Dissertation

submitted to the
Combined Faculties for the Natural Sciences and for Mathematics
of the
Ruperto-Carola University of Heidelberg, Germany
for the degree of
Doctor of Natural Sciences

Presented by
Dipl.-Phys. Matthias Hock
born in Munich, Germany

Date of oral examination: July 2, 2014

Modern Semiconductor Technologies for Neuromorphic Hardware

Referees:	Prof. Dr. Karlheinz Meier
	Prof. Dr. Peter Fischer

Modern Semiconductor Technologies for Neuromorphic Hardware

Neuromorphic hardware is a promising tool for neuroscience and technological applications. This thesis addresses the question to what extent such systems can benefit from advances in CMOS scaling using the existing BrainScales Hardware System as a reference. A 65 nm process technology was selected and basic characteristics were evaluated using prototype chips. A system providing a large number of programmable voltage and current sources, based on capacitive storage cells, was developed. A novel scheme for refreshing the cells is presented. This system has been characterized in silicon. Two components required in a concept for synapse implementation, consisting of primarily digital circuits, were developed and tested in a prototype chip. One is an orthogonal dual-port SRAM with a specialized structure where every 8 bit word stored in the memory can be accessed by a single operation from either port. The second is an 8 bit current DAC which is used for generating postsynaptic events. Finally the analog neuron implementation from the existing system was transfered to the 65 nm process technology using thick-oxide transistors. Simulations suggest that comparable performance can be achieved. In conclusion, modern process technologies will contribute to successful realization of large-scale neuromorphic hardware systems.

Moderne Halbleiter-Technologien für Neuromorphe Hardware

Neuromorphe Hardware ist ein vielversprechender Ansatz für neurowissenschaftliche und technische Anwendungen. Diese Arbeit behandelt die Frage, in welchem Umfang solche Systeme von Fortschritten in der CMOS Technologie profitieren können. Eine 65 nm Prozess-Technologie wurde ausgewählt und wesentliche Eigenschaften mit Hilfe von Prototypen-Chips untersucht. Ein System welches große Zahlen an programmierbaren Spannungs- und Stromquellen, basierend auf kapazitiven Speicherzellen, bereitstellen kann wurde entwickelt. Ein neuartiges Verfahren für das Auffrischen der Zellen wird vorgestellt. Das System wurde *in silico* getestet. Zwei Komponenten die für die Realisierung eines Synapsenkonzepts, basierend auf vorwiegend digitalen Schaltungen, benötigt werden wurden entwickelt. Eine davon ist ein orthogonaler dual-port SRAM mit einer speziellen Struktur, die es erlaubt, auf jede 8 bit Einheit mit einem einzelnen Speicherzugriff von jedem der beiden Ports zuzugreifen. Die zweite ist ein 8 bit Strom DAC, der für das Generieren von postsynaptischen Aktionspotentialen benötigt wird. Abschließend wurde die Neuron-Schaltung des existierenden Systems mit Hilfe von thick-oxide Transistoren in die 65 nm Prozess Technologie übertragen. Simulationen zeigen, dass vergleichbares Verhalten erreicht werden kann. Moderne Prozess-Technologien werden zur erfolgreichen Umsetzung von groß-skaligen neuromorphen Hardware Systemen beitragen.

Contents

1 Introduction

Understanding the human brain is a great challenge for science. Researchers from various disciplines have attempted to elucidate the structure and function of this highly complex system. Philosophers wonder at this very foundation of human nature. Physicians seek to cure neurological and psychiatric disorders such as Alzheimer's Disease and schizophrenia. Neuroscientists investigate the brain's components and their interaction by systematic experiments. The individual components, such as neurons and synapses, are understood fairly well, using sophisticated experimental methods such as the patch clamp technique. Characterizing the high-level functions and interconnections of these components remains a challenge. However, the complex interactions between populations of neurons provide the basis for information processing within the brain. The possibilities of analyzing these processes in living subjects are limited. Therefore mathematical modeling is commonly used to investigate high-level brain functions. Early attempts date back to 1943, see McCulloch and Pitts [1943]. Many current models are based on work presented in Hodgkin and Huxley [1952]. Abstract mathematical descriptions can be used in computer simulations. The objectives of such models range from detailed descriptions, accounting for individual ion channels within single neurons to approaches covering entire functional units of the cortex. However, conventional computer systems operate sequentially, processing transitions between discrete states using only a limited number of processing units [von Neumann 1945]. In contrast, within the brain all neurons are processing information in parallel. This fundamental discrepancy leads to a limited efficiency of simulation systems based on standard computer hardware [Morrison et al. 2005].

An alternative, more efficient approach is the physical emulation of biological components. Processing and transmission of information in the brain is based on electrical phenomena. Each neuron posses a membrane which is characterized by a capacitance and a certain potential. The potential is modulated by currents resulting from the activity of ion channels located within the membrane. Whenever the membrane potential crosses a certain threshold, an action potential or so-called spike is generated by the neuron. This event is characterized by a sudden increase in the membrane potential, followed by a steep decrease and a relatively slow repolarization. This signal is transmitted via the neuron's axon to the dendrites of other, neighboring or distant neurons. The transmission of information between individual neurons occurs at synapses. These connect the axons of the presynaptic neuron to the dendrites of the postsynaptic neuron. Synapses are either excitatory or inhibitory. At excitatory synapses an incoming spike leads to an increase in the membrane potential of the postsynaptic neuron, whereas at inhibitory synapses a spike leads to a decrease in the membrane potential of the postsynaptic neuron. Spikes are considered as digital signals, the information transmitted is only encoded in the timing, not in amplitude or shape of the action potential. The synapses have a distinct property, referred to as the synapse's weight, characterizing its quantitative impact on the membrane potential of the postsynaptic neuron.

Electronic circuits can be designed to mimic the behavior of neurons and synapses, referred to as neuromorphic hardware. Early attempts to build such circuits were described e.g. by Mead in 1988 [Mead and Mahowald 1988, Mead 1990]. Currently various approaches to realize neuromorphic hardware are pursued, a selection is outlined in [Indiveri et al. 2011]. These systems are all based on integrated circuits, allowing for emulation of large neural networks operating at biological time-scales or even faster. In analogy to neurophysiology, all functional units in neuromorphic hardware can operate in parallel, which is an intrinsic advantage compared to simulations based on standard computer technology. However, the realization of networks of biologically relevant dimensions remains a challenge. The neocortex of the human brain contains roughly $2 \cdot 10^{10}$ neurons [Pakkenberg et al. 2003]. None of the existing systems is able to emulate networks in this order of magnitude. Even the emulation of functional units such as the human visual cortex, containing $5 \cdot 10^{5}$ neurons [Wandell et al. 2007], are currently out of reach. Advances in integrate circuit technology are one building block towards improved neuromorphic hardware.

Once available, such systems can be used as an efficient experimentation platform for neuroscientists. A large-scale system implementing biologically inspired information processing might not only be a valuable tool for neuroscientists. Moreover it offers the possibility to utilize this strategy for technical applications. Undoubtedly biology has better solutions for certain tasks in information processing than current efforts using standard computer technology. Typical examples are image processing or the control of complex patterns of physical movement. These are qualities required for instance in the field of robotics. A unique feature of biological systems is the ability of learning. Implementation of biologically inspired learning processes in technical applications would enable novel paradigms in information processing.

In summary, the development of large-scale neuromorphic hardware can be a powerful tool for neuroscientists and open new ways for information processing in technical applications.

1.1 Outline of This Work

This thesis evaluates to what extent large-scale neuromorphic hardware systems based on mixed-signal chips can benefit from recent progress in CMOS technology. The reference system for the evaluation is the currently available BrainScaleS Hardware System. Four different modern semiconductor manufacturing processes of the 45 nm and 65 nm node were evaluated based on process documentation and simulations. Based on this data a TSMC 65 nm low-power process technology was selected for a more thorough investigation, including the development of prototype chips. Circuits for assessment of basic properties of this process technology, such as device mismatch and leakage currents, were integrated into a prototype chip. The question whether it is possible to transfer existing analog circuits to the new process technology using thick-oxide transistors was addressed. As a proof of concept an operational amplifier was successfully transfered. Further more complex components required in mixed-signal neuromorphic hardware were developed and tested. An analog parameter storage system based on capacitive memory cells which provides a large number of programmable voltage and current sources was designed. A novel scheme for programming and refreshing the individual cells is presented. The parameter storage system was integrated into a prototype chip and its performance was characterized. Currently the option of implementing a synapse array, based on the architecture of the existing design, but using mostly digital circuits is being developed. Two key components required for the new concept were developed and implemented. The first of these components is an SRAM block with two orthogonally orientated ports, with a novel, application-specific structure. The second component is a unit element current DAC, designed to produce short current pulses. Both circuits have been tested in a prototype chip. In conclusion, this thesis demonstrates that the 65 nm process technology is suited for implementation of mixed-signal neuromorphic hardware.

1.2 Non Disclosure Agreements

To develop circuits for implementation in modern semiconductor fabrication process detailed knowledge about some aspects of the underlying technology is required. Therefore the foundry offering the process usually provides a “Process Design Kit” (PDK) to the customer. This set of files contains general documentation on the process technology. Furthermore detailed simulation models for the transistors, typically based on BSIM models [Hu et al. 1998] described in the SPICE format [SPICE 2014], are included. Usually also libraries of standard cells for automated synthesis of digital designs are included. An important aspect are the design rules for drawing layouts. These are required by the routing software which automatically connects standard cells in digital design as well as for development of custom circuits. The information included in the PDK is typically considered a corporate secret that must not be shared with third parties, especially not with competing chip manufacturers. As a consequence, customers must accept a Non Disclosure Agreement (NDA) in order to access the PDK of a process technology. Due to Non Disclosure Agreements, some details of the process technologies discussed in the following cannot be published within this thesis.

2 Existing Neuromorphic Hardware

There have been various attempts to build electronic circuits mimicking the behavior of biological brains. Early work in this field has been conducted by VLSI pioneer Carver Mead in 1988 [Mead and Mahowald 1988, Mead 1990].

Since then, systems based on a wide variety of different approaches have been developed. These cover e.g. designs implementing small numbers of precise emulations of Hodgkin-Huxley models based on complex analog circuits [Chen et al. 2010]. An entirely different strategy is used in the SpiNNaker Project [Furber et al. 2013]. It uses large numbers of digital processors, which are combined to a massively parallel computing system, in order to model the behavior of large neural networks. A comprehensive overview, comparing multiple different approaches on the emulation of spiking neural networks, can be found in Indiveri et al. [2011].

For the work presented in this thesis the BrainScaleS Hardware System is used as a reference for the evaluation of different process technologies. The BrainScaleS Hardware System is a highly configurable VLSI[1] implementation of neuromorphic circuits, fabricated in a 180 nm process technology. It utilizes wafer-scale integration of mixed signal ASICs[2] in order to allow for the energy efficient emulation of large scale neural networks. In the following a general overview of the system is presented. The focus in this description however is on the circuits and concepts which are relevant for the work described in the following. Further information on the BrainScaleS Hardware System can be found in Schemmel et al. [2010] and Brüderle et al. [2011].

2.1 The HICANN Chip

The HICANN chip is the building block of the BrainScaleS Hardware System. It is a mixed signal chip, featuring 512 highly configurable analog neuron implementations as well as about 115 k synapse circuits. Additionally there is digital control logic for configuration of the analog circuits as well as the transmission of the neural events. The time constants of the all analog circuits are scaled compared to the time constants found in biology by a factor of about 10^4. This speed up factor makes the system attractive for extensive parameter sweeps and to investigate longterm learning effects. The physical size of a single HICANN chip is $5 \times 10\,\mathrm{mm}^2$.

In Figure 2.1 an overview for the various parts of the chip is shown. The analog part of HICANN is build from two identical blocks, the so called Analog Neural

[1] Very Large Scale Integration

[2] Application Specific Integrated Circuit

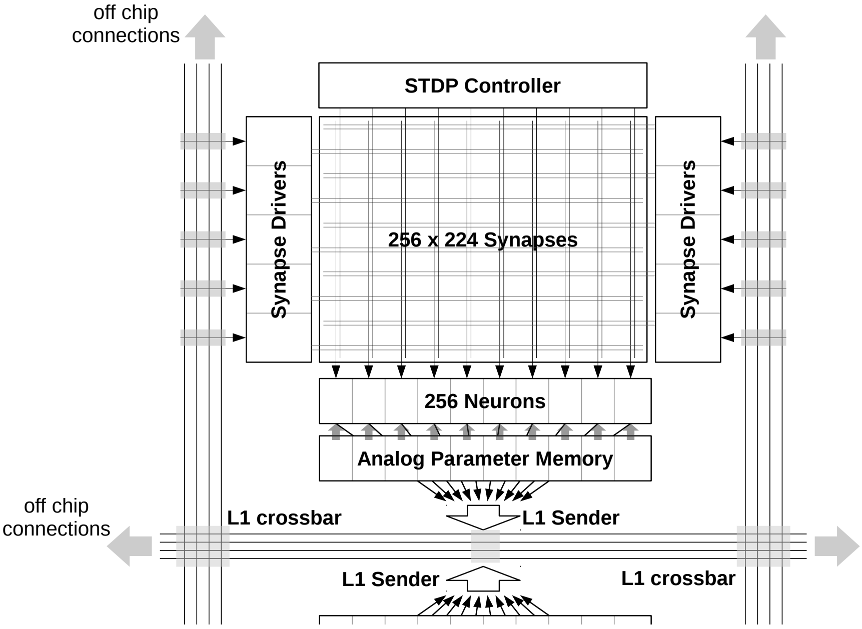


Figure 2.1: Simplified diagram of the main building blocks of a HICANN chip. Only the upper half of the chip is shown. The lower half is identical but mirrored relative to the horizontal L1 lines in the center of the chip.

Network cores (ANN core). The description given in the following refers to the upper ANN core, the lower one is identically organized but its orientation is mirrored. Within one ANN core 256 neuron circuits are placed in a row. On top of the neuron row the synapse array, implementing 256 × 224 synapse circuits, is located. Each neuron circuit receives input from the 224 synapses in the same column. In order to emulate neurons which receive input from more synapses, it is possible to connect multiple neuron circuits. Up to 64 neuron circuits can be configured to emulate a single neuron with up to 14336 synaptic inputs. Obviously the total number of neurons which can be emulated by one HICANN chip decreases accordingly.

The analog neuron circuit is based on the Adaptive Exponential Integrate-and-Fire model, published by R. Brette and W. Gerstner [Brette and Gerstner 2005]. A detailed description of the electronic implementation used in the HICANN chip can be found in Millner [2012]. An important aspect of the neuron circuit is its high flexibility. Each neuron can be configured by 22 individual analog parameters, it requires 11 programmable current sources and 11 programmable voltage sources. Furthermore there are 3 analog parameters which are shared among multiple neuron circuits. On the one hand, the high number of parameters offers various possibilities to modify the characteristics of the neurons, enabling the circuits to mimic the behavior of different types of neurons found in biology [Izhikevich 2004]. On the other hand, the individually adjustable parameters are important to compensate for neuron-to-neuron variation which is introduced by device mismatch. The work presented in Schwartz [2013] is dedicated to calibration of the neuron circuits by adjusting its parameters. In Section 2.1.3 the analog parameter storage system providing the individually programmable voltages and currents is described.

A flexible bus network, termed the Layer 1 (L1) bus, allows for routing of neural events within the HICANN chip. Further it provides the capability to directly connect multiple HICANN chips and exchange neural events between them. Information on the L1 system can be found in Schemmel et al. [2008] and Hock [2009]. The external interface of the chip is termed the Layer 2 (L2) bus which has been developed by the TU Dresden, [Hartmann et al. 2010]. It allows for transmission of configuration data to the chip but is also capable of exchanging neural events.

2.1.1 Processing of Synaptic Events

If the membrane voltage of a neuron circuit crosses its threshold voltage a spike is generated. The event is then digitally processed and sent to the asynchronous L1 bus network. The data transmitted on the bus is a 6 bit identification number which is statically assigned to the presynaptic neuron which emitted the spike. The event is routed in the L1 system to its target, a *synapse_driver* circuit. The *synapse_driver*, is the interface between the L1 communication network and the synapse array. It transmits the lower 4 bit of the received neuron number into an attached row of synapses. Each synapse circuit in the array features an address decoder which compares the data sent by the *synapse_driver* to its address, which is stored in local SRAM cells. In case the numbers are matching, the synapse is activated and produces a postsynaptic event.

The postsynaptic events in the system are represented by current pulses. Every synapse features a unit element current DAC, offering 4 bit resolution. The amplitude of the pulses is proportional to the weight stored in the sending synapse. The length of the pulse is controlled by the *synapse_driver* circuit. Along with the address it sends the `strobe` signal, gating the output of the DACs, to the synapse row. The outputs of all DACs in one column of the synapse array are connected to a shared wire which connects them to the synaptic input circuit of the postsynaptic neuron. This circuit evaluates the integrated amount of charge transfered by the postsynaptic current pulse.

2.1.2 Plasticity

In neural networks the properties of many components change over time, depending on of the activity in the network. These effects are summarized by the term *plasticity*. An overview for the various synaptic plasticity mechanisms observed in biology is presented in Morrison et al. [2008]. In the synapse array of the HICANN chip two plasticity mechanisms are implemented, Short-Term Plasticity (STP) and Spike-Timing Dependent Plasticity (STDP). The STP functionality is implemented in the *synapse_driver* circuit, modulating the strength of the synaptic events by adjusting the length of the `strobe` signal. This mechanism is not further described as it is not relevant for the circuits discussed within this thesis. General information on STP can be found in Tsodyks and Markram [1997], the implementation in the HICANN chip is described in Schemmel et al. [2007].

STDP is a mechanism which changes the weight of individual synapses, dependent on the relative timing between pre- and postsynaptic events. The biological background is presented e.g. in Gerstner et al. [1996], Markram et al. [1997] or Bi and Poo [1998]. According to the concept of STDP, a synapse changes its weight w depending on the correlation between pre- and postsynaptic events. The time difference $\Delta t_{ij} = t_i - t_j$, where t_i is the time at which the presynaptic event occurred and t_j the time of the postsynaptic event, is evaluated for each pair of pre/post spikes. In case the time difference Δt_{ij} is positive, i.e. the presynaptic event occures before the post synaptic event, a causal correlation between the events is assumed. For negative values Δt_{ij} the correlation is assumed to be acausal. The change of the synapse weight Δw_{ij}, triggered by a single pair of pre- and postsynaptic spike, can be described by the following model:

$$\Delta w_{ij} = \pm F_{\pm}(w) \exp\left(-\frac{|\Delta t_{ij}|}{\tau_{\pm}}\right) \tag{2.1}$$

$F(w)$ describes a positive, weight dependent factor and $\tau_{\pm}$ the time constants. The positive signs apply in case Δt_{ij} is positive, the negative ones apply for Δt_{ij} being negative. As a result, pre-before-post spike pairs lead to an increasing weight of the synapse, whereas post-before-pre pairs lead to a reduction of the weight. In literature many STDP rules, using different factors $F_{\pm}(w)$, can be found. Various examples are discussed in Morrison et al. [2008] or Abbott and Nelson [2000]. In the HICANN chip two different components are involved in the realization of STDP. The exponentially weighted time interval between pre- and postsynaptic events is evaluated locally in every synapse by an analog circuit. The results are accumulated and stored in the synapse, represented by the voltage on a capacitor. On top of the synapse array an digital STDP controller is located, implementing the term $F_{\pm}(w)$ and performing the weight updates for the synapses. This controller sequentially processes the individual synapses by reading the accumulated correlation voltage stored on the capacitors and comparing it against an adjustable threshold. In case the value is above the threshold, the controller reads the current weight of the synapse, calculates an update and reprograms the weight of the synapse accordingly. After reading the accumulated voltage, the controller resets the storage capacitor in the synapse.

Accounting for the accumulation happening in the hardware implementation Equation 2.1 needs to be modified as follows:

$$\Delta w_{ij} = \pm F_{\pm}(w) \sum_{n} \exp\left(-\frac{|\Delta t_{ij,n}|}{\tau_{\pm}}\right) \tag{2.2}$$

In this context n is the number of pre-post spike pairs which occurred since the last reset of the accumulation capacitor. For detailed information about the STDP implementation used in the HICANN chip see Schemmel et al. [2006; 2007]. The concept used in the HICANN chip allows for an efficient implementation of STDP, however it lacks flexibility. The controller implemented in the HICANN chip supports only a fraction of the diverse rules which are discussed among neuroscientists. The work presented in Friedmann [2013] describes the development of a processor which can replace the currently used STDP controller in order to allow for a wide variety of STDP rules.

2.1.3 The Analog Parameter Storage System

As mentioned before, the analog circuits in the HICANN chip, especially the neuron circuits, require a large number of individually adjustable parameters. In total, each chip requires 12384 programmable voltage and current sources, more than 90 % of which are used for configuration of the neuron circuits. In the HICANN chip these adjustable parameters are provided by a system based on floating gate transistors. Today the usage of floating gate-based storage is a widespread technology in consumer electronics as non volatile digital memory. However, these devices can also be used to store continuous, analog values. In the following a brief overview of the floating gate parameter storage system is given. A more detailed descriptions can be found in Kononov [2011, Chapter 3] and Millner [2012, Chapter 9].

Basic Principle

Floating gate devices are typically based on transistors featuring a completely isolated gate, referred to as *floating*. The amount of charge trapped on the floating gate determines the gate-source voltage of the transistor, controlling its drain current. Above the floating gate a second gate, termed the control gate, is located. A voltage difference between the control gate and the potential in the channel of the underlying transistor can be used to change the amount of charge trapped on the floating gate, utilizing either a process called Fowler-Nordheim tunneling [Lenzlinger and Snow 1968] or hot-electron-injection [Duffy and Hasler 2003].

Using a programming process including feed back, it is possible to set the drain current of the transistor to a predefined target value. This way floating gate devices can be used as programmable analog current sources. Integrating the floating gate device into a source follower circuit allows to read the voltage on the floating gate, the circuit can be used as a programmable voltage source. Once programmed, floating gate devices typically provide constant output over extended periods of time. Without any potentials applied to the control gate, the tunneling rates discharging the floating gate are negligible for most applications. The absolute numbers depend on thickness and quality of the material isolating the floating gate as well as the precision required.

In the 180 nm process used for the HICANN chip, only a single layer of polysilicon is available to form transistors gates. It is not possible to place an additional control gate on top of the floating one. A way how to integrate floating gate devices into a single-poly process has been demonstrated by K. Ohsaki [Ohsaki et al. 1994]. Instead of a single one, three transistors are required to realize a cell. The gates of these transistors are connected, forming the floating gate. Two of them, termed the *control gates*, are connected as MOS capacitors and required for programming the cells. The third transistor is used for readout. This concept is used for the parameter storage cells of the HICANN chip.

In general, floating gate transistors are an area and power efficient solution to realize programmable voltage and current sources. However, to achieve sufficient tunneling rates during the programming process, voltages significantly above the specified supply voltage range of the process are required [Wu et al. 2012]. The cells implemented in the HICANN chip require a programming voltage of about 11 V. The effective area consumption for a single parameter is $210\,\mu\mathrm{m}^2$. Once programmed, the drift of the output is negligible on the time scales of neuromorphic experiments.

Programming Scheme

On the HICANN chip the floating gate cells in the parameter storage system are organized in four arrays, each holding 1548 voltage and 1548 current cells. The output values of the cells are changed by applying short voltage pulses to the control gates of the individual cells. The process is controlled by comparing the output of the individual cells against a reference generated by a 10 bit voltage DAC after each programming pulse. The output of current cells is converted to a voltage by connecting it to a $150\,\mathrm{k}\Omega$ resistor to allow for comparison with the reference voltage. Number and duration of the pulses required to reach a target value strongly depend on the absolute voltage on the gate. To reduce the number of pulses required to program it to high values the duration of the individual pulses is doubled in case a cell does not reach its target value within a specified number of pulses. Controlling the programming process by comparison of the output against the reference helps to eliminate any cell-to-cell variation introduced by device mismatch. However, after each programming pulse the DAC and comparator need to sequentially process all of the individual cells in the array.

Limitations of the Floating Gate System

As mentioned before, floating gate devices provide an area efficient way to implement programmable current and voltage sources. Once programmed, no power is consumed by the storage devices themselves. However, several limitations are observed in the system which is currently used.

Since every cell in the array is read out by the controller after each individual pulse the programming process is rather slow. With the currently used settings for the controller, writing a full set of parameters takes about 4 s [Hartel 2014]. Due to the high speed up of the neuronal circuits in the chips, the duration of individual experiments is very short. In case each experiment requires a different parameter setup, the reprogramming process might take significantly longer than the experiment itself. In such a scenario the user can not exploit the high speed-up factor of the neural circuits on the hardware.

Changing of parameters during an experiment is not possible with the floating gate-based system. The programming controller applies high voltage pulses to the control gates of the cells when changing their value. As a result, voltages and currents outside the regular operating range are observed at the output of the cells during the process. It can be expected that this has a severe impact on any running experiment.

Furthermore the implementation of floating gate devices increases the overall complexity of the hardware system. In total three additional supply voltages are required to operate the floating gate parameter system in the HICANN chip. Two of these supply voltages are significantly higher than the regular supply voltage. As a consequence specialized ESD[3] protection mechanisms are required and it is essential to follow the correct power up sequence for the individual voltages.

2.2 Wafer-Scale Integration

In order to build a system large enough to allow for the emulation of biologically relevant networks, multiple HICANN chips need to be combined. Due to the accelerated operation, the communication bandwidth required to transmit neural events in the system is very high. According to Schemmel et al. [2010] the presynaptic event rate easily reaches 1.5 Gevents/s per HICANN. Therefore the BrainScaleS Hardware system uses wafer-scale integration to provide the required bandwidth between the individual chips at a reasonable power efficiency.

The HICANN chips are produced on standard 20 cm wafers in a 180 nm CMOS process technology. However, the wafer is not cut into individual dies after fabrication, but used as one unit. As mentioned before, the HICANN chips include an interface, the L1 bus, which allows for direct connection between individual chips. Eight HICANN chips which are placed within a an area of $20 \times 20\,\mathrm{mm}^2$, a so called “reticle”, and their L1 links are directly connected. Figure 2.2 shows a photo of such a wafer. A single chip as well as a reticle are highlighted. For technical reasons, no electrical connections between the in total 48 reticles per wafer can be realized during manufacturing. An additional metal layer, called the post processing layer in the following, is therefore applied to the wafers at the Fraunhofer Institute for Reliability and Microintegration, Berlin. It creates connections between the L1 links of adjacent HICANN chips located in different reticles. Additionally it provides contact pads on the wafers surface which are used for power supply and to provide an external communication interface. Via these pads the wafer is connected to the main PCB using elastomeric connectors[4]. The external communication interface,

[3] Electro Static Discharge

[4] ”Zebra elastomeric connectors”, www.fujipoly.com

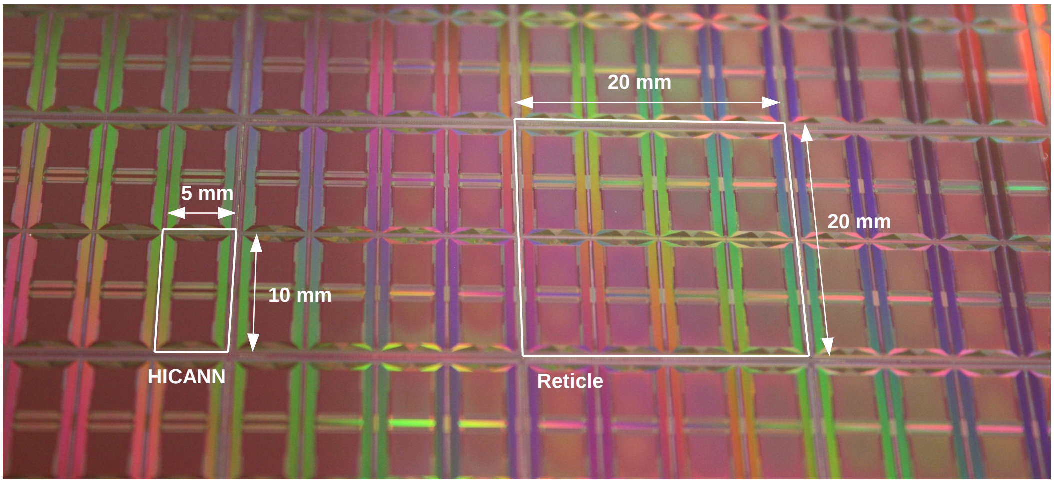


Figure 2.2: Photograph showing a section of a wafer with HICANN chips before the additional post processing layer is applied. A single HICANN chip and a reticle are highlighted.

the L2 bus, connects the individual HICANN chips to FPGA[5] based communication units, which have been developed by the TU Dresden [Hartmann et al. 2010]. These are connected to a cluster of conventional PCs via Ethernet. The communication units also provide the option of interconnecting multiple wafer modules. The host PCs are used to configure the circuits on the wafer and control the execution of experiments. Figure 2.3 shows a photograph of an assembled wafer module. Overall, one wafer module features 348 HICANN chips, providing total numbers of about 200 k neurons and 60 M synapses. The power supplies, the main PCB and the cooling system a designed to allow for a maximum power consumption of 1 kW per wafer. Further there are plans to interconnect multiple wafer modules, to allow for emulation of even larger networks. This can be realized by digital high-speed links between the FPGA based communication units.

[5] Field Programmable Gate Array

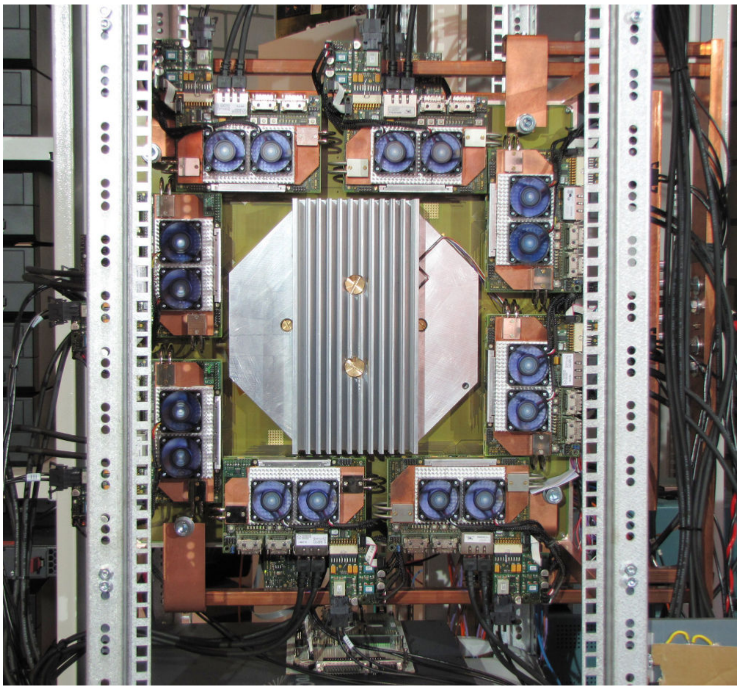


Figure 2.3: Photograph showing an assembled wafer module mounted in a rack. The wafer is covered by the central aluminum bracket, it is surrounded by eight communication units.

3 Modern Process Technologies

The demand for evermore powerful digital circuits has motivated scaling of the CMOS technology throughout the past four decades. These developments were accompanied by multiple challenges. However, with higher resolution in lithography, the introduction of new materials and better understanding of the underlying physics, these challenges could be met and the performance of the resulting systems continually increased, as proposed by Moore's Law [Moore 2006]. Given the high packing density, fast circuit speed, and lower power dissipation, CMOS technology has become the prevailing technology for VLSI applications.

However, these considerations apply mainly for digital circuits. In mixed-signal systems also functional units built from analog circuits are integrated. The performance of these analog building blocks does not necessarily improve with newer CMOS technologies. With every new generation of CMOS processes, the minimum transistor dimensions are constantly decreasing. Simultaneously the nominal supply voltages are also constantly decreasing. This is required to limit the dynamic power consumption of digital circuits, despite the growing numbers of transistors. While the overall performance of digital circuits benefits from the reduced supply voltage, it imposes challenges for analog designs.

The effects of CMOS scaling on analog circuit performance is dealt with in many studies. For reference, see e.g. Annema et al. [2005], Young [2010], Kinget [2007], Sansen et al. [1998]. Taken together, these studies indicate that lower supply voltages are associated with additional effort for the design of analog circuits.

When transferring an exiting mixed-signal system to newer process technology, the performance of the digital components will improve, whereas the realization of analog circuits with equal functionality will require additional design effort.

3.1 Evaluation of Different Process Technologies

The existing BrainScaleS Hardware System is manufactured in a 180 nm process technology, available since 1999. To justify the associated efforts, a considerable advantage should be anticipated when transferring the existing system to a more recently developed process technology. As of today (2014) mass production for processes technologies of the 22 nm node is established [ITRS 2011]. The development of a new system requires the evaluation of multiple prototype chips. Therefore, an important condition for the selection of a new process technology, which is not directly related to any technical characteristics, is that the fabrication of prototype chips based on Multi-Project Wafers (MPWs) has to be offered. For multi-project wafers organizations such as e.g. CMP[1], EUROPRACTICE IC Service[2] or MOSIS[3] collect multiple designs of small chips from various customers and send them to the process vendor as a single design. That way the mask production costs can be split among all participants, allowing for reasonably priced prototype chips. Only process technologies for which MPW runs are offered on a regular basis are interesting candidates for future neuromorphic hardware projects.

Four different process technologies of either the 65 nm or the 45 nm node were preselected. For all of them MPW runs for prototyping are offered by one of the organizations mentioned above and a PDK for the Virtuoso IC Design Platform[4] is available. These process technologies were evaluated based on the documentation included in the PDK and simulation results. First of all basic characteristics such as the maximum possible number of metal layers or the diameter of the wafers were compared. Furthermore a small custom layout was drawn in each of the process technologies to test complexity of the design rules. As expected the design rules for the 45 nm were more restrictive, slightly increasing the effort for the development of custom circuits.

More than half of the overall wafer area of the BrainScaleS Hardware System is covered by digital circuits. These circuits strongly benefit from the increased device density offered by a new process technology. This suggests employing one of the 45 nm process technologies which provide higher integration density. However, static leakage currents in digital devices need to be considered. A significant proportion of the digital part is custom SRAM which is used to store configuration data for analog circuits. Therefore custom SRAM was used as an example for a digital component in simulations for the estimation of leakage currents. These simulations indicate that leakage of the digital components has the potential to substantially contribute to the overall power consumption.

Finally a 65 nm low power process technology offered by TSMC[5] was selected as a promising candidate for future large-scale neuromorphic systems. The main advantage is the anticipated leakage current. Simulations suggest that leakage within the 65 nm TSMC process is by more than one order of magnitude lower than within the alternative process technologies. This process was characterized more thoroughly, including the development of prototype chips, to evaluate whether the effort of porting the system is justified.

[1] Circuits Multi-Projets, http://cmp.imag.fr

[2] http://www.europractice-ic.com

[3] http://www.mosis.com

[4] Cadence Design Systems, Inc., San Jose, CA, USA

[5] Taiwan Semiconductor Manufacturing Company, Ltd., Hsinchu, Taiwan

3.2 Main Characteristics of the TSMC 65 nm Process Technology

Several key features of the new process technology offer benefits compared to the 180 nm process used so far. These are detailed in the following. The BrainScaleS Hardware System uses wafer-scale integration, see Schemmel et al. [2010] and Chapter 2.2. This allows to realize a high communication bandwidth between the individual components with adequate power efficiency. The new process is based on 30 cm wafers offering about twice the area per wafer compared to the 20 cm wafers which are currently used. Therefore a significantly larger number of devices can be connected using power-efficient on-wafer communication. On the other hand, without substantial changes to the design of the underlying chips, the demands for the supporting structures like the main PCB, the cooling system and the mechanical mounting of the system will increase with wafer size. The standard transistors in the process are designed to operate at 1.2 V. The higher speed of the transistors helps to increase the performance of the digital part significantly. Additionally there is the option to use thick-oxide transistors which are optimized for 2.5 V operation. With decreasing size of the transistors the density of digital components increases. For instance the area covered by a single bit of custom SRAM decreases by a factor of about 9, see Chapter 5.1. Standard cell logic is also assumed to offer a 8-fold higher density compared to the former process. The number of metal layers available increases from 6 to 9, improving the routing resources. This simplifies the realization of the long-range connections required for the transmission of neural events. Further the additional metal resources can allow for thorough shielding of sensitive signals. In the BrainScaleS Hardware System a crosstalk problem has been identified, see Friedmann [2013, Section 5.3.4]. In this case sufficient shielding cannot be provided due to limited resources.

Overall, the new process offers considerable improvements in the digital domain.

3.3 Analog Circuits in the 65 nm Process Technology

As mentioned before the development of complex analog circuits in modern process technologies is challenging, mainly due to the low supply voltage. However, the development of analog low voltage circuits is not within the scope of this thesis. An alternative option to realize analog circuits in the 65 nm process technology is to use a higher supply voltage in conjunction with the thick-oxide transistors available. This offers the possibility to operate circuits with a supply voltage of up to 2.5 V. Considering the fact that this is even more than the supply voltage used for the analog circuits in the BrainScaleS Hardware System, it seems feasible to transfer these circuit designs to the new process technology. This option has been investigated within this work. In Chapter 5.5 the transfer of an existing operational amplifier design is described. In Chapter 8 the transfer of the full circuit used for neuron emulation in the BrainScaleS Hardware System is described.

4 Prototype Chips and Experimental Setup

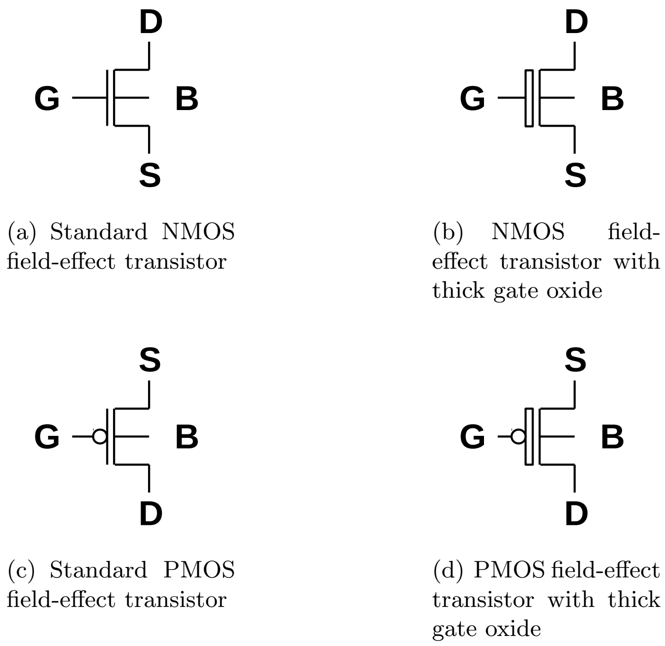
After an initial evaluation of different process technologies, the focus was shifted towards the TSMC 65 nm low power process. Various circuits have been developed for this technology within this thesis and their performance has been evaluated in simulations. Additionally two prototype chips have been developed and tested. The aim was not only to verify the simulation results but also to test the full design flow required to bring the circuits to silicon. The prototype chips developed in the TSMC 65 nm process were submitted for fabrication within the MiniASIC program of EUROPRACTICE[1].

Here a short overview of the circuits implemented in the individual chips is presented. All prototype chips have been tested using the same experimental setup which is described afterwards.

4.1 Schematic Diagrams

For most of the circuits presented within this thesis the corresponding schematic diagrams are shown. The term *schematic* will be used in the following as an abbreviation for *schematic diagram*. All transistors available in the TSMC 65 nm process technology are enhancement type metal-oxide-semiconductor field-effect-transistors (MOSFETs). In literature many different symbols representing MOS transistors in schematic diagrams can be found. To avoid confusion, the symbols used in the following are presented in Figure 4.1. Additionally the following conventions apply: If no bulk contact is shown for an NMOS transistor, its bulk is connected to ground. In case no bulk contact is shown for a standard PMOS transistor, its bulk is connected to the 1.2 V supply. For thick-oxide PMOS transistors the bulk is connected to the 2.5 V supply. In integrated circuits MOS transistors are usually completely symmetric regarding drain and source contact. Consequently there is no marking in the symbols distinguishing these terminals of the transistor. The terminal connecting to the channel of an NMOS transistor which is at the lower potential is considered to be the source contact, the one at the higher potential is referred to as the drain contact. For PMOS transistors the channel contact at the higher potential is referred to as source contact and the one at the lower potential is considered to be its drain.

[1] www.europractice-ic.com



(a) Standard NMOS field-effect transistor

(b) NMOS field-effect transistor with thick gate oxide

(c) Standard PMOS field-effect transistor

(d) PMOS field-effect transistor with thick gate oxide

Figure 4.1: Symbols used in the schematics shown within this thesis. All transistors in the TSMC 65 nm process technology are enhancement type metal-oxide-semiconductor field-effect-transistors.

As it is common practice, an inverted digital signal is marked by a bar over the signal's name.

4.2 Design Flow

The full process from simulating a few transistors of a simple analog circuit to a final mixed-signal chip containing several million transistors involves many different design steps which rely on correct usage of multiple software tools. In the following a short overview of the tools used for the implementation of the prototype chips in the TSMC 65 nm process technology is presented. A detailed description of the steps required to realize VLSI mixed-signal chips can be found in Grübl [2007, Chapter 2].

The analog circuits of the prototype chips have been designed and simulated using Virtuoso Analog Design Environment[2]. Additionally digital designs, described in the high-level hardware description languages Verilog [Verilog 2006] and System Verilog [SystemVerilog 2004], have been integrated into the chips. These are syn-

[2] Cadence Design Systems, Inc., San Jose, CA, USA

thesized into a gate-level netlist using Design Compiler[3]. The synthesized netlist is mapped to a library of standard cells provided by TSMC and implemented by the Encounter Digital Implementation[2] tool. For the static timing analysis PrimeTime[3] is used. The layout of the final chip is checked for violations of the design rules provided by TSMC by Calibre DRC[4]. Calibre is also used for checking the equivalence of the layout against the toplevel netlist of the full design. The implementation flow was created by A. Grübl[5] and A. Hartel[5]. The digital logic in all test chips is designed for reliable operation at a clock frequency of 500 MHz.

4.3 The First Prototype Chip

The first prototype chip contains several analog circuits that allow for testing of basic characteristics of the process technology. In order to test the full design, optimization and verification flow for a digital design of reasonable complexity, an early version of the embedded plasticity processor developed within the Brain-i-Nets project [Brain-i Nets 2012] is included. Details on this 32-bit CPU[6], which has been developed for implementation in the BrainScaleS wafer-scale hardware system, can be found in [Friedmann 2013, Chapter 5.2].

Several analog test circuits are implemented, the individual blocks are listed below. The digital circuits used to configure and control the analog blocks have been written in Verilog based on an example design provided by S. Friedmann[5]. For configuration of the digital control logic a JTAG interface is implemented [JTAG 2001]. The physical implementation of the digital circuits and the overall assembly of the components was done by G. Sidlauskas[5] and A. Grübel.

Analog and digital circuits on the chip are supplied by separate power nets in order to isolate the analog circuits from noise generated by the digital circuits. For both domains a 1.2 V supply as well as a 2.5 V supply, used in conjunction with thick-oxide transistors, are implemented. The in total four different power nets are termed VDD12D, VDD25D, VDD12A and VDD25A. Consequently there are also two separate ground nets, GNDD and GNDA, used for either digital or analog circuits.

In the first version of the chip, the processor was not working due to a mistake in the interface connecting it to the instruction memory. The memory was generated using a memory compiler tool provided from TSMC. The mistake in the interface was not discovered during simulations because the memory model from TSMC, delivered along with the memory macro, behaves differently than the actual memory. As a consequence TSMC offered a free run to submit a new version of the chip with a fixed memory interface. This opportunity was also used to include some minor changes in the analog part of the chip for the second revision. When describing

[3] Synopsys, Inc., Mountain View, CA, USA
[4] Mentor Graphics, Inc., Wilsonville, OR, USA
[5] Kirchhoff Institute for Physics, Heidelberg University
[6] Central Processing Unit

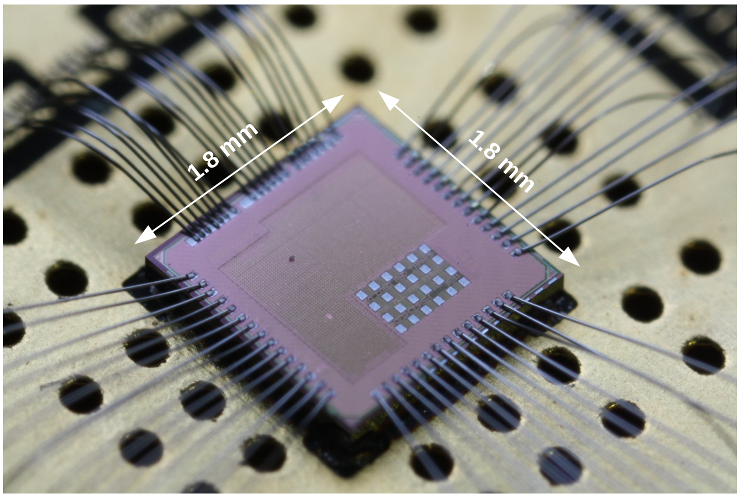


Figure 4.2: Photograph of the first prototype chip. In the lower right corner, the separate bond pads for the floating gate cells are visible. All other circuitry is covered by a grid of wires on the uppermost metalization layer, which distributes the supply voltage.

experimental results, the chips of the first revision are tagged by the number $1a.x$ where 1 refers to the first prototype chip, a refers to the version and x specifies the number of the individual chip. Accordingly results obtained from a chip of the second version will be tagged by $1b.x$. Figure 4.2 shows a photograph of the first prototype chip bonded to a PCB using wedge bond technology. The physical size of the chip is $1.8 \times 1.8\,\mathrm{mm}^2$.

The following list presents an overview for the circuits which have been implemented into the first version of the first prototype chip.

- The first prototype chip contains different designs of capacitive current storage cells. These devices can be used as programmable current sources which are typically required for adjustable analog circuits. The integrated circuits and experimental results are presented in Chapter 6.2.

- An operational amplifier, originally designed for the 180 nm process, see Millner [2008], has been transferred to the new process using thick-oxide transistors. Information on the transfer process and measurement results are presented in Chapter 5.5.

- To simplify the power management in future wafer-scale systems the option of integrated power switches is evaluated. The 1.2 V supply voltage of the test chip can be switched using an internal transistor, see Chapter 5.4.

- In modern semiconductor fabrication processes device mismatch is an important issue. To explore the properties of the process and compare the results against the predictions of Monte Carlo simulations, 4 different sets of 256 identical transistors are implemented. All transistors are supplied with the same gate potential and the resulting drain currents can be measured individually for every transistor. Experimental results obtained from this circuit are not presented within this work but can be found in Graf [2011].

- A small array of 8 × 8 custom SRAM cells is implemented and tested. Information on the design of the cells as well as experimental results are presented in Chapter 5.1.
- In the BrainScaleS Hardware System programmable analog parameters are realized using floating gate devices. To see if this is an option also in the new process, 8 simple single-poly floating gate cells have been integrated. The contacts of the cells are connected to separate bond pads. Experimental results are presented not within this work but can be found in Hüll [2014].

The following changes have been made to the analog circuits for the second version:

- Two separate sense wires are added for the power switching transistor in order to measure the voltage drop over the transistor directly and without any deviations caused by the ohmic resistance of the metal routing and the bond wires, see Section 5.4.3.
- To investigate the spatial distribution of transistor mismatch in 2 dimensions the linear arrangements of transistors were replaced by an array of 48 × 128 identical transistors, measurement results are discussed in Chapter 5.3.
- The floating gate cells have been replaced by an array of 3072 custom-designed SRAM bits. The supply voltage pin of the memory has been connected to a separate bond pad, allowing for direct measurements of the leakage current drawn in a static situation, see Chapter 5.2.

4.4 The Second Prototype Chip

The circuits in the second test chip are focused on the development of more complex circuits, aiming directly at neuromorphic hardware as a target application. The analog circuits implemented are listed below. The digital control logic required for configuration and operation of the analog circuits has been written by A. Hartel. The general characteristics of the second chip are similar to the first one. The physical size is identical as well as the structure of the power and ground nets. Again a JTAG interface is used to control the digital part of the chip. Figure 4.3 shows a photo of the second prototype chip.

Correct operation of the analog circuits in conjunction with the digital control logic has been verified by a mixed-signal simulation environment. It combines

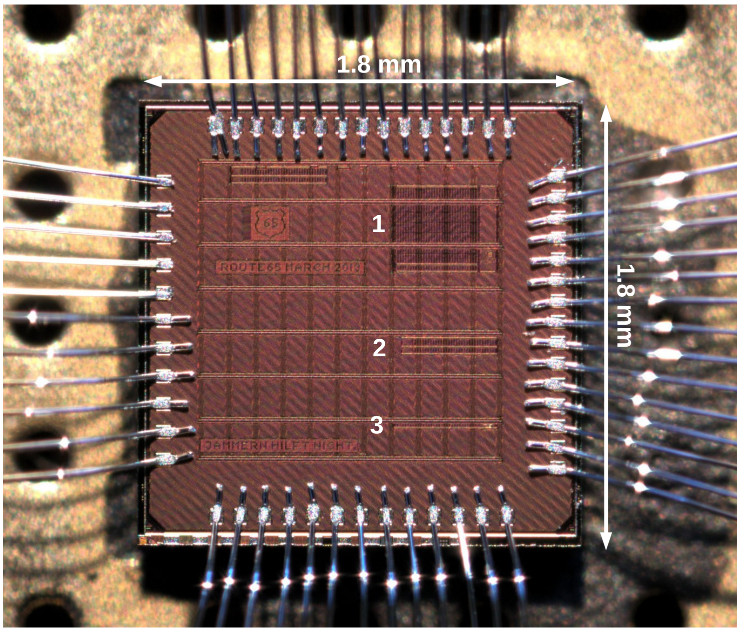


Figure 4.3: Photograph of the second prototype chip. The relative dimensions of the analog circuits on the chip are visible. 1: Array of 32 × 24 capacitive memory cells, 2: Synapse RAM, 3: 32 Synapse DACs.

an analog simulation of the custom analog circuits, using Spectre[2] or Ultrasim[2], with a event driven RTL[7] simulation of the digital components. This simulation environment was set up by A. Hartel. The physical implementation of the digital parts and assembly of all components of the chip was done by A. Hartel and A. Grübel.

The following circuits are integrated into the second prototype chip:

- An array of 32 × 24 analog voltage and current storage cells, including control circuits that manage the programming and refreshing of the cells. The design and experimental results are discussed in Chapter 6.3.
- A block of custom dual port SRAM, holding 256 × 32 bits, is tested. The timing for the SRAM operation is generated by analog circuits to allow for accesses within a single clock cycle. The internal structure of the memory is different from regular dual port SRAM. Specific modifications were made to fit the demands of storing synapse weights in an array optimized for digital processing of synaptic events. Details can be found in Chapter 7.2.
- A row of 32 unit-element current DACs with a resolution of 8 bit is integrated. It is designed to generate postsynaptic pulses in a synaptic array. A description of the design and experimental results are presented in Chapter 7.6.

Four individual chips, tagged chip 2.0 to 2.3, have been used for experiments. Chip 2.0 was severely damaged during testing, therefore results from this chip are only available for a limited number of measurements. Chip 2.1 shows an internal defect, the current consumption on the analog 2.5 V analog supply is 10 mA higher than for the other test chips. This observation is probably related to the fact that one of the output amplifiers which buffers an internal voltage is not working. The exact cause or characteristics of the defect could not be determined. Nevertheless, reasonable measurement results were obtained from chip 2.1 in most cases. However, some results show a high noise level, compared to the other test chips, which is considered to be a result of the internal defect.

4.5 The Experimental Setup

For testing of all prototype chips a basically identical experimental setup was used. Therefore a general description is given in the following. Figure 4.4 presents an overview of the experimental setup. In Table 4.1 the measurement instruments and signal generators used are listed.

[7] Register Transfer Level

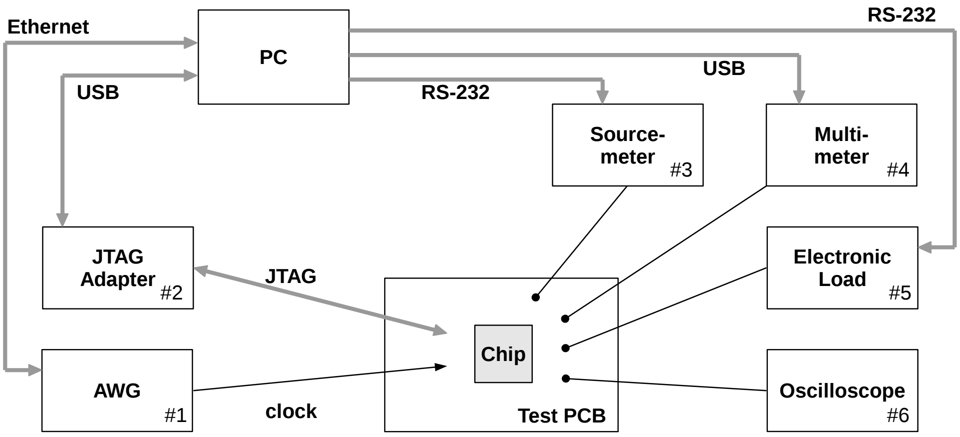


Figure 4.4: Overview for the setup used to test the prototype chips. Detailed information on the devices is given in Table 4.1

#	Device	Model
1	Waveform Generator	Tektronix AWG 7102
2	USB-JTAG Adapter	Xilinx Platform Cable USB II
3	Sourcemeter	Keithley 2635 SYSTEM Sourcemeter
4	Electronic Load	BK Precision 8500
5	Oscilloscope	LeCroy Wave Runner HRO 64Zi
6	Multimeter	Keithley 2100 Multimeter

Table 4.1: Measurement devices and signal generators used for testing of the prototype chips.

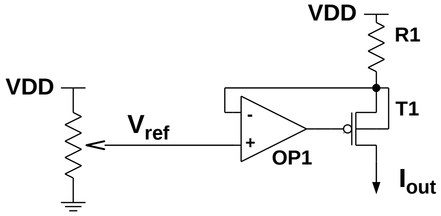


Figure 4.5: Schematic of the circuit used to generate bias currents for the prototype chips.

4.5.1 The Printed Circuit Boards

For testing, each of the prototype chips is glued to a custom-designed Printed Circuit Board (PCB). Wedge bonding technology is used to create electrical connections between the chips and the PCB. The two different versions of the first prototype chip use the same PCB, for the second prototype chip a similar one was designed.

The main function of the test boards is to provide the connectors required for attachment of the measurement instruments and signal sources. Only a small number of active components is used on the boards. To achieve a low noise level, the analog supply voltages at levels of 1.2 V and 2.5 V for the chips are generated locally by linear voltage regulators. The digital supply voltages are directly generated by a standard laboratory voltage source. For several circuits implemented on the chips precisely adjustable bias currents in a range down to 20 nA are required. These are generated on the PCB by standard voltage-to-current converting circuits using operational amplifiers. The schematic of such a current source is shown in Figure 4.5. The operational amplifier OP1 controls the gate voltage of the transistor T1 such that the voltage drop over the resistor is equal to the reference voltage connected to its positive input. A corresponding circuit based on an NMOS transistor is used to provide adjustable current sinks. The reference voltages for the current sources as well as bias voltages for the chips are generated using precision potentiometers.

4.5.2 Software Setup

The experimental setup is controlled by a software program running on the host PC. It is written in C++ [cpp] and controls the operation of the digital part of the test chips via the implemented JTAG interface. The connection between chip and the host PC is realized using the Xilinx Platform Cable USB II. To control the USB-JTAG adapter cable the open source software UrJTAG [UrJTAG 2014] is integrated into the software framework. The general setup, especially the integration of the UrJTAG software, was done by S. Friedmann for testing of the processor implemented in the first prototype chip.

Additionally the software framework allows to control most of the other instruments and devices, read back the measured data and stores it for further processing. The interfaces used for the individual devices are shown in Figure 4.4. The clock for the digital part of the prototype chip is generated by an Arbitrary Waveform Generator (AWG), it is configured by the PC over its Ethernet interface. Internship student J. Kunz helped to integrate the remote control functions of the AWG into the software framework. The integration of the Sourcemeter and the Electronic Load into the software framework was supported by internship student C. Graf.

5 Initial Evaluation of the 65 nm Process Technology

5.1 Custom Static Random Access Memory

Static Random Access Memory (SRAM) is typically the most efficient way to implement digital data storage into mixed signal VLSI chips. The term “static” refers to the fact that it reliably holds the stored data as long as it is powered, however it is volatile. In general it provides fast operation, a rather low static power consumptions and a reasonable density of bits per area.

In general, other memory concepts, such as Dynamic Random Access Memory (DRAM), provide a higher density. However DRAM is typically not suited for implementation into mixed signal chips. The single bits only have a limited storage time, a controller that performs regular refresh operations is required. This increases the overall complexity of the memory significantly and leads to a rather high static power consumption. The maximum possible density for DRAM is only achieved in chips fabricated in a technology that offers special process options.

In highly configurable neuromorphic mixed-signal circuits, configuration data for analog circuits is typically stored in custom SRAM. The internal state of the SRAM bits needs to be connected to the circuit which is configured, therefore the individual bits are integrated into the full custom layouts of the analog circuits. In the BrainScaleS Hardware System more than 38 MB of custom SRAM are integrated per wafer, covering approximately 10 % of the total silicon area.

Custom cells of standard 6T SRAM have been designed in the 65 nm process technology and implemented into the first prototype chip.

5.1.1 SRAM Operation

Figure 5.1 shows a schematic of a standard 6 transistor SRAM cell. The central element of SRAM cells is the latch built from transistors T2 to T6. These are connected to form two cross coupled inverters. Once written to one configuration, the latch holds its state, presenting a logical `1` at the output of one inverter (`Q`) and a logical `0` at the output of the other $\overline{\mathtt{Q}}$. In order to write or read the state of the latch, two additional transistors, T1 and T2, are required. These are typically termed the *access transistors* of the cell and connect the nodes `Q` and $\overline{\mathtt{Q}}$ to external signals.

For efficient implementation, SRAM cells are typically arranged in an array. In this configuration, the signal controlling the gates of the access transistors is shared

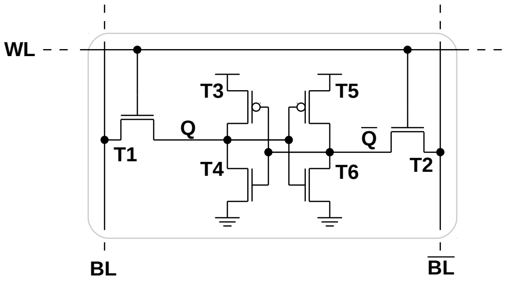


Figure 5.1: Schematic for a standard 6 transistor SRAM cell.

among all cells in a row, it is referred to as the wordline (`WL`). The wires that allow for accessing the state of the latch via the access transistors are shared between all cells in a column, and called bitlines (`BL`). One of these lines is connected to the access transistor at positive output of the latch `Q`, the other one to the access transistor at the inverted output $\overline{\texttt{Q}}$. In the following the process of accessing SRAM cells is briefly described. A detailed description of the mechanisms involved can be found in e.g. in Chandrakasan et al. [2000, Chapter 14.2].

Writing the Cell

In order to write the state of latch, the bitlines have to be set accordingly by the SRAM control logic and the wordline is activated. If the cell is already in the correct state this does not have an effect. If the current state of the cell is different from the one it is written to, the external signals have to overwrite the value which is actively held by the inverters. This is only possible if the conductivity of the access transistors is sufficiently low compared to the conductivity of the PMOS transistors in the inverters. Figure 5.2 shows the voltage at the bitlines and the internal nodes `Q` and $\overline{\texttt{Q}}$ of an SRAM cell during a write process. Since the wordline is shared among all cells in a row and each column has an individual pair of bitlines, a full row can be written at once.

Reading the Cell

To read a single cell of SRAM it is sufficient to activate the wordline, the internal state of the cell will be visible at the bitlines. In a large array of cells, the situation is more complicated. The long bitlines, used for reading and writing of all cells within a column, have a significant capacitance. As a consequence it is possible that the bitline pair of a cell is charged to opposing digital values from a previous operation when a read access is initiated. If the wordline of a cell is activated in this situation, it can happen that the state of the cell is overwritten by the current state of the bitlines. In this case the stored information is lost. To prevent this situation a process called *precharge* is required before the wordline is activated. Before each read access, both bitlines of a column are charged to the supply voltage by the SRAM control logic. In this state the bitlines BL and $\overline{\texttt{BL}}$ are not logically inverted to each other. Once the bitlines are charged, the controller stops driving them and the wordline is activated. If the balance between the dimensions of all transistors involved is correct, this state can not destroy the state of the cell. Instead, one of the bitlines, depending on the logical value stored in the cell, is discharged to 0 while the other one remains at 1. The state of the bitlines can then be read by the SRAM control circuit. Figure 5.2 shows the voltage at the bitlines and the internal nodes Q and $\overline{\texttt{Q}}$ of an SRAM cell during a read process. For large arrays it takes a significant amount of time for the controller to charge the full bitline to the supply level. An even longer time interval is required to discharge one of the bitlines as this has to be done by the transistors in the cell. The bitline capacitance is connected to ground via a serial connection of an access transistor and an NMOS transistor of one of the inverters in the latch. To achieve a high memory density, the transistors in the cells are typically as small as possible, offering only a limited conductivity. In order to reliably detect a 0 by the digital circuits reading the state of the bitlines, the respective bitline needs to be discharged to level significantly below half of the supply voltage.

However, there are several possibilities to speed up the reading process of SRAM, a common option is the usage of so-called *sense amplifiers*. To read the state of a single cell of the SRAM it is not necessary to wait until the bitline which is representing a 0 is fully discharged. Instead it is possible to evaluate the difference between the two bitlines of a pair using an amplifier with a differential input stage. That way a small voltage difference on the bitlines is sufficient to detect the correct state of the cell that is read. The topic of sense amplifiers is discussed in more detail in Chapter 7.2. Further there are implementations of SRAM which include additional transistors in every cell in order to speed up the reading process, examples can be found in Athe and Dasgupta [2009].

SRAM Controller

In order to access SRAM, the bitlines have to be set accordingly and the wordline needs to be activated with appropriate timing. The simplest option is to use a controller realized in digital logic for this task. In the HICANN chip such digital controllers are used for all custom SRAM cells implemented. Since the minimum time interval which can be realized by such a controller is one clock cycle, the duration of a read access is at least two clock cycles, regardless of the time actually required by the memory.

Another possibility is to use a controller which generates the sequence of the signals based on analog delay elements, an option which is particularly interesting for high speed applications. It is possible to realize memory controllers which perform complete access operations within one cycle of the clock used in the system. An SRAM array using analog delay elements for timing control is described in Section 7.3.2.

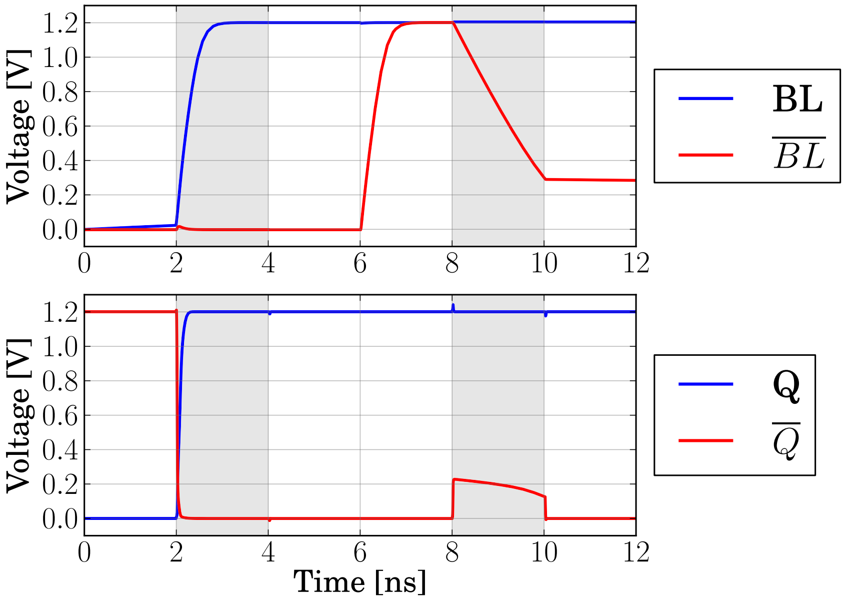


Figure 5.2: Write and read process of an SRAM cell. The voltage at the bitlines as well as the voltage at the internal nodes `Q` and $\overline{\texttt{Q}}$ are shown. During the intervals marked by the grey background the worldline of the cell is active. In the interval from 2 to 4 ns the cell is written to `1`. The following read process is initiated at 6 ns by the precharging of the bitlines. Subsequently the wordline is activated in the interval from 8 to 10 ns where the cell discharges the inverted bitline $\overline{\texttt{BL}}$.

5.1.2 SRAM Macro Blocks

Since SRAM is required in many digital designs and its performance is critical, process vendors typically provide IP[1] macro blocks of highly optimized memory cells. Within these macros the cell density is significantly higher than for custom SRAM implementations. The layout of the single cells is aggressively optimized and tested in silicon by the vendor. The regular structure within the SRAM arrays and the extensive testing allows for reliable designs which use minimum spacings and dimensions that clearly violate the design rules that apply for custom designs. It is not possible to achieve a comparable density for custom SRAM which has to be consistent with the general design rules. In case of the TSMC 65 nm process the area consumption of a single bit of 6T SRAM is $0.5\,\mu\mathrm{m}^2$.

Besides a high density, SRAM macro blocks provide high speed access to the cells. This is achieved by optimized control logic and the usage of sense amplifiers. Macro blocks are the ideal choice where large amounts of SRAM are required by digital designs such as processors. The Plasticity Processor implemented in the first prototype chip uses two macro blocks of 16 k bit size each as data and instruction memory, see Friedmann [2013, Chapter 5.2].

However, macro blocks are not suited to store configuration data of analog circuits. Despite the fact that the cells are very compact, the usage of macro blocks is inefficient for small memory arrays. In this case the overall density is reduced significantly, due to the area overhead caused by the integrated control logic and the sense amplifiers. Further, for configuration memory used in analog circuits it is required that the internal state of each bit is statically wired to the circuit which is configured. This is not possible in macros, the state of the single bits can only be accessed for individual rows using the interface of the attached control logic.

5.1.3 Implementation Details

The custom SRAM cells described in the following have been designed for to be used as configuration memory for analog circuits. Optimizing the density is more important than speed. As mentioned before, the dimensions of the transistors in an SRAM cell have to be balanced carefully to allow for reliable read and write accesses to the memory. Simulations of the 6T SRAM cells have shown that a configuration in which all transistors have identical dimensions is working reliably. The robustness has been verified by simulations covering different process corners and using Monte Carlo simulations to account for device mismatch. Though these dimensions might not result in the fastest possible operation, the identical dimensions of all transistors are helping to design a compact layout. Figure 5.3 shows a sketch of the transistor arrangement that was found to cover the smallest area per bit. In both dimensions of the array, the orientation of neighboring cells is mirrored to each other. Bitline and worldline contacts are shared between neighboring cells to save area. In this

[1] Intellectual Property

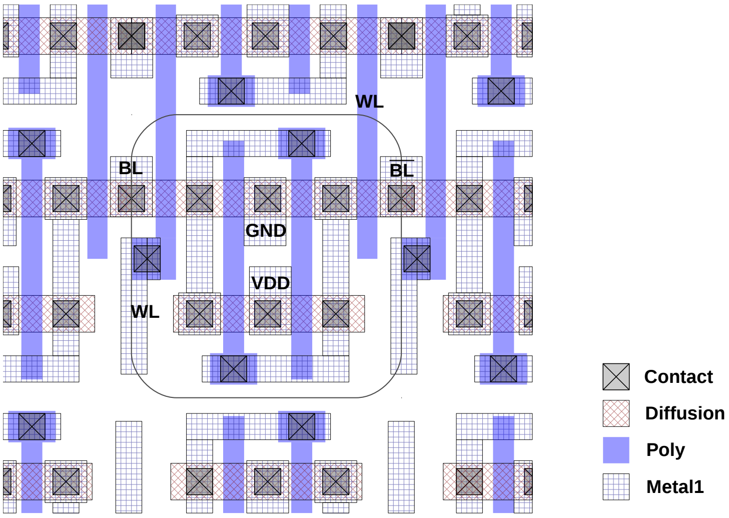


Figure 5.3: Sketch of the layout for the custom SRAM cells. Only the layers Diffusion, Poly, Contact and Metal1 are shown. A single cell is highlighted by the gray box. The orientation of the cells is such that the wordlines run vertically, the bitlines horizontally (not shown). Not to scale.

arrangement one bit covers an area of $1.1\,\mu\mathrm{m}^2$. This is more than twice the area consumed by a single bit in an SRAM macro block.

A small array of 8 × 8 bits of the custom SRAM cells is integrated into the first prototype chip. The bit- and wordlines are connected to a configurable digital SRAM controller which has been written by J. Schemmel [Schemmel et al. 2014, Section 4.4.2].

5.1.4 Experimental Results

The SRAM is tested by writing all 64 cells to random values, reading back the memory contents afterwards and comparing it to the originally written data. The digital SRAM controller is configured to use the shortest possible time intervals. For writing the bitlines are driven to the correct logical values and the wordline is activated for a single clock cycle. For reading the bitlines are precharged for one cycle and then the wordline is activated for one more cycle to read back the value. Due to the short bitlines in the small array, the time constants of all operations are significantly below 2 ns, the duration of one cycle at the nominal clock frequency of the test chip. The design has been tested for reliability at the nominal supply voltage of 1.2 V and a the nominal clock frequency of 500 MHz. Two individual chips have been tested, for each a total amount of 1 MiB of random data has been written and read back. No errors have been detected in this experiments, the basic design of the SRAM cell seems to work reliably. However, the number of programming cycles that were tested are low, compared to typical test procedures for digital circuits. It is limited by the slow JTAG interface connecting the prototype chip to the PC. The absolute number of cells in the array is also too small to detect any potential yield problems. Furthermore, the probability for a cell in the array being affected by severe device mismatch is rather low.

J. Kunz, an internship student, was involved in testing the SRAM array implemented in the first prototype chip. He carried out additional tests such as sweeps of the supply voltage and the clock frequency. For more detailed information on these aspects see his report [Kunz 2012]. It can be summarized as follows. Reliable operation of the memory is possible down to 700 mV supply voltage for a reduced clock frequency of 100 MHz. It was also tested down to which level the supply voltage can be reduced in between accesses without losing the information in the memory. The content is preserved as long as the supply voltage is reduced not further than 250 mV. These additional experiments were performed in order to evaluate options how to reduce the power consumption caused by static leakage currents, an issue that will be discussed in the next section.

Overall, the custom drawn SRAM implemented in the first prototype chip is working reliably. However, one has to keep in mind that the dimensions of the array on the prototype chip is much smaller than in any typical application. The significance of the results presented here is therefore limited.

5.2 Static Power Consumption of Custom SRAM

Ongoing, aggressive scaling of the CMOS technology allows for constantly increasing device densities in integrated circuits. To limit the power consumption in active digital circuits, the supply voltage of the circuits is constantly decreased. In order to sustain a high speed for the circuits, the threshold voltage of the transistors needs to be scaled accordingly. As a result of the scaling, the impact of leakage currents on the overall power consumption constantly increases. An comprehensive overview for the various leakage mechanisms found in deep-submicrometer process technologies is presented in Anis [2003]. According to Qi et al. [2006], the subthreshold leakage current is typically dominating in process technologies of the 65 nm node. The impact of gate-oxide tunneling is also reported to contribute significantly in this regime, see Sirisantana and Roy [2004] Simulations for the TSMC 65 nm low power process technology indicate that leakage currents are relevant for the overall power consumption of wafer-scale systems. In order to verify the simulation results, the static power consumption of custom 6T SRAM cells has been measured.

5.2.1 Experimental Results

The static power consumption for a single digital circuit is difficult to determine as the absolute currents involved are small. To measure the static power consumption of custom 6T SRAM, a memory block containing 3072 bit is integrated into the second version of the first prototype chip. All bit- and wordlines are statically

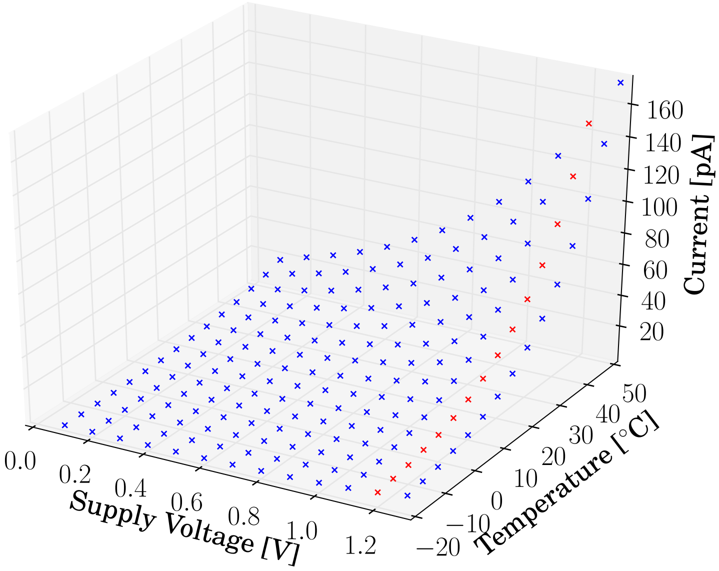


Figure 5.4: Leakage current per bit of 6T SRAM over supply voltage and temperature. The plot is based on measurement results obtained from a block of 3072 SRAM cells implemented into the prototype chip. The nominal supply voltage of 1.2 V is marked in red. The measurement error on the current, obtained from multiple independent repetitions of the measurement, are too small to be visible. The absolute error on the temperature is estimated to be 1 °C, for clarity no according error bars are depicted.

connected to ground. To allow for precise measurements, the power supply of the memory cells is connected to a separate bond pad. Furthermore this pad is not connected to any Electro Static Discharge (ESD) protection circuits to avoid additional contributions from leakage in these circuits [Sarbishaei 2007]. As a consequence the chips needs to be handled carefully.

To generate the supply voltage and measure the current the Sourcemeter was used. The test PCB carrying the chip was placed in a climate chamber to control the temperature of the setup. Figure 5.4 shows the result of measuring the current consumption of the memory at different voltages and temperatures. The results are divided by 3072 in order to display the average leakage current per single bit.

In Figure 5.5 the result of a simulation evaluating the leakage current caused by a single bit for different corners and different temperatures at the nominal supply voltage of 1.2 V is presented . The leakage current correlates strongly with the process corner. The corresponding set of measured data is also shown, it matches the simulation results based on typical transistor characteristics. For the fast corner and at 50 °C the leakage is about 25 times larger than for the typical corner at the

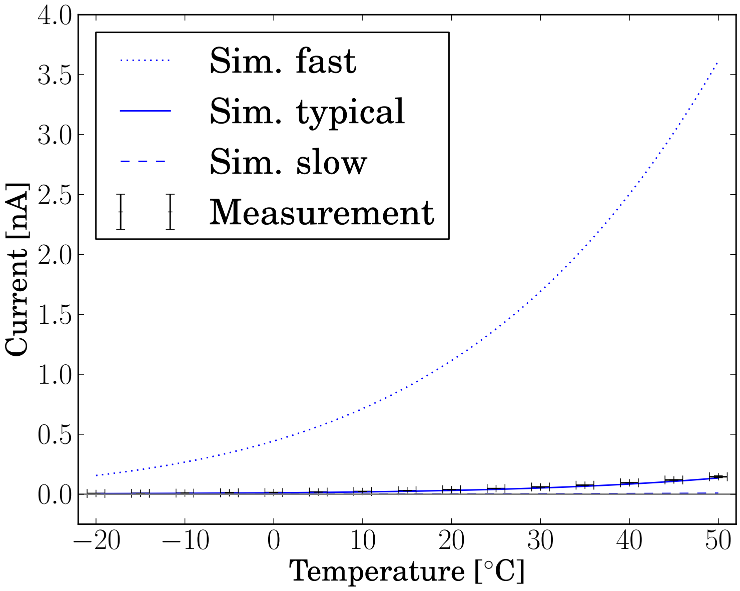


Figure 5.5: Simulation of the leakage current per bit of 6T SRAM at a supply voltage of 1.2 V over temperature. Results are shown for typical (solid), fast (dotted) and slow (dashed) process corner. The result of the corresponding measurement from Figure 5.4 is also shown. The measurement results correspond to the simulation for typical process characteristics.

same temperature.

5.2.2 Static Power Consumption of a Wafer-Scale System

The results of simulations and measurements of the leakage current caused by custom SRAM can be used to derive a rough estimation for the static power consumption of a full wafer-scale system. The leakage current per area is assumed to be comparable for custom SRAM and digital standard cell logic. In both cases minimum length transistors are arranged at high density. Based on numbers from the current BrainScaleS Hardware System, it can be further assumed that at least half of the area of a wafer is covered by custom SRAM or standard cell logic. For a wafer of 30 cm diameter and typical characteristics, operated at 50 $°C$, the estimated power consumption caused by leakage is about 12 W for the digital part alone. For a wafer with fast corner characteristics this number changes to more than 300 W. This number emphasizes the importance of considering not only the active but also static power consumption of the system during development. To assess the actual static power consumption of standard cell logic in this process technology further investigation is required. The SRAM cells described here are based on standard core transistors. Using the high threshold voltage option available for the core transistors is an option to reduce the leakage current in the SRAM cells. However, this measure reduces the speed of the memory cells.

The static power consumption caused by analog circuits is much more difficult to estimate, it strongly depends on the details of the circuits implemented. However this part will also contribute to the static power consumption of the system.

5.3 Transistor Mismatch

Manufacturing of integrated circuits in modern process technologies is a complex process involving many different steps. Statistical fluctuation of various parameters during the fabrication process lead to a statistical variation in the characteristics of the produced devices. These parameter fluctuations lead, depending on their origin, to wafer-to-wafer, die-to-die and also device-to-device variation [Bowman et al. 2002, Agarwal and Nassif 2007].

As a result of die-to-die variations, the characteristics of all devices of the same type and located within the same die are shifted against the expected value. These are typically caused by a global deviations in the dopant concentrations, charge carrier mobility or the thickness of the gate oxide. In order to test the robustness of circuits against die-to-die variation during the design phase, usually additional simulation models are provided by the foundries. These describe the characteristics of the transistors for several corner cases of the expected die-to-die variation.

The variation between the characteristics of two identically designed devices on the same die is termed device mismatch. This variation is typically caused by statical fluctuation of the dopant densities in the channels of the transistors and variation in their geometry, introduced by uncertainties in lithography process. In analog precision circuits device mismatch is an important issue. In many circuits good matching between pairs or within groups of transistors is crucial. The relative variation of the characteristics of a transistor is related to its size. As a rule of thumb it is assumed that the parameter variation is proportional to $1/\sqrt{w \cdot l}$, where w denotes the width and l the length of the transistors gate [Pelgrom et al. 1989, Lovett et al. 1998]. Therefore the transistors used in analog circuits are typically large, compared to the minimum size which can be realized in the respective process technology.

However, the matching between devices does not only depend on their absolute size but also on the general structure of the layout. Typically devices in close spatial proximity match better than devices located in larger distance. This is based on the assumption that variations in the process parameters are happening continuously and at least some of them are changing on spacial scales larger than the size of the individual devices. Another layout strategy for precise matching of analog transistors is to split each of the transistors into multiple fingers and arrange them in an interleaved fashion. If multiple transistors need to be matched, the individual fingers should be arranged in a common centroid structure, see Long et al. [2005]. These measures compensate for global gradients in the characteristics of the devices. In general a layout should be as homogeneous as possible to achieve good matching.

To account for the effect of device mismatch during the design process, models including information on the expected distribution for multiple parameters of the device models, are typically provided by the semiconductor foundries. The models allow for Monte Carlo simulations. In each simulation run, the parameters used for each individual transistor are randomly drawn from the corresponding distribution. Multiple simulation runs allow for an estimation of the impact of devices mismatch on the circuit. As mentioned before, spatial distance of the devices as well as the structure of the layout are assumed to have an impact on the matching. However, typical Monte Carlo simulation setups are not able to account for these effects. Therefore it is interesting to measure the transistor mismatch for a typical layout and compare it to results of Monte Carlo simulations.

Circuits allowing for quantitative characterization of transistor mismatch are implemented into the first prototype chip. The measurements and analyses of the results were preformed in collaboration with internship students, only the most important results will be presented here. A detailed evaluation of the results can be found in Kunz [2012].

5.3.1 Measuring the Mismatch in an Array of Transistors

In the second version of the first prototype chip a homogeneous array of transistors has been implemented.

Implementation Details

An array of 48 × 128 identical transistors is included in the chip. The length of the transistors gate is 600 nm, the width is 400 nm. The total dimensions of the array are 116 μm × 164 μm. The drain current of the individual transistors can be measured.

The setup which is used to select a single transistor in the array for testing is shown in Figure 5.6. Transistor T1 is the transistor that is under test. To monitor the drain current of a single transistor at the shared output of the array, the corresponding row and column has to be enabled using the digital signals `col_en` and `row_en`. The gate voltage for T1 is generated per column by feeding the externally supplied reference current I_{ref} to the transistor T3, which is 16 times wider than T1. The current measured at I_{in} is expected to be 1/16 of I_{ref}. In the layout, T3 is split into 16 fingers that are distributed homogeneously over the column, every 8 instances of T1 a finger of T3 is placed. Since one finger of T3 has the same dimensions as one instance of T1 the layout is completely regular. Transistor T5 is used to multiplex I_{ref} to the single column that is activated. T4 is used to reliably pull down the gate voltage of the columns which are disabled.

To select a row in the array, the drain terminals of all current controlling transistors are connected to the output line using transistor T2. Since only the transistor

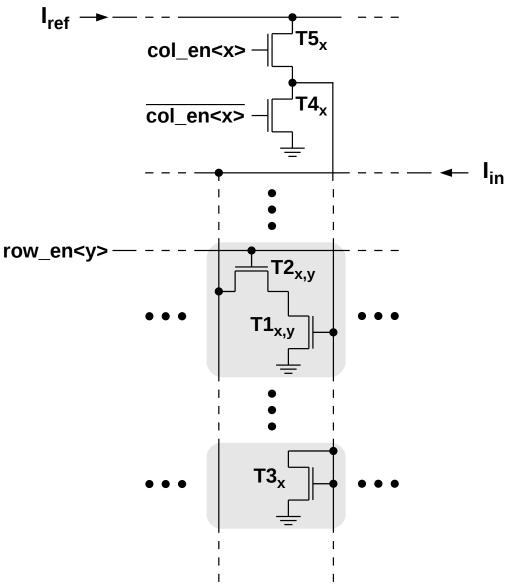


Figure 5.6: Schematic of the test circuits in the transistor array used for measuring device mismatch. Transistor T1_{xy} is one instance of the transistors under test. The current flowing at I_{in} is the drain current of a single transistor in the array, selected using the column and row enable signals.

Chip	$I_{ref}/16$	STD(I_{in})	STD(I_{in})/I_{in}	ΔI_{in}
	[μA]	[μA]	[%]	[nA]
1b.2	0.25	0.021	8.6	0.23
	3.00	0.162	5.4	1.16
	10.00	0.352	3.5	0.63
1b.3	0.25	0.020	8.0	0.27
	3.00	0.164	5.5	0.31
	10.00	0.351	3.5	0.71
1b.4	0.25	0.022	8.9	0.16
	3.00	0.164	5.5	0.35
	10.00	0.354	3.5	0.78
Simulation	0.25	0.036	14.2	-
	3.00	0.202	6.7	-
	10.00	0.384	3.8	-

Table 5.1: Results for the measurement of the mismatch test array on three different chips and for three different reference currents. STD(I_{in}) is the standard deviation of the drain currents of the individual transistors in the array. ΔI_{in} is the average measurement error of the individual drain currents, obtained from 8 independent repetitions. Additionally the result of a Monte Carlo simulation for the transistors is presented.

T1 in the selected column is supplied with a gate potential other than 0 V, only one transistor in the array is producing the current I_{in}.

Experimental Results

The current I_{in} is measured using the Sourcemeter applying a voltage of 1.2 V at the chips input pad. Table 5.1 shows the variation measured between the drain currents of the individual transistors for different reference currents. The average measurement error for the drain current of an individual transistor, obtained from 8 repetitions, is significantly below the variation observed between the individual transistors. Additionally the results of Monte Carlo simulations for the transistor mismatch are presented. For all combinations of chips and reference currents, the observed variation is below the variation predicted by Monte Carlo simulations. As an example, Figure 5.7 shows the measured drain currents for all transistors in chip 1.b3 for a nominal current of 3 μA. Additional measurements and a quantitative analysis of the spatial distribution of the device mismatch, can be found in [Kunz 2012].

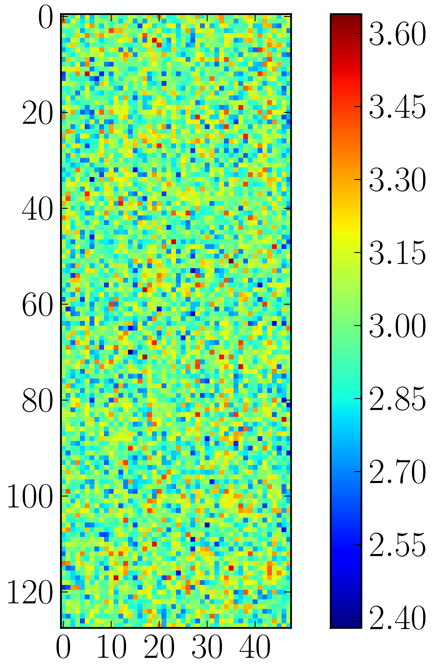


Figure 5.7: Drain current of the 48 × 128 transistors included in chip 1.b3 for a nominal current of $3\,\mu$A. The average current over all elements is $I_{in} = 2.986\,\mu$A, the standard deviation observed is $\Delta I_{in} = 0.164\,\mu$A.

5.4 Integrated Power Management

The BrainScaleS Hardware System uses wafer-scale integration to allow for a high communication bandwidth between the neural circuits at a comparatively low power consumption. However, wafer-scale integration also involves several technological challenges. One of the most important aspects is defect management. In modern semiconductor process technologies the chips on a wafer are tested after fabrication for electrical defects. Chips which fail in any of the tests are marked and not further processed after cutting the wafer. Manufacturers typically consider the yield of working chips that is achieved a corporate secret and do not publish any corresponding data. However, it can be expected that a full wafer will always contain some faulty devices or connections. This means that it is necessary to test all devices on a wafer before it is taken into operation and that the system needs to be able to cope with defects. The nature of possible defects covers a broad range from faulty transistors to open connections or shorts in the metalization layers.

In neuromorphic hardware, many errors only have local consequences, rendering single components useless, but do not disturb the operation of the system. This is especially true if individual neuron or synapse circuits are affected. Due to the very flexible routing options for neural events on the HICANN chip, it is possible to facilitate network operation bypassing faulty components. This requires that the software algorithm which maps the neural network of interest to the individual hardware neurons and synapses is aware of all defects present in the system. Information on the mapping process can be found in Brüderle et al. [2011] and HBP SP9 partners [2014, Chapter 14.2].

The consequences of a defect are more severe if a central component of a HICANN chip is affected. Large fractions of the digital control logic are essential for successful operation, any defect in these parts renders a full HICANN chip useless. But even in this case the working chips on the wafer can be used to facilitate network operation.

Most problematic however are short circuits between any of the supply voltages and ground or between different supply voltage nets. In this case operation of intact components of the system can only be achieved by disabling the power supply for the part of the wafer which is affected by such a defect. In the current BrainScaleS Hardware System discrete switch transistors placed on the main PCB are used for power management, see Section 5.4.1. In the following, the option of integrating power switches directly into the chips of future systems is discussed.

5.4.1 Power Switches in the Current System

In the BrainScaleS Hardware System the power supplies can be disconnected with a per reticle granularity using discrete switching transistors located on the main PCB. For detailed information see Güttler [2010]. These switches are used during the power up process of a wafer module to enable the different supply voltages of an individual reticle in the correct order. The reticles themselves are activated one after another to avoid excessively high current transients on the power supply rails.

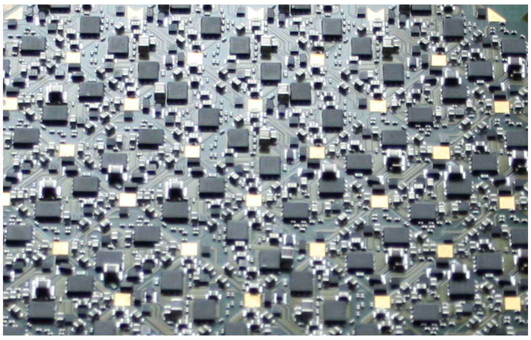


Figure 5.8: Photograph of the power switch transistors on the main PCB of a wafer module. Photograph by Maurice Güttler.

Furthermore the switches are also used to permanently disable the power supply of reticles in which serious manufacturing errors have been detected. Because of the high number of 12 different supply voltage nets for the HICANN chip, the same number of transistors is required for every single one of the 48 reticles per wafer. The high number of switch transistors leads, in conjunction with the required blocking capacitance for the individual supply voltages, to a very dense arrangement of discrete devices on the main PCB, see Figure 5.8. The high number of active devices on the top side of the board causes reliability problems in the assembly process of the main PCB [Husmann 2013]. Despite the high number of switch transistors on the main PCB, a single short in the power supply of one HICANN chip renders a full reticle, containing eight HICANN chips, useless. Furthermore, especially if the disabled reticle is close to the center of the wafer, a significant fraction of the L1 bus fabric, which transmits neural events across the wafer, is not available. The general architecture of the L1 bus is very flexible and most of the working neural components on the wafer can be connected to each other, but the overall bandwidth in the L1 system is reduced.

5.4.2 Integrated Power Switches for Future Systems

In a future hardware system, built in the 65 nm process, the problem of switching the power supply voltages becomes even more critical than in the current system. The number of neural circuits per area is expected to increase and the absolute area per wafer doubles, due to the larger wafer diameter of 30 cm. Using a set of discrete transistors per reticle, as in the old system, would lead to a higher density and a higher absolute number of devices on the main PCB. As mentioned in Section 5.4.1, manufacturing and assembly of the main PCB is already a challenge in the existing system.

In order to simplify the main PCB, the option of using on-chip transistors to switch the power supply nets is evaluated. This would also allow to switch the supply voltages with a single chip granularity, minimizing the impact of single damaged chips on the system. The most critical aspect of this approach is that a short circuit at the input side of any of the power switches renders a whole wafer useless. An option to increase the reliability is to avoid using structures of minimum width or with minimum spacings in the corresponding layouts. Furthermore the question which fraction of the chip area will be covered by the power management structures is crucial. The R_{on} of the integrated transistors is proportional to the area it covers. A trade off has to be made between increasing the power efficiency and blocking of valuable chip area. Assuming that manufacturing errors which affect robust layout structures, e.g. induced by dust particles, are distributed homogeneously over the total area, the probability of a switch transistor being affected depends on their relative size. In the following, the question how much area needs to be invested in order to allow for on-chip power switches for the 1.2 V supplies is discussed.

Minimizing the R_{on} of a 1.2 V Transistor

The largest fraction of the power consumption of a future system will be drawn from the 1.2 V core supply nets. Minimizing the ohmic resistance of the enabled power switch R_{on} is important to achieve a high power efficiency at an acceptable area consumption. The drain current of a MOS transistor in the ohmic region can be described by Equation 5.1.

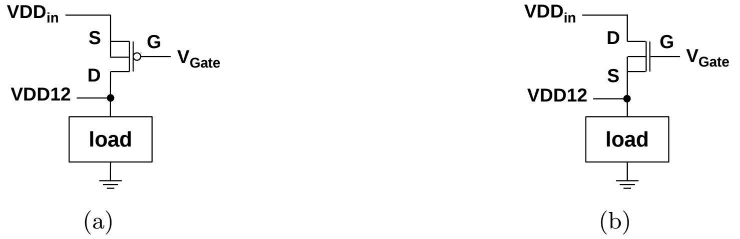
$$I_D = K' \cdot \frac{w}{l} \cdot ((V_{GS} - V_{th})V_{DS} - \frac{V_{DS}^2}{2}) \tag{5.1}$$

K' is a process technology-dependent parameter, w is the width and l the length of the transistor's gate. For small values of V_{DS}, which are typical for transistors used as switches, the quadratic term can be neglected. The drain current can be assumed to be proportional to the drain-source voltage. As a result, an effective ohmic resistance for the transistor switch in the on state can be estimated by:

$$R_{on} = \frac{1}{K'} \cdot \frac{l}{w} \cdot \frac{1}{(V_{GS} - V_{th})} \tag{5.2}$$

The resistance of the switch transistor can be tuned to the desired value by adjusting the factor l/w. As mentioned before, robustness is an important issue for the power management system, therefore a value larger than the possible minimum should be chosen for the length l. Adjusting the width w allows for tuning of the transistor's resistance in order to match the desired trade-off between efficiency and area consumption. To minimize R_{on}, the low-threshold-voltage option offered for the core transistors can be used. That way the effective gate voltage $V_{GS,eff} = V_{GS} - V_{th}$ is increased.

The most convenient way to switch the power supply of a load is using a PMOS transistor, with its source connected to the supply and the drain connected to the load, see Figure 5.9(a). To enable the switch, the gate of the PMOS transistor needs to be connected to ground. In this configuration, the gate-source voltage is as large



(a) (b)

Figure 5.9: Using a PMOS or an NMOS transistor as a switch for the 1.2 V supply voltage of a load. In (a) V_{Gate} needs to be 0 in order to turn on the transistor, in (b) a V_{Gate} of VDD12 + $V_{GSmax} = 2.4$ V is required to achieve the minimum R_{on} for the transistor.

as possible within the supply voltage boundaries. However the intrinsic conductivity of a NMOS transistor is significantly larger than for a PMOS transistor since the mobility of electrons in silicon is higher than the mobility of holes. In Equation 5.2 this has to be accounted for by using different constants K'_n or K'_p.

If an NMOS transistor is used instead, a gate potential higher than the supply voltage is required to obtain a significant gate-source voltage at the transistor, see Figure 5.9(b). For the lowest possible R_{on}, the gate potential has to be V_{source} plus the maximum gate-source voltage allowed. Since in the on state the source of the transistor is at the supply voltage level, this means that twice the supply voltage is required at the gate. Circuits that can be used to double the supply voltage and use it to drive the gate of an NMOS transistor can be found in literature, e.g. Liu et al. [2011]. The simplest and most robust solution in the system described here is to use thick-oxide transistors and the 2.5 V supply for controlling a 1.2 V NMOS switch. The only challenge is to ensure that the gate-source voltage V_{GS} of the switch transistor never exceeds the nominal maximum of 1.2 V during the power-up and the power-down process.

The option of using an on-chip power switch for the 1.2 V supply, based on a low-threshold voltage NMOS core transistor, has been evaluated in detail. A transistor which allows for a load current of up to 1.2 A at a voltage drop of about 100 mV maximum is tested. The length of the transistor is chosen to be 80 nm to increase the robustness of the design. The total width of the transistor needs to be 6 mm to match the specification given above in typical process corner. In the layout, the gate of the transistor is split into 400 parallel connected fingers of 15 μm width each. The total area covered by the layout of the transistor is 1880 μm^2.

The gate potential for the transistor is generated by the circuit presented in Figure 5.10, it was designed by J. Schemmel. Transistor T1 is the large low threshold transistor with a standard gate oxide, which is used to connect the 1.2 V supply voltage from VDD12_{in} to $\text{VDD12}_{\text{out}}$. The crucial devices for generating the gate potential of T1 are the transistors T2 and T3 which have identical dimensions.

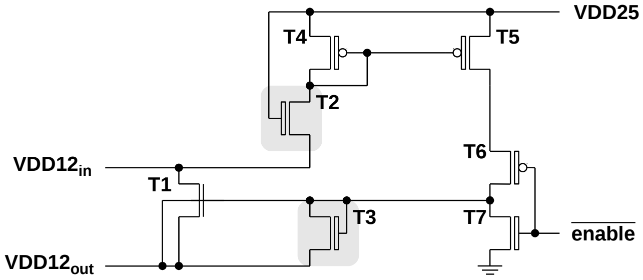


Figure 5.10: Circuit that generates the gate voltage for the power switch transistor T1. See text for further explanation.

Since gate and source of T2 are both connected to supply voltages, $V_{GS,T2}$ always equals VDD25 – VDD12$_{in}$ = 1.3 V. The resulting drain current is mirrored using T4 and T5 and fed to the diode-connected transistor T3. Since T3 has the same dimensions as T2 and their drain-source current is identical, $V_{GS,T3}$ equals $V_{GS,T2}$, regardless of the absolute potential at the source of T3. The transistors T6 and T7 are used to trigger the switching process. As long as $\overline{\texttt{enable}}$ is high, the current generated by T5 is stopped at T6 and the gate of T1 is discharged by T7. To switch the transistor into the `on` state, $\overline{\texttt{enable}}$ is set to zero. Now the current generated by T5 is able to charge the gate of the switch transistor until it reaches a level of 1.3 V over its source level.

The described circuit has been integrated into the first prototype chip and tested, the results are presented in Section 5.4.3.

Switching the 2.5 V Supply Voltage

A future neuromorphic system built in the TSMC 65 nm process will also require a 2.5 V supply in case circuits based on thick-oxide transistors are used. Implementing analog circuits based on thick-oxide transistors is an interesting option which is discussed in Chapter 5.5 and Chapter 8.

Using a thick-oxide PMOS transistor is again the simplest solution to switch the 2.5 V supply. The gate potentials required to enable or disable the switch are within the regular supply voltage range. Using an NMOS transistor offers a lower R_{on} per area. In this case an absolute gate potential of 5 V is required to enable the transistor. Therefore a circuit controlling the gate voltage needs to be supplied by a voltage of at least this level. Such a high voltage can be generated internally from the 2.5 V supply by using a charge pump circuit, see e.g. Liu et al. [2011]. However, integrating a charge pump circuit in every chip increases the number of devices involved in power management. Consequently the risk of a random error affecting the power management of a wafer increases. As an alternative, an external 5 V supply, distributed over the main PCB, could be used.

Another option is to use a switch built from core transistors. Circuits that allow

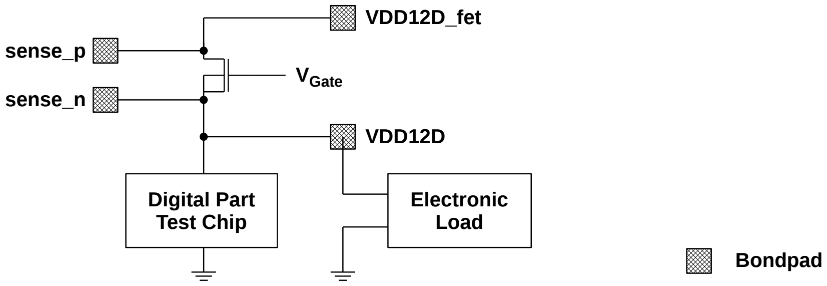


Figure 5.11: Diagram showing the integration of the power switch transistor into the digital 1.2 V power supply net of the test chip. The bond pads used for testing the transistor are shown. The gate voltage V_{Gate} is generated by the circuit shown in Figure 5.10.

to switch voltages which exceed the maximum V_{GS} of the single transistors involved can be found in literature. It is possible that such a circuit, built from stacked core transistors, offers superior efficiency for a 2.5 V supply voltage switch. compared to an implementation based on a single thick-oxide transistor.

However, it is not yet clear if a significant amount of current will be drawn from the 2.5 V supply in a future system. Therefore the options for implementing 2.5 V switches have not been evaluated in detail.

5.4.3 Experimental Results for the 1.2 V Switch Circuit

The 1.2 V switch transistor and the circuit which controls its gate voltage have been implemented into the first prototype chip and tested. Figure 5.11 gives an overview how the switch transistor is integrated into the digital 1.2 V power supply net of the test chip. The drain of the transistor is connected to the bond pad `VDD12D_fet`, here the supply voltage of the test chip is connected in regular operation. The source of the transistor is connected to the internal 1.2 V digital power net of the test chip, which is accessible via the bond pad `VDD12D`. This additional pad serves two different purposes. First of all it can be used to directly supply the digital 1.2 V net of the test chip in case the power switch transistor is not working correctly. It can further be used to connect an external load to the net, allowing for testing of the transistor at high currents. The digital part of the test chip, an early implementation of the Plasticity Processor [Friedmann 2013, Chapter 5.2], does not consume more than about 40 mA at the maximum possible clock frequency. Both bond pads, `VDD12D_fet` and `VDD12D`, are wider than regular. Six bond wires can be connected to each. This is necessary to allow for large test currents. To measure the voltage drop over the transistor, omitting the inevitable voltage drop over the bond wires and the on-chip routing, sense wires are connected to source and drain of the transistor and accessible via separate bond pads.

The voltage drop over the transistor is measured in dependence of the load cur-

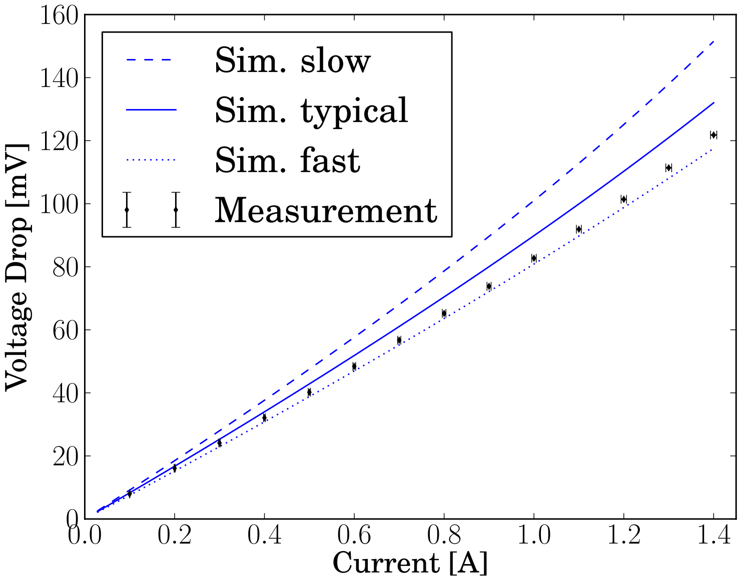


Figure 5.12: Measurement results for the voltage drop over the power switch transistor at different load currents. Additionally corresponding simulation results are shown for different process corners. The measurement results are close to the simulation for the fast corner. The error on the load current is estimated to be 0.5 %, the error on the voltage drop, estimated from three repetitions of the measurement, is below 1 mV. The corresponding error bars are too small to be visible.

rent, the results are presented in Figure 5.12. For this measurement, a supply voltage of 1.3 V is connected to the drain of the transistor via pad `VDD12_fet`. The current consumption of the digital part of the test chip is reduced to significantly less than 1 mA by disabling its clock signal. The current through the transistor is controlled by a programmable electronic load which is connected between the `VDD12D` pad and ground. The resulting voltage drop over the transistor is measured by a multimeter connected to the sense pads. The measurement error has been estimated based on three repetitions of the full measurement procedure, it is below 1 mV. In order to determine the accuracy of the programmable load, a separate measurement is used to compare the currents generated against measurements taken by a multimeter. As a result, the average error of the programmable load is estimated to be about 0.5 %. At a load of 1 A, the resistance of the switch transistor is $R_{on} = 85\,\mathrm{m}\Omega \pm 1\,\mathrm{m}\Omega$.

Figure 5.12 additionally shows simulation results for the voltage drop over the transistor for different process corners. For all simulations the temperature has been set to 27 °C. The measurement results are comparable to the simulation results for the fast corner.

5.4.4 Estimating the Area Consumption for on-chip Switches

Based on the measurement results presented, the area required by on-chip switches for the 1.2 V supply of a future system can be estimated. Since dimensions and current consumption of a future hardware system, built in the 65 nm process technology, are unknown, the corresponding numbers of the BrainScaleS Hardware System are used. In the existing system, most of the HICANN chips' power consumption is drawn at the two main 1.8 V supplies, one is used for digital and one for analog circuits. Per supply voltage and reticle, one individual switch transistor is located at the main PCB. The transistors used for the 1.8 V supplies provide an R_{on} of 5 mΩ. The transistors and the main PCB are designed for a maximum current consumption of 8 A for the digital supply and 8 A for the analog supply of every reticle, each of which contains 8 HICANN chips. These numbers lead to a maximum power loss of 0.64 W at the switches of a single reticle.

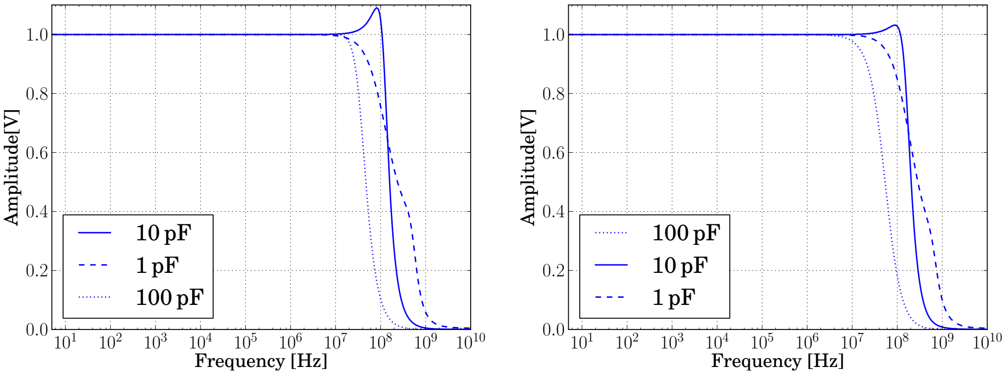
In order to achieve an effective R_{on} in range of 5 mΩ a transistor 4.25 times wider than the implementation used in the test chip needs to be integrated into each of the 8 chips within a reticle. The resulting area consumption per chip, scaling linearly with the width of the transistor, is about 4000 μm^2. Accounting for two such transistors, one for the analog and one for the digital supply, the area consumption per chip is 8000 μm^2, which corresponds to approximately 0.016 % of the total chip area.

Simulations suggest that under worst case conditions - a chip in slow process corner operating at a temperature of 75 °C - the resistance of the transistor is 1.4 times higher than the measurement result used in this estimation. This has to be considered when actually designing a system using on-chip power switches.

5.5 Transferring an Existing Operational Amplifier Design

The analog circuits used in the BrainScaleS Hardware System have been developed and improved over several years. To benefit from the circuits already available, the option of porting an existing circuit implemented in the 180 nm process directly to the 65 nm process using thick-oxide transistors is explored. As an example of reasonable complexity an operational amplifier designed by S. Millner [Millner 2008] was chosen. The amplifier uses a parallel folded-cascode input stage with g_m compensation and a class-AB output stage. It provides a rail-to-rail input voltage range as well as rail-to-rail output voltages.

In a first step the schematic has been rebuilt using exactly the same transistor dimensions. The minimum length and width of the thick-oxide transistors in the new process are larger than the respective dimensions for the standard transistors in the 180 nm process. As a consequence, small transistors in the design needed to be enlarged. These transistors have been scaled such that the w/l ratio remains equal to the one in the original design. The asymmetry between N- and PMOS transistors, caused by the different mobilities of electrons and holes in the semiconductor, is less pronounced in the new process. Wherever the ratio between



(a)

(b)

Figure 5.13: Simulation comparing the AC characteristics of the original amplifier, designed for the 180 nm process (a) and the transfered amplifier based on the thick-oxide transistors of the 65 nm process (b).

NMOS and PMOS conductivity is relevant, the width of the PMOS transistors was decreased. The effect of all changes to transistor dimensions has been monitored using simulations. Finally some transistor dimensions have been adjusted based on comparison between simulations of the original and the new circuit.

In simulations, the amplifier built from thick-oxide transistors in the 65 nm process achieves comparable performance as the original design. As one example, Figure 5.13 shows a simulation of the AC characteristics for the original amplifier design compared to the corresponding behavior of the transfered circuit. In both cases the input is a sine signal with an amplitude of 1 V, symmetric to half of the supply voltage. The bias currents are set to 3 μA. As suggested by S. Millner, the stability of the amplifier circuits is improved by using a 50 Ω serial resistor at the output.

The layout of the amplifier in the new process covers an area of 480 μm^2. Despite the fact that the transistor dimensions are basically the same as in the original design, it was possible to reduce the area consumption by 36 % compared to the 180 nm version.

In summary, comparable performance between original and transfered design was achieved in simulations, the transfered design is based on an identical schematic, only the dimensions of the transistors have been adapted. These results indicate that, using thick-oxide transistors, it is possible to transfer existing analog circuits to the new process with limited effort. In Chapter 8 the transfer of the neuron implementation used in the HICANN chip to the 65 nm process is described.

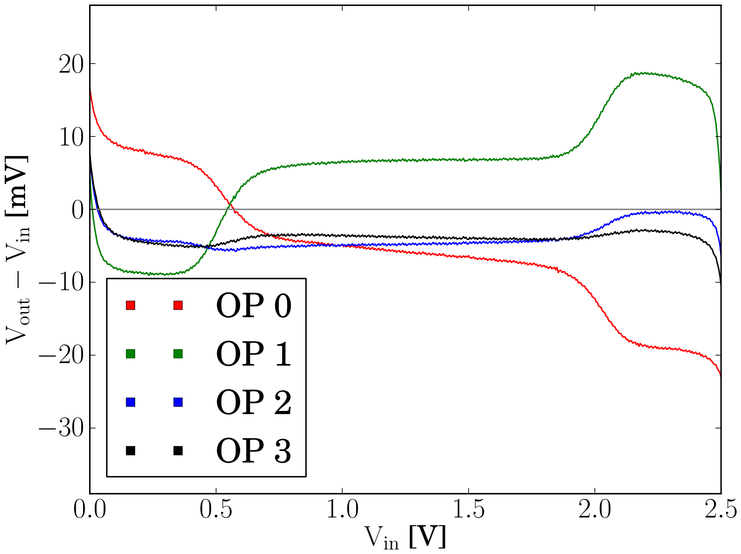


Figure 5.14: DC characteristics of four operational amplifiers from chip 1a.4, operating as unity gain buffers. The difference between output and input voltage is plotted over the input voltage. A significant voltage dependent offset is visible, especially for the amplifiers OP0 and OP1.

Experimental Results

Six instances of the amplifier have been integrated into the first prototype chip and can be tested. A typical application for the amplifier is to use it as a unity gain buffer, driving signals connected to bond pads of chip. Therefore results testing it in the respective configuration are presented. Figure 5.14 shows the DC characteristics of the four individual amplifiers of chip 1a.4 which can be tested in unity gain configuration. The input voltage V_{in} is swept over the full supply range and the resulting output voltage is recorded. In the Figure the difference $V_{in} - V_{out}$ is plotted.

All amplifiers are affected by an offset voltage, a common problem in design and implementation of operational amplifiers, see e.g. Gray and Meyer [1982]. For OP0 and OP1 the offset is not constant but shows a significant input voltage dependency. This is a result of the fact that the amplifier design features two parallel pairs of input transistors, one built from NMOS and one from PMOS transistors, each of which can be differently affected from device mismatch and dominates the characteristics of the amplifier for different input voltage ranges. See Millner [2008] for detailed information on the design of the amplifier.

These results presented in Figure 5.14 are relevant because the amplifier has been used as output buffer for several circuits implemented in the second prototype chip.

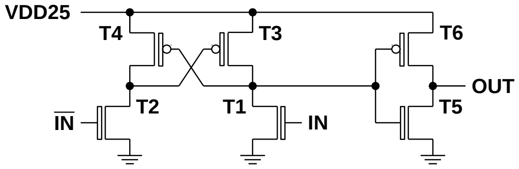


Figure 5.15: Schematic of the circuit used to shift digital signals from the 1.2 V domain to the 2.5 V domain.

5.6 Digital Level Shifters

In mixed-signal chips the analog parts need to be interfaced with the digital circuits. If thick-oxide transistors are used to implement analog circuits, it is necessary to shift the voltage levels of the signals interfacing the different parts. In most cases, digital signals from the 1.2 V domain are used to control or configure the analog circuits operating at 2.5 V. The standard level-shifting circuit shown in Figure 5.15 can be used for this task. This design, using minimum size thick-oxide NMOS transistors and PMOS transistors of minimum width and twice the minimum length, has been used in both prototype chips. Including a realistic load, simulations suggest that rise and fall time of the output signal are below 100 ps in the typical process corner. The layout of the circuit covers about 11 μm^2 of chip area. In case only small currents in the analog part need to be controlled by digital signals, it is sufficient to connect the 1.2 V signal directly to the gate of a thick-oxide NMOS transistor without using a level-shifting circuit.

To realize a digital output of a 2.5 V supplied analog circuit it is sufficient to use a single digital inverter built from thick-oxide transistors but supplied with only 1.2 V. This circuit covers about 3 μm^2 of area and offers rise and fall times of less than 100 ps with a realistic capacitive load at the output.

6 Analog Parameter Memory System

This chapter presents a system designed to provide large numbers of programmable voltage and current sources for analog neuromorphic hardware chips. These are required to allow for flexible circuits which can be adapted to cover a larger range of biological models. Another important aspect is the compensation for variation between the individual circuits caused by device mismatch.

As described in Chapter 2, each HICANN chip requires more than 14 k programmable parameters. A full wafer of the BrainScaleS Hardware System contains more than 4.8 M individually adjustable current and voltage sources, more than 90 % of which are used to configure the neuron circuits. A comparable neuromorphic hardware system, designed in the 65 nm process and based on 30 cm wafers, will have an even higher demand for programmable parameters. These numbers emphasize the importance of realizing a power- and area-efficient parameter system.

In mixed-signal projects, programmable voltage and current sources are typically realized using digital-to-analog converters (DACs). A wide variety of different implementations for DAC circuits can be found in literature. However, a different strategy can be pursued if power and area efficiency are important and the parameters are supposed to be constant during an individual experiment. A limited number of DACs can be used to sequentially generate the required analog values and store the results in multiple analog memory cells. This scheme is applied in the HICANN chip. Cells based on floating gate devices are used to store the voltages and currents, see Section 2.1.3. For the TSMC 65 nm process technology a scalable parameter storage system based on capacitive memory cells has been developed and tested in a prototype chip. The general design as well as performance measurements will be presented in the following.

6.1 The Concept of Capacitive Memory

The basic idea of storing voltages and currents using capacitors is shown in Figure 6.1(a). To store a voltage, a capacitor is charged to the target voltage and then disconnected using a switch. The same principle is used in digital random access memory (DRAM). However, in that application only binary states are stored on the capacitor. Following the same concept for analog precision applications imposes various additional challenges.

Currents can be stored using the same concept as for voltages, see Figure 6.1(b). A reference current is drawn from a diode connected transistor, the resulting gate

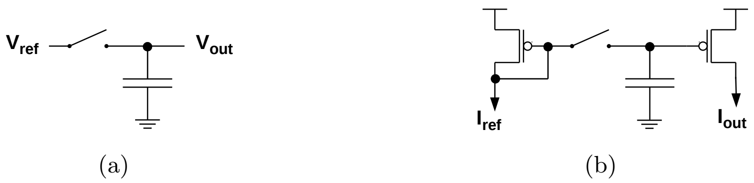


Figure 6.1: Simplified schematic showing the basic concept of capacitive storage cells. (a) shows a cell storing voltages, (b) a corresponding design for currents.

voltage is stored on the capacitor. The output current is generated using an additional transistor, identical to the one used for programming. Its gate is connected to the stored voltage, as a result the output current resembles the reference current used during the programming step.

However, due to leakage currents discharging the capacitor, the stored voltage drifts over time. Therefore the stored values need to be refreshed periodically in any application where a stable output is required over extended periods of time.

There are also implementations of capacitive memory cells that offer significantly longer storage times, see e.g. Wojtyna [2008]. In these systems active circuits are used to minimize or compensate for the leakage currents. Considering the high cell density required and the tight power envelope in large-scale neuromorphic systems, inserting additional transistor circuits consuming constant bias currents is not an option. To achieve a high power efficiency, the leakage currents discharging the capacitor need to be reduced by passive measures. Low output drift rates are essential to reduce the rate of power-consuming refresh cycles while still providing sufficient accuracy.

6.2 The First Generation of Capacitive Memory Cells

A first generation of capacitive storage cells has been developed in the 65 nm process technology and tested in the first prototype chip. The focus during development was on minimizing the leakage currents discharging the storage capacitors. These depend on the implementation of the switch which disconnects the capacitor. But also the technology used to realize the on-chip capacitance can affect the storage time of the circuit. Three versions of current storage cells, which use different implementations of the switch, are presented.

6.2.1 Choosing the Capacitor Technology

To minimize the leakage discharging the storage capacitor not only the contribution of the switch has to be considered. Depending on the technology for implementation of the capacitor, leakage in the device itself might contribute to the discharging process. Typically the best option to implement capacitors into an integrated circuit is to use a metal-insulator-metal capacitor, abbreviated as MIM Cap. These are simple plate capacitors formed by two layers in the metalization stack. To achieve a reasonable capacitance per area, a special process step is used to significantly reduce the distance between the plates of the capacitor compared to the regular vertical spacing between the metal layers. The dielectric material used to separate the plates of the capacitor typically provides a very good isolation. For detailed information on MIM Caps see Allen and Holberg [2011, p. 46ff].

However, in the TSMC 65 nm process technology the option of using MIM Caps is only given for the metal layers `M7` and `M8`. Implementing large numbers of capacitors blocks a significant amount of area in these metal layers. According to current plans for the architecture of a 65 nm neuromorphic hardware system, `M8` is required for long-distance routing of synaptic events and must not be blocked by the parameter storage cells. Therefore the capacitor has to be implemented as a transistor gate capacitance. The gate oxide of the 1.2 V standard transistors is very thin, leading to the largest capacitance per area in the given process. Simulations however indicate that the tunneling current through the gate oxide of the standard transistors is large enough to significantly reduce the storage times. As a result, the best option for implementation of the capacitors of the analog storage cells is to use the gates of thick-oxide transistors. In the simulation models of these devices no tunneling through the gate oxide is included. Measurements with floating gate devices based on the thick-oxide transistors of the process, presented in Hüll [2014], indicate that the tunneling current for these transistors is indeed negligible for typical applications.

6.2.2 Optimizing the Switch

There are several options for realization of the switch shown in Figure 6.1 using transistors. The goal is to minimize the current flowing through the switch in the `off` state, using only passive measures. Typically, transmission gates built from an NMOS and a PMOS transistor in parallel are used to switch voltages, providing a low `on` resistance over the full supply voltage range. However the parallel connection of two transistors inevitably leads to a higher leakage than a single transistor switch. When a single transistor is used as a switch, the range within which good conductivity in the `on` state can be provided is reduced. The gate-source voltage of the single transistor needs to be larger than its threshold voltage in order to achieve a low resistance. As a consequence, an NMOS switch can only be operated for voltages from 0 to $VDD-V_{th}$, assuming that the gate of the transistor is connected to VDD in the `on` state. For PMOS switches the situation is reversed, it can operate for voltages in the range from VDD down to $|V_{th}|$. Using single-transistor switches therefore reduces the range of the voltages that can be stored on the capacitor. This restriction has a negative effect on the dynamic range for voltage storing cells. Nevertheless, in order to optimize the storage time this option is pursued in the following.

The leakage current through a transistor in the `off` state can be minimized by applying a negative gate-source voltage. In a system without a negative supply voltage, this can be realized for an NMOS transistor if its source potential is at a voltage larger than 0. This is typically true for the stored voltage V_{stored}, connected to one side of the switch. During the `off` state, the other side of the switch can be connected to the supply voltage using additional transistors. As the stored voltage is always below VDD, the corresponding contact of the transistor is considered to be its source. Consequentially the gate-source voltage of the switch transistor is $-V_{stored}$. The same concept can be used for a PMOS switch. In the `off` state the gate is at VDD, the voltage at its source needs to be below that level. The strategy of using a negative gate-source voltage to minimize the leakage through a transistor in a single supply system is known as "analog T-switch" concept, described in Ishida et al. [2006].

However, the leakage current through the channel of the switch is not the only effect changing the amount of charge stored on the capacitor. The reversed bias current through the bulk diodes of the switch transistors is an aspect that needs to be considered. Due to their opposing bulk potentials, a combination of NMOS and PMOS transistors can be used in order to cancel the diode currents. However, precise balancing of the currents can only be achieved at a certain voltage level. Compensating the effect over a large dynamic range is only possible by actively controlling the bulk potentials in dependence of the stored voltage. This option has not been pursued as implementing switch transistors in individually isolated substrate wells comes at a high price in terms of area consumption. Furthermore, active circuits, increasing the overall power consumption, are required to drive the bulk potential to the optimum level.

As mentioned in Chapter 6.2.1, simulations indicate that the impact of electrons tunneling through the gate oxide of the standard transistors is significant. The resulting current scales linearly with the area of the gate, therefore the effect is most prominent in the capacitor. However, the full circuit needs to be built from thick-oxide transistors in order to achieve the maximum storage time possible.

A general problem in switched capacitor circuits is charge injection [Laker and Sansen 1994, 700ff]. When changing the state of the switch, the gate potential of the transistor(s) involved changes from 0 to VDD or vice versa. Due to the parasitic gate-drain and gate-source capacitances in MOS transistors, this transition has an effect on the circuits connected to the switch. If the switch is used to disconnect a small capacitor, the voltage on the capacitor can be changed significantly by the switching process. In transmission gates, built from an NMOS and a PMOS transistor, the effect is partially canceled by the fact that the gates of the two transistors are driven by inverted signals. If only a single transistor is used as a switch, the gate capacitance of an additional transistor can be used to compensate for charge injection [Eichenberger and Guggenbuhl 1990]. Its gate needs to be driven by the inverted switch signal and its drain and source need to be connected to the storage capacitor. The ratio between the size of the actual switch and the size of the compensation transistor is critical for the success of this approach.

6.2.3 The Implemented Storage Cells

Considering the aspects mentioned above, capacitive current storage cells have been designed and tested in the first prototype chip. The circuits are built entirely from thick-oxide transistors. To find the best possible switch configuration, three different types of cells have been implemented. The schematics of these cells are presented in Figure 6.2.

The basic concept for all types of cells is the same, only the transistor implementation of the switch is different. The transistors associated with the switch are highlighted by the gray boxes. The cells are designed to sink a programmable current using an NMOS output transistor. During programming, the write enable signal `W` is high, T1 and the switch is conducting. Transistors T2 and T7 operate as current mirror for the reference current I_{ref} and capacitor C1 is charged to the according gate voltage. To store the value, `W` returns to low, so that T1 and the switch are no longer conducting. As long as C1 holds the correct voltage, the current I_{out} equals the reference current used for programming the cell. The capacitance of C1 is in the range of 95 fF. As it is realized as a gate capacitance the exact value depends on the stored voltage.

The cells are designed to cover a dynamic range of 0 to 10 μA. The transistors T2 and T7 have identical dimensions of w $= 0.8\,\mu$m and l $= 1.6\,\mu$ m. According to simulations, mirroring currents in the range from 0 to 10 μA leads to a voltage range of 0 to 1.02 V at the gates of the transistors.

A general limitation of the described architecture is that the time required to charge the capacitor as well as the gate-source capacitances of the transistors T2 and T7 to the correct voltage depends on the reference current. Especially for a low current of e.g. 5 nA it takes more than 5 μs to initially charge the capacitor in the cell. For a refresh process, where the difference between the stored voltage on the capacitor and the generated voltage is minimal, the problem is less severe. In this case only the gate-source capacitance of T2 has to be charged. Transistor T1, present in all cell types, is required to connect multiple cells to the same reference current source, the digital write enable signal `W` is generated individually for each cell.

In the cells of Type 1, shown in Figure 6.2(a), T5 is used as the switch transistor disconnecting the capacitor. To reduce the leakage current through T5 when storing a value, T4 is used to pull node n0 to 2.5 V, in order to create a negative gate-source voltage for T5. In this situation the terminal of T5 which is connected to the stored voltage has to be considered the transistors source contact. Since the voltage stored is always larger than 0, therefore a negative gate-source voltage for T5 is ensured. The transistor T3 is used to prevent the gate capacitance of T2 to be also charged to 2.5 V. Minimizing the capacitance of n0 helps to reduce the time required to write the cell. T6 is used to counterbalance the charge injection effect of T5.

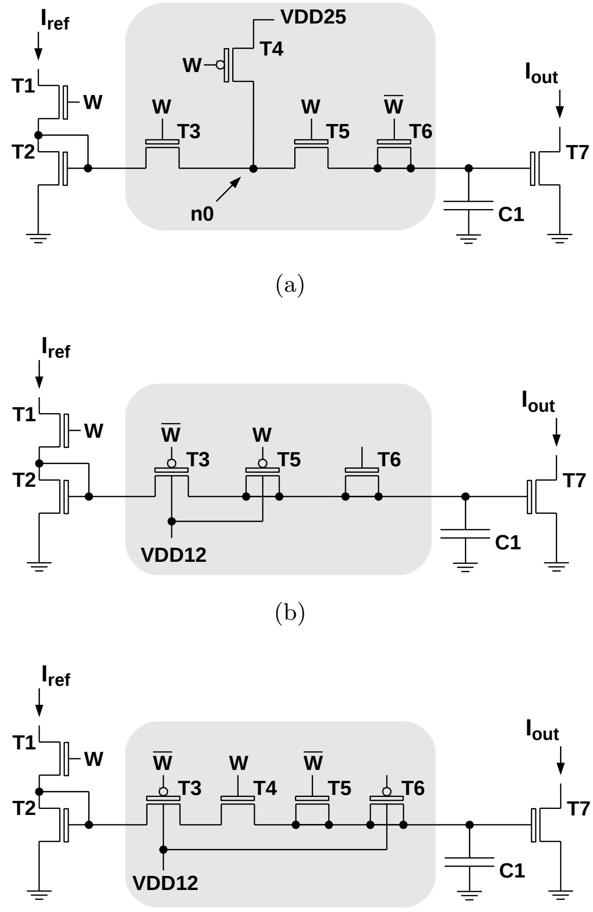
In the cells of Type 2, shown in Figure 6.2(b), PMOS transistor T3 is used to disconnect the capacitor. Using a PMOS transistor as a switch results in a low conductivity for low voltages. Since the cell uses an NMOS output transistor, low voltages have to be stored in order to realize small output currents. This obvious limitation could be solved by replacing transistors T2 and T7 by PMOS transistors and thereby changing the characteristics of the cells output from being a current sink to sourcing a current. However, to simplify the test setup and to allow for direct comparison of the results from different cell types, the architecture is not changed. The problem of low conductivity of the PMOS transistor for low voltages can be avoided by using sufficiently long programming times when testing the cells. The gate of T2 discharges to 0 as soon as I_{ref} stops, therefore a negative source-gate voltage, minimizing the leakage current through T3, is generated. Other than for the cells of Type 1, no additional transistor is required. T5 is used to counterbalance the charge injected by T3 during the switching process. The bulk of both PMOS transistors T3 and T5 is connected to the 1.2 V supply rather than the 2.5 V supply. This is done to reduce the leakage current caused by the reversed substrate diodes at the source and drain contacts of the transistors. As a consequence the stored voltage is not allowed to exceed a level of 1.2 V, otherwise the substrate diodes would be biased in forward direction. Since the stored voltage is below 1.02 V over the full specified dynamic range, this is no limitation. T6 is an NMOS transistor added to counterbalance the leakage current of the bulk diodes of the PMOS transistors by the leakage of its bulk diodes.

In the cells of Type 3, shown in Figure 6.2(c), a serial connection of the PMOS transistor T3 and the NMOS transistor T4 is used to disconnect the capacitor. The leakage current is limited by the transistor that has the lower conductivity in this setup. Since T3 is a PMOS transistor, the performance of the cells for low voltages on the capacitor is again restricted. In this case changing the type of the mirroring transistors T2 and T7 does not help as the NMOS transistor T4 provides only limited conductivity for voltages close to the supply level. Nevertheless this setup was included for testing. T5 is used to compensate for charge injection during the switching process, T6 balances the leakage current caused by the bulk diodes of the other transistors involved.

6.2.4 Implementation Details

In the first prototype chip the three cell types described before are implemented. A block containing 8 instances from the cells of Type 1 are implemented, the same is true for the cells of Type 2. For Type 3 two blocks containing 8 cells each are integrated into the chip. The layout of one block covers an area of about 702 μm^2, the area consumption per cell is therefore 88 μm^2.

For each cell an individual write signal is generated in the digital part of the chip. A `select` bit can be activated for each of the 24 cells via the JTAG interface. The chip features a digital input pad, `cm_write`, which is used to control the timing of the write process. `AND` gates between the `select` bits and the global `cm_write` are used to generate an individual write signal for each cell. Levelshifter circuits are integrated into each block, increasing the level of the write signals to 2.5 V. The reference current I_{ref} used to program the cells is supplied by an external current



(a)

(b)

(c)

Figure 6.2: Schematics of current cells designed for the first prototype chip. (a): Type 1, (b) Type 2, (c) Type 3. The basic design is identical, however the implementation of the switch, marked by the gray box, is different for the three individual designs.

source via an analog input pad. The output of all cells, the drain contact of T7, can be individually connected to a analog bond pad using a configurable multiplexer.

6.2.5 Experimental Results

Only a short overview of the measurement results is presented, as the cells implemented in the first prototype chip suffer from various fundamental limitations.

The performance of the capacitive storage cells is measured for the test chips 1a.0, 1a.3 and 1a.4. The Sourcemeter is used to produce a voltage of 2.5 V at its output and to measure the current flowing into the selected storage cell. The device also features digital general purpose IOs, one of which is used to generate the `cm_write` signal controlling the write process. All functions of the Sourcemeter can be controlled by a PC using its RS-232 interface. However, latency and jitter for this interface are in range of milliseconds. Therefore the option of transmitting scripts written in Lua [Lua 2014] via RS-232 to the internal memory of the Sourcemeter is used. These are executed by a microcontroller integrated into the device, offering precise control over the timing of the measurements. According to the documentation, a uncertainty below 50 μs can be expected for the duration of programmed time intervals.

In order to measure the drift of the output current over time the following protocol has been used: A single storage cell is selected and the `cm_write` signal is activated in order to charge the cell. Next, the initial output current I_{init} is measured while `cm_write` is still active. Ideally I_{init} should be identical to the value of I_{ref}, however due to device mismatch of the transistors T2 and T7 some deviation is observed. Then `cm_write` is deactivated and immediately a sequence of 20 measurements with an interval of 0.1 s in between the individual measurements is started. The results of the measurements are written to an internal buffer and read back later via RS-232 for further processing. The full sequence is repeated 20 times for each cell.

Figure 6.3 shows some examples for results of such a measurement. The output current for six cells from chip 1a.0, two of each type have been randomly selected, are shown. The reference current level was set to 10 μA, marked by the horizontal line in the figure. The effect of device mismatch in the current mirror has been compensated by multiplying each data point by the factor I_{ref}/I_{init}. The error bars denote the standard deviation obtained from 20 repetitions. For each of the cells the drift of the output over time is visible. However, the first sample of all measurement is significantly different from the reference current I_{ref}. It seems the output currents are shifted towards lower output currents when `cm_write` is disabled.

Output Current Offset

The offset visible in Figure 6.3 is investigated in more detail. It is not a result of mismatch in the current mirror as this effect has been eliminated using I_{init}. The small statistical error of the measurements also suggests that it is caused by a

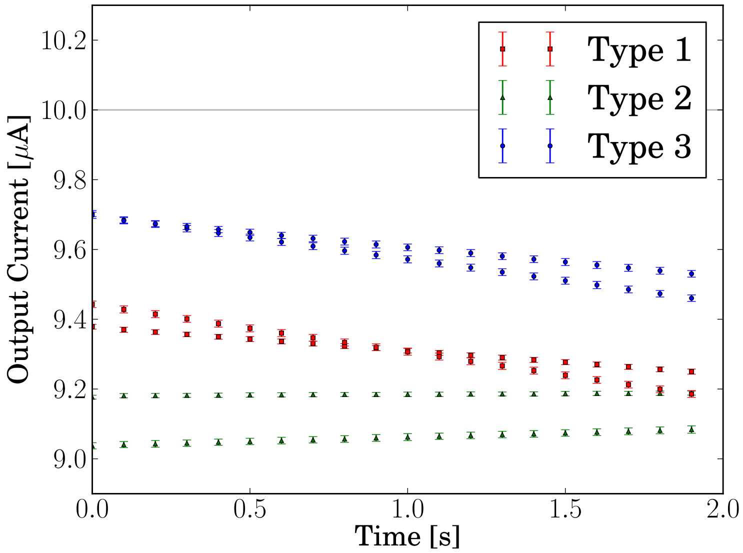


Figure 6.3: Output current over time for 6 randomly chosen cells from chip 1a.0. The impact of device mismatch has been compensated for by multiplying each data point by the factor I_{ref}/I_{init}. The error bars indicate the standard deviation obtained from 20 repetitions of the measurement.

systematic effect.

The measurement protocol described before has been used to characterize all cells on the three chips, each chip was tested at three different values for I_{ref}. For each trace a linear function $f(t) = m \cdot t + b$ is fitted to the data. The offset introduced by the end of the programming process is the value $f(t = 0) = b$. In Figure 6.4 histograms over the offsets observed for the different types of storage cells are shown. The value of the reference current used, $I_{ref} = 5\,\mu\text{A}$, is marked by a vertical line. In all cases the offset leads to a lower output current than expected. For all cell types a significant cell-to-cell variation of the offset can be observed.

The reason for this offset lies within the design of the cells. The `write` signal of the cell, abbreviated by `W` in the schematic, is used simultaneously to disable I_{ref} for the addressed cell as well as to disable the switch, disconnecting the storage capacitor from T2. As a result transistor T1 is limiting the reference current, leading to a reduced gate voltage at T2, before the storage capacitor is completely disconnected by the switch in the cells. Therefore the voltage stored on the capacitor is shifted towards lower values, leading to an output current smaller than I_{ref}. The large cell to cell variation of the effect can be explained by the threshold voltage variation of the switch transistors. In order to allow for precise programming of the cells, it needs to be ensured that the switch in the cell is in `off` state before I_{ref} is disabled. The offset caused by the problematic design of the cells is so large, that it masks any potential distortions caused by charge injection.

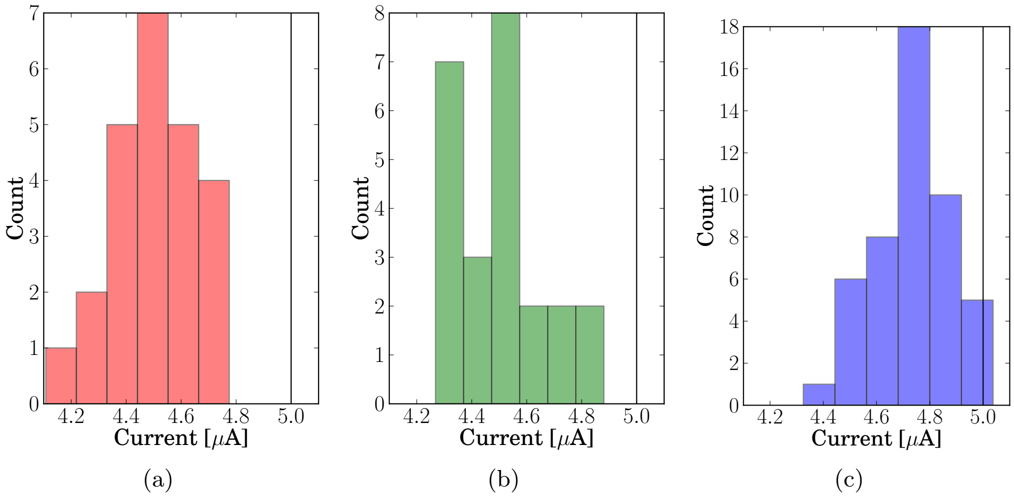


Figure 6.4: Output current offset for all capacitive memory cells available on the chips 1a.0, 1a.3 and 1a.4. A reference current of $I_{ref} = 5\,\mu$A, marked by the vertical line, was used. (a) shows the results for cells of Type 1, (b) for Type 2 and (c) for Type 3.

Output Current Drift

As mentioned before, the output current has been measured over time for each cell on the three test chips and a linear function has been fitted to the resulting data. Regardless of the offset observed, the slope obtained from the fit characterizes the drift of the output current over time. In Figure 6.5 the average drift rate, measured over all cells of the same type, is shown for different reference currents. The error bars indicate the standard deviation over the results from individual cells.

The cells of type 2 show the smallest absolute drift rate at reference currents of $5\,\mu$A and $10\,\mu$A. For the lowest value of $I_{ref} = 1\,\mu$A the drift rate is, relative to the results obtained from other cell types, rather large. Since the performance at low currents is critical, this behavior is considered non ideal. The variation of the drift rate between individual cells of the same type is significant. In case of the cells of type 2, even the direction of the drift is different for individual cells.

The cells of type 3 show a very low drift for small currents. However, the design of these cells is not considered to be usable in an actual parameter storage system. Using an NMOS and a PMOS transistor in series in the switch limits the dynamic range drastically in a system where short programing times are important.

The cells of type 1 show a small drift rate at low currents and the design is suitable for implementation in a parameter storage system. The drift at larger output currents is worse than for the other cell types.

However, the absolute drift rates measured for all of the cells are in a range which is considered to be acceptable for a parameter storage system. In a system offering

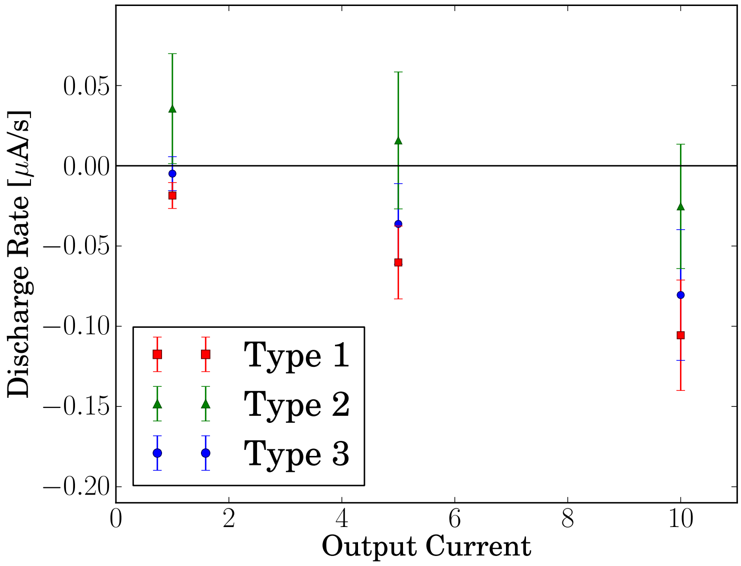


Figure 6.5: Drift rate of the output current, averaged over all cells of the same type, for all cells on the three test chips. The error bars denote the standard deviation over the individual cells.

a 10 bit resolution and a dynamic range from 0 to 10 μA, one LSB equals to about 10 nA. Drift rates better than 0.2 μA/s, as observed for all of the cells, correspond to about 0.02 LSB/ms. Assuming that the output drift between two refresh cycles should be less than 1 LSB, an update is required at least every 50 ms.

6.3 The Parameter Storage System

A full parameter storage system based on capacitive memory cells has been developed and implemented in a second prototype chip. It includes not only the storage cells for currents and voltages but also the infrastructure required to program and regularly refresh a large number of these cells. First the design of the voltage and current storing cells used in the system will be described. Based on the measurement results obtained from the first prototype chip, the design has been improved compared to the cells presented in Section 6.2.3. Later the architecture of the programming system, which performs regular refresh cycles for all cells, will be presented.

6.3.1 Design Goals for the Parameter Storage System

The general characteristics of the system should match the demands of a mixed signal neuromorphic hardware chip which has similar characteristics as the HICANN chip, cf. Chapter 2.1. The system is designed to provide a number of 24 programmable voltage and current sources for each of the 512 neuron circuits on the chip. The resolution for adjusting each parameter is 10 bit. However, it is assumed that for the operation of a neuromorphic hardware systems a precision of 8 bit is sufficient. The cells need to be built from thick-oxide transistors, in order to avoid a degradation of the storage time by tunneling through the gate-oxide of the transistors. The current cells are designed to cover a dynamic range from 0 to 2 μA. Reliable operation at small currents in the nano ampere range is important to allow for the power efficient realization of long time constants in the neuron circuits. The dynamic range of the voltage cells should be as large as possible but does not need to include the supply rails. A programming system, providing the reference voltages and currents to the cells, is required. Due to the expected drift of the values stored in the cells, periodic refresh cycles need to be performed for all cells during operation. In order to allow for interactive experiments, the possibility to reprogram parameters during an experiment is required. The area and particularly the power consumption of the system have to be optimized in order to make it suitable for integration in a wafer-scale neuromorphic hardware system.

6.3.2 Analog Part of the Current Cells

Based on the experimental results obtained from the first prototype chip, a new design for the current cells has been developed. The schematic is shown in Figure 6.6, the dimensions of all transistors are given in Table 6.1. There are several significant differences to the current storing cells presented in Section 6.2.3. The cell is designed as a current source, using a PMOS output transistor. Consequentially PMOS transistors are also used as switches since the range of stored voltages has to enclose the supply voltage in order to realize small output currents. Figure 6.7 shows a simulation of the output current of a current cell in dependence of the gate voltage which is stored on the capacitor. In order to realize the desired dynamic range of 0 to 2 μA, voltages in the range from 2.5 V down to approximately 0.8 V have to be stored. This range fits the operating range of the PMOS switch transistors. Other than in the first generation of current cells, the gate voltage for the output transistor, which is stored on the capacitor, is no longer generated locally. The cells are programmed using a reference voltage, instead of a reference current. Detailed information on the programming system can be found in Chapter 6.3.4.

The leakage through the switch is reduced the same way as in the cells of type 1 described in Section 6.2.3. The transistor T2 pulls the node connecting T1 and T3 to ground to ensure a negative gate-source voltage for T3. Figure 6.8 shows a simulation of the voltage stored on the capacitor drifting over time for two cells. For one the initial voltage on its capacitors was set to 0, for the other it was set to 2.5 V. It becomes evident that two effects are involved in changing the stored voltage. As only PMOS transistors, with their bulks being connected to the 2.5 V supply, are used, the reversed bias current of all substrate diodes involved are charging the capacitor. For low voltages on the capacitor this effect dominates, the stored

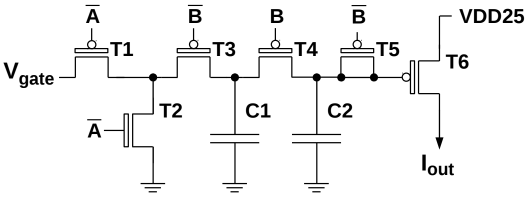


Figure 6.6: Schematic of the current cell developed for the capacitive parameter storage system. The cell operates as a current source.

Device	length $[\mu m]$	width $[\mu m]$
T1	0.32	0.8
T2	0.28	0.4
T3	0.32	0.8
T4	0.33	0.8
T5	0.32	0.4
T6	10.32	0.8
C1	5.16	1.8
C2	5.16	0.6

Table 6.1: Dimensions of the devices shown in the schematic of the current cell shown in Figure 6.6.

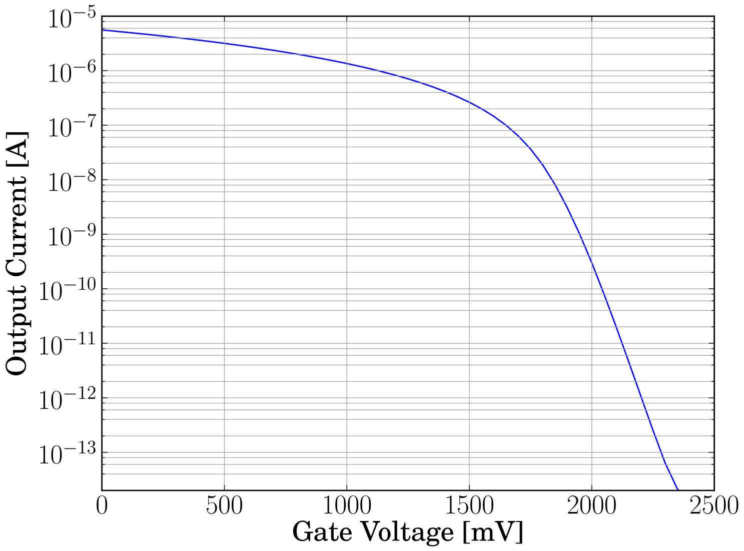


Figure 6.7: Simulation of the output current of a current cell in dependence of the gate voltage stored on the cell's capacitor. In order to allow for small currents, voltages close to the supply level are required.

voltage is rising. The leakage through the switch transistor T3 is discharging the capacitor, because the drain of T3 is pulled to ground by T2. This effect strongly depends on the gate-source voltage of T3, which depends on the stored voltage. As a consequence the stored voltage drops rapidly for high voltages. Equilibrium between both effects is observed at a level of 2.3 V, the cells are drifting towards this voltage. The output current for a cell in this state is, according to simulations, only 60 fA.

When testing the cells implemented in the first prototype chip, it was not possible to evaluate the effect of charge injection due to a fundamental mistake in the design. The circuit has been further optimized regarding this aspect using simulations. For the new version a scheme already used in the Facets Stage 1 Hardware System to isolate the distortions caused by the refresh process from the output has been adapted, see Ostendorf [2007]. Two storage capacitors C1 and C2 are used and connected sequentially by transistor T4. Both capacitors are implemented as gate capacitances of NMOS transistors. T4 is switched inverse to transistor T3, which connects the reference voltage to C1 during the refresh process. The output of the cell can not be disturbed by this process as T4 is not conducting. When the refresh process ends, T4 becomes conducting and the voltage in the cell is updated towards the voltage to which C1 was charged. T5 is used to compensate the charge injection effect from T4, preventing the output from being disturbed by the process. A drawback of this scheme is that the output of a cell can not be updated to a new target value in a single refresh process. Due to the sequential connection of

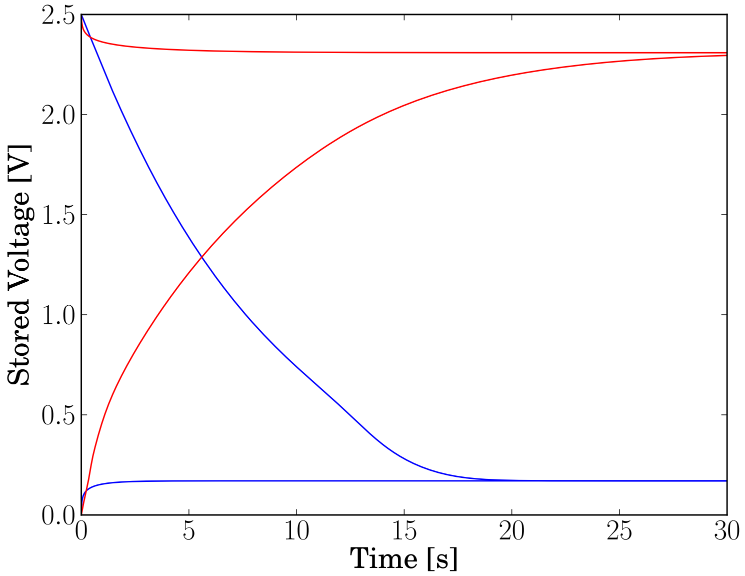


Figure 6.8: Drift of the voltage stored in the capacitive memory over time. In red the traces of the voltage on the capacitors of two current cells are shown. For one cell the voltage is initially set to a value of 0, for the other to 2.5 V. In blue the corresponding traces for the voltage stored in two voltage cells are shown.

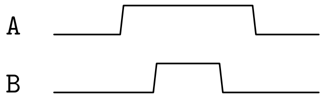


Figure 6.9: Relative timing of the signals A and B.

the two storage capacitors the voltage stored at C2 approaches its target value exponentially. This aspect is discussed in more detail in Section 6.3.5.

To allow for a higher analog precision of the programming process two control signals, termed A and B in the following, are used for the new cells. The cell is programmed by positive pulses with 2.5 V amplitude on these signals. The temporal sequence used is shown in Figure 6.9, the wider pulse of signal A embraces a shorter one on signal B.

When A turns high, the node connecting T1 and T3 is no longer pulled to ground as T2 is no longer conducting. Instead it is charged to the level of the reference voltage V_{Gate} through T1. Next, when the voltage has settled, T3 is activated by the rise of signal B, charging the first storage capacitor C1 to V_{Gate}. The end of the programming process is initiated by B turning back to 0, disconnecting C1 from the reference voltage. Simultaneously C1 and C2 are connected via T4. Finally the node between T1 and T3 is pulled down to ground again when A turns low.

6.3.3 Analog Part of the Voltage Cells

The voltage cells are based on the design of the current cells presented in Section 6.3.2. The schematic is shown in Figure 6.10, the dimensions of all transistors are given in Table 6.2. In this case the switches are based on NMOS transistors because the desired dynamic range of the output voltage is centered around 1 V. The upper boundary of the dynamic range is determined by the highest source voltage at which the switch transistors are conducting. Despite the different polarity of the switches, the structure of the circuit is identical to the design of the current cells. Since T3 is an NMOS transistor, T2 is used to pull the node connecting T1 and T3 to VDD in order to ensure a negative gate-source voltage. Figure 6.8 shows a simulation of the voltage stored on the capacitors drifting over time for two voltage cells. For one the initial voltage on its capacitors is set to 0, for the other it is set to 2.5 V. The two effects changing the voltage at the capacitor are the same as described for the current cells, however the polarities are reversed. The leakage current through the switch transistor T3 is charging the capacitor as the drain of T3 is pulled to the supply voltage. The reversed bias current trough the substrate diodes is discharging the capacitor since NMOS transistors are used in the cell. The equilibrium state of the stored voltage is at 170 mV. Below this level the leakage through T3 is rater high, resulting in very short storage times.

The output transistor, T6 in the schematic of the current cell, is not required in the voltage cells. In order to keep the layout of current and voltage cells as iden-

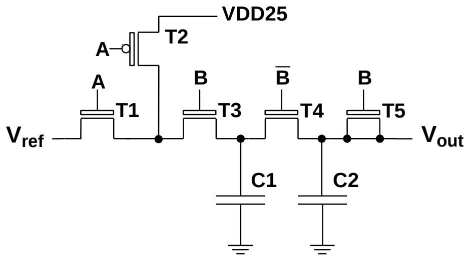


Figure 6.10: Schematic of the voltage cells used in the parameter storage system.

Device	length [μm]	width [μm]
T1	0.400	1.6
T2	0.280	0.4
T3	0.380	0.8
T4	0.415	0.8
T5	0.440	0.4
C1	5.160	1.8
C2	5.160	0.6
T6 (C2)	10.32	0.8

Table 6.2: Dimensions of the devices in the voltage cell shown in Figure 6.10.

tical as possible, it is nevertheless integrated and used as an additional capacitor, increasing the effective capacitance of C2. A larger value of C2 helps to reduce the noise level at the output of the cell. Decreasing the ratio between C1 and C2 however limits the maximum rate at which the output voltage can be changed, see Section 6.3.5. To save area and achieve faster reprogramming of the cells it is possible to omit T6 in future revisions of the voltage cells.

In the floating gate parameter storage system of the HICANN chip, a source follower is implemented within every cell, buffering the output voltage. A source follower is a very area efficient option to buffer a voltage, nevertheless it consumes additional current and area. Furthermore the threshold voltage variation affecting the individual source followers, leads to an increased variation of the output voltage from cell to cell. In the new system the output voltage of the cells is not buffered, this means that no current can be drawn from the outputs. A buffer has to be installed at the input of any circuit receiving a voltage from the parameter memory whenever a low impedance voltage source is required. The designer of these circuits can decide individually for every parameter which output impedance and dynamic range is required and chose a sufficient voltage buffering circuit. As mentioned before, the largest amount of the parameters is used to configure the neuron circuits. In the neuron circuit implemented in the BrainScaleS Hardware System, all programmable voltage parameters are connected to the very high impedance inputs of operational or transconductance amplifiers. This design is suited to operate with a parameter system providing only high impedance voltage outputs. However, as a consequence of the high output impedance of the voltage cells, the wires between parameter memory and receiving circuits are highly sensitive to crosstalk from other signals. This issue needs to be addressed in the layout, the voltage outputs have to be shielded properly, see Section 6.3.7.

6.3.4 The Programming System

The storage cells described above are able to store voltages and currents if they are provided with accurate reference voltages and periodically refreshed. In comparable systems, such as the one used in the Facets Stage 1 Hardware System, a single DAC is used to program all cells in the system sequentially. For large scale systems the DAC has to be very fast in order to provide a sufficient update rate for an individual cell. Furthermore fast digital logic circuits are necessary to read the digital target values from a digital memory and transfer them to the DAC. Another aspect, limiting the minimum time required to refresh a single cell, is the RC time constant of the wire connecting the DACs output with the storage cell. However, the impact of the RC delay can be reduced if the order in which the cells are updated is sorted according to their target values. In this case the voltage of the net distributing the DACs output to the cells only has to change incrementally. This scheme is also used in the Facets Stage 1 Hardware System to increase the programming speed.

A large number of different DAC architectures with different characteristics is available from literature. Optimizing speed, area and power consumption simultaneously is only possible up to a certain degree. Inspired by the sorted writing scheme in which the DAC only has to increase its output voltage incrementally, a system working in a more parallel fashion has been developed. The concept is first explained only for voltage storing cells, the corresponding circuits to program the current cells are described later. Instead of using a DAC generating the reference values for individual cells, a slowly and linearly increasing voltage ramp is generated and distributed to all the voltage cells in the array. Time is measured by a 10 bit counter which is started synchronously to the voltage ramp. The counter value is also distributed over the array to every cell. To program an individual cell to its target value, a refresh process has to be triggered at the point in time at which the voltage ramp has the correct analog value. To achieve this, each cell includes 10 bit of custom SRAM, holding the digital representation of its target value as well as asynchronous logic that compares the value stored in the internal SRAM to the current value of the counter. Whenever a match between counter and the internal SRAM is detected, the cell refreshes its internal analog value to the present value of the voltage ramp. The concept of addressing the individual cells in the array by the counter value can be compared to the concept of content-addressable memory which is used in digital high-speed search applications [Krikelis and Weems 1994]. Using the proposed scheme, all voltage cells in the array are refreshed within a single period of the ramp. The system is designed to operate with a ramp period in the order of 1 ms, the corresponding operating frequency for the 10 bit counter is about 1 MHz. Depending on the storage time provided by voltage and current cells, breaks can be added in between the single programming cycles in order to reduce the power consumption.

In the following first the individual components of the programming system are described, in Section 6.3.7 an overview for the full implementation of the system is presented.

Content Addressable Memory Implementation

Each cell contains a digital part that holds the 10 bit digital target value in custom SRAM memory and compares it asynchronously against the counter value. To compare each SRAM bit with the corresponding bit of the counter a logical equivalence operation has to be evaluated. This can be implemented as a standard CMOS `XNOR` gate, a solution that needs 14 transistors per bit. A much more area efficient way is to use transmission gate logic, this way the same functionality can be implemented using just 4 transistors. The results of the 10 single bit comparisons are evaluated by an standard CMOS `AND` gate providing 10 inputs to check for full equivalence between stored value and counter. A schematic of the comparison logic of a cell is shown in Figure 6.11. The two transistors T1 and T2 are working as an `XOR` gate. If the result of the operation is a logical `1`, the node connecting T1 and T2 can not reach the supply voltage level since both are NMOS transistors. The voltage rises only up to $VDD - V_{th}$ before the transistor pulling the node up enters the subthreshold regime and the further rising of the signal is slowed down drastically.

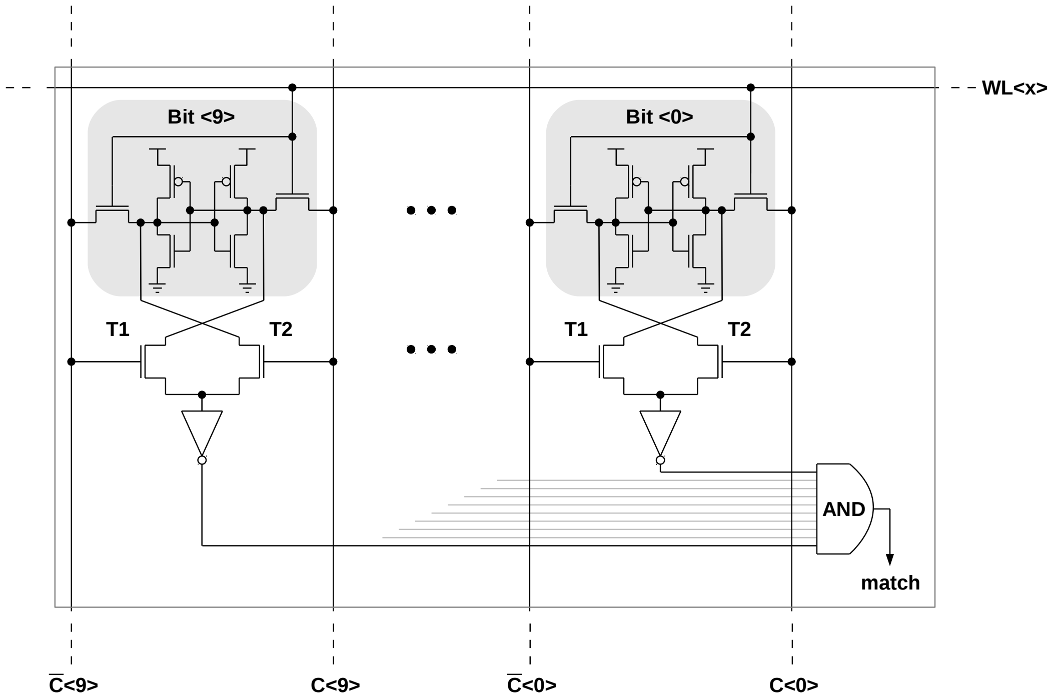


Figure 6.11: Schematic of the comparison logic implemented in each cell. The digital value stored in the SRAM bits is compared to the present value of the counter.

The following inverter is used to restore full voltage swing for the signal. However, it has to be ensured that the voltage at the input of the inverter reliably exceeds its trigger level. This was verified in a series of simulations covering the effects of different process corners as well as degraded supply voltage. More than 2000 runs of Monte Carlo simulations, accounting for transistor mismatch, did not lead to a single failure of the circuit.

A drawback of this implementation is that not only the counter signals $C_{<9:0>}$ are necessary but also the inverted signals $\overline{C}_{<9:0>}$ need to be routed to every cell. The inverted signals are generated along with the counter signals. Each pair of C_x and $\overline{C}_x$is routed in parallel to minimize the impact from the counter on the sensitive analog signals due to crosstalk. To save routing resources, the wires for the counter signals $C_x$and $\overline{C}_x$ are shared with the bitlines D_x and $\overline{D}_x$ of the corresponding SRAM bit. This is possible since the SRAM is not accessed while the counter is running. Changing the digital target values of cells while the parameter memory is in operation has to be done during the breaks in between individual ramp cycles when the counter is not active.

Experiments have shown that the implementation as shown in Figure 6.11 and used in the second prototype chip is not ideal regarding the dual use of bitline and

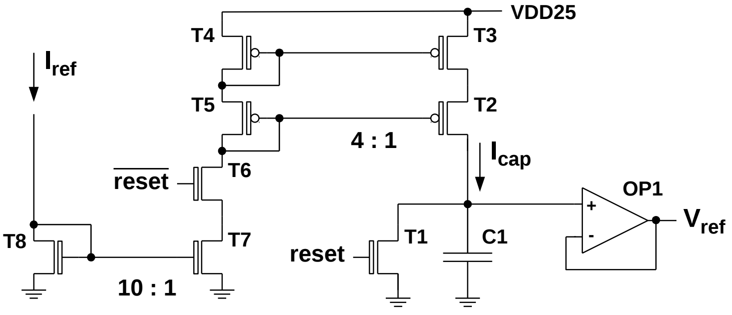


Figure 6.12: Schematic of the circuit which generates the linearly increasing voltage ramp V_{ref}.

counter signals. When reading back the data stored in the SRAM for debugging, the SRAM controller needs to precharge the bitlines: $\mathrm{D_x}$ and $\overline{\mathrm{D}}_\mathrm{x}$ are connected to the supply voltage. In this state the two signals are no longer inverted to each other, instead the transistors T1 and T2 in the comparison gate are both conducting simultaneously, shortening the internal state of the SRAM. In regular operation of the parameter storage system no read operations on the SRAM are required. However, any attempt to read back SRAM data destroys the content of the memory. A simple option how to avoid this problem in future revisions is presented in Section 6.5.1.

Generating the Reference for Voltage Cells

The linear voltage ramp used to program the voltage cells is generated by charging a capacitor with a constant current. The schematic of the *ramp_gen* circuit is shown in Figure 6.12. When the counter reaches its maximum value the reset signal is activated, switching off the current and shortening the capacitor voltage to ground. During the break between consecutive ramp cycles the reset signal is kept active.

The capacitor is implemented as a MIM Cap with a capacitance of 820 fF, the constant current charging it needs to be 1.64 nA to charge it up to 2 V within a timespan of 1 ms. To achieve good linearity the current source charging the capacitor needs to have a high output resistance, thus the current needs to be independent of the voltage at the capacitor. A two stage current mirror, build from the PMOS transistors T2 to T5, is used at this point, see e.g. Sansen [2006, p. 95]. Figure 6.13 shows the output current of the mirror over the voltage on the capacitor. For voltages larger than 1.95 V at the output, the transistors involved drop out of the saturation regime and the current is no longer constant. For comparison, the output current of a standard single stage current mirror is shown. The gates of the single stage mirror have twice the length of the ones used in the double stage mirror in order to compare circuits covering the same amount of chip area. The output resistance of the double stage mirror is significantly larger. The mirror is

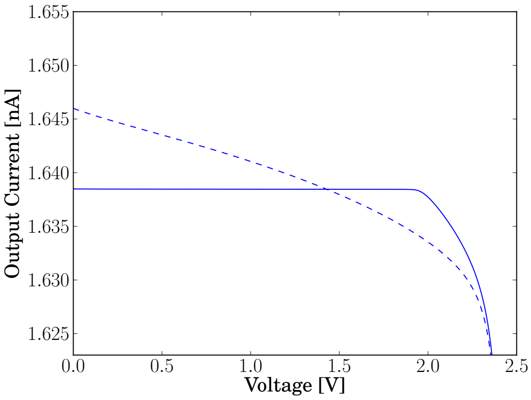


Figure 6.13: The output current of the current mirror which generates I_{cap} over the voltage at C1 (solid). For comparison the output characteristics of a single stage current mirror covering the same amount of chip area is plotted (dashed).

also used, in conjunction with the second one build from the NMOS transistors T7 and T8, to divide the externally supplied bias current by a factor of 40. The transistors in the current mirrors are subject to device mismatch, the same is true for the capacitor C1. Therefore the reference current I_{ref} needs to be calibrated for each individual instance of the *ramp_gen* circuit. Typically only one *ramp_gen* circuit is required per chip. The voltage at the capacitor V_{ref} is buffered using an operational amplifier and distributed to all voltage storage cells.

Generating the Reference for Current Cells

The linear voltage ramp used to program the voltage cells could be used to program the current cells as well. But in this case the relation between the digital code programmed to the cell and its output current is not linear. Furthermore global process variations, changing the characteristics of the output transistors, would directly affect the voltage to current conversion in the cells. Therefore the linearly increasing voltage V_{ref} is fed to a standard voltage to current converting circuit, generating a linearly increasing current. The schematic of the *VI_conv* circuit is shown in Figure 6.14. The OP1 is generating the gate voltage for T1 such that the current through R1 is $I_{R1} = V_{ref} \cdot R1$. The gate voltage that is required for T1 to produce the linearly increasing current ramp is distributed to the current storing cells as the reference voltage V_{gate}. To reduce the area consumed by R1 the transistor T1 is 32 times wider than the output transistors in the cells, the length is identical. The current through R1 ranges therefore from 0 to 64 μA. The value of R1 is programmable in a range from 8 kΩ to 48 kΩ. It is implemented as a string of 6 resistor elements with a value of 8 kΩ, each node between the single elements can be connected to ground using NMOS transistors. If non of the transistors is

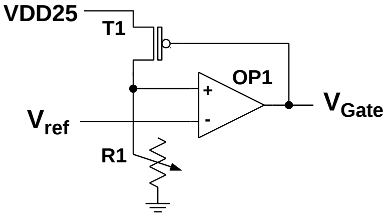


Figure 6.14: Schematic of the circuit which converts the voltage ramp V_{ref} to a linearly increasing drain current in transistor T1. It is used to generate the programming voltage V_{Gate} for the current cells.

active R1 is completely disconnected from ground and no current is flowing in the circuit. The default setting for R1 is 32 kΩ, leading to an output range of 0 to 2 μA for the current storage cells, when V_{ref} ranges from 0 to 2 V.

The option of changing the value of R1 was introduced to test different dynamic ranges for the output of the current cells. Another aspect is that the circuit described can become unstable, depending on the bias of OP1 and its capacitive load. The load depends on the parasitic capacitance of the network of wires distributing V_{gate} over the array of memory cells. Stability has been verified in simulations accounting for the parasitic capacitance and covering different process corners. However, reducing the value of R1 is a fall back solution to operate the circuit in case it is not stable within the planned operating range.

Refreshing a Cell

The refresh process for the individual storage cells is described in Section 6.3.2. Two digital signals, termed A and B, needs to be activated in correct sequence. During the operation of the system these signals need to be generated within every cell whenever a match between the internally stored digital value and the counter signals is detected. To generate this sequence of the signals without implementing analog delay circuits an additional clock signal is used. Its frequency is by a factor of two higher than the frequency of the counter and it is distributed to all cells. The timing of all digital signals which are involved in an update process is shown in Figure 6.16(a). When the match is detected, signal A is immediately activated. The signal B is activated a quarter of a counter period later, and stays high for half a counter period. Synchronously with the end of the counter period A returns back to the low state. This scheme is chosen to avoid distortions on the analog signals by crosstalk from the digital counter signals. The storage capacitor is connected to the reference voltage only while B is active, during this interval the counter signals are static.

In the second prototype chip the circuit presented in Figure 6.15, termed *pulse_gen*, is implemented in each storage cell to generate the signals A and B. During tests of the prototype chip it was discovered that the timing of the signals generated by the

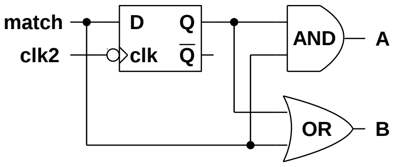


Figure 6.15: Schematic of the circuit which generates the signals A and B.

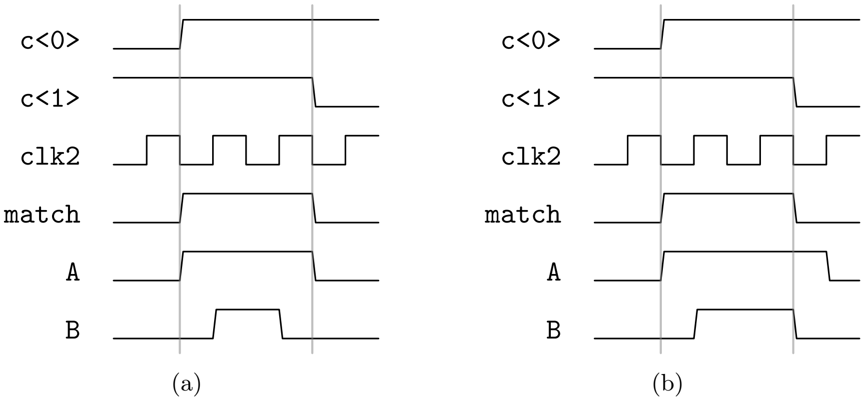


Figure 6.16: Timing of the signals involved in the programming cycle of a cell. The vertical lines mark the timespan in which the value stored in the internal SRAM and the counter value are identical. (a) shows the correct timing of the signals. (b) shows the incorrect timing generated by the logic implemented in the prototype chip. The signals A and B are both active a quarter of a counter period longer than intended.

pulse_gen circuit is not correct. The timing actually generated is shown in Figure 6.16(b), the signals A and B return to zero a quarter of a counter period later than intended. As a consequence, B is turning low simultaneously with the transition of the counter signals. Under certain conditions, this mistake has severe consequences for the analog accuracy of the cells. A more detailed discussion on this issue is given in Section 6.4.2. A simplified circuit which generates the correct timing for signals A and B is presented in Section 6.5.2.

Since the analog part of the cells is entirely built from the thick-oxide transistors, the level of the logic signals A and B needs to be shifted to the 2.5 V domain. For this purpose two instances of the levelshifter circuit described in Chapter 5.6 are implemented in each cell.

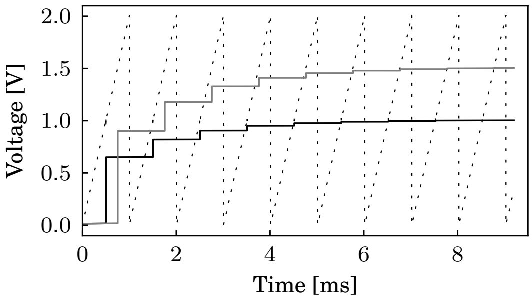


Figure 6.17: Simulation of the refresh process for two voltage cells. One is programmed to the digital code 511 (black) and one to 767 (grey). V_{ref} is also shown (dotted). Taken from Hock et al. [2013].

6.3.5 Estimating the Setup Time

As mentioned in Section 2.1.3, it can happen that the time required for initial programming of the floating gate-based parameter storage cell in the BrainScaleS Hardware System limits the maximum rate at which independent experiments can be executed. Therefore the setup time for a wafer-scale system using the capacitive parameter storage system is estimated. In the new system the setup time is determined by two aspects. In a first step all the SRAM cells in the digital part of the parameter storage cells have to be programmed. Assuming that the same number of parameters as in the current wafer-scale system is used, the total amount of configuration data which needs to be transmitted is 4.76 M × 10 bit = 5.95 MB. Again referring to the specification of the BrainScaleS Hardware System, the available bandwidth for the digital data links connecting the wafer module to the host PC is assumed to be 1 GB/s [Müller 2014], resulting in a time consumption of about 6 ms for the transmission of the data.

In a second step the counter is started, triggering the refresh processes which change the analog output of the cells. Due to the internal structure of the cells, using two capacitors which are connected in series by a switch, the output of a cell can not reach a new target value within a single refresh cycle. In the following the process of the output value approaching its target value is described in more detail. Figure 6.17 shows a circuit simulation of the refresh process for two voltage cells. Starting from $V_{out} = 0$ one in programmed to a digital code of `511`, the other to `767`.

Estimating the Setup Time for a Voltage Cell

In the following, the number of refresh cycles required by the voltage cells to reach an output voltage which deviates less than 1 LSB from the target value is estimated. In Figure 6.18 a simplified schematic of a voltage cell is shown. The two switches S1 and S2 are operated by the non overlapping signals Φ_1 and Φ_2. Whenever Φ_1 is

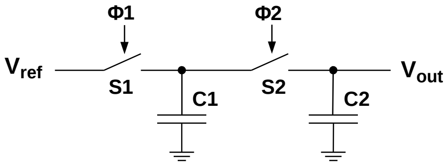


Figure 6.18: Simplified schematic of a voltage cell.

active and S1 is closed, the amount of charge on C1 is:

$$Q_1 = C_1 \cdot V_{ref} \tag{6.1}$$

The charge on C2 depends on the number of cycles n which have already been applied:

$$Q_2(n) = C_2 \cdot V_{out}(n) \tag{6.2}$$

Whenever Φ_2 is active and S2 is closed, the voltage between both capacitors is equalized. After n cycles, the output voltage $V_{out}(n)$ can be written as:

$$V_{out}(n) = \frac{Q_1 + Q_2(n)}{C_1 + C_2} = \frac{C_1 \cdot V_{ref} + C_2 \cdot V_{out}(n-1)}{C_1 + C_2} \tag{6.3}$$

This allows to calculate the change of V_{out} for each refresh cycle:

$$V_{out}(n) - V_{out}(n-1) = \frac{C_2}{C_1 + C_2}(V_{out}(n-1) - V_{out}(n-2)) \tag{6.4}$$

Assuming $V_{out}(0) = 0$, the output voltage after n cycles can be written as the sum over the individual steps:

$$V_{out}(n) = \frac{C_1 \cdot V_{ref}}{C_1 + C_2} + \frac{C_1 \cdot V_{ref} \cdot C_2}{(C_1 + C_2)^2} + \frac{C_1 \cdot V_{ref} \cdot C_2^2}{(C_1 + C_2)^3} + ... \tag{6.5}$$

$$= \frac{C_1 \cdot V_{ref}}{C_1 + C_2} \sum_{i=0}^{n-1} \left(\frac{C_2}{C_1 + C_2}\right)^i \tag{6.6}$$

This a geometric series. In general the partial sums of a geometric series are given by the following expression, provided that $r \neq 1$.

$$a \cdot \sum_{i=0}^{i=n} r^i = a \cdot \frac{1 - r^n}{1 - r} \tag{6.7}$$

In our example $r < 1$ therefore the series converges and the limit is V_{ref}:

$$a \cdot \sum_{i=0}^{i=\infty} r^i = a \cdot \frac{1}{1-r} = \frac{C_1 \cdot V_{ref}}{C_1 + C_2} \cdot \frac{1}{1 - \frac{C_2}{C_1 + C_2}} = V_{ref} \tag{6.8}$$

The fact that V_{out} converges against V_{ref} is expected from the structure of the circuit. However, combining Equation 6.7 and Equation 6.8 allows to calculate the distance between the output voltage V_{out} of a cell and its target value V_{ref}:

$$V_{ref} - V_{out}(n) = \left(\frac{C_2}{C_1 + C_2}\right)^n \cdot V_{ref} \tag{6.9}$$

This result can be used to estimate the number of cycles required by the cell to minimize the difference between actual output voltage and target value to less than 1 LSB. However, the calculation assumes ideal switches and does not account for the limited conductivity of their transistor implementation. Due to the slow operation of the refresh system this is not considered to be a limitation during normal operation. The switch S1 is operated by signal B which is active for about 0.5 μs for each refresh cycle. Accordingly signal $\overline{\text{B}}$, operating switch S2, is constantly active during the remaining period of the counter cycle. Another deviation is introduced by the fact that the assumption of B and $\overline{\text{B}}$ being non-overlapping is not correct. Instead $\overline{\text{B}}$ is slightly delayed against B. However, any time interval in which both switches are simultaneously active speeds up the process of the cell approaching its target value.

The capacitors C1 and C2 are implemented as gate capacitances of MOS transistors. Therefore their capacitance shows some voltage dependency, this is another effect neglected in the calculation leading to Equation 6.9. For the examples presented in the following, the capacitance of the devices in the middle of the supply voltage range is used. In case of the voltage cells, the capacitance of C1 is 57 fF. The effective size of the second capacitor is given by the capacitance of C2, 66 fF in case of the voltage cells, and the capacitance connected to the output of the cell. Assuming that an instance of the amplifier described in Chapter 5.5 is connected to the output of the cell, the additional capacitance is in a range of 42 fF. However, for the current implementation of the neuron circuit, the capacitance of all voltage parameter inputs is significantly smaller. The input capacitance of the rather large amplifier is used here in order to estimate a worst case scenario. These numbers lead to a effective capacitance ratio of $C_2/(C_1 + C_2) = 0.62$. For a transition from 0 to the maximum output value of 1023, the difference between actual output voltage and target voltage can be expected to be less than 1 LSB after 15 refresh cycles.

For the worst case scenario of a large load capacitance and reprogramming the output voltage from the minimum to the maximum possible value, a time interval of about 15 ms is required in order to obtain a stable output for the voltage cells. Adding the 6 ms required by the transmission of the configuration data, the total setup time for the system is estimated to be better than 21 ms in a typical scenario.

Estimating the Setup Time for the Current Cells

In case of the current cells, the voltage stored on C2 approaches its target value in the same manner as described for the voltage cells. However the effective value of C2 does not depend on the capacitance at the output of the cells. It is given by the capacitance of C2 and the gate capacitance of transistor T6, see Figure 6.6. The effective capacitance ration is therefore smaller than for the voltage cells, $C_2/(C_1 + C_2) = 0.54$. As a result the voltage at the gate of the output transistor approaches its target value faster than the output of the voltage cells. The overall setup time of the system is not limited by the current cells.

6.3.6 Estimating the Output Resistance of the Voltage Cells

Evaluating the charging processes of the capacitors during a refresh operation allows to estimate the output resistance of the voltage cells. The arrangement of the switches S1 and S2 as well as capacitor C1, shown in Figure 6.18, corresponds to a typical implementation of a resistor element in switched-capacitor circuits [Caves et al. 1977]. The effective resistance of such a circuit is given by

$$R_{eff} = \frac{1}{C \cdot f} \tag{6.10}$$

where C is the capacitance and f is the frequency at which the switches are operated by non-overlapping signals. For the voltage cells, operated at a refresh frequency of 1.2 kHz and with a capacitance of 57 fF for C1, this results in a value of $R_{out} \approx 15\,\mathrm{G\Omega}$. The value given is only an estimation. The signals B and $\overline{\texttt{B}}$, operating the switches, are slightly delayed against each other instead of being non overlapping. Any time interval in which both switches are simultaneously active leads to a reduction in the output resistance. However, accurate output voltages can only be expected if the circuit connected to the cell provides an significantly larger input resistance. As a result the cells can only be used to control the gate potentials of MOS transistors, for any other application a buffer has to be inserted.

6.3.7 Implementation of the Parameter Storage System

The components for a new parameter storage system presented so far are designed to be used in a future neuromorphic chip. It is assumed that the basic architecture of this chip is organized in a similar fashion as in the HICANN chip, see Chapter 2.1. The parameter system needs to provide programmable voltage and current sources to the neuron circuits which are arranged in a horizontal row. Consequently the parameter storage cells are arranged in an array located close to the row of neuron circuits. Each column of the array is assigned to a single neuron circuit.

Figure 6.19 shows an overview for such an array of parameter storage cells and the additional circuits required for programming and refreshing the cells. The voltage cells are located in the upper half of the array, minimizing the distance between the output of the cells and the neuron circuits. This is important due to the high output resistance of the voltage cells, rendering them very sensitive to crosstalk. The current cells are located in the lower half of the array.

The layout of the new parameter storage system is organized in a way that up to 24 cells, current or voltage type, can be edge connected to form a column. The

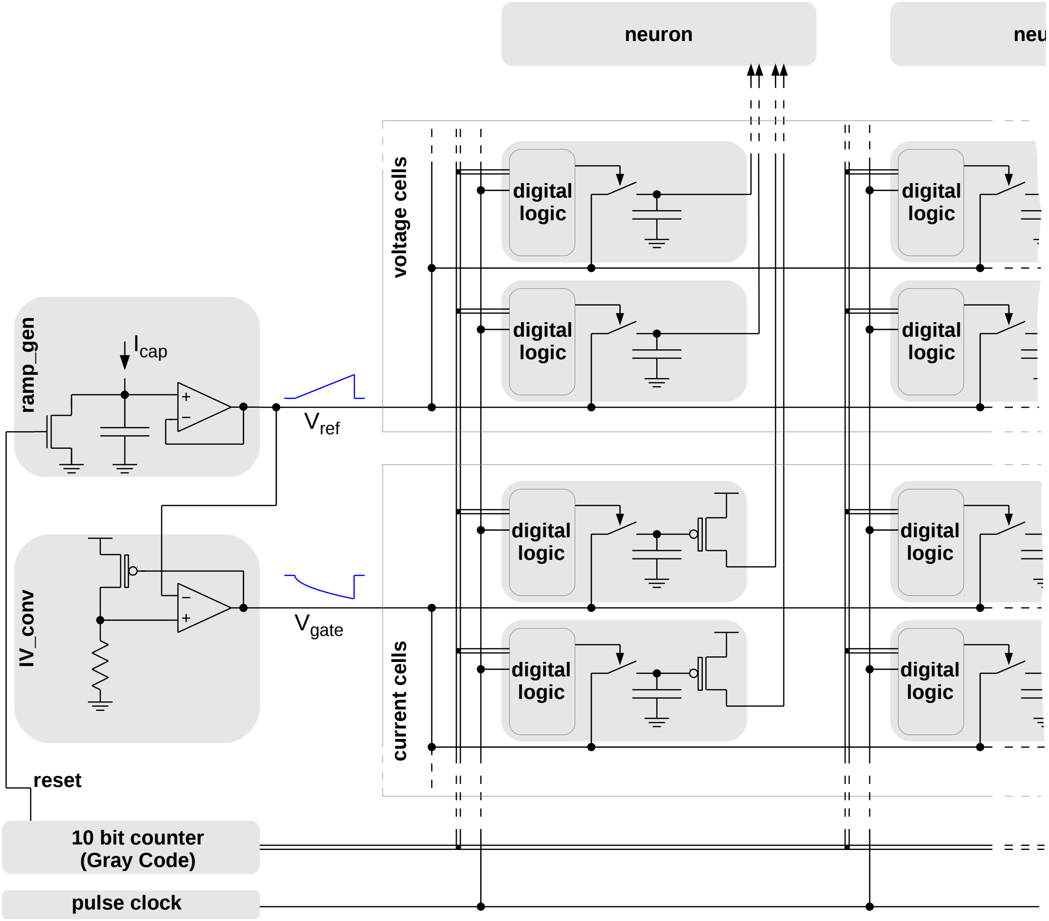


Figure 6.19: Overview of the full parameter storage system as it is implemented in the second prototype chip.

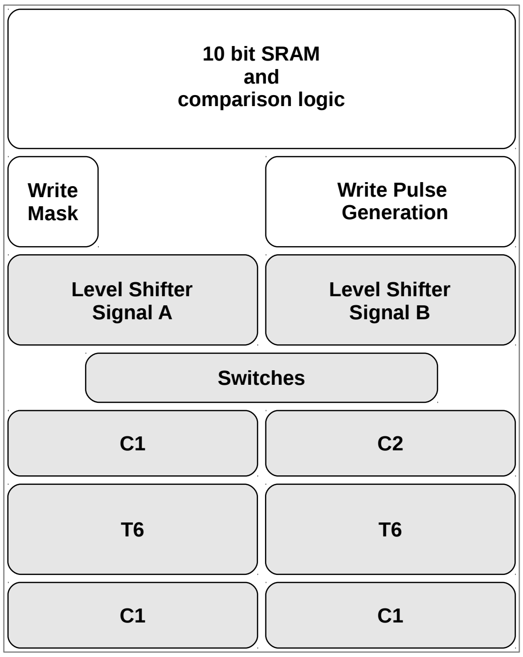


Figure 6.20: Sketch of the layout of a current cell. Size and position of the individual components corresponds to their size and position in the actual layout. The blocks marked in gray are built from thick-oxide transistors.

ratio of voltage to current cells within the column can be chosen according to the demands of the neuron circuit. To connect the outputs of all parameter cells in the same column to the neuron circuit, 24 vertically orientated wires are included in the layout of the cells. To prevent crosstalk between the individual parameters, additional wires connected to ground are placed between the vertical wires. When assembling a column, vias determining which parameter is connected to which of the vertical wires have to be added.

In Figure 6.20 the structure of the layout of a single storage cell is shown. Size and position of the individual components corresponds to their size and position in the actual layout. The names for the capacitors and the transistor in the lower half of the figure refer to the schematic of the current cell shown in Figure 6.6. The layout of the voltage cells is very similar. In the voltage cells the output transistor T6 is not required, instead it is used to increase the capacitance of C2, see Figure 6.3.3. The overall chip area covered by a single cell is $11.76\,\mu\text{m} \times 14.7\,\mu\text{m} = 173\,\mu\text{m}^2$. The chip area consumed by the supplementary circuits *ramp_gen* and *VI_conv* is about $38\,\mu\text{m} \times 84\,\mu\text{m} \approx 3200\,\mu\text{m}^2$.

6.4 Experimental Results

A full implementation of the parameter storage system is included in the second prototype chip. However, due to the limited physical dimensions of the chip the size of the cell array has been reduced to 32 × 24 parameter storage cells. Each of the 32 columns contains 12 current and 12 voltage storing cells. In the following individual cells are identified by their coordinates (m, n) where m specifies the column and n row of the cell.

The results of measurements performed to characterize and test the parameter storage system are presented in this section. A general description of the experimental setup and the measurement devices used is given in Chapter 4.5. Most parts of the system work as intended and its performance can be characterized. However, several mistakes in the implementation and general limitations of the design have been identified during the experiments. In Chapter 6.5 solutions for the identified problems are presented.

6.4.1 Test Circuits in the Prototype Chip

To allow for testing of the parameter storage cells some additional circuits are required. Due to the limited number of bond pads available, multiplexers are used to allow for measuring of the output voltages and currents of all 768 cells in the array. All voltages that can be monitored from the outside of the chip are buffered using the operational amplifier described in Chapter 5.5. In order to test the performance of the *ramp_gen* circuit, V_{ref} is not only buffered to be distributed to all of the voltage cells, there is also an additional buffer connecting it to the bond pad `cm_vref_buf`.

Voltage Cells

On top of every column there are 12 transmission gates multiplexing the output of the voltage cells in the respective column to one wire which is shared between all columns. The transmission gates are build from thick-oxide transistors. These can be controlled via the JTAG interface. Levelshifter circuits are used to generate 2.5 V signals from the 1.2 V signals provided by the digital part of the chip. At the end of the shared line the voltage is buffered and connected to the bond pad `cm_vout<0>`.

Additionally, three cells, located at the right edge of the array, are directly connected to individual buffers without any multiplexer stages in between, their outputs are connected to the bond pads `cm_vout<1>` to `cm_vout<3>`. These three extra bond pads have been implemented for two reasons. First to see the time dependent effects on the output of the voltage cells without the low pass filtering effect of the shared line connected to all multiplexers. Second, as a fall back solution for the case that the multiplexer stage does not work as intended.

The first tests involving voltage cells revealed a serious fault in the concept of the multiplexer circuits. The output of the voltage cells is not buffered, the output voltage is identical to the voltage on the second capacitor C2. The amount of charge which can be transfered to C2 by a single programming process is limited, the effective output resistance of the cells is in range of 15 GΩ, see Section 6.3.5. It was expected that it takes a significant amount of time for one voltage cell to charge the overall capacitance of the shared wire. However, it was observed that the voltage measured at `cm_vout<0>` can hardly be influenced at all by the value programmed to individual voltage cells. The reason for this is the leakage through of the 384 transmission gates connected to the shared line. The resistance of the transmission gates in the `off` state is very high, but the equivalent resistance of 383 disabled transmission gates in parallel is in the same order of magnitude as the output resistance of a single cell for which the transmission gate is enabled. As a result, the voltage level on the shared line resembles the average voltage for all cells in the array and does not allow for any precise measurement of the voltage level at the output of a single cell.

All measurements for voltage cells presented in the following are based on results obtained from the three cells per chip which are connected to an individual output buffer. However, these cells are also connected to the common readout wire by a transmissions gate. As discussed in Section 6.3.3, the leakage current through the switch disconnecting the storage capacitors from the programming voltage has been minimized in order to allow for a long storage time. The leakage current through the transmission gate connected to the output of the cell is expected to have a severe impact on the storage time. To avoid this effect, the voltage on the shared wire and inside the cell which is tested needs to be kept identical. Therefore all voltage cells have been programmed to the same or at least a similar value as the cell which is tested when measuring the storage time. Experiments have shown that when operated at a high refresh rate, the additional leakage through the transmission gate has no measurable impact on the output voltage of the cells.

Current Cells

For the current cells a multiplexing scheme similar to the one described for the voltage cells is implemented. On top of every column there are 12 thick-oxide PMOS transistors located which can be used to connect the output of an individual cell to a shared wire. This wire is directly connected to the bond pad **`cmin case multiple of these are programmed to similar target values._iout<0>`**. Again level shifters have to be used for the digital control signals. Additionally, the bias current used to generate the voltage reference ramp can also be multiplexed to the pad `cm_iout<0>`. As a fall back solution the output nodes of three cells located at the right side of the array are connected directly to the bond pads `cm_iout<1>` to `cm_iout<3>`. In case of the current cells multiplexing of the currents to the shared output works as intended, all 384 cells in the array can be tested.

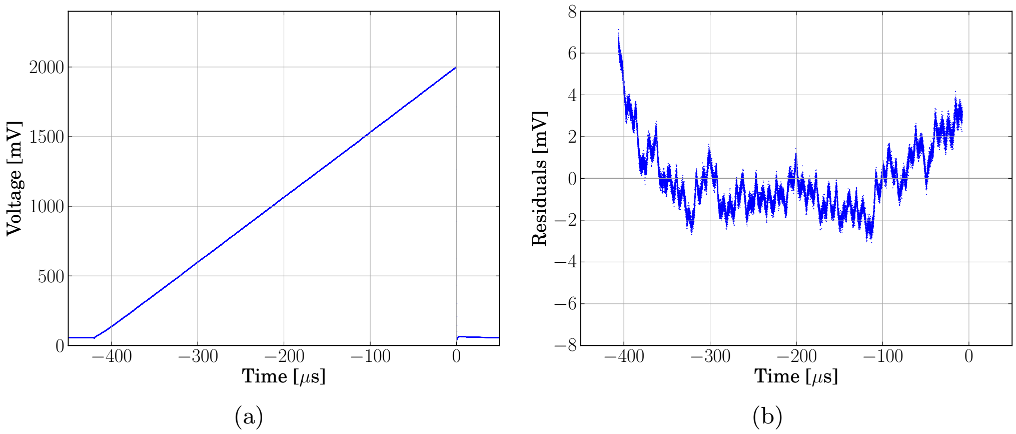


Figure 6.21: (a) oscilloscope recording of the linearly increasing voltage V_{ref} used for programming the voltage cells. The average over 100 individual traces is plotted. (b) shows the residuals against a linear fit.

6.4.2 Dynamic Range and Linearity for the Voltage Cells

Voltage Ramp Generation

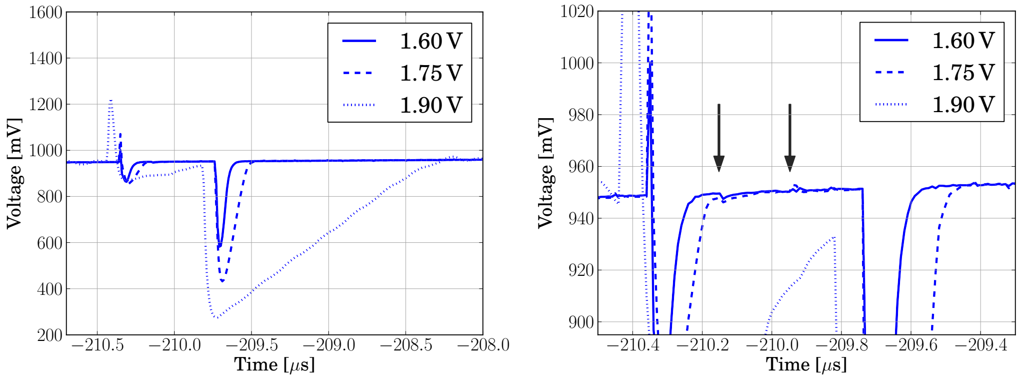
The first step on the way to analog output voltages or currents is the generation of the linearly increasing voltage ramp V_{ref} by the *ramp_gen* circuit, see Figure 6.12. Testing the ramp generation can be done by measuring the signal at the bond pad `cm_vref_buf`. Figure 6.21(a) shows an oscilloscope recording of the ramp voltage at chip 2.1 over one counter cycle. To reduce the impact of random noise the oscilloscope was configured to average over 100 trigger events. Figure 6.21(b) shows the residuals between the measured trace and a linear fit. A significant deviation of up to 6 mV from the ideal behavior can be observed for voltages close to 0. Some structured distortions are visible on the trace over the full voltage range. Since the trace shows an average over 100 runs the source of the noise needs to have a fixed timing correlation to the generation of the ramp. It is probably caused by crosstalk from digital signals involved in generating the counter. Linearity and noise might be affected by the characteristics of the readout amplifier, see Chapter 5.5. It is not clear if the noise is actually present on the V_{ref} signal distributed in the array of cells or if only the readout path is affected by crosstalk from digital signals.

The counter is operated at a frequency of 2.5 MHz, resulting in a duration of $0.4\,\mu\text{s} \cdot 1024 = 409.6\,\mu\text{s}$. Every counter cycle is followed by a break, during which the counter is stopped and the `reset` signal is active, which also lasts 409.6 μs. Unless otherwise specified, this configuration of the ramp timing is used for all measurements presented in the following. The slope of the ramp is controlled by I_{ref}, the maximum voltage of the ramp is used as a reference for the setting of I_{ref}. In Figure 6.21(a) the peak voltage of the ramp is set to 2.0 V ± 5 mV using the oscilloscope.

The voltage measured for V_{ref} during the reset state ranges between 60 mV and 80 mV, depending on which chip is tested. This offset is a combination of the individual offsets of the operational amplifier in the ramp generation circuit, OP1 in Figure 6.12, and the additional output amplifier driving the signal to the bond pad. The offset of the reset voltage is of no concern for the voltage cells. Since the output voltage drift is rather high for voltages close to 0, see Figure 6.8, it is not recommended to operate them at output voltages below 100 mV. For the dynamic range of the current cells the offset is a problem, it limits the minimum currents which can be produced reliably. Here the performance at the lower edge of the dynamic range is critical. The issue is discussed in more detail in Section 6.4.3 and in Section 6.5.3.

The reset signal of the *ramp_gen* circuit is triggered simultaneously with the counter reaching the code `1023`. Cells programmed to `1023` are therefore refreshed during the steep drop of the reference voltage caused by the reset. In this case the resulting output voltage is rather unstable. However, for the oscilloscope recording shown in Figure 6.21(a) all voltage cells in the array have been programmed to the digital code `1023`. This setting was chosen to avoid any potential impact of the refresh process of the cells on the V_{ref} signal.

In Figure 6.22(a) an oscilloscope recording of V_{ref} for the situation of all 384 voltage cells being programmed to `511` is shown. The effect of the simultaneous refresh process is clearly visible. Figure 6.22(b) shows a zoom on the time interval in which the refresh process is happening. The impact of the simultaneous programming of all cells is severe, distortions with an amplitude of several hundred millivolts are visible on the V_{ref} signal. The distortions are caused by two different effects. When transistor T1 in a voltage cells gets activated, the node connecting T1, T2 and T3 needs to be discharged from 2.5 V to the current value of V_{ref}, this causes the ramp voltage to rise. The second and probably more severe effect is the charge injection caused by switching the transistors T1 and T2. Both are controlled by the signal `A` which turns `1` when a programming cycle is initiated. At the end of the programming process a large dip in the ramp voltage, consistent with charge injection caused by signal `A` turning `0`, is visible. The time constant of the decay of the distortions can be controlled by the bias setting of the operational amplifier which buffers V_{ref}. In Figure 6.22 the effect of different settings of the amplifiers bias voltage is shown. The bias voltage controls the gate of a PMOS transistor which generates the bias current for the amplifier. A lower voltage leads to a higher bias current. A setting of $V_{bias} = 1.6\,\mathrm{V}$ leads, according to simulations, to a rather high average current consumption of $45.5\,\mu$A for the buffer. Though distortions on the ramp where expected to some degree, the impact of simultaneously programming all cells is surprisingly severe. However, to prevent the effect of affecting the accuracy of the refresh process, the temporal sequence of the signals `A` and `B` was introduced during the design phase. Figure 6.22(b) shows that this strategy is basically working. The points at which signal `B` switches to start and end the actual programming process of the cells are marked by arrows. During



(a)

(b)

Figure 6.22: (a) effect of programming all 384 voltage cells to the same digital code of `511` at various settings of the bias voltage for the buffer in the *ramp_gen* circuit. (b) shows a zoom on the refresh process.

the interval in which `B` is active, and the cell's capacitor C1 is connected to V_{ref}, no significant distortion is visible as long as the bias voltage of the buffer is below 1.75 V.

In the prototype chip, the timing of the signals `A` and `B` is not ideal due to a mistake in the digital *pulse_gen* circuit, see Section 6.3.4. The relative timing between `A` and `B` is correct, but in relation to the counter signals both signals are activated longer than intended. This causes a temporal overlap in the programming processes of different cells which have consecutive digital target values. The signal `B` of a cell at a digital code of `x` is deactivated at the same time `A` is activated for cells programmed to `x+1`. Measurements suggest that delays in the implementation even lead to a short overlap of these signals. As a consequence the programming process for a cell at `x` is affected by the distortion on V_{ref} which is caused by signal `A` of cells programmed to `x+1`. Output voltage deviations of more than 100 mV have be observed for a single cell at a code of `x`, if all of the remaining cells in the array are programmed to a target value of `x+1`. As a consequence of the incorrect timing generation, the voltage cells of the array implemented in the chip can not be operated simultaneously at random target values.

The same problem applies to the current cells, the programming voltage for the current cells, V_{Gate}, is also distorted if many cells are programmed at once, see Section 6.4.3. An improved version of the *pulse_gen* circuit, generating the correct timing sequence, is presented in Section 6.5.2. Nevertheless it is necessary to limit the impact of the programming process on the reference voltage. A strategy how this can be achieve is presented in Section 6.5.5.

Voltage Cells

As explained in Section 6.4.1, reading the output voltages of the cells in the array is not possible for most of the cells. The readout multiplexers are not working correctly due to the extremely high output impedance of the voltage cells. Precise measurements can only be obtained from the three cells per chip which are connected to individual output amplifiers. Furthermore it has to be ensured that, due to the incorrect timing of the programming signals, the cells under test are not affected by the programming process of other cells in the array. As a consequence the measurements presented in the following are made using the accessible cells, the rest of the array is programmed to a digital value of `1023`. Only when measuring storage times a different setting is used, for details see Section 6.4.6.

The dynamic range of the voltage cells was expected to be in a range between about 0 and 2 V. Figure 6.23 shows the output of a voltage cell over the full range of digital codes, the peak voltage of the ramp is set to 2 V. For every digital code the output voltage has been measured 16 times, in between the single measurements the cell has be programmed to a digital code of `0` in order to avoid any correlation between the single measurements. The average measurement error, determined from the standard deviation within the 16 repetitions, is below 0.5 mV and therefore too small to be visible in this plot. At the lower end, the minimum output voltage is about 80 mV, which is consistent with the offset that has been observed for the programming voltage V_{ref}. Measuring the output of the voltage cells also involves a readout amplifier, possibly contributing to the offset. At the upper end a steep rise, starting at about 1.8 V, and followed by saturation at 1.95 V can be observed. This behavior was not expected. The reason for both, rise and saturation was identified to be the limited conductivity of the NMOS switch transistors in the cell at high voltages. The gate voltage of the enabled transistors is 2.5 V and sufficient conductivity is possible as long a V_{DS} is larger or equal to V_{th}. Therefore saturation was expected to happen at a higher voltage of $VDD - V_{th} \approx 2\,\mathrm{V}$. This estimation neglects the body effect, see e.g. Razavi [2001, p. 23], the bulk of the transistors is not at the same potential as its source contact but always connected to ground. As a result, the effective gate voltage is decreased and limited conductivity affects the cell for voltages larger than 1.8 V.

The device names in the following refer to the schematic of the voltage cells shown in Figure 6.10. The steep rise is caused by transistor T2, pulling the node connecting transistors T1, T2 and T3 to 2.5 V as long as the cell is holding a value. During a refresh process the node is typically quickly discharged towards the ramp voltage V_{ref} as soon T1 is activated. But for V_{ref} being higher than 1.8 V, the conductivity of T1 is not sufficient to discharge the node before T3 becomes activated. Despite the fact that conductivity of T3 is also limited, the storage capacitor is affected and the voltage at the storage capacitor rises. For voltages larger that 1.95 V, the conductivity of the NMOS transistors is so low, that no additional charge is transmitted to the storage capacitor and the output of the cell saturates. In order to map the available dynamic voltage range to the full

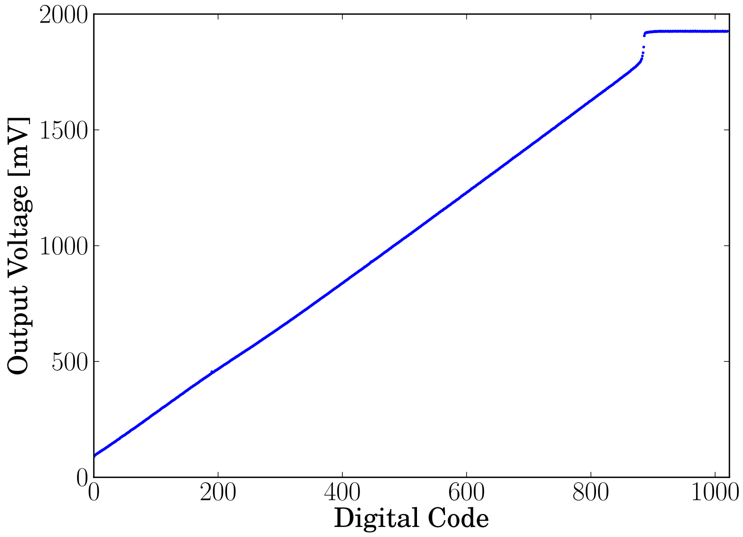


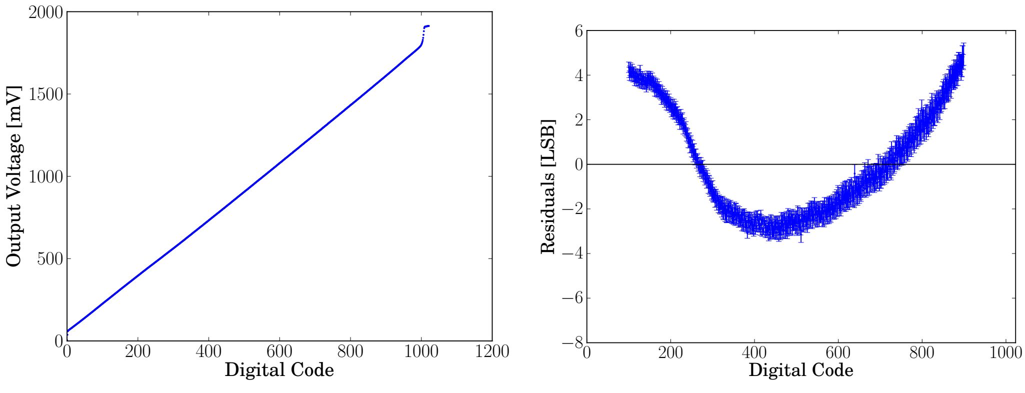
Figure 6.23: Output voltage of voltage cell (15,0) of chip 2.1 over digital codes. For output voltages larger than 1.8 V the linearity is severely disrupted by a saturation effect. See text for further explanation.

range of digital codes available, I_{ref} is reduced so that the voltage ramp reaches a peak level of only $V_{ref} = 1.85\,\mathrm{V}$. This setting is used for all measurements involving voltage cells which are presented in the following. As a consequence, the output voltage is only affected by the limited conductivity for digital codes larger than about `1000`. Up to this point, corresponding to an output voltage of 1.8 V, the output characteristic of the cell is linear. Accordingly 1 LSB step corresponds to 1.8 V / 1000 = 1.8 mV for the voltage cells.

Figure 6.24(a) shows the output voltage of cell (0,13) at chip 2.2 over the full range of digital target values with the optimized setting of I_{ref}. The average error on the single data points is 0.34 mV, obtained from 16 independent repetitions of the measurement. The characteristics of this cell are considered to be typical, the aspect of cell-to-cell variation is discussed in Section 6.4.4.

To investigate the linearity of the output voltage a linear fit is applied for the range from `100` to `900`. The resulting residuals are shown in Figure 6.24(b). A significant integral non-linearity is visible, the residuals are within a range of ± 5 LSB. It is caused by a systematic effect as the residuals are changing gradually in a continuous manner. The nonlinearity of the voltage ramp, see Figure 6.21(b), as well as the characteristics of the output amplifier are effects that contribute to the nonlinearity observed here. Furthermore the parasitic capacitances in the switch transistors are voltage depended. Therefore the efficiency of the charge injection compensation may vary over the dynamic range, leading to nonlinear characteristics of the output voltage.

However, systematic nonlinearity at this level is not considered to be a problem.



(a) (b)

Figure 6.24: Characteristics of the voltage cell (0,13) of chip 2.2. (a) shows the output voltage over the full range of digital codes. The average error on the individual measurements, obtained from 16 repetitions, is too small to be visible. (b) shows the residuals of the output voltage relative to a linear fit applied in the range from 100 to 900. The error bars indicate the standard deviation over 16 repetitions of the measurement.

As long as the effect is comparable for all cells on the chip, it can be compensate by a single calibration step.

6.4.3 Dynamic Range and Linearity for the Current Cells

Generating the Programming Voltage for the Current Cells

The voltage V_{Gate}, used to program the current cells, is generated by the *VI_conv* circuit described in Section 6.3.4. The prototype chip does not provide a possibility to measure V_{Gate} directly, testing of this circuit has to be done indirectly by measuring the output characteristics of the current cells. During the design phase simulations suggested that the voltage to current converting circuit can become unstable under certain conditions. In that case the voltage V_{Gate} shows oscillations of rather small amplitude, compared to the absolute level of V_{Gate}. Measurements suggest that this is happening in the prototype chip even more likely than expected from simulation. This indicates that parasitic effects were not modeled with sufficient accuracy in the simulation setup. However, tuning of the bias current for OP1 and configuring the resistor R1 allows for a stable configuration. Best results are obtained at a bias voltage of 1.9 V for the operational amplifier in *VI_conv* circuit.

As already described for the voltage cells, multiple cells being programmed to the same value lead to distortions on the programming voltage. Charge injection from switching of transistors T1 and T2 of the voltage cells affects V_{Gate}. For robust

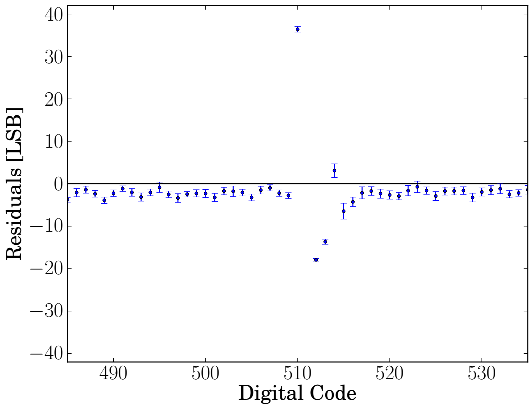
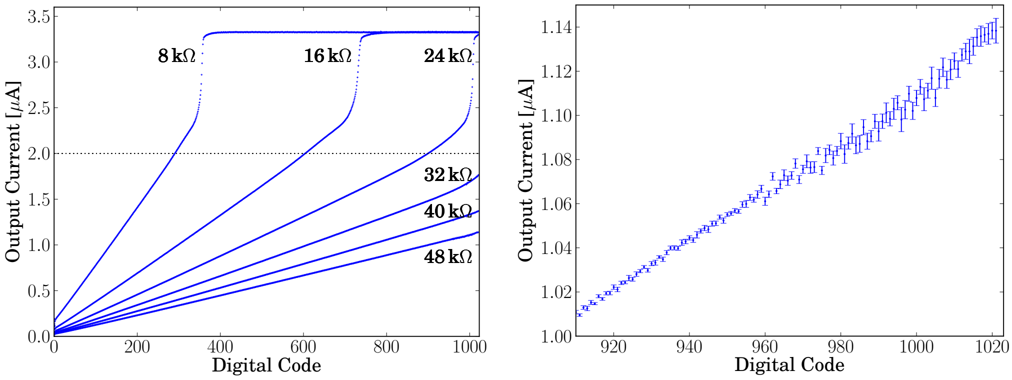


Figure 6.25: Residuals between the output current of a cell and a linear fit over digital codes. The remaining cells in the array are programmed to `511`, the bias voltage of the buffers is set to 1.9 V. Severe distortions of the output current can be observed for digital codes close to `511`. This result indicates that the programming voltage V_{Gate} is distorted by the simultaneous refresh of the full array.

operation in case all current cells in the array are programmed to the same level, a lower level of the bias voltage, resulting in a larger bias current for the amplifier, is required. In case of the current cells, the effect can not be measured directly as V_{Gate} is not accessible. However it can be observed by indirect measurements. Figure 6.25 shows the residual between the output current and a linear fit over digital codes for a current cell. The remaining cells in the array are programmed to `511` and the bias voltage of the buffer is set 1.9 V. The output of the tested cell is significantly distorted for the digital codes that are close to the code to which the full array is programmed.

However, the instability of the *VI_conv* circuit prevents the usage of large bias currents to reduce the impact of this effect. A strategy how to achieve an improved version of the *VI_conv* circuit, providing stability and a sufficiently high bandwidth, is presented in Section 6.5.3. Unless otherwise specified, the bias voltage of the amplifier is set to 1.9 V for all measurements presented in the following. The situation of programming all cells to the same target value is avoided.

Figure 6.26(a) shows the output characteristics of a current cell for different settings of R1. For smaller values of R1 the stability is improved, however the usable range of digital codes is reduced as saturation effects limit the output current. The instability of the output current is not visible in the figure as the amplitude of the distortions is rather small. Figure 6.26(b) shows a detail of the trace recorded at a setting of R1 = 48 kΩ, the error bars denote the standard deviation of the 16 individual measurements which are taken at each digital code. A measure for stability of the *VI_conv* circuit is to compare the average value of the standard deviations



(a)

(b)

Figure 6.26: (a) shows the output current of cell (0,11) from chip 2.2 over digital codes for different settings of the resistor R1 used in the *VI_conv* circuit. The measurement error, determined from 16 repetitions, is too small to be visible. The specified upper edge of the dynamic range, 2 μA, is marked by a horizontal line. (b) shows a zoom of the trace recorded at a setting of R1 = 48 kΩ. The size of the measurement error indicates instability of the *VI_conv* circuit.

measured at every digital code for individual settings of R1. The corresponding data is shown in Table 6.3. For values larger than R1 = 24 kΩ the average standard deviation rises significantly. The numbers are only an indicator for the problem, not a precise measure. The Sourcemeter used to measure the output current integrates over a timespan of 20 ms for each measurement, low-pass filtering the input signal. The actual amplitude of the distortions of the output currents in case the *VI_conv* circuit becomes instable is larger.

In order to test the current cells with a stable V_{Gate} a setting of R1 = 24 kΩ is used in the following. To achieve the maximum possible resolution of digital codes, the slope of the V_{ref} voltage ramp is decreased. I_{ref} is reduced to a value which leads to a peak voltage for V_{ref} of only 1.4 V ± 5 mV for all measurements involving current cells. With the chosen settings, the average maximum output current of the cells is in a range of 1.7 μA. Consequently 1.7 μA / 1024 ≈ 1.7 nA corresponds to 1 LSB step.

Despite the effect of different settings for R1, Figure 6.26(a) also shows the available dynamic range for the current cells. Details regarding this characteristic of the cells are discussed in the following.

Current Cells

The performance of single current cells in terms of linearity and dynamic range is presented in the following. Figure 6.27(a) shows a sweep over the full range of

R1 $[k\Omega]$	Average STD $[nA]$
8	0.6
16	0.4
24	0.4
32	0.7
40	3.9
48	8.4

Table 6.3: The output current of cell (0,11) from chip 2.2 was recorded at different settings of R1 in the *VI_conv* circuit. The average standard deviation for the individual data samples indicates instability of the circuit for large values of R1.

digital codes for cell (30,11) of chip 2.2, Figure 6.27(b) shows the residuals relative to a linear fit applied in the range from `10` to `1022`. The integral nonlinearity is comparable to the one observe for voltage cells. Again the spread of the residuals is continuous, clearly indicating that a systematic effect is the source.

Covering the upper end of the specified dynamic range is possible, however with the setting chosen for the measurements, the highest output current available is about 1.7 μA. For smaller values of the resistor R1 in the *VI_conv* circuit the output current range can be extended to a maximum of 2.4 μA, see Figure 6.26(a). From that point the linearity of the output is severely degraded by the limited conductivity of the switch transistors. The effect has already been discussed for the voltage cells where it limits the maximum output voltage of the cells to 1.8 V. However, different than for the voltage cell this is not happening within the specified operating range and is therefore no limitation.

An important aspect is the performance of the current cells at the lower end of their dynamic range. Figure 6.28 shows the output of three randomly selected current cells from three different chips at low digital codes. While the output current at `0` is in range of only 5 nA for all cells, it rises quickly to about 60 nA within the next steps. From this point on a linear ascent of the output currents can be observed, the slope is within the expected range.

This result indicates that the dynamic range of the current cells is affect by an general offset in range of 60 nA. The behavior for the digital codes below `5` is caused by a distortion of V_{ref}, introduced by disabling the `reset` signal. Measurements with the oscilloscope show that V_{ref} is affected by charge injection from the transistor T1 in the *ramp_gen* circuit when the `reset` signal is released. As a result, V_{ref} is temporarily pushed towards lower voltages at the beginning of a counter cycle. Since V_{Gate} is derived from V_{ref}, the effect also has an impact on the current cells.

The lower limit of about 60 nA for the output of the current cells is not expected, in simulations the system is able to reliably generate smaller output currents. Re-

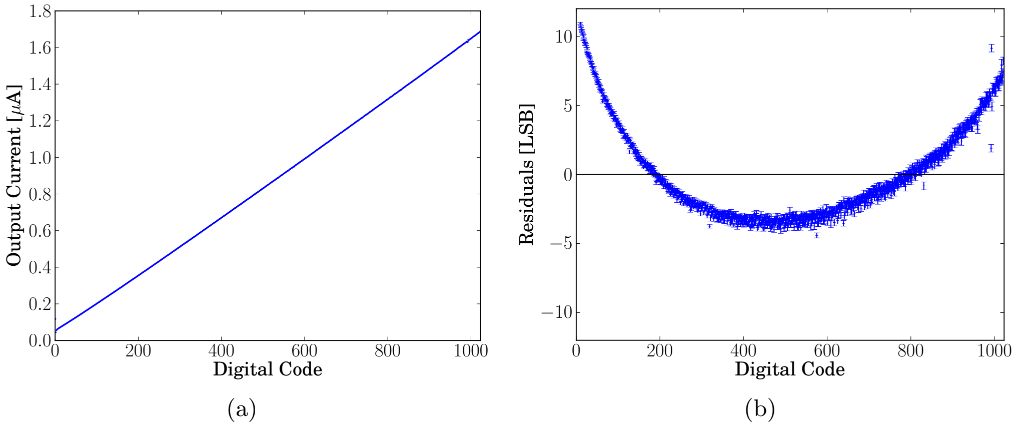


Figure 6.27: Characteristics of the current cell (30,11) of chip 2.2. (a) shows the output current over the full range of digital codes. (b) shows the residuals of the output current relative to a linear fit.

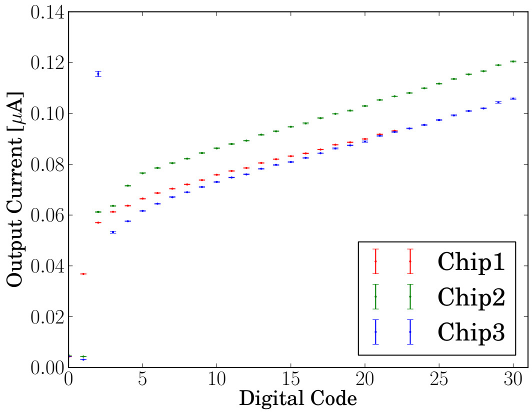


Figure 6.28: Output current over low digital codes for three randomly selected current cells from three different chips. The error bars indicate the standard deviation over 16 repetitions of the measurement. The output currents show an offset in range of 60 nA. The output currents for digital codes below 5 are a result of a transient distortion of V_{Gate}, caused by deactivation of the reset signal.

sponsible for the shift towards higher output currents are probably the input offsets of the two operational amplifiers involved in generating V_{Gate}. One is used in the *ramp_gen* circuit, buffering V_{ref}, the second one is generating V_{Gate} in the *VI_conv* circuit. The general problem was not identified during the design phase, as the amplifiers have not been tested using Monte Carlo simulations. Since the performance of the parameter storage system in the low current range is critical, an option to compensate for the offset needs to be implemented in future version of the *VI_conv* circuit. A detailed analysis of the problem, along with options how to solve it, is discussed in Section 6.5.3

6.4.4 Cell-to-Cell Variation

So far most of the measurement results presented have been obtained from individual voltage and current cells. These cells have been selected as examples, their behavior is considered to be typical for the respective type of cells. In the following, measurement results comparing the performance of multiple cells per chip are presented. As mentioned before, only three voltage cells per chip can be tested, here the statistical significance of the results is limited. In case of the current cells 348 cells per chip can be tested and quantitative results are obtained.

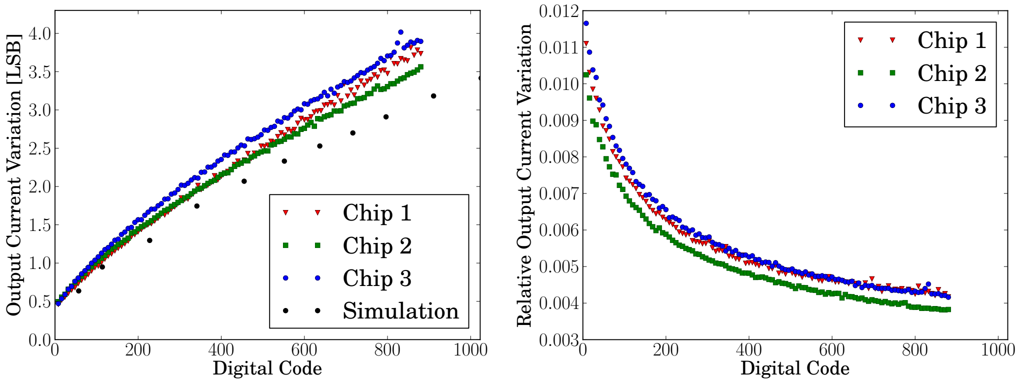
Voltage Cells

According to Monte Carlo simulations, the voltage cells are rather robust against cell-to-cell variation. A simulation, including 200 Monte Carlo parameter sets, of a voltage cell which is programmed to a digital code of `512` results in an output voltage variation of only 245 μV, which is significantly below 1 LSB. The simulation results are similar over the full dynamic range.

The high precision can be explained by the fact that the transistors involved are operating as digital switches, the circuit is not relying on precise transistor parameters. Only the variation of the parasitic capacitances in the transistors has an effect on the stored voltages, as this changes the efficiency of the charge injection compensation. Due to the mistake in the readout circuits it is not possible to test more than three voltage cells per chip, see Section 6.4.1. Therefore it is not possible to verify the simulation results. In fact it is not even possible to compare the output voltages of the three cells which are accessible on the same chip. Each cell uses an individual output buffer, the individual input offset variation of the amplifier is much larger than the expected cell-to-cell variation of the cells, see Chapter 5.5.

Current Cells

The variation between individual current cells is an important aspect. Monte Carlo simulations suggest that it is clearly dominated by the device mismatch of the output transistor T6. To measure the cell-to-cell variation, all cells of a chip are individually programmed to the same digital code and the corresponding output current is measured for each cell. During the measurement of a single cell, the



(a)

(b)

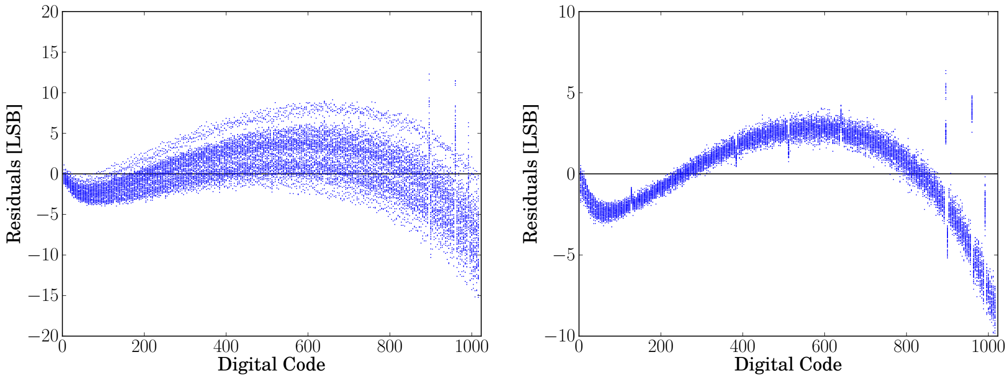
Figure 6.29: Mismatch between the individual current cells. The standard deviation of the output currents of all cells on the same chip is plotted over digital codes. Additionally the result of a Monte Carlo simulation, accounting only for mismatch of the output transistor of the current cells, is shown.

remaining cells are programmed to `1023` to avoid the crosstalk problem mentioned before. The digital codes are swept in steps of `8` for a range from `0` to `880`, covering an output current range between 60 nA and about 1.5 μA. Figure 6.29 shows the resulting standard deviation of the output currents over all cells on the three chips in absolute and relative units. The results are similar for all three chips, the visible difference is probably caused by the fact that the *IV_conv* circuits of the chips have not been calibrated. Device mismatch in the conversion resistor R1 leads to slightly different absolute output current ranges.

Additionally the results of Monte Carlo simulations, accounting only for the variation of the output transistor, are shown in Figure 6.29(a). According to the documentation, the Monte Carlo models predict the relative mismatch for transistors in close spatial proximity. Since the output transistors of the single current cells are distributed over an area of about 350 μm × 180 μm, the slightly larger variation observed in measurements fits the simulation results. The only option to reduce the variation significantly is to increase the size of the output transistor. The variation of the drain current of a transistor decreases only with the square root of its gate area, see Chapter 5.3. In the layout of the current cells the output transistor covers already a significant fraction of the total area. Reducing the variation is only possible at the cost of significantly larger cells.

In Section 6.4.3 the characteristics of a single current cell have been discussed. Figure 6.27(b) shows that the output characteristics of the cell is affected by a significant integral nonlinearity. The distribution of the samples clearly indicates that a systematic effect is the cause.

In order to investigate the variation of the nonlinearity between the cells on the



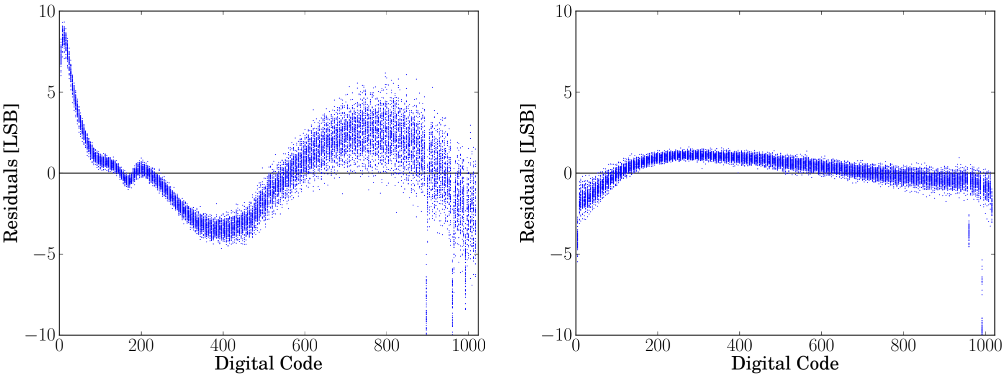
(a)

(b)

Figure 6.30: Residuals of the output current against a linear fit for 32 cells from chip 2.3. For (a) all data points have been considered for a single linear regression, the cell-to-cell variation is clearly visible. For (b) the cell-to-cell variation caused by the device mismatch of the output transistor is eliminated by applying individual fits for every cell. The systematic non-linearity becomes clearly visible. Note the different scaling of the y-axis between (a) and (b). For individual digital codes at the upper end of the dynamic range the output currents deviates by more than 5 LSB from the expected value.

same chip, the results of sweeping digital codes for multiple cells is presented. To reduce the time required for the measurement only a fraction of 32 cells per chip, homogeneously distributed over the array, has been included. Figure 6.30(a) shows the residuals of all output current traces recorded on chip 2.3 against a linear fit. All data samples shown have been included in the linear regression. The impact of the cell-to-cell variation is clearly visible in the spread of the data points. The device mismatch of the output transistor mostly causes a variation of the slope of the output characteristics. To eliminate the cell-to-cell variation a linear regression is applied to every individual cell. The resulting residuals are shown in Figure 6.30(b), the spread of the data samples is significantly reduced. The structure of the remaining nonlinearity is very similar for all cells, this emphasizes the assumption that it is caused by a global effect. For the digital codes `896`, `960` and `992` the resulting output currents deviate by more than 5 LSB from the expected value. According behavior is visible for all cells. However, the source of this effect has not yet been identified.

Figures 6.31(a) and 6.31(b) show the residuals against individual linear fits for the chips 2.1 and 2.2, in each case the results of 32 cells are plotted. Again the basic structure visible in the residuals is identical for the cells within the same chip. However it deviates significantly from chip to chip, the course of the residuals



(a) (b)

Figure 6.31: Residuals of the output current against linear fit for current cells of chips 2.1 (a) and chip 2.2 (b). To eliminate the variation introduced by device mismatch an individual linear fit is performed for each cell. The systematic non-linearity of chip 2.1 has a significantly different characteristic than for the chips 2.2 and 2.3. Again the output current for individual digital codes deviates significantly from the expected values.

shows a fundamentally different behavior. The source of the nonlinearity is not yet identified, according behavior could not be reproduced in simulations. Again the output current of few individual codes at the upper end of the dynamic range show a deviation of more than 5 LSB from their expected value.

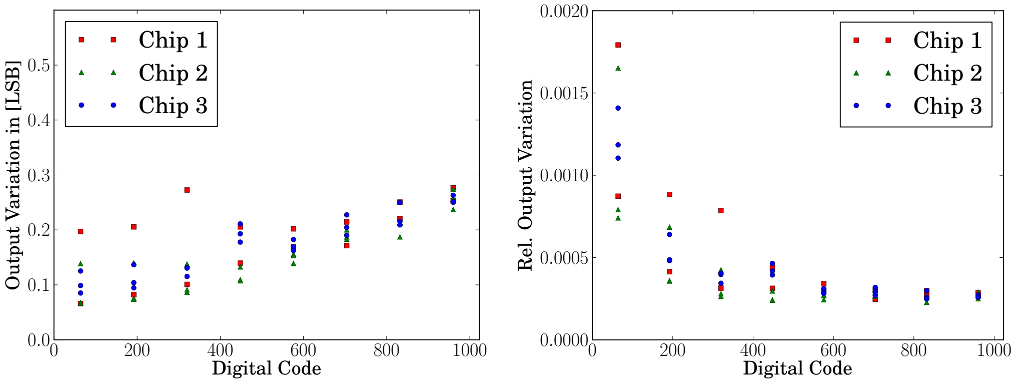
6.4.5 Reproducibility

An important characteristic of any analog parameter storage system is reproducibility. Calibration of single storage cells, or the circuits connected to it, is only possible down to the accuracy within which the cells behavior can be reproduced.

In a first step, the average output voltages and currents are measured multiple times at the same digital code, between the individual measurements the cell is reprogrammed to a random value before it is written back to the test code. During the measurement the analog value stored in the cell is constantly refreshed. This way it is only tested if the reprogramming of the digital code in the SRAM has any permanent effect on the output of the cells. In a second step the variation of the output value after each individual analog refresh cycle is measured.

Voltage Cells

The impact of reprogramming the target value of a voltage cell on its average output voltage is investigated. The accessible voltage cells are tested at eight different



(a)

(b)

Figure 6.32: Variation of the output voltage for repetitive programming of voltage cells over digital codes. The output voltage of every usable cell has been reprogrammed and measured 400 times for each of the digital codes. (a) shows the resulting absolute standard deviation over the measurements, in (b) the relative variation of the output voltage is shown. The standard variation is significantly below 1 LSB for all codes.

digital codes that cover the full dynamic range. Each cell is reprogrammed 400 times for every digital code, after each programming process the output voltage is measured. In between the individual cycles the cell is programmed to a random digital code to ensure independent programming processes and measurements. The average output voltage is measured using the Sourcemeter which is configured to integrate the input signal over a time interval of 20 ms. The analog value stored in the cell is refreshed at a rate of 1.2 kHz.

Figure 6.32(a) shows the resulting standard deviation of the single measurement for each cell over digital codes. In Figure 6.32(b) the relative variation is shown, the standard deviation is divided by the average output current. The variation between the individual measurements is significantly below 1 LSB.

However, since reprogramming the digital code of a cell is a digital process, a low variation is expected in this measurement. The analog value stored in the cell is updated 25 times during the interval in which the Sourcemeter takes one sample, compensating for transient distortions in the output voltage.

The observed variation is assumed to be caused by the general noise level in the experimental setup and the accuracy of the Sourcemeter. In order to verify that the reprogramming processes do not contribute to the variation, the experiment has been repeated for chip 2.2 without reprogramming the cell in between the individual measurements. The results of both measurements do not show any significant difference.

Next the output voltage variation introduced by individual refresh cycles is investigated. The digital part of the capacitive memory storage system provides the `cm_trigger` signal which is active during the counter cycle and 0 during the intermediate breaks. The `cm_trigger` signal, which is accessible at a bond pad of the chip, can be used to trigger external measurement devices. The duration of the breaks in between two analog refresh cycles is increased to 26.6 ms. A setup comparable to the one described for the measurements presented in Section 6.2.5 is used. A script is transmitted to Sourcemeter and executed by its internal microprocessor. The `cm_trigger` signal is connected to a general purpose IO which is used as a trigger input, sensitive to the falling edge of the signal. The Sourcemeter is configured to use a trigger delay of 1 ms, take a measurement with an integration time of 2 ms and store the result in an internal data buffer. This procedure is executed in a loop by the internal microcontroller. Afterwards the data stored in the buffer is read back using the RS-232 connection.

In Figure 6.33 the results of this measurement are presented. For each accessible voltage cell 380 voltage measurements, each taken after an individual refresh process of the cell, have been taken. The resulting standard deviation over these measurements is plotted over digital codes. Comparison with Figure 6.32, showing the variation of the average output voltage during operation, shows that accounting for individual refresh processes introduces only an insignificant amount of additional variation. Overall, the observed variation of the output voltages is significantly below 1 LSB and probably limited by the general measurement uncertainty in the setup.

Current Cells

The same measurements as described for the voltage cells are used to characterize the reproducibility for the current cells. First the effect of reprogramming a cell on the average output current is investigated. In order to limit the time required by the measurement, only a subset of 32 cells is taken into account. These are distributed homogeneously over the array. Each cell is reprogrammed 400 times for every digital code, after each programming process the output current is measured. The Sourcemeter is configured to integrate over a timespan of 20 ms, within this interval the cell is refreshed 25 times. In between each measurement the cell is programmed to a random digital code to ensure independent programming and measurement processes.

Figure 6.34(a) shows the standard deviation over the single measurements from 32 cells per chip. In Figure 6.34(b) the relative variation is shown, the standard deviation is divided by the average output current at the respective code. For the cells on chip 2.1 the variation of the single measurements is significantly larger than for chip 2.2 and 2.3. As mentioned in Chapter 4.4, the capacitive memory system on chip 2.1 is affected by an internal defect. Though the exact source of the problem is unknown, it is assumed that the increased variation visible in Figure 6.34 is linked to this issue. Excluding the results obtained from chip 2.1, the absolute standard

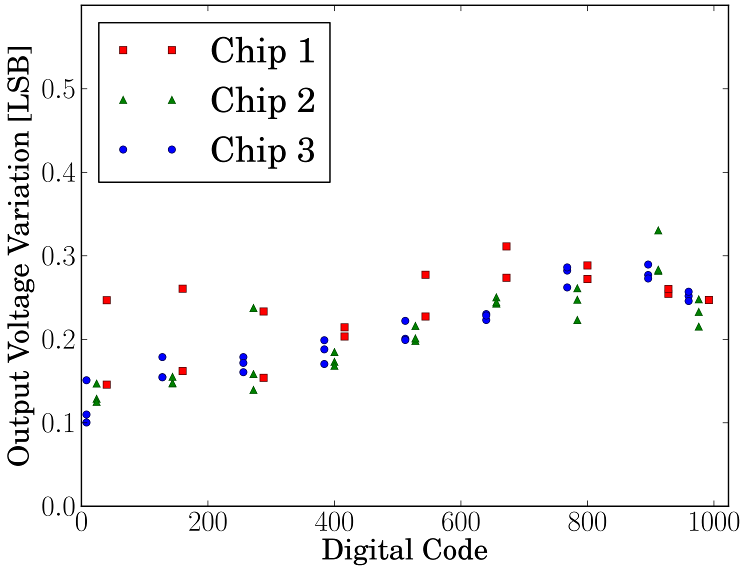
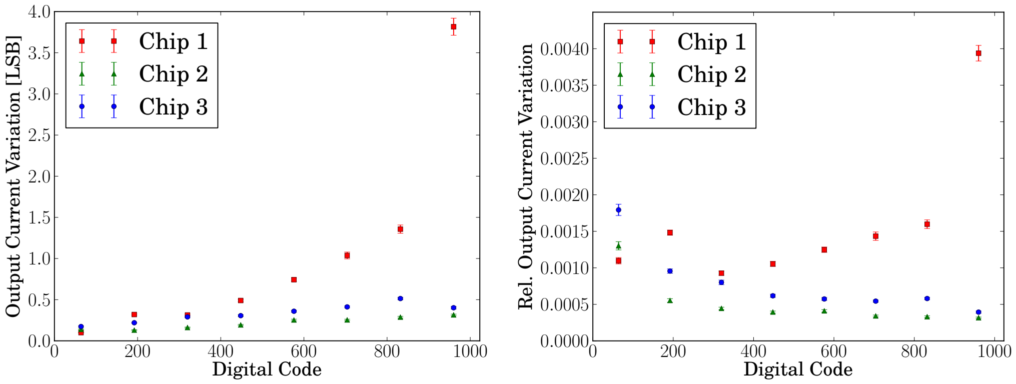


Figure 6.33: Variation of the output voltage after individual refresh cycles. The standard deviation of 380 measurements is plotted over digital codes.

deviation observed is below 0.5 LSB.

Again the results have been compared to the results of an identical experiment in which the cell are not reprogrammed in between the single measurements, in order to verify that the observed variation is not related to the reprogramming. The results of both measurements do not show any significant difference. As for the voltage cells, the variation between the single measurements of the output current is assumed to be caused by the general noise level in the experimental setup, at least for chips 2.2 and 2.3.

Next the impact of individual refresh cycles, using the same setup as described for the voltage cells, is investigated. The output current is measured 380 times after individual refresh cycles, using the `cm_trigger` signal, for all current cells on the test chips. The result is shown in Figure 6.35, the average standard deviation for the individual cells is plotted against digital codes. The maximum variation observed is in range of 2 LSB for chips 2.2 and 2.3, chip 2.1 again shows a significantly higher variation. In comparison to Figure 6.34(a), the variation observed between individual refresh cycles is larger by about 1 LSB. This might be caused by the limited precision of the refresh processes. However, when measuring currents, the accuracy of the Sourcemeter can be expected to degrade if short integration times are chosen. Considering the overall noise level in the system, the variation is within an acceptable range for the chips 2.2 and 2.3.



(a)

(b)

Figure 6.34: Variation of the output current for repetitive programming of current cells over digital codes. On every chip a subset of 32 cells, homogeneously distributed over the array, was tested. The output current of every cell was reprogrammed and measured 400 times for each of the digital codes. (a) Shows the average standard deviation for all cells on the same chip, in (b) the relative variation of the output currents is shown. The variation is below 0.5 LSB, only the results for chip 2.1 show a significantly higher variation than the cells on the other test chips.

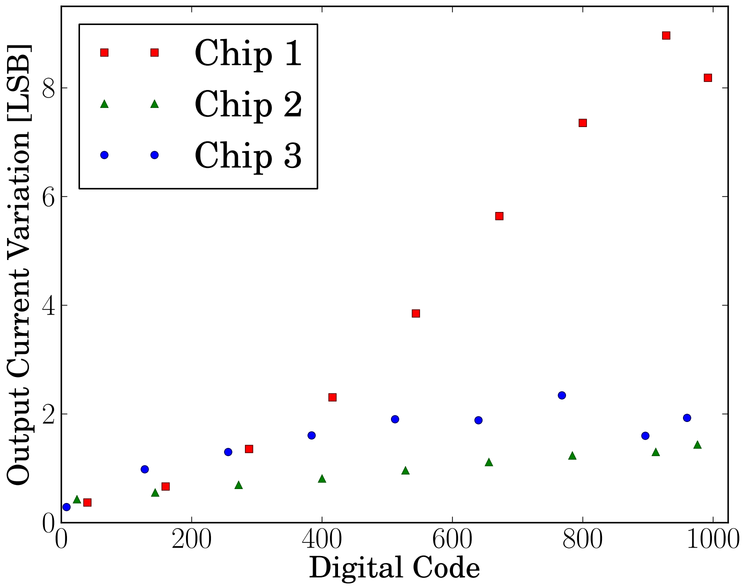


Figure 6.35: Variation of the output current after individual refresh cycles. The standard deviation obtained from 380 measurements for each cell on the respective chip is plotted over digital codes.

6.4.6 Storage Time

An important characteristic of the parameter storage system is the drift rate of the output voltages and currents. Low drift rates allow for longer breaks between the power consuming refresh cycles.

Voltage Cells

An overview over the drift of the three accessible voltage cells of chip 2.2 is given in Figure 6.36. The voltages are sampled at a rate of 40 samples per second using the Sourcemeter. The cells were initially programmed to the digital codes `8`, `511` and `960`. Before stopping the refresh cycles, the output voltage is measured multiple times to get an accurate start level V_{init} for the individual cells. This initial output voltage is used to compensate the offset introduced by the individual output amplifiers of the cells. The equilibrium state towards which the cells are drifting varies significantly from cell to cell. Comparable behavior is also observed for the voltage cells on the other test chips. From simulations an equilibrium level of 170 mV is expected for the voltage cells, cf. Section 6.3.3.

For quantitative measurements of the output drift a setup similar to the one used in the reproducibility measurements, see Section 6.4.5, is used. The break between two programming cycles is configured to be 26.6 ms and again the `cm_trigger` signal is used. To allow for accurate timing, a script controlling the measurements is

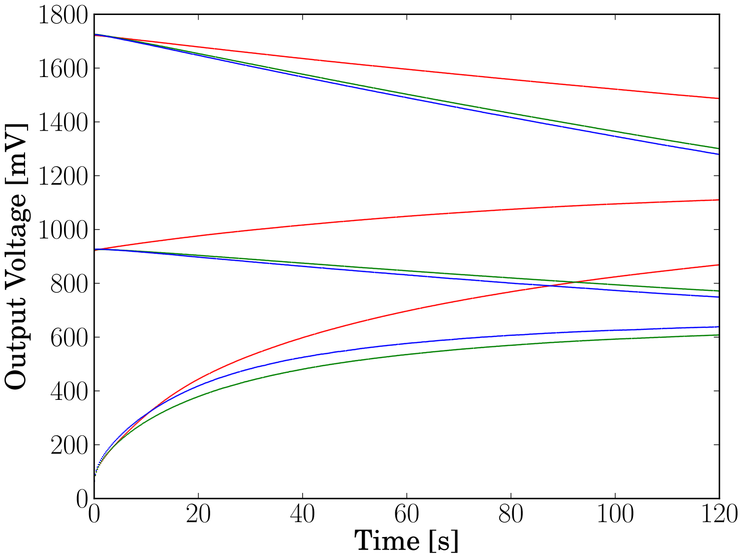


Figure 6.36: Output voltage over time after disabling the refresh process. Traces are shown for the three accessible voltage cells of chip 2.2. (0,13) is shown in red, (0,14) in green and (0,15) in blue. The cells were initially programmed to the digital codes `8`, `511` and `960`. The voltage traces are calibrated to compensate for the offset caused by the individual output amplifiers.

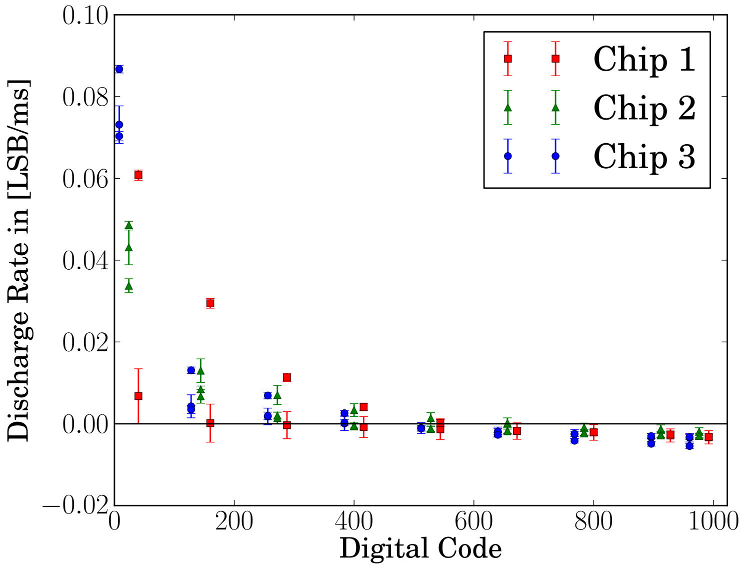


Figure 6.37: Drift of the output voltage over digital codes for the voltage cells of the three chips. The error bars indicate the uncertainty for the individual data points.

loaded into the memory of the Sourcemeter, it is executed by the internal microprocessor. At each falling edge of the `cm_trigger` signal, indicating the end of refresh cycles, the following measurement procedure is triggered. After an trigger delay of 1 ms a first measurement of the output voltage with an integration time of 2 ms is performed. Next, after a delay of 18 ms, a second measurement of 2 ms duration is taken. The effective distance between the two measurement intervals is 20 ms. The procedure is repeated 380 times for each cell and different digital codes. Evaluating the average difference between first and second measurement and dividing it by the effective time interval allows to determine the average drift of the cells' output voltage.

The results are presented in Figure 6.37. For all accessible voltage cells on the three test chips the drift of the output voltage is measured for nine different digital codes, covering the full dynamic range. The error on the individual measurements includes two components. The statistical uncertainty of the average slope can be obtained from the standard deviation observed for the 380 repetitions. Additionally the systematic uncertainty of the timing generated by the microprocessor in the Sourcemeter has to be taken into account. According to the documentation of the device, a systematic deviation of 100 μs has to be assumed for the interval between the two measurements. Consequentially an additional error of 0.5 % has to be added to the statistical error of the slope.

For low digital codes, corresponding to low output voltages, the cells drift towards a higher voltage level at a rather high rate. This behavior is consistent with simulation results, the leakage through the switch transistor dominates the drift of the cell if low voltages are stored, see 6.3.3. For higher voltages the drift rate decreases until its sign changes at digital codes around `511`. At this point an equilibrium between the reversed bias current through the substrate diodes of the transistors and the leakage through the switch is reached. For even larger digital codes the absolute drift rate slightly increases, the stored voltages are drifting towards lower values. As long as voltages below 200 mV are avoided, the measured drift rates allow for breaks of several milliseconds between the refresh cycles of the programming system. The equilibrium state of the voltage cells observed in measurements is different from what is expected from simulations where the cells drift towards 170 mV.

Current Cells

An overview over the drift of the current cells is given in Figure 6.38. The output current traces of 9 randomly chosen current cells over time, after disabling the refresh process, are shown. The cells have been initially programed to the digital codes `0`, `512` and `960`. Before stopping the refresh cycles, the output current is measured multiple times to get an accurate start level I_{init} for the individual cells. This initial output current is used to compensate the cell-to-cell variation introduced by the device mismatch of the output transistors. All current cells shown in the Figure drift towards an output current close to 0. This behavior is consistent with simulations, where the voltage on the storage capacitors drifts towards a level of 2.3 V, resulting in an output current below 1 pA. However, a wide spread in the drift rates of individual cells is already observed in this small subset of cells.

In order to obtain quantitative results for the drift rate of the current cells the same setup as used in case of the voltage cells is used. The break between two refresh cycles is increased to 26.6 ms and the `cm_trigger` signals is used to trigger a measurement procedure controlled by the microprocessor in the Sourcemeter. 1 ms after the falling edge of `cm_trigger` two measurements with a distance of 20 ms are taken. The integration time for each measurement is 2 ms. Again the difference of the two measurements is used to determine the slope of the output current signal.

The results are presented in Figure 6.39. The average drift of all cells is shown for the three test chips, the error bars denote the standard deviation over the results for the individual cells. The average error on a single measurement, based on the statistical error of the measured voltage difference and the systematic uncertainty of the time interval, is 0.003 LSB/ms. The drift rate for all cells is negative, the outputs are drifting towards lower currents. The absolute drift rates are larger than observed for the voltage cells.

6.4.7 Temperature Dependence of the Storage Time

All measurements presented so far have been performed at room temperature. The measurements for the leakage current caused by an array of SRAM cells, presented

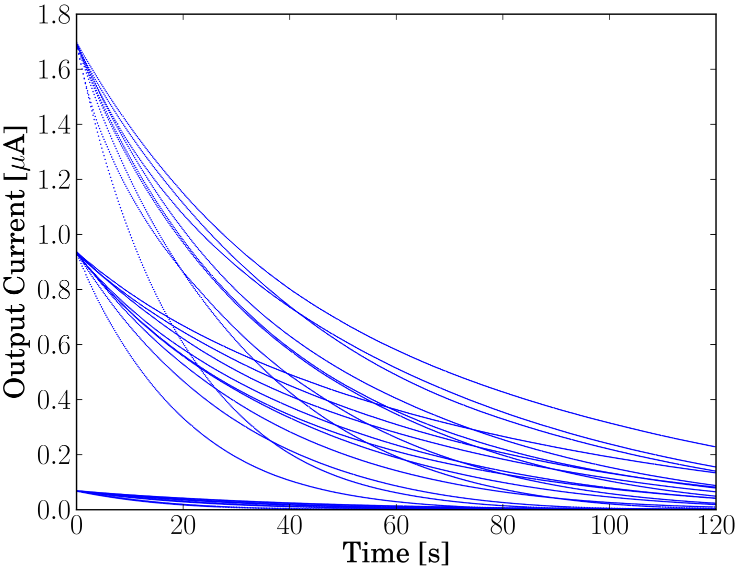


Figure 6.38: Output current over time after disabling the refresh process. Traces are shown for 9 randomly chosen current cells of chip 2.3. The cells were programmed to the digital codes `8`, `512` and `960`. The output voltages are calibrated to compensate for the device mismatch of the output transistors.

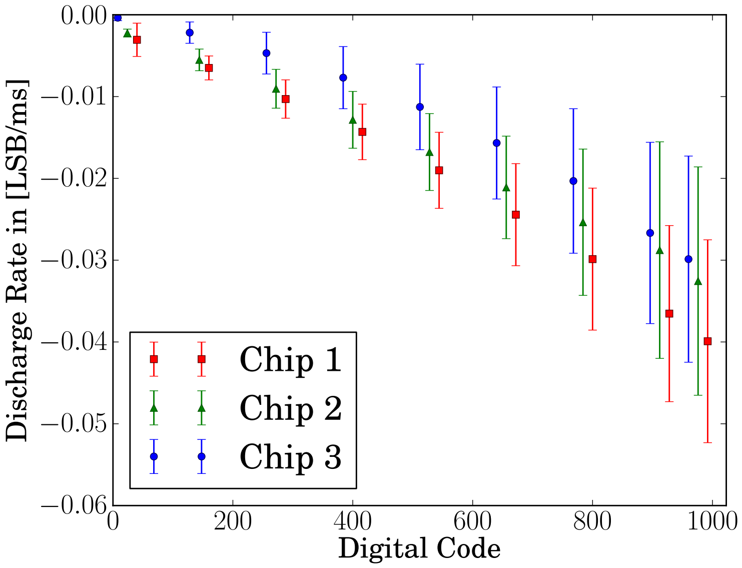


Figure 6.39: Average drift rate of the output current over digital codes for all current cells of the three test chips. The error bars indicate the standard deviation over the results from all cells within the same chip. The average error of the individual measurements is 0.003 LSB/ms.

in Chapter 5.2, have shown a significant temperature dependence. It can be assumed that the storage time of the capacitive memory cells is also affected by temperature change. In order to investigate the effect, the PCB carrying the test chip 2.2 is placed in a climate chamber. The protocol described in Section 6.4.6 for measuring the drift rate of the current cells is repeated at different temperatures. The results are presented in Figure 6.40. The drift rate of the cells increases significantly with higher temperatures.

Figure 6.40 shows the average drift of the current cells for different output currents and temperatures. The error bars, showing the standard deviation over all cells, indicate that there is a significant cell-to-cell variation. The minimum refresh frequency which is required to operate the system, is limited by the cell with the largest drift rate. Figure 6.41 shows a histogram over the drift rates for all cells of chip 2.2 at a temperature of 50 °C and programmed to a digital code of 976. This configuration can be assumed to be a worst case scenario regarding the drift of parameters. The highest absolute drift rate observed is 0.32 LSB/ms, this number has to be used when estimating the minimum refresh rate required to achieve a certain accuracy.

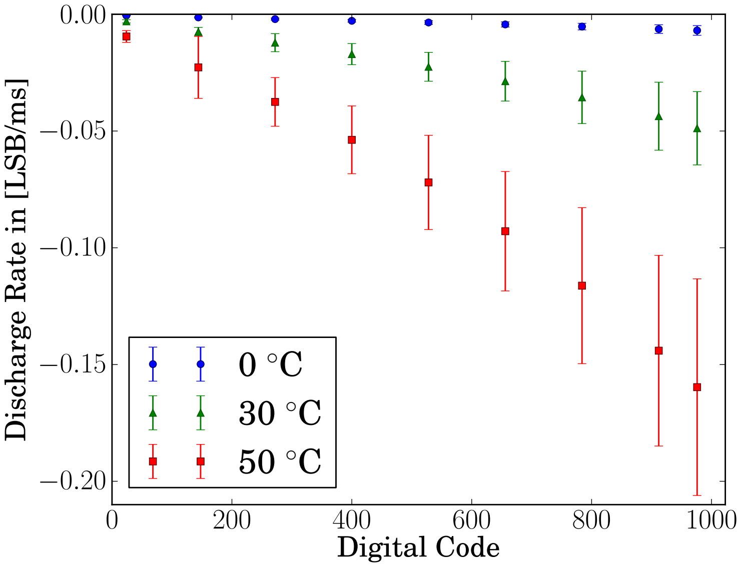


Figure 6.40: Drift rate of the output currents over digital codes for the cells from chip 2.2, measured at different temperatures. A significant temperature dependence of the drift rate is visible.

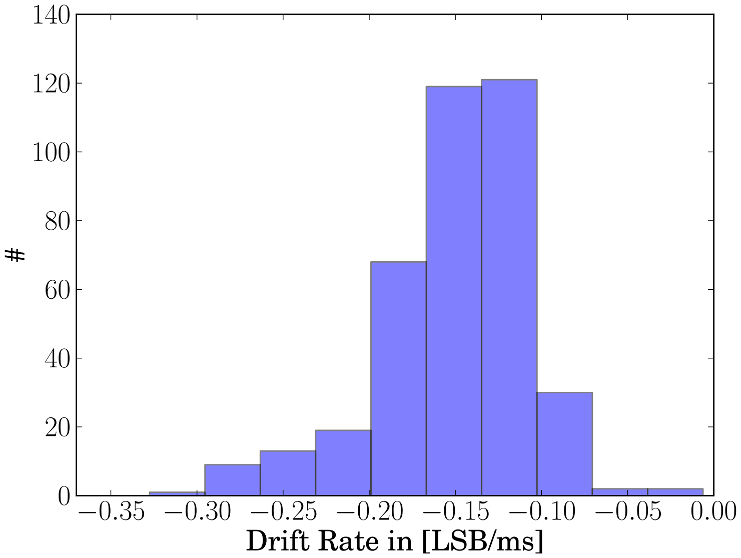


Figure 6.41: Histogram of the drift rate for all current cells of chip 2.2 at a temperature of 50 °C and a digital code of `976`. The highest absolute drift rate observed is 0.32 LSB/ms.

6.4.8 Reprogramming the Target Value of a Cell

As discussed in 6.3.5 the output of a cell cannot reach a new target value within a single refresh cycle. Due to the sequential arrangement of the two storage capacitors in the cell an exponential approximation towards the new target value is expected. The timescale at which this is happening is important for the overall setup time, required by the parameter storage system to produce stable output signals after initialization.

Voltage Cells

The setup time of the cells is determined by the ratio between the sizes of two capacitors used in the cell. Since there are no buffering circuits integrated into the voltage cells, the capacitance connected to the output of the cell adds to the capacitance of the second capacitor C2, see Section 6.10. The three voltage cells on the prototype chip that can be tested are directly connected to the inputs of output amplifiers. These have a rather large input capacitance of about 42 fF. The transistors at the input of the amplifier are larger than any of the transistors which are connected to voltage parameters in the design of the neuron circuit used in the BrainScaleS Hardware System. Therefore the results obtained from measurements can be considered to be a worst case scenario regarding setup time of the voltage cells. For the measurements shown in the following the default timing for the programming process is used. The period of the programming cycle and the subsequent break is 0.82 ms.

Figure 6.42 shows the output voltage of a single voltage cell during reprogramming, the trace is recorded using an oscilloscope. The red trace shows a reprogramming from the maximum output voltage down to a digital code of `8`.

This plot does not allow for a precise measurement of the number of cycles required by the cell to minimize the distance to its target value to less than 1 LSB. However, it can be estimated that this point is reached after about 20 ms. The number of programming cycles required in the worst case to change the output voltage of a cell to a lower value is therefore assumed to be 25. The two blue traces show a reprogramming process increasing the output voltage of the cell. In case of the upper trace, the cell is programmed to a target value of `1020`. The output does not reach a stable value within 45 ms. This is not only an effect of the architecture using two capacitors but it is also caused by the limited conductivity of the NMOS switch transistors at high voltages, see Section 6.4.2. The lower blue trace shows the programing towards a digital target value of `767`. Here the programming is not affected by limited transistor conductivity and again an interval of 20 ms, respectively 25 programming cycles is required for the change of the output voltage. However, the numbers given are just estimations based on the oscilloscope recording of a single cell.

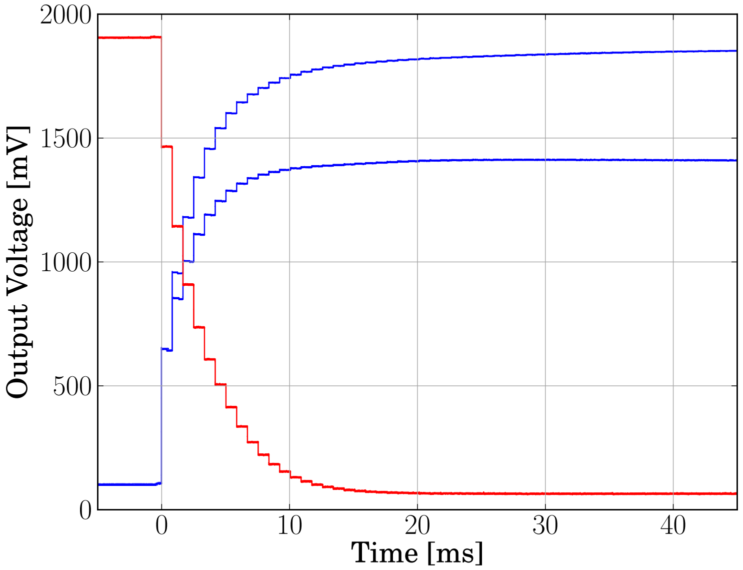


Figure 6.42: Oscilloscope recording of the output voltage of cell (0,13) from chip 2.3. The red trace shows reprogramming of the cell from a digital code of `1020` to `8`. In blue the traces for reprogramming the cell from `8` to `1020` and to `767` are shown. To reduce the level of random readout noise each trace shows the average over 10 trigger events.

Current Cells

To measure the currents generated by the parameter storage system with an oscilloscope, the output currents have to be converted to a voltage. Therefore the pin `iout<0>` is connected to ground using a resistor with a nominal resistance of $R_{ext} = 240\,\mathrm{k}\Omega$. The voltage over the resistor is recorded by the oscilloscope. To convert the measured voltage to the corresponding current the input resistance of the oscilloscope, specified to be $1\,\mathrm{M}\Omega$, needs to be taken into account. The total resistance of the parallel connection of R_{ext} and the input voltage of the oscilloscope has been measured directly using the Sourcemeter. This results in $R_{conv} = 193.6\,\mathrm{k}\Omega \pm 0.5\,\mathrm{k}\Omega$, this value has been used to convert the measured voltages to the corresponding currents for the following plots. The maximum output current of $2\,\mu\mathrm{A}$ leads to a voltage drop of about 400 mV. As a result the drain-source voltage of the output transistors in the cells is always above 2 V, the transistors can not drop out of the saturation regime.

Figure 6.43 shows the output current for cell (22,6) of chip 2.3 during reprogramming. The red trace shows the process of programming the cell from a digital code of `1020` down to `8`. Since the capacitance at the output of the cell does not affect the ratio of the internal capacitors, the programming process is happening significantly faster than for the voltage cells. The new target value is reached within 10 ms, less than 15 programming cycles are required. As for the recording of the voltage cell described above, this number is only an estimation.

However, the first programming cycle is obviously not decreasing the output current, instead it is increased. This behavior is caused by a mistake in the digital controller of the parameter array. During the reprogramming of the SRAM the counter has to be stopped. In the current implementation of the digital controller the `reset` signal of the *ramp_gen* circuit is not activated during this break. As a consequence the current I_{ref} is charging the voltage V_{ref} to the maximum possible value. Accordingly V_{Gate} is at the lowest possible value and the current produced in the *VI_conv* circuit is at its maximum during the first programming cycle. The reset after the first cycle discharges the capacitor in the *ramp_gen* circuit and allows for normal operation in the following cycles. The issue is discussed in more detail and along with other reset related problems in Section 6.5.4.

The blue trace in Figure 6.43 shows the programming of the cell from `0` up to `1020`. In this case it takes about 20 ms, respectively 25 cycles until the target value is reached. For the current cells a faster change of the output, compared to the voltage cells, has been expected since the capacitance at the output side of the cell is lower. Obviously other effects than just the ratio between the capacitors are affecting the setup time of the cells in case high output currents need to be reached. As the conductivity of the switch in the current cell is not limiting the process, this effect is observed for currents larger than $2.4\,\mu\mathrm{A}$, further investigation is required.

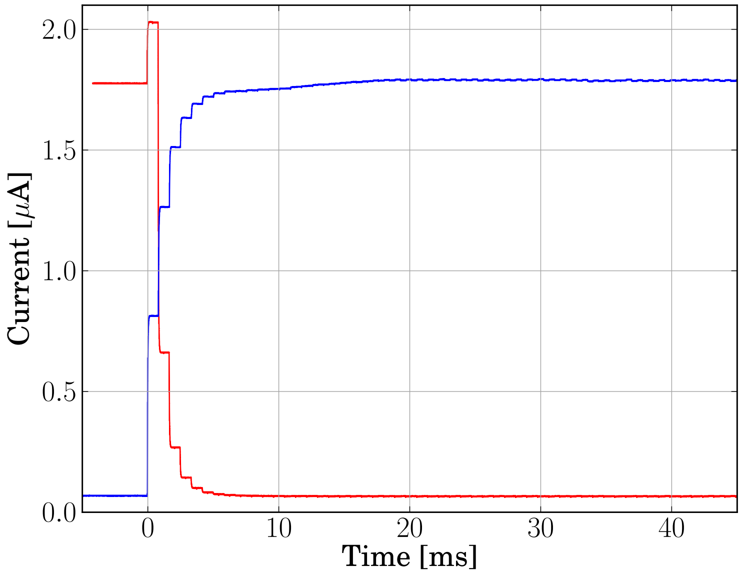


Figure 6.43: Oscilloscope recording of the output current of cell (22,6) from chip 2.3 during reprogramming. The trace for programming the cell from a digital code of `1020` to `8` is shown in red, in blue the traces for programming the cell from `8` to `1020` is shown. To reduce the level of random readout noise the trace shows the average over 10 trigger events.

6.4.9 Transient Distortions

In 6.4.5 measurements regarding the variation of the output signals of the storage cells after individual programming cycles are presented. The output signal of several cells has been recorded using an oscilloscope to investigate the output characteristics of the parameter storage cells during the refresh process.

Voltage Cells

For each of the voltage cells the stored value is refreshed periodically during regular operation of the system. It is important to isolate the output of the cells against noise introduced by the operation of the switches during the refresh process. Figure 6.44 shows the output voltage of the three voltage cells accessible on chip 2.3 over one counter cycle. The falling edge of the `cm_trigger` signals has been used to trigger the oscilloscope. To suppress the impact of random noise the traces show the average over 100 trigger events. The cells are programmed to a digital code of `511`. To compensate the offset introduced by the individual output amplifiers the average output voltage of every cell has been subtracted.

A significant amount of crosstalk from the voltage ramp V_{ref} to the output of the cells is visible. Capacitive coupling through the switch transistors can not be responsible for the effect, see Figure 6.10. In the voltage cells, the node connecting T1 and T3 is pulled to the supply voltage by T2, isolating V_{ref} from the voltage stored on the capacitors C1 and C2. To avoid capacitive coupling to the internal voltages of the cells, the wires distributing V_{ref} over the array have been shielded by metal structures connected to ground in the layout. The crosstalk effect is more severe for cells (0,14) and (0,15) where the amplitude of the distortion reaches 3 mV. Cell (0,13) is significantly less affected. The effect is systematic, the same pattern is visible for the other test chips. In the layout of the array there is no asymmetry that explains the different behavior observed for the three cells. Outside of the memory array however, the wires connecting the output voltages of the cells (0,14) and (0,15) to the readout amplifiers are running in parallel to the wire distributing V_{ref} along the edge of the array. Despite the fact that the length of the parallel segments is only about 32 μm, this is the most probable explanation for the severe crosstalk observed for the cells (0,14) and (0,15). This shows how sensitive the high impedance voltage signals are and emphasizes the necessity of proper shielding for the wires connecting the voltage cells to the neuron circuits. For cell (0,13) the readout wire cannot be affected by crosstalk, as it is routed in large distance to the net distributing V_{ref}. The source of the crosstalk observed for cell (0,13) is not yet identified. Probably the shielding in the layout of the cell needs to be improved.

During the rise of V_{ref}, an additional, regular pattern is visible on the voltage trace. This is probably caused by capacitive crosstalk from the digital counter signals. In the layout the analog part of the cell is shielded against the digital counter signals running on `M4` by wide metal wires distributing power and ground on layer `M3`. Furthermore every counter signal $\mathrm{C_x}$ is routed in parallel to its inverted

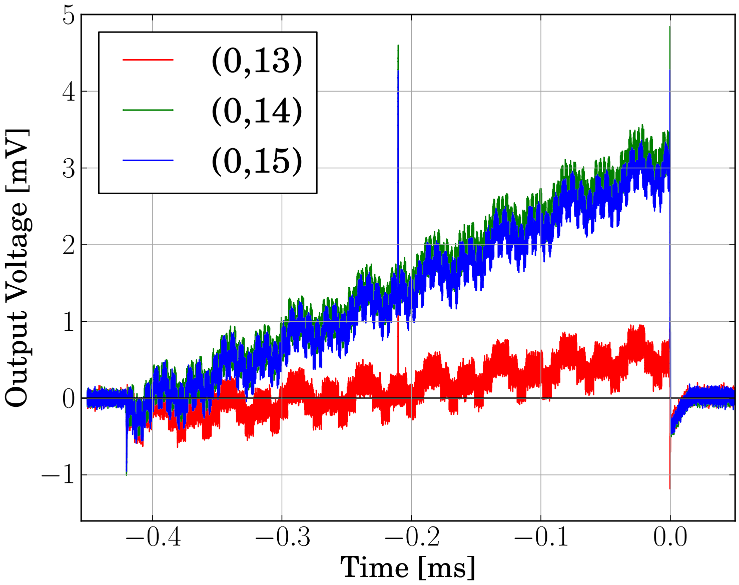


Figure 6.44: Oscilloscope recording of the output voltage of the three voltage cells on chip 2.3 during a refresh cycle. The cells are programmed to a digital code of `511`. To eliminate the offset caused by the individual output amplifiers, the average output voltage for each cell has been subtracted. For all three cells significant crosstalk from the reference voltage V_{ref} as well as the counter signals are visible. Furthermore the actual refresh process, occurring at -0.21 ms, is also clearly visible. To reduce the impact of random noise, each trace shows the average over 100 trigger events. See text for further details.

counterpart $\overline{\mathrm{C}}_{\mathrm{x}}$ to reduce the impact of crosstalk on other nets. Nevertheless the impact of the counter signals is visible. Due to the fact that the amplitude of this distortions is below 0.5 LSB the effect is not considered a problem.

Another significant distortion visible in 6.44 is caused by the update process of the cell itself. This happens at -0.21 ms, a peak with a height of about 3 mV is visible for all cells. This is considered a serious problem, it seems the compensation for the charge injection of the switch transistors is not working as effective as expected. Canceling the charge injection from the programming process requires precise balancing of the size of the switching transistors. The optimum sizes for the transistors also depends on the capacitance on the output side of the cells. In simulations, also when including the output amplifier at the cells output, the impact of the refresh process is well below 1 mV. So far the source of this discrepancy has not been identified and further investigation is required.

Current Cells

Figure 6.45 shows the output current for three current cells from chip 2.3. The cells are programmed to a digital code of `511`, the average output current has been subtracted in order to compensate for systematic cell-to-cell mismatch. Similarly to the situation described for the voltage cells significant crosstalk from the programming voltage V_{Gate} is visible. The output current increases by almost 3 nA, corresponding to 1.7 LSB, during the programming cycle. The three cells are affected to the same degree, despite the fact that they are located at different positions in the array. The global readout line is running in a significant distance to any wires distributing V_{Gate} outside the array. Both facts indicate that the source of the crosstalk is within the array. As discussed for the voltage cells, crosstalk through the parasitic capacitances of the switch transistors can be ruled out. The voltage stored on the capacitors is shielded against the wire distributing V_{Gate} by additional wires connected to ground in the layout. Probably the shielding of the wires distributing V_{Gate} within the array needs to be improved.

The reset of the programming voltage is causing the most prominent distortion. To avoid this effect the reset mechanism in the *ramp_gen* circuit can be modified. Avoiding the large transients on the signals V_{ref} and V_{Gate} caused by the fast reset process can help to decrease the overall noise level in the system. A more detailed discussion on improvements of the reset mechanism can be found in 6.5.4.

The update process is suppressed very well, at -0.21 ms no significant distortion can be observed, neither in the picture shown nor when zooming closer into the time interval in question. The high frequency noise that is visible on the traces does not change depending on whether the counter is running or not. However the source of the noise has to be synchronous to the programming process, otherwise it would have been suppressed by the averaging over multiple trigger events. The main clock of the chip, constantly operating at 100 MHz, possibly introduces the noise at some point in the experimental setup.

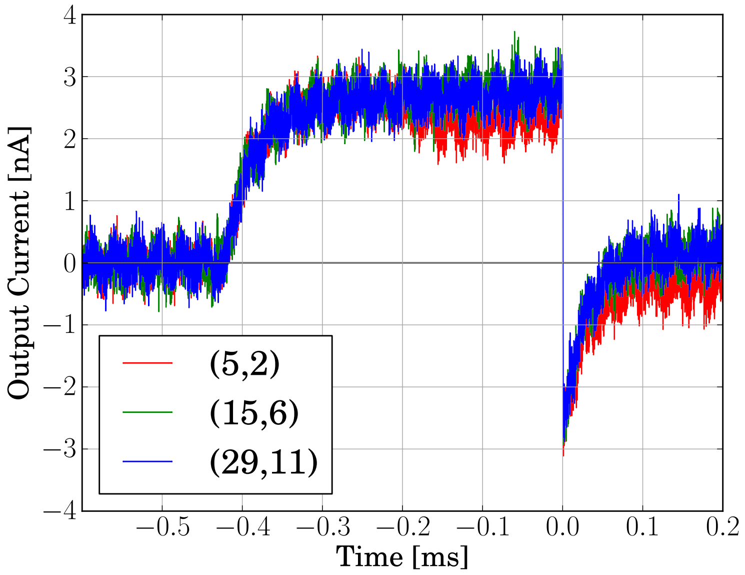


Figure 6.45: Oscilloscope recording of the output current of the three current cells of chip 2.3 during a refresh cycle. The cell is programmed to a digital code of `511`. To eliminate the offset caused by device mismatch of the output transistor, the average output current for each cell has been subtracted. For all three cells significant crosstalk from the programming voltage V_{Gate} is visible. The update process for the cell itself, occurring at -0.21 ms, is not visible. See text for further details.

6.4.10 Power Consumption

The power consumption is an important performance figure for the capacitive parameter storage system. Using the prototype chip it is not possible to measure the power consumption of the capacitive memory system directly as other circuits are connected to the same power supply nets. However it is possible to configure all components of the capacitive memory system into an inactive state in which their power consumption is negligible. This option allows to measure the baseline current consumption of the chip and compare it to the current consumption measured for an operating parameter storage system. In order to measure the power consumption including the active storage system, the standard parameters described in Section 6.4.2 are used. One cycle of the voltage ramp is configured to last 409.6 μs, followed by a break of the same duration. The resulting refresh rate for the cells is 1.2 kHz. In the following, measurement results for the current consumption for each of the individual supply voltages of chip 2.3 are presented. The Sourcemeter is used to generate the supply voltage and simultaneously measure the current drawn by the chip.

Digital Supply

The parameter system uses both of the digital supply nets, VDD12D and VDD25D. The VDD12D supply is required by the logic generating the counter and the buffers which are distributing the counter signals to all cells in the array. Inside each cell the logic comparing the counter to its internally stored target value is also supplied by this net. The VDD25D supply is only required by the pair of levelshifter circuits implemented in each cell, converting the programming signals `A` and `B`. The baseline consumption of the chip on the digital supplies can be measured by configuring the parameter storage system into the reset state. The counter is disabled and all digital signals in the system are static. The difference in the current consumption between active and inactive parameter storage system is measured to be 72.9 μA ± 0.5 μA for the VDD12D supply. For the VDD25D supply a value of 0.3 μA ± 0.2 μA is obtained. The errors are estimated, based on three repetitions of each measurement. The results are also summarized in Table 6.4.

Analog Supply

From the two analog supply nets, VDD12A and VDD25A, only the latter is required by the parameter storage system. It is used by the current cells to generate the output current, see Section 6.3.2. In the prototype chip, the output current can only be generated in cells which are connected to the shared readout line by their output multiplexer. The contribution of the actual output currents to the overall power consumption of the system depends on the values programmed to the cells and is well known. For the measurements presented in the following the multiplexers of all current cells are disabled. Therefore the output currents do not contribute to the

Supply Net	I $[\mu A]$	ΔI $[\mu A]$	$Power$ $[\mu W]$	$\Delta Power$ $[\mu W]$
VDD12D	72.9	0.5	87.5	0.6
VDD25D	0.3	0.2	0.8	0.5
VDD25A	100.5	0.5	251.3	0.6
Total:			339.6	1.0

Table 6.4: Power consumption of the parameter storage system on the prototype chip. The results were obtained by comparing the power consumption of the chip for an active and an inactive parameter storage system.

measured power consumption. The VDD25A supply is further used by the circuits *ramp_gen* and *VI_conv*, generating the programming voltages V_{ref} and V_{Gate}.

To measure the baseline current consumption of the chip for the analog VDD25A supply, not only the reset needs to be activated but several more steps are required. As mentioned before, the multiplexers for the output currents are disabled, preventing the cell from drawing any current. The bias current of the amplifiers used in *ramp_gen* and *VI_conv* is generated by a PMOS transistor. It can be cut off by setting the respective bias voltage to 2.5 V. Furthermore the active reset signal stops the current I_{cap} in the *ramp_gen* circuit. In the *VI_conv* the value of resistor R1 is configurable. The current through T1 and R1 can be completely disabled by configuring all switches that connect the single elements of R1 to ground into the `off` state.

For a measurement of the current consumption including the active storage system, the counter is configured the same way as mentioned above is. The bias current of the *ramp_gen* circuit is configured such that the peak voltage of V_{ref} is 1.85 V. The bias voltage of the operational amplifiers is set 1.6 V. All voltage and current cells are programmed to a digital code of `511`. Since no output currents are produced the current consumption is assumed to be independent of the values stored in the cells. Comparison of the two measurements shows a difference of 100.5 μA ± 0.5 μA in the power consumption. The error is again estimated from three repetitions of the measurement. Table 6.4 summarizes the results obtained for all of the individual supply nets.

Estimating the Power Consumption of a Large System

The parameter storage system in the prototype chip features an array of 32 × 24 cells. The layout is organized such that each of the 32 columns can be attached to a neuron circuit. However, in future neuromorphic systems larger numbers are required for each chip. The HICANN chip used in the current BrainScaleS Hardware System features in total 512 neuron circuits. In the following an estimation for the power consumption of a system featuring 512 × 24 individual parameters is given.

Multiplying the power consumption measured for the prototype chip by a factor of 16 allows to determine an upper boundary for the power consumption of a system featuring 512 × 24 cells. This results in a total estimated power consumption of 5.4 mW. However, the observed power consumption is partially caused by components which are required only once per array, regardless of its size. Therefore this calculation overestimates the actual consumption for the system.

In order to obtain more accurate data it is necessary to estimate which fraction of the power consumption is caused directly by the storage cells and needs to be scaled and which fraction is independent of the arrays size. For the VDD25D supply the situation is clear. It is only used by the cells, the corresponding current consumption scales linearly with the size of the array. For the VDD12D supply the situation is different. It is required by the counter, the buffer circuits distributing the counter signals through the array and the individual cells. The consumption of the buffers and the cells can be assumed to scale linearly with the size of the array. The counter is required only once per array, regardless of its size. However, the power consumption of the small number of logic gates required to realize the counter is assumed to be negligible compared to the power consumed by the buffers driving the counter signals over large distances. Therefore linear scaling leads only to a small overestimation of the current consumption on the VDD12D supply.

Since the output currents of the array are not taken into account, the VDD25A supply is only used by the supplementary circuits *ramp_gen* and *VI_gen*. These circuits are required only once per array, but the load connected to the amplifiers which buffer V_{ref} and V_{Gate} depends on the size of the connected array. In fact these buffers are responsible for the largest part of the power consumption of the supplementary circuits. In *ramp_gen* the only other circuit consuming power is the current mirror generating the current I_{cap}, see Figure 6.12. In normal operation with the standard settings, I_{cap} equals to about 4 nA, the mirror consumes in total five times that value. During the breaks in between the ramp cycles, I_{cap} is disabled by transistor T6. For the timing configuration used, the average consumption of the mirror generating I_{cap} is 10 nA, its contribution to the overall consumption of the supplementary circuits is negligible. The *VI_conv* generates a linearly increasing current, flowing through T1 and R1, see Figure 6.14. Since T1 is 32 times wider than the output transistor of the current cells, the current ramp ranges from 0 to about 64 μA. Accounting for the breaks between the refresh cycles, the average current flowing trough this path is 16 μA. This fraction is consumed only once per array, regardless of the number of cells.

These considerations suggest that the average current consumption of the buffers in the supplementary circuits is 84.5 μA. This number corresponds well to the value of 45.5 μA obtained from simulations for a single buffer at a bias voltage of 1.6 V, see Section 6.4.2. It is not possible to reliably estimate the power consumption of the operational amplifiers required for a large scale parameter array. The buffers currently integrated into the system are small two-stage Miller OTAs, which are not suited to drive large loads. The power efficiency of the supplementary circuits can be increased significantly by using amplifiers featuring a class-AB output stage [Callewaert and Sansen 1989]. However, this aspect needs further investigation.

Supply Net	I $[mA]$	$Power$ $[mW]$
VDD12D	1.12	1.34
VDD25D	0.005	0.013
VDD25A	0.69	1.73
Total:		3.08

Table 6.5: Estimated power consumption of a parameter storage system featuring 512 × 24 cells and operating at a refresh rate of 1.2 kHz. The numbers are derived from measurement results obtained from the smaller system included in the prototype chip.

Another possibility to optimize the efficiency, which is not implemented in the prototype chip, is to disable the buffers during the break in between the refresh cycles. Applying linear scaling for the consumption of the buffers but considering the fact that their consumption can be reduced by at least a factor of two if they are disabled during breaks, their power consumption in the full scale system can be estimated to be 676 μA. Adding the fraction required by the generation of the linearly increasing current ramp, the power consumption on the VDD25A supply of a full scale system is about 692 μA.

Table 6.5 summarizes the results of the estimations for a system providing the parameters for 512 neuron circuits. The estimated total power consumption is 3.1 mW. However, this number is based on measurements in which the refresh rate of the system has been set to 1.2 kHz. The power consumption is assumed to drop almost linearly with an decrease in the refresh rate, during the break in between to cycles the system is hardly consuming any power if the amplifiers are disabled. The measurements presented in Section 6.4.6 and Section 6.4.7 suggest, that even at temperature of 50 °C the storage time of the cells is sufficient to allow for a reduction of the refresh rate to 0.25 kHz. This setting leads to an estimated power consumption of less than 1 mW for a system providing programmable voltage and current sources to 512 neuron circuits.

6.4.11 Defective Cells

So far only the results of cells considered to be working correctly have been presented. Variation in the analog performance is observed and single samples that deviate significantly from their target values are present for some cells. These limitations are all consistent with causes like device mismatch, crosstalk or related to the error in *pulse_gen* circuit generating the programming pulses. However, during testing a small group of cells which have to be considered completely non functional has been identified.

Voltage Cells

Only a limited number of voltage cells can be tested. From the 9 voltage cells usable on test chips, it is not possible to measure correct output voltages for cell (0,14) of chip 2.1. The output voltage measured for this cell is constant, regardless of the value that is programmed to the cell. Further the absolute value of this constant voltage depends on the impedance connected to the according bond pad. This behavior suggests that the amplifier buffering the output voltage of the cell is not working. Otherwise the output impedance of the chip would be low enough that the measured output voltage is level is not dependent on the load impedance. At least not for load impedances variations in a range above 1 MΩ. It is not possible to verify whether the connected voltage cell itself is working correctly or not. Further, there is no information on what might have caused the failure of the amplifier. Since its output is connected to a bond pad, damaging by external causes such as an ESD event is possible. But also a defect of a single transistor or wire involved in providing the bias current for the amplifier can explain the behavior observed.

Current Cells

For the current cells systematic tests for all individual cells in the test chips are possible. A number of cells have been identified that are affected by a defect in their digital part. Figure 6.46 shows the output current over digital codes for two cells belonging to that group. Comparing the measured output characteristics to a table of the Gray coding used for the counter suggests that for these cells a single bit in the digital part is stuck at a fixed value. In the intervals of digital codes where the output current is changing with the correct slope, the stuck bits value is the same as its target value. For those intervals at which the output current is changing with the wrong slope, the stuck bit is in the wrong state. Checking which bit changes in the Gray code for the digital numbers at which wrong behavior of the output current occurs allows for identification of the stuck bit and the value it is stuck at. In case of Figure 6.46(a) the output current is correct for the digital codes `0` to `63`, but for `64` to `127` the output current decreases again. In the Gray code used, the 7th bit is supposed to flip from `0` to `1` at a code of `64`. All discontinuities observed are consistent with the situation of the 7th bit of the cell being stuck at `0`. In case of Figure 6.46(b) the 4th bit is stuck at a value of `1`. Similar problems are observed for in total 6 cells on the three test chips that have been scanned for broken cells.

In each cell there are two different components involved in the processing of single bits. The SRAM bit itself and the following comparison logic, see Figure 6.11. The best option to identify the source of the problem would be to read the content of the SRAM of a broken cell. However, this is not possible in case of the capacitive memory cells. As described in Section 6.3.4 the precharge state, required to initiate the reading process of the SRAM, causes the comparison gate to short the internal state of the SRAM cells, deleting the stored information.

Cell (6,1) on chip 2.3 is also affected by failure of a single bit, but in this case

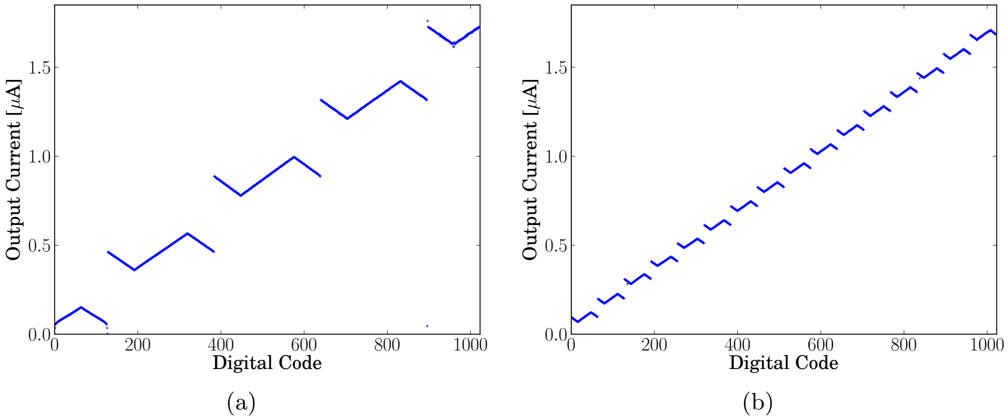


Figure 6.46: Output current over digital codes for defective current cells. (a) shows cell (15,2) on chip 2.1, comparing the output current to the Gray coding used suggests that the 7th bit is stuck at `0`. (b) shows cell (20,10) on chip 2.1, here the 4th bit is stuck at `1`.

the faulty bit is not stuck but rather instable. The typical protocol used to record sweeps over digital codes for single memory cells includes resetting the cell to `0` in between the single measurements to obtain independent results. The full procedure is repeated at least 16 times in order to increase the measurement precision and to determine the standard deviation of the results. Applying this measurement protocol to cell (6,1) reveals only small deviations from ideal behavior in the output characteristics. If the measurement protocol is changed such that the cell is programmed directly to its target value and 16 measurements are taken without any intermediate reprogramming of the cell, the result changes significantly. Figure 6.47 shows the output current over digital codes and the corresponding residuals for this situation. At most of the digital codes the output current has the correct value, all bits in the digital part are evaluated correctly. Most interesting are the few samples for which the output current deviates significantly from the expected value. The periodicity suggests that the 4th bit is causing the problem. It is not stuck at `1` continuously but it shows an unstable behavior. For single digital codes the bit is `1` when it is supposed to be `0`.

The different results for experiments in which the SRAM is reprogrammed prior to each measurement and experiments that program the memory only once indicate that the SRAM bit is unstable and changes its state during the measurement procedure. This becomes evident from the raw measurement data recorded for the data points which show a large standard deviation over the 16 individual measurements of the output current. In these cases the output of the cell is correct for the first measurements, followed by a single transition towards a wrong output current. No occurrence of the current flipping back to the correct value has been observed.

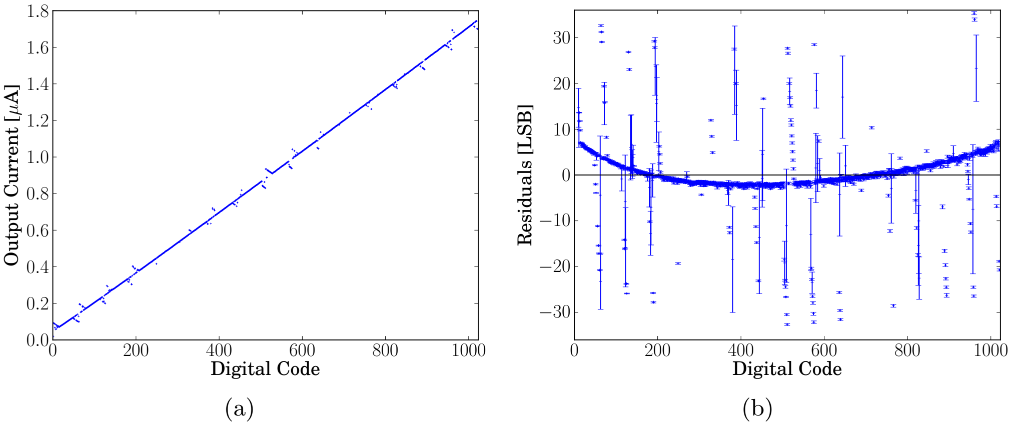


Figure 6.47: (a) shows the output current over digital codes for cell (6,1) from chip 2.3, (b) shows the corresponding residuals relative to a linear fit. The 4th bit shows an unstable behavior. For some samples where it is supposed to be `0` it is `1` instead. The error bars indicate the standard deviation over 16 repetitions of the measurement. The large error bars of some samples show that the bit flips in between the single measurements.

The results obtained from testing cell (6,1) on chip 2.3 have shown that the problem of single bit failure observed for multiple parameter storage cells originates from a wrong value stored in the SRAM. Since the basic SRAM design has been proven to be stable in previous measurements, see Section 5.1.4, this can only be caused by the comparison logic, shortcutting the internal state of the cell. This scenario is happening when reading the memory. In the precharge phase both bitlines are set to `1` and the data in the cells is deleted. In normal operation the counter signals `C` and $\overline{\texttt{C}}$ should be always inverted to each other. However this is not necessarily true during a transition from one counter state to the next. If the two signals do not change their state simultaneously, both can be high for a short time interval. This interval might be sufficient to delete the content of individual SRAM cells.

To investigate this effect in simulation, a delay between the counter signals `C` and $\overline{\texttt{C}}$ is introduced in the simulation testbench. It is set to 200 ps, according to the timing verification tool used for the digital part this is the highest value that can be expected. Still, no failure of the SRAM cells was observed, within 200 ps the content of the cells is not destroyed. However, single cells of SRAM being very sensitive to the shortening over the comparison logic are the most reasonable explanation for the incorrect operation of observed for single cells in the capacitive memory.

For the three test chips a quota of more than 0.5 % of cells is affected by this problem. The uncertainty of this percentage is rather large as the absolute number of broken cells and chips that have been tested is low. Since every neuron circuit features more than 20 parameters, all of which have to be working correctly, a fraction of more than 0.5 % defective parameter storage cells is not acceptable.

As mentioned Section 6.3.4 the current implementation of the comparison logic also prevents reading accesses to the SRAM. In the next revision of the parameter storage system the comparison logic has to be changed, to allow for reliable SRAM operation. An option how this can be realized by only minor changes to the current design is presented in Section 6.5.1.

6.5 Modifications for the Next Revision

During the experiments various problems that limit the performance of the cells have been identified. Some are caused by mistakes in the circuits, the behavior of the actual implementation is different from the intended behavior. Furthermore several possibilities for improvement of the performance of the system have been identified. Some effects which have been observed need further investigation. However for the problems for which the source could be clearly determined, possible solutions will be presented in the following.

6.5.1 Comparison Logic

The first problem detected during tests of the digital part of the prototype chip was that the SRAM implemented into the storage cells can not be read correctly. This problem is caused by the comparison logic used to compare the content of the SRAM to the counter value in every cell. The precharge process initiating a read access causes the comparison gate to short the internal state of the SRAM, deleting the stored information, see Section 6.3.4. Later in the testing process, a small number of cells has been identified in which single bits are stuck at either `1` or `0`, see Section 6.4.11. The most plausible explanation for this problem is again that the SRAM is shorted by the comparison logic during an transition of the counter signals. Due to asymmetry introduced by device mismatch, the bit flips always to the same state and is effectively stuck.

The comparison logic needs to be improved to allow for robust operation of the system. Using a classic implementation of an XNOR gate in static CMOS logic would result in a significantly larger area consumption as it requires 14 transistors for every bit. The simplest solution, requiring only minimal changes to the current design of the cells, is to interchange the inputs of the comparison gate which is currently used. Figure 6.48 shows a schematic of this solution for the comparison logic for a single bit. Compared to the original design shown in Figure 6.11, the orientation of the transistors T1 and T2 is changed. In this case the gates of the two NMOS transistors are connected to the internal states of the latch in the SRAM bit. In static operation these signals are always inverted to each other, preventing the counter lines from being shorted. However, when reading the SRAM the voltage level inside the SRAM bit rises at the node that is representing a

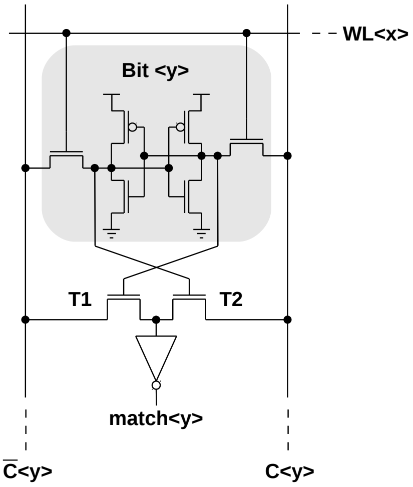


Figure 6.48: Schematic for a corrected comparison logic. The transistors T1 and T2 cannot short the internal state of the SRAM bit.

`0` when the wordline is activated and the charge stored on the bitline needs to be discharged through the SRAM bit, see Figure 5.2. Simulations show that the voltage at the `0` node of the SRAM bit is always below the threshold voltage of the NMOS transistors. An additional option to increase the robustness in this setup is to disconnect the logic gate from other signals during the SRAM access by adding transmission gates which are controlled by the wordline signal. Since this circuit is required 10 times in each parameter storage cell, it is important to use the most area efficient solution possible.

6.5.2 Update Pulse Generation

Every cell of the parameter memory compares the digital value stored in its local SRAM to the current value of the Gray counter that is distributed over the array. When a match is detected, the switch transistors need to be activated to perform an update of the stored analog value. As described in Section 6.3.2 the two signals `A` and `B` are required in this process. In the current implementation these two signals are generated by the *pulse_gen* circuit described in Section 6.3.4. In Figure 6.16(b) the timing that is generated by the circuit is shown, the signals `A` and `B` are activated longer than intended. As a consequence crosstalk between different cells in the array can be observed, see Section 6.4.2.

A simple option to generate `A` and `B` with correct timing is described in the following. First of all, the timing of the `match` signal generated by the comparison logic of the cell is identical to the behavior required for signal `A`, it can be used directly to trigger the first stage of the switches. For `B` a pulse shorter than the positive phase of the `match` signal has to be produced. To realize this, without im-

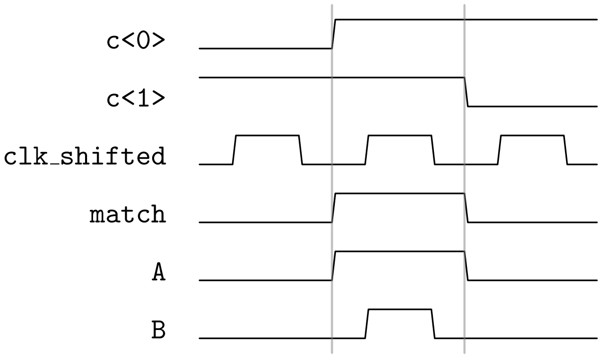


Figure 6.49: Timing diagram for an improved strategy to generate the signals A and B.

plementing analog delay circuits in the cells, an additional timing signal is required. In the current implementation the signal clk2, a clock signal that has twice the frequency of the counter, is distributed over the array for this purpose. A better choice for the timing signal is to use a clock signal with the same period as the counter, but delay it by a quarter of a counter period. This situation is shown in timing diagram 6.49. Signal B can now be generated by a single AND gate with its inputs connected to match and the timing signal clk_shifted. Using this strategy, the generation of A and B requires less transistors than the current implementation. As a consequence, the area consumption of the cells can be reduced.

6.5.3 Reference Voltage for the Current Cells

Dynamic Range

The most severe analog problem of the parameter storage system is its inability to generate output currents smaller than about 60 nA. Measurements for the behavior of the cells at low currents are presented in Section 6.4.3. A minimum output current of about 60 nA indicates that the maximum value of the gate voltage stored in the cell is not larger than 1.7 V, see Figure 6.7. This limitation is probably not related to the design of the current cells but originates from the generation of the reference voltage V_{Gate}, which is used to program the current cells. V_{Gate} is generated by the *VI_conv* circuit, see Figure 6.14. In this circuit the operational amplifier OP1 is used to control the current through T1 such that the voltage drop over the resistor R1 equals the reference voltage V_{ref}.

Typically any transistor implementation of an operational amplifier is affected by an offset voltage V_{OS}, see e.g. Gray and Meyer [1982]. As a consequence the voltage difference between the inputs of the amplifier is not driven towards zero, as it is expected in circuits using negative feedback, but towards V_{OS}. The input offset voltage of operational amplifiers typically arises from device mismatch, affecting the differential input stage. This results in a randomly distributed offset voltage which can have either polarity. Some amplifier designs however are additionally affected by a systematic offset voltage. Further the offset voltage is not stable but varies with parameters such as temperature or input common mode.

Assuming an offset voltage V_{OS1} for the amplifier in the *VI_conv* circuit, the current generated is shifted by $\Delta I = V_{OS1}/R_1$ against the target value V_{ref}/R_1. However, V_{ref} is generated by the *ramp_gen* circuit which is described in Section 6.3.4. Here OP1 is used to buffer the voltage V_{Cap} on the capacitor, the reference voltage is therefore also affected by the offset of an amplifier, $V_{ref} = V_{Cap} + V_{OS1}$. Accounting for the offsets in both amplifiers involved, the current generated in the *VI_conv* circuit is shifted by $\Delta I = (V_{OS1} + V_{OS2})/R_1$. A current offset in range of 60 nA in the output of the current cells translates to an offset of about 2 μA for the current in the *VI_gen* circuit as T1 is 32 times wider than the output transistors of the current cells. Consequentially the combined voltage offset of the two amplifiers needs to be in range of 48 mV in order to explain the current offset.

Measurements indicate that V_{ref} is shifted by a significant positive offset, see Section 6.4.2. In the reset state a value of 60 mV to 80 mV, varying from chip to chip, is observed. Since the voltage over the capacitor in *ramp_gen* is shorted to ground during the reset, the measured voltage is caused by the amplifier buffering the voltage on the capacitor and the additional output amplifier driving V_{ref} to the bond pad `cm_vref`. It is not clear which fraction of the offset is caused by which of the two amplifiers, therefore the minimum voltage level of the internal V_{ref} signal is unknown. However, it is unlikely that the full offset is generated by the output amplifier alone, see Chapter 5.5.

The offsets observed for V_{ref} as well as for the output of the current cells are similar for all three test chips. This suggests that the amplifiers used are not only affected by random device mismatch but mainly by a systematic offset. In order to reliably generate small output currents, the *VI_conv* circuit needs to be modified. First the source of the systematic offset needs to be analyzed, probably the design of the amplifiers can be improved. Further it is possible to connect the negative input of the amplifier in the *VI_ref* circuit not to the buffered V_{ref} signal but directly to capacitor in the *ramp_gen* circuit. This way the offset V_{OS1}, introduced by the amplifier buffering V_{ref}, is eliminated. The simplest solution to compensate for any offset introduced by the amplifier in the *VI_conv* circuit is to intentionally introduce a systematic voltage offset in the amplifiers design, shifting the output voltage towards higher values. This intentional offset has to be larger than the maximum offset which might by introduced by random device mismatch. As a result, the output currents will be shifted towards lower values. These will stay constant at their absolute minimum value for the lowest digital codes. In a calibration step, the digital codes have to be shifted in the control software in order to align the start of the linear ascent of the output currents with the digital code of `0`.

Stability

As for any system using feedback, stability is an important issue for the *VI_conv* circuit. A comprehensive discussion of feedback and stability in linear systems can be found in Laker and Sansen [1994, Chapter 3].

Simulations and measurements have shown that the *VI_conv* circuit, used to generate V_{Gate}, becomes instable under certain conditions. In the prototype chip it is necessary to reduce the bandwidth of the amplifier by reducing its bias current, in order to increase the phase margin. In order to generate V_{Gate}, a very low bandwidth in range of kilohertz is sufficient. However, as described in Section 6.4.2, significant distortions have been observed on V_{ref} and V_{Gate} if all cells are programmed to the same digital code, refreshing their internal value simultaneously. In Section 6.5.5 a possibility to reduce this problem by improving the design of the individual cells is discussed. Further it has also been shown that a higher bias current for the amplifier can help to suppress the effect, see Figure 6.22. To actively compensate the distortions on the programming voltages in a large array, the amplifiers need to have a rather high bandwidth and need to be able to provide high output currents. The best option to allow for stability in the voltage-to-current conversion and sufficient driving strength to distribute V_{Gate} in a large array is to decouple these aspects by using two amplifiers. A slow amplifier in the feedback loop, ensuring a stable conversion, and a second, powerful amplifier buffering V_{Gate}. As described before, an intentional offset needs to be introduced in the design of the amplifier responsible for the conversion. If a second amplifier is used to buffer V_{Gate}, this device will introduce an additional offset. Consequently the intentional offset needs to be increased accordingly.

6.5.4 Improving the Reset for the Reference Voltages

Multiple small problems related to the reset of the capacitor in the *ramp_gen* circuit which is generating V_{ref} have been observed. Several options how to improve the reset process by changes in the digital controller and the *ramp_gen* circuit are presented.

Digital Controller

In 6.4.8 the output traces of cells during the process of reprogramming is shown. It has been observed that the first update process after reprogramming the SRAM in the cells is changing the output value of voltage and current cells towards the maximum output value possible, independently of the actual target value of the cell. This is caused by a minor mistake in the digital controller, the reset of the *ramp_gen* circuit is not activated during the reprogramming of the memory. Therefore I_{ref} continues to charge the capacitor until the maximum possible value, close to the 2.5 V supply, is reached. When the counter is restarted, V_{ref} is still at the high level and the first programming process refreshes the cell towards a high output voltage. The same is true for the current cells as, V_{Gate} is derived from V_{ref}. After the first

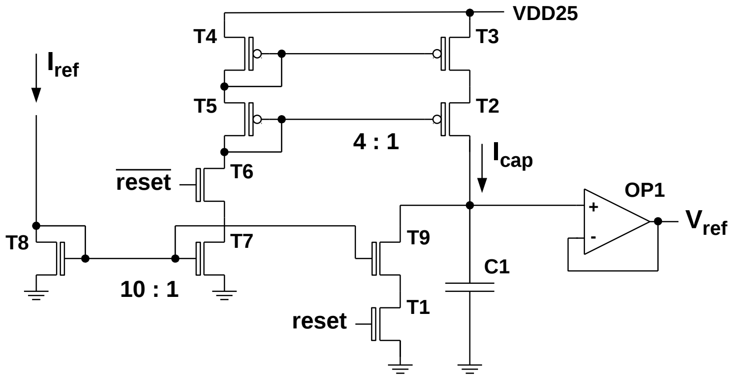


Figure 6.50: Schematic for an improved version of the *ramp_gen* circuit. Transistor T9 slows down the discharging process of C1 when the `reset` signal is activated.

cycle of the counter, the capacitor is reset again and the cell is updated towards the correct target value by the following cycles. Activating the reset during the SRAM programming and releasing it at the beginning of the next counter cycle, as it is done during the breaks in between programming cycles, solves the issue.

However the timing of the reset release at the beginning of the counter cycle also needs a minor adjustment. In the current implementation, the reset is deactivated at the same time the counter is starting. Therefore V_{ref} is already rising during the first interval, in which the counter is at a digital code of `0`. This is not ideal, a voltage cell programmed to `0` should sample the lowest possible value of V_{ref}. Therefore the start of the rise of V_{ref} has to be aligned with the counter changing its state from `0` to `1`. The same applies for the current cells, a cell at `0` should sample the highest possible value of V_{Gate}. Overall the effect of this adjustment is rather low, the maximum error caused by the misalignment of counter and reset in the current implementation of the controller is 1 LSB.

Analog Aspects

On oscilloscope recordings of the output currents and voltages in the parameter storage system significant crosstalk of the reset process is visible, see e.g. Figure 6.45. Instead of shortening the capacitor directly to ground, it is possible to discharge it slowly by a controlled current. This avoids high voltage transients in all circuits of the parameter memory system. Figure 6.50 shows an improved schematic that includes this option. The transistor T9 is added to the first stage of the current mirror dividing the reference current I_{ref}. If T7 and T9 have the same dimensions, capacitor C1 is discharged 4 times faster than it is charged during the ramp generation. Placing the reset switch T1 between the source of T9 and ground limits the distortion of the voltage on C1 when T1 is disabled at the end of the reset period.

6.5.5 Distortions affecting the Reference Voltages

As shown in Figure 6.22, severe distortions can be observed on the V_{ref} signal in case all cells are programmed to the same value. The effect is caused by charge injection from the switch transistors of the cells. This problem can be compensated by using a sufficiently strong buffer to drive V_{ref} over the array. A more energy efficient solution would be to add an additional transistor in every cell that compensates the impact of the switching of transistors T1 and T2. Figure 6.20 indicates that there is some space left in the layout of the cells that allows to add one more switch transistor. The levelshifter used for signal `A` is already generating signal $\overline{\texttt{A}}$ which is required to switch the compensation transistor. Therefore this option can be implemented with only limited effort. Extensive simulations are required to determine the optimum size for the compensation transistor. But since the precision of the cell does not directly depend on this compensation, any compensation that limits the charge injection will help to reduce the distortions on V_{ref}.

The reference voltage for the current cells V_{Gate} can not be monitored directly in the test chip. Measurements indicate that it is also affected by charge injection from the switch transistors activated by signal `A`, see Section 6.4.3. The effect can be reduced the same way as described for V_{ref} by introducing a compensation transistor in the cells.

6.5.6 Adding an Inactive Digital Code

In the current implementation of the digital control circuit of the parameter storage system the counter cycles through the codes from `0` to `1023` during every refresh cycle. During the break between two cycles the counter holds its maximum value of `1023` until the reset signal is released. Since the counter covers every possible 10 bit code in its sequence, all cells in the array are updated in each cycle. In normal operation it is intended and important that all parameters are refreshed in every cycle. However for the purpose of debugging and characterization of the parameter storage system it would be useful if there is a code which is not reached by the counter. Stopping the counter at `1022` would rise the option to program cells to `1023` to prevent them from being refreshed. This feature can be useful to analyze potential digital and analog crosstalk effects in the array. The performance of a single cell can be characterized for both situations, the rest of the array being active or completely passive. Comparing the results can help to identify the source of noise and distortions that are visible at the output of the cell.

6.6 Introducing Different Types of Cells

Building a single storage cell that satisfies all possible demands which might be required by future neuron implementations is impossible. Compromises have to be made, especially regarding the dynamic ranges of output currents and voltages.

However, a few aspects can be covered better if different types of cells, specializing in certain features, are introduced. Some options are described in the following.

6.6.1 Current Cells

Dynamic Range

As mentioned before, the reliable generation of small currents is an important characteristic for the parameter storage systems. These are required to implement power efficient neuron circuits which emulate processes requiring long time constants. On the other hand there are also processes, such as the generation of action potentials, which happen on rather short time scales, requiring larger currents. In the current neuron implementation used in the HICANN chip adjustable current sources providing up to $8\,\mu$A are required. It is difficult to cover a large dynamic range from nano amperes up to multiple micro amperes with the same analog storage cell. In the HICANN chip, current mirror stages are integrated for multiple parameters. These are used to multiply or divide the current generated in the parameter storage system before it is fed to the neuron circuit. However, these mirroring stages consume additional power and chip area. Furthermore the additional device mismatch increases the effective cell-to-cell variation of the parameter storage system.

Another possibility to cover a large range of programmable currents is to implement different designs of storage cells, optimized for different dynamic ranges. In the parameter storage system discussed here, the dynamic range of the current cells can be adjusted by changing the dimensions of the output transistor.

The layout of the current cells offers a simple way to introduce a cell that generates 4 times larger output currents than the ones described. The output transistor T6, see Figure 6.6, is realized by the serial connection of two individual transistors. With only minor changes in the layout, it is possible to connect these two transistors in parallel. The dynamic range of such a cell covers the maximum value used in the neuron implementation of the HICANN chip without introducing additional current mirrors.

Current Source versus Current Sink

The output of the current cells is working as a current source, the output current flow is directed from the supply voltage of the cell towards ground. Some circuits in the neuron implementation need a current sink, conducting a controlled current to ground. In the HICANN chip the parameter storage cells are also working as current sources. Current mirrors in the neuron circuit are used to change the direction of the current if required. As mentioned before, these current mirrors are consuming additional power, chip area and are a source of additional variation.

Therefore it is interesting to investigate the option of using two different types of current storage cells, one version which sources and one that sinks a programmable current. A current sinking storage cell integrated into the capacitive parameter storage system requires an NMOS output transistor. Accordingly, the dynamic range of the voltages stored on the capacitors needs to be shifted towards lower voltages, a range at which the voltage cells described are operating. Current sinking cells can be implemented by just adding an NMOS output transistor to the voltage cells which have already been tested. However, to program such cells an additional programming voltage $V_{Gate,N}$ is required, a complementary version of the $V_{Gate,P}$ used for the current sourcing cells. It can be generated in analogy to $V_{Gate,P}$ using a complementary version of the *VI_conv* circuit.

6.6.2 Voltage Cells

The design of the voltage cells discussed here is optimized for operation closer to the lower end of the voltage range and optimum performance in the middle of the supply voltage range. The maximum voltage that can be produced is 1.8 V.

It is possible that parts of a future neuron implementation can be simplified when programmable voltages closer to the supply level of 2.5 V are available. In the current implementation, the maximum output voltage is limited by the switches based on NMOS transistors, their conductivity drops with increasing voltage. The current sourcing cells used so far are operating with high voltages stored at the capacitors since PMOS transistors are used as switches. Voltage cells operating closer to the supply level can be realized by using a current sourcing cell and omitting the output transistor.

However to program both types of cells the reference voltage V_{ref} needs to cover the full supply voltage range. This is not possible using the *ramp_gen* circuit described in Section 6.3.4, with this implementation the linearity of the ramp is significantly degraded for voltages above 2 V. Another option is to generate two separate reference voltages, covering different voltage ranges.

7 Strategies for a Future Synapse Array

On average, several thousand synapses are found per neuron in the cortex of mammals [Pakkenberg et al. 2003]. Consequently the circuits emulating synapses are by far the most frequent functional units in neuromorphic hardware. Due to their large number, the area and power consumption of the synapse implementation is critical.

In this chapter the options for realizing circuits emulating synapses in the next generation of neuromorphic hardware are discussed. Since digital circuits benefit significantly from the new process technology, the question whether it is possible to realize processing of synaptic events using primarily digital circuits is addressed. Possible benefits from such an approach are increased flexibility and reliability. However, area and power consumption of the digital circuits must be considered and compared to implementations based on mixed-signal approaches.

An architecture which allows for processing of synaptic events by digital circuits has been developed in collaboration with S. Friedmann[1], A. Hartel[1] and J. Schemmel[1]. The general concept is still in an early state of the development. Many of the assumptions it is based on, as well as the conclusions drawn have a high level of uncertainty. The basic ideas underlying the new concept will be presented in the following.

The high speed up factor for the operation of the analog circuits, which is in order of 10^4 compared to biological realtime, is an important feature of the current BrainScaleS Hardware System. However it also leads to an excessive bandwidth demand for the transmission of neural events. Therefore reducing the speed up to a factor of 10^3, which is currently considered for future systems, is an interesting option. While it is difficult to realize long and precisely controllable time constants in compact analog circuits, digital circuits benefit from slower operation.

In order to refine the concept and to evaluate whether it allows for an improved overall performance, two key components have been designed and tested. In Chapter 7.2 the design of a custom high-speed SRAM array featuring two orthogonally oriented ports is presented. The internal structure of the memory is customized for its application in the synapse array. The second component is an 8 bit current DAC which is required to generate the postsynaptic events, assuming that an analog circuit is used for the emulation of the neurons. The design as well as measurement results are presented in Chapter 7.6.

[1] Kirchhoff Institute for Physics, Heidelberg University

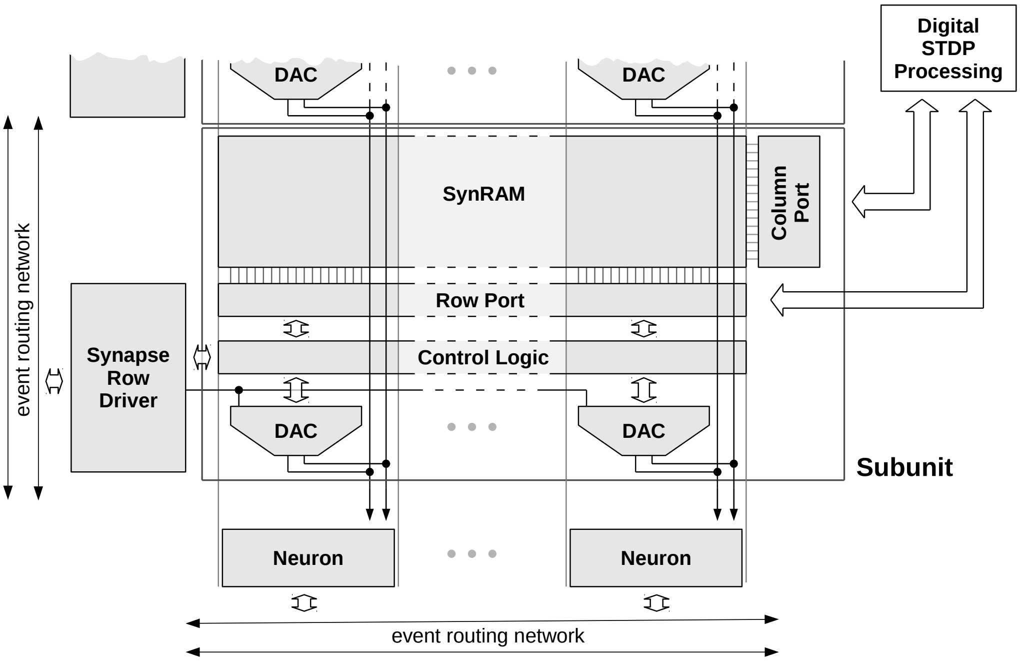


Figure 7.1: Sketch of the architecture for the new synapse array.

7.1 Architecture of the New Synapse Array

The basic strategy for the processing of synaptic events in the new synapse array is similar to the one used in the HICANN chip. A brief description of the existing system is given in Chapter 2.1, for detailed information see Schemmel et al. [2007; 2010].

Figure 7.1 shows an overview of the architecture of the new synapse array. As in the HICANN chip, all synapses within one column are assigned to the same postsynaptic neuron circuit. The major change compared to the existing system is that the synapse array is segmented into subunits. Multiple subunits are stacked to form the full synapse array. Each of these subunits includes a dual-port memory holding the weights and addresses for multiple rows of synapses. Theses memory blocks are termed "*SynRAM*" in the following. Further a layer of control logic and a row of DAC circuits are included in each subunit. The motivation for these changes in the architecture is to allow for an increased resolution of 8 bit for the synaptic weights and the processing of STDP using digital circuits, as detailed in the next section.

7.1.1 Increasing the Weight Resolution

The resolution of the weights stored in individual synapses of the BrainScaleS Hardware System is 4 bit. However, Pfeil et al. [2012] and in Friedmann [2013, Chapter 2.3] suggest that higher resolution leads to significantly better results for various tasks involving STDP-based learning processes. The new synapse array is designed to allow for a resolution of 8 bit for the synaptic weights.

As in the current system, the postsynaptic events are represented by short current pulses. The amplitude of the pulses must be proportional to the weight of the synapse. The area consumption of the DACs which generate the postsynaptic pulses increases significantly as a consequence of the higher resolution. Therefore it is not possible to implement an individual DAC in every synapse as in the BrainScaleS Hardware System. Instead, the number of DACs is reduced and every DAC is shared by several synapses. Each DAC needs to process the postsynaptic events generated by all synapses located in the same column of the respective subunit. Since a DAC can only produce one postsynaptic event at a time, the problem of possible collisions arises. When two synapses assigned to the same DAC receive an event within the same time slot, one of the events has to be dropped. The resulting drop rate depends on the overall activity in the array, the length of the time slots occupied by one synaptic event and the number of synapses sharing one DAC. The time for processing a single synaptic event depends on the duration of the postsynaptic pulse, but also on the time required to transfer the weight of the respective synapse from the memory to the DAC. The speed of the memory holding the weights is therefore critical. The work presented in this chapter was done to obtain some reliable information on how much area the DAC and the synaptic weight memory cover and what results can be expected. These numbers are required to determine the optimal values for other variables in the concept. Especially the number of synapses sharing the same DAC, which is equal to the number of synapse rows integrated into one subunit, is a critical parameter for the overall concept.

7.1.2 Processing Synaptic Events

The processing of synaptic events follows a similar sequence as described for the HICANN chip in Section 2.1.1. When a neuron circuit generates an action potential, an event, encoding the number of the neuron, is transmitted by the digital event routing network. Presynaptic events sent over the event routing network are received by the *synapse_driver* circuits. In the new system, each *synapse_driver* is associated to one subunit of the synapse array. The *synapse_driver* splits the received neuron number into a row address and a column address. Both are sent to the control logic layer in the subunit. Using the row port of the *SynRAM*, the control logic reads the information associated with the addressed row of synapses from the memory. The memory contains the individual addresses and the weights for the individual synapses. The address of each synapse in the row is compared to the column address provided by the *synapse_driver*. In case of a match, the DAC of the corresponding column is enabled and provided with the weight of the respective synapse. The DAC then generates a postsynaptic current pulse. The amplitude of the pulse is proportional to the weight of the synapse. The duration of the pulse is controlled by a separate signal generated in the *synapse_driver*. An important aspect of this concept is that for the processing of each individual synaptic event data needs to be read from the SynRAM.

7.1.3 STDP in the New System

From a technical point of view the implementation of STDP is a challenging task. The correlation between pairs of pre- and postsynaptic events needs to be evaluated continuously for every individual synapse. In the HICANN chip local analog circuits, which measure the weighted correlation for individual pairs and accumulate the results for each synapse are used. These values are evaluated by a global digital controller which updates the weights of the synapses according to the implemented STDP rule. The approach of combining analog and digital circuits allows for an efficient implementation, however it lacks flexibility. One restriction is the limited set of rules which can be realized using the digital controller of the HICANN chip. To overcome this limitation a programmable processor, which can replace the controller and allows for a wide range of different rules, has been developed, see Friedmann [2013, Chapter 3]. However, the analog part still limits the flexibility of the system. The exponential shape and time constant of the weight function are determined by the analog circuit in the synapse and cannot be modified. Further the controller can only operate with the accumulated correlation information and not account for individual events.

Since modern process technologies offer smaller and faster digital circuits, the realization of an all-digital STDP concept is considered an option for a system built in the 65 nm process technology. Digital processing of each individual spike allows for total flexibility regarding the implementation of STDP and for operating the system at lower speed up factors. A general description of such a system is presented in [Friedmann 2013, Chapter 6.2].

To allow for digital processing of individual spikes, the STDP controller units need fast access to the synaptic weights stored in the synapse array. In case of a presynaptic event, the STDP processors need to be able to read and process the weights of all synapses in the corresponding row. When a postsynaptic neuron emits a spike, the STDP controllers need to process all synapse weights in the corresponding column. The question how and at which price in terms of area consumption a synapse weight memory block providing this functionality can be implemented is addressed in the following.

7.2 Topology of the Synapse Memory

The central component of the new concept for the synapse array is the *SynRAM* block which holds the synaptic weights. Read operations on the memory are required for the processing of every synaptic event. The duration of the memory accesses, especially the read operation, is therefore critical.

The structure of the *SynRAM* is organized in columns, each of which is 8 bit wide. Each column holds the data associated with synapses which are connected to the same postsynaptic neuron. The information associated with a single synapse is stored in two rows within the column. One row holds the 8 bit weight of a synapse. The other row holds the address of the synapse, which is 4 to 6 bit wide, depending on the size of the system. The remaining bits in that row can be used for implementation of additional features of the STDP logic. As an example, the logical resolution of the synapse weight can be increased by adding two bits of lowest significance. This enables the controller to internally operate on synapses with 10 bit resolution. The resolution of the synapse DAC does not change. The height of the postsynaptic pulses is still proportional to the value of the 8 most significant bits of the extended weight. The spare bits could also be used to signal the STDP processing units to apply different update rules for individual synapses.

7.2.1 Dual-Port SRAM

As described in Section 7.1.3, the STDP processors need to have fast access to the synapse weight data, either within a full row or a full column. In case of a postsynaptic event the STDP controller needs to process all synapses which are connected to the neuron that generated the action potential. Therefore the weights of all synapses in the same column need to be read and processed by the controller. In case of a presynaptic event, the situation is reversed. The STDP controller needs to process the weights of all synapses which received the action potential. These are located in the same row.

To allow for efficient operation of the STDP controllers, the *SynRAM* is implemented as dual-port SRAM. Other than in typical applications for dual-port SRAM, the ports are oriented orthogonally to each other. A schematic of such an SRAM cell is shown in Figure 7.2, the two ports are named A and B. The cell has two pairs of access transistors. The bitlines of port A, $\mathrm{BL_A}$ and $\overline{\mathrm{BL}}_\mathrm{A}$, are oriented vertically, the corresponding wordline $\mathrm{WL_A}$ is oriented horizontally. For port B the orientation of bitlines and wordline is inverse.

In an array of such SRAM cells, port A allows for accessing a full row from the memory array by activating the corresponding wordline and reading or writing the single bits using the vertical bitlines. Accordingly, port B allows to access the contents of the memory column-wise. However, using orthogonal dual-port SRAM alone is not sufficient to provide optimum access for a digital STDP controller because more than one bit of information is associated to a single synapse. According to the plans for the new synapse array, 16 bit of memory are required for every

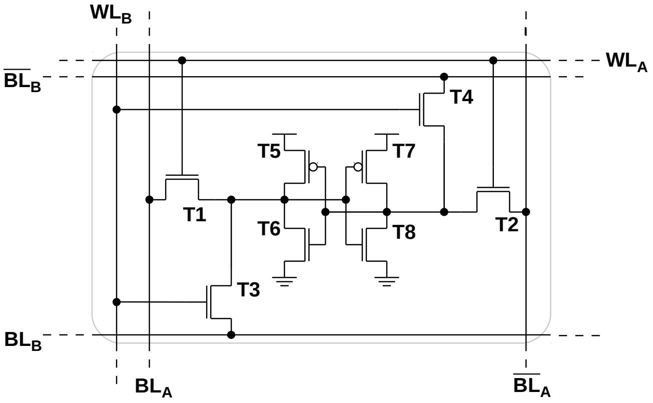


Figure 7.2: Schematic of a regular dual-port SRAM cell with orthogonally oriented ports. Port A uses the transistors T1 and T2 to access the latch built from transistors T5 to T8. The bitlines of port A, $\mathrm{BL_A}$ and $\overline{\mathrm{BL}}_\mathrm{A}$, are oriented vertically, the associated wordline $\mathrm{W_B}$ runs horizontally through the array. For port B the situation is reversed: Port B uses the transistors T3 and T4 to access the latch. The bitlines of port B, $\mathrm{BL_B}$ and $\overline{\mathrm{BL}}_\mathrm{B}$, run horizontally. Consequently $\mathrm{W_B}$ is oriented vertically.

synapse. These are split into 8 bit for the weight information and 8 bit for storing the address of the synapse as well as additional technical information. If the memory bits belonging to a single synapse are arranged in a rectangular field of 8 × 2 bit of regular dual-port SRAM, it takes two read accesses to retrieve the information related to a single synapse for the port reading rows of the memory. However, to retrieve the same information using the port reading columns it would take 8 successive operations. An alternative approach is presented in the following.

7.2.2 Modified Dual-Port Structure

Based on regular orthogonal dual-port SRAM, a memory structure optimized for its application in the synapse array has been developed. It allows to access any 8 bit word stored in the memory by a single operation, using either one of the two ports. In order to access the 8 × 2 bit assigned to a single synapse, two accesses are required.

As mentioned before, the STDP controller needs to access the information associated with a full row of synapses in case of presynaptic event. The memory architecture described in the following provides a dedicated port using horizontal wordlines and vertically oriented bitlines. All signals associated with this port will be marked by the index `row`. Accordingly, information associated with all synapses within a column needs to be processed in case of a postsynaptic event. The signals associated with the port which are dedicated to this task will be marked by the index `col` in the following. In order to simplify visualization, the internal

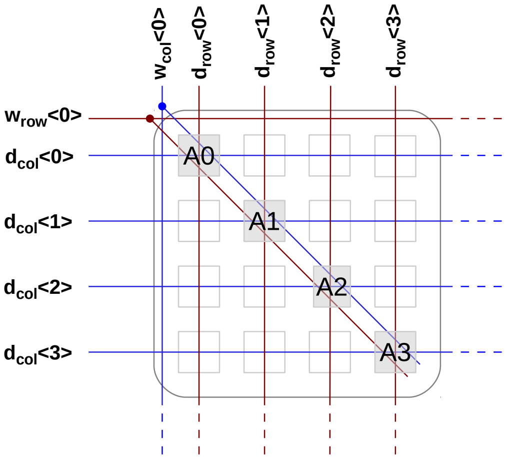


Figure 7.3: Diagram of the ideal logical structure for storing the data bits A0 ... A3 associated with a single synapse in a memory array. Due to the diagonal structure all bits can be read or written within a single access either from the `col` port (blue) or the `row` port (red).

architecture of the *SynRAM* will be explained for a situation where one synapse is associated with only a single row of 4 bits. Furthermore, only one bitline `BL` will be shown per port, its inverted counterpart $\overline{\texttt{BL}}$ is omitted.

From a logical point of view the structure sketched in Figure 7.3 would be ideal. Arranged in a diagonal fashion, it is possible to access all bits of a single synapse from both ports by a single access operation. However, this structure does not allow for an efficient implementation of the synapse memory on a chip. It is necessary to use a regular pattern of SRAM cells as a base for the memory, as the layout benefits significantly if signals and resources, such as contacts to the diffusion regions, between neighboring cells can be shared.

Implementing the structure exactly as shown in Figure 7.3 in a regular array of SRAM cells is not possible, only a fraction of the total number of cells could be used. The diagonal structure described needs to be transformed into an area efficient arrangement, being as close as possible to a standard dual-port SRAM array. Figure 7.4 shows the option of stacking multiple diagonally oriented synapses A ... D within a column. For better visibility (a) only shows the signals required by the `row` port (red), (b) instead shows only the signals associated with the `col` port. Depending on the total number of synapses stacked in this pattern, the utilization of the memory area can be very high. However, this architecture requires an additional layer of multiplexing logic for the bitlines of the `col` port. Depending on which synapse is accessed a different subset of 4 bitlines needs to be connected to the logic controlling the memory. Simultaneous access is only possible for synapses which have a distance that is greater than the number of bits per synapse. Especially for

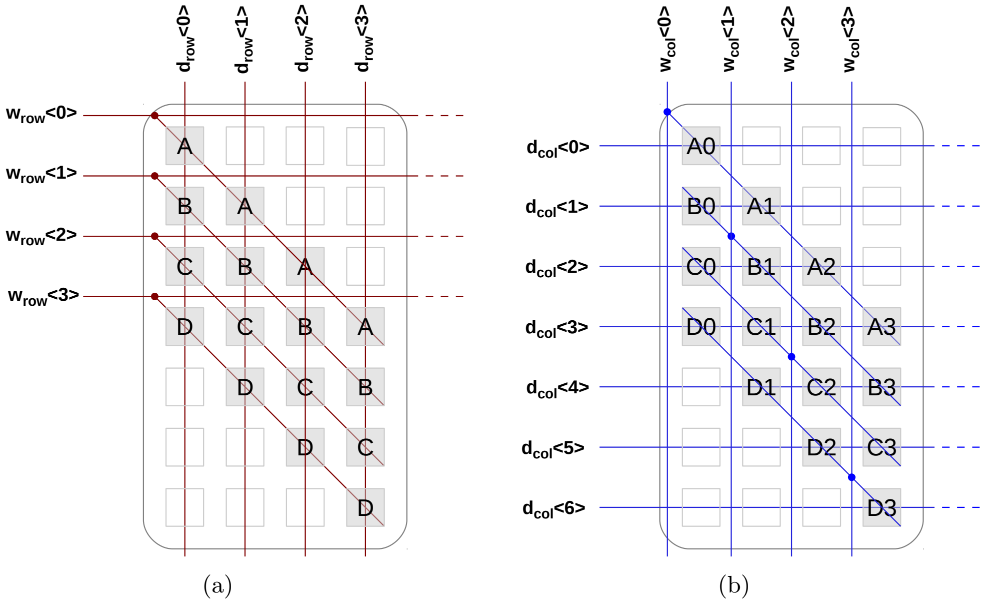
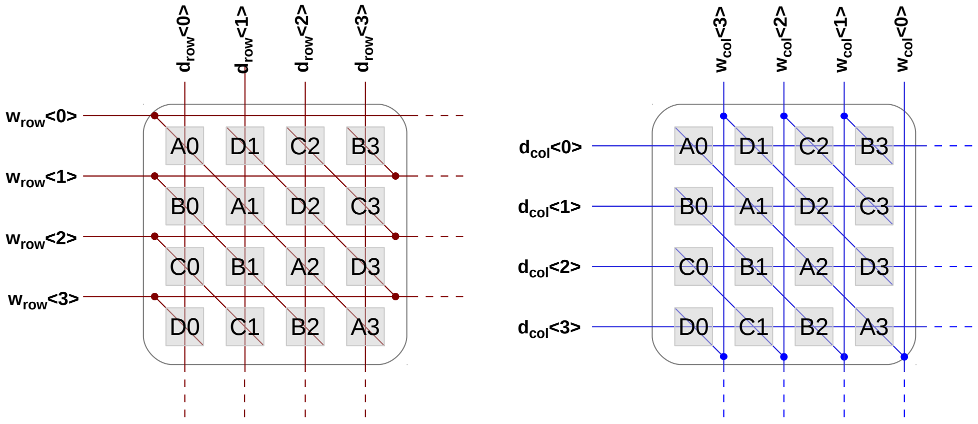


Figure 7.4: Possible strategy to improve the area utilization for diagonally arranged synapse memory. For better visibility (a) only shows the signals required by the `row` port (red), whereas (b) shows only the signals associated with the `col` port. Note that for the `col` port a different subset of the 6 horizontal bitlines has to be used, depending on which of the synapses A ... D is accessed.

large numbers of stacked synapses, managing the selection of the correct bitlines requires a significant amount of additional logic.

The architecture finally implemented in the *SynRAM* is shown in Figure 7.5. Again the signals of the two different ports are shown in separate pictures. The columns in the memory array are partitioned in a stack of squared sub blocks. In the simplified example, using a column width of 4 bits, one block holds 16 bit. The first synapse A covers the bits in the diagonal of the square, representing the ideal structure described before. The next synapse B covers the row below the diagonal set of bits. This row has only 3 bits, used for B0 to B2. Therefore the single bit located in the upper right corner of the memory block is additionally assigned to this synapse, holding B3. The position is ideal from a routing point of view, the bit is located in close spacial proximity to the correct bit- and wordline for both ports. The same can be done for the next diagonal row holding the data for synapse C. As there are only 2 bits in the lower left corner, C0 and C1 are assigned to these positions, two more are added from the upper right corner, holding C2 and C3. D is finally implemented the same way as B but mirrored relative to the first diagonal.

Using this structure, the ordering of the bits is only correct for the synapse



(a) (b)

Figure 7.5: Stucture of a building block of the *SynRAM*. For better visibility the signals associated with the two ports are shown in two separate pictures. Every bit in the array is utilized and the data associated with each synapse can be accessed from either of the ports in a single operation. Note that the order of the horizontal bitlines is different for each synapse in case the `col` port is used.

covering the bits on the diagonal of the block identical for both ports. For the other synapses in the block the relative ordering of the `col` bitlines to the numbering of the single bits is shifted in a cyclic manner. E.g. the bits associated to synapse C are presented in the order C2, C3, C0, C1 at the `col` port. This cyclic shift can be compensated by an 8 bit barrel shifter [Gorgin and Kaivani 2007], which can be implemented efficiently in a custom circuit by using three layers of transmission gates.

Using the described architecture, all bits of a regular memory array can be utilized. For both ports the bitlines can be routed straight through the array, keeping their capacitance as low as possible. For the wordlines the situation is slightly more complicated, the global wires which are running straight through the array need to be extended by local diagonal wires, connecting it to the cells.

A layout for a *SynRAM* array with the proposed structure has been developed. The area consumption of a single bit of custom standard dual-port SRAM is 1.5 μm^2/bit. For the structure of a dual-port SRAM with diagonally oriented sub units the area consumption per bit increases slightly to 1.7 μm^2/bit. In standard dual port SRAM the wordlines are shared between all cells within a row or a column respectively, therefore it is possible to save area by sharing gate contacts between neighboring cells in the layout. Due to the diagonal routing of the wordlines in the memory architecture described here, the gates of the access transistors have to be isolated from the neighboring cells. Therefore a different cell layout is required, increasing the area consumption per bit by about 12 %. The layout is based on building blocks containing 4 bit in a 2 × 2 bit arrangement. These building blocks include the wires for routing of bitlines, wordlines, power and ground. Multiple blocks can be combined by edge connection to form memory columns of the required size. The layout uses the metal layers `M1` to `M5`.

The concept for the synapse array discussed so far is based on a column width of 8 bit. Each block consequently holds 64 bits, storing the information associated to 4 individual synapses. Due to the flexible structure based on small building blocks, the layout can be adjusted to different column dimensions if required.

7.3 Controlling the Access Operations

Accessing the SynRAM works identically to the operation of regular SRAM. A brief description of the basic write and read procedure for SRAM is given in Section 5.1.1. A more detailed discussions of SRAM operation can be found e.g. in Chandrakasan et al. [2000, Chapter 14.2].

As mentioned before, speed is critical for the *SynRAM*. An access operation has to be performed for the processing of every single post- or presynaptic event. Therefore common strategies to speed up SRAM operations are used in the circuits controlling the memory. Standard voltage-mode sense amplifiers, evaluating the voltage difference between the bitlines of a memory column, are used to speed up the reading process. The full circuit controlling the bitlines, including the sense amplifier, is described in Section 7.3.1. In order to allow for the shortest possible timing for the precharge process and the activation of the wordline during read and write access, the control logic uses analog delay elements based on replica bitlines. The circuits used to generate the timing are described in Section 7.3.2. These measures aim at performing an access over the `row` port within a single cycle of a clock operating at a frequency of 500 MHz. For the `col` port the bitlines are significantly longer, here the goal is to perform an access operation within two clock cycles.

7.3.1 The Sense Amplifier

A common strategy to reduce the read time in large arrays of SRAM is to use sense amplifiers. As described in Section 5.1.1, both bitlines of a column are charged to the supply voltage when initiating a read access. When the corresponding wordline is activated, the addressed memory cell starts to discharge one of the bitlines. In large SRAM arrays the bitlines have a significant capacitance. Since two small transistors connected in series are connecting the bitline to ground, the discharge rate is limited. Using only standard digital circuits to detect a logical `0` for the bitline requires the voltage level to drop significantly below half of the supply voltage. The read time can be reduced by introducing an amplifier which compares the voltage between the two bitlines of a column. This allows for reliable detection of

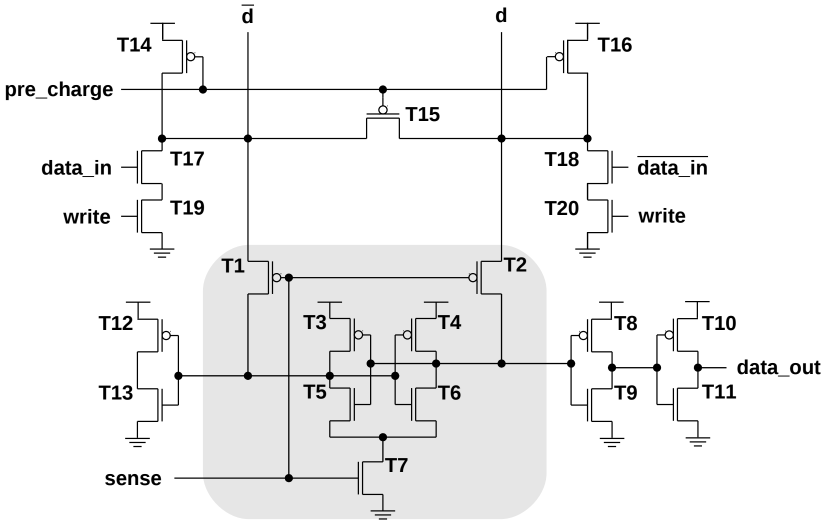


Figure 7.6: Schematic of the *BL_ctrl* circuit which controls the bitlines of the SRAM cells in the *SynRAM*. The sense amplifier is built from transistors T1 to T7, highlighted by the gray box. The inverter built from T8 and T9 reads the internal state of the latch in the sense amplifier. T12 and T13 are required to ensure equal capacitance at the internal nodes of the latch. Transistors T14 to T16 are used for precharging the bitlines. T17 to T20 are required for write operations.

small voltage differences, therefore the state of the addressed cell can be detected after a shorter time interval.

A conventional voltage-mode sense amplifier is used in the *SynRAM* to speed up the reading process, see e.g. Sinha et al. [2003]. Figure 7.6 shows a full schematic of the circuit controlling the bitlines of the *SynRAM*, referred to as *BL_ctrl*. The sense amplifier is highlighted by the gray box. The transistors required for precharging of the bitlines and for writing to the SRAM are also included in this circuit.

The sense amplifier is based on a latch built from transistors T3 to T6, the bitlines of the respective SRAM column are connected to the latch via transistors T1 and T2. The inverters of the latch are not directly connected to ground, instead this is done via transistor T7. The sense amplifier is controlled by the digital `sense` signal. When a read access is initiated the `sense` signal is low. The latch is deactivated as it has no connection to ground, but its internal nodes are connected to the bitlines via T1 and T2. When the precharge process is finished, the wordlines are activated and the addressed SRAM cell starts to discharge one of the bitlines. Once the voltage difference between the bitlines is assumed to be sufficiently large, the wordline is deactivated and the `sense` signal is activated instead. This transition disconnects the bitlines from the sense amplifier and activates the latch by enabling T7. Since the nodes of the latch circuit are charged to different voltages, the latch will flip into the same state as the one in the addressed SRAM bit. For readout the state of the latch is buffered using two inverters built from T8 to T11. The inverter built from T12 and T13 is required to keep the capacitive load at the internal nodes of the latch symmetric.

The minimum voltage difference, which determines the minimum read time, required for reliable comparison of the bitline signals depends on different aspects. The sensitivity of the comparator strongly depends on the mismatch-induced asymmetry of the two inverters in the latch circuit. This effect depends on the size of the transistors T3 to T6 and can be predicted using Monte Carlo simulations. The size of the transistors can be adjusted to achieve the required sensitivity. Additionally random noise and crosstalk in the circuit are distorting the absolute voltage levels on the bitlines and need to be taken into account. However, it is difficult to make reliable estimations for the overall noise level on the bitlines in the final chip.

In the *SynRAM*, the circuit shown in Figure 7.6 is used for both ports. Though the optimum overall dimensions for a *SynRAM* unit are not yet determined, it is clear that the width of the memory has to be significantly larger than its height. For the design of the memory circuits a width of 1024 bit and a height of 64 bit has been assumed. The horizontally oriented bitlines of the `col` port are 16 times longer than the vertical ones of the `row` port. As a result, the voltage on the `row` bitlines drops quickly when performing a read access and a low sensitivity of the amplifier is sufficient. For the long bitlines of the `col` port, sensitivity is more important to achieve fast reading. Therefore different implementations of the amplifier are used for the different ports. In Table 7.1 the dimension of the critical transistors for the sense amplifier of the `row` port and `col` port are listed. The effect of the different transistor dimensions is clearly visible in Monte Carlo simulations of the comparison process. The smaller `row` amplifier requires a voltage difference of 120 mV for reliable detection, no error has been observed at this level over 2000 simulation runs. For the larger amplifier, used for the `col` port, no errors have been detected within 2000 simulation runs down to a level of 35 mV between the bitlines. However, these simulations did not account for transient noise which will be present in an actual implementation.

The schematic shown in Figure 7.6 includes not only the sense amplifier but also the transistors required for precharging the bitlines and writing SRAM cells. During the precharge phase the bitlines are connected to the supply voltage by the transistors T14 and T16. In case of an aggressively optimized timing for the precharge process, it is possible that the bitlines do not reach the full supply level. In this case mismatch between the charging transistors can lead to non identical voltage levels at the two bitlines, a situation that increases the probability for a failure of the following reading process. A standard way to prevent this situation is to shortcut the bitlines directly, using the additional transistor T15, during the precharge phase.

For writing an SRAM cell, one of the bitlines has to be driven to `1` and the other one to `0`. In the *SynRAM* this is realized by initiating the write process by

Port	Device	length $[nm]$	width $[nm]$
row	T1,T2	60	400
	T3,T4	80	400
	T5,T6	80	240
	T7	60	480
col	T1,T2	60	240
	T3,T4	215	1200
	T5,T6	215	1200
	T7	60	560

Table 7.1: Dimensions of the transistors of the sense amplifiers for `row` and `col` port. The transistors used in the sense amplifiers of the `col` port are significantly larger, in order to reduce the effect of device mismatch.

a precharge phase, charging both bitlines to the supply voltage. In a second step one of the bitlines is discharged to ground over transistors T17 and T19 or T18 and T20, depending on the input data. The transistors used to charge and discharge the bitlines are also different in size for the two ports. For the `col` port significantly wider transistors are implemented since a larger capacitive load has to be driven.

7.3.2 Generating the Access Timing

To write or read the memory, several digital control signals have to be generated in correct sequence and with correct timing. As mentioned in Section 5.1.1 an controller based on RTL logic can be used to control the SRAM access. In this case the temporal resolution for all control signals is restricted to half clock cycles. For the *SynRAM* speed is critical, accessing the data over the `row` port should be possible within a single cycle of the system clock which is running at 500 MHz. For the `col` port one access should be finished within two clock cycles. Optimizing the timing of single or dual clock cycle SRAM access operations requires a higher temporal resolution than half clock periods. Using a digital controller, this can only be achieved when the controller is operating relative to an additional clock running at a significantly higher frequency than the system clock. Another solution, not depending on an additional high frequency clock signal, is to use analog delay elements to control the duration of the SRAM operation phases. However, if the timing of the SRAM operation is aggressively optimized, typical analog delay elements, such as a chain of current starved inverters, require precise tuning. An elegant solution to implement delay elements in SRAM circuits is presented in Amrutur and Horowitz [1998]. A replica bitline, connected to replica cells, is added to the memory array and used as a delay element. That way the delay is affected to the same amount as the actual bitlines by variation of global parameters such as process corner, temperature or

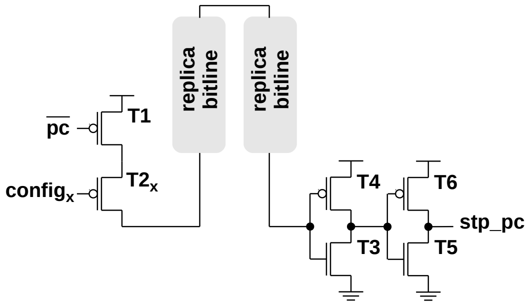


Figure 7.7: Circuit used to control the duration of the precharge phase.

supply voltage and the timing is inherently adjusted to these conditions. In modern process technologies not only the transistor characteristics are subject to process variation, also the ohmic resistance and the parasitic capacitances in the metalization layers shows significant systematic fluctuations. Using replica bitlines in the timing generation accounts for all global characteristics that change the properties of the actual bitlines. The circuits used to generate the timing for the signals that control the precharge phase, the wordline and sensing are inspired by the circuits discussed in Amrutur and Horowitz [1998].

The concept of the circuits used in the current *SynRAM* design are described in the following. The circuits generating the timing are basically identical for both ports. Only the dimensions of individual transistors are different due to the highly asymmetrical aspect ration of the memory block.

Precharge Timing

Reading SRAM is a critical procedure, the internal states of the SRAM cells must not be destroyed. Therefore the bitlines of the SRAM need to be reliably charged to a high voltage level before the wordline is activated and the bitlines are connected to the internal latches of the cells. This process is called the precharge phase. The duration of this phase depends on the driving capability of the transistor charging the bitline and the RC characteristic of the bitline itself. The minimum time interval required for the precharge phase is affected by process variations changing the characteristics of the transistors as well as the parasitic capacitances and resistances in the metalization layers. In order to start the read process as soon as possible, it is necessary to detect when the bitlines are charged to a sufficient level. In the *SynRAM* this is done by the circuit shown in Figure 7.7.

It uses a replica bitline that is added to the array in order to model the real bitlines in the memory. The dummy bitline is charged by PMOS transistor T1, the other end of the bitline is connected to the input of an inverter. As soon as the voltage level at the end of the bitline crosses the triggering voltage of the inverter, the signal `stp_pc` is generated, signaling that the precharge phase can be stopped. The transistors in the first inverter, T3 and T4, are significantly larger than minimum size to reduce the mismatch-induced variation of its triggering voltage. The ratio between the length of the actual bitline and the length of the replica bitline is used to determine the voltage level that is reached on the actual bitlines when the stp_pc signal is activated. Furthermore the relative size of the transistors charging the actual bitlines and the one charging the replica bitline can be used to adjust the timing generation process. In the *SynRAM* the effective width of T1 is configurable. Another aspect with impact on the timing is the propagation delay caused by the control logic and the various buffer stages that are required to drive the control signals along the memory array to the individual *BL_ctrl* circuits. Especially for the row port, where the bitlines are rather short, the contribution of these additional delays is significant.

To precisely optimize the ratio between the length of the actual and the replica bitlines simulations accounting for all parasitic effects in the layout of the full memory block are required. However, if a netlist including parasitic data for the full layout is used, the simulation times become prohibitively long. Therefore a simplified setup, accounting only for the parasitic properties of word- and bitlines, is used. In order to allow for compensation of the resulting inaccuracy, the width of the transistor T1 is configurable over a wide range in the implementation of the *SynRAM*. Based on these simulations, the length of the replica bitline is chosen to be twice as long as the regular bitlines. In the layout, the two bitlines of an additional column are connected in series to realize the replica bitline for row port. The two bitlines of an additional row are connected in series as a replica bitline for the col port. For the row port the ratio between the effective width of the transistor charging the replica bitline, T1 in Figure 7.7, and the effective width of transistors charging the actual bitlines, T14 and T16 in Figure 7.6 is configurable in a range from 1/6 to 2/3 in four steps. At the setting leading to the slowest possible timing, the replica bitline is charged by 1/6 of the current which is charging the actual bitlines. As a result the voltage at the actual bitlines reliably reaches a voltage of 1.2 V by the time the precharge process is stopped. The timing can be optimized by increasing the width of the transistor charging the replica bitline.

In case of the col port the timing of the precharge circuits is generated as described for the row port. However, there is one significant difference. In order to speed up the process for the significantly longer bitlines, precharge transistors are not only located in the *BL_ctrl* circuit but also at the opposing end of the bitlines. The ratio between the combined width of the transistors charging the actual bitlines and the effective width of the transistors charging the replica bitline can be adjusted over a range from 1/10 to 1/1 in eight steps.

Wordline Timing

When an access operation to the SRAM is initiated, the bitlines are precharged. Next the wordline for the selected row or column of the array is activated. When writing, one of the bitlines is discharged quickly by either of the large write tran-

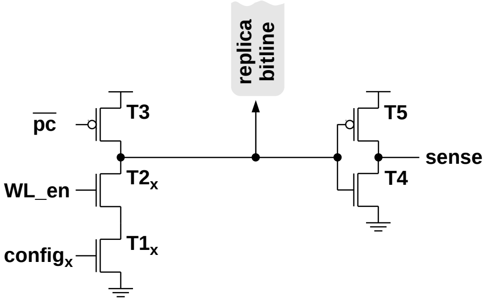


Figure 7.8: Circuit used to control the timing for activation of the wordline.

sistors included in the *BL_ctrl* circuit. When reading, one of the bitlines is slowly discharged by the addressed SRAM cell. The wordline needs to stay activated until the voltage difference between the two bitlines is large enough to allow for reliable operation of the sense amplifier.

The timing for the write/read phase is again generated using a replica bitline. The corresponding schematic is shown in Figure 7.8. The replica bitline is charged by T3 during the precharge phase. When the `WL_en` signal is activated, the replica bitline is discharged through the NMOS transistors T2 and T1. These transistors are replicating the access transistor and the NMOS transistor of an inverter of the latch in an SRAM cell. The inverter built from transistors T4 and T5 is used to detect when the voltage level on the replica bitline drops below 600 mV. The output signal of the inverter is used to disable the wordline, stopping the discharge process of the bitline, and trigger the sense amplifier. The voltage on the actual bitlines in the moment the `sense` signal is activated is determined by the ratio between the length of the actual bitline and the length of the replica bitline. Additionally multiple cell replicas can be used in parallel to speed up the discharge process of the replica bitline. In the *SynRAM*, the number of replica cells used to discharge the replica bitline is configurable. However, if the ratio between the replica bitline and the actual bitlines is chosen correctly, the timing is generated correctly over a wide range of conditions, without the need for adjusting any parameters. As for the precharge timing, the propagation delay in the control logic and the buffer stages adds a fixed offset to the interval the wordline is activated.

A simulation setup, accounting for the parasitic properties of the bit- and wordlines involved, has been used to determine the relative length of the replica bitline and the number of replica cells which are used to discharge it. Reliable operation is achieved for a replica bitline which has half the length of an actual bitline, using 4 replica cells to discharge it. In order to account for uncertainty in the simulation setup, the number of replica cells can be configured in a range from 1 to 8.

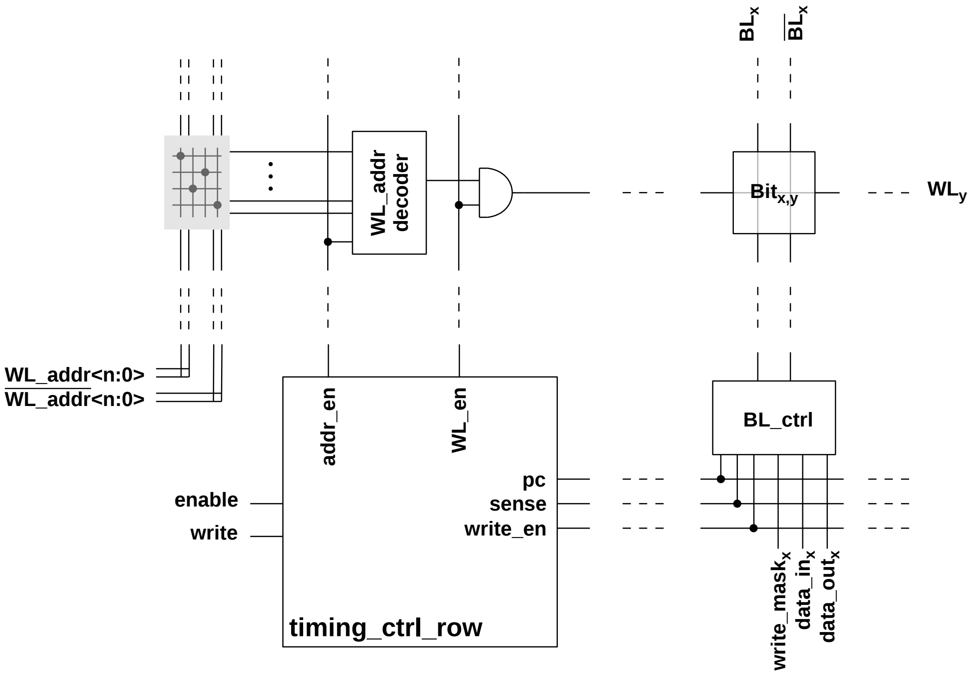


Figure 7.9: Overview of components and signals involved in access operations of the `row` port.

7.3.3 Control Logic

Here, the basic architecture of the *SynRAM* control logic and the signals involved will be presented. An overview over the individual components and signals involved in accessing the memory through the `row` port is given in Figure 7.9. The control logic for the `col` port is basically identically organized, only the physical orientation of the components is different. The following description refers to the `row` port.

The control logic for the `row` port is located in the *timing_ctrl_row* module. This module has two external inputs, `enable` and `write`. When the `enable` signal is activated, an SRAM access procedure is triggered. The state of the `write` signal determines whether a read or write operation is performed. The circuits used to generate the timing signals, described in Section 7.3.2, are located within the *timing_ctrl_row* module.

The global signals `pc`, `sense` and `write_en` are distributed horizontally, controlling the *BL_ctrl* circuits for all columns. The *BL_ctrl* circuit has been described in Section 7.3.1. Besides the timing control signals, each *BL_ctrl* instance is supplied with the input data that is to be written to the memory. Further it provides an output for data which has been read from the memory. An additional input allows to use a write mask, in case not all bits in a memory row need to be written. The

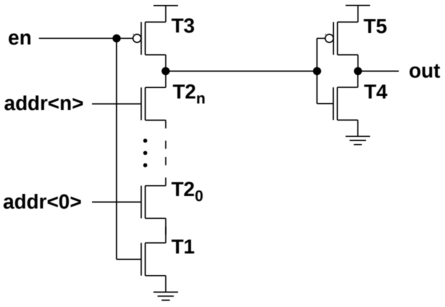


Figure 7.10: Schematic of the address decoder circuit used in the *SynRAM*.

`write_mask`$_x$ signal and the global `write_en` are connected to the inputs of an `AND` gate (not shown in Figure 7.6), the output of the gate is enabling the actual write process for the respective column.

For each access to the memory, the row to be read or written has to be selected. This is done by setting the wordline address `WL_addr`, the width n of the address needs to be adjusted to the number of rows in the memory. The address signals and the inverted address are routed vertically along the edge of the memory. Each row is eqipped with an address decoder. The n inputs of each address decoder are horizontally crossing the `WL_addr` and $\overline{\texttt{WL\_addr}}$ signals. The individual addresses of the rows are coded in an array of vias, connecting each input x to either `WL_addr`$_x$ or $\overline{\texttt{WL\_addr}_x}$. The standard address decoder circuit shown in Figure 7.10 is used to evaluate the state of the address inputs for each row. As long as `addr_en` is low, the output of the decoder is `0` because the input of the inverter built from transistors T4 and T5 is pulled to VDD by T3. When an access operation is triggered, the `addr_en` signal is activated immediately by the *BL_ctrl* circuit. In case all address inputs of the circuits are `1`, which is the case for only a single row, the input of the inverter is discharged through the chain of NMOS transistors and the output of the address decoder turns active. For all rows in which at least one address input is at `0`, the output of the address decoder stays at `0`.

Once the end of the precharge phase is detected by the timing control circuit, `WL_en` is activated and the wordline of the addressed row turns `1`, connecting the vertical bitlines to the latches of the SRAM cells in the respective row. When the end of the write/read phase is indicated by the wordline timing circuit, the `WL_en` is deactivated and the `sense` signal is activated instead. Triggered by the `sense` signal the sense amplifiers evaluate the voltage difference between the bitlines, regardless of whether a write or read access was performed. In case of a write operation the value written is also read by the amplifier. The activation of the sense signal is the last step of an access operation.

Simulations suggest that an access operation on the `row` port is completed within less than 2 ns, allowing for a clock frequency of up to 500 MHz. For the `col` port every access is completed within less than 4 ns, it can be operated at a clock frequency of up to 250 MHz.

7.3.4 Interface to Digital Standard Cell Logic

The memory block itself is designed to allow for a write or read access within a single clock cycle. If the `enable` signal for one of the ports is activated synchronously to a positive edge of the clock signal, the *timing_ctrl* circuit performs the requested SRAM operation based on the internal timing generation. In case of a read access, the data read from the memory can be sampled from the outputs of the sense amplifiers at the next rising edge of the clock signal. Simultaneously, a consecutive access operation can be initiated.

However, the memory block needs to be integrated into a digital RTL design which is synthesized from a circuit description written in a high-level hardware description language (HDL) such as Verilog [Verilog 2006], System Verilog [SystemVerilog 2004] or VHDL [VHDL 1988]. In order to allow for reliable data transmission to and from the memory, the timing characteristics of each input and each output needs to be specified. This information is required by the software realizing the physical implementation of the HDL code, it has to ensure that the timing constraints of all pins are met. To simplify the timing characterization for the memory, a layer of custom flip-flops is attached to all inputs and outputs. This way it is only necessary to characterize the flip-flops used in order to allow for an automated timing verification.

A. Hartel developed a workflow that allows for automated timing characterization of the flip-flops at the inputs and outputs of the *SynRAM* block. It uses the Cadence Spectre simulator in order to simulate the circuit under various conditions. The corresponding setup, hold and clock-to-output time are extracted from the resulting waveforms. The results are written to a file in the “liberty” format, containing the information required by the Cadence Encounter Digital Implementation software to integrate the memory in the design.

A drawback of the additional flip-flop stages is the increased latency of two additional clock cycles for each access operation. The throughput in a series of consecutive accesses is not reduced. In Figure 7.11 the RTL timing diagram of the external signals of the `row` port is shown for a write and a read access.

7.4 Implementation Details for the Prototype Chip

The circuits of the *SynRAM* have been designed for a memory block of 1024 × 64 bit and have been tested on the second prototype chip. However, the physical size of the prototype chip did not allow to test a full sized *SynRAM* unit. Instead a smaller sub block of 256 × 32 bit, corresponding to 32 × 16 synapses, has been implemented. The total amount of data that can be stored in the memory block is 1 kiB. The dimensions of the *BL_ctrl* circuits, including the sense amplifiers, and the circuits for timing generation are not changed compared to the version designed for a full sized array. The length of the replica bitlines used in the timing generation is

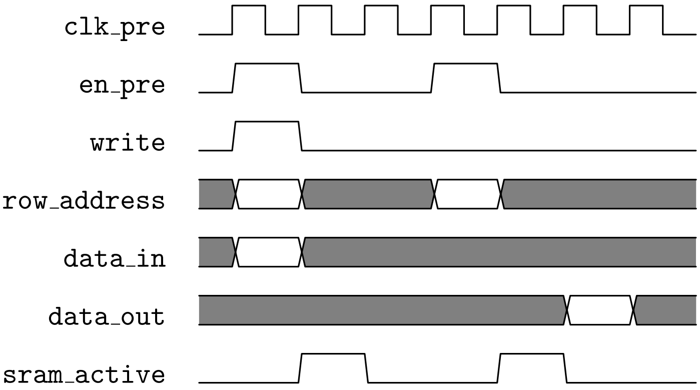


Figure 7.11: RTL timing diagram for the control signals of the *SynRAM* for a write and a read access, including the layer of integrated custom flip-flops. `sram_active` is an internal signal of the *SynRAM* control logic.

adjusted such that the ratio to the length of the actual bitlines is correct. A sketch of the layout implemented into the prototype chip is shown in Figure 7.12.

The layout of the actual memory block has physical dimensions of $376\,\mu\mathrm{m} \times 36\,\mu\mathrm{m}$. Overall, including memory, column and row driving circuits, interface flip-flops and the control logic the block has a size of $432\,\mu\mathrm{m} \times 74\,\mu\mathrm{m}$. The resulting ratio between actual memory area and total area is about 0.42, resulting in a large effective area consumption per bit. However, this low efficiency is mostly caused by the small dimensions of the block. For a full sized *SynRAM* block, featuring 1024×64 bits, the ratio between memory area to total area is 0.63. The flip-flops included for all inputs and outputs of the memory block are implemented as standard master-slave flip-flops in the current layout. One custom master-slave flip-flop covers an area of about $7.3\,\mu\mathrm{m}^2$. A more area efficient implementation that can be used instead is presented in Section 7.6.3.

Operating at 500 MHz and performing an access within every clock cycle, the `row` port interface of the memory block provides a bandwidth of 16 GB/s. The JTAG interface of the prototype chip is not able to provide such a high bandwidth. Therefore the digital controller connected to the memory provides 4 registers of 256 bit width each that can be written or read sequentially using the JTAG interface. Via additional JTAG commands, four consecutive write operations can be triggered. The data stored in the registers is written to four rows of the memory within four clock cycles. In a corresponding manner, four consecutive read operations can be performed. The data stored in 4 arbitrary rows of the SRAM is transfered into the registers of the controller within four clock cycles.

The nominal operation frequency of the `col` port is 250 MHz, only half of the frequency the `row` port is designed for. In the prototype chip the clock signal for the `col` port is derived from the main clock by a clock divider, reducing the frequency

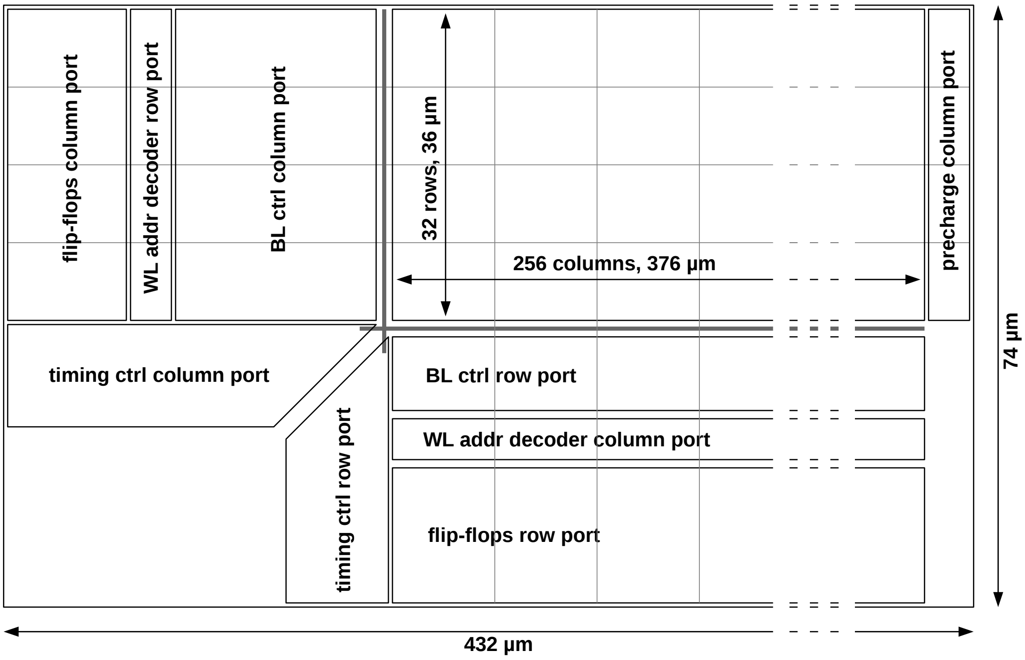


Figure 7.12: Sketch of the layout of the *SynRAM* block integrated into the prototype chip. The relative size of the components corresponds to their size in the actual layout. The location of the replica bitlines used for timing generation is marked by the gray bars.

by a factor of two. The bandwidth of the `col` port interface is lower. This port is 32 bit wide and operated at 250 MHz, resulting in a bandwidth of only 1 GB/s. To test the `col` port, the controller provides the same options as for the `row` port. The contents of four registers of 32 bit width can be written to the SRAM within four cycles of the `col` clock, corresponding to eight cycles of the chip clock. Accordingly, the content of 4 columns in the memory can be read to the registers at the full data rate.

As mentioned in 4.4, verification of the test controller and the memory block itself has been done using a mixed-signal simulation setup.

7.5 Experimental Results for the SynRAM

Both ports of the memory block have been tested at various settings of the timing parameters and for different clock frequencies. To verify if the memory is working correctly under the given parameters, the following test procedure is used. The procedure is identical for both ports, here it is described for the `row` port. As described in Section 7.4 the controller of the SRAM block can write or read four rows at the maximum speed. In the test sequence, four rows with the numbers m to $m+3$ are written with random data in four consecutive clock cycles. In the next step, these four rows are read back at the maximum speed and the data retrieved is compared to the data send. This is repeated 32 times for $m = 0$ to $m = 31$. For m being larger than 28 the incrementing of the row number results in values larger than 31. In this case the row number is wrapped over and the series is continued from $m = 0$. During one test cycle, every bit in the memory is written and read four times, in total 4 kiB of test data are transfered. Every row has been tested at every possible position in the sequence of the four consecutive accesses. To increase the statistical significance of the results, multiple test cycles are typically performed for each set of parameters.

One test cycle for the `col` port is the according procedure for writing and reading columns instead of rows. Four consecutive columns, starting from n, are written and read back at full speed. This is repeated for $n = 0$ to $n = 255$. During the procedure each bit in the array is written and read four times, a total amount of 4 kiB of random test data is processed.

7.5.1 Testing the Row Port

The `row` port allows for access to full rows of the memory, using the vertically oriented bitlines and the horizontally oriented wordlines. As described in Section 7.3.2, the timing generation can be tuned by means of two parameters, `PC_config` and `WL_config`. In a first step the fastest possible setting for the timing parameters is evaluated. Next the maximum clock frequency at which the `row` port can be operated is determined. Finally the reliability of the memory, when operated at its nominal clock frequency, is tested.

	PC_config	WL_config
Chip 2.1	4	6
Chip 2.2	4	6
Chip 2.3	4	6

Table 7.2: Optimized timing parameters for the `row` port. The results are identical for all three prototype chips.

Optimizing the Timing Parameters

The length of the precharge phase can be adjusted by changing the number of transistors which are charging the replica bitline, see Section 7.3.2. The available range is from 1 to 4 transistors, all of which have identical dimensions. The length of the period during which the wordline is active, is controlled by an additional replica bitline. Here the number of replica memory cells which discharge the line can be adjusted in a range from 1 to 8. For both parameters the respective time interval gets shorter the more transistors or replica cells are enabled. For the parameter optimization the clock frequency of the test chip is set to 500 MHz. The absolute timing in the *SynRAM* control logic does not depend on the clock frequency. The only constrain is that the SRAM accesses have to be fast enough to finish within one clock period. According to simulations, a period of 2 ns is sufficient for a memory access in the small block implemented in the prototype chip, even when `PC_config` and `WL_config` are set to low values. To determine the fastest possible setting for the precharge phase, the parameter `PC_config` is swept. During this test `WL_config` is set to 3, according to simulations a rather slow setting and sufficient for a reliable read process. For every value of `PC_config` 16 test cycles are performed, every bit in the memory is written and read 64 times. The maximum value for `PC_config` at which no error has been detected is the setting providing the maximum possible speed while still ensuring sufficient charging of the bitlines. This value is then used in the optimization process of `WL_config`. Again the array is tested using 16 cycles for every possible value of `WL_config` and the resulting error rate is recorded. This procedure has been repeated for three test chips. The maximum values which allow for reliable operation of the memory are presented in Table 7.2. The results are identical for all three chips. In case of the precharge configuration, the maximum possible setting of `PC_config` = 4 does not lead to any errors.

Sweeping the Clock Frequency

Next, the absolute time required for a single memory access, using the setting found by the parameter optimization process, is determined. The clock frequency is swept and at every frequency 16 test cycles are performed. As soon as the clock period becomes shorter than the time required to finish a memory access, errors are detected. Figure 7.13 shows the resulting error rate over clock frequency for

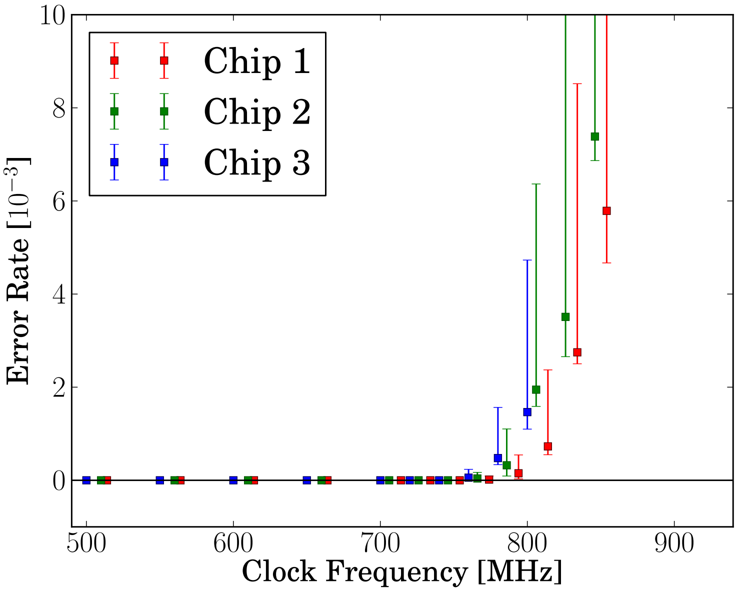


Figure 7.13: Error rate over clock frequency for SRAM accesses over the `row` port for the three test chips. The data points show the average over 16 repetitions of a test that writes and reads the full array 4 times. In total 32 kiB of data are written and read for each setting of clock frequency. The error bars indicate minimum and maximum error rate observed within the 16 runs.

the three test chips. Up to a clock frequency of at least 740 MHz all three chips are working reliably. For higher clock frequencies a steep rise in the error rate can be observed.

However, the errors observed at high frequencies are not necessarily caused by the memory being to slow to perform an access within one clock period. Another possibility is that the digital controller logic, providing the test data and reading it back, fails. The timing for the standard cell logic in the chips has been constraint and qualified for operation at up to 500 MHz, the nominal clock frequency of the chip. Assuming the prototype chips to be in typical process corner, correct operation can be expected for higher clock frequencies, as the verification process also covers chips in the slow process corner. Nevertheless, for frequencies about 50 % above the specified value, it is reasonable to assume that the controller logic fails.

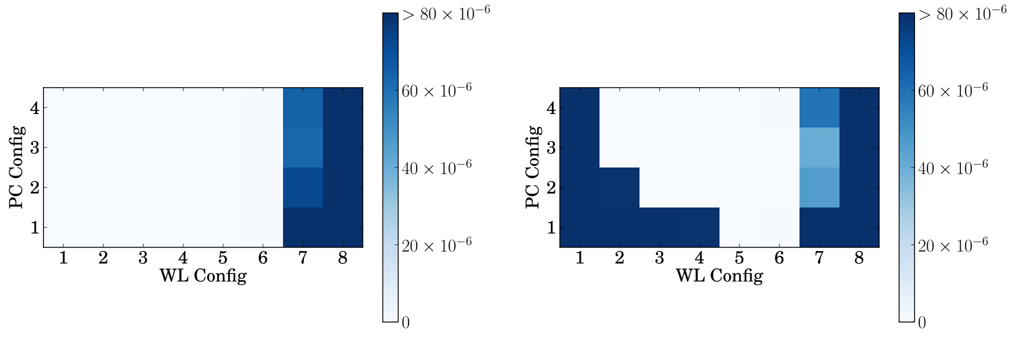
To distinguish between the possible scenarios for the failure of the memory tests at high frequencies, additional experiments are required. One possibility is to operate the memory at the maximum possible clock frequency, which had been determined using the optimized parameter set, and vary the timing parameters. Assuming that the memory needs the full clock period to perform an access operation, any change towards lower values, reducing the speed of the SRAM operation, will result in a high failure rate. If slowing down the memory operation still allows for reliable accesses, there is still headroom within the clock period. In this case the errors observed at high clock frequencies most likely originate in the digital control logic of the chip and not in the memory itself.

For chip 2.2 a full sweep over `PC_config` and `WL_config` has been performed at two different clock frequencies, 500 MHz and 740 MHz. The results are shown in Figure 7.14. According to Figure 7.14(a) the memory is working reliably for all settings of the timing parameters, except for the case `WL_config` is set to 7 or 8. This means that even at the slowest possible setting of `PC_config` $= 0$ and `WL_config` $= 0$ the duration of an access operation is short enough to be finished within 2 ns. As already seen in the results of the parameter optimization process, the precharge phase is sufficiently long for reliable operation, even at the fastest possible setting of `PC_config` $= 4$. For the wordline the situation is different. If the delay that controls the activation of the wordline is too short, the memory operation fails. This happens at settings of `WL_config` ≥ 7.

At a clock frequency of 740 MHz the situation changes, see Figure 7.14(b). The memory operation now fails if the parameters `PC_config` and `BL_config` are set to low values, resulting in slow memory operation. In these cases the access operation is not finished within one clock period. However, in relation to the fastest possible setting of `PC_config` $= 4$ and `BL_config` $= 6$, the memory operation can be slowed down significantly, still retaining reliable operation at a clock frequency of 740 MHz. This indicates that the errors which can be observed at frequencies slightly higher than 740 MHz are not caused by the memory itself but rather by failure of the control logic implemented in the chip. It is not possible to determine the maximum operating frequency of the `row` port for the small *SynRAM* block implemented on the prototype chip. Since the errors observed are not caused by the memory itself, no further insight can be gained from a detailed error analysis.

Extended Reliability Test

In the tests described so far, 16 test cycles have been used for each set of parameters. Every bit in the array has been written and read back 64 times. In total 32 kiB of random test data are processed. To further reduce the statistical uncertainty in the results a more extensive test is done. It uses the parameter setting obtained from the optimization process and the chips are operated at their nominal clock frequency of 500 MHz. In this test a total amount of 1.25 MiB is written and read, every bit is tested 1280 times. All three chips passed this test, no errors have been detected. Significantly larger amounts of data can not be tested since the JTAG interface provides only a low bandwidth connection to the prototype chips and the experiment duration becomes prohibitively long.



(a)

(b)

Figure 7.14: Error rate over the setting of the timing parameters, measured using the `row` port of chip 2.2. (a) shows the result for a clock frequency of 500 MHz, (b) is recorded at 740 MHz. For every data point 32 kiB are written and read back.

7.5.2 Testing the Column Port

Using the `col` port, full columns of the array can be read using the horizontally oriented bitlines and the vertically oriented wordlines. It is tested in an analogous manner to the procedures described for the `row` port. An important difference to the `row` port is that the clock controlling the `col` port is generated in the standard cell logic of the chip, its running at half the frequency of the main clock, see Section 7.4. All clock frequencies given in the following relate to the main clock.

Optimizing the Timing Parameters

Since the concept of the timing generation for the `col` port is the same as for the `row` port, the parameters `PC_config` and `WL_config` have been optimized using the same strategy as described in Section 7.5.1. The only difference is that in case of the `col` port the number of transistors available to control the length of the precharge phase is 8 instead of 4. For each setting 32 kiB of random test data have been used. The maximum parameters that allow for reliable operation of the memory are listed in Table 7.3. Other than for the `row` port, here the maximum setting is not identical for all chips. In case of chip 2.1 the timing for the wordline needs to be slowed down by one step of `WL_config`, compared to the two other test chips.

Sweeping the Clock Frequency

Next the clock frequency is swept to find the maximum value that still allows for reliable operation. Figure 7.13 shows the error rate of the `col` port over clock frequency at the optimal setting of the timing parameters. Up to a frequency of at least 800 MHz for the main clock all three chips are working reliably. Since the

	PC_config	WL_config
Chip 2.1	8	4
Chip 2.2	8	5
Chip 2.3	8	5

Table 7.3: Optimized timing parameters for the col port. Chip 2.1 requires a lower setting of WL_config than the other chips.

clock frequencies at which the digital controller is operating during the tests of the col port are even higher than for the tests of the row port, it is likely that again the digital control logic fails before the SRAM itself is operating too slow to finish an access within two cycles of the main clock.

To verify this, a full sweep over all possible settings of the timing parameters is done for chip 2.2 at two different clock frequencies. In Figure 7.16(a) the result for 500 MHz is shown. The situation is the same as for the row port, the memory is working correctly for all settings of the timing parameters, except for the case that WL_config is larger than 6. In this case the phase in which the wordline is activated is too short to allow for reliable accesses. Even the shortest possible setting for the duration of the precharge phase, PC_config = 8, results in sufficient charging of the bitlines.

The situation changes for a clock frequency of 820 MHz, see Figure 7.16(b). In case the timing for the wordline is set to the slowest possible setting, WL_config = 1, high error rates are measured. The duration of an access is too long to be finished within the two clock cycles of the main clock. For WL_config = 6, the reliability of the memory shows some dependence on the setting of PC_config. Operation at the fast wordline timing is only if the bitlines have been sufficiently precharged. For slower settings of the wordline configuration, reliable operation is also possible if the bitlines are only charged to a lower level. This is not surprising, except for the fact that the effect is not observed at a clock frequency of 500 MHz. A possible explanation is a drop of the supply voltage as a result of the higher current consumption when operating the memory at a higher frequency. However, for a wide range of parameter settings no errors are detected. Even for a setting of PC_config = 1 and WL_config = 2, which leads to significantly slower SRAM operation than with the optimized parameter set, the memory is working reliably. This proves that upper limit of the clock frequency is not determined by the memory itself but by the digital controller logic of the prototype chip. Again it is not possible to determine the maximum operation frequency for the small *SynRAM* block implemented in the prototype chip.

Extended Reliability Test

In the tests described so far, a maximum of 16 test cycles has been used to measure the error rate for a parameter set. To reach a higher statistical significance, the

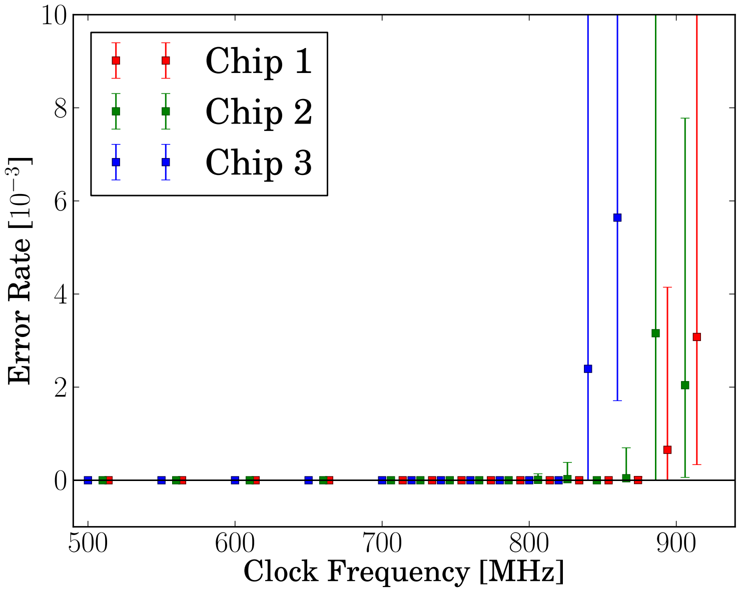


Figure 7.15: Error rate over clock frequency for SRAM accesses over the `col` port for the three test chips. The data points show the average over 16 repetitions of a test that writes and reads the full array 4 times. In total 32 kiB of data are written and read per frequency. The error bars indicate minimum and maximum error rate observed within the 16 runs. On the x-axis the clock frequency at which the digital part of the chip operates is shown. Note that the `col` port of the *Syn_RAM* is operating at half that frequency.

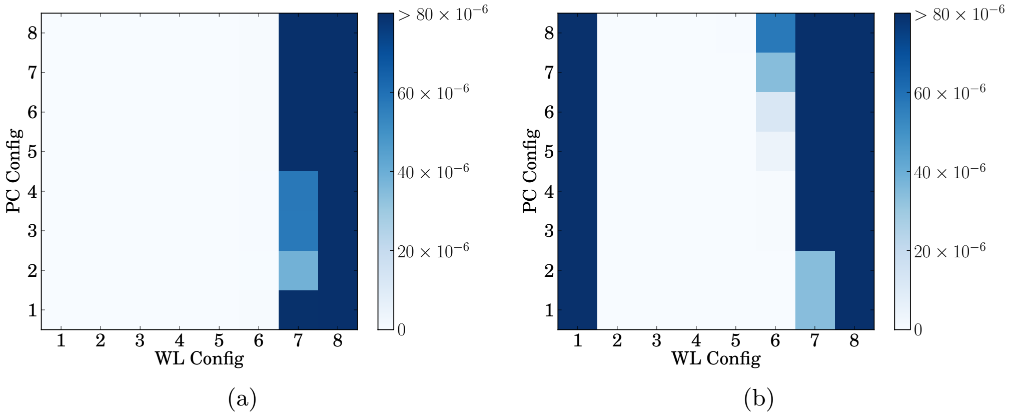


Figure 7.16: Error rate over the setting of the timing parameters, measured using the `col` port of chip 2.2. (a) shows the result for a clock frequency of 500 MHz, (b) is recorded at 820 MHz. For every data point 16 kiB are written and read back.

memory is tested extensively at the chips nominal clock frequency of 500 MHz. The timing parameters are set to the values found in the optimization process. In this test a total amount of 1.25 MiB is written and read, every bit is tested 1280 times. All three chips passed this test, no errors have been detected.

7.6 SynDAC

For the new synapse array architecture a current DAC is required to generate the post synaptic pulses which are delivered to the neuron circuits, see 7.1. It is designed to produce rectangular current pulses with a maximum amplitude of 8 μA and a width of a few nano seconds. The amplitude of the pulses needs to be adjustable with a resolution of 8 bit.

In Figure 7.17 an overview of the architecture of the *SynDAC* circuit is given. It is built from three parts. The analog part, performing the actual digital to analog conversion, is discussed in Section 7.6.2. The *pulse_ctrl* circuit is required by the analog part to generate precise current pulses, details are described in Section 7.6.4. In the synapse array the DAC needs to be connected to a layer of synthesized standard cell logic. As discussed in Section 7.3.4 for the *SynRAM*, the integration of the DAC is simplified if flip-flops are integrated for each digital input. However, in case of the *SynRAM* latches are used instead of flip-flops in order to save area. The details of the interface are described in Section 7.6.3.

Architecture and specifications for the *SynDAC* are, except for the increased resolution, comparable to the synapse DACs used in the HICANN chip. Therefore the existing design is briefly discussed in the following.

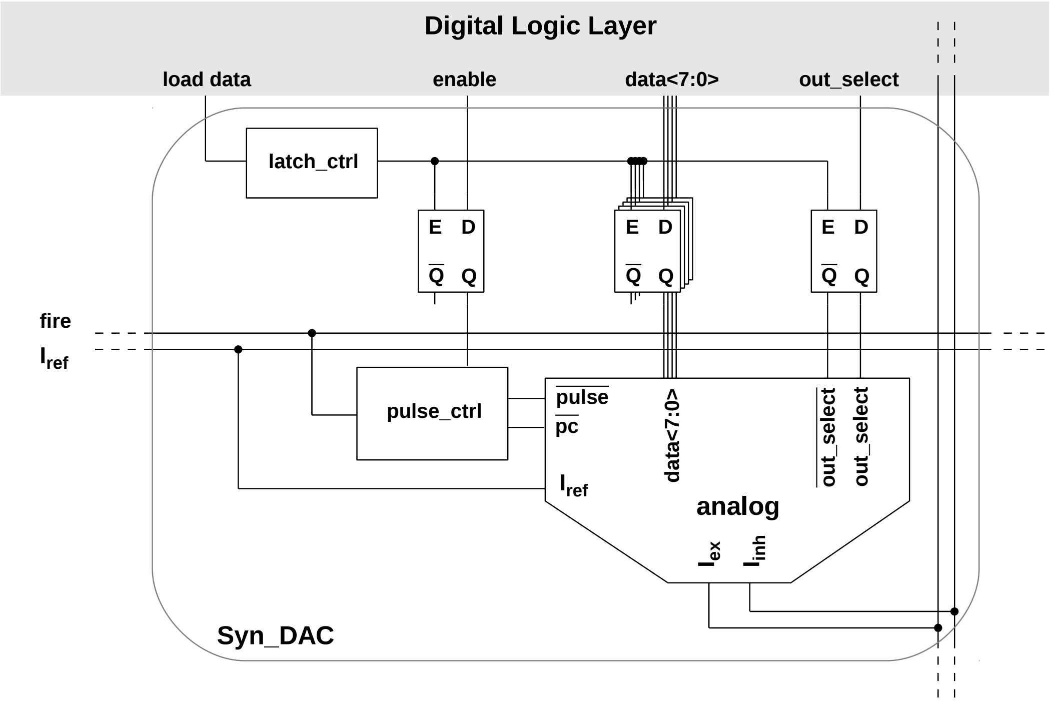


Figure 7.17: Block diagram showing the components of the *SynDAC* circuit.

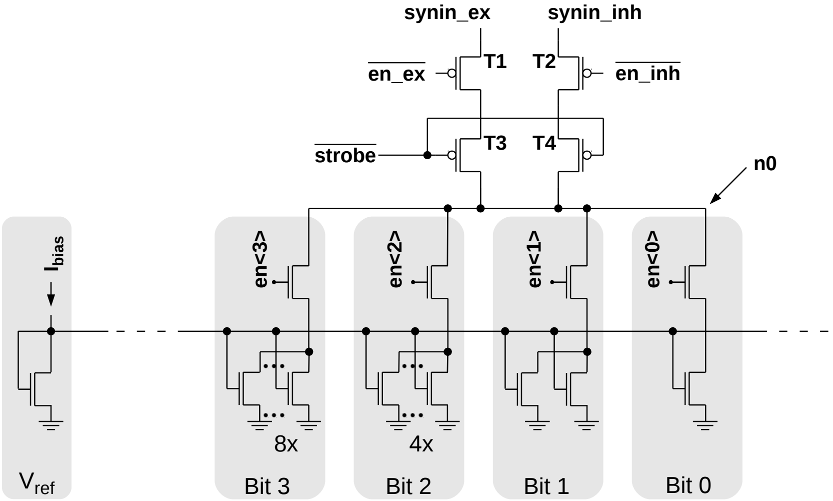


Figure 7.18: Schematic of the 4 bit current DAC used in the HICANN chip to generate postsynaptic events. The reference voltage which controls the current sources in the DAC is derived from a bias current in the *row_driver* circuit and distributed to the DACs in the associated row.

7.6.1 Synapse DAC of the HICANN chip

In the HICANN chip every single synapse is equipped with an individual unit element current DAC with a resolution of 4 bit. A schematic of the circuit is shown in Figure 7.18. Realizing a unit element DAC with 4 bit resolution requires 16 individual elements. Each element consists of a current source and a switch connecting the current source to the output of the DAC. The current source is implemented as a transistor, operated in the saturation regime, which replicates a reference current. The number of elements assigned to the individual bits starts at one for the LSB and is doubled for every subsequent bit. The digital input bits of the DACs in a synapse are statically connected to the memory bits holding the weight of the synapse.

In case of a synaptic event a current pulse is generated, whose amplitude is proportional to the weight of the synapse. The length of the pulse is controlled by the `strobe` signal which enables transistor T3. Each synapse can be configured to be either excitatory or inhibitory. The transistors T1 and T2 are used to statically connect the output of the DAC to the respective input of the neuron circuit.

The `strobe` signal is generated in the *row_driver* circuit and distributed along the respective row of synapses. The average duration of a pulse is 4 ns, the length is modulated by the STDF implementation in the *row_driver*.

In preparation of the development of the new DAC for the 65 nm process technology, the old design has been simulated. Some results are shown in Figure 7.19. Figure 7.19(a) shows a simulation of the output current of the DAC during a pulse for different digital codes. A significant overshoot at the beginning of the pulse is visible. Even for a digital code of 0, a short current pulse is produced as soon as the transistors T3 and T4 are enabled. The synaptic input circuit of the neuron integrates the amount of charge drawn by the current pulse. Figure 7.19(b) shows the amount of charge drawn per pulse at the output of the synapse DAC over digital codes in relative units. Due to the current peak at the beginning of each pulse, the integrated output signals show a significant offset. For a digital code of 0, an amount of charge corresponding to about 10 % of the maximum possible output value is drawn.

The reason for this offset is the parasitic capacitance of the node n0 connecting the individual elements to the transistors T3 and T4. In between pulses, this capacitance is discharged by the DAC elements which are active. When a pulse is produced, T3 and T4 get activated and a large current is flowing into the DAC, charging the node n0 and causing the peak visible in Figure 7.19(a). After n0 is charged to the voltage level at synaptic input of the neuron, the current is controlled by the current sources, as intended.

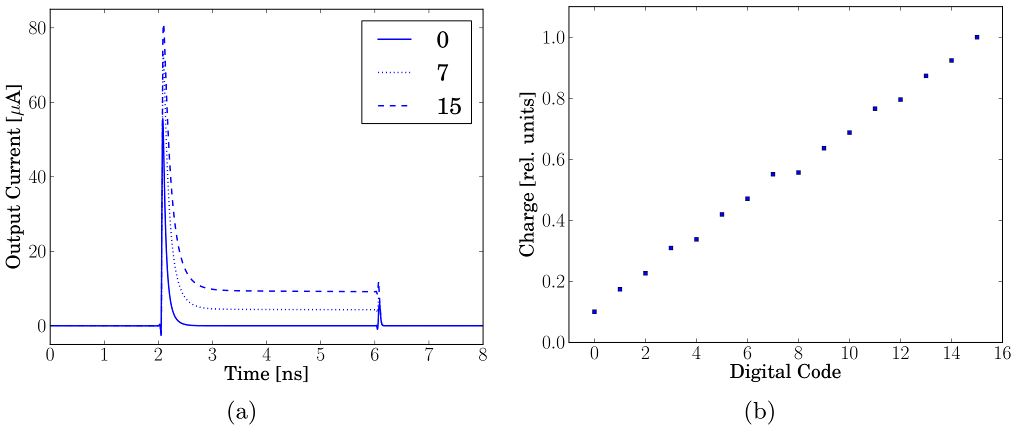
Unit element current DACs, especially when designed to produce short pulses, usually use the concept of current steering to prevent the circuit from dropping out of its operation point when inactive, see e.g. Myderrizi and Zeki [2010]. Current steering can not be used in the synapses, as the constant current would increase the power consumption of synapse array drastically. For a single HICANN chip which includes about 120 k synapses and a maximum output current of 8 μA for each synapse DAC, standard current steering would lead to a constant power consumption of 1 W. For a whole wafer, featuring 348 HICANN chips, the additional power consumption would be in range of 330 W. For the *SynDAC* design an alternative approach, presented in 7.6.4, is chosen to ensure that the transistors controlling the output current are at their respective operating point when producing an output signal.

7.6.2 Architecture of the Analog Part

The basic architecture of the *SynDAC* is inspired by a standard unit element current DAC. It is designed to conduct a controlled current from a node being held at 1.2 V to ground in short pulses. An overview of the analog part of the DAC is presented in Figure 7.20.

Realizing a standard unit element DAC with a resolution of 8 bit requires a total number of 256 identical current sources. The number of current sources assigned to the individual bits starts at one for the LSB and is doubled for every subsequent bit up to 128 current sources for the MSB. These current sources can be implemented as identical MOS transistors which are operating in the saturation regime and share the same gate potential. The resulting output current for each transistor is given by:

$$I_D = \frac{K'}{2} \frac{w}{l} (V_{GS} - V_{th})^2 \tag{7.1}$$



(a)

(b)

Figure 7.19: Simulation results for the 4 bit DAC generating postsynaptic events in the synapses of the HICANN chip. (a) shows the current produced at the output of the DAC for three different digital codes. Significant overshoot is visible at the beginning of the pulse, even for a digital code of `0`. (b) shows the amount of charge drawn by the DAC during a single pulse over digital codes. Note that the output signal at a digital code of `0` is in range of 10 % of the maximum possible output signal.

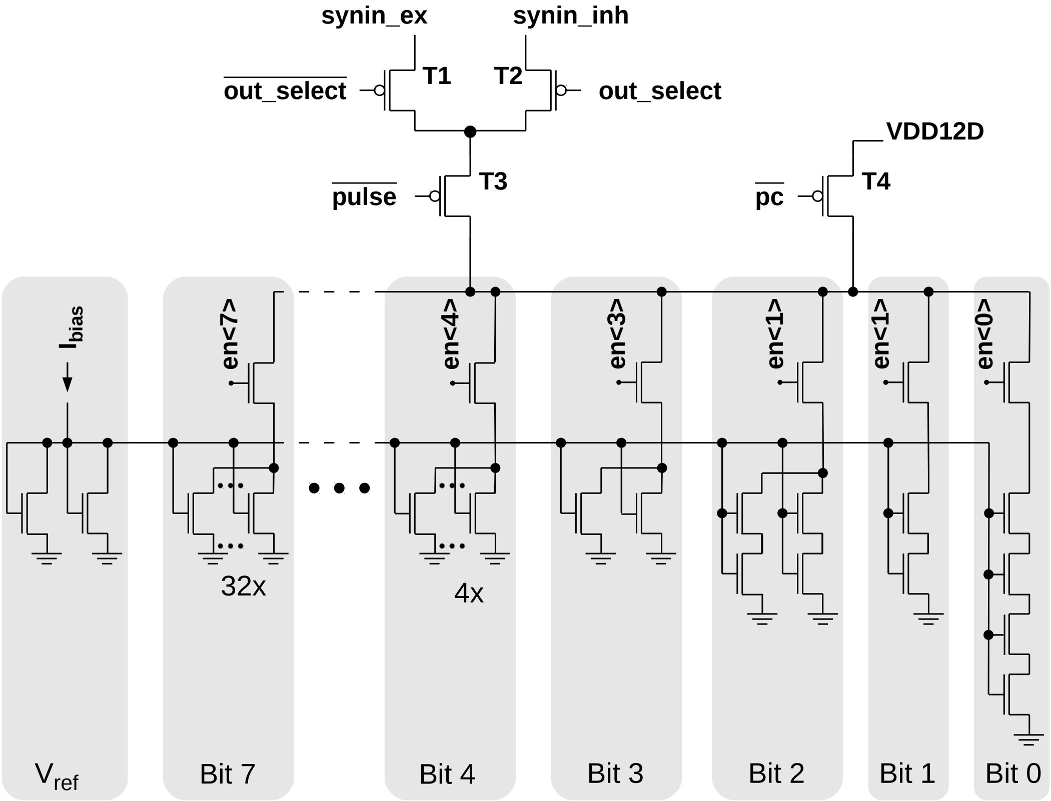


Figure 7.20: Schematic of the analog part of the *SynDAC*.

K' is a technology dependent constant, and w and l are the dimensions of the transistors gate. The effect of channel length modulation is neglected, see Section 7.6.5 for further discussion. The fact that the current sources are built from MOS transistors offers an option to reduce the required number of unit elements. Instead of using only parallel connection of transistors to realize the correct output current ratio between the sources assigned to the individual bits, it is also possible to use division of the currents by connecting multiple transistors in series. This is possible due to the geometry factor w/l in Equation 7.1. The current produced by a MOS transistors in inversely proportional to its length. Further the current produced by a serial connection of n MOS transistors with gate length l, sharing the same gate potential, corresponds to the current produced by a single transistor with a length of $n{\cdot}l$ [Galup-Montoro et al. 1994]. In the design of the *SynDAC* the two bits of lowest significance are realized by serial connection of two, respectively four, transistors. Instead of using a single transistor, bit 2 is implemented as an arrangement of 2 × 2 transistors to reduce the impact of device mismatch. By dividing the current for the two bits of lowest significance, the total number of transistors required is reduced from 256 to 73.

7.6.3 Digital Interface

Each DAC needs to be interfaced with synthesized standard cell logic. As discussed for the *SynRAM* block in Section 7.3.4, this is simplified if each input is equipped with a flip-flop. In case of the *SynRAM*, standard master-slave flip-flops have been used to synchronize the digital inputs and outputs of the design. However, these require 36 transistors per instance and a custom layout covers an area of about $7.3\,\mu\text{m}^2$. The area consumption can be optimized by using latches instead. Each instance covers only about half the area of a master-slave flip-flop. While latches are level sensitive, an edge triggered behavior is required for an interface operation synchronously to the clock signal. This can be emulated using a circuit that generates a short positive pulse whenever it is triggered by an edge of the clock signal. This functionality is implemented in the *latch_ctrl* unit, the according schematic is shown in Figure 7.21. For any static input, the output is `0` as the two inputs of the `AND` gate are always inverted to each other. Whenever the level at the input changes from `0` to `1`, the new state arrives earlier at the lower input of the gate and until the signal at the upper input turns `0` the output of the circuit is active. The duration of the pulse needs to be long enough to reliably exceed the hold time of the latch [Bhasker and Chadha 2009, Chapter 8.1]. A number of 5 inverters, built from transistors with a gate length of $\text{l} = 180\,\text{nm}$ and a width of $\text{w} = 200\,\text{nm}$, has been proven to be sufficient by simulations covering all process corner cases and Monte Carlo runs. The resulting pulse width is typically 140 ps and varies by about ± 32 ps for slow and fast process corner. One of these pulse generating circuits is implemented in every DAC, triggering the 10 latches which store the input signals provided by the digital control logic. Figure 7.22 shows a timing diagram for the signals involved in the process of loading data into the latches of a DAC. The latches

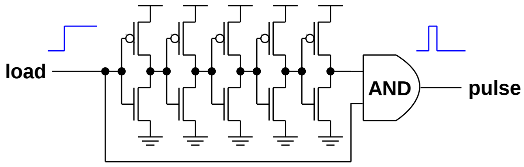


Figure 7.21: Schematic of the circuit generating the `latch_enable` signal which is required to store the input data in the latches.

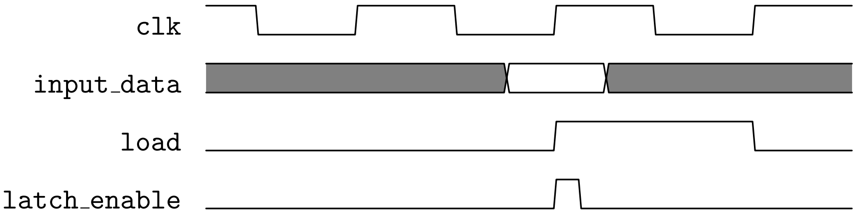


Figure 7.22: Timing diagram for the signals of the digital interface of the *SynDAC*

are implemented as transparent D-latches, built from four `NAND` gates. Each one requires 16 transistors and covers an area of $3.2\,\mu\text{m}^2$ in the layout. Including the area covered by one pulse generator for 10 latches, this implementation covers only 48 % of the area required to implement 10 custom master-slave flip-flops.

7.6.4 Pulse Generation

The length of the pulses generated by the DAC should be as short as possible in order to save power and to reduce the probability for collisions between presynaptic events. On the other hand the pulses need to be long enough so that the *row_driver* circuit is able to modulate its length with reasonable resolution and dynamic range, emulating the effect of STDF [Schemmel et al. 2007]. Furthermore the analog precision benefits from longer pulses, the relative impact of any distortions occurring at beginning or end of a pulse is reduced. The optimum pulse length for operation of the full system is not yet determined. It strongly depends on the implementation details of the *row_driver* circuits, which is not yet designed. An average pulse length of 2.5 ns has been assumed for simulations and experiments. Together with the maximum output current of $8\,\mu$A, the amount of charge transfered by one synaptic event is smaller or equal to 20 fC.

Typically the strategy of current steering is required to realize current pulses as short as 2.5 ns at the output of a current DAC. In Section 7.6.1 the effect of operating an unit element DAC without current steering is discussed. The internal node connecting all elements of the DAC is discharged in between pulses. Therefore a high current is drawn from the output of the DAC at the beginning of a pulse,

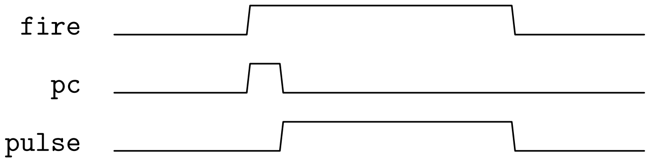


Figure 7.23: Timing relation between the signals used for the generation of output pulses for the *SynDAC*.

recharging the internal node of the DAC and disturbing the actual output signal. In case of the *SynDAC* the consequences are even more severe than for the DAC used in the HICANN chip. Due to the higher resolution, resulting in a larger number of elements in the DAC, the internal capacitance is significantly larger. Furthermore, due the shorter pulse length, the absolute amount of charge per pulse is smaller.

In the *SynDAC* the problem is solved by charging the internal node prior to every pulse. For this purpose transistor T4 is included in the design, see Figure 7.20. When a `fire` signal is sent to the DAC, the *pulse_ctrl* circuit generates a short pulse, activating the `pc` signal for about 200 ps. As a consequence, the internal node of the DAC is connected to the 1.2 V supply voltage by transistor T4. According to simulations, this time interval is sufficient for all internal signals of the DAC to settle at their operating point. Coincidentally with the deactivation of the `pc` signal the `pulse` signal is activated, connecting the DAC's output to the either the inhibitory or the excitatory input of the assigned neuron circuit. The timing of the signals involved in the generation of a pulse is shown in Figure 7.23. The *pulse_ctrl* unit uses the same circuit as shown in Figure 7.21 to generate the precharge pulse. Since the length of the precharge pulse is less than 10 % of the actual pulse length, the impact on the overall current consumption is negligible.

Using the 1.2 V supply voltage for precharging the DAC is beneficial for the layout as no separate net needs to be routed. However, this requires the neuron circuit to use a voltage level of 1.2 V as a reference potential for the integrator circuit at its synaptic input. Assuming that the neuron circuit for the new system will be based on the design of current neuron design used in the HICANN chip, this constraint can be met. In the current neuron design the reference level of the synaptic input circuit is typically set 1 V, close to the half of the 1.8 V supply voltage [Millner 2012, Chapter 3]. The neuron circuits used in a system based on the 65 nm process technology will probably be built from the thick-oxide transistors using a supply voltage of 2.5 V. Therefore it seems possible to realize an integrator circuit operating with a reference level of 1.2 V for the synaptic input.

7.6.5 Accuracy

The dimensions of the transistors operating as current sources are crucial for the accuracy of the DAC. These have an impact on the output resistance of the transistors as well as in the device mismatch affecting the characteristics of the DAC. For the design described here, the dimensions l = 1 μm and w = 0.25 μm have been chosen.

Output Resistance

The output resistance of the transistors is limited by the effect of channel length modulation. Accounting for this effect, the drain current of a MOS transistor in the saturation regime is given by:

$$I_D = \frac{K'}{2}\frac{w}{l}(V_{GS} - V_{th})^2 \cdot (1 + \lambda V_{DS}) \tag{7.2}$$

The factor λ accounts for the channel length modulation, introducing a dependency of the drain current on the drain-source voltage. Its value depends on the absolute length of the transistor. For a detailed discussion of channel length modulation see [Razavi 2001, p. 25].

However, in the application discussed here, the output resistance is not critical because the DACs output will be kept at a constant voltage level. The wire connecting the output of all *SynDACs* to the synaptic input circuit of a neuron is crossing the full synapse array, its absolute length will be in range of a few millimeters. The parasitic capacitance of this wire helps to maintain a constant level at the DACs output during the generation of a pulse. Additionally the switch transistor of every enabled element acts as a cascode stage, reducing the voltage swing at the drain of the transistor used as current source, [Sansen 2006, p.92]. Simulations suggest that for the chosen transistor dimensions a voltage drop of 100 mV at the output of the *SynDAC* leads to an error in range of only 0.2 % for the output current.

Device Mismatch

More important for the accuracy of the DAC than the output resistance is the impact of device mismatch. The design of a unit element DAC is based on the assumption that each of the enabled elements, supplied with the same gate voltage, contributes the same amount current to the total output current. As mentioned in Chapter 5.3, the characteristics of the individual elements are subject to statistical variation. In a standard unit element design, the total output current I_{out} of the DAC at a digital code of `k` is:

$$I_{out}(k) = \sum_{i=1}^{i=k} I_{elem,i} \tag{7.3}$$

As the output current is the sum of the independent currents contributed by the individual elements of the DAC, the error of the output current $\Delta I_{out}(k)$ is therefore given by:

$$\Delta I_{out}(k) = \sqrt{k} \cdot \Delta I_{elem} \tag{7.4}$$

For many applications of DACs the differential non linearity is an important characteristic, it is defined by:

$$DNL(i) = \frac{I_{out}(i+1) - I_{out}(i)}{1LSB} - 1 \tag{7.5}$$

where $1\,LSB$ equals the ideal step size for a change of the least significant bit in the input data. For a current DAC with n bit resolution it is defined by:

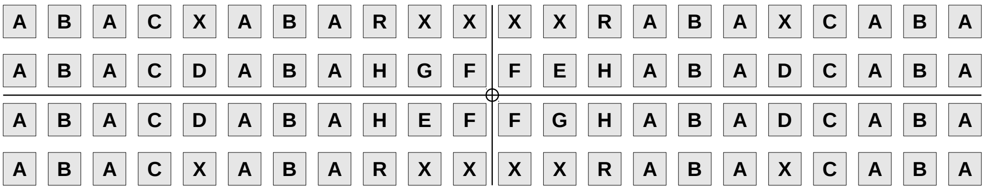
$$1\,LSB = \frac{I_{out}(2^n)}{2^n} \tag{7.6}$$

The DNL describes the deviation between the change of the output current when the digital code changes by `1` and the expected change of 1 LSB. If the unit elements are enabled individually, according to the the digital input code, the DNL equals ΔI_{elem} for all transitions. However, the translation of the n bit of binary input data into 2^n individual control signals requires a significant amount of logic. This approach is known as "thermometer code" architecture [Myderrizi and Zeki 2010]. In most designs of unit element DACs each individual bit m of the input data is statically connected to the enable signals of a group of 2^m elements. The output current is the sum over the output currents of the activated bits. The error of the output current for each individual bit m is given by Equation 7.4, where k is 2^m. The static assignment of elements to the individual bits of the binary input data has severe consequences for the transitions between consecutive codes. For a transition towards a code which is represented by a single active bit m, the output currents before and after the transition are generated by non overlapping groups of elements. As a result, the error on the change in output current at such a transition is given by:

$$\begin{aligned} \Delta(I_{out}(2^m) - I_{out}(2^m - 1)) &= \sqrt{(\Delta I_{out}(2^m))^2 + (\Delta I_{out}(2^m - 1))^2} \\ &= \Delta I_{elem} \cdot \sqrt{2^{m+1} - 1} \end{aligned} \tag{7.7}$$

This results in the maximum possible error for m being the most significant bit. For an n bit DAC, the worst DNL can be expected for the transition between the digital codes 2^{n-1} and $2^{n-1} - 1$, which happens in the middle of the dynamic range. In this case the output current between largest possible independent groups of elements are compared.

For the *SynDAC* the situation is slightly different than described so far, as it is not a strict unit element design. The two bits of lowest significance are generate by dividing the current in the element by two, respectively by four, using a serial connection of transistors. The error of these two bits is not necessarily related to the errors of the other bits by Equation 7.4. However, the basic results of the considerations presented above still hold true. At the transition between the digital codes 2^{8-1} and $2^{8-1} - 1$, the largest groups of non overlapping elements are involved in the transition, leading to the largest expected DNL and highest probability for non monotonic behavior. According results are obtained from experiments, see Figure 7.25.



A	B	A	C	X	A	B	A	R	X	X	X	X	R	A	B	A	X	C	A	B	A
A	B	A	C	D	A	B	A	H	G	F	F	E	H	A	B	A	D	C	A	B	A
A	B	A	C	D	A	B	A	H	E	F	F	G	H	A	B	A	D	C	A	B	A
A	B	A	C	X	A	B	A	R	X	X	X	X	R	A	B	A	X	C	A	B	A

Figure 7.24: Sketch of the arrangement used for the single elements in the layout of the DAC. The letters A to H are assigned to Bit 7 to Bit 0, R marks the two transistors used to generate V_{ref}, X marks dummy transistors ensuring homogeneity of the layout. See also Table 7.4.

7.6.6 Layout

The digital part of each DAC, including the latches storing the input data, the logic used to generate the timing for the precharge operation and the switches which connect the current sources to the output, covers an total area of $8.45\,\mu$m $\times\ 11.76\,\mu$m $\approx 100\,\mu$m^2. In the analog part, the 72 single transistors operating as current sources are arranged within a regular array of 22 columns and 4 rows. To achieve good relative matching between the individual bits, the associated groups of transistors are arranged symmetrically to the center of the array. A sketch of the transistor array is shown in Figure 7.24, the meaning of the letters is shown in Table 7.4. The arrangement is a trade-off between an optimum common centroide layout [Long et al. 2005] and a structure which requires only limited routing resources. In the layout used in the prototype chip two diode-connected transistors are used per *SynDAC* to generate V_{ref}, see Section 7.6.7 for details. Some positions in the array are covered by filler transistors which have the primary function to keeping the layout homogeneous. Further these are used as additional blocking capacitance for V_{ref}.

7.6.7 Implementation Details

A row of 32 *SynDACs* is implemented in the second prototype chip. All *SynDACs* share the same reference voltage V_{ref}. In every instance, two diode-connected transistors are integrated into the transistors array to generate V_{ref} from an externally supplied bias current. Therefore the bias current is distributed among 64 individual transistors in total. As a consequence the external bias current equals the maximum output current of an individual *SynDAC*. Accordingly an external bias current of $8\,\mu$A is supplied.

A digital controller, written by Andreas Hartel, is used to generate the input data for the *SynDAC* in the test chip. The the same 8 bit word is written to the latches of all instances. The `enable` bit as well as the `out_select` bit can be set individually for each DAC. Two different modes for the `fire` signal can be selected. It is possible to configured it to be continuously active or to be activated

Device	#	Function
A	32	Bit 7
B	16	Bit 6
C	8	Bit 5
D	4	Bit 4
E	2	Bit 3
F	4	Bit 2
G	2	Bit 1
H	4	Bit 0
R	2	generating V_{ref}
X	14	Filler, blocking V_{ref}

Table 7.4: Function and number of the individual transistors in the layout of the analog part of the *SynDAC*, see Figure 7.24

periodically. The output of each *SynDAC* can be connected directly to a bond pad of the test chip using multiplexers. This allows for precise measurements of the DACs output in constant current operation, see Section 7.7.1.

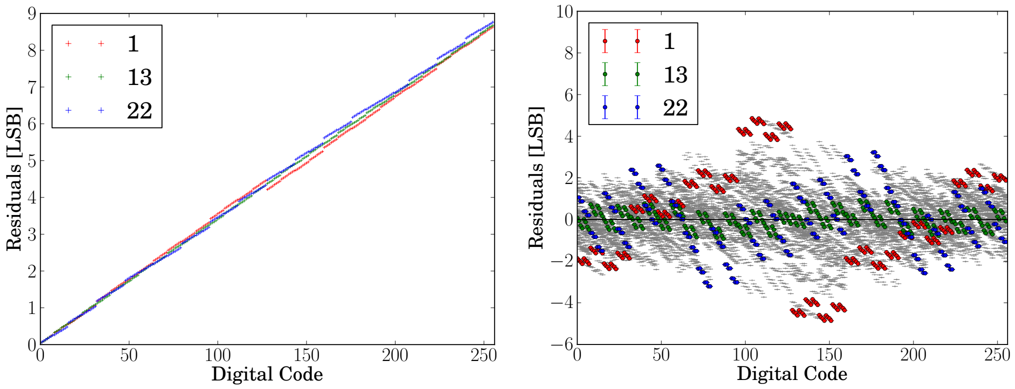
To test the DACs performance in pulsed operation, the digital controller can be configured to periodically activate the `fire` signal, period and pulse width can be configured in multiples of clock cycles. To characterize the *SynDAC* in pulsed operation, the single current pulses can be integrated with help of an on-chip capacitor, see Section 7.7.2.

7.7 Experimental Results for the SynDAC

Initially the DAC is tested in continuous current mode. This allows for precise measurement of the mismatch for the individual groups of transistors which are assigned to one bit. In a second step current pulses are generated and the amount of charge transfered by the pulses is measured. A general description of the experimental setup used for testing of the prototype chips is given in Chapter 4.5.

7.7.1 Continuous Current Operation

Testing the DACs performance using static currents allows for precise evaluation of the errors introduced by device mismatch. In this setup the `fire` signal is configured to be constantly active. An arbitrary 8 bit value is written to the input data registers of the DACs. Only a single DAC is enabled using its internal `en` bit and the corresponding multiplexer is activated such that the selected DAC is directly connected to a bondpad of the test chip. The current flowing into the activated DAC is measured using the Sourcemeter which is producing a voltage of 1.2 V at its output.



(a) (b)

Figure 7.25: (a) Output current of three *SynDACs* from chip 2.2 over digital codes. Out of the 32 instances, two which are severely affected by device mismatch (DAC 1 and DAC 22) and one with good matching (DAC 13) have been chosen as examples. The measurement error on the current, obtained from 8 independent repetitions, is below 4 nA for all data points and therefore too small to be visible in the plot. (b) Residuals of the output current of all DACs from chip 2.2 over digital codes. The residuals for the three *SynDACs* shown in (a) are highlighted.

The output current was measured against digital codes for all DACs on the four test chips 2.0 to 2.3, in total 128 individual DACs are tested. Figure 7.25(a) shows the output current of three *SynDACs* from chip 2.2 over the digital codes. Out of the 32 instances, two which are severely affected by device mismatch (DAC 1 and DAC 22) and one with good matching (DAC 13) have been chosen as examples. The impact of device mismatch is clearly visible and leads to a significant differential non linearity for the DACs. Figure 7.25(b) shows the residuals against a linear fit for all *SynDACs* from chip 2.2. The residuals for the three *SynDACs* shown in Figure 7.25(a) are highlighted.

As expected, see Section 7.6.5, the transition between the states `0111 1111` and `1000 0000` is the most problematic one. As an example, for *SynDAC* 1 in Figure 7.25(a), the output current is decreased by almost 8 LSB instead of being increased by 1 LSB. Figure 7.26 shows a histogram of the current step measured at the MSB flip for the 128 individual DACs tested. The measurement shows an average step size of 0.88 LSB and a standard deviation of 3.62 LSB. For an ideal DAC a step of 1 LSB is expected. Consequently the average DNL(127) is - 0.12 ± 3.62 for the *SynDAC*. Non-monotonic behavior, a decrease instead of an increase in output current, is observed at the transition from `127` to `128` for about 40 % of the DACs. A Monte Carlo simulation of the corresponding setup, based on 2000 individual parameter sets, shows an average step size of 1.2 LSB and a standard deviation of

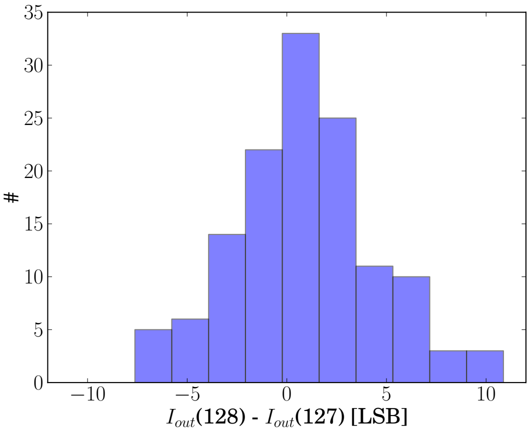


Figure 7.26: Histogram of the difference between the output current at a digital code of 128 and the output current at 127 for the 128 individual SynDACs available on the prototype chips 2.0 to 2.3. The step size is expected to be 1 LSB.

4.1 LSB.

In order to ensure monotonicity for the DAC, the DNL must be reliably smaller than 1 for all input codes. For the presented *SynDAC* however, values of up to 3.5 can be expected within 1 standard deviation for the DNL(127). In this case the uncertainty of the DNL is proportional to the relative error of a single element used in the DAC, see Equation 7.7. Since the relative error of the output current of a single element can be assumed to be proportional to the factor $1/\sqrt{w \cdot l}$ [Lovett et al. 1998], a drastic increase in the transistor size would be required to ensure monotonicity for the *SynDAC*. An option to reliably avoid the issue of non-monotonicity in a unit element DAC is to change the architecture to a design based on thermometer coding [Myderrizi and Zeki 2010].

7.7.2 Pulsed Operation

The DAC is designed to produce short current pulses. As the synaptic input circuit of the neuron integrates the signals generated in the synapse array, the amount of charge that is drawn per pulse by the DAC needs to be evaluated. To test the DAC in pulsed operation, the digital controller in the prototype chip allows for activation of the `fire` signal. Pulse rate and pulse width are configurable in multiples of clock cycles, additionally the frequency of the externally supplied clock can be changed to precisely generate the desired timing. For the measurements presented in the following, the clock is set to 400 MHz and the pulse width is set to 1 cycle, resulting in a pulse duration of 2.5 ns. The circuit shown in Figure 7.27 is integrated into the prototype chip and can be used to measure the output characteristics of the DACs implemented. The capacitor C1 is first charged to 1.2 V and then discharged by the

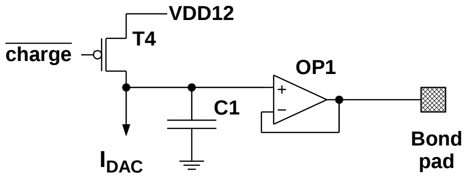


Figure 7.27: Schematic of the circuit integrated into the second prototype chip to measure the amount of charge drawn by the DAC when producing pulses.

current drawn during a pulse by the active DAC. The voltage on the capacitor is buffered and connected to a bond pad for external measurement. C1 is implemented as a MIM Cap and has a nominal capacitance of 919 fF. A single pulse of 2.5 ns duration and 8 μA amplitude discharges the capacitor by about 22 mV.

The absolute measuring accuracy of this method is limited. Due to device mismatch the capacitance of C1 is not known precisely, furthermore the parasitic capacitance of the wire connecting the output of all DACs to C1 changes the effective capacitance. Nevertheless, the circuit allows for reliable measurements of the relative amount of charge that is drawn by a pulse of the DAC. It is important that the voltage on the capacitor does not drop too much since the DAC is specified to operate at a constant output voltage of 1.2 V. As mentioned in Section 7.6.5, an error of about 0.2 % for the output current is expected in case the output voltage changes by 100 mV. The impact of a drop in range of 20 mV is assumed to be negligible. For the measurements presented in this section, the following measurement protocol has been applied. The `charge` signal is activated for 40 ns, charging the capacitor C1 to 1.2 V. Next a pulse of 2.5 ns length is generated in a single DAC, discharging C1. This is followed by a break of 1280 ns before the capacitor is charged again. The voltage level on the capacitor after the pulse can be sampled during the break period using an oscilloscope. But since only relative measurements are of interest it is also possible to measure the average voltage on the capacitor during a continuous repetition of the measurement protocol. The Sourcemeter, which is configured to perform analog integration over a period of 20 ms for each sample, is used to measure the average voltage on the capacitor. The voltage drop is proportional to the relative amount of charge which is drawn by the DAC in every pulse.

Figure 7.28 shows the relative amount of charge drawn by a DAC over digital code in pulse operation. The measurement results for all 96 *SynDAC* circuits on the test chips 2.1 to 2.3 are shown. The data is obtained from the measured voltage drop on the readout capacitor as described above. The average measurement error for each data point, obtained from 8 independent repetitions of the measurement, is 0.0025. Additionally corresponding simulation results are shown. The simulation setup includes all 32 *SynDAC* circuits and the readout capacitor. Only one of the *SynDACs* is producing current pulses of 2.5 ns width, the remaining ones are

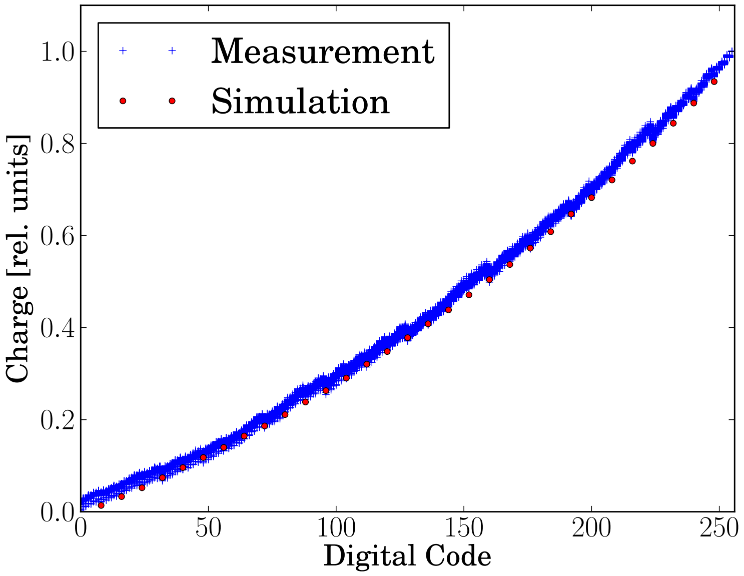


Figure 7.28: Output characteristics of 96 *SynDACs* in pulsed operation. The relative amount of charge drawn by a single pulse is plotted against digital codes for all DACs. The pulse duration was set to 2.5 ns. Additionally the simulation results for the corresponding setup are shown. The average measurement error for the single data samples, obtained from 8 independent repetitions of the measurement, is 0.0025 and therefore too small to be visible in this plot. In pulsed operation the output of the DAC shows significant non-linearity.

disabled. The simulation results match the measurement results.

Operating the DAC with pulses obviously degrades the linearity of the output. This effect is also visible in simulations and its origin has been identified. When the DAC produces a pulse, the voltage at the drain terminal of the active transistors rises quickly. This voltage slope couples through the parasitic drain-to-gate capacitance of the active transistors onto the reference voltage, leading to an increased output current of the DAC. The effect gets stronger the more transistors are active, therefore the deviation of the output is larger for higher digital codes. Figure 7.29 shows a simulation of the reference voltage V_{ref} over time for different digital codes. Note that the absolute amplitude of the distortions on V_{ref} is below 1 mV.

Additional blocking capacitance for the reference voltage can be used to counteract the effect. But to achieve a noticeable effect, very large amounts of capacitance are required. In the experiment presented, only one of the 32 DACs on the chip is active, consequently the gate-to-source and gate-to-drain capacitances in all transistors of the 31 inactive DACs serve as blocking capacitance for the reference voltage. Nevertheless the effect is clearly visible in simulation and measurement.

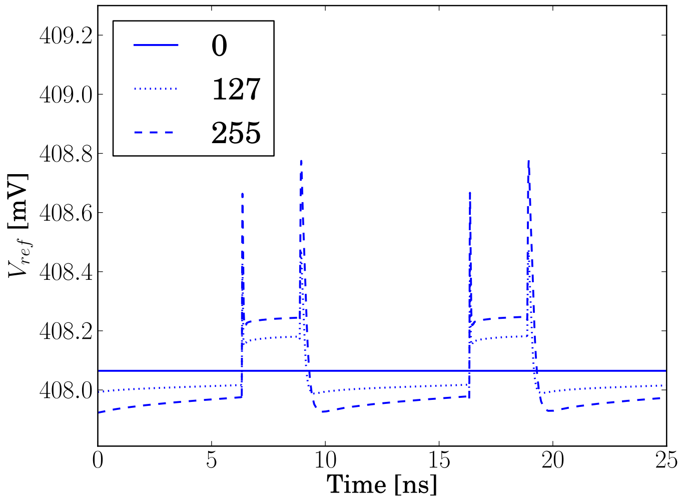


Figure 7.29: Reference voltage over time for different digital codes programmed to a *SynDAC* in pulsed operation. Every 10 ns a pulse of 2.5 ns length is triggered. The distortion of V_{ref} is responsible for the non-linearity of the *SynDAC* in pulsed operation.

An important aspect, visible in Figure 7.28, is that for a digital code of `0` the resulting output signal is close to 0. The average relative output signal at `0` for all 96 DACs tested in pulsed operation is 0.020 ± 0.003 (in relative units). This is an significant improvement compared to the synapse DACs used in the HICANN chip were the minimum output signal possible is in range of 10 % of the maximum output signal, see Section 7.6.1. The low standard deviation over all DACs located at the three different chips 2.1 to 2.3 shows that a systematic effect is causing the remaining deviation, suggesting that better results can be obtained by further optimization. Probably the timing for the precharge phase is too aggressive, extending the length by 50 % might be sufficient to achieve better results.

8 Neurons in the 65 nm Process Technology

Emulation of neuron behavior is typically the most complex task on the circuit level within the development of analog neuromorphic hardware. In literature a large number of mathematical models describing the dynamics of neurons on various levels of accuracy can be found. A very common but rather simplistic description of neuron behavior is the Leaky-Integrate-and-Fire model (LIF) [Pospischil et al. 2011]. Models such as the Hodgkin-Huxley model on the other hand are very accurate, accounting for the individual types of ion channels found in biology [Hodgkin and Huxley 1952]. A high level of detail increases the effort required for implementation, typically leading to a high area and energy consumption of the circuits. The neuron circuits described in Chen et al. [2010] are highly accurate implementations of Hodgkin-Huxley models. However, such an approach results in a large area consumption per neuron, only two neuron circuits fit onto a single chip.

For large-scale applications, aiming at the implementation of neuron numbers of biological relevance, a trade-off between model accuracy and the consumption of resources has to be found. The neuron implementation used in the BrainScaleS Hardware System is based on the Adaptive Exponential Integrate-and-Fire model (AdEx), published by R. Brette and W. Gerstner [Brette and Gerstner 2005]. A detailed description of the neuron circuit can be found in Millner [2012, Chapter 3].

As other circuits in the BrainScaleS Hardware System, the design of the neuron circuit has been developed, tested and improved over many years. In order to benefit from the circuits already available, the option of porting the existing neuron implementation directly to the 65 nm process technology using thick-oxide transistors is evaluated. In Chapter 5.5 the successful transfer of an operational amplifier is described. The same strategy was used to transfer the entire neuron circuit. The focus during this process was to test whether it is possible to rebuild the complex analog circuit in the 65 nm process. Despite known problems and limitations, the basic design was not modified. The goal was to achieve matching behavior between the circuits, there was no intention to improve the circuit.

The UMC 180 nm process has a core supply voltage of 1.8 V, the thick-oxide transistors of the TSMC 65 nm process technology are designed to operate with 2.5 V. It is possible to operate circuits with only 1.8 V in the new process technology. However, complex analog circuits, such as the neuron implementation, can benefit significantly from the higher supply voltage. The additional voltage headroom can be used to increase the dynamic range of the signals involved. The transfered circuit of the neuron is operated at a supply voltage of 2.5 V. For comparison to the original design, operating with a supply voltage of 1.8 V, the absolute voltage levels are shifted accordingly.

8.1 Transfer of Neuron Components

The neuron implementation uses multiple instances of operational amplifiers as well as an operational transconductance amplifier. In a first step the amplifiers used in the neuron circuit have been transfered, using the same strategy as described in Chapter 5.5. The schematic of the circuits and all transistor dimensions are kept identical to the original design. In case these dimensions violate the minimum width or length restrictions of the thick-oxide devices, the dimensions of the respective transistor are enlarged, keeping the w/l ratio identical to the original design. Finally the dimensions of individual transistors are tuned, based on simulations comparing the transfered circuit against the original design. The aim was to keep the characteristics as close as possible to the original circuits. As one example for a transfered amplifier, the operational transconductance amplifier used in the neuron is compared to the original circuit in the following.

8.1.1 The Operational Transconductance Amplifier

The operational transconductance amplifier (OTA) is an important component of the neuron circuit, it is used 7 times in each instance. A detailed description of the original circuit can be found in Millner [2012, Chapter 3.3]. The characteristics of an ideal OTA are such that it produces an output current which is proportional to the voltage difference between its inputs, see Equation 8.1.

$$I_{out} = g_m(I_{bias}) \cdot (V_{in+} - V_{in-}) \tag{8.1}$$

The transconductance g_m of an OTA typically depends on the bias current supplied to the circuit. In the neuron circuit the g_m OTA is required to have a linear dependency on I_{bias}.

In Figure 8.1 some characteristics of the original 180 nm design (left panels) and the transfered 65 nm version (right panels) are compared. In the first row the output current is shown as a function of the differential voltage at the input for different bias currents. In both cases the common mode of the input voltages is set to half of the supply voltage. The range in which the OTA is operating in a linear fashion is similar for both implementations. However, the systematic input offset, the differential input voltage at which an output current of 0 is produced, is reduced from 8 mV to 4 mV in the 65 nm version. The second row shows the output current as a function of the differential input voltage for different common modes of the input voltages. A bias current of 1 μA is used. The input common mode is shifted by ± 400 mV relative to half of the supply voltage. The 65 nm version shows a higher common mode rejection than the original OTA. The third row shows the output current as a function of the differential input voltage for different voltages at the output. The bias current is again 1 μA. The output voltage is shifted by ± 400 mV relative to half of the supply voltage. Ideally the output current of the OTA should not depend on the voltage at the output.

The AC characteristics of both versions of the circuit are also comparable. At a bias current of 1 μA, the $-3dB$ bandwidth of the 65 nm version is 115 MHz, compared to 92 MHz for the 180 nm version. Overall, the transfered version provides equal or even better performance than the original circuit.

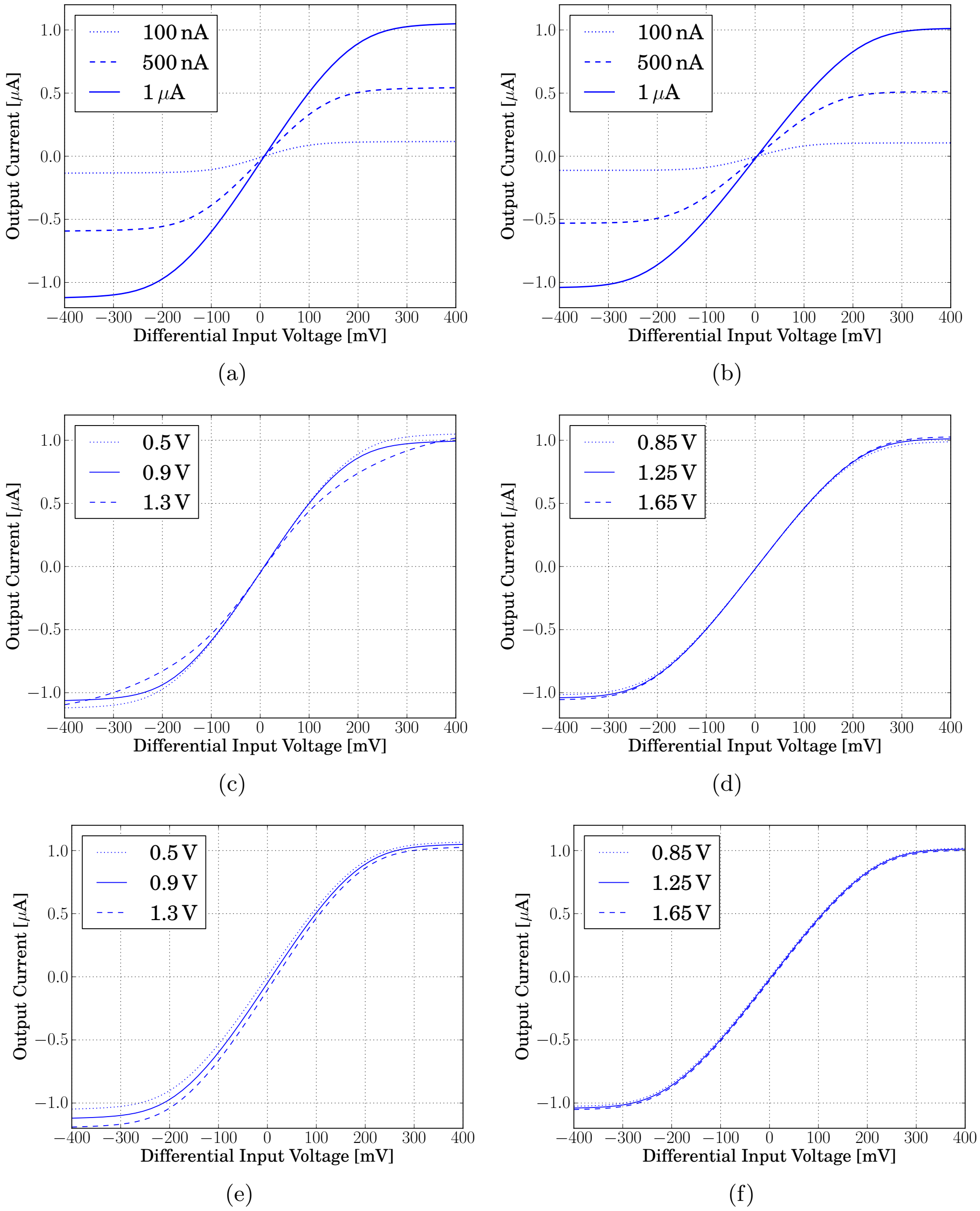
8.2 The 65 nm Neuron Circuit

Following the amplifiers, the individual subunits of the neuron circuit have been ported to the 65 nm process and their performance was individually compared to their original counterparts. Finally all components were connected to a full replica version of the neuron circuit, built entirely from thick-oxide transistors in the 65 nm process technology. The schematic is identical to the original design, only the dimensions of the transistors have been modified.

8.2.1 Stimulating the 65 nm Neuron Circuit

In order to test the neuron circuit, its reaction to an external current injected onto the membrane is investigated. Using an external current stimulus to test and characterize the behavior of neurons is discussed in e.g. Izhikevich [2004]. These experiments have been performed for the 180 nm neuron circuit, several of the patterns described in Izhikevich [2004] have been successfully reproduced in simulations and measurements by S. Millner [Millner 2012, Millner et al. 2010].

Different firing patterns can also be observed in simulations of the 65 nm neuron implementation, examples are presented in Figure 8.2. During these simulations the synaptic input circuit of the neuron was disabled because a fundamental design problem was identified during experiments with the 180 nm neuron design [Kleider 2014]. So far no extended quantitative characterization of the transfered circuit has been performed. However, the presented simulation results suggest that an implementation of a neuron circuit, based on the existing design and providing comparable performance, can be realized in the new process technology.



(a)

(b)

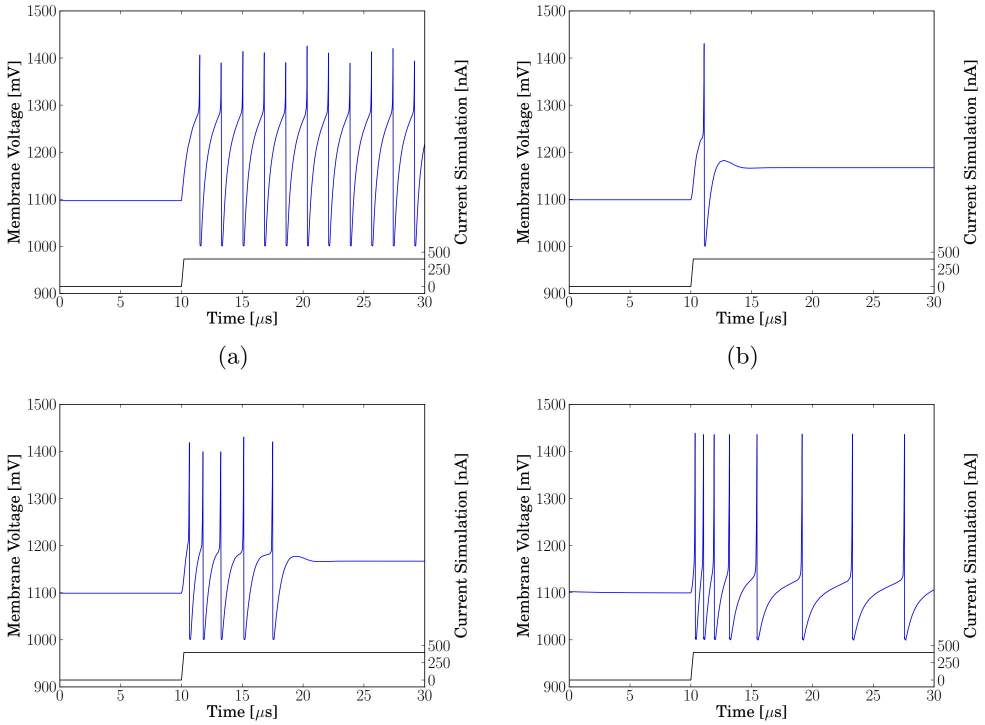
(c)

(d)

(e)

(f)

Figure 8.1: Characteristics of the operational transconductance amplifier used in the neuron circuit. The left panels simulation results for the original 180 nm design are shown, the right panels present the corresponding simulations for the 65 nm version. (a) and (b) show the output current as a function of the differential input voltage for different bias currents. In (c) and (d) the common mode of the input voltage is varied by ± 400 mV relative to half of the respective supply voltage. In (e) and (f) the voltage at the output of the circuit is varied, again by ± 400 mV relative to half of the respective supply voltage.



(a)

(b)

(c)

(d)

Figure 8.2: Membrane voltage of the neuron circuit (blue, left y-axis) resulting form a current stimulus (black, right y-axis). (a) shows tonic spiking, (b) phasic spiking, (c) phasic bursting and (d) spike frequency adaptation.

9 Discussion

The scope of this thesis is to evaluate if and to what extent large-scale neuromorphic hardware systems based on mixed-signal circuits can benefit from the progress made in semiconductor manufacturing. The reference for this evaluation is the current BrainScaleS Hardware System, presented in Chapter 2, which is realized in a 180 nm process technology. After an evaluation of several process technologies based on documentation and simulations a 65 nm low-power process offered by TSMC was considered a promising candidate for future neuromorphic systems. In a first prototype chip several basic process characteristics have been investigated. Further more complex circuits, aiming directly at integration in neuromorphic hardware, have been designed and tested in a second prototype chip. These include an analog parameter storage system and components for a synapse array which is primarily based on digital circuits. Additionally the option of transferring the neuron circuits used in the BrainScaleS Hardware System to the new process technology is explored. In the following, the main results of this thesis will be summarized and discussed.

9.1 Basic Circuits Realized in the 65 nm Process Technology

9.1.1 Static Power Consumption of Custom SRAM

One of the main criteria for the process selection was the comparatively low leakage of digital components, an important aspect for wafer-scale integration. This issue was also investigated by measurements, using an array of custom SRAM cells as a test vehicle. The measurement results are consistent with simulations. The results obtained for custom SRAM have been used for a rough estimation of the static power consumption of the digital part of a wafer-scale system, see Section 5.2.2. For wafers in slow and typical process corner the results are acceptable. However, the simulation result for a wafer in fast process corner, operating at 50 °C, results in a value of approximately 300 W. This number indicates that static power consumption of digital circuits should be considered during development.

9.1.2 Device Mismatch

Device mismatch is reported to be an increasing problem for analog circuits realized in recent process technologies, see e.g. Agarwal and Nassif [2007]. Therefore Monte Carlo simulations, using transistor models accounting for device variation, are an essential tool for the development of analog circuits. The accuracy of the provided models has been verified by comparison to measurement results obtained from dedicated test circuits included in the first prototype chip. A reasonable correspondence between measurement and simulation was observed. In regular layouts the variation measured is even lower than predicted by simulations, see Section 5.3.

9.1.3 Integrated Power Management

The new process technology is based on 30 cm wafers, an aspect which raises new challenges for wafer-scale integration. A larger and even more complex main PCB is required if the basic architecture of the BrainScaleS Hardware System is retained. Therefore the option of integrating power management structures into the wafers, rather than using a large number of discrete devices on the main PCB, was evaluated. A circuit allowing to switch the 1.2 V supply voltage using an low-threshold NMOS transistor has been tested. Based on the dimensions and power consumption of the HICANN chip, the estimated area consumption of on-chip power switches for the 1.2 V supply is in range of 8000 μm^2, which corresponds to less than 0.02 % of the overall chip area. A critical aspect within this approach is that any defect leading to a short in an integrated power switch renders a full wafer useless. In order to decrease the probability for such a defect, the layout avoids any structures using minimum spacing. However, the expected simplification of the main PCB might compensate for the loss of individual wafers.

9.2 The Capacitive Parameter Storage System

A parameter storage system for large scale neuromorphic hardware, based on capacitive storage cells, has been realized in the 65 nm process technology. It allows to provide large numbers of programmable current and voltage sources required to adjust and calibrate the properties of analog circuits, most of all the neuron implementations.

The system presented has a nominal resolution of 10 bit, however it is assumed that an accuracy in range of 8 bit is sufficient for operating neuromorphic hardware. However, the degree of e.g. non-linearity and transient distortions which can be tolerated by a neuromorphic hardware system has not been determined previously. Therefore it is difficult to conclusively assess the measurement results.

A novel scheme for programming and refreshing of the capacitive memory cells is introduced. Typical strategies are based on a single DAC, sequentially updating the individual cells. The scheme proposed here operates in a parallel fashion. All cells are supplied with a global reference signal, which changes slowly over time and covers the full dynamic range. Simultaneously a 10 bit counter is used to measure time, its value is distributed to all cells in the array. Each individual cell contains 10 bit of SRAM, storing the digital representation of its target value. Asynchronous logic within every cell compares the value stored in its local memory to the counter. Whenever a match is detected, the cell updates its internal analog value towards the present value of the reference signal.

This architecture is well suited for scaling. An arbitrary number of cells can be refreshed within one period of the reference signals. In the experiments described, the period of one cycle of the refresh system was typically set to 0.41 μs and the counter operated at a frequency of about 2.5 MHz. The slow operation of all components of the programming system simplifies the integration of the parameter memory into a chip. The system relies on the possibility to efficiently integrate memory and digital logic into every cell. Further sensitive analog as well as multiple digital signals need to be routed through the array. Sufficient shielding requires significant amounts of routing resources. Therefore the system benefits from the compact logic as well as the large number of metal layers available in the 65 nm process technology. The implementation of a comparable cells in a 180 nm process technology would result in significantly higher area consumption. Memory and comparison logic can be assumed to cover 10 times more area, the overall size of the cells would increase by approximately a factor of four.

The design of the individual cells utilizes the analog-t-switch strategy [Ishida et al. 2006] in order to achieve low drift rates for the voltage stored on the internal capacitors. The analog part of the cells is built from thick-oxide transistors to obtain the optimum storage time and to allow for a larger dynamic range of the voltage cells. A single parameter storage cell covers about 175 μm^2 of chip area. The current cells are designed to deliver currents of up to 2 μA, the output of the voltage cells covers a dynamic range from 100 mV to 1.8 V.

In the following the most important results are summarized and their consequences for a large-scale system are discussed.

9.2.1 Experimental Results

An implementation of the capacitive parameter storage system featuring 32 × 24 storage cells was integrated into a prototype chip and tested. A general limitation for the significance of the experimental results arises from a conceptual error in the readout multiplexers of the voltage cells, see Section 6.4.1. As a result, only three voltage cells per chip can be accessed, these are equipped with individual output amplifiers connected directly to separate bond pads. In total, only 8 voltage cells on three chips have been tested. Nevertheless, the basic characteristics of the voltage cells could be determined. For the current cells there is no such limitation, all 384 cells per chip can be tested.

Several problems, on the cell level as well as in the programming system, have been identified in this implementation. Possible solutions for these issues are presented in Section 6.5. Despite the limitations introduced by errors in the implementation of the system, it was possible to characterize most aspects of the analog performance of the system.

Dynamic Range and Linearity

Overall, the results regarding the dynamic range of the voltage and current cells are within the expected range. The voltage cells are able to reliably produce output voltages in the range of 100 mV to 1.8 V. The lower limit is due to the short storage time for lower voltages, the upper limit is given by the limited conductance of the NMOS transistors used as switches.

Depending on the settings of the programming system, the current cells can produce output currents of up to 2.4 μA. However, the performance of the current cells at the lower end of the dynamic range observed in the prototype chip is not satisfactory. The option of providing adjustable currents in the range of few nano amperes is crucial for the energy-efficient realization of neuron circuits. Measures to circumvent the problem, which is caused by the input offset voltage of operational amplifiers involved in the generation of the reference voltages, are presented in Section 6.5.3.

For voltage and current cells a systematic non-linearity is observed, see Section 6.4.2 and Section 6.4.3. A thorough evaluation of the effect is only possible for the current cells. The deviations observed are in a range of ± 10 LSB. While all current cells located on the same chip show the same pattern of non-linearity, the effect varies significantly from chip to chip. It probably originates from device mismatch in the circuits generating the reference voltages for the system.

Cell-to-Cell Variation and Reproducibility

From Monte Carlo simulations a very low cell-to-cell variation is expected for the voltage cells. However, this could not be verified due to the limited number of accessible cells. Further each cell uses an individual output amplifier introducing additional variation, significantly larger than the expected variation between the cells. For the current cells a significant cell-to-cell variation is observed, it is mostly caused by device mismatch of the cells' output transistor. Over the full dynamic range the maximum standard deviation of the output currents between all cells tested is in a range of 4 LSB. The measurement results are within the range expected from Monte Carlo simulations.

Due to device mismatch, the neuron circuits of the BrainScaleS Hardware System need individual calibration for each parameters. The work described in Schwartz [2013] is dedicated to this task. It can be expected that future neuron implementations also require calibration. The cell-to-cell variation introduced by the parameter storage system will be eliminated by the process.

An essential requirement for successful calibration however is a precise reproducibility of output currents and voltages for the individual cells. Repeated, independent programming to the same digital code leads to a variation of less than 1 LSB on the average output currents and voltages of the cells, see Section 6.4.5.

Parameter Drift

The drift of the output voltages and currents over time determines the minimum refresh rate required. It is assumed that the output of the cells is allowed to drift by up to 1 LSB between two refresh cycles.

At room temperature, the voltage cells show large drift rates of up to 0.1 LSB/ms for output voltages below 100 mV. This behavior is expected from simulations and leads to the restriction that the voltage cells should not be used in a range below 100 mV. For larger voltages the observed drift rates are below 0.02 LSB, see Section 6.4.6. The drift of the voltage stored on the capacitors of the current cells is, according to simulations, lower than for the voltage cells, see Figure 6.8. Due to the characteristics of the output transistor the drift rate observed for the output currents are systematically larger. It increases with larger output currents. The average drift rates observed at high output currents are in range of 0.04 LSB/ms, see Section 6.4.6.

However, the minimum refresh rate required for the operation of the system is limited by the highest drift rates which can be expected under realistic operating conditions. The drift rates show a strong temperature dependence as well as significant cell-to-cell variation. The largest drift rate measured, for a cell at a temperature of 50 °C and an output current of more than 1.6 μA, was 0.32 LSB/ms, see Section 6.4.7. As a result, a minimum refresh rate of 0.32 kHz is required to operate the system at this temperature.

Transient Distortions

For current and voltage cells transient distortions have been observed at the outputs during the refresh cycles, see Section 6.4.9. In both cases crosstalk from the respective programming voltage to the output of the cells occurs. While the crosstalk is considered not critical for the voltage cells, the effect causes deviations in order of ± 1.5 LSB for the current cells. The issue needs further investigation as the exact source of the problem is not yet identified. The suggested improvements for the reset process, see Section 6.5.4, will help to reduce the effect. Improving the shielding in the layout might be sufficient to reduce the effect to an acceptable level. One of the most critical limitations regarding analog quality is the distortion which can be observed for voltage cells during the actual refresh process of the individual cell, see Figure 6.44. The observed amplitude of about 3 mV, corresponding to 1.7 LSB, indicates that the compensation of the charge injection is not as efficient as expected from simulations. Using a refined simulation setup which accounts also for parasitic effects, it is presumably possible to readjust the dimensions of the compensation transistors. The fact that no distortion from the refresh process of the individual cells is visible for the current cells demonstrates that it is possible to reduce the effect.

Operating the Entire Array

The implementation of the digital part of the cells in the prototype chip does not generate the correct timing for the signals controlling the refresh process, see Section 6.3.4. No significant consequences for the operation of single parameter storage cells have been observed. However, this effect causes a severe crosstalk between different cells, if these are programmed to similar target values. The situation of multiple cells being programmed to identical or similar values is a typical scenario for a system providing parameters to neuron circuits. This scenario could not be tested using the prototype chip. A simplified version of the digital logic in the cells which generates the correct timing for the refresh process is presented in Section 6.5.2.

Power Consumption

Based on measurements performed with the prototype chips, the power consumption of an active system providing parameters to 512 neuron circuits can be estimated, see Section 6.4.10. In a worst case scenario this estimation results in a value of about 5 mW. This value is based on an refresh rate of 1.2 kHz and assumes constant biasing of all amplifiers. However, measurements of the cell's drift rate indicate that a refresh rate of about 0.3 kHz is sufficient. Combined with some minor improvements to the design of the system, which are described in Section 6.4.10, a power consumption in range of 1 mW seems achievable. For a wafer-scale system providing the same number of neurons as the BrainScaleS Hardware System, the parameter memory required would consume about 0.5 W, which is negligible compared to the total power consumption of such a system.

9.2.2 Setup Time

An important aspect for a neuromorphic hardware system is the setup time required to initialize the system before an actual experiment can be initiated. Due to the internal structure of the parameter storage cells, the output of a cell cannot directly reach a new target value within a single refresh process. Instead, multiple refresh cycles are required, see Section 6.3.5 and Section 6.4.8, until the new value is reached. The resulting setup time of the parameter storage system is estimated to be in range of 15 ms when using a refresh rate of 1.2 kHz.

This value needs to be considered in relation to the overall setup time of a wafer-scale system. In the BrainScaleS system a full set of configuration data consists of 44 MB, 87 % of which are the individual weights and addresses of the synapse circuits [Brüderle et al. 2011]. Even if only limited by the bandwidth between host PC and a wafer module, the transmission of the configuration data for an experiment takes more than 40 ms. If the neuron configuration is transmitted first, the analog outputs can settle at the correct values during the transmission of the remaining configuration data. In this case the the overall setup time is not affected by the setup time of the parameter storage system.

If the initial configuration of future systems is significantly faster or in a series of experiments in which only the neuron configuration varies, the setup time of the parameter storage system has an impact on the maximum rate at which independent experiments can be performed. An option to improve the setup time is to introduce dynamic scaling of the refresh rate. At initialization or when parameters are changed, the refresh rate can be increased. Once the analog outputs have reached their target values, the refresh rate can be reduced in order to optimize the power consumption of the system.

9.2.3 Comparison with the Existing Floating Gate-Based System

Due to its low drift rate, floating gate-based memory is an elegant solution to realize energy-efficient and compact analog parameter storage devices. However, the physical implementation into standard mixed-signal ASICs is difficult. In the BrainScaleS Hardware System three additional supply voltages are required to operate the parameter system, two of which are outside of the nominal supply voltage range of the process technology.

On the contrary, a system based on capacitive memory integrates well into mixed-signal ASICs. Due to the required refresh cycles, it will always consume more power than a floating gate-based system during operation. However, the estimated power consumption of the system presented here is considered to be acceptable, compared to the expected power consumption of the overall system.

For the presented system, the area consumption per parameter is 175 μm^2, this is about 20 % less than the area covered by a single floating gate parameter cell in the BrainScaleS Hardware System. The accuracy of the cells, especially regarding reproducibility, is significantly better than the results obtained from the floating gate system, see Kononov [2011] and Millner [2012, Chapter 9]. In the existing BrainScaleS Hardware System, the setup time of the parameter memory depends on the required accuracy. For typical experiments the programming process takes several seconds [Hartel 2014]. Compared to this value, the capacitive memory system, which features a setup time in range of 20 ms, provides a significant advantage. Another difference to the existing system is that the capacitive parameter storage cells allows to change parameters during an experiment.

The most important benefit from replacing the floating gate-based storage cells by the capacitive parameter storage system is the simplification of the overall wafer-scale system.

9.2.4 Summary

The system presented is an area- and energy-efficient option to realize analog parameter memory for large-scale neuromorphic hardware applications. It offers a nominal resolution of 10 bit, but due to several limitations it does not reach 10 bit accuracy. However, the high reproducibility of the cells allows for calibration in scenarios where the actual precision is not sufficient. For a final assessment of the proposed parameter storage system, the problems identified in current implementation need to be corrected. An implementation of a system with the full dimensions has to be tested in silicon for final verification.

For future implementations several additional option regarding the design of the system should be considered. In the prototype chip the voltage cells are operated without an integrated buffer in order to save area and power. However, the high output resistance of the cells, approximately 15 GΩ at a refresh rate of 1.2 kHz, renders the wires connecting the signals to the neuron circuits extremely sensitive to crosstalk, see Figure 6.44. Additional experiments are required to evaluate whether it is possible to shield these wires sufficiently in an actual neuromorphic system to allow for reliable operation. Further the option of introducing specialized cells is discussed in Section 6.6.2. Based on the designs of the existing cells it is possible to introduce additional cells, providing different dynamic ranges. These cells can help to simplify the design of the neuron circuits and increase the overall power efficiency. The question whether it is worth the effort depends on the design of the future neuron circuits which is not yet available.

9.3 SynArray

Based on the synapse array of the existing system, a new architecture for a future synapse array was developed in collaboration with S. Friedmann, A. Hartel and J. Schemmel. It takes advantage of the comparatively faster and more compact digital circuits available in the 65 nm process technology. The aim is to improve the performance of the synapse circuits compared to the synapse circuits implemented in the BrainScaleS Hardware System. Due to the compact SRAM cells it is possible to increase the weight resolution of the synapses to 8 bit. However, it is not feasible to implement an 8 bit DAC for the generation of postsynaptic events within each synapse. Instead, a scheme in which multiple synapses share one DAC is introduced. Digital control logic which reads the weight of an activated synapse and transfers the data to the assigned DAC needs to be integrated into the array.

Further, the option of implementing STDP functionality using purely digital circuits is considered. Compared to the existing system, this would increase the flexibility regarding the update rules. Such an approach requires the STDP control logic to have fast access to the synapse weights. The weights of either a full row or a full column of synapses need to be processed in case of a pre- or a postsynaptic event, respectively. The memory holding the weights needs to provide appropriate interfaces.

Size and performance of the actual STDP controllers is crucial for the efficiency of the entire concept. However, the evaluation of this aspect is not within the scope of this thesis. In Friedmann [2013, Chapter 6.2] considerations regarding the general architecture of digital STDP controllers are presented. The proposed concept is in an early state of development and the validity of the assumptions made remains to be determined. In order to gain some reliable figures regarding area consumption and performance, two of the components required have been implemented and tested in a prototype chip. These are a memory block designed to hold the digital representation of the weights and addresses of the synapses as well as an 8 bit current DAC for the generation of postsynaptic pulses.

9.3.1 SynRAM

An array of dual port SRAM cells has been developed as a memory for storing synaptic weights. The memory block provides two access ports, orientated orthogonally to each other. Its internal structure is organized such that each 8 bit word stored in the memory can be accessed within a single operation from either of the two ports. This feature is crucial to allow for efficient interfacing with digital STDP controllers.

In order to speed up the reading operations, standard voltage-mode sense amplifiers are used. The timing of the access operations is generated by circuits using replica bitlines as delay elements. In simulations an array size of 1024 × 64 bit, corresponding to 128 × 32 synapses, was used. Access operations using the row port are possible within 2 ns, for the column port operation can be performed reliably within 4 ns.

A smaller memory block, holding 256 × 32 bit, was included in a prototype chip. However, it was not possible to determine the maximum speed of operation. When increasing the clock frequency, the digital control logic of the prototype chip fails before the memory itself reaches its limits. For the row port reliable single-cycle accesses have been demonstrated for frequencies up to 740 MHz, see Figure 7.13. For the column port, operating relative to a clock running at half the frequency of the main clock, successful operation has been demonstrated for up to 400 MHz, see Figure 7.13. The sense amplifiers and the replica bitlines contribute significantly to the area consumption of the SynRAM. For an array holding 1024 × 64 bit, about 40 % of the total area are covered by these circuits. The memory contributes 42 μm^2 to the effective area consumption of a single synapse.

9.3.2 SynDAC

The second component developed for the new synapse architecture is an 8 bit DAC, designed to produce short current pulses. Its architecture is based on the unit element approach. The two bits of least significance however are realized by serial connection of transistors, instead of using only parallel connections, in order to reduce the total number of current sources required. Multiple instances of the DAC were implemented into a prototype chip and have been tested.

When operated in a constant current mode, the impact of device mismatch in the DAC can be evaluated precisely. Mismatch leads to a significant differential non-linearity, resulting in non-monotonic characteristics for many DACs. The effect is most prominent at the transition between the digital codes 127 and 128. Averaged over 128 instances, the change observed in the output current is 1.2 LSB, the standard deviation is 4.1 LSB. During operation of neuromorphic hardware the weights of the synapses are typically modified by learning algorithms such as STDP. As a result of a non-monotonic relation between the programmed weight and the amplitude of the postsynaptic pulses, it is possible that some of these algorithms converge towards a local rather than the global minimum. However, the large stochastic variation in the activity of neural networks might prevent such issues. This aspect needs further investigation on a theoretical level.

When used in a synapse array the DAC needs to produce short current pulses. The synaptic input circuit of the connected neuron integrates the amount of charge transfered during such a pulse. The DAC has also been tested in a mode of operation where it periodically produces current pulses with a length of 2.5 ns. Its output signal is integrated using an on-chip capacitor. Evaluation of the voltage on the capacitor allows for relative measurements of the amount of charge transfered per pulse. As a result of capacitive crosstalk between the output of the DAC and the reference voltage which controls the transistors operating as current sources, a systematic non-linearity is observed, see Section 7.7.2. However, this is not considered a problem for the application in the synapse array. Due to its systematic nature the effect can be compensated for by calibration in case linearity is required.

In the HICANN chip a comparable DAC circuit, offering a resolution of 4 bit, is used to produce postsynaptic pulses. The offset observed in the output characteristics of this circuit has been successfully suppressed in the new DAC design. This is achieved by precharging the DAC's output capacitance before triggering the actual pulse.

The overall area consumption of one DAC circuit is 182 μm^2. Assuming one DAC being shared among 32 synapses, it contributes 6 μm^2 to the effective area of a single synapse.

9.3.3 Summary

A conclusive assessment regarding the new architecture for the synapse array is difficult to make. Important components, such as the digital STDP controllers and the *synapse_driver* circuit, have not been developed yet. Further the optimum number of synapse rows per subunit, an important variable in the concept, have not been determined so far.

The average firing rate of a neuron can be estimated to be in order of 10 Hz [Wilson et al. 1994]. Based on the assumption that one subunit contains 32 rows of synapses and accounting for the speed up factor of 10^3, the average rate of presynaptic events which needs to be processed within each subunit is 320 kHz. All components of the synapse array are expected to operate at a clock frequency of 500 MHz. On average, the processing of a single presynaptic event needs to be completed within about 1500 clock cycles. However, in biology the phenomenon of *bursting* is observed, where single neurons emit sequences of multiple action potentials at a high rate. As an example, firing rates of up to 400 Hz have been reported for neurons located in the human visual cortex during bursts [Wandell et al.

2007]. But even in a scenario where half of the presynaptic neurons connected to one subunit are firing at such rates, the average interval between the spikes is in a range of 80 clock cycles. The generation of postsynaptic events can be performed easily within this frame. Reading the data associated with a synapse row from the *SynRAM*, evaluating the address, transferring the weight information to the DAC and generation of the actual postsynaptic current pulse is possible within less than 10 clock cycles. To cope with collision between events, a FIFO[1] queue-based event buffer can be implemented in the *synapse_driver*. However, the maximum possible event rate will probably be limited by the STDP controllers, not by the generation of postsynaptic events.

The overall area required for the implementation of one synapse in the new system cannot yet be determined. However, it is possible to compare the area covered by memory and DAC to the corresponding numbers for the existing system, not accounting for circuits related to STDP in both systems. In the new system, DAC and memory cover about $50\,\mu\text{m}^2$. Despite the fact that the weight resolution is increased by a factor of 16, these components cover only half the area consumed by the corresponding components in the 180 nm system.

In the example presented, the area consumption of the memory storing weight and address is dominating over the area consumption for the DAC. One option to improve the area efficiency is to include a larger number of synapse rows into one subunit, since the contribution of the sense amplifiers and the control logic to the total area of a subunit is constant. However, assuming a number of 32 synapse rows per subunit, the length of the bitlines is only $72\,\mu\text{m}$. Therefore the option of omitting the sense amplifiers for the `row` port can be evaluated. This would slow down the access operation, but the area efficiency of the memory increases by more than 30 %.

Overall, the performance, area and power consumption of the STDP logic are expected to be the most critical aspect of the new concept.

9.4 Neuron Implementations in the New Process Technology

The accurate emulation of neurons typically requires the most complex analog circuits within mixed-signal neuromorphic hardware systems. In the BrainScaleS Hardware System a circuit implementing the Adaptive Exponential Integrate-and-Fire model is used [Brette and Gerstner 2005, Millner 2012]. Within this thesis, the option of transferring the existing circuit directly to the 65 nm process technology using thick-oxide transistors has been evaluated. This option has been demonstrated for an individual operational amplifier, see Chapter 5.5. Using an identical schematic, only a limited number of adjustments to the dimensions of the transistors was required to achieve a performance comparable to the original design.

[1] First-In-First-Out

Further the transfer of the full neuron circuit used in the BrainScaleS Hardware System has been evaluated using simulations. Again, a circuit based on a schematic which is identical to the original design has been used. By adjusting transistor dimensions it was possible to achieve qualitatively comparable behavior. The adaption of the neuron was performed as a proof of concept. The result indicates that it is possible to reuse concepts and circuits from the existing system. In order to obtain a neuron circuit which can be used in a future system, some known limitations of the current design need to be corrected.

Even when developing a new neuron design from scratch, using thick-oxide transistors at least for parts of the circuit is the most promising approach. Such a strategy for the implementation of analog circuits in modern process technologies is also suggested e.g. in Annema et al. [2005], Mak and Martins [2010] and in Nauta and Annema [2005]. The higher supply voltage simplifies the design of the circuits and the signal-to-noise ratio benefits from larger dynamic ranges.

9.5 Conclusion

The networks which can be investigated using available neuromorphic hardware systems are orders of magnitude smaller than the networks found in the brains of mammals. This raises the question whether it is possible to scale the concept of the BrainScaleS Hardware System to such dimensions. The work presented focuses on the question to what extent a transition to a 65 nm process technology can contribute to this effort. Increasing the performance of digital circuits is the motivation for the ongoing scaling of the CMOS technology. Therefore the digital components in a mixed-signal system undoubtly benefit from any transition towards a more recent process technology. In particular, for the transition from a 180 nm to a 65 nm process technology it can be assumed that the area consumption of digital circuits decreases by a factor of eight.

In contrast, the properties of modern process technologies introduce various challenges for the design of analog circuits. Especially the reduced supply voltage renders the usage of many standard analog design strategies impossible. Specialized low-voltage solutions are available for many standard applications. However, these typically require a significantly larger number of components, which also leads to an overall higher power consumption [Annema 1999]. Nevertheless, analog circuits also benefit from some aspects of modern process technologies. Examples are the extended routing resources or the high conductance of minimum length transistors, allowing for compact analog multiplexer implementations. The availability of area-efficient memory permits the implementation of additional configuration and calibration features.

Overall, when selecting a modern process technology for a mixed-signal system, one has to decide whether the benefits for digital circuits are overcompensating for the additional difficulties anticipated for complex analog circuits. These considerations are linked to the question which fraction of the functional units in the system are realized based on analog or digital circuits. Further, some components can be realized using either digital or analog circuits. The chosen process technology has a significant impact on these decisions.

For the 65 nm process technology, using thick-oxide transistors to realize analog circuits is beneficial. As demonstrated for the neuron circuit, these devices also enable the transfer of existing components. However, the area consumption of analog circuits will not decrease significantly.

In conclusion, it is possible to build a system, comparable to the existing BrainScaleS Hardware System, using the 65 nm process technology. Assuming comparable features of the analog components, their area consumption remains constant. The digital components, currently covering approximately half of the total area, will require only about one tenth of their original size. This offers the possibility to either increase the quality of the circuits emulating neurons and synapses or to increase their number. With the latter option, accounting also for the increased diameter of the wafers, a maximum of four times as many neurons and synapses can be implemented per wafer.

Still, a large number of wafers needs to be interconnected in order to build a system large enough to achieve biologically relevant dimensions. For realization of such a system, the individual wafer modules should be designed as simple and compact as possible. As one example, the aspect of simplifying the power management for a wafer has been addressed within this thesis. Replacing the floating gate-based analog parameter storage system also contributes to a simplification of the overall system.

In the current system, the event routing and communication resources required for inter-wafer communication are realized using dedicated submodules based on FPGAs. The next step towards a system which allows for a higher integration density is to integrate the functionality of these modules directly into the wafers. This project remains to be planned in detail. However, it can only be realized if highly power- and area-efficient memory and logic is available.

The 65 nm process technology facilitates the implementation of a large-scale system by two means. On the one hand, the number of neurons and synapses per wafer can be moderately increased. On the other hand, it may allow for a significant simplification of the inter-wafer communication. The latter might be the more important contribution towards a large-scale system, providing neuron and synapse numbers in biologically relevant dimensions.

10 Outlook

As discussed within this thesis, several advantages are expected from the transition from a 180 nm process technology towards a 65 nm process technology for a large scale neuromorphic hardware system based on mixed-signal chips. The 65 nm process technology used in this work was introduced in 2007. CMOS scaling still continues. As of today (2014), chips produced in 22 nm process technologies are available in mass production [ITRS 2011]. This raises the question what can be expected if the latest or even future process technologies are used to implement neuromorphic hardware.

The basic considerations mentioned before remain true. Digital circuits benefit drastically from CMOS scaling, while the implementation of complex analog circuits becomes more challenging. However, also in the latest process technologies the option of using thick-oxide transistors is typically available, see e.g. Narasimha et al. [2012], an aspect which supports the development of analog circuits in these technologies. Nevertheless, at some point digital implementation will be the most efficient solution for all components of a neuromorphic system. At which technology node this point is reached remains to be determined.

One example of neuromorphic hardware based on purely digital circuits is presented in Seo et al. [2011] and Merolla et al. [2011]. The chips described are realized in a 45 nm process technology and designed to operate in biologically realistic time scales. One chip with a size of 4.2 mm^2 emulates 256 neurons which can be interconnected by 262 k binary synapses. Every neuron circuit, implementing a leaky integrate-and-fire model, covers 3300 μm^2 of chip area.

In direct comparison, the area covered by each neuron circuit in the BrainScaleS Hardware System, including the associated parameter storage cells, is about twice as large. Another parameter which can be considered a measure for the integration density is the ratio of the number of neurons implemented to the total chip area. Based on the dimensions of a single HICANN chip, this ratio is about five times lower for the BrainScaleS Hardware System.

However, within this system analog circuits implement the significantly more sophisticated Adaptive-Exponential Integrate-and-Fire neuron model. As demonstrated by quantitative analysis [Pospischil et al. 2011] the AdEx model represents the behavior of biological cells more adequately than the simpler LIF model. Further the HICANN chip provides a resolution of 4 bit for the synaptic weights rather than only binary synapses. Finally, the BrainScaleS Hardware System operates 10^4 times faster than biological realtime.

For experiments which require a high number of rather simple components operating in realtime, digital circuits are already an interesting option. However, if complex models are desired and accelerated operation is beneficial, analog circuits currently provide superior performance over existing digital approaches.

Besides scaling of CMOS technology, further developments may have an impact on the future of neuromorphic hardware systems.

A novel device frequently discussed in the context of neuromorphic hardware development is the memristor. The concept was introduced by L. Chua in 1978 [Chua and Kang 1976]. However, only in 2008 it could be realized using semiconductor technology [Strukov et al. 2008]. Provided that memristors can be integrated into neuromorphic systems, these devices are an interesting option to realize programmable analog storage elements. Snider [2008] demonstrates the option of adding an array of memristive devices on top of a CMOS chip. In addition, this study evaluated the option of using these devices for implementation of synapses in a neuromorphic system.

In the course of CMOS technology development, wafer size has continually increased. Currently wafers of 45 cm in diameter are used. In systems using wafer-scale integration, such as the BrainScaleS Hardware System, more components can interact via power-efficient on-wafer communication links. However, increasing wafer size imposes additional challenges for the mechanical realization of such a system. This aspect needs further investigation, presumably the introduction of novel approaches is required. A particularly interesting development in the chip packaging industry is 3D integration [Lau 2010]. Multiple dies are thinned, stacked and can be interconnected by through silicon vias (TSV). Even the combined packaging of multiple thinned wafers, an approach termed 3D-Wafer-Level-packaging, is under development [Pieters and Beyne 2006, Ramm et al. 2010]. These technologies may contribute significantly to the realization of highly integrated large-scale neuromorphic hardware systems.

Using modern technology it may by possible to built a neuromorphic hardware system, encompassing the dimension of the human brain.

Abbreviations

AdEx Adaptive Exponential Integrate-and-Fire.

ASIC Application Specific Integrated Circuit.

CMOS Complementary Metal Oxide Semiconductor.

CPU Central Processing Unit.

DAC Digital to Analog Converter.

DNL Differential Non-Linearity.

DRAM Dynamic Random Access Memory.

ESD Electro Static Discharge.

FPGA Field Programmable Gate Array.

HDL Hardware Description Language.

LIF Leaky Integrate-and-Fire.

LSB Least Significant Bit.

MIM Cap Metal Insulator Metal Capacitor.

MPW Multi-Project Wafer.

NDA Non Disclosure Agreement.

PCB Printed Circuit Board.

PDK Process Design Kit.

RTL Register Transfer Level.

SRAM Static Random Access Memory.

STDP Spike-Timing Dependent Plasticity.

STP Short-Term Plasticity.

VLSI Very Large Scale Integration.

Bibliography

L. F. Abbott and S. B. Nelson. Synaptic plasticity: taming the beast. *Nature neuroscience*, 3:1178–1183, 2000.

K. Agarwal and S. Nassif. Characterizing Process Variation in Nanometer CMOS. In *Design Automation Conference, 2007. DAC '07. 44th ACM/IEEE*, pages 396–399, June 2007.

P. Allen and D. Holberg. *CMOS Analog Circuit Design*. Oxford University Press, 3rd edition, 2011. ISBN 9780199765072.

B. Amrutur and M. Horowitz. A replica technique for wordline and sense control in low-power sram's. *Solid-State Circuits, IEEE Journal of*, 33(8):1208–1219, Aug 1998. ISSN 0018-9200. doi: 10.1109/4.705359.

M. Anis. Subthreshold leakage current: challenges and solutions. In *Microelectronics, 2003. ICM 2003. Proceedings of the 15th International Conference on*, pages 77–80, Dec 2003. doi: 10.1109/ICM.2003.1287726.

A.-J. Annema. Analog circuit performance and process scaling. *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, 46(6): 711–725, Jun 1999. ISSN 1057-7130. doi: 10.1109/82.769780.

A.-J. Annema, B. Nauta, R. van Langevelde, and H. Tuinhout. Analog circuits in ultra-deep-submicron CMOS. *Solid-State Circuits, IEEE Journal of*, 40(1): 132–143, Jan 2005. ISSN 0018-9200. doi: 10.1109/JSSC.2004.837247.

P. Athe and S. Dasgupta. A comparative study of 6t, 8t and 9t decanano sram cell. In *Industrial Electronics Applications, 2009. ISIEA 2009. IEEE Symposium on*, volume 2, pages 889–894, Oct 2009. doi: 10.1109/ISIEA.2009.5356318.

J. Bhasker and R. Chadha. *Static Timing Analysis for Nanometer Designs: A Practical Approach*. Springer Publishing Company, Incorporated, 1st edition, 2009. ISBN 0387938192, 9780387938196.

G. Q. Bi and M. M. Poo. Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 18 (24):10464–10472, Dec. 1998. ISSN 0270-6474. URL `http://www.jneurosci.org/content/18/24/10464.abstract`.

K. Bowman, S. Duvall, and J. Meindl. Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration. *Solid-State Circuits, IEEE Journal of*, 37(2):183–190, Feb 2002. ISSN 0018-9200. doi: 10.1109/4.982424.

Brain-i Nets. Brain-i-Nets: Novel Brain-Inspired Learning Paradigms for Large-Scale Neuronal Networks. `http://www.brain-i-nets.kip.uni-heidelberg.de`, 2012.

R. Brette and W. Gerstner. Adaptive Exponential Integrate-and-Fire Model as an Effective Description of Neuronal Activity. *J. Neurophysiol.*, 94:3637 – 3642, 2005.

D. Brüderle, M. A. Petrovici, B. Vogginger, M. Ehrlich, T. Pfeil, S. Millner, A. Grübl, K. Wendt, E. Müller, M.-O. Schwartz, and et al. A comprehensive workflow for general-purpose neural modeling with highly configurable neuromorphic hardware systems. *Biological Cybernetics*, 104:263–296, 2011.

L. Callewaert and W. Sansen. Class AB CMOS Operational Amplifiers with Very High Efficiency. In *Solid-State Circuits Conference, 1989. ESSCIRC '89. Proceedings of the 15th European*, pages 129–132, Sept 1989. doi: 10.1109/ESSCIRC.1989.5468139.

J. Caves, S. Rosenbaum, M. Copeland, and C. Rahim. Sampled analog filtering using switched capacitors as resistor equivalents. *Solid-State Circuits, IEEE Journal of*, 12(6):592–599, Dec 1977. ISSN 0018-9200. doi: 10.1109/JSSC.1977.1050966.

A. P. Chandrakasan, W. J. Bowhill, and F. Fox. *Design of High-Performance Microprocessor Circuits*. Wiley-IEEE Press, 1st edition, 2000. ISBN 078036001X.

H. Chen, S. Saïghi, L. Buhry, and S. Renaud. Real-time simulation of biologically realistic stochastic neurons in VLSI. *IEEE Transactions on Neural Networks*, 21(9):1511–1517, 2010. URL `http://dblp.uni-trier.de/db/journals/tnn/tnn21.html#ChenSBR10`.

L. O. Chua and S. M. Kang. Memristive devices and systems. *Proceedings of the IEEE*, 64(2):209–223, 1976.

C. Duffy and P. Hasler. Modeling Hot-Electron Injection in pFET's. *Journal of Computational Electronics*, 2(2-4):317–322, December 2003. ISSN 1569-8025.

C. Eichenberger and W. Guggenbuhl. On charge injection in analog mos switches and dummy switch compensation techniques. *Circuits and Systems, IEEE Transactions on*, 37(2):256–264, Feb 1990. ISSN 0098-4094. doi: 10.1109/31.45719.

S. Friedmann. *A new approach to learning in neuromorphic hardware*. PhD thesis, Ruprecht-Karls Universität Heidelberg, July 2013. URL `http://www.ub.uni-heidelberg.de/archiv/15359`.

S. Furber, D. Lester, L. Plana, J. Garside, E. Painkras, S. Temple, and A. Brown. Overview of the SpiNNaker System Architecture. *Computers, IEEE Transactions on*, 62(12):2454–2467, Dec 2013. ISSN 0018-9340. doi: 10.1109/TC.2012.142.

C. Galup-Montoro, M. Schneider, and I. Loss. Series-parallel association of FET's for high gain and high frequency applications. *Solid-State Circuits, IEEE Journal of*, 29(9):1094–1101, Sep 1994. ISSN 0018-9200. doi: 10.1109/4.309905.

W. Gerstner, R. Kempter, J. Van Hemmen, H. Wagner, et al. A neuronal learning rule for sub-millisecond temporal coding. *Nature*, 383(6595):76–78, 1996.

S. Gorgin and A. Kaivani. Reversible barrel shifters. In *Computer Systems and Applications, 2007. AICCSA '07. IEEE/ACS International Conference on*, pages 479–483, May 2007. doi: 10.1109/AICCSA.2007.370925.

C. Graf. Transistor Mismatches bei einem Strom-DAC in 65nm-Technologie. Internship report (German), Heidelberg University, 2011.

P. Gray and R. Meyer. MOS operational amplifier design-a tutorial overview. *Solid-State Circuits, IEEE Journal of*, 17(6):969–982, Dec 1982. ISSN 0018-9200. doi: 10.1109/JSSC.1982.1051851.

A. Grübl. *VLSI Implementation of a Spiking Neural Network.* PhD thesis, Ruprecht-Karls-University, Heidelberg, 2007. URL `http://www.kip.uni-heidelberg.de/Veroeffentlichungen/details.php?id=1788`. Document No. HD-KIP 07-10.

M. Güttler. Konzeptoptimierung und Entwicklung einer hochintegrierten Leiterplatte. Diploma thesis (German), University of Heidelberg, HD-KIP-10-68, 2010.

A. Hartel. personal communication, 2014.

S. Hartmann, S. Schiefer, S. Scholze, J. Partzsch, C. Mayr, S. Henker, and R. Schiiffny. Highly integrated packet-based aer communication infrastructure with 3gevent/s throughput. In *Electronics, Circuits, and Systems (ICECS), 2010 17th IEEE International Conference on*, pages 950–953, Dec 2010. doi: 10.1109/ICECS.2010.5724670.

HBP SP9 partners. *Neuromorphic Platform Specification.* Human Brain Project, Mar. 2014.

M. Hock. Test of Components for a Wafer-Scale Neuromorphic Hardware System. Diploma thesis, University of Heidelberg, HD-KIP-09-37, `http://www.kip.uni-heidelberg.de/Veroeffentlichungen/details.php?id=1935`, 2009.

M. Hock, A. Hartel, J. Schemmel, and K. Meier. An analog dynamic memory array for neuromorphic hardware. In *Circuit Theory and Design (ECCTD), 2013 European Conference on*, pages 1–4, 2013.

A. L. Hodgkin and A. F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *J Physiol*, 117(4):500–544, August 1952. ISSN 0022-3751. URL `http://view.ncbi.nlm.nih.gov/pubmed/12991237`.

C. Hu, W. Liu, and X. Jin. *The BSIM3v3.2 MOSFET Model*, Dec 1998.

S. Hüll. Testen eines Floating-Gate Analogspeichers in 65 nm Single-Poly Technologie. Bachelor thesis (German), University of Heidelberg, 2014.

D. Husmann. personal communication, 2013.

G. Indiveri, B. Linares-Barranco, T. Hamilton, A. van Schaik, R. Etienne-Cummings, T. Delbruck, S.-C. Liu, P. Dudek, P. Häfliger, S. Renaud, J. Schemmel, G. Cauwenberghs, J. Arthur, K. Hynna, F. Folowosele, S. Saighi, T. Serrano-Gotarredona, J. Wijekoon, Y. Wang, and K. Boahen. Neuromorphic silicon neuron circuits. *Frontiers in Neuroscience*, 5:1–23, 2011. ISSN 1662-453X. doi: 10.3389/fnins.2011.00073. URL `http://www.frontiersin.org/Neuromorphic_Engineering/10.3389/fnins.2011.00073/abstract`.

K. Ishida, K. Kanda, A. Tamtrakarn, H. Kawaguchi, and T. Sakurai. Managing subthreshold leakage in charge-based analog circuits with low-VTH transistors by analog T- switch (AT-switch) and super cut-off CMOS (SCCMOS). *Solid-State Circuits, IEEE Journal of*, 41(4):859–867, April 2006. ISSN 0018-9200. doi: 10.1109/JSSC.2006.870761.

cpp. *Programming Language C++*. ISO/IEC 14882, July 1998.

ITRS. International Technology Roadmap for Semiconductors. `www.itrs.net/Links/2011itrs/2011Chapters/2011PIDS.pdf`, 2011.

E. M. Izhikevich. Which Model to Use for Cortical Spiking Neurons? *IEEE Transactions on Neural Networks*, 15:1063–1070, 2004. URL `http://www.izhikevich.org/publications/whichmod.htm`.

JTAG. IEEE Standard Test Access Port and Boundary-Scan Architecture. *IEEE Std 1149.1-2001*, pages i–200, 2001. doi: 10.1109/IEEESTD.2001.92950.

P. Kinget. Designing analog and RF circuits for ultra-low supply voltages. In *Solid State Device Research Conference, 2007. ESSDERC 2007. 37th European*, pages 58–67, Sept 2007. doi: 10.1109/ESSDERC.2007.4430882.

M. Kleider. personal communication, 2014.

A. Kononov. Testing of an Analog Neuromorphic Network Chip. Diploma thesis (English), University of Heidelberg, HD-KIP-11-83, 2011.

A. Krikelis and C. Weems. Associative processing and processors. *Computer*, 27 (11):12–17, 1994. ISSN 0018-9162. doi: 10.1109/2.330035.

J. Kunz. Vermessung des Transistor Missmatches eines 65nm Stromspiegels und Testen von 65nm SRAM-Speicherzellen. Internship report (German), Heidelberg University, 2012.

K. R. Laker and W. M. C. Sansen. *Design of Analog Integrated Circuits and Systems*. McGraw-Hill,Inc, 1994. ISBN 007036060.

J. Lau. Evolution and outlook of tsv and 3d ic/si integration. In *Electronics Packaging Technology Conference (EPTC), 2010 12th*, pages 560–570, Dec 2010. doi: 10.1109/EPTC.2010.5702702.

M. Lenzlinger and E. Snow. Fowler-nordheim tunneling into thermally grown sio2. *Electron Devices, IEEE Transactions on*, 15(9):686–686, Sep 1968. ISSN 0018-9383. doi: 10.1109/T-ED.1968.16430.

J. Liu, Y. Allasasmeh, and S. Gregori. Fully-integrated charge pumps without oxide breakdown limitation. In *Electrical and Computer Engineering (CCECE), 2011 24th Canadian Conference on*, pages 001474–001477, May 2011. doi: 10.1109/CCECE.2011.6030708.

D. Long, X. Hong, and S. Dong. Optimal two-dimension common centroid layout generation for MOS transistors unit-circuit. In *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, pages 2999–3002 Vol. 3, May 2005. doi: 10.1109/ISCAS.2005.1465258.

S. Lovett, M. Welten, A. Mathewson, and B. Mason. Optimizing MOS transistor mismatch. *Solid-State Circuits, IEEE Journal of*, 33(1):147–150, Jan 1998. ISSN 0018-9200. doi: 10.1109/4.654947.

Lua. Website. `http://www.lua.org`, 2014.

P.-I. Mak and R. Martins. High-/mixed-voltage rf and analog cmos circuits come of age. *Circuits and Systems Magazine, IEEE*, 10(4):27–39, Fourthquarter 2010. ISSN 1531-636X. doi: 10.1109/MCAS.2010.938636.

H. Markram, J. Lübke, and B. Sakmann. Regulation of Synaptic Efficacy By Coincidence of Postsynaptic Aps. *Science*, 275:213–215, 1997.

W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, pages 127–147, 1943.

C. A. Mead. Neuromorphic Electronic Systems. *Proceedings of the IEEE*, 78:1629–1636, 1990.

C. A. Mead and M. A. Mahowald. A silicon model of early visual processing. *Neural Networks*, 1(1):91–97, 1988.

P. Merolla, J. Arthur, F. Akopyan, N. Imam, R. Manohar, and D. Modha. A digital neurosynaptic core using embedded crossbar memory with 45pj per spike in 45nm. In *Custom Integrated Circuits Conference (CICC), 2011 IEEE*, pages 1–4, Sept 2011. doi: 10.1109/CICC.2011.6055294.

S. Millner. An Integrated Operational Amplifier for a Large Scale Neuromorphic System. Diploma thesis, University of Heidelberg, HD-KIP-08-19, 2008.

S. Millner. *Development of a Multi-Compartment Neuron Model Emulation.* PhD thesis, Ruprecht-Karls Universität Heidelberg, November 2012. URL `http://www.ub.uni-heidelberg.de/archiv/13979`.

S. Millner, A. Grübl, K. Meier, J. Schemmel, and M.-O. Schwartz. A VLSI Implementation of the Adaptive Exponential Integrate-and-Fire Neuron Model. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 1642–1650, 2010.

G. E. Moore. Cramming more components onto integrated circuits, reprinted from electronics, volume 38, number 8, april 19, 1965, pp.114 ff. *Solid-State Circuits Society Newsletter, IEEE*, 11(5):33–35, Sept 2006. ISSN 1098-4232. doi: 10.1109/N-SSC.2006.4785860.

A. Morrison, C. Mehring, T. Geisel, A. Aertsen, and M. Diesmann. Advancing the boundaries of high connectivity network simulation with distributed computing. *Neural Comput.*, 17(8):1776–1801, 2005.

A. Morrison, M. Diesmann, and W. Gerstner. Phenomenological models of synaptic plasticity based on spike timing. *Biological Cybernetics*, 98(6):459–478, June 2008. ISSN 0340-1200. doi: 10.1007/s00422-008-0233-1.

E. Müller. personal communication, 2014.

I. Myderrizi and A. Zeki. Current-steering digital-to-analog converters: Functional specifications, design basics, and behavioral modeling. *Antennas and Propagation Magazine, IEEE*, 52(4):197–208, Aug 2010. ISSN 1045-9243. doi: 10.1109/MAP.2010.5638288.

S. Narasimha, P. Chang, C. Ortolland, D. Fried, E. Engbrecht, K. Nummy, P. Parries, T. Ando, M. Aquilino, N. Arnold, R. Bolam, J. Cai, M. Chudzik, B. Cipriany, G. Costrini, M. Dai, J. Dechene, C. Dewan, B. Engel, M. Gribelyuk, D. Guo, G. Han, N. Habib, J. Holt, D. Ioannou, B. Jagannathan, D. Jaeger, J. Johnson, W. Kong, J. Koshy, R. Krishnan, A. Kumar, M. Kumar, J. Lee, X. Li, C. Lin, B. Linder, S. Lucarini, N. Lustig, P. McLaughlin, K. Onishi, V. Ontalus, R. Robison, C. Sheraw, M. Stoker, A. Thomas, G. Wang, R. Wise, L. Zhuang, G. Freeman, J. Gill, E. Maciejewski, R. Malik, J. Norum, and P. Agnello. 22nm high-performance soi technology featuring dual-embedded stressors, epi-plate high-k deep-trench embedded dram and self-aligned via 15lm beol. In *Electron Devices*

Meeting (IEDM), 2012 IEEE International, pages 3.3.1–3.3.4, Dec 2012. doi: 10.1109/IEDM.2012.6478971.

B. Nauta and A.-J. Annema. Analog/rf circuit design techniques for nanometer-scale ic technologies. In *Solid-State Circuits Conference, 2005. ESSCIRC 2005. Proceedings of the 31st European*, pages 45–53, Sept 2005. doi: 10.1109/ESSCIR.2005.1541556.

K. Ohsaki, N. Asamoto, and S. Takagaki. A single poly eeprom cell structure for use in standard cmos processes. *Solid-State Circuits, IEEE Journal of*, 29(3): 311–316, Mar 1994. ISSN 0018-9200. doi: 10.1109/4.278354.

B. Ostendorf. Charakterisierung eines Neuronalen Netzwerk-Chips. Diploma thesis (German), University of Heidelberg, HD-KIP 07-12, 2007.

B. Pakkenberg, D. Pelvig, L. Marner, M. J. Bundgaard, H. J. G. Gundersen, J. R. Nyengaard, and L. Regeur. Aging and the human neocortex. *Experimental gerontology*, 38(1):95–99, 2003.

M. Pelgrom, A. C. J. Duinmaijer, and A. Welbers. Matching properties of mos transistors. *Solid-State Circuits, IEEE Journal of*, 24(5):1433–1439, Oct 1989. ISSN 0018-9200. doi: 10.1109/JSSC.1989.572629.

T. Pfeil, T. C. Potjans, S. Schrader, W. Potjans, J. Schemmel, M. Diesmann, and K. Meier. Is a 4-bit synaptic weight resolution enough? - Constraints on enabling spike-timing dependent plasticity in neuromorphic hardware. *Frontiers in Neuroscience*, 6(90), 2012. ISSN 1662-453X. doi: 10.3389/fnins.2012.00090.

P. Pieters and E. Beyne. 3d wafer level packaging approach towards cost effective low loss high density 3d stacking. In *Electronic Packaging Technology, 2006. ICEPT '06. 7th International Conference on*, pages 1–4, Aug 2006. doi: 10.1109/ICEPT.2006.359749.

M. Pospischil, Z. Piwkowska, T. Bal, and A. Destexhe. Comparison of different neuron models to conductance-based post-stimulus time histograms obtained in cortical pyramidal cells using dynamic-clamp in vitro. *Biological Cybernetics*, 105(2):167–180, 2011. ISSN 0340-1200. doi: 10.1007/s00422-011-0458-2. URL `http://dx.doi.org/10.1007/s00422-011-0458-2`.

X. Qi, S. Lo, A. Gyure, Y. Luo, M. Shahram, K. Singhal, and D. MacMillen. Efficient subthreshold leakage current optimization - leakage current optimization and layout migration for 90- and 65- nm asic libraries. *Circuits and Devices Magazine, IEEE*, 22(5):39–47, Sept 2006. ISSN 8755-3996. doi: 10.1109/MCD.2006.272999.

P. Ramm, A. Klumpp, J. Weber, N. Lietaer, M. Taklo, W. De Raedt, T. Fritzsch, and P. Couderc. 3d integration technology: Status and application development.

In *ESSCIRC, 2010 Proceedings of the*, pages 9–16, Sept 2010. doi: 10.1109/ESSCIRC.2010.5619857.

B. Razavi. *Design of Analog CMOS Integrated Circuits.* McGraw-Hill, Inc., New York, NY, USA, 1 edition, 2001. ISBN 0072380322, 9780072380323.

W. Sansen, M. Steyaert, V. Peluso, and E. Peeters. Toward sub 1 V analog integrated circuits in submicron standard CMOS technologies. In *Solid-State Circuits Conference, 1998. Digest of Technical Papers. 1998 IEEE International*, pages 186–187, Feb 1998. doi: 10.1109/ISSCC.1998.672428.

W. M. C. Sansen. *Analog Design Essentials (The International Series in Engineering and Computer Science).* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. ISBN 0387257462.

H. Sarbishaei. *Electrostatic Discharge Protection Circuit for High-Speed Mixed-Signal Circuits.* PhD thesis, University of Waterloo, 2007.

J. Schemmel, A. Grübl, K. Meier, and E. Muller. Implementing Synaptic Plasticity in a VLSI Spiking Neural Network Model. In *Proceedings of the 2006 International Joint Conference on Neural Networks (IJCNN).* IEEE Press, 2006.

J. Schemmel, D. Brüderle, K. Meier, and B. Ostendorf. Modeling Synaptic Plasticity within Networks of Highly Accelerated I&F Neurons. In *Proceedings of the 2007 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 3367–3370. IEEE Press, 2007.

J. Schemmel, J. Fieres, and K. Meier. Wafer-Scale Integration of Analog Neural Networks. In *Proceedings of the 2008 International Joint Conference on Neural Networks (IJCNN)*, 2008.

J. Schemmel, D. Brüderle, A. Grübl, M. Hock, K. Meier, and S. Millner. A Wafer-Scale Neuromorphic Hardware System for Large-Scale Neural Modeling. In *Proceedings of the 2010 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1947–1950, 2010.

J. Schemmel, A. Grübl, S. Millner, S. Friedmann, and A. Hartel. Specification of the HICANN Microchip. FACETS and BrainScaleS project internal documentation, 2014.

M.-O. Schwartz. *Reproducing Biologically Realistic Regimes on a Highly-Accelerated Neuromorphic Hardware System.* PhD thesis, Ruprecht-Karls Universität Heidelberg, 2013. URL `http://www.ub.uni-heidelberg.de/archiv/14631`.

J. Seo, B. Brezzo, Y. Liu, B. Parker, S. Esser, R. Montoye, B. Rajendran, J. Tierno, L. Chang, D. Modha, and D. Friedman. A 45nm cmos neuromorphic chip with a scalable architecture for learning in networks of spiking neurons. In *Custom Integrated Circuits Conference (CICC), 2011 IEEE*, pages 1–4, Sept 2011. doi: 10.1109/CICC.2011.6055293.

M. Sinha, S. Hsu, A. Alvandpour, W. Burleson, R. Krishnamurthy, and S. Borkar. High-performance and low-voltage sense-amplifier techniques for sub-90nm SRAM. In *SOC Conference, 2003. Proceedings. IEEE International [Systems-on-Chip]*, pages 113–116, Sept 2003. doi: 10.1109/SOC.2003.1241474.

N. Sirisantana and K. Roy. Low-power design using multiple channel lengths and oxide thicknesses. *Design Test of Computers, IEEE*, 21(1):56–63, Jan 2004. ISSN 0740-7475. doi: 10.1109/MDT.2004.1261850.

G. Snider. Spike-timing-dependent learning in memristive nanodevices. In *Nanoscale Architectures, 2008. NANOARCH 2008. IEEE International Symposium on*, pages 85–92, 2008. doi: 10.1109/NANOARCH.2008.4585796.

SPICE. Website. http://bwrc.eecs.berkeley.edu/classes/icbook/spice/, 2014.

D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams. The missing memristor found. *Nature*, 453(7191):80–83, 2008.

SystemVerilog. *SystemVerilog 3.1a Language Reference Manual.* Accellera, 2004.

M. Tsodyks and H. Markram. The neural code between neocortical pyramidal neurons depends on neurotransmitter release probability. *Proceedings of the national academy of science USA*, 94:719–723, Jan. 1997.

UrJTAG. Website. `http://www.urjtag.org`, 2014.

Verilog. Ieee standard for verilog hardware description language. *IEEE Std 1364-2005 (Revision of IEEE Std 1364-2001)*, pages 1–560, 2006. doi: 10.1109/IEEESTD.2006.99495.

VHDL. Ieee standard vhdl language reference manual. *IEEE Std 1076-1987*, pages 1–, 1988. doi: 10.1109/IEEESTD.1988.122645.

J. von Neumann. First draft of a report on the EDVAC. Technical report, Moore School of Electrical Engeneering Library, University of Pennsylvania, 1945. Transscript in: M. D. Godfrey: Introduction to “The first draft report on the EDVAC” by John von Neumann. IEEE Annals of the History of Computing 15(4), 27–75 (1993).

B. A. Wandell, S. O. Dumoulin, and A. A. Brewer. Visual field maps in human cortex. *Neuron*, 56(2):366–383, Oct 2007.

F. A. Wilson, S. P. O’Scalaidhe, and P. S. Goldman-Rakic. Functional synergism between putative gamma-aminobutyrate-containing neurons and pyramidal neurons in prefrontal cortex. *Proceedings of the National Academy of Sciences*, 91(9): 4009–4013, 1994. URL `http://www.pnas.org/content/91/9/4009.abstract`.

R. Wojtyna. A concept of current-mode long-term analog memory for neural-network learning on silicon. In *Signal Processing Algorithms, Architectures, Arrangements, and Applications (SPA), 2008*, pages 121–126, 2008.

Y.-D. Wu, K.-C. Cheng, C.-C. Lu, and H. Chen. Embedded Analog Nonvolatile Memory With Bidirectional and Linear Programmability. *Circuits and Systems II: Express Briefs, IEEE Transactions on*, 59(2):88–92, 2012. ISSN 1549-7747. doi: 10.1109/TCSII.2012.2184371.

I. Young. Analog mixed-signal circuits in advanced nano-scale CMOS technology for microprocessors and SoCs. In *ESSCIRC, 2010 Proceedings of the*, pages 61–70, Sept 2010. doi: 10.1109/ESSCIRC.2010.5619780.

Acknowledgements

Finally I want to express my gratitude to everyone who supported this work, especially:

Prof. Dr. Karlheinz Meier for supervision and the opportunity to work in the Electronic Vision(s) Group.

Prof. Dr. Peter Fischer as the second referee of this thesis.

Dr. Johannes Schemmel for supervision and lots of helpful and interesting discussions on technical issues.

Ralf Achenbach and Markus Dorn from the ASIC Lab for technical assistance with the CAD software and bonding of the prototype chips.

Simon Friedmann, Andreas Grübel, Andreas Hartel and Gvidas Sidlauskas for great teamwork during development of the prototype chips.

All proof readers, especially Simon and Andi.

Eric Müeller for competent support regarding any kind of software problems.

The entire Electronic Vision(s) Group for general support, being great coworkers and such an amazing bunch of crazy people.

Simon Friedmann for scientific and technical advice, interesting discussions about life, the universe and everything as well as being a good friend.

Christine Engeland for a lot of proof reading, support and much more.

The work reported was funded by the Seventh Framework Program of the EC under grant agreement no. 269921 (BrainScaleS).