Synaptic Rewiring in Neuromorphic VLSI for Topographic Map Formation

Simeon A. Bamford

Doctor of Philosophy
Institute for Adaptive and Neural Computation
School of Informatics
University of Edinburgh

2009

Abstract

A generalised model of biological topographic map development is presented which combines both weight plasticity and the formation and elimination of synapses (synaptic rewiring) as well as both activity-dependent and -independent processes. The question of whether an activity-dependent process can refine a mapping created by an activity-independent process is investigated using a statistical approach to analysing mapping quality. The model is then implemented in custom mixed-signal VLSI. Novel aspects of this implementation include: (1) a distributed and locally reprogrammable address-event receiver, with which large axonal fan-out does not reduce channel capacity; (2) an analogue current-mode circuit for Euclidean distance calculation which is suitable for operation across multiple chips; (3) slow probabilistic synaptic rewiring driven by (pseudo-)random noise; (4) the application of a very-low-current design technique to improving the stability of weights stored on capacitors; (5) exploiting transistor non-ideality to implement partially weight-dependent spike-timing-dependent plasticity; (6) the use of the non-linear capacitance of MOSCAP devices to compensate for other non-linearities. The performance of the chip is characterised and it is shown that the fabricated chips are capable of implementing the model, resulting in biologically relevant behaviours such as activity-dependent reduction of the spatial variance of receptive fields. Complementing a fast synaptic weight change mechanism with a slow synapse rewiring mechanism is suggested as a method of increasing the stability of learned patterns.

Acknowledgements

I'd like to thank my supervisors Prof Alan Murray and Prof David Willshaw for their guidance and support and for giving me a great deal of freedom to experiment. Their depth and breadth of knowledge is inspirational, and they are gentlemen whose reputation precedes them and whom it is a pleasure to be associated with.

I'd like to say that the chip I created is a success for the whole neuromorphic engineering community. It contains a bias generator circuit designed and implemented by Andre Van Schaik and Tobi Delbruck; a soma circuit designed by Giacomo Indiveri; AER sender circuitry designed by Kwabena Boahen, laid out by Elisabetta Chicca and reworked for cadence by Vasin Boonsobhak; aspects of an STDP circuit by Adria Bofill-i-Petit; and it implements a grid communications scheme demonstrated by Paul Merolla. It was extremely useful to attend the 2007 Telluride Neuromorphic Engineering Workshop; many useful discussions were had there.

I benefitted greatly from the practical advice of Katherine Cameron, with additional help in design and layout from Robert Henderson, Martin Reekie, Bruce Rae, Keith Muir and at least one revelatory insight from Thomas Koickal. Susan Kivlin helped me with her surface-mount soldering skills, Keith Muir helped me to learn PCB design and Thomas Clayton helped me to learn VHDL coding for FPGAs. Guy Billings saved me a lot of time by providing me with the basis for the code used for neural network simulations and Adrian Haith helped me with maths on several occasions.

There have also been helpful discussions with or other practical help from Anthony Walton, Zhijun Yang, Tong-Boon Tang, Luiz Gouveia, Adria Bofill-i-Petit, Vasin Boonsobhak, Juan Huo, Yaxiong Zang, Mark Muir, Iain Lindsay, Tobi Delbruck, Giacomo Indiveri, Elisabetta Chicca, Shih-Chii Liu, Paul Hasler, Anand Chandrasekaran, Paul Merolla, Srinjoy Mitra, Matthias Hennig, Jim Bednar, Judith Law, Chris Palmer, David Sterratt, Mark van Rossum, Adam Barrett, Chris Williams, Amos Storkey, John Quinn, Lawrence York, Tom Griffiths, Sen Song, and Martin Ling.

I'd like to thank the management and administration of the Edinburgh University Doctoral Training Centre in Neuroinformatics and Computational Neuroscience and all those responsible for its creation, for engendering a fantastic environment for learning. The centre, and my research, has been funded by EPSRC and MRC, to whom I am very grateful.

Finally I'd like to thank my partner, Bea Brogi for her love and devotion.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Simeon A. Bamford*)

Contents

Chapter 1

Introduction

1.1 Rationale

The aim of this project was to investigate how to implement synaptic rewiring (the formation and elimination of synapses) in neuromorphic VLSI. In particular it may be desirable to allow the weights of synaptic connections (which can change rapidly and have frequently been stored in volatile memory on capacitors) to influence network topology (which changes slowly and can be stored in stable memory elements), in order to overcome an existing problem of how to stably store learnt patterns in neuromorphic systems. The phenomenon of biological topographic map formation was chosen as a case-in-point to constrain designs. It will be seen that there are existing neuromorphic systems which achieve synaptic rewiring through the manipulation of off-chip look-up tables. An alternative was however conceived whereby synapse circuits could advantageously store details of their incoming connectivity locally and change this based on the outcome of a locally implemented learning rule.

1.2 Statement of hypothesis

Formally, given the above rationale, this is an investigation of the hypothesis that (1) synaptic rewiring can be implemented in neuromorphic VLSI by circuitry distributed throughout the synapses of a neural array, that (2) this ability can be used to model the

phenomenon of topographic map formation, and that (3) this ability can increase the stability of patterns learnt by a neuromorphic system.

This hypothesis has guided the following investigation, embodying as it does the core of the electronic system created, a link to the biological phenomena on which it is based, and a suggestion of the practical benefits that it may generate.

1.3 Thesis structure

In brief, the structure of this thesis is as follows. Firstly, a model of topographic map formation is developed. There follows the presentation of circuitry relating to: the functioning of neurons and synapses; the construction of neural networks by systems for the delivery of spikes; and the rewiring of these networks according to topographic constraints. Finally a functioning circuit implementation of the model is demonstrated.

Due to the diverse subject areas which this thesis covers, each chapter begins with a literature review relevant to the contents of the chapter. The detailed chapter-by-chapter structure is as follows.

Chapter 2: Modelling topographic map formation

The literature on biological topographic map formation is reviewed, as is that on synaptic plasticity, including synapse formation and elimination. While synapse formation and elimination is undoubtedly an important part of topographic map formation, most computational models fail to model this explicitly but rather assume its equivalence with weight plasticity.

A generalised model of topographic map development is presented, which is based on spiking (integrate-and-fire) neurons, and combines both weight plasticity and the formation and elimination of synapses (synaptic rewiring) as well as both activity-dependent and -independent processes.

A novel approach to analysing mapping quality is developed, which allows independent consideration of the development of a projection's preferred locations and the spread of receptive fields. This is used to address statistically the question of whether an activity-

dependent process can refine a mapping created by an activity-independent process and to assess the effect of the interplay between two forms of plasticity.

Computational simulations were carried out and it is shown that (within the model): (a) Synapse formation and elimination embed in the network topology changes in the weight distributions of synapses due to the activity-dependent synaptic modulation rule used (Spike-Timing-Dependent Plasticity - STDP); (b) the variance of receptive fields can be reduced by an activity dependent mechanism, with or without spatially correlated inputs; (c) the accuracy of preferred locations will not necessarily improve when synapses are formed based on distributions with on-average perfect topography.

The remainder of the thesis is about the implementation of the above model in neuromorphic VLSI.

Chapter 3: Silicon neuron and synapse

The literature on the field of Neuromorphic VLSI is reviewed, focusing on neuron and synapse implementations, including implementations of spike-timing-dependent plasticity (STDP).

The design choices which have been made stem from the following principles: information should be kept locally to minimise its transmission where possible; limited quantisation of time is acceptable; and the scaling of numbers of neurons and synapses should be borne in mind when considering area constraints.

Neuron and synapse circuits are presented. Novelty includes: (a) A switched capacitor implementation of excitatory and leakage currents onto the membrane; (b) very-low-current design technique applied to the design of a STDP circuit, resulting in weights which retain traces of their learnt values over tens of seconds; (c) the use of MOSCAP non-linearities to offset other non-linearities, resulting in broadly linear behaviours over wide voltage ranges; and (d) a simple approach to introducing a controlled amount of weight dependence into STDP.

The performance of these circuits is characterised, based on chip results; their ability to perform equivalently to the neurons and synapses used in the computational model in chapter 2 is demonstrated; The effects of process variation are considered and in particular it is shown that the homeostatic nature of STDP helps to reduce this problem.

Chapter 4: A distributed and locally reprogrammable address-event receiver

Techniques for transmitting spikes in neuromorphic systems are reviewed, focusing on address-event representation.

A design is presented for a run-time-reprogrammable address-event decoder, where the decoding elements are distributed through the synaptic array and act simultaneously. This allows a spike to be received simultaneously by all the synapses on the axonal arbour. This decoder is compatible with existing address-event senders. The scalability of this system is compared with existing systems, with respect to silicon area, energy and speed (thus, channel capacity), as numbers of neurons and synapses in a system increase. It is shown that this system scales particularly well in terms of speed as synaptic fan-out increases. This advantage is demonstrated with chip results which show firstly the simultaneous receipt of spikes by many synapses, then the simultaneous functioning of 8 chips, configured in a grid arrangement [Merolla et al., 2007], achieving spike delivery rates which are arguably higher than any other system published to date.

Alternative designs are discussed, namely: a word-serial distributed decoder; floating gate memory for semi-analogue addressing; and an integrated look-up-table.

Chapter 5: Synaptic rewiring and Euclidean distance calculation

Literature is reviewed which relates to neuromorphic approaches topographic mapping and synaptic rewiring. There is then a presentation of the reasoning behind the approach of storing connectivity information locally within neurons. This approach is then used to argue for the location of circuitry for implementing synaptic rewiring within each synapse. Such circuitry is then presented and its functioning demonstrated. In the model of synaptic rewiring adopted, the connection rule requires a calculation of the distance between a neuron and the ideal location of a potential pre-synaptic partner; consequently, circuitry for Euclidean distance calculation is presented, whose novel features are current-mode operation across multiple chips and the capability of implementing both wrap-around (toroidal) and non-wrap-around topologies. The ability of the rewiring circuitry together with the distance calculation circuitry to allow the formation of radially symmetric receptive fields with arbitrary relationships of connection probability to distance from the centre is then demonstrated.

Chapter 6: The model implemented

As the capabilities of the fabricated chip have been demonstrated in chapters 3, 4 and 5, in this chapter, chip results are presented which qualitatively match the simulation results presented in chapter 2, thus demonstrating its fitness for the intended purpose. Similarities and differences between the model and its implementation are discussed.

Finally the project is summarised, conclusions are drawn and future work is proposed.

Chapter 2

Modelling Topographic Map Formation

2.1 Introduction

The hypothesis presented in section 1.2 includes the idea that circuitry which implements synaptic rewiring may be used to model the phenomenon of topographic map formation. This chapter lays the groundwork, by presenting a novel model of topographic map formation which includes the process of synaptic rewiring. Firstly, the subjects of synaptic plasticity and topographic map formation are reviewed. The question of how to assess the quality of topographic mappings is addressed with the novel statistical methods of analysing receptive field development. Simulation results from the model are presented.

2.1.1 Synaptic plasticity

The connections between neurons, known as synapses, are crucial elements in neural systems, both for communication and transfer of information and for their plasticity, which provides a basis for the memory and adaptability of organisms. The term "synaptic plasticity" encompasses the formation and elimination of synapses and changes in their physiological strength. These processes will be discussed in the following two sections.

2.1.1.1 Synapse formation and elimination

In order to construct the neural networks of the nervous system, neurons form synapses. In the vertebrate brain, chemical synapses are predominant (cf. gap junctions), and typically

form as follows. A neuron grows processes; these are typically specialised into dendrites, which convey incoming signals and are typically branched over a short range, and axons, which convey outgoing signals and may travel for a relatively long distance of millimetres or more before branching (the branching of the axon results in an "axonal arbor"). When an axon branch comes into close proximity with a dendrite, (which may be aided by the formation of an outgrowth on the dendrite known as a spine), a synapse may form. This is a specialised region in which neurotransmitter chemicals can be released from the axon. This release is typically in response to a wave of depolarisation known as a spike, which travels along the axon from the body of the neuron. The neurotransmitter can then cause depolarisation or hyperpolarisation in the dendrite (these are, respectively, the effects of excitatory and inhibitory synapses), which propagate towards the body of the neuron and may in turn cause further spikes to be transmitted (or alternatively may inhibit further spikes). Neurons in one area of the brain often develop axons which grow *en masse* to a different area to innervate the neurons there. These axons are guided by concentrations of marker chemicals [Dickson, 2002]. The growing end of an axon continuously grows and retracts finger-like protrusions called filopodia which carry sensors for various chemicals and, in a manner which is not completely understood, allow the axon to be guided. The area-by-area structure of the brain and the guidance of axons to form connections between different brain areas are prerequisites for the development of topographic maps, as will be seen later in section 2.1.2.

Synaptic connections can also be eliminated. This process is well studied, for example at the neuromuscular junction. In neonatal mammals, each muscle fibre is innervated by axons from several different motor neurons and then during development most of these synapses are eliminated so that in the adult, each muscle fibre is innervated by only one motor neuron. The process is known to be competitive and various mechanisms have been proposed to account for this [Buffelli et al., 2004], including Hebbian mechanisms, see below in section 2.1.1.2.

When a synapse is eliminated, the axon branch that leads to it is retracted [Bishop et al., 2004]. Unoccupied dendritic spines also retract [Trachtenberg et al., 2002]. The formation and elimination of synapses (as well as the remodelling of axons and dendrites that underlies it) is collectively referred as synaptic rewiring [Chklovskii et al., 2004].

2.1.1.2 Hebbian learning

The physiological strength of a synapse refers to the efficacy with which the arrival of a spike at a synapse affects the membrane potential in the post-synaptic neuron. The terms "synaptic strength" and "weight" are used interchangeably. In the model presented in section 2.1.1.3, based on an established neurophysiological model [Song and Abbott, 2001], the peak synaptic conductance corresponds to synaptic weight; when a spike arrives at the synapse this is assumed to achieve an instantaneous rise in the conductance of the post-synaptic neural membrane towards an excitatory reversal potential, which then decays. Synapse strength can also be modelled more concisely as a combination of factors such as the probability of vesicle release, post-synaptic receptor density, etc [Vogelstein et al., 2007].

Hebb [1949] proposed that where one neuron consistently took part in causing another one to fire then the connection between them would become strengthened. From this postulate, the term "Hebbian learning" has come to encompass a class of systems in which changes in strengths of the synapses between neurons are related to the correlation of their activity. Long term potentiation refers to the increase in the physiological strength of a synapse, which is known to be triggered by changes in intracellular Ca^{2+} concentration and can occur following stimulation (typically at high frequency in experimental protocols) of both pre- and post-synaptic neurons [Malenka and Nicoll, 1999]; long term depression refers to a complementary reduction in physiological strength (which can be induced by lower frequency stimulation).

In models of synaptic plasticity in which coincidence of pre- and post-synaptic activity causes potentiation, additional constraints are typically applied to prevent the run-away potentiation of synapses, such as global normalisation or decay of synaptic strength [Miller et al., 1989]. Such constraints could be seen as simplifications of homeostatic processes, which serve to keep the overall activity of neurons in balance [Turrigiano, 2007].

Although in this thesis, a Hebbian mechanism is used to modulate synapses in order to develop topographic mappings (see below in section 2.1.4), the same mechanism can serve as a basis for learning and memory; thus although map development is usually addressed in separate literature to memory, in this thesis, issues of learning and memory storage are considered, and the terms "synaptic learning" and "synaptic modulation" are used interchangeably (where appropriate).

2.1.1.3 Spike-timing-dependent plasticity

Hebb's postulate implies causality. For a pre-synaptic spike to cause a post-synaptic neuron to fire it is necessary that the pre-synaptic spike precede the post-synaptic spike. Bi and Poo [1998] observed that in cultured hippocampal neurons, the potentiation or depression of a synapse was dependent on the temporal order of induced pre- and post-synaptic activity. In this study [and in Markram et al. 1997, Zhang et al. 1998], pre-synaptic activity preceding post-synaptic activity cause potentiation (and *vice-versa*) in accordance with the causality condition, though in other studies the opposite temporal dependence has been observed [Bell et al., 1997]. Such Spike-Timing-Dependent Plasticity (STDP), as it has become known, was predicted prior to these observations in computational and VLSI models [Gerstner et al., 1996, Hafliger et al., 1997] (the historical antecedents of STDP are traced in more detail by Morrison et al. [2008] p. 481), and has since been investigated extensively in computational neuroscience.

Song et al. [2000] modelled STDP in a way which has been used in many subsequent studies. The model is summarised here, as it forms a basis for work in this project. STDP was implemented such that a pre-synaptic spike at time t_{pre} and a post-synaptic spike at time t_{post} modify the corresponding peak synaptic conductance by $g \rightarrow g + g_{max}F(\Delta t)$, where $\Delta t = t_{pre} - t_{post}$ and:

$$F(\Delta t) = \left\{ \begin{array}{ll} A_+.e^{(\frac{\Delta t}{\tau_+})}, & if \Delta t < 0 \\ -A_-.e^{(\frac{-\Delta t}{\tau_-})}, & if \Delta t \geq 0 \end{array} \right\} \tag{2.1}$$

where $A_{+/-}$ are magnitudes relative to the range of conductance and $\tau_{+/-}$ are time constants for potentiation and depression respectively. This is cumulative for all pre- and post-synaptic spike pairs. g is bounded in the range $0 \leq g \leq g_{max}$. The peak synaptic conductance g is then used to instantaneously increase the excitatory conductance of the neuronal membrane on the arrival of a spike and this conductance decays exponentially thereafter.

This model was used to show that in a neuron whose dendritic synapses implemented STDP, the synaptic weights would diverge into a strong group and weak group, with the effect that (a) output spike rate was held within a narrow range relative to the range of mean input frequencies applied, and (b) groups of synapses whose input spikes were more correlated, i.e. more likely to arrive within a narrow time window of each other, would be preferentially strengthened over synapses whose input spikes were less correlated. (a)

is interesting as it is a form of homeostatic regulation and (b) is interesting as it allows unsupervised learning based on input correlations. The divergence of incoming synaptic weights for a neuron into a bimodal population, however, is an effect of the choice of models for STDP and the other results mentioned do not necessarily depend on it. In the update rule for synaptic conductance shown above, the change in conductance is not dependent on the initial conductance (except to the extent that the conductance is bounded); that is to say, it is an additive update rule.

Other authors investigated multiplicative update rules, which assume linear attenuation of depression as a lower boundary is approached [Van Rossum et al., 2000], or this in addition to linear attenuation of potentiation as an upper boundary is approached [Kistler and Van Hemmen, 2000, Rubin et al., 2001]. With these rules, a unimodal distribution of synaptic weights results, but competition fails to achieve robust segregation of groups of synapses. Following this, Gutig et al. [2003] developed a generalised STDP rule which allowed for a tunable degree of weight dependence. In the model, a single pair of pre-synaptic and post-synaptic action potentials with time difference $\Delta t \equiv t_{post} - t_{pre}$ induces a change in synaptic efficacy Δw given by:

$$\Delta w = \left\{ \begin{array}{ll} -\lambda f_-(w).e^{(\frac{\Delta t}{\tau})}, & if \Delta t < 0 \\ \lambda f_+(w).e^{(\frac{-\Delta t}{\tau})}, & if \Delta t \geq 0 \end{array} \right\}$$

where λ is a learning rate, w is the initial synaptic efficacy, τ is a single time constant for both depression and potentiation, and $f_+(w)$ and $f_-(w)$ are updating functions which in general are weight dependent. They introduced a family of updating functions:

$$\begin{aligned} f_+(w) &= (1-w)^\mu \\ f_-(w) &= \alpha w^\mu \end{aligned}$$

Thus by changing the parameter μ, the model can capture a range of update rules between completely additive and completely multiplicative — see figure 2.1, (α allows for different learning rates for potentiation and depression).

They then investigated various parameters for this model to determine under what conditions a bimodal distribution would result and to what extent such symmetry breaking

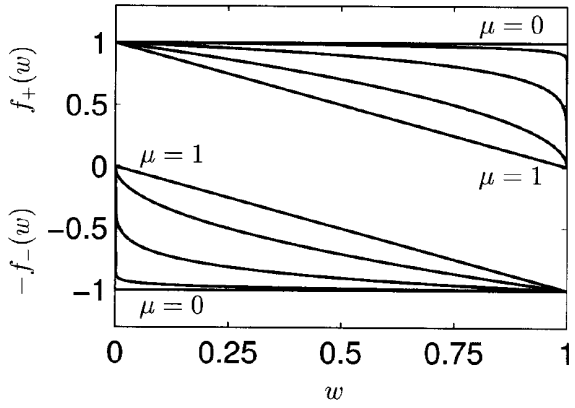


Figure 2.1: Synaptic updating functions with varying weight dependence. Figure taken from Gutig et al. [2003] p. 3698: "Effect of the parameter μ on the updating functions $f_+(w)$ (top half) and $f_-(w)$ (bottom half) for $\mu =$ 0, 0.02, 0.15, 0.5, 1 ($\alpha = 1$). As μ increases, the curves change from the constant additive updating curves ($\mu = 0$, horizontal lines at 1 and -1) to the multiplicative updating functions with linear weight dependence ($\mu = 1$, straight lines with slope -1)."

would capture the correlational structure in the incoming activity. The last point is important since Song et al. [2000] also showed that two groups of inputs which were intra-correlated but not inter-correlated would compete to control the output of the neuron, i.e. one group of synapses would end up potentiated and the other depressed, and the outcome would be random. Thus there is a question of how strong correlational cues have to be in order to affect the resulting synaptic weight distributions. Gutig et al. [2003] therefore investigated the parameters μ, and the amount of correlation within groups of correlated inputs. They found that there are constrained regions of the parameter space in which the correlational structure of the inputs can be captured by resulting synaptic weight distributions, and that the sensitivity of the outcome to differences in the amount of correlation between competing groups was maximised with $0 < \mu < 1$. Thus a degree of weight dependence can improve the ability of STDP to act as a correlation detection mechanism. Whilst a bimodal distribution can always form in the presence of a sub-group of correlated inputs and another group of uncorrelated inputs, for larger values of μ the equilibrium weights of the two groups will not diverge to extremes but may be close together, and the separation will not persist if the correlations are removed. Available evidence currently suggests that a unimodal distribution is more realistic than a bimodal one [Song et al., 2005]. Interestingly, Morrison et al. [2007] have more recently shown that the original results from Bi

and Poo [1998] are best described by a learning rule which cannot be captured by the formalism of Gutig et al. [2003] since increasing weight appears to have a positive influence on the magnitude of a potentiating update.

There is ongoing debate about the nature of STDP, the molecular mechanisms that give rise to it and its relevance as a candidate mechanism for memory and learning. To give some example of the range of questions that exist: STDP-like behaviour can arise from a synaptic update rule dependent on post-synaptic membrane voltage rather than post-synaptic spikes [Brader et al., 2007]; there are experiments which indicate that individual synapses may have binary strengths and experience all-or-nothing plasticity events [Petersen et al., 1998], which are apparently at odds with studies showing synapses have unimodal distributions; and there are questions over how the contributions of different spike pairs should be combined [Sjostrom et al., 2001, Butts et al., 2007]. Experiments demonstrating the nature of STDP have typically used *in vitro* preparations [Bi and Poo, 1998] or unrealistic levels of stimulation [Zhang et al., 1998] leading to questions about their relevance to normal cellular processes.

Notwithstanding the above, learning rules similar to the formalism of Song et al. [2000] have been used to investigate: topographic map formation [Song and Abbott, 2001]; the response to latency in inputs [Guyonneau et al., 2005]; visual feature map learning [Masquelier and Thorpe, 2007]; receptive field reorganisation [Young et al., 2007]; learning cross-modal spatial transformations [Davison and Fregnac, 2006] etc. The study of Jun and Jin [2007] is particularly notable as a study of the formation of synfire chains using a combination of STDP and a form of rewiring plasticity, similar to the model presented in this thesis.

In the computational model described in section 2.2, weight independent STDP is adopted, in order to reduce the parameter space of the model and in keeping with the aforementioned body of work, whereas for the neuromorphic VLSI implementation described in section 3.4.4, partially weight-dependent update rules are investigated due to synergistic properties of the silicon substrate. For the intended application, these rules should be expected to achieve the same effects though possibly with varying degrees of efficiency.

2.1.1.4 The link between synaptic weight change and rewiring

Both forms of plasticity discussed above, namely synapse rewiring and changes in synapse weight, have the potential to change the performance of a neural network and therefore act as a basis for development (as well as for learning and memory). It is therefore natural to question whether there is a relationship between them. At the neuromuscular junction a reduction in synaptic efficacy precedes synapse withdrawal (as judged by quantal release probability and post-synaptic receptor density,[Colman et al., 1997, Balice-Gordon and Lichtman, 1993]). One interpretation of this is that the weakness of a synapse is a causal factor in its elimination. Experiments in which kittens are deprived of visual input in one eye are also informative (as pointed out by Miller [1998]); responses of cells in the primary visual cortex ("V1") previously tuned to the deprived eye become less responsive to input from that eye (and sometimes more responsive to input from the open eye) over a period of hours [Mioche and Singer, 1989], with the change in response complete within two days [Hensch et al., 1995], whereas axonal remodelling associated with this change is incomplete after 4 days [Antonini and Stryker, 1996]. This is an illustration of the different timescales over which the different forms of plasticity operate: in general, weight plasticity operates over a period of minutes to hours whereas structural changes operate over a scale of days to weeks or longer [Chklovskii et al., 2004]. Synaptic weight potentiation may also be a causal factor in the formation of new synapses, as demonstrated in hippocampal slice cultures where synapse formation follows LTP induction [Toni et al., 1999].

2.1.2 Topographic maps in the brain

A topographic map is an area of the brain where the response to input parameters varies continuously across the area. Where a sheet of neurons in one area (the "source" area) innervates a sheet of neurons in another ("target") area, the mapping between the areas can be said to be topographic if neighbouring neurons in the target area are (maximally) responsive to the activity of neighbouring neurons in the source area [Udin and Fawcett, 1988]. Alternatively, if a sheet of neurons is responsive either more or less directly to a sensory modality such as vision, the sheet of neurons could be said to have a topographic map of the sensory input if, to take the example of vision, cells in neighbouring parts of the sheet of neurons are receptive to stimulation from neighbouring parts of the visual field.

There are numerous topographic mappings present in most vertebrate brains. Indeed, the development of topographic maps across the cortex is a process that goes hand in hand with the division of the cortex into functional areas [Sur and Rubenstein, 2005]. Most sensory modalities project topographically; for example, the retina is responsive across its two-dimensional surface to the two dimensions of the visual field and the retina then projects topographically (or "retinotopically") to the superior colliculus in mammals or the corresponding optic tectum in lower vertebrates. It also projects retinotopically to the various layers of the Lateral Geniculate Nucleus ("LGN"). The projections from the LGN to V1, and to higher cortical areas involved with vision continue, to maintain retinotopic organisation. Audition is also topographically organised, with the range of frequency sensitivities of the cells of the cochlear projected smoothly across the cochlear nucleus. Note that the term "projection" is used here to indicate a set of directional connections from one brain area or layer of cells to another.

2.1.2.1 Receptive fields

Most projections between brain areas exhibit many-to-many connectivity, that is to say, an axon from one source neuron makes connections with many target neurons, and likewise the dendritic tree of one target neuron forms synapses with many source neurons. Thus, a target neuron is unlikely to exhibit sensitivity to excitation from only one source neuron; rather it is said to have a "receptive field", which is a region of the source area over which the neuron is responsive to stimulation. In an alternative nomenclature, the term "receptive field" refers to the area in the visual field (or other sensory space) in which stimulation causes activity in the target neuron whilst a patch of a neural sheet from which a target neuron receives connection is referred to as an incoming "connection field" [Nakahara et al., 2006]. These terms are used interchangeably here. The region of a target neural sheet to which the axonal arbor of a source neuron projects may be called a "projective field" [Venier et al., 1997].

Gaze et al. [1974] demonstrated two ways in which receptive fields can be assessed; a target location was recorded from (albeit mainly using multi-unit rather than single cell recordings); then the boundaries of an area of the input space were determined in which stimulation repeatably caused a discernible increase in spiking activity in the target; alternatively a point in the input space was determined for which target response was maximal.

Projections from both eyes, originally growing along separate optic nerves, are mixed

together in the optic chiasm, so that the topographic maps present in the cortex receive input from both eyes, (in those areas which map parts of the visual field which are visible from both eyes). Although these projections are intermixed, tracing the connections from the eyes to V1 (via LGN) reveals that in many mammals there are alternating stripes in V1 in which cells predominantly receive input from one eye or the other [Hubel et al., 1977]; these are known as patterns of *ocular dominance*. Thus the receptive fields of V1 neurons can be seen as covering areas of both retinae, though with a preference for one of them. This preference may be mediated by differing numbers or weights of synapses from the respective areas.

In fact, many different parameters of the visual input have representations which vary across V1, such as: orientation – the tendency of a neuron to respond to edges between regions of differing illumination which are oriented in a particular direction [Hubel and Wiesel, 1962, Erwin et al., 1995]; spatial frequency – a neuron may be maximally responsive to stimuli whose illumination varies with a given frequency across a portion of the visual field (judged perpendicularly to any preferred orientation) [De Valois et al., 1982]; and colour sensitivity [Tootell et al., 1988].

2.1.3 The purpose of topographic organisation

Reasons why topographic organisation may exist in the brain or purposes it may serve are presented here, under the headings of dimension reduction, wiring efficiency and multimodal integration.

2.1.3.1 Dimension reduction

One way to think about orientation stripe patterns in V1 (introduced in section 2.1.2.1) is that three dimensions of information from the visual field, namely azimuth, elevation and ocularity, are compressed into the two (predominant) dimensions of the cortical area V1. There is a caveat that area V1 also has 3 dimensions; however, ocularity is strongest in one layer (layer 4) of V1, which has very little depth, and to the extent that ocularity exists in the cells of other layers it corresponds to the preferences of layer 4 cells in the corresponding location, hence the term *ocular dominance column* (generally, cells within confined columns through the cortex are responsive to the same stimuli). Whilst retinotopy is maintained across the entire area of V1, at a closer scale the representation of

ocularity across the area oscillates from one eye to the other. In fact, the many different parameters of the visual input introduced above such as orientation, colour, etc, also have representations across V1 which vary continuously (though sometimes with abrupt discontinuities) at fine scale. The sensitivity to orientation, for example, has been shown to be finely ordered, down to the scale of neighbouring neurons [Ohki et al., 2005, 2006]. These variations in input preference at fine scale can be seen as the effect of attempting to reduce many dimensions of input stimuli into two physical dimensions.

2.1.3.2 Wiring efficiency

Knudsen et al. [1987] sought to define computational maps as maps (whether topographic or not) which apply some transformation to the patterns of activity which pass through them rather than simply relaying information. By this definition, orientation maps as defined above are computational maps, since orientation-selective neurons sample an area of the visual space with such a receptive field as to be primarily sensitive to edges oriented in a particular direction; thus there is a mapping from neurons in LGN primarily sensitive to centre-surround contrast to those in V1 sensitive to orientation. In order to perform such transformations, connections are needed with neurons representing local and neighbouring locations in visual space rather than far apart locations. Thus, if neurons which represent neighbouring parts of the input space (or more generally, neighbouring values for input parameters) are close together then the total amount of wiring required can be shorter and thus the wiring of the brain can be more efficient. There would be evolutionary pressure towards such a solution, due to the costs of wiring to an organism in terms of energy, volume and perhaps speed of transmission.

Chklovskii and Koulakov [2004] argued that since neural networks with the same connection topology will have equivalent functionality regardless of the physical location of the neurons, and differ only in the wiring cost, it is hard to justify the existence of topographic maps without reference to wiring cost.

2.1.3.3 Multimodal integration

The superior colliculus (“SC”) is the mammalian equivalent of the optic tectum in lower vertebrates. The optic tectum is a key part of the visual system. In mammals the visual pathways in the cortex provide most of the functionality of visual perception. However,

the SC retains a role in gaze control. A stimulus in the visual field can invoke eye saccades or head or torso movements to orientate the gaze in the direction of the stimulus. The topographic map of the visual field also acts as a map of potential gaze orientations, as activity in a circumscribed patch of the SC precedes eye saccades towards the corresponding target location [King, 2004]. Gaze changes can also be invoked by auditory or somatosensory stimuli. Whilst neurons in the superficial layer of the SC are responsive to input from retinal ganglion cells, neurons in successively deeper layers receive input from the inferior colliculus (part of the auditory pathway), and the somatosensory cortex respectively. (In addition they receive input from many cortical and thalamic areas [Harting et al., 1992]). The auditory and somatosensory projections are ordered so as to represent the body and the space around it in the same coordinate frame as the visual projection and are overlaid so that neurons which are vertically aligned are responsive to the same part of space, albeit via different modalities. This would give an organism a basis for multimodal sensory integration. Experiments with cats show that the effect of visual and auditory stimuli from the same direction sum non-linearly both in terms of the activity they cause in the SC and in the startle response invoked [Holmes and Spence, 2005].

2.1.4 The development of topographic maps

The development of topographic maps between two brain areas requires that axons grow from the source area to the correct target area and then form synapses with neurons in the correct location. This is a highly complex process, which involves axon guidance to the correct area, finding the correct layer and topographic point and creation of synapses on appropriate parts of the target cell(s) [Benson et al., 2001]. This thesis excludes consideration of axon guidance to the correct brain area and additionally does not consider the questions of how to target the correct layer (in projections to layered tissue) or how to terminate on the correct part of a dendritic tree. Rather it is primarily concerned with the development of receptive fields. Although the model which is to be the main subject of this thesis assumes a mechanism for finding correct topographic position, models which hypothesise mechanisms for the establishment of correct topographic locations are reviewed here, in order to place the model in context.

A much-studied example of a topographic mapping is that between the retina and the optic tectum, or the SC in mammals, and in the following discussion this mapping will be used as an example.

In Xenopus (the African Clawed Frog), the mapping between each retina and the contralateral tectum to which it projects is such that the dorsal-ventral axis (from top to bottom — literally from back to belly) of the retina is projected across the medial-lateral axis (from the centre out sideways) of the tectum whilst the nasal-temporal axis (from the nose to the temples) of the retina is projected across the rostral-caudal axis (from head to tail) of the tectum [Gaze et al., 1974]; this is also true in animals as diverse as the Golden Hamster [Finlay et al., 1978] etc. In Xenopus, the projection forms while the areas themselves grow, the retina growing by marginal accretion of cells [Straznicky and Gaze, 1971] whilst the tectum grows in a curvilinear fashion starting from the most rostral-lateral point with addition of cells in caudal and medial directions [Straznicky and Gaze, 1972]. Retinal Ganglion Cell (RGC) axons from the retina enter the contralateral tectum and are immediately directed towards an appropriate topographic location, branching as they approach it and forming synapses with the tectal cells' dendrites. In chicks and rodents by contrast, the axons grow parallel to the dorsal SC, overshooting the appropriate location and then axon branches form, which accumulate in an appropriate location whilst other branches retract [McLaughlin et al., 2003b]; thus the spatial spread of the axonal arbors reduce, and this is often referred to as topographic refinement. Because, in Xenopus, the areas grow while the mapping is forming, appropriate retinotopic locations for RGCs change during development with respect to the actual cells in the tectum; at each point in development from as early as can be measured (developmental stage 47) such order as can be discerned is topographic, consistent with the adult ordering [Gaze et al., 1974]. From this it can be deduced that the pattern of efferent (outward) connectivity for individual RGCs and afferent (inward) connectivity for tectal cells changes during development; initially RGCs from the centre of the eye project to a central location in the existing tectum, but this location is rostral-lateral with respect to the full grown tectum. Later the same RGCs project to the centre of the tectum, to cells which were not formed initially, whilst the tectal cells in the rostral lateral tectum have receptive fields in the nasal ventral retina. This requires that new axon branches and synapses are formed whilst old synapses and axon branches are retracted. (There is a caveat based on this evidence that old synapses may simply become weaker rather than being eliminated, since Gaze et al. [1974] believed, but could not prove, that they were recording from terminal arborisations of optic nerve fibres when they assessed the activity resultant from activation of retina; McLaughlin's visualisations of axon retraction at later stages albeit in a different organism are suggestive of axon retraction rather than simply of the weakening of remaining synapses).

More facts about map development are pertinent and will be introduced during the following discussion on models.

2.1.5 Models of topographic map formation

Models of topographic map development are introduced here which seek to explain the development of topographic organisation and receptive fields, under the headings: activity dependence; weight vs wiring plasticity; lateral interactions in the target layer; and growth of areas. During this discussion, the tenets of the model which is presented in this thesis will be revealed; each section concludes by stating the relevance of the discussion to the model which will be presented.

2.1.5.1 Activity dependence

Models of topographic map formation can be divided into those which require activity of the participant cells (that is to say, electrical or spiking activity) in order to form the map and those which do not. The following sections discuss these separately and then how they can be combined.

Activity-independent models Sperry [1943] proposed that in order to establish topographic maps, each pair of cells which are to be connected should have unique chemical markers to signal that they should connect to each other. However, the sheer number of unique signatures that would be required to be genetically programmed makes this unlikely. He then proposed [Sperry, 1963] that the target area be labelled by two orthogonal gradients of chemicals which in-growing axons could use to be guided to the correct location (this is known as the chemoaffinity hypothesis). Experimental evidence in favour of this hypothesis has followed recently, with the discovery of various candidate marker chemicals, the most well-studied of which is the ephrin family of membrane-bound molecules and their associated receptors. To take the example of the mapping from RGCs to the optic tectum, EphA receptors are found in RGCs, distributed in an increasing gradient across the nasal-temporal axis of the retina [Flanagan and Vanderhaeghen, 1998], whilst their associated membrane-bound ephrin-A ligands are expressed in a decreasing gradient along the rostral-caudal axis of the tectum. There is a complementary arrangement in the other dimension, where EphB receptors are distributed in an increasing

gradient in the dorsal-ventral direction across the retina, whilst their associated ephrin-B ligands are distributed in an increasing gradient along the medial-lateral axis of the tectum [Mann et al., 2002].

Allowing then that there exist molecular gradients which may act as cues for developing axons, Prestige and Willshaw [1975] distinguished between two ways the interactions may work. In Type I matching, source neurons have an affinity for a specific patch of the target area, whereas in Type II matching, all source neurons have maximum affinity for one end of the target area. Models which hypothesise type I mechanisms, in which all axons independently find their own positions using interactions between receptor and ligand, can include those which require counter gradients [Gierer, 1983] and those which require set points [Honda, 1998]. An example of a set point mechanism is as follows: it is known that axons enter the tectum from the rostral end and grow caudally; if each axon were to continue growing until interactions between receptors and ligands reach a critical level, then those axons with fewer receptors will reach a zone of greater ligand density before the required interaction strength is attained; if the effect of the presence of the ligand on the growing axon is repulsion then such a mechanism is achieved, and indeed this is the normal mode of interaction for Ephs and ephrins [Orioli and Klein, 1997]. Type II models by contrast require competitive interaction between axons, all having maximum affinities for the same limited space, with the winners being determined by the strengths of interactions between receptors and ligands [Koulakov and Tsigankov, 2004]. Neither of these types of mechanisms can account for a range of results which have been attained by performing specific surgical manipulations to the retinotectal system. These manipulations are summarised in [Goodhill and Xu, 2005]. Briefly, there have been experiments in which either the tectum or the retina have been rotated, half of the retina or tectum or non-complementary halves of both have been removed, in which a portion of tectum has been rotated, and in which two halves of retinae have been swapped. In each case any developed projections are cut and then allowed to regenerate following surgery, and the resulting mapping is analysed, to assess the interactions between presumed marker gradients in the abnormal mapping. For example, when the nasal half of the retina is removed, connections from the remaining half reform in a smooth gradient across the entire rostral-caudal axis of the tectum. This is inconsistent with a type II rule, since the available space in the tectum has not decreased thus all connections from the temporal retina should still be able to occupy the same space in the tectum as before the lesion. If marker gradients are assumed to be constant in the remaining tissue then it is also inconsistent with a type I

rule, since the fixed immutable relationship between retinal and tectal locations would not change. This suggests that marker gradients and concentrations may be subject to change (at least following such surgeries). Two recent models propose mechanisms by which marker concentrations in the tectum may change [Frean, 2006, Willshaw, 2006] (the latter is a detailed instantiation of a previous theoretical model [Willshaw and von der Malsburg, 1979]). In both models the amount of marker chemical in an innervating axon is supposed to induce changes in the level of the corresponding marker chemical in the tectal cell(s) with which it is connected. Such models are promising due to their ability to explain the results of the aforementioned lesion studies.

In summary, this section has presented examples of models in which maps form irrespective of the spiking or other electrical activity of the neurons involved.

Activity-dependent models By contrast to the examples in the previous section, other models show how maps can form based on Hebbian reinforcement of the correlated activity of neighbouring cells. Willshaw and von der Malsburg [1976] presented a model in which two 2-dimensional sheets of neurons represent two brain areas between which a topographic mapping should form, with excitatory connections from each neuron of the source layer to each neuron of the target layer. Neighbouring source-layer neurons are simultaneously active and target-layer neurons excite neighbouring or nearby neurons whilst inhibiting distant neurons. With a Hebbian mechanism applied to the weights of the synapses, the mapping develops so that neighbouring source-layer neurons maintained strong connections with neighbouring target-layer neurons whilst other connections become weakened (in fact this is one study, such as referred to in section 2.1.1.2, in which global synaptic normalisation is required in order to constrain the weight distributions yielded by the Hebbian mechanism). Intuitively, activity induced into neighbouring cells is self-reinforcing whilst activity induced into non-neighbouring cells causes them to compete, with the activity in those cells which benefit from more local reinforcement winning the competition; the higher activity levels both pre- and post-synaptically in neighbouring cells then causes Hebbian synaptic reinforcement. In order for the maps to be oriented correctly, however, an initial bias in the correct direction is required. Therefore some element of the explanation of topography involves processes more fundamental than the activity-dependent forces invoked in this model.

Similar mechanisms have been used in studies which seek to explain ocular dominance

column formation [Miller et al., 1989, Erwin and Miller, 1998]. In these studies, two input areas were used, with intra-correlations in their activity stronger than any inter-correlations, mimicking the supposed effect of input from two eyes being brought together on a single cortical area. Goodhill [1993] also addressed ocular dominance formation, though with a more abstract synaptic update rule which allowed that a global competition be evaluated and that all synapses within a region of the winning location should be strengthened. These assumptions are similar to the mechanism of the Kohonen map [Kohonen, 1982], a highly successful data clustering algorithm which took its inspiration from the process of biological topographic map formation. In an alternative activity-dependent mechanism [proposed by Elliott et al., 1996, Elliott and Shadbolt, 1998, 1999], activity in axons induces the release of neurotrophic factor from tectal cells, which diffuses locally, causing formation of synapses from nearby axons.

The above models assume that activity in the source layer of a mapping is spatially correlated, that is, that neighbouring or nearby neurons in the source layer are more like to be co-active. This would be a reasonable assumption for the visual system since positions close together in the visual field are more likely to receive similar stimuli. In the case of mammals though, retinotopic projections largely form prior to birth or to the opening of the eyes, when an organism would not receive any visual input. However in developing eyes which have not yet opened there are spontaneous waves of activity [Wong, 1999], which would provide spatial correlations in the absence of genuine visual input. There are also spontaneously generated spatially correlated patterns of activity in other areas, such as V1 [Chiu and Weliky, 2001]. The role of retinal waves and of spatial correlations in general is nevertheless debatable. The refinement of axonal arborisation and the development of ocular dominance are two phenomena which are at least partly activity dependent and are the key subjects in the model presented in this thesis. Regarding the refinement of axonal arborisation, in amphibia and fish (those animals where RGCs directly target an appropriate location), blocking all synaptic activity doesn't affect the formation of topography [O'Rourke et al., 1994], which suggests that activity independent mechanisms are responsible. However in a study involving mutant fish with no spiking activity in their RGCs, the axonal arbors were less refined [Gnuegge et al., 2001], suggesting that in at least some cases, activity dependent processes play an important role. Regarding the question of whether such activity needs to have spatial correlations, McLaughlin et al. [2003b] showed that mutant mice ("β2 knockouts" — lacking the gene for the β2 subunit of the nicotinic acetylcholine receptor, important for generating retinal waves), with RGC

activity in a key developmental period which is spontaneous but was thought not to be spatially correlated, have enlarged axonal arbors, suggesting that spatial correlations are necessary for some some map refinement. A recent publication, though, has cast doubt on this and other studies involving the aforementioned β2 knockout mice [Sun et al., 2008], since it showed that spatial correlations do exist where they were thought not to; they nonetheless had altered properties including a higher frequency, such that the original results might still be interpreted as showing that the precise nature of the spatial correlations in the retina are important for development. Butts et al. [2007] modelled a mechanism by which retinal wave activity could cause retinotopic refinement in LGN assuming a learning rule based on the timing of bursts (which they showed is not incompatible with various forms of STDP); this study is interesting in that the mechanism requires spatio-temporal correlations in the input structure, but this was used simply to reinforce existing preferences laid down by an activity-independent wiring mechanism. The model presented in this thesis works on a similar principle.

Regarding the development of ocular dominance, Ruthazer et al. [2003] showed how blocking NMDA receptors (understood to be necessary for activity-dependent plasticity) can prevent the formation of ocular dominance patterns, suggesting that an activity dependent mechanism may underlie ocular dominance development. By contrast, Crowley and Katz [1999] found that ocular dominance patterns could form even when all visual input was removed early in development.

Linsker [1986a,b,c] used a correlational mechanism similar in style to Willshaw and von der Malsburg [1976] to demonstrate the formation of spatial opponent cells and orientation-specific cells, arranged in columns. Notably, this model assumes that axons from the source layer terminate in random distributions around a pre-defined and immutable location in the target layer; the model presented in this thesis uses the same assumption. Linsker then used this framework as a basis for investigating the detailed formation of receptive fields. (A similar framework lies behind LISSOM models [Miikkulainen et al., 2005] and their descendants [Bednar et al., 2004]). The model shows how such receptive fields can develop, at least in principle, without any spatial correlations in the input; experimental results support such a possibility [Crair et al., 1998].

Combining activity-dependent and independent mechanisms The two previous sections have given examples of models in which either activity-independent processes or

activity-dependent processes give rise to topographic map formation. There are fair bodies of biological evidence to support each type of process, which raises the question of how they should be combined. According to some authors, activity-independent processes form broad topography whilst activity-dependent processes play a role in map refinement and receptive field development Ruthazer and Cline [2004]. However, especially given the various organisms, points in development and forms of receptive field development which have been studied, it is unsurprising that the relative contribution of these two types of mechanism continues to be a subject of debate. Recent studies have investigated the roles of molecular guidance and neural activity in parallel; e.g. [Pfeiffenberger et al., 2006] showed (with anteriorgrade tracing of RGC axons) that mice lacking genes for ephrin-A molecules have severely disrupted topography in the dorsal-ventral axis of the superior colliculus, β2 knockouts (with abnormal retinal waves during a critical period) had nearly normal topography but less refined axonal arbors, while mice with both genes missing had almost complete absence of topography in the dorsal ventral axis. They also observed that with both genes missing, topography was not so disrupted in the LGN, suggesting that other mechanisms are at work there (possibly due to the 3D laminar structure of the LGN). Cang et al. [2008] used functional optical imaging in the superior colliculus to support the previous study, finding that mice lacking only ephrin-As had topography in the dorsal-ventral axis which was global disrupted but locally clustered, compared with ephrin-A and β2 knockouts in which clustering was not apparent, suggesting that this clustering is the effect of the normal patterns of retinal wave activity.

Some models actively seek to combine different developmental processes (for example the abstract multiple constraints model of Fraser and Perkel [1990]), whilst a combination of processes is implicit in others, e.g. Butts et al. [2007] as noted above. Beyond this, it may not even be possible to separate the phenomena of molecular guidance and neural activity on which the different types of models are based, since, for example, Hanson and Landmesser [2004] showed that blocking normal bursting activity can disrupt pathfinding processes normally attributed to molecular guidance, and Bouzioukh et al. [2006] showed that changing the level of a guidance molecule can affect synaptic transmission.

The model presented in this thesis includes activity-dependent and -independent processes. Specifically, it looks at how an activity-dependent process may affect axonal arbor spread (or equivalently, receptive field spread) and ocular dominance pattern formation, given certain assumptions about the activity-independent process.

2.1.5.2 Weight vs wiring plasticity

A number of the models presented above model map development through networks with fixed connectivity where synaptic weights are subject to change [Willshaw and von der Malsburg, 1976, Miller et al., 1989, Goodhill, 1993, Song and Abbott, 2001, Willshaw, 2006]. In such models, a synaptic weight of zero is often interpreted as meaning that the synapse has been retracted or otherwise does not exist. Other models have considered the formation and elimination of synapses with fixed weight [Elliott and Shadbolt, 1999]. A mathematical equivalence between such models has been demonstrated under certain conditions [Miller, 1998].

There have been few attempts to include both forms of plasticity in a model. Miikkulainen et al. [2005] predominantly used synaptic weight change but supplemented this with a synaptic elimination process, which occurs periodically for all weights below a certain threshold. Willshaw and von der Malsburg [1979] include both synaptic weight change and rewiring; a more recent version of the same model [Willshaw, 2006] used only synaptic weight change, noting that (p. 2708) “A synaptic strength can be interpreted as the probability of a given retinal axon contacting a given tectal cell.”.

The model presented in this thesis considers both of these processes. Notwithstanding any mathematical equivalence as aforementioned, there are compelling reasons to include both processes. Firstly, since both processes are known to exist and to operate alongside each other, any model which seeks to fully explain topographic map development must ultimately include the two together. A second point regards the accuracy of topographic projection. Although a neuron may receive afferents from a wide area of the input space, topographic mapping can be very fine, theoretically even with precision higher than the spacing between neurons. This is because the location of peak sensitivity may contribute activation to nearby afferents of a target neuron in such proportions as to elicit its peak response. The ability of synapses to have different strengths allows further opportunities for the refinement of precision, over and above that afforded by the discrete formation of synapses. This is especially true where a receptive field consists of relatively small numbers of synapses. Thirdly, each synapse which exists has a cost to the organism in terms of the volume of brain it takes up and the energy required to maintain it and to carry it around. In these terms, all-to-all connectivity is prohibitively expensive, but rather only the most useful connections should exist. Chklovskii et al. [2004] argued that the ability of a brain to rewire its connections could substantially increase its capacity to store infor-

mation (which in the context of topographic maps might be interpreted as the formation of patterned receptive fields which reflect the statistics of the input activity); they raised the question of how the brain might implement search over a space of possible network topologies. The model presented hereafter proposes a possible mechanism, within the domain of topographic map formation. Finally, as noted in section 2.1.1.4, synaptic weight change and rewiring happen on different timescales. In general this may have practical consequences. For example, in associative memory studies, Fusi et al. [2005] demonstrate how allowing synaptic plasticity between states with different transition rates can improve the performance of the resulting neural network in terms of its ability to store new memories whilst retaining old ones, a finding which could be mapped to the differing timescales of weight and rewiring plasticity.

As noted above, some models only suppose elimination of synapses without formation. Whilst for the development of retinotopy the dominant trend appears to be over-elaboration of axonal arbors followed by pruning, nevertheless new synapses are added. The consideration of how, at least in Xenopus, receptive field location may change during development due to the changing ways that retina and tectum areas develop [Gaze et al., 1974] also suggests that synapse formation is a vital element for explaining such phenomena.

2.1.5.3 Lateral interactions in target layer

The work of Willshaw and von der Malsburg [1976] has been described above, in which, in order to refine receptive fields through coincident activity of neighbouring neurons, lateral interactions between cells in the target layer are hypothesised. These were used earlier by von der Malsburg [1973] to demonstrate a possible mechanism for the formation of orientation-selective receptive fields in V1, and have been used extensively since in various activity-dependent models of topographic map formation and receptive field development Linsker [1986a,b,c], Miller et al. [1989], Goodhill [1993], Song and Abbott [2001]. Specifically, there are short-range excitatory interactions and inhibitory interactions at longer range, which are often modelled as a difference of Gaussians, creating a "Mexican hat" profile. Such interactions are known to exist between cortical cells, though their actual profile is more complex [Kisvárday et al., 2006]. A major contribution of the models of Miikkulainen et al. [2005] was to apply the same plasticity rules to lateral synapses as was applied to feed-forward synapses (from the source to the target layer). Song and Abbott [2001] applied STDP rules to excitatory though not inhibitory connec-

tions, in order to explore how the spike-timing dependence of their learning rule would affect the performance of the lateral connections. They found that excitatory connections could act as a guide towards the development of similar preferences between neighbours and then be weakened at a later point in development, though this prediction has not yet been observed *in vivo*. They also found that, given an initial bias towards a desired topology, their strongly stabilising weight-independent STDP rule could act to refine the topology without the need for inhibitory connections. Linsker [1986b] showed how the interplay between long and short range connections could be responsible for the formation of orientation preferences in the absence of correlated input. The range of excitatory lateral interactions was shown by Goodhill [1993] to influence the width of ocular dominance stripes. Elliott and Shadbolt [1998] then showed that the amount of correlation between the inputs from the two eyes could affect the width of stripes, a result which is consistent with observations that in kittens with strabismus (thus having less inter-correlations between the inputs from the two eyes), ocular dominance stripes are wider (and more sharply delineated) [Lowel, 1994].

The model which is presented in this thesis assumes short-range excitatory lateral interactions but no longer-range inhibitory interactions. This follows from Song and Abbott's observation, as noted above. Intuitively, the purpose of long-range inhibitory interactions within an activity-dependent model is to ensure that different parts of the map develop different input preferences. However, if an activity-independent process is assumed to create and maintain a broad topography, arguably different regions of the target layer are constrained to be innervated by different parts of the input space and therefore differing input preferences are enforced, rendering long-range inhibitory interactions redundant. The same argument could also be applied to short-range excitatory interactions and such possibilities are investigated.

2.1.5.4 Growth of areas

As noted above in section 2.1.4, in some animals the mapping between retina and tectum forms as the areas themselves are growing and therefore projections must change during development; this was modelled by Willshaw and von der Malsburg [1976] and Willshaw [2006]. In an alternative approach to development, Miikkulainen et al. [2005] extended their LISSOM models such that new neurons would be inserted between existing ones, so that the mapping between points in the source and target spaces remained constant.

Whilst mappings between areas are typically continuous, there can be some transformation in scale or rotation and this can vary across a mapping and can change during development. For example, the mapping between the retina and V1 is such that central (foveal) locations in the retina map to medial positions in V1 whilst peripheral locations in the retina map to lateral positions in V1. The foveal area is over-represented in V1, such that moving a given small distance across medial V1 shifts the receptive field centre a smaller amount across the visual field compared to the same shift across lateral V1 [Albus, 1975, Adams and Horton, 2003].

The model presented in this thesis does not explicitly include either the growth of areas or transformations between areas; however, it is compatible with both and could be extended to include them, as will be discussed in section 2.4.4.

2.2 Model

2.2.1 Overview

In this section a model of map formation is presented. The model is intended to be general to the extent that it could apply equally to retinotectal, retinocollicular, retinogeniculate or geniculocortical projections. In brief, this model proposes the following:

1. Activity independent processes fully specify a topographic mapping between a source and target area and guide axons from the source area towards their "ideal" location in the target area, i.e. the location dictated by the topographic mapping. The mechanism that yields this mapping is unspecified; it could be thought of as a type I chemoaffinity mechanism (as defined in section 2.1.5.1) with fixed affinities [Gierer, 1983, Honda, 1998], though other mechanisms could be inserted.

2. Axon branching leads to formation of synapses over an area surrounding the ideal topographic location (broadly in line with, for example, the innervation of the tectum [McLaughlin et al., 2003a], though simplifying the directional overshoot of axons observed in chick and rodent).

3. Competitive Hebbian learning detects correlations in input patterns due to spatial proximity in the source area, such that synapses from more spatially clustered afferent neurons are strengthened at the expense of synapses from neurons which are

more distant from other afferents. The effective spread of the receptive fields of target neurons in the source area is thereby reduced; this follows the model of Song and Abbott [2001]. To the extent that receptive fields contain input-specific features, such as ocular dominance segregation, then these arise from this process.

4. Preferential elimination of weak synapses (as discussed in section 2.1.1.4) allows the reduction of spread to be embedded in the network topology, offering a possible cause for the reduction in axonal arbor spread seen by, for example, McLaughlin et al. [2003b].

5. To the extent that this process continues, with further creation and elimination of synapses, there is the potential for the spread to be reduced further.

The model can be seen as a unique synthesis of existing ideas and elements of models. The primary purpose is to investigate the interplay between two types of plasticity, weight change and rewiring, as they relate to topographic mapping and receptive field development. The phenomena which are focused on are changes in spread of receptive field and the development of ocular dominance.

2.2.2 Details of the model

In this section, greater detail is given, and distinctions are drawn between those properties which are general and those specific ones which have been adopted in order to develop a system which is amenable to tractable simulation and analysis. Thus, a class of models is defined, though only a small subset have been simulated within this project.

There are two layers, the input layer and the target layer. Layers are 2D spaces on which neurons are located; the words “layer” and “area” are used interchangeably hereafter. Each location in one layer has a corresponding ideal location in the other, such that one layer maps smoothly and completely to the other. In general, the layers could be of any shape and the transformation that maps one layer to the other could be any that does not require discontinuities. For simulation, neural areas are square grids of neurons, the two layers are the same size, periodic boundaries are imposed to avoid edge artefacts (see section 2.3.2.1).

Each cell in the target layer can receive a maximum number of afferent synapses. It can be said then that each cell has a certain synaptic capacity (a concept explored by Bougeois

and Rakic [1993]). For simulation, all target layer neurons have the same synaptic capacity. A set of connections from one layer to another is referred to hereafter as a *projection*; this can also refer to a set of connections from one layer to itself. There are two excitatory projections, a feed-forward projection from the input layer to the target layer and a lateral projection from the target layer back to itself. Axons within these projections compete for the synaptic capacity of the target neurons. As noted in section 2.1.5.3, for simplicity, inhibitory lateral interactions are not implemented in this model. Some fundamental features of this model also function without excitatory lateral interactions, as will be demonstrated later in section 6.2.3.

It is assumed that an unspecified activity-independent process is capable of guiding the formation of new synapses so that they are distributed around their ideal locations. A Gaussian distribution is assumed, since a process which is initially directed towards a target site and then randomly branches on its way would yield a Gaussian distribution of terminations around the target site. To implement the Gaussian distributions, where a target neuron has fewer than its maximum number of dendritic synapses, the remaining slots are considered "potential synapses". At a fixed rate, a synapse from the neurons of the target layer is randomly chosen. If it is a potential synapse, a possible pre-synaptic cell is randomly selected (for the simulations which follow, the last cell to have fired is used as a possible pre-synaptic partner) and synapse formation occurs when:

$$r < p_{form} e^{-\frac{\delta^2}{2\sigma^2_{form}}} \tag{2.2}$$

where r is a random number uniformly distributed in the range $(0,1)$, p_{form} is the peak formation probability, δ is the distance of the possible pre-synaptic cell from the ideal location of the post-synaptic cell and σ^2_{form} is the variance of the connection field. In other words, a synapse is formed when a uniform random number falls within the area defined by a Gaussian function of distance, scaled according to the peak probability of synapse formation, (which occurs at $\delta = 0$). This is a rejection sampling process.

Lateral connections are formed by the same means as feed-forward connections, though σ_{form} can be different for each projection. For simulation, p_{form} was set so as to allow the same overall probability of formation for each projection. This is because in the absence of a general rule for the relative numbers of feed-forward vs lateral connections formed, starting with equal numbers of each is a good basis for observing the relative development of these projections.

If the synapse which has been selected from the neurons of the target area already exists (i.e. it is an actual synapse, rather than a potential one) then it is considered for elimination. In general it is proposed that the probability of elimination should be some monotonically decreasing function of weight. For simulation, due to the nature of the chosen learning rule (weight-independent spike-timing-dependent plasticity) which tends to deliver a bimodal weight distribution, the probability of elimination has been simplified to one of two values, with a higher value for synapses with weights below a certain threshold $\left(P_{elim-dep}\right)$ and *vice-versa* $\left(P_{elim-pot}\right)$.

In general, synapses implement some competitive Hebbian learning rule, such that correlations in inputs to a given target neuron result in preferential strengthening of those synapses at the expense of the strength of other synapses. The neuron model and inputs used should be of types which support the chosen synaptic process. For simulation, the synapses, neurons and type of input are based on the model of Song and Abbott [2001], i.e. with integrate-and-fire neurons with synaptic modulation governed by STDP. The criticisms of STDP raised in section 2.1.1.3 should be borne in mind; the model inevitably loses some generality due to this decision. Most models of map formation use more abstract learning rules. This may be partly motivated by computational constraints but also by a desire for simplicity. Miikkulainen et al. [2005] noted that choosing STDP makes it difficult to interpret the interacting units of a model as any aggregation larger than a single neuron. Nevertheless the model of Song and Abbott [2001] has been chosen because: (1) its applicability to the modelling of topographic map formation has already been demonstrated, such that existing results can be built on; (2) the use of STDP is attractive because it implements a competitive Hebbian learning rule without requiring additional processes for weight normalisation; (3) STDP is a form of weight change which is known to occur in biological neurons, and as such, has the potential to add biological realism to a model (although lack of knowledge of suitable parameters may undermine this advantage) — the study of Young et al. [2007] is a case in point as it showed how qualitatively different predictions arose from applying a spike timing dependent learning rule as opposed to a rate-based learning rule, which better matched an observed phenomenon. In the computational model described here, weight independent STDP is adopted, both in keeping with Song and Abbott [2001] and in order to reduce the parameter space of the model; for the neuromorphic VLSI implementation which will be described in section 3.4.4, partially weight-dependent update rules are investigated due to synergistic properties of the silicon substrate, and in so doing the space of possible models explored is expanded, for added

generality. The detail of the model used, including neuron and synapse dynamics, inputs and initial conditions, is given in algorithm 1.

An unjustified assumption, used by Song and Abbott [2001] and adopted here, is that new synapses start strong and then get weakened; the opposite case seems more likely when the process of synapse formation is considered, since for a synapse to grow to be a big synapse with many vesicles and a large post-synaptic density it must first pass through a stage in which it is a small synapse with few vesicles and a small synaptic density. This assumption has been used for simplicity because it avoids the need for any additional homeostatic mechanisms to kick-start the activity of the network.

2.3 Methods

2.3.1 Experimental parameters

In this section, the process by which the model was parameterised is explained. Parameters for the following simulations are given in table 2.1.

Simulations were run with a C++ function, with initial conditions created and data analysis carried out with Matlab. Simulations used a time step of 0.1ms and rewiring simulations typically settled within 5 minutes of simulated time; therefore 3,000,000 iterations were performed, and at each time step several matrix operations were performed over all synapses. Full scale simulations were therefore very computationally intensive, necessitating the use of relatively small numbers of neurons and synapses; this in turn necessitated a rigorous statistical approach. The actual numbers used for simulations and later for implementation were arrived at after a lengthy period of experimentation. The size of the grids representing neural layers was 16×16 (i.e. number of neurons in a layer, $N_{layer} = 256$), enough that discernable patterns of ocular dominance might be observed (c.f. 25×25 [Miller et al., 1989]; 32×32 [Goodhill, 1993]; 20×20 [Elliott and Shadbolt, 1999]). In determining the maximum fan-in, or number of potential afferent synapses per target neuron, it was found that as the fan-in reduced, the performance of STDP as a correlation detector degraded and the bimodal distributions generated were less extreme; this could be compensated to some extent by building stronger correlational cues into the inputs, as shown by Bofill-i Petit [2005] who achieved strong segregation between just 6 synapses with carefully constructed inputs. Therefore the choice of fan-in represents a compromise

Algorithm 1 Model summary

There are two layers of the same size, the *input* and *target* layers; each is a square grid of neurons with periodic boundaries, and the ideal location of each neuron in the input layer is the location with the same coordinates in the target area. Each target-layer neuron has the same number of *potential synapses*; these are dendritic locations in which actual synapses may form; synapses can be with a pre-synaptic neuron from either the input or target layer, including the post-synaptic neuron itself.

Initial conditions: all potential synapses start formed, with conductance g_{max}.

Input: neurons are independent Poisson processes. A stimulus location s is randomly chosen and firing rates are set to $f_{base} + f_{peak} exp\left(-d/2\sigma^2_{stim}\right)$, where d is the distance from s. With a period t_{stim}, s moves and the process repeats.

Neuron dynamics (target-layer): the membrane voltage V_{mem} is described by:

$$\tau_{mem}\frac{\delta V_{mem}}{\delta t} = V_{rest} - V_{mem} + g_{ex}(t)\left(E_{ex} - V_{mem}\right)$$

E_{ex} = excitatory reversal potential; V_{rest} = resting potential; τ_{mem} = membrane time constant. Upon reaching a threshold V_{thr}, a spike occurs and V_{mem} is reset to V_{rest}. A pre-synaptic spike at time 0 causes a synaptic conductance at time $t \geq 0$ of $g_{ex}(t) = ge^{\frac{-t}{\tau_{ex}}}$ (τ_{ex} = synaptic time constant); this is cumulative for all pre-synaptic spikes.

STDP: a pre-synaptic spike at time t_{pre} and post-synaptic spike at t_{post} modify the synaptic conductance by $g \rightarrow g + g_{max}F(\Delta t)$, where $\Delta t = t_{pre} - t_{post}$ and $F(\Delta t) = A_+ exp\left(\Delta t/T_+\right)$ if $\Delta t < 0$, otherwise $F(\Delta t) = -A_- exp\left(-\Delta t/T_-\right)$, where $A_{+/-}$ are magnitudes and $\tau_{+/-}$ are time constants for potentiation and depression respectively. This is cumulative for all pre- and post-synaptic spike pairs; g is bounded in $(0, g_{max})$.

Synaptic rewiring: At a fixed rate f_{rew}, a potential synapse is randomly chosen. If it is an actual synapse, the elimination rule is used, otherwise the formation rule is used.

Formation rule: A possible pre-synaptic neuron is randomly selected from either the input or target layer and synapse formation occurs if:

$$r < p_{form} e^{-\frac{\delta^2}{2\sigma^2_{form}}}$$

r = uniform random number in $(0,1)$; p_{form} = peak formation probability; δ = distance of possible pre-synaptic neuron from ideal location of post-synaptic neuron; σ^2_{form} = variance of the connection field. p_{form} and σ^2_{form} may differ based on which layer the possible pre-synaptic neuron is from.

Elimination rule: If the synapse's conductance is below $0.5g_{max}$ it is eliminated with probability $p_{elim-dep}$, otherwise probability $p_{elim-pot}$ is used.

between amount of computation necessary and the desire to use more realistic input spike trains. Data is scarce on actual number of dendritic synapses in areas such as the tectum or superior colliculus, so it is difficult to say what a biologically realistic number might be — it may vary greatly between different organisms, brain areas and developmental stages. For most of the simulations in this chapter the maximum fan-in (S_{max}) was 32, though it was increased to 64 for one set of experiments.

For the silicon implementation described later in this thesis, the maximum fan-in was 64. Although it was possible to observe in simulation the amount of variation in neuron and synapse behaviour that could be expected (see section 3.5.7) it was difficult to predict how that might impair the performance of the model, various simulations of such effects notwithstanding. Therefore having achieved good results with a fan-in of 32 in simulations (see below in section 2.4) extra synaptic capacity was allowed for in the implementation to give a margin for error, since this would not impair speed performance as it did in simulations.

Data is scarce on appropriate values for the probabilities governing synapse formation and elimination. However, dendritic spines have been imaged extending and retracting over periods of hours compared with others stable over a month or more [Grutzendler et al., 2002, Trachtenberg et al., 2002]. In the simulations, much higher rates were used so that synapses had several chances to rewire during the short periods for which it was tractable to run simulations, while maintaining a large difference between these probabilities (in practice a factor of 180 was used, representing the difference between 4 hours and 1 month, i.e. $p_{elim-pot} = p_{elim-dep}/180$). The value of p_{form} works together with the rewiring rate ($f_{rew} = 10^4 Hz$, an arbitrary choice), the number of synapses ($16 \times 16 \times 32 = 8192$), σ_{form}, and the topology of the area to define the actual rate of formation. $\sigma_{form-feedforward}$ was given a larger value than $\sigma_{form-lateral}$, in line with generic parameters given in Miikkulainen et al. [2005]. Once a synapse has been eliminated there is no computational benefit from not being formed again as soon as possible, therefore $p_{form-lateral} = 1$, so that if a possible pre-synaptic partner is presented whose ideal location matches the location of the post-synaptic neuron, then the match is accepted. Since $\sigma_{form-feedforward} > \sigma_{form-lateral}$, $p_{form-feedforward}$ should be less than $p_{form-lateral}$ in order to balance the overall probability of synapse formation with each afferent layer; in fact, to achieve this balance, $p_{form-feedforward} = p_{form-lateral} . \sigma^2_{form-lateral} / \sigma^2_{form-feedforward}$. The mean formation rate can then be calculated. $P_{elim-dep}$ was set at half the mean formation rate so that weak

synapses would be eliminated half as often as potential synapses became actual synapses, so that the majority of the potential synapses would be formed at any point. In practice, for the parameters given, depressed synapses were eliminated after an average of 33s whereas strong synapses would only be eliminated with a probability of ≈ 0.05 within a 5 minute simulation.

Regarding inputs, the stimulus location changed regularly every 0.02s. This regularity is a move away from the model of Song and Abbott 2001 in which t_{stim} was chosen according to an exponential distribution; this was a necessary concession to provide stronger correlation cues (i.e. more effective symmetry breaking) given the smaller number of synapses per neuron. A further concession was the more extreme values of the base and peak firing frequencies, f_{base} and f_{peak}. The spread of the stimulus, σ_{stim}, was chosen to be between the values of $\sigma_{form-feedforward}$ and $\sigma_{form-lateral}$ and f_{peak} was set so as to keep the overall mean firing rate at a value, f_{mean}, which was chosen to allow sufficient difference between f_{base} and f_{peak}. Butts et al. [2007] following Sjostrom et al. [2001] suggested a way in which the STDP rule itself could be modified to deal better with burst-based spiking, but this possibility has not been pursued in this project.

For the neuron and synapse dynamics, parameters were set starting from parameters given in Song and Abbott [2001]. A_+ was increased 20-fold as a concession to limited computational resources for simulations (this should not qualitatively change the model since many plasticity events are still needed to potentiate a depressed synapse). Then key parameters were varied, in order to maintain key conditions, which were: The total weight should be approximately 50% of the maximum possible; the average target neuron firing rate should approximately match the average input firing rate; and the total weight of lateral synapses should roughly match the weight of feed-forward ones. The parameters which were varied are as follows. The peak synaptic conductivity, g_{max}, was varied, since this affects the amount of stimulus the neurons receive and thus their firing rates. The ratio of time constants for depression and potentiation, τ_-/τ_+, was varied, since this affects the relative weights of feed-forward and lateral synapses, (since correlated feed-forward synapses benefit less from symmetry breaking when the ratio increases); in practice τ_+ was held constant whilst τ_- was varied, as in Song and Abbott [2001]. The ratio of depression to potentiation, $B = A_-\tau_-/A_+\tau_+$, was varied, since this affects the balance of weights; in practice, A_- was treated as the free parameter in order to vary B, however B is quoted, since its meaning is more intuitive. In the interests of simplicity, B was constrained to

Table 2.1: Simulation parameters

Wiring	Inputs	Membrane	STDP
$N_{layer} = 16 \times 16$	$f_{mean} = 20Hz$	$V_{rest} = -70mV$	$A_+ = 0.1$
$S_{max} = 32$	$f_{base} = 5Hz$	$E_{ext} = 0V$	$B = 1.2$
$\sigma_{form-feedforward} = 2.5$	$f_{peak} = 152.8Hz$	$V_{thr} = -54mV$	$\tau_+ = 20ms$
$\sigma_{form-lateral} = 1$	$\sigma_{stim} = 2$	$g_{max} = 0.2$	$\tau_- = 64ms$
$p_{form-lateral} = 1$	$t_{stim} = 0.02s$	$\tau_m = 20ms$	
$p_{form-feedforward} = 0.16$		$\tau_{ex} = 5ms$	
$p_{elim-dep} = 0.0245$			
$P_{elim-pot} = 1.36e^{-4}$			
$f_{rew} = 10^4Hz$			

having the same value for different projections feed-forward vs lateral). In practice, it was difficult to find a good set of parameters, since they are interdependent. For example, varying the spike rate changes the balance of the weights, and *vice versa*. Moreover, a single set of parameters inevitably leads to different results depending on the nature of the inputs and depending on whether rewiring was implemented, so there are inevitably confounding factors when attempting to compare these different cases.

Initial placement of synapses was performed by iteratively generating a random pre-synaptic partner and carrying out the formation rule. Feed-forward and lateral connections were placed separately, up to their initial number of 16 synapses each.

2.3.2 Analysing topographic map quality

2.3.2.1 Previous approaches

In order to address the question of how developmental processes affect the quality of topographic maps there must be some way of assessing this quality. Goodhill and Sejnowski [1996] reviewed various measures of assessing neighbourhood preservation based on comparing the similarity of pairs of positions and their images in the mapped space. By comparing the performance of various different measures on a set of mappings they made it clear that different assumptions about what aspect of the mapping is important lead to different quality judgements. Many of the methods reviewed would be overly complex for

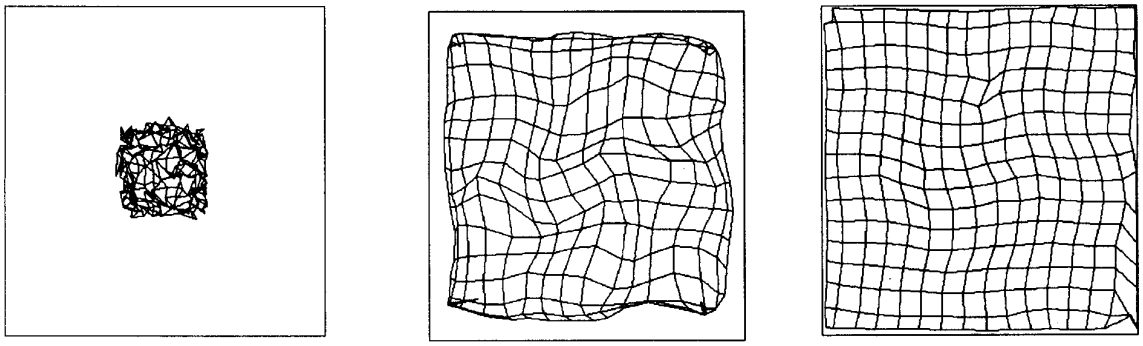


Figure 2.2: Map formation sequence from Goodhill 1993, figure 2.

the problem at hand, since they do not assume that there is an ideal frame of reference against which a proposed mapping can be compared. The following discussion focuses on cases where there is a known ideal mapping against which a mapping can be judged. The Procrustes method [Zaidel and Iacoboni, 2003, p. 77] is another general approach which is inappropriate here. In the Procrustes method, a linear transformation is calculated which, when applied to one set of co-ordinates, minimises their summed squared distance error from another set. Mapping errors are thus divided into systematic errors (defined by the resulting transform) and residual errors. Such a distinction is not important here.

Goodhill [1993] presented a model of topographic map formation which focused on the conditions for achieving ocular dominance stripe formation. In this model, all-to-all connectivity was assumed between neurons in two (square) grids, representing the retina and a cortical mapping thereof. The weights of connections were initialised randomly with a bias towards the topographically appropriate locations, based on identity of the square spaces. The topography of the mapping was then illustrated by plotting the centre of mass of the connections for each of the cortical cells in the retinal space; this can be interpreted as the centre of the receptive field for each cortical cell. The result is shown in figure 2.2, which gives a sequence of snapshots from initial through to final topography. The mapping seems to unfold during development, with the retinal position for each cortical cell converging on the topographically appropriate location. However, the mapping was initialised based on perfect topography and the systematic shift away from the topographically appropriate location towards the centre of the retinal space is purely an effect of the space being bounded, since a cell towards the edge of the area has more synapses from retinal locations to one side of its topographically appropriate location than the other.

This was pointed out by Elliott and Shadbolt [1999] who make a complementary contri-

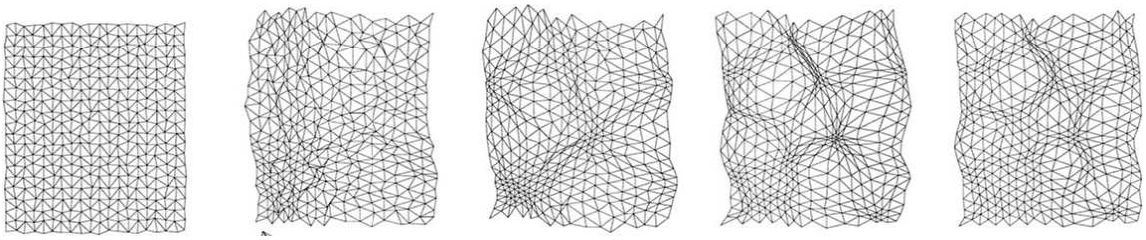


Figure 2.3: Map formation sequence from Elliott and Shadbolt 1999, figure 2.

bution. In a model of topographic map formation based on activity-dependent release of diffusable neurotrophins, they worked again with two square spaces but defined these with a toroidal, or wrap-around, topology. They then defined the centre of mass measurement for a target cell x as:

$$\vec{M}_x^{CoM} = \frac{\sum_i \vec{p}_{xi} s_{xi}}{\sum_i s_{xi}} \tag{2.3}$$

where $\vec{p}_{xi}$ is the spatial position of afferent i relative to the spatial position of the afferent that would uniquely project to target cell x were topography perfect, and s_{xi} is the number of synapses between afferent i and target cell x (in this model, synapses have unitary weight). The results of the system are shown in figure 2.3, again in a formation sequence from left to right. Now it can be seen that the topology starts off as perfect on average, due to an initial bias towards correct topology, and it appears that the effect of the activity-dependent development mechanism is in fact to introduce systematic shifts away from perfect topology, in this case due to varying activity levels in the source layer.

Both authors pointed out that their visualisations of topography give incomplete information, and supplemented them with visualisation of receptive field spread, showing that this reduces during development. However, the centre of mass measure of topographic position is affected by the total weight of the afferents (whether this is measured as a sum of the weights of individual synapses, or as the number of unitary synapses), since the fewer contributions there are to the final position, the more randomness will be evident in the final positions. A novel method of analysing results to avoid this effect is presented below in section 2.3.2.2.

2.3.2.2 Current approach

For calculating the preferred location for each target cell, the use of the centre of mass measure as in Elliott and Shadbolt [1999] would be erroneous. The space is toroidal but, as the authors note, the centre of mass is always calculated relative to perfect projections. Therefore the calculation of preferred location would be skewed by the choice of reference point from which synapses' coordinates are measured. This bias has been avoided by the novel method of searching for the location around which the afferent synapses have the lowest weighted variance (σ^2_{aff}), i.e.:

$$\sigma^2_{aff} = \underset{\vec{x}}{\operatorname{argmin}} \frac{\sum_i w_i |\vec{p}_{xi}|^2}{\sum_i w_i} \tag{2.4}$$

where i is a sum over synapses, $\vec{x}$ is a candidate preferred location, $|\vec{p}_{xi}|$ is the minimum distance from that location of the afferent for synapse i and w_i is the weight of the synapse (if connectivity is evaluated without reference to weights, synapses have unitary weight). This has been implemented with an iterative search over each whole number location in each dimension and then a further iteration to locate the preferred location to 1/10th of a unit of distance (the unit is the distance between two adjacent neurons). Note that in the non-toroidal case this location is equivalent to the centre of mass, as used in Goodhill [1993]; hereafter it will be referred to as the "preferred" location.

Having calculated the preferred location for all the neurons in the target layer, the mean of the distance of the preferred location from the ideal location was taken to give a mean Absolute Deviation (*AD*) for the projection. By reporting both mean *AD* and mean σ_{aff} for a projection there is a basis for separating the spread of the receptive fields from the deviation of their preferred locations from their ideal locations. However *AD* and σ_{aff} are both dependent on the numbers and strengths of synapses and these can change during development. Therefore to observe the effect of the activity-dependent development mechanism irrespective of changes in synapse number and strength, comparison was made in two ways. Firstly, for evaluating change in mapping quality based only on changes in connectivity without considering the weights of synapses, a new map was created by taking the final number of synapses for each target neuron and randomly placing them in the same way as the initial synapses were placed. σ_{aff} and *AD* were then calculated for each neuron in each of the maps and the means of these (i.e. mean σ_{aff} and mean *AD*) were compared, applying significance tests between the values of two populations of neurons,

i.e. all the neurons on the final map *vs* all those on the reconstructed map. Having established what effect there was on connectivity, the additional contribution of weight changes was considered, by creating a new map with the same topology, taking the final weights of synapses for each target neuron and randomly reassigning these weights amongst the existing synapses for that neuron. The two maps were then compared as described above.

2.4 Results and discussion

Three main experiments were carried out: case 1 had both rewiring and input correlations, as described in section 2.2; case 2 had input correlations but no rewiring; case 3 had rewiring but no input correlations (i.e. all input neurons had rate f_{mean}). The results are given in table 2.2.

For comparisons, mean σ_{aff} and mean AD were each calculated for the feed-forward connections of the following networks: (a) the initial state with weights not considered (recall that all weights were initially maximised) - these results are suffixed "*init*", i.e. mean AD_{init}; (b) the final ("fin") network with weights not considered but only connectivity ("*con*") with all synapses weighted equally, i.e. mean $AD_{fin-con}$; (c) for comparison with mean $AD_{fin-con}$, the final number of synapses for each target neuron, randomly placed ("$shuf$") in the same way as the initial synapses (not applicable for simulations with no rewiring), i.e. mean $AD_{fin-con-shuf}$; (d) the final network including weights, i.e. mean $AD_{fin-weight}$; (e) for comparison with mean $AD_{fin-weight}$, the final connectivity for each target neuron with the actual weights of the final synapses for each target neuron randomly reassigned amongst the existing synapses, i.e. mean $AD_{fin-weight-shuf}$. Results were compared using Wilcoxon Signed-Rank (WSR) tests on AD and σ_{aff} for incoming connections for each target neuron over the whole target layer for a single simulation of each of the two conditions under consideration.

2.4.1 Receptive field spread and the effect of rewiring

The effect of rewiring can be seen by comparing case 1 (with rewiring) and case 2 (without rewiring). Considering topology change, in case 1 mean $\sigma_{aff-fin-con}$ drops to 1.95, c.f. 2.32 for mean $\sigma_{aff-fin-con-shuf}$; this drop is significant. In case 2 mean $\sigma_{aff-fin-con}$ is constrained to remain at mean $\sigma_{aff-ini} = 2.36$. Considering weight change, in case 1,

Table 2.2: Summary of simulation results: case 1: rewiring and input correlations; case 2: input correlations and no rewiring; case 3: rewiring and no input correlations

Case	**1**	**2**	**3**
Target neuron mean spike rate	24.7	17.4	10.5
Final mean number of feed-forward incoming synapses per target neuron	14.1	NA	12.5
Weight as proportion of max for the initial number of synapses	0.60	0.36	0.33
Mean $\sigma_{aff-init}$	2.36	2.36	2.36
Mean $\sigma_{aff-fin-con-shuf}$	2.32	NA	2.32
Mean $\sigma_{aff-fin-con}$	1.95	2.36	2.17
p (WSR $\sigma_{aff-fin-con}$ vs $\sigma_{aff-fin-con-shuf}$)	2.4×10^{-25}	NA	5.0×10^{-6}
Mean $\sigma_{aff-fin-weight-shuf}$	1.88	2.10	1.99
Mean $\sigma_{aff-fin-weight}$	1.70	1.98	1.95
p (WSR $\sigma_{aff-fin-weight}$ vs $\sigma_{aff-fin-weight-shuf}$)	2.7×10^{-27}	8.7×10^{-6}	0.028
Mean AD_{init}	0.78	0.78	0.78
Mean $AD_{fin-con-shuf}$	0.89	NA	0.90
Mean $AD_{fin-con}$	0.83	0.78	0.93
p (WSR $AD_{fin-con}$ vs $AD_{fin-con-shuf}$)	0.31	NA	—
Mean $AD_{fin-weight-shuf}$	0.92	1.36	1.21
Mean $AD_{fin-weight}$	0.95	1.58	1.34
p (WSR $AD_{fin-weight}$ vs $AD_{fin-weight-shuf}$)	0.48	0.0012	—

mean $\sigma_{aff-fin-weight}$ drops to 1.70, c.f. 1.88 for mean $\sigma_{aff-fin-weight-shuf}$. In case 2, mean $\sigma_{aff-fin-weight}$ drops to 1.98, c.f. 2.10 for mean $\sigma_{aff-fin-weight-shuf}$. Both drops are significant.

Mean $\sigma_{aff-fin-weight}$ appears to be lower in case 1 than case 2. It is not possible to say for sure that this superior reduction of variance is due to the effect of the rewiring mechanism because the different numbers and weights of final synapses in each case make a comparison impossible. However, there is a good reason to believe that this is so: the drop in mean $\sigma_{aff-fin-con}$. This drop on its own indicates that the rewiring mechanism has helped to reduce variance and would also lay the groundwork for different final measures of σ_{aff} when weights are considered.

It can be seen then that (a) the weight-changing learning rule causes some reduction in the variance of the receptive fields, and (b) when the rewiring mechanism is applied, the network topology develops such that a variance reduction can be observed in the placement of the synapses, irrespective of their weight. Since the rewiring mechanism on its own can only generate synapse distributions according to the variance used by the formation rule it has no means to reduce this variance except the influence from the effect of the weight change mechanism, whereby outlying synapses are weakened and become subject to preferential elimination. Thus, the variance reduction is caused by the weight-change mechanism and becomes embedded in the network topology as a result of the rewiring mechanism.

It can also be seen qualitatively that the effect of rewiring is to embed in the connectivity of the network input preferences which arise through the weight changes mediated by the learning rule. STDP favours causal inputs with the lowest latency and local excitatory lateral connections tend to lose the competition with excitatory feed-forward connections as they have a higher latency [Song and Abbott, 2001]. The extreme of this effect can be seen in synapses from a target neuron back to itself (“recurrent” synapses). The placement rule allows these synapses to form, but they only ever receive a pre-synaptic spike immediately following a post-synaptic spike and therefore they are always depressed by the learning rule. Figure 2.4-left shows the initial density of incoming lateral synapses from pre-synaptic partners at given distances out from the post-synaptic neuron. It can be seen that the average neuron receives more synapses from itself (those with zero distance of pre-synaptic neuron from post-synaptic neuron) than from any of its closest neighbours. Figure 2.4-middle shows the final distribution where synapses are weighted. The recurrent

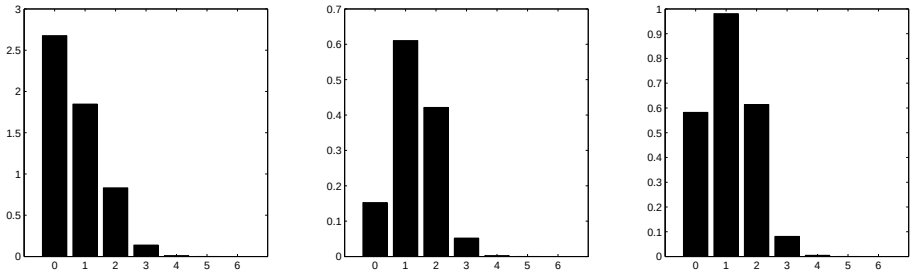
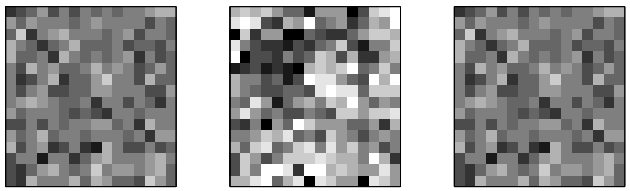


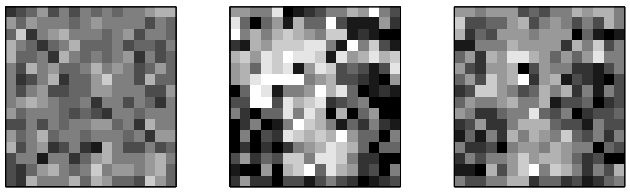
Figure 2.4: Normalised weight density of incoming lateral synapses (weight/unit area; y-axis) radially sampled and interpolated at given distances of pre-synaptic neuron from post-synaptic neuron (x-axis), averaged across population. Left: Initial connectivity (weights maximised); middle: final connectivity, weighted by final synaptic weights; right: final connectivity, not considering weights, i.e. each synapse is considered to have unity weight.

synapses have been depressed much more than their neighbours. Figure 2.4-right shows the final distribution only considering numbers of synapses and not their weights. The proportion of recurrent synapses to lateral synapses with neighbours has reduced from the initial state, due to the preferential elimination of the weak recurrent synapses.

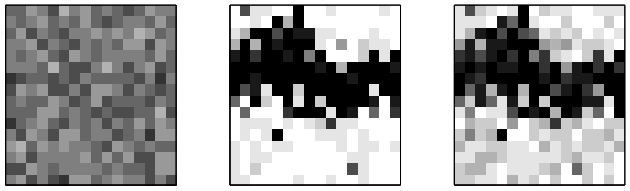
As a further demonstration of the effect of rewiring, a simulation was carried out with the input neurons divided into two groups, mimicking the effect of binocular inputs. The groups were interspersed in a chequered pattern, i.e. each input neuron was in the opposite group to its 4 orthogonally adjacent neurons; the stimulus location switched between the two groups every time it changed. To keep the overall input rate the same, the peak firing rate was doubled. Figure 2.5(b)-left shows the initial preference of each target neuron for input neurons in the two groups. Figure 2.5(b)-middle shows the final ocular dominance map where synapses are weighted. Although the space used was too small and the result of the learning rule with a small number of synapses too random for familiar striped ocular dominance patterns to emerge, ocular dominance zones can be seen. This pattern is reflected in the final map of connectivity in figure 2.5(b)-right, where synaptic weights are not considered. For comparison, the results in figure 2.5(a) are from an experiment which differed only in that rewiring was not performed; in this case, although some pattern of ocularity preference arises in the weights of the connections, this pattern cannot be transferred to the network topology. This, therefore, is another example of weight patterns caused by input activity becoming embedded in connectivity patterns by the rewiring mechanism.



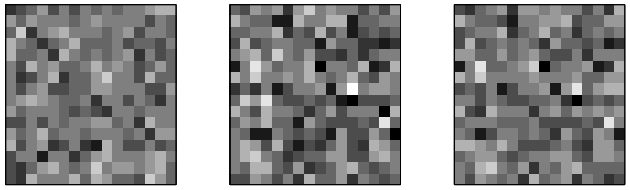
(a)



(b)



(c)



(d)

Figure 2.5: Ocular preference maps. Within each raster, each cell represents a target neuron, and is shaded on a scale from white to black according to the (weighted) proportion of its afferent synapses which are from one of two separately intra-correlated input spaces interspersed in the input space. Left: Initial connectivity (weights maximised); middle: final connectivity, weighted by final synaptic weights; right: final connectivity, not considering weights, i.e. each synapse is considered to have unity weight. (a) No rewiring. (b) With rewiring. (c) As (b), but with double the maximum number synapses (i.e. $N_{syn} = 64$), and $\sigma_{form-feedforward} = 1.99$, $\sigma_{form-lateral} = 2.49$, with initial synapse numbers and peak formation probabilities adjusted to create a corresponding ratio of feed-forward to lateral synapses. (d) Control experiment with full intercorrelation of input spaces (case 1 in table 2.2).

The effect is shown more clearly in a further simulation (figure 2.5(c)) in which double the number of synapses were used (i.e. 64 in total), and in which, parameters were optimised for the production of ocular dominance segregation. In particular the range of the lateral excitatory connections were increased (with a balancing reduction in the range of the feed-forward projection), based on the knowledge from Goodhill [1993] that this can reduce the spatial frequency of an ocular dominance pattern. In this case the target layer became divided into two continuous bands of opposing input preference. The relatively good results achievable with 64 afferent synapses per neuron informed the decision to implement this amount of fan-in on the chip which will be presented in chapter 3.

2.4.2 Receptive field centres

Considering the effect of the algorithm on mean *AD*, in case 2 mean $AD_{fin-weight}$ is significantly increased c.f. mean $AD_{fin-weight-shuf}$. In case 1 the corresponding change is not significant. In case 1 the drop in mean $AD_{fin-con}$ c.f. mean $AD_{fin-con-shuf}$ is not significant.

The basic action of weight-independent STDP on a set of incoming synapses for a single neuron is to deliver a bimodal weight distribution [Song and Abbott, 2001]. Where there are input correlations these cause the more correlated inputs to be maximised and the less- or un-correlated inputs to be minimised. The effect of both the input correlations and the local excitatory lateral synapses on each individual incoming connection field then should be to cause a patch of neighbouring synapses to become potentiated and for outliers from this patch to be depressed. This could be more simply thought of as choosing a subset of the synapses. Ideally the subset which is chosen will be the subset which is most tightly clustered, in other words, the subset with the lowest variance. This is also true if the sample variance measure is used instead of the population variance. It can be proven that for samples drawn from normally distributed data, the sample variance is independent from the sample mean. That is to say, the centre of mass of the tightest cluster is no more likely to be located towards the ideal location than if the same number of synapses were placed randomly according to the initial distribution. This is only true for samples drawn from a normal distribution; however, in this case the population of synapses from which the sample is drawn has a finite number of data points and therefore only approximates to a normal distribution. The effect of this is to introduce an additional error in *AD*, such that the final value of *AD* is likely to be slightly higher than if the corresponding number

of synapses is distributed according to the initial distribution. This increase though is only slight, such that it does not necessarily pass the significance tests (as the number of afferent synapses for a neuron increases, this increase should tend to zero). Rewiring cannot be expected to do anything to eliminate this error since it can only enhance existing trends.

The result of the learning rule is not to drive the preferred location towards the ideal. This suggests that, given the assumptions in this model, any improvement in the quality of topography as judged by reduction in the distance of preferred locations from ideal locations (as opposed to reduction in the spread of receptive fields) observed in the development of maps in biology is likely to be due to such activity-independent mechanisms as exist.

The final results in this respect are not inconsistent with biological topographic maps. In V1 receptive field centres of cells within a single cortical column have been found to be distributed randomly with a standard deviation which is comparable to the areas of the receptive fields (thus, in the terms used here, comparable to the spreads of the receptive fields, as judged by the standard deviation) [Hubel and Wiesel, 1968, Creutzfeldt et al., 1974].

2.4.3 The role of input correlations

Considering the role of input correlations, in case 3 (rewiring but no input correlations) mean $\sigma_{aff-fin-con} = 2.17$, vs 2.31 for mean $\sigma_{aff-fin-con-shuf}$; this is significant. Mean $\sigma_{aff-fin-weight} = 1.95$ vs 1.99 for mean $\sigma_{aff-fin-weight-shuf}$; this is significant.

The slight drop in mean $\sigma_{aff-fin-weight}$ is a sufficient cue to drive the narrowing of the incoming connection fields, as evidenced by the drop in mean $\sigma_{aff-fin-con}$. It was shown [Linsker, 1986b, Miikkulainen et al., 2005] that functional architecture can form in the absence of any input except uncorrelated random noise. Here a complementary result can be seen in which receptive field spread reduces without input correlations. However, the mechanisms are different, since Linsker's result was described in terms which require lateral inhibition, which is not present in this model. Rather there are two possible mechanisms. Firstly, a target neuron may receive two or more synapses from a single input neuron. These synapses will always be correlated and are likely to reinforce each other, becoming more likely to be amongst the potentiated neurons and thus narrowing the spatial variance. Secondly, a spike from a single input neuron will excite a given target neuron and any other of its neighbours which have a synapse from that input. Thus the neuron will

also tend to receive some excitation from lateral connections because of that spike. The smaller range of $\sigma_{form-lateral}$ should selectively enhance the input from a smaller range of locations, leading to a reduction in variance.

2.4.4 Limitations of the model

Although a reduction in the spread of receptive fields is seen, it is on a small scale. It is possible that the small numbers of synapses limit the effect. Though figures are not available for the results of Song and Abbott [2001, figure 6], the variance reduction achieved appears to be larger (with 200 feed-forward connections cf. ≈ 16 in these simulations, albeit in a 1D mapping scenario). They also noted that in their simulations, altering the range of the spatial correlations could affect the tightness of the final projection. Quantitative information on the variance reduction seen in biology is sparse. However, whilst axonal arbor spread reduces quite substantially in rodents [McLaughlin et al., 2003b] and chicks, in fish and amphibians it does not necessarily reduce in absolute size, only relatively to the area of the tectum, which expands during development [McLaughlin et al., 2003a]. Thus, it may be counter-productive to judge the observed variance reduction quantitatively in a model with intended generality, but rather sufficient simply to observe that such an effect is possible. Nevertheless further work would be needed to fully characterise the performance of the variance reduction phenomenon under different parameters.

A possible way this model could be extended is to allow that axon branching should be guided by the existence of axons, such that an input neuron is more likely to form synapses with target cells which are close to target cells which it is already innervating. This might be expected to model axonal arbor development more accurately. In addition there is no mechanism in this model for the preferential sprouting of synapses due to potentiation [Toni et al., 1999] – a complementary way in which the two types of plasticity could interact.

As is common in this field, point neurons have been modelled, that is to say, there has been no consideration of the spatial and temporal summation and filtering performed by transmission of post-synaptic potentials along dendritic trees. Such considerations are likely to be important for a complete understanding of map formation, especially since it is known that temporal learning windows can be different at different locations on the dendritic tree [Kampa et al., 2007].

The development of a mapping from the start has not been simulated, nor has the growth of areas been addressed. Rather, this model assumes a starting time at some point during development, in order to assess the effect of the proposed learning rules. This is reasonable, as it mimics the progress from an initially diffuse mapping towards a final mapping, which is thought to be at least partly dependent on activity [Simon et al., 1992]. Nevertheless, this focus limits the scope of the model. Additionally, as noted previously, initialising weights at their maximum is unlikely to reflect the reality of synapse development.

In order to model both the growth of areas and the effects of lesions on redevelopment as discussed in section 2.1.5.1, a promising possibility would be to combine the mechanisms used in this model with the activity-independent process described by Willshaw [2006]. This would involve replacing the probabilities of synapses forming which are currently based on distances from fixed ideal locations, with probabilities which Willshaw modelled as synaptic strengths but which represent affinity of an axon towards a particular target location. These affinities change according induced levels of ephrin ligands, which in turn change according to a developmental rule which can allow for growing areas and for abnormalities of the types introduced by lesion studies. More generally, the probability of synapse formation is a promising place in which to intervene within this model in order to somehow capture the aforementioned effects. potentially yielding a model where the topography could develop and the areas themselves could grow, which could replicate compression and expansion studies etc, and which in addition could allow for the formation of functional architecture such as ocular dominance, based on statistical structure in input spike trains.

2.5 Conclusions

A model of topographic development has been presented which includes both weight and wiring plasticity. There are three key assumptions: (a) synapses preferentially form in locations to which their axons are guided, (b) weights of dendritic synapses of a neuron are modified according to a competitive Hebbian learning rule, and (c) weaker synapses are more likely to be eliminated. In order to instantiate the model, more assumptions have been made, the main one being that the weight-change mechanism is a form of spike-timing-dependent plasticity.

It has been found that whilst spatially correlated inputs help to create patterns of synaptic

weights which favour narrower projections, spatial correlations are not necessary for some reduction of variance to occur. A weight-change mechanism and a rewiring mechanism can work together such that the rewiring mechanism acts to embed patterns of synaptic strengths in the network topology; this is as one would expect, though it has not been demonstrated quantitatively before. The accuracy of preferred locations for target neurons will not necessarily improve when synapses are initially distributed around ideal locations. The division of mapping quality into the quantities of mean σ_{aff} and mean *AD* is a useful means for investigating these effects, and a method of applying statistical significance tests has been demonstrated which avoids possible biases in order to extract highly significant effects from small-scale simulations.

Chapter 3

Silicon neuron and synapse

3.1 Introduction

In this chapter the literature on the field of neuromorphic Very-Large-Scale Integration (VLSI) is reviewed, focusing on neuron and synapse implementations, including implementations of Spike-Timing-Dependent Plasticity (STDP). Neuron and synapse circuits are presented. Novel aspects of these circuits include: (a) very-low-current design technique applied to existing design for STDP circuit, resulting in weights which retain traces of their learnt values over tens of seconds; (b) the use of MOSCAP non-linearities to offset other non-linearities, resulting in broadly linear behaviours over wide voltage ranges, with application to linear synaptic integration and other problems; and (c) a novel way of introducing weight-dependence into an STDP implementation. The performance of these circuits is characterised, based on chip results; the effects of mismatch are measured; and the extent of the circuits' ability to perform equivalently to the neurons and synapses used in the computational model in chapter 2 is discussed.

3.2 Literature review

3.2.1 Overview

Neuromorphic engineering is the discipline of creating integrated electronic circuits which mimic neural computation in biological nervous systems, both to inform computational

neuroscience and in pursuit of superior engineering solutions for classes of problems where biology currently outperforms artificial devices [Sarpeshkar, 2006]. Hardware which implements neural computation has a number of potential benefits such as low-power computation and real-time control systems, though much of this potential has not yet been realised. There is particular potential in the creation of implantable devices to interface with biological nervous systems damaged through injury or disease [Vogelstein, 2007].

3.2.2 Neuron models; the analogue approach

Electronic models of nerves cells have a long history. Lapicque originally modelled nerve membrane as a capacitor in 1907 [as cited by Brunel and van Rossum, 2007]. This laid a foundation for the integrate and fire model as it is now known, which appeared from the 1960s [e.g. Stein, 1965], in which synaptic currents are integrated on a capacitor representing a membrane, until a threshold voltage is crossed, at which point a spiking event is triggered and the membrane capacitor voltage is reset to a resting level. Hopfield [1982] modelled recurrent neural networks at a more abstract level (though in a way which could be readily implemented in integrated circuits), catalysing the field of artificial neural networks. More recently, a large body of work has investigated the plausibility and benefits of implementing electronic models of nerve cells and neural networks in CMOS (a form of integrated silicon technology).

It is possible to model the electrical behaviour of nerve cells at many different levels of detail. The perceptron [Rosenblatt, 1958] stands at one end of the spectrum, being a simple embodiment of a non-linear summation function of its inputs. At the other end stands the Hodgkin and Huxley [1952] model, which describes currents flowing through a portion of neural membrane (a “compartment”) due to various different populations of membrane-bound ion channels. Many such compartments can then be linked together to create arbitrarily complex neuronal morphologies. Recently a neural model of intermediate complexity, called the simple model, has gained popularity [Izhikevich, 2003].

CMOS implementations of such diverse neural models have been proposed, for example the integrate-and-fire neuron [Mead, 1989] (see figure 3.1 for this example), Hodgkin-and-Huxley-like neural dynamics [Mahowald and Douglas, 1991, Simoni et al., 2004], the perceptron [Aunet et al., 2004], a neuron with similar properties to the simple model [Wi-

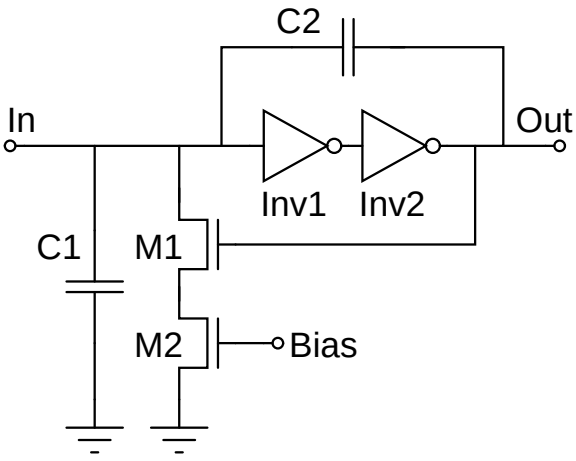


Figure 3.1: Integrate-and-fire neuron, modified from Mead [1989, p. 189], who called it an Axon-Hillock circuit. Currents from synapses (not shown) are sourced on to the *In* node and charge accumulates on capacitor *C1*. When the voltage reaches the threshold of the inverter Inv1, the *Out* node, which represents the voltage of the neuron's membrane, starts to rise. This raises the *In* node further, due to capacitive coupling through C2. *Out* therefore quickly rises to *Vdd* (a global high voltage node which provides power to the inverters) representing a spike. This switches on transistor M1, and the charge on C1 drains away through M1-M2 at a rate set by *Bias*; this implements an absolute refractory period, in which the neuron cannot produce another spike. When the threshold of Inv1 is crossed again, there is another fast feedback reaction that resets *In* to a low level, so that the process can repeat.

jekoon and Dudek, 2006], and neurons with dendritic compartments [Elias, 1993]. The integrate-and-fire neuron remains popular with computational neuroscientists and neuromorphic engineers alike [Schultz and Jabri, 1995, Van Schaik, 2001, Indiveri, 2003b]; it is the integrate-and-fire neuron which is used in this thesis, following the model developed in chapter 2.

Common to all these implementations of neural systems is the use of analogue electronics; as the membrane of a neuron can be modelled as a capacitor in parallel with resistors, so circuit implementations often use a physical capacitor, the voltage across which represents the voltage between the interior and exterior of the cell, and currents onto and off from it mediated by resistive elements (typically transistors). In this way the electronic circuit implemented in silicon is physically analogous to the ionic electrical behaviour of a neuron. As such, these approaches stand in contrast to computer simulations of neural systems, in

which numerical representations of quantities such as membrane voltage are stored and manipulated digitally and sequentially in order to solve the differential equations which govern their evolution through time. The choice of suitably simple and compact circuit representations for neurons allows many of them to be fabricated within a single die, so that neural networks can be created. Since all circuitry can be simultaneously active, the activity of a neural network can be simulated in parallel. This holds the potential to simulate neural network behaviour faster than can be achieved in a Von Neumann architecture, since there is not necessarily a bottle-neck in terms of processing capacity as the size of networks increase (though see chapter 4). A complementary body of work has investigated the use of alternative digital architectures such as Field-Programmable Gate Arrays (FPGA) [Glackin et al., 2005], hardware accelerators [Porrmann et al., 2002] and regular processors networked in a massively parallel way [Jin et al., 2008], to partially parallelise the digital implementation of neural networks.

The analogue approach also has potential energy savings with respect to digital simulation of the same systems, especially in designs which make use of the MOSFET's subthreshold region of operation. In this region, drain-source currents vary exponentially with gate-source voltages, potentially allowing wide dynamic ranges of behaviour with currents which can be on the order of nano-amps down to pico-amps or less. Working with analogue circuits, however, typically incurs a cost in accuracy, as identically designed circuits vary in performance, particularly with lower currents; this is referred to as "mismatch". Analogue electrical quantities (i.e. voltage, current, etc) are also sometimes used to represent continuous quantities in the nervous system of a non-electrical nature, for example, concentrations of a neurotrophic factor [Taba and Boahen, 2002]. Analogue circuitry is not ubiquitous within neuromorphic approaches; as spikes can be modelled as discrete events (as in integrate-and-fire models), so they are often implemented as digital signals. The transmission of spiking events and the connecting of neural circuits to create networks is reviewed in chapter 4.

3.2.3 Synapse circuits

Just as there are many ways to model synapses, so there have been a diversity of approaches to their silicon implementation, which will be reviewed here. Aspects of synapse physiology such as weight, conductance increase towards reversal potential, dendritic filtering and temporal profile will be considered.

In systems which transmit spiking events as brief digital pulses, the simplest approach to a synapse circuit might be to use the pulse to instantaneously source current onto the membrane capacitor for the duration of the pulse. However, this ignores many useful properties of a synapse's behaviour. Firstly, a synapse has a weight, such that the same spike arriving at synapses with different weights will generate different currents; this can be implemented by controlling the magnitude of the current by a representation of the synapse weight [Satyanarayana et al., 1992, Fusi et al., 2000, Chicca et al., 2003a]. Secondly, the effect of a spike arriving at a synapse is actually to cause an increase in the permeability of the membrane to a certain type of ion, and thus an increase in conductance towards an associated reversal potential. Modelling this effect precisely would require the use of resistive elements (or their emulation by some means). Thirdly, synapses are distributed over a neuron's dendritic tree and post-synaptic potentials are filtered and combined as they travel towards the soma. Attempts to model these phenomena include Elias and Northmore [1995] and Rasche and Douglas [2001]. Finally, a synaptic current is not instantaneous but rather has a temporal profile, typically rising quickly then decaying over a period of milliseconds. Modelling such temporal dynamics can significantly change the dynamics of the membrane voltage, for example by extending the length of the depolarisation kernel caused by a spike and allowing a greater time window in which co-operation from other spikes can lead to an action potential. One implementation of this, the "reset and discharge synapse", instantaneously charged a capacitor to a given level depending on the weight of the synapse and then linearly drained it, using the subthreshold exponential V_{gs} to I_{ds} relationship of a transistor to convert the resulting negative-going ramp into an exponentially decaying current representing synaptic input [Lazzaro 1994, as cited by Bartolozzi and Indiveri, 2007]. Extending the temporal effect of a synapse in this way makes it possible that another spike will arrive during the period in which the first spike has an effect. In this case, typical computational models would sum the effects of these spikes linearly; however, the aforementioned circuit cannot achieve this, as the arrival of a spike resets the capacitor, entirely overwriting the output of the circuit with the effect of the most recent spike. A series of modifications to this approach ("linear charge-and-discharge" synapse, "current-mirror integrator", "log-domain integrator" and "differential-pair integrator"; reviewed by Bartolozzi and Indiveri, 2007) have ultimately allowed the linear integration of spikes. However, solutions to this problem remain far from perfect, since of the two circuits which achieve linear integration, the log-domain integrator has difficulty sourcing enough current for practical purposes whilst the differential-pair integrator only integrates

linearly in a steady state condition — the first few spikes in any burst have a disproportionately large effect. A novel approach to linear synaptic integration is presented below in section 3.5.3.

3.2.3.1 Synaptic weight change

Synaptic weight plasticity is a fundamental element of adaptability and memory. As this has been modelled by computational neuroscientists, so it has been implemented in VLSI. For example, there are implementations of short term depression [Rasche and Hahnloser, 2001, Chicca et al., 2003b] and facilitation [Liu, 2003]; STDP [Hafliger et al., 1997, Gordon and Hasler, 2002, Indiveri, 2003a, Bofill-i Petit and Murray, 2004, Koickal et al., 2007]; bimodal probabilistic STDP [Arthur and Boahen, 2005]; bimodal probabilistic plasticity based on membrane voltage level, following a model by Brader et al. [2007], [Fusi et al., 2000, Badoni et al., 2006]; and a model of plasticity based on intracellular calcium levels by Shouval et al. [2002], [Rachmuth and Poon, 2003].

As discussed in chapter 2, STDP learning rules can be weight-dependent or -independent. The implementation of [Koickal et al., 2007] is weight-independent, with the temporal learning windows modelled as a pair of decaying exponentials, as did Song et al. [2000]. The weight-independent implementation of Indiveri et al. [2006] (though see section 3.4.4.1) is more compact, though with a temporal learning window which is less easy to relate to biological data such as from Zhang et al. [1998]. The implementation of Hafliger et al. [1997] implements only the potentiation aspect of STDP, does so in a weight-dependent manner and only performs weight updates based on single pairs of pre- and post-synaptic spikes. The operation of STDP on single pairs of spikes only, is a form of the learning rule which has since been incorporated in a computational model [Izhikevich et al., 2004]; other implementations mentioned allow all possible pre-synaptic to post-synaptic spike pairs to contribute to plasticity. The implementation of Bofill-i Petit and Murray [2004] has weight-dependent potentiation and weight-independent depression, as observed by Bi and Poo [1998] [though see Morrison et al., 2007]. Features of this implementation are that the degree of weight-dependence is tunable and that it is relatively compact with exponentially decaying temporal learning windows. Section 3.4.4 presents an alternative approach to a compact implementation of weight-dependent STDP with exponentially decaying learning windows and all-to-all spike pairing.

3.2.3.2 Synaptic weight stability

A major obstacle to the implementation of weight plasticity is that for long term plasticity there needs to be a way of holding a continuously-valued variable at a constant level. Synaptic weight is often implemented as a voltage across a capacitor [Chicca et al., 2003b, Bofill-i Petit and Murray, 2004]. However this is subject to leakage such that any learnt value will be lost over a period of milliseconds or seconds.

One solution to this problem focuses on the observation that weight-independent STDP tends to produce a bimodal distribution of weights such that weights are either almost completely potentiated or almost completely depressed if averaged over time; also that certain biological studies are suggestive of synaptic plasticity with an all-or-nothing nature [Petersen et al., 1998]. Given this, Indiveri [2003a] used weak positive feedback from an amplifier to drive the weight value either upwards or downwards away from a central threshold, yielding a distribution of weights which are bi-stable; they can take any value in the short term but in the long term they have only two stable states, potentiated or depressed. Brader et al. [2007] then introduced a computational model of weight plasticity which was explicitly bi-stable; this has been implemented [Fusi et al., 2000, Badoni et al., 2006]. Arthur and Boahen [2005] went further, modelling weights as having only two states even in the short-term and using a static ram element to store this value, which could then be switched if an accumulation of either potentiating or depressing events surpassed a certain threshold. In a complementary approach, an analogue memory element was created with many stable states [Hafliger and Riis, 2003], though the implementation is bulky, requiring an amplifying element for each stable state.

An alternative solution is to use floating gate technology. A piece of semiconductor which is completely isolated by oxide insulator can hold its charge indefinitely, yet be charged or discharged by electron tunnelling or hot-electron injection. If such an isolated node is then used as the gate of a transistor, it is referred to as a floating gate and its voltage can be repeatedly used to affect system behaviour. If synapse weights are stored as charge on floating gates then there is the additional advantage that any learnt patterns of weights is not erased in the absence of a power supply. [Hasler et al., 1995, Gordon and Hasler, 2002]. Floating gate technology has higher technical requirements, for example, higher voltage power rails or the generation of higher voltages with charge pumps. More problematically, many synaptic learning algorithms make frequent weight updates, often as frequently as spiking events, which reduces the lifetime of floating gate transistors due to

eventual dielectric breakdown caused by hot-electron injection.

Another approach is to digitise the weight with a chosen level of accuracy. It can then be stored in standard computer memory (which may physically consist of capacitors, static RAM cells or floating gates depending on the technology). Such a digital memory has previously been used to periodically refresh local analogue storage of weights on capacitors [Eberhardt et al., 1989, Satyanarayana et al., 1992]. More recently it has been used advantageously to allow synapses to become virtual rather than physical devices, so that one physical device can act as all the incoming synapses for a neuron, by sequentially receiving the weight information for each incoming spike and acting accordingly [Goldberg et al., 2001, Vogelstein et al., 2007]. This achieves a possible saving in area but with higher communication overheads, which will be discussed in chapter 4.

In this work, each synapse has a physical instantiation with locally stored weight, for lower communication overheads. Further implications of this decision are discussed in chapter 5. One implication of this decision is that the total area of a neuron scales linearly with the number of incoming synapses. As synaptic fan-in increases, the area of the neuron becomes increasingly dominated by the synapse circuitry. Therefore whilst efforts to miniaturise the circuitry which performs the central functions of the neuron are commendable, the contribution of neuron circuitry to overall chip area becomes increasingly irrelevant as fan-in increases. Given this, the implementation of processes central to the neuron have been allowed to be reasonably expansive, whereas synapse circuitry is relatively compact.

Regarding the aforementioned approach of introducing bistability, Bofill-i Petit [2005] argued that (weight-independent) STDP is inherently bistable and that this fact should be utilised to bypass the problem of volatile weight storage, rather than, for example, introducing explicit bistability. Such an approach is viable in situations where the input which leads to learnt patterns of weights is continuous. In the absence of such input, weight distributions will converge due to leakage currents. This thesis, however, realises a complementary approach: by allowing weight distributions (which can change rapidly and are stored in volatile memory on capacitors) to influence network topology (which changes slowly and is stored in stable memory elements), features learnt from input can continue to influence the behaviour of a network long after the original input has occurred and the immediate memory trace has faded. Notwithstanding this, steps are also taken to reduce leakage currents so as to maximise the lifetimes of memory traces stored on

capacitors.

3.2.4 Use of clocks

Neuromorphic circuitry often carries out analogue computations in a continuous manner, for example, the continuous evolution of membrane voltage through the integration of synaptic currents. Where discrete spiking events occur, a significant body of work has investigated their transmission in an asynchronous fashion (see chapter 4). Furthermore separate neural circuits which exist within a chip can all carry out their processes in parallel, rather than being constrained to operate sequentially. Thus, neuromorphic circuitry differs from standard computing architectures in that there is not necessarily any need for a central clock to synchronise processing. Nonetheless, some use of regularly timed clocks has been beneficial, and will be reviewed here.

Elias and Northmore [1995] implemented compartmental models of dendritic trees. Each compartment was modelled as a capacitor with resistances between compartments and from each compartment towards a resting potential. However the resistances of biological membrane can be on the order of giga-ohms. Such large resistances can be achieved by MOSFETs but at the expensive of linearity. The alternative used by Elias and Northmore was to implement a resistance as a switched capacitor (these are explained in the text of figure 3.2); by using small capacitances and low clock frequencies, they achieved biologically realistic timescales for the propagation of charge along their artificial dendrites. It was also used by Glover et al. [1998] and Bofill-i Petit [2005] to create membrane leak conductances in single-compartment integrate-and-fire neurons. Vogelstein et al. [2007] used switched capacitors in a slightly different way, as part of a quantised synapse circuit. By opening or closing the circuit to individual members of a set of differently sized capacitors incorporated within a switched capacitor arrangement, variably sized packets of charge could be delivered to a neuron. This usage does not require a regular clock, since the switching cycle occurs asynchronously whenever a spike is transmitted.

Switched capacitors are used in multiple situations in the circuitry presented in this thesis. In particular, a novel circuit for implementing membrane currents is shown in section 3.4.3. In general, the benefits of switched capacitor circuits are the ability to implement constant conductances; the controllability of these conductances over a very wide range; and good matching between components, due to the relatively good matching of capac-

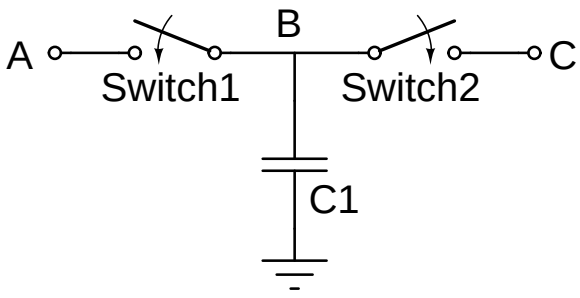


Figure 3.2: Switched capacitor. The switches (normally transistors or transmission gates) are controlled by two non-overlapping clock signals. On the first clock signal, switch 1 closes and charge is shared between nodes A and B, so that depending on the voltage of A, some charge may be stored on capacitor C1. Then switch 1 opens and 2 closes, and charge is shared with node C. If there is a voltage difference between A and C then this cycle causes a packet of charge to pass between them whose magnitude depends both on the capacitance of C1 and on the voltage difference. The higher the frequency of the clock signal, the more of these packets of charge are allowed to pass. For a large number of clock cycles, the behaviour of the switched capacitor approximates a resistor whose resistance is defined as $1/fC$, where f is the frequency of the clocks and C is the capacitance of the capacitor.

itors in CMOS cf. transistors or resistors. The disadvantages are: a higher energy requirement due to the energy cost of implementing global clock signals; the possibility of the synchronicity introduced by clocked processes creating synchronicity in spike trains and qualitatively affecting network behaviour; and the additional possibility of unintended behaviour due to parasitic capacitive coupling between the clocks and other nodes. These problems can be balanced against each other: the higher the frequency of a clocked process, the higher the energy cost but the less any synchronicity introduced is likely to affect network behaviour, since it is a closer approximation to continuous operation. In this project then, a limited quantisation of time is accepted for some processes.

3.3 Justification for neuromorphic implementation

The general goals of the discipline of neuromorphic engineering have been given above in section 3.2.1. The combination of weight plasticity and synaptic rewiring is a timely concern [Chklovskii et al., 2004, Jun and Jin, 2007]. In particular, embedding learnt synaptic weight distributions in network topologies by synaptic rewiring has the potential to resolve the ongoing problem of how to store volatile memories in neuromorphic systems. Topographic mapping between brain areas is widespread in the vertebrate brain, suggesting that it plays an important role. There have been attempts by neuromorphic engineers to model these processes (which will be reviewed in chapter 5) but there is plenty of work to do. In particular, there have been no neuromorphic systems which explicitly combine activity-dependent and -independent processes. Furthermore, whilst STDP and other learning rules have been implemented, few studies have investigated the behaviour of such implementations at the network level [though see Taba and Boahen, 2002, Chicca et al., 2003a]. Meanwhile Morrison et al. [2008] noted that large scale network simulations of STDP are few and far between, partly due to the computationally intensive simulation that is required. Perhaps then, neuromorphic VLSI could offer a platform to computational neuroscientists for the efficient simulation of STDP or other event-based learning rules.

3.4 Circuit designs

Where simulation results are indicated, they have been generated by Cadence software, normally running the Spectre simulator and based, where appropriate, on a technology

library for the AMS C35B4 process. Note that *Gnd* refers to "ground", a global node against which all voltages are measured, whereas *Vdd* refers to a global node which acts as a source of high voltage. In the process used there is 3.3V from *Vdd* to *Gnd*.

3.4.1 Neuron Circuit

The circuitry which implements the threshold, spike and reset mechanism of the integrate-and-fire neurons is slightly modified from that described in Indiveri [2003b] and is explained in appendix A.

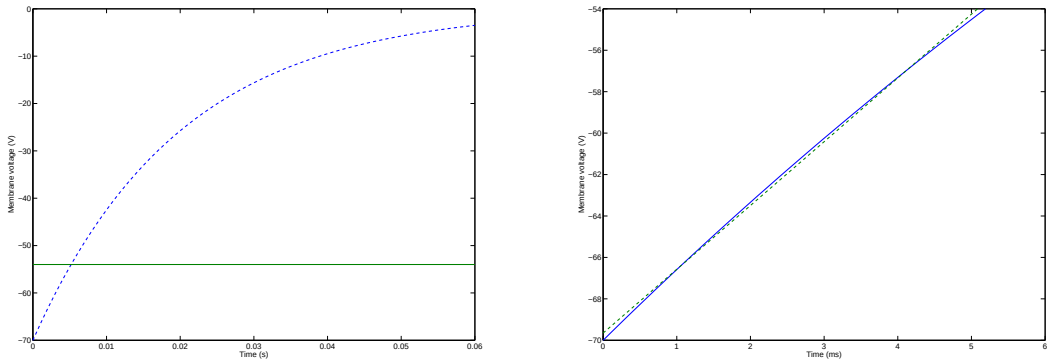
3.4.2 Synaptic conductance

3.4.2.1 Equivalence of synaptic conductance and current

In the neural model in Song et al. [2000] adopted in chapter 2, the influence of excitatory synapses is included as a kernel of increased conductance towards an excitatory reversal potential. Figure 3.3(a) shows the curved charging profile that results from a (fixed) conductance towards the excitatory reversal potential. Although this profile is non-linear, only the section of the curve below the threshold voltage is used, since above this level, the membrane is reset. The region of the curve that needs to be implemented can be closely approximated by a straight line, as demonstrated in figure 3.3(b). A straight line represents the charging profile which would result from a constant current onto the membrane, irrespective of membrane voltage. Therefore to simplify the implementation of excitatory conductance, it has been assumed that excitatory synaptic currents are not dependent on the membrane potential, an assumption which is justifiable when $E_{ex} \gg V_{thr}$. Consequently, *synaptic conductance* is referred to synonymously with *synaptic current*. Similar modelling studies of STDP to the one on which the present model is based use explicitly current-based synapses [Morrison et al., 2007].

3.4.2.2 Integration of increments to synaptic conductance

Figure 3.4 shows the circuitry which maintains a voltage which represents synaptic conductance. Transistors M1-M2 exist in the synapse; there are therefore multiple pairs of these transistors, one for each synapse. The synapse stores its weight value as a voltage on



(a) (b)

Figure 3.3: Ideal membrane charging profile. (a) The dotted line shows the membrane capacitor charging from resting potential (-70mV) towards the excitatory reversal potential (0V) through a fixed conductance; the solid line shows the threshold voltage (-54mV). (b) The solid line shows the membrane capacitor charging from resting potential (-70mV) towards the excitatory reversal potential (0V) through a fixed conductance, up to the threshold voltage (-54mV); the dotted line gives the best linear fit to this curve.

a capacitor. This weight value is used to modulate the increase of synaptic conductance upon incoming spikes. The weight value, *nWeight*, is prefixed "*n*" because it is negatively defined i.e. a low voltage corresponds to a strong synapse. Upon a negatively-going timed digital pulse (*nPrePulseSynCond*) from the address-event decoder, with a typical duration of $\approx 10ns$, current is sourced through M1-M2 into the *SynCond* node. The charge on this node represents the amount of synaptic conductance. The current through M2 due to *nWeight* would ideally be linearly related to *nWeight*. Whilst it is not entirely linear, the quadratic function of M2 in the saturated strong inversion region is partially offset by the source voltage of M2 which varies as a function of *nWeight* with a leading squared term. The effect of this can be seen in figure 3.5, where the V-I function is shown in comparison to a V-I curve for a single pMOSFET. The function is almost linear over most of the range of *nWeight*.

Between pulses, the leakage current through M1-M2 is held at sub-pA levels by a positive V_{gs}, i.e. *nPrePulseSynCond* rests at *Vdd* whilst *nWeightHigh* is held below *Vdd*, in a manner suggested by Linares-Barranco and Serrano-Gotarredona [2003]. This allows many synapses to be connected to the *SynCond* node (64 in the fabricated chip), without

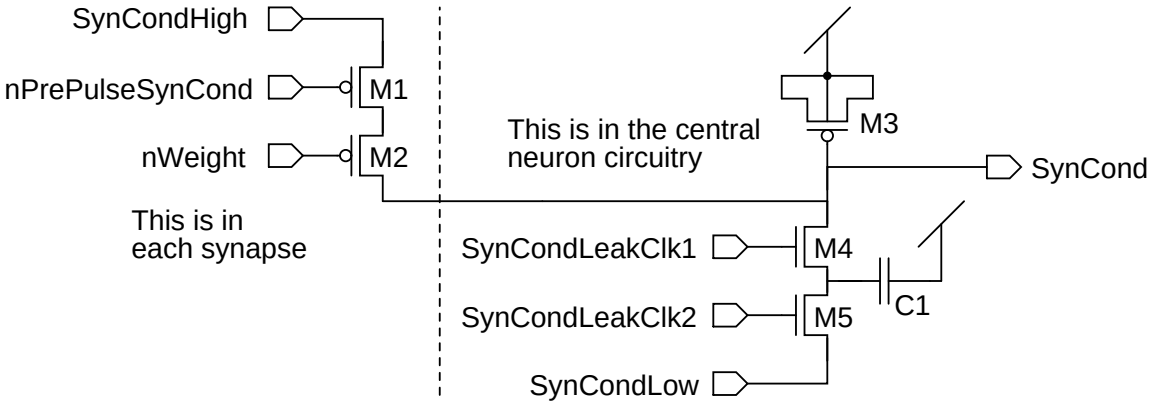


Figure 3.4: Circuitry for synaptic conductance dynamics. Upon a timed pulse (*nPrePulseSynCond*) from the address-event decoder, with a typical duration of $\approx 10ns$, current is sourced through M1-M2 into the *SynCond* node. Between pulses, the leakage current through M1-M2 is held at sub-pA levels by a positive V_{gs}, i.e. *nPrePulseSynCond* rests at *Vdd* whilst *SynCondHigh* is held below *Vdd*. A regular clock signal *SynCondLeakClk1-2* implements a conductance from *SynCond* to *SynCondLow* through switched capacitor arrangement M4-M5 and C1.

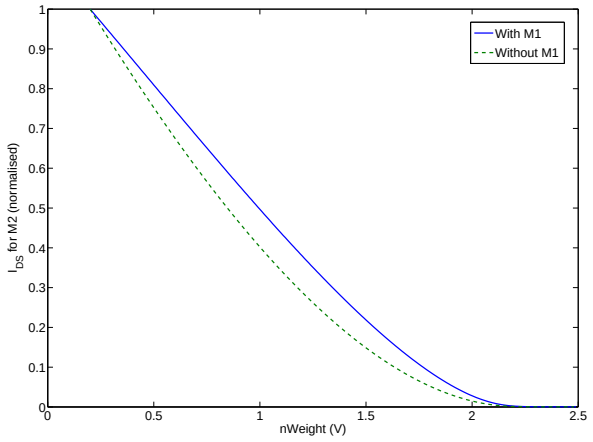


Figure 3.5: Current through transistor M2 as a function of its gate voltage *nWeight.* Simulated currents through transistor M2 with its drain connected to *Gnd*. Currents are normalised to aid comparison. Solid line: M1-M2 to configured as in 3.4 with the source of M1 connected to *nWeightHigh*. Dashed line: transistor M2 with its source connected directly to *nWeightHigh*.

creating a large current into it.

In order for the effect of synaptic inputs to sum linearly, in accordance with the model, it is necessary for the effect of each pulse from a synapse to raise the value of *SynCond* by the same amount, irrespective of the value of *SynCond*. However, the current through M1-M2 is also partly dependent on *SynCond*; as *SynCond* rises, the current through M1-M2 is decreased. Figure 3.6 shows a simulation of a pair of transistors configured as M1-M2 in figure 3.4, continuously charging an ideal capacitor. As the voltage on the capacitor increases, M2 goes out of saturation and thereafter the current supplied reduces resulting in a charging profile which is initially straight (to first order approximation) but then curves. The right graph shows that when the synapse is weak, i.e. when *nWeight* is high, linearity is maintained throughout the charging profile, because M2 does not go out of saturation. However the behaviour of *SynCond* is dominated by strong synapses rather than weak ones so it is more important to address the non-linearity in the case of strong synapses.

The problem described above is partially offset by the use of pMOSFET M3 as a capacitor, i.e. as a "MOSCAP" device. The gate capacitance of a transistor is composed of gate-source, gate-drain and gate-bulk capacitances (C_{GS}, C_{GD} and C_{GB} respectively). Each of these vary non-linearly with V_{GS} with transitions as the transistor moves between different regions of operation (off, saturation and non-saturation); a model of this behaviour can be found in Allen and Holberg [2002, pp. 79-86]. Whilst a MOSCAP device is never saturated, it can pass between its off and non-saturation regions. Figure 3.7 shows how capacitance for a pMOSCAP changes with V_{GS}, as well as the voltage curve which results from charging with a constant current. Intuitively, if the synapses supply a current which decreases with the voltage of *SynCond* and if the voltage of a pMOSCAP will increase more sharply when charged when the voltage of SynCond is high, then these effects should cancel each other out to some extent, to deliver a charging profile which deviates less from a linear profile than for either constant (ideal) charging of a MOSCAP, or for charging through M1-M2 of an ideal capacitor. The resulting simulated charging profile can be seen in figure 3.8, against a linear fit. The effect of the MOSCAP is to extend the range of the *SynCond* signal which charges approximately linearly.

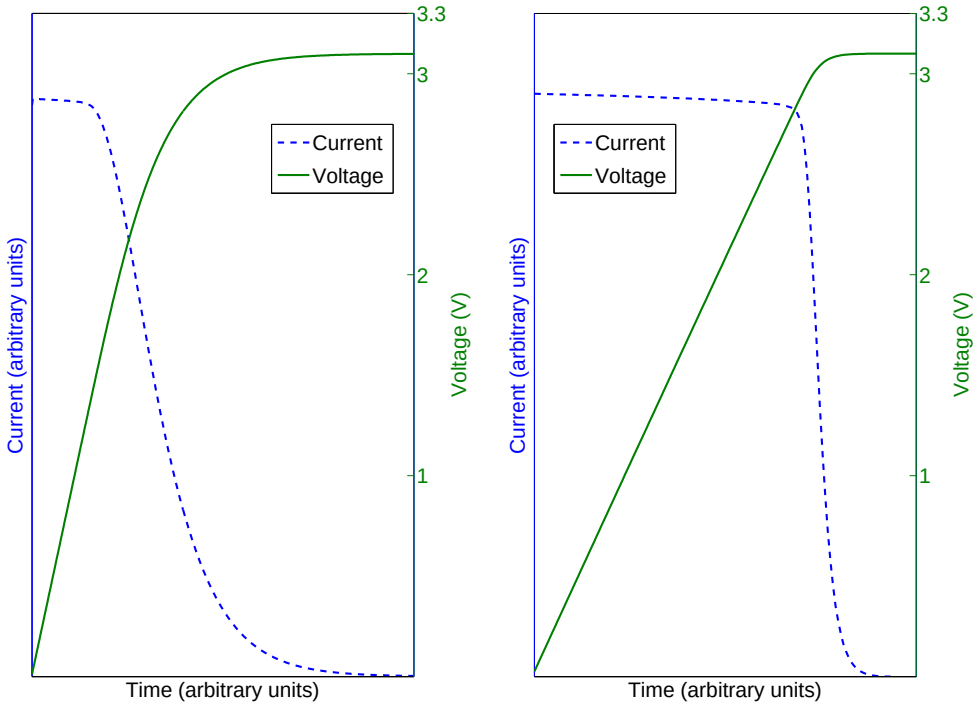


Figure 3.6: Ideal capacitor charging. Simulation of an ideal capacitor being continuously charged from *Gnd* by a pair of transistors configured as M1-M2. Left: *nWeight* = 0.2V (i.e. a strong synapse); right: *nWeight* = 2V (i.e. a weak synapse).

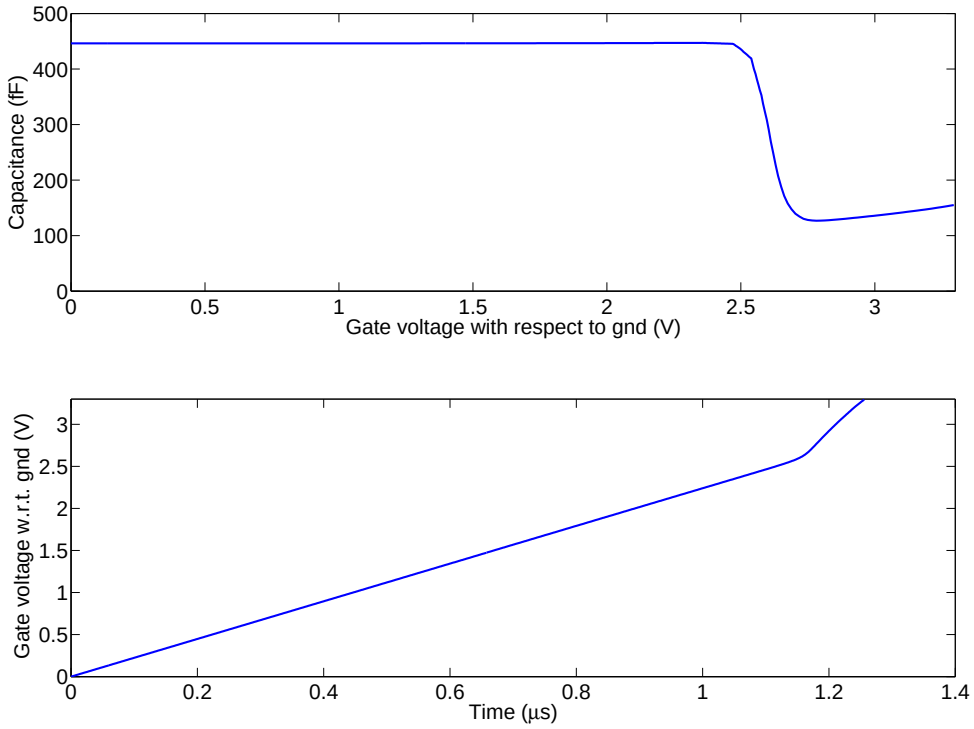


Figure 3.7: MOSCAP capacitance profile. Top: how the capacitance of a pMOSFET (configured as M3 in figure 3.4 with dimensions $10x10\mu$) varies with its gate voltage (simulated). Bottom: the voltage on the gate of the pMOSFET with respect to *Gnd* as it is charged with a constant current ($1\mu A$).

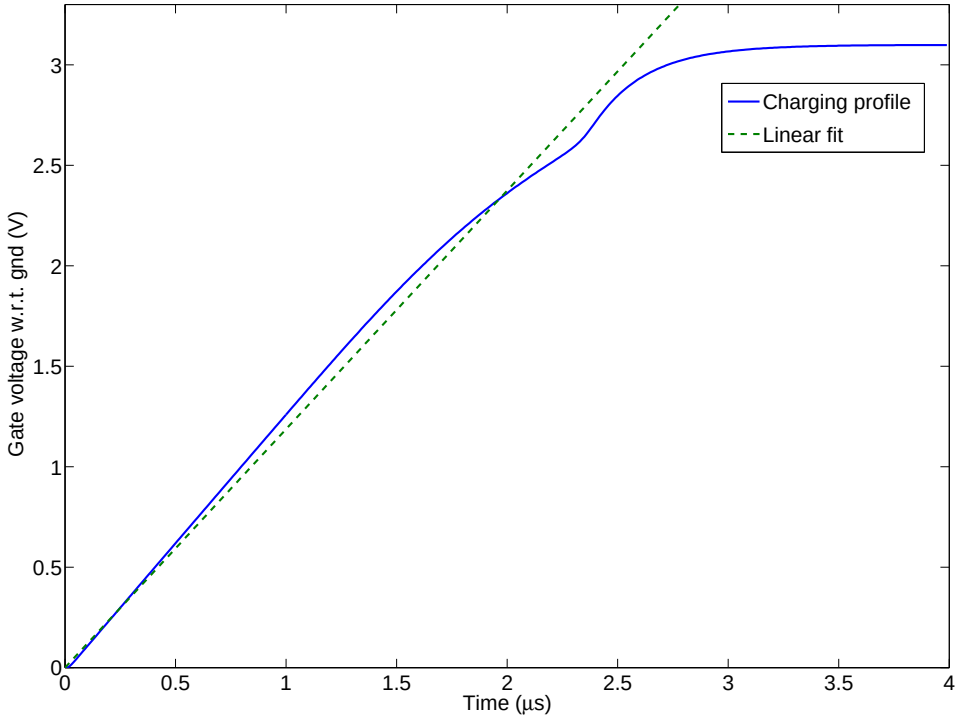


Figure 3.8: SynCond charging simulation. The solid line shows a pMOSFET, configured as M3 in figure 3.4 with dimensions $10x10\mu m$, being continuously charged from *Gnd* by a pair of transistors configured as M1-M2, with *nWeight* = 0.2V. The dashed line is the best linear fit for the data up to 3V with the y-intercept fixed at 0V.

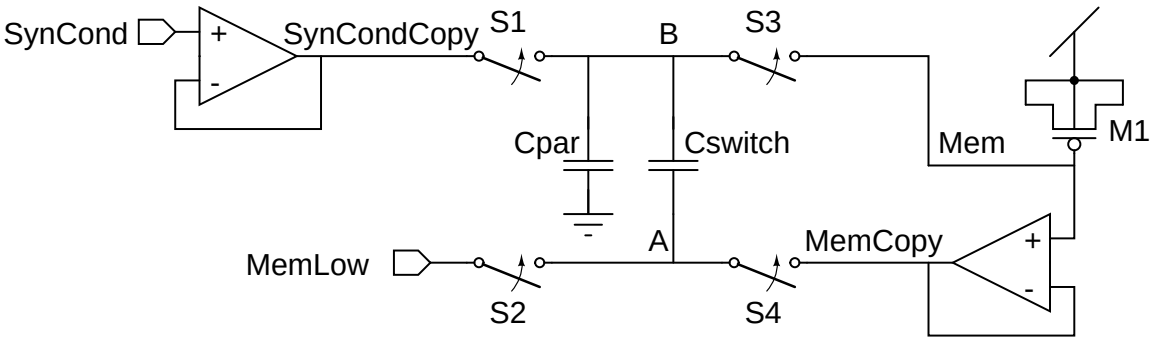


Figure 3.9: Circuitry which creates a synaptic current onto the membrane and a leak conductance towards the resting potential. *Cpar* represents parasitic capacitance on node B. The circuit is explained in the text.

3.4.2.3 Decay of synaptic conductance

The *SynCond* node is drained away through a resistor implemented as a switched capacitor (M4-M5 and C1), allowing the time constant to be altered over a wide range by controlling the non-overlapping clock signals *SynCondLeakClk1-2*. It drains away towards a low voltage *SynCondLow*, which is defined as the point of zero excitatory membrane conductance.

3.4.3 Membrane currents

The level of *SynCond* is used to create a current onto the membrane (the *Mem* node) which varies linearly with *SynCond* but is independent of *Mem*. This is achieved by the circuit shown in figure 3.9, which simultaneously implements a conductance from *Mem* towards the resting potential of the membrane *MemLow*.

SynCond is buffered with unity gain to create the voltage $V_{SynCondCopy}$, whilst *Mem* is buffered with unity gain to create the voltage $V_{MemCopy}$. Assume that the membrane has previously been charged to a certain level V_{Mem_Init} (below its spiking threshold) so as to observe how this circuit implements a conductance from *Mem* to *MemLow*. Also assume for simplicity that $V_{MemLow} = Gnd$ (in practice it is constrained to equal $V_{SynCondLow}$ and is raised above *Gnd*). Finally, assume that the capacitance of the *Mem* node, C_{Mem}, is a perfect capacitor rather than a MOSCAP device. Switches S1 and S2 close simultaneously, discharging node *A* to *Gnd*, and charging node *B* to $V_{SynCondCopy}$. Then switches S1 and S2 open again. Next, in practical operation, switches S3 and S4 close simultaneously, but

to simplify the understanding of this circuitry, assume that S3 closes first. Note that from node *B* there is an additional capacitor C_{Par} to *Gnd*; this models the parasitic capacitance on that node. When S3 closes, the charge density (thus voltage) on nodes *B* and *Mem* balance, so that:

$$V_B = V_{Mem} = \frac{V_{SynCondCopy}C_{Par} + V_{Mem\_Init}C_{Mem}}{C_{Par} + C_{Mem}} \tag{3.1}$$

Since at this stage the assumption is that S2 and S4 are both open, the capacitance C_{Switch} does not affect the charge balancing above in equation 3.1, and the floating node *A*, previously set to *Gnd*, now goes to:

$$V_A = \frac{V_{SynCondCopy}C_{Par} + V_{Mem\_Init}C_{Mem}}{C_{Par} + C_{Mem}} - V_{SynCondCopy} \tag{3.2}$$

When switch S4 closes, the voltage on node *A* is driven up from the value for V_A given above in equation 3.2, to $V_{MemCopy}$. This causes nodes *B* and *Mem* to rise, so that:

$$V_B \Rightarrow V_B + \left(V_{MemCopy} - V_A\right) \frac{C_{Switch}}{C_{Par} + C_{Switch} + C_{Mem}} \tag{3.3}$$

However, $V_{MemCopy}$ is a buffered copy of V_{Mem}, and as V_{Mem} changes (being linked with V_B through switch S3), so does $V_{MemCopy}$. i.e.:

$$V_{MemCopy} \Rightarrow V_{Mem} \tag{3.4}$$

Thus there is feedback, resulting in a final value for V_{Mem} which can be evaluated by substitution of equations 3.1, 3.2 and 3.4 into equation 3.3, and rearranging. This yields:

$$V_{Mem\_Final} = V_{Mem\_Init}\frac{C_{Mem}}{C_{Par} + C_{Mem}} + V_{SynCondCopy}\frac{C_{Switch} + C_{Par}}{C_{Par} + C_{Mem}} \tag{3.5}$$

It can be seen from equation 3.5 that the effect of a single cycle of the switches is to set V_{Mem} to a proportion of its old value plus a proportion of $V_{SynCondCopy}$. Both of these proportions are based on the relative sizing of the capacitors, including the parasitic capacitance of node *B*. Careful sizing of C_{Mem} and C_{Switch}, bearing in mind the likely value of C_{Par} due to layout, yields a single switched-capacitor-driven process which simultaneously implements both a current into *Mem* linearly related to $V_{SynCond}$, representing an excitatory conductance, and a current out of *Mem* linearly related to V_{Mem}, representing a leak

conductance towards the neuron's resting potential. Previous membrane implementations typically treat these processes separately.

The magnitude of the leak conductance is:

$$g_{Leak} = \frac{f C_{Par} C_{Mem}}{C_{Par} + C_{Mem}}$$

where f is the frequency of the clock. The time constant for decay of the membrane potential is:

$$\tau_{Mem} = \frac{C_{Par} + C_{Mem}}{f C_{Par}}$$

The effects of both V_{Mem} and $V_{SynCondCopy}$ are actually relative to V_{MemLow} rather than *Gnd*, requiring that $V_{MemLow} = V_{SynCondLow}$, as stated previously. Equation 3.5 can be restated as:

$$\begin{aligned} V_{Mem\_Final} &= V_{MemLow} \\ &\quad + (V_{Mem\_Init} - V_{MemLow}) \frac{C_{Mem}}{C_{Par} + C_{Mem}} \\ &\quad + (V_{SynCondCopy} - V_{SynCondLow}) \frac{C_{Switch} + C_{Par}}{C_{Par} + C_{Mem}} \end{aligned}$$

Switch S2 is implemented with an nMOSFET and the others are implemented with transmission gates (an nMOSFET and a pMOSFET in parallel), since they may pass high or low values. The choice of a MOSCAP device M1 to implement capacitance C_{Mem} allows greater capacitance per unit area. Providing that *MemThresh* is set so that *Mem* is always towards *Gnd*, the non-linear capacitance of the MOSCAP device will not affect the profile of *Mem*; if it were non-linear it would in any case be irrelevant to the functioning of the neuron.

The use of a single clock to control excitatory synaptic conductance and membrane leak makes the circuit less flexible than it could be, however an approach to parameterising this circuit which allows the magnitude of these two processes to be varied independently is described in appendix E. This circuit is sensitive to mismatch, since any offset in the amplifier DA2 will be accumulated with each clock cycle, affecting the resting potential. Some evidence of this mismatch can be seen in appendix E. This is a disadvantage which

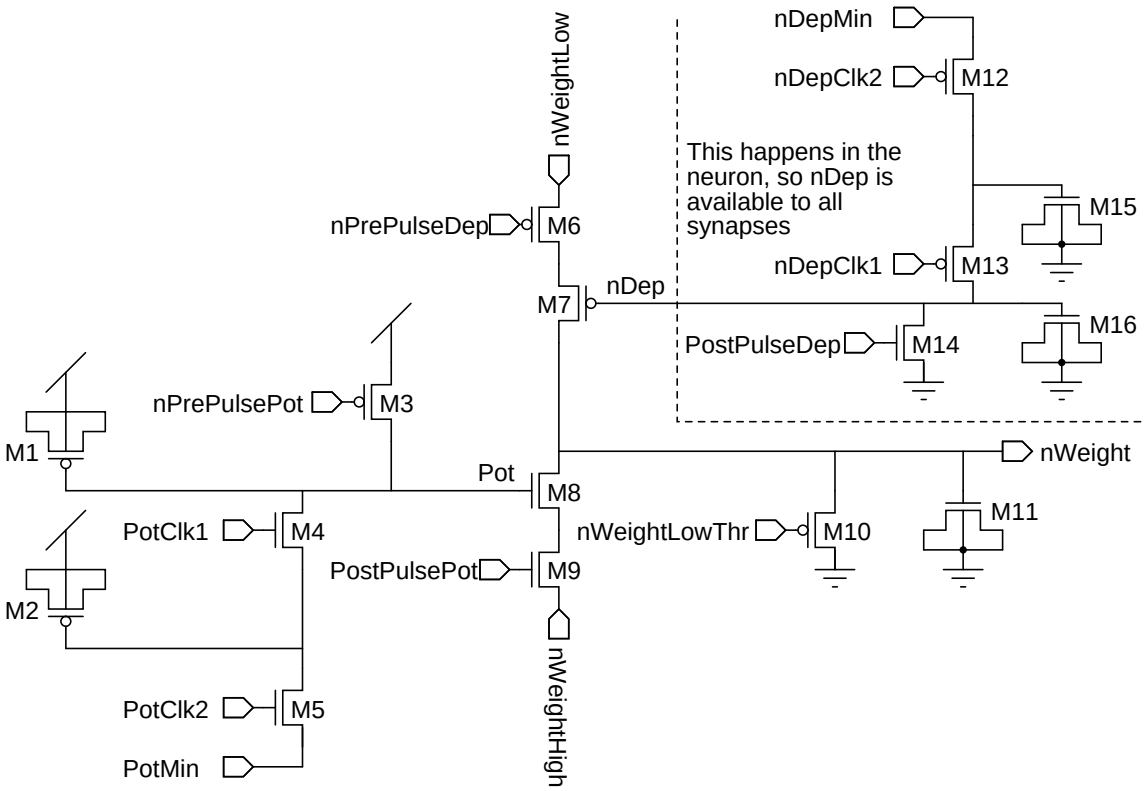


Figure 3.10: Spike-timing-dependent plasticity circuit. Explanation in text.

may outweigh the advantage of compactness, in comparison to a more standard approach such as a parasitic-insensitive switched capacitor integrator [Allen and Holberg, 2002, p. 523].

3.4.4 Spike-Timing-Dependent Plasticity (STDP)

The implementation of STDP (see figure 3.10) has elements of previous implementations by both [Bofill-i Petit and Murray, 2004] and Indiveri et al. [2006]. The weight of the synapse is held on a capacitor and is allowed to vary between reference voltages *nWeightHigh* and *nWeightLow*. As these are raised and lowered from *Gnd* and *Vdd* respectively, the effect is to choke the currents through M6 and M9 to sub-pA levels between pulses, allowing the weight value to remain stable over a relatively long period (see section 3.5.6). In the synapse there is also a voltage *Pot*, maintained on a capacitor, which represents the potential for potentiation. This is raised during a timed pre-synaptic pulse (*nPrePulsePot*) with M3 and then decays away with a time constant typically of tens of milliseconds through switched capacitor M2 and M4-M5. A complementary mechanism

(M12-M16) maintains a potential for depression (*nDep*). However, this value is common to all synapses and so is created once for all of them in the neuron's central circuitry to save space. *Pot* and *nDep* then modulate the changes of weights which then occur during timed pre- and post- synaptic pulses (*nPrePulseDep* and *PostPulsePot*). Note that although occurring simultaneously, *nPrePulsePot* and *nPrePulseDep* are created by different tunable pulse generators, and likewise for *PostPulsePot* and *PostPulseDep*, allowing a high degree of control over the parameters of the potentiation and depression (A_+ and $-A_-$ as they are labelled in chapter 2). Likewise the time constants τ_+ and τ_- can be altered by varying the frequency of the clocks *PotClk1-2* and *nDepClk1-2*.

3.4.4.1 Weight dependence of plasticity

The model in chapter 2 used weight-independent STDP, both to reduce the number of parameters and in keeping with the previous model on which it was based [Song and Abbott, 2001]. Such a learning rule requires upper and lower bounds to be placed on the weight value, so that it cannot grow indefinitely (the lower boundary is usually considered to be a synapse of zero conductance). Where weight is represented as a voltage value, CMOS can provide bounding of the weight for free, since a voltage outside the range defined by the high and low power rails (*Gnd* and *Vdd*) cannot be easily obtained or maintained.

It has been shown, however, that a moderate degree of weight-dependence can improve the ability of STDP to act as a correlation detection mechanism [Gutig et al., 2003]. Furthermore, whilst it has been shown that weight-dependence reduces the duration of a memory trace [Billings and van Rossum, 2008], making learnt weight distributions more volatile, it is expected that the introduction of synaptic rewiring as described in chapter 2 should counter-act this by embedding learnt patterns in the network topology. For these reasons, the implementation described here departs from the model described previously by allowing a degree of weight-dependence in the learning rule.

The circuit presented in figure 3.10, naturally delivers a degree of weight-dependence, due to the fact that currents through M6-M7 and M8-M9 will vary depending on the difference between *nWeight* and *nWeightLow* and *nWeightHigh* respectively (this will also be true for the STDP circuit presented by Indiveri et al. [2006], though this fact has not been reported). To understand the rationale for this circuit it is helpful to consider an alternative circuit, shown in figure 3.11. The basic arrangement of pairs of transistors to pull *nWeight*

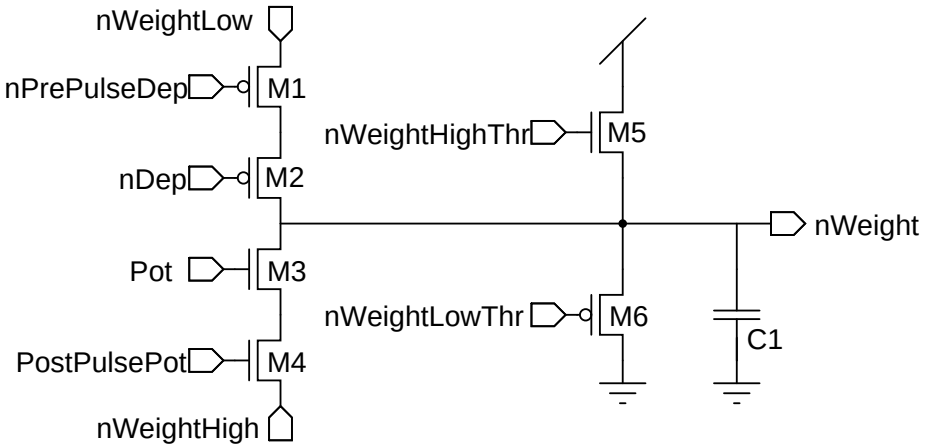


Figure 3.11: Idealised STDP circuit. Explanation in text.

up and down, shown as M6-M9 in figure 3.10, is preserved here as transistors M1-M4. The potentials for potentiation and depression, *Pot* and *nDep* respectively, are assumed to be generated in the same way. The transistor gated by the *nWeightLowThr* bias, shown here as M6, is complemented by an nMOSFET M5, while the MOSCAP device has been replaced with an ideal capacitor C1.

The effect on this circuit of potentiation and depression events was simulated and the results are shown in figure 3.12. The effect of each type of event was simulated for a range of initial values for *nWeight* and for a range of values of *Pot* and *nDep* respectively. It can be seen that as the initial level of *nWeight* moves closer to its boundary in either direction, the amount of weight-dependence increases. qualitatively this is similar to the behaviour described by the model of Gutig et al. [2003] for intermediate values of μ (i.e. between 0 and 1), as shown in figure 2.1.

The point in the range of *nWeight* at which the weight dependence becomes apparent, however, depends on the level of *Pot* or *nDep* respectively. This can only be described as a shortcoming of this circuit, since *Pot* (*nDep*) represents a combination of the time-weighted effects of all pre-(post-)synaptic spikes prior to the post-(pre-)synaptic spike which caused the potentiation (depression) event; no such dependency has been discerned from such biological assays which have been conducted (although neither have such dependencies been discounted).

In figure 3.12, the left- and right-most dotted lines show the boundary of achievable results for the normal range of operation, since *nWeight* is bounded in the range 0.2-3.1V. However, if the bias *nWeightLowThr* is lowered from *Vdd*, then the right-most boundary

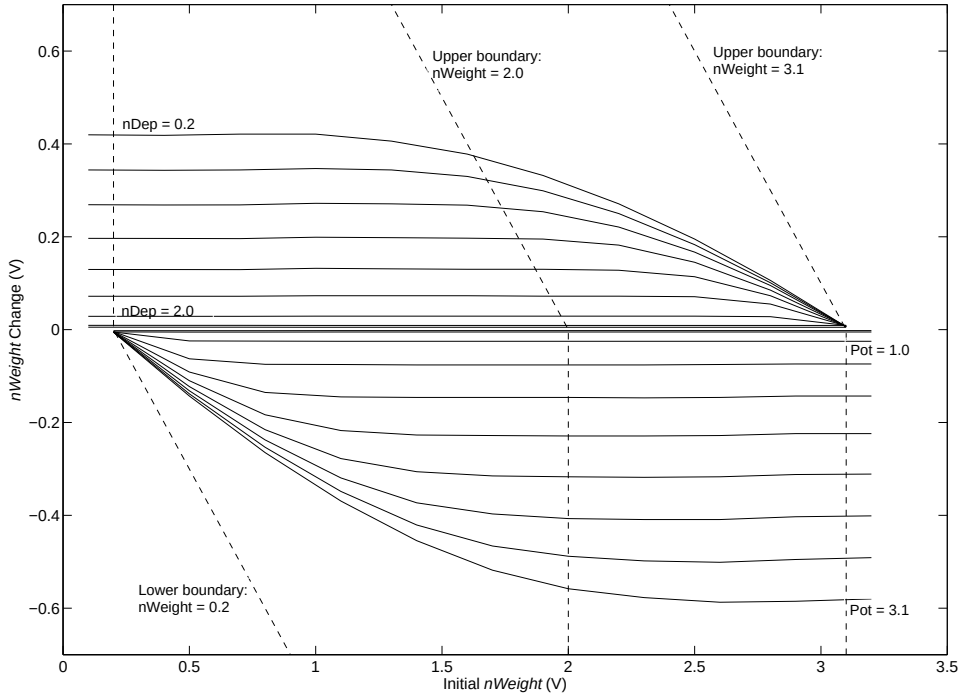


Figure 3.12: Performance of idealised STDP circuit. Simulation results from the circuit shown in figure 3.11. *nWeight* was initialised at each of a set of values in the range 0.1-3.1V, at intervals of 0.3V. For each initial value of *nWeight*, *nDep* was lowered to each of a set of values in the range 0.2-2.6V, at intervals of 0.3V. For each of these conditions, a single active-low pulse was applied to *nPrePulseDep* for 10ns and the resulting rise in *nWeight* was recorded; these results are shown on the top half of the graph, with lines linking sets of results for each value of nDep. The bottom half of the graph shows a corresponding experiment with potentiation, where *nWeight* was initialised in the range 0.2-3.2V (step size 0.3V), *Pot* was set in the range 0.4-3.1V (step size 0.3V) and pulses were sent in to *PostPulsePot* for 2.5ns (the difference in pulse width and the difference in magnitude between potentiation and depression reflects a choice of sizes for transistor pairs M1-M2 and M3-M4, which were optimised for implementing the parameters of the model in chapter 2). The outer boundary dashed lines mark the region of the graph which can be reached in normal operation. The inner dashed line marked “Upper boundary: nWeight = 2.0” marks an arbitrary upper boundary for nWeight values which can be achieved if *nWeightLowThr* is set accordingly (about 1.95V).

is shifted leftwards: for a suitable value (about 1.95V), the maximum achievable value of *nWeight* becomes 2V (the boundary shown by the central dotted line), since if it goes higher, transistor M6 in figure 3.11 conducts to bring it down again. This redefines the range of *nWeight*, and in so doing it redefines the weight dependence. Now potentiation is weight dependent whilst depression is (virtually) weight-independent, in accordance with the (albeit disputed) observation of Bi and Poo [1998]. Alternatively, by applying a *nWeighHighThr* bias of greater than *Gnd* to transistor M5 in figure 3.11, a lower boundary can be set on *nWeight*, which again would redefine the weight-dependence. By setting both *nWeightLowThr* and *nWeightHighThr* at intermediate values, the symmetry of the weight-dependence could be maintained (as in the formalism of Gutig et al. [2003]) whilst the amount of weight-dependence would be reduced (equivalent to reducing the value of μ); this would be achieved at the expense of reducing the operational range of *nWeight*.

This is only a partial solution to the tuning of weight dependence, since the range of *nWeight* delivered by the circuit needs to be interpreted correctly by the circuit which generates a synaptic conductance. Such flexibility could be simply achieved by applying a bias to the source of transistor M1 in figure 3.4 which is either independent or at least appropriately shifted from *nWeightLowThr*, allowing the point at which the highest value of *nWeight* results in a negligible raise in *SynCond* to be varied.

Figure 3.13 shows the effect of replacing the ideal capacitor C1 in figure 3.11 with a MOSCAP. The left graph shows the effect of using a pMOSCAP. If the range of nWeight is capped at 2V then disruptions caused by the capacitance profile are entirely avoided - they lie outside the effective range on nWeight. Thus, a weight-dependence profile is achieved which is in line with Bi and Poo [1998] whilst benefitting from the increased capacitance per unit area offered by a MOSCAP cf. a Poly-Poly capacitor (whose capacitance profile is essentially ideal).

If on the other hand, an nMOSCAP is used, as shown in the right graph in figure 3.13, this illustrates yet another way to alter the weight-dependence profile and it is this latter option which has been implemented in this project. The weight-dependence of potentiation is reduced, albeit in a non-linear way, whilst the weight-dependence of depression is increased, albeit in a way which has not yet been modelled or observed. Thus on average the weight-dependence of potentiation and depression are more balanced, in accordance with the formalism of Gutig et al. [2003]. By assisting the transition of weights through the region in which the capacitance of the nMOSCAP is reduced, the weight profiles are

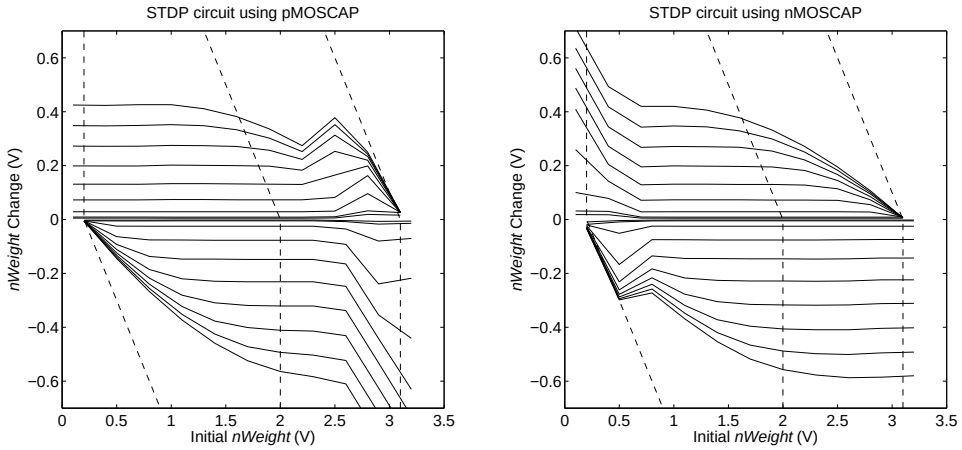


Figure 3.13: Performance of idealised STDP circuit. Simulation results from the circuit shown in figure 3.11 but with the capacitor C1 replaced with a MOSCAP device. Left: C1 is replaced with a pMOSCAP, referred to *Vdd*; right: C1 is replaced with an nMOSCAP, referred to *Gnd*. All simulation details are as in figure 3.12 and the same boundary lines are shown, for reference.

likely to affected, but not in ways that are likely to make a qualitative difference to the performance of the learning rule; this is because STDP will continue to adjust the total weight of the incoming synapses to each neuron to achieve homeostatic feedback on its output firing rate and because all synapses will continue to be in competition to control the firing of the neuron. As before, the implementation benefits from the increased capacitance per unit area of the MOSCAP.

3.4.5 Switched capacitors and clocks

As noted in section 3.4.3, the membrane currents are implemented with a switched capacitor circuit driven by a pair of non-overlapping clock signals. In total, 4 pairs of non-overlapping clock signals are used, the others being the decay of the *SynCond* node (section 3.4.2) and the decay of the potentials for potentiation and depression, (section 3.4.4). The principle advantage to the use of switched capacitors is that the conductances (and therefore the time constants) that they implement can be altered over a very wide range simply by changing the frequency of the clocks. At the expense of slightly more area, the switched capacitors which implement exponential decays on the *Pot*, *nDep* and *SynCond* nodes could be replaced with parasitic insensitive circuits [Allen and Holberg, 2002, p.

514]. The delivery of these clock signals is discussed in appendix C.4.

3.4.6 Pulse generators and bias generators

The circuits which have been described depend on a set of precisely timed pulses and accurate voltage biases for correct operation. The methods for generating these are described in appendix B.

3.5 Results and discussion

The chips were fabricated using the AMS 0.35μ 4-metal 2-poly process. For the chip results displayed hereafter, the following experimental set up was used. Each chip contains an array of $8 \times 4 = 32$ neurons. Each neuron has 64 synapses with reprogrammable 9-bit address-event receivers (which will be described in chapter 4). 8 chips were organised in a grid arrangement [Merolla et al., 2007] so that any neuron on any chip could send or receive spikes (address-events) with negligible delay. Input address-events could be sequenced from a PC and streamed with time stamps to an FPGA (Xilinx Spartan 3 on an Opal Kelly XEM3010 integration module). The FPGA would then transmit the address-events at the correct times (where time is measured in microseconds relative to the beginning of a simulation). On demand, the *nWeight* value of any one synapse could be buffered out to a pad, via on-chip unity-gain buffers. In addition one neuron on each chip (the one in the right-most bottom-most position, i.e. position Y=7 X = 3 according to a 2D zero-based addressing scheme) has its *SynCond*, *Mem* and *nDep* values buffered out to pads via unity-gain buffers, as well as the *Pot* value of its final synapse (i.e. synapse 63, according to a zero-based addressing scheme). Analogue values were either sampled by an oscilloscope (Agilent 54622D) or by an ADC (Texas Instruments TLV2553).

General parameters for simulations given in appendix D were used, unless otherwise noted.

3.5.1 Spikes in, spikes out

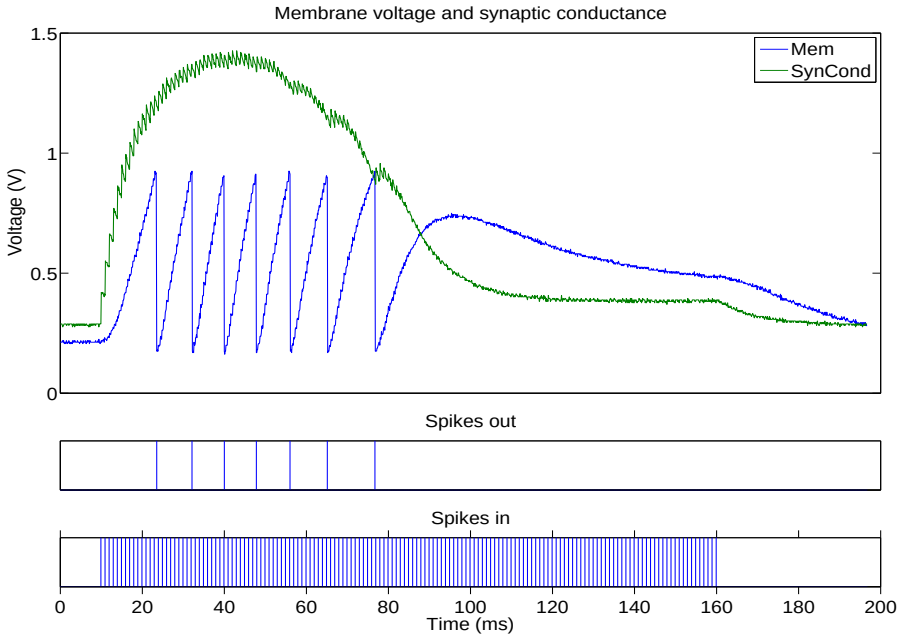
In this section the basic properties of the neuron and synapse designs are demonstrated. Figure 3.14(a) shows how *SynCond* and *Mem* varied, and shows the output spikes gen-

erated. From the start of a burst of incoming spikes, *SynCond* rose rapidly and this was followed by the rise of *Mem*, leading to the first spike. *SynCond* continued to rise and the output spike rate increased. Upon each output spike there was a potentiation event which increased the weight of the synapse, but upon each pre-synaptic spike there was a depression event which decreased the weight of the synapse. The depression overcame the potentiation so that *SynCond* peaked short of its maximum level around time 40ms and then started to fall. The rise of *Mem* and the output spike rate then fell accordingly until after time 80ms the level of *SynCond* was too low to allow the neuron to spike; after this, *Mem* peaked below its spiking threshold and then fell away. The weight approached its minimum value at around 100ms, after which *SynCond* approached a new plateau. Once the burst finished at 160ms, both *SynCond* and *Mem* decayed back towards their resting levels. Figure 3.14(b) shows a closer view of the same trace, in which the time discretisation of the decay of *SynCond* and synaptic current into *Mem* can be seen. This is discussed further in section 3.5.2.

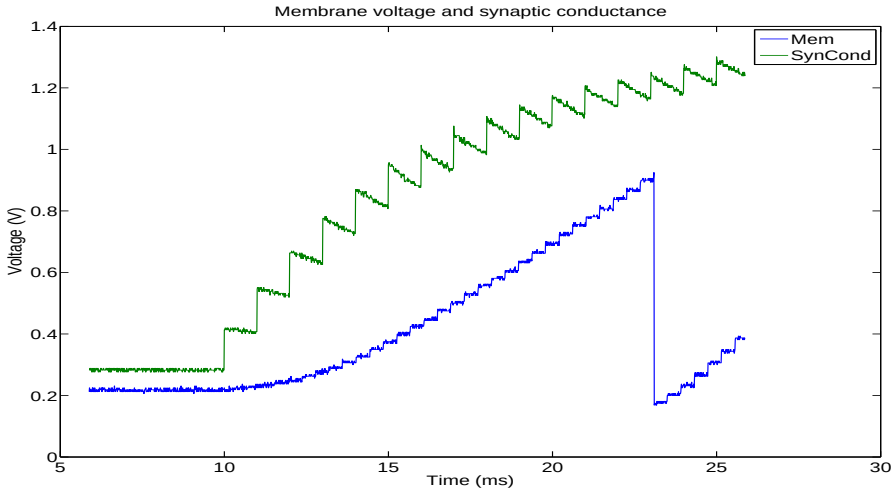
Figure 3.15 shows the results of an experiment (described in the figure caption) to characterise the influence of input spike rate on *SynCond* and of *SynCond* on output spike rate. It can be seen that as input spike frequency increases, *SynCond* increases but the rate of increase slows, as *SynCond* approaches its limit. As *SynCond* is reduced it reaches a point at which the output rate reaches zero. Just above this, at 0.58V is the lowest level of *SynCond* which elicits output spikes. Bearing in mind that *SynCond* represents the level of a current onto the membrane, it can be said that the rheobase of the neuron occurs at the current which corresponds to a *SynCond* of 0.58V.

3.5.2 Membrane decay

As noted in section 3.4.3, if C_{mem} and C_{Switch} can be sized carefully, bearing in mind the likely value of the parasitic capacitance C_{Par} due to layout, then a useful range of membrane current magnitudes can be delivered by the circuit. In practice, sizing the capacitors was a trade-off; larger capacitors, and in particular a larger C_{Mem}, allow the period of *SynCondClk* to be faster, so that the discretised charging and discharging of *SynCondClk* is a better approximation to continuous time. The compromise taken was to set C_{Mem} at a size which did not dominate the neuron circuitry and to accept the longer clock period this would entail. For C_{Mem} of $9.7 \times 10.4\mu m$ (1.3% of the area of the central neuron circuitry) and C_{Switch} of $4 \times 4\mu m$, it was found that a clock period of $358\mu s$ gave $\tau_m \approx 20ms$, though



(a)



(b)

Figure 3.14: The effect of pre-synaptic spikes on a neuron. A single neuron was configured so that a single synapse was connected to a chosen pre-synaptic address. A burst of spikes was sent from the pre-synaptic neuron at regular intervals of 1ms. The burst started at time 10ms and continued until time 160ms. Parameters were as in appendix D except $\tau_{ex} \approx 10ms$, $\tau_m \approx 22ms$ and $PrePulseSynCond \approx 9ns$. (a) Top: membrane voltage and synaptic conductance. Middle: post-synaptic (output) spikes. Bottom: pre-synaptic (input) spikes. (b) Close up of membrane voltage and synaptic conductance.

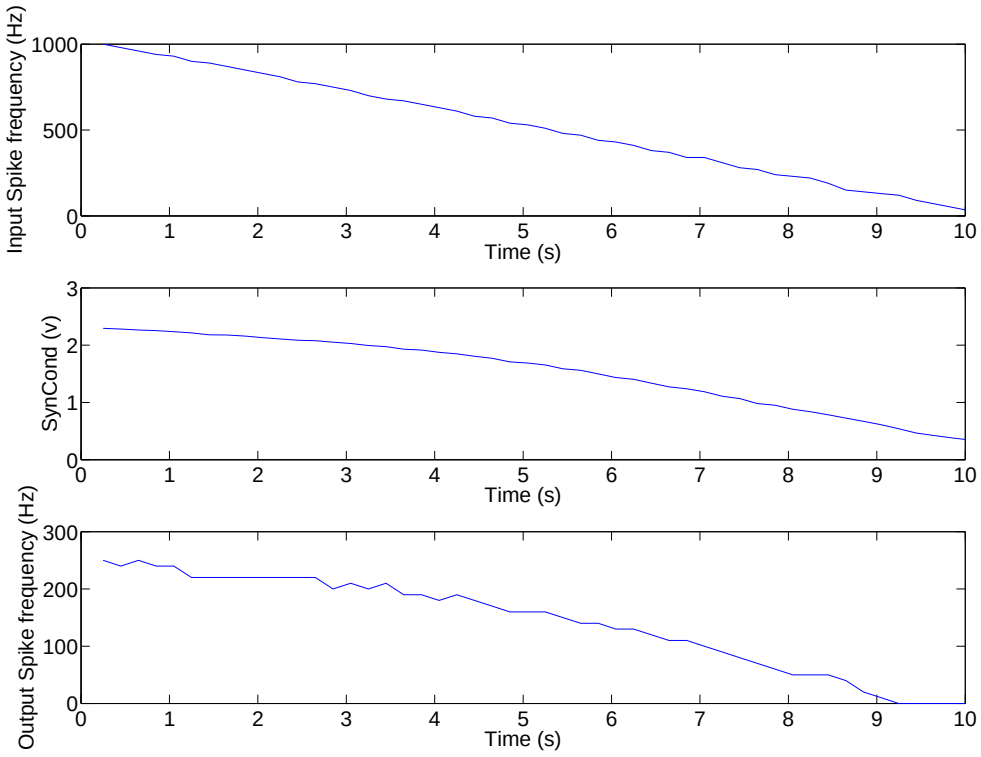


Figure 3.15: The relationship of *SynCond* to output rate. With one pre-synaptic neuron connected to one synapse of a post-synaptic neuron, a regular pre-synaptic pulse train was sent (with depression inhibited). The frequency was held constant for periods of 200ms and then linearly reduced to a new constant level. This was repeated over a period of 10s. Parameters were as in appendix D except $\tau_{ex} \approx 10ms$, $\tau_m \approx 22ms$ and $PrePulseSynCond \approx 9ns$. Input and output frequency and mean SynCond were recorded for the second half of each 200ms period (the first 100ms of data was discarded to allow the performance to settle). Top: input spike frequency; Middle: SynCond; Bottom: output spike frequency.

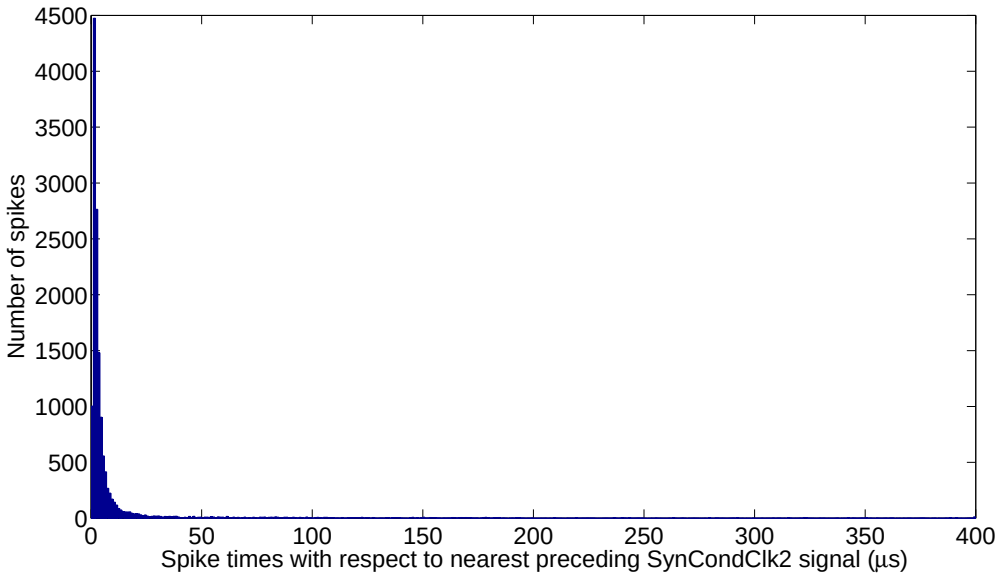


Figure 3.16: Histogram of spike times with respect to nearest preceding *SynCondClk2* signal, for a 2 second period in which Poisson spike trains were delivered to all neurons with all synapses connected to a random pre-synaptic partner and with depression inhibited. Both in-going and out-coming spikes were timestamped with an accuracy of $1\mu s$.

with some variation between neurons. The parameterisation process which yielded the value of this clock period is described in appendix E.

The clock period of $358\mu s$, which delivers the desired time constant, is longer than the $100\mu s$ time step of the simulations in chapter 2 and thus the corresponding simulations which can be delivered by the fabricated chip are of coarser time discretisation. An advantage of this longer clock period is that less energy is dissipated in the charging and discharging of the global clock nodes, whilst a possible disadvantage is that synchronisation of spike times may lead to unwanted network behaviours, although $358\mu s$ is still much faster than the time constants for STDP (20ms or higher). Figure 3.16 gives a histogram of spike times with respect to the nearest preceding *SynCondClk2* signal (the signal upon which any increments to *Mem* occur). It can be seen that spike times are strongly correlated with the *SynCondClk2* signal; the existing variation is mostly within $10\mu s$ of the signal and is likely due to a combination of: propagation time of the clock signal; variable time to spike due to the positive feedback in DA1, M1 and M4-M6 in figure A.1; queueing of address-events in the arbitration circuitry; and delay in travelling around the grid.

3.5.3 Linearity of synaptic integration

A theory of how to improve the linearity of synaptic integration by the use of a MOSCAP device was developed in section 3.4.2. However, in practice, the *SynCond* node has a large parasitic capacitance as a result of contributions from each of the 64 synapse circuits to which it is attached. This capacitance is of the same order of magnitude as the capacitance due to the pMOSCAP and has the characteristic of an ideal capacitance. The result of this is shown in figure 3.17. It can be seen that the charging profile is curved for strong synapses, in agreement with the simulated behaviour of an ideal capacitor, shown in figure 3.6. The contribution of the pMOSCAP is still present - this is more apparent in the graph to the right showing the charging profile for a weak synapse. Thus in the fabricated chip, linearity of integration has not been achieved as well as was hoped in this case. A more successful application of the use of the MOSCAP non-linearity is presented in section 3.5.5.

3.5.4 Spike Timing Dependent Plasticity

The basic behaviour of the circuitry for STDP is shown in figure 3.18, where, similarly to in figure 3.14, a single potentiated synapse received a periodic stream of spikes lasting 100ms. As before, the neuron generated 7 output spikes before the synapse became too weak to drive the neuron to fire, as can be seen by the *Mem* trace. The biases *nDepMin* and *PotMin* were generated by programmable chip-wide current bias generators and were set to voltages which should generate currents through M7 and M8 respectively in figure 3.10 of 1nA. Empirically it can be seen that these biases caused a resting level for *nDep* and *Pot* of approximately 2.45V and 0.6V respectively. As the input spikes arrived, *Pot* (representing the potential for potentiation) was driven upwards with each incoming spike, reaching a high resting level of $\approx 2.8V$ by 30ms, at which point it was held in balance by the leak conductance implemented by the switched capacitor M2 and M4-M5. nWeight started at $\approx 0.2V$ (i.e. a strong synapse) and as the first spikes arrived it rose to $\approx 0.35V$ in the period before the neurons started to generate output spikes. According to simulations, approximately $100mV$ of this rise would have been due to the rise in *Pot* transmitted through the gate-drain capacitance on transistor M8; the remainder would have been due to small packets of charge sourced through M6-7 on each pre-synaptic spike despite the weak bias offered by *nDep*. Upon the first spike generated, at time 25ms, *nWeight* was

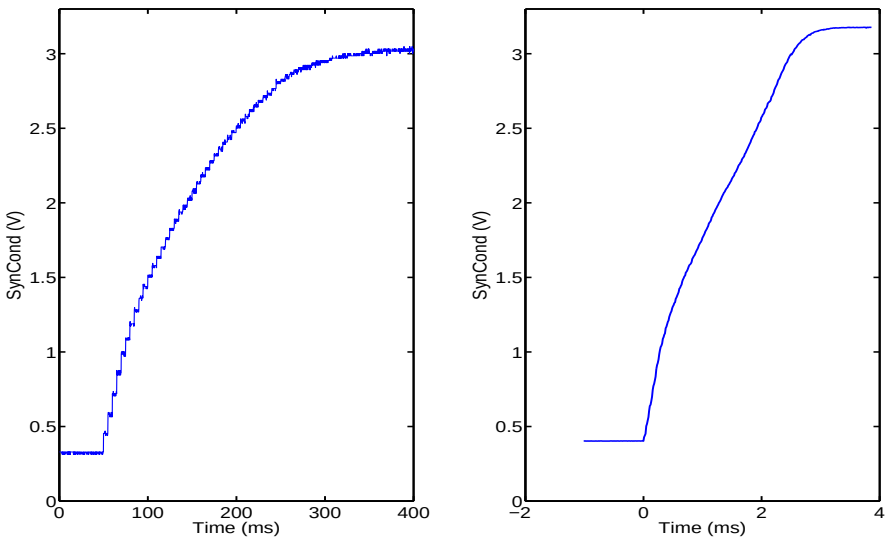


Figure 3.17: The charging of the *SynCond* node. Left: *SynCond* was charged from its resting level of $\approx 0.3V$ by a series of minimal width pulses to different consecutive synapses, where each synapse had maximum weight ($nWeight \approx 0.2V$). The pulses came every 5ms, starting at time 50ms. Right: *SynCond* was charged from slightly above its resting level ($\approx 0.4V$) by a series of minimal width pulses to a single synapse. The pulses came every $10\mu s$, starting at time 0s. The graph is smooth because the sample period was $\approx 12\mu s$. The synapse started weak ($nWeight \approx 2V$) and the weight decreased slightly during the early part of the charging due to small amounts of depression which cannot be eliminated; this explains the bend in the graph around $SynCond = 1 - 1.5V$.

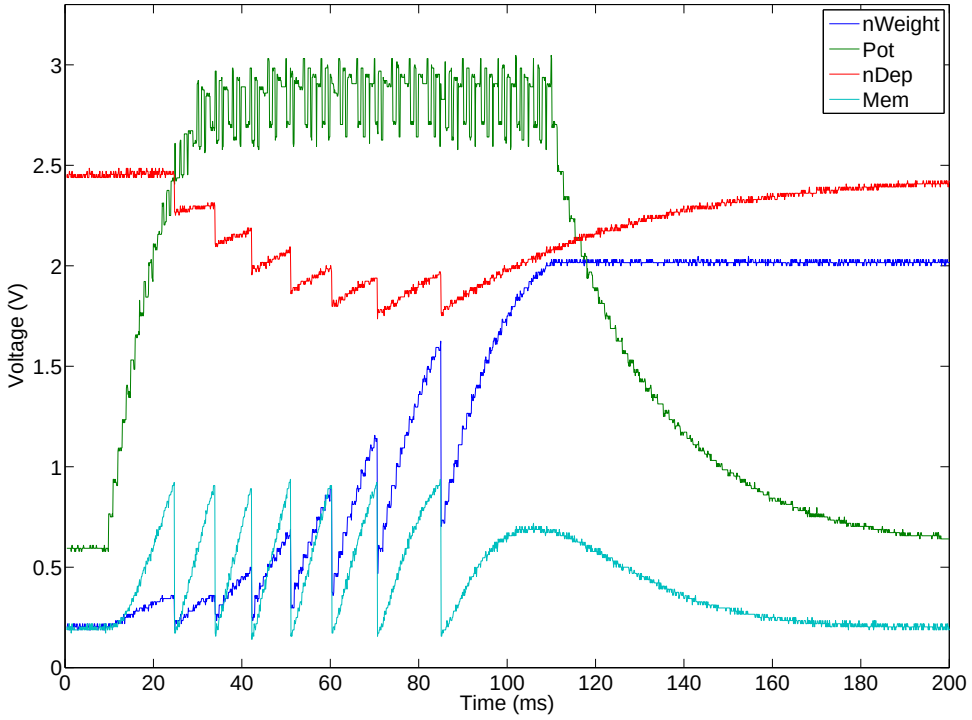


Figure 3.18: Spike Timing Dependent Plasticity in action. A regular stream of spikes was sent to one synapse. The stream started at time 10ms and continued until time 110ms, with a frequency of 1kHz. Parameters were as in appendix D except $\tau_{ex} \approx 10ms$, $\tau_m \approx 22ms$ and $PrePulseSynCond \approx 9ns$.

sharply lowered back to its minimum value of $\approx 0.2V$ by the *PostPulsePot* signal acting on M9; nDep was also lowered by the PostPulseDep signal acting on M14, representing an increase in the potential for depression. Thereafter, each pre-synaptic spike caused a larger increase to *nWeight*, as can be seen by the steeper gradient of *nWeight* after each output spike. The final few output spikes were not sufficient to restore *nWeight* to its minimum level and thus the average level of *nWeight* increased, until after time $\approx 90ms$ the synapse was no longer strong enough to drive the neuron to fire. No further potentiation took place and by the time the input spikes ceased, nWeight was reduced to $\approx 2V$, close to the level which represents minimum strength, where it then stayed as there were no further plasticity-causing events. *Pot* and *nDep* then decayed away back to their respective minima. Notice that *Pot* decayed faster than *nDep*, since $\tau_+ \approx 20ms$ and $\tau_- \approx 64ms$

3.5.5 Potential for potentiation: linearity of integration

The technique of using a MOSCAP device to achieve greater linearity in the charging of a capacitor through a transistor has been applied more successfully in the charging of the *Pot* node, as can be seen in figure 3.19. As spikes arrive, the node is charged from 0.2V all the way up to 3.1V closely following a straight line. (note that *PotMin* is usually set to a higher level, around 0.6V). A comparison is given to the expected results if an ideal capacitor were used.

It is worth noting that integration is approximately linear only up to a maximum level, beyond which further inputs cease to have any effect. This applies to all nodes where inputs are integrated and subject to an exponential decay, namely, *Pot*, *nDep* and *SynCond*. This is discussed further in appendix F. With regards to *Pot* and *nDep*, there is a trade-off to be made between the ability of *Pot* and *nDep* to integrate spikes and the ability to modify the parameters A_+ and A_-. The pulses *PrePulsePot* and *PostPulseDep* cause the inputs to *Pot* and *nDep*, and they work together with the pulses *PostPulsePot* and *PrePulseDep* to define the parameters A_+ and A_-. By increasing the duration of these pulses to increase A_+ or A_-, the ability of *Pot* and *nDep* respectively to integrate multiple spikes is reduced.

3.5.6 Weight stability

With *nWeightHigh* and *nWeightLow* set to *Gnd* and *Vdd* (3.3V) respectively, *nWeight* discharges rapidly in the absence of plasticity events, at a rate (measured in the range 1.7V

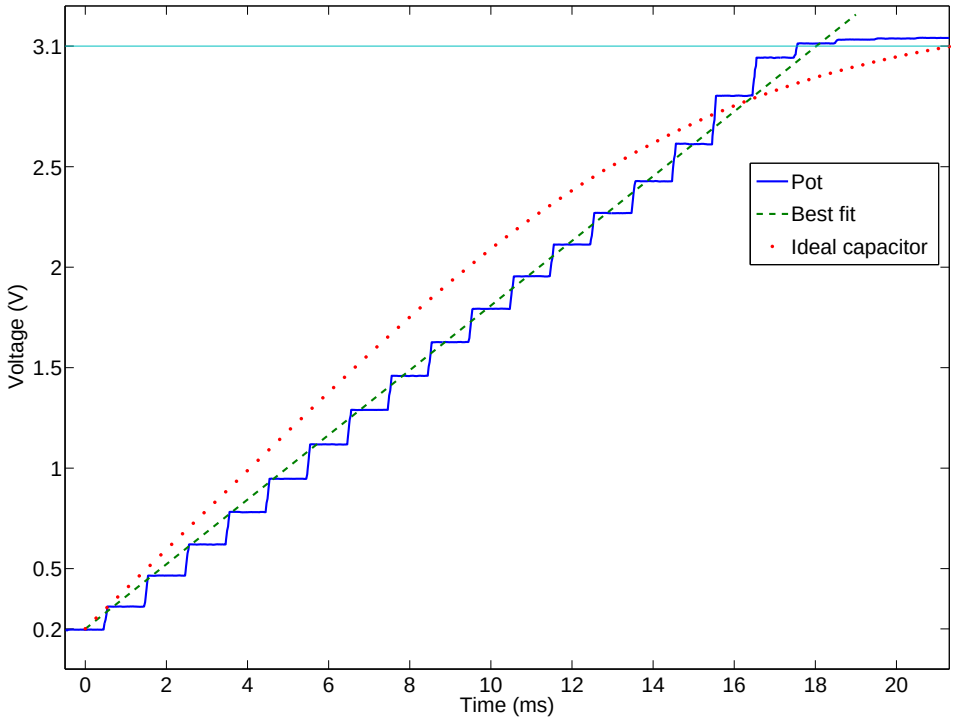


Figure 3.19: *Pot* charging. A stream of spikes was sent from a single neuron and simultaneously received by a single synapse on each of 7 neurons spread across 7 different chips. The spikes had a frequency of 1000 Hz, starting at time 0.5ms. Meanwhile *PotClk* was disabled so the *Pot* nodes would not decay. The *Pot* node of each of these synapses was sampled and the solid line gives the mean of these samples from each sampling sweep. The dashed line gives the best linear fit for this curve up to the selected desired maximum level of 3.1V with the y-intercept fixed at 0.2V (the minimum achievable resting level of *Pot*, i.e. *PotMin*). The dotted line shows simulation results for an ideal capacitor being charged through a transistor configured as M3 in figure 3.10 (with its drain connected to the ideal capacitor), with time scaled so that the best fit line also applies to it for the range 0.2V-3.1V.

down to 0.7V) of 692V/s; for an expected nWeight capacitance of $\approx 0.5pF$ this implies a current of 346fA. Discharging to *Gnd* is to be expected since a pMOSFET has a higher (negatively-defined) threshold with respect to *Vdd* than an nMOSFET of the same dimensions has with respect to *Gnd*; thus when a pMOSFET is gated by *Vdd* it is more deeply subthreshold than an nMOSFET gated by *Gnd*. When *nWeightHigh* is raised to 0.2V, and *nWeightLow* is lowered to 3.1V, the rate at which *nWeight* changes is greatly reduced, as shown in figure 3.20. Respectively raising and lowering these levels further does not decrease the rate of change any further, suggesting that reverse diode leakage to substrate is dominant at this point. Although learnt weight values begin to decay immediately, some trace of the learned memory is retained over several minutes.

3.5.7 Mismatch

There are many sources of mismatch in the circuits, with effects at each stage of the process from delivering a spike to raising membrane potential to adapting synaptic weights. In this section, results are presented which indicate the extent of these effects in the fabricated system. If each source of mismatch were investigated and characterised separately it may give a misleading impression of the cumulative effects. This is because the neural system being implemented has homeostatic properties. In particular, it was shown by Song et al. [2000] that the effect of STDP on a set of synapses afferent to a single neuron may be to adjust the overall amount of weight of the synapses, achieving a smaller change in output spike rates in response to a change; the change may be in the input spike rate or in the maximum strength of a synapse. Therefore to investigate the effects of mismatch on the performance of neurons, the chips were configured so that each neuron had its synapses connected to exactly the same set of input neurons (with no lateral connections) and typical spiking input was provided. Then, any differences in performance between the neurons could be attributed to a combination of mismatch and electronic noise, effectively giving an upper limit for the divergence that can be expected due to mismatch.

The connectivity was typical for the experiments to be performed, except that all 64 synapses were feed-forward; they were connected randomly according to a Gaussian distribution around a single pre-synaptic location. Poisson distributed spike trains with frequency 20Hz were sent for 10s from all input neurons. The results are shown in figures 3.21-3.22. Figure 3.21(a) is a histogram of the output spike rates for the neurons whilst 3.21(b) shows how these spike rates were distributed around the neurons. There are some

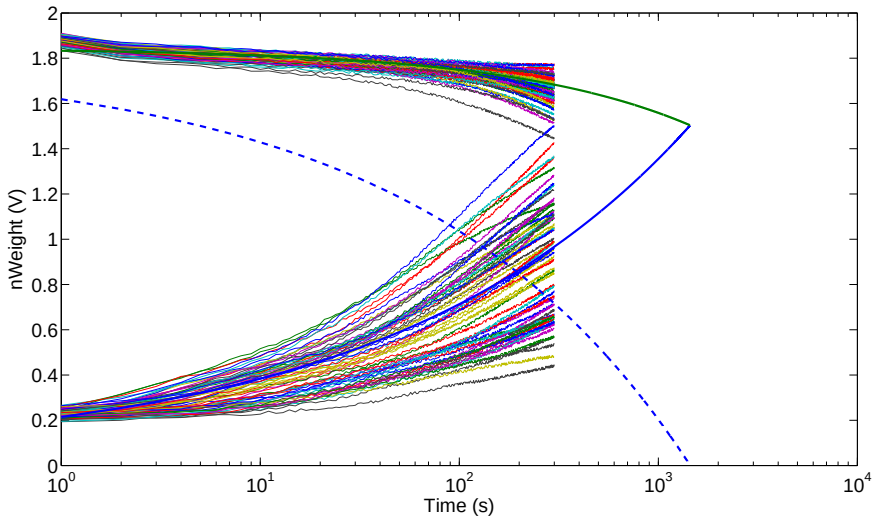


Figure 3.20: Weight stability. All synapses were initially connected, to set *nWeight* to $nWeightHigh = 0.2$ (whilst $nWeightHigh = 3.1$ and $nWeightLowThresh = 1.95$). *nWeight* was then sampled for each synapse each second, up to 300s. A selection of these traces are shown for 64 synapses (those of neuronY0X0). In a separate experiment, all synapses were initially connected to a single pre-synaptic location, then minimised (i.e. *nWeight* raised) by the raising of $nDepMin$ and the sending of a burst of spikes from the pre-synaptic location. *nWeight* was then sampled for each synapse each second, up to 300s; traces are shown for the same 64 synapses. Low synaptic weights rise at different rates whilst high weights fall at different rates. The weight of each synapse asymptotes towards a resting level at which the currents into and out of the node balance. Simplistically, the trajectory of each *nWeight* can be modelled as an exponential decay. As an arithmetic average of exponential curves can be modelled with a power-law curve [Anderson, 2001], best power-law fits of the rising and falling trends (over all synapses, not just those pictured) are shown and extended out to the point at which they cross over. While the rising and falling curve for an individual synapse will not cross but only converge, the distance (in Volts) between the rising and falling trend lines up to the crossing point (the dotted line) is indicative of the extent to which a weight will remain deflected away from its resting level after learning has occurred; thus the rate of decay of this distance is indicative of the rate at which learnt memories decay. This distance decays to 90% of its maximum in 8s, 75% in 43s, 50% in 227s and 0% in 24 mins. The relatively fast drop in *nWeight* around 1.9V in the first few seconds is due to discharging through M10 in figure 3.10.

overall differences between chips and some random differences between neurons within the same chip. There also appears to be a slight systematic variation within chips such that the last and to a lesser extent the first row of neurons have lower spike rates than the rows in the middle; this is probably the result of differences in the pulse lengths at each neuron, as pulses are broadcast laterally across the chip, see section 4.5.2; this could be resolved in a mature implementation with more attention to the design of the clock distribution network, or alternatively by adopting a pulse-width independent synapse design, as will be proposed in section 7.2.2. Furthermore, the bottom-right neuron in every chip has a lower spike rate; this is almost certainly because buffers on these neurons' *Mem* and *SynCond* nodes for test purposes added capacitance and affected their functioning. A histogram of the mean normalised weight for the synapses of a neuron is given in figures 3.21(c), and 3.21(d) gives the distribution of these weights around the neurons. Again there is some variation between chips and some variation between neurons on the same chip. Importantly, however, there is an inverse relationship between the spike rate of a neuron and the mean weight of its synapses — those neurons which spiked faster ended up with more depressed synapses (this can be seen more clearly in figure 3.21(e); the Pearson correlation coefficient is -0.61). This is the homeostatic effect of STDP in action; a lower mean weight for incoming synapses will make the neuron less likely to fire, providing negative feedback on the divergence of spike rates. Although the result is a spread of spike rates which may or may not be deemed acceptable for any given application, it can at least be deduced that the spread of spike rates is not so severe as it would be were it not for the implementation of STDP.

Figure 3.22(a) shows output spike trains for the neurons. Although there is some variation in the time of each spike, it appears that neurons generally produce a similar pattern of spikes, i.e. the same pattern of input elicits a similar pattern of output, at least over a short time scale. Figure 3.22(b) shows the trajectory due to STDP of the weights of the same synapse in a selection of different neurons, over 10s. Although there is significant variation in the overall weight of this synapse in different neurons, even the highest and lowest weight traces demonstrate a similar trajectory with many of the same small-scale fluctuations over the entire duration of the trace. The overall divergence in the weights may be due to differences in the relative strengths of transistors M7 and M8 in figure 3.10. However, results later in section 6.2.3 show that when no strong pattern of weights is implied by the statistics of the input spike trains, the behaviour of the system will ultimately diverge chaotically over time given identical inputs, due only to electronic noise, so it is

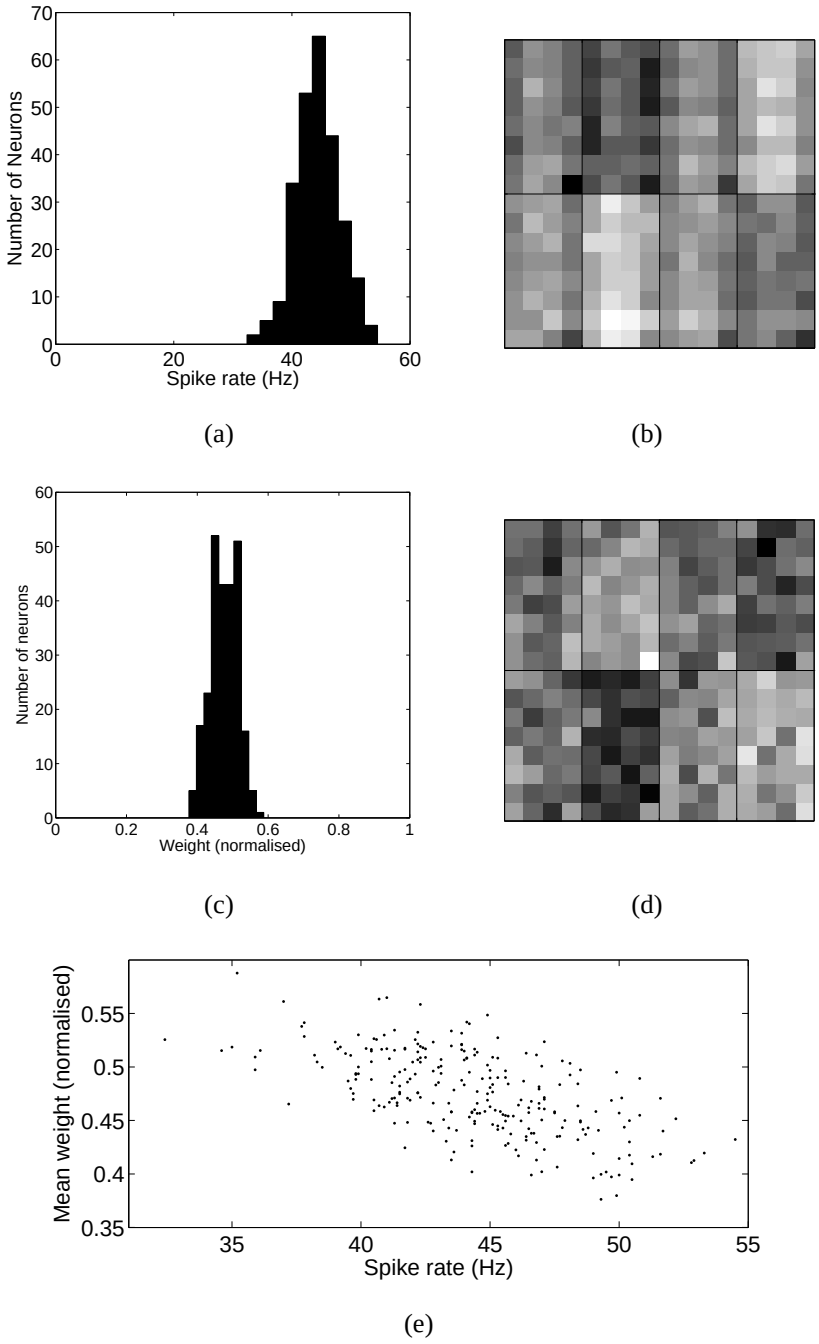


Figure 3.21: Effects of mismatch. (a) Histogram of output spike rates. (b) map showing variation in output spike rate across the chips, on a scale from white (fastest) to black (slowest); dividing lines show the boundaries of the individual chips which make up the neural area. (c) Histogram of mean weight of all synapses for each neuron; weights were normalised by linear interpolation of *nWeight* values between 1.9V (weight 0) and 0.2V (weight 1). (d) map showing variation in mean weight across the chips, on a scale from white (strongest) to black (weakest). (e) Mean weight vs. spike rate.

unlikely that the divergence seen here is purely the effect of mismatch.

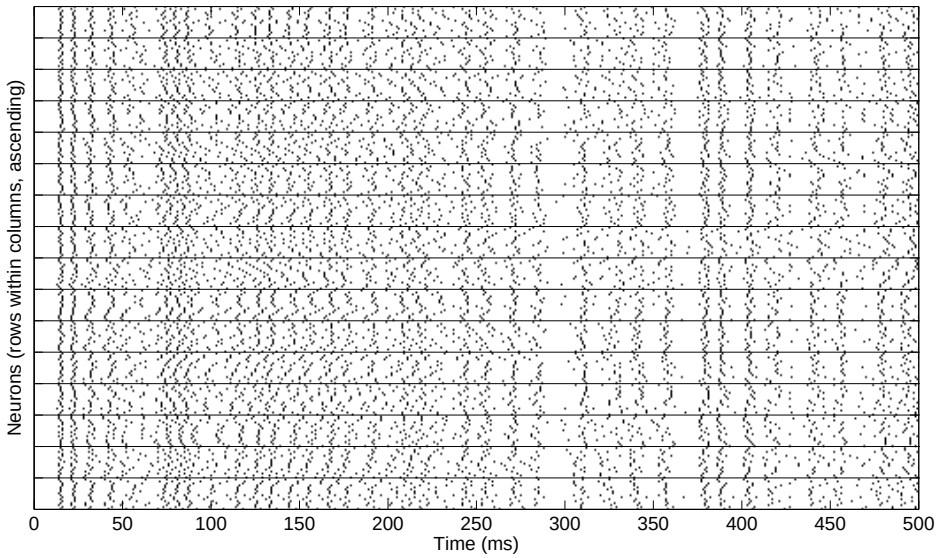
3.6 Conclusions

Literature from the field of neuromorphic engineering has been reviewed with a focus on analogue CMOS implementations of integrate-and-fire neurons and synapses which implement STDP. Circuitry has been presented which implements the single-compartment integrate-and-fire neuron required by the model in chapter 2. In addition a circuit for implementing STDP has been presented which introduces a degree of weight-dependence; this works on an entirely different principle to the only other published circuit which (explicitly) does so [Bofill-i Petit and Murray, 2004], by largely exploiting the characteristics of existing devices, resulting in a similarly compact implementation.

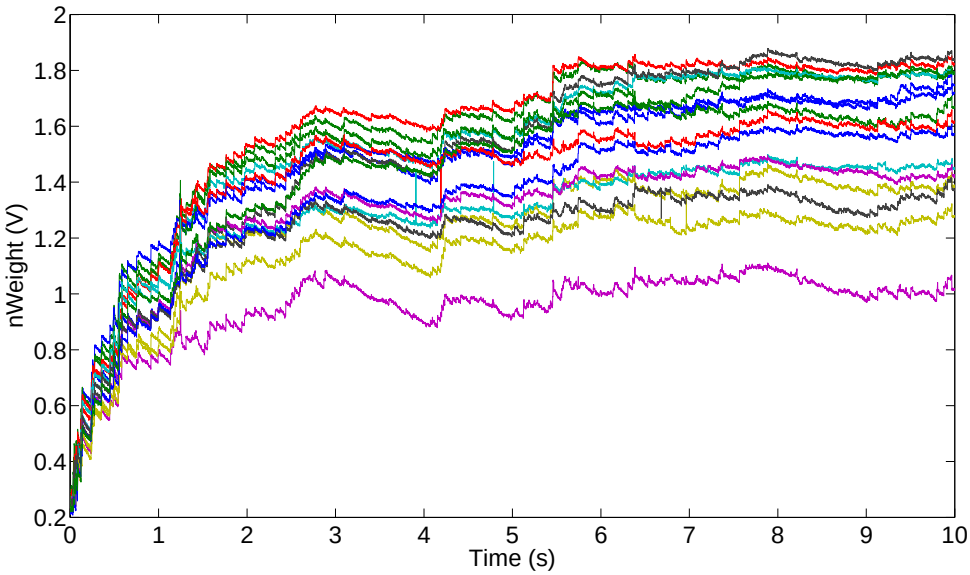
Switched capacitors have been used in a novel arrangement which simultaneously implements membrane currents from excitatory synapses and conductance towards a resting potential, for a potential space saving cf. existing switched-capacitor approaches, which handle these mechanisms separately [Glover et al., 1998]. However this mechanism has no real advantages over an existing design of switched capacitor integrator.

The STDP circuit presented is similar to that presented by Indiveri et al. [2006] in that pre- and post-synaptic pulses cause brief currents onto and off from a capacitor which represents the weight of a synapse, but it differs by raising(lowering) the source potential of the transistors gated by the spikes with respect to *Gnd*(*Vdd*) to choke leakage currents, resulting in weights which retain traces of their learnt values over tens of seconds. Though this is still much shorter than the duration of LTP and LTD observed in cultures of many hours, it nevertheless allows the learnt effects of many sets of inputs to accumulate and it will be seen in chapter 6 that it is long enough to allow learnt patterns to become embedded in the network topology, at least given the artificially increased synaptic rewiring rates which are achievable in a silicon implementation.

The capacitance profile of MOSCAP devices has been used in two novel ways. Firstly it has been used to increase the broad linearity of charging profiles which are non-linear due to the limited ability of transistors to act as ideal current sources. This was applied to the charging of the *SynCond*, *Pot* and *nDep* nodes, though it was most successful in the case of the *Pot* node. Secondly it has been used to alter the weight-dependence profile of STDP



(a)



(b)

Figure 3.22: Effects of mismatch, continued. (a) Output spike times for each neuron for the first 500ms; a black dot represents a ms in which the neuron produced at least one spike; lines divide each row of neurons. (b) *nWeight* for Synapse 0 for each of 16 neurons in a diagonal line across the area, i.e. Y0X0, Y1X1, ... Y15X15; For each synapse, *nWeight* was sampled every $160\mu s$.

as applied to the charging and discharging of the *nWeight* node, in a way which allows the overall behaviour to match broadly, if not in detail, an established model for STDP [Gutig et al., 2003], whilst benefiting from the smaller area per unit capacitance offered by MOSCAPs.

Chapter 4

A Distributed and Locally-Reprogrammable Address-Event Receiver

4.1 Introduction

The hypothesis presented in section 1.2 is based on the idea that synaptic rewiring could be implemented in neuromorphic VLSI by circuitry distributed throughout the synapses of a neural array. This chapter lays the groundwork by developing circuitry for receiving spikes at synapses.

Techniques for transmitting spikes in neuromorphic systems are reviewed, with a focus on Address-Event-Representation. A design is presented for an address-event receiver, where the receiving elements are distributed through the synaptic array and act simultaneously. This allows a spike to be received simultaneously by all the synapses on the axonal arbor. This receiver is compatible with existing address-event senders. The receiver is reprogrammable during run-time, allowing synaptic rewiring to be implemented. The scalability of this system is analysed and compared against existing systems with respect to silicon area, energy and time (thus, channel capacity), as numbers of neurons and synapses in a system increase. It will be shown that this system scales particularly well in terms of speed as synaptic fan-out increases. Results are presented from fabricated chips, which show first the simultaneous receipt of a spike by many synapses, then the simultaneous functioning of 8 chips (containing a total of 256 neurons and 16,384 synapses),

configured in a grid arrangement, achieving high (arguably record-breaking!) spike delivery rates. Finally, alternative designs are discussed as possible future work.

4.2 Literature review

There is a need to form interconnects between many integrated neuron circuits to create neural networks. The most simplistic approach to this would be to use a dedicated wire to connect each pair of neurons; however, in the worst case, where all-to-all connectivity is required between neurons, this approach would quickly become intractable, since the number of wires required would scale as N^2, where N is the number of neurons. Such a problem could be ameliorated to some extent as the number of neurons on a chip increases by the use of increasing number of layers for interconnect [Joyner et al., 2004]; however, it would cause a particular problem for multi-chip systems, where the number of Input/Output (I/O) pads that are available to relay signals between neurons implemented on different chips becomes the limiting factor. From the beginnings of the field of neuromorphic engineering, therefore, alternative approaches to this problem have been sought and the following review focuses on solutions which have been attempted within this field. The related issue of reconfigurability of neural networks is mainly deferred to chapter 5.

Eberhardt et al. [1989] created a system that used individual I/O pads for the input and the output of each neuron, with the synapses for each neuron implemented on a separate chip from the neurons themselves. All-to-all connectivity was achieved by using a crossed grid of wires to access each element of a square grid of synapses. The number of pads necessary for each neuron was limited to 1 each for input and 1 each for output, since each (voltage-carrying) input wire from a pre-synaptic neuron accessed all related synapses in a horizontal direction and all the outputs to each post-synaptic neuron sourced currents which were summed together on a single vertical wire. Thus the number of I/O pads required was $2N$, where N is the number of neurons serviced by a synapse chip (32 in the case of the system implemented). Such chips could be tiled in grids in order to implement larger networks; however, the number of synapses which could be created on a chip is limited to $(P/2)^2$ where P is the maximum number of pads which can be created, limiting the extent to which increasing synapse density can improve the scalability of such a system. Interestingly, this system is comparable to the solution adopted by the brain, wherein the stereotypical neuron has a single output wire (axon) and a small num-

ber of input wires (dendrites); these branch as necessary, with the branch points being relatively close to the synapses that are formed between them. The brain, however, does not implement all-to-all connectivity, but rather its connectivity is extremely sparse. In all but the simplest of nervous systems there is some indeterminism in the development of connections (and indeed in the development of neurons), such that different organisms of the same species develop brains which are broadly the same but differ in the detail of their network topologies. Mueller et al. [1989] and Satyanarayana et al. [1992] both created systems with perceptron-like neurons, with continuous analogue voltage output levels. These were continuously connected to a selection of each other's synapses within the same chip, via matrices of switches which could be reprogrammed by an external mechanism. Such an approach changes the problem from that of providing all-to-all connectivity to providing the possibility of all-to-all connectivity, but the scalability remains unchanged since there must exist N^2 switches and $2N$ wires. Arguably Eberhardt et al. [1989] is also an example of this, since a synapse could be programmed with a zero weight, which could be interpreted as the lack of a synapse between the two respective neurons. Coggins et al. [1995] gave an example of a fixed topology Multi-Layered Perceptron (MLP) with all-to-all connectivity between layers, all implemented with dedicated wires within a single chip, showing that this approach can be suitable for small-scale neural networks with particular applications.

In order to reduce the number of wires necessary to implement interconnect, multiplexing is adopted, that is, interconnect between many pairs of neurons is implemented by shared wires. Neal [2000] proposed a method in which each layer of a MLP with all-to-all connectivity passed analogue output values to the next layer sequentially along a chain of capacitors, switched according to a "bucket brigade" arrangement (previously used and better explained by Coggins et al. [1995], though for a slightly different purpose). This method required that the output of the network was calculated under the control of a global clock signal; it had the advantage that synaptic weights were stored separately from the circuit for multiplication of the weights with inputs, such that each neuron required only one multiplier circuit to implement all of its afferent synapses.

The approaches mentioned above are suitable for networks where the neurons output analogue levels of activation; this may represent a rate at which a biological neuron produces spikes. Such an analogue output level can be represented digitally, with the possible advantages of speeding transmission and immunity to noise during transmission. This could

be by means of numeric analogue-to-digital and digital-to-analogue conversion, but such an approach could suffer both from the size of circuitry required and from loss of accuracy in digitisation. Hirai et al. [1989] designed a system in which all values stayed in the digital domain. Tomberg et al. [1989] also did so, using 16 bit values for synaptic weights, but all bits were interpreted as sign bits of two's complement numbers; this reduced the complexity of circuitry necessary for addition and multiplication at the expense of reducing the range of values which could be stored and thus increasing the rounding error in conversion for a given number of bits. Alternative "pulse stream" approaches were described by Hamilton et al. [1992], in which an analogue output level can be used to drive oscillators so as to create pulsed digital signals which carry the analogue information in the form of pulse width, pulse amplitude, pulse frequency or the difference in phase between two pulses. The pulsed signals can then be multiplexed. One proposed scheme involved neurons taking turns to send the next pulse on a line, with the delay from the previous pulse representing its level of activation. The scheme is self-timed, rather than relying on a central clock; the receiving circuitry must stay synchronised with the sending circuitry in order to correctly de-multiplex the incoming signal. In order to increase accuracy in transmission in this scheme, longer delays must be allowed for; as the number of neurons which share the line increases, the rate at which they can be sampled decreases, and this rate will fluctuate as the average level of activation changes.

An analogue level of activation for a neuron may represent the rate at which the neuron produces spikes. Where spiking neurons are directly implemented, the signal to be transmitted from a neuron to a post-synaptic synapse is a discrete all-or-nothing pulse, which can be idealised as an event, occurring at a particular time. Indeed, the integrate-and-fire neuron can be seen as a device which turns an analogue level of activation (specifically, current input) into a stream of events with a rate which is related to the level of activation (linearly, in the simplest case); thus, spiking neurons can be seen as a special case of the pulse-stream approach, in which neural activation is encoded as pulse frequency. Spiking neuromorphic systems, however, allow the possibility that the time of individual spike events may carry information or otherwise influence the behaviour of a neural network; this timing information is a requirement for learning rules such as STDP. In spiking neuromorphic systems, the time taken for axonal transmission to occur in biology is frequently ignored and simply implemented as fast as possible, whereas some systems model the temporal dynamics of dendrites [Elias and Northmore, 1995].

Considering, then, how spiking events may be multiplexed, various approaches have been proposed. Tuffy et al. [2007] proposed a time-multiplexing architecture. Each neuron within a layer had a time slice in which it could control a node common to all the neurons of the layer. The control of the node passed from neuron to neuron by means of a shift register, connected in a loop and synchronised by a global clock. On receiving control of the node, a neuron would raise it if it had produced a spike since the last time it had control. The system was designed to implement all-to-all connectivity between layers; each neuron in the subsequent (target) layer therefore had one synapse for each input neuron. The synapses for each target neuron operated on a time slice which was also generated by a shift register. The shift register for the synapses of each target neuron was synchronised to the shift register for the input neurons; therefore at any point, within each target neuron there would be one synapse receptive to the input neuron which had control of the common node. At that point those synapses would latch the value of the common node and use it to produce input to their target neuron. As the number of neurons sharing an outgoing communication node increase, the time necessary to complete a cycle of all the neurons increases such that the time resolution of the transmission decreases. The capacitance of the node which an input neuron must drive also increases both with the number of input neurons and especially with the number of target neurons, increasing the energy cost of transmission.

4.2.1 Address-event representation

The time-multiplexing architecture of Tuffy et al. [2007] and the scheme of Hamilton et al. [1992] have in common that each neuron is sequentially polled for its output; then either the fact of a spiking event or the implied rate of spiking, respectively, is relayed. An alternative approach is to transmit spiking events only when they occur, so that the capacity of a multiplexed channel is used to transmit information about neural activity rather than inactivity. This is the basis of Address-Event Representation (AER). An array of neurons all share a bus (i.e. a set of wires). When one of the neurons generates a spike, this event is transmitted by placing the unique digital address of the source neuron on the bus; this can then be decoded by receiving circuitry and appropriate synapses targeted. This approach was first used in the theses of Sivilotti [1991] and Mahowald [1992] and has since been extended and improved. AER exploits the large difference in frequency between the spiking behaviour of biological neurons (on the order of 10-1000Hz) and

the capability of digital electronic communication (many MHz). When more than one neuron generates events whose transmission would overlap in time, the events can either be discarded [Mortara et al., 1995] or all but the first discarded [Abusland et al., 1996], or they can be queued using arbitration circuitry so that all are delivered [Sivilotti, 1991, Mahowald, 1992, Boahen, 2000].

The number of wires required to transmit, in parallel, the unique binary-encoded addresses of N neurons scales as $log_2(N)$, so the number of pads and wires necessary to interconnect chips is achievable for very large neural networks, e.g. 37 wires would be sufficient to encode unique addresses for the 10^{11}neurons of a human-sized nervous system. The development of word-serial AER reduces the number of wires required still further [Boahen, 2004]. In this scheme, the unique digital address of a neuron is broken into words, which are transmitted serially. When the words correspond to the X and Y portions of an address within a 2D grid, this protocol can minimise time losses due to channel arbitration, as the arbitration circuitry operates first in one dimension then in the other [Boahen, 2000].

Arbitrated word-parallel AER is the spike transmission protocol which has been adopted in this project; it functions as follows. The wires for carrying the addresses of the neurons are supplemented with two extra wires, a request (*Req*) line and an Acknowledge (*Ack*) line. The sending circuit loads the address onto the bus then raises *Req* (i.e. it goes from a low voltage, *Gnd*, to a high voltage, *Vdd*). The receiving circuit latches the address data then raises *Ack*. Once the sending circuit receives the *Ack* signal, it responds by lowering the *Req* signal and new data may be placed on the bus. The receiving circuit waits until both *Req* is lowered and until it is ready to receive the next event, before it lowers *Ack*. The sending circuit waits until *Ack* is lowered before it raises the *Req* signal, to repeat the protocol for the next event. The system is asynchronous (since no regular clocks are needed); it is self-timed by means of a “four-phase handshake”; spike transmission proceeds at the fastest rate achievable by both the sending and receiving circuits.

Faster asynchronous data transfer protocols are possible; if data is encoded with a one-of-N code, for example one-in-four, then from an initial null state where all the wires are at low voltage, the raising of exactly one wire transmits 2 bits of data concurrently with a request, eliminating the need for a delay between loading data and raising a specific request line. This was applied to interconnect for general processing, with additional wires for packet-switching, [Bainbridge and Furber, 2002] and is being applied specifically to a neural network simulator, [Plana et al., 2007]. An alternative called “one-change-in-

four" recently proposed by Kwabena Boahen (Stanford University, California, personal communication) seeks to reduce transmission time further by implementing a two-phase handshake, where a request is made by means of changing the state of one out of a set of four data wires. The receiving circuit then flips the state of *Ack* to acknowledge receipt. Each possible transition implies 2 bits of address data and enough sets of four wires are used in parallel to encode the address. This will eliminate the need for wires to undergo a second transition in order to return to the null state.

AER was originally conceived as a point-to-point protocol. If each neuron in one neural layer has a unique connection to only one neuron in a corresponding neural layer with a precise topographic mapping between the layers, the outgoing bus can be decoded directly by a row-and-column decoder on a receiving chip, and spikes are delivered correctly to the same location on a corresponding chip [as in Sivilotti, 1991, Mahowald, 1992]. Simplistically this type of one-to-one connectivity can be observed in some places in the nervous system, for example the connections from cone receptors to bipolar cells, at least in the fovea [Kandel et al., 2000, ch. 26]. More commonly however, as noted in section 2.1.2.1, neurons make connections to many other neurons (i.e. they have a large "fan-out") and receive large numbers of incoming connections ("fan-in"). As two examples, Xiong et al. [1994] found an average fan-out of 167 for retinal ganglion cells in the tectum of the hamster, whilst Palkovits et al. [1971] found an average fan-in of 85,000 onto the Purkinje cells of the cat. Despite such apparently large fan-in and fan-out, these networks are nevertheless very sparsely connected, when the overall number of neurons in the respective brain areas are considered, such that implementing dedicated synapses for each possible pair of neurons, as in the all-to-all network topologies reviewed above, would be extremely wasteful of circuitry. In order to implement arbitrary many-to-many network connectivity, address-events are commonly received not directly by a neural array chip but rather by a digital microcontroller and are then compared to a look-up table in memory in order to find out which outgoing address-events should be sent [Deiss et al., 1999, Mitra et al., 2006]. These are then sent sequentially to one or more receiving neural arrays. Essentially then, a look-up table is used to convert source neuron addresses into target neuron or synapse addresses; this is referred to below as the "look-up table" approach. This approach reduces the capacity of the bus in the presence of large fan-out. If, for example, the number of neurons which can be supported by a bus is $\approx 10^5$ [Boahen, 2000] and an average fan-out of 1000 is desired, the number of neurons which can then be supported is reduced to ≈ 100.

A similar approach was also used to map source-neuron addresses to target synapse addresses which stored in a look-up table not a verbose mapping between all source-target address pairs but rather a receptive field template which was used to calculate appropriate target addresses for each source address [Liu et al., 2001]; this system saved memory space at the level of the microcontroller at the expense of topological freedom – all axonal arbors and dendritic trees in the projection from one layer to another were identical. As the complexity of neuromorphic systems has increased, digital microcontrollers have been used to implement more aspects of the neural network model. This trend has been extended by Vogelstein et al. [2007] where other synaptic variables (number of release sites, probability of release and quantal post-synaptic response — the product of these is essentially the synaptic weight) were also held in the look-up table, allowing each neuron to have a single "general purpose" synapse circuit which acts as a number of virtual synapses.

An alternative approach to implementing axonal fan-out was demonstrated by [Serrano-Gotarredona et al., 1999]. This system took advantage of the observation that 2D topographic projections between neural layers in the visual system, for example between LGN and V1, can be usefully modelled as functions such as Gaussians or Gabor filters. When a spike was delivered to the receiving neural array by AER, it was used to target not just the neuron in the topographically appropriate location but also all those within a certain distance in each dimension, with the length of the pulse and thus the weight of the connection modified according to a separable approximation to the desired filter, which could be programmed. This therefore implemented a specialised type of axonal fan-out without reducing the channel capacity, but it did so at the expense of adaptability; all axonal arbors were constrained to be the same shape and this could not be adapted by a learning rule.

4.3 The broadcast approach

In this project, in order to overcome the bottleneck on channel-capacity as fan-out increases, whilst maintaining the possibility of adaptability by means of synaptic rewiring, an alternative approach has been taken. Address-events from a sending chip are directly received by a receiving chip and broadcast across the receiving chip's neural array. Simultaneously, each synapse compares that address to an address which is stored locally to the synapse, to establish whether the address-event was intended for itself. Many synapses can

store the same desired address and thus arbitrarily large axonal arbors can be implemented without reducing bus capacity. This will be referred to as the “broadcast approach”. This approach has been mooted before e.g. Deiss et al. [1999]:

> “Ideally each node should recognise its relevant source events, but our present multi-neuron chips use a DSP chip and lookup table to implement the fan-out from source address to the individual target synaptic addresses."

However to date, no such system has been implemented. In this section the broadcast approach is described in comparison to the existing look-up table approach, as described in section 4.2.1 above. The look-up table approach allows the use of receiving circuitry as described by Boahen [2000], which is shown in fig 4.1 (a).

The receiving circuitry which implements the broadcast approach is shown in figure 4.1 (b). The chip-level address-event receiver is compatible with existing address-event transmitters. An incoming request is acknowledged immediately and triggers local latching of the address bus and a timed delay followed by a timed pulse to synapses. Synapses do not acknowledge receipt of an event, rather the chip-wide broadcast is timed to last long enough for all synapses to receive it. A minimum cycle time is imposed sufficient to allow for the timed delays before the acknowledge signal is dropped. An example timing diagram for the receipt of a single address-event is given in figure 4.1 (c).

In contrast to the aforementioned approach of Vogelstein et al. [2007], more rather than less information is stored locally at each synapse circuit. Specifically, the digital address of the pre-synaptic neuron is stored, alongside the analogue synaptic variables representing weight and the potential for potentiation, as discussed in chapter 3. Given the local availability of information about incoming connectivity, neurons can take advantage of other information stored locally at the soma and in the synapses in order to change incoming connectivity. By additionally storing a binary variable at each synapse indicating whether or not the synapse actually exists, the synaptic weight is used to inform the decision whether to disconnect, in accordance with the model presented in chapter 2. The synapse circuit therefore becomes a circuit representing a potential synapse, part of the neuron’s total synaptic capacity. This is supplemented with a chip-wide mechanism for implementing synaptic connection, where the probability of a synapse forming with a given pre-synaptic neuron is influenced by the distance between the ideal location of that neuron and the post-synaptic neuron, allowing receptive fields to form according to 2D probabilistic distributions. This system is detailed in chapter 5.

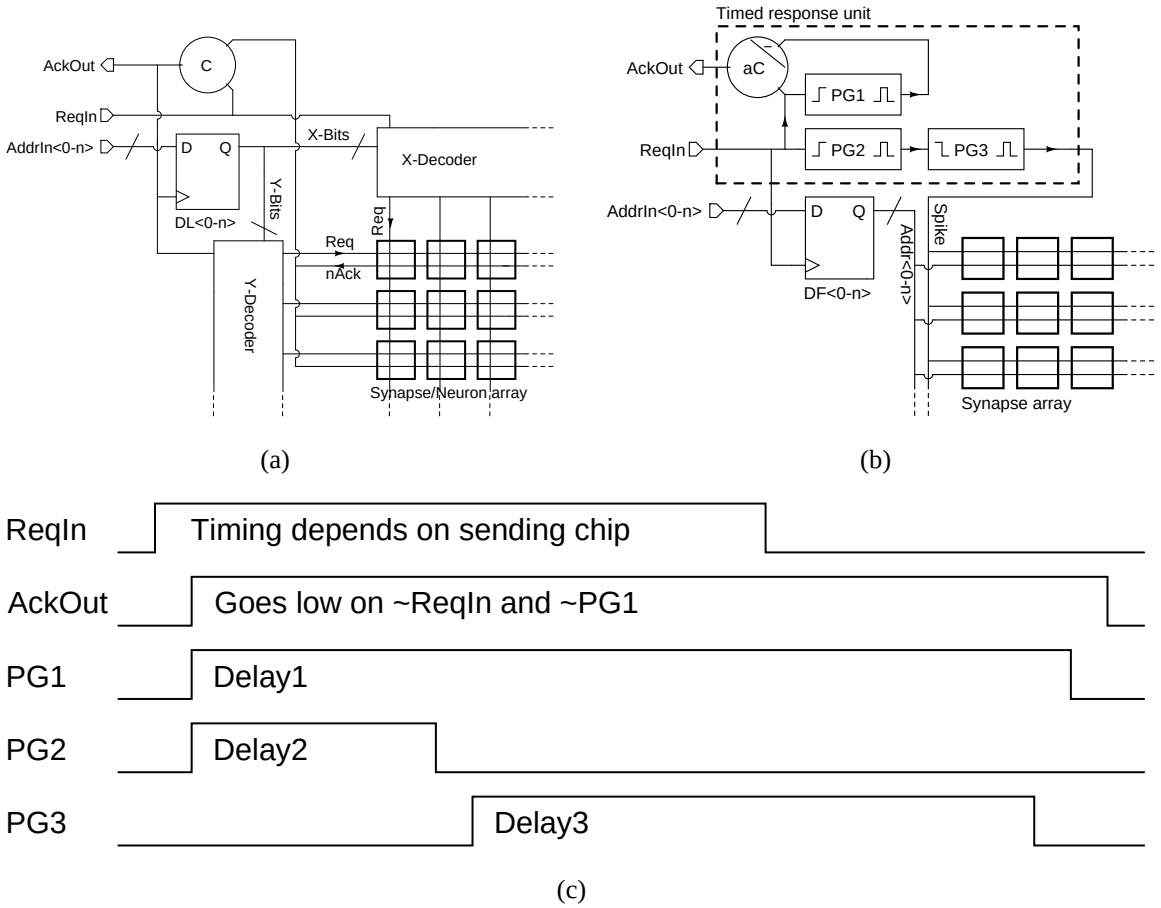


Figure 4.1: (a) Address-event receiver circuitry, functionally equivalent to that described in Boahen [2000]. The incoming request *ReqIn* triggers the raising of the global acknowledge, *AckOut*, and the decoding of the incoming address; a synapse (or neuron) is targeted; when this acknowledges, *AckOut* is lowered (once *ReqIn* has also been lowered), allowing the next event to be transmitted. (b) Proposed address-event receiver. Upon *ReqIn* going high, *AckOut* is immediately driven high, and a pulse generator (*PG1*) is also triggered, the output of which stays high for a precisely-timed (adjustable) period thereafter. *AckOut* stays high until *ReqIn* and *PG1* both drop. *ReqIn* also triggers the local latching of the incoming address bus. Once latched, the address is broadcast across the chip and all synaptic address-event receivers simultaneously compare this address to their own stored address to decide whether it is correct. From the rising of *ReqIn* there is a short delay (implemented by *PG2*) to allow the address data to propagate across the chip, before a pulse (*Spike*) is sent out across the chip (implemented by *PG3*) triggering those synapses with correct addresses to accept the event. The pulse generated by *PG1* is timed to be long enough to accommodate the joint delays of *PG2* and *PG3* before allowing *AckOut* to drop and the cycle to repeat. (c) Example timing diagram for timed response unit.

4.4 Scalability of broadcast approach

4.4.1 Area

Each synapse, in order to implement its address-event receiver, must store as many bits in memory elements as the width of the incoming address bus. The total area of the receiving circuitry across the chip, or across the system, for a multi-chip system, then scales as $S_{max}Nlog_2(N)$, where N is the number of neurons in the system and S_{max} is the maximum fan-in, i.e. the number of dendritic (or incoming) synapses allowed per neuron. The $S_{max}N$ term represents the number of synapse circuits in the system and the $log_2(N)$ term represents the number of bits necessary to encode a neuron's address within each synapse. At first glance this scales poorly compared to the look-up table approach, which employs row and column decoders allowing the area of the receiving circuitry to scale as $\sqrt{S_{max}N}log_2(S_{max}N)$, where the $\sqrt{S_{max}N}$ term represents the number of row or column decoder elements necessary to decode a target synaptic address and the $log_2(S_{max}N)$ term represents the number of bits necessary to encode a synaptic address (each decoder element must store one dimension (i.e. half the bits) of the synaptic addresses it encodes for). Importantly, however, the look-up table approach requires that additional memory external to the neural array is used to store the look-up table (typically on an external memory chip), in which area is required which scales as $S_{av}Nlog_2(S_{max}N)$, where S_{av} is average fan-out. The $S_{av}N$ term is the number of axonal (or outgoing) synapses in the system and the $log_2(S_{max}N)$ term is the number of bits necessary to encode a dendritic (or incoming) synaptic address. The costs of microcontrollers and RAM are not normally considered, whether in terms of chip area or power consumption. This is acceptable for test systems, but if total power budget and space are considered (for a hypothetical implantable system, for example) it can be seen that in the broadcast approach, the chip space necessary to implement memory is simply being distributed throughout the neural array, rather than stored in a separate dedicated chip.

It is also worth noting that the scaling expression above for the on-chip area required by the look-up table approach only holds for a single-chip system. If the system is spread across multiple chips then the expression for the look-up table approach becomes $C\sqrt{S_{max}N_{chip}}log_2(S_{max}N_{chip})$ where C is the number of chips in the system and N_{chip} is the number of neurons per chip. Therefore as a neural network is scaled up by networking together more chips and the ratio of C/N_{chip} goes up, the on-chip area scaling advantage

with respect to the broadcast approach due to row and column decoding ceases to accrue. The scaling expressions above are summarised in table 4.1. Vogelstein's approach [Vogelstein et al., 2007] is also included for comparison; this is included because it is a special case of the look-up table approach in which there is only one target synapse address per target neuron.

Table 4.1: Scaling of area, energy usage and speed

System	On-chip receiver area	Off-chip memory area	Internal buffering energy per spike sent	Time per spike sent
Broadcast	$S_{max}Nlog_2(N)$	none required	$S_{max}Nlog_2(N)$	unity
Look-up table	$C\sqrt{S_{max}N_{chip}}$ $log_2(S_{max}N_{chip})$	$S_{av}Nlog_2(S_{max}N)$	$S_{av}C\sqrt{S_{max}N_{chip}}$	S_{av}
Vogelstein et al. [2007]	$C\sqrt{N_{chip}}$ $log_2(N_{chip})$	$S_{av}Nlog_2(N)$	$S_{av}C\sqrt{N_{chip}}$	S_{av}

N_{chip} = number of neurons per chip;
C = number of chips in system;
N = number of neurons in system = $N_{chip}C$;
S_{max} = maximum fan-in, i.e. number of dendritic synapses allowed per neuron;
S_{av} = average fan-out.

The actual difference in area requirements between these approaches should not be overlooked. Memory on a dedicated RAM chip takes up much less space than in the design presented in this chapter, for three reasons. Firstly it is not integrated with address-bus monitoring circuitry but rather optimised for its purpose. S-RAM cells use just 6 transistors compared to 12 used in the monitor bit design and D-RAM cells can be much smaller again, consisting of just one transistor and one capacitor; in section 4.7 a distributed word-serial address-event receiver is proposed to reduce this drawback. Secondly, dedicated memory can arguably be smaller as it can be implemented in more recent processes with smaller geometry, whilst analogue neurons may need to be implemented with larger geometry to limit mismatch; if homeostatic neural algorithms are implemented, however, such constraints need not apply, as suggested by section 3.5.7. Thirdly dedicated memory is also less costly simply because it is mass-produced; however, whilst chip area is much more expensive on trial ASICs than on mass-produced memory, this may not always be

the case if neuromorphic circuitry comes into mainstream demand. Notwithstanding the possible solutions to these issues, the broadcast approach currently yields synapses of significantly larger on-chip area, resulting in higher production costs for the foreseeable future. However, if this increase of area can be tolerated for a given technology, then it can be tolerated equally both as miniaturisation proceeds and as the size of neural network implemented expands. Meanwhile the broadcast approach can be expected to continue to support larger neural networks with large average fan-outs after the existing approaches run out of "bandwidth".

4.4.2 Energy usage

In the broadcast approach, each incoming address event must be broadcast across the neural array to each synapse. Consequently each synapse contributes a capacitive load to the on-chip buffering and therefore energy consumption will scale linearly with $S_{max}N$. This term includes internal buffering to the neurons but does not include buffering from the pads to the peripheral latches shown in figure 4.1 (b), because the look-up table approach has an equivalent cost. The chips which have been fabricated each contain 2048 synapses and based on the analysis in appendix C therefore use $\approx 1.8nJ$ per incoming spike for internal buffering. This is likely to be comparable to the energy necessary to transfer a spike externally between chips, though as die sizes increase the energy cost of internal buffering can be expected to become increasingly dominant. In the look up table approach there is no need to broadcast the address across the chip; rather the spike signal can be targeted to the row and column of the correct synapse within the neural array. The energy cost of internal buffering should therefore be lower and should scale *per incoming spike* as $C\sqrt{S_{max}N_{chip}}$. In the broadcast approach, however, energy usage remains constant per address-event sent, whilst in the look-up table approach energy usage *per spike sent* increases linearly with axonal fan-out, as each axonal synapse requires a separate spike to be transmitted between chips and the correct synapse targeted. Bearing this in mind, scaling expressions for energy are given in Table 4.1. This suggests that if the choice of which approach to use is to be determined by energy usage then there will be a ratio (ignoring the complexity introduced by multi-chip systems) of S_{av} / $S_{max}N$ above which the broadcast approach can be expected to outperform the look-up table approach. In other words the broadcast approach may perform better in terms of energy for densely connected systems but it will not perform so well for sparsely connected systems. Here,

the additional energy costs of the microcontroller and RAM in the look-up table approach have not been considered; how these scale depends on the implementation.

4.4.3 Time

To ensure that communication succeeds in the broadcast approach, each communication cycle is deliberately slower than the average cycle time which could be achieved if the sender were allowed to proceed with the next event as soon as a synapse acknowledges, as in figure 4.1 (a), though the difference could be no more than a small factor. However the time taken in the broadcast approach does not increase with S_{av} whereas in the look-up table approach it increases linearly (as shown in table 4.1). In fact the scaling of time in the broadcast approach may be above unity for a couple of reasons. Firstly, if S_{max} were increased whilst N_{chip} was held constant within a given technology and for a given synapse design, this would result in a larger chip; therefore the broadcast could take longer, although (a) this could be offset by the use of more powerful buffers, (b) it is more likely in such a case that more chips would be used rather than bigger chips, due to the constraints of yield. Secondly, the time the address receiver takes to compare its stored address with the incoming address will scale as $log_2(N)$, due to the increased number of receiver bits which contribute to the NAND gate. However this is likely to be inconsequential compared to the broadcast time, and so has not been included in table 4.1. In any case, neither of these effects would significantly change the conclusion: the principle advantage of the broadcast approach is that as axonal fan-out increases, a speed advantage accumulates.

4.5 Implementation

4.5.1 Synaptic address-event receiver circuitry

The total area of the synapse scales as the number of bits necessary to encode a neuron's address in the system. It is therefore necessary to make the storage of each bit and its associated circuitry as compact as possible. This has been achieved using a static memory element, with a transmission-gate implementation of an XNOR gate for comparison with the incoming address bit. The result of the comparison contributes to a NAND gate for the whole receiver, the output of which ("nAeCorrect") indicates whether or not the incoming

address is correct. Additional circuits allow for overwriting. The synaptic address receiver circuitry is shown in fig 4.2.

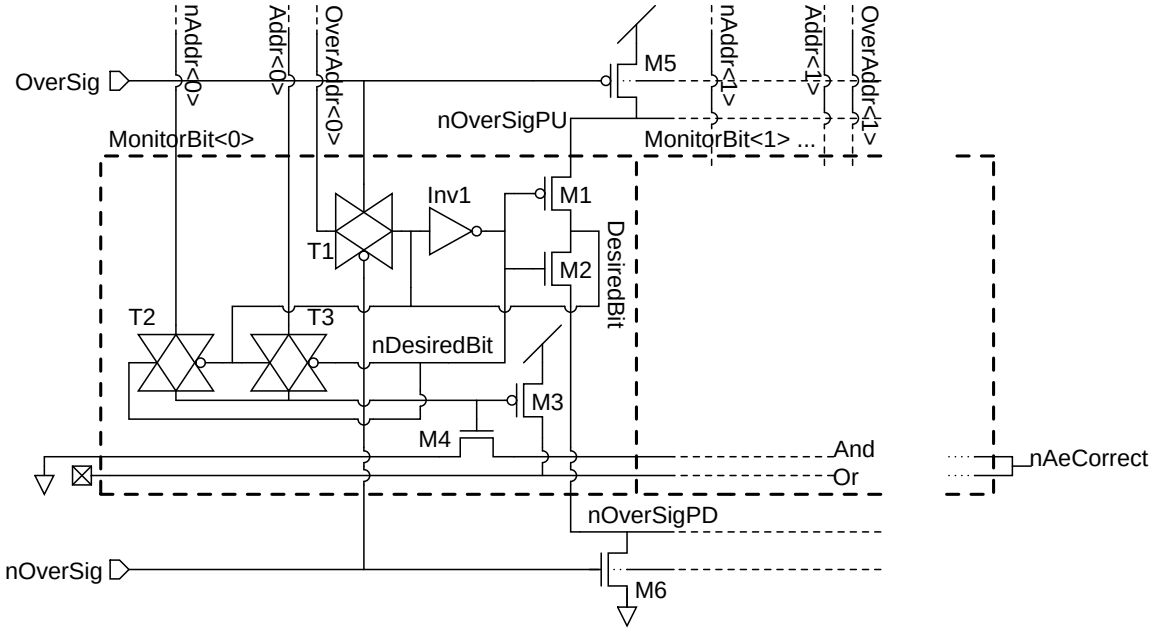


Figure 4.2: Address-event receiver circuitry. The receiver is composed of a chain of blocks each of which monitors a single bit of the incoming address bus; one of these "monitor bits" is shown here (the zeroth bit). A bit of the address (*DesiredBit*) is stored in a memory element composed of Inv1 and M1-M2. An XNOR is continuously performed between *DesiredBit* and the incoming address bit ($Addr<0>$) by means of T2-T3 (the incoming bit's complement $nAddr<0>$ is also required). The result of the XNOR contributes to a NAND gate implemented throughout the receiver array by transistors M3-M4. The result is *nAeCorrect*, indicating whether the full incoming address matches the full stored address. When *OverSig* goes high (and its complement *nOverSig* goes low), this is the signal for the receiver's address to be overwritten with the address on the *OverAddr* bus, a separate bus broadcasting a pre-synaptic address for consideration. *OverSig* chokes off transistors M1-M2 using transistors M5-M6 (these are common for all the monitor bits) while T1 opens, allowing *DesiredBit* to take the value of $OverAddr<0>$.

In the fabricated chip, additional elements were included in this circuit to allow the easy read-out of stored values, which amounted to two additional transistors per monitor bit plus additional circuitry at higher levels. These have been excluded from figure 4.2 for the purpose of clarity. In fact, due to a design error, these circuits did not work correctly and were therefore unused, whilst an alternative approach was used to read out stored values,

as described in appendix H.

4.5.2 Broadcast

The signal labelled *Spike* in figure 4.1 is an active-low pulse, which is labelled *nPrePulseSynCond* in figure 3.4 and is used to generate an increase in synaptic conductance. In fact, two other signals are broadcast across the chip, *nPrePulsePot* and *nPrePulseDep*, which are used to control STDP, as in figure 3.10. Whilst it is not important for these signals to reach all synapses at the same time, it is important for them to last the same length of time at each synapse, since this affects the parameters g_{max}, A_+ and A_-. In the fabricated chip, these signals are generated by pulse generators, buffered to reach each row and then buffered again along each row. In a mature implementation, more attention could be paid to the design of the clock distribution network [Friedman, 2001]. Within each synapse, the result of the address-event receiver is used to decide whether these pulses should be applied to the synapse, as shown in figure 4.3.

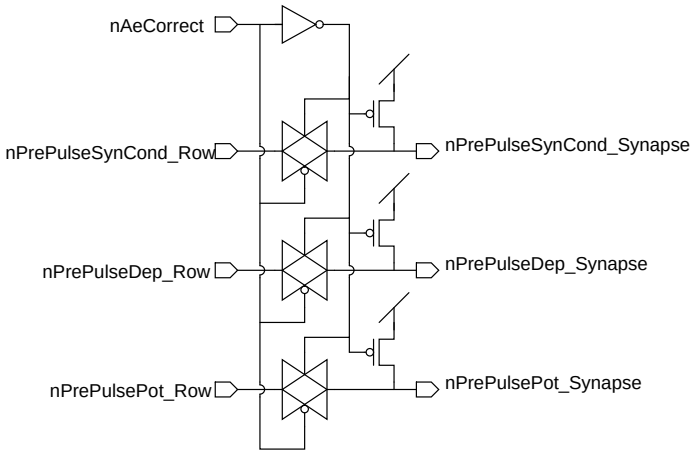


Figure 4.3: Pulse receiver. The signal *nAeCorrect*, which is produced by the address-event receiver, is used to open or close gates for three timed pulses which are broadcast along each row of neurons: *nPrePulseSynCond, nPrePulsePot* and *nPrePulseDep*.

4.5.3 Layout

The chips were fabricated using the AMS 0.35μ 4-metal 2-poly process. Layout considerations are discussed in appendix C.

4.5.4 Multi-chip system

As stated previously in section 3.5, each chip contains an array of $8 \times 4 = 32$ neurons. Each neuron has 64 synapses with reprogrammable 9-bit address-event receivers (there are therefore 2048 synapses per chip). In order to achieve the required number of neurons for a 16×16 layer, 8 chips are therefore required. 8 chips were organised in a grid arrangement [Merolla et al., 2007] so that any neuron on any chip could send or receive spikes (address-events) with negligible delay. Input address-events could be sequenced from a PC and streamed with time stamps to an FPGA (Xilinx Spartan 3 on an Opal Kelly XEM3010 integration module). The FPGA would then transmit the address-events at the correct times (where time was measured in microseconds relative to the beginning of a simulation). Spikes sent by the neurons on the chip were received by the FPGA; these were time-stamped and sent to the PC; they were also optionally merged into the stream of input spikes and sent back to the chips, in order to implement lateral or recurrent connections. The grid system is shown in figure 4.4. Whilst there are alternative schemes which would minimise the number of transmissions necessary for spike delivery, this scheme allows for intervention in recurrent spike delivery should it be required (though this has not been needed in practice).

The grid system used differs from that used by Merolla et al. [2007] in that the addresses used are absolute, not relative. The addressing scheme used in this thesis is as follows. Neurons are referred to by their zero-based coordinates within a layer. For example the top-most left-most neuron in a layer is called neuron Y0X0, whereas the bottom-most right-most neuron is called neuron Y15X15. Where it is not obvious from the context whether the input of target layer is referred to, a neuron can have an "Area" prefix, 0 for the input layer (simulated) and 1 for the target layer (fabricated), e.g. neuron Area1Y7X3. Synapses are referred to by their zero-based index, for example synapse Y7X3Syn63 for the final synapse in neuron Y7X3. Chips are referred to by their 1-based coordinates within the target layer, as can be seen in figure 4.4, for example, the bottom-most right-most chip is called chip Y2X4.

When the system is initialised, each chip must be programmed with its location within the area. This is transmitted to each chip from the FPGA via a cross-chip shift register. It is then latched by the chip and used to form the most significant bits (in each dimension) of the address-events sent by neurons on the chip.

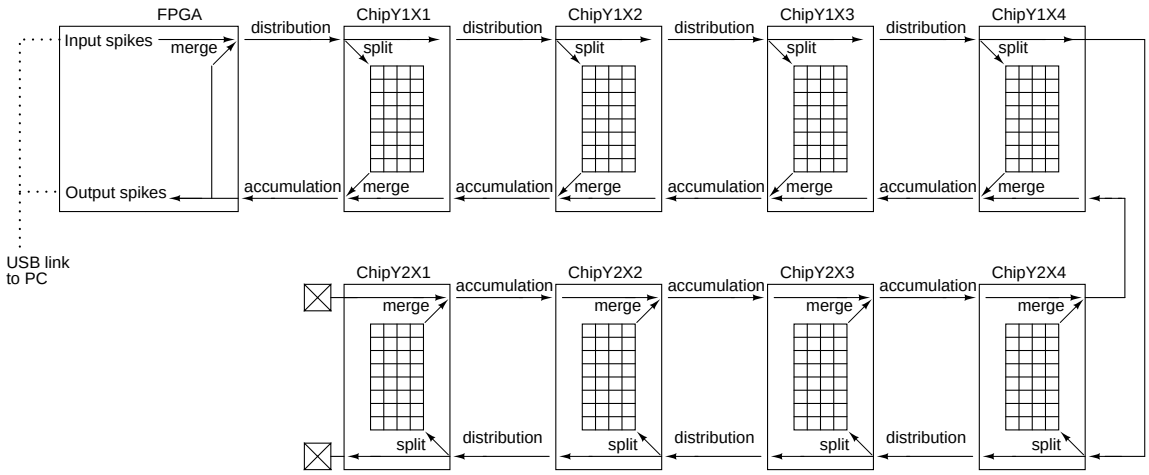


Figure 4.4: Grid system. Timestamped input spikes from the PC are sent at the correct time (or as soon as possible thereafter if the bus is not free) to the first chip (chip Y1X1). The first chip broadcasts this across its neural array and simultaneously (again depending on bus availability) transmits it to the second chip, and so on. The spikes are thus distributed throughout the grid using the chain of buses labelled *distribution*. Spikes generated by neurons in the final chip in the chain, in this case chip Y2X1, are transmitted to the next chip (chip Y2X2). This chip receives those spikes and merges them sequentially, using arbitration circuitry, with any spikes from its own neural array, before transmitting them on. Thus spikes generated from the chips' neurons accumulate along the chain of buses labelled *accumulation*, until they arrive at the FPGA. Here they are timestamped and sent to the PC. They are also, optionally, merged with input spikes and redistributed across the chips, allowing lateral or recurrent synapses to be implemented.

4.6 Results

For the chip results displayed hereafter, the experimental set up is as described is section 3.5. Where simulation results have been given they have been generated by Cadence software, running either the Spectre or the UltraSim simulator and based, where appropriate, on a technology library for the AMS C35B4 process.

4.6.1 Simultaneous receipt of a spike by many synapses

Figure 4.5 gives results which demonstrate the ability of synapses to simultaneously receive the same spike.

4.6.2 Channel capacity

The ability of the distribution chain of buses to distribute spikes is demonstrated in figure 4.6. A burst of spikes from all input neurons was distributed from the FPGA starting at 0s (the first four are shown). Spikes were passed through each chip in the chain with a latency of $\approx 13ns$. The rate at which spikes can be distributed is limited by the total broadcast cycle time as defined by the (programmable) length of the pulse generated by pulse generator PG1 as shown in figure 4.1(c) plus various latencies imposed by the system. The delivery of a spike to a chip and the broadcast within it in the test system took upwards of 60ns. In fact in the test system created, the FPGA contributed the greatest delay; thus the speed achieved was far short of address-event delivery speeds achieved in recent publications (e.g. 41.66MHz Berge and Hafliger [2007]; 78.125MHz [Fasnacht et al., 2008] - two systems for delivery of address-events, which would in fact be compatible with the system presented here, although the rates reported are for network links rather than delivery to end points). Nevertheless, even with spikes being *sent* at only $\approx 4.7MHz$, as the network was configured with an average fan-out of 64, spikes were being *received* at a rate of $\approx 300MHz$. Although it is an unfair comparison, it is true to say that this is faster than any spike delivery rate which has been reported for AER in the neuromorphic engineering literature to date, and increasing the fan-out would increase the spike delivery rate by the same degree. Figure 4.7 shows a burst of spikes generated by the chips in a separate experiment being transmitted along the accumulation chain of buses.

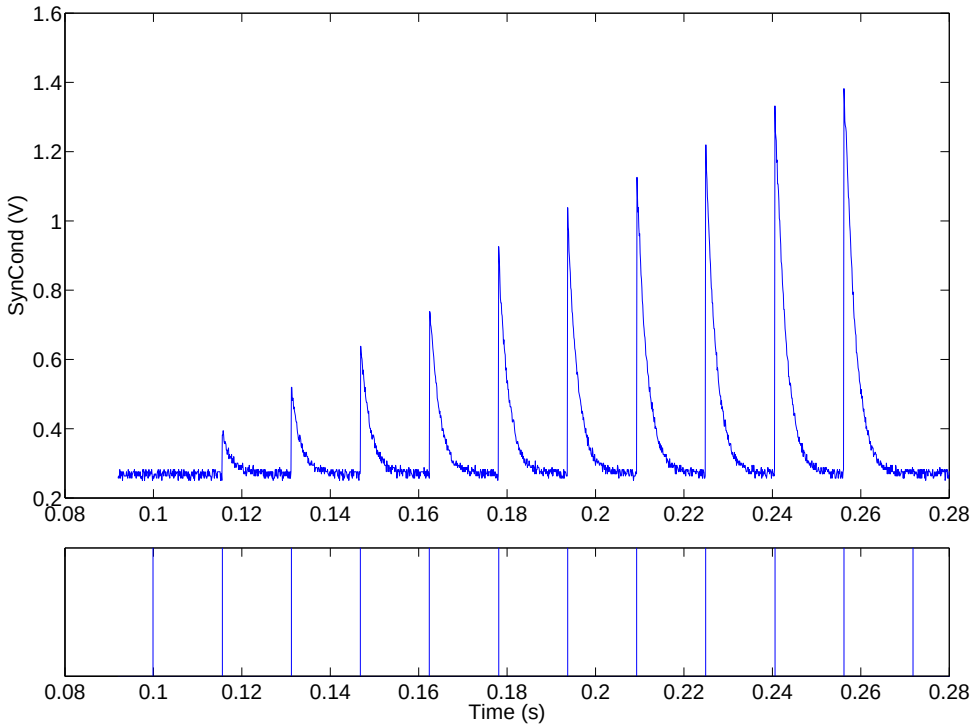


Figure 4.5: Simultaneous receipt of a spike by multiple synapses. Synapses of a neuron were programmed so that Syn0 was receiving from neuron Area0Y0X1, Syn1 and Syn2 were receiving from neuron Area0Y0X2, Syn3, Syn4 and Syn5 were receiving from neuron Area0Y0X3, and so on, with one more synapse receiving from each incrementally higher neuron address, up to Syn45-54, i.e. 10 synapses, which were all receiving from neuron Area0Y0X10. A sequence of spikes was sent in, starting at time 0.1s with frequency of 64Hz. The first spike in the sequence was from neuron Area0Y0X0, the next from neuron Area0Y0X1, and so on. SynCondLeak period was $40\mu s$, implementing a time constant for the decay of SynCond of $\approx 2.5ms$. The upper plot shows SynCond for the neuron whilst the lower plot shows incoming spike times. The first spike did not elicit any response from the neuron since no synapse was programmed to receive from that address. Thereafter each spike caused a progressively (and approximately linearly) larger instantaneous increase in SynCond, as more synapses simultaneously received each subsequent spike.

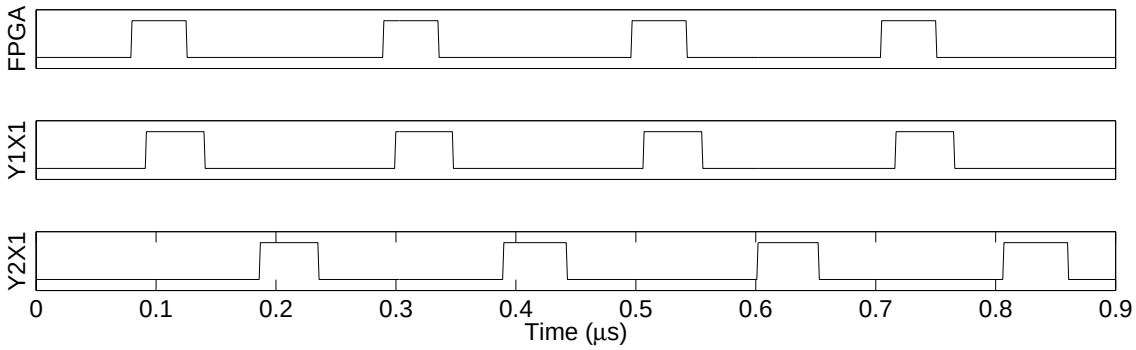


Figure 4.6: Address-events being distributed at maximum speed (with a minimal delay imposed, i.e. delay 1 in figure 4.1(c)). Every synapse was programmed to receive from a randomly selected neuron in the input layer; thus, each input layer neuron had an average fan-out of 64. A set of address-events all with time stamp 0 were distributed from the FPGA as fast as possible. Graphs show the distribution buses' request signals output from: Top - the FPGA; Middle - the first chip in the chain (Y1X1); Bottom - the final chip in the chain (Y2X1). A rate of 4.74MHz was achieved. This includes all delays in the receiving and sending chain in the FPGA, PCB and chips. The largest delay is in the FPGA in this implementation (approx 150ns per cycle). The address-events took 107ns to pass through the grid, about $\approx 13ns$ latency per chip, whereas the receive and broadcast time of a chip was $\approx 60ns$.

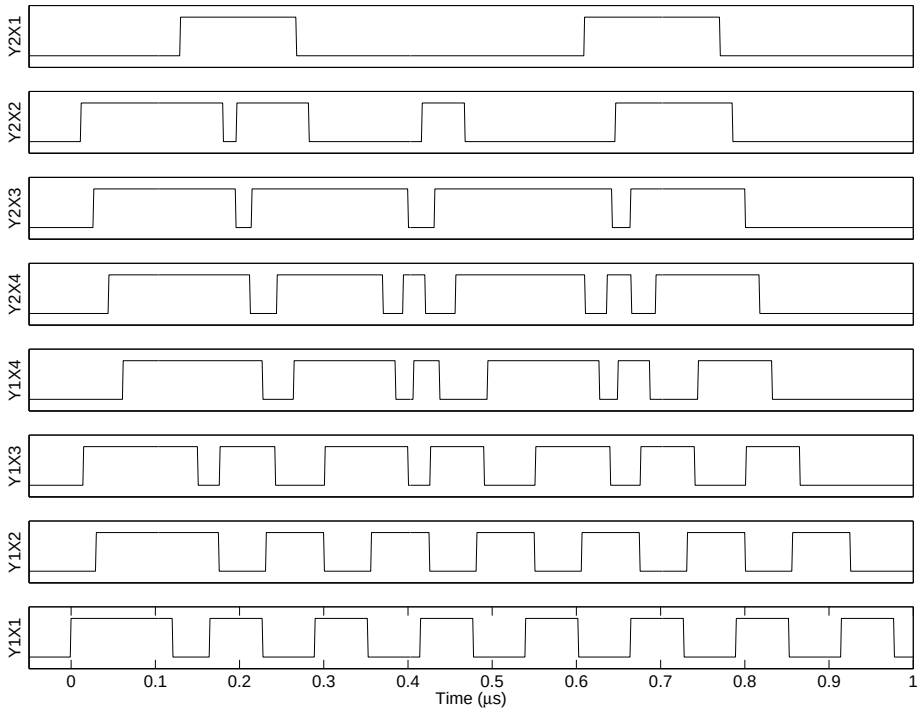


Figure 4.7: Spikes being accumulated. Every synapse was programmed to receive from a randomly selected neuron in the input layer. A burst of spikes was delivered from all input neurons in sequence according to their address, at a rate of 1MHz. The first burst of output spikes generated by the target layer neurons is shown, with time relative to the first request, which came from chip Y1X1. While this request was high, chip Y1X3, further back in the chain, generated a spike at time $\approx 0.02\mu s$. This spike was transmitted through chip Y1X2 and was queued in chip Y1X1 until the handshake for the first spike had completed, whereupon it was transmitted by chip Y1X1, at time $\approx 0.16\mu s$. The next spike, from chip Y2X2, can be seen to be transmitted through most of the chain, and so on.

4.7 Discussion

The address-event receiver which has been implemented redefines synapse circuits as potential synapses and, in a straight-forward manner, shifts the burden of decoding and receiving spike events into them. Alternative designs are possible and may prove beneficial. For example, the adoption of word-serial AER would limit the number of address bus monitoring elements in the synapse and promote the adoption of a more standard choice for a repeating memory element. If standard 6-transistor S-RAM elements were used then the size of the repeating memory element would be $\approx 30\mu m^2$ for the same technology and additional read-out circuitry would not be required. Alternatively a DRAM architecture could reduce the size of the repeating unit to as little as $\approx 2\mu m^2$ for the same technology, though this would probably be unusable in this application since it depends on peripheral circuitry which operates sequentially. As a further alternative, whilst floating gate technology is not best suited to storing synaptic weights because the high frequency of changes usually required by synaptic learning rules would lead to eventual dielectric breakdown, the low rates of synaptic rewiring in natural systems make storage of pre-synaptic addresses on floating gates an attractive option. Analogue storage of many address bits on a single gate (a form of multi-valued logic) could be explored for a possible space saving [Holler et al., 1990].

Adopting a monitor bit design similar to that of standard Content Addressable Memory [Pagiamtzis and Sheikholeslami, 2006] could achieve a space saving, at the expense of a more complex comparison cycle, since it requires that lines which indicate a match are pre-charged before a comparison takes place.

A drawback of the transmission gate implementation of the pulse receiver which was used is that if an unusually large number of synapses within a row are connected to a particular pre-synaptic address, the capacitance on the broadcast signals will increase, possibly affecting the parameters for this set of synapses which depend on pulse duration, e.g. g_{max}, with respect to other synapses in the same row. This could be avoided by using active gates (e.g. NAND gates). More generally, the speed performance of the protocol could be optimised by decoupling the delivery of an address event from the production of currents which drive the synapse's processes, since in general the pulse lengths necessary are longer than the minimum time necessary to simply register a pulse. Such optimisation, however, would require the implementation of pulse generators at each synapse, or alter-

native mechanisms for implementing synaptic processes; a proposal along these lines is given in section 7.2.2. A small, simple improvement to speed performance would be to replace PG1 in figure 4.1(b) with a state machine driven by PG2 and PG3, eliminating one source of delay from the present design.

Alternatively, the look-up table approach could be improved in ways which overcome some of the drawbacks mentioned previously. In a multi-chip system, each chip could have a dedicated microcontroller for receiving address-events and converting them into the target synapse address-events using a dedicated look-up table. Thus at the level of the network, events would be passed as single events which encode the address of their source neuron and only when they neared their destinations would they be expanded to a set of events encoding the addresses of the target synapses. This would reduce the number of events which would have to be passed through the wider network. Individual chips could therefore be sized so that the capacity of this incoming bus was never exceeded. Such a system was created by Deiss et al. [1999], a system which implemented an inter-chip broadcast of address-events. If in addition the microcontroller and look-up table circuitry were integrated into the same die as its associated neural array, then energy and speed savings could be achieved. This approach is being pursued by Shih-Chii Liu (Institute of Neuroinformatics, Zurich, personal communication) and should result in an interesting comparison with the present work.

Khan et al. [2008] have proposed a method of extending the look-up table approach to multi-chip systems. Many chips would be situated in a (toroidal hexagonal) grid by means of two-way communication links. Each chip would have a look-up table and collectively these would define the path or paths by which a packet (essentially an address-event) would have to be routed in order to reach all of the destinations. The look-up table size would be minimised by (a) assuming default routing paths for events (b) masking of parts of neurons' addresses in the routing table to use the same entries for packets with similar addresses and identical destinations (c) encoding a single routing table entry to define multiple routes. As with the above proposal, packets would encode the addresses of source neurons until they reached their target chips (in fact, target *cores*, since each chip would contain multiple processing cores) and only when they arrived would individual destination synapses be targeted. Moreover the number of transmissions of a packet could be minimised with the route branching only as necessary to reach all target chips. Essentially, this system suggests a method by which a network could be expanded beyond the

limits of capacity of a single global bus. Although within cores, computation of the behaviour of synapses and neurons is performed in a digital sequential manner, parallel or analogue neural arrays of the type described here could equally be used within such a network (though see section 5.6 for a discussion of the implications for network rewiring).

A further interesting possibility was proposed by Tuffy et al. [2007], in which neurons would send address-events on a shared wire in the form of a sine wave at a unique frequency. All synapses would then monitor the wire with bandpass filters selective for a particular frequency representing their intended pre-synaptic neuron. This system would have the advantage that neurons could signal concurrently with their oscillating signatures superimposed. However, in order to have a large enough range of frequencies available to represent a suitably large address space, synapses would have to implement compact, high-Q bandpass filters, for which the use of dedicated MEMS elements has been proposed; it remains to be seen whether fabrication difficulties can be overcome (Liam McDaid, University of Ulster, Londonderry, personal communication).

4.8 Conclusions

A design has been presented for an address-event receiver, which is composed of elements which are distributed through the synaptic array and act simultaneously on broadcast address-events. This allows a spike to be received simultaneously by all the synapses on the axonal arbor, allowing for arbitrarily large axonal arbors to be implemented without reducing channel capacity. This receiver is compatible with existing address-event senders. The receiver is reprogrammable during run-time, allowing synaptic rewiring to be implemented; the additional circuitry necessary to make use of this local reprogrammability will be presented in chapter 5. The scalability of this system has been analysed and compared against existing systems with respect to the silicon area, transmission energy and transmission time required, as numbers of neurons and synapses in a system increase. This system scales particularly well in terms of speed as synaptic fan-out increases. Results have been presented from fabricated chips. In particular, spike sending rates have been shown which, when multiplied by the axonal fan-out being implemented, can be interpreted as spike delivery rates which are in excess of those achieved by any published AER-based neuromorphic system to date, demonstrating the potential speed advantage. Alternative designs have been suggested which may further improve the scalability of

such a system.

Chapter 5

Synaptic rewiring and Euclidean distance calculation

5.1 Introduction

According to the title and topic of this thesis, literature is reviewed which relates to neuromorphic approaches to topographic mapping and synaptic rewiring. There is then a presentation of the reasoning behind the approach of storing connectivity information locally within neurons. This approach is used to argue for the location of circuitry for implementing synaptic rewiring within each synapse. Such circuitry is then presented and its functioning demonstrated. In the model of synaptic rewiring adopted, the connection rule requires a calculation of the distance between a neuron and the ideal location of a potential pre-synaptic partner. Consequently, circuitry for Euclidean distance calculation is presented, whose novel features are current-mode operation across multiple chips and the capability of implementing both wrap-around (toroidal) and non-wrap-around topologies. The ability of the rewiring circuitry together with the distance calculation circuitry to allow the formation of radially symmetric receptive fields with arbitrary relationships of connection probability to distance from the centre is then demonstrated.

5.2 Literature review

5.2.1 Neuromorphic approaches to topographic mapping

In this section, there is a discussion of the benefits of physically laying out integrated neuron circuits according to proximity of the biological neurons they are in tended to model; this leads on to an exploration of previous approaches to neuromorphic systems which implement topographic maps.

5.2.1.1 Physical proximity of neuron circuits

In section 4.2.1, the system of Serrano-Gotarredona et al. [1999] was discussed, in which an address-event which arrived at one neuron also influenced a number of nearby neurons, approximating the effect of an axonal arbor. This system made use of the physical proximity of neurons on a chip, analogous to their intended physical proximity in a layer of brain tissue, to simplify its operation, since the separable projective fields it implemented were constructed by a combination of row-wise and column-wise operations. There have been other attempts to implement connectivity by exploiting the physical proximity of neuron circuits; for example, Boahen and Andreou [1992] showed how synapses between neighbouring cells (both directional chemical synapses and non-directional gap junction synapses) could be implemented, using transistors to control the flow of current between nodes in neighbouring cells which represented levels of excitation; wiring for such an arrangement is trivial providing that cells which are neighbouring in the layer of brain (or in this case, retina) being modelled are also physically neighbouring when integrated. Not only does this suggest an alternative approach to implementing synaptic interconnect to those reviewed in section 4.2, it demonstrates how it may be advantageous to place neuron circuits according to the physical locations of the neurons which they are intended to represent. This chapter demonstrates another way in which the layout of neuron circuits according to their intended topography can be utilised.

5.2.1.2 Sensor arrays

In fact, the cells in the aforementioned chip by Boahen and Andreou [1992] were laid out according to the physical arrangement of the modelled biological structure for a more

fundamental reason than easing the wiring of their interconnect; along with a previous "silicon retina" by Mead and Mahowald [1988] and many systems thereafter, the cells in one of the layers implemented represented photoreceptor cones and responded to differing light levels by means of phototransistors; thus layout in a 2D surface upon which incident light could be focused was essential for capturing spatial information from the visual field, as it is in the retina. Other examples of the use of physical layout for sensing include the use of MEMS elements for wind direction sensing by Zhang et al. [2007]. Another system along similar lines used a grid of resistive and amplifying elements to model both the output of hair cells and their interaction with the fluid dynamics of the cochlea [Hamilton et al., 2007], although this was not actually a sensing system (an earlier example of a similar system can be found in [Mead, 1989]).

5.2.1.3 Topographic mapping and receptive fields

The formation of topographic maps was demonstrated in simulation by Elliott and Kramer [2002], based on input from a silicon retina. This was based on a model of neurotrophin release mentioned in section 2.3.2.1 [Elliott and Shadbolt, 1999]). A simplified version of the same model was implemented in silicon by Taba and Boahen [2002]. In this system, spikes from simulated retinal waves caused the release of neurotrophin from their post-synaptic neurons. The level of neurotrophin in the extracellular medium was modelled by the voltage in a charge spreading network, implemented as a single pFET channel laid out as a hexagonal lattice connecting all the nodes across the chip. As neurotrophin was released into the extracellular medium it spread laterally across the area and was gradually taken back up by synapses all across the chip. At any time, therefore, the charge representing neurotrophin level formed gradients across the chip from areas of high activity to areas of low activity. This acted as a cue which allowed reinforcement of neighbourhood relationships to take place. This is another example of a system in which the analogous physical location of circuits representing cells was put to good use. The system of Merolla and Boahen [2003] implemented a model of the spontaneous development of orientation preferences of simple and complex cells again using horizontal charge spreading to implement interconnect. The connectivity of the system was hybrid, since short-range regular interconnect was implemented by direct connection whereas long-range interconnect was implemented over multiplexed channels, by means of AER. The system of Choi et al. [2005] modelled orientation preferences in V1; it also used a combination of directly im-

plemented short-range interconnect and AER for long-range interconnect. It is notable for implementing V1 using multiple chips working together; the chips were designed identically and were each capable of producing neurons with simple Gabor-like receptive fields with either a horizontal or diagonal orientation, but by simple one-to-one arithmetic translation of address-events between chips by dedicated hardware, four chips of the same design could be used to implement four different orientation preferences. There were therefore neurons on each of the chips which were responsive to identical topographic locations but different orientations; these neurons would be intermingled to some extent in V1, so the complete representation of V1 was implemented by a combination of the neural areas on all of the chips. Whilst the local interconnect implemented directly between neighbouring cells was used to approximate the effect of feed-forward axonal arbors, local (inhibitory) feedback between neurons in the same topographic area was implemented through AER. The system presented in this thesis uses AER for all interconnect (see section 4.5.4); even when a neuron forms a recurrent synapse with itself, the spikes from the neuron are transmitted as address events and broadcast back to all neurons so that the appropriate synapse circuits can receive them, including those of the self-same neuron. Although there are some assumptions about network topology embedded in the chip design, these are not in the form of directly implemented local interconnect; rather the method used to form topographic mappings is explained below in sections 5.3-5.4.

5.2.2 Synaptic rewiring in neuromorphic systems

Some VLSI neural networks with reconfigurable topology have been mentioned previously in section 4.2. For example Mueller et al. [1989] and Satyanarayana et al. [1992] created systems in which neurons were connected to each other's synapses on the same chip, via matrices or chains of switches which could be reprogrammed by an external mechanism. Thus, although for any one topology there was a dedicated wire for each connection, different topologies could be achieved using the same sets of wires. These two systems serve to illustrate a couple of basic points about rewiring. Firstly, both of these systems implemented synapses with weights (which were also programmable). Thus the practical distinction between synaptic weight change and synaptic topology change discussed in section 2.1.1.4 is respected. It would be possible to implement functionally equivalent neural networks using all-to-all connectivity and setting the weights of unwanted synapses to zero, but for a hardware implementation of sparsely connected neural

networks this would be an extremely inefficient use of silicon area. Secondly, these two systems both offer reconfigurability of the synaptic connections, but by a seemingly inconsequential difference in implementation, the system of Mueller et al. [1989] would be better able to offer run-time reconfiguration than the system of Satyanarayana et al. [1992]. This is because the switches in the former could be independently targeted for reprogramming whilst those in the latter could only be reprogrammed by passing a string of data in for all the switches; thus in the latter, in order to change one connection, the activity in the whole system would be disrupted at least temporarily (though this may be very fast; this ability is not in fact important for the MLP-like systems being implemented). Run-time reconfiguration is a requirement for implementing a model of synaptic rewiring, since in the brain synapses are formed and eliminated whilst others continue to function.

For systems which use AER, with network topology implemented by re-routing address-events with digital look-up tables, run-time reconfigurability is possible providing only that the look-up tables are stored in memory which can be selectively overwritten. Such an approach was used in the aforementioned system of Taba and Boahen [2002] to implement a model of topographic map formation. The model was based on biological synaptic rewiring, although it contained a number of simplifying assumptions. Axons would follow gradients of neurotrophin; if there was a higher level of neurotrophin in a neighbouring area the axon would eliminate its current synaptic connection and immediately form a new one in that neighbouring location; this would displace the axon currently occupying that synapse, which would immediately form a new synapse in the location that had just been vacated by the invading axon. This was implemented by swapping the target synapse addresses associated with the two axons in the look-up table. The neurotrophin gradient calculation was carried out locally to the synapse and then the intention to swap locations was transmitted to the microcontroller using a specialised address-event. The system presented in sections 5.3-5.4 also has a mechanism for generating an intention to rewire which is local to the synapse, but in addition, the entire rewiring process is completed locally to the synapse.

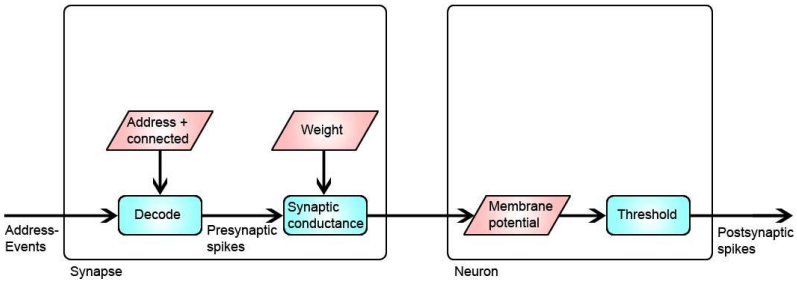
5.3 Rationale

Figure 5.1 presents a series of flow charts which demonstrate the reasoning for implementing synaptic functions locally. Figure 5.1 (a) shows the basic operation of an integrate-

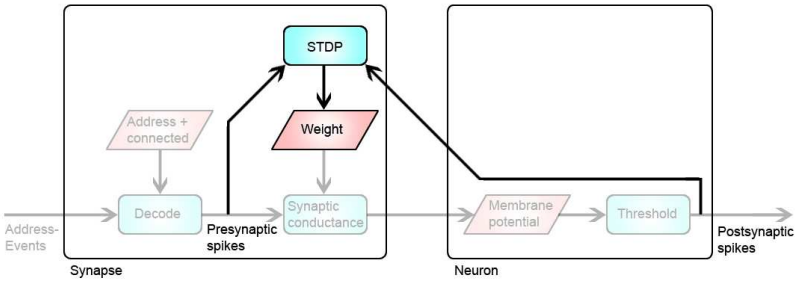
and-fire neuron. For simplicity, the neuron is shown with only one dendritic synapse. Address-events which are broadcast across the chip are received by the receiver circuitry presented in chapter 4. As well as storing the address of the pre-synaptic neuron, the synapse also stores one bit which indicates whether it is connected, and only if this bit is set will any address-event be received. If an address-event is received, the resulting pulse is used to create a synaptic conductance. This process is modified by the weight of the synapse, which, as indicated, is also stored locally in the synapse. The synaptic conductance affects the membrane potential, which is stored as a charge on a capacitor in the circuitry for the central functions of the neuron. If this potential passes a threshold then an outgoing spike is generated. These processes are implemented by the circuitry presented in chapter 3.

Figure 5.1 (b) introduces the additional function of STDP. This requires information about the pre-synaptic and post-synaptic spikes (specifically it requires their relative timings), and it affects the weight of the synapse. Since pre- and post-synaptic spikes are necessarily available locally, and since synaptic weight is stored locally, there is an advantage if the STDP function can be implemented locally, rather than, say, in the periphery of the chip or by an off-chip mechanism, since there does not need to be any long-distance communication, with the overheads of energy and possibly time that this would imply. Circuitry which can implement STDP, which is amenable to relatively compact integration within a synapse circuit has been presented in section 3.4.4.

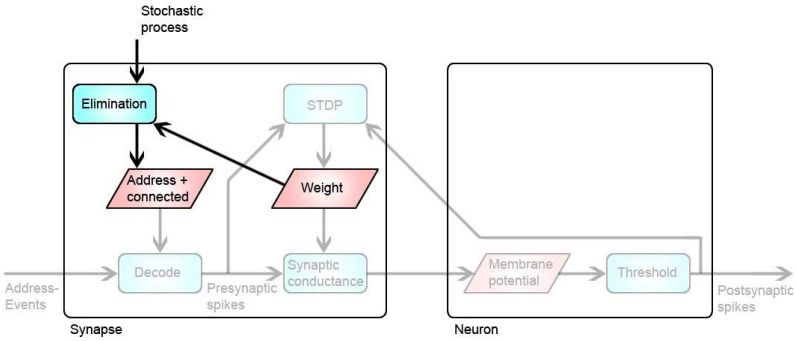
The approach of implementing functions locally to the synapse and neuron is then pursued towards its logical conclusion with the implementation of the synaptic rewiring rules presented in chapter 2. Figure 5.1 (c) introduces the synapse elimination rule. This requires the weight information, as this affects the probability of elimination. As it is probabilistic it also needs a stochastic process as input. If elimination occurs, the effect is to reset the bit which encodes the connectedness of the synapse. Information is therefore available locally, with the possible exception of a stochastic process. Finally, figure 5.1 (d) introduces the synapse formation process. This also requires a stochastic input. Additionally it requires a randomly chosen potential pre-synaptic partner; this information is not shown as being generated locally; it must be transmitted to the synapse in at least some cases, since it is used to change the address stored in the synapse if formation is successful, therefore the means to transmit this address to the synapse must exist (the possibility of using incoming address-events as an available source of potential pre-synaptic partners is dis-



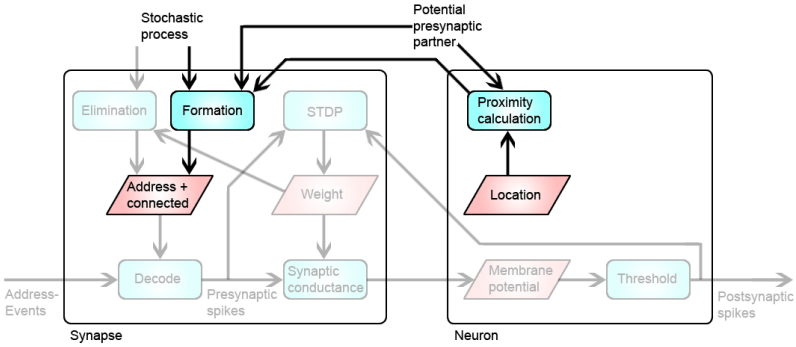
(a) Basic integrate-and-fire operation



(b) STDP



(c) Synapse elimination



(d) Synapse formation

Figure 5.1: Flow charts demonstrating the functions of a neuron. Explanation in text.

cussed in section 5.4.1). The probability of formation is influenced by the proximity of the ideal location of the pre-synaptic neuron to the post-synaptic neuron's location. Location is a property of the neuron and therefore circuitry for calculating proximity is localised in the circuitry for the neuron's central processes. In fact, rather than explicitly encoding the location of each neuron, the location is implied by the neuron circuit's physical location on the chip, as will be seen in section 5.4.2.

5.4 Circuitry

In this section, circuitry necessary for implementing the formation and elimination rules is described.

5.4.1 Synaptic rewiring circuitry

Synapses can be individually targeted for rewiring by a chip-wide mechanism, which employs row and column decoders in the periphery. This allows for both the explicit setting of synaptic variables from an off-chip control mechanism, and for ongoing probabilistic rewiring.

The circuitry which implements the connection and disconnection algorithm is shown in fig 5.2. When a synapse is selected as a candidate for rewiring, its behaviour depends on its state of connectedness, stored in a static memory element. If it is connected then it is considered for disconnection. Its analogue weight voltage (*nWeight* in figure 3.10) is compared to a voltage *nProbDisconnect*, randomly chosen according to a probabilistic distribution. If the weight is below the random value then the synapse is disconnected. The random value is common for all the synapses on the chip but is only used at one synapse at a time and changes between each usage, avoiding the possibility of correlation between synapses. In this implementation the voltage is produced off-chip, but it could be produced on-chip by a random number generator and a DAC in a mature implementation. By changing the probability distribution of *nProbDisconnect*, different relationships of weight to probability of elimination can be implemented, for example, a thresholded rule, as used in this project, or alternatively a linear interpolation between high and low probabilities, etc.

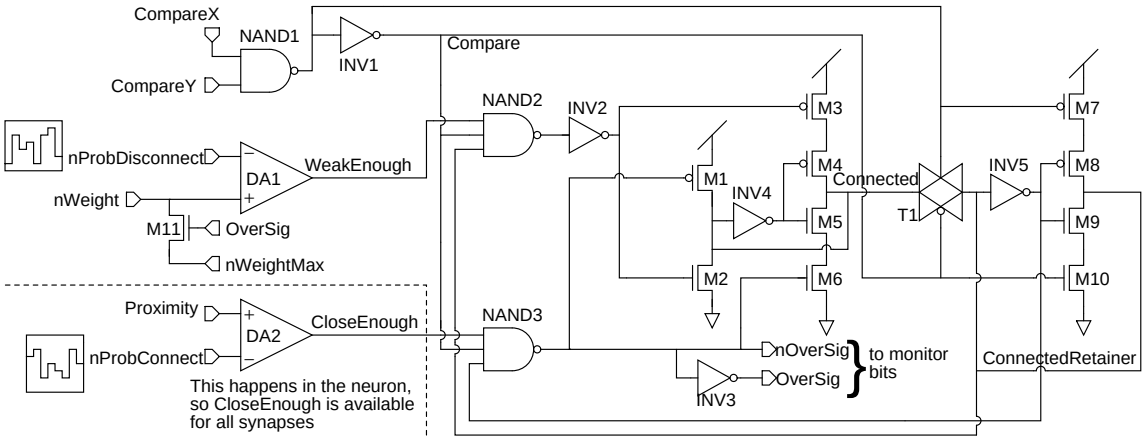


Figure 5.2: Circuitry for synaptic rewiring. The synapse's *Connected* state is stored in a memory element composed of INV4 and M4-M5. This state can be overridden by a disconnection signal from NAND2 and INV2, using M2-3, or by a connection signal from NAND3 using M1 and M6. Compare is driven high by the targeted conjunction of the *CompareX* and *CompareY* signals from row and column decoders, to indicate that rewiring is under consideration. While *Compare* is high, the *Connected* state is latched in a separate memory element *ConnectedRetainer* (INV5 and M8-M9). This ensures that only connection or disconnection can occur, avoiding oscillations during the *Compare* signal. The complete condition for connection is that (a) *Proximity* is above the random value *nProbConnect* such that the neuron is judged to be *CloseEnough* (the output of differential amplifier DA2) to the potential pre-synaptic partner; (b) the *Compare* signal is high; and (c) the synapse is not currently connected i.e. *ConnectedRetainer* is low. On connection, the override signal *Oversig* and its complement are sent to the address-event receiver, allowing the address under consideration to override the address stored in the monitor bits; *nWeight* is also set to its strongest value, by M11. A complementary set of conditions apply to disconnection, the first being that *nWeight* is above *nProbDisconnect*.

If the synapse is disconnected and it is then selected as a candidate for rewiring, the possibility of it taking a new pre-synaptic partner is considered. The pre-synaptic partner to be considered is randomly chosen, differing from the simulations in chapter 2 where the last neuron to have fired was chosen. The method of choosing the last neuron to have fired would be attractive as it could further reduce the amount of communication required, since it is an existing source of random addresses which is already available at the synapse. However, in practice this was not used, both due to practical difficulties in implementation, and to allow rewiring to be explicitly controlled where necessary without interrupting spiking input. A consequence of the difference between the two methods is that if the last address-event was used, neurons which fire more would end up with larger axonal arbors due to having more opportunities to form synapses. However since in simulations efforts were made to balance the firing rates between areas, and since in a mature implementation additional homeostatic processes might be expected to keep neural firing rates in broad agreement, these differences are likely to be slight and have not been investigated in this project.

The randomly chosen synapse addresses come from off-chip in the test implementation but they could come from an on-chip pseudo-random-number generator in a mature implementation. The potential pre-synaptic partner is latched separately by each chip and is broadcast across all chips at the point that a rewiring consideration takes place, using a different address bus to the one used to transmit spikes. This allows a cross-chip calculation to take place, providing a value, which can be made available at any one neuron, of the geometric proximity of that neuron to the incoming address. The synapse under consideration then compares this proximity value to a random value *nProbConnect*, similar to the random value for disconnection but separate, created according to a probabilistic distribution for synapse formation. If the proximity value is higher than the random value then the synapse becomes connected and it adopts the broadcast address of the potential pre-synaptic partner in consideration as its new stored address.

By changing the probability distribution of *nProbConnect*, differently shaped connection fields can be created, for example, Gaussian connection fields as used in the model in chapter 2, or other shapes such as cylinders or cones. In fact any radially symmetric shape where the height (or probability of connection) decreases monotonically with distance from the centre can be created; the example of an isodensitic bounded (i.e. cylindrical) receptive field is given in section 5.5.3.

The key parameters from the model in chapter 2 which influence synapse formation are $\sigma_{form-feedforward}$, $P_{form-feedforward}$, $\sigma_{form-lateral}$ and $P_{form-lateral}$. As the standard deviation and peak formation probability changes depending on the area from which the potential pre-synaptic partner is chosen (i.e. whether the projection is feed-forward or lateral), *nProbConnect* must be chosen from a different probabilistic distribution depending on this area. For this test system, the potential pre-synaptic partner addresses are generated off-chip alongside the values for *nProbConnect*. However, in order to demonstrate one way in which this circuitry may be generalised, values of *nProbConnect* are generated for each of two distributions, *nProbConnectFF* for feed-forward connections and *nProbConnectLat* for lateral connections; peripheral circuitry on the chip then selects the correct value to broadcast to the neurons based on the part of the address of the potential pre-synaptic partner which indicates its area (one bit in this case, since the test system has been designed to allow only two incoming projections).

In the model under consideration there is a strong topographic mapping between successive neural layers, which applies to both the feed-forward projection and to the lateral projection, but this assumption is not essential to the system described here. The effect of proximity on the probability of rewiring can be eliminated altogether if it is not required, by reducing the distribution of *nProbConnect* to a binary choice between an extremely high value, where the synapse will not connect no matter how high the proximity, and an extremely low value, where the synapse will definitely connect regardless of proximity. This fact can be used for directly controlling rewiring where necessary; this will be demonstrated in section 5.5.1.

5.4.2 Euclidean Distance Circuit

5.4.2.1 Basic circuit

Euclidean distance calculation is based on the known principle of using the squared V-I relationship of saturated MOSFETs in strong inversion [Cilingiroglu and Aksin, 1998]. The circuit which calculates Euclidean distance is presented in figure 5.3 and its functioning is described in detail here.

(A) shows the layout of chip. A 2D array of neurons contains a cell, marked *T* for target, which contains a synapse which is not currently connected and has been randomly selected to carry out its rewiring rule. A pre-synaptic neuron is also randomly selected, whose ideal

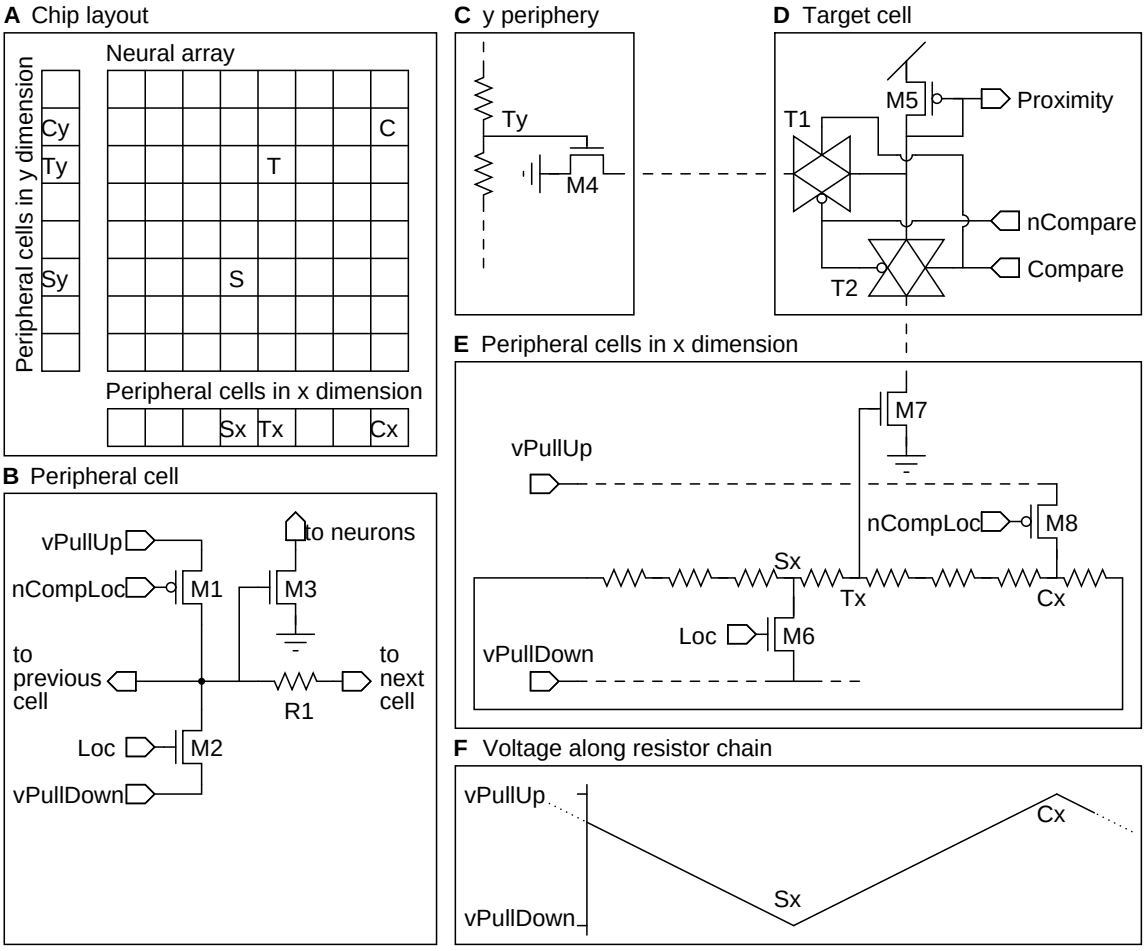


Figure 5.3: Euclidean distance calculation circuit (single chip, toroidal topology). (A) Layout of the chip; (B) circuit for each peripheral cell; (C) a section of the circuit along the periphery in the Y dimension; (D) circuit inside each neuron; (E) circuit along the periphery in the X dimension; (F) idealised voltage along the chain of resistors in (E).

location in this neural area is marked *S* for source. These locations are transmitted to the chip (or simultaneously to all the chips in a multi-chip system) and decoded in peripheral row and column decoders (not shown). The complementary location, marked *C*, is the cell which is the furthest away from the ideal location (assuming a wrap-around topology). Along two edges of the chip are a row and column of identical peripheral cells. Peripheral cells corresponding to the target, ideal and complementary locations are marked *Tx/y, Sx/y* and *Cx/y* respectively.

(B) shows the circuit within each peripheral cell. A central node in each cell is connected to the central node of its two neighbours via a resistor R1. In the Sx and Sy, the *Loc* signal is raised by the row and column decoders, switching on M2 so that the central node is pulled down to a voltage reference *vPullDown*. Likewise, in Cx and Cy, the *~CompLoc* signal is lowered, switching on M1 so that the central node is pulled up to the voltage *vPullUp*. The central node gates M3, whose source is at *Gnd* and whose drain is at the end of a wire which spans across the chip and is available to all neurons in the row or column corresponding to the peripheral cell.

(E) shows the circuit along one edge of the chip. All of the central nodes for each peripheral cell connect via resistors to form a chain of resistors which travels along the edge of the chip and then wraps around at the edge. For simplicity, the only transistors shown are those which would be active in the case shown in (A). (F) gives a graph of voltage along the chain of resistors (as the number of cells tends to infinity). At the ideal location *Sx*, the voltage is at *vPullDown*. It then rises linearly in each direction, reaching a maximum of *vPullUp* at the complementary location *Cx*. The voltage on the resistor chain therefore represents the distance (in one dimension) from the ideal location, on a linear scale from *vPullDown* to *vPullUp*. *vPullDown* is set to be approximately the threshold voltage of the nMOSFETs gated by the nodes of the chain of resistors (i.e. M3 in (B)). If these nMOSFETs are saturated, the currents through them will be proportional (to first order approximation) to the square of the distance of their node away from the ideal location. There is also a circuit, which is equivalent to that shown in (E), along the vertical edge of the chip, as indicated by (C) (for simplicity, only the transistor which is active in this case is shown).

(D) shows the circuitry inside the target cell where the synapse is to carry out its rewiring rule. Two transmission gates T1-T2 are opened and the currents are allowed to flow through the nMOSFETs in the two corresponding peripheral cells, M4 and M7. These

currents join together to travel through a single diode-connected pMOSFET M5. Providing that M4, M7 and M5 all stay in saturation, and providing that only one cell has its transmission gates opened at one time, the voltage created at the gate of M5 is proportional to the square-root of the sum of the square of the distance from the ideal location of the potential pre-synaptic neuron to the targeted post-synaptic neuron in each dimension, which is the Euclidean distance between them. (Note that the term "proximity" is used in preference to "distance" simply because the circuit produces a decreasing voltage value as distance increases, thus proximity is more intuitive.)

To summarise then, an analogue current-mode circuit uses the squared V-I relationship of saturated transistors in strong inversion to directly implement a calculation of Euclidean distance. As transistors are operated in strong inversion the currents required are on the order of tens of microamps, but these currents need only flow for a few μs while a calculation is being carried out; given the rate at which synapses rewire in biology, the duty cycle of this circuit *per synapse* can therefore be extremely small. During a calculation, three currents flow, one through each resistor chain and one through the pMOSFET. Suitable sizing of the resistors and the nMOSFETS respectively can limit the magnitudes of these currents. The circuitry inside each neuron is rather compact, just one transistor connected by two transmission gates. (It would be possible to further refine the circuit to use single transistors in place of each of the transmission gates if strict assumptions are made about the possible range of *vPullDown* to *vPullUp*). The circuitry in the peripheral cells is rather less compact, partly because of the need to integrate resistor components with suitably large resistances and partly because longer nMOSFETs result in smaller currents. However, the area required by the peripheral circuitry scales as $C\sqrt{N_{chip}}$ (according to the definitions in section 4.4) therefore the area required becomes increasingly irrelevant as the number of neurons integrated on each chip increases.

5.4.2.2 Circuit for non-toroidal topology

A toroidal topology has been used in the model presented in chapter 2 for mathematical convenience. Biological topographic maps, however, typically do not have a toroidal topology. (Interestingly there is no reason in principle why they could not; some animals, for example the rabbit, have almost 360° vision, auditory maps, at least in the barn owl cover all directions with respect to the head, and most animals' somatic sensorium extends to all surfaces of their bodies; topographic maps with toroidal topology might therefore

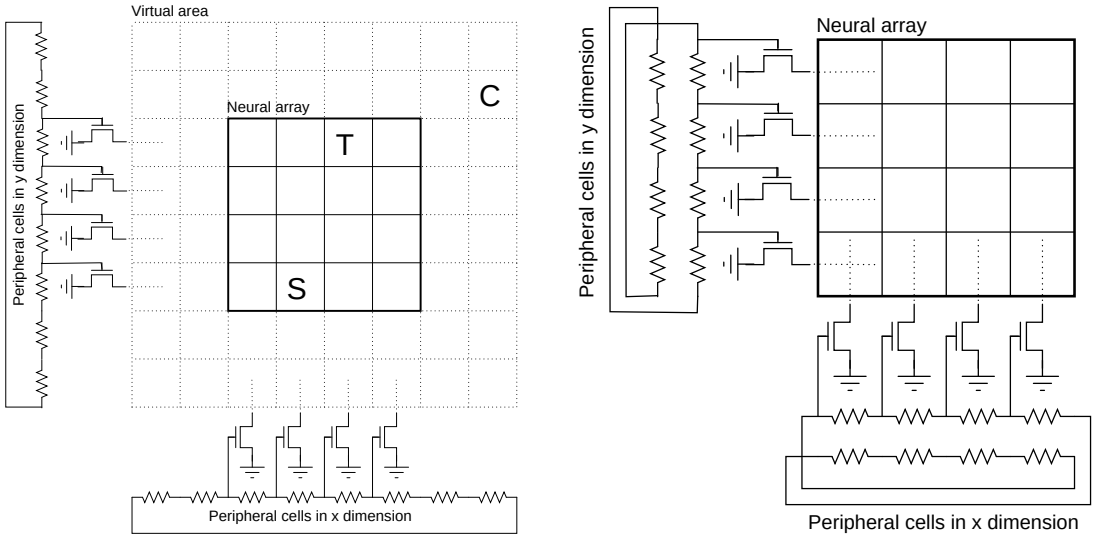


Figure 5.4: Proximity circuit for non-toroidal topology. The circuit is shown for a chip with a 4×4 neural array. (a) The actual neural array is extended into a virtual space twice as large in each dimension, by extension of the peripheral chain of resistors in each dimension. The source *S* and target *T* locations are always in the actual neural array, whereas the complement *C* of the source location is always in the virtual area; thus voltage on the resistor chain always increases away from S towards the edges of the chip in each dimension. Calculation then proceeds as for the circuit presented in figure 5.3. (b) For compact implementation, each peripheral cell contains the nodes and resistors for the corresponding source location and its virtual complementary location.

be useful for implementing sensory processing, although the evidence from the animal kingdom is that such innovation is not necessary). To implement non-toroidal topologies (as has been done in most models of topographic map formation, e.g. Goodhill 1993) an alternative version of this circuit is required. The circuit is shown in figure 5.4.

PThe fabricated chips contain circuits for both toroidal and non-toroidal topologies and these circuits share the same wires across the neural array and transistors within the neurons. That is, there is only duplication of circuitry in the peripheral cells.

5.4.2.3 Multi-chip circuit

The need for an array of chips to implement a suitable neural area has been described in 4.5.4. The circuit which has been presented above can be easily extended to multi-chip

systems. The wiring scheme is shown in figure 5.5 (for simplicity, only the circuit for toroidal topology is shown.)

It can be seen that wires pass between chips in order to implement the chains of resistors, and that there is one chain of resistors for each row and each column of chips. As the complete circuit is implemented over different dies, process variation between the dies may affect the performance of the circuit more than would be expected within a single die. The effect of mismatch will be considered in section 5.5 below.

5.5 Results

5.5.1 Synaptic rewiring

A demonstration of the ability of a neuron to rewire one of its synapses is shown in fig 5.6.

5.5.2 Proximity values

In order to parameterise the circuit with values for *vPullDown* and *vPullUp*, an experiment was carried out, the results of which are shown in figure 5.7(a). For values for the gate voltage of transistor M3 in figure 5.3 in the range $\approx 0.4V$ up to $\approx 2V$, *Proximity* falls approximately linearly from its maximum level of $\approx 2.6V$. With the gate voltage above $\approx 2V$, the rate of change of *Proximity* reduces as the nMOSFETs go out of saturation. 0.4V is therefore a good value for *vPullDown*. For linear performance across the entire range, 2V would be a good value for vPullUp; however, by extending *vPullUp* further into the non-linear region for *Proximity*, the total range of *Proximity* can be extended, allowing greater accuracy in the comparison with *nProbConnect* for high proximities, whilst incorrect Euclidean distance calculations will only occur for pre-synaptic neurons whose ideal location is far from the post-synaptic neuron. For the intended use of this circuit to create a Gaussian profile, such neurons will rarely be chosen as pre-synaptic partners. Thus for most simulations *vPullUp* has been extended to 2.5V to maximise accuracy where it is most required.

Figure 5.7(b-c) shows the results of the cross-chip *Proximity* calculation. (b) gives mean results from neurons on each of 8 different chips whilst (c) shows these results separately to give some indication of the effects of mismatch. In (c), each data set individually

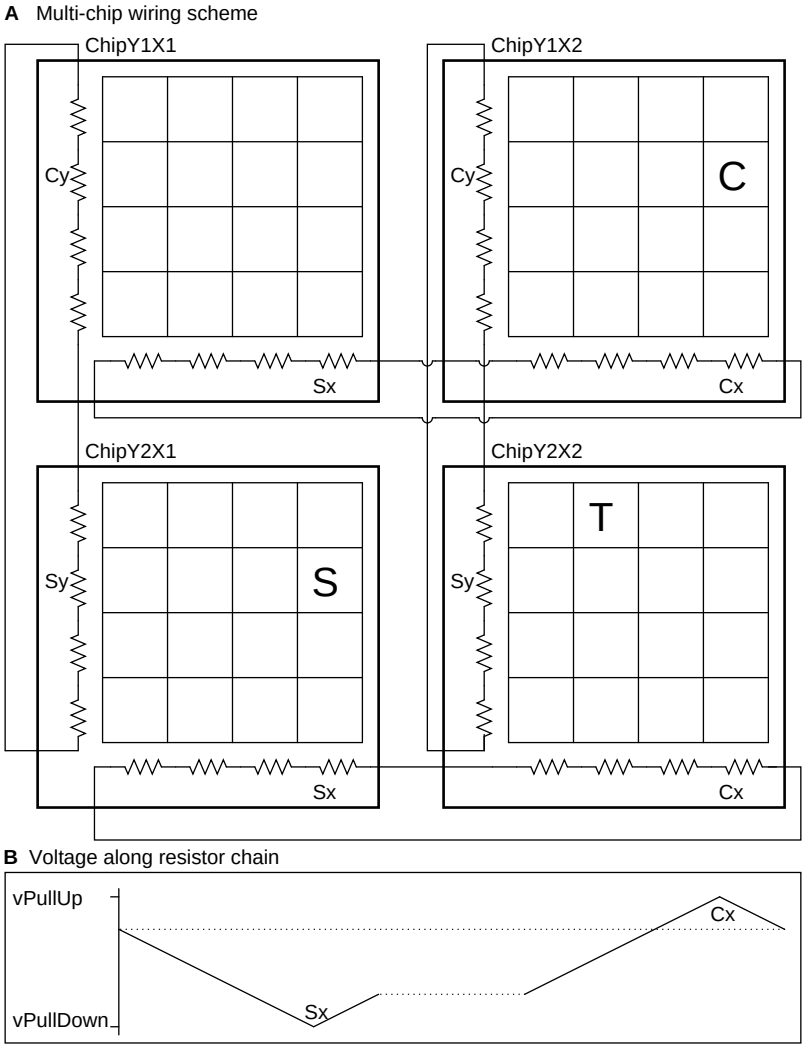


Figure 5.5: Proximity circuit for multi-chip system, shown for a 2×2 array of chips each with a 4×4 array of neurons. (A) layout of chips. The source *S* and target *T* locations can be anywhere on any of the chips, whereas the complement *C* to the source location is guaranteed to be on a different chip than *S*. There is one complete row of resistors for each row of chips and likewise for columns. In all other respects, the circuit functions as with the circuit presented in figure 5.3. (B) The idealised voltage profile along the chains of the resistors in the X dimension is shown for the case illustrated in (A). Voltage rises across across the neural arrays from *vPullDown* at *Sx* on the left chips up to *vPullUp* at *Cx* on the right chips. Dotted lines represent the voltage of resistor chains at links between chips.

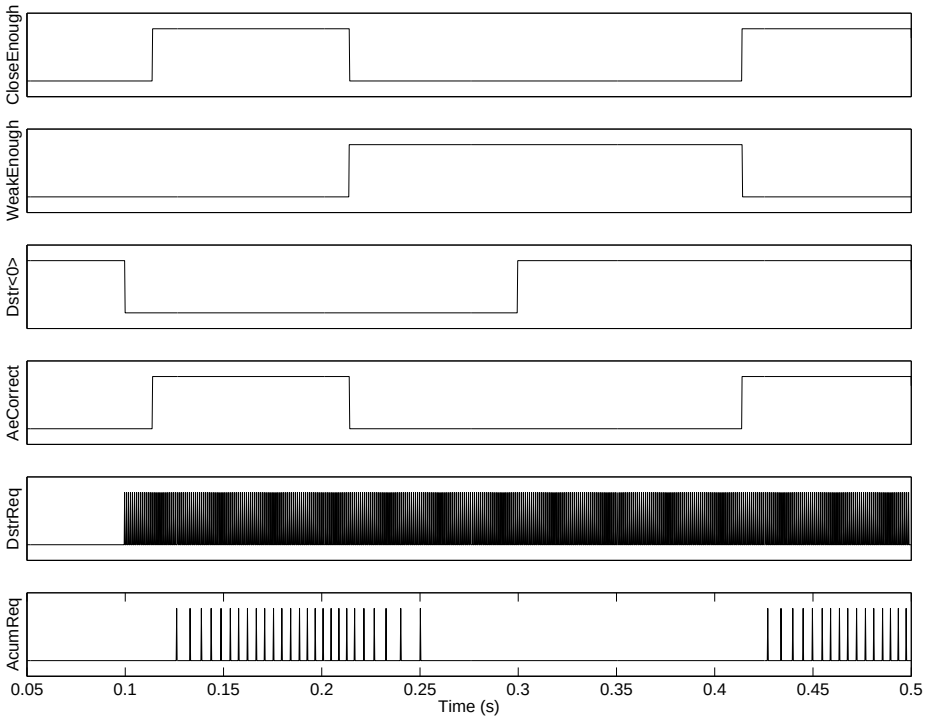


Figure 5.6: Trace showing the rewiring of a synapse. At the beginning of the simulation all synapses were disconnected. Shortly after time 0.1s, a rewiring event was sent to synapse Y15X15Syn63 (such events were sent periodically at 10KHz thereafter) where the pre-synaptic address was Area0Y0X0 and *nProbConnect* was 0V, i.e. connection was guaranteed. *CloseEnough* (for neuron Y15X15) rose, permitting connection. Spikes were sent from neuron Area0Y0X0 starting at time 0.1s with frequency 1KHz, continuing to time 0.3s, whereupon the spikes were sent instead from neuron Area0Y0X1 (the change can be seen from the LSB on the distribution bus *Dstr<0>*; the spikes can be seen from the request signal *DstrReq*). Immediately after the connection of the synapse, the *AeCorrect* signal from the synapse rose, showing that the incoming address on the bus was being received, and shortly thereafter the neuron started to produce a string of output spikes, (see the request signal of the accumulation bus *AcumReq);* depression was disabled. Shortly after time 0.2s, rewiring events were sent to the same synapse with *nProbConnect=3.3V* and *nProbDisconnect=0V*, i.e. disconnection was guaranteed. *CloseEnough* dropped while *WeakEnough* raised, the synapse became disconnected, *AeCorrect* dropped and the neuron stopped emitting spikes shortly thereafter. Shortly after time 0.4s, rewiring events were sent to the same synapse with connection guaranteed to pre-synaptic address Area0Y0X1. Connection took place and as spikes were being sent from neuron Area0Y0X0 by that point, *AeCorrect* raised and the neuron started to emit spikes again.

achieves good linearity and the main effect of mismatch is a shift in the output voltage, suggesting that the main cause of mismatch is threshold variation in the pMOSFET that generates *Proximity* at its gate. In general, this variation is likely to cause different neurons to have incoming connection fields with different amounts of spread. Applied to Gaussian distributions, a downward/upward shift of *Proximity* is likely to increase/decrease kurtosis respectively. Mismatch between neurons on the same chip cannot be assessed because only one *Proximity* node per chip has been made accessible, but it is likely to be less severe than mismatch between chips.

5.5.3 Ability to form probabilistic distributions

The circuitry within each synapse which implements the connection and disconnection rules has been demonstrated above in section 5.5.1, in trials which eliminated the influence of analogue values for *Proximity* and *nWeight* by using only extreme (digital) values for *nProbConnect* and *nProbDisconnect*. In this section, the ability of the analogue values to influence rewiring is demonstrated.

5.5.3.1 Insufficient open-loop amplification

In fact, the design presented has a defect in the ability to form synapses. In figure 5.2, the amplifier *DA2* which compares *Proximity* to *nProbConnect* is in open loop configuration and, as it is a wide range amplifier [Mead, 1989, page 80] it typically outputs a voltage close to Vdd or Gnd. Only when *Proximity* and *nProbConnect* are very close will it output an intermediate value, and the following gate *NAND3* applies further amplification to this signal to yield *nOverSig*. Nevertheless in a system where there are a large number of comparisons made, a proportion of these result in an intermediate value for nOverSig. Since this is applied separately to T1 in figure 4.2 for each monitor bit, mismatch in the transistors which make up the transmission gate provide an overriding signal of varying strength to each of the monitor bits, such that some bits take their new value whilst others do not. Thus in some cases where the decision that the pre- and post-synaptic neurons are close enough to connect is marginal, the pre-synaptic address can be stored wrongly in the synapse by the failure to latch some bits. Experiments established that for the standard ranges of values used for Proximity and nProbConnect in order to implement the model from chapter 2, there is a 1/136 chance per bit of not taking the correct value upon synapse

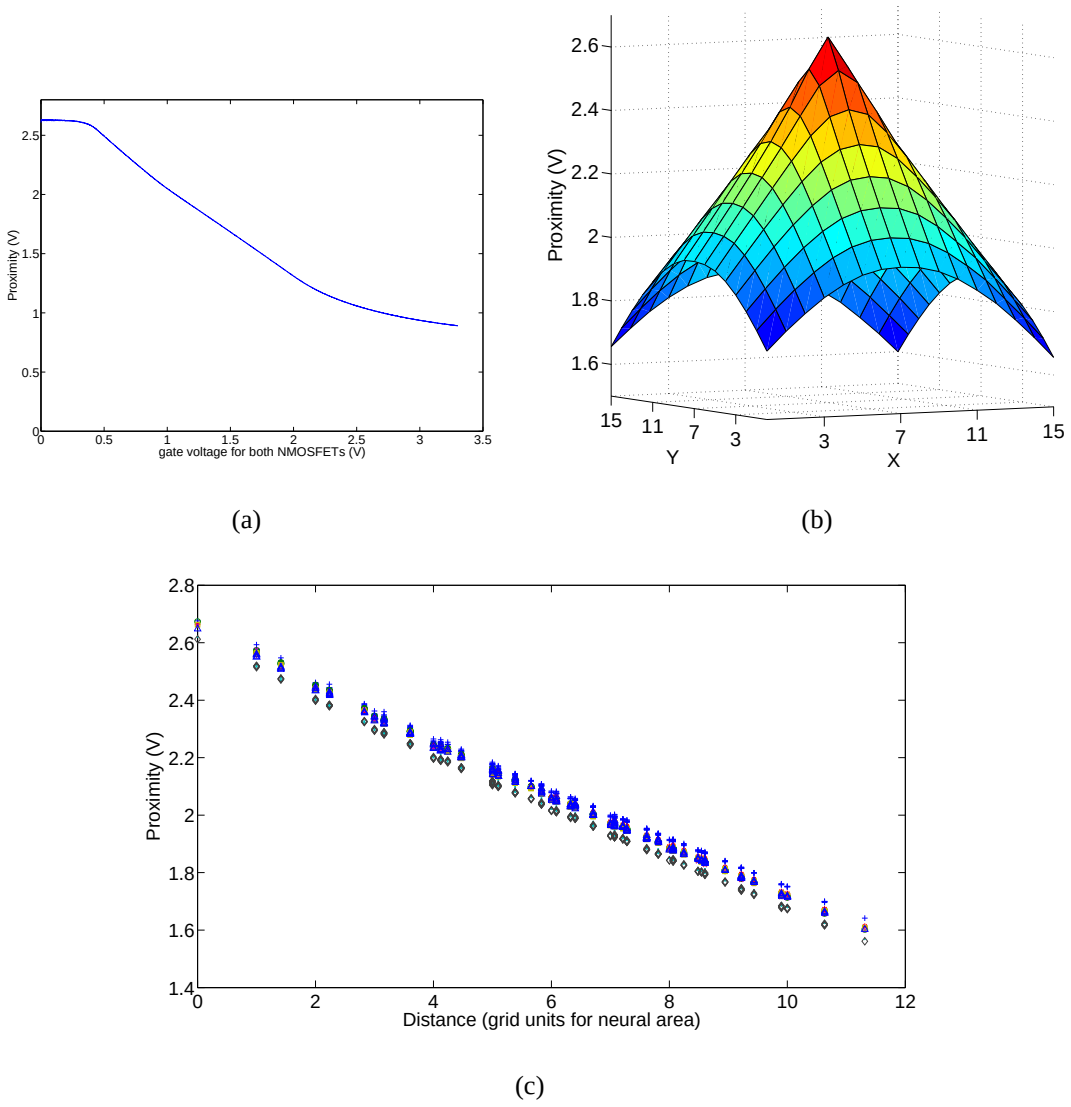


Figure 5.7: Proximity. (a) A single chip was configured so that the voltage along both of its resistor chains was linked and externally controlled. This value was swept in the range *Gnd-Vdd* (x-axis) and the *Proximity* value calculated by the bottom-right neuron (Y7X3) was recorded. (b)-(c) For the bottom-right neuron on each of 8 chips, a synapse was considered for connection with a neuron in each possible source location and *Proximity* was recorded. $vPullDown = 0.4V$; $vPullUp = 2V$. (b) The results were shifted so that the highest *Proximity* in each case occurred at Y7X7, and the mean was taken for each location. The lowest *Proximity* occurred at the complementary position i.e. Y15X15. (c) *Proximity* is plotted against the distance which it is intended to represent. A different symbol is used for the data points from each target neuron.

formation. As there are 9 bits per synapse this would give a maximum 6.4% chance of a synapse taking an incorrect value, if the probabilities per bit were independent within a synapse; in fact the probabilities per bit are not independent within a synapse, such that the actual chance of a synapse taking an incorrect value is lower. It may, however, be high enough to significantly affect statistics relating to mapping quality. To correct the design, one approach would be to introduce an additional amplification stage between DA2 and NAND3, for example a pair of inverters or a further open-loop differential amplifier. As amplification in this stage was increased, this would decrease the distance between *Proximity* and *nProbConnect* that would result in an intermediate output value, until a desired level of accuracy was achieved. To achieve 100% accuracy, a memory element could be inserted between DA2 and NAND3, with a similar design to INV5 and M7-M10, which was activated shortly before the compare signal, ensuring that a stable all-or-nothing output state was achieved.

To compensate for this problem for the purposes of implementing the model, two measures were adopted. Firstly, rather than generating initial random synaptic distributions for each neuron based on ideal parameters, the initial distributions were instead created by selection from a set of synapses which had previously formed on the chips. The results of several attempts to form distributions with a given set of parameters were pooled, with synapse formations recorded separately for each neuron on each physical chip. Then a subset of incoming (dendritic) synapses was chosen from the pool for each neuron, up to the required numbers. Note that this procedure is required in any case in order to allow for the effects of mismatch, since each physical neuron has a different effective standard deviation measured relative to its ideal location ($\sigma_{measured}$), for an applied standard deviation (σ_{form}) and a given number of synapses, due to mismatch (see section 5.5.3.3). This implies that chips cannot be interchanged if the same pooled data is to be used. It is then still possible to apply statistical tests to the *de facto* before and after variances in order to observe changes which indicate the action of the synaptic learning rule, although the initial mean $\sigma_{measured}$ may differ from the intended value.

Secondly, the implemented synaptic address bits happen to initially latch at *Gnd* on start-up, therefore if a synapse is formed during simulation for the first time, any bits which do not correctly take their new value would retain their existing zero value. This has the potential to systematically skew incoming connection fields in favour of pre-synaptic addresses which contain more zeros. To overcome this, an initial pool of synapse formations

was generated as above. During start-up, all synapses were connected to synapses selected from the initial pool and then disconnected. A second phase of synapse pool formation was then carried out, where address bits which did not latch correctly would retain their previously programmed values, strongly skewing the incorrect address away from the tendency to an excess of zeros (from 1/136 per bit to 1/18496 per bit, assuming independence between trials). Further iterations of this process could be carried out if it was deemed necessary. This correction has been applied in all results presented hereafter.

5.5.3.2 Creating differently shaped receptive fields

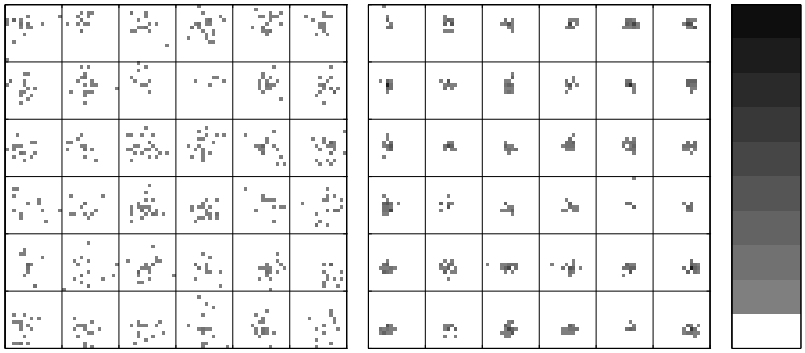
The method for generating the signal *nProbConnect* is described in appendix G. The method used to read the final connectivity following rewiring is described in appendix H.

Figure 5.8 demonstrates the ability of the system to form receptive fields with a Gaussian profile.

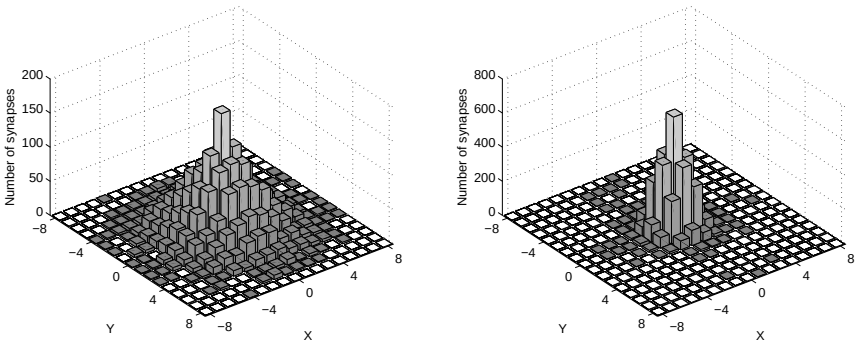
Figure 5.9 demonstrates another possible receptive field shape, in this case a bounded isodensitic receptive field. In this case the uncertainty at the boundary due to mismatch is apparent, and a small number of outliers can be seen, which are probably the result of the address-bit latching mismatch problem described above.

5.5.3.3 Variation in variance

For a desired σ_{form}, each neuron develops a different $\sigma_{measured}$. This is partly expected due to the random nature of the input stream, and partly due to mismatch between neurons in the circuitry which generates the *CloseEnough* signal. This mismatch is partly due to the MOSFETs that generate the *Proximity* voltage, as suggested by figure 5.7. There are also likely to be differing offsets between neurons for amplifier DA2 in figure 5.2. The effect of this mismatch is more apparent when σ_{form} is small, as is shown in figure 5.10. (a) demonstrates visually that outliers can be seen more easily for a small σ_{form}, for example the connection field of the neuron which is 5th down and 3rd across on the right grid is more diffuse than those of its neighbours. (b) shows this effect as an inversely proportional relationship between σ_{form} and the coefficient of variation of $\sigma_{measured}$.



(a)



(b)

Figure 5.8: Gaussian receptive field formation. *nProbConnectFF/Lat* signals were created based on parameters $\sigma_{form-feedforward} = 2.5$, $p_{form-feedforward} = 0.16$, $\sigma_{form-lateral} = 1$ and $p_{form-lateral} = 1$. With no synapses initially connected, rewiring was run for 50 seconds with 10,000 rewiring iterations per second (≈ 30 rewiring opportunities in total per synapse)*; nProbDisconnect* was maximised, i.e. no synapse elimination. Afterwards there were an average of 16.1 feed-forward and 18.0 lateral synapses per target neuron. Mean $\sigma_{aff-ff} = 2.30$, Mean $\sigma_{aff-lat} = 0.97$. left: feed-forward receptive fields; right: lateral receptive fields. (a) Receptive fields for a subset of target neurons (in the range Y5-10, X5-10). Within each receptive field, white space indicates no synapses formed with neurons in that part of the afferent area; squares of increasingly darker grey shades indicate higher numbers of synapses with the neuron in that position (max=9 as shown in the scale-bar); (b) combined receptive fields for all target neurons, with the afferent neuron whose ideal location matches the location of the target neuron centred at (0,0).

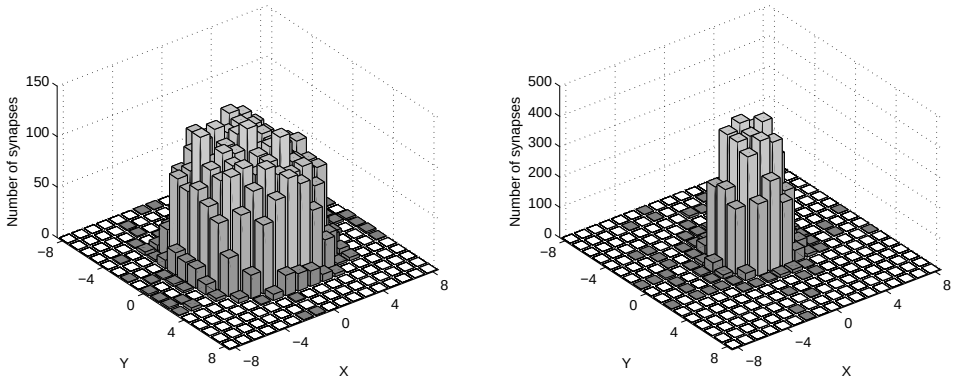


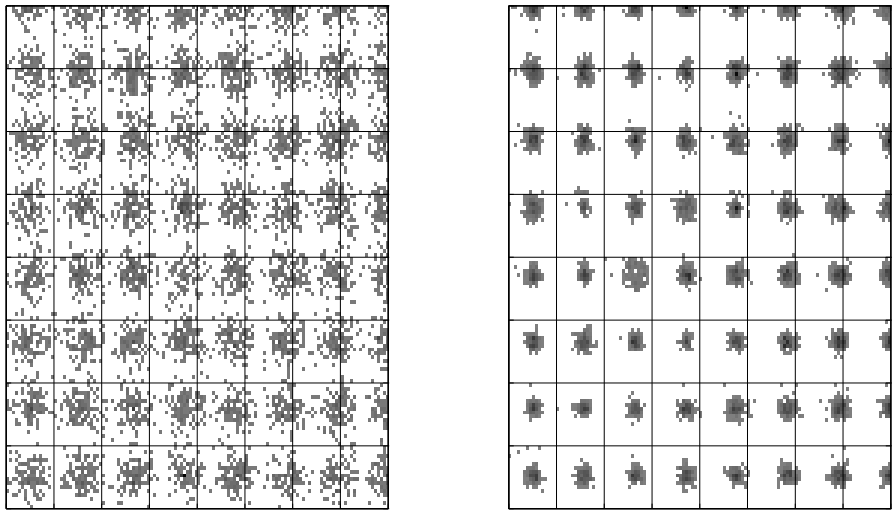
Figure 5.9: Formation of isodensitic bounded fields. The experiment was as in figure 5.8 but with *nProbConnect* created based on a formation probability which was flat up to a certain boundary distance and zero thereafter. Combined receptive fields for all target neurons are shown, with the afferent neuron whose ideal location matches the location of the target neuron centred at position (0,0); Left: feed-forward projection ($boundarydistance = 5$; $p_{form-feedforward} = 0.25$); right: lateral projection ($boundarydistance = 2.5$; $p_{form-lateral} = 1$).

5.5.4 Non-toroidal topology

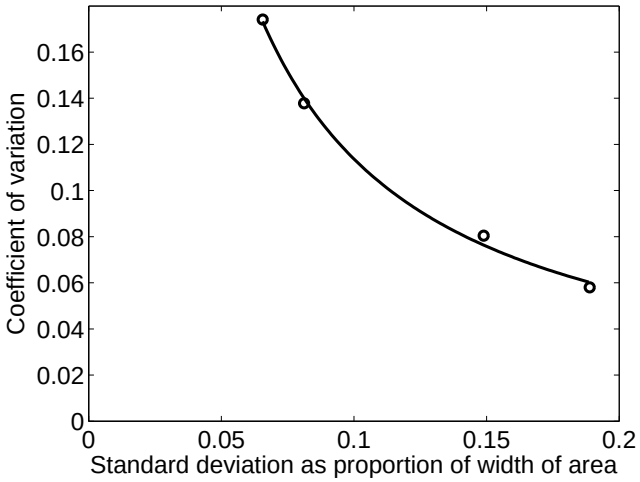
Figure 5.11 shows the ability of the chips to calculate proximity based on a non-toroidal topology and the effect of this on receptive field formation. In the receptive fields of neurons towards the edge of the chip, more connections form with pre-synaptic partners whose ideal locations are in the area which is available, resulting in denser sampling.

5.6 Discussion

The synapse design which has been presented allows synapses to be rewired during operation, without interrupting spike delivery and neuron functions. Whilst it is possible to impose an arbitrary network topology by external programming, it is also possible to allow a probabilistic topology to form and, if desired, to continue to develop within the

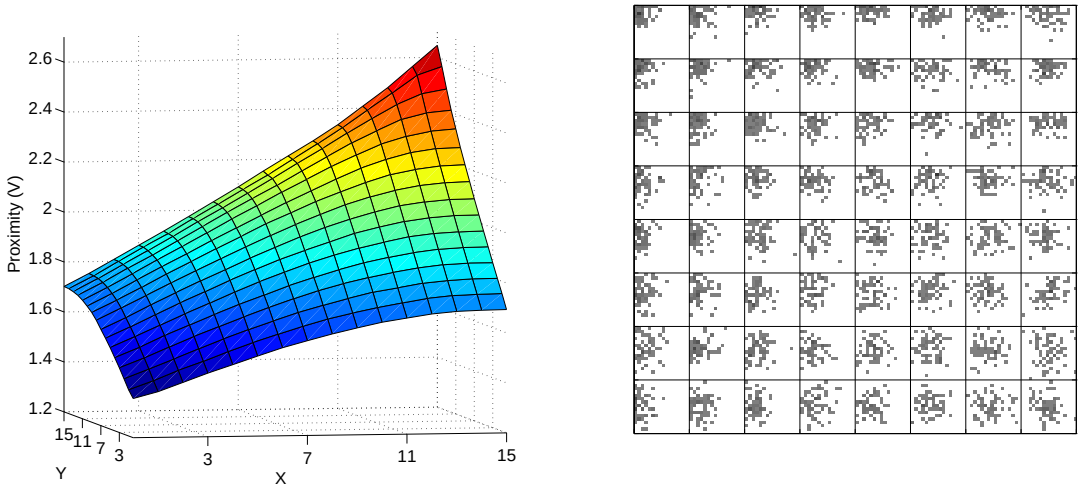


(a)



(b)

Figure 5.10: Variation of $\sigma_{measured}$ vs σ_{form}. (a) the initial pool of possible synaptic connections, created as described in section 5.5.3.1, for neurons Y=0-7, X=8-15 only; left: feed-forward synapses, $\sigma_{form-ff} = 2.5$; right: lateral synapses, $\sigma_{form-lat} = 1$. Each neuron has more than 64 afferent synapses for each projection. (b) for four initial pools of connectivity formed based on different σ_{form}, the coefficient of variation of $\sigma_{measured}$ is plotted. For generality, σ_{form} is expressed as a proportion of the width of the area (i.e. $\sigma_{form}/16$). The best fit line is shown for a $1/x$ curve.



(a) (b)

Figure 5.11: Non-toroidal topology. (a) Proximity value generated by neuron Y15X15 with respect to every possible pre-synaptic location. (b) Gaussian receptive field formation. *nProbConnectFF* was created based on $\sigma_{form-feedforward} = 2.5$, and $p_{form-feedforward} = 1$. *nProbDisconnect* allowed no synapse elimination and only feed-forward connections were considered. With no synapses initially connected, rewiring was run for 100 seconds with 10,000 rewiring iterations per second. Receptive fields for the feed-forward projection are shown for a example set of target neurons (neurons in the range Y0-7, X0-7). White space indicates that no synapses formed with neurons in that part of the afferent area; squares of increasingly darker grey shades indicate higher numbers of synapses with the pre-synaptic neuron in that position.

system according to biologically realistic principles, without any details of the topology being made available off-chip. In other words this system allows a black-box approach to network wiring at the level of individual synapses, allowing a neural network designer to concentrate on higher-level building blocks. Rewiring probabilities can be made arbitrarily low, even achieving biologically-realistic rates of synapse formation and elimination, i.e. hours, days or months between events [Grutzendler et al., 2002, Trachtenberg et al., 2002]. Although supporting stochastic processes were generated off chip for this test system, these could be integrated on-chip.

The proximity calculation circuits presented deliver a measure of distance which is isodirectional, resulting in fully radially symmetric receptive fields. This sets it apart from the systems of Serrano-Gotarredona et al. [1999] and Choi et al. [2005] which could achieve receptive fields with radial symmetry of limited order with angular phase linked to the axes of the chips. The measure of distance is also linear. Linearity is not in fact necessary for the system described, as supporting probability distributions could be profiled to compensate for any monotonically decreasing measure of proximity, but the calculations necessary to generate the probability distributions are simplified with a linear solution. This may prove advantageous if random value generation was moved on-chip. To produce random values on chip would require a linear feedback shift register configured as a pseudo-random number generator, followed by any digital arithmetic necessary to create the appropriate distributions (for example by implementing the calculation presented in appendix G), followed by a DAC to create the analogue voltage required. Alternatively it is possible to generate analogue noise by amplifying thermal noise [Alspector et al., 1988]; this could then be profiled through various analogue transformations to match the probability of adaptation.

The cross-chip proximity calculation circuit represents an advance in multi-chip neuromorphic systems. Whilst multiple chips have previously been used together in single systems, each chip has either represented a separate neural layer [Serrano-Gotarredona et al., 2006, Chicca et al., 2007] or else a separate set of cells or function within a layer [Choi et al., 2005]. Multiple chips have also been used to implement contiguous segments within a spinal cord, which arguably have a topographic organisation [Patel et al., 2006]. The system presented here, however, consolidates the use of multiple chips to create a single expansive area, by implementing a 2D cross-chip proximity calculation, which requires that all chips have a dedicated place within a 2D lattice of chips which form a

complete neural area.

As this system scales, speed of operation may eventually become an issue, since the maximum speed at which rewiring can take place is inversely proportional to the number of synapses in the neural area. Assuming, arbitrarily, that each synapse should be given one opportunity to rewire each hour, then at the speed of 10KHz used in this test system, 3.6×10^7 synapses can be accommodated.

The inverse relationship of σ_{form} to the coefficient of variation of $\sigma_{measured}$ suggests that this effect will become more problematic if the size of neural areas were scaled up and σ_{form} were not scaled up accordingly. That is, if receptive fields over small regions of a large neural area were desired, the resulting receptive fields would be poorly matched in size. There are a few possible solutions to this. Firstly, the approach taken in the results presented in this chapter, of raising vPullUp beyond the ability of the pMOSFET to source current, can be extended to some extent, stretching the range of the *Proximity* measure which represents close values for greater accuracy at the expense of accuracy at a distance. Secondly, weakening the pMOSFET would extend this effect still further. Finally, with more complex decoding circuitry it would be possible to set two complementary locations on each resistor chain, a certain distance away from the source location in each direction; this would set a boundary for the range over which the proximity calculation would work.

Is it necessary to recreate the physical layout of neural areas? Section 2.1.3.2 presented the viewpoint that the purpose of topographic organisation in brains may be to improve wiring efficiency by minimising wiring length. Since individual wires between connected neurons are not implemented in this system, but rather, all spikes are broadcast to all neurons, the potential advantage of topographic organisation may no longer apply. However it is likely that in order to create neural areas on the scale of those in the human cortex, further advances are necessary. One possibility is to assume that the probability of a neuron being connected to a pre-synaptic partner whose ideal location is beyond a certain distance from it is too low to be worth modelling, allowing a layer to be subdivided into different regions each with its own bus and a higher-level routing system controlling communication between different regions; in this case, keeping neurons which are more interconnected close to each other, whether on the same chip or on chips which are nearby in terms of their communications network, continues to be advantageous. As discussed in section 4.7, Khan et al. [2008] have presented an approach to minimising the amount of communication in the inter-chip routing of spikes; this approach is most efficient when connections

are predominantly local.

The approach of distributing the circuitry which achieves rewiring throughout the synaptic array has been argued for in section 5.3 above. As such, this design serves to highlight a pole on a spectrum of design choices regarding the amount of functionality implemented within synapses and indeed on-chip as opposed to in external digital processing. However, hybrid approaches are possible and may prove beneficial. Rewiring functions could be centralised to a single circuit on the periphery of each chip. This would remove about 20% of the area of the synapse design in the present system, with the expense that the synapse would have to buffer its analogue weight value out to the periphery and some additional signal rails would be required. A hybrid approach might be to implement disconnection in the synapse and connection at the level of the chip or the network. The decision to disconnect could then be easily based on the weight and would result in a simple digital message, (perhaps simply a pulse) to peripheral circuitry. This could then trigger the reassignment of that potential synapse circuit by external circuitry with the benefit that the implications for higher level network routing tables [for example, those described by Khan et al., 2008] could be resolved at the same time.

The model in chapter 2 assumes a fixed relationship between neural areas. Additionally, the neural areas are of the same size and shape and there is no transformation in the mapping between the areas. This allows for the address of the pre-synaptic neuron in the source area to be directly decoded and used to specify a cell in the target area as the ideal location of the pre-synaptic neuron. However it was noted in section 2.1.5.4 that the model is not incompatible with transformations between areas. To apply transformations between areas in this system, some transformation must be applied to the source address and the location in the target area which is used as its ideal location. For simple transformations such as rotation through right-angles, mirroring, or expansion or compression by multiples of 2, simple bit-wise or bit-shifting operations on the source address prior to decoding would be sufficient [as in Choi et al., 2005]. Any linear transformation could be achieved by matrix multiplication, and for ultimate generality, there could be a piecewise arbitrary mapping between addresses implemented by a look-up table; note that this proposal differs from the use of the look-up table identified in section 4.2.1 since source neuron addresses would not be converted to target synapse addresses but rather to source neuron addresses in a transformed topology. Another interesting possibility may be to replace the resistors in the peripheral cells of the proximity circuit with transistors so that their resistances could

be dynamically altered in order to skew proximity calculations. In order to implement the suggestion in section 2.4.4 of combining the model with a model for the development of areas, such as that described by Willshaw [2006], there would need to be a further mechanism, possibly implemented across the chip, which maintained details of the afferent mapping, which could then be used to modify the transformation applied to the source addresses.

One more contemporary approach is worth mentioning. Anand Chandrasekaran (Stanford University, California, personal communication) is developing a system whereby point-to-point AER is used to deliver address-events from a source neuron to a location in a target chip which represents the ideal location of the neuron. From there they travel through local reconfigurable interconnect composed of wires and switch matrices in the style of a FPGA to their target neurons. Such a system would not lose channel capacity in the presence of large fan-out and would maintain the ability to implement arbitrary axonal arbors and synaptic rewiring, where the extent of axonal arbors would be constrained only by the competition for limited available interconnect resources; the brain has a similar resource problem as there is only a limited volume in which neuropil can develop, although it has three dimensions to work with rather than two.

5.7 Conclusions

An approach has been presented of choosing the location of circuitry for neural and synaptic functions based on minimising the need to transmit information. This approach has been used to argue for the location of circuitry for implementing synaptic rewiring within each synapse. Such circuitry has been developed and results have been presented which demonstrate its functioning. Notwithstanding a design error for which solutions have been suggested, the circuit is capable of connecting and disconnecting a synapse in response to either explicit external programming or a probabilistic learning rule, proposed in chapter 2. The connection rule requires a calculation of the distance between a neuron and the ideal location of a potential pre-synaptic partner; consequently, circuitry for Euclidean distance calculation has been presented. This circuitry is based on established principles for calculating Euclidean distance, but it is a novel formulation for the specific task; its notable features include current mode operation across multiple chips and the capability of implementing both toroidal and non-toroidal topologies. The ability of the rewiring

circuitry, together with the distance calculation circuitry, to allow the formation of radially symmetric receptive fields with arbitrary relationships of connection probability to distance from the centre, has been demonstrated.

The first part of the hypothesis which was proposed in chapter 1, that synaptic rewiring can be implemented in neuromorphic VLSI by circuitry distributed throughout the synapses of a neural array, has therefore been proven.

Chapter 6

Map formation model implemented in VLSI

6.1 Introduction

In section 1.2 a hypothesis was proposed, that (1) synaptic rewiring can be implemented in neuromorphic VLSI by circuitry distributed throughout the synapses of a neural array, that (2) this ability can be used to model the phenomenon of topographic map formation, and that (3) this ability can increase the stability of patterns learnt by a neuromorphic system. Topographic map formation was therefore explored in chapter 2, culminating in a model which captures many general features of this process. A chip was then fabricated and its capabilities have been explored in chapters 3, 4 and 5. In particular, circuitry for synaptic rewiring has been developed which is distributed throughout a neural array (chapter 5), built on top of a novel design for a distributed address-event receiver (chapter 4).

In this chapter, chip results are presented which qualitatively match the simulation results presented in chapter 2, thus demonstrating its fitness for the intended purpose. In so doing, part 2 of the hypothesis is proven, at least to the extent that the many diverse phenomena related to topographic map formation have been captured in the model. Similarities and differences between the model and its implementation are discussed. Additionally, in order to address part 3 of the hypothesis, the advantage of increased memory duration offered by synaptic rewiring is demonstrated.

6.2 Results

6.2.1 Effects of rewiring

The experiment and control from chapter 2 which establish the difference in behaviour between rewiring and no rewiring, that is case 1 and case 2 from table 2.2, were duplicated using the fabricated chips. The experimental set up was as described in section 3.5, with the multi-chip spike delivery system as described in section 4.5.4. The initial number of feed-forward and lateral synapses was 32 each, to make full use of the capacity in the fabricated system. Parameters used were as in table D.1. Most parameters given in table 2.1 were therefore matched, with the exception that it is difficult to quantify A_+ (the proportion of g_{max} by which synaptic weight is raised on an immediate potentiation event) and B (the ratio $A_-\tau_-/A_+\tau_+$) given the weight dependence in the synapse circuit and other slight non-linearities. Rather, the biases which work together to create A_+ and A_- were treated as free parameters in order to achieve similar weight distributions and output spike rates to those achieved in simulation. g_{max} (as created by the bias *PrePulseSynCond*) was also treated as a free parameter; it had to change to accommodate the greater number of synapses used.

Initial connectivity for each neuron was randomly chosen from a previously generated pool of possible pre-synaptic partners, as described in section 5.5.3.1. “Shuffled” versions of the final connectivity were created in the same way, for the purpose of significance tests, as described in section 2.4. At the end of the experiment, weights were recorded for each synapse and then connectivity was recorded, using the method described in appendix H.

The experiments ran for 5 minutes of simulated time; this took 5 minutes, plus a period of less than 1 minute for set up of the chip and data read out, compared with many hours for the simulations in chapter 2, which were carried out using a compiled C++ function on a single PC within a cluster.

In order to interpret weights on a normalised scale, *nWeight* voltages were converted by linear interpolation between 0.2V (representing 1, or maximum weight) and 1.9V (representing 0, or minimum weight). Any outliers from this range were constrained to the maximum or minimum value.

Results are given in table 6.1, including the results of relevant significance tests; comparable results from the simulations in chapter 2 are included for reference. As with the

simulations, in case 3 (with rewiring) the variance of the final connectivity (not considering weights) is significantly lower than a shuffled version of the final connectivity in which the same number of synapses are redistributed randomly according to the method for constructing the initial connectivity, and the variance of the final weight distribution is significantly lower than a shuffled version of those weights within the same set of synaptic connectivity for each neuron. AD results are also included for completeness. However, whilst not all comparisons involving AD follow the same pattern, the reason why these should be disregarded was explained in section 2.4.2.

6.2.2 Weight distribution

The weight distributions generated tend to be bimodal, at least for a range of parameters around those given in section 6.2.1 above. An example is given in 6.1. In this experiment, uncorrelated spike trains were used as input. The only correlations were those where a target neuron had two or more synapses with a single pre-synaptic partner; this was enough for the development of a slight bias towards higher weights with pre-synaptic neurons whose ideal locations were closer to the location of the post-synaptic neuron, due to the increased sampling of that space in the network connectivity; this demonstrates one of the effects described in section 2.4.3. The distribution which formed was initially uni-modal; then, while there was little change in overall weight, the distribution gradually became bi-modal over $\approx 20s$, as a group of synapses within each neuron gained control over its firing. The bi-modal divergence was not extreme, however. Introducing correlations between sub-groups of synapses can increase the divergence, as shown below in section 6.2.3.

According to the analysis in [Gutig et al., 2003], as weight dependence of STDP increases, the effect on a stable distribution of weights of afferent synapses to a single post-synaptic neuron is that the poles of a bi-modal distribution move closer together until a critical point is passed at which they form a uni-modal distribution. However, there are many other factors that can influence the weight distribution, so the extent of divergence observed in experiments such as these cannot on its own be taken as evidence of the effect of weight dependence.

Table 6.1: Summary of simulation results: cases 1-2: from simulations; cases 3-4: from implementation; case 1 and 3: rewiring and input correlations; case 2 and 4: input correlations and no rewiring

Case	**1**	**2**	**3**	**4**
Target neuron mean spike rate	24.7	17.4	11.2	15.4
Final mean no. feed-forward synapses per target neuron	14.1	NA	25.4	NA
Weight as proportion of max for the initial no. of synapses	0.60	0.36	0.50	0.41
Mean $\sigma_{aff-init}$	2.36	2.36	2.94	2.94
Mean $\sigma_{aff-fin-con-shuf}$	2.32	NA	2.94	NA
Mean $\sigma_{aff-fin-con}$	1.95	2.36	2.51	2.94
p (WSR; $\sigma_{aff-fin-con}$ vs $\sigma_{aff-fin-con-shuf}$)	2.4×10^{-25}	NA	6.8×10^{-29}	NA
Mean $\sigma_{aff-fin-weight-shuf}$	1.88	2.10	2.45	2.91
Mean $\sigma_{aff-fin-weight}$	1.70	1.98	2.16	2.48
p (WSR; $\sigma_{aff-fin-weight}$ vs $\sigma_{aff-fin-weight-shuf}$)	2.7×10^{-27}	8.7×10^{-6}	2.3×10^{-22}	7.3×10^{-33}
Mean AD_{init}	0.78	0.78	0.80	0.80
Mean $AD_{fin-con-shuf}$	0.89	NA	0.74	NA
Mean $AD_{fin-con}$	0.83	0.78	0.92	0.80
p (WSR; $AD_{fin-con}$ vs $AD_{fin-con-shuf}$)	0.31	NA	1.2×10^{-4}	NA
Mean $AD_{fin-weight-shuf}$	0.92	1.36	1.31	0.93
Mean $AD_{fin-weight}$	0.95	1.58	1.32	0.88
p (WSR; $AD_{fin-weight}$ vs $AD_{fin-weight-shuf}$)	0.48	1.2×10^{-3}	0.12	0.14

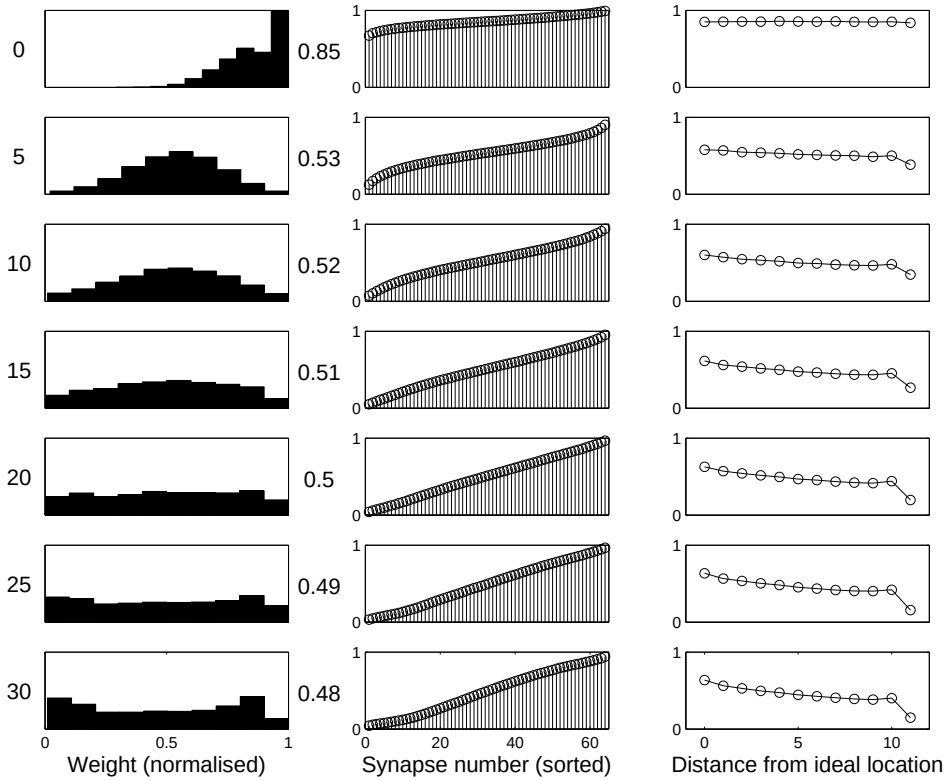


Figure 6.1: Emergence of bimodal weight distribution. All synapses were connected with input layer neurons (i.e. a fully feed-forward network) with $\sigma_{form-ff} = 2.5$. All parameters were as in section 6.2.1 except that $\tau_{-} = 40ms$, $\tau_{ex} = 10ms$ and the biases which yield g_{max}, A_{+} and A_{-} were adjusted to achieve mid-range weights. Spike trains from input neurons were selected from a Poisson distribution with a fixed rate of 20Hz (i.e. there were no spatial correlations). The weight of each synapse was sampled every 5 seconds for 30 seconds; each complete set of weight samples took over 0.5 seconds to complete, thus some of the weights shown for the zeroth second have already reduced from their initial maximum value. Top to bottom: results at each sample point; the time of the sample point is given to the left. Left: histogram of weights. The number to the right of each histogram is the total weight as a proportion of the total if all weights were maximised. Centre: The mean weight (y-axis) for the nth synapse (x-axis) in each neuron, where the synapses in each neuron are sorted in ascending order of weights; this demonstrates that the distributions seen in the histograms are generally present within each neuron, rather than an aggregate effect. Right: The mean weight (y-axis) for all synapses whose pre-synaptic neurons have ideal locations of various distances from the location of the post-synaptic neuron (x-axis), where distance has been rounded to the nearest whole number.

6.2.3 Persistence of learnt patterns

A set of experiments described here demonstrate the ability of the rewiring mechanism to increase the persistence of learnt patterns.

In each neuron, 32 synapses were connected to each area, input from both areas was generated by the PC and spikes generated by the chips were not relayed back to the chips. Thus the network was completely feed-forward, with input coming from two areas each mapped across the same space, simulating a binocular projection converging on a single area.

For the first 240s, spiking input consisted of simultaneous spikes from a randomly chosen 50% of the neurons in one input area. after 20ms there was another simultaneous set of spikes from neurons in the other input area. After a further 20ms this pattern repeated, with different randomly chosen neurons constituting the "flash" each time. Between these flashes there were Poisson-distributed spikes from all neurons in both input areas with the rate set so that the total spike rate per pre-synaptic neuron including spikes from flashes was 20Hz. For a further 30s afterwards spiking input consisted just of Poisson-distributed spikes from all neurons in both input areas with a fixed rate of 20Hz. Thus, there was a longer period of input with intra-ocular but not inter-ocular correlations, followed by a shorter period of input with no correlations. The spike trains were designed to contain strong correlation cues and therefore departed further from such biological realism as there was in those employed by Song and Abbott [2001], in order to achieve a clear demonstration.

Figure 6.2 shows the results for an experiment in which there was no rewiring. Parameters were as in appendix D, except that $\tau_{-}20ms$, $\tau_{ex} = 10ms$ and the biases which yield g_{max}, A_{+} and A_{-} were adjusted to achieve mid-range weights. During the intra-ocular correlations, strong ocular preferences quickly developed in the weights of the incoming synapses to each target neuron. This is shown by ocular dominance maps. In addition a measure of ocularity is given to the left of the maps. This is the mean of the ocularity for each target neuron, which is calculated as:

$$Ocularity = 2\left|\frac{\sum_i w_i^1}{\sum_i w_i} - \frac{1}{2}\right|$$

where w_i is the weight of the ith synapse for the target neuron and w_i^1 is the weight of the ith synapse only for synapses with pre-synaptic neurons from area 1. The measure

therefore gives a value of 1 if ocular preferences are complete, but 0 if there are no ocular preferences. Strong preferences developed during the first 20-30s. There was further slight improvement until 70s after which the ocularity measure did not change. In contrast to the results given in figure 2.5 there are no spatial correlations in the ocular preferences, as there were no lateral connections to set up such correlations. After 240s when the intra-ocular correlations in the input disappeared, the ocular preferences quickly collapse until after 30s there is not much evidence of them remaining.

In this process, the weight histogram shows that the strong correlational cues in the input caused strong and relatively rapid bimodal divergence, compared with figure 6.1 in which there are no correlational cues in the input. Then, when the correlational cues were removed the weight distribution reverted to a less clearly bimodal distribution, similar to that achieved in figure 6.1.

Figure 6.3 gives results for the same experiment but with rewiring. For these experiments, $p_{elim-pot} = 0$, i.e. synapses were only eliminated when weaker than 50% weight. During a rewiring experiment it is not possible to observe the connectivity of the chip, as the method for observing it is destructive to weights, therefore results are given only for the end of an experiment. In (a) and (b), the experiment was stopped after 240s, at the end of the correlated phase. Both experiments used identical initial connectivity, spike trains and rewiring probabilities, that is to say, all inputs to the chips were identical. Nevertheless different ocular preferences arise; this demonstrates the indeterminate performance of the chips, at least in a case where different outcomes should be equally likely. Therefore, observing different patterns of weights or connectivity (or ocular preference) for different experiments in which identical input is supplied and the experiments are stopped at different points cannot be taken to indicate that the rewiring mechanism makes the pattern of connections or weights volatile. Experiments (c) and (d) (described below) also use identical initial connectivity, spike trains and rewiring probabilities.

In (a) and (b), a pattern of connectivity in the weights (centre) became embedded in the connection preferences (left), with the result that the ocular preference resulting from the combination of connections and weights (right) was stronger than that achieved with no rewiring mechanism (figure 6.2) after the same amount of time, as judged by the ocularity measures (0.91 vs 0.73 respectively).

In (c), the experiment was stopped after an additional 30s of uncorrelated spike trains. Whilst the preferences due to the weights (centre) declined to a similar extent as those

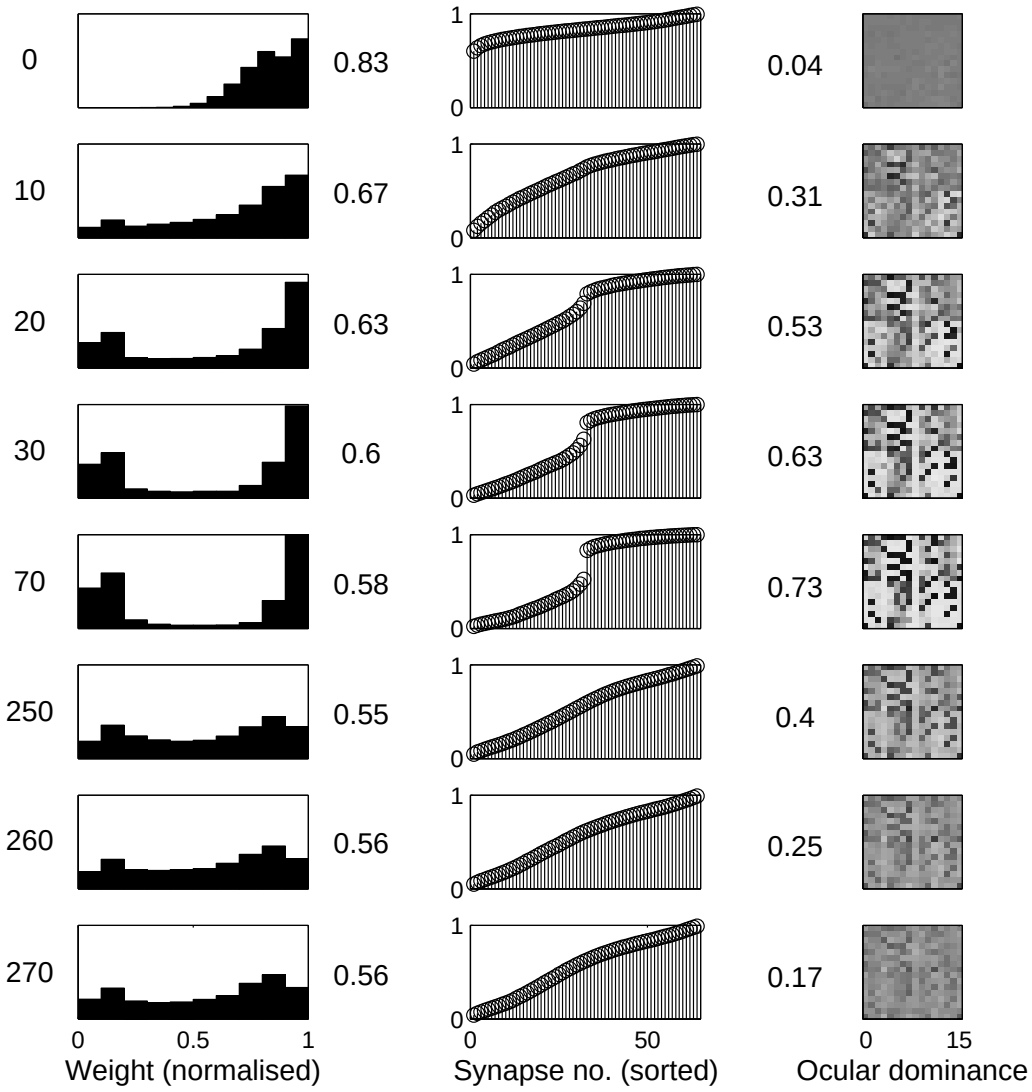


Figure 6.2: Persistence of learnt patterns without rewiring. Top to bottom: the weights of all synapses were sampled at the times labelled to the left. Left: histogram of weights. The number to the right of each histogram is the mean normalised weight. Centre: The mean weight (y-axis) for the nth synapse (x-axis) in each neuron, where the synapses in each neuron are sorted in ascending order of weights. Right: ocular preferences for neurons. Within each raster, each cell represents one target-layer neuron. The shade of the pixel gives the weighted sum of the synapses connected to area 1 as a proportion of the weighted sum of all synapses on a scale from white to black. The number to the left of the ocular dominance map is the related ocularity measure, as defined in the text.

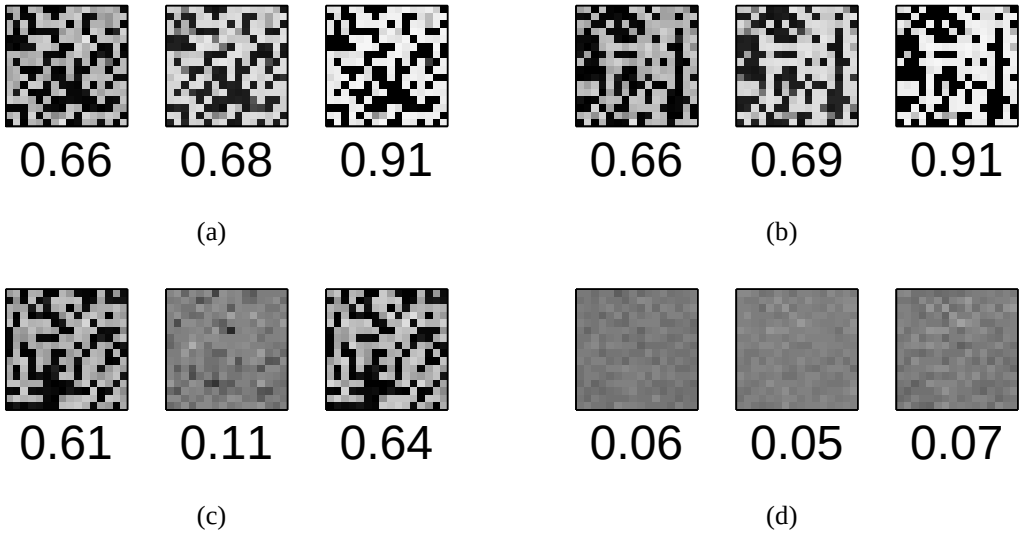


Figure 6.3: Persistence of learnt patterns with rewiring. Ocular preferences for neurons. Within each raster, each pixel represents one target-layer neuron. The shade of the pixel gives the preference of synapses for area on a scale from white (area 1) to black (area 0). Left (within each sub-figure): preference due only to the connections; this is calculated as the number of synapses connected to area 1 as a proportion of all connected synapses. Centre: preference due only to the weights irrespective of the relative numbers of synapses; this is calculated as the summed weight of the synapses connected to area 1 divided by their number, as a proportion of the summed weight of all connected synapses divided by their number. Right: preference due to the combination of connectivity and weights; this is calculated as the summed weight of the synapses connected to area 1 as a proportion of the summed weight of all synapses. (a)-(d): the results of 4 separate experiments. (a)-(b): After 240s of spike trains with strong intra-ocular correlations; the results of two separate experiments using identical initial connectivity, spike trains and rewiring probabilities; (c) after 240s of intra-ocular correlations followed by 30s of uncorrelated Poisson spike trains; (d) after 30s of uncorrelated Poisson spike trains. The number under each map is the related ocularity measure, as defined in the text.

in the experiment with no rewiring after the same time (0.11 vs 0.17 respectively), the preferences due to the connectivity (left) only declined slightly (from 0.66 to 0.61), with the result that the overall preferences (right) remain comparatively strong. (d) is a control experiment in which the same 30s of uncorrelated spike trains as used previously were sent to the chips immediately, without any correlated input first. No strong ocular preferences formed, demonstrating that the ocular preferences observed in (c) were not an artefact of the rewiring mechanism.

The experiments are summarised in figure 6.4, which shows the trends of the ocularity measure through time. The relatively slow rate of change of the ocularity measure for connectivity compared with weights is apparent.

6.2.4 Ocular dominance patterns

The model in chapter 2 was used to demonstrate two phenomena relating to topographic map formation and receptive field development. The first was reduction in the spread of receptive fields, which was used to quantitatively evaluate the change in network topology due to learnt patterns. The second was the development of patterns of ocular dominance, as an additional qualitative view, since patterns which form due to synaptic weight changes can be seen in the patterns of connectivity. To attempt the same with the fabricated chips, a further experiment was carried out, similar to case 1 as described in section 6.2.1, except that the input neurons were divided into two groups, mimicking the effect of binocular inputs. The groups were interspersed in a regular diagonal pattern, i.e. each input neuron is in the opposite group to its 4 adjacent neurons; the stimulus location switched between the two groups every time it changed. To keep the overall input rate the same the peak firing rate was doubled. The experiment therefore mirrored those reported in section 2.4.1.

After 5 minutes, preferences had developed not only for the centre of the distribution (figure 6.5, left) but also for one of the input spaces (right). Although the lateral synapses were largely depressed (centre right), they had sufficient influence that discernable regions of the area formed similar preferences. The result is qualitatively similar to the results presented in figure 2.5 for a lower number of afferent synapses per target neuron. The boundaries are not as clear as the best results achieved in simulation for the same number of afferent synapses; this might be improved with more optimisation of parameters, but it is not unexpected, given the non-idealities in the design. There is however

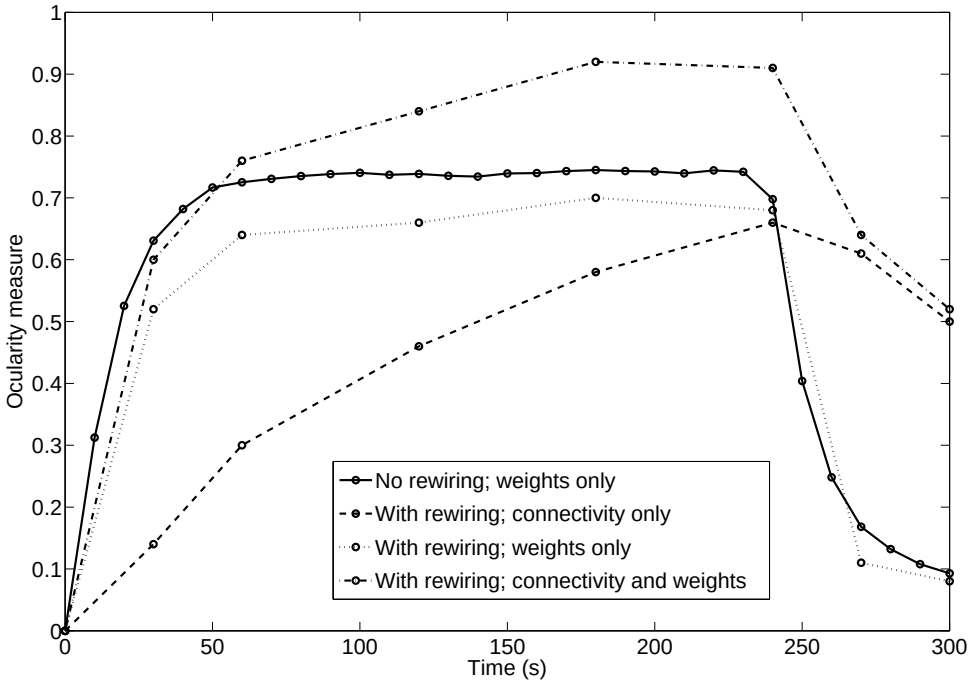


Figure 6.4: Persistence of learnt patterns summary. The ocularity measure (defined in the text) is plotted against time for an experiment in which stimuli with strong intercorrelations for 4 minutes are followed by 1 minute of stimuli with no intercorrelations. The solid line shows data points from the experiment with no rewiring shown in figure 6.2. The other three lines show the ocularity measure for connectivity only (dashed line), weights only (dotted line), and connectivity combined with weights (dashed-dotted line). For these three lines, the three data points at each time come from a single experiment, which terminated at the given time, allowing the connectivity and weights to be read out. These data points include those given in figure 6.3(a) (240s) and 6.3(c) (270s).

a noticeable vertical division between the upper and lower half of the map, which lies along chip boundaries, and the boundary between chips Y2X1 and Y2X2 stands out, on some rows. This suggests that the design suffers from unwanted intra-chip correlations. A likely cause is the choice of a transmission gate implementation of the pulse receiver, as discussed in section 4.7. Alternative designs which could avoid this problem are proposed in section 7.2.2.

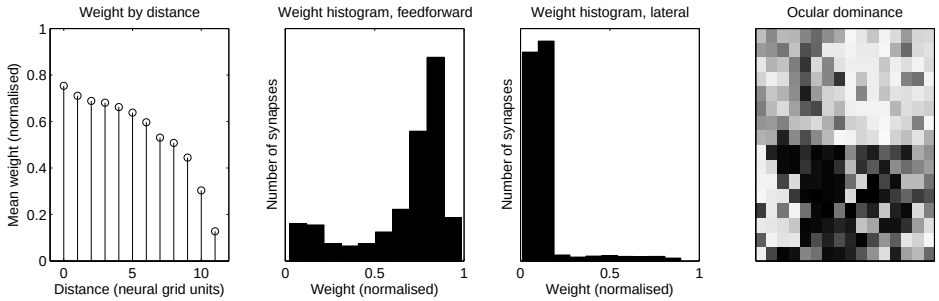


Figure 6.5: Ocular dominance patterns. Results after 5 minutes of input from two interspersed input spaces with intra- but not inter-correlations. Left: the mean weight of synapses according to the distance of the ideal location of the pre-synaptic neuron from the post-synaptic neuron (with distance rounded to the nearest whole number). Left-centre: histogram of weights of feed-forward synapses. Right-centre: histogram of weights of lateral synapses. Right: Ocular dominance map, where each cell represents a target layer neuron and is shaded according to the sum of feed-forward weights from one of the two interspersed input areas as a proportion of the sum of all weights, on a scale from white (1) to black (0).

6.3 Discussion

6.3.1 Differences between model and implementation

There are various differences between the simulations presented in chapter 2 and the experiments in section 6.2.1. It will be argued here that such differences do not diminish the ability of the fabricated chips to model the process of topographic map formation.

The introduction of weight-dependence in STDP in the circuitry for STDP was discussed in section 3.4.4.1. Whilst the model used weight-independent STDP, both to reduce the number of parameters and in keeping with the previous model on which it was based [Song

and Abbott, 2001], weight-dependence was introduced in the chip due to other theoretical and practical considerations. Firstly, it has been shown that a moderate degree of weight-dependence can improve the ability of STDP to act as a correlation detection mechanism [Gutig et al., 2003]. Secondly, weight-dependence reduces the duration of a memory trace [Billings and van Rossum, 2008], allowing the ability of synaptic rewiring to reduce volatility to be demonstrated, as in section 6.2.3. Finally, it is simply serendipitous that the silicon substrate allows various forms of weight-dependence to be implemented "for free" instead of by explicit design, [c.f. Bofill-i Petit and Murray, 2004]. The fact that two different forms of STDP can achieve qualitatively the same results suggests that the model is more robust than if it were tied to a specific form; the fact that non-linearities were introduced into the weight-dependence profile by the use of MOSCAPs for weight storage, without undermining the ability of the system to function, furthers this suggestion.

Another difference between the model and its implementation has been discussed in section 5.4.1. In the model, the potential pre-synaptic neuron for a potential synapse which is implementing its formation rule was chosen as the last neuron to have fired, whereas for the fabricated system the pre-synaptic partner considered was randomly chosen. The differences between these approaches have been discussed; however, here it is sufficient to note that since in simulations efforts were made to balance the firing rates between areas and since in a mature implementation additional homeostatic processes might be expected to keep neural firing rates in broad agreement, these differences are likely to be slight and are likely to be reduced rather than enhanced due to further anticipated developments.

In contrast to the simulations, the fabricated system is non-deterministic, such that different equally likely outcomes can arise even when identical input is provided, as seen in section 6.2.3. This indeterminism arises from analogue electrical noise, which is not present in digital simulations. Analogue noise is also present in the brain, and could be seen as an advantage for a system which seeks to model the brain, since a system whose performance does not change qualitatively in the presence of noise demonstrates such robustness as is characteristic of biological neural networks; indeed, noise is frequently added to computational simulations of neural systems and in some cases the results are found to be beneficial to performance [Mitaim and Kosko, 2004]. In a similar way, there are numerous sources of mismatch which cause permanent differences in performance between synapses and neurons. The combined effect of these has not been to undermine the performance of this system, as judged by the similarity between results in sections 2.4

and 6.2.1. It remains to be seen, however, whether such performance can be maintained while progressing towards the use of processes with smaller geometries and the greater mismatch they entail. Some loss of performance is noticeable in the ability to form clearly segregated ocular dominance patterns (comparing the results in sections 2.4.1 and 6.2.4), but it is difficult to conclude that this is due to either mismatch or electronic noise, since parameterisation of the model can have such a great effect on performance (the best results in section 2.4.1 were achieved after a detailed guided search through the parameter space). A particular problem is noticeable whereby unwanted correlations lead to similar preferences within chips or rows of neurons; this may be solved with an alternative design (see section 7.2.2).

There are various other limitations of the fabricated chips, for example the ability of a neuron to integrate the effects of incoming pulses on synaptic conductance and thus membrane potential at only a limited rate, and the coarser discretisation of time in some clocked processes. To some extent the fabricated chip has been constructed based on the particular requirements of the simulations it was intended to model. Providing certain limits are not exceeded the system will perform well, but the chips do not offer an infinitely reconfigurable neural simulator in the same way as a general purpose computer does. This drawback, coupled with the high-expense and long development time necessary to design and fabricate such a system, makes it unlikely that such systems will be of general use to computational neuroscientists, although for simulations of sufficient scale the advantage in terms of speed may ultimately create a demand.

Notwithstanding this, the chips have a limited generality as neural network simulators. One demonstration of this is the ability to use a portion of the address space originally intended for lateral connectivity as input from a second, independent input space. With more chips, arbitrarily complex networks could be constructed, and if the mechanisms for forming probabilistic distributions around an ideal location were not required, arbitrarily shaped or sized neural layers could be implemented. Generalising the circuitry for proximity calculation would therefore be useful future work.

6.3.2 Memory stability

In section 2.1.5.2, theoretical reasons for including both weight and rewiring plasticity in a model of topographic map formation were presented, and evidence suggesting a causal

link between the two were given. Then in section 3.2.3.2, a practical reason was given for creating a causal link between weight and rewiring plasticity. To recap this practical reason: by allowing weight distributions (which can change rapidly and are stored in volatile memory on capacitors) to influence network topology (which changes slowly and is stored in stable memory elements), features learnt from input can continue to influence the behaviour of a network long after the original input has occurred and the immediate memory trace has faded. This process has been demonstrated in section 6.2.3. Specifically it has been shown that by adding a rewiring mechanism to a weight change mechanism, a learnt pattern based on correlations in the input to the network can persist in a stronger form for longer when the correlations which set up the pattern are removed. Although this is shown for a rewiring mechanism which operates on a timescale of tens of seconds (i.e. not much slower than the weight change mechanism) it is intuitively obvious that the same effect should occur if the rewiring mechanism were slowed down further, perhaps to biologically realistic rewiring speeds.

This observation, however, highlights a limitation of this approach, if the rewiring rate is lowered then just as it will take longer for memory traces stored in the network topology to fade, so it will take longer for them to form. The trade-off between learning and forgetting rates for a neural network is an active area of study, for example by Fusi et al. [2005], who proposed that by operating learning within a single synapse at a range of different timescales, an optimum balance of learning and forgetting rates could be achieved. This finding can be related to the mechanism presented here. Synaptic weight change is a fast process, so that patterns of input correlations can quickly be learnt. These patterns will persist in the weights while they persist in the inputs, as shown by figure 6.2. Rewiring by contrast is a slower process, so it should slowly learn persistent patterns in the inputs. Although it has not been demonstrated here, it also seems likely that a pattern which appeared intermittently in the inputs such that it was present on average, would be slowly learnt by the rewiring mechanism. It has also been shown that the rewiring mechanism can allow for an amplification of the strength of the learnt pattern beyond what is achievable by weight changes alone. Thus the system allows both for fast, ephemeral changes and slow, durable changes in the learnt pattern, as shown in figure 6.4.

6.4 Conclusion

In chapter 1 a hypothesis was proposed, that (1) synaptic rewiring can be implemented in neuromorphic VLSI by circuitry distributed throughout the synapses of a neural array, that (2) this ability can be used to model the phenomenon of topographic map formation, and that (3) this ability can increase the stability of patterns learnt by a neuromorphic system. In section 6.2.1, results of experiments using the fabricated chips have been presented, in which the outcomes of statistical significance tests match results from the simulation of a model of topographic map formation presented in chapter 2, thus demonstrating their fitness for the intended purpose. The second part of the hypothesis has therefore been proven, at least to the extent that the many diverse phenomena related to topographic map formation have been captured in the model. In section 6.2.3 the advantage of increased memory duration offered by synaptic rewiring has been demonstrated, proving the third part of the hypothesis. Similarities and differences between the model and its implementation have been discussed; the differences have not been found to undermine the ability of the fabricated chip to model topographic map formation.

Chapter 7

Summary and Conclusions

In this chapter, the work which has been carried out is summarised and recommendations are made for how the work should continue. Finally, conclusions are drawn relating to the hypothesis.

7.1 Work carried out

7.1.1 Model

A model of topographic development was produced which includes both weight and wiring plasticity. There are three key assumptions: (a) synapses preferentially form in locations to which their axons are guided, (b) weights of dendritic synapses of a neuron are modified according to a competitive Hebbian learning rule, and (c) weaker synapses are more likely to be eliminated. Synaptic rewiring is therefore modelled as a pair of stochastic processes. In order to instantiate the model in a form amenable to simulation, more assumptions were made, the main ones being that the neurons are single-compartment integrate-and-fire neurons and the synaptic weight-change mechanism is a form of spike-timing-dependent plasticity. A range of computational simulations were then performed.

A method of analysing map quality was devised whereby the quantities of mean *AD* (absolute distance of the centre of mass of a receptive field from its ideal location), and mean σ_{aff} (variance of the receptive field about its centre of mass) were considered separately, and statistical significance tests were applied to compare developed maps to carefully constructed controls.

It was found that whilst spatially correlated inputs help to create patterns of synaptic weights which favour narrower projections, spatial correlations are not necessary for some reduction of variance to occur. A weight-change mechanism and a rewiring mechanism can work together such that the rewiring mechanism acts to embed patterns of synaptic strengths in the network topology; this is as one would expect, though it has not been demonstrated quantitatively before. It was also demonstrated qualitatively, with the embedding of developed spatial patterns of ocular dominance in the network topology. The accuracy of preferred locations for target neurons will not necessarily improve when synapses are initially distributed around ideal locations.

7.1.2 Circuitry

Circuitry was then developed which is capable of implementing the model. Novel features of this implementation are described here.

A circuit for implementing STDP was developed which introduces a degree of weight-dependence. This works on a different principle to the only other published circuit which explicitly does so, by exploiting the characteristics of existing devices, resulting in a similarly compact implementation. The STDP circuit also uses negative V_{gs} in order to reduce the leakage currents from a weight capacitor down to the limits imposed by the technology, such that a learnt weight distribution decays slowly over a period of minutes. This is long enough to allow learnt patterns to become embedded in the network topology, at least given the artificially increased synaptic rewiring rates which are achievable in a silicon implementation.

The capacitance profile of MOSCAP devices was used in two novel ways. Firstly it was used to increase the broad linearity of charging profiles which are non-linear due to the limited ability of transistors to act as ideal current sources. Secondly it was used to alter the weight-dependence profile of STDP in a way which allows the overall behaviour to match broadly, if not in detail, an established model for STDP, whilst benefiting from the smaller area per unit capacitance offered by MOSCAPs.

A novel design was developed for an address-event receiver, where the decoding elements are distributed through the synaptic array and act simultaneously on broadcast address-events. This allows a spike to be received simultaneously by all the synapses on the axonal arbor, allowing for arbitrarily large axonal arbors to be implemented without reducing

channel capacity. This receiver is compatible with existing address-event senders. The receiver is reprogrammable during run-time, allowing synaptic rewiring to be implemented. The scalability of this design was compared against existing systems with respect to the silicon area, and energy and time required for spike transmission, as numbers of neurons and synapses in a system increase. According to this analysis, the design scales particularly well in terms of speed as synaptic fan-out increases.

An approach was presented of choosing the location of circuitry for neural and synaptic functions based on minimising the need to transmit information. This approach was used to argue for the location of circuitry for implementing synaptic rewiring within each synapse.

Circuitry was developed and results were presented which demonstrate its functioning. Notwithstanding a design error for which solutions have been suggested, the circuit is capable of connecting and disconnecting a synapse in response to either explicit external programming or the probabilistic learning rules of the model.

The probabilistic rule for synapse connection requires a calculation of the distance between a neuron and the ideal location of a potential pre-synaptic partner. Consequently, circuitry for Euclidean distance calculation was developed. This circuitry was based on established principles for calculating Euclidean distance, but it is a novel formulation for the specific task. Its notable features include current mode operation across multiple chips and the capability of implementing both toroidal and non-toroidal topologies. The rewiring circuitry and distance calculation circuitry together allow the formation of radially symmetric receptive fields with arbitrary monotonically decreasing relationships of connection probability to distance from the centre.

The circuitry described above was implemented in the form of fabricated silicon chips using the AMS $0.35\mu m$ process. 8 of these chips were connected together, both in a grid arrangement for spike delivery, and in the arrangement of a neural layer for distance calculation. This system was used to demonstrate the development of receptive fields, in experiments analogous to those carried out in simulation. Analogous significance tests were performed demonstrating qualitatively similar performance to that of the simulated model. Additional results were gathered relating to learning stability.

7.2 Future work

7.2.1 Model

Although the model was developed primarily as a basis for a neuromorphic system rather than as a tool for computational neuroscience research, its contribution to computational neuroscience can nevertheless be evaluated. For example, the conclusion that activity dependent topographic refinement simply enhances an existing trend produced by an activity-independent mechanism echoes the conclusion of Butts et al. [2007], whilst the observation that receptive field spread reduces without input correlations is similar to the findings of Linsker [1986b] and Miikkulainen et al. [2005] that functional architecture can form in the absence of any input except uncorrelated random noise, although an alternative mechanism is suggested. The model has combined some suggestions previously made in the literature about the link between synaptic weight change and synaptic rewiring in order to demonstrate quantitatively a way in which the two can work together which is consistent with many findings in the biological literature.

However, the model has not been developed to the point where it can make experimental predictions. For example, it would be tempting to predict that in amphibia or fish, the centre of mass of the retinal receptive field for a neuron of the optic tectum, (as judged by the locations of the RGCs which are afferently connected to it, perhaps judged by retrograde tracing from a single cell) will be no closer its ideal location (as judged by a smoothed average of all projections, perhaps established by functional optical imaging) later in development than in a constructed topography based on addition or elimination of synapses to the final observed topography according to the distribution observed earlier, irrespective of the smaller relative spread of the receptive fields later in development. Various methodological issues may prevent such an experiment from succeeding. Then, if this turned out to be true, it might lend weight to the idea that receptive field refinement is simply Hebbian reinforcement of random tight clusters in a receptive field which is initially randomly formed based on developmental pressure towards an ideal location dictated by guidance molecules. However if it turned out not to be true this would not necessarily falsify that hypothesis, because it might be that lateral interactions in the optic tectum increase pressure towards an ideal location; the apparent lack of effect of lateral interactions in the simulations performed may be because of the very small scale at which the model has been implemented or it may be because the learning rule applied does not sufficiently re-

ward bursting behaviour, as suggested by Butts et al. [2007]. Thus in order to develop this model further, there should be further experimentation with lateral interactions at various scales, and with alternative Hebbian learning rules.

More generally, further experimentation with different sizes of neural layers, variances of receptive fields, numbers of afferent synapses and other parameters would all help to develop its credentials as a generalised model.

A possible way this model could be extended is to allow that axon branching should be guided by the existence of axons, such that an input neuron is more likely to form synapses with target cells which are close to target cells which it is already innervating. This might be expected to model axonal arbor development more accurately. In addition there is no mechanism in this model for the preferential sprouting of synapses due to potentiation, as suggested by some literature. As is common in this field, there has been no consideration of the spatial and temporal summation and filtering performed by transmission of post-synaptic potentials along dendritic trees. Such considerations are likely to be important for a complete understanding of map formation, especially since it is known that temporal learning windows can be different at different locations on the dendritic tree. The development of a mapping from the start has not been simulated, nor has the growth of areas been addressed. Rather, this model assumes a starting time at some point during development, in order to assess the effect of the proposed learning rules. Additionally, initialising weights at their maximum is unlikely to reflect the reality of synapse development.

In order to model both the growth of areas and the effects of lesions on redevelopment, a promising possibility would be to combine the mechanisms used in this model with the activity-independent process described by Willshaw [2006]. This would involve replacing the probabilities of synapses forming which are currently based on distances from fixed ideal locations, with probabilities which Willshaw modelled as synaptic strengths but which represent affinity of an axon towards a particular target location. These affinities change according induced levels of ephrin ligands, which in turn change according to a developmental rule which can allow for growing areas and for abnormalities of the types introduced by lesion studies. More generally, the probability of synapse formation is a promising place in which to intervene within this model in order to somehow capture the aforementioned effects. This could potentially yield a model where the topography could develop and the areas themselves could grow, which could replicate compression and expansion studies etc, and which in addition could allow for the formation of func-

tional architecture such as ocular dominance, based on statistical structure in input spike trains.

Adding all these abilities would not necessarily make sense for computational neuroscience, since it could result in a model which mimicked everything and explained nothing. However, as a basis for the development of neuromorphic systems, where engineering solutions are expected to be generated though an attempt to mimic biology, it could prove fruitful.

7.2.2 Circuitry

The distributed address-event receiver which has been implemented breaks new ground, yet it may benefit from redesign, in any of the following ways. Firstly, stored bits of addresses might be stored on floating gates to achieve non-volatile storage. Analogue storage of many address bits on a single gate could be explored for a possible space saving. Secondly, the adoption of word-serial AER could limit the number of receiver elements in the synapse since they could be used to compare a word at a time against words stored in S-RAM cells in order to receive an entire address-event. Alternatively, adopting a monitor bit design similar to that of standard Content Addressable Memory could achieve a space saving. Both of these suggestions would come at the expense of a more complex comparison cycle and both may lead to the adoption of a standard S-RAM cell for bit storage.

The present transmission gate implementation of the pulse receiver has a drawback whereby if an unusually large number of synapses within a row are connected to a particular presynaptic address, the capacitance on the broadcast signals will increase. This is a likely cause of the problem with ocular dominance pattern formation shown in section 6.2.4. This could be avoided with the use of active gates. Thirdly however, a more generally useful improvement than the previous one would be to decouple the delivery of spikes from the implementation of neural processes. At the moment the delivery of spikes is by a pulse (in fact, a set of them), the precise lengths of which are used to deliver measured amounts of charge. This causes two problems: (a) it prevents the spike broadcast and receivers from being optimised for speed; (b) it appears to have introduced systematic variation in the performance of neurons due to the limitations of the clock distribution network. Some of the processes in the neuron and synapse have been made insensitive to the duration of

controlling pulses by the use of switched capacitors, and this approach could be extended to cover all of the analogue processes of the neuron and synapse. This would require a state machine to be built into each synapse (and into each neuron) capable of generating a pair of non-overlapping pulses in response to a single brief pulse; however since the pulses would not need to be of precisely measured duration these could be created by circuits which use minimum geometry transistors. The present design uses two pulse generators in the design of the neuron which contain capacitors of greater than minimum geometry (in order to create controlling pulses for STDP), so there would be a potential space saving in the neuron. A possible design is outlined in figure 7.1.

Regarding the use of switched capacitors, those which have been used to implement exponential decays could be replaced with parasitic insensitive circuits at the expense of slightly more area; eliminating parasitic capacitances would also eliminate mismatch in the parasitic capacitances. Moreover, although the circuit presented in section 3.4.3 for generating membrane currents achieves a slight gain in compactness by means of combining a linear integration with an exponential decay, the disadvantages of sensitivity to mismatch and relatively restrictive parameterisation may outweigh the benefit, compared to the combined use of a parasitic-insensitive switched capacitor transresistor and integrator.

The implementation of synaptic rewiring rules locally at each synapse has the potential to minimise intra- and inter-chip communication, especially if, in a mature implementation, the probabilistic processes necessary to drive the learning rules could be generated on chip. Two interesting avenues to explore in this respect would be (a) the possibility of amplifying analogue noise and then shaping it, and (b) the modification of the connection rule to use the most recently delivered spike as a potential pre-synaptic partner. Notwithstanding this, alternatives should be considered. If the rewiring rules were implemented in shared peripheral circuitry there could be a substantial space saving. A hybrid approach whereby synapse elimination is performed locally at the synapse whereas connection is a globally controlled process may be a beneficial approach. Centralising the control of rewiring, or at least some central circuit for receiving information about rewiring which occurs, would be a pre-requisite for using the chips in a network of the type envisaged by Khan et al. [2008], where address-events are routed only to chips where they are required. The circuit which implements the connection rule contains a design error which corrupts a small proportion of the addresses stored; a solution for this has been proposed and should

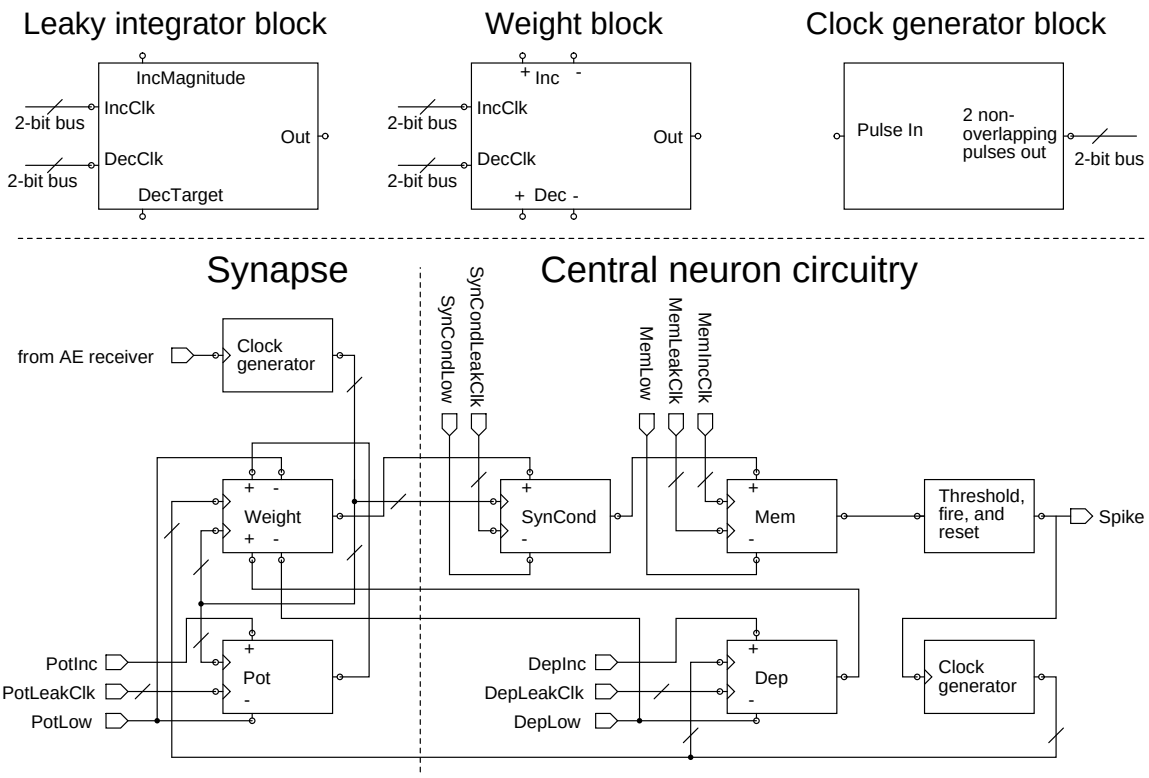


Figure 7.1: Pulse-duration-insensitive neuron — high-level design. The synapse and neuron are mainly composed of two repeating blocks — a leaky integrator and a clock generator, whose I/O ports are shown above, but whose detailed workings are left unspecified here (although switched capacitor function is intended and designs are suggested in section 3.4.3). The leaky integrator maintains an internal voltage across a capacitor which is then buffered to the *Out* port. One complete cycle of the pair of non-overlapping clock signals *IncClk* increments the internal voltage by some proportion of the difference between *IncMagnitude* and *DecTarget*, whereas a complete cycle of *DecClk* decrements the internal voltage by some proportion of the difference between the internal voltage itself and *DecTarget*. This block is used for *SynCond*, *Mem*, *Pot* and *Dep*. The *Weight* block would differ according to the exact learning rule desired; in the version shown, one complete cycle of *IncClk* would increment *Weight* by some proportion of the difference between *Inc+* and *Inc-*, and likewise for the decrement caused by a cycle of *DecClk*. The clock generator block in the synapse takes a single pulse which has been received and outputs two non-overlapping pulses on a 2-bit bus (all the buses shown in the diagram convey such clock signals), triggering a decrement to *Weight* and an increment to *SynCond* and *Pot*. Likewise the clock generator in the neuron triggers an increment to *Dep* and an increment to *Weight*. All other processes are governed by global clock signals.

be included if the circuit is used again.

A preliminary discussion of the scalability of the proximity calculation circuitry has been given in section 5.6, including suggestions on how to minimise variability. If the use of this circuit is perpetuated, these issues should be explored more thoroughly, in order to assess the ultimate potential of the circuit.

It would be interesting to investigate the possibility of implementing non-trivial mappings between areas and the development of such mappings. Two methods that might be used to achieve this are: (a) applying a transformation to the addresses of address-events; (b) replacing the resistors of the proximity calculation circuit with transistors and then dynamically controlling their resistances in order to skew distances across the map.

The homeostatic properties of STDP appear to have been useful in this implementation. Numerous non-idealities of the proposed circuits were known in advance of fabrication, but simulations of the map formation model which included variation extracted from monte-carlo simulations showed that the neural algorithm was robust against them. This robustness held true in practice; a notable example is the way that STDP acted to compensate for systematic variation due to a limitation of the clock distribution network. Therefore, whilst attempts to minimise the effects of mismatch and noise are good practice, it may ultimately prove more fruitful for neuromorphic engineers to attempt to implement other mechanisms of neural homeostasis, in order to take full advantage of further silicon miniaturisation.

The potential benefit of using synaptic rewiring to stabilise learnt memory traces has been demonstrated. Nevertheless, the limitations of this approach in terms of the trade-off between learning and forgetting rates for a neural network is apparent from the theoretical literature. Therefore it may be useful to combine the system proposed here with other attempts to overcome this problem, for example the semi-supervised learning scheme of Brader et al. [2007].

7.3 Conclusions

This project was an investigation of the hypothesis that (1) synaptic rewiring can be implemented in neuromorphic VLSI by circuitry distributed throughout the synapses of a neural array, that (2) this ability can be used to model the phenomenon of topographic

map formation, and that (3) this ability can increase the stability of patterns learnt by a neuromorphic system. Circuitry has been developed within the context of a neuromorphic system, in which synaptic rewiring is implemented by circuitry distributed throughout the synapses of a neural array. This circuitry has been integrated in silicon chips and these have been used to demonstrate synaptic rewiring, thus proving the first part of the hypothesis. A model of topographic map formation has been developed, which focuses on the interplay between synaptic weight change and synaptic rewiring, and its relevance to the current developmental neuroscience literature has been argued for. Instantiations of the model were simulated and statistical tests as well as qualitative observations were used to demonstrate its core features. Similar instantiations of the model were implemented using the fabricated chips, and the application of the same statistical tests achieved the same results (there were also notable similarities in qualitative observations), thus proving the second part of the hypothesis, at least to the extent that the many diverse phenomena relating to topographic map formation have been captured by the model. In separate experiments it has been shown that with the addition of the synaptic rewiring mechanism to the synaptic weight change mechanism, a learnt pattern based on correlations in the input to the network can persist in a stronger form for longer when the correlations which set up the pattern are removed; this has proven the third and final part of the hypothesis.

Appendix A

Neuron circuit

The circuitry which implements the threshold, spike and reset mechanism of the integrate-and-fire neurons is based on that described in Indiveri [2003b] and can be seen in figure A.1. The main difference is that a differential amplifier is used instead of a source follower to set the threshold, allowing a greater range of thresholds to be chosen without implications for power consumption.

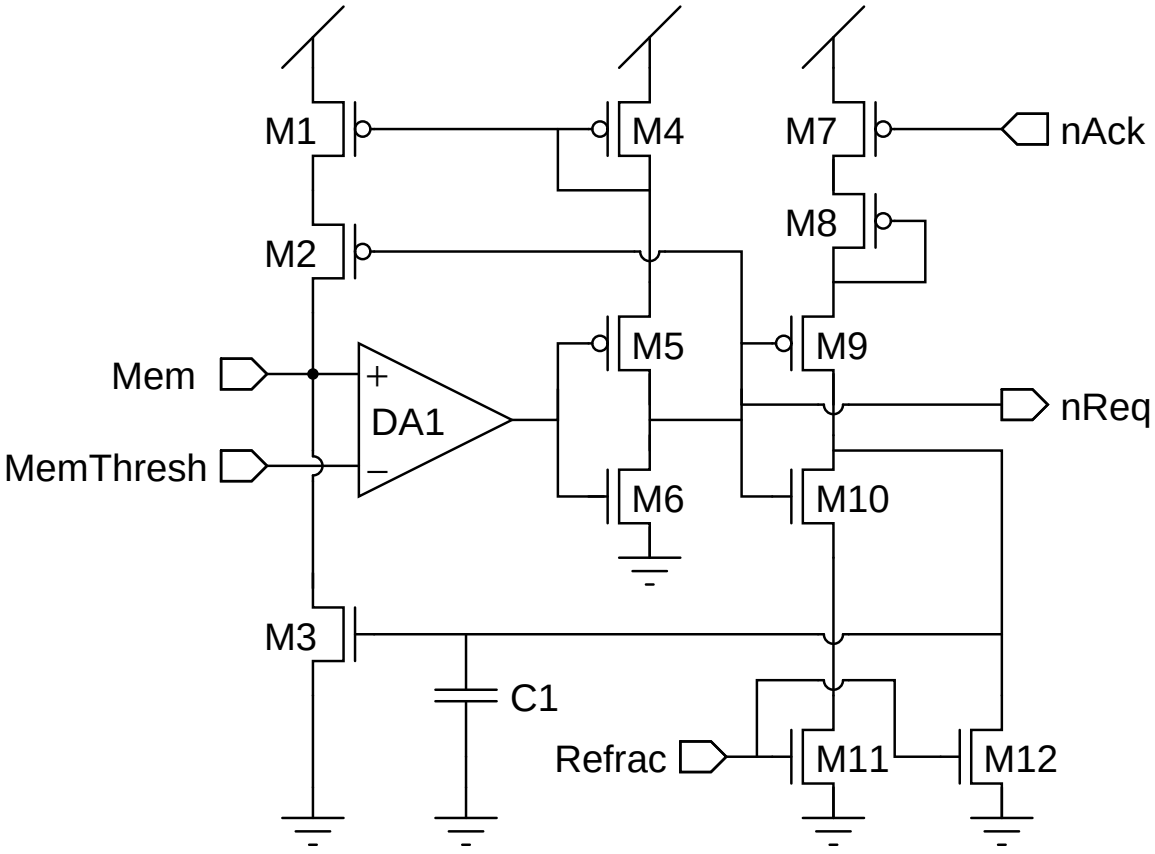


Figure A.1: Threshold, spike and reset circuit. Synaptic currents (not shown) flow into and out of the capacitance of the membrane (the *Mem* node). When the voltage of *Mem* rises to approach the threshold *MemThresh*, the output of DA1 starts to rise above *Gnd* (this circuit differs from that presented in Indiveri [2003b] only in that this amplifier replaces a source follower). This causes current to start to flow through M6 and consequently through M4. The current in M4 is mirrored onto M1, sourcing current into *Mem*; this causes positive feedback which quickly pushes the output of DA1 high, cutting off the current through M5. *nReq* is therefore lowered and this triggers a request to spike-sending circuitry. It also triggers pulse generators which create two timed post-synaptic pulses used in the STDP mechanism, as described in section 3.4.4. The output of inverter M9-M10 however is prevented from rising until *nAck* is lowered, indicating that the spike has been successfully delivered; during this period it is held weakly low by a subthreshold current through M12 set up by the *Refrac* bias. Once the output of M9-M10 rises, C1 is charged and *Mem* is discharged through M3. The current through M9 is therefore shut off and the output of M9-M10 is restored to *Gnd* through a subthreshold leak current through M11 set up by the *Refrac* bias. This implements an absolute refractory period which can be altered by adjusting *Refrac*.

Appendix B

Pulse generators and bias generators

The circuits which have been described depend on a set of precisely timed pulses and accurate voltage biases for correct operation. The design of pulse generator used is similar to that described by Bofill-i Petit [2005, page 60] so is not sufficiently novel to warrant detailed description here. Essentially, a transistor is biased to generate a current from a capacitor to *Gnd*. Upon a trigger which marks the start of the pulse, the capacitor is raised to *Vdd*. Thereafter the capacitor drains until a threshold is crossed, at which point the pulse finishes. By varying the bias voltage applied, the duration of the pulse can be varied over many orders of magnitude. The pulse generator implemented has a minimum pulse duration of about 8ns.

The pulse generator circuit described above is one example of a circuit which needs a voltage bias in order to generate a current. However transistors can be expected to be mismatched in their threshold voltage both between transistors on the same chip and especially between transistors on different chips. A voltage applied to the gate of two identical transistors will therefore generate different currents through them. This is especially a problem in multi-chip systems as the variations between different chips are typically more extreme, meaning that a bias voltage generated externally and applied equally to two different chips will give rise to different currents within the chips. In the example above this would equate to pulses of different lengths, which might mean, for example, that the synapses on one chip are more likely to become depressed than those on another chip. It would be possible to generate a different set of bias voltages for each chip used and calibrate them independently in order to match performance, but this approach would soon become intractable as the number of chips in a system increases. This problem has been

circumvented by employing programmable bias generator circuits as described by Delbrück and Lichtsteiner [2006]. A reliable master current is generated on each chip, which is then mirrored and subdivided according to digital words (these are initially streamed onto the chip with a shift register). The resulting currents are then converted to voltages by transistors whose dimensions and source voltages are matched to those transistors across the chip which the voltages are then used to bias. This scheme does not eliminate mismatch between circuits within a single chip though it should greatly reduce mismatch between chips.

Appendix C

Layout considerations

C.1 Area usage

The chips were fabricated using the AMS 0.35u 4-metal 2-poly process. The area of the synaptic address monitor bit is $11.1\mu m \times 15.95\mu m = 177\mu m^2$. The system simulated has 512 neurons i.e. grids of 16×16 neurons in each of two layers. The target layer is fabricated whereas the input layer is simulated; nevertheless a 9-bit address is required to uniquely identify neurons. Therefore each synapse has a 9-bit receiver. This receiver takes up 56% of the total synapse area, which is $11.1\mu m \times 255.95\mu m = 2841\mu m^2$. The remaining area is dedicated to: storing the additional synaptic variables; implementing the connection and disconnection circuitry; creating an increase in the neuron's level of synaptic current when a spike arrives; and implementing spike-timing-dependent plasticity. Each neuron has 64 potential synapses, and the synaptic array takes up 98.6% of the total area of the neuron ($740.275\mu m \times 255.95\mu m = 0.189mm^2$), where the remaining area is dedicated to the storage of the neuron's variables, its central (integrate and fire) functions and its sending circuitry. Each chip is of total area $\approx 14mm^2$, of which $\approx 6mm^2$ is used to implement neurons and the remaining area is dedicated to peripheral circuitry, i.e. pads, bias generators, AER input buffers and timing control, AER output arbiters, peripheral cells for proximity calculation and rewiring, clock buffers, analogue output buffers, etc. In order to achieve the required number of neurons for a layer, 8 chips were required. The choice of die-size reflects a budgetary constraint; for larger-scale production systems larger die sizes would be more efficient since there would be a greater ratio of chip area dedicated to implementing neurons *vs.* peripheral circuitry.

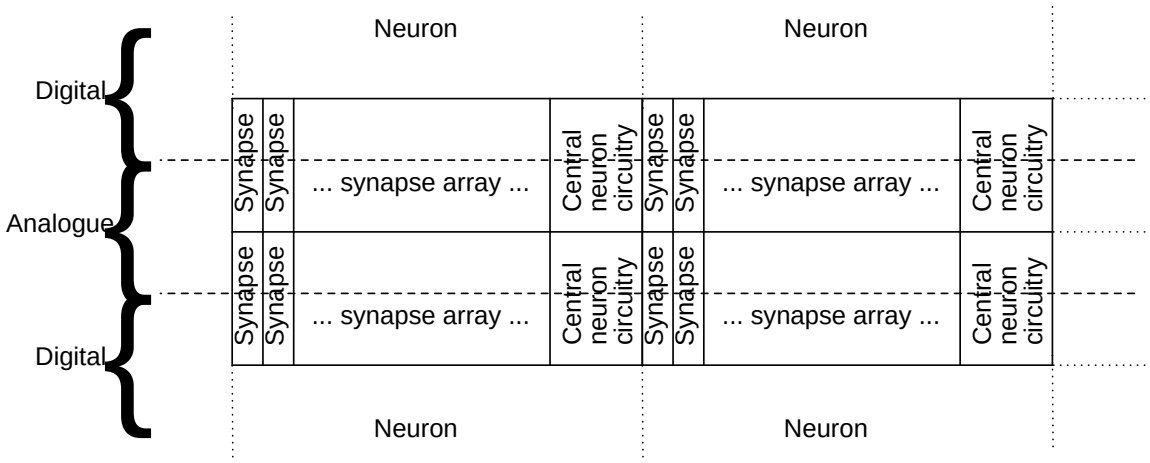


Figure C.1: Layout scheme to maximise separation of analogue and digital domains. Within a neuron, all synapses are arranged in a single row. Both synapses and the central neural circuitry are designed with vertical separation between analogue and digital circuitry. Every alternate row of neurons is flipped vertically to double the average separation of digital and analogue circuitry. Thus, there are broad horizontal bands of alternating digital and analogue circuitry. Where possible, routing of global and shared signals and biases is horizontal across the chip, within the appropriate bands.

C.2 Digital and analogue separation

In order to reduce the coupling of digital onto analogue signals, maximum separation was sought between circuitry which carried predominantly analogue signals and that which carried predominantly digital signals. This was achieved as shown in figure C.1. Other standard methods of reducing coupling included the use of separate digital and analogue power supplies which are only brought together outside the chip, and the use of separate n-wells with appropriate bulk connections for pMOSFETs and guard rings of substrate contacts for areas of the bulk containing nMOSFETs carrying analogue signals.

C.3 Energy cost of spiking

The chips each contain 2048 synapses and based on a simulation including capacitances extracted from layout, each synaptic address-monitor consumes per incoming spike: 227fJ for delivery of the spike signal; 69.6fJ per address bit (assuming that each incoming address bit makes a transition with 50% probability each spike); and 5fJ pumped through

the NAND gate that determines whether the correct address has been received. The 9-bit address-monitors therefore consume 859fJ each per spike. This figure includes internal buffering to the neurons but does not include buffering from the pads to the peripheral latches.

C.4 Switched capacitor clocks

As noted in section 3.4.5, 4 pairs of non-overlapping clock signals are used to implement conductances. To reduce the impact of transients on the power rails and other biases due to parasitic capacitances from the clocks, the pulses are slowed down by (a) being buffered across the chip with weak inverters and (b) being routed in minimum-width polysilicon. This should ensure that the pulses rise and then fall in a wave which propagates slowly across the chip. Although the chip was not designed in such a way as to allow this feature to be tested, simulations suggest that the wave should take about 100ns to propagate across the chip.

Appendix D

General parameters for chip experiments

General parameters given in table D.1 were used in all chip experiments unless otherwise noted.

Name	Value
Clock periods:	
Pot	1330μs, so that $\tau_+ \approx 20ms$
Dep	1490μs, so that $\tau_- \approx 64ms$
SynCondLeak	80μs, so that $\tau_{ex} \approx 5ms$
SynCond	358μs, so that $\tau_m \approx 20ms$
Voltages from external DAC	
MemThresh	0.89V*
nWeightLowThr	1.95V
*GeneralHigh***	3.1V, i.e. $Vdd - 0.2V$
*GeneralLow***	0.2V, i.e. $Gnd + 0.2V$
DistGenPu	2.5V
DistGenPd	0.4V
Currents from on-chip generator	
Refrac	10nA, giving refractory period $\ll 1ms$
PrePulseSynCond	4μA, giving a pulse of $\approx 33ns$
PostPulseDep	19.3μA, giving a pulse of $\approx 9ns$
PostPulsePot	19.3μA, giving a pulse of $\approx 9ns$
PrePulsePot	19.3μA, giving a pulse of $\approx 9ns$
PrePulseDep	6μA, giving a pulse of $\approx 23ns$
nDepMin	1nA
PotMin	1nA

Table D.1: General parameters for chip experiments

* *MemThresh*: 0.89V would be the correct value to implement a threshold of -54mV if *Vdd* is defined as the excitatory reversal potential of 0V and *Gnd* is defined as the resting potential of -70mV. However, as there is no practical need for these conventions to be respected, the value of *MemThresh* is an arbitrary choice.

** *MemLow* and *SynCondLow* are constrained to be the same bias, as a condition for the correct functioning of the circuit for membrane currents (see section 3.4.3), and *nWeightHigh* is also included as the same bias in the fabricated chip, for reasons of economy; they are collectively referred to as *GeneralLow*. Likewise *SynCondHigh* and *nWeightLow* are collectively *GeneralHigh*.

Appendix E

Parameterisation of neuronal processes

In this section the parameterisation process is described which yielded clock periods and pulse durations which give appropriate values for τ_m and g_{max}. Other parameters can be set in similar ways.

E.1 Setting τ_m

The period of *SynCondClk* was set to a guessed value of $400\mu s$ and membrane decays were observed for neurons on different chips, as described in figure E.1. The time it took for *Mem* to decay from 0.8V down to 1/e of this level with respect to the recorded mean resting potential was noted. Table E.1 gives 3 such data points. There is a certain variation in both the recorded mean resting potential and in the time constant. The variation in the mean resting potential is probably due to mismatch in the amplifiers in figure 3.9, (though there may be some contribution from additional amplifiers which buffered the *Mem* signal out to pads), whilst the apparent offset of the mean values above the intended 0.2V is likely due to the small remaining currents into *SynCond* through transistors M1-2 in figure 3.4, as judged by the difference between the resting levels of *SynCond* and *Mem* which can be observed in figure 3.14 (recordings over 8 chips give average resting levels for *SynCond* and *Mem* as 0.3V and 0.26V respectively); (there could also be a contribution from from differences in performance between the off-chip DAC that created the bias of 0.2V and the ADC that recorded the output). From the mean τ_m which was given by a

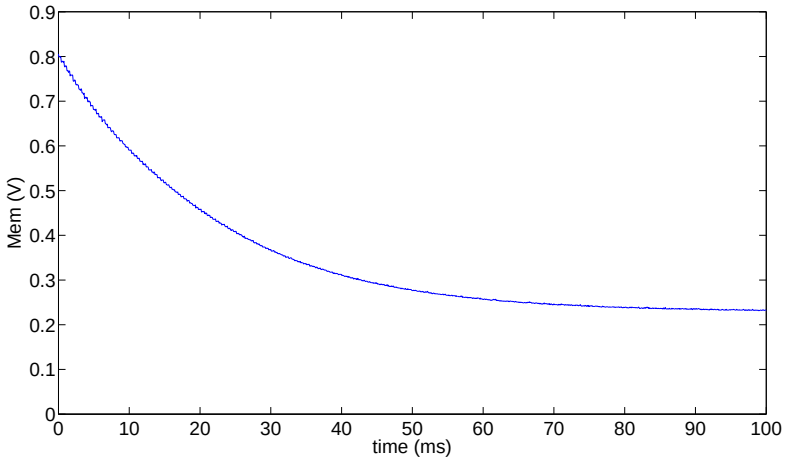


Figure E.1: Membrane decay. *SynCondLeakClk* was given a very short period whilst the membrane threshold was raised to a high level, then a fast burst of spikes was delivered to one synapse whilst synaptic depression was inhibited. Thus *Mem* was raised to a high level and was then allowed to drop whilst *SynCond* quickly decayed to its resting level of *GeneralLow* $= 0.2V$. *SynCondClk* period $= 400\mu s$. The decay of *Mem* from 0.8V is shown.

Mean resting potential (V)	1/e voltage w.r.t. 0.8v (V)	Time of decay to 1/e voltage (ms)
0.2303	0.4399	20.7
0.2797	0.4711	22.2
0.2434	0.4482	22.0

Table E.1: Membrane decay time constants for $400\mu s$ clock period for three neurons on different chips

known *SynCondClk* period, it is possible to derive the *SynCondClk* period which will give any desired value of τ_m by straightforward multiplication.

E.2 Setting g_{max}

The next step in parameterising the neuron is to set g_{max} i.e. the instantaneous rise in conductance upon a spike to a synapse of maximum weight; this is defined in terms of the leak conductance and the desired value is $g_{max} = 0.2g_{leak}$. Thus if 5 spikes arrive simultaneously to synapses with maximum weight, the level to which *SynCond* rises should be such that if it were sustained, V_m would stabilise halfway between V_{rest} and E_{ex}, in the

ideal model, if it were not reset due to crossing its threshold. This is because at that point the excitatory and leak current would be equal:

$$g_{ex}(V_{ex} - V_m) = g_{leak}(V_m - V_{rest})$$

Substituting $g_{leak} = 5g_{max}$ and rearranging:

$$g_{ex} = 5g_{mem}\frac{V_m - V_{rest}}{V_{ex} - V_m} \tag{E.1}$$

The level of *SynCond* at which output spikes start to occur, i.e. the level which achieves the rheobase current, was determined experimentally to be 0.58V, as shown in section 3.5.1. The rheobase current causes V_m to stabilise just on its spiking threshold of -54mV in the ideal model. So using equation E.1, a sustained excitatory conductance of:

$$g_{ex} = 5g_{mem}\frac{-54mV + 70mV}{0V + 54mV}$$

$$g_{ex} = 1.48g_{mem}$$

should cause *Mem* to stabilise at its threshold voltage (if it were prevented from firing and resetting). Therefore, 1.48 (≈ 1.5) pulses to maximum strength synapses should raise *SynCond* from its resting potential of $\approx 0.3V$ to its rheobase level of 0.58V. Using this guideline it was possible to set the *nPrePulseSynCond* pulse in figure 3.4 at an appropriate duration, approximately 27ns for the fabricated chip. The design could be optimised for speed by sizing transistors M1-2 wider and using correspondingly shorter pulses.

Appendix F

Limits of integration

The integration performed on nodes such as *Pot* is approximately linear only up to a maximum level, beyond which further inputs cease to have any effect. This applies to all nodes where inputs are integrated and subject to an exponential decay, namely, *Pot*, *nDep* and *SynCond*. For a periodic input it would be possible to calculate the maximum input rate which could be sustained without overloading the respective node. However for Poisson distributed spike trains, as used extensively in computational neuroscience, including in the model being implemented, it is always possible for a burst of spikes to occur which is sufficiently dense as to overload the input node. However, with careful design, it should be possible to allow a large enough number of spikes to be integrated instantaneously that a burst with more than that number of spikes would be sufficiently rare within the intended inputs as to not adversely affect the operation of the neural network being implemented. This was the approach taken when choosing the sizes of capacitors and charging transistors used in the fabricated chip. For example, figure 3.19 shows 17 spikes taking Pot from its minimum to its maximum. Conservatively it might be seen as desirable not to allow more than 17 spikes to arrive during a 20ms period, in which Pot should decay to 1/e of its initial value. The peak spike rate used in any of the simulations was $\approx 300Hz$ (not continuously but only for brief periods for any one neuron). By integrating the Poisson distribution it can found that for a spike rate of 300Hz, 18 or more spikes will arrive within 20ms with a probability of just 5.6×10^{-5}. From this, the likelihood of occurrence could be calculated for a given simulation period and size of network. In fact, the incoming spike trains are not truly Poisson-distributed, since there is a maximum rate allowed by the spike delivery circuitry (see section 4.6). In the same way there is additional scope for

the relaxation of such constraints with regards to *nDep*, since it is driven by post-synaptic spikes and the rate at which these can be generated is limited by the ability of the clocked synaptic conductance circuitry to raise *Mem* to its threshold.

Appendix G

Creating random input distributions

In order to generate the signal *nProbConnect*, equation 2.2 was re-arranged for distance:

$$distance < Re\left(\sqrt{-2\sigma^2 ln\left(\frac{r}{p_{form}}\right)}\right)$$

This condition was then put in terms of *Proximity*, using the peak *Proximity* voltage and its gradient with respect to distance, taken from results such as those given in figure 5.7:

$$Proximity > PeakProximityVoltage - abs\left(\frac{\delta ProximityVoltage}{\delta distance}\right).Re\left(\sqrt{-2\sigma^2 ln\left(\frac{r}{p_{form}}\right)}\right)$$

The term on the right was then used to generate values for *nProbConnect*, based on the random number *r*. *nProbConnect* was constrained to a minimum value of 0V, which guarantees connection, since *Proximity* cannot go so low. Conversely if *r* is greater than p_{form} then *nProbConnect* was set to 3.3V, which guarantees that connection does not occur. A similar procedure was used to create *nProbDisconnect*, based on the chosen relationship of weight to elimination probability.

Appendix H

Reading connectivity

The method used to read the final connectivity following rewiring is as follows. The potential for depression was maximised by raising the current that generates the bias *nDepMin*. Then for each synapse in turn, the following steps were carried out. *nWeight* was lowered towards *Gnd* for all synapses across the system, by temporarily lowering *nWeightLowThresh*. The synapse had its *Compare* signal raised; this doubled as a signal for the synapse to buffer out its *nWeight* value onto a global node, using an amplifier included for testing purposes only. *Meanwhile nProbConnect* and *nProbDisconnect* were maximised; thus the weight of the synapse could be recorded without changing its connectivity. An event was then sent in for each source address in the system, until *nWeight* rose, indicating that the synapse had received the event and had become depressed. This process, when automated, took up to 45s to complete for all 16384 synapses. This method was destructive to weights and could not be used during a simulation but only after the simulation had finished and weights had been read.

Appendix I

Publications

Peer-reviewed conference papers

1. SA Bamford, AF Murray, and DJ Willshaw. Synaptic rewiring for topographic map formation. *International Conference on Artificial Neural Networks (ICANN)*, 2008

2. SA Bamford, AF Murray, and DJ Willshaw. Large developing axonal arbors using a distributed and locally-reprogrammable address-event receiver. *International Joint Conference on Neural Networks (IJCNN)*, 2008.

Synaptic Rewiring for Topographic Map Formation

Simeon A. Bamford[1], Alan F. Murray[2], and David J. Willshaw[3]

[1] Doctoral Training Centre in Neuroinformatics, sim.bamford@ed.ac.uk,
[2] Institute of Integrated Micro and Nano Systems
[3] Institute of Adaptive and Neural Computation,
University of Edinburgh

Abstract. A model of topographic map development is presented which combines both weight plasticity and the formation and elimination of synapses as well as both activity-dependent and -independent processes. We statistically address the question of whether an activity-dependent process can refine a mapping created by an activity-independent process. A new method of evaluating the quality of topographic projections is presented which allows independent consideration of the development of a projection's preferred locations and variance. Synapse formation and elimination embed in the network topology changes in the weight distributions of synapses due to the activity-dependent learning rule used (spike-timing-dependent plasticity). In this model, variance of a projection can be reduced by an activity dependent mechanism with or without spatially correlated inputs, but the accuracy of preferred locations will not necessarily improve when synapses are formed based on distributions with on-average perfect topography.

1 Introduction

The development of topographic mappings in the connections between brain areas is a subject that continues to occupy neuroscientists. There have been a number of investigations of the development of maps through networks with fixed connectivity and changes to synaptic weights [1–5]. Other models have considered the formation and elimination of synapses with fixed weight [6]. Indeed a mathematical equivalence between such models has been demonstrated for certain conditions [7]. There have been few attempts to include both forms of plasticity in a model (though see [8, 9]) however since both forms of plasticity are known to exist, we have created a model of topographic map development which combines both forms of plasticity and we explore some of the consequences of this model. This work is part of a project to implement synaptic rewiring in neuromorphic VLSI [10], however the results presented here are purely computational.

Theories of topographic map formation can be divided by the extent to which activity-dependent processes, based on Hebbian reinforcement of the correlated

activity of neighbouring cells, are deemed responsible for the formation of topography. Some assume that activity-independent processes, based on chemoaffinity [11] provide an approximate mapping, which is then refined [12]. Others [5] show how activity-independent processes may fully determine the basic topography, thus relegating the role of activity-dependent processes to the formation of "functional architecture" e.g. oculardominance stripes etc. [13]. Our model is in the latter of these categories, assuming that synapses are placed with on-average perfect topography by an activity-independent process. Miller [7] gives evidence that the decision whether newly sprouted synapses are stabilised or retracted may be guided by changes in their strengths; this is a basis for our model.

2 Model

This generalised model of map formation could equally apply to retino-tectal, retino-geniculate or geniculo-cortical projections. There are 2 layers (i.e. 2D spaces on which neurons are located), the input layer and the network layer. Each location in one layer has a corresponding "ideal" location in the other, such that one layer maps smoothly and completely to the other. For simplicity neural areas are square grids of neurons and the 2 layers are the same size (16 x 16 in the simulations presented here). We have worked with small maps due to computational constraints; this has necessitated a rigorous statistical approach. Periodic boundaries are imposed to avoid edge artefacts.

Each cell in the network layer can receive a maximum number of afferent synapses (32 in our simulations). Whilst we acknowledge arguments for the utility of inhibitory lateral connections in building functional architecture [8] we simplified our model using the finding [4] that a topographic projection could form in the absence of long-range lateral inhibition. Thus, two excitatory projections are used, a feed-forward and a lateral projection; these projections compete for the synaptic capacity of the network neurons. We assume that an unspecified activity-independent process is capable of guiding the formation of new synapses so that they are distributed around their ideal locations. We assume a Gaussian distribution, since a process which is initially directed towards a target site and then randomly branches on its way would yield a Gaussian distribution of terminations around the target site. Our model does not specify the underlying mechanisms that cause an axon to be guided towards an ideal location. Thus it is not fundamentally incompatible with lesion studies which show shifts or compression of maps, but rather, to achieve such reorganisation some mechanism for specifying and changing ideal locations would need to be added.

To implement the Gaussian distributions, where a network cell has less than its maximum number of synapses, the remaining slots are considered "potential synapses". At a fixed "rewiring" rate a synapse from the neurons of the network layer is randomly chosen. If it is a potential synapse a possible pre-synaptic cell is randomly selected and synapse formation occurs when:

$$r < p_{form}.e^{-\frac{\delta^2}{2{\sigma_{form}}^2}} \tag{1}$$

where r is a random number uniformly distributed in the range $(0, 1)$, p_{form} is the peak formation probability, δ is the distance of the possible pre-synaptic cell from the ideal location of the post-synaptic cell and ${\sigma_{form}}^2$ is the variance of the connection field. In other words, a synapse is formed when a uniform random number falls within the area defined by a Gaussian function of distance, scaled according to the peak probability of synapse formation, (which occurs at $\delta = 0$). This is essentially a rejection sampling process.

Lateral connections are formed by the same means as feed-forward connections though σ_{form} is different for each projection and p_{form} is set correspondingly to allow the same overall probability of formation for each projection. In the absence of a general rule for the relative numbers of feed-forward vs lateral connections formed, starting with equal numbers of each is a good basis for observing the relative development of these projections; $\sigma_{form-feedforward}$ is given a larger value than $\sigma_{form-lateral}$, in line with generic parameters given in [8].

If the selected synapse already exists it is considered for elimination. In general we propose that the probability of elimination should be some monotonically decreasing function of weight. Due to the nature of the learning rule we have chosen (STDP; see below in this section), which tends to deliver a bimodal weight distribution, we have simplified probability of elimination to one of 2 values with a higher value for synapses with weights below a certain threshold ($p_{elim-dep}$) and vice versa ($p_{elim-pot}$). Data is scarce on appropriate values for these probabilities, however dendritic spines have been imaged extending and retracting over periods of hours compared with others stable over a month or more [14]. We have used much higher rates so that synapses have several chances to rewire during the short periods for which it was tractable to run simulations, while maintaining a large difference between these probabilities (in fact we used a factor of 180 representing the difference between 4 hours and 1 month).

The rest of our model is strongly based on [4]. We use integrate and fire neurons, where the membrane potential V_{mem} is described by:

$$\tau_{mem}\frac{\delta V_{mem}}{\delta t} = V_{rest} - V_{mem} + g_{ex}(t)(E_{ex} - V_{mem}) \tag{2}$$

where E_{ex} is the excitatory reversal potential, V_{rest} is the resting potential and τ_{mem} is the membrane time constant. Upon reaching a threshold V_{thr}, a spike occurs and V_{mem} is reset to V_{rest}. A presynaptic spike at time 0 causes a synaptic conductance $g_{ex}(t) = ge^{\frac{-t}{\tau_{ex}}}$ (where τ_{ex} is the synaptic time constant); this is cumulative for all presynaptic spikes. Spike-Timing-Dependent Plasticity (STDP) is known to occur in biology at least in vitro, and has been used recently to explain map reorganisation in vivo [15]. STDP is implemented such that a presynaptic spike at time t_{pre} and a post-synaptic spike at time t_{post} modify the corresponding synaptic conductance by $g \rightarrow g + g_{max}F(\Delta t)$, where $\Delta t = t_{pre} - t_{post}$ and:

$$F(\Delta t) = \left\{ \begin{array}{l} A_+.e^{(\frac{\Delta t}{\tau_+})}, \quad if \Delta t < 0 \\ -A_-.e^{(\frac{-\Delta t}{\tau_-})}, if \Delta t \geq 0 \end{array} \right\} \tag{3}$$

where $A_{+/-}$ are magnitudes and $\tau_{+/-}$ are time constants for potentiation and depression respectively. This is cumulative for all pre- and post-synaptic spike pairs. g is bounded in the range $0 \leq g \leq g_{max}$.

Parameters were set starting from parameters given in [4]. A+ was increased 20-fold as a concession to limited computational resources for simulations (this should not qualitatively change the model since many plasticity events are still needed to potentiate a depressed synapse). Then key parameters were changed; namely g_{max} (the peak synaptic conductivity), $\tau-/\tau+$ (the ratio of time constants for depression and potentiation) and B (the ratio of potentiation to depression, i.e. A_+/A_-) were changed to maintain key conditions, being: the total weight should be approximately 50% of the maximum possible; the average network neuron firing rate should approximately match the average input firing rate; and the total weight of lateral synapses should roughly match the weight of feed-forward ones. In the interests of simplicity we did not allow for different values of B for different projections feedforward vs recurrent). An unjustified simplification is that new synapses start strong and then get weakened; the opposite case seems more likely. We have used this for simplicity because it avoids the need for any homeostatic mechanisms to kick-start the network.

Each input cell was an independent Poisson process. A stimulus location was chosen and mean firing rates were given a Gaussian distribution around that location based on a peak rate f_{peak} and variance ${\sigma_{stim}}^2$ which was added to a base rate f_{base}. The stimulus location changed regularly every 0.02s. This regularity is a move away from biologically realistic inputs (c.f. [4]); this was a necessary concession to provide stronger correlation cues given the smaller number of synapses per neuron. A further concession was the more extreme values of f_{base} and f_{peak}. σ_{stim} was chosen to be between the values of $\sigma_{form-feedforward}$ and $\sigma_{form-lateral}$ and f_{peak} was set so as to keep the overall mean firing rate at a mean value f_{mean} which gave sufficient difference between f_{base} and f_{peak}.

3 Results

Simulations were run with a C++ function, with initial conditions created and data analysis carried out with Matlab. Simulations used a time step of 0.1ms. Parameters are given in table 1. The mean frequency of rewiring opportunities per potential synapse was 1.22Hz (depressed synapses were therefore eliminated after an average of 33s). Initial placement of synapses was performed by iteratively generating a random pre-synaptic partner and carrying out the test for formation described in section 2. Initially feed-forward and lateral connections were placed separately, each up to their initial number of 16 synapses. Weights were initially maximised. Runs were for 5 minutes of simulated time.

For calculating the preferred location for each target cell, the use of the "centre of mass" measure as in [6] would be erroneous because the space is toroidal and therefore the calculation of preferred location would be skewed by the choice of reference point from which synapses' coordinates are measured. In [6] the reference point for calculating centre of mass of the dendritic synapses of

Table 1. Simulation parameters

for STDP	for rewiring	for inputs
$g_{max} = 0.2$	$\sigma_{form-feedforward} = 2.5$	$f_{mean} = 20Hz$
$\tau_m = 0.02s$	$\sigma_{form-lateral} = 1$	$f_{base} = 5Hz$
$\tau_+ = 0.02s$	$p_{form-lateral} = 1$	$f_{peak} = 152.8Hz$
$\tau_- = 0.064s$	$p_{form-feedforward} = 0.16$	$\sigma_{stim} = 2$
$A_+ = 0.1s$	$p_{elim-dep} = 0.0245$(= 0.5∗mean formation rate)	
$B = 1.2$	$p_{elim-pot} = p_{elim-dep}/180 = 1.36 * 10 - 4$	

a target cell was chosen as the predefined ideal location, therefore the measures of distance of preferred location were skewed towards the ideal locations dictated by the model. We avoided this by the novel method of searching for the location around which the afferent synapses have the lowest "weighted variance" (${\sigma_{aff}}^2$), i.e.:

$$\sigma_{aff}^2 = \underset{\boldsymbol{x}}{\operatorname{argmin}} \frac{\sum_i w_i . |\boldsymbol{p}_{xi}|^2}{\sum_i w_i} \tag{4}$$

where i is a sum over synapses, $\boldsymbol{x}$ is a candidate preferred location, $|\boldsymbol{p}_{xi}|$ is the minimum distance from that location of the afferent for synapse i and wi is the weight of the synapse (if connectivity is evaluated without reference to weights, synapses have unitary weight). We implemented this with an iterative search over each whole number location in each dimension and then a further iteration to locate the preferred location to 1/10th of a unit of distance (the unit is the distance between two adjacent neurons). Note that in the non-toroidal case this measure is equivalent to the centre of mass, as used in [3].

Having calculated the preferred location for all the neurons in the network layer we took the mean of the distance of this preferred location from the ideal location to give an Average Absolute Deviation (AAD) for the projection. By reporting both AAD and mean σ_{aff} for a projection we have a basis for separating its variance from the deviation of its preferred location from its ideal location. However AAD and mean σ_{aff} are both dependent on the numbers and strengths of synapses and these can change during development. Therefore to observe the effect of the activity-dependent development mechanism irrespective of changes in synapse number and strength we made comparison in two ways. Firstly, for evaluating change in mapping quality based only on changes in connectivity without considering the weights of synapses we created a new map taking the final number of synapses for each network neuron and randomly placing them in the same way as the initial synapses were placed. We then calculated σ_{aff} and AD for each neuron in each of the maps and compared the averages of these (i.e. mean σ_{aff} and AAD), applying significance tests between the values of two populations of neurons, i.e. all the neurons on the final map vs all those on the reconstructed map. Having established what effect there was

on connectivity we considered the additional contribution of weight changes by creating a new map with the same topology, taking the final weights of synapses for each network neuron and randomly reassigning these weights amongst the existing synapses for that neuron. We then compared the two maps as described above.

Three main experiments were carried out: Case 1 had both rewiring and input correlations, as described in section 2; case 2 had input correlations but no rewiring; case 3 had rewiring but no input correlations (i.e. all input neurons fired at fmean). The results are given in table 2. For comparisons, mean σ_{aff}

Table 2. Summary of simulation results: Case 1: Rewiring and input correlations; Case 2: Input correlations and no rewiring; Case 3: Rewiring and no input correlations

Case	**1**	**2**	**3**
Network neuron mean spike rate	24.7	17.4	10.5
Final mean no. feed-forward incoming synapses per network neuron	14.1	NA	12.5
Weight as proportion of max for the initial no. of synapses	0.60	0.36	0.33
Mean $\sigma_{aff-init}$	2.36	2.36	2.36
Mean $\sigma_{aff-final-con-shuffled}$	2.32	NA	2.32
Mean $\sigma_{aff-final-con}$	1.95	2.36	2.17
Mean $\sigma_{aff-final-weight-shuffled}$	1.88	2.10	1.99
Mean $\sigma_{aff-final-weight}$	1.70	1.98	1.95
AAD_{init}	0.78	0.78	0.78
$AAD_{final-con-shuffled}$	0.89	NA	0.90
$AAD_{final-con}$	0.83	0.78	0.93
$AAD_{final-weight-shuffled}$	0.92	1.36	1.21
$AAD_{final-weight}$	0.95	1.58	1.34

and AAD were each calculated for the feed-forward connections of the following networks: (a) The initial state with weights not considered (recall that all weights were initially maximised) these results are suffixed "*ini*", i.e. AAD_{ini}; (b) the final ("fin") network with weights not considered but only connectivity ("*con*") with all synapses weighted equally, i.e. $AAD_{fin-con}$; (c) for comparison with $AAD_{fin-con}$, the final number of synapses for each network neuron, randomly placed ("*shuf*") in the same way as the initial synapses (not applicable for simulations with no rewiring), i.e. $AAD_{fin-con-shuf}$; (d) the final network including weights, i.e. $AAD_{fin-weight}$; (e) for comparison with $AAD_{fin-weight}$, the final connectivity for each network neuron with the actual weights of the final synapses for each network neuron randomly reassigned amongst the existing synapses, i.e. $AAD_{fin-weight-shuf}$. Results were compared using Wilcoxon Signed-Rank (WSR) tests on AD and σ_{aff} for incoming connections for each network neuron over the whole network layer for a single simulation of each of the two conditions under consideration.

4 Discussion

We observe the effect of rewiring by comparing case 1 (with rewiring) and case 2 (without rewiring). Considering topology change, in case 1 mean $\sigma_{aff-fin-con}$ drops to 1.95, c.f. 2.32 for mean $\sigma_{aff-fin-con-shuf}$; this drop is significant (WSR, p=2.4e-25). In case 2 mean $\sigma_{aff-fin-con}$ is constrained to remain at mean $\sigma_{aff-ini} = 2.36$. Considering weight change, in case 1 mean $\sigma_{aff-fin-weight}$ drops to 1.70, c.f. 1.88 for mean $\sigma_{aff-fin-weight-shuf}$. In case 2, mean $\sigma_{aff-fin-weigh}$ drops to 1.98, c.f. 2.10 for mean $\sigma_{aff-fin-weight-shuf}$. Both drops are significant (WSR, p=2.7e-27 and 8.7e-6 respectively).

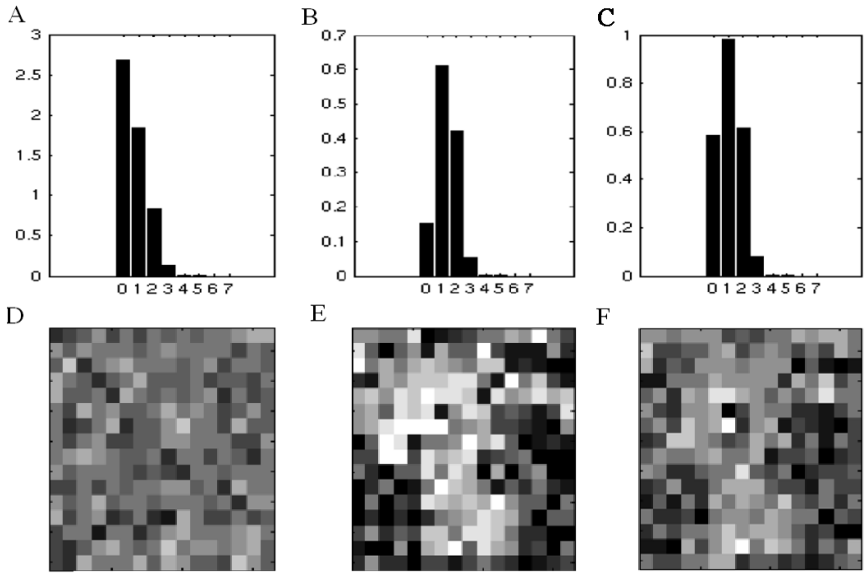


Fig. 1. A-C: Normalised weight density of incoming lateral synapses (weight/unit area; y-axis) radially sampled and interpolated at given distances of pre-synaptic neuron from post-synaptic neuron (x-axis), averaged across population. D-F: ocular preference, i.e. preference for cells from the two intra-correlated input spaces interspersed in the input space, for each network cell on a scale from white to black. A,D: initial. B,E: final, considering synaptic weights. C,F: final, all synapses with unitary weight.

Mean $\sigma_{aff-fin-weight}$ appears to be lower in case 1 than case 2. We cannot say for sure that this superior reduction of variance is due to the effect of the rewiring mechanism because the different numbers of final synapses in each case make a comparison impossible, however there is a good reason to believe that this is so: the drop in mean $\sigma_{aff-fin-con}$. This drop on its own indicates that the rewiring mechanism has helped to reduce variance and would also lay the groundwork for different final measures of σ_{aff} when weights are considered.

We can also see qualitatively that the effect of rewiring is to embed in the connectivity of the network input preferences which arise through the weight

changes mediated by the learning rule. STDP favours causal inputs with the lowest latency and local excitatory lateral connections tend to lose the competition with excitatory feed-forward connections as they have a higher latency [4]. The extreme of this effect can be seen in synapses from a network neuron back to itself (recurrent synapses). The placement rule allows these synapses to form, however these synapses only ever receive a pre-synaptic spike immediately following a post-synaptic spike and therefore they are always depressed by the learning rule. Figure 1A shows the initial density of incoming lateral synapses from pre-synaptic partners at given distances out from the post-synaptic neuron. It can be seen that the average neuron receives more synapses from itself (those at x-position 0) than from any of its closest neighbours. Figure 1B shows the final distribution where synapses are weighted. The recurrent synapses have been depressed much more than their neighbours. Figure 1C shows the final distribution only considering numbers of synapses and not their weights. The proportion of recurrent synapses to lateral synapses with neighbours has reduced from the initial state, due to the preferential elimination of the weak recurrent synapses.

As a further demonstration of the effect of rewiring a simulation was carried out with the input neurons divided into two groups, mimicking the effect of binocular inputs. The groups were interspersed in a regular diagonal pattern, i.e. each input neuron is in the opposite group to its 4 adjacent neurons; the stimulus location switched between the two groups every time it changed. To keep the overall input rate the same the peak firing rate was doubled. Figure 1D shows the initial preference of each network neuron for input neurons in the two groups. Figure 1E shows the final ocular dominance map where synapses are weighted. Although the space used was too small and the result of the learning rule with a small number of synapses too random for familiar striped ocular dominance patterns to emerge (c.f. [3]) ocular dominance zones can be seen. This pattern is reflected in the final map of connectivity in Figure 1F, where synaptic weights are not considered; another example of weight patterns caused by input activity becoming embedded in connectivity patterns.

Considering the effect of the algorithm on AAD, in case 2 $AAD_{fin-weight}$ is significantly increased c.f. $AAD_{fin-weight-shuf}$ (WSR, p=0.0012). In case 1 the corresponding change is not significant (WSR, p=0.48). In case 1 the drop in $AAD_{fin-con}$ c.f. $AAD_{fin-con-shuf}$ is not significant (WSR, p=0.31).

The basic action of weight-independent STDP on a set of incoming synapses for a single neuron is to deliver a bimodal weight distribution [4]. Where there are input correlations these cause the more correlated inputs to be maximised and the less- or un-correlated inputs to be minimised. The effect of both the input correlations and the local excitatory lateral synapses on each individual incoming connection field then should be to cause a patch of neighbouring synapses to become potentiated and for outliers from this patch to be depressed. The location of the patch will be random; it is likely to form near the ideal location because there should be a denser concentration of synapses there, however the centre of the patch is unlikely to fall exactly on the ideal location but rather a certain mean distance from it. This introduces a shift of preferred location

from the ideal location. Rewiring cannot be expected to eliminate this error but it might be expected to allow the patch to move towards the centre as σ_{aff} reduces due to the preferential placement of synapses towards the centre. However in our simulations AAD did not improve. The slight drop in $AAD_{fin-con}$ c.f. $AAD_{fin-con-shuf}$ is not significant but in any case a drop in AAD could only be a result of the reduction in mean σ_{aff} because $AAD_{fin-weight}$ does not decrease, rather it stays the same (as in case 1) or increases (as in case 2). That is to say, the result of the weight changes is not to drive the preferred location towards the ideal. Rather, the improvement of topography is driven by the continued placement of synapses towards the ideal location; the activity-dependent mechanism simply facilitates by allowing the incoming connection field to be narrowed by the preferential elimination of outliers.

Considering the role of input correlations, in case 3 (rewiring but no input correlations) mean $\sigma_{aff-fin-con} = 2.17$, vs 2.31 for mean $\sigma_{aff-fin-con-shuf}$; this is significant (WSR, p=5.0e-6). Mean $\sigma_{aff-fin-weight} = 1.95$ vs 1.99 for mean $\sigma_{aff-fin-weight-shuf}$; this is significant (WSR, p=0.028).

The slight drop in mean $\sigma_{aff-fin-weight}$ is a sufficient cue to drive the narrowing of the incoming connection fields, as evidenced by the drop in mean $\sigma_{aff-fin-con}$. It was shown [8] that functional architecture could form in the absence of any input except uncorrelated random noise. We show that this applies to topographic map refinement as well, although our explanation differs: A spike from a single input neuron will excite a given network neuron and any other of its neighbours which have a synapse from that input. Thus the neuron will also tend to receive some excitation from lateral connections because of that spike. Network neurons sample afferent neurons more densely around their ideal locations so they are more likely to share an afferent with a neighbour if that afferent is close to their ideal location. Thus synapses from afferents closer to the ideal location are more likely to be potentiated. Therefore the gradient of connection density set up by activity-independent placement acts as a cue which allows the preferential elimination of outliers, giving a reduction in variance.

5 Conclusions

We have presented a model of topographic development including both weight and wiring plasticity, which follows the reasonable assumptions that synapses preferentially form in locations to which their axons are guided and that weaker synapses are more likely to be eliminated. We have shown that spatially correlated inputs help to create patterns of synaptic weights which favour narrower projections, but the spatial correlations are not necessary for some reduction of variance to occur (extending a result from [8]). A weight-change mechanism and a rewiring mechanism can work together to achieve a greater effect than the weight changes alone, with the rewiring mechanism acting to embed patterns of synaptic strengths in the network topology; this is as one would expect, though it has not been demonstrated quantitatively before, to our knowledge. The accuracy of preferred locations for network neurons however may not necessarily improve

when synapses are formed based on distributions with on-average perfect topography to start with. The novel division of mapping quality into the quantities of mean σ_{aff} and AAD is therefore a useful means for investigating these effects, and we have demonstrated a method of applying statistical significance tests to extract highly significant effects from small-scale simulations. Future work will include the introduction of weight-dependent STDP and of noise in the measures of proximity and weight used for formation and elimination.

Acknowledgements

We are grateful to Guy Billings for providing the basis of the simulator code. This work was funded by EPSRC.

References

1. Willshaw, D., von der Malsburg, C.: How patterned neural connections can be set up by self-organisation. Proc. R. Soc. Lond. B. **194** (1976) 431–445
2. Miller, K., Keller, J., Stryker, M.: Ocular dominance column development: analysis and simulation. Science **245** (1989) 605 615
3. Goodhill, G.: Topography and ocular dominance: a model exploring positive correlations. Biological Cybernetics **69** (1993) 109–118
4. Song, S., Abbott, L.: Cortical development and remapping through spike timing-dependent plasticity. Neuron **32** (Oct 2001) 339–350
5. Willshaw, D.: Analysis of mouse Epha knockins and knockouts suggests that retinal axons reprogramme target cells. Development **133** (2006) 2705–2717
6. Elliott, T., Shadbolt, N.: A neurotrophic model of the development of the retinogeniculocortical pathway induced by spontaneous retinal waves. Journal of Neuroscience **19** (1999) 7951–7970
7. Miller, K.: Equivalence of a sprouting-and-retraction model and correlation-based plasticity models of neural development. Neural Computation **10** (1998) 529–547
8. Miikkulainen, R., Bednar, J., Choe, Y., Sirosh, J.: Computational Maps in the Visual Cortex. Springer, New York (2005)
9. Willshaw, D., von der Malsburg, C.: A marker induction mechanism for the establishment of ordered neural mappings. Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences **287** (1979) 203–243
10. Bamford, S., Murray, A., Willshaw, D.: Large developing axonal arbors using a distributed and locally-reprogrammable address-event receiver. International Joint Conference on Neural Networks (IJCNN) (2008)
11. Sperry, R.: Chemoaffinity in the orderly growth of nerve fiber patterns and connections. Proc. Natl. Acad. Sci. USA **50** (1963) 703–709
12. Ruthazer, E., Cline, H.: Insights into activity-dependent map formation from the retinotectal system. Journal of Neurobiology **59** (2004) 134–146
13. Swindale, N.: The development of topography in the visual cortex: a review of models. Network: Computation in Neural Systems **7** (1996) 161–247
14. Grutzendler, J., Kasthuri, N., Gan, W.: Long-term dendritic spine stability in the adult cortex. Nature **420** (2002) 812–816
15. Young, J., Waleszczyk, W., Wang, C., Calford, M., Dreher, B., Obermayer, K.: Cortical reorganisation consistent with spike timing- but not correlation-dependent plasticity. Nature Neuroscience **10** (July 2007) 887–895

Large Developing Axonal Arbors Using a Distributed and Locally-Reprogrammable Address-Event Receiver

Simeon A. Bamford, Alan F. Murray, David J. Willshaw

***Abstract*— We have designed a distributed and locally reprogrammable address event receiver. Incoming address-events are monitored simultaneously by all synapses, allowing for arbitrarily large axonal fan-out without reducing channel capacity. Synapses can change input address, allowing neurons to implement a biologically realistic learning rule locally, with both synapse formation and elimination.**

I. Introduction

Neuromorphic engineers create integrated electronic circuits which mimic neural computation in biological nervous systems, both to inform computational neuroscience and in pursuit of superior engineering solutions for classes of problems where biology currently outperforms artificial devices [1]. There is a need to form interconnects between many integrated neuron circuits to create neural networks. In many applications such as topographic map development [2], reconfigurability in the connections is essential to underpin map formation and maintenance. In a topographic map, one (typically 2D and sensor-driven) layer of neurons maps its connections to another layer such that neighbouring relationships between neurons in one layer are preserved in the other. In order for such a mapping to develop, neurons gradually change their patterns of connections according to both innate preferences and feedback induced by network input [3].

Time-division multiplexing facilitates massively-parallel connection between spiking neuron circuits across multiple chips. Specifically, spikes are treated as address-events; the unique address of a neuron within a neural array is transmitted on an address bus. This approach was first used in the theses of Sivilotti and Mahowald [4] and [5] and has since been extended and improved. Boahen [6] gives a good summary of this still-evolving technique. Within this Address-Event Representation (AER) protocol, the number of wires required to connect N neurons scales as $log(N)$, such that the number of pins and wires necessary to interconnect chips is achievable. The development of word-serial AER reduces the number of wires required still further [7]. AER exploits the large difference in frequency between the spiking behaviour of biological neurons (on the order of 10-1000Hz) and the capability of digital electronic communication (many MHz). Approximately 100,000 neurons can share a single bus [6] if biological spike rates are desired.

This work has been funded by EPSRC. Authors details: Simeon Bamford, Neuroinformatics Doctoral Training Centre, University of Edinburgh. Alan Murray, Institute of Integrated Micro and Nano Systems, University of Edinburgh. David Willshaw, Institute of Adaptive and Neural Computation, University of Edinburgh. To whom further communication should be addressed: sim.bamford@ed.ac.uk

AER was originally conceived as a point-to-point protocol. If each neuron in one neural layer has a unique connection to only one neuron in a corresponding neural layer in a topographic map arrangement, the outgoing bus can be decoded directly by a row-and-column decoder on a receiving chip, and spikes are delivered correctly to the same location on a corresponding chip (as in [4] [5]). Simplistically this type of one-to-one connectivity can be observed in some places in the nervous system, for example the connections from cone receptors to bipolar cells, at least in the fovea ([8] ch. 26). More commonly however neurons make connections to many other neurons (i.e. they have a large "fan-out") and receive large numbers of incoming connections ("fan-in"). As two examples, Xiong *et al* [9] found an average fan-out of 167 for retinal ganglion cells in the tectum of the hamster, whilst Palkovits *et al* [10] found an average fan-in of 85,000 onto the Purkinje cells of the cat. In order to implement arbitrary many-to-many network connectivity, address-events are commonly received not directly by a neural array chip but rather by a microcontroller and are then compared to a look-up table in memory in order to find out which outgoing address-events should be sent (e.g. [11]). These are then sent sequentially to one or more receiving neural arrays. This approach reduces the capacity of the bus in the presence of large fan-out. If an average fan-out of 1000 is desired for example, a bus can only support about 100 neurons.

The use of a microcontroller and a look-up table in memory has also been used to implement synaptic rewiring, where the connectivity between neurons changes with time according to a biologically inspired learning rule [12]. In the scheme of Taba and Boahen [13], information from the receiving synapse is transmitted off-chip back to the microcontroller where it is used to modify the look-up table. This is part of a trend of using the microcontroller to implement more of the neural network model. This trend has been extended by Vogelstein *et al* [14] where other synaptic variables (number of release sites, probability of release and quantal post-synaptic response — the product of these is essentially the synaptic weight) are also held in the look-up table, allowing each neuron to have a single "general purpose" synapse circuit which acts as a number of virtual synapses.

II. Proposed system

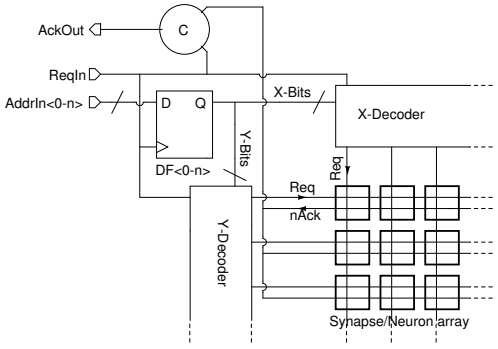
In order to overcome the bottleneck on channel-capacity as fan-out increases, we have taken an alternative approach in which more information is stored in synapse circuits within the neural array. Details of incoming connectivity are stored,

along with synaptic variables such as an analogue voltage representing synaptic weight. Address events from a sending chip are directly received by a receiving chip and broadcast across the receiving chip's neural array. Simultaneously, all synapses compare that address to a locally-stored address to establish whether the address-event was intended for it. Many synapses can store the same desired address and thus arbitrarily large axonal arbors can be implemented without reducing bus capacity. Synapses do not acknowledge receipt of an event, rather the chip-wide broadcast is timed to last long enough for all synapses to receive it. We compare our approach to the "look-up table" approach in which source neuron addresses are mapped to target synapse addresses using a look-up table, an example of which is Mitra *et al* [15]. The look up table approach allows the use of receiving circuitry as described by Boahen [6], which is shown in fig 1a. The receiving circuitry which implements our system is shown in fig 1b. In our system, to ensure that communication succeeds, each communication cycle is deliberately slower than the average cycle speed which could be achieved if the sender were allowed to proceed with the next event as soon as a synapse acknowledges, as in Fig 1a. However as average fan-out increases our solution outperforms any system which implements fan-out serially.

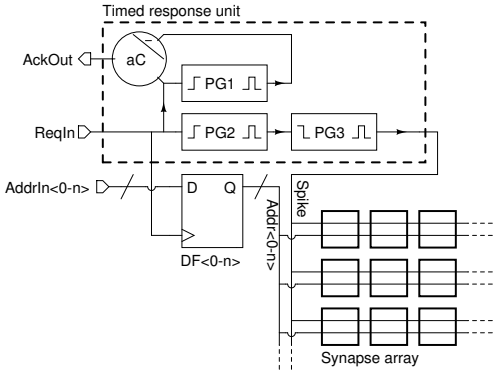
III. Scalability of proposed system

A. Area

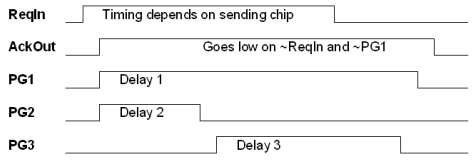
Each synapse, in order to implement its address bus monitor, must store as many bits in memory elements as the width of the incoming address bus. The total area of the monitoring circuitry across the chip (or across the system, for a multi-chip system) then scales as $S_{max}Nlog_2(N)$, where N is the number of neurons in the system and S_{max} is the maximum fan-in, i.e. number of dendritic (or incoming) synapses allowed per neuron. The $S_{max}N$ term represents the number of synapse circuits in the system and the $log_2(N)$ term represents the number of bits necessary to encode a neuron's address within each synapse. At first glance this scales poorly compared to the look-up table approach, which employs row and column decoders allowing the area of the receiving circuitry to scale as $\sqrt{S_{max}N}log_2(S_{max}N)$, where the $\sqrt{S_{max}N}$ term represents the number of row or column decoder elements necessary to decode a target synaptic address and the $log_2(S_{max}N)$ term represents the number of bits necessary to encode a synaptic address (each decoder element must store one dimension (i.e. half the bits) of the synaptic addresses it encodes for). Importantly however the look-up table approach requires that an external memory chip is used, in which area is required which scales as $S_{av}Nlog_2(S_{max}N)$, where S_{av} is average fan-out. The $S_{av}N$ term is the number of axonal (or outgoing) synapses in the system and the $log_2(S_{max}N)$ term is the number of bits necessary to encode a dendritic (or incoming) synaptic address. The costs of microcontrollers and RAM are not normally considered, whether in terms of chip area or power consumption. This is acceptable for test systems,



(a) AER receiver circuitry, functionally equivalent to that described in [6]. The incoming request "ReqIn" triggers the raising of the global acknowledge "AckOut" and the decoding of the incoming address; a synapse (or neuron) is targeted; when this acknowledges, AckOut is lowered (once ReqIn has also been lowered), allowing the next event to be transmitted.



(b) Our AER receiver. Upon ReqIn going high, AckOut is immediately driven high and also a pulse generator (PG1) is triggered, the output of which stays high for a precisely-timed (adjustable) period thereafter. AckOut stays high until ReqIn and PG1 both drop. ReqIn also triggers the local latching of the incoming address bus. Once latched the address is broadcast across the chip and all synaptic address-monitors simultaneously compare this address to their own stored address to decide whether it is correct. From the rising of ReqIn there is a short delay (implemented by PG2) to allow this to happen before a pulse ("Spike") is sent out across the chip (implemented by PG3) triggering those synapses with correct addresses to accept the event. The pulse generated by PG1 is timed to be long enough to accommodate the joint delays of PG2 and PG3 before allowing AckOut to drop and the cycle to repeat.



(c) Example timing diagram for our response unit.

Fig. 1.

TABLE I
SCALING OF AREA, ENERGY USAGE AND SPEED

System	On-chip receiver area	Off-chip memory space	Internal buffering energy per spike sent	Speed per spike sent
Ours	$S_{max}Nlog_2(N)$	none required	$S_{max}N$	unity
Look-up table	$C\sqrt{S_{max}N_{chip}}log_2(S_{max}N_{chip})$	$S_{av}Nlog_2(S_{max}N)$	$S_{av}C\sqrt{S_{max}N_{chip}}$	S_{av}
Vogelstein [14]	$C\sqrt{N_{chip}}log_2(N_{chip})$	$S_{av}Nlog_2(N)$	$S_{av}C\sqrt{N_{chip}}$	S_{av}

N_{chip} = number of neurons per chip; C = number of chips in system; N = number of neurons in system = $N_{chip}C$; S_{max} = maximum fan-in, i.e. number of dendritic synapses allowed per neuron; S_{av} = average fan-out.

however if total power budget and space are considered (for a hypothetical implantable system, for example) it can be seen that in our approach the chip space necessary to implement memory is simply being distributed throughout the neural array, rather than stored in a separate dedicated chip.

It's also worth noting that the scaling expression above for the look-up table approach only holds for a single-chip system. If the system is spread across multiple chips then the expression for the look-up table approach becomes $C\sqrt{S_{max}N_{chip}}log_2(S_{max}N_{chip})$ where C is the number of chips in the system and N_{chip} is the number of neurons per chip. Therefore as a neural network is scaled up by networking together more chips and the ratio of C/N_{chip} goes up, the on-chip area scaling advantage with respect to our system due to row and column decoding is eroded. The scaling expressions above are summarised in table I. Vogelstein's approach [14] is also included for comparison; this is included because it is a special case of the look-up table approach in which there is only one target synapse address per target neuron.

Note that we do not wish to overlook the actual difference in area requirements between these approaches. Memory on a dedicated RAM chip takes up much less space than in our design, partly because it is not integrated with decoder circuitry but rather optimised for its purpose and partly because it does not need to be implemented in a process suitable for mixed signals and can therefore benefit from smaller feature sizes. Beyond this it is also less costly simply because it is mass-produced. Our approach yields synapses of significantly larger on-chip area resulting in higher production costs for the foreseeable future. If however this increase of area can be tolerated for a given technology, then it can be tolerated equally both as miniaturisation proceeds and as the size of neural network implemented expands. Meanwhile we can expect our approach to continue to support larger neural networks with large average fan-outs long after the existing approaches run out of "bandwidth". Whilst chip area is much more expensive on trial ASICs than on mass-produced memory, this may not always be the case if neuromorphic circuitry comes into mainstream demand.

B. Energy usage

In our approach, each incoming address event must be broadcast across the neural array to each synapse. Consequently each synapse contributes a capacitive load to the on-chip buffering and therefore energy consumption will scale linearly with $S_{max}N$. This figure includes internal buffering to the neurons but does not include buffering from the pads to the peripheral latches shown in figure 1, because the look-up table approach has an equivalent cost. The chips we are fabricating each contain 2048 synapses, and based on the analysis in section VI will therefore use 1.76nJ per incoming spike for internal buffering. We expect this to be comparable to the energy necessary to transfer a spike externally between chips, though as die sizes increase we expect the energy cost of internal buffering to become increasingly dominant. In the look up table approach there is no need to broadcast the address across the chip and the spike signal can be targetted to the row and column of the correct synapse within the neural array. The energy cost of internal buffering should therefore be lower and should scale *per incoming spike* as $C\sqrt{S_{max}N_{chip}}$. In our system however, energy usage remains constant per address-event sent, whilst in the look-up table approach energy usage *per spike sent* increases linearly with axonal fan-out, as each axonal synapse requires a separate spike to be transmitted between chips and the correct synapse targetted. Bearing this in mind, scaling expressions for energy are given in Table I. This suggests that if the choice of which approach to use is to be determined by energy usage then there will be a ratio (ignoring the complexity introduced by multi-chip systems) of S_{av} / $S_{max}N$ above which our system can be expected to outperform the look-up table approach. Here we have not considered the additional energy costs of the microcontroller and RAM in the look-up table approach; how this scales depends on the implementation.

C. Speed

As noted above, in our system each communication cycle is deliberately slower than the average cycle speed which can theoretically be achieved in the look-up table approach, though we expect this difference to be no more than a small factor. However the time taken in our approach does not increase with S_{av} whereas in the look-up table approach it increases linearly (as shown in table I). Additionally even in the case where fan-out = 1, the look-up table approach introduces a small latency due to the need for the microcontroller to receive, process and send a spike, though this latency is normally considered insignificant for neural systems running on a biological time scale.

IV. Local synaptic rewiring

As the details of incoming connectivity are stored locally to the synapse, neurons can take advantage of other information stored locally at the soma and in the synapses in order to change incoming connectivity. Specifically, by also storing a binary variable at each synapse indicating whether or not the synapse exists, we use the synaptic weight (an analogue voltage stored on a capacitor) to inform the decision to disconnect. This follows Miller [16], who gives evidence that the decision whether newly sprouted synapses are stabilised or retracted is guided by changes in physiological strengths. The synapse circuit therefore becomes a circuit representing a potential synapse, part of the neuron's total synaptic capacity (a concept explored in [17]). We supplement this with a chip-wide mechanism for implementing synaptic connection, where the probability of a synapse forming with a given pre-synaptic neuron is influenced by the distance between that neuron and the post-synaptic neuron, allowing receptive fields to form according to 2D probabilistic distributions, as if the axons were guided according to some version of the chemoaffinity hypothesis [18]. The details of the neural learning algorithm we use are being published separately however a brief summary is given here:

This generalised model of map formation could equally apply to retino-tectal, geniculo-cortical, or other projections. Implemented cells are considered to be in a 2D "layer" of neurons. There are two excitatory projections to this layer, one from a simulated source layer and one a lateral projection from the cells themselves. Each location in one layer has a corresponding ideal location in the other, such that one layer maps smoothly and completely to the other; for simplicity there is no transformation from source location to ideal location; the address spaces are identical, though our implementation would allow for transformations to be inserted. Each cell in the network layer can receive a maximum number of afferent synapses (64 in our implementation). The projections compete for the synaptic capacity of the network neurons. We assume that an unspecified activity-independent process is capable of guiding the formation of new synapses so that they have a distribution around their ideal locations which is monotonically decreasing with distance. To implement this, where a network cell has less than its maximum number of synapses, the remainder are considered potential synapses. At a fixed rewiring rate a synapse is randomly chosen. If it is a potential synapse a possible pre-synaptic cell is randomly selected and, if for example we choose a Gaussian function of distance, then synapse formation occurs when:

$$r < p_{form}.e^{-\frac{\delta^2}{2{\sigma_{form}}^2}} \tag{1}$$

where r is a random number uniformly distributed in the range $(0, 1)$, p_{form} is the peak formation probability, δ is the distance of the possible pre-synaptic cell from the ideal location of the post-synaptic cell and ${\sigma_{form}}^2$ is the variance of the connection field. In other words a synapse is formed when a uniform random number falls within the area defined by a Gaussian function of distance, scaled according to the peak probability of synapse formation, (which occurs at $\delta = 0$). Lateral connections are formed by the same means as feed-forward connections though our implementation allows different parameters for equation 1 for each projection, or indeed a different function for each projection. If the selected synapse already exists it is considered for elimination. The probability of elimination should be some monotonically decreasing function of weight and is implemented in a similar manner. Weights themselves vary according to a synaptic learning rule - we have chosen a form of spike-timing-dependent plasticity. For completeness, we briefly present ideal models of the neurons and synapses we have implemented. Based on [19], we use integrate and fire neurons, where the membrane potential V_{mem} is described by:

$$\tau_{mem}\frac{\delta V_{mem}}{\delta t} = V_{rest} - V_{mem} + g_{ex}(t)(E_{ex} - V_{mem}) \tag{2}$$

where E_{ex} is the excitatory reversal potential, V_{rest} is the resting potential and τ_{mem} is the passive membrane time constant. Upon reaching a threshold V_{thr}, a spike occurs and V_{mem} is reset to V_{rest}. To simplify implementation, we use a linear approximation to membrane excitation, which is justifiable when $E_{ex} \gg V_{thr}$. Other parameters are highly modifiable. A presynaptic spike at time 0 causes a synaptic conductance $g_{ex}(t) = ge^{\frac{-t}{\tau_{ex}}}$ (where τ_{ex} is the synaptic time constant); this is cumulative for all presynaptic spikes. Spike-timing-dependent plasticity is implemented such that a presynaptic spike at time t_{pre} and a post-synaptic spike at time t_{post} modify the corresponding synaptic conductance by $g \rightarrow g + g_{max}F(\Delta t)$, where $\Delta t = t_{pre} - t_{post}$ and:

$$F(\Delta t) = \left\{ \begin{array}{ll} A_+.e^{(\frac{\Delta t}{\tau_+})}, & if \Delta t < 0 \\ -A_-.e^{(\frac{-\Delta t}{\tau_-})}, & if \Delta t \geq 0 \end{array} \right\} \tag{3}$$

where $A_{+/-}$ are magnitudes and $\tau_{+/-}$ are time constants for potentiation and depression respectively. This is cumulative for all pre- and post-synaptic spike pairs. g is bounded in the range $0 \leq g \leq g_{max}$.

V. Proposed circuit

A. Address-event receiver circuitry

Our chip-level address-event receiver is compatible with standard address-event transmitters. An incoming request is acknowledged immediately and triggers local latching of the address bus and a timed delay followed by a timed pulse to synapses. A minimum cycle time is imposed. In our circuit this is about 20ns, which also allows for the effect of parasitic capacitances extracted from layout; this could be improved if the synapse design was optimised for speed. The circuitry which implements this is shown in fig 1b and a timing diagram is given in fig 1c.

B. Synaptic address monitor circuitry

The total area of the synapse scales as the number of bits necessary to encode a neurons address in the system. It is therefore necessary to make the storage of each bit and its associated circuitry as compact as possible. We have used a static memory element with a transmission-gate implementation of an XNOR gate for comparison with the incoming address bit. The result of the comparison contributes to a NAND gate for the whole monitor, the output of which ("nAeCorrect") indicates whether or not the incoming address is correct. Additional circuits allow for overwriting and read-out (though read-out may not be necessary in a final implementation). The synaptic address monitor circuitry is shown in fig 2, omitting read-out circuitry in the interests of clarity.

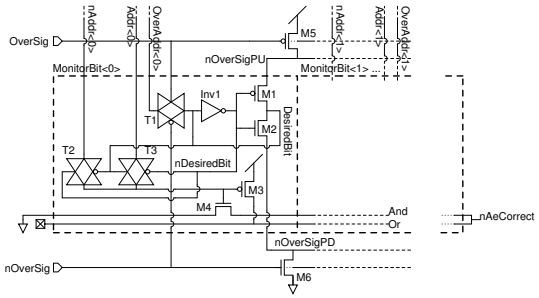


Fig. 2. Address monitor circuitry. The monitor is composed of a chain of bits; one bit is shown here (the zeroth bit). A bit of the address ("DesiredBit") is stored in a memory element composed of Inv1 and M1-2. An XNOR is continuously performed between DesiredBit and the incoming address bit ("Addr<0>") by means of T2-T3 (the incoming bit's complement "nAddr<0>" is also required). The result of the XNOR contributes to a NAND gate implemented throughout the monitor array by transistors M3-4. The result is nAeCorrect, indicating whether the full incoming address matches the full stored address. When OverSig goes high (and its complement nOverSig goes low), this is the signal for the monitor's address to be overwritten with the address on the "OverAddr" bus, a separate bus latching a recently received spike for consideration. OverSig chokes off transistors M1-2 using transistors M5-6 (these are common for all the monitor bits) while T1 opens, allowing DesiredBit to take the value of OverAddr<0>. Readout circuitry is not shown for clarity; this is an additional choked inverter with the same design as M1-2 & 5-6, opened onto a common outgoing bus during the "Compare" signal (see fig 3).

C. Synaptic rewiring circuitry

Synapses can be individually targeted for rewiring by an additional chip-wide mechanism, employing row and column decoders in the periphery. This allows both for the explicit setting and read-out of synaptic variables from an off-chip control mechanism for the purpose of testing the circuit, and for ongoing probabilistic rewiring, where synapses are randomly selected at a given rate as candidates for rewiring. The randomly chosen synapse addresses come from off-chip in our test implementation but could come from an on-chip random-number generator in a mature implementation.

When a synapse is selected as a candidate for rewiring its behaviour depends on its state of connectedness, stored in a static memory element. If it is connected then it is considered for disconnection. Its analogue weight value is compared to a voltage randomly chosen according to a probabilistic distribution. If the weight is below the random value then the synapse is disconnected. The random value is common for all the synapses on the chip but is only used at one synapse at a time and changes between each usage, avoiding the possibility of correlation between synapses. In our implementation the voltage is produced off-chip, but could be produced on chip by a random number generator and a DAC in a mature implementation. It is also possible to generate analogue noise for use in this way [20] which could then be profiled to match the probability of adaptation.

If the synapse is disconnected and it is selected as a candidate for rewiring then the possibility of it taking a new pre-synaptic partner is considered. The pre-synaptic partner considered is the last address to have arrived on the incoming bus. This is latched separately by the chip and also broadcast across the chip at the point that a rewiring consideration takes place. This allows a chip-wide calculation to take place providing a value, available at each neuron, of the geometric proximity of that neuron to the incoming address. The synapse under consideration then compares this proximity value to a random value, similar to the random value for disconnection but separate, created according to a probabilistic distribution for synapse formation. If the proximity value is higher than the random value then the synapse becomes connected and it adopts the incoming address in consideration as its new stored address. The circuitry which implements the connection and disconnection algorithm is shown in fig 3.

Regarding the proximity value, the incoming address may be from a neuron in the same neural layer, even a recurrent spike from the neuron itself, or it may be from a neuron in an afferent layer. We are considering a model in which there is a strong topographic mapping between successive neural layers, but this assumption is not essential to the system we describe. The effect of the proximity on the probability of rewiring can be eliminated altogether if it is not required, by reducing the probabilistic distribution to a binary choice between an extremely high value (where the synapse will not connect no matter how high the proximity) and an extremely low value (where the synapse will definitely connect regardless of proximity). The circuitry for creating the proximity value will be published separately. Briefly however it is an analogue current-mode circuit capable of operating across a neural layer composed of multiple chips and capable of delivering proximity values based on either toroidal (wrap-around) or non-toroidal spaces. It creates voltage gradients along both dimensions of the area upwards from the ideal location of the pre-synaptic partner and then allows a Euclidean distance to be created at a chosen node based on circuitry fundamentally similar to that described in [21].

Whilst it is possible to impose an arbitrary network topology by external programming, it is also possible to allow a probabilistic topology to form and, if desired, to continue to

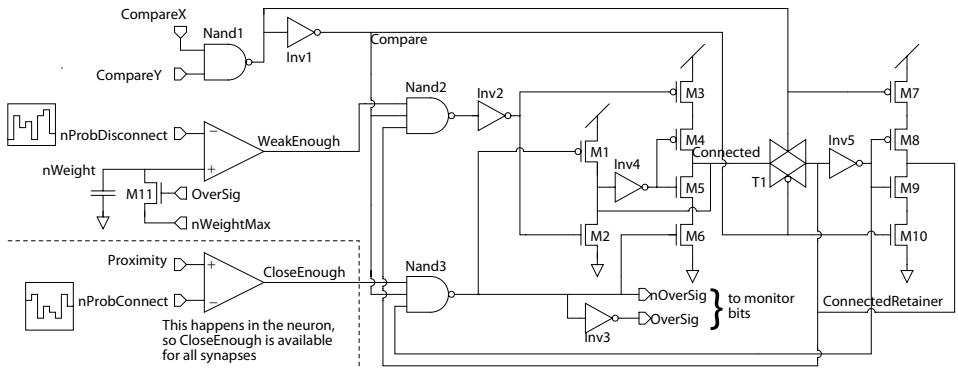


Fig. 3. Circuitry for synaptic rewiring. The synapse's "Connected" state is stored in a memory element composed of Inv4 and M4-5. This state can be overridden by a disconnection signal from Nand2 and Inv2, using M2-3, or by a connection signal from Nand3 using M1 and M6. "Compare" is driven high by the targeted conjunction of the CompareX and CompareY signal from row and column decoders, to indicate that rewiring is under consideration. While Compare is high, the Connected state is latched in a separate memory element "ConnectedRetainer" (Inv 5 and M8-9). This ensures that only connection or disconnection can occur, avoiding oscillations during the Compare signal. On connection, the override signal "Oversig" and its complement are sent to the address monitor, allowing the address under consideration to override the monitor's stored address; nWeight is also set to its strongest value (M11).

develop within the system according to biologically realistic principles, without any details of the topology being made available off-chip. In other words this system allows a black-box approach to network wiring at the level of individual synapses, allowing a system designer to concentrate on higher-level building blocks. Rewiring probabilities can be made arbitrarily low, even achieving biologically-realistic rates of synapse formation and elimination, i.e. hours, days or months between events [22].

VI. Simulation results and layout

A simulation demonstrating the ability of a neuron to rewire one of its synapses is shown in fig 4.

A high level neural network simulation implemented in C++/Matlab has shown the ability of a system with these capabilities and parameters to be capable of performing biologically realistic topographic map formation, even when mismatch ranges taken from Monte Carlo simulations of circuits are applied to the simulation (results not shown here).

The chip is being fabricated in AMS 0.35u 4-metal 2-poly process. The area of the synaptic address monitor bit is $11.1\mu m \times 15.95\mu m = 177\mu m^2$. We are creating a test system with 512 neurons (spread across multiple chips), therefore each synapse has a 9-bit receiver. This takes up 56% of the total synapse area, which is $11.1\mu m \times 255.95\mu m = 2841\mu m^2$. The remaining area is dedicated to: storing the additional synaptic variables; implementing the connection and disconnection circuitry; creating an increase in the neuron's level of synaptic current when a spike arrives; and implementing the synaptic weight change algorithm (spike-timing-dependent plasticity). Each neuron has 64 potential synapses, and the synaptic array takes up 98.6% of the total area of the neuron ($740.275\mu m \times 255.95\mu m = 0.189mm^2$), where the remaining area is dedicated to the storage of the neuron's variables, its central (integrate and fire) functions and its sending circuitry (the neuron circuit is novel, using a switched capacitor approach; this will be described in a separate publication). The layout of the synaptic address-monitor bit is shown in fig 5, excluding upper metal signal and power rails for clarity.

The need to buffer the address out to all the synapses as well as the spike signal requires that a significant capacitive load is overcome. Buffers which achieve this within a reasonable timescale (<5ns) are placed in the periphery of the chip. The ratio of address buffer to synaptic monitor area is $\approx 0.5\%$ and we would expect this ratio to remain approximately constant as neural network size is scaled up within the same technology.

Regarding power consumption, based on simulation with extracted capacitances, the synapse consumes per spike: 227fJ for delivery of the spike signal; 69.6fJ per address bit (assuming that each incoming address bit makes a transition with 50% probability each spike); and 5fJ pumped through the NAND gate that determines whether the correct address has been received. Our 9-bit synapses therefore consume 859fJ each per spike. This figure includes internal buffering to the neurons but does not include buffering from the pads to the peripheral latches shown in figure 1 (this is not included because the look-up table approach would be expected to have an equivalent cost).

VII. Discussion

The address event receiver we have implemented redefines synapse circuits as potential synapses and, in a straightforward manner, shifts the burden of decoding and receiving spike events into them. Pursuing this design choice we have moved other functions into the synapse, namely the ability to implement a developmental model which involves synaptic rewiring. Thus the circuit we have created highlights an alternative pole on a spectrum of design choices regarding the amount of functionality implemented within synapses. Hybrid approaches are clearly possible and may

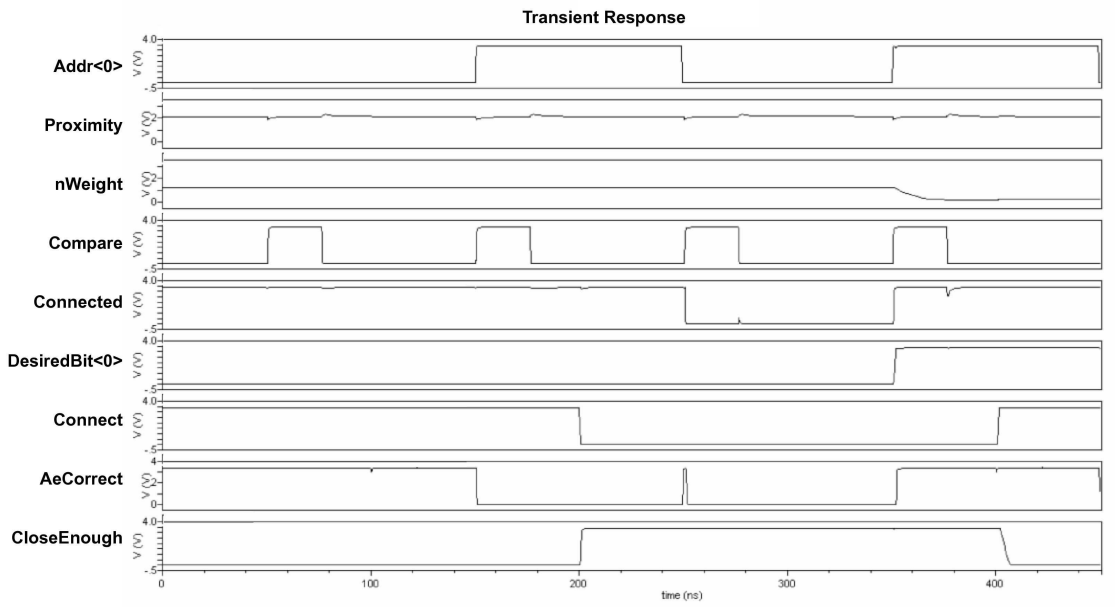


Fig. 4. Trace showing the rewiring of a synapse. The first synapse of a neuron is initially connected("Connected"=true=vdd) to pre-synaptic address 000000000 (only the least significant bit is shown: "DesiredBit¡0¿"). The incoming address starts as 000000000, switches to 000000001 at 150ns, and then switches back and so on every 100ns thereafter (only the least significant bit is shown: "Addr<0>"). "AeCorrect" is the (inverted) output of the NAND gate composed of all monitor bits and this initially indicates that the incoming address is correct, until 150ns at which point the incoming address changes. The random value for connection is initially lower than the "Proximity" value (i.e. nProbConnect is higher) thus "CloseEnough" is false (= 0), until it they switch to respectively high values at 200ns. The random value for disconnection and the corresponding thresholded value "WeakEnough" happen to mirror the aforementioned values (they are not shown here). nProbDisconnect is compared to "nWeight". The two rewiring consideration ("Compare") events at 50ns and 150ns therefore fail to disconnect the neuron because WeakEnough is low. Once WeakEnough goes high the next Compare event at 250ns causes disconnection. Now, although the incoming address matches the stored address, AeCorrect is false, thus the synapse will not accept a spike. At the following Compare event at 350ns, CloseEnough is true and the disconnected synapse is free to connect to the currently latched incoming address, 000000001. Thus DesiredBit goes high and AeCorrect now indicates that the incoming address 000000001 is correct. nWeight is also driven to its minimum (= strong synapse) — a feature of the learning rule we have implemented.

prove beneficial. For example, the adoption of word-serial AER would limit the number of decoder elements in the synapse and promote the adoption of a more standard choice for a repeating memory element. If standard 6-transistor S-RAM elements were used then the size of the repeating memory element would be $\approx 30\mu m^2$ for the same technology and additional read-out circuitry would not be required. Alternatively a DRAM architecture could reduce the size of the repeating unit to as little as $\approx 2\mu m^2$, though noise and power consumption issues may prove prohibitive. As a further alternative, whilst floating gate technology is not best suited to storing synaptic weights because the high frequency of changes usually required by synaptic learning rules would lead to eventual dielectric breakdown, the low rates of synaptic rewiring in natural systems make storage of pre-synaptic addresses on floating gates an attractive option. Analogue storage of many address bits on a single gate could be explored for a possible space saving [23].

Rewiring functions could be centralised to a single circuit on the periphery of each chip. This would remove about 20% of the area of our synapse design, with the expense that the synapse would have to buffer its analogue weight value out to the periphery and some additional signal rails would be required. Note that in our present design, there are two sets of row and column decoders necessary for synaptic rewiring. One targets a synapse for rewiring whilst the other creates a reference location for the proximity calculation. The area required for these scales as $C\sqrt{S_{max}N_{chip}}log_2(S_{max}N_{chip})$ and $C\sqrt{N_{chip}}log_2(N_{chip})$ respectively. This area requirement would not change with the aforementioned proposal.

VIII. Conclusion

We have designed a distributed and locally reprogrammable address event receiver, which allows for arbitrarily large axonal fan-out without reducing channel capacity. Our approach has been mooted before e.g. [11]:

> "Ideally each node should recognise its relevant source events, but our present multi-neuron chips use a DSP chip and lookup table to implement the fan-out from source address to the individual target synaptic addresses."

To our knowledge, however, no such system has been implemented. There is a precedent for simultaneous receipt of events by multiple neurons, in which the same spike was

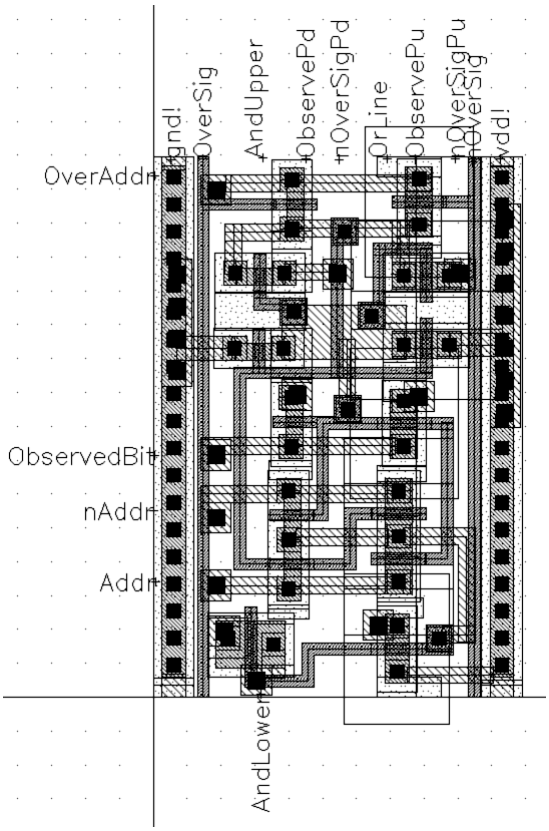


Fig. 5. Layout of synaptic address-monitor bit, in AMS 0.35μ 4-metal 2-poly. Two intermeshing signal layers M2 and M3, and the power layer, M4, have been removed for clarity, though their pin labels and contacts downwards to M1 (larger black squares) are shown. Signal names broadly follow those in fig 2.

delivered to each neuron within a defined area on a chip, implementing a geometrical projective field [24], but this connectivity pattern is fixed and therefore cannot contribute to learning. Our approach also allows for locally implemented probabilistic synaptic rewiring according to a biologically realistic learning rule. Future work will be on demonstrating the abilities of the fabricated chip. Information-theoretic analyses considering constraints of space and power consumption are also anticipated.

ACKNOWLEDGEMENTS

Although not described here, the chip being fabricated contains sections of circuitry acquired from Giacomo Indiveri, Tobi Delbrück and others at INI Zurich. AER Sender schematics were reworked for Cadence by Vasin Boonsobhak. We are grateful to Katherine Cameron for her help and to many people for helpful discussions at the Telluride Neuromorphic Engineering Workshop.

REFERENCES

[1] R. Sarpeshkar, "Borrowing from biology makes for low-power computing," *IEEE Spectrum*, vol. 43, pp. 24–29, May 2006.

[2] D. Willshaw and D. Price, *Modelling Neural Development*. MIT Press, 2003, ch. Models for topographic map formation, pp. 213–244.

[3] H. Cline, "Sperry and Hebb: oil and vinegar?" *Trends in Neurosciences*, vol. 26, pp. 655–661, Dec 2003.

[4] M. Sivilotti, "Wiring considerations in analog VLSI systems, with application to field-programmable networks," Ph.D. dissertation, California Institute of Technology, 1991.

[5] M. Mahowald, "VLSI analogs of neuronal visual processing: A synthesis of form and function," Ph.D. dissertation, California Institute of Technology, 1992.

[6] K. Boahen, "Point-to-point connectivity between neuromorphic chips using address- events," *Circuits and Systems, IEEE Transactions on*, vol. 47, pp. 416–434, 2000.

[7] ——, "A burst-mode word-serial address-event link i: Transmitter design," *Circuits and Systems, IEEE Transactions on*, vol. 51, pp. 1269–1280, 2004.

[8] E. Kandel, J. Schwartz, and T. Jessel, *Principles of Neural Science; 4th Edition*. McGraw-Hill Medical, 2000.

[9] M. Xiong, S. Pallas, S. Lim, and B. Finlay, "Regulation of retinal ganglion cell axon arbor size by target availability: Mechanisms of compression and expansion of the retinotectal projection," *Journal of Comparative Neurology*, vol. 344, pp. 581–597, Oct 1994.

[10] M. Palkovits, P. Magyar, and J. Szentagothai, "Quantitative histological analysis of the cerebellar cortex in the cat," *Brain Res.*, vol. 34, pp. 1–18, 1971.

[11] S. Deiss, R. Douglas, and A. Whatley, *Pulsed Neural Networks*, 1999, ch. A pulse-coded communications infrastructure for neuromorphic systems, pp. 157–178.

[12] D. Chklovskii, B. Mel, and K. Svoboda, "Cortical rewiring and information storage," *Nature*, vol. 431, pp. 782–788, 2004.

[13] B. Taba and K. , Boahen, "Topographic map formation by silicon growth cones," in *Neural Information Processing Systems, Proceedings of,*, 2002.

[14] R. Vogelstein, U. Mallik, J. Vogelstein, and G. Cauwenberghs, "Dynamically reconfigurable silicon array of spiking neurons with conductance-based synapses," *IEEE Transactions on Neural Networks*, vol. 18, pp. 253–265, 2007.

[15] S. Mitra, S. Fusi, and G. Indiveri, "A VLSI spike-driven dynamic synapse which learns only when necessary," in *Proc. IEEE International Symposium on Circuits and Systems*, 2006, pp. 2777–2780.

[16] K. Miller, "Equivalence of a sprouting-and-retraction model and correlation-based plasticity models of neural development," *Neural Computation*, vol. 10, pp. 529–547, 1998.

[17] J. Bougeois and P. Rakic, "Changes of synaptic density in the primary visual cortex of the macaque monkey from fetal to adult stage," *Journal of Neuroscience*, vol. 13, pp. 2801–2820, 1993.

[18] R. Sperry, "Chemoaffinity in the orderly growth of nerve fiber patterns and connections," *Proc. Natl. Acad. Sci. USA*, vol. 50, pp. 703–709, 1963.

[19] S. Song and L. Abbott, "Cortical development and remapping through spike timing- dependent plasticity," *Neuron*, vol. 32, pp. 339–350, Oct 2001.

[20] J. Alspector, R. Allen, V. Hu, and S. Satyanarayana, "Stochastic learning networks and their electronic implementation," in *Neural information processing systems; Proceedings of the First IEEE Conference*, 1988, pp. 9–21.

[21] U. Cilingiroglu and D. Aksin, "A 4-transistor euclidean distance cell for analog classifiers," in *IEEE International Symposium on Circuits and Systems*, 1998, pp. 84–87.

[22] J. Trachtenberg, B. Chen, G. Knott, G. Feng, J. Sanes, E. Welker, and K. Svoboda, "Long-term in vivo imaging of experience-dependent synaptic plasticity in adult cortex," *Nature*, vol. 420, pp. 788–794, 2002.

[23] M. Holler, S. Tam, H. Castro, and R. Benson, "An electrically trainable artificial neural network (etann) with 10240 floating gatesynapses," pp. 50–55, 1990.

[24] T. Serrano-Gotarredona, A. Andreou, and B. Linares-Barranco, "Aer image filtering architecture for vision-processing systems," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 46, pp. 1064–1071, Sept 1999.

Bibliography

A Abusland, TS Lande, and M Hovin. A VLSI communication architecture for stochastically pulse- encoded analog signals. In *IEEE Circuits and Systems Society*, volume 3, pages 401–404, 1996.

DL Adams and JC Horton. A precise retinotopic map of primate striate cortex generated from the representation of angioscotomas. *Journal of Neuroscience*, 23:3771–3789, 2003.

K Albus. A Quantitative Study of the Projection Area of the Central and the Paracentral Visual Field in Area 17 of the Cat I. The Precision of the Topography. *Experimental Brain Research*, 24:159–179, 1975.

PE Allen and DR Holberg. CMOS Analog Circuit Design (second edition). 2002.

J Alspector, RB Allen, V Hu, and S Satyanarayana. Stochastic learning networks and their electronic implementation. In *Neural information processing systems; Proceedings of the First IEEE Conference*, pages 9–21, 1988.

RB Anderson. The power law as an emergent property. *Memory and Cognition*, 29(7): 1061–1068, 2001.

A Antonini and MP Stryker. Plasticity of geniculocortical afferents following brief or prolonged monocular occlusion in the cat. *The Journal of Comparative Neurology*, 369 (1):64–82, 1996.

JV Arthur and K Boahen. Learning in Silicon: Timing is Everything. In *Advances in Neural Information Processing Systems*, 2005.

S Aunet, B Oelmann, S Abdalla, and Y Berg. Reconfigurable subthreshold CMOS perceptron. In *IEEE International Joint Conference on Neural Networks*, volume 3, pages 1983– 1988, 2004.

D Badoni, M Giulioni, and V Dante. An aVLSI recurrent network of spiking neurons with reconfigurable and plastic synapses. In *ISCAS*, 2006.

WJ Bainbridge and SB Furber. CHAIN: A Delay Insensitive CHip Area INterconnect. *IEEE Micro special issue on Design and Test of System on Chip*, 142(4):16–23, 2002.

RJ Balice-Gordon and JW Lichtman. In vivo Observations of Pre- and Postsynaptic Changes during the Transition from Multiple to Single Innervation at Developing Neuromuscular Junctions. *Journal of Neuroscience*, 13(2):834–855, 1993.

C Bartolozzi and G Indiveri. Synaptic dynamics in analog VLSI. *Neural Computation*, 19 (10):2581–2603, 2007.

JA Bednar, A Kelkar, and R Miikkulainen. Scaling self-organizing maps to model large cortical networks. *Neuroinformatics*, 2:275–302, 2004.

CC Bell, VZ Han, Y Sugawara, and K Grant. Synaptic plasticity in a cerebellum-like structure depends on temporal order. *Nature*, 387:278 – 281, 1997.

DL Benson, DR Colman, and GW Huntley. Molecules, maps and synapse specificity. *Nature Reviews Neuroscience*, 2:899–909, 2001.

HKO Berge and P Hafliger. High-Speed Serial AER on FPGA. *IEEE International Symposium on Circuits and Systems*, 2007.

GQ Bi and MM Poo. Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. *J. Neurosci.*, 18: 10464–10472, 1998.

G Billings and MCW van Rossum. Memory retention and Spike Timing Dependent Plasticity. *Submitted*, 2008.

DL Bishop, T Misgeld, MK Walsh, W Gan, and JW , Lichtman. Axon branch removal at developing synapses by axosome shedding. *Neuron*, 44:651–661, 2004.

KA Boahen. Point-to-point connectivity between neuromorphic chips using address-events. *Circuits and Systems, IEEE Transactions on*, 47:416–434, 2000.

KA Boahen. A burst-mode word-serial address-event link i: Transmitter design. *Circuits and Systems, IEEE Transactions on*, 51:1269–1280, 2004.

KA Boahen and AG Andreou. A contrast sensitive silicon retina with reciprocal synapses. In *Advances in Neural Information Processing Systems*, pages 764–772, 1992.

A Bofill-i Petit. *An analogue VLSI study of temporally-asymmetric Hebbian learning*. PhD thesis, University of Edinburgh, 2005.

A Bofill-i Petit and AF Murray. Synchrony detection and amplification by silicon neurons with stdp synapses. *Neural Networks, IEEE Transactions on*, 15:1296–1304, 2004.

JP Bougeois and P Rakic. Changes of synaptic density in the primary visual cortex of the macaque monkey from fetal to adult stage. *Journal of Neuroscience*, 13:2801–2820, 1993.

F Bouzioukh, G Daoudal, J Falk, D Debanne, G Rougon, and V Castellani. Semaphorin3A regulates synaptic function of differentiated hippocampal neurons. *European Journal of Neuroscience*, 23(9):2247–2254, 2006.

JM Brader, W Senn, and S Fusi. Learning real world stimuli in a neural network with spike-driven synaptic dynamics. *Neural computation*, 19(11):2881–2912, 2007.

N Brunel and MCW van Rossum. Lapicque's 1907 paper: from frogs to integrate-and-fire. *Biological Cybernetics*, 97:337–339, 2007.

M Buffelli, G Busetto, C Bidoia, M Favero, and A Cangiano. Activity-Dependent Synaptic Competition at Mammalian Neuromuscular Junctions. *News in Physiological Sciences*, 19:85–91, 2004.

DA Butts, PO Kanold, and CJ Shatz. A burst-based "Hebbian" learning rule at retino-geniculate synapses links retinal waves to activity-dependent refinement. *PLoS Biol*, 5 (3):e61, 2007.

J Cang, L Wang, MP Stryker, and DA Feldheim. Roles of Ephrin-As and Structured Activity in the Development of Functional Maps in the Superior Colliculus. *Journal of Neuroscience*, 28(43):11015–11023, 2008.

E Chicca, D Badoni, V Dante, M D'Andreagiovanni, G Salina, L Carota, S Fusi, and P Del Giudice. A VLSI recurrent network of integrate-and-fire neurons connected by plastic synapses with long-term memory. *IEEE Transactions on Neural Networks*, 14: 1297–1307, 2003a.

E Chicca, G Indiveri, and R Douglas. An Adaptive Silicon Synapse. In *ISCAS*, 2003b.

E Chicca, AM Whatley, P Lichtsteiner, T Delbruck, P Del Giudice, R Douglas, and G Indiveri. A Multichip Pulse-Based Neuromorphic Infrastructure and Its Application of Orientation Selectivity. *IEEE Transactions on Circuits and Systems - I: Regular Papers*, 54(5):981–993, 2007.

C Chiu and M Weliky. Spontaneous Activity in Developing Ferret Visual Cortex In Vivo. *The Journal of Neuroscience*, 21(22):8906–8914, 2001.

DB Chklovskii and AA Koulakov. Maps in the Brain: What Can We Learn from Them? *Annual Review of Neuroscience*, 27:369–392, 2004.

DB Chklovskii, BW Mel, and K Svoboda. Cortical rewiring and information storage. *Nature*, 431:782–788, 2004.

TYW Choi, PA Merolla, JV Arthur, KA Boahen, and BE Shi. Neuromorphic Implementation of Orientation Hypercolumns. *IEEE Transactions on Circuits and Systems - I: Regular Papers*, 52(6):1049–1060, 2005.

U Cilingiroglu and DY Aksin. A 4-transistor euclidean distance cell for analog classifiers. In *IEEE International Symposium on Circuits and Systems*, pages 84–87, 1998.

R Coggins, M Jabri, B Flower, and S Pickard. A hybrid analog and digital VLSI neural network for intracardiacmorphology classification. *Solid-State Circuits, IEEE Journal of*, 30(5):542–550, 1995.

H Colman, J Nabekemura, and JW Lichtman. Alterations in Synaptic Strength Preceding Axon Withdrawal. *Science*, 275:356–361, 1997.

MC Crair, DC Gillespie, and MP Stryker. The role of visual experience in the development of columns in cat visual cortex. *Science*, 279(5350):566–570, 1998.

OD Creutzfeldt, GM Innocenti, and D Brooks. Vertical organization in the visual cortex (area 17) of the cat. *Experimental Brain Research*, 21:313–336, 1974.

JC Crowley and LC Katz. Development of ocular dominance columns in the absence of retinal input. *Nature Neuroscience*, 2:1125 – 1130, 1999.

AP Davison and Y Fregnac. Learning cross-modal spatial transformations through spike-timing- dependent plasticity. *J. Neurosci.*, 26:5604–5615, 2006.

RL De Valois, DG Albrecht, and LG Thorell. Spatial frequency selectivity of cells in macaque visual cortex. *Vision Research*, 22(5):545–559, 1982.

SR Deiss, RJ Douglas, and AM Whatley. A pulse-coded communications infrastructure for neuromorphic systems. In W Maas and CM Bishop, editors, *Pulsed neural networks*, pages 157–178. 1999.

T Delbrück and P Lichtsteiner. Fully programmable bias current generator with 24-bit resolution per bias. In *IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 2849–2852, 2006.

B Dickson. Molecular mechanisms of axon guidance. *Science*, 298:1959–1964, 2002.

S Eberhardt, T Duong, and A Thakoor. Design of parallel hardware neural network systems from custom analog VLSI building block chips. In *Artificial Neural Networks, International Joint Conference on*, volume 2, pages 183–190, 1989.

JG Elias. Artificial dendritic trees. *Neural Computation*, 5:648–663, 1993.

JG Elias and DPM Northmore. Switched capacitor neuromorphs with wide-range variable dynamics. *IEEE Transactions on Neural Networks*, 6(6):1542–1548, 1995.

T Elliott and J Kramer. Coupling an aVLSI Neuromorphic Vision Chip to a Neurotrophic Model of Synaptic Plasticity: The Development of Topography. *Neural Computation*, 14:2353–2370, 2002.

T Elliott and NR Shadbolt. Competition for neurotrophic factors: Ocular dominance columns. *Journal of Neuroscience*, 18:5850–5858, 1998.

T Elliott and NR Shadbolt. A neurotrophic model of the development of the retinogeniculocortical pathway induced by spontaneous retinal waves. *Journal of Neuroscience*, 19: 7951–7970, 1999.

T Elliott, CI Howarth, and NR Shadbolt. Axonal processes and neural plasticity i: Ocular dominance columns. *Cerebral Cortex*, 6:781–788, 1996.

E Erwin and KD Miller. Correlation-based development of ocularly matched orientation and ocular dominance maps: determination of required input activities. *Journal of Neuroscience*, 18:9870–9895, 1998.

E Erwin, K Obermayer, and K Schulten. Models of Orientation and Ocular Dominance Columns in the Visual Cortex: A Critical Comparison. *Neural Computation*, 7(3):425–468, 1995.

DB Fasnacht, AM Whatley, and G Indiveri. A Serial Communication Infrastructure for Multi-Chip Address Event Systems. In *IEEE International Symposium on Circuits and Systems*, pages 648–651, 2008.

BL Finlay, SE Schneps, KG Wilson, and GE Schneider. Topography of visual and somatosensory projections to the superior colliculus of the golden hamster. *Brain Research*, 142:223–235, 1978.

JG Flanagan and P Vanderhaeghen. The ephrins and Eph receptors in neural development. *Annual Review of Neuroscience*, 21:309–345, 1998.

SE Fraser and DH Perkel. Competitive and Positional Cues in the Patterning of Nerve Connections. *Journal of Neurobiology*, 21(1):51–72, 1990.

M Frean. A model for adjustment of the retinotectal mapping, based on eph- dependent regulation of ephrin levels. In *Fifteenth Annual Computational Neuroscience Meeting, Edinburgh*, 2006.

G Friedman. Clock Distribution Networks in Synchronous Digital Integrated Circuits. *Proceedings of the IEEE*, 89(5):665–692, 2001.

S Fusi, M Annunziato, D Badoni, A Salamon, and DJ Amit. Spike-Driven Synaptic Plasticity: Theory, Simulation, VLSI Implementation. *Neural Computation*, 12:2227–2258, 2000.

S Fusi, PJ Drew, and LF Abbott. Cascade models of synaptically stored memories. *Neuron*, 45:599–611, 2005.

RM Gaze, MJ Keating, and SH Chung. The evolution of the retinotectal map during development in xenopus. *Proc R Soc Lond B Biol Sci.*, 185:301–330, 1974.

W Gerstner, R Kempter, JL van Hemmen, and H Wagner. A neuronal learning rule for sub-millisecond temporal coding. *Nature*, 383:76–78, 1996.

A Gierer. Model for the retino-tectal projection. *Proceedings of the Royal Society of London, B*, 218:77–93, 1983.

B. Glackin, M McGinnity, P Maguire, QX Wu, and A Belatreche. A Novel Approach for the Implementation of Large Scale Spiking Neural Networks on FPGA Hardware. In *IWANN*, pages 552–563, 2005.

MA Glover, A Hamilton, and LS Smith. Analogue VLSI integrate and fire neural network for clustering onset and offset signals in a sound segmentation system. In LS Smith and A Hamilton, editors, *Neuromorphic systems: engineering silicon from neurobiology*, pages 238–250. World Scientific, 1998.

L Gnuegge, S Schmid, and SCF Neuhauss. Analysis of the Activity-Deprived Zebrafish Mutant macho Reveals an Essential Requirement of Neuronal Activity for the Development of a Fine-Grained Visuotopic Map. *The Journal of Neuroscience*, 21(10):3542–3548, 2001.

DH Goldberg, G Cauwenberghs, and AG Andreou. Probabilistic synaptic weighting in a reconfigurable network of VLSI integrate-and-fire neurons. *Neural Networks*, 14: 781–793, 2001.

GJ Goodhill. Topography and ocular dominance: a model exploring positive correlations. *Biological Cybernetics*, 69:109–118, 1993.

GJ Goodhill and TJ Sejnowski. Quantifying neighbourhood preservation in topographic mappings. In *Proceedings of the 3rd Joint Symposium on Neural Computation*, pages 61–82, 1996.

GJ Goodhill and J Xu. The development of retinotectal maps: A review of models based on molecular gradients. *Network: Computation in Neural Systems*, 16(1):5–34, 2005.

C Gordon and P Hasler. Biological learning modeled in an adaptive floating-gate system. *IEEE International Synnmposium on Circuits and Systems*, 2002.

J Grutzendler, N Kasthuri, and WB Gan. Long-term dendritic spine stability in the adult cortex. *Nature*, 420:812–816, 2002.

R Gutig, R Aharonov, S Rotter, and H Sompolinsky. Learning input correlations through nonlinear temporally asymmetric Hebbian plasticity. *Journal of Neuroscience*, 23(9): 3697–3714, 2003.

R Guyonneau, R Van Rullen, and SJ Thorpe. Neurons Tune to the Earliest Spikes Through STDP. *Neural Computation*, 17:859–879, 2005.

P Hafliger and HK Riis. A multi-level static memory cell. In *IEEE International Symposium on Circuits and Systems*, 2003.

P Hafliger, M Mahowald, and L Watts. A spike-based learning neuron in analog VLSI. In MC Mozer, MI Jordan, and T Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, page 692, 1997.

A Hamilton, AF Murray, DJ Baxter, S Churcher, HM Reekie, and L Tarassenko. Integrated pulse stream neural networks: results, issues, and pointers. *IEEE Transactions on Neural Networks*, 3:385–393, 1992.

TJ Hamilton, C Jin, and A van Schaik. A Basilar Membrane Resonator for an Active 2-D Cochlea. pages 2387–2390, 2007.

MG Hanson and LT Landmesser. Normal Patterns of Spontaneous Activity Are Required for Correct Motor Axon Guidance and the Expression of Specific Guidance Molecules. *Neuron*, 43(5):687–701, 2004.

JK Harting, BV Updyke, and DP Van Lieshout. Corticotectal projections in the cat: anteriorgrade transport studies of twenty-five cortical areas. *Journal of Comparative Neurology*, 324:2379–414, 1992.

P Hasler, C Diorio, BA Minch, and C Mead. Single transistor learning synapses. In *Advances in Neural Information Processing Systems*, 1995.

DO Hebb. *The Organization of Behavior: A Neuropsychological Theory*. Wiley, New York, 1949.

TK Hensch, MC Crair, ES Ruthazer, M Fagiolini, DC Gillespie, and MP Stryker. Robust two-day ocular dominance plasticity revealed by single-unit recording and intrinsic signal imaging of kitten area 17. In *Society for Neuroscience Abstracts*, volume 21, page 2023, 1995.

Y Hirai, K Kamada, M Yamada, and M Ooyama. A digital neuro-chip with unlimited connectability for large scaleneural networks. In *Neural Networks, International Joint Conference on*, volume 2, pages 163–169, 1989.

AL Hodgkin and AF Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physiol.*, 117:500–544, 1952.

M Holler, S Tam, H Castro, and R Benson. An electrically trainable artificial neural network (etann) with 10240 Floating gate synapses. pages 50–55, 1990.

NP Holmes and C Spence. Multisensory integration: Space, time and superadditivity. *Current Biology*, 15:762–764, 2005.

H Honda. Topographic Mapping in the Retinotectal Projection by Means of Complementary Ligand and Receptor Gradients: a Computer Simulation Study. *Journal of Theoretical Biology*, 192:235–246, 1998.

JJ Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proc. NatL Acad. Sci. USA*, 79:2554–2558, 1982.

DH Hubel and TN Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *Journal of Physiology*, 160:106–154, 1962.

DH Hubel and TN Wiesel. Receptive fields and functional architecture of monkey striate cortex. *J. Physiol.*, 195:215–243, 1968.

DH Hubel, TN Wiesel, and S LeVay. Plasticity of Ocular Dominance Columns in Monkey Striate Cortex. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 278(961):377–409, 1977.

G Indiveri. Neuromorphic bistable VLSI synapses with spike-timing-dependent plasticity. In *Advances in Neural Information Processing Systems*, 2003a.

G Indiveri. A low power adaptive integrate-and-fire neuron circuit. In *IEEE International Symposium on Circuits and Systems (ISCAS)*, volume 4, pages 820–823, 2003b.

G Indiveri, E Chicca, and R Douglas. A VLSI array of low-power spiking neurons and bistable synapses with spike-timing dependent plasticity. *IEEE Transactions on Neural Networks*, 17:211–221, 2006.

EM Izhikevich. Simple Model of Spiking Neurons. *IEEE Transactions on Neural Networks*, 14:1569– 1572, 2003.

EM Izhikevich, JA Gally, and GM Edelman. Spike-timing Dynamics of Neuronal Groups. *Cerebral Cortex*, 14:933–944, 2004.

X Jin, SB Furber, and JV Woods. Efficient Modelling of Spiking Neural networks on a Scalable Chip Multiprocessor. In *IEEE International Joint Conference on Neural Networks*, 2008.

JW Joyner, P Zarkesh-Ha, and JD Meindl. Global interconnect design in a three-dimensional system-on-a-chip. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 12(4):367– 372, 2004.

JK Jun and DZ Jin. Development of Neural Circuitry for Precise Temporal Sequences through Spontaneous Activity, Axon Remodeling, and Synaptic Plasticity. *PLoS One*, 8, 2007.

BM Kampa, JJ Letzkus, and GJ Stuart. Dendritic mechanisms controlling spike-timing-dependent synaptic plasticity. *Trends in Neurosciences*, 30(9):456–463, 2007.

ER Kandel, JH Schwartz, and TM Jessel. *Principles of Neural Science; 4th Edition*. McGraw-Hill Medical, 2000.

MM Khan, DR Lester, LA Plana, A Rast, X Jin, E Painkras, and SB Furber. In *Neural Networks, IEEE International Joint Conference on*, pages 2849–2856, 2008.

AJ King. The superior colliculus. *Current Biology*, 14:335–338, 2004.

WM Kistler and JL Van Hemmen. Modeling Synaptic Plasticity in Conjunction with the Timing of Pre- and Postsynaptic Action Potentials. *Neural Computation*, 12:385–405, 2000.

ZF Kisvárday, DS Kim, UT Eysel, and T Bonhoeffer. Relationship Between Lateral Inhibitory Connections and the Topography of the Orientation Map in Cat Visual Cortex. *European Journal of Neuroscience*, 6(10):1619 – 1632, 2006.

EI Knudsen, S du Lac, and SD Esterly. Computational maps in the brain. *Annual Review of Neuroscience*, 10:41–65, 1987.

T Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43:59 – 69, 1982.

TJ Koickal, A Hamilton, SL Tan, JA Covington, JW Gardner, and TC Pearce. Analog VLSI Circuit Implementation of an Adaptive Neuromorphic Olfaction Chip. *IEEE Trans. Circuits and Systems*, 54(1):60–73, 2007.

AA Koulakov and DN Tsigankov. A stochastic model for retinocollicular map development. *BMC Neuroscience*, 5:30, 2004.

B Linares-Barranco and T Serrano-Gotarredona. On the Design and Characterization of Femtoampere Current-Mode Circuits. *IEEE Journal of Solid-State Circuits*, 38(8): 1353–1363, 2003.

R Linsker. From basic network principles to neural architecture: emergence of spatial-opponent cells. *Proc. Natl Acad. Sci. USA*, 83:7508–7512, 1986a.

R Linsker. From basic network principles to neural architecture: emergence of orientation selective cells. *Proc. Natl Acad. Sci. USA*, 83:8390–8394, 1986b.

R Linsker. From basic network principles to neural architecture: emergence of orientation columns. *Proc. Natl Acad. Sci. USA*, 83:8779–8783, 1986c.

SC Liu. Analog VLSI Circuits for Short-Term Dynamic Synapses. *EURASIP Journal on Applied Signal Processing*, 7:620–628, 2003.

SC Liu, J Kramer, G Indiveri, T Delbrück, T Burg, and R Douglas. Orientation-selective aVLSI spiking neurons. *Neural Networks*, 14(6-7):629–643, 2001.

S Lowel. Ocular dominance column development: strabismus changes the spacing of adjacent columns in cat visual cortex. *Journal of Neuroscience*, 14:7451–7468, 1994.

M Mahowald. *VLSI Analogs of Neuronal Visual Processing: A Synthesis of Form and Function*. PhD thesis, California Institute of Technology, 1992.

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

RC Malenka and RA Nicoll. Long-Term Potentiation - A Decade of Progress? *Science*, 285:1870–1874, 1999.

F Mann, S Ray, WA Harris, and CE Holt. Topographic Mapping in Dorsoventral Axis of the Xenopus Retinotectal System Depends on Signaling through Ephrin-B Ligands. 35 (3):461–473, 2002.

H Markram, J Lubke, M Frotscher, and B Sakmann. Regulation of synaptic efficacy by coincidence of postsynaptic Aps and Epsps. *Science*, 275:213–215, 1997.

T Masquelier and SJ Thorpe. Unsupervised learning of visual features through spike timing dependent plasticity. *PLoS Computational Biology*, 3:e31, 2007.

T McLaughlin, R Hindges, and DDM O'Leary. Regulation of axial patterning of the retina and its topographic mapping in the brain. *Current Opinion in Neurobiology*, 13 (1):57–69, 2003a.

T McLaughlin, CL Torborg, MB Feller, and DDM O'Leary. Retinotopic map refinement requires spontaneous retinal waves during a brief critical period of development. *Neuron*, 40:1147–1160, 2003b.

C Mead. *Analog VLSI and Neural Systems*. Addison Wesley, 1989.

CA Mead and MA Mahowald. A Silicon Model of Early Visual Processing. *Neural Networks*, 1:91–97, 1988.

P Merolla and K Boahen. A Recurrent Model of Orientation Maps with Simple and Complex Cells. In *Advances in Neural Information Processing Systems*, 2003.

PA Merolla, JV Arthur, BE Shi, and KA Boahen. Expandable Networks for Neuromorphic Chips. *IEEE Transactions on Circuits and Systems - 1: Regular Papers*, 54(2):301–311, 2007.

R Miikkulainen, JA Bednar, Y Choe, and J Sirosh. *Computational Maps in the Visual Cortex*. Springer, New York, 2005.

KD Miller. Equivalence of a sprouting-and-retraction model and correlation-based plasticity models of neural development. *Neural Computation*, 10:529–547, 1998.

KD Miller, JB Keller, and MP Stryker. Ocular dominance column development: analysis and simulation. *Science*, 245:605– 615, 1989.

L Mioche and W Singer. Chronic Recordings From Single Sites of Kitten Striate Cortex During Experience-Dependent Modifications of Receptive-Field Properties. *Journal of Neurophysiology*, 62(1):1989, 1989.

S Mitaim and B Kosko. Adaptive Sotchastic Resonance in Noisy Neurons Based on Mutual Information. *IEEE Transactions on Neural Networks*, 15(6):1526–1540, 2004.

S Mitra, S Fusi, and G Indiveri. A VLSI spike-driven dynamic synapse which learns only when necessary. In *Proc. IEEE International Symposium on Circuits and Systems*, pages 2777–2780, 2006.

A Morrison, A Aertsen, and M Diesmann. Spike-timing-dependent plasticity in balanced random networks. *Neural Computation*, 19:1437–1467, 2007.

A Morrison, M Diesmann, and W Gerstner. Phenomenological models of synaptic plasticity based on spike timing. *Biological Cybernetics*, 98:459–478, 2008.

A Mortara, EA Vittoz, and P Venier. A Communication Scheme for Analog VLSI Perceptive Systems. *Solid State Circuits, IEEE Journal of*, 30(6):660–669, 1995.

P Mueller, J Van der Spiegel, D Blackman, T Chiu, T Clare, J Dao, C Donham, T Hsieh, and M Loinaz. A general purpose analog neural computer. *Neural Networks, International Joint Conference on*, 1989.

H Nakahara, K Morita, RH Wurtz, and LM Optican. Saccade-Related Spread of Activity Across Superior Colliculus May Arise From Asymmetry of Internal Connections. *Journal of Neuropysiology*, 96:765–774, 2006.

MJ Neal. An analog VLSI design for a neuron with a choice of learning rules. *Neurocomputing*, 30:185–200, 2000.

K Ohki, S Chung, YH Chung, P Kara, and RC Reid. Functional imaging with cellular resolution reveals precise microarchitecture in visual cortex. *Nature*, 433:597–603, 2005.

K Ohki, S Chung, P Kara, M Hubener, T Bonhoeffer, and RC Reid. Highly ordered arrangement of single neurons in orientation pinwheels. *Nature*, 442:925–928, 2006.

D Orioli and R Klein. The eph receptor family:axonal guidance by contact repulsion. *Trends in Genetics*, 13(9):354–359, 1997.

NA O'Rourke, HT Cline, and SE Fraser. Rapid remodeling of retinal arbors in the tectum with and without blockade of synaptic transmission. *Neuron*, 12:921–934, 1994.

K Pagiamtzis and A Sheikholeslami. Content-addressable memory (CAM) circuits and architectures: A tutorial and survey. *Solid-State Circuits, IEEE Journal of*, 41(3):712–727, 2006.

M Palkovits, P Magyar, and J Szentagothai. Quantitative histological analysis of the cerebellar cortex in the cat. *Brain Res.*, 34:1–18, 1971.

GN Patel, MS Reid, DE Schimmel, and SP DeWeerth. An Asynchronous Architecture for Modeling Intersegmental Neural Communication. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 14(2):97–110, 2006.

CCH Petersen, RC Malenka, RA Nicoll, and JJ Hopfield. All-or-none potentiation at Ca3-ca1 synapses. *Proc. Natl. Acad. Sci. USA*, 95:4732–4737, 1998.

C Pfeiffenberger, J Yamada, and DA Feldheim. Ephrin-As and Patterned Retinal Activity Act Together in the Development of Topographic Maps in the Primary Visual System. *Journal of Neuroscience*, 26(50):12873–12884, 2006.

LA Plana, SB Furber, S Temple, M Khan, Y Shi, and J Wu. A GALS Infrastructure for a Massively Parallel Multiprocessor. *IEEE Design and Test of Computers*, 24(5): 454–463, 2007.

M Porrmann, U Witkowski, and H Kalte. Implementation of Artificial Neural Networks on a Reconfigurable Hardware Accelerator. In *Proceedings of the 10th Euromicro Workshop on Parallel, Distributed and Network-based Processing*, 2002.

MC Prestige and DJ Willshaw. On a role for competition in the formation of patterned neural connexions. *Proceedings of the Royal Society of London. Series B, Biological Sciences*, 190:77–98, 1975.

G Rachmuth and CS Poon. Design of a Neuromorphic Hebbian Synapse Using Analog VLSI. In *Proceedings of the 1st international IEEE EMBS Conference on Neural Engineering*, 2003.

C Rasche and RJ Douglas. Forward- and backpropagation in a silicon dendrite. *IEEE Transactions on Neural Networks*, 12(2):386–393, 2001.

R Rasche and HR Hahnloser. Silicon synaptic depression. *Biological Cybernetics*, 84: 57–62, 2001.

F Rosenblatt. The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. *Psychological Review*, 65:386–408, 1958.

J Rubin, DL Lee, and H Sompolinsky. Equilibrium properties of temporally asymmetric hebbian plasticity. *Phys Rev Lett*, 86:364–367, 2001.

ES Ruthazer and HT Cline. Insights into activity-dependent map formation from the retinotectal system: A middle-of-the-brain perspective. *Journal of Neurobiology*, 59: 134–146, 2004.

ES Ruthazer, CJ Akerman, and HT Cline. Control of Axon Branch Dynamics by Correlated Activity in Vivo. *Science*, 301:66–70, 2003.

R Sarpeshkar. Borrowing from biology makes for low-power computing. *IEEE Spectrum*, 43:24–29, May 2006.

S Satyanarayana, YP Tsividis, and HP Graf. A reconfigurable VLSI neural network. *Solid-State Circuits, IEEE Journal of*, 27:67–81, 1992.

SR Schultz and MA Jabri. Analogue VLSI integrate-and-fire neuron with frequency adaptation. *Electronics Letters*, 31(16):1357–1358, 1995.

R Serrano-Gotarredona, M Oster, P Lichtsteiner, A Linares-Barranco, R Paz-Vicente, F Gómez-Rodríguez, H Kolle Riis, T Delbrück, SC Liu, S Zahnd, AM Whatley, R Douglas, P Häfliger, G Jimenez-Moreno, A Civit, T Serrano-Gotarredona, A Acosta-Jiménez, and B Linares-Barranco. AER Building Blocks for Multi-Layer Multi-Chip Neuromorphic Vision Systems. In *Advances in Neural Information Processing Systems*, 2006.

T Serrano-Gotarredona, AG Andreou, and B Linares-Barranco. AER image filtering architecture for vision-processing systems. *IEEE Transactions on Circuits and Systems - I: Fundamental Theory and Applications*, 46:1064–1071, Sept 1999.

HZ Shouval, MF Bear, and LN Cooper. A unified model of NMDA receptor-dependent bidirectional synaptic plasticity. *Proceedings of the National Academy of Science*, 99 (16):10831–10836, 2002.

DK Simon, GT Prusky, DDM O'Leary, and M Constantine-Paton. N-Methyl-D-aspartate receptor antagonists disrupt the formation of a mammalian neural map. *Proc. Nadl. Acad. Sci. USA*, 89:10593–10597, 1992.

MF Simoni, GS Cymbalyuk, ME Sorensen, RL Calabrese, and SP DeWeerth. A Multiconductance Silicon Neuron With Biologically Matched Dynamics. *IEEE Transactions on Biomedical Engineering*, 51(2):342–354, 2004.

M Sivilotti. *Wiring considerations in analog VLSI systems, with application to field-programmable networks*. PhD thesis, California Institute of Technology, 1991.

PJ Sjostrom, G Turrigiano, and S Nelson. Rate, timing, and cooperativity jointly determine cortical synaptic plasticity. 32:1149–1164, 2001.

S Song and LF Abbott. Cortical development and remapping through spike timing- dependent plasticity. *Neuron*, 32:339–350, Oct 2001.

S Song, KD Miller, and LF Abbott. Competitive hebbian learning through spike-timing-dependent synaptic plasticity. *Nature*, 3:919–926, Sept 2000.

S Song, PJ Sjostrom, M Reigl, S Nelson, and DB Chklovskii. Highly Nonrandom Features of Synaptic Connectivity in Local Cortical Circuits. *PLOS Biology*, 3(3):507–519, 2005.

RW Sperry. Visuomotor co-ordination in the newt (triturus viridescens) after regeneration of the optic nerve. *J. Comp. Neurol.*, 79:33–55, 1943.

RW Sperry. Chemoaffinity in the orderly growth of nerve fiber patterns and connections. *Proc. Natl. Acad. Sci. USA*, 50:703–709, 1963.

RB Stein. A Theoretical Analysis of Neuronal Variability. *Biophysical Journal*, 5:173–194, 1965.

K Straznicky and RM Gaze. The growth of the retina in xenopus laevis: an autoradiographic study. *J Embryol Exp Morphol*, 26:67–79, 1971.

K Straznicky and RM Gaze. The development of the tectum in xenopus laevis: an autoradiographic study. *J Embryol Exp Morphol*, 28:87–115, 1972.

C Sun, DK Warland, JM Ballesteros, D van der List, and LM Chalupa. Retinal waves in mice lacking the beta2 subunit of the nicotinic acetylcholine receptor. *Proceedings of the National Academy of Science of the UNited States of America*, 105(36):13638–13643, 2008.

M Sur and JLR Rubenstein. Patterning and Plasticity of the Cerebral Cortex. *Science*, 310:805–810, 2005.

B Taba and K Boahen. Topographic map formation by silicon growth cones. In *Neural Information Processing Systems, Proceedings of,*, 2002.

J Tomberg, T Ritoniemi, K Kaski, and H Tenhunen. Fully digital neural network implementation based on pulse densitymodulation. In *Custom Integrated Circuits Conference, Proceedings of the IEEE*, pages 12.7/1–12.7/4, 1989.

N Toni, PA Buchs, I Nikonenko, CR Bron, and D Muller. LTP promotes formation of multiple spine synapses between a single axon terminal and a dendrite. *Nature*, 402: 421–425, 1999.

RB Tootell, MS Silverman, SL Hamilton, RL De Valois, and E Switkes. Functional anatomy of macaque striate cortex. III. Color. *Journal of Neuroscience*, 8:1569–1593, 1988.

JT Trachtenberg, BE Chen, GW Knott, G Feng, JR Sanes, E Welker, and K Svoboda. Long-term in vivo imaging of experience-dependent synaptic plasticity in adult cortex. *Nature*, 420:788–794, 2002.

F Tuffy, lJ McDaid, VW Kwan, J Alderman, TM McGinnity, JM Santos, PM Kelly, and H Sayers. Inter-neuron communication strategies for spiking neural networks. *Neurocomputing*, 71:30–44, 2007.

G Turrigiano. Homeostatic signaling: the positive side of negative feedback. *Current Opinion in Neurobiology*, 17:318–324, 2007.

SB Udin and JW Fawcett. Formation of topographic maps. *Annual Review of Neuroscience*, 11:289–297, 1988.

MCW Van Rossum, GQ Bi, and GG Turrigiano. Stable hebbian learning from spike timing-dependent plasticity. *Journal of Neuroscience*, 20:8812–8821, 2000.

A Van Schaik. Building blocks for electronic spiking neural networks. *Neural Networks*, 14:617–628, 2001.

P Venier, A Mortara, X Arreguit, and EA Vittoz. An integrated cortical layer for orientation enhancement. *IEEE Journal of Solid-State Circuits*, 32:177–186, 1997.

RJ Vogelstein. *Towards a spinal neuroprosthesis: Restoring locomotion after spinal cord injury*. PhD thesis, Johns Hopkins University, Baltimore, Maryland, 2007.

RJ Vogelstein, U Mallik, JT Vogelstein, and G Cauwenberghs. Dynamically reconfigurable silicon array of spiking neurons with conductance-based synapses. *IEEE Transactions on Neural Networks*, 18:253–265, 2007.

C von der Malsburg. Self-organization of orientation sensitive cells in the striate cortex. *Kybernetik*, 14:85–100, 1973.

JHB Wijekoon and P Dudek. Simple analogue VLSI circuit of a cortical neuron. In *IEEE International Conference on Electronics, Circuits, and Systems (ICECS)*, pages 1344–1347, 2006.

DJ Willshaw. Analysis of mouse Epha knockins and knockouts suggests that retinal axons reprogramme target cells to form ordered retinotopic maps. *Development*, 133:2705–2717, 2006.

DJ Willshaw and C von der Malsburg. How patterned neural connections can be set up by self-organisation. *Proc. R. Soc. Lond. B.*, 194:431–445, 1976.

DJ Willshaw and C von der Malsburg. A marker induction mechanism for the establishment of ordered neural mappings: its application to the retinotectal problem. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 287:203–243, 1979.

ROL Wong. Retinal waves and visual system development. *Annual Review of Neuroscience*, 22:29–47, 1999.

M Xiong, SL Pallas, S Lim, and BL Finlay. Regulation of retinal ganglion cell axon arbor size by target availability: Mechanisms of compression and expansion of the retinotectal projection. *Journal of Comparative Neurology*, 344:581–597, Oct 1994.

JM Young, WJ Waleszczyk, C Wang, MB Calford, B Dreher, and K Obermayer. Cortical reorganisation consistent with spike timing- but not correlation-dependent plasticity. *Nature Neuroscience*, 10:887–895, July 2007.

E Zaidel and M Iacoboni. *The Parallel Brain: The Cognitive Neuroscience of the Corpus Callosum*. MIT Press, 2003.

LI Zhang, HW Tao, CE Holt, WA Harris, and MM Poo. A critical window for cooperation and competition among developing retinotectal synapses. *Nature*, 395:37–44, 1998.

Y Zhang, A Hamilton, R Cheung, B Webb, P Argyrakis, and T Gonos. Integration of Wind Sensors and Analogue VLSI for an Insect-Inspired Robot. In *Artificial Neural Networks, International Work-Conference on*, pages 438–446, 2007.