Dominik Schmidt

Automated Characterization of a Wafer-Scale Neuromorphic Hardware System

Master Thesis

KIRCHHOFF-INSTITUT FÜR PHYSIK

Department of Physics and Astronomy
University of Heidelberg

Master Thesis
in Physics
submitted by
Dominik Schmidt
born in Wetzlar, Hessen

1. November 2014

Automated Characterization of a Wafer-Scale Neuromorphic Hardware System

This Master Thesis has been carried out by Dominik Schmidt at the
Kirchhoff Institute for Physics
Ruprecht-Karls-Universität Heidelberg
under the supervision of
Prof. Dr. Karlheinz Meier

Automated Characterization of a Wafer-Scale Neuromorphic Hardware System

Modeling neuronal networks is an effective way to better understand the brain or to tackle highly complex problems in machine learning. Neuromorphic hardware systems implement the emulation of neuronal networks on application-specific integrated circuits. These emulations can be highly accelerated and generally consume less power than conventional computer simulations. However, the fabrication of integrated circuits introduces transistor variations which lead to variations in neuron dynamics, hindering a precise emulation of neuronal networks. This thesis presents a software framework for automated characterization and calibration of the BrainScaleS Wafer-Scale neuromorphic chip, called HICANN. This calibration compensates the heterogeneity that is caused by transistor variations. Methods for the calibration of neuron parameters are introduced and their effectiveness is examined. It is shown that the variation across neuron parameters can be significantly reduced in an automated fashion, allowing for the emulation of a basic neural network on the wafer-scale system. Furthermore, advanced measurement methods grant insights on additional neuron characteristics such as the membrane capacitance, leading to a better understanding of the circuits.

Automatisierte Charakterisierung eines neuromorphen Hardwaresystems auf Wafer-Ebene

Die Modellierung von neuronalen Netzen ist ein effektives Mittel, um das Gehirn besser zu verstehen oder Probleme des maschinellen Lernens zu lösen. Neuromorphe Hardwaresysteme implementieren die Emulation von neuronalen Netzen in anwendungsspezifischen integrierten Schaltkreisen. Diese Emulationen können stark beschleunigt sein und deutlich weniger Strom verbrauchen, als konventionelle Computersimulationen. Die Herstellung von integrierten Schaltkreisen bringt allerdings Variationen in Transistorcharakteristika mit sich, die zu Variationen in Neuronen-Parametern führen. Mit dieser Arbeit wird ein Software-Framework vorgestellt, welches das automatisierte Charakterisieren und Kalibrieren des BrainScaleS Wafer-Scale neuromorphen Chips HICANN ermöglicht. Diese Kalibrierung kompensiert die durch Transistorvariationen hervorgerufene Heterogenität. Im Fokus der Arbeit stehen die Methoden zur Kalibrierung von Neuronenparametern sowie Untersuchungen hinsichtlich deren Effizienz. Es wird gezeigt, dass sich Variationen von Neuronenparametern signifikant reduzieren lassen, was die Emulation von großskaligen neuronalen Netzen ermöglicht. Darüberhinaus geben erweitere Charakterisierungsmethoden Einblicke in zusätzliche Neuronencharakteristika wie beispielsweise die Membrankapazität. Dies führt zu einem besseren Verständnis der Schaltkreise.

Contents

1 Introduction

The human brain is one of the most fascinating living structures in the known universe. Its capabilities to store and process huge amounts of data while consuming less power than a standard light bulb is unmatched by today's technology. Considering this, it is no surprise that understanding the human brain is one of the key scientific challenges in the twenty-first century. The significance of neuroscience is reflected in the amount of large-scale scientific projects that appeared in the last years. Scientists from many scientific areas have come together to start projects like the Human Brain Project in Europe, the BRAIN Initiative in the United States or the Brain/MINDS project in Japan. The scales of these projects are compared to the scale of the human genome project at the end of the twentieth century.

In 2006, the World Health Organization estimated that nearly one in six people worldwide suffers from some form of neurological disorders (*Aarli et al.*, 2006). A better understanding of the causes for neurological disorders within the brain itself is vital to improving the overall mental health of the public.

Besides benefits in medical research, a better understanding of how the brain processes data can result in innovative approaches to computing and artificial intelligence. Already, simulations of brain-inspired artificial neural networks perform pattern recognition tasks exceedingly well (*Krizhevsky et al.*, 2012). However, simulating a detailed neural network of the size of the human brain would consume more than 10 megawatts of power – the amount a large hydroelectric plant is able to produce – while being magnitudes slower than the biological counterpart. This is one of the motivations for scientific groups worldwide to come up with new ideas on how to implement biologically inspired physical models of neurons on application-specific integrated circuits (ASICs), so-called neuromorphic devices.

One of these groups is the Electronic Vision(s) group at the University of Heidelberg. Within the FACETS-project, the group successfully developed an analog integrated circuit called Spikey, emulating several hundreds neurons with tens of thousands of synapses. It was proven that basic functional neural networks (*Pfeil et al.*, 2013) as well as more sophisticated classifiers (*Schmuker et al.*, 2014) can be implemented in Spikey. The next step to a larger scale emulation of the brain was the start of the BrainScaleS project in the beginning of 2011, which is still running until spring 2015. The BrainScaleS project involves three approaches: In-vivo biological experiments to gather actual brain data, the simulation of neural networks on petascale computers and the development of a wafer-scale neuromorphic device to emulate neural networks on a large scale.

The use of analog neuromorphic devices for the emulation of neural networks has many benefits. The most compelling one is the low power consumption compared to supercomputer simulations of similar size. Another major difference is that analog circuits

run in continuous time unlike traditional, clocked computers. The designer of a neuromorphic circuit can also choose the time scales of the model to be much smaller than biological time scales, resulting in a device that not only consumes a lot less power than a supercomputer, but also runs highly accelerated. The BrainScaleS waferscale-system incorporates both of these advantages, being approximately up to 10^5 times faster than biological neurons. The power consumption of one synaptic transmission is estimated to be several orders of magnitude lower on the BrainScaleS waferscale-system than in a supercomputer simulation.

The designers of neural network models often need to precisely define neuron parameters in order to maintain the successful function of a model. Running the model on a computer simulation will always result in the same outcome. On the other hand, the fabrication of analog circuits introduces variations in transistor size which lead to a deviation of neuron parameters on different instances of the circuit. These deviations often make it hard to directly transfer neural networks from software implementation onto the hardware. However, the high configurability of the BrainScaleS hardware system includes tuning of almost every parameter down to the level of single neurons, enabling the calibration of most circuits. By means of calibration, transistor variations can be overcome to a certain extend, facilitating the transfer of neural models from software to hardware. Because of that, calibration is an important step to make the hardware system accessible to users with functional neural networks waiting to be implemented on a highly accelerated system.

During this master thesis, a software framework was developed for automated characterization and calibration of a High Input Count Analog Neural Network (HICANN) chip, which is the neuromorphic component of the BrainScaleS waferscale system. This thesis covers methods and results of the calibration process for neuron circuits on the HICANN chip.

First, the hardware parts that are important for understanding the calibration are explained in detail and effects of transistor size mismatches are illustrated. To get a basic neural network running on a calibrated system, the calibration focuses on Leaky Integrate-and-Fire (LIF) neuron parameters, while adaptation and exponential term are turned off. For each parameter, measurement and calibration methods are explained and results of the calibration on one HICANN chip are shown. Finally, the methods and results are discussed and an outlook is given over future improvements to the calibration methods.

2 The HICANN Chip

A main goal of the BrainScaleS project is the development of a large-scale analog neuromorphic hardware system, called the Hybrid Multiscale Facility (HMF). The HMF consists of two parts: a neuromorphic hardware module emulating highly accelerated neuron circuits and a computer cluster that supports the communication with the hardware and runs software simulations. The core of the neuromorphic hardware is the 20 cm silicon wafer, which is produced in a UMC 180 nm process. Up to 384 High Input Count Analog Neural Network (HICANN) chips (see fig. 2.1) are located on one wafer, organized in reticles of eight chips each. The novelty of this system is that the wafer is not cut into separate chips, but left uncut, and a communication layer connecting the chips is added via post-processing (*Schemmel et al.*, 2010).

The HICANN chip itself is a highly configurable mixed-signal processor, where analog circuits are used to implement physical models of neurons, while digital circuits are used for communication. A large area on the chip is taken by the Analog Neural Network Core (ANNCORE) which contains 512 analog neuron circuits and 112.000 synapses in total. The ANNCORE is surrounded by the digital communication circuits. Synapse drivers on the edges of each synapse array convert digital spike information to an analog signal that is transferred to the synapses. Switches in the synapses can be programmed to connect the synapse drivers with the synaptic input circuits inside the neurons, where the shape of the post-synpatic potential (PSP) is modeled and trasmitted to the membrane. This setup allows for mapping of almost arbitrary network topologies onto the hardware.

In this chapter, the relevant parts of the neuron circuit as well as the physical model it emulates will be described in detail. Documentation of all other parts of the chip can be found in *Schemmel et al.* (2012) and *Millner* (2012).

Since variations in transistor characteristics (transistor mismatch) appear in every fabricated circuit, it is important to know their influence on the behaviour of the circuits. In addition to the ideal circuit behaviour, these effects are illustrated. Information on their estimated magnitudes is mainly based on Monte-Carlo simulations done in *Kiene* (2014).

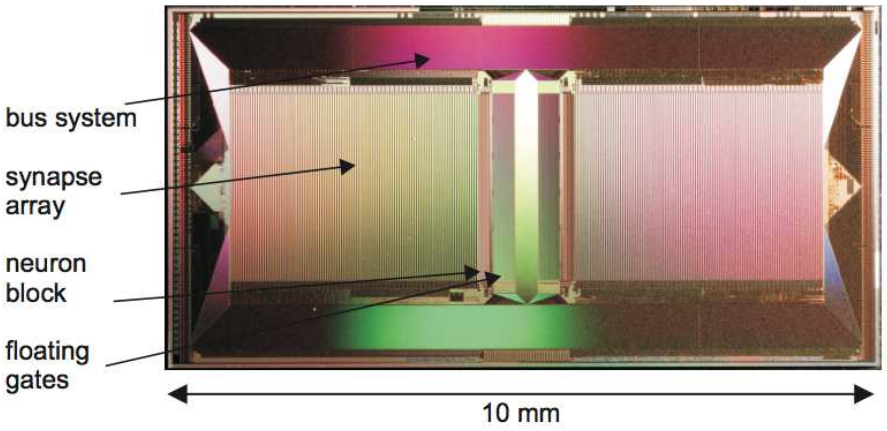

Figure 2.1 Photograph of a HICANN chip. Source: *Schwartz* (2013).

2.1 Adaptive Exponential Integrate-and-Fire Model

The neuron model implemented on the HICANN chip is the Adaptive Exponential Leaky Integrate-and-Fire (AdEx) neuron model introduced by *Brette and Gerstner* (2005). The AdEx model is an enhancement of the well-established Leaky Integrate-and-Fire (LIF) neuron model, adding two terms that allow imitation of many spiking patterns observed in biology, such as phasic spiking, bursting or spike frequency adaptation.

The model dynamics are described by two differential equations:

$$C\frac{dV}{dt} = -g_l(V - V_{\text{rest}}) + g_l\Delta_T \exp\left(\frac{V - V_{\text{exp}}}{\Delta_t}\right) - w + I, \tag{2.1}$$

$$\tau_w \frac{dw}{dt} = a(V - V_{\text{rest}}) - w, \tag{2.2}$$

where C is the capacity of the membrane, g_l the leakage conductance and V_{rest} the resting potential, together forming the leaky integrator part. The exponential term consists of the parameters V_{exp} and Δ_T, which are the effective threshold potential and slope factor of the exponential term. This term facilitates the sharp rise of the membrane voltage after a certain threshold is crossed. The adaptation behaviour is implemented in the term w, where τ_w is the adaptation time constant and a the adaptation parameter controlling the strength of subthreshold adaptation. Incoming spikes or other injected currents are summarized by the term I in the equation.

The spiking of neurons is not implemented in the differential equations but by an additional reset condition. If the membrane voltage V crosses a certain threshold V_t, then

$$V \to V_{\text{reset}}, \tag{2.3}$$

$$w \to w + b, \tag{2.4}$$

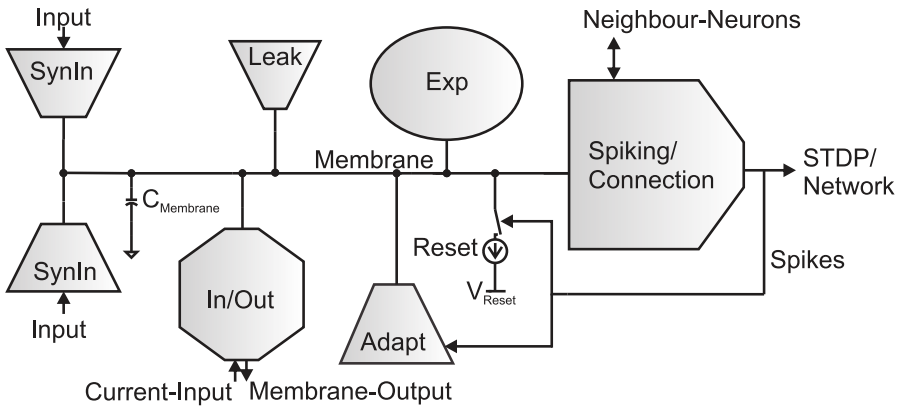

Figure 2.2 Simplified neuron schematic showing the individual modules. Source: *Millner* (2012).

where V_{reset} is the reset potential and b the increase of adaptation in case of a spike. As such, b regulates the strength of spike-frequency adaptation.

2.2 The Dendritic Membrane Circuit

A dendritic membrane circuit consists of several modules connected to a membrane capacitor, each representing a certain part of the AdEx neuron model (fig. 2.2). The size of the capacitor can be switched between 2.16 nF and 0.16 nF, allowing for a wider range of time constants. To increase the number of inputs per neuron, up to 64 dendritic membrane circuits can be connected to form one large neuron. Another feature of the neuron circuit is that single modules can be turned off. To explain the function of a basic neuron as a starting point, the focus will be on describing the modules that emulate LIF parameters. Adaptation and exponential terms are turned off during all experiments presented in this thesis (see chapter 3 for the basic experiment setup) and are therefore not explained here.

2.2.1 The operational transconductance amplifier

The core device of the neuron circuit is the operational transconductance amplifier (OTA), which appears seven times in the whole circuit. Its main function is to serve as a tuneable conductance that emulates the ion channels of biological neurons.

The defining characteristic of an ideal conductance is that the output current I_{out} is linear to the voltage difference V_d at the terminals:

$$I_{out} = g \cdot V_d. \tag{2.5}$$

However, in the implemented OTA, this only holds true within a certain range, called the linear range. Theoretically, the upper limit for the V_d is proportional to $\sqrt{I_b}$, where I_b is the bias current (*Millner*, 2012). For values of V_d above this limit, the output current

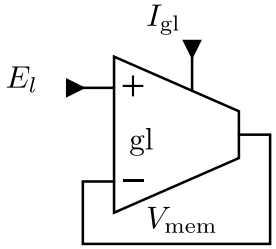

Figure 2.3 operational transconductance amplifier functioning as the leakage term. In the linear range, the output current is proportional to the difference $\mathrm{V_d}$ between $\mathrm{E_l}$ and $\mathrm{V_{mem}}$. Source: *Millner* (2012).

of the OTA approaches the limit I_b. Simulations show that differential voltages larger than 100 mV are no longer within the linear range (*Kiene*, 2014).

The most basic use of an OTA within the neuron circuit is a direct connection of the leakage potential $\mathrm{E_l}$ to the membrane, with the OTA emulating the leakage conductance $\mathrm{g_l}$ (see fig. 2.3). The strength of the conductance can be controlled by the bias current $\mathrm{I_{gl}}$, which is also the maximum current that the OTA can provide. In this instance, the OTA permanently drives the membrane potential towards $\mathrm{E_l}$, leading to a constant steady state voltage in the absence of other inputs. In biology, this steady state voltage is known as the resting potential $\mathrm{V_{rest}}$.

Transistor Mismatch Effects

The primary effect of transistor mismatch in the OTA is a shift of differential input common mode $\Delta\mathrm{V_{in}}$, i.e. the differential voltage at which no current flows at the output of the OTA. This voltage should ideally be zero. Monte Carlo simulations done in *Kiene* (2014) show an effective $\Delta\mathrm{V_{in}}$ of

$$\Delta\mathrm{V_{in}} : \mu = -12\,\mathrm{mV}, \sigma = 23\,\mathrm{mV} \tag{2.6}$$

at a bias current of 2 µA. As a neuron-level consequence of this, a variation in resting potentials across all neurons should be observable:

$$\mathrm{V_{rest}} = \mathrm{E_l} + \Delta\mathrm{V_{in}} \tag{2.7}$$

2.2.2 Spike Detection and Reset

The spike detection is implemented as a differential OTA with additional feedback. If the membrane voltage $\mathrm{V_{mem}}$ surpasses a certain threshold θ, a spike event is triggered. This event is sent to the inter-HICANN communication layer, the plasticity controller, the adaptation circuit and the reset circuit of this neuron, triggering the reset condition introduced in eqs. (2.3) and (2.4).

When a spike is triggered inside the reset circuit, the membrane gets pulled toward the potential $\mathrm{V_{reset}}$ by a strong current $\mathrm{I_{reset}}$. This happens for a certain amount of time called the refractory period τ_{ref}. This time depends on the current $\mathrm{I_{pl}}$ where a lower $\mathrm{I_{pl}}$ leads to a longer τ_{ref}. The reset voltage $\mathrm{V_{reset}}$ is shared among blocks of 128 neurons. Each shared floating gate block provides the reset voltage for either odd or even neurons on the top neuron block or odd or even neurons on the bottom block (see fig. 2.4).

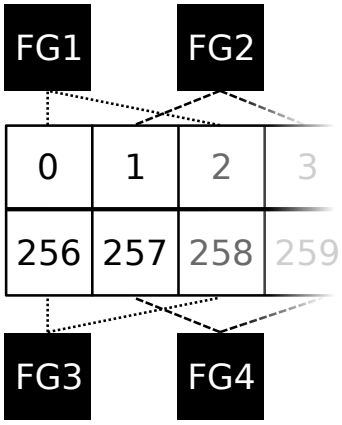

Figure 2.4 Connection scheme of V_{reset} to the neurons. Each floating gate block provides the voltage to either odd or even neuron numbers on either the top or the bottom neuron block.

Transistor Mismatch Effects

The main effect of transistor mismatch in the comparator is a variation of the effective threshold θ. Measurement and correction methods for this effect are explained in section 3.9.

2.2.3 Synaptic Input

The function of the synaptic input term is to convert very short (5 ns pulse length at 200 MHz clock frequency) rectangular current pulses to post-synaptic conductances (PSCs) with shapes that mimic biological PSCs.

The current pulses from the synapse arrive at the input I_{syn} of the operational amplifier (OP) and are integrated by the capacitor C and the resistive element R (fig. 2.5), while V_{syn} is kept at a constant value. R can be adjusted by a control voltage V_{syntc} in order to change the time constant of the integrator. The integrated pulse is then converted to a current by OTA_1 which scales with the bias current I_{conv}. Finally, by using this current as bias in OTA_0, it is translated to a conductance between the synaptic reversal potential E_{syn} and the membrane voltage V_{mem}, emulating the behaviour of conductance-based synapses. For detailed analysis and simulation of the synaptic input circuit, see *Kiene* (2014).

Transistor Mismatch Effects

There are mainly three devices at which transistor mismatches cause large variations. Simulations show that the OP amplifier has a total offset of

$$\Delta V_{OP} : \mu = 0.7\,\text{mV}, \sigma = 8.7\,\text{mV}, \tag{2.8}$$

while both OTAs are equal to the one described in section 2.2.1. Combining the offsets of OTA_1 and the OP results in a total offset of

$$\Delta V_{OTA_1} : \mu = -13\,\text{mV}, \sigma = 26\text{mV} \tag{2.9}$$

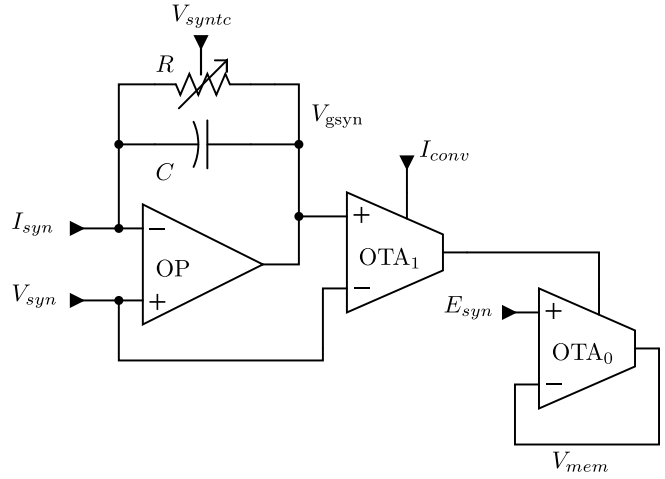

Figure 2.5 Simplified schematic of the synaptic input term. When a spike event occurs, a rectangular current pulse is injected via I_{syn} and converted into an alpha-shaped conductance in OTA_0. Source: *Millner* (2012).

at OTA_1. Unlike the leakage offset, this offset cannot be corrected for since the integrator always works relative to $\mathrm{V_{syn}}$ and OTA_1 is connected to that same voltage. This problem could be solved by connecting it to a separate voltage line, thus decoupling it from the integrator, as suggested in *Kiene* (2014).

The result of a negative $\Delta \mathrm{V}_{\mathrm{OTA}_1}$ is a constant positive current flowing into the bias gate of OTA_0. This generates a nonzero conductance $\mathrm{g_{syn}}$ even in the absence of spikes. On neuron level, the resting pontentials will be shifted towards the reversal potentials, even if no spikes are incoming. Theoretically, the effective resting potential of a neuron when combining the leakage conductance towards $\mathrm{E_l}$ with the unwanted leakage conductances towards the synaptic reversal potentials $\mathrm{E_{syn<i,x>}}$ will be

$$\mathrm{V_{rest,eff}} = \frac{\mathrm{g_l} \cdot \mathrm{E_l} + \mathrm{g_{syni}} \cdot \mathrm{E_{syni}} + \mathrm{g_{synx}} \cdot \mathrm{E_{synx}}}{\mathrm{g_l} + \mathrm{g_{syni}} + \mathrm{g_{synx}}}, \tag{2.10}$$

where i denotes the inhibitory synaptic input and x the excitatory synaptic input.

2.2.4 Input/Output Circuit

To conduct single neuron experiments, it is helpful to observe not only spiking events but also the membrane voltage of a neuron. Furthermore, the injection of currents directly into a neuron can be used to induce a certain behaviour and extract neuron characteristics. The input/output circuit (called In/Out in fig. 2.2) implements both of these tasks. A simplified schematic of this circuit is found in fig. 2.6. The first part of the circuit connects one of the four HICANN's current stimulation circuits to the neuron via a transmission gate. The connection scheme for the current stimulus generators is identical to the one for $\mathrm{V_{reset}}$ found in fig. 2.4. It is important to note that the connection of the membrane circuit to a current stimulation line increases the total

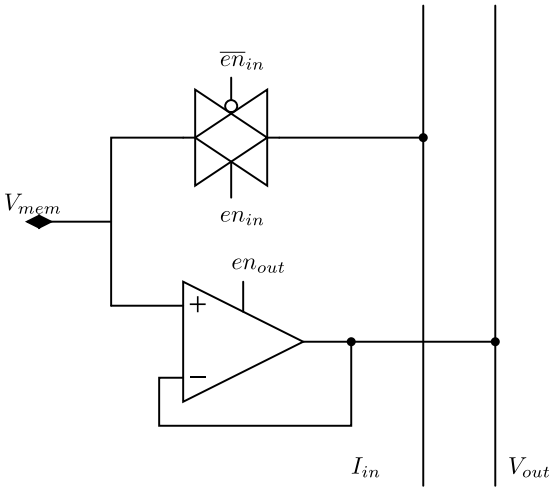

Figure 2.6 Simplified schematic of the input/output circuit. Source: *Millner* (2012).

capacitance of that neuron circuit (see section 4.3 for measurements). The second part of the circuit consists of an OTA that connects the membrane voltage to the line V_{out} which is connected to the HICANN readout amplifier. In the following chapters, this OTA is called the neuron readout amplifier or neuron readout buffer. Since there are two HICANN readout amplifiers and four input current generators on the chip, this setup enables readout of up to two neurons simultaneously, while enabling the stimulus of up to four neurons simultaneuously.

Transistor Mismatch Effects

In the current input circuit, no significant transistor mismatch effects are visible since it only consists of a single transmission gate. The output circuit however consists of an OTA, where transistor mismatch leads to the previously mentioned shifts. Thus, a shift in the effective readout voltage is expected. Measuring this shift requires knowing the actual membrane voltage and comparing it to the measured membrane voltage (see section 3.4). Another transistor mismatch effect are shifts in the transmitted voltages of the HICANN's readout amplifiers. These shifts however are constant over time, so each amplifier will always introduce the same voltage shift. Correcting for this shift is only important when using both amplifiers. Since only one of the amplifiers is used throughout this thesis, the readout shift of the HICANN's readout amplifier will not be correctied.

2.3 Floating Gates

To provide all neuron circuits with adjustable parameters, four blocks of analog floating gate cells (described in *Lande et al.* (1996)) are placed on the HICANN chip. One floating gate block consists of 24 lines with 129 floating gate cells each. Each line stores 128 neuron parameters and one global parameter. In total, over 3000 parameter values can be stored on one floating gate block. The floating gate cells themselves can provide voltages in a range from 0 V to 1.8 V or currents in a range from 0 µA to 2.5 µA. The

Table 2.1 Parameter naming in different domains. Hardware parameters describe the observable behaviour of a neuron while floating gate parameters are used to control the behaviour. For the remaining floating gate parameters, see table 2.3.

Model (full name)	Hardware	HW range	Floating gate
resting potential	V_{rest}	0 V to 1.2 V	E_l
spike threshold	V_t	0 V to 1.2 V	V_t
reset potential	V_{reset}	0 V to 1.2 V	V_{reset} (global)
excitatory synaptic reversal potential	E_{synx}	0 V to 1.2 V	E_{synx}
inhibitory synaptic reversal potential	E_{syni}	0 V to 1.2 V	E_{syni}
excitatory synaptic time constant	τ_{synx}		V_{syntcx}
inhibitory synaptic time constant	τ_{syni}		V_{syntci}
refractory period	τ_{ref}	0.01 µs to 6 µs	I_{pl}
leakage conductance	g_l	0.1 µS to 6 µS	I_{gl}
membrane time constant	τ_{mem}	0.5 µs to 15 µs	I_{gl}
adaptation coupling parameter	a		$I_{gladapt}$
adaption time constant	τ_w		I_{radapt}
spike-triggered adaptation	b		I_{fire}
effective threshold potential	V_{exp}	0 V to 1.2 V	V_{exp}
slope factor	Δ_t		I_{bexp}

programming is done by the floating gate controller, which is also located on the chip. The main reasons for using floating gate cells on the chip are the stability of programmed values (*Kononov*, 2011), the small size of the cells and the low power consumption.

However, since programming of a single floating gate cell will interfere with other floating gate cells, the whole array needs to be reprogrammed if only one value is changed. Currently, this is done prior to every experiment, totalling to more than half of the overall configuration time. Reprogramming the floating gates also introduces imprecisions in their output values, leading to slightly different values after each reprogramming. Thus, two consecutive runs of the same experiment will always result in a different outcome. More on floating gate variations can be found in chapter 3.

2.4 Parameter Translation

Since the hardware implementation of the AdEx-Model operates in a different range of parameters than biological neurons, a translation from biological to hardware parameters needs to be done.

To describe the transformations, three different parameter spaces can be defined (see table 2.1). Parameters that are used in the model and in software will be called biological (Bio) parameters, their scaled counterparts on the hardware will be called hardware (HW) parameters. Hardware parameters are controlled by voltages and currents that can be set on the hardware, which will be called floating gate (FG) parameters. Parameter

translation can therefore be divided into two separate translations. First, the biological domain needs to be mapped onto the hardware domain. Second, the dependencies of hardware parameters on floating gate parameters need to be measured.

From Biology to Hardware

The first transformation step can be defined by comparing the operating ranges of biological and hardware parameters. Typical membrane voltages in biological neurons range from $-100\,\mathrm{mV}$ to $0\,\mathrm{mV}$ with typical resting potentials at $-65\,\mathrm{mV}$ (*PyNN*, 2014), whereas the hardware system operates in a total voltage range from $0\,\mathrm{V}$ to $1.8\,\mathrm{V}$. The transformation from one domain to the other is linear:

$$V_{\mathrm{hw}} = V_{\mathrm{bio}} \cdot v_{\mathrm{scale}} + v_{\mathrm{shift}}. \tag{2.11}$$

The scaling factor is derived from the total dynamic ranges in biology $\Delta\ \mathrm{V}_{\mathrm{dyn,bio}}$ and hardware $\Delta\mathrm{V}_{\mathrm{dyn,HW}}$:

$$v_{\mathrm{scale}} = \frac{\Delta\mathrm{V}_{\mathrm{dyn,HW}}}{\Delta\mathrm{V}_{\mathrm{dyn,bio}}}. \tag{2.12}$$

The voltage shift is obtained by mapping a reference voltage of the biological to a reference voltage of the hardware domain. Here, we choose the lower boundary of the dynamic range as the point of reference to obtain

$$v_{\mathrm{shift}} = \min\left(\mathrm{V}_{\mathrm{dyn,\ HW}}\right) - v_{\mathrm{scale}} \cdot \min\left(\mathrm{V}_{\mathrm{dyn,\ bio}}\right). \tag{2.13}$$

Biological time constants are translated to hardware time constants by dividing by the chooseable hardware speedup factor t_{speedup} which is usually chosen to be around 10^4. Thus, all time constants will be transformed by

$$\tau_{\mathrm{mem,\ HW}} = \frac{\tau_{\mathrm{mem,\ bio}}}{t_{\mathrm{speedup}}} \tag{2.14}$$

$$\tau_{\mathrm{syn,\ HW}} = \frac{\tau_{\mathrm{syn,\ bio}}}{t_{\mathrm{speedup}}} \tag{2.15}$$

$$\tau_{\mathrm{ref,\ HW}} = \frac{\tau_{\mathrm{ref,\ bio}}}{t_{\mathrm{speedup}}}. \tag{2.16}$$

The leakage conductance is translated by using the hardware membrane capacitor and the scaled membrane time constant:

$$g_{\mathrm{leak,\ HW}} = \frac{C_{\mathrm{mem,\ HW}}}{\tau_{\mathrm{mem,\ HW}}}. \tag{2.17}$$

Simulations in *Schwartz* (2013) have shown that this transformation does not significantly change the neuron dynamics.

Realistic transformation

By examining the design of the hardware described in this chapter, the numbers for an ideal transformation can be obtained. First, the total range of the hardware from 0 mV to 1.8 mV can not be fully used because the OTA only behaves similar to a conductance within a differential voltage range of ±100 mV. Thus, the total dynamic range should be 200 mV at most. We choose the difference of inhibitory and excitatory synaptic reversal potentials to equal this range. Assuming that biological reversal potentials are at −100 mV and 0 mV, we obtain a scaling factor of:

$$v_{\text{scale}} = \frac{200\,\text{mV}}{100\,\text{mV}} = 2. \tag{2.18}$$

By placing the dynamic range at 800 mV to 1000 mV (the center of the supply voltage of 1.8 V) and using the lower end (i.e. the inhibitory synaptic reversal potential) as reference point, we obtain a voltage shift of

$$v_{\text{shift}} = 800\,\text{mV} - v_{\text{scale}} \cdot (-100\,\text{mV}) = 1000\,\text{mV}. \tag{2.19}$$

The reset potential V_{reset} can only be set for blocks of 128 neurons, which means that neurons cannot have individually different reset potentials. This problem can be solved by choosing V_{reset} as a point of reference for all other potentials, instead of the reversal potential. A typical biological reset potential is at −65 mV which results in 870 mV when using eqs. (2.18) and (2.19). Then, an alternative shift for all potentials can be introduced to each neuron:

$$v_{\text{shift},i} = 870\,\text{mV} - v_{\text{scale}} \cdot v_{\text{reset, bio},i}. \tag{2.20}$$

Using this transformation, neurons with different V_{reset} values will indeed operate in different ranges, but their dynamics will stay unchanged since all potential differences stay the same.

The speedup factor can not be set directly but follows from the properties of the hardware, where the factor of 10^4 is only an approximation. If the hardware membrane time constant is known, the speedup factor for all time constants is calculated by

$$t_{\text{speedup}} = \frac{\tau_{\text{mem,HW}}}{\tau_{\text{mem, bio}}} \tag{2.21}$$

Given this transformation and the estimated speedup factor of 10^4, table 2.2 summarizes the parameter transformation for the default pyNN parameters that are used in a Leaky Integrate-and-Fire neuron (*PyNN*, 2014).

Table 2.2 Summary of a realistic parameter translation. Biological values are taken from the default Leaky Integrate-and-Fire pyNN parameter set.

pyNN name	HW name	Bio value	HW value
v_rest	E_l	−65 mV	870 mV
v_reset	V_{reset}	−65 mV	870 mV
v_thresh	V_t	−50 mV	900 mV
e_rev_E	E_{synx}	0 mV	1000 mV
e_rev_I	E_{syni}	−100 mV	800 mV
tau_m	τ_{mem}	20 ms	2 ms
tau_refrac	τ_{ref}	0.1 ms	0.01 ms
tau_syn_I	τ_{syni}	5 ms	0.5 ms
tau_syn_e	τ_{syni}	5 ms	0.5 ms

Table 2.3 Overview over all technical parameters that do not directly affect neuron model parameters. Typical settings are also provided where they are known. For parameters that directly control model parameters, see table 2.1

Description	parameter name	typical setting
Neuron parameters		
bias current for synaptic input (max. conductance)	$I_{conv<i,x>}$	set to max.
integrator bias in synapse	$I_{intbb<i,x>}$	set to 2 µA
spike threshold comparator bias	$I_{spikeamp}$	set to 2 µA
voltage level of line to synapse array	$V_{syn<i,x>}$	set to 1 V
Shared parameters		
bias of floating gate array amplifiers	int_op_bias	set to max.
dll reset voltage	V_{dllres}	set to 0.36 V
global bias of neuron readout amplifiers	V_{bout}	set to 0.25 µA
bias for buffer of V_{exp}	V_{bexp}	2.0 µA − 2.5 µA
short-term plasticity in facilitation mode	V_{fac}	
current used to pull down membrane after reset	I_{breset}	set to max.
short-term plasticity in depression mode	V_{dep}	
STDP readout compare voltage causal	V_{thigh}	
max. synaptic weight	$V_{gmax<0,1,2,3>}$	set all to 80 mV
STDP readout compare voltage acausal	V_{tlow}	
V2 in STDP circuit (see *Schemmel et al.*, 2006)	$V_{clr<a,c>}$	
STDF reset voltage	V_{stdf}	
V1 in STDP circuit (see *Schemmel et al.*, 2006)	V_m	set to 0 V
bias for short-term plasticity	V_{bstdf}	
bias for DTC in short-term plasticity circuit	V_{dtc}	
bias for STDP readout circuit	V_{br}	

3 Calibration towards Neural Network Experiments

The implementation of large scale neural networks on the wafer-scale system is a highly complex problem. For this purpose, the automated mapping and routing tool "marocco" was developed (*Jeltsch*, 2014). To prepare hardware configurations for network experiments, marocco processes descriptions of neural networks written in PyNN (*Davison et al.*, 2008), where neuron parameters are usually defined in the biological parameter domain. The translation from biological domain to the hardware domain is done as described in section 2.4. Without calibration, the final translation from hardware domain to corresponding floating gate settings is done using the ideal transformations obtained by *Schwartz* (2013). These transformations use simulations of neuron circuits on a transistor level as a reference. However, the fabrication of integrated circuits introduces transistor variations which lead to deviations in neuron dynamics. The ideal transformations therefore only hold true when averaging over many neurons on a HICANN chip. Without calibration, individual neurons will most likely not behave as desired. Due to the large number of neurons present in the wafer-scale system, automatic and robust methods for characterization and calibration need to be developed.

This chapter introduces the setup and methods used to automatically characterize and calibrate neurons on a HICANN chip, generating data for the calibration backend. Each calibration method is described in detail, applied to all neurons on a HICANN chip and tested for its quality. Since the aim of this calibration is the reliable mapping of neural networks, a network using two calibrated HICANN chips is presented at the end of this chapter.

3.1 Basic Experiment Setup

To successfully characterize a complete neuron circuit, the dependencies of indiviual dynamics, e.g. membrane time constant, to the input parameters, e.g. leakage OTA bias current have to be measured. For an overview of all relations, see table 2.1. Since it is only possible to read analog membrane voltages and digital spikes from the neuron circuits, methods to extract neuron parameters from these measurements need to be developed. Many methods in this chapter are based on the work of *Schwartz* (2013). The main difference to this work however is the activation of the synaptic input term, i.e. setting the parameters I_{convi} and I_{convx} to their maximum value, instead of setting them to zero, which is essential for a neuron embedded in a network, but introduces many challenges to the calibration (see section 2.2.3).

To store calibration data and make it accessible for marocco, the calibration backend "calibtic" is used. Calibtic stores functions that take the ideal floating gate parameters and return calibrated floating gate parameters. Given a successful calibration, the returned parameters will be chosen such that the neuron behaves as intended by the ideal transformation, i.e. like the mean of all neurons for the desired floating gate value. In that way, homogeneity across neurons is increased.

3.2 Calibration software

To generate these functions in an automated and user-friendly way, the software framework "cake" was delevoped during the scope of this master thesis. Before running cake, the user can choose settings such as the target dynamic range on the hardware, the ranges for all calibrations, the HICANN coordinate and many more in a configuration file that is passed on to the software. To generate the calibration functions from hardware measurements, the software consecutively processes each neuron parameter in the following way.

First, a number of measurement steps are picked from the specified hardware parameter range. For each step, the floating gates are programmed to the desired values and voltage traces for all neurons are recorded. The measured voltage values are corrected for the readout error that is caused by the neuron readout amplifier (see section 2.2.4). The traces are then processed by an analyzer which extracts the important observables (hardware parameters) from the shape of the trace. After all measurements are finished, the measured hardware parameters are translated to corresponding floating gate settings via the ideal transformations. The ideal transformations for all potentials are $V_{fg} = V_{hw}$ since the voltages from the floating gates are directly applied to the circuit. The ideal transformations for time constants or conductances need to be extracted by taking the mean behaviour of all neurons on a chip or by simulating the neuron circuit on a transistor level (*Schwartz*, 2013). Finally, the transformed results are fitted with a polynomial function, where the measured floating gate values provide the x-data and the floating gate values that were set provide the y-data. The result of this fit is a function $\mathrm{V_{set}}(\mathrm{V_{measured}})$ describing the relationship between input parameter and measured behaviour of the neuron (see fig. 3.1). Using this function, the neuron can be set to a desired behaviour more precisely than by using only the ideal transformation. However, the precision of this function is limited since it is only a fit to a limited amount of data points. Increasing the number of measurement points would potentially improve the fit, but at the cost of total calibration time. Therefore, the number of measurement points is chosen such that the calibration is successful while taking as little time as possible.

The basic parameter settings for all calibration steps can be found in table 3.1. These settings ensure that the adaptation and exponential terms are turned off so that the neurons behave like LIF neurons. In each calibration, some of these settings are changed (see table 3.2) to create the desired membrane voltage traces. All measurements are done with the big capacitance setting and the speedup factor set to normal. The results presented in this chapter were obtained on a vertical setup equipped with one HICANNv2

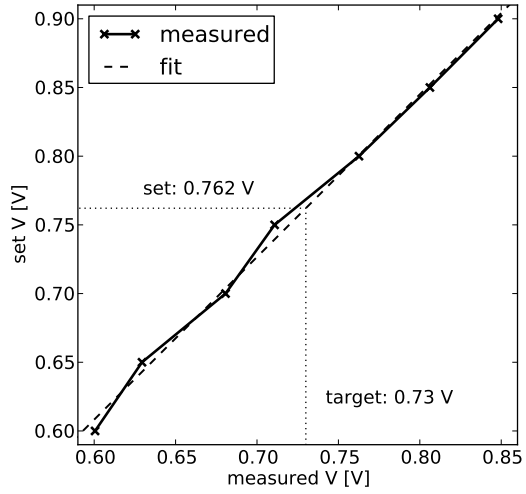

Figure 3.1 Plot of one measurement showing the basic principle of a calibration. Measurement steps for this measurement were chosen from 0.6 V to 0.9 V in steps of 50 mV (y-axis). A fit is done to obtain a function that describes the required parameter setting that yields a desired result (x-axis).

chip. The software works as well on the wafer-scale system without any adjustments.

3.3 Measure of calibration quality

To quantify the success of the calibration, the mean and standard deviation of a measured parameter across all neurons before and after calibration is examined. After an ideal calibration, the mean should equal the desired parameter and the standard deviation should be zero, so that all neurons behave in the desired way. Two effects lead to a nonzero standard deviation before calibration: the deviations caused by transistor mismatch, which will be systematic and constant for each trial, and the deviations caused by reprogramming the floating gates, which are a statistical error (see section 2.3). This leads to a total standard deviation of

$$\sigma_{total} = \sqrt{\sigma_n^2 + \sigma_t^2}, \tag{3.1}$$

where σ_n is the systematic neuron-to-neuron variability and σ_t the statistical trial-to-trial variability. Thus, the purely systematic neuron-to-neuron variation can be calculated when knowing the trial-to-trial variation:

$$\sigma_n = \sqrt{\sigma_{total}^2 - \sigma_t^2}. \tag{3.2}$$

The goal of the calibration is to minimize both σ_n and the distance of the mean measured value to the desired value, while σ_t cannot be changed since it is given by the quality of the floating gates. There are two ways to quantify σ_n from measurements. If σ_t is known, σ_n can be calculated from eq. 3.2. An easier way to obtain σ_n however is to average measurement results over many trials, effectively cancelling out trial-to-trial variations. The latter method is used to generate data for histograms in this chapter.

Table 3.1 Base parameters for all calibrations. These settings are used to emulate Leaky Integrate-and-Fire neurons. For the specific calibration routines, some of the parameters are changed to values from table 3.2.

Neuron Parameters							
E_l	900 mV	V_t	1000 mV	V_{synx}	1000 mV	$I_{spikeamp}$	2000 nA
E_{syni}	800 mV	I_{pl}	2000 nA	V_{syni}	1000 mV	I_{intbbi}	2000 nA
E_{synx}	1000 mV	I_{gl}	1000 nA	V_{syntci}	1420 mV	I_{intbbx}	2000 nA
I_{convi}	2500 nA	V_{exp}	1800 mV	V_{syntcx}	1420 mV	I_{radapt}	2500 nA
I_{convx}	2500 nA	I_{bexp}	2500 nA	I_{fire}	0 nA	I_{rexp}	2500 nA
$I_{gladapt}$	0 nA						
Shared Parameters							
V_{reset}	500 mV	I_{bstim}	2500 nA	V_{stdf}	0 mV	int_op_bias	2500 nA
V_{thigh}	0 mV	V_{gmax2}	80 mV	V_{dllres}	350 mV	V_{gmax3}	80 mV
V_m	0 mV	V_{bout}	750 nA	V_{tlow}	0 mV	V_{bstdf}	0 mV
V_{bexp}	2500 nA	V_{gmax0}	80 mV	V_{dtc}	0 mV	V_{fac}	0 nA
V_{clra}	0 mV	V_{br}	0 mV	I_{breset}	2500 nA	V_{clrc}	0 mV
V_{dep}	0 mV	V_{gmax1}	80 mV				

Table 3.2 Specific hardware settings for different parameter calibration routines (target parameter on top). The settings from table 3.1 are replaced by these specific parameters in order to induce the voltage traces needed for characterization.

V_{reset}		V_t		E_{synx}		E_{syni}	
E_l	1300 mV	E_l	1300 mV	E_l	900 mV	E_l	900 mV
V_t	900 mV	V_{reset}	400 mV	V_{reset}	200 mV	V_{reset}	200 mV
I_{convx}	0 nA	I_{convx}	0 nA	I_{convx}	2500 nA	I_{convx}	0 nA
I_{pl}	10 nA	I_{pl}	2000 nA	V_t	1300 mV	V_t	1300 mV
I_{convi}	0 nA	I_{gl}	1500 nA	I_{convi}	0 nA	I_{convi}	2500 nA
I_{gl}	1100 nA	I_{convi}	0 nA	I_{gl}	0 nA	I_{gl}	0 nA
E_l		I_{pl}		V_{syntci}		V_{syntcx}	
I_{gl}	calibrated	E_l	1200 mV	V_t	1200 mV	V_t	1200 mV
V_t	1200 mV	V_{reset}	500 mV	E_l	900 mV	E_l	900 mV
I_{convx}	2500 nA	V_t	800 mV	E_{syni}	800 mV	E_{syni}	800 mV
I_{convi}	2500 nA	I_{gl}	1000 nA	E_{synx}	1000 mV	E_{synx}	1000 mV

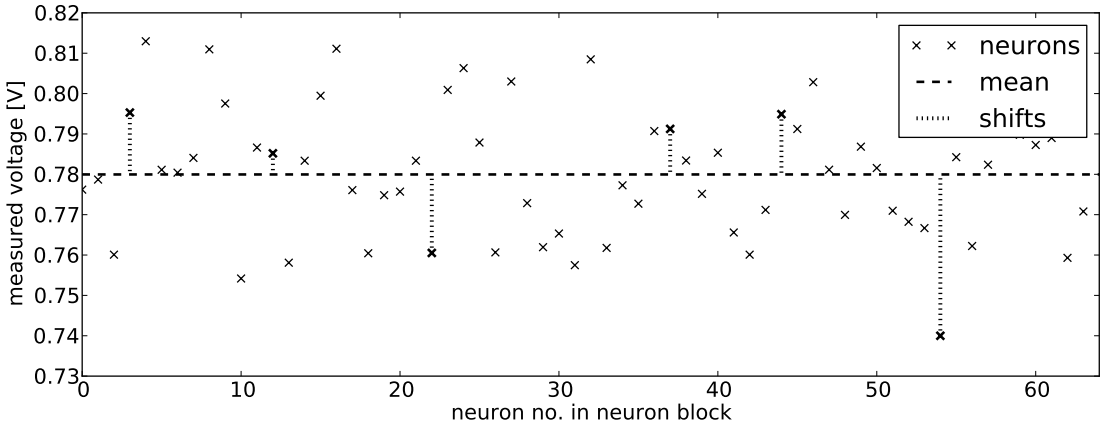

Figure 3.2 Measurement of V_{mem} across one block of 64 interconnected neurons. The readout shift of one neuron is the difference of measured V_{mem} (crosses) to the mean over the whole block (dashed line). The dotted lines illustrate the acquired readout shifts for 6 of the 64 neuron circuits.

3.4 Neuron Readout Buffer Offsets

To correctly measure all membrane voltages in the following calibration steps, the offsets of the neuron readout buffers (readout shifts) need to be measured first (see section 2.2.4). These offsets are systematic and constant for each neuron. A method to measure the readout buffer offsets is given in *Millner* (2012). In this section, it is explained in more detail and validated by a second measurement method.

Methods

One way to measure the readout shifts of all neuron output amplifiers is to ensure that a large number of neuron circuits are physically connected to one voltage line. If the number of neurons is sufficiently large and the individual readout buffer shifts are symetrically distributed around zero, the voltage on this line can be calculated by taking the mean over all neurons connected to the line. The difference between mean voltage and the measured voltage of an individual neuron yields the readout shift for that neuron (see fig. 3.2).

To achieve that many neurons share one voltage line, the 512 neurons on the chip are interconnected to form 8 large neuron blocks with a size of 64 neuron circuits each. This is the largest number of interconnected circuits possible (see chapter 2). Since the connection between neuron circuits has a very low resistance, the membrane voltages of all neurons that are interconnected should be the same. After measuring each neuron's individual membrane potential $V_{\mathrm{mem},i}$, a mean membrane voltage $\overline{V_{\mathrm{mem}}}$ over all neurons whithin a 64-size block can be calculated. The readout shift of each neuron is then calculated via

$$V_{\mathrm{shift},i} = V_{\mathrm{mem},i} - \overline{V_{\mathrm{mem}}}. \tag{3.3}$$

To validate this measurement and ensure that interconnected neurons indeed share one voltage, a second method to obtain shared voltages is applied. This method utilizes the fact that blocks of 128 neurons are connected to the same reset potential (see section 2.2.2). Again, by measuring the individual $V_{\text{reset},i}$ for all neurons across that block, a mean $\overline{V_{\text{reset}}}$ can be calculated for each block of 128 neurons. The reset shift of each neuron i is then calculated via

$$V_{\text{shift},i} = V_{\text{reset},i} - \overline{V_{\text{reset}}}. \tag{3.4}$$

The advantage here is that both measurements can be conducted independently, which is used to validate the assumptions made in each method (i.e. "interconnected neurons have the same membrane voltage"). Indeed, the neuron circuits might see slightly different reset potentials due to current mirrors that are placed between each membrane and the reset potential line. Using this method, we effectively measure the combined effects of V_{reset} variations and readout shifts. However, if the variation in V_{reset} across a block of neurons is not much larger than the spread in readout shifts, a strong correlation should be visible.

Results

The distribution of readout shifts for both methods is shown in figs. 3.3 and 3.4. By definition, the mean of the readout shifts for both methods is zero. The standard deviations of readout shifts are very similar, being $\sigma_M = 14.94\,\text{mV}$ and $\sigma_R = 15.05\,\text{mV}$ for membrane potential method and reset potential method respectively; fig. 3.5 shows that both measurement methods correlate very well with a Pearson-R of 0.968. In conclusion, the measurement of the readout shift via interconnected neurons is a viable method since it does not suffer from additional shifts. Therefore, all following calibation routines make use of the acquired readout shifts to correct measured voltages.

3.5 Reset Potential

Of all the neuron parameters described in this chapter, the reset potential is the only shared parameter (see section 2.2.2). Four different values of V_{reset} can be set on one HICANN, with 128 neurons sharing the same reset potential. The connection scheme of neurons to the floating gate blocks can be found in fig. 2.4. Being a shared parameter, V_{reset} cannot be calibrated for each neuron, but only across the four blocks.

Methods

Measurement of V_{reset} is done by setting E_l above V_t so that the neuron is constantly spiking. To facilitate visibility of the reset and give the membrane enough time to get pulled towards V_{reset}, the refractory period is set to a large value by choosing a low value for I_{pl}. The resulting value for V_{reset} is then obtained by an algorithm which extracts the baseline of the voltage trace in the following way: First, the times of sharp drops are

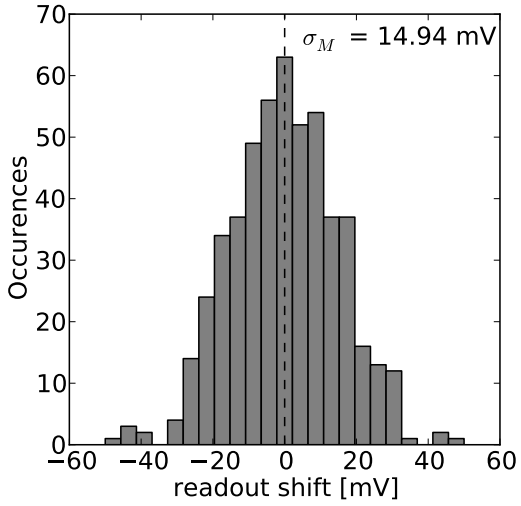

Figure 3.3 Distribution of readout shifts measured with the V_{mem} method.

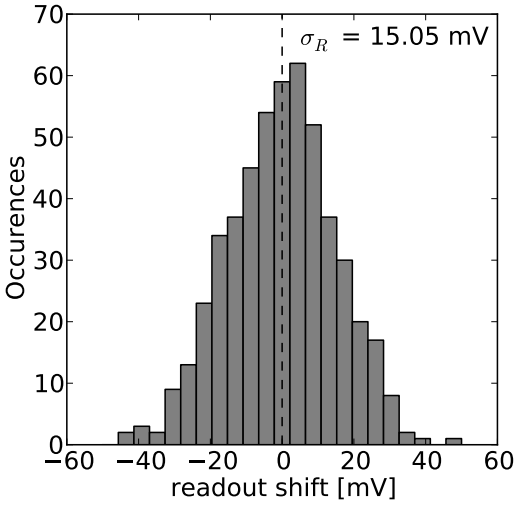

Figure 3.4 Distribution of readout shifts measured with the V_{reset} method.

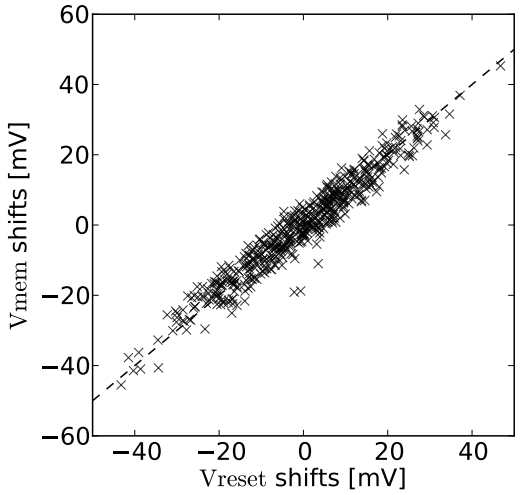

Figure 3.5 Correlation of readout shifts measured via V_{reset} method and via V_{mem} method.

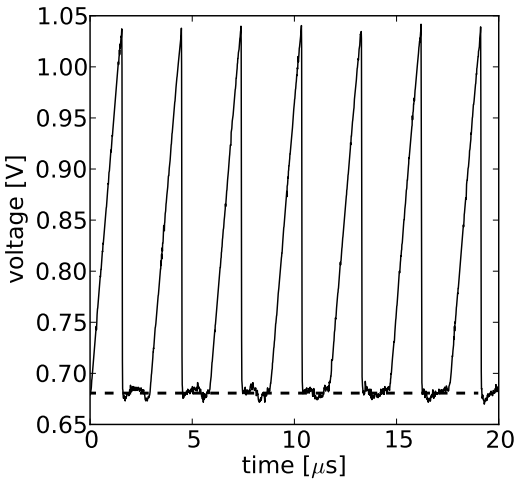

Figure 3.6 Typical membrane voltage trace for a measurement of V_{reset}. The dashed line shows the measured value.

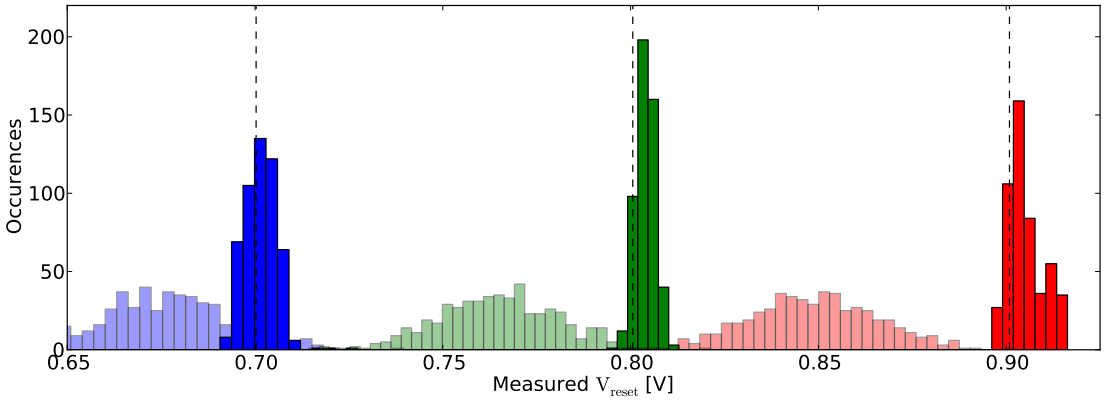

Figure 3.7 Distribution of measured V_{reset} before (faded) and after (solid) calibration for multiple target values (dashed lines). Different colors belong to different measurement steps.

found via the first derivative of the voltage trace. Then, the mean time it takes for the membrane to rise again is measured. For each spike that was obtained in the first step, the mean over the membrane voltage from the time of spike until the time the membrane rises again is taken. The returned baseline is the mean of all spikes that were found.

After measurement of V_{reset} across all neurons is done, the mean over all neurons that share the same floating gate block is calculated and used as measured V_{reset} for this floating gate block.

Results

A typical membrane voltage trace for these measurements can be seen in fig. 3.6. The choice of low I_{pl} increases the visibility of V_{reset} since the membrane is held at the reset potential for up to a few microseconds. The distributions of V_{reset} before and after calibration are shown in fig. 3.7. The neuron-to-neuron variation after calibration is decreased from around 8 % of the total dynamic range of 200 mV to only 1 %. An overview over the results is given in table 3.3.

3.6 Synaptic Reversal Potentials

The synaptic reversal potentaial of a conductance based synapse is the voltage towards which the membrane potential is pulled if the neuron receives a spike from another neuron. The total dynamic range for the membrane potential of a neuron is defined by its inhibitory and excitatory reversal potentials E_{syni} and E_{synx} since its membrane potential can never reach values that are lower than E_{syni} or larger than E_{synx}. In the synaptic input circuit, E_{syn} is connected to one of the inputs in OTA_0 (see fig. 2.5). Since both synaptic input circuits and their calibration routines are identical, only the method for measuring E_{synx} will be described.

Table 3.3 Results of the V_{reset} calibration. The leftmost column shows the target V_{reset} values, while all other columns show measured V_{reset} values. The neuron-to-neuron variation is reduced by up to a factor of six. The trial-to-trial variation after calibration is larger than the neuron-to-neuron variation.

	before calibration		after calibration		
target [mV]	mean [mV]	σ_n [mV]	mean [mV]	σ_n [mV]	mean σ_t [mV]
700	676.94	17.97	701.40	4.03	3.80
750	717.32	16.04	751.62	3.60	4.16
800	764.96	16.09	803.99	2.50	3.86
850	805.63	16.78	853.48	2.59	3.95
900	848.66	16.92	905.54	4.72	3.74

Methods

To measure E_{synx}, the membrane leakage conductance is set to the smallest value by setting the leakage OTA bias current I_{gl} to zero. Then, only the excitatory synaptic input circuit is activated by setting its synaptic OTA bias I_{convx} to the maximum value while setting I_{convi} to zero. With those settings, the synaptic input E_{synx} is the only source for currents onto the membrane. Due to the leakage currents from OTA_1 in the synaptic input circuit (see section 2.2.3), the conductance towards E_{synx} is non-zero. If this conductance, which usually leads to unwanted shifts in the resting potential, is large enough, the membrane is driven towards the synaptic reversal potential. That way, a flaw in the design of the synaptic input is exploited to serve as a measurement method for E_{synx}. The conductance towards E_{synx} is further increased by sending a very high rate of strong excitatory input spikes from the background generator to the neuron. Given that the conductance between membrane and reversal potential is sufficiently large, the membrane potential rests at E_{synx} (see eq. (2.10)). The reversal potential is then acquired by taking the mean over a membrane voltage trace (see fig. 3.8).

Results

The results of E_{synx} and E_{syni} calibration are shown in figs. 3.9 and 3.10. Detailed results are found in tables 3.4 and 3.5. The neuron-to-neuron variation of the excitatory reversal potential was lowered from 11 % to 6 %, while the variation of the inhibitory reversal potential was lowered from around 11 % to 2 % of the dynamic range.

However, this measurement method does not work on all neurons (see the right membrane voltage traces in fig. 3.8). There are several factors that can potentially disturb the success of this method. First, neurons could be connected to defect synapse drivers, not receiving any spikes at all. Second, insufficiently calibrated V_{syntc} values or broken synaptic input circuits can lead to incoming spikes not being properly forwarded to the membrane. Also, despite setting the bias currents I_{gl} and I_{convi} to zero, leakage currents towards E_l or the E_{syni} can still occur and disturb the membrane. Currently, neurons that

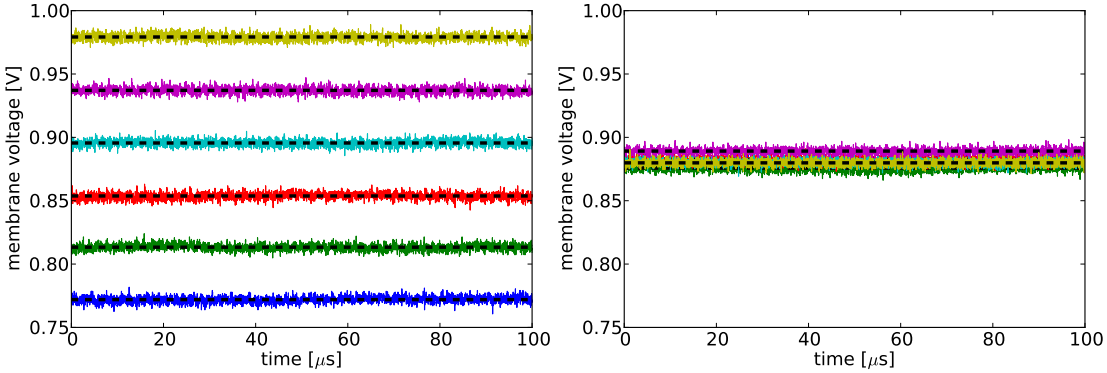

Figure 3.8 Typical membrane voltage traces for the measurement of E_{synx}. Different colors show different settings of E_{synx}, while the dashed lines show the measured mean value. The traces on the left are taken from a neuron that can be measured with this method, while the traces on the right are taken from a neuron where this measurement method does not work. Despite setting different values for E_{synx}, the membrane voltage does not change, but stays constant at the leakage potential E_l.

Table 3.4 Results of E_{synx} calibration. The leftmost column shows the target E_{synx} values, while all other columns show measured E_{synx} values. The arrow shows the value that will be chosen for experiments (upper end of the dynamic range, see section 2.4).

	before calibration		after calibration		
target [mV]	mean [mV]	σ_n [mV]	mean [mV]	σ_n [mV]	mean σ_t [mV]
850	798.98	21.65	848.78	4.41	6.33
900	838.88	21.26	900.93	6.05	5.15
950	878.97	21.29	952.62	8.77	4.97
$\longrightarrow$ 1000	919.50	22.42	1004.60	11.82	5.07
1050	962.56	23.37	1053.44	33.08	5.52

cannot be measured with this method are blacklisted by the algorithm and marked as broken. However, future calibration routines could implement alternative measurement methods for E_{syn} for neurons which cannot be measured with this method (see chapter 6). In addition, a synapse driver defect detection tool is currently in development. This tool can be used to systematically scan synapse drivers and mark broken drivers. Then, only the ones that are working well can be chosen for this method.

On the HICANN chip that is installed in the depicted vertical setup, the algorithm marks 30 of the 512 neurons as broken. However, this value depends strongly on the chip itself, where the depicted chip has a particularly large number of neurons marked as broken. On HICANN chips on the wafer, the number of neurons marked as broken is typically between 5 and 10.

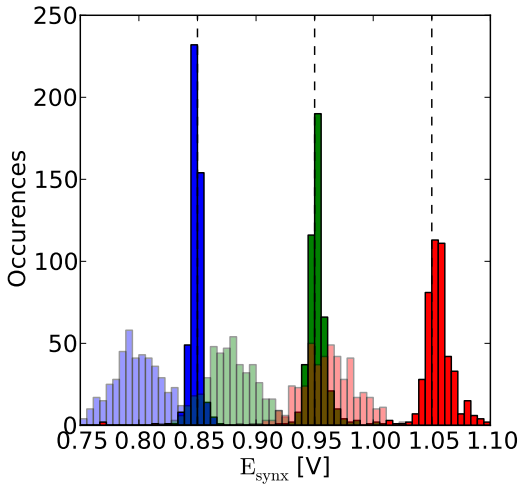

Figure 3.9 Results of E_{synx} calibration before (faded) and after (solid) calibration for different target values (dashed lines).

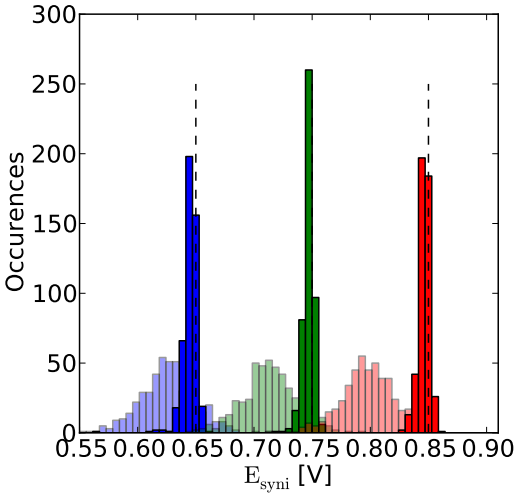

Figure 3.10 Results of E_{syni} calibration before (faded) and after (solid) calibration for different target values (dashed lines).

Table 3.5 Results of E_{syni} calibration. The leftmost column shows the target E_{syni} values, while all other columns show measured E_{syni} values. The arrow shows the value that will be chosen for experiments (lower end of the dynamic range, see section 2.4).

	before calibration		after calibration		
target [mV]	mean [mV]	σ_n [mV]	mean [mV]	σ_n [mV]	mean σ_t [mV]
650	625.92	23.98	644.98	6.72	6.33
700	670.56	24.03	696.74	5.66	5.15
750	712.76	23.39	746.82	4.53	4.97
$\longrightarrow$ 800	756.60	23.21	798.15	4.21	5.07
850	797.67	22.36	846.20	4.77	5.52

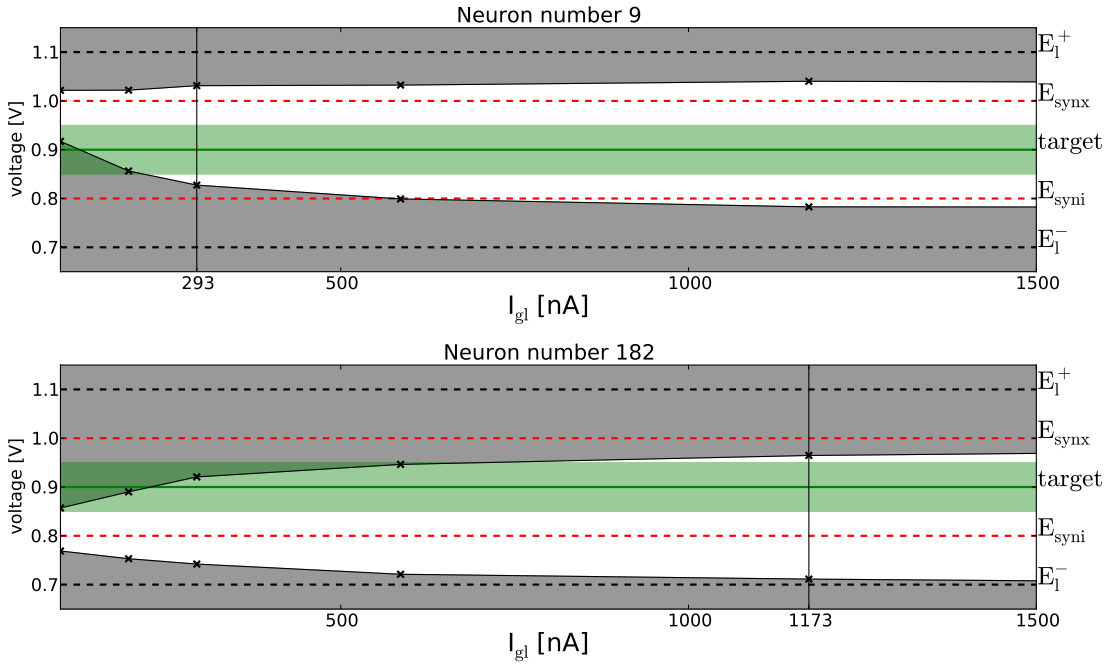

Figure 3.11 Illustration showing the principle of finding the lowest leakage current where a target of 900 mV can be still be reached. For each setting of I_{gl}, two measurements with different values for E_l are done. The two E_l values that are set in each step are shown as black dashed lines (E_l^+ = target + 200 mV and E_l^- = target − 200 mV). The red dashed lines show the configured values for E_{syni} and E_{synx}. The black crosses are the recorded resting potentials V_{rest}^+ and V_{rest}^- for both settings of E_l. The upper and lower resting potentials enclose an area which is marked as unshaded. This area is considered *reachable* in the sense that for any voltage V with $V_{rest}^- < V < V_{rest}^+$, there exists an E_l between E_l^+ and E_l^- that will result in $V_{rest} = V$. The unreachable area is shaded grey. The lowest measured I_{gl} value where the reachable area fully includes the green target area (target ± 50 mV) is chosen by the algorithm. In the upper plot, an I_{gl} of 293 nA is chosen, while in the lower plot, an I_{gl} of 1173 nA is chosen.

3.7 Leakage Conductance for spike impact maximization

To maximize the effect of incoming post-synpatic potentials (PSPs) on the neuron membrane, the leakage conductance g_l should be chosen as low as possible by decreasing the bias current I_{gl}. Due to leakage currents from the synaptic input circuit (see section 2.2.3), decreasing I_{gl} will strongly influence the resting potential. Choosing a value that is too low might lead to a loss of control over the resting potential. The target of this calibration method is to find the smallest I_{gl} for which a desired resting potential V_{rest} can still be reached. This is a very crude calibration method in a sense that only one value for I_{gl} is chosen for which a working neuron is obtained, restricting any choice over the membrane time constant. The advantage of this method however is that it is a fast and simple way to get neurons that are working well. An alternative method for calibration of membrane time constants is explained in section 4.2.

Methods

In order to find out if a desired V_{rest} can be reached with a given I_{gl}, two measurements with different values for E_l with the same I_{gl} are done (see fig. 3.11). In the first measurement, E_l is set 200 mV below the target V_{rest}, while it is set to 200 mV above the target V_{rest} in the second measurement. For each of the two measurements, the resulting resting potentials V^+_{rest} and V^-_{rest} are measured. If the desired V_{rest} lies between the two measured resting potentials, it is considered reachable. If both measured values are above or below the target, it is considered unreachable. The calibration routine applies this method for six values of I_{gl} to find the lowest I_{gl} where V_{rest} can still be reached. The target V_{rest} is set to the center of the dynamic range, i.e. 900 mV.

After an I_{gl} for each neuron is found, the resulting time constants can be measured with the method found in section 4.1.

Results

The resulting distribution of chosen I_{gl} values after calibration can be found in table 3.6. Setting the leakage conductance for each neuron to a constant value results in fixed membrane time constants. This resulting distribution of time constants after calibration is shown in fig. 3.12. The trial-to-trial variations depend on the absolute value of τ_{mem} for each neuron. Therefore, only the relative trial-to-trial variations of time constants are shown in figs. 3.13 and 3.14. Figure 3.14 shows that sixty percent of all neurons have a relative error of τ_{mem} of 5 % or lower.

3.8 Resting Potential

The membrane leakage potential E_l is used to control the resting potential V_{rest} of the neuron. It is connected to the membrane via the leakage OTA, which is driven by the bias current I_{gl} (see section 2.2.1).

Methods

Before measuring E_l, the bias current I_{gl} needs to be set to a constant value. It is important to keep the bias current constant during and after calibration. When I_{gl} values were identified with the method from section 3.7, it makes sense to choose those values. The synaptic reversal potentials are set to their target values (800 mV and 1000 mV) and also kept constant throughout the sweep of E_l. The synaptic inputs need to be turned on by setting $I_{convi} = I_{convx} = 2.5$ nA during calibration. All of these settings must be kept constant even after calibration, since any changes will strongly influence the resting potential of the neuron, rendering the calibration useless. When all parameters are set, the resulting resting potential V_{rest} is measured by taking the mean over a membrane trace in the absence of any (spike or current) input.

Table 3.6 Distribution of I_{gl} values after calibration for spike impact maximization. The I_{gl} steps were chosen such that the leakage conductances between steps are approximately equally spaced.

I_{gl} [nA]	no. of neurons
97	241
195	52
293	46
586	70
999	48
1173	37
2346	18

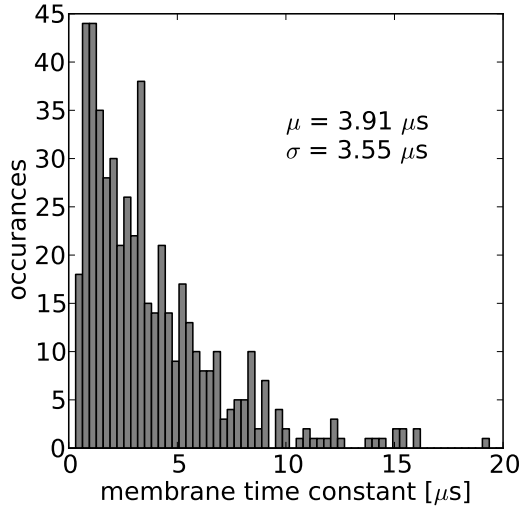

Figure 3.12 Distribution of membrane time constants after calibration for spike impact maximization.

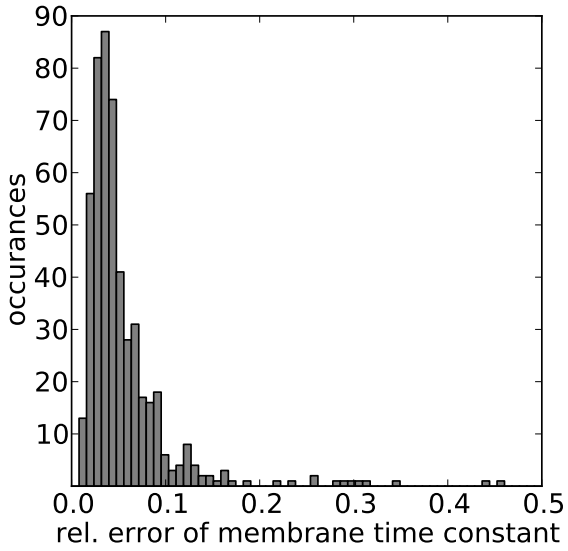

Figure 3.13 Distribution of relative trial-to-trial errors of membrane time constants after calibration for spike impact maximization.

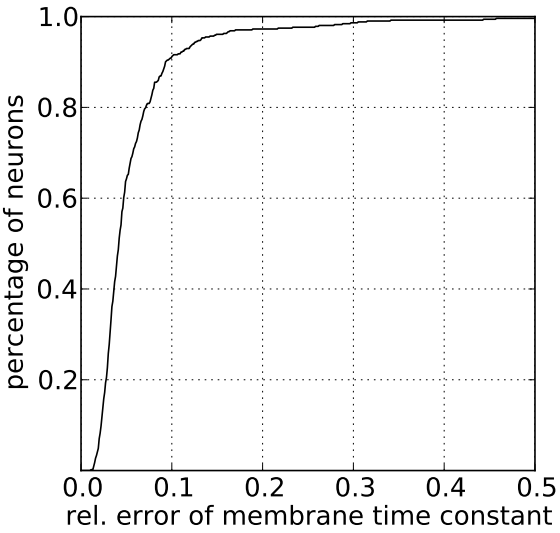

Figure 3.14 Cumulative relative trial-to-trial errors of membrane time constants after calibration for spike impact maximization.

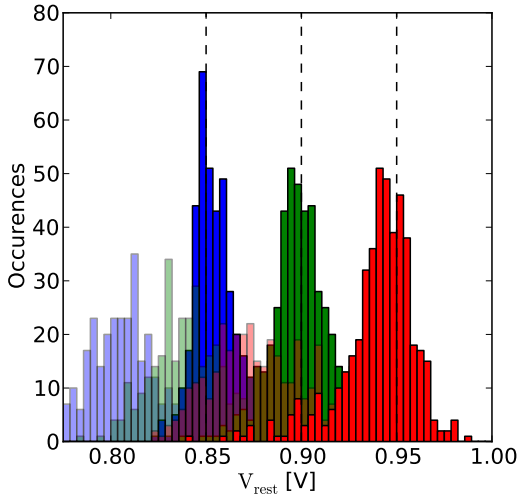

Figure 3.15 Results of E_l calibration before (faded) and after (solid) calibration for different target values (dashed lines).

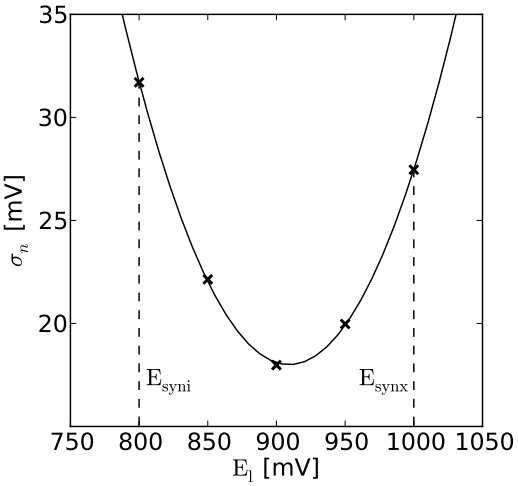

Figure 3.16 Neuron-to-neuron variability of V_{rest} as a function of the chosen target setting.

Results

The distribution of measured resting potentials before and after calibration for three different steps are shown in fig. 3.15. Since the strongest disturbances to the resting potential come from the synaptic input, the neuron-to-neuron variation depends on the value of E_l that is chosen (see fig. 3.16). Setting a value that is closer to one of the reversal potentials will result in some neurons being pulled strongly towards the opposite reversal potential (as seen in the red histogram in fig. 3.15). This effect increases the variability of V_{rest} at the borders of the dynamic range. When setting E_l to the center of the dynamic range, the effects of both synaptic inputs overlap, leading to a point of smallest neuron-to-neuron variation (see fig. 3.16). Summarized, the relative variation of resting potentials across neurons was reduced from 19 % to 9 % of the dynamic range of 200 mV.

3.9 Spike Threshold

The spike threshold V_t is used in the comparator of the spike detection circuit to trigger a spike if $V_{mem} > V_t$. It is one of the few parameters that are completely independent of any other parameters in the circuit.

Methods

Similar to the V_{reset} calibration method, the neuron is put into a state where it is constantly spiking by setting E_l above V_t (see table 3.2). However, in contrast to the V_{reset} measurement method, an increase of visibility via a refractory period is not possible.

Table 3.7 Results of the E_l calibration. The leftmost column shows the target V_{rest} values, while all other columns show measured V_{rest} values. Neuron-to-neuron variation is decreased by a factor of two at best. The trial-to-trial variations are significantly smaller than the neuron-to-neuron variations.

	before calibration		after calibration		
target [mV]	mean [mV]	σ_n [mV]	mean [mV]	σ_n [mV]	mean σ_t [mV]
750	768.61	50.16	795.67	46.12	8.24
800	800.66	44.41	828.87	31.65	6.33
850	832.59	40.26	862.39	22.16	5.15
900	863.43	37.15	900.90	17.96	4.97
950	893.89	34.90	939.94	19.95	5.07
1000	923.73	33.47	976.53	27.40	5.52
1050	956.21	33.39	1009.76	40.13	6.54

Table 3.8 Results of V_t calibration. The leftmost column shows the target V_t values, while all other columns show measured V_t values. The neuron-to-neuron variation is decreased to only 1 % of the dynamic range of 200 mV. After calibration, the trial-to-trial variation is larger than the neuron-to-neuron variation.

	before calibration		after calibration		
target [mV]	mean [mV]	σ_n [mV]	mean [mV]	σ_n [mV]	mean σ_t [mV]
800	787.65	10.35	801.24	2.80	3.71
850	830.75	9.72	850.33	2.57	3.41
900	873.81	10.34	900.14	2.13	3.37
950	917.49	9.83	949.20	2.01	3.25
1000	961.15	8.96	998.63	2.16	3.22
1050	1004.80	10.23	1050.36	2.81	3.33
1100	1049.75	10.09	1100.94	3.58	3.51

To obtain as many spikes as possible, the refractory period is set to a smaller value by increasing I_{pl}. A spike detection algorithm taken from (*Billauer*, 2012) and converted to Python is used to detect the heights at which each spike occurs.

Results

The results of the V_t calibration are shown in the histograms in fig. 3.17 and in table 3.8. Since measurement of this parameter is completely independent of all other parameters, σ_n is very low. The calibration led to a reduction from 5 % to 1 % of the dynamic range of 200 mV of 200 mV.

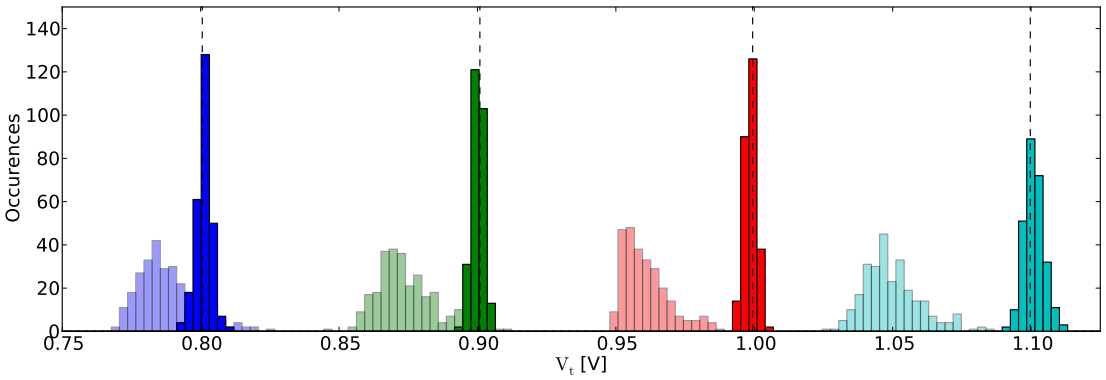

Figure 3.17 Histogram of measured V_t before (faded) and after (solid) calibration for four different target settings (dashed lines).

3.10 Synaptic Time Constants

The synaptic time constants of a neuron are controlled by the two voltages V_{syntcx} and V_{syntci}, where x denotes the excitatory and i the inhibitory synaptic time constant. Since circuits and calibration methods for both voltages are identical, only V_{syntcx} is described here.

Methods

The time constant of the synaptic input is proportional to the resistance R of the resistive element in the integrator (see section 2.2.3). R depends exponentially on V_{syntcx}, which makes precise control over the synaptic time constant very challenging. In addition, R changes by several orders of magnitude depending on the input voltage (*Kiene*, 2014). To find a useful point of operation for each neuron, this calibration method searches for the value of V_{syntcx} where PSP impact is maximized. The PSP impact is measured by periodically sending spikes to the sypnase and then taking the standard deviation σ of the trace. After recording the spike impact of different V_{syntcx} values, the one with the highest σ is considered to be the best value. This value is then stored in the database to be used for all experiments. This method results in fixed synaptic time constants for each neuron and is therefore only used as a preliminary calibration.

Results

Typically measured membrane voltage traces are shown in fig. 3.18. The calibration method chooses the trace with the highest impact, which happens at $V_{syntcx} = 1544\,\text{mV}$ for the depicted neuron. Increasing the parameter leads to a more rounded PSP shape, indicating a longer time constant, as expected. Similar behaviour can be observerd in almost every neuron (*Klähn*, 2013).

The measured dependency of membrane potential standard deviation on V_{syntc} is shown in figs. 3.19 and 3.20. For each neuron, the maximum value was chosen. The result-

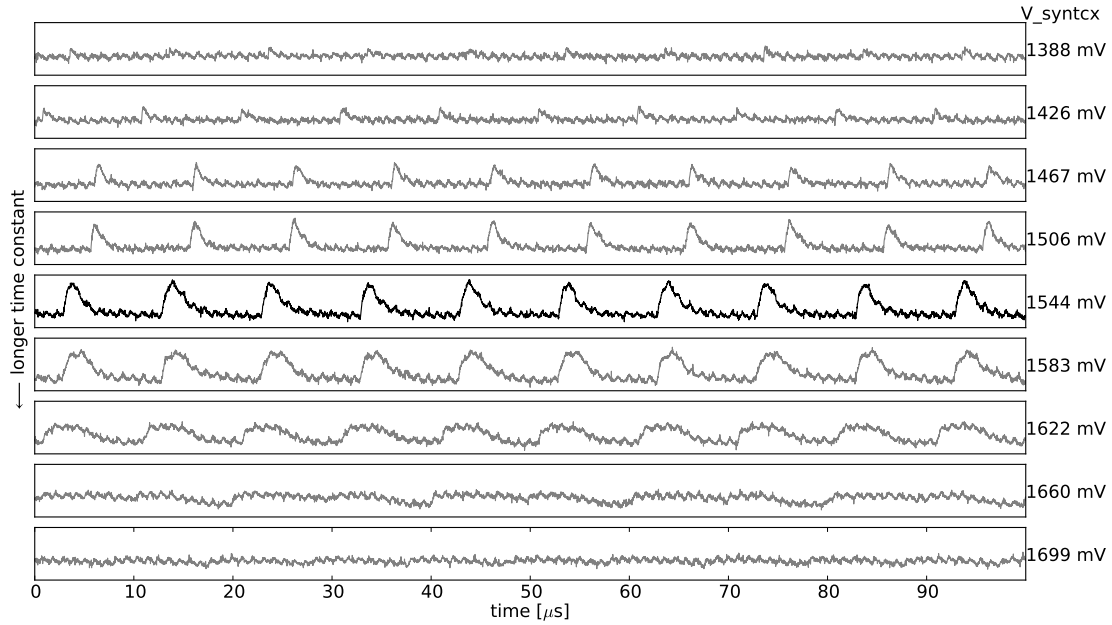

Figure 3.18 Membrane voltage traces of PSPs on one neuron with different values of V_{syntcx}. The trace with the highest standard deviation is shown in black, while the traces that were not chosen are shown in grey.

ing distributions of chosen V_{syntcx} and V_{syntci} values are shown in figs. 3.21 and 3.22. The calibration routine scans for V_{syntc} values in a range from 1300 mV to 1700 mV because values outside of this range do not result in any PSPs. The similarity of both histograms matches the expectation since the circuits for excitatory and inhibitory synaptic input are completely identical.

3.11 Refractory Period

The refractory period τ_{ref} of a neuron is the time it takes for the neuron to be excitable again after a spike. On the hardware, this time is controlled by the pulse current I_{pl}, where lower currents result in a longer refractory time.

Methods

The refractory period is measured by setting the neuron to a constant-spiking state, with parameters equal to those of the V_{reset} calibration. For different values of I_{pl}, a spike detection algorithm measures the interspike intervals (ISIs). Afterwards, the ISI with the smallest possible refractory period, ISI_0, is obtained by setting I_{pl} to the maximum possible value. The refractory periods for all other settings of I_{pl} are obtained by calculating

$$\tau_{ref}(I_{pl}) = ISI(I_{pl}) - ISI_0, \tag{3.5}$$

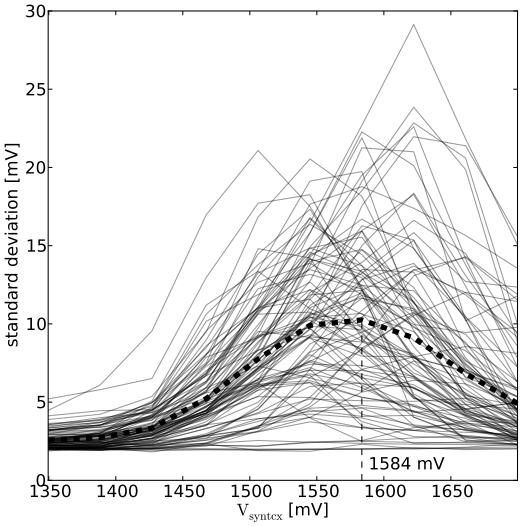

Figure 3.19 Measured membrane voltage standard deviations for some neurons (solid lines) and the mean standard deviation (dashed line) depending on V_{syntcx}. The maximum of the mean is marked (vertical dashed line).

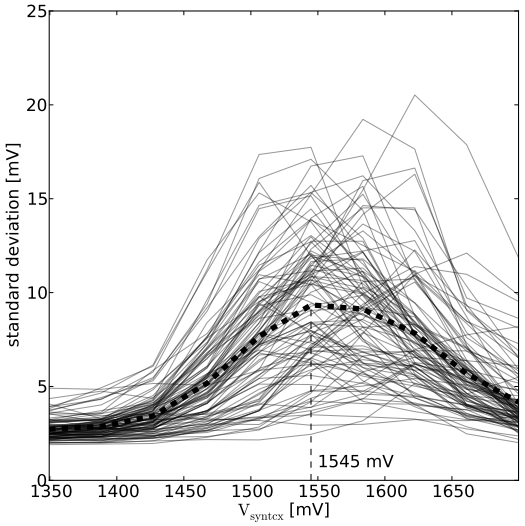

Figure 3.20 Measured membrane voltage standard deviations for some neurons (solid lines) and the mean standard deviation (dashed line) depending on V_{syntci}. The maximum of the mean is marked (vertical dashed line).

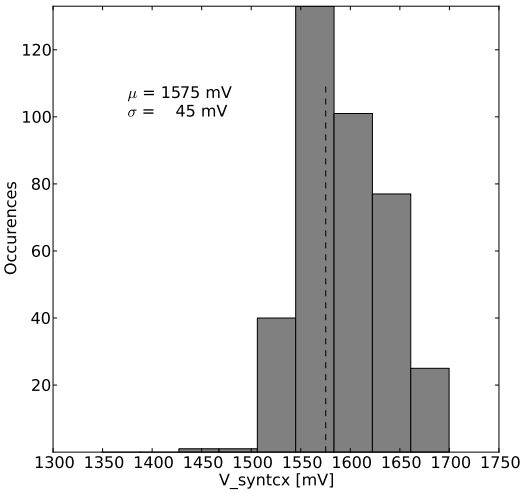

Figure 3.21 Distribution of chosen V_{syntcx} values across all neurons on one HICANN.

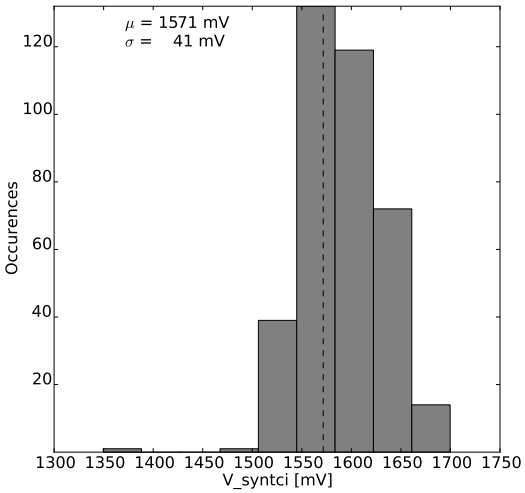

Figure 3.22 Distribution of chosen V_{syntci} values across all neurons on one HICANN.

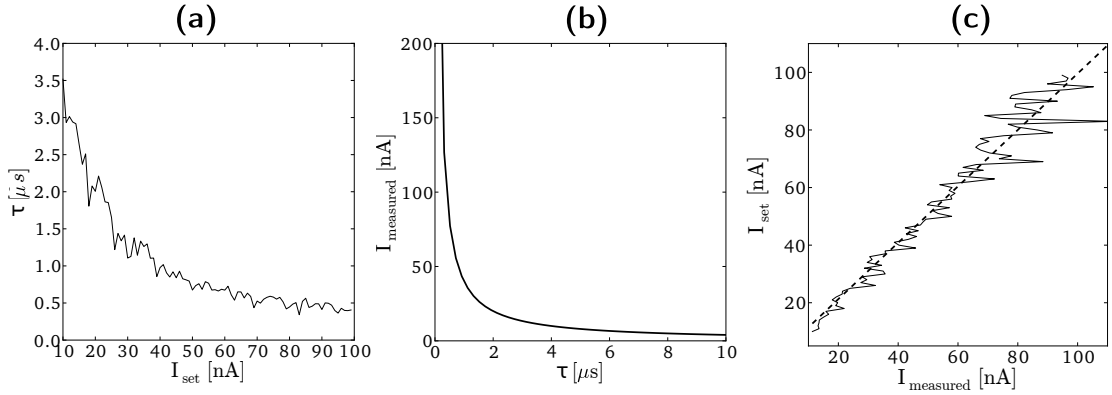

Figure 3.23 The three steps of obtaining a calibration for I_{pl}. First, τ_{ref} is measured for different configured values of I_{pl} (a). The measured refractory periods are then transformed via the ideal transformation (b) from eq. (3.6) and x- and y-axes are swapped. Finally, the curve is fitted with a straight line (c). The dashed line represents the fitted function that is stored in the calibration backend calibtic.

per definition leading to a τ_{ref} of zero for the maximum I_{pl}. The obtained refractory periods are transformed to their corresponding ideal floating gate parameter values (see fig. 3.23) via the ideal transformation taken from *Schwartz* (2013):

$$I_{pl} = \frac{1}{0.025 \frac{1}{\mu s \mu A} \cdot \tau_{ref} + 0.0004 \frac{1}{\mu A}}. \tag{3.6}$$

This transformation ensures that both x- and y-data for the fit have the same unit µA and the resulting function complies with the design of the calibration backend calibtic (see section 3.2).

Results

The calibration results are summarized in table 3.9. Histograms for three settings of I_{pl} are shown in fig. 3.24. The calibration reduces the neuron-to-neuron variability by up to a factor of four. Since long refractory periods can only be induced with very low settings for I_{pl}, floating gate variations lead to large relative σ_t.

Table 3.9 Results of the I_{pl} calibration. The two leftmost columns show the I_{pl} values that were set and their ideal transformations (target), while the other columns show measured values and errors. The calibration decreases neuron-to-neuron variations by up to a factor of four. For long refractory periods, the trial-to-trial variations are very large.

		before calibration		after calibration		
I_{pl} [nA]	target [µs]	mean [µs]	σ_n [µs]	mean [µs]	σ_n [µs]	mean σ_t [µs]
10	3.984	3.798	3.196	2.394	1.103	0.459
20	1.984	2.628	2.262	1.782	0.583	0.306
30	1.317	1.368	0.967	1.277	0.294	0.188
40	0.984	0.923	0.600	0.959	0.175	0.124
50	0.784	0.696	0.449	0.757	0.120	0.082
60	0.651	0.536	0.330	0.607	0.083	0.061
70	0.555	0.472	0.273	0.514	0.068	0.050
80	0.484	0.394	0.237	0.449	0.060	0.041
90	0.428	0.359	0.207	0.395	0.050	0.037
100	0.384	0.319	0.187	0.364	0.047	0.035
200	0.184	0.156	0.082	0.177	0.034	0.030
500	0.064	0.057	0.049	0.059	0.023	0.030
1000	0.024	0.026	0.041	0.020	0.019	0.031
1500	0.011	0.007	0.040	0.010	0.020	0.028
2000	0.004	0.014	0.045	0.004	0.019	0.029

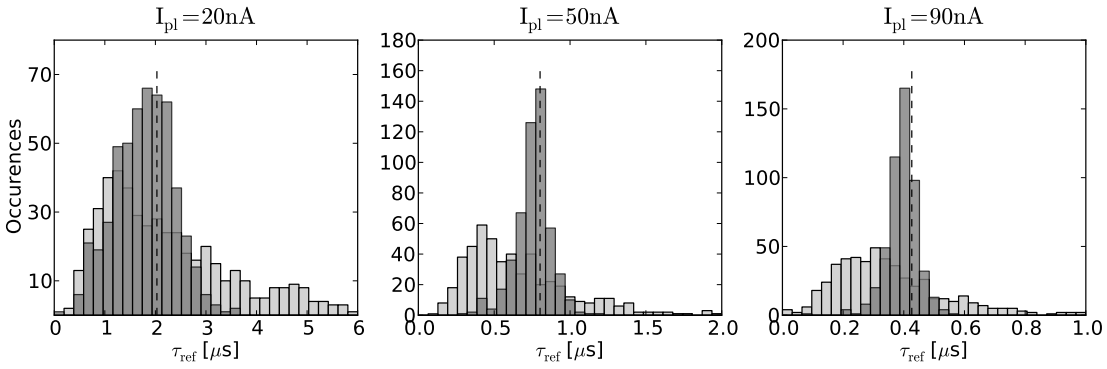

Figure 3.24 Result of the I_{pl} calibration for three different values of I_{pl}. Results before calibration are shown in light grey, results after calibration are shown in dark grey.

3.12 Interconnecting Neurons

When mapping a large neural network to the hardware, the number of maximum inputs for single neuron circuits is not sufficient to support routing without a severe loss of synaptic connections. To increase the maximum number of possible inputs per neuron, the HICANN chip supports the interconnection of neuron circuits. The mapping and routing tool marocco interconnects neurons by default, so all network experiments on the HICANN chip will make use of this feature. It is therefore important to measure the effects of interconnecting neuron circuits on the outcome of the calibration. To examine these effects, the distributions of all parameters after calibration were measured for blocks of 16 interconnected neurons.

Results for V_{reset}, V_t and E_l calibration can be found in tables 3.10 to 3.12. The variations in V_{reset} and V_t are not significantly changed by the interconnection. A possible explanation is that for each neuron block, only the spike dectection and reset circuit of one of the neurons in the block is activated. The distribution of reset and threshold potentials should therefore be a subset of the distribution over all neurons. The neuron-to-neuron variation in V_{rest} is reduced by a factor of up to four when inteconnecting neurons, suggesting that variations in resting potentials are averaged out over all neurons in a block. A significant reduction in neuron-to-neuron variation was also examined in the membrane time constants after calibration. With interconnected neuron blocks, the mean membrane time constant was changed from 3.91 µs for single neuron circuits to 1.6 µs for neuron blocks. The neuron-to-neuron variation was reduced by a factor of 8 from $\sigma_n = 3.55\,\mu s$ to $\sigma_n = 0.44\,\mu s$.While the reduced neuron-to-neuron variation can be explained by the interconnection, the large change in mean membrane time constant is unexpected. A possible reason for this change is the measurement method that is used to acquire the time constant. As explained in section 4.1, measuring the time constant with the current stimulus generator adds a constant capacitance of about 600 fF to the neuron. For single neuron circuits, this amounts for up to 20 % to 30 % of the total membrane capacitance. Since the membrane capacitance of an interconnected neuron of size 16 is 16 times the capacitance of a single neuron, the additional capacitance of the stimulus generator only amounts to 1.5 % of the total capacitance. This leads to a reduction of the disturbance caused by the stimulus generator, effectively decreasing the measured time constant.

An examination of the remaining LIF parameters $E_{syn<i,x>}$ and $V_{syntc<i,x>}$ for the whole block is not meaningful since each neuron circuit in the block has its own synaptic input term. The neuron block will therefore have many different synaptic reversal potentials and time constants, depending on which neuron circuit a spike arrives at. In conclusion, the measurement results suggest that the interconnection of neurons does not interfere with the outcome of the calibration, even if the calibration was done with single neuron circuits.

Table 3.10 Results of the V_{reset} measurements for interconnected neuron blocks of size 16 after calibration with single neurons. Compared to table 3.3, the values do not differ much.

target [mV]	mean [mV]	σ_n [mV]	mean σ_t [mV]
700	702.58	2.60	3.38
750	754.51	2.48	3.07
800	804.68	2.58	1.89
850	854.07	2.86	1.65
900	906.90	2.98	3.11

Table 3.11 Results of V_t measurements for interconnected neuron blocks of size 16 after calibration with single neurons. The values are similar to those of single neuron circuits found in table 3.8.

target [mV]	mean [mV]	σ_n [mV]	mean σ_t [mV]
800	801.35	2.58	3.03
850	850.31	2.36	3.53
900	900.03	2.90	3.49
950	949.66	2.68	3.09
1000	998.45	2.68	3.21
1050	1050.34	2.75	3.46
1100	1099.59	3.73	3.35

Table 3.12 Results of E_l measurements for interconnected neuron blocks of size 16 after calibration with single neurons. After interconnection, the neuron-to-neuron variation is much smaller than the variations for single neuron circuits found in table 3.7.

target [mV]	mean [mV]	σ_n [mV]	mean σ_t [mV]
800	817.03	6.74	1.48
850	856.26	4.29	1.23
900	899.26	4.00	1.12
950	941.31	4.96	1.29
1000	980.89	7.43	1.28
1050	1017.30	12.39	1.60

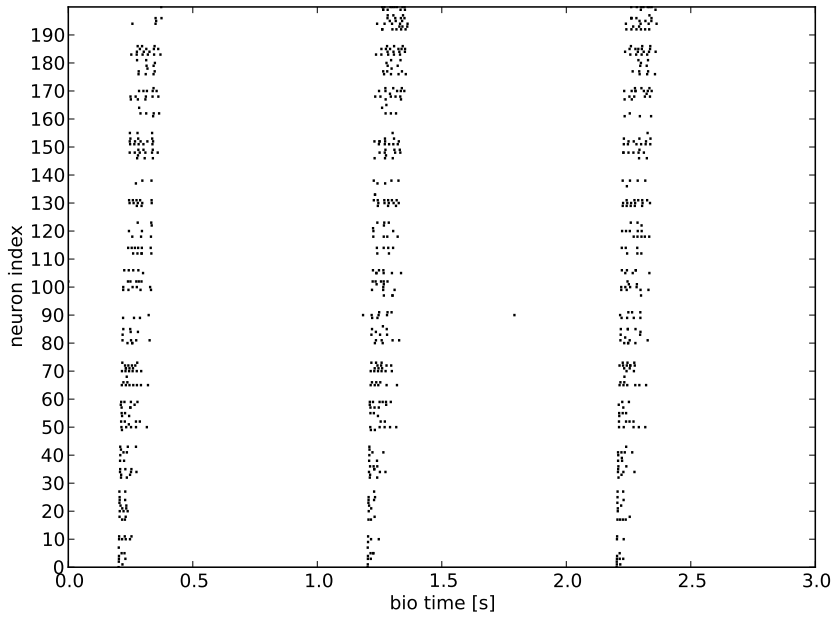

Figure 3.25 Rasterplot of a feed-forward neural network emulation with 200 neurons across two calibrated HICANN chips. Spikes are sent to population 0 (located on the lower end) in one second intervals and propagate upwards through the chain.

3.13 Neuron network experiment

To show that the goal of the calibration is achieved, a neural network emulation was set up by *Sebastian Schmitt* and *Paul Müller*. Since this experiment was not done by myself, the setup and results are only briefly described here.

The neural network is a feed-forward network of 200 neurons across two calibrated HICANN chips on one reticle of the wafer. The neurons are built by interconnecting four neuron circuits to increase the input count. Each link of the chain consists of a population of 12 neurons with excitatory connections to the next link in the chain. At one second intervals, spike inputs are sent to population number 0 and propagate upwards to the end of the chain.

The whole network description is written in PyNN and translated to a hardware network via the mapping and routing tool marocco (*Jeltsch*, 2014). This means that neurons are arbitrarily chosen, in contrast to previous experiments, where neurons were carefully picked, tuned and manually connected.

Figure 3.25 shows a raster plot of the neuron response. The spike signal indeed propagates from neuron number 0 further up the chain. When running the same experiment with an uncalibrated system, no response at all is visible, showing that calibration is essential for the reliable setup of large neural networks.

4 Advanced Characterization Methods

The previous chapter showed calibration methods that successfully allow basic neural network experiments. For the experiment that was presented, only the most vital neuron characteristics were calibrated. However, other neuron characteristics were not measured. This chapter gives an overview over some advanced characterization methods, especially concerning the membrane time constant. First, the membrane time constants on the chip are examined and possible ranges that can be set are measured. After the time constants are sufficiently characterized, a preliminary calibration method is presented that shows an alternative approach to the I_{gl} calibration shown in chapter 3. Finally, a method to increase the accuracy of modeling hardware neuron dynamics is introduced.

4.1 Characterization of the Membrane Time Constant

To achieve an emulation of biological membrane time constants from 1 ms to 100 ms, different approaches are combined in the HICANN chip. Primarily, the membrane time constant τ_{mem} scales inversely with the leakage bias current I_{gl}. This scaling alone is not sufficient to cover a range of τ_{mem} over two orders of magnitude without leading to large trial-to-trial variations for long τ_{mem}. To enable the large range of time constants, there are two mechanisms that further scale the time constant. First, the so-called speedup setting can be set to the three different values slow, normal or fast. This setting is used to configure a current mirror that divides the current I_{gl} by a certain factor s before it arrives at the leakage OTA. The fast setting simply mirrors the current without any changes and therefor has a factor of $s_{fast} = 1$. The normal and slow settings correspond to the dividers $s_{norm} = 3$ and $s_{slow} = 27$, lowering the total bias current arriving at the leakage OTA. In addition to the speedup factor, the size of the membrane capacitor can be set to either 2.16 pF or 0.16 pF, theoretically changing the time constant by a factor of 13. In this section, an analysis on the effectiveness of this approach is done by introducing and applying methods to measure τ_{mem} and the total possible hardware range.

Methods

Extraction of the membrane time constant τ_{mem} requires a free decay of the membrane voltage to its resting potential. For this purpose, the injection of a step current with strength I is used to raise the membrane voltage by a height V_I. After the input is turned off, the voltage decays exponentially back to the resting potential:

$$V_{mem}(t) = V_{rest} + V_I \cdot e^{-t/\tau_m}. \tag{4.1}$$

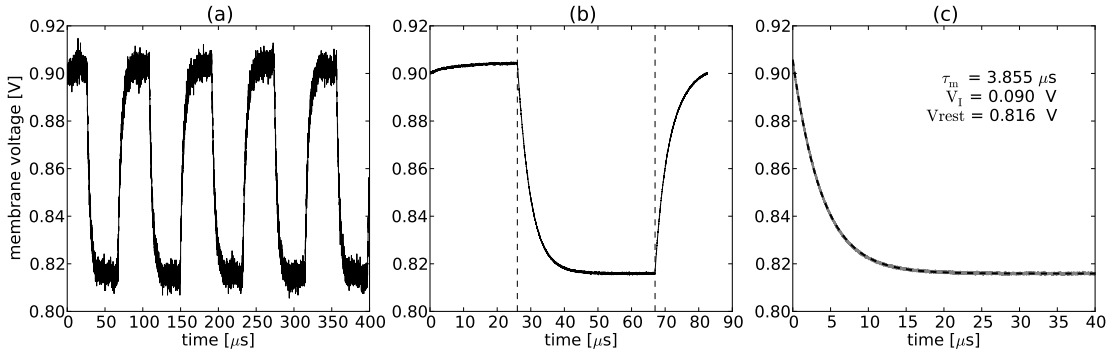

Figure 4.1 The principle of measuring the membrane time constant. (a) A large number of free decays of the membrane potential is induced via step currents. (b) The repeating pattern is averaged (solid line) and a fitting range (dashed lines) is found by a Sobel Filter (*Sobel and Feldman*, 1968). (c) A least squares fit of an exponential decay (see eq. (4.1)) is applied to the data to extract the membrane time constant. The fit (dashed line) perfectly covers the data (solid grey line).

To improve the quality of the data, many repetitions of the free decay are measured and averaged (see fig. 4.1). The range of data that is used for the fit is acquired by a Sobel Filter (*Sobel and Feldman*, 1968) that detects the edges in the voltage trace. A least squares fit method of eq. (4.1) to the averaged data extracts the time constant as well as the resting potential. The initial parameters for the fit are taken directly from the pre-processed voltage trace (fig. 4.1 c), with the last point in the trace as initial V_{rest} and the time of $\frac{1}{e} \cdot V_I$ elevation as initial τ_{mem}.

This method can only be applied when the membrane voltage does not rise more than 100 mV above V_{rest}. As soon as the voltage difference gets larger, an exponential function no longer accurately describes the behaviour and the fit will yield unsatisfactory results (see appendix, fig. A.4). To avoid voltages that are too high, the measurement method searches for a current that will yield a large enough membrane elevation while keeping the differential voltage below 100 mV.

Since this method uses the current stimulation to generate a signal, the total capacitance of the neuron is increased by the line capacitance of the current stimulation line (see section 4.3). The time constant that is measured with a current stimulus is therefore larger than the actual time constant that the neuron will have when the stimulus line is disconnected. The line capacitance is estimated to be about 600 fF (*Millner*, 2012).

To determine the maximum possible hardware range, measurements of the time constant for all speedup settings (fast, normal and slow) are conducted. A quantity of interest is the number of neurons that can be set to certain time constant. To get this number, each neuron's individual potential range is obtained by measuring its membrane time constants for 10 different settings of I_{gl}. The highest and lowest measured τ_{mem} are extracted from these measurements and taken as the upper and lower boundary for that neuron's membrane time constant range. This method is applied for all three speedup

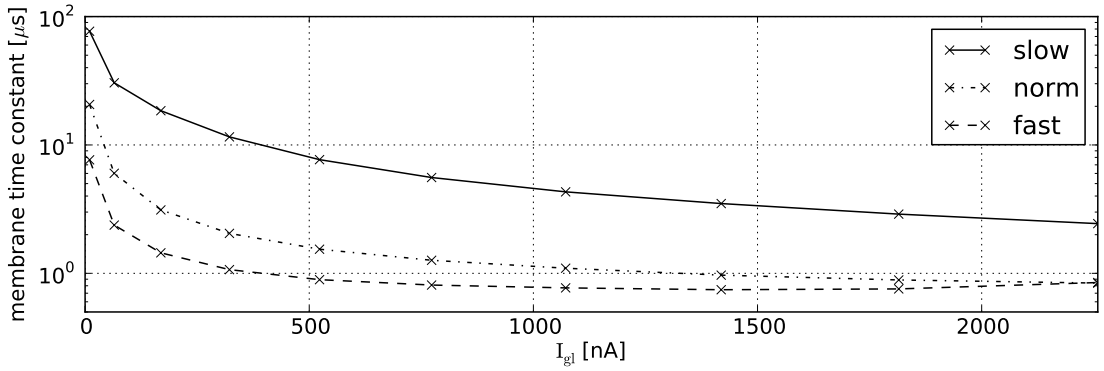

Figure 4.2 Measurement of mean time constants for different settings of I_{gl} and all three speedup settings with the capacitance set to the bigcap setting. The y-axis is plotted logarithmically since the time constants span a range of two orders of magnitude.

settings and for all neurons. The fraction of neurons that can reach a certain time constant is then calculated by counting how many neurons include this time constant in their range and dividing by the total number of neurons on the chip. Since this method will be used to obtain the potential possible hardware range, disturbances by unwanted leakage currents are turned off by setting both I_{conv} bias currents to zero.

Results

The mean time constants for different settings of the speedup factor and the bias current I_{gl} for the bigcap setting is shown in fig. 4.2. As seen in the plot, the time constants range from 1 µs to 100 µs, covering two orders of magnitude. Interestingly, the time constant does not fall monotonically for the fast setting. For I_{gl} values above 1.5 µA, the time constant rises again. Reasons for this behaviour are currently unknown. Simulations of the neuron circuit on a transistor level could give insights into this phenomenon.

The same plot for the smallcap setting is found in fig. 4.5. While the overall time constants are indeed smaller than for the bigcap setting, the theoretical decrease by a factor of 13 is not observed. In fact, the time constants are only decreased by a factor of up to 3. A possible explanation for this is that the true membrane capacitances of the neuron are not as specified. Measurements on the total membrane capacitance of the neuron are shown in section 4.3.

The results on the possible hardware range for the bigcap setting are shown in figs. 4.3 and 4.4. The slow setting yields best results for long time constants, while the percentage of neurons drops for time constants below 4 µs. Normal and fast settings do not deviate much for very short time constants below 1 µs. This is also in agreement with the observation that the time constant for the fast setting does not decrease monotonically with higher I_{gl}, emphazising the need for further simulations of the circuit.

Results on the possible hardware range for the smallcap setting can be found in figs. 4.6 and 4.7.

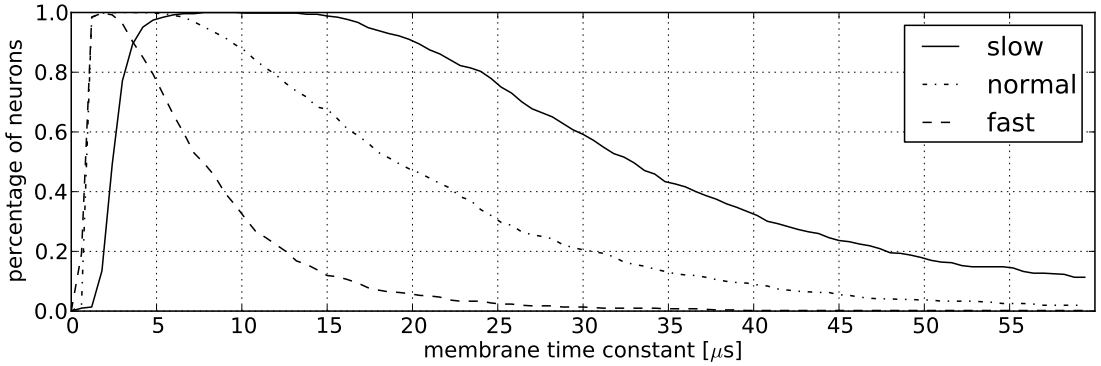

Figure 4.3 Measurement of the possible hardware range for τ_{mem} with the capacitance set to the bigcap setting. The percentage of neurons that can potentially reach a certain membrane time constant is plotted for the three different speedup settings. As expected, for slower settings, the long time constants can be reached by more neurons. Assuming that a reliable setup is given where more than 98 % of neurons behave as desired, the total range of time constants is between 1 µs and 6 µs for the normal setting and between 5 µs and 15 µs for the slow setting.

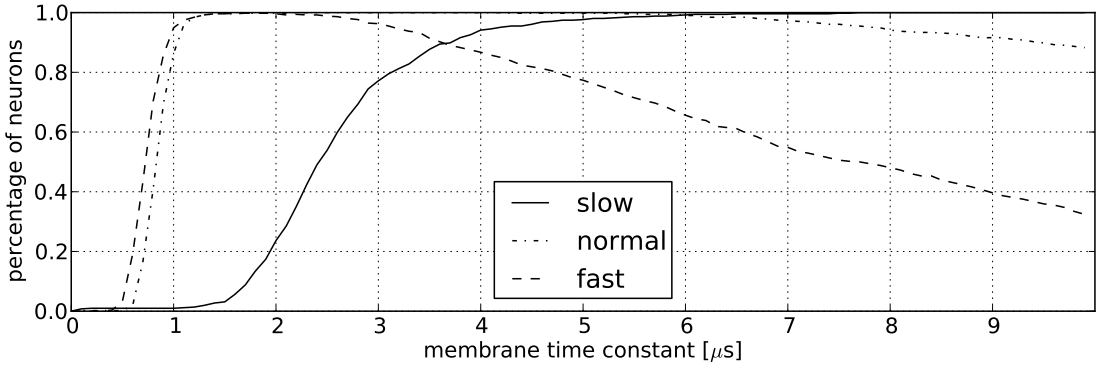

Figure 4.4 The plot from fig. 4.3 for time constants up to 10 µs. The normal and fast speedup settings do not differ much for time constants below 2 µs, while the normal setting yields better results for time constants above 2 µs.

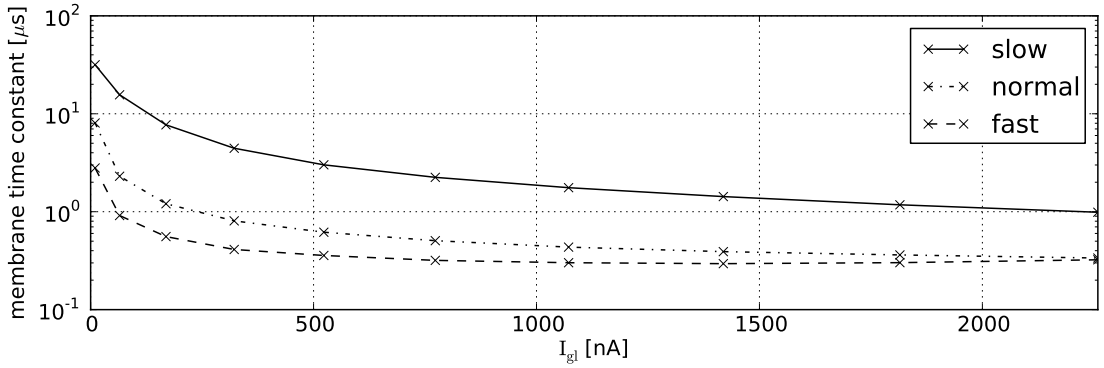

Figure 4.5 Measurement of mean time constants for different settings of I_{gl} and all three speedup settings with the small capacitance setting.

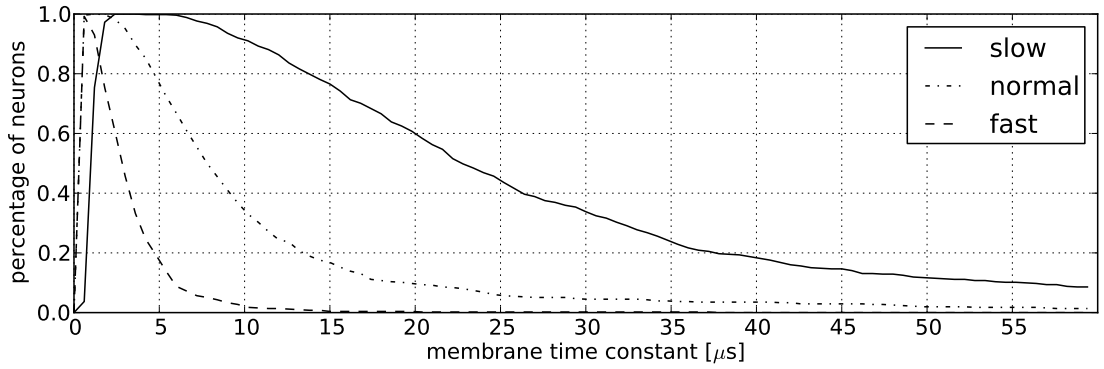

Figure 4.6 Measurement of the possible hardware range for τ_{mem} in the smallcap setting. The percentage of neurons that can potentially reach a certain membrane time constant is plotted for the three different speedup settings.

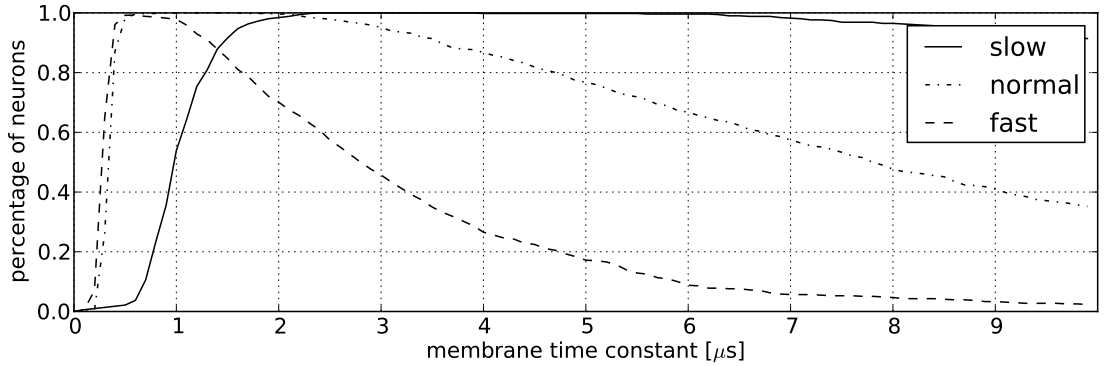

Figure 4.7 The plot from fig. 4.6 for time constants up to 10 µs. With the smallcap setting, the membrane time constant can be set as low as 0.5 µs.

For all results presented here, it has to be noted that the time constants measured with this method are distorted since the measurement uses the current stimulation. In *Millner* (2012) it is estimated that connecting the current stimulus adds about 600 fF to the total capacitance of the neuron. It can be assumed that the time constant scales linearly with the neuron's membrane capacitance (see section 4.3). Assuming the capacitance of the current generator to be 600 fF assuming the total membrane capacitances from section 4.3, the actual time constants without the current stimulus connected can be estimated to be up to 20 % smaller.

4.2 Calibration of the Membrane Time Constant

The calibration of I_{gl} in section 3.7 puts the neuron in a state where the impact of single incoming spikes on the membrane is maximized. The downside of that method is that a fix I_{gl} for each neuron is obtained, restricting any freedom over the membrane time constant. An alternative approach for calibrating I_{gl} is similar to the approach for I_{pl} found in the previous section.

Method

To obtain calibration functions for I_{gl} in the same way as described in section 3.11 for I_{pl}, an ideal transformation needs to be defined (see section 3.2). The ideal transformation for I_{pl} was directly taken from *Schwartz* (2013). However, it was not possible to reconstruct and confirm the ideal transformation that was given for I_{gl} in *Schwartz* (2013). Therefore, it has to be extracted from measurements. The transformation is acquired by measuring the membrane time constant for all neurons on a HICANN chip and for different settings of I_{gl} (see section 4.1). The measured time constants were averaged over all neurons to extract the ideal behaviour. The function that is fitted to the ideal behaviour was suggested in *Schwartz* (2013):

$$I_{\mathrm{gl}} = a \cdot \left(\frac{1}{\tau_{\mathrm{mem}}}\right)^2 + b \cdot \frac{1}{\tau_{\mathrm{mem}}} \tag{4.2}$$

Results

The results for the I_{gl} calibration are preliminary and only available for the normal speedup setting with the big capacitance. The ideal transformation that was obtained by fitting eq. (4.2) to the mean τ_{mem} over all neurons is

$$I_{\mathrm{gl}} = 607.61\,\mu\mathrm{A}\mu\mathrm{s}^2 \cdot \left(\frac{1}{\tau_{\mathrm{mem}}}\right)^2 - 202.3\,\mu\mathrm{A}\,\mu\mathrm{s} \cdot \frac{1}{\tau_{\mathrm{mem}}} \tag{4.3}$$

The results of the I_{gl} calibration are summarized in table 4.1. Since the calibration needs an ideal transformation, i.e. a transformation that approximately holds true for most neurons on the chip, a range needs to be chosen where most neurons can operate in. This leads to a reduction of range especially for neurons that can be set to long time

Table 4.1 Results of the I_{gl} calibration. The two leftmost columns show the I_{gl} values that were set and their ideal transformations (target), while the other columns show measured values and errors.

		before calibration		after calibration		
I_{gl} [nA]	target [µs]	mean [µs]	σ_n [µs]	mean [µs]	σ_n [µs]	mean σ_t [µs]
10	2.66	7.42	9.00	2.70	0.96	0.12
65	1.87	3.61	3.22	1.86	0.34	0.06
169	1.39	2.17	1.24	1.35	0.15	0.04
322	1.09	1.56	0.67	1.06	0.09	0.03
523	0.90	1.24	0.42	0.90	0.08	0.02
773	0.77	1.05	0.29	0.79	0.07	0.02
1072	0.66	0.93	0.22	0.73	0.08	0.02
1419	0.59	0.85	0.17	0.71	0.16	0.02
1815	0.53	0.81	0.15	0.73	0.20	0.02
2259	0.48	1.39	0.02	0.82	0.24	0.02

constants. The effect of this reduction is reflected in the target values (second left column in table 4.1), which are much lower than the actual mean values of an uncalibrated chip. However, to achieve homogeneity across a chip, this step is neccesary. To summarize, the calibration is successful for I_{gl} values below 1 µA, where the relative neuron-to-neuron variation is reduced by a factor of 2 to 3.

4.3 Membrane Capacitance

Each HICANN chip is equipped with four current stimulus generators that enable the injection of time-dependend current stimuli. These stimuli can be used in experiments to trigger the firing of neurons or to extract information about their dynamics. One information that can be extracted through current stimulation is the membrane capacitance of the neuron. According to the hardware specifications, the membrane capacitance has two settings, 2.165 pF for the bigcap setting and 0.164 pF for the smallcap settings. However, measurements on the chip suggest that the total capacitance of the neuron is between 3.6 pF to 4.0 pF *Millner* (2012). This means that the total capacitance of the neuron is about 70 % larger than specified. In this section, the capacitance measurement method from *Millner* (2012) is conducted for both bigcap and smallcap settings. These measurements are compared with measurements of the time constants done in section 4.1.

Methods

In *Millner* (2012), the capacitance of the neuron is measured by injecting a constant current into the neuron. When turning all leakages off by setting I_{convi}, I_{convx} and I_{gl} to 0 µA and deactivating adaptation and exponential term (see section 3.1), the neuron will

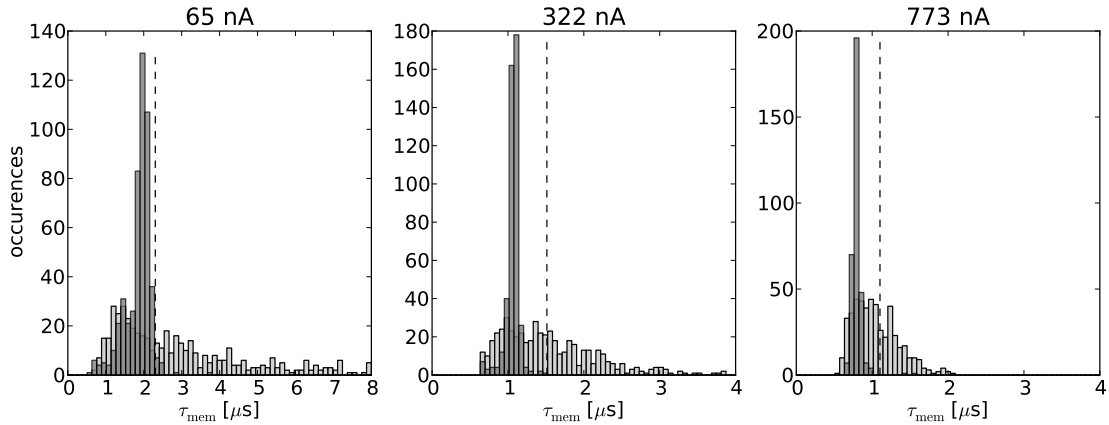

Figure 4.8 Distributions of τ_{mem} for three different settings of $\mathrm{I_{gl}}$ before (light grey) and after (dark grey) calibration.

be constantly spiking. The spike frequency depends on the current I, the reset potential V_{reset}, the threshold potential V_{t} and the capacitance C in the following way:

$$f = \frac{I}{C \cdot A} \tag{4.4}$$

where A is the trace amplitude $A = V_{\mathrm{t}} - V_{\mathrm{reset}}$. This can also be written as

$$(f \cdot A) = \frac{1}{C} \cdot I, \tag{4.5}$$

where a linear fit can extract C if $f \cdot A$ is measured as a function of I. Thus, by measuring spike frequency as well as the reset and threshold potentials and knowing the current, the total capacitance can be calculated.

It is important to note that eq. (4.4) only holds true under the assumption that the reset happens instantaneously. In the hardware neuron however, the reset is not instantaneous but takes a finite amount of time. The measured spike rates therefore need to be corrected for that time. To acquire the average time it takes for a reset, the time differences between all maxima and subsequent minima of a voltage trace are averaged. This time can then be used to correct the spike frequency.

The current injection is done via the current stimulus generator, where the output current is controlled by an adjustable voltage V_{ref}. Up to a voltage of about 1 V, the output current scales linearly with the voltage:

$$I_{\mathrm{out}}(V_{\mathrm{ref}}) = V_{\mathrm{ref}} \cdot 1.95\,\mu\mathrm{A\,V^{-1}} \tag{4.6}$$

Due to variations in production, the slope of $1.95\,\mu\mathrm{A\,V^{-1}}$ can vary between different instances of the current generator. Since there are four different current stimulus generators on one chip, the current I will vary for neurons connected to different current generators. Therefore, all capacitance measurements will be separated between the four

Table 4.2 Results of the linear fits to the lines in fig. 4.10. The error estimates are taken from the linear fitting method.

block	C_b [pF]	C_s [pF]	$\frac{C_b}{C_s}$
0	4.321 ± 0.105	1.856 ± 0.053	2.327 ± 0.087
1	4.634 ± 0.098	1.973 ± 0.047	2.349 ± 0.075
2	4.333 ± 0.104	1.857 ± 0.052	2.333 ± 0.086
3	4.327 ± 0.095	1.851 ± 0.049	2.338 ± 0.080

blocks of neurons which are connected to the same line.

This measurement of capacitances can be compared to the measurements of time constants. In theory, the time constant τ_{mem} of a neuron should be linear to the membrane capacitance C_{mem}. The ratio of time constants for bigcap and smallcap settings should therefore be equal to the ratio of both capacitances C_b and C_s for the bigcap and smallcap settings:

$$\frac{\tau_b}{\tau_s} = \frac{C_b}{C_s}. \tag{4.7}$$

To examine if this relation holds true on the HICANN, the measurements from section 4.1 are used to calculate time constant ratios and compare those with the ratios of membrane capacitances.

Results

A typical membrane voltage trace that results from injecting a constant current and turning off all other leakage terms is shown in fig. 4.9. The crosses drawn in the trace denote the single sample points of the analog-to-digital converter (ADC). It can be seen from the trace that the membrane needs approximately 3 to 4 sample points to get pulled down to the reset potential. Assuming the ADC sample frequency to be about 100 MHz, this amounts to a time of 30 ns to 40 ns. This effect is not significant for low spike rates, where the time between spikes is larger than 10 µs. However, for spike rates of 1 MHz, this amounts to a reduction in total spike rate of 3 % to 4 %, introducing a significant distortion. Therefore, the reset time correction introduced in the methods section is needed to correctly measure the spike frequencies.

The results of spike frequency measurement for different injected currents is shown in fig. 4.10. As expected, the spike frequency of the neurons rises linearly with the injected current up to a point of about 1.1 V. This is the point at which the current that flows out of the generator does not increase anymore (*Millner*, 2012). For the linear fit, the values for V_{ref} above 1 V are not used. The value for $V_{\text{ref}} = 0$ V is also not used since most neurons do not spike at all. Results of linear fits to each of the lines in fig. 4.10 are shown in table 4.2.

The measured membrane capacitance for the bigcap setting is about two times larger than specified, while it is up to 12 times larger for the smallcap setting. These measurements suggest an additional capacitance at the neuron of roughly (2.24 ± 0.10) pF for the

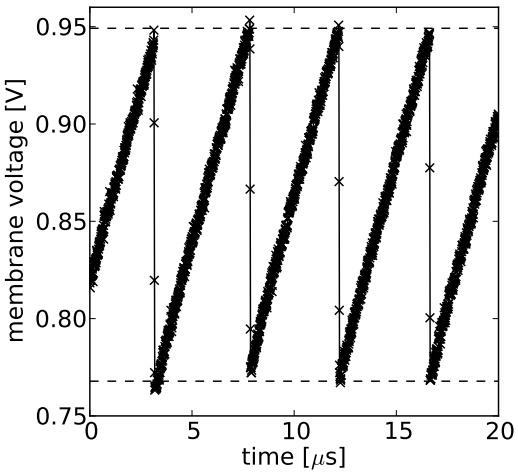

Figure 4.9 Example of a resulting voltage trace of the current injection experiment. The measured V_t and V_{reset} are marked with dashed lines. The sample points of the ADC are denoted by the crosses, showing that the neuron takes about 3 sample points to get pulled to the reset potential.

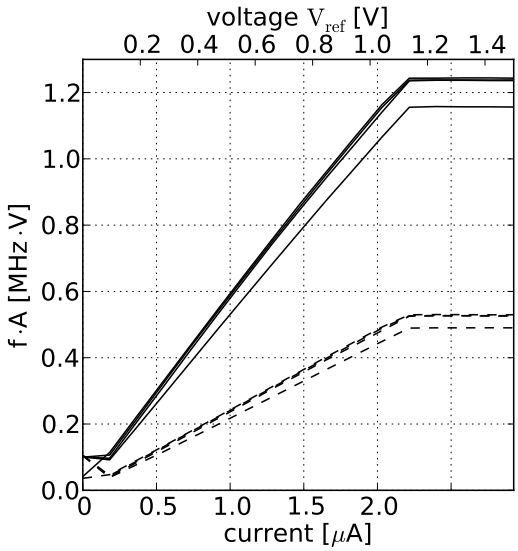

Figure 4.10 Dependency of $f \cdot A$ on the current that is injected. The mean $f \cdot A$ over each block of 128 neurons is plotted. The four dashed lines show the four blocks in the bigcap setting, while the four solid lines show the four blocks in the smallcap setting. Above 1.1 V, the current output does not increase anymore, explaining the plateaus at higher voltages. The points for $V_{ref} = 0\,\mathrm{V}$ as well as $V_{ref} > 1\,\mathrm{V}$ are not used in the linear fit.

bigcap and $(1.72 \pm 0.05)\,\mathrm{pF}$ for the smallcap setting. In *Millner* (2012), some sources for additional capacitances are explained. Parasitic capacitances of the transistors are estimated to be about 100 fF, while the capacitance of the line stimulus is estimated to be about 600 fF. These estimations however do not account for the full 2 pF and suggest that there are other factors influencing this measurement method. A possible factor could be leakage currents by any of the neuron circuit modules, since they cannot be completely cut off from the neuron.

To compare the ratios of capacitances to the ratios of time constants, the results from section 4.1 were used. The mean ratio over all time constants measured in section 4.1 is $R_\tau = 2.520 \pm 0.059$. This value agrees well with the mean measured ratio of the capacitances, which is $R_C = 2.336 \pm 0.164$. The conclusion is that the membrane time constant indeed scales linearly with the membrane capacitance of the neuron.

5 Discussion

Many methods to characterize and calibrate neuron circuits on the HICANN chip were introduced in previous chapters. The results show a reliable calibration that allows for the mapping of neural networks from software description to the wafer-scale hardware system. However, especially concerning future development of calibration routines, the choice of methods and possible alternatives need to be discussed.

Calibration routines

The calibration presented in chapter 3 proved to be a viable method for reducing the variability of neuron dynamics. While V_{reset}, V_t and V_{rest} measurements are very robust, the E_{syn} method revealed some problems. The number of failures in the E_{syn} calibration depends strongly on the chip that is calibrated, ranging from 6 neurons that are marked as broken, to up to 130 neurons. The goal is to keep the yield of working neurons as large as possible. To test if this problem originates from the hardware itself and not from the chosen measurement method, alternative methods for acquiring E_{syn} need to be examined. One alternative method would include sending spikes to a neuron and measuring the resulting standard deviation of the voltage trace while lifting the membrane potential up stepwise. At the point where the membrane is equal to the reversal potential, the standard deviation will have a minimum, since the incoming spikes do not have any effect on the membrane potential. If the problems that were revealed are sourced in the hardware itself, this method would fail for the same number of neurons since broken synapse drivers or synaptic input circuits values would still lead to spikes not being properly forwarded to the neuron.

As already stated in section 3.10, the calibration routines for V_{syntc} are preliminary. An advanced method for calibration of V_{syntc} would include the actual measurement of the synaptic time constant. This could be achieved by measuring and averaging voltage traces for many PSPs, similar to the τ_{mem} measurement method in section 4.1. From the shape of the averaged PSP, a fit can extract different parameters like the synaptic time constant and the reversal potential. The success of this fit will rely heavily on the chosen PSP function that is fitted to the data. In addition, since synaptic and membrane time constants will overlap in the shape of the PSP, a reliable measurement of the membrane time constant beforehand would increase the success of a PSP fit.

Conceptual approach of the calibration

The concept of the calibration presented in this work is based on the presumption that the calibration backend calibtic stores functions which take target floating gate (FG) parameters as input and return calibrated FG parameters. This calibrated FG parameter is chosen in a way that the neuron behaves like a neuron after the ideal transformation, i.e. like the mean of all neurons on a chip if set to the target FG parameter. The idea behind this approach is to keep the calibration process strictly separated from the other parameter translations introduced in section 2.4. This approach works well when the ideal transformation approximately holds true for all neurons. More precisely, the neurons all have to share a common domain of definition, which in this case is the range of hardware parameters they can be set to. This is not a problem for all the potentials, since all neurons can indeed be set to all values in the domain of the ideal transformation. The calibration of time constants however exposed challenges in this design approach, since the ranges of available time constants deviate more strongly from neuron to neuron. Particularly, very long time constants can only be reached by a fraction of neurons. When calibrating with the given approach, the domain of definition for the ideal transformation is chosen such that the possible parameter ranges of all neurons are reduced to the range where most neurons can operate in. This effectively abolishes long time constants for the sake of homogeneity.

It is therefore worthwhile to discuss the conceptual approach of the current calibration software. An alternative approach would be an implementation of the calibration not as the very last step after the biological parameter was translated to a hardware parameter and then to a floating gate parameter. Instead, the calibration could function as an intermediate step when translating from hardware to floating gate parameter. Instead of using functions that convert FG parameters to calibrated FG parameters, the backend could use functions that convert the hardware parameters directly to the corresponding FG parameters. After characterization, each neuron could then be assigned its own valid hardware parameter range. These ranges could be used by the mapping and routing tool marocco (*Jeltsch*, 2014) as the basis of which neurons to pick when translating from software to hardware. This approach would dispose of the ideal transformations, eliminating one source of errors.

Measurement of time constants

The measurement method for the membrane time constants described in section 4.1 differs strongly from the method introduced in *Schwartz* (2013). In the latter, the method to measure τ_{mem} is to set $\mathrm{E_l}$ slightly above $\mathrm{V_t}$ so that the neuron is constantly spiking. For an ideal LIF neuron, the resulting spike frequency f gives rise to the membrane time constant in the following way:

$$\tau_{\mathrm{mem}} = \frac{1}{f} \cdot \ln\left(\frac{V_{\mathrm{reset}} - V_{\mathrm{rest}}}{V_t - V_{\mathrm{rest}}}\right)^{-1}. \tag{5.1}$$

With this setup, the time constant can be derived by measuring the firing rate f, the resting potential V_{rest}, the firing threshold V_t and the reset potential V_{reset}.

The main advantage of this method is its simplicity. However, it is not possible to exactly measure V_{rest} with this method because a resting state is never reached. Instead, V_{rest} needs to be measured in a separate experiment, being prone to trial-to-trial errors. By using Gaussian error propagation on eq. (5.1), an error in the estimation of V_{rest} would result in the following error in τ_{mem}:

$$\frac{\Delta\tau}{\tau} = \ln\left(\frac{V_{reset} - V_{rest}}{V_t - V_{rest}}\right)^{-1} \frac{V_{reset} - V_t}{(V_{reset} - V_{rest})(V_{rest} - V_t)} \cdot \Delta V_{rest}. \tag{5.2}$$

Since V_{reset} and V_t are directly measured from the trace, their errors will be neglected; they can be assumed to be much smaller than the uncertainty in V_{reset}. From eq. (5.2), it can be seen that the relative error in τ_{mem} is inversely proportional to $V_{rest} - V_t$. To stay in the exponential range of the OTA, the potential differences need to be as small as possible. This dependency can easily lead to large relative errors in τ_{mem}. As an example, for a realistic configuration of V_{reset}=850 mV, V_t=900 mV and V_{rest}=930 mV, an uncertainty in the estimation of V_{rest} of 1 % would result in an uncertainty of 22 % in τ_{mem}. This dependency limits the precision of this method, potentially leading to very large variations between measurement trials.

The method used in this thesis uses the fitting of an exponential function to averaged voltage traces. The uncertainties for the measured τ_{mem} values are estimated by the fit method. These uncertainties are typically in the 0.1 % order, being two orders of magnitude lower than the errors that appear in the frequency method. While this method has a much lower uncertainty in the measured τ_{mem}, it distorts the actual membrane time constant of the neuron by adding the capacitance of the current stimulus line to the membrane capacitance of the neuron. In *Millner* (2012), this capacitance is estimated to be about 600 fF, which would be up to 30 % of the neuron's total capacitance. By using the current stimulus method throughout this thesis, a predictable systematic distortion was chosen over unpredictable uncertainties in the measurement method.

6 Outlook

The calibration methods described in this thesis greatly improve the performance of network emulations on the wafer-scale system. It was shown that for the first time on the wafer-scale system, a neural network emulation can be conducted without having to choose, tune and connect each neuron manually. This chapter gives an overview over the role that the calibration process will play in future operation and development of neuromorphic hardware systems.

Scaling up Calibration

When considering the future of the wafer-scale system, an important aspect is the feasibility of calibrating a large number of HICANN chips. Currently, the full calibration of one chip takes about one hour, but that time could well be increased by a factor of two if all remaining parameters are to be calibrated as well. Assuming the calibration of one HICANN chip to take take two hours for simplicity, sequentially calibrating all 384 chips on one wafer will take 768 hours, or 32 days. At the end of the BrainScaleS project, it is planned to have eight working wafer-scale systems, while in the Human Brain Project (HBP), at least 20 wafer-scale systems are planned. This would lead to a total calibration time of 256 days for BrainScaleS and 640 days for HBP, assuming that measurements will run twenty-four hours a day. These numbers immediately highlight the need for a parallelization of the calibration process. Fortunately, a parallelization across wafers is possible without any additional effort, since each wafer will be connected to a separate control computer. In that way, the full bandwidth capabilities will be utilized and the speed of the communication with the system does not depend on the number of wafers that are used (*Müller*, 2014). Running the calibration process on all wafers in parallel will decrease the total calibration time of any number of wafers to 32 days – the calibration time of one wafer. However, spending a whole month on non-stop calibration would still not be very feasible. Therefor, a parallelization across reticles is the next logical step. If reticles are calibrated in parallel, the total calibration time will be decreased from 32 days to about 16 hours – the time it takes to calibrate all eight HICANNs on one reticle. This time could still be cut in half by making use of both analog readout channels on each reticle, calibrating two HICANNs per reticle at once. In conclusion, a completely parallelized calibration that fully exploits the capabilities of the system would take only four times longer for any number of chips on any number of wafers than it would take for a single HICANN.

Executable system specification (ESS)

The ESS is a detailed software simulation of the wafer-scale system. One of its purposes is to aid developers of neuronal network models to cope with hardware limitations at an early stage, before translating the model to the hardware. While hardware restrictions on mapping and routing as well as bandwidth limitations are simulated in the ESS, the variations of neuron model parameters are not yet implemented. Instead, AdEx neurons with ideal parameters are used to calculate the dynamics of the membrane potential. Including model parameter variations that were presented in this thesis in the ESS would increase the accuracy of the hardware simulation. The effects of parameter variations on the model could then be systematically examined to better understand where the model needs to be adjusted. Approaches on how to compensate for hardware limitations are presented in *Petrovici et al.* (2014), where information on parameter variations that were acquired during the scope of this thesis were used.

HICANNv3

Besides allowing neural networks to run on the hardware, the calibration process gives deep insight into the technical challenges of the hardware system. In the current HICANN version, there are design problems in the neuron circuit that became apparent during the calibration process, especially concerning the synaptic input circuit. To gain a deeper understanding of the problem sources, the synaptic input circuit was thoroughly simulated and analyzed in *Kiene* (2014). One problem that was observed in the circuit is the exponential dependence of the integrator's resistance on the voltage V_{syntc} (see section 2.2.3). Another problem that makes the calibration much more challenging are the synaptic leakage currents towards the reversal potential E_{syn} in the absence of spike input. A decrease of these leakages could greatly improve the precision of resting potentials. Simulations from *Kiene* (2014) also show that, since the capacitance in the integrator circuit seems to be too small, only a handful of spikes are enough to saturate the synaptic input.

Suggested improvements to the synaptic input circuit include the replacement of the resistive element by a more linear device and the implementation of a larger capacitor, potentially sacrificing a few rows of synapse lines. To counter the synaptic leakage currents, an additional control voltage could be introduced that is connected to OTA_1 instead of V_{syn}. This control voltage would allow the calibration of OTA_1 to cancel out the offset currents that are generated by the integrator. The problems that were found in the circuit and the fact that most of them can be eliminated led to considering the development of a next chip revision, the HICANNv3.

HICANN-DLS

Parallel to using the current version of the HICANN chip and considering the development of a HICANNv3, the Electronic Vision(s) group is developing the next generation of the chip, the HICANN-DLS. Among the countless features of the next generation hardware, there will be innovative changes on how plasticity and parameter storage on

the chip will work. Parameters will no longer be stored in a floating gate array which needs to be fully reprogrammed each time a parameter is changed. Instead, a parameter storage will be used which can be reprogrammed on-the-fly and updates substantially faster than the floating gate implementation (*Hock*, 2014). Plasticity will be implemented with a plasticity processing unit (PPU) that can be programmed in the C programming language (*Friedmann et al.*, 2013). The PPU will potentially be able to read out and process membrane voltage traces, spikes or the parameter storage memory. It will also be able to send any signals to the communication layer, including the signals needed to change parameters on-the-fly. The combination of new parameter storage and PPU would allow for a completely different approach to calibration. For example, the PPU could be programmed in a way to calibrate the neurons incrementally by comparing the distance from measured to target value and simply changing the parameter until the target is reached. In the current version, this approach would take far too long since reprogramming all floating gates takes between forty seconds and one minute.

However, it will take a few more years until a working wafer-scale system equipped with the HICANN-DLS chip will be operational. Until then, putting the current HICANN into service is a high priority goal. A thorough understanding of the current system through characterization and calibration is a vital part of this process and aids the developers of future chip generations to ensure the success from early on.

A Appendix

Modeling the operational transconductance amplifier

The common technique when bringing neuronal network models from software to hardware is the compensation of distortions introduced by the hardware (*Petrovici et al.*, 2014). For the waferscale system, most of the hardware distortions such as bandwidth limitations or mapping and routing capabilities can be simulated by the ESS before mapping a network to the actual hardware. However, one distortion that is not yet modeled is the nonlinearity of the OTAs. Instead, the ESS uses an ideal AdEx model to simulate neuron dynamics. A possible way to model the OTA behaviour was given in *Stöckel* (2014). This section introduces an alternative way to model the dynamics of the OTA based on a sigmoidal output function.

Model and Methods

The output current of a leakage conductance in the AdEx model is

$$I_{\text{out}} = g_l \cdot (V_{\text{mem}} - E_l) \tag{A.1}$$

with the leakage conductance

$$g_l = \frac{I_{\text{out}}}{V_{\text{mem}} - E_l} = \text{const.} \tag{A.2}$$

This behaviour leads to an exponential voltage decay to the leakage potential $\mathrm{E_l}$ after the membrane voltage is excited.

However, the conductance of an OTA is not a constant value. As described in section 2.2.1, the current provided by the OTA is approximately linear to the differential voltage $\Delta V = V_{\text{mem}} - E_l$ if the differential voltage stays below 100 mV. For larger values of ΔV, the output current starts to saturate, with the maximum output current being the OTA bias current $\mathrm{I_{bias}}$. The effective conductance of the OTA beyond this point is no longer constant. Currently, these saturation effects are avoided by reducing the dynamic range to 200 mV. However, the behaviour of the OTA can be modeled by a sigmoidal function

$$I_{\text{out}} = 2 \cdot I_{\text{bias}} \cdot \left(\frac{1}{1 + \exp\left(-\frac{E_l - V_{\text{mem}}}{\kappa}\right)} - 0.5 \right) \tag{A.3}$$

where $\mathrm{E_l}$ is the leakage potential, V_{mem} the membrane potential, κ the slope of the curve and $\mathrm{I_{bias}}$ the maximum output current.

The resulting conductance g_l will then depend on the differential voltage $\Delta V = V_{\text{mem}} - E_l$ by

$$g_l = \frac{dI}{d(\Delta V)} = \frac{2 \cdot I_{\text{bias}} \cdot e^{\frac{-\Delta V}{\kappa}}}{\kappa \cdot \left(e^{\frac{-\Delta V}{\kappa}} + 1\right)} \tag{A.4}$$

The maximum conductance $g_{l,\max}$ occurs at $\Delta V = 0$:

$$g_{l,\max} = \frac{I_{\text{bias}}}{\kappa} \tag{A.5}$$

Using eq. (A.3), a membrane voltage trace can be constructed by insterting I_{out} into

$$\frac{dV_{\text{mem}}}{dt} = \frac{I}{C}. \tag{A.6}$$

Solving eq. (A.6) yields

$$V_{\text{mem}}(t) = \frac{1}{C} \int I_{\text{out}} \tag{A.7}$$

which can be solved numerically to generate membrane voltage traces. To test how well this model represents voltage traces from the hardware, a curve fit to measured data is done and compared to a curve fit of the exponential model to the same data. To test if the OTA model also covers the exponential part, a second fit is done. For this fit, the membrane trace is cut so that the maximum differential voltage used in the fit is 100 mV.

Naturally, it is of interest to extract useful information about the neuron characteristics from the abstract parameters of the model. Parameters that can be extracted are the leakage conductange g_l and the time constant τ_{mem}. Since these parameters are not inherently well defined in the OTA model, we need to compare them to the parameters of the exponential fit within the linear range. The conductance of the OTA model is no longer constant, but depends on the differential voltage. However, the leakage conductance at any given differential voltage can be calculated via eq. (A.4). An effective leakage conductance $\overline{g_{l,\text{OTA}}}$ can then be calculated by taking the mean over the conductance within the linear range (i.e. from $-100\,\text{mV}$ to $100\,\text{mV}$) (see fig. A.6). The time constant $\tau_{\text{mem,OTA}}$ is obtained by inserting $g_{l,\text{OTA}}$ into $\tau_{\text{mem}} = \frac{C}{g_l}$. An alternative way to obtain $\tau_{\text{mem, OTA}}$ from the model parameters is to find the time it takes the membrane voltage to decay from $100\,\text{mV}$ to $36.8\,\text{mV}$, i.e. $\frac{1}{e}$ of elevation.

Results

The resulting output current of eq. (A.3) is visualized in figs. A.1 and A.2. Qualitatively, this figure compares well to OTA simulations shown in (*Millner*, 2012). Membrane voltage traces generated by eq. (A.7) are shown in fig. A.3. These simulations illustrate how the voltage decays linearly, not exponentially, for large differential voltages. The result of fitting the model to a measured membrane voltage trace is shown in fig. A.4. This

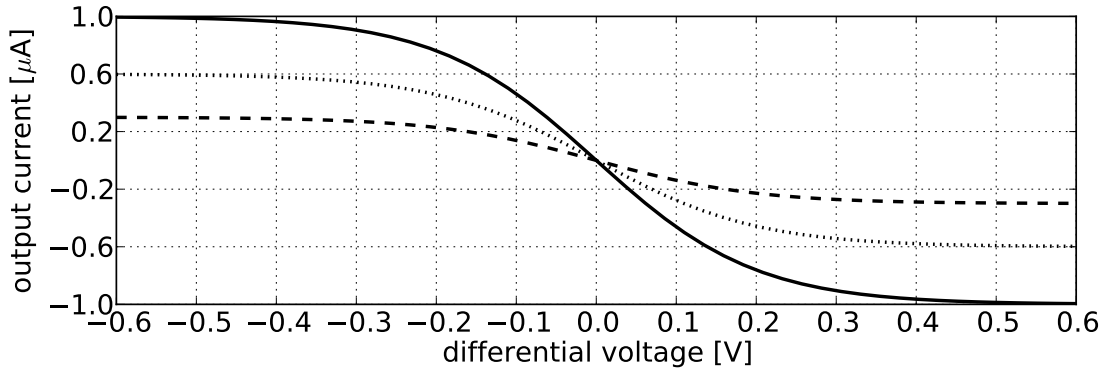

Figure A.1 Output current depending on the differential voltage for I_{bias} of 0.3 µA (dashed), 0.6 µA (dotted) and 1.0 µA (solid) and a slope κ of $-0.1\,V^{-1}$. Curves were generated with the model in eq. (A.3).

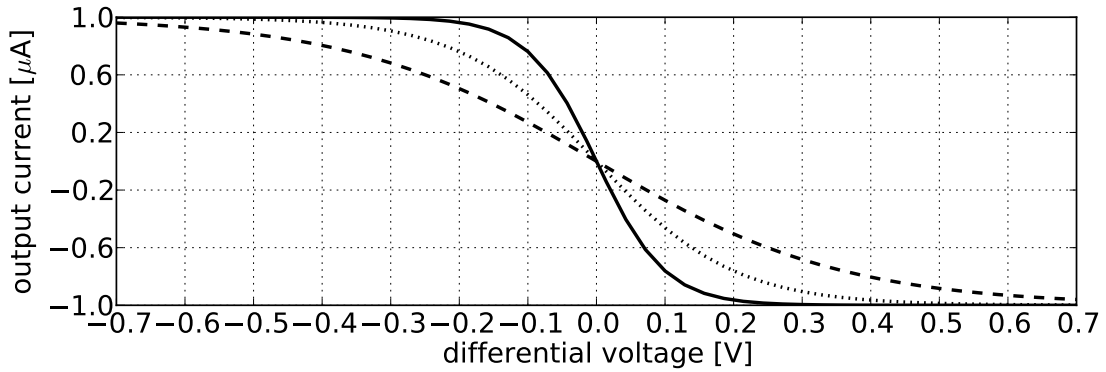

Figure A.2 Output current depending on the differential voltage for κ of $-0.18\,V^{-1}$ (dashed), $-0.1\,V^{-1}$ (dotted) and $-0.15\,V^{-1}$ (solid) with I_{bias} set to 1 µA. Curves were generated with the model in eq. (A.3).

plot clearly shows that the OTA model more accurately describes the OTA behaviour in this voltage regime since the fitted curve is indistinguishable from the measured trace. The result of a fit within the exponential range is shown in fig. A.5. Again, the OTA model represents the measured curve very well, while the exponential model yields a similar result.

The extracted membrane time constants and conductances are summarized in table A.1. Both models give similar results if the differential voltage stays below 100 mV. However, the OTA model yields the same result independent of the differential voltage range. In conclusion, the OTA model presented in this section is a viable option for a correct simulation of the conductances found on the hardware neuron.

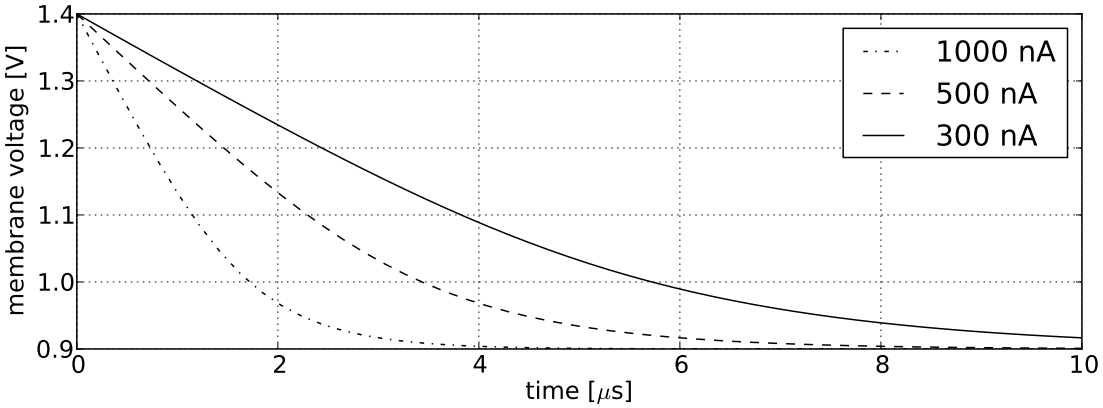

Figure A.3 Membrane voltage traces generated by model eq. (A.7) for different values of I_{bias}. The leakage potential is set to 0.9 V, while the slope κ is set to $-0.1\,V^{-1}$. The capacitance C is set to 2.165 fF, matching the large capacitance setting of the hardware neuron.

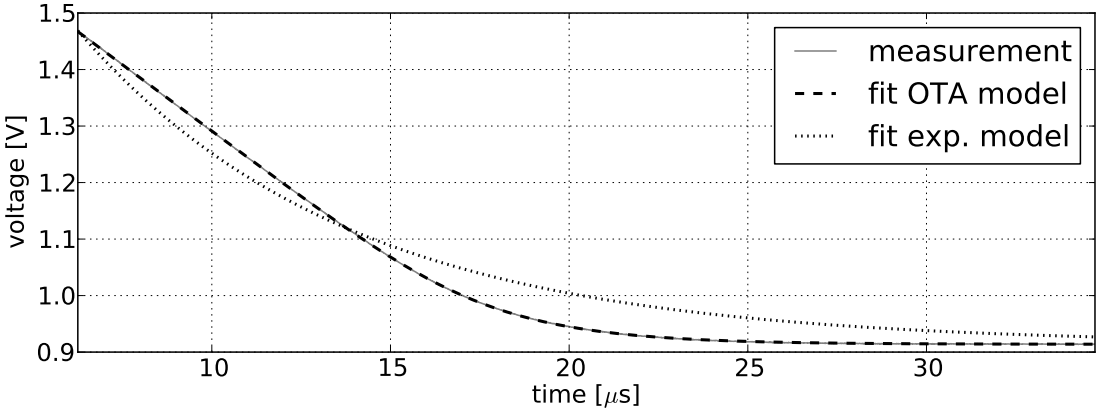

Figure A.4 Fit of OTA model and exponential model to a measured membrane voltage trace. The measured trace is the result of averaging 24 traces and therefor contains very little noise. The OTA model overlaps perfectly with the measured curve, while the exponential model is not able to cover the saturation effect.

Table A.1 Results of the two fits found in figs. A.4 and A.5.

	τ_{mem1} [µs]	g_{l1} [µS]	τ_{mem2} [µs]	g_{l2} [µS]
OTA model	2.91	1.20	2.93	1.19
exponential model	7.56	0.46	2.92	1.20

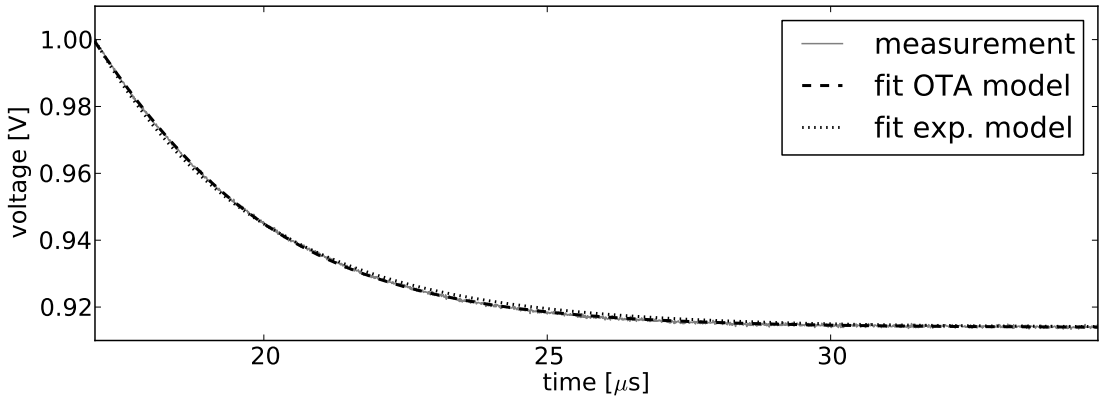

Figure A.5 Fit of OTA model and exponential model to a measured membrane voltage trace. The measured trace is the result of averaging 24 traces and therefor contains very little noise. Since the differential voltage is below 100 mV, the exponential model also overlaps well with the measured trace.

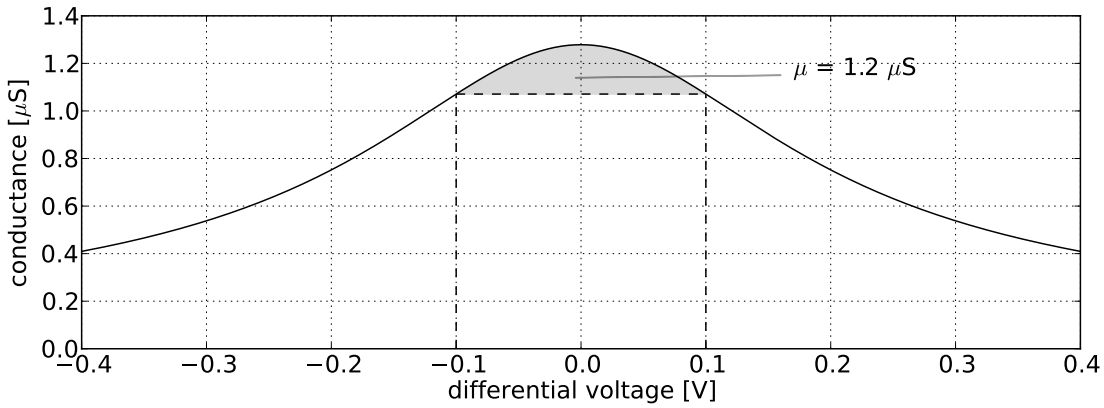

Figure A.6 Conductance resulting from the OTA model with parameters measured by the fit in fig. A.4. To compare this result with the result acquired through exponential fitting, a mean conductance of the exponential range ($\pm$ 100 mV) can be done (shaded area).

Bibliography

Aarli, J. A., T. Dua, A. Janca, and A. Muscetta, *Neurological disorders: public health challenges*, World Health Organization, 2006.

Billauer, E., peakdet: Peak detection using matlab, [`http://billauer.co.il/peakdet.html`; accessed: 2014-10-27], 2012.

Brette, R., and W. Gerstner, Adaptive exponential integrate-and-fire model as an effective description of neuronal activity, *J. Neurophysiol.*, *94*, 3637 – 3642, doi:NA, 2005.

Davison, A. P., D. Brüderle, J. Eppler, J. Kremkow, E. Muller, D. Pecevski, L. Perrinet, and P. Yger, PyNN: a common interface for neuronal network simulators, *Front. Neuroinform.*, *2*(11), 2008.

Friedmann, S., N. Frémaux, J. Schemmel, W. Gerstner, and K. Meier, Reward-based learning under hardware constraints - using a risc processor embedded in a neuromorphic substrate, *Frontiers in Neuroscience*, *7*(160), doi:10.3389/fnins.2013.00160, 2013.

Hock, M., Modern semiconductor technologies for neuromorphic hardware, Ph.D. thesis, Ruprecht-Karls-Universität Heidelberg, 2014.

Jeltsch, S., A scalable workflow for a configurable neuromorphic platform, Ph.D. thesis, Ruprecht-Karls-Universität Heidelberg, 2014.

Kiene, G., Evaluating the synaptic input of a neuromorphic circuit, Bachelor thesis, Ruprecht-Karls-Universität Heidelberg, 2014.

Klähn, J., Untersuchung und management von synapsendefektverteilungen in einem großskaligen neuromorphen hardwaresystem, Bachelor thesis (German), University of Heidelberg, HD-KIP 13-36, 2013.

Kononov, A., Testing of an analog neuromorphic network chip, Diploma thesis, Ruprecht-Karls-Universität Heidelberg, hD-KIP-11-83, 2011.

Krizhevsky, A., I. Sutskever, and G. E. Hinton, Imagenet classification with deep convolutional neural networks, 2012.

Lande, T., H. Ranjbar, M. Ismail, and Y. Berg, An analog floating-gate memory in a standard digital technology, in *Microelectronics for Neural Networks, 1996., Proceedings of Fifth International Conference on*, pp. 271 –276, doi:10.1109/MNNFS.1996.493802, 1996.

Millner, S., Development of a multi-compartment neuron model emulation, Ph.D. thesis, Ruprecht-Karls University Heidelberg, 2012.

Müller, E., Novel operation modes of accelerated neuromorphic hardware, Ph.D. thesis, Ruprecht-Karls-Universität Heidelberg, 2014.

Petrovici, M. A., et al., Characterization and compensation of network-level anomalies in mixed-signal neuromorphic modeling platforms, *PLOS ONE*, doi:dx.doi.org/10.1371/journal.pone.0108590, 2014.

Pfeil, T., et al., Six networks on a universal neuromorphic computing substrate, *Frontiers in Neuroscience*, *7*, 11, doi:10.3389/fnins.2013.00011, 2013.

PyNN, A Python package for simulator-independent specification of neuronal network models, [`http://neuralensemble.org/docs/PyNN/reference/neuronmodels.html`; accessed: 2014-10-15], 2014.

Schemmel, J., 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.

Schemmel, J., 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)*, pp. 1947–1950, 2010.

Schemmel, J., A. Grübl, S. Millner, and S. Friedmann, Specification of the HICANN microchip, FACETS and BrainScaleS project internal documentation, 2012.

Schmuker, M., T. Pfeil, and M. P. Nawrot, A neuromorphic network for generic multivariate data classification, *Proceedings of the National Academy of Sciences*, *111*(6), 2081–2086, 2014.

Schwartz, M.-O., Reproducing biologically realistic regimes on a highly-accelerated neuromorphic hardware system, Ph.D. thesis, Universität Heidelberg, 2013.

Sobel, I., and G. Feldman, A 3x3 Isotropic Gradient Operator for Image Processing, never published but presented at a talk at the Stanford Artificial Project, 1968.

Stöckel, D., Measuring the leakage current module characteristic of the hicann neuron circuit, 2014.

Acknowledgments

At this point I would like to thank the following people who supported me during my master thesis:

My supervisors Prof. Dr. Karlheinz Meier and Dr. Johannes Schemmel for providing this great atmosphere to work in.

Mitja Kleider, Sebastian Schmitt, Christoph Koke, Paul Müller, Thomas Pfeil and Eric Müller for proofreading my thesis and helping me polish it to this level.

Syed Ahmed Aamir, Matthias Hock, Simon Friedmann and all the other hardies for answering my questions about electronics and integrated circuits, with which I had no experience whatsoever when I started in the group.

Finally, the whole Elecronic Vision(s) group for their scientific support and entertainment.

Statement of Originality (Erklärung):

I certify that this thesis, and the research to which it refers, are the product of my own work. Any ideas or quotations from the work of other people, published or otherwise, are fully acknowledged in accordance with the standard referencing practices of the discipline.

Ich versichere, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Heidelberg, October 31, 2014

.....
(signature)