Department of Physics and Astronomy
University of Heidelberg

Bachelor Thesis in Physics
submitted by

Malte Wehrheim

born in Bremen Nord (Germany)

October 2019

Reconstruction of Synaptic Weight on the Neuromorphic BrainscaleS-1 System

This Bachelor Thesis has been carried out by Malte Wehrheim at the
KIRCHHOFF-INSTITUTE FOR PHYSICS
RUPRECHT-KARLS-UNIVERSITÄT HEIDELBERG
under the supervision of
Dr. Johannes Schemmel

Abstract Neuromorphic computing aims to imitate the behaviour of neurons and synapses of the brain to solve complex problems. In the Electronic Vision(s) group this task is done using electronic circuits. In this thesis we try reconstructing the synaptic weight by analysing postsynaptic potentials (PSPs) to work towards a synaptic weight calibration. To do this we looked at different approximations to describe the conductance based (COBA) AdExp neurons used in the BrainscaleS-1 wafer scale system. The AdExp model contains additional terms to the standard "Leaky Integrate and Fire" (LIF) neuron model. We found that approximations only describe the course of the PSPs for very small weights which means that we cannot use them to calibrate for a wide range of weights. For this reason we used numeric python modules to fit the not analytically solvable solution to the COBA neurons. We used the fit to find neuron parameters best suitable to measure the synaptic weights. To reconstruct an accurate weight we had to compensate for nonlinear hardware behaviour and adjust our fit to work even with noise on the fitted data. In conclusion a better understanding of the synaptic weight was developed in this thesis which will help in future experiments.

Zusammenfassung Neuromorphes Computing nutzt Modelle um das Verhalten von den Neuronen und Synapsen des Gehirns zu imitieren und mit diesen komplexe Aufgaben zu bewältigen. In der Electronic Vision(s) Gruppe werden hierfür elektronische Schaltkreise verwendet. In dieser Arbeit befassen wir uns mit der Bestimmung des synaptischen Gewichts durch das Analysieren von Postsynaptischen Potentialen. Dadurch soll auf eine Kalibrierung des synaptischen Gewichts hingearbeitet werden. Hierfür haben wir uns mit Näherungen für die im BranscaleS-1 Wafersystem verwendeten Konduktanz basierten AdExp Neuronen beschäftigt, die eine Weiterentwicklung des Standard "Leaky Integrate and Fire" Neuron Modells darstellen. Dabei sind wir jedoch zu dem Schluss gekommen, dass das Fitten der numerisch bestimmten Lösung die einzige Möglichkeit ist, um das Gewicht für einen größeren Parameter Bereich zu rekonstruieren. Mithilfe des Fits haben wir Neuronparameter bestimmt mit denen die Rekonstruktion des Gewichts am Besten funktionierte. Um ein genaues synaptisches Gewicht zu bestimmen haben wir Wege gesucht, um nichtlineare Effekte der Hardware zu kompensieren und unsere Fit-Routine angepasst, um auch exakt mit rauschbehafteten Messdaten arbeiten zu können. Insgesamt führt diese Arbeit zu einem besseren Verständnis des synaptischen Gewichts, was in zukünftigen Experimenten hilfreich sein wird.

Erklärung

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

Heidelberg, den ...,

Contents

1 Introduction

As the performance increase of conventional computers built after the Von Neumann principle is coming to an end with a decline in the in Moore's law predicted performance increase [1], the search for new kinds of computers becomes more and more important [2]. Inspired by the human brain, which is very energy efficient and able to excel at a wide range of tasks, neuromorphic hardware presents itself as an interesting field of study. The neuromorphic BrainscaleS-1 system from the Electronic Vision(s) group from the University of Heidelberg utilizes electronic circuits to imitate the behaviour of biological neurons and synapses [3].

1.1 Leaky Integrate and Fire Model Theory

The neurons used in the BrainscaleS-1 system are so called conductance based AdExp neurons [3]. This model describes the behaviour of neurons by using an analogy to electronic circuits to reduce the complexity of the biological neurons. The states of the neuron (i.e. voltages of the neurons membrane) follow the following differential equation.

$$\begin{aligned} -C_\mathrm{m}\frac{dV}{dt} =& g_\mathrm{l}(V-V_\mathrm{l}) - g_\mathrm{l}\Delta_\mathrm{th} exp(\frac{V-V_\mathrm{th}}{\Delta_\mathrm{th}}) \\ &+ g_\mathrm{e}(t)(V-E_\mathrm{e}) + g_\mathrm{i}(t)(V-E_\mathrm{i}) + w(t) \end{aligned} \tag{1}$$

With g_l leak conductance, V_l resting potential, $g_\mathrm{e/i}(t)$ total synaptic conductance, $E_\mathrm{e/i}$ synaptic reversal potential, Δ_th slope factor, V_th threshold potential and $w(t)$ adaptation factor. As we only look at non spiking behaviour $((V(t) < V_\mathrm{th})\forall t)$ and set parameters scaling the adaptation current to zero the AdExp model is identical to the "Leaky Integrate and Fire" model. We will from now on call it the LIF model. The states of the neuron therefore follow the following differential equation [4]:

$$C_\mathrm{m}\frac{dV}{dt} = -g_\mathrm{l}(V-V_\mathrm{l}) - \sum_{i=1}^{k} I_\mathrm{i}^\mathrm{syn}(t). \tag{2}$$

As can be seen, the change over time of the membrane potential V depends on the difference of the potential to a leak or resting potential V_l and the leakage conductance g_l. The current flowing on the membrane is given by the sum over all k synaptic connections which each contribute a current $I_\mathrm{i}^\mathrm{syn}$. The form of this current depends on the used neuron model. For the current based neuron model (CUBA) the current is given by:

$$I^\mathrm{syn}(t) = w \cdot exp\Big(-\frac{t-t_0}{\tau_\mathrm{syn}}\Big) \cdot \Theta(t-t_0). \tag{3}$$

The current changes after the arrival of a synaptic spike at time t_0 and decays exponentially with the synaptic time constant τ_syn. The amplitude of the current is linearly scaled with the synaptic weight w.
A model closer to biology and used in the BrainscaleS-1 system is the conductance based neuron model (COBA). Here the current flowing on the membrane additionally depends on the difference of the membrane voltage to a reversal potential.

$$I^\mathrm{syn}(t) = w \cdot exp\Big(-\frac{t-t_0}{\tau_\mathrm{syn}}\Big) \cdot \Theta(t-t_0) \cdot (V - V_\mathrm{i}) \tag{4}$$

The only difference between CUBA and COBA is the introduction of the difference of the membrane voltage V and the reversal potential V_i. The reversal potential will from now on be called E_syn for synaptic reversal potential.
It is important to note, that the currents are estimated to change instantaneously after the arrival of a spike. Depending on the synapse model one would introduce another exponential factor which describes the rise of the current. However the rising time constant is very small for the synapse we look at compared to the synaptic time constant τ_syn and the membrane time constant $\tau_\mathrm{m} = \frac{c_\mathrm{m}}{g_\mathrm{l}}$ and thus this factor can be neglected. As mentioned before the firing part of the neurons is also neglected here. The membrane voltage would rise exponentially after reaching a threshold and then reset to a fixed reset potential for a time τ_ref. The neuron then emits a spike travelling to its postsynaptic partners. Though this part is really important for computation with neurons we will not look at this feature as we set the threshold voltage over our excitatory reversal potential. Ideally we will then have no spiking behaviour of the analysed neurons.
Though the conductance based model better describes the biological neuron

behaviour it comes with different problems. The differential equation for current based neurons is easily solvable. For the conductance based differential equation however an analytical solution is not possible.

1.2 Realization on Hardware

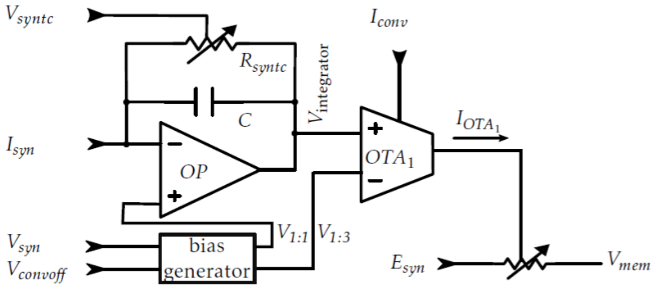


Figure 1: Synaptic input circuit of HICANN revision 4/4.1 to realize the course of the synaptic conductance seen in Eq. (2). The synaptic weight is here introduced as the current I_{syn} that flows on an integrator and then scales the resistive element between reversal potential and membrane. Figure from [5].

On hardware the scaling of the synaptic current is realized by controlling a resistive element between the membrane and the reversal potential. The circuit can be seen in Figure 1. This circuit looks identical for the excitatory and the inhibitory input of the neurons. The only difference being a change in the reversal potential (here E_{syn}, for inhibitory smaller than the resting potential for excitatory higher than the resting potential) and a change in the resistive element to realize a constant conductance independent of the applied voltages.

After the arrival of a spike the current I_{syn} flows on the integrator for a constant time t_{syn}. The voltage of the integrator is then translated to a current by OTA_1. The current I_{OTA_1} then linearly scales the conductance of the resistive element. In the integrator we find another resistive element which is scaled by the voltage V_{syntc}. This resistive element allows us to change the synaptic time constant which describes the decay of the conductance after the spike $\tau_{\text{syn}} = R_{\text{syntc}} \cdot C$. Also a bias is applied to the OTA and the operation amplifier in the integrator to improve the stability of the circuit. These biases are created in the bias Generator scaled by two parameters V_{syn} and V_{convoff}.

The linear scaling of the conductance with the synaptic weight happens in the input current I_{syn}. Ideally the current is given by

$$I_{\text{syn}} = V_{\text{gmax}} \cdot g_{\text{scale}} \cdot \frac{w}{g_{\text{div}}}. \tag{5}$$

In Figure 2 we see how the scaling is done in hardware. The scaling describes

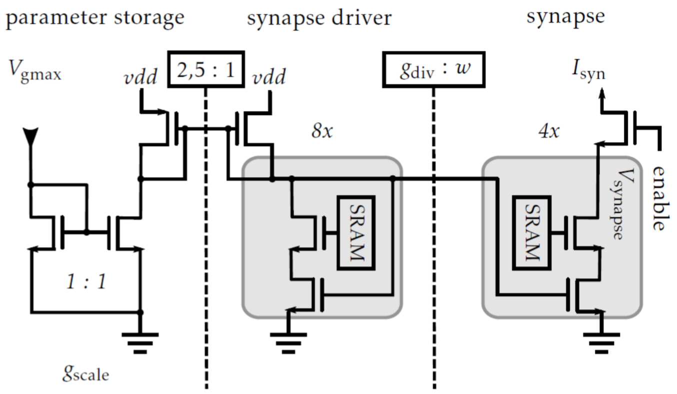


Figure 2: Scaling of the current I_{syn} of the synaptic input circuit. The reference current vgmax is scaled using differently sized current mirrors depending on the input parameters g_{div} and bit-weight w and the for all HICANNs fixed scaling factor g_{scale}. Figure from [5].

the steps from the bias current scaled by V_{gmax} to the current I_{syn}. The w here is a bit-weight which can take values of 0 to 15 by switching on and of 4 transistors of different sizes. The divisor g_{div} works similarly and takes values from 1 to 30 (realized with 8 transistors). Though the transistors do not scale with potencies of 2 like for the bit weight (scaling factors of

1+1+2+2+4+4+8+8). The factor g_{scale} is fixed for all HICANN and only dependent on the revision (4. for revision 4 and .4 for revision 4.1). The factor g_{div} is chosen for the complete row of synapses. V_{gmax} can be chosen from one of four possible global parameters per quadrant of the chip. As we only look at one HICANN with a external spike input the whole range of parameters is available to us.

1.3 Task of Thesis

This thesis tries to connect the synaptic weight in the theoretical model with the realization of the synaptic weight on hardware. It builds up on the work in [5] and continues the work towards the recreation of the synaptic weight. The hardware underlies variations between each neuron, synapse and even each individual transistor. These variations come e.g. from differences in the doping strength or variations in the size of each transistor. For this reason the actual synaptic weight is different between neurons for identical setup parameters.

In this thesis we will show how to reconstruct the synaptic weight by analysing postsynaptic potentials. This way a systematic comparison of hardware measurements and simulation will be possible.

2 Towards a Synaptic Weight Reconstruction

The approach to reconstruct the synaptic weight is to analyze the postsynaptic potential after the arrival of a spike. As seen in section 1 the voltage course follows the differential equation from Eq. (2). As the synaptic weight is a parameter in the differential equation we expect to be able to reconstruct it from the height and shape of the PSP.

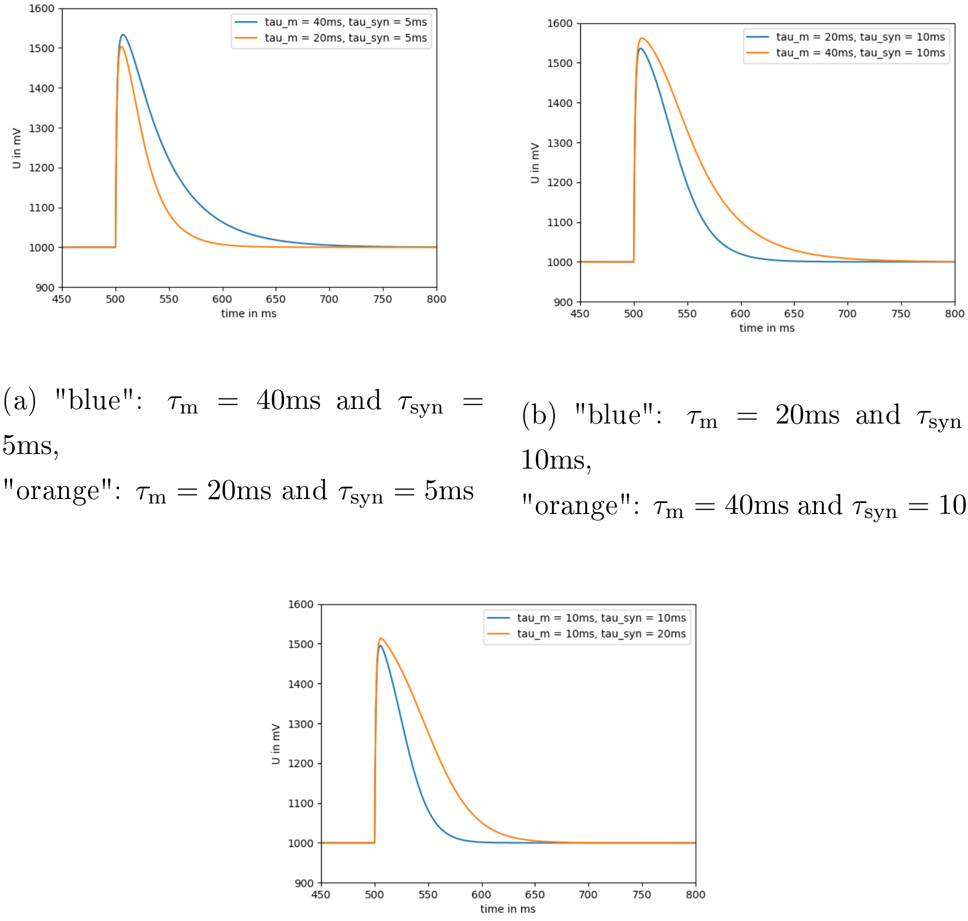
2.1 Impact of Neuron Parameters on the Postsynaptic Potential

In order to get a better understanding of the solution of Eq. (2) we use the simulation software NEST [6]. In the simulation we look at the membrane voltage of a single neuron and are able to analyse the neuron behaviour after the arrival of a spike. With NEST we are able to choose each neuron parameter individually. We can therefore directly observe the impact of changes in the parameters. We will study the effect of the parameters in Eq. (2) i.e. membrane and synaptic time constants τ_{m} and τ_{syn}, resting and reversal potential V_{rest} and E_{syn} and the weight w and membrane capacity C_{m}.

Apart from the shape of the PSP we will look at the height explicitly. The height is easy to measure which is why a reconstruction of the synaptic weight only involving the height and the neuron parameters would be very efficient. For this reason we also simulated the heights of the PSP while changing neuron parameters. While iterating over weight and one chosen neuron parameter the other parameters were fixed at $C_{\mathrm{m}} = 0.2\mu\mathrm{F}$, $V_{\mathrm{rest}} = 1000\mathrm{mV}$, $E_{\mathrm{syn}} = 1600\mathrm{mV}$, $\tau_{\mathrm{m}} = 20\mathrm{ms}$ and $\tau_{\mathrm{syn}} = 5\mathrm{ms}$. However for the iteration over τ_{syn} and τ_{m} both time constants were set to 20 ms for easier comparison of the simulated curves to each other. The dimension of the weight used in the simulation is µS.

We will begin with analysing the effect of the membrane and the synaptic time constant.

2.1.1 Effect of Membrane and Synaptic Time Constant



(a) "blue": $\tau_{\mathrm{m}} = 40\mathrm{ms}$ and $\tau_{\mathrm{syn}} = 5\mathrm{ms}$,
"orange": $\tau_{\mathrm{m}} = 20\mathrm{ms}$ and $\tau_{\mathrm{syn}} = 5\mathrm{ms}$

(b) "blue": $\tau_{\mathrm{m}} = 20\mathrm{ms}$ and $\tau_{\mathrm{syn}} = 10\mathrm{ms}$,
"orange": $\tau_{\mathrm{m}} = 40\mathrm{ms}$ and $\tau_{\mathrm{syn}} = 10\mathrm{ms}$

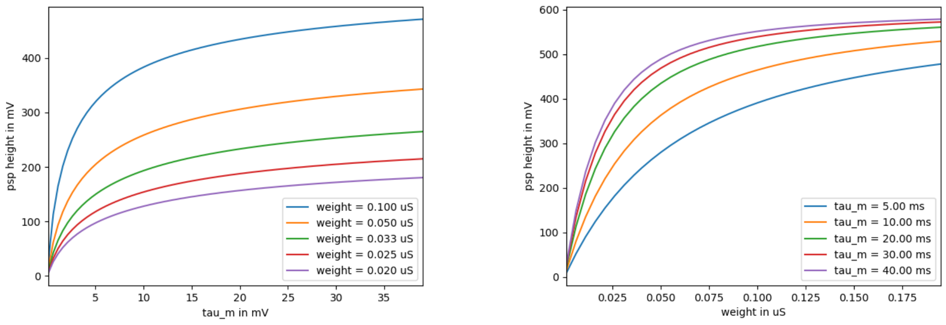
(c) "blue": $\tau_{\mathrm{m}} = 10\mathrm{ms}$ and $\tau_{\mathrm{syn}} = 10\mathrm{ms}$,
"orange": $\tau_m = 10\mathrm{ms}$ and $\tau_{\mathrm{syn}} = 20\mathrm{ms}$

Figure 3: Simulated PSPs with different synaptic and membrane time constants (time constants shown in subcaptions). Other neuron parameters are fixed at $V_{rest} = 1000\mathrm{mV}$, $E_{\mathrm{syn}} = 1600\mathrm{mV}$, $w = 0.8\mu\mathrm{S}$ and $C_{\mathrm{m}} = 1\mu\mathrm{F}$.

We see that a longer membrane time constant leads to a longer PSP (see Figure 3). We also find that the height of the PSP depends on the neuron parameters. The maximum in these examples can be seen in the "orange" curve in Figure 3b with $\tau_{\mathrm{m}} = 40\mathrm{ms}$ and $\tau_{\mathrm{syn}} = 10\mathrm{ms}$. This also makes sense as these are the longest time constants in this example. The current flow seen in Eq. (4) decaying with τ_{syn} charges the membrane over a longer time and the large membrane time constant leads to a slower decay of the charged capacitor. In the "orange" curve in Figure 3a and "blue" curve in Figure 3c we have the smallest pair of time constants. For this reason the height of the PSP is smaller.

We also observe that a longer synaptic time constant also increases the duration of the PSP (see "orange" curve in Figure 3c). Though here the membrane voltage decays quicker than with higher membrane time constant the current flowing on the membrane happens for a longer time. This leads to the longer PSP.

Another interesting result can be seen in the "blue" curve in Figure 3b and "orange" curve inFigure 3c. Here the values of the two time constants were switched. As the course of the PSP is different between the two figures we know that the solution of Eq. (2) is not symmetric in the time constants.

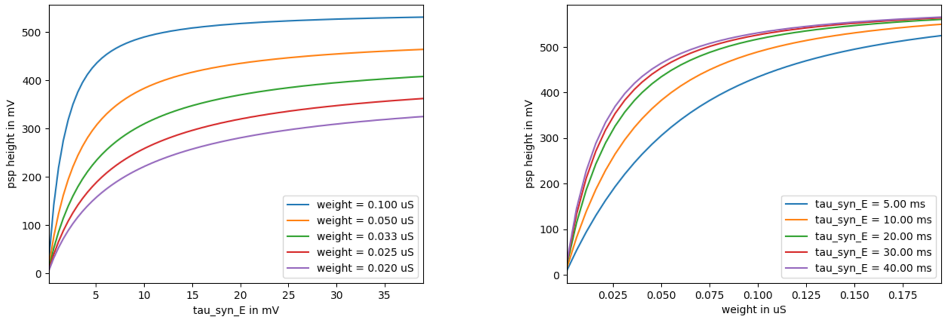


(a) PSP height over τ_{m} with different weight

(b) PSP height over weight with different τ_m

Figure 4: Effect of the neuron parameter τ_{m} on the height of the PSP in simulation.

In Figure 4 and Figure 5 we see that the two time constants have a similar but not identical effect on the PSP height. In Figure 5b we see that the



(a) PSP height over τ_{syn} with different weight

(b) PSP height over weight with different τ_{syn}

Figure 5: Effect of the neuron parameter τ_{syn} on the height of the PSP in simulation.

maximal membrane potential approaches higher values for lower weight than in Figure 4b even though the time constants were altered in a identical way. From these plots we again find that the time constants are not symmetrical and cannot freely be exchanged.

2.1.2 Effect of Resting and Reversal Potential

In Figure 6 we see the PSP with changed resting and reversal potential. While changing the parameters the difference of the two potentials was kept the same. With a change like this we do not see differences in the voltage course but only a offset. In Figure 8 and Figure 7 we find linear relations between the simulated PSP height and the resting and reversal potential. The only difference is the sign of the slope of the two relations. We also see in Figure 8, that the height approaches zero when the resting potential approaches the reversal potential. Because there is no voltage difference, no additional current is flowing onto the membrane. For this reason a spike does not lead to a measurable peak. We find the same effect in Figure 7. Here a height of zero is measured when the reversal potential approaches the resting potential. The explanation for this is identical.

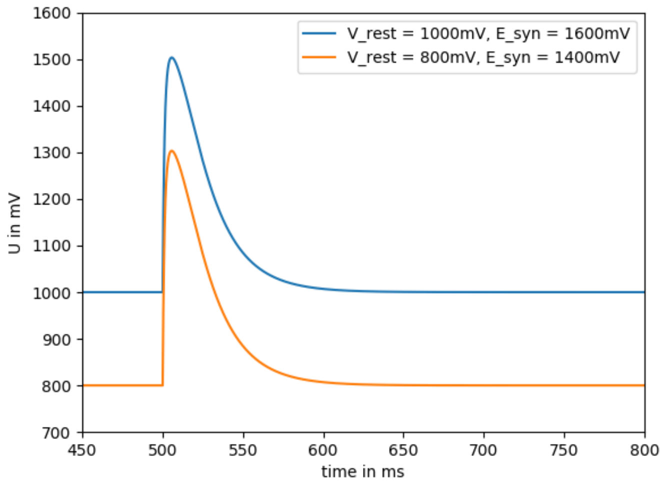
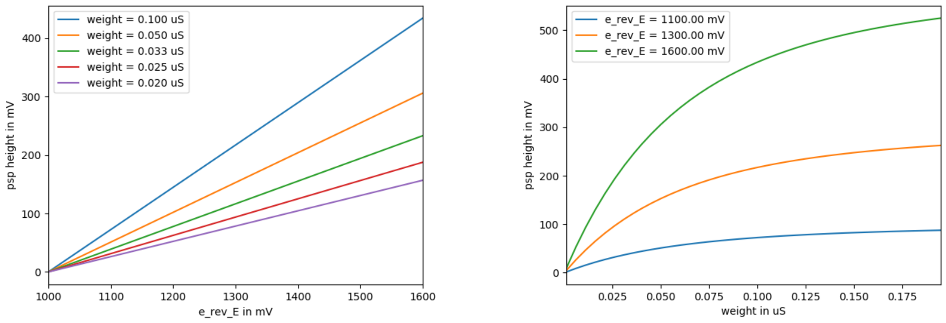


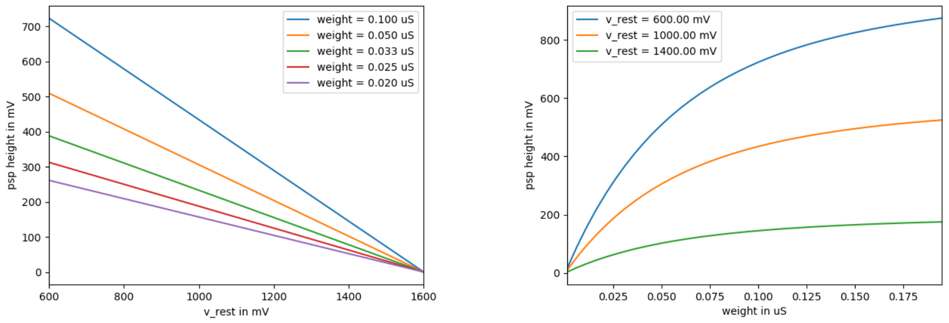
Figure 6: Simulated PSPs with different resting and reversal potential (potentials shown in legend). Other neuron parameters are fixed at $\tau_\mathrm{m} = 20\mathrm{ms}$, $\tau_\mathrm{syn} = 5\mathrm{ms}$, $w = 0.8\mu\mathrm{S}$ and $C_\mathrm{m} = 1\mu\mathrm{F}$.



(a) PSP height over E_syn with different weight

(b) PSP height over weight with different E_syn

Figure 7: Effect of the neuron parameter E_syn on the height of the PSP in simulation.



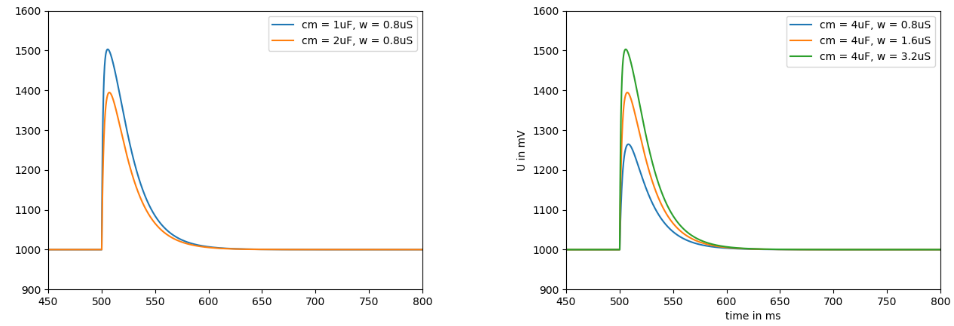
(a) PSP height over V_{rest} with different weight

(b) PSP height over weight with different V_{rest}

Figure 8: Effect of the neuron parameter V_{rest} on the height of the PSP in simulation.

2.1.3 Effect of Weight and Membrane Capacity

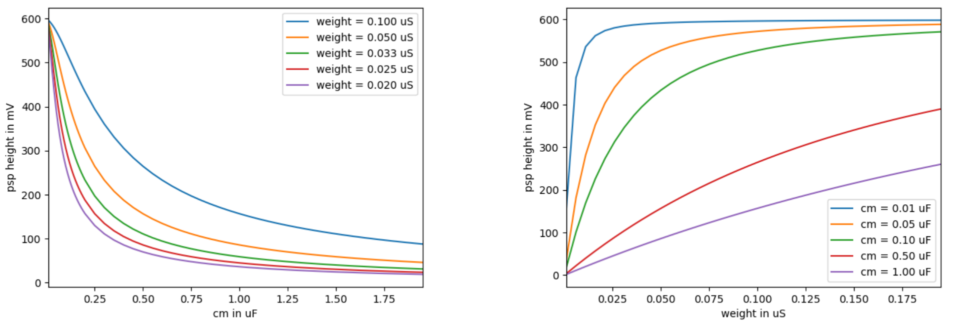
In Figure 9 we see that the membrane capacity and the synaptic weight have the opposite impact on the PSP. We see the same PSP for the "blue" curve in Figure 9a and "green" curve inFigure 9b as well as for the "orange" curve in Figure 9a and "orange" curve in Figure 9b. The reason for this is that the ratio of synaptic weight and capacity is the same for these simulations. As we are not able to measure the membrane capacity of our neuron circuits anyway this motivates us to conduct our reconstruction of the parameters for the ratio of $\frac{w}{C_{\text{m}}}$.



(a) "blue": $C_\mathrm{m} = 1\mu\mathrm{F}$ and $w = 0.8\mu\mathrm{S}$, "orange": $C_\mathrm{m} = 2\mu\mathrm{F}$ and $w = 0.8\mu\mathrm{S}$

(b) "blue": $C_\mathrm{m} = 4\mu\mathrm{F}$ and $w = 0.8\mu\mathrm{S}$, "orange": $C_\mathrm{m} = 4\mu\mathrm{F}$ and $w = 1.6\mu\mathrm{S}$ "green": $C_\mathrm{m} = 4\mu\mathrm{F}$ and $w = 3.2\mu\mathrm{S}$

Figure 9: Simulated PSPs with different synaptic weight and membrane capacitance (weight and capacitance shown in subcaptions). These weights and capacity were arbitrarily chosen. However the $\frac{w}{C_\mathrm{m}}$ chosen is in the order of values found in biology. Other neuron parameters are fixed at $V_\mathrm{rest} = 1000\mathrm{mV}$, $E_\mathrm{syn} = 1600\mathrm{mV}$, $\tau_\mathrm{m} = 20\mathrm{ms}$ and $\tau_\mathrm{syn} = 5\mathrm{ms}$.



(a) PSP height over C_m with different weight

(b) PSP height over weight with different C_m

Figure 10: Effect of the neuron parameter C_m on the height of the PSP in simulation.

2.2 First Approach

In [5] the following equation was fitted on averaged PSPs recorded from multiple measurements with single spikes to calibrate parameters of the synaptic input circuit.

$$V(t) \approx V_{\mathrm{rest}} + \Theta(t_0) \cdot A \cdot (exp\left(\frac{t_0 - t}{\tau_{\mathrm{m}}}\right) - exp(\left(\frac{t_0 - t}{\tau_{\mathrm{syn}}}\right)) \tag{6}$$

Here A is given by

$$A = \frac{w \cdot (E_{\mathrm{syn}} - V_{\mathrm{rest}}) \cdot \tau_{\mathrm{g}}}{C_{\mathrm{m}}}. \tag{7}$$

with τ_{g} given by:

$$\tau_{\mathrm{g}} = (\frac{1}{\tau_{\mathrm{syn}}} - \frac{1}{\tau_{\mathrm{m}}})^{-1} \tag{8}$$

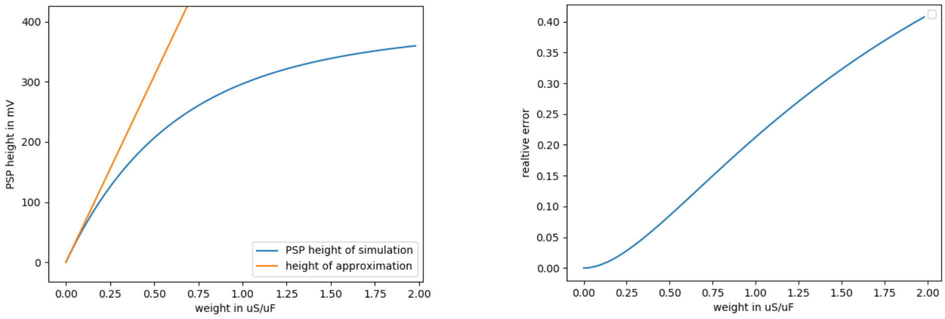
Using the height of the PSP (with $\tau = \frac{\tau_{\mathrm{syn}}}{\tau_{\mathrm{m}}}$) as a parameter given by:

$$h = A \cdot (\tau^{\frac{\tau}{1-\tau}} - \tau^{\frac{1}{1-\tau}}) \tag{9}$$

start parameters were easier to be estimated.

This gives us a first estimation of the synaptic weight only depending on the neuron parameters and the height of the PSP. This approximation only holds for small peaks as we expect a saturation effect from the conductance based model as the membrane voltage approaches the reversal potential. This effect can be seen in Figure 11a. Without a point of failure the approximation begins to deviate from the simulated PSP heights immediately. We see a monotonic rise in the relative error (see Figure 11b). A reconstruction based on this approximation would therefore only be accurate for very low weights. Since we want to reconstruct for a wider range of weights different approaches will be necessary.

In Eq. (6) we also see that the two time constants τ_{m} and τ_{syn} appear in a symmetrical fashion, we however showed in Section 2.1 that this is not the case for the actual voltage course. For small PSP with low weight where this approximation is expected to hold the time constant could have a symmetric effect, however this was not further investigated in this thesis.



(a) height of simulated PSP versus the weight used in simulation with approximation given by Eq. (9)

(b) Relative difference of PSP height from approximation and hardware measurement given by $1 - \frac{h_{\text{simulated}}}{h_{\text{approx}}}$

Figure 11: Comparison of the PSP height from approximation and hardware measurement. Height approximation is qualitative description of quality of approximation. Neuron parameters used were $V_{\text{rest}} = 900\text{mV}$, $E_{\text{syn}} = 1300\text{mV}$, $\tau_{\text{m}} = 20\text{ms}$ and $\tau_{\text{syn}} = 2\text{ms}$.

2.3 Numeric Solution of COBA Differential Equation

As previously mentioned an analytical solution of Eq. (2) is not possible however a lot of research is still done to find approximated solutions to the differential equation. In [7] the equation was solved up to the point of containing a gamma function. This equation accurately describes the voltage course of a conductance based LIF neuron after a spike arrives at t=0. However the Gamma function still does not allow an analytical solution.

$$\begin{aligned} V(t) =& \exp(-\frac{t}{\tau_{\text{m}}^{\text{l}}} + \frac{\tau_{\text{s}}}{\Delta\tau_{\text{m}}^{\text{s}}} e^{-\frac{t}{\tau_{\text{s}}}}) \cdot \Big(E_{\text{l}} e^{-\frac{\tau_{\text{s}}}{\Delta\tau_{\text{m}}^{\text{s}}}} - E_{\text{s}}(e^{-\frac{\tau_{\text{s}}}{\Delta\tau_{\text{s}}^{\text{m}}}} - \exp[\frac{t}{\tau_{\text{m}}^{\text{l}}} - \frac{\tau_{\text{s}}}{\Delta\tau_{\text{m}}^{\text{s}}} e^{-\frac{t}{\tau_{\text{s}}}}]) \\ & - \Gamma[-\frac{\tau_{\text{s}}}{\tau_{\text{m}}^{\text{l}}}, \frac{\tau_{\text{s}}}{\Delta\tau_{\text{m}}^{\text{s}}} e^{-\frac{t}{\tau_{\text{s}}}}, \frac{\tau_{\text{s}}}{\Delta\tau_{\text{m}}^{\text{s}}}](\frac{\tau_{\text{s}}}{\Delta\tau_{\text{m}}^{\text{s}}})^{\frac{\tau_{\text{s}}}{\tau_{\text{m}}^{\text{l}}}} \frac{\tau_{\text{s}}}{\tau_{\text{m}}^{\text{l}}} (E_{\text{s}} - E_{\text{l}}) \Big) \end{aligned} \tag{10}$$

Here $\Delta\tau_{\text{m}}^{\text{s}} \widehat{=} \frac{w}{C_{\text{m}}}$, $\tau_{\text{m}}^{\text{l}} \widehat{=} \tau_{\text{m}}$ and $\tau_{\text{s}} \widehat{=} \tau_{\text{syn}}$. The Gamma function with three arguments is the incomplete Gamma function defined by

$$\Gamma(a, x, y) = \int_{\text{x}}^{\text{y}} t^{\text{a}-1} e^{-t} dt \tag{11}$$

In [8] this equation was further approximated.

$$\begin{aligned} V(t) =& V_{\mathrm{L}} + (V_0 - V_{\mathrm{L}}) e^{(-t-t_0)(\frac{1}{\tau_{\mathrm{m}}^{\mathrm{L}}} + \frac{1}{\tau_{\mathrm{m}}^{\mathrm{e}}(t_0)} + \frac{1}{\tau_{\mathrm{m}}^{\mathrm{i}}(t_0)})} \\ &+ \frac{V_{\mathrm{e}} - V_{\mathrm{L}}}{\tau_{\mathrm{m}}^{\mathrm{e}}(t_0)} (\frac{1}{\tau_{\mathrm{m}}^{\mathrm{L}}} + \frac{1}{\tau_{\mathrm{m}}^{\mathrm{e}}(t_0)} + \frac{1}{\tau_{\mathrm{m}}^{\mathrm{i}}(t_0)} - \frac{1}{\tau_{\mathrm{e}}})^{-1} \\ &\cdot [e^{-\frac{t-t_0}{\tau_{\mathrm{e}}}} - e^{-(t-t_0)(\frac{1}{\tau_{\mathrm{m}}^{\mathrm{L}}} + \frac{1}{\tau_{\mathrm{m}}^{\mathrm{e}}(t_0)} + \frac{1}{\tau_{\mathrm{m}}^{\mathrm{i}}(t_0)})}] \end{aligned} \tag{12}$$

The formalism is a little bit different $\tau_{\mathrm{m}}^{\mathrm{e}}(t_0)$ is the time constant resulting from the synaptic conductance and the membrane capacity. It is therefore given by $\tau_{\mathrm{m}}^{\mathrm{e}}(t_0) \widehat{=} \frac{w}{C_{\mathrm{m}}}$. τ_{e} is the synaptic time constant τ_{syn} and $\tau_{\mathrm{m}}^{\mathrm{L}}$ is the membrane time constant τ_{m}. In Eq. (12) the exponential in the Gamma function was approximated with the first order Taylor expansion. The accuracy of the approximation therefore quickly decreases and the approximation only describes the voltage course for times much smaller than the synaptic time constant. In Figure 12 the approximated PSPs with identical neuron parameters and weight were plotted. We immediately see that the approximations deviate from the simulation. We therefore will not be able to use any approximation for our weight reconstruction and have to use the numeric solution if we want to fit any measured or simulated data.

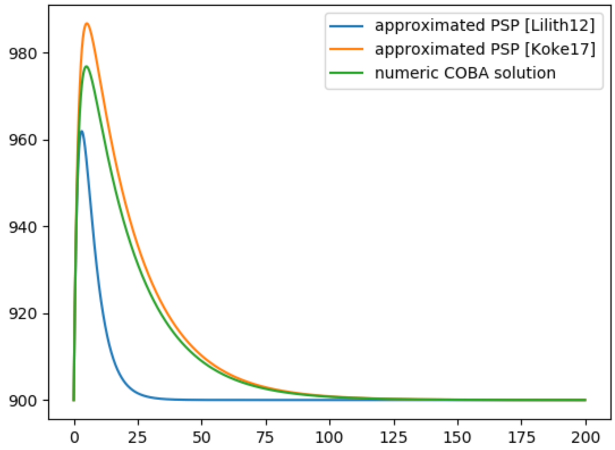
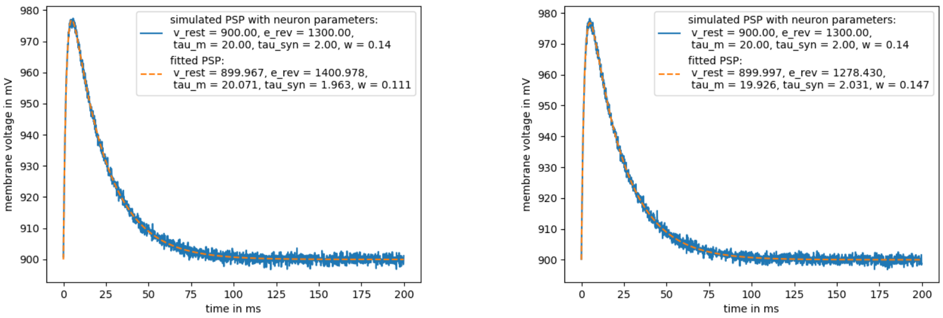


Figure 12: Comparison of the mentioned approximations of the neuron behaviour. Chosen neuron parameters were $V_{\text{rest}} = 900\text{mV}$, $E_{\text{syn}} = 1300\text{mV}$, $\tau_{\text{m}} = 20\text{ms}$, $\tau_{\text{syn}} = 2\text{ms}$ and $w = 0.14\frac{\text{µS}}{\text{µF}}$. We see that the approximation from [5] describes the course better then the course from [8].

2.4 Fitting the Numeric Solution and First Hardware Measurement

2.4.1 Fitting of Simulation Data

We use Eq. (10) to fit simulation data from NEST. We find that fitting Eq. (10) using the python module mpmath to numerically solve the incomplete Gamma function takes a long time (order of seconds). This as a problem as we want to measure the weight for a large number of neurons and a wide range of parameters. Using numerical methods to solve the differential equation from Eq. (2) directly, proved to be faster. For this the odeint function from the python module scipy [9] was used. We found that the fitted function describes the simulation data very well (χ^2_{red} in the order of 1e-8, which can be explained by numerical errors) and is able to reconstruct the used parameters for the simulation from the voltage course. When introducing some artificial Gaussian noise we are still able to reconstruct the data (σ of 0.1mV). However with increasing noise the reconstructed weight and reversal potential deviate from the used parameters for the simulation (see Figure 13, Gaussian noise with σ of 1.0mV). This will be a problem when we later want to reconstruct the weight from the measured PSPs as the measured PSPs also contain noise (see Figure 15).

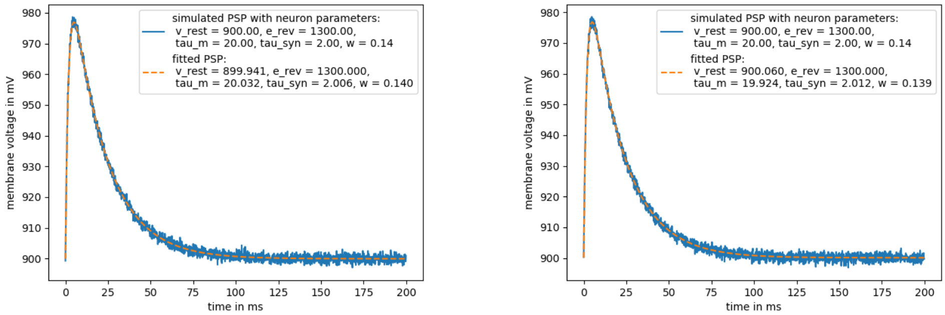


(a) Fitted parameters: $V_{\text{rest}} = 900.0\text{mV}$, $E_{\text{syn}} = 1401.0\text{mV}$, $\tau_m = 20.01\text{ms}$, $\tau_{\text{syn}} = 1.96\text{ms}$ and $w = 0.111 S/F$

(b) Fitted parameters: $V_{\text{rest}} = 900\text{mV}$, $E_{\text{syn}} = 1278.4\text{mV}$, $\tau_m = 19.93\text{ms}$, $\tau_{\text{syn}} = 2.03 s$ and $w = .147 S/F$

Figure 13: Fitted PSPs with some random gaussian noise and free reversal potential (different noise for different plots). Setup parameters: $V_{\text{rest}} = 900\text{mV}$, $E_{\text{syn}} = 1300\text{mV}$, $\tau_m = 20\text{ms}$, $\tau_{\text{syn}} = 2\text{ms}$ and $w = 0.140 S/F$

With the noise the fit is not able to differentiate between a increasing weight or a increasing reversal potential. As can be seen in Figure 13 the fitted reversal potential varies here from 1280mV to 1400mV. The weight varies from 0.111S/F to 0.147S/F. Higher variations are also possible depending on the noise. We will therefore need to keep the reversal potential fixed to be able to reconstruct the synaptic weight (done in Figure 14). As can be seen the weight is reconstructed correctly with the fixed reversal potential. To get an accurate weight however we will need to find other ways to fix the reversal potential to its exact value for each neuron. For the first fitted measurements



(a) Fitted parameters: $V_{\text{rest}} = 899.9\text{mV}$, $E_{\text{syn}} = 1300\text{mV}$, $\tau_m = 20.03\text{ms}$, $\tau_{\text{syn}} = 2.01\text{ms}$ and $w = 0.140S/F$

(b) Fitted parameters: $V_{\text{rest}} = 900.1\text{mV}$, $E_{\text{syn}} = 1300\text{mV}$, $\tau_m = 19.92\text{ms}$, $\tau_{\text{syn}} = 2.01\text{ms}$ and $w = 0.139S/F$

Figure 14: Fitted PSPs with some random Gaussian noise and fixed reversal potential (different noise for different plots). Setup parameters: $V_{\text{rest}} = 900\text{mV}$, $E_{\text{syn}} = 1300\text{mV}$, $\tau_m = 20\text{ms}$, $\tau_{\text{syn}} = 2\text{ms}$ and $w = 0.140\text{S/F}$

we will just use the setup reversal potential parameter. This way variations in the reversal potential between neurons appear as additional variations in the weight.

2.4.2 First Measurement on Hardware

We now look into the first measurements on hardware. Here we varied neuron parameters and compared these to the fitted parameters. The measurement was conducted on wafer 33 HICANN 297. In the measurement routine multiple PSPs will be measured. A averaged PSP from these measurements will then be returned for each neuron and step of the bit-weight. This way the noise is reduced to the values seen in the plots (e.g. Figure 15). Otherwise the membrane voltage underlies a noise with a spread of $\sigma_{\text{mem}} \approx 5\text{mV}$.
In Figure 17, Figure 18 and Figure 21 we see corresponding fitted neuron parameter plotted at differently set 4-bit weights. The used neuron parameters were $V_{\text{rest}} = .8\text{V}$, $E_{\text{syn}} = 1.2\text{V}$ and $g_{\text{div}} = 30$. Also two values for V_{gmax} were

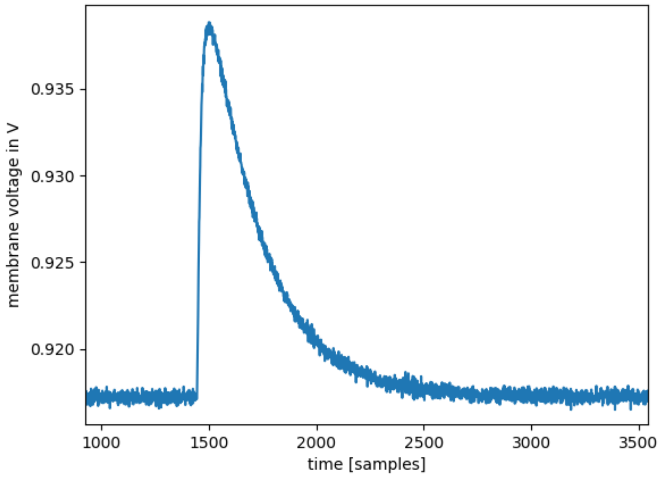


Figure 15: Measurement of Neuron 1 on wafer 33 HICANN 297. The chosen parameters were $V_{\text{rest}} = 900\text{mV}$, $E_{\text{syn}} = 1300\text{mV}$, $\tau_{\text{m}} = 20\text{e} - 7\text{s}$, $\tau_{\text{syn}} = 2\text{e} - 7\text{s}$, $g_{\text{div}} = 30$, $V_{\text{gmax}} = .9\text{V}$ and bit-weight 5. The relative noise is comparable to the noise seen in Figure 13b. For this reason we will have problems when fitting hardware measurements as the noise will impact our fit as seen in simulation.

used, firstly V_{gmax} was set at .9V and later the lowest weight of $V_{\text{gmax}} = 0.0\text{V}$ was used. In the plots if no V_{gmax} is mentioned it is set at .9V and if a low V_{gmax} is mentioned it is set as 0.0V.

In Figure 17 we see that the fitted membrane time constant is approximately constant for all bit weights except for the "green" measurement with $\tau_{\text{syn}} = 5\text{e} - 7\text{s}, \tau_{\text{m}} = 40\text{e} - 7\text{s}$ and the "orange" measurement with $\tau_{\text{syn}} = 10e - 7\text{s}, \tau_{\text{m}} = 20\text{e} - 7\text{s}$. A fitted time constant varying over different weights indicates a deformation of the PSP (e.g. by saturation of the OTA translating the integrated voltage to the bias current to the resistive element between membrane and reversal potential, see Figure 16), another hint for this is the increasing χ^2_{red} in Figure 22. However we cannot be certain as we do not have access to the integrator voltage and therefore cannot observe this effect directly. For the lower weights ($V_{\mathrm{gmax}} = 0.0$V) we see no change in the

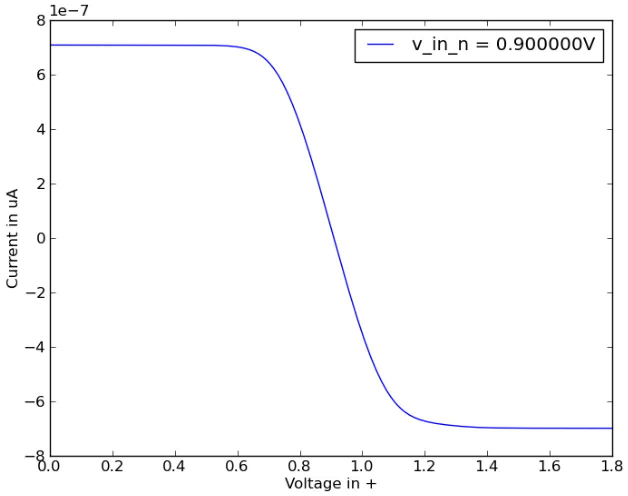


Figure 16: Voltage current characteristic of the OTA with a voltage of .9V at the negative Pin. See a roughly linear course for a small range (good accuracy for $V_+ = .9 \pm .1$V) around .9V for the positive pin and then a saturation. This could explain a deformation of PSPs for higher weight (i.e. higher integrator voltage to the OTA). Simulation done using cadence circuit simulation.

fitted time constant as e.g. the OTA probably does not saturate. This effect can also be observed in Figure 19 where we compare two neuron fits with a $V_{\mathrm{gmax}} = 0.0$V and $V_{\mathrm{gmax}} = 0.9$V. We also expect a higher synaptic time constant to have a similar effect, as the voltage in the integrator does not decay as fast, it is more likely for the OTA to saturate for higher synaptic time constant. We also see quite a spread between measurements with the same setup parameter and low weight ("red" and "purple"). One reason for this could be trial to trial variations however for the lower bit weight the "blue" and the "purple" curve are very close to each other, which could hint at an impact of the fit depending on the synaptic time constant. However we did not investigate this further.

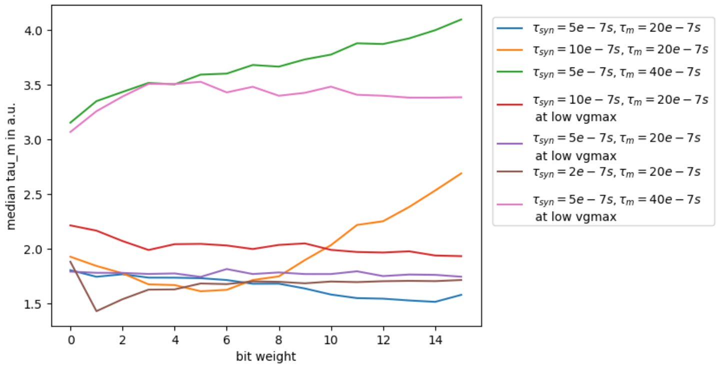


Figure 17: Median of fitted Membrane time in e-6s constant at different 4-bit weight for measurement runs with different neuron parameters, all 512 neurons on HICANN used for median. Lower synaptic time-constants and lower weight (low $V_{\text{gmax}} = 0.0\text{V}$) lead to more weight independent membrane time constant. Also lower membrane time constants (here 20e-7s) are better realized on hardware.

In Figure 18 we see a similar but even stronger effect. We see that the fitted synaptic time constants of measurements with $V_{\text{gmax}} = .9\text{V}$ all seem to increase with increasing weight. The size of the increase seems to depend on the time constant itself (smallest synaptic time constant of 2e-7 has the smallest deviation for the highest bit weight, contrary biggest time constant of 10e-7 even starts to deviate for smaller bit-weights). The deformation by e.g. OTA saturation therefore seems to impact the synaptic time constant fit even more. However we also see that the spread between different measurements seems to be much smaller for low weight and same setup synaptic time constant ("purple" and "pink") compared to the one in Figure 17. The membrane time constant therefore seems to have a smaller impact on the synaptic time constant fit than vice versa.

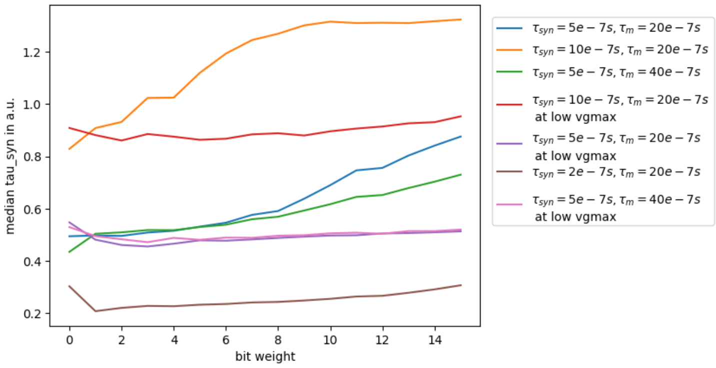


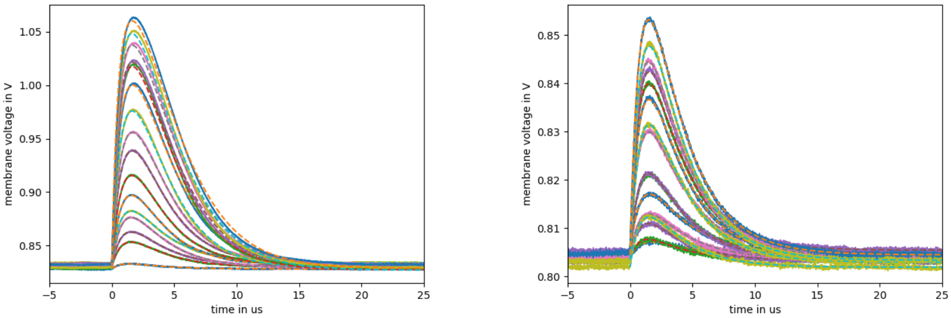
Figure 18: Median of fitted synaptic time constant at different 4-bit weight for measurement runs with different neuron parameters, all 512 neurons on HICANN used for median.

Next we look at our parameter of interest, the synaptic weight. We will first look at the measurements with the V_{gmax} of .9V. As expected we see a roughly linear increase in weight when increasing the 4-bit weight (see Figure 21). However we also see some nonlinear behaviour identical for all of the different measurement runs. This nonlinear increase was shown in [5] in hardware simulation. There formula 5.5) (see Eq. (13)) describes a increase of the synaptic weight depending on each bit individually.

$$I_{\text{syn}}(V_{\text{gmax}}, g_{\text{div}}, w) = V_{\text{gmax}} \cdot g_{\text{scale}} \cdot \frac{w}{g_{\text{div}}} + i_0 + i_1 w_1 + i_2 w_2 + i_4 w_4 + i_8 w_8 \quad (13)$$

As the 4-bit weight activates different transistors (see Figure 2), a parasytic charging effect leads to the observed nonlinearity. Characteristically for a effect depending on the individual bits are the observed dips at the bit weights 4, 8 and 12 which is when many of the transistors are switched on or off.

We also see that the fitted weights differ for different measurement runs. This effect seems to depend on the synaptic time constant. The higher the used time constant the higher the fitted weight. Same time constants lead to nearly identical fitted weights. Again we suspect a deformation of the PSP that alters the fitted weight. To understand how the different parameters impact the quality of the fit we look at the χ^2_{red} values of the fit. As we did not take



(a) Fitted trace of Neuron 1 on wafer 33 HICANN 0. Chosen parameters were $V_{\text{rest}} = 800\text{mV}$, $E_{\text{syn}} = 1200\text{mV}$, $\tau_m = 20\text{e} - 7\text{s}$, $\tau_{\text{syn}} = 10\text{e} - 7\text{s}$, $g_{\text{div}} = 30$, $V_{\text{gmax}} = 0.9\text{V}$

(b) Fitted trace of Neuron 1 on wafer 33 HICANN 0. Chosen parameters were $V_{\text{rest}} = 800\text{mV}$, $E_{\text{syn}} = 1200\text{mV}$, $\tau_m = 20\text{e} - 7\text{s}$, $\tau_{\text{syn}} = 10\text{e} - 7\text{s}$, $g_{\text{div}} = 30$, $V_{\text{gmax}} = 0.0\text{V}$

Figure 19: Fitted Hardware traces of the measurement with $\tau_m = 20\text{e} - 7\text{s}$ and $\tau_{\text{syn}} = 10\text{e} - 7\text{s}$. We see the effect, that the fit describes the measurement less accurately for high weights in Figure 19a. This explains the change in reconstructed parameters seen in Figure 17 and Figure 18.

any errors into account we are not able to draw any quantitative conclusions from this graph. However we clearly see that the fits with higher synaptic time constant deviate from the measurement for lower weights ("orange" reaches $\chi^2_{\text{red}} = 4\text{e} - 5$, "blue" reaches $\chi^2_{\text{red}} = 4\text{e} - 6$ and "green" reaches $\chi^2_{\text{red}} = 1.6\text{e} - 6$). We also see that again the low weight V_{gmax} measurements are better fitted by the used model (this leads us to believe that saturation of the synaptic input circuit leads to deformation of the PSPs). We also see that the lowest synaptic time constant of $\tau_{\text{syn}} = 2e - 7\text{s}$ is best fitted for the measurements with the higher weight. For this reason we will use this time constant for following measurements.

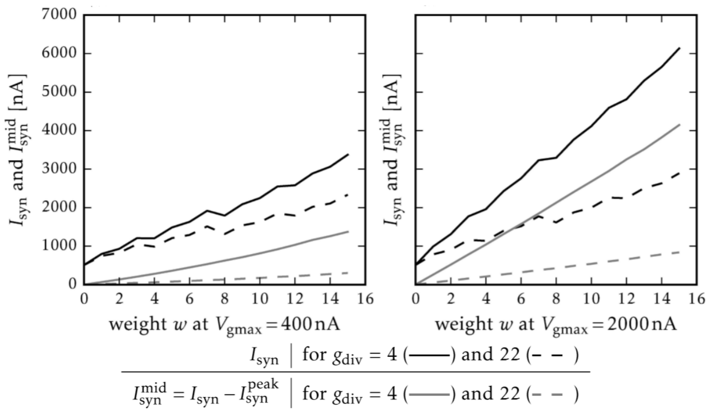


Figure 20: Simulated current I_{syn} in hardware simulation with different setup V_{gmax} and g_{div}. Similar non linear behaviour with changing bit weights was reproduced when fitting the PSPs from the measurement. Figure from [5]

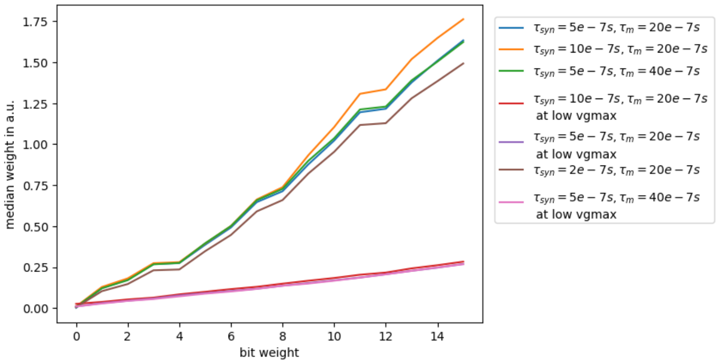


Figure 21: Median of fitted weight at different 4-bit weight for measurement runs with different neuron parameters, all 512 neurons on HICANN used for median. Reproduced the nonlinear weight effect seen in Figure 20.

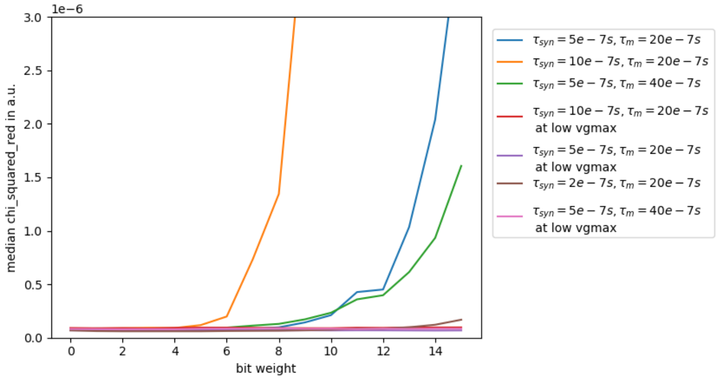
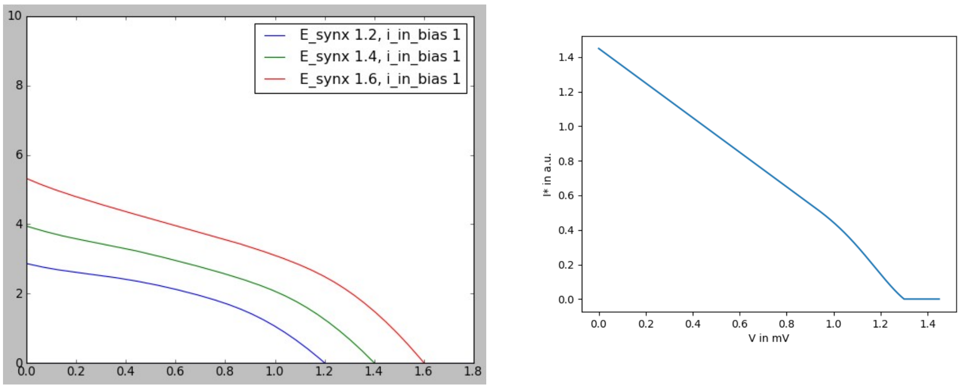


Figure 22: χ^2_{red} of the fits at different 4-bit weight for measurement runs with different neuron parameters. The higher synaptic time constants seems to have a big impact on the quality of the fit which hints at saturation in the synaptic input circuit.

2.5 Improving the Fitting

As mentioned before we were not able to fit the reversal potential, as the fit could not differentiate between an increasing weight and an increasing reversal potential. We therefore want to identify the used reversal potential and fix it at that value to make the measured weights comparable for different neurons. If the used reversal potential is not correct, the fits quality will decrease and we will also calculate weights which would be higher or lower than they actually are.

In hardware simulation another effect was found which we will need to account for to get a working and accurate weight measurement. In Figure 23a we see that the current voltage characteristic of the resistive element is not linear. Instead we see a linear decline (as expected for a classical resistor) and then a sharp decline as the potential of the membrane approaches the setup reversal potential. The COBA LIF equation (Eq. (2)) contains a conductance independent of the voltage difference. Our fitted function therefore can only describe a PSP whose voltages are always in the linear regime of the current voltage characteristic. We can do this by conducting our measurements further away from the reversal potential using a lower resting potential and choosing our neuron parameters to get a smaller height of the PSP (e.g. small synaptic and membrane time constants). However we still need to adjust the reversal potential as the setup reversal potential is calibrated by measuring the maximum voltage reached of the membrane. In Figure 23a the calibration would lead to measured reversal potentials of 1.6V ("red"), 1.4V ("green") and 1.2V ("blue"). These voltages however do not describe the effective reversal potential which is described by the continuation and subsequent cutting of the linear range and the x-axis. This would be the course of a classical resistor with voltage independent conductance.



(a) Course of the current voltage characteristic of the resistive element between the membrane and the reversal potential at fixed bias current for different reversal potentials. Done with cadence circuit simulation.

(b) Additional Conductance dependence added to Eq. (2). This way we are able to reproduce the current voltage characteristic seen in hardware simulation. Done by adding a forth power term to the conductance term in Eq. (2).

Figure 23: Comparison of the current voltage characteristics of the hardware simulation and an approximation in Python. This way we want to simulate the effect the nonlinearity has on the measured PSPs.

The only observable we have access to is the membrane voltage. For this reason we cannot reproduce Figure 23a on hardware. To look for the from now on called effective reversal potential (i.e. value of potential that determines the course of the PSP in case that the voltages do not enter a range were the conductance behaves nonlinearly) we remind ourselves of Figure 8a. There we saw that the height of the PSP approaches 0 as the resting potential approaches the reversal potential. As only the height of the PSPs is plotted we are able to reproduce this plot on hardware.

To first look at what we expect we alter our differential equation Eq. (2) by adding an additional factor to the conductance. This factor depends on the difference of the membrane potential and the reversal potential. In our custom simulation (see Figure 23b) we chose a forth power term to approximate the behaviour seen in Figure 23a. This is just an approximation and only qualitatively describes the hardware simulation. We are now able to simulate the PSPs by solving our new differential equation. We chose an effective reversal potential of 1.45 V and a reversal potential of 1.3 V. These parameters were chosen this way as we used a reversal potential of 1.3 V for our following measurements and found that the effective reversal potential was in the range of 1.45V. The bend of the curve was chosen to start at .9 V, this was motivated by what we saw in Figure 23a.

In Figure 24 we see the simulated height of the solution to our altered differential equation over the chosen resting potential. We see that the linear approximation on the first simulated points crosses the x-axis at the effective reversal potential. This was done by using the scipy odeint function which we also use to fit the measurements. This motivates us to use a measurement run like this to find the effective reversal potential of our neurons to better fit the data and gather a weight independent of the resting potential.

Due to a power outage wafer 33 was not usable for this part of the experiment. For this reason we switched to wafer 30 HICANN 0 for our measurement.

To gather the data we had to shut off the connections to inhibitory synapses by setting the bias current of the OTAs in the inhibitory synaptic input circuits to zero. Without setting the OTA bias to zero the resistive elements between the membrane and the inhibitory reversal potential do not reach a conductance of zero even when no synaptic input is present (inhibitory spikes to the neuron). This leads to a leak current pulling the membrane potential towards the reversal potential. The value of the potential in the case of no

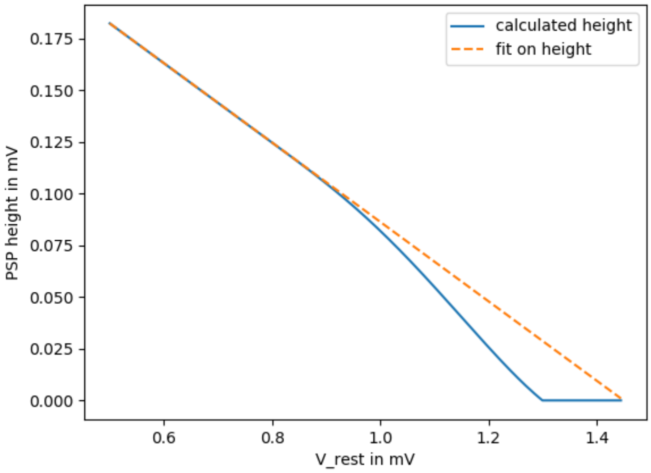
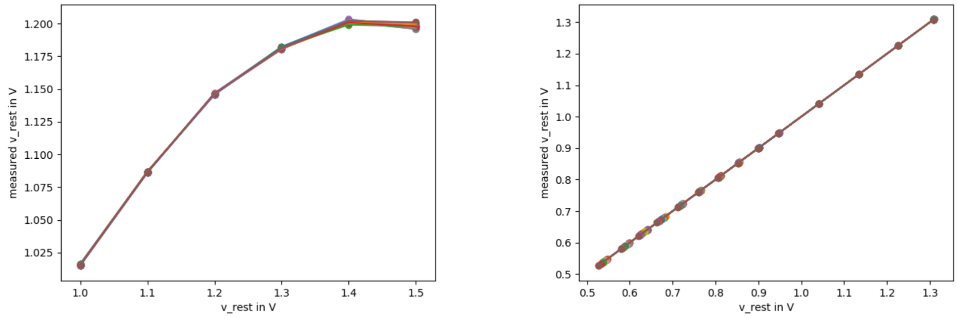


Figure 24: Expected course of the height of the PSP over the used resting potential with the nonlinearity of the resistive element added to Eq. (2). The linear continuation for lower V_{rest} cuts the x-axis at the effective reversal potential.

synaptic input is therefore not only influenced by the setup leakage potential but also the reversal potentials. This effect is stronger for higher potentials as the current flowing from the membrane to the reversal potential increases with the voltage difference. Since we do not need the inhibitory synaptic input circuit for our measurement we are easily able to eliminate this error by switching the inhibitory input off completely which sets the conductance of the inhibitory resistive element to zero. This way the inhibitory reversal potential was not pulling the membrane (i.e. resting potential) to lower values (see Figure 25b). Without turning of the inhibitory input we are not able to achieve resting potentials close to our reversal potential of 1.3V (see Figure 25a).

We were able to replicate the expected results on hardware as can be seen in Figure 26. We see a similar course of the height as we expected from Figure 24. The small discrepancies from measurement and simulation are



(a) Fitted resting potential versus setup resting potential with inhibitory synaptic input turned on

(b) Fitted resting potential versus setup resting potential with inhibitory synaptic input turned off

Figure 25: Measured resting potential versus the setup resting potential with the inhibitory synaptic input turned off and on. Without disabling the inhibitory synaptic input the inhibitory reversal potential pulls the membrane down, the membrane does not reach the desired value.

probably caused by our chosen nonlinear conductance model and differently chosen start of bend and effective reversal potential. With this measurement we are able to reconstruct the effective reversal potential by fitting on the points where we did not see any bend in the measured height. To get a better approximation we averaged over all the weight settings. This leads to the following distribution of effective reversal potentials (see Figure 27). This gives us an average effective reversal potential of $E\_rev\ast = (1.48 \pm 0.16)\text{V}$ (the star denotes the effective reversal potential contrary to the setup reversal potential). The effective reversal potential is therefore higher than the setup reversal potential and the cross of our curves in Figure 26 with the x-axis. This proves our expectations correct. To now test if our calculated effective reversal potential is the reversal potential affecting the membrane we look at different measurement runs with changing resting potential. In this measurement run we kept all other neuron parameters fixed $(E_{\text{syn}} = 1.3\text{V}, \tau_{\text{syn}} = 2e - 7\text{s}, \tau_{\text{m}} = 20\text{e} - 7\text{s}, g_{\text{div}} = 30, V_{\text{gmax}} = .9\text{V})$. We iterate over the resting potential from .4V to 1.3V. The measurements are

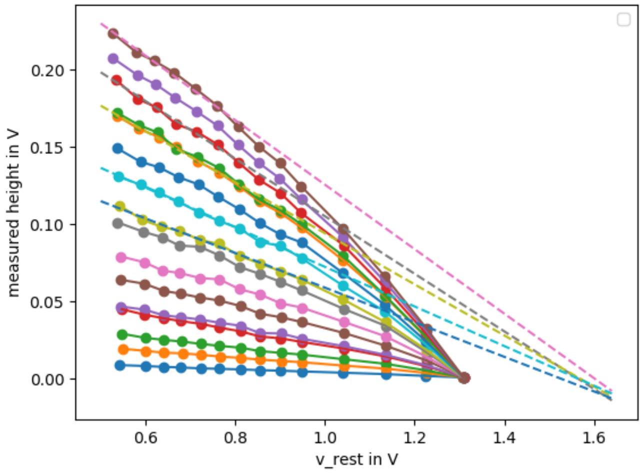


Figure 26: Median PSP height over the used resting potential in hardware measurements. Each dot indicates a measurement run. The different coloured lines are different chosen 4-bit weight. For some weights the linear approximation to the first data points is plotted. They cut the x-axis at the effective reversal potential.

then fitted two different ways. First we will fit the run with our fixed setup reversal potential, afterwards we will fit with the calculated effective reversal potential for each neuron. Simulation results can be seen in Figure 28. We hope to find a from the resting potential independent weight for our calculated effective reversal potential (see Figure 28b) and an increase in the fitted weight when using the setup reversal potential (see Figure 28a). The wiggles seen in Figure 28a are numerical errors, which arise from finite stepsizes when solving the differential equation. With different simulation software these could be reduced.

In Figure 29 we see the comparison of the two ways of fitting. Figure 29b shows the fit with the calculated effective reversal potential. The weight is approximately constant in the range of the resting potential where the

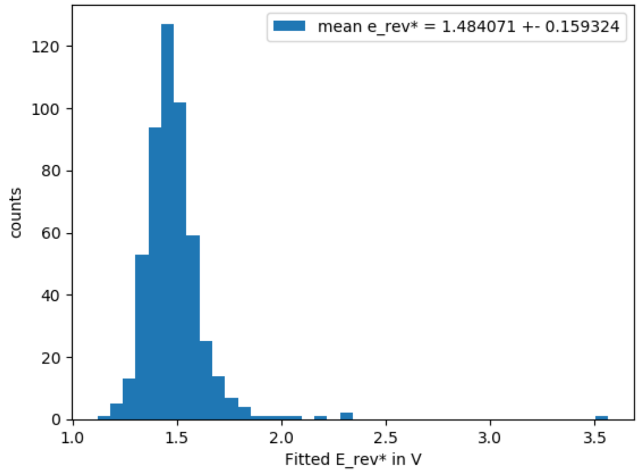
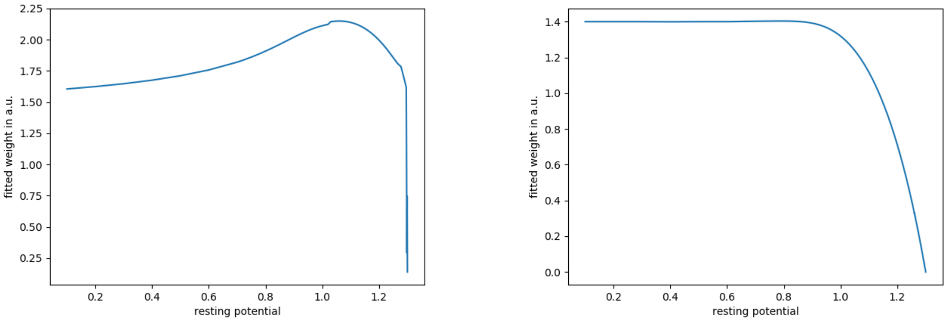


Figure 27: Histogram of the effective reversal potentials reconstructed by fitting Figure 26 for each neuron individually. The effective reversal potential is higher than the setup reversal potential as expected from the python simulation.

conductance does not yet bend. Only for the higher resting potentials the fitted weight deviates from this constant value as the fit does not describe the data accurately due to the bend in the conductance. However we saw the same effect in the simulation with the artificial bend in the conductance (Figure 28b). In Figure 29a we see that the fitted weight is not constant but instead monotonically rising. Again this was expected after looking at the simulation in Figure 28a. We therefore seem to be able to eliminate the effect of the bending conductance in the linear regime and should be able to reconstruct the synaptic weight from our fit.



(a) Fitted weight versus fitted resting potential. Used setup reversal potential for the fit. Fitted on simulated data with the above nonlinear conductance

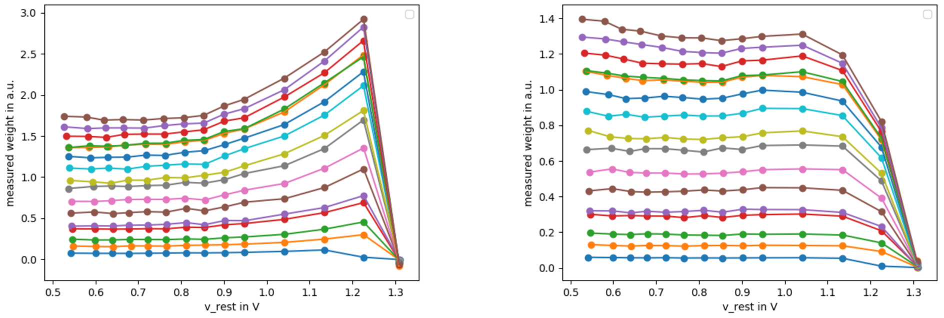
(b) Fitted weight versus fitted resting potential. Used effective reversal potential for fit. Fitted on simulated data with the above nonlinear conductance

Figure 28: Fitted weight of our custom simulation.

2.6 Results Improved Fitting

We now analyze the measurements conducted to find the effective reversal potential, which we will fit again with the new fixed reversal potential. From Figure 17 and Figure 18 we expect roughly constant time constants for the different weights, since we used the best parameters found of $\tau_{\mathrm{m}} = 20e - 7\mathrm{s}$ and $\tau_{\mathrm{syn}} = 2e - 7\mathrm{s}$. Also we expect that the chosen resting potential has no impact on the other fitted neuron parameters (we already saw that the weight was independent of V_{rest} in Figure 29b). However we find that the neuron parameters are not fitted as expected.

In Figure 30 we can see that the synaptic time constant varies for different weights. This was expected as the different weights lead to different saturation of the OTA and differently deformed PSPs. However we also see that the variation for different weights varies depending on the used resting potential. For higher resting potentials we expect a variation, as the conductance begins to bend as the membrane voltage approaches the reversal potential and the differential equation no longer accurately describes the behaviour.



(a) Median of fitted weight versus fitted resting potential. Used setup reversal potential for the fit. Result roughly follows the expectations from Figure 28a.

(b) Median of fitted weight versus fitted resting potential. Used effective reversal potential for fit. Result roughly follows the expectations from Figure 28b.

Figure 29: Comparison of the different fits. Figure 29b with calculated effective reversal potential, Figure 29a with setup reversal potential. The weight is constant when the effective reversal potential is used otherwise the fitted weight increases as the resting potential approaches the reversal potential. Wiggles in the plot are the result from finite step sizes in the numeric solution of our expanded differential equation.

On the other hand the different course for lower resting potential cannot be explained this way.

In Figure 31 we also see a variation of the time constant with increasing weight. This behaviour was not seen in Figure 17. However even more interesting is the increase in the membrane time constant for lower resting potential (i.e. larger difference of resting and reversal potential). Only for resting potentials which approach the reversal potential the fitted synaptic time constant approaches the setup time constant of $\tau_{\mathrm{m}} = 20e-7\mathrm{s}$. It is also interesting that this effect seems to be only slightly impacted by the chosen weight. Even for a bit-weight of 0 ("blue" curve) the rise in the time constant is significant. This hints, that this is not an effect of the synaptic input circuit but might be another non constant conductance effect of the resistive

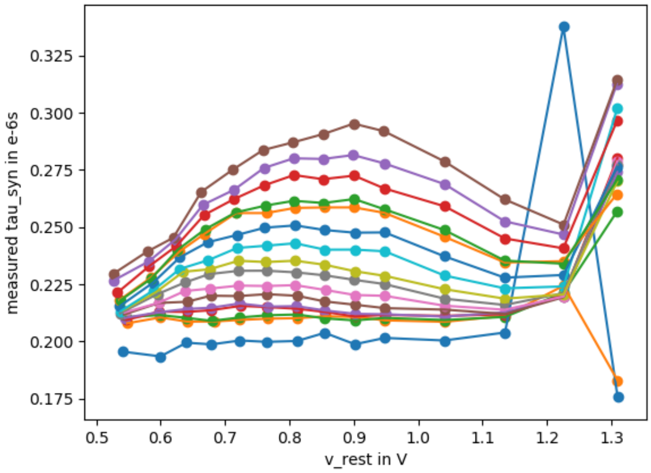


Figure 30: Fitted median synaptic time constant with used effective reversal potential for Fit. Different curves represent different bit weights. See constant synaptic time constant for low weight ("blue") as expected. Spread of time constant at fixed resting potential probably caused by saturation and corresponding deformation of PSP. Weird behaviour for high resting potential can be explained by the small PSPs. These lead to failing or inaccurate Fits as the PSP starts to disappear in the noise of the measurement.

element which is used to set up the membrane time constant. Analysing the circuit to understand this effect was out of the scope of this thesis.
When looking at the χ^2_{red} values of the fits we see that the fit function does not describe the data as accurately for lower resting potentials (maximum of χ^2_{red} at $V_{\mathrm{rest}} = .7\mathrm{V}$) but then improves again for even lower resting potentials. We also tested if the effective reversal potential impacted the fit and fitted with the setup reversal potential as well (see Figure 33). We see that the Fit of the membrane time constant is not impacted by the Fit with or without the effective reversal potential. The Fit of the synaptic time constant

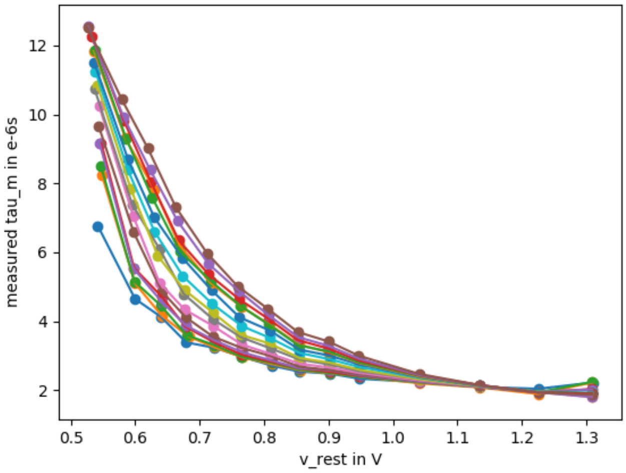


Figure 31: Fitted median membrane time constant with used effective reversal potential for Fit. Different curves represent different bit weights. Find unexpected dependence of fitted membrane time constant on the resting potential independent of the weight (even low weight "blue" not constant). Also see spread for different weights though this was expected due to saturation and deformation of the PSP.

appears slightly higher with the setup reversal potential compared to the effective reversal potential. However we still do not get the expected results of constant fitted time constants.

As this measurement disagrees with the previously done measurement (stronger dependence on the weight for the time constants) seen in Figure 18 and Figure 17, we repeat the previous measurement again on wafer 33 HICANN 297.

2.6.1 Repeated Measurement on Wafer 33

As we need to measure with all the different resting potentials anyway we also redo the calculation of the effective reversal potential to see if we get

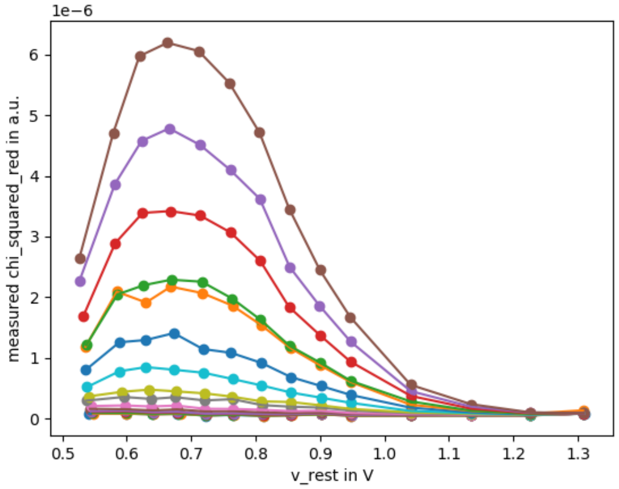
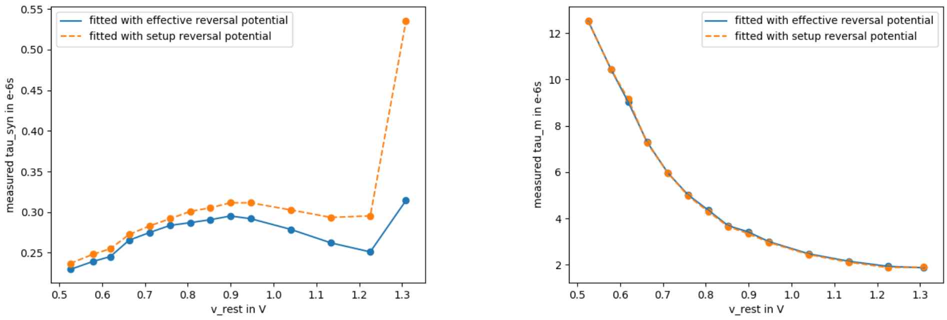


Figure 32: χ^2_{red} with used effective reversal potential for Fit. Different curves represent different bit weights. Find an increase that does not just scale with the resting potential (i.e. height of the PSP) so it cannot just be caused by relative fitting errors. Unfortunately we do not have a sure explanation at this point in time.

similar results (see Figure 34). We achieve similar results though the mean effective reversal potential is lower than for the other HICANN on the other wafer. We again find the same behaviour as before when fitting the weight with and without the effective reversal potential. When looking at the fitted time constants we also find the same behaviour as on wafer 30 HICANN 0. We again find the same behaviour as before when fitting the weight with and without the effective reversal potential (see Figure 36). The course of the time constant therefore seems not to be the effect of faulty hardware but instead is an actual physical property of the chip.

Since we were still not able to reproduce the results of Section 2.4 we look again into the changes made between these measurements. Apart from switching from the setup to the effective reversal potential (which did not



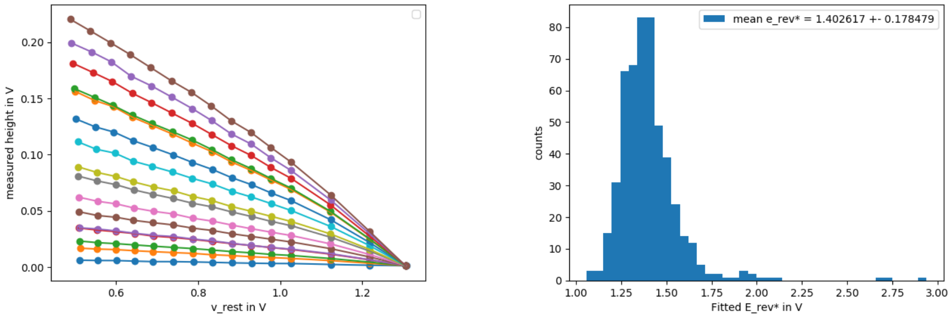
(a) Comparison of the Fit of synaptic time constant with the reversal potential fixed at the effective and setup reversal potential.

(b) Comparison of the Fit of membrane time constant with the reversal potential fixed at the effective and setup reversal potential.

Figure 33: Comparison of Fits of time constants with bit-weight = 15, '-' is the fit with effective reversal potential, '- -' is the fit with setup reversal potential. Only small difference for Fit of synaptic time constant nearly no difference for Fit of membrane time constant. Therefore our correction is not the reason for the resting potential dependent membrane time constant.

seem to have an impact as seen in Figure 33) we changed the reversal potential from 1.2V to 1.3V and turned off the inhibitory synaptic input to achieve a wider range of available V_{rest}. We will therefore switch the synaptic input circuit back on and look whether we can achieve the old measured data again.

The results can be seen in Figure 37, we find that we still have a dependence of the fitted parameters on the resting potential. The dependence seems to be shifted compared to Figure 36b. However at a resting potential of .9V we find that the time constants are roughly independent of the used weight as expected. This is also at the same difference of resting potential and reversal potential as the one in the measurements in Section 2.4. As it seems we were just lucky to choose parameters where the fitted parameters were constant for the used weights and also were approximately in the range of the setup parameters.



(a) Height of the PSPs versus the resting potential. Used to calculate the effective reversal potential by fitting to the linear course of the curve.

(b) Distribution of the fitted effective reversal potentials for each individual neuron. Reconstructed by fitting on the linear course of Figure 34a.

Figure 34: Measurement to reconstruct the effective reversal potential on wafer 33 HICANN 297. Done like before with wafer 30 HICANN 0. The mean effective reversal potential seems to be lower for this HICANN. This cannot be explained at this point in time since the same measurement method was used.

We did not find a reason why the fits are dependent on the resting potential, as analysing the circuit is out of the scope of this thesis.

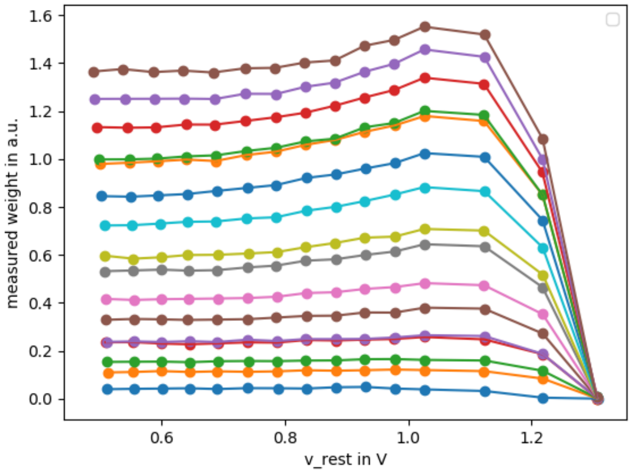
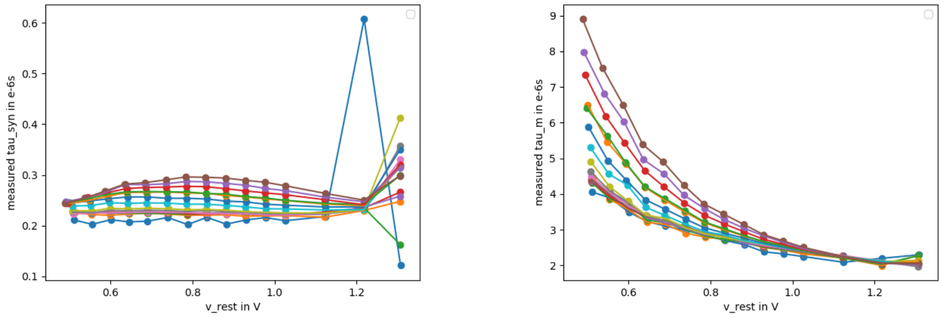


Figure 35: Redone measurement of Figure 29 on wafer 33 HICANN 297. Median of fitted weight with used effective resting potential. Find the same effect as seen on wafer 30 HICANN 0. This proves that this is a real hardware effect that has to be accounted for to get a correctly estimated synaptic weight.

2.7 Comparison of Hardware With Nest Simulation

With our fit routine we are able to reconstruct neuron parameters, which we can use in simulation to simulate PSPs. These PSPs are very close to those measured on hardware. For this reason we believe that we understand what our fit does. We hope that our fitted neuron parameters represent the actual hardware parameters if we are in a regime where the synaptic input does not saturate (low synaptic time constant and low weights), we do not reach membrane potentials where we are in a regime of non constant conductance of the resistive element (resting potential further away from the reversal potential) and the membrane time constant does not deviate from our setup membrane timeconstant. This leaves us with neuron parameters of $V_{\text{rest}} = .9\text{V}$, $E_{\text{syn}} = 1.3\text{V}$, $\tau_m = 20\text{e} - 7\text{s}$ and $\tau_{\text{syn}} = 2\text{e} - 7\text{s}$. Also we can not increase the weight indefinitely to make sure that the OTA does not



(a) Fitted median synaptic time constant with used effective reversal potential for Fit. Different curves are different bit weights w ("brown" 15, "blue" 0).

(b) Fitted median membrane time constant with used effective reversal potential for Fit. Different curves are different bit weights w("brown" 15, "blue" 0).

Figure 36: Redone measurement of Figure 31 and Figure 30 on wafer 33 HICANN 297. Find same courses of the fitted time constants. This also proves that the seen effects are real hardware effects and not any faults on the wafer. Outliers in Figure 36a for high resting potential happend due to the small size of the PSP. The fit routine then has problems accurately fitting the data as the PSP vanishes in the noise.

saturate.
As we want to demonstrate that we can translate the used synaptic weight as a parameter of V_{gmax}, g_{div} and the bit weight to the biological weight used in simulation we have to rescale our measured quantities. The voltages on hardware scale from 0V to 1.8V. this translates to -120mV to 60mV. The translation is given by:

$$U_{bio}[\mathrm{mV}] = (U_{\mathrm{HW}}[\mathrm{V}] - 1.2\mathrm{V})/10 \cdot 1000\frac{\mathrm{mV}}{\mathrm{V}} \tag{14}$$

Also the hardware is sped up by a factor of 1e4 compared to biology. This leads to the following translation:

$$t_{\mathrm{bio}}[\mathrm{ms}] = t_{\mathrm{HW}}[\mathrm{s}] \cdot 10^4 \cdot 10^3\frac{\mathrm{ms}}{\mathrm{s}} \tag{15}$$

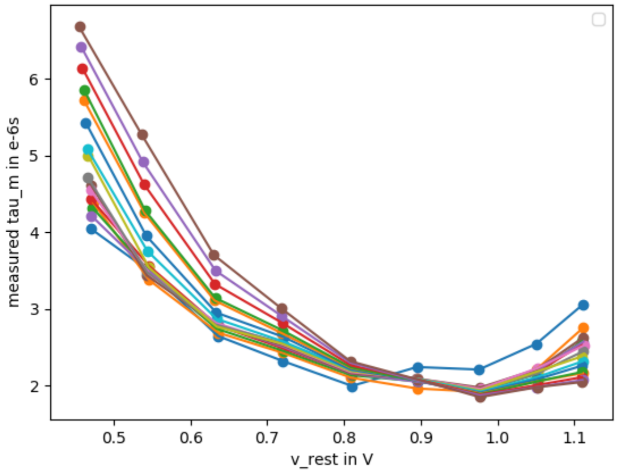


Figure 37: Fitted median membrane time constant with used effective reversal potential for Fit. Different curves are different bit weights. Done on wafer 33 with I_{convi} turned back on. Same behaviour as without the inhibitory synaptic input.

As we added an additional factor of 1e6 when fitting the data, this factor has to be removed additionally.

Lastly we need to rescale the weight. As the dimension of the weight in our case is a conductance per Farad (i.e. a reciprocal time). We also need to rescale it like the time, though the inverse way.

$$w_{\text{bio}}[\mathrm{S/F}] = w_{\text{HW}}[1/\mathrm{s}] \cdot 10^{-4} \cdot 10^{-3} \frac{\mathrm{msS}}{\mathrm{F}} \tag{16}$$

Again the factor from the fitting of 1e6 needs to be removed first. However here we need to multiply by this factor.

We now look at the rescaled PSPs and compare these to the simulated data with the fitted parameters used as the neuron parameters for the simulation. For this comparison we used the the measurement with weight independent membrane time constants found in Figure 37. We therefore chose the measurement with a resting potential of 0.9V from this run. In order to discuss the Fits we will first look at a histogram of the χ^2_{red} values of the fit (See Figure 38).

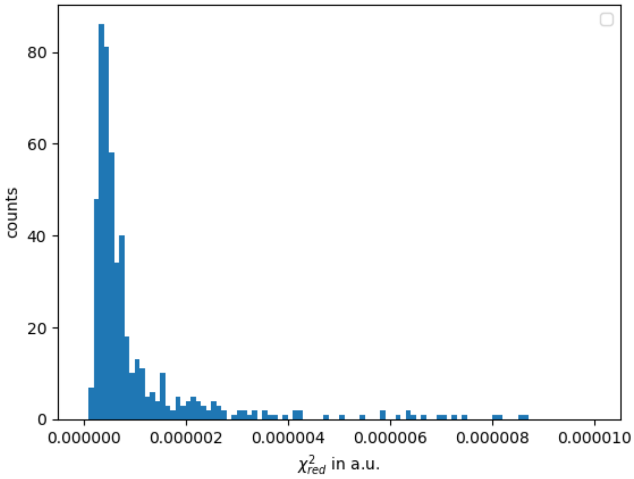
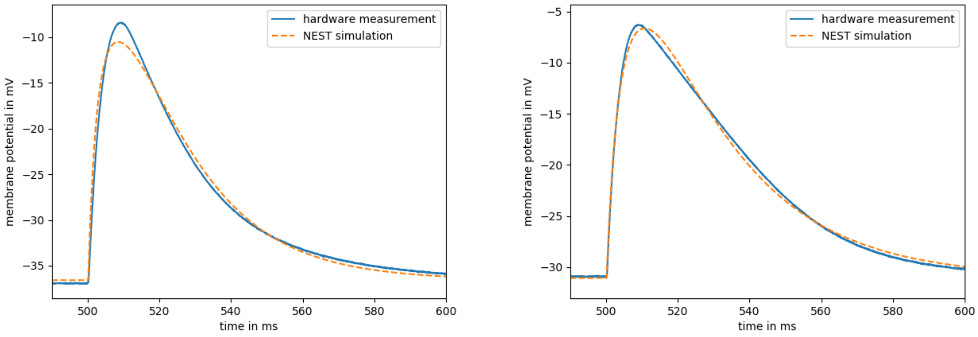


Figure 38: χ^2_{red} of the fits. Six outliers above 1e5 were cut off but will be discussed separately.

We will first look at the fits with high χ^2_{red} in Figure 39. We do not find any identical reasons for the low quality of the the PSPs. Two PSPs Figure 46b and Figure 46d show a wrong estimation of the resting potential. Reason for this could lie in the fit routine. In order to fit the data we had to define the start of the rise of the PSP as fitting the time t_0 of the arrival of the spike proved to be too inaccurate and time intensive. When fitting the differential equation the start condition was chosen to be the resting potential. A wrong estimation of the start time could therefore lead to a wrong estimation of the resting potential. For Figure 39a, Figure 39b and Figure 46c we also find a high fitted synaptic time constant. However we cannot be certain if this is the cause or the result of the low quality fit. One could work on better fitting these PSPs by e.g. refitting with different start parameters or fixing certain parameters. Intersting is also, that we find that in Figure 46b the synaptic and membrane time constant seem to have switched values. Since we know that the PSP is not symmetric under the two time constants this is probably caused by some deformation of the PSP which did not allow the fit to distinguish the difference of the two time constants.



(a) Neuron 135

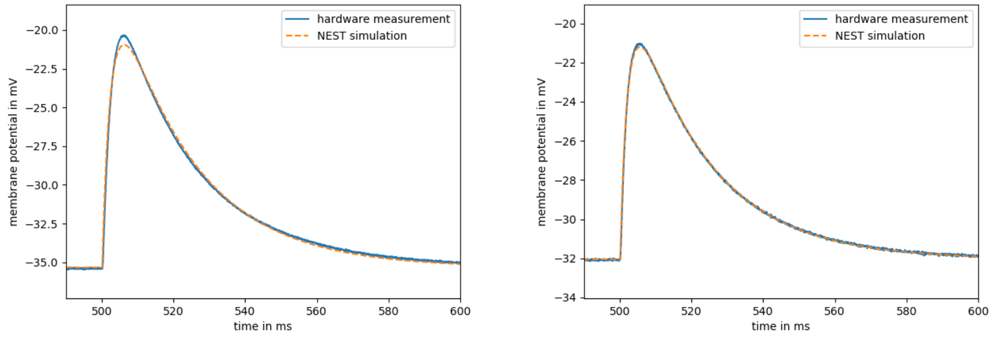
(b) Neuron 208

Figure 39: Rescaled Hardware PSPs with fits producing high χ^2_{red} with the PSPs simulated using NEST with the fitted neuron parameters in table. Neuron Parameters for Hardware measurement were $V_{\mathrm{rest}} = 900\mathrm{mV}$, $E_{\mathrm{syn}} = 1300\mathrm{mV}$, $\tau_{\mathrm{m}} = 20\mathrm{e}-7\mathrm{s}$, $\tau_{\mathrm{syn}} = 2\mathrm{e}-7\mathrm{s}$, $g_{\mathrm{div}} = 30$, $V_{\mathrm{gmax}} = .9\mathrm{V}$ and bit-weight 15. The rest of the traces are shown in section 4

Neuron	V_{rest} [mV]	E_{syn} [mV]	τ_{m} [ms]	τ_{syn} [ms]	C_{m} [µF]	w [µS]
135	-36.6	5.4	19.55	6.52	1	.30
208	-31.1	19.0	25.99	6.84	1	.17
281	-39.1	10.4	25.89	2.58	1	.21
284	-39.6	13.0	4.23	21.59	1	.18
389	-35.4	23.2	23.15	4.53	1	.16
396	-40.8	6.0	33.58	2.4	1	.21

We now look at the first six neurons on this HICANN seen in Figure 40. We see that most of these seem to be fitted accurately. Only in Figure 40a and Figure 47c a discrepancy is visible, however we again are not able to explain this.

Lastly we look at a histogram of the height of the fitted PSPs (see Figure 41. The PSPs with higher χ^2_{red} are indicated. We can see, that a higher PSP



(a) Neuron 0

(b) Neuron 1

Figure 40: Rescaled Hardware PSPs of first six neurons with the PSPs simulated using NEST with the fitted neuron parameters in table. Neuron Parameters for Hardware measurement were $V_{\text{rest}} = 900\text{mV}$, $E_{\text{syn}} = 1300\text{mV}$, $\tau_{\text{m}} = 20\text{e} - 7\text{s}$, $\tau_{\text{syn}} = 2\text{e} - 7\text{s}$, $g_{\text{div}} = 30$, $V_{\text{gmax}} = .9\text{V}$ and bit-weight 15. The rest of the traces are shown in section 4.

Neuron	V_{rest} [mV]	E_{syn} [mV]	τ_{m} [ms]	τ_{syn} [ms]	C_{m} [µF]	w [µS]
0	-35.3	18.6	21.87	2.72	1	.16
1	-32.1	6.4	21.29	2.4	1	.19
2	-33.5	17.3	13.61	2.47	1	.16
3	-27.5	4.6	17.13	2.73	1	.21
4	-24.2	10.5	25.46	2.99	1	.20
5	-28.99	10.0	20.04	2.81	1	.17

seems to be connected to a higher χ^2_{red}. This can partly be explained because a relative error in the fit parameters leads to a higher absolute χ^2_{red}. However the χ^2_{red} are much higher compared to the height of the PSP. This can be seen in Figure 42. Here we plotted the χ^2_{red} divided by the square of the height. If the decreased accuracy happened just due to relative errors of the fit parameters the course would be uniformly distributed. The found effect of high PSPs being described less accurately by the fit is therefore probably also due to saturation effects on the hardware.

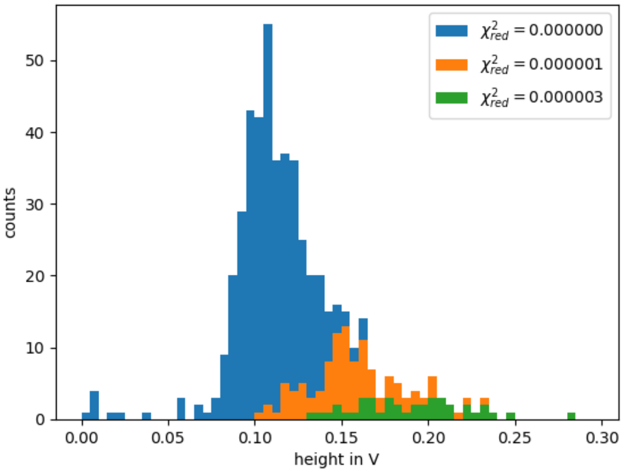


Figure 41: Histogram of the heights of the PSP, different colours indicate different quality of the Fits. Higher PSPs do not seem to be fitted as well (high χ^2_{red}).

2.8 Varying Of Weight Parameters

When calibrating parameters on the BrainscaleS-1 system many measurements at varied setup parameters are conducted. By then analyzing the membrane the desired parameter is reconstructed. This way the dependency of the setup and measured parameter can be estimated. If the function is known that describes this dependency the parameters in this function can be fitted to find the corresponding setup parameter for each desired setting.
As the synaptic weight is a parameter described by three setup parameters (g_{div}, V_{gmax} and bit-weight w) we need to calibrate each parameter individually. This will make it more difficult as the parameters influence eachother. However we want to use the whole parameter range for our calibration.
We can see in Figure 43a that the synaptic weight depends on the parameter g_{div} even when we set V_{gmax} to zero. This was also found in [5] in hardware simulations and was incorporated into the synaptic current I_{syn} by adding

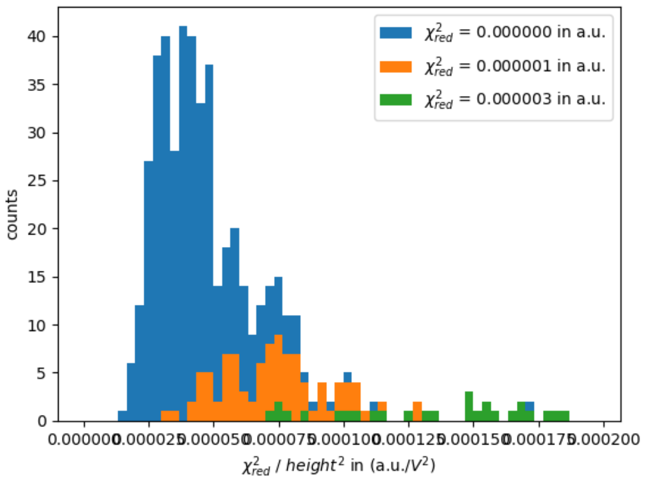
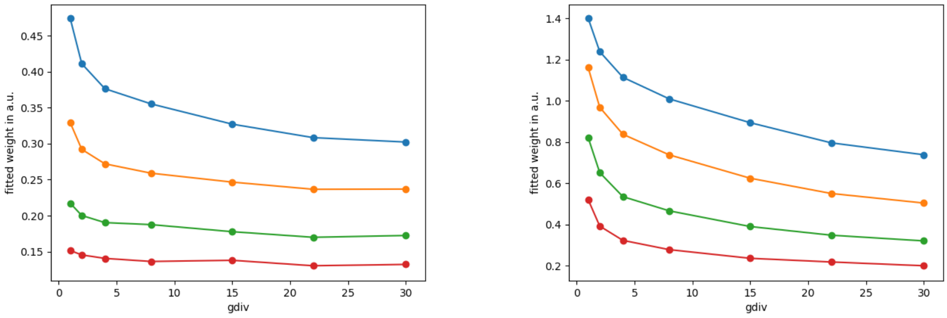


Figure 42: χ^2_{red} divided by the square of the height to show that reduction in fit quality is not only caused by a relative scaling of the PSP. The cause for high χ^2_{red} is probably caused by saturation and a variation from our fitted model.

additional correction terms. These were however not further explained.

$$I_syn(V_{\text{gmax}}, g_{\text{div}}, w) = V_{\text{gmax}} \cdot g_{\text{scale}} \frac{w}{{g_{\text{div}}}^{\gamma}} + \frac{\beta_1 w + \beta_2 w^2}{g_{\text{div}}} + i_0 + i_1 w_1 + i_2 w_2 + i_4 w_4 + i_8 w_8 \tag{17}$$

We also see that the courses roughly appear to be reciprocal in g_{div} with a on the bit-weight depending offset. We also see that the course of the curves drastically changes for Figure 43a but does not for Figure 43b. This way we can conclude, that the correction factor is much smaller then the factor depending on vgmax, as the course for low bit-weight in Figure 43a seems to disappear in the weight variations. In Figure 44 we again found the expected nonlinear behaviour for varying bit-weight. This was already explained in Section 2.4. However we do not see the nonlinear behaviour for lower bit-weight. The reason for this is probably that the weight variations are too small (in absolute values) to be recondtructed accurately. As the



(a) $V_{\text{gmax}} = 0.00\text{V}$

(b) $V_{\text{gmax}} = 0.01\text{V}$

Figure 43: Median fitted weight versus the used g_{div} parameter. Different coloured curves are different bit-weights (from the top: 15, 13, 10, 8). See roughly the expected course. With a with g_{div} decreasing fitted weight and a bit-weight dependent offset.

PSPs are very small (due to the $V_{\text{gmax}} = 0.0\text{V}$) the fit seems to struggle to reconstruct this effect. For this reason we are not able to fit Eq. (17) to the measurement to reconstruct the i_i. Maybe this could be improved by doing multiple measurements, however this was not done here. In Figure 45 we find a nonlinear relation of the synaptic weight and the parameter V_{gmax}. This was unexpected as it was not shown in hardware simulation of the synaptic current I_{syn} for this reason we have to suspect that it is coming from saturation in the synaptic input circuit.

Unfortunately this means that we are not able to reconstruct a synaptic weight of the form shown in Eq. (17). We will therefore not be able to find a function to calculate the synaptic weight after fitting the parameters of said function. However we should be able to use our results to create a lookup table to find the synaptic weights for some fixed parameters. To be able to calibrate the synaptic weight we need to understand the hardware better and have better ways of characterizing saturation effects which might occur.

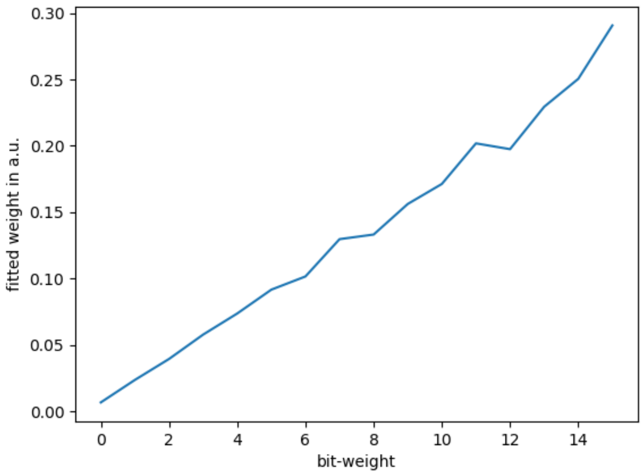


Figure 44: Median fitted weight versus the used bit-weight parameter. Course as expected for higher bit-weight. For lower bit-weight nonlinear course not as strong as in Figure 21. This is caused by small PSPs, which means that small variations tend to disappear in the noise.

3 Conclusion and Outlook

In this thesis we showed the necessary steps to measure the synaptic weight by fitting PSPs. After finding that only the solution to the differential equation without any approximations allows to fit the data we analysed PSPs with different neuron parameters to find those which lead to the best results. This way we found that a small synaptic time constant allows the fitting to work best. For this reason we used a synaptic time constant of $\tau_{\text{syn}} = 2e-7$s. We did not use smaller time constants as we can not be certain that smaller time constants are possible for all neurons and we wanted the neurons to be comparable (i.e. have roughly the same neuron parameters). As we found that we could not fit the weight and reversal potential simultaneously we had to fix the reversal potential in our fit. To do this we also had to compensate for

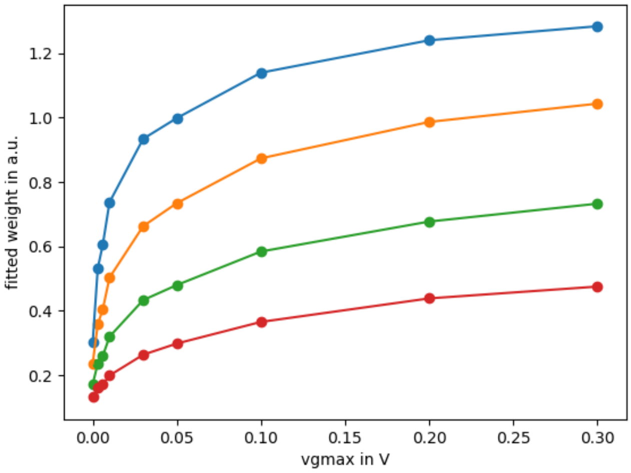


Figure 45: Median fitted weight versus the used V_{gmax} parameter with fixed $g_{\text{div}} = 30$. Different coloured curves are different bit-weights (from the top: 15, 13, 10, 8).

a nonlinear hardware effect, where the conductance of the resisitve element between the reversal potential and membrane had an additional dependence on the difference of potentials. We were able to develop a measurement routine to compensate for this effect and fix the reversal potential in the fit to its exact value. To prove that this worked we showed that the reconstructed weight is now independent on the used resting potential.

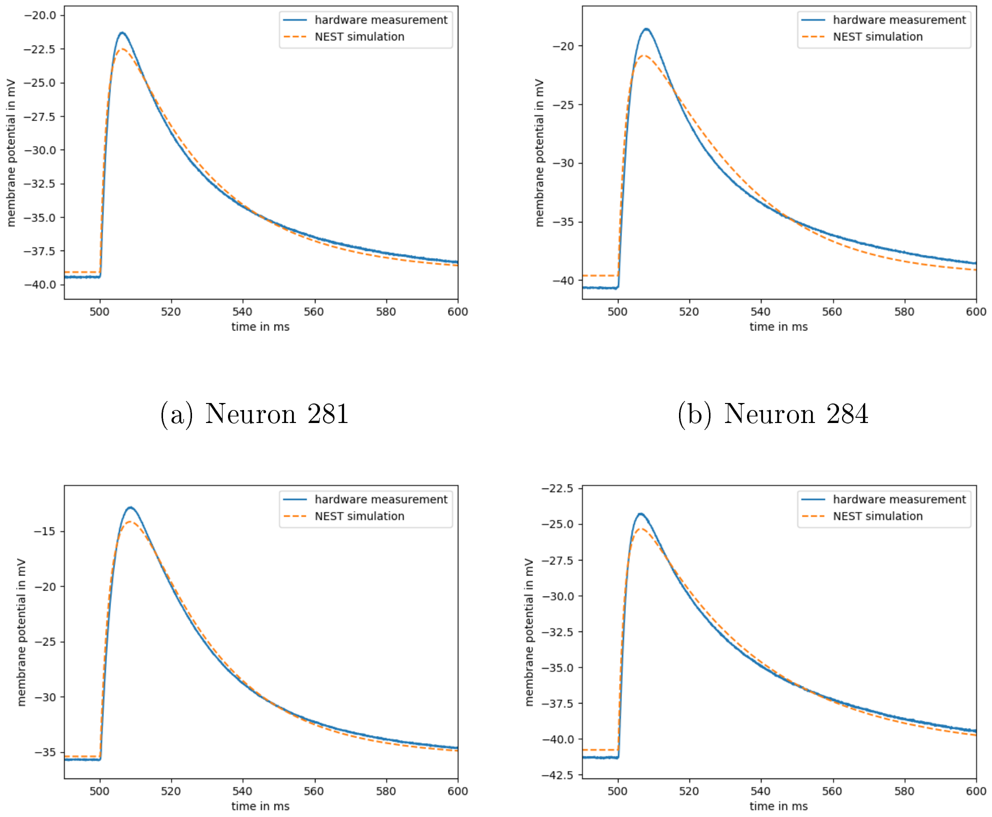
However during the measurement we found that our fit reconstructed a membrane time constant dependent on the resting potential. Since the quality of the fit however did not decrease drastically and we observed this effect independent of the weight (i.e. independent of the integrator voltage in the synapic input circuit) we can expect that this is a real effect on the hardware and not another saturation effect deforming the PSP. Analysing the circuit was out of the scope of this thesis.

Lastly we showed the impact of the parameters V_{gmax}, g_{div} and the bit-weight w. We showed that though the effect of these parameters on the synaptic current I_{syn} is known we did not find the same effect on the synaptic weight. This leads us to believe, that saturation effects do not allow the conductance to scale linearly with the input current I_{syn}. For this reason further work will be necessary to calibrate the synaptic weight, as these saturation effects need to be accounted for.

As we were able to reconstruct the weight in simulation we proved that this approach works, however we need to make the fitting more robust and do some adjustments to compensate for variations on hardware. Maybe smoothing the measured data first could lead to better fitting results. This might also improve the algorithm to find the beginning of the spike and therefore reduce outliers seen in Figure 39. We were able to compensate for hardware nonlinearities for one hardware effect (effective reversal potential due to voltage dependent resistive element), but more work would need to be done to find out more about the observed saturation, which we cannot explain yet. Nevertheless it could be shown that the analog synapses and neurons closely follow the mathematical model. Also the better understanding of the synaptic weight developed in this thesis will help in future experiments.

4 Appendix

4.1 Remaining figures



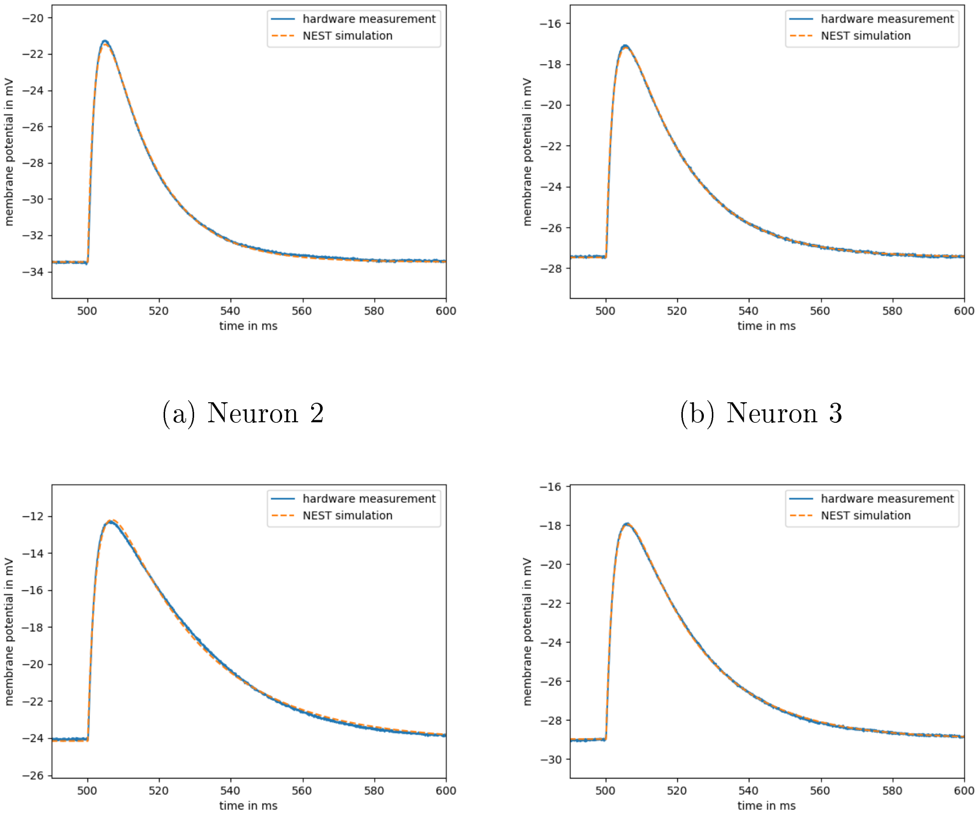
(a) Neuron 281

(b) Neuron 284

(c) Neuron 389

(d) Neuron 396

Figure 46: Remaining traces with high χ^2_{red} of Figure 39



(a) Neuron 2

(b) Neuron 3

(c) Neuron 4

(d) Neuron 5

Figure 47: Remaining traces of first 6 neurons of Figure 40

4.2 Used Versions of Python Modules

NumPY Version 1.16.4 by the Numpy Developers
http://www.numpy.org/
matplotlib Version 2.2.3 by The Matplotlib development team
https://matplotlib.org/
pandas Version 0.24.1 by the PyData Development Team
http://pandas.pydata.org
SciPy Version 1.2.1 by the SciPy Developers
http://www.scipy.org/

4.3 Used Versions of Software From the Group

project name	Commit ID	Commit Message
ztl	2934a12003c14e08643cf2b4b3cbe7553e860f08	Add support for other compilers
vmodule	2eb1aeadd2338f61c7098273dff95fbc2eb1e5d7	Reduce warnings
sthal	2d896e0e4cac02b2e54bd0c09809926401f3ef82	Revert "Lazy population of HICANN ADC config"
sctrltp	446f16d85866aa27031cf032b649a97c27213e7b	Fix Timeout-Problems
redman	690d48b10652844f778ca9f39c5f3387a7b48297	Add missing serialization of id
rant	8d5d65484667852ccfdb52843f0b6d4b6b067324	Fix rant-tests build
pywrap	961e7e3ff5921d1eef610fc76c5c513e97415fc6	Add add_optional_vector_converter_for
pyublas	cb8db753399719cd356afa1b2c749eb54c7c420a	Check for same condition in instantiation and declaration
pythonic	e9628388d2eb0ce34db770c660ea37718d97dd3d	Add support for other compilers
pypluplus	064993baaea33e81c93655d79d9a1a6204b4acd0	Fix deprecation warnings on newer boost versions
pygccxml	8ae9e19ae00c4152fa5a381eb9e663561c07345f	fix has_public_binary_operator to take name hiding etc into account
logger	8355792fd3e591d08381575fcf5c4b2547b5fe3d	Compile fix for gcc 4.4 (i.e. pywrap)
lib-rcf	8355792fd3e591d08381575fcf5c4b2547b5fe3d	Fix rcf_enable_warnings flag
lib-boost-patches	5d74d1ddd3fa2e1da534c753e6fa58931fb8aed4	Add LICENSE and a short README
hwdb	81f96bd86d89ffc5c3e3a66cd44db98eb5d74cfc	Move IB6 to CUBE3
hicann-system	14db2df8b4df50924564ac1e882284333f1bfbc9	Fix name clash of mutex object and std::mutex
hate	c7226ece407813b4e57da7852021a39de2fb4221	Rename is_in to is_in_type_list
halco	cc8854af538a399e924788f4188b15560b9a4a3f	Add unique Omnibus addresses for fisch backends

halbe	7a8979cf71909e80e74ee8e0f2e64bdaaedc3560	Set BASEDIR_NMPM _SOFTWARE to find files during tests
euter	5b6d4c8c1eb2b52514801bbc80841786641eeabe	Fix library folder hierarchy
code-format	f3f6424b08e58e64e43e08346ba3ad29f15410c5	Change clang-format cpp-standard from Cpp03 to Auto
chip-teststand	0bf4b4e4915e020d187383cc5d4b87cecc30fab9	Use 'doxygen' tool in wscript instead of 'documentation'
cd-denmen-teststand	71512c2bba2f4f5439c5fb41ac65f0688646b3a5	Fix unit of bias current
calibtic	99f5801685281581c70c3726af88c91d96087bf5	Log what is loaded
cake	8894bfc46ed7c173538636f3b6bd5a7cbcc60ee2	Added scripts used to Fit measured data and compare to bio simulation
bitter	aa18d4a73a994a7e8590addbc40f6dc34a439b24	Fix warning
symwaf2ic	104b0998ee50e735dacaf59a1829fe476ed5ca1d	Add BSS2 FPGA spec

4.4 Own Code Used

Scripts used to generate the data which is analysed in this thesis were placed here: https://gerrit.bioai.eu/c/cake/+/8338

References

[1] J. Dean, D. Patterson, and C. Young, "A new golden age in computer architecture: Empowering the machine-learning revolution," *IEEE Micro*, vol. 38, no. 2, pp. 21–29, 2018.

[2] T. N. Theis and H.-S. P. Wong, "The end of moore's law: A new beginning for information technology," *Computing in Science & Engineering*, vol. 19, no. 2, pp. 41–50, 2017.

[3] J. Schemmel, D. Brüderle, A. Grübl, M. Hock, K. Meier, and S. Millner, "A wafer-scale neuromorphic hardware system for large-scale neural modeling," *Proceedings of the 2010 IEEE International Symposium on Circuits and Systems (ISCAS"10)*, pp. 1947–1950, 2010.

[4] R. Brette and W. Gerstner, "Adaptive exponential integrate-and-fire model as an effective description of neuronal activity," *Journal of Neurophysiology*, vol. 94, no. 5, pp. 3637–3642, 2005. PMID: 16014787.

[5] C. Koke, *Device Variability in Synapses of Neuromorphic Circuits.* PhD thesis, Universität Heidelberg, 2017.

[6] M.-O. Gewaltig and M. Diesmann, "Nest (neural simulation tool)," *Scholarpedia*, vol. 2, no. 4, p. 1430, 2007.

[7] M. Rudolph and A. Destexhe, "Analytical integrate-and-fire neuron models with conductance-based dynamics for event-driven simulation strategies," *Neural Computation*, vol. 18, no. 9, pp. 2146–2210, 2006.

[8] M. Rudolph-Lilith, M. Dubois, and A. Destexhe, "Analytical integrate-and-fire neuron models with conductance-based dynamics and realistic postsynaptic potential time course for event-driven simulation strategies," *Neural Computation*, vol. 24, no. 6, pp. 1426–1461, 2012. PMID: 22364504.

[9] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J.

van der Walt, M. Brett, J. Wilson, K. Jarrod Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. Carey, İ. Polat, Y. Feng, E. W. Moore, J. Vand erPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and S. . . Contributors, “SciPy 1.0–Fundamental Algorithms for Scientific Computing in Python,” *arXiv e-prints*, p. arXiv:1907.10121, Jul 2019.

Acknowledgements

I would like to thank Dr. Johannes Schemmel for giving me the opportunity to work in the Electronic Vision(s) group on this interesting topic.

I am especially thankful to Dr. Sebastian Schmitt for his incredible support and general helpfulness at every hour, including proof reading this thesis.

Thanks to Jose Montés for his help with the cake calibration framework and the nice talks at lunch.

I also want to thank Sebastian Billaudelle for the help working with the hardware simulation and any hardware related questions.

Thanks to the rest of the people of the Electronic Vision(s), who answered any questions that came up.

Last but not least I want to thank my family and friends for always supporting me throughout my studies and for finding many of my spelling errors.