

**Fakultät für Physik und Astronomie**  
**Ruprecht-Karls-Universität Heidelberg**

Diplomarbeit  
im Studiengang Physik

vorgelegt von  
**Boris Ostendorf**

aus Kiel

2007



# **Charakterisierung eines Neuronalen Netzwerk-Chips**

Die Diplomarbeit wurde von Boris Ostendorf ausgeführt am  
Kirchhoff-Institut für Physik  
unter der Betreuung von  
Herrn Prof. Dr. Karlheinz Meier



## Charakterisierung eines Neuronalen Netzwerk Chips

In der vorliegenden Arbeit werden Teile des neuronalen Netzwerkchips Spikey auf Funktionstüchtigkeit untersucht. Spikey ist ein VLSI ASIC und implementiert ein pulskodiertes Neuronales Netzwerk mit 384 Neuronen, die auf einem erweiterten *Integrate-and-Fire* Modell beruhen, sowie knapp  $10^5$  Leitfähigkeits-basierten Synapsen. Die verwendeten Modelle benötigen für eine physikalische Realisierung in einem Mikrochip analoge Spannungswerte. Die zu diesem Zwecke verwendeten Spannungsquellen werden in der Arbeit intensiv untersucht. Es werden Kalibrierungsroutinen entwickelt, die lineare Abweichungen automatisch ermitteln und korrigieren können, die gefundenen Parameter werden im XML-Format gespeichert.

Eine zweite untersuchte Schaltung dient der Erzeugung von *Short Term Plasticity*. *Short Term Plasticity* beschreibt einen Adaptionsmechanismus, der in Kraft tritt, wenn ein präsynaptisches Neuron in einem kurzen Zeitraum wiederholt feuert. Dabei ändern sich kurzzeitig die Übertragungsfunktionen der Synapsen und die induzierten postsynaptischen Potenziale werden verstärkt oder abgeschwächt. Die zur Nachbildung dieses Mechanismus verwendete Schaltung ist vielfältig einstellbar, um in den Simulationen einen großen Variationsbereich abdecken zu können. In der Arbeit wird diese Schaltung auf Funktionsfähigkeit hin untersucht. Es lässt sich daraufhin ein Parametersatz ermitteln, mit dem das biologische Vorbild gut nachbildet wird. Die Schaltung erfüllt also die an sie gestellten Anforderungen.

## Characterization of a Neural Network Chip

In the presented thesis parts of the neural networkchip Spikey will be examined in terms of functional efficiency. Spikey is a VLSI ASIC and implements a pulscoded neural network with 384 neurons, which are based on an advanced *Integrate-and-Fire* model, and almost  $10^5$  conductance-based synapses. For a physical representation in a microchip the used models depend on analog voltages. The voltagesources used for this purpose are intensively examined in this thesis. Calibration-routines are developed which are capable of automatically acquiring and correcting linear variances. The found parameters are saved using the XML-format.

A second examined circuit is used to generate *Short Term Plasticity*. *Short Term Plasticity* describes an adaption-mechanism which occurs if a presynaptic neuron fires repeatedly in a short period of time. It is expressed in a change of the synaptic transfer function, the induced postsynaptic potentials are strengthened or weakened. The circuit used for modeling this mechanism is adjustable in many ways in order to achieve a great span of variation possibilities. In this thesis the circuit is examined in respect to operability. Thereupon a set of parameters is deduced that fits the biological specimen well. Thus the circuit meets the demands.



# Inhaltsverzeichnis

<b>1</b>	<b>Einführung und Motivation</b>	<b>1</b>
<b>2</b>	<b>Grundlagen Mikroelektronik</b>	<b>3</b>
2.1	pn-Übergang . . . . .	3
2.2	MOSFET . . . . .	4
2.3	Transmission-Gate und Charge Injection . . . . .	8
2.4	Stromspiegel . . . . .	9
2.5	Inverter . . . . .	10
2.6	Mikrochipherstellung . . . . .	11
<b>3</b>	<b>Grundlagen Neuronale Netze</b>	<b>13</b>
3.1	Biologische Neuronen . . . . .	14
3.2	Neuron- und Netzwerkmodelle . . . . .	16
3.2.1	Neuronenmodelle der ersten und zweiten Generation . . . . .	17
3.2.2	Ein Modell aus der dritten Generation: Leaky Integrate-and-Fire . . . . .	18
3.2.3	Leitfähigkeits-basierte Synapsen . . . . .	19
3.3	Lernen und Plastizität . . . . .	20
3.3.1	Lernmodelle . . . . .	20
3.3.2	Spike Time Dependent Plasticity . . . . .	21
3.3.3	Short Term Plasticity . . . . .	21
3.4	High-Conductance State . . . . .	23
<b>4</b>	<b>Spikey-Chip</b>	<b>25</b>
4.1	Neuron-Modell . . . . .	25
4.2	Chipaufbau . . . . .	26
<b>5</b>	<b>Test-Hardware</b>	<b>31</b>
5.1	Backplane und Nathan . . . . .	31
5.2	Recha . . . . .	33
5.3	Recha V2 . . . . .	34
<b>6</b>	<b>Messungen</b>	<b>41</b>
6.1	Analogspannungsquellen Spikey 1 . . . . .	41
6.1.1	Messungen . . . . .	41
6.1.2	Schaltungsaufbau der Analogspannungsquellen . . . . .	42
6.1.3	Erste Kalibrierung: Beseitigung der scheinbar zufälligen Fehler . . . . .	44
6.1.4	Zweite Kalibrierung: Korrektur der Steigung . . . . .	44

6.2	Analogspannungsquellen Spikey 2 . . . . .	47
6.2.1	Änderungen der Schaltung . . . . .	47
6.2.2	Auswirkungen der Änderungen . . . . .	47
6.2.3	Kalibrierung der Analogspannungsquellen . . . . .	48
6.3	Short Term Plasticity . . . . .	52
6.3.1	Schaltungsaufbau der Synapsentreiber . . . . .	52
6.3.2	Messergebnisse . . . . .	54
<b>7</b>	<b>Zusammenfassung und Ausblick</b>	<b>59</b>
<b>A</b>	<b>Kalibrierung der Neuronen</b>	<b>i</b>
<b>B</b>	<b>Schaltpläne</b>	<b>v</b>
<b>C</b>	<b>Lagenpläne</b>	<b>x</b>
<b>D</b>	<b>Bestückung</b>	<b>xv</b>
<b>E</b>	<b>Ergänzende Abbildungen</b>	<b>xix</b>
E.1	Ersatzschaltbilder zur Short Term Plasticity Erzeugung . . . . .	xix
E.2	Lötstempel . . . . .	xx
<b>F</b>	<b>Konfigurationsdateien zur Short Term Plasticity Messung</b>	<b>xxi</b>
	<b>Literaturverzeichnis</b>	<b>xxviii</b>

# Kapitel 1

## Einführung und Motivation

Diese Arbeit entsteht aus mehreren Motiven. Da wären Neugier, Wissensdrang. Wie funktioniert mein Gehirn. Ist es möglich es zu simulieren? Können wir es schaffen, Teile davon nachzubauen?

Weitere Motive sind Ehrgeiz, das Streben danach etwas zu Lernen und noch viele mehr. All diese Wünsche, Bedürfnisse und Empfindungen gehören zu unserer Persönlichkeit. Sie entstammen unserem Gehirn, einer leistungsfähigen Recheneinheit, und unterscheiden uns von künstlichen Recheneinheiten, die in bestimmten Bereichen noch viel leistungsfähiger sind. Das Gehirn stattet uns mit Intelligenz aus, wir können unbekannte Problemstellungen erfassen und lösen. Es ermöglicht uns, aus den so gewonnenen Erfahrungen zu lernen. Doch was das Bewusstsein ausmacht, wie Gefühle entstehen und wie die Intelligenz begründet ist, weckt seit Jahrtausenden das Interesse des Menschen und ist bis heute größtenteils ungeklärt.

Die Ansätze, das Gehirn zu verstehen sind vielfältig. Den Grundstein für viele neuere Lernregeln legte zum Beispiel der Arzt und Verhaltensforscher Iwan Pavlov gegen Ende des 19. Jahrhunderts. An Experimenten mit Hunden entdeckte er das Prinzip der Konditionierung.

Ein anderer Weg besteht darin, die Grundbausteine des Gehirns zu verstehen. Anfang des 20. Jahrhunderts wurde das Neuron als Grundbaustein identifiziert. Der Anatom Ramón y Cajal benutzte eine von Camillo Golgi entwickelte Färbetechnik, um einzelne Neuronen dunkel zu färben. Er postulierte, dass das Nervensystem aus Milliarden einzelner Neuronen besteht, die polarisiert werden und über Synapsen miteinander kommunizieren.

Durch Messungen mit Elektroden konnte das Verhalten von einigen wenigen Neuronen in *in-vitro* Experimenten bestimmt werden. Anschließend wurde versucht, aus den gefundenen Ergebnissen Modelle zu entwickeln, die das Neuron beschreiben. Ein sehr bekanntes und ausführliches Modell wurde 1952 von Hodgkin und Huxley vorgestellt.

Das Wissen um das exakte Verhalten eines Neurons bringt jedoch allein keine Erkenntnis über das Entstehen von Gefühlen oder die Bildung von Intelligenz. Diese entstehen aus dem Zusammenspiel von vielen Milliarden Neuronen, die ein großes Netzwerk bilden. Die Zielsetzung, das Verhalten von vielen vernetzten Neuronen zu studieren, kann man von zwei Richtungen angehen. Die erste beginnt beim Gehirn selbst. Es werden Messungen am lebenden Gehirn durchgeführt, bei denen unter anderem versucht wird, den Ort und den zeitlichen Verlauf von Netzwerkaktivität zu messen.

Die andere Richtung beginnt bei einigen wenigen künstlichen Neuronen, die ein künstliches neuronales Netzwerk bilden. Die Simulation von Neuronen in Software ist eine beliebte Methode, stellt

jedoch hohe Anforderungen an die Rechenleistung. Die Genauigkeit des verwendeten Neuronmodells und die Anzahl der Neuronen haben großen Einfluss auf die benötigte Rechenzeit.

In der *Electronic Vision(s)* Gruppe im Kirchhoff-Institut für Physik in Heidelberg werden Alternativen zur Simulation in Software erarbeitet. Es wurden bereits zwei verschiedene Mikrochips entwickelt, die eine physikalische Repräsentation eines neuronalen Netzwerkes implementiert haben. Der erste Chip, HAGEN<sup>1</sup>, beruht auf dem Perzeptron-Modell. Das McCulloch-Pitts-Modell stellt die Basis für die verwendeten Neuronen dar. Der HAGEN-Chip wurde bereits in vielen Experimenten verwendet [2, 3, 4, 5, 6].

Der Nachfolge-Chip Spikey verwendet ein pulskodiertes Netzwerk, mit Neuronen, die auf einem erweiterten *Integrate-and-Fire* Modell basieren. Die Aufgabe des Autors ist es, einige kritische Funktionen des Spikey-Chips zu untersuchen. Bei Abweichungen vom gewünschten Verhalten sollen Strategien entwickelt und implementiert werden, mit dem Ziel, die Fehler zu korrigieren.

Die *Electronic Vision(s)* Gruppe ist am Forschungsprojekt FACETS<sup>2</sup> beteiligt, in dem intensiv im Bereich der neuronalen Netze geforscht wird. So werden biologische Experimente durchgeführt, Modelle von neuronalen Netzwerken entwickelt und diese in Software als auch in Hardware simuliert bzw. nachgebaut.

Dieses Dokument ist in fünf Teile gegliedert, gefolgt von einer Zusammenfassung.

- Der Schwerpunkt der Untersuchungen liegt auf zwei Funktionen. Zur Verständnis des beobachteten Verhaltens ist es nötig, die zu Grunde liegenden Schaltungen in ihrer Funktionalität zu beschreiben. Kapitel 2 gibt eine Einführung in die in der Mikroelektronik oft verwendeten und auch hier zur Anwendung kommenden Grundschaltungen.
- In Kapitel 3 wird eine Einleitung zur Theorie der künstlichen Neuronalen Netzwerke gegeben. Das biologische Vorbild sowie die in der Praxis oft verwendeten Netzwerk- und Neuronmodelle werden vorgestellt.
- Die Implementierung eines der Modelle in einem Mikrochip wird in Kapitel 4 beschrieben. Dort werden die grundlegenden Funktionen und der Aufbau des Spikey-Chips erläutert.
- Kapitel 5 handelt von der zum Betrieb des Chips benötigten Hardware. Die schon beim HAGEN-Chip verwendete Backplane und die Nathan-Platine werden vorgestellt, sowie die Spikey-spezifische Recha-Platine. Letzteres wurde vom Autor überarbeitet und um Zusatzfunktionen erweitert.
- In Kapitel 6 folgt die Beschreibung und Analyse der am Spikey-Chip durchgeführten Messungen. Die untersuchten Schaltungen werden im Detail vorgestellt und das jeweils gemessene Verhalten ergründet. Weiterhin wird die entwickelte Software vorgestellt, die auf der Basis des gemessenen Verhaltens versucht Korrekturen anzubringen.

Fachbegriffe werden bei ihrer ersten Erwähnung im Text kursiv dargestellt, danach werden sie als bekannt vorausgesetzt. Spannungen sind, wie in der meist englischsprachigen Referenzliteratur, mit  $V$  abgekürzt.

---

<sup>1</sup>HAGEN: Heidelberg Analog Evolvable Network [1]

<sup>2</sup>FACETS: Fast Analog Computing with Emergent Transient States [7]

## Kapitel 2

# Grundlagen Mikroelektronik

Um das Verhalten eines Mikrochips charakterisieren zu können, ist es nötig, die zu Grunde liegende Physik sowie die zur Anwendungen kommenden Schaltelemente zu kennen.

### 2.1 pn-Übergang

Einer der für die Mikroelektronik bedeutsamsten Werkstoffe, Silizium, ist ein *Halbleiter*. Die elektrische Leitfähigkeit von Halbleitern liegt bei Normalbedingungen zwischen der von Isolatoren und Metallen. Die Leitfähigkeit ist temperaturabhängig, am absoluten Nullpunkt sind reine Halbleiter Isolatoren, mit steigender Temperatur nimmt die Leitfähigkeit zu. Dieses Verhalten wird vom Bändermodell beschrieben. Das energetisch höchste voll besetzte Band wird als *Valenzband*, das erste leere darüberliegende Band als *Leitungsband* bezeichnet, zwischen ihnen liegt die *Bandlücke*. Durch thermische Anregung können Elektronen vom Valenzband in das Leitungsband gehoben werden. Sie sind nicht mehr an einzelne Atome gebunden, sondern können sich frei bewegen, gleichzeitig hinterlassen die Elektronen ein positiv geladenes Loch im Valenzband. Mit zunehmender Temperatur steigt die Anzahl der freien Ladungsträger, die Ladungsträgerkonzentration, und somit auch die Leitfähigkeit.

Die Anzahl der freien Ladungsträger lässt sich durch *Dotierung* erhöhen. Bei diesem Vorgang werden Fremdatome mit einer anderen Wertigkeit gegenüber Silizium in den Halbleiter eingebracht. Silizium liegt in der vierten Hauptgruppe des Periodensystems, seine vier Elektronen auf der äußersten Schale gehen Bindungen mit den Nachbarn ein. Wird nun ein Siliziumatom durch ein fünfwertiges Fremdatom ersetzt, so gehen vier Elektronen Bindungen zu den Nachbarn ein, das fünfte Elektron dagegen hat keinen Bindungspartner. Bei einem dreiwertigen Fremdatom fehlt ein Elektron für die Bindung, dies entspricht einem Loch. Das Elektron bzw. das Loch sind nur schwach an den Atomrumpf gebunden und werden bei genügend hoher Temperatur ins Leitungsband bzw. Valenzband abgegeben, der Atomrumpf wird ionisiert. Da das fünfwertige Atom ein Elektron zuviel hat, das an das Leitungsband abgegeben wird, wird es Donator genannt, man spricht von einer n-Dotierung. Das dreiwertige Atom dagegen kann ein Elektron aufnehmen um die offene Bindung zu schließen und wird deshalb als Akzeptor bezeichnet, diese Dotierung wird p-Dotierung genannt.

Bei einem pn-Übergang grenzt ein p-dotierter Bereich an einen n-dotierten. Aufgrund des Konzentrationsgradienten diffundieren die in ihren Ursprungsbereichen vorhandenen Majoritätsladungsträger in den anderen Bereich, also Elektronen in den p-dotierten Bereich und Löcher in den n-dotierten Bereich. Die Ladung der zurückbleibenden ionisierten Atomrümpfe wird nicht mehr ausgeglichen,

es entsteht eine *Raumladungszone*, der Potenzialunterschied ist die *Diffusionsspannung*  $V_D$ . Die Ladungsträger müssen nun eine Potenzialbarriere überwinden, um in den anderen Bereich zu gelangen. Der Strom der Majoritätsladungsträger wird Diffusionsstrom oder auch *Rekombinationsstrom*  $I_R$  genannt, da die Ladungsträger hinter der Raumladungszone mit den dort vorherrschenden Majoritätsladungsträgern rekombinieren.

Im Gleichgewichtszustand wird der Rekombinationsstrom ausgeglichen durch den Driftstrom oder auch *Erzeugungsstrom*  $I_E$ . Aufgrund von thermischer Anregung werden im Halbleiter ständig Elektron-Loch-Paare erzeugt, die kurz darauf durch Rekombination wieder verschwinden. Entstehen sie jedoch in oder nahe bei der Raumladungszone, so werden die Majoritätsladungsträger von der Potenzialbarriere abgestoßen, während die Minoritätsladungsträger in den anderen Bereich gezogen werden. Der Strom der Minoritätsladungsträger ist der Erzeugungsstrom. Im Stromgleichgewicht werden die Ströme von Gleichung 2.1 beschrieben [8], wobei der Vorfaktor  $a$  nur schwach temperaturabhängig ist, und das Verhalten durch den exponentiellen Faktor bestimmt wird.

$$|I_E| = |I_R| = a(T)e^{-eV_D/k_B T} \quad (2.1)$$

Eine Diode besteht aus einem pn-Übergang und kann als Stromgleichrichter dienen, denn der Strom durch den Übergang ist abhängig von der angelegten elektrischen Spannung. Der dotierte Halbleiter hat eine gute Leitfähigkeit, nur in der Raumladungszone gibt es kaum freie Ladungsträger. Diese wird auch *Verarmungszone* genannt, deshalb kann man davon ausgehen, dass die angelegte Spannung über der Raumladungszone abfällt. Die angelegte Spannung wird zur Diffusionsspannung addiert, dementsprechend ändert sich das Stromgleichgewicht.

Im n-dotierten Bereich ist die Raumladungszone positiv geladen, da die Elektronen wegdiffundiert und nur die positiv geladenen Atomrümpfe übrig sind. Wird der positive Pol der angelegten Spannung an die n-Dotierung gelegt, so steigt die Potenzialbarriere für den Rekombinationstrom. Die Stärke des Erzeugungsstroms wird dagegen nicht beeinflusst, sie ist abhängig von der Anzahl der erzeugten Minoritätsladungsträger, die Erhöhung der Potenzialdifferenz beschleunigt diesen Vorgang nicht. Man spricht bei dieser Schaltung von der *Sperrrichtung* der Diode, bei genügend hoher Spannung wird  $I_R$  aufgehoben, es fließt nur noch  $I_E$ .

Ist die äußere Spannung dagegen mit dem negativen Pol an den n-dotierten Bereich angeschlossen, so spricht man von der *Durchlassrichtung*. Die Spannung reduziert die Potenzialbarriere,  $I_R$  wächst exponentiell. Zusammengefasst ergibt sich folgende Gleichung, wobei der Sättigungsstrom  $I_s$  gleich  $I_E$  aus Formel 2.1 ist.

$$I(U) = I_s \left( e^{eU/k_B T} - 1 \right) \quad (2.2)$$

In Abbildung 2.1 ist die Kennlinie einer Diode gezeigt, zu beachten sind die unterschiedlichen Skalen. In der linken Hälfte ist die Diode in Sperrrichtung geschaltet, es fließt nur der konstante Sperrstrom. Auf der rechten Seite, der Durchlassrichtung, steigt der Strom exponentiell an.

## 2.2 MOSFET

Abbildung 2.2 zeigt den Schnitt durch einen  $NMOS^3$  Feld-Effekt-Transistor. In einer schwach p-dotierten Wanne liegen zwei Bereiche mit n-Dotierung. Die Art der Dotierung dieser Kontakte bestimmt das Verhalten und den Namen des Transistors. n-dotierte Bereiche in einer p-dotierten Wanne

<sup>3</sup>MOS: Metal Oxide Semiconductor.

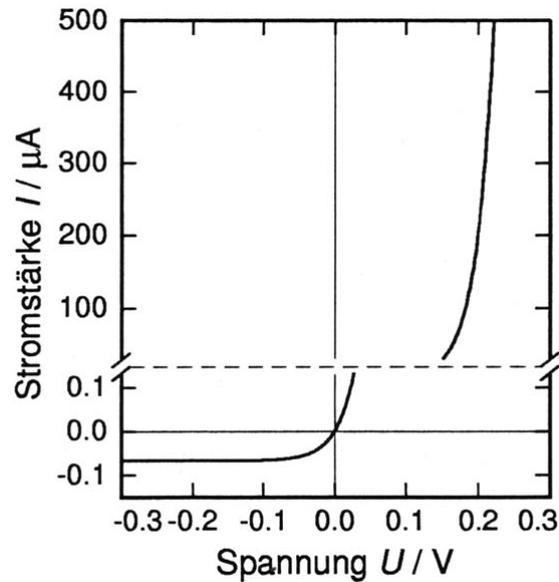


Abbildung 2.1: Strom-Spannungskennlinie einer pn-Diode. Entnommen aus [8]

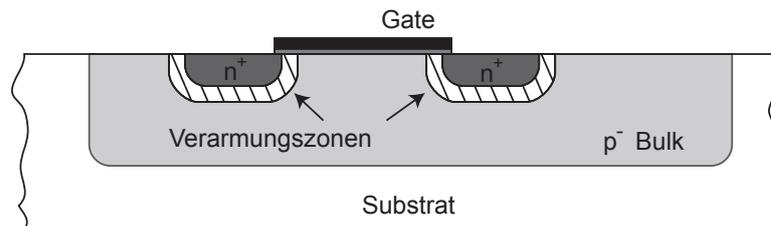


Abbildung 2.2: Schnitt durch einen NMOS-Transistor

bilden einen NMOS, p-dotierte Kontakte in einer n-Wanne einen PMOS. Die Kontakte sind *Source*- und *Drain*-Anschluss des Transistors, sie unterscheiden sich äußerlich nicht. Erst durch Anlegen einer Spannung wird definiert, welcher Kontakt welche Rolle übernimmt. Beim NMOS wird der Anschluss mit dem negativen Potenzial *Source* genannt, beim PMOS der mit dem positiven. Isoliert über dem Bereich zwischen den beiden Kontakten liegt das *Gate*. Zusätzlich gibt es noch einen vierten Anschluss, den sogenannten *Bulk*-Anschluss. Der *Bulk* ist der Bereich zwischen und um *Source* und *Drain* herum. Im einfachsten und meistens gewünschten Falle liegen *Source* und *Bulk* auf dem gleichen Potenzial.

An den pn-Übergängen zwischen *Bulk* und *Source* bzw. *Drain* bilden sich Verarmungszonen. Legt man eine positive Spannung  $v_{GS}$  zwischen *Gate* und *Bulk* an, so entsteht um das *Gate* herum ein elektrisches Feld. Dieses Feld sorgt dafür, dass die Majoritätsladungsträger im *Bulk* vom *Gate* abgestoßen werden. Gleichzeitig werden Minoritätsladungsträger angezogen, die mit den Majoritätsladungsträger rekombinieren, unter dem *Gate* entsteht eine weitere Verarmungszone, die an den Rändern mit

denen der pn-Übergänge verschmilzt. Überschreitet die *Gate*-Spannung einen bestimmten Schwellwert  $V_T$  so übersteigt die Anzahl der Minoritätsladungsträger die der Majoritätsladungsträger, der Bereich unter dem *Gate* ist invertiert. Die vorherrschenden Ladungsträger in *Source*, *Drain* und unter dem *Gate* sind nun gleichartig. Es bildet sich ein *Kanal*, in dem die Minoritätsladungsträger von *Source* nach *Drain* fließen können. Durch Anlegen einer Spannung  $v_{DS}$  zwischen *Source* und *Drain* wird ein Stromfluss hervorgerufen. Dieser ist bis zu einem gewissen Wert von  $v_{DS}$  proportional zu  $v_{GS}$ , der Transistor befindet sich im *ohmschen Bereich*. Für den Strom von *Drain* nach *Source* gilt Gleichung 2.3, die sogenannte *Sah-Gleichung* [9], wobei  $K'$  der Transkonduktanz-Parameter ist, der proportional zur Beweglichkeit der Ladungsträger ist.  $W$  ist die Breite des Transistors und  $L$  die Länge.

$$i_D = \frac{K'W}{2L} [2(v_{GS} - V_T)v_{DS} - v_{DS}^2] \quad (2.3)$$

Durch das positive Potential am *Drain* vergrößert sich dessen Verarmungszone mit steigendem  $v_{DS}$ . Wird  $v_{DS}$  größer als  $v_{GS} - V_T$ , so dringt die Verarmungszone unter das *Gate* vor und schnürt den Kanal ab. Eine weitere Erhöhung von  $v_{DS}$  erhöht den Strom kaum noch. Dies ist der *Sättigungsbereich* des Transistors, der durch Gleichung 2.4 beschrieben wird. Den linken Teil der Gleichung erhält man aus der *Sah-Gleichung* durch Einsetzen von  $v_{DS} = v_{GS} - V_T$ , der rechte Teil spiegelt die geringe Zunahme von  $i_D$  mit  $v_{DS}$  wieder. Je größer die Verarmungszone um das *Drain*, desto kürzer ist der Kanal, den die Ladungsträger durchlaufen müssen. Die effektive Kanallänge wird kürzer und der Strom steigt, dies wird durch das Produkt aus  $v_{DS}$  mit dem Kanallängenmodulationsparameter  $\lambda$  verdeutlicht.

$$i_D = \frac{K'W}{2L} (v_{GS} - V_T)^2 (1 + \lambda v_{DS}) \quad (2.4)$$

Abbildung 2.3 zeigt den Strom in Abhängigkeit von  $V_{DS}$  für eine Schar von  $V_{GS}$ -Werten. Links ist der ohmsche Bereich, die Steigung ist dort zunächst fast linear und geht dann in den gesättigten Bereich über, in dem der Strom nur noch geringfügig ansteigt.

Die Gleichungen 2.3 und 2.4 beschreiben den *Drainstrom* im Bereich  $v_{GS} > V_T$ , die Zone unter dem *Gate* ist stark invertiert, der Transistor befindet sich im Bereich der *starken Inversion*. Reduziert man  $v_{GS}$ , so stellt man fest, dass der *Drainstrom* für  $v_{GS}$  unterhalb von  $V_T$  nicht null ist, sondern exponentiell abfällt. Dies ist der *Subthreshold-Bereich*, oder auch der Bereich der *schwachen Inversion*. Der *Drainstrom* ist in diesem Bereich abhängig von vielen Parametern, die in Simulationsprogrammen verwendeten Gleichungen sind deshalb komplex und unanschaulich. Gleichung 2.5 ist stark vereinfacht, sie ist bei Raumtemperatur gültig, wenn *Bulk* und *Source* auf dem gleichen Potenzial liegen. Den Übergang zwischen schwacher und starker Inversion kann sie nicht beschreiben, dort sind komplexere Gleichungen erforderlich.

$$i_D = \frac{W}{L} I_{D0} \exp\left(\frac{qV_{GS}}{nk_B T}\right) \quad (2.5)$$

$n$  und  $I_{D0}$  sind prozessabhängige Parameter. Der *Drainstrom* im Übergang zwischen schwacher und starker Inversion wird in Abbildung 2.4 gezeigt.

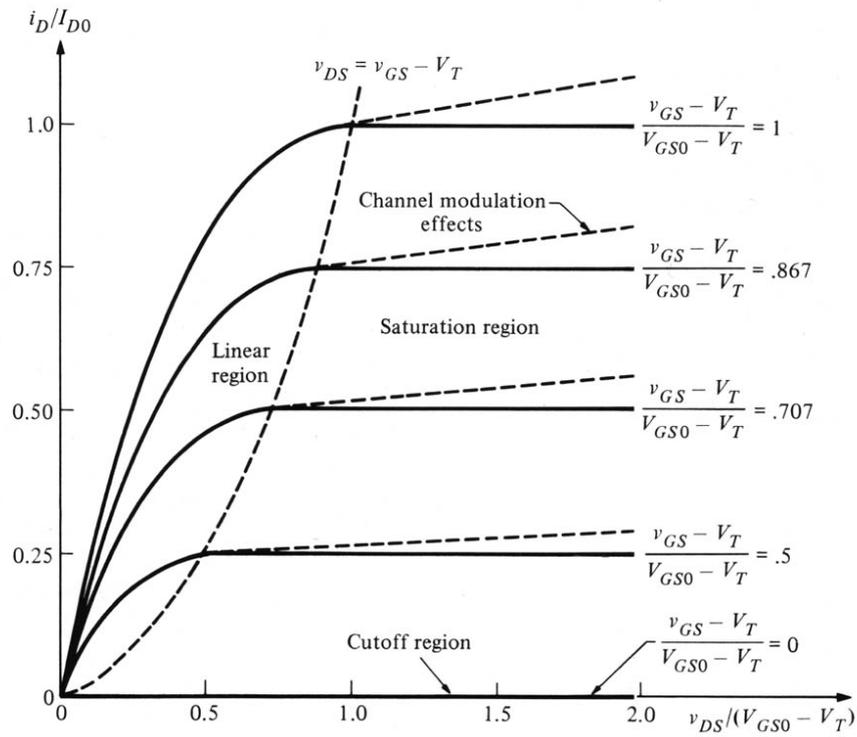


Abbildung 2.3: Drainstrom eines MOS-Transistors in Abhängigkeit von  $V_{DS}$

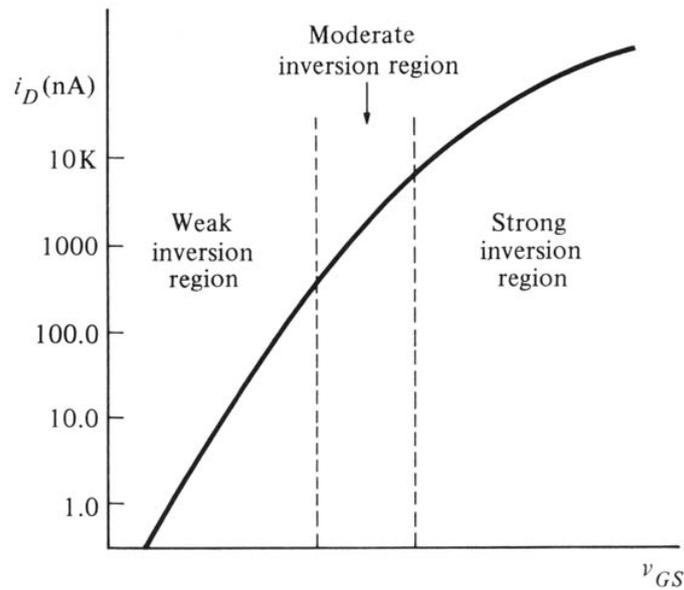
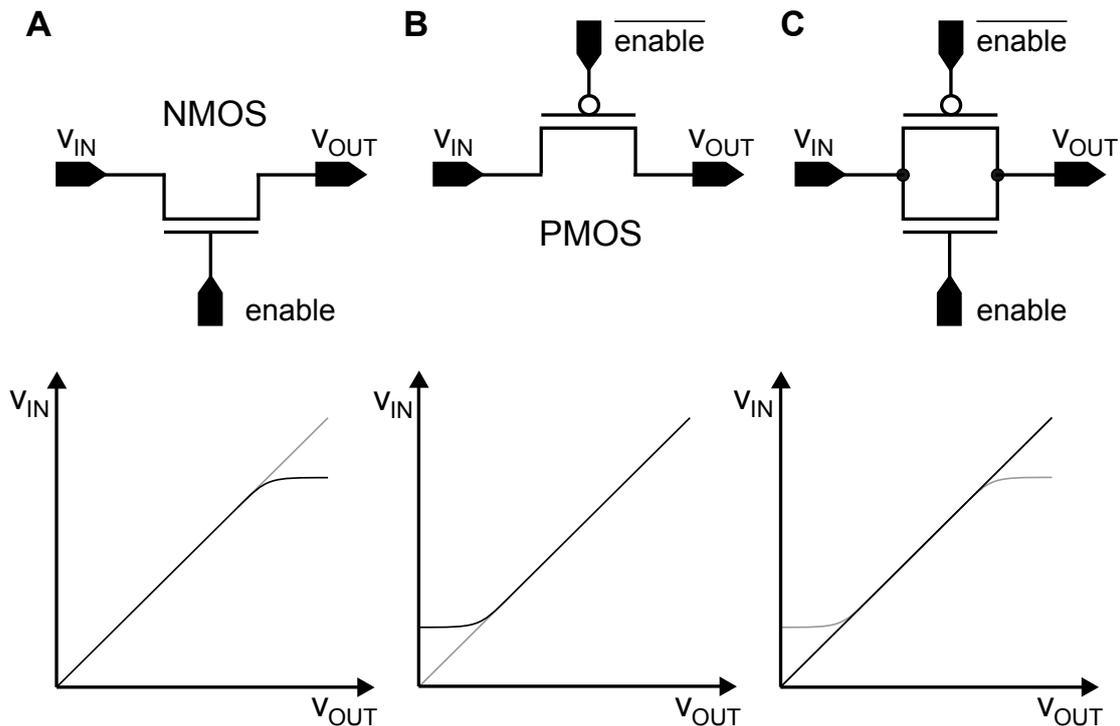


Abbildung 2.4: Übergang zwischen den verschiedenen Arbeitsbereichen eines MOS-Transistors. Entnommen aus [9].



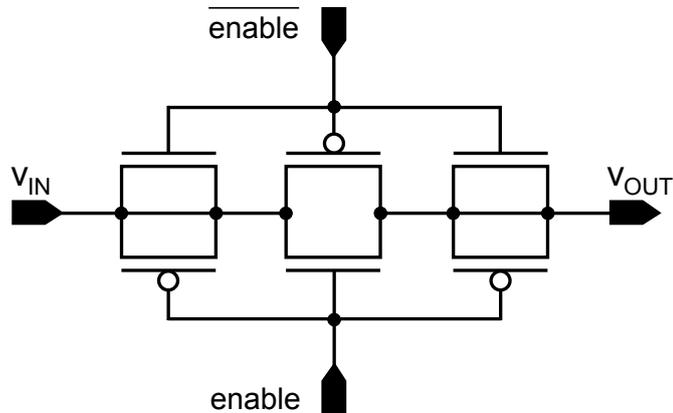
**Abbildung 2.5:** Ein Schalter realisiert durch einen einzelnen NMOS (A) bzw. PMOS (B) sowie ein *Transmission-Gate* (C)

### 2.3 Transmission-Gate und Charge Injection

Eine oft benötigte Grundschaltung ist der Schalter, sowohl ein NMOS als auch ein PMOS-Transistor kann als Schalter benutzt werden. In Abbildung 2.5 A und B werden die zu Grunde liegenden Schaltungen gezeigt.

Allerdings tritt dabei der Effekt auf, dass der NMOS Signale nahe der Versorgungsspannung abschneidet, der PMOS Signale nahe der Masse. Dies liegt daran, dass  $v_{GS}$  immer kleiner wird, je näher man den beiden Extrempunkten kommt, bis der Transistor schließlich in den *Subthreshold*-Bereich gelangt und abschaltet. Die einfachste Methode einen Schalter zu realisieren, der im ganzen Spannungsbereich arbeitet, ist eine Kombination aus NMOS und PMOS. Die Schaltung dieses sogenannten *Transmission-Gates* ist in Abbildung 2.5 C gezeigt. Allerdings muss zum Betrieb des *Transmission-Gates* das *enable*-Signal invertiert und nicht-invertiert vorliegen, um beide Transistoren gleichzeitig einschalten zu können. Die beste Linearität zwischen Eingangs- und Ausgangsspannung erhält man, wenn man die Transistoren so dimensioniert, dass die *Drainströme* gleich sind. Aufgrund der geringeren Beweglichkeit der Löcher im PMOS wird dieser meist um den Faktor drei breiter gewählt als der NMOS.

Schaltet man ein *Transmission-Gate* ab, so verschwindet die Verarmungszone unter dem *Gate*, der Kanal bricht zusammen. Die Ladungsträger im Kanal verschwinden nicht, sondern werden nach *Source* oder *Drain* gedrängt, man nennt diesen Effekt *Charge Injection*. Bei empfindlichen Anwendungen kann sich diese Ladungsmenge negativ bemerkbar machen, sie verändert die Spannung an



**Abbildung 2.6:** Ein *Fully Compensated Transmission-Gate* zur Vermeidung von *Charge Injection*

den Anschlüssen des Transistors. Da man zum Betrieb des *Transmission-Gates* ohnehin das *enable*-Signal invertiert und nicht-invertiert benötigt, kann man an beiden Enden noch zwei *Transmission-Gates* anschließen, die invertiert zum mittleren betrieben werden. *Source* und *Drain* der jeweiligen Transistoren sind kurzgeschlossen, die Breite der Transistoren wird im Vergleich zum mittleren *Transmission-Gate* halb so groß gewählt. Die Schaltung ist in Abbildung 2.6 zu sehen.

Die zusätzlichen, kurzgeschlossenen *Transmission-Gates* wirken als zuschaltbare Kondensatoren. Sie sind invertiert zum eigentlichen *Transmission-Gate* geschaltet, ihre Kanäle bilden sich genau zu dem Zeitpunkt, an dem der Kanal des mittleren *Gates* zusammenbricht, die *Charge Injection* wird kompensiert.

## 2.4 Stromspiegel

Ein in der Mikroelektronik oft eingesetztes Schaltungselement ist der *Stromspiegel*. Mit dem Stromspiegel ist es möglich, einen Strom zu kopieren, es lässt sich aber auch das Verhältnis zwischen dem generierten Strom und dem Referenzstrom einstellen. Abbildung 2.7 zeigt einen Stromspiegel.

Da *Gate* und *Drain* von  $M_1$  auf dem gleichen Potenzial liegen, ist  $M_1$  entweder ausgeschaltet oder im gesättigten Bereich. Wird  $i_{IN}$  von einer Stromquelle vorgegeben, so stellt sich automatisch ein  $v_{GS}$  gemäß Gleichung 2.4 ein. Ist die am Ausgang anliegende Spannung ausreichend groß, um  $M_2$  in Sättigung zu bringen, also  $v_{OUT} > v_{GS} + V_T$ , so gilt:

$$\frac{i_{OUT}}{i_{IN}} = \left( \frac{L_1 W_2}{W_1 L_2} \right) \left( \frac{1 + \lambda v_{DS2}}{1 + \lambda v_{DS1}} \right) \quad (2.6)$$

und wenn  $v_{DS1} = v_{DS2}$ :

$$\frac{i_{OUT}}{i_{IN}} = \left( \frac{L_1 W_2}{W_1 L_2} \right) \quad (2.7)$$

Durch die Wahl von  $W_1/L_1$  und  $W_2/L_2$  lässt sich also der Ausgangstrom einstellen.

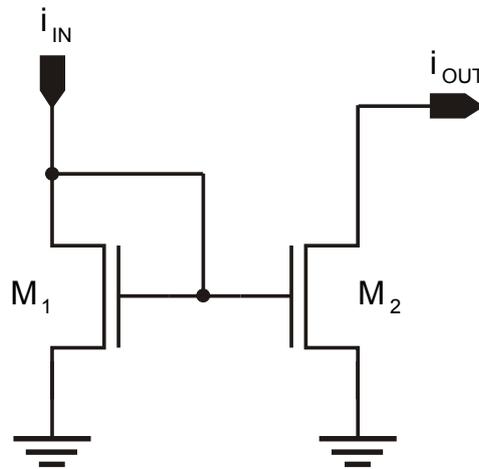
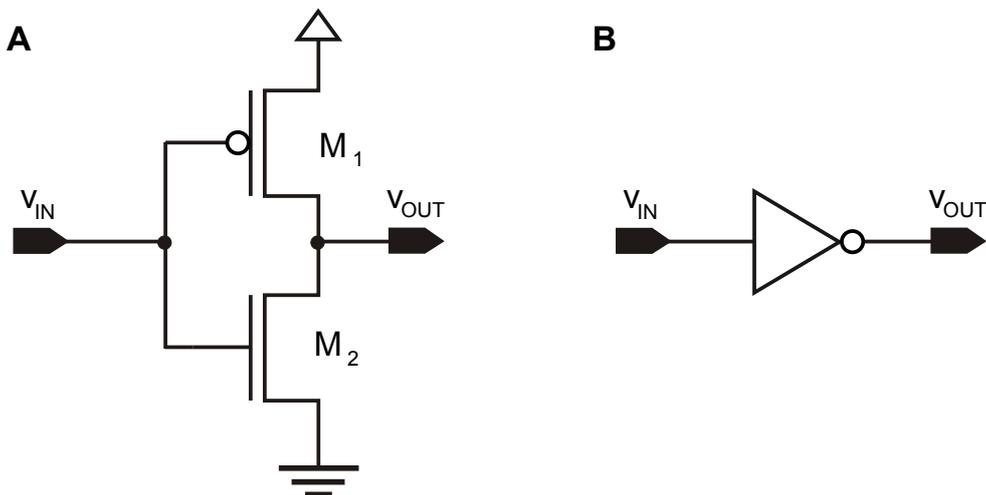


Abbildung 2.7: Ein einfacher Stromspiegel

## 2.5 Inverter

Wie im Fall des *Transmission-Gates* gesehen, ist es bei der Verwendung von Digitalschaltungen oftmals nötig, ein Signal zu invertieren. Eine einfache Schaltung mit nur zwei Transistoren ist in Abbildung 2.8 gezeigt. Liegt am Eingang die Versorgungsspannung an, so zieht der NMOS den Ausgang auf Masse. Liegt hingegen Masse am Eingang an, so zieht der PMOS den Ausgang aufs Potential der Versorgungsspannung.

Abbildung 2.8: Ein *Push-Pull*-Inverter, Schaltplan (A) und Ersatzschaltbild (B).

## 2.6 Mikrochipherstellung

Mikrochips werden heutzutage meist aus dem Halbleiter Silizium in einem fotolithografischen Verfahren hergestellt. Das Silizium wird zum Beispiel mit dem Zonenschmelzverfahren [8] gereinigt, gleichzeitig ordnet sich die Kristallstruktur, es entsteht ein großer Einkristallstab. Gängige Stabdurchmesser sind 8 oder 20, neuerdings auch 30 Zentimeter. Die Stäbe werden in Scheiben, sogenannte *Wafer*, zersägt. Der *Wafer* bildet die Grundlage für einen Mikrochip. Über Fotolithografie können ausgewählte Bereiche dotiert und damit Transistoren gebildet werden. Anschließend werden mit dem gleichen Verfahren mehrere Metalllagen aufgebracht, die die Verbindungen zu den einzelnen Transistoren herstellen. Fotolithografie besteht im Wesentlichen aus fünf Schritten:

1. Material auftragen
2. Fotolack auftragen
3. Belichten
4. Ätzen
5. Fotolack entfernen

Das Material wird in diesem Fall in belichteten Bereichen weggeätzt, in unbelichteten von Fotolack geschützt. Es kann aber auch erst nach dem Ätzen aufgebracht werden, es haftet dann nur in den vom Fotolack befreiten Bereichen, der Rest wird mit dem Fotolack entfernt. Das Material ist vom Prozessschritt abhängig. In den ersten Schritten werden Schutzlacke und polykristallines Silizium sowie Isolatormaterial verwendet. Mit den Lacken werden Bereiche vor Dotierung geschützt, das Polysilizium dient als *Gate* der Transistoren. Danach wechseln sich Isolatormaterial und Metall ab, als Metall kommen Aluminium oder Kupfer in Frage.



## Kapitel 3

# Grundlagen Neuronale Netze

Während ein Computer mathematische Algorithmen besser und schneller lösen kann als ein Mensch, gibt es viele Bereiche in denen das menschliche Gehirn dem Computer weit voraus ist, beispielsweise bei der Mustererkennung. Weiterhin ist das Gehirn robuster gegenüber Ausfall von einzelnen Komponenten und es ist in der Lage, unvollständige oder fehlerhafte Informationen zu verarbeiten. Ist zum Beispiel ein Buchstabe in einem Wort unleserlich oder sind Buchstaben vertauscht, so wird das Wort meist trotzdem erkannt, bei kaum gebremstem Lesetempo.

Der hierfür entscheidende Unterschied zwischen Gehirn und Computer liegt in der Art der Informationsverarbeitung. Ein Prozessor in einem Computer arbeitet mit hoher Geschwindigkeit sequentiell eine Reihe von Befehlen ab, heutzutage sind Taktfrequenzen im einstelligen GHz-Bereich Standard. Das Gehirn dagegen arbeitet vergleichsweise langsam, es werden Feuerraten im Bereich von einigen zehn Hz gemessen [10], dafür aber massiv parallel. Es gibt keinen Hauptprozessor, sondern viele kleine Recheneinheiten, die *Neuronen*. Dieses Prinzip wird auch in der Computerindustrie aufgegriffen. Während Supercomputer schon seit einigen Jahren über mehrere Tausend Prozessoren verfügen, werden mittlerweile auch Personal Computer mit Multicore-Prozessoren ausgeliefert. Die Anzahl der Neuronen im Gehirn wird jedoch lange nicht erreicht. So liegt die Dichte der Neuronen der Großhirnrinde bei über  $10^4$  Neuronen pro Kubikmillimeter, jedes Neuron ist mit bis zu  $10^4$  weiteren Neuronen verbunden.

Der Aufbau und das Verhalten des Gehirns werden seit Hunderten von Jahren untersucht, das Verhalten einzelner Neuronen kann inzwischen sehr genau beschrieben werden. Das bekannteste Modell wurde 1952 von Hodgkin und Huxley [11] entworfen. Es beschreibt sehr detailliert die Vorgänge an den Verknüpfungspunkten der Neuronen, den *Synapsen*. Die Erkenntnisse zum Verhalten einzelner oder weniger Neuronen werden meist *in-vitro* mit *Patch-Clamp*-Experimenten gewonnen, bei denen mit Hilfe von Elektroden das elektrische Potenzial der Neuronen gemessen wird [12].

Die Arbeitsweise von komplexeren Strukturen oder gar des ganzen Gehirns selbst ist bisher nur bruchstückhaft verstanden und Schwerpunkt intensiver Forschungen. Zum einen wird das lebende Gehirn untersucht. Der Ort und der zeitlich Verlauf von Gehirnaktivität, ausgelöst durch äußere Reize, ist dabei von besonderem Interesse. Zur Messung der Gehirnaktivität kommen meist nicht-invasive Verfahren wie *EEG*<sup>4</sup> oder *fMRT*<sup>5</sup> zum Einsatz, aber auch invasive Verfahren, zum Beispiel eines, das Aktivität mit spannungssensitiven Farbstoffen darstellt [13].

Simulation von Neuronmodellen und Neuronalen Netzwerken ist ein anderer Ansatz. Zum Beispiel wird im *Blue Brain Project* [14] versucht, eine neocorticale Säule zu simulieren, benutzt wird

---

<sup>4</sup>EEG: Elektroenzephalografie

<sup>5</sup>fMRT: funktionelle Magnetresonanztomographie

dazu ein *Blue Gene* Supercomputer. Ein Supercomputer ist nötig, um die angestrebten  $10^4$  Neuronen in einer akzeptablen Zeit simulieren zu können. Trotz Verwendung von 8192 Prozessoren besteht immer noch das Problem, dass diese nur die Formeln des verwendeten Modells berechnen können, sie repräsentieren nicht die Neuronen selbst. Durch speziell angepasste Hardware wie zum Beispiel *FPGAs*<sup>6</sup> [15, 16] oder speziell entwickelten Chips ist es möglich die Rechenzeit zu reduzieren. Weitere Rechenleistung lässt sich einsparen durch die Wahl eines einfacheren Neuronmodells. Ein weit verbreitetes Modell ist das *Integrate-and-Fire* Modell, das das Verhalten eines Neurons phänomenologisch beschreibt, die genauen inter- und intrazellulären Vorgänge aber vernachlässigt.

Weiterhin kann man spezielle Mikrochips entwickeln, sogenannte *ASICs*<sup>7</sup>, die nicht dazu dienen, eine Softwaresimulation in Hardware zu beschleunigen, sondern die eine physikalische Repräsentation des neuronalen Netzwerkes implementieren. *Spikey* ist ein solcher Chip mit 384 Neuronen und annähernd  $10^5$  Synapsen.

### 3.1 Biologische Neuronen

Es gibt viele verschiedene Arten von Neuronen [17]. So besteht der *cerebrale Kortex*, die Großhirnrinde, aus sechs verschiedenen Schichten, die nach den darin vorkommenden Neuronen unterschieden werden. Viele Neuronen sind sogenannte *Interneuronen* und dienen nur Verbindungszwecken innerhalb des Kortex. Für den Kortex spezifisch sind die *Pyramidenzellen* und *bedornen Sternzellen*. Die Pyramidenzellen sind die größten Zellen des Kortex und dienen der Weiterleitung von Signalen aus dem Kortex heraus, die bedornen Sternzellen empfangen Signale aus anderen Teilen des Gehirns. Abbildung 3.1 zeigt schematisch den Aufbau eines Neurons und dessen Verbindungen mit anderen Neuronen, die Synapsen.

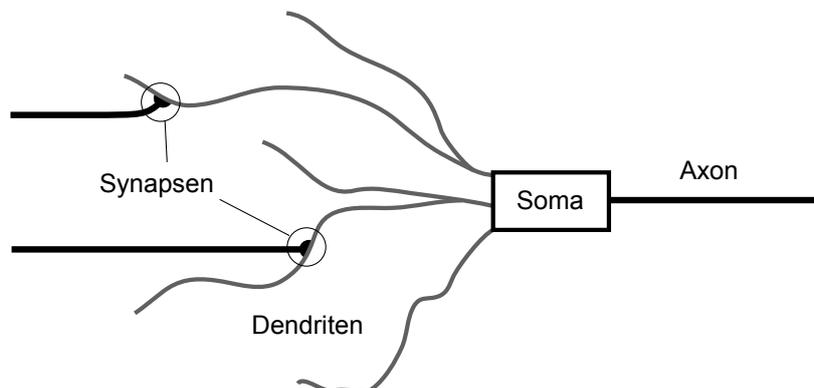


Abbildung 3.1:

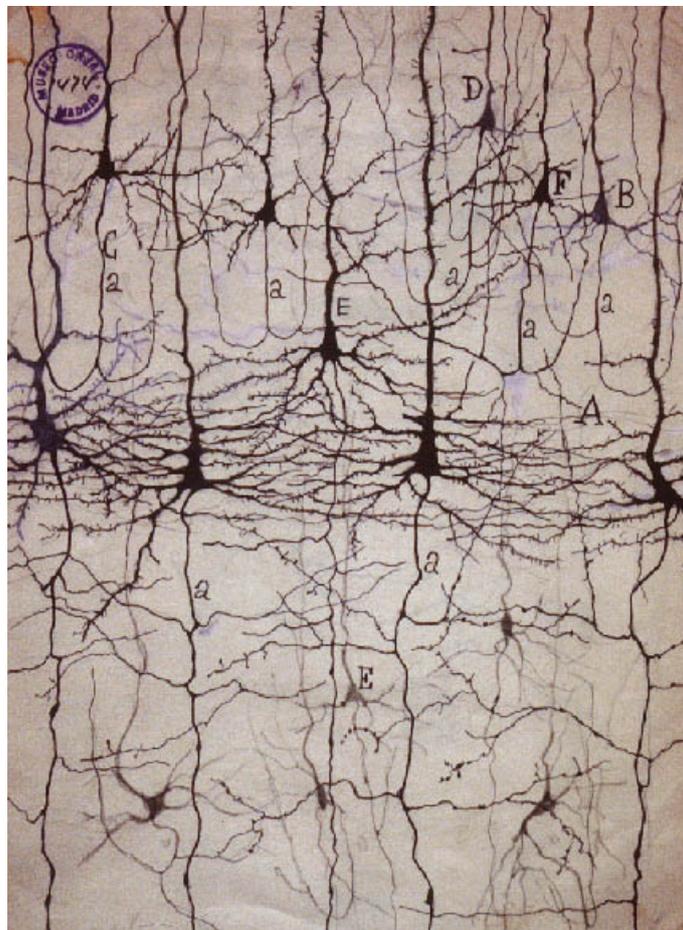
Aufgabe und Form der verschiedenen Neuronen weichen zwar mitunter stark voneinander ab, die wesentlichen Bestandteile sind jedoch die gleichen. Die meist in alle Richtungen verzweigten *Dendri-*

<sup>6</sup>FPGA: Field Programmable Gate Array; ein programmierbarer Logikbaustein

<sup>7</sup>ASIC: Application Specific Integrated Circuit

ten stellen die Eingänge der Zelle dar. Das *Axon* ist der Ausgang der Zelle. Er ist unterschiedlich stark verzweigt und kann im Fall der Pyramidenzellen sehr lang sein und bis in das Rückenmark reichen. Getrennt werden Dendriten und Axon vom *Soma*, dem Zellkern, in welchem die Eingangssignale ausgewertet werden und das Ausgangssignal generiert wird. Das Ausgangssignal, das sogenannte *Aktionspotenzial* (AP) oder auch *Spike*, ist ein elektrischer Puls, der das Axon entlangläuft und über die Synapsen am Ende des Axons auf andere Neuronen koppelt. Ein Neuron, dessen Axon an der Synapse liegt und das Signale liefert wird als *präsynaptisch* bezeichnet. Das, dessen Dendrit an der Synapse liegt als *postsynaptisch*. Ein Neuron im Kortex hat oftmals mehr als  $10^4$  Ein- und Ausgänge.

In Abbildung 3.2 wird eine Zeichnung von Ramón y Cajal gezeigt. Die dargestellten Neuronen stammen aus dem visuellen Kortex einer Katze, und wurden mit einer von Golgi entwickelten Silbernitrat-Färbemethode sichtbar gemacht. Gut zu erkennen sind einige Pyramidenzellen, deren nach unten führende Axone, im Bild mit *a* gekennzeichnet, und die nach oben und zu den Seiten hin verästelten Dendriten.



**Abbildung 3.2:** Zeichnung einiger Neuronen aus dem Visuellen Kortex einer Katze von Ramón y Cajal [18].

Das Innere des Neurons ist durch eine *Membran* von der Umgebung getrennt. Die Membran hat einen hohen elektrischen Widerstand, über ihr fällt die *Membranspannung* ab, die Potenzialdifferenz

zwischen der extrazellulären Flüssigkeit und dem Neuroninneren. Die Membranspannung entsteht durch Konzentrationsunterschiede von verschiedenen Ionenarten, zum Beispiel Kalium- oder Natriumionen. Eingebettet in die Membran sind ionenspezifische Kanäle, die nur bestimmte Ionen durchlassen. Einige dieser Kanäle sind ständig geöffnet, andere haben einen Schließmechanismus. Zusätzlich sind auch ionenspezifische Pumpen vorhanden, die dafür sorgen, dass sich im Ruhezustand Konzentrationsunterschiede ausbilden. Da das Potenzial der extrazellulären Flüssigkeit konstant ist, ist die Änderung der Membranspannung gleich der vom Potenzial im Inneren des Neurons, dem *Membranpotenzial*. Membranspannung und Membranpotenzial können deshalb synonym verwendet werden.

Gelangt ein präsynaptisches Aktionspotenzial an eine chemische Synapse, so werden dort Neurotransmitter ausgeschüttet, die durch den synaptischen Spalt diffundieren und auf der postsynaptischen Membran an Rezeptoren koppeln. Dies löst eine biochemische Signalkette aus, die damit endet, dass einige Ionenkanäle geöffnet werden und Ionen in das Neuroninnere strömen. Im Ruhezustand ist das Neuroninnere im Vergleich zur extrazellulären Flüssigkeit negativ *polarisiert*. Werden Ionen eingelassen, die das Potenzial im Inneren anheben, die also *depolarisierend* wirken, so wird die Synapse *exzitatorisch* genannt. Bei einer *inhibitorischen* Synapse dagegen wird das Neuroninnere noch negativer und die Polarisation verstärkt, man spricht von *Hyperpolarisation*.

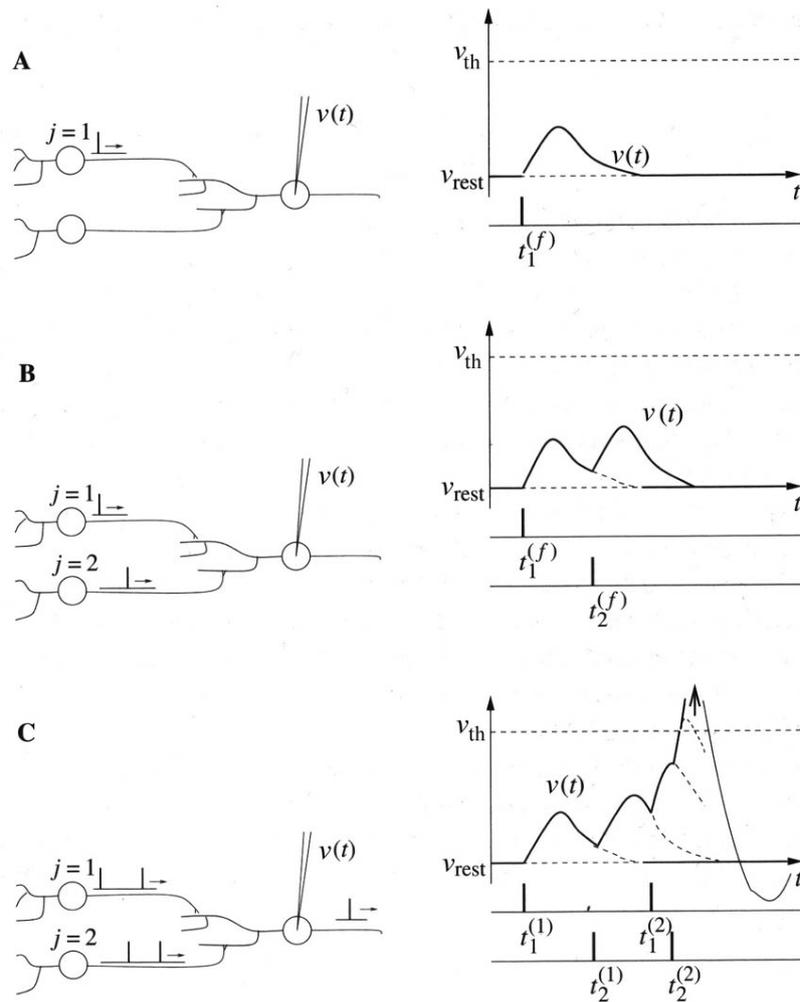
Abbildung 3.3 zeigt zwei prä- und ein postsynaptisches Neuron, die über exzitatorische Synapsen koppeln. Das Membranpotenzial des postsynaptischen Neurons wird gemessen, eine Messelektrode ist angedeutet, und der zeitliche Verlauf des Potenzials an der Messelektrode dargestellt. Während in Teil **A** die Reaktion auf ein einzelnes Aktionspotenzial gezeigt wird, ein sogenanntes *postsynaptisches Potenzial*, sieht man in Teil **B** die lineare Überlagerung von zwei dicht aufeinander folgenden postsynaptischen Potenzialen. Der Anstieg des Membranpotenzials wird durch den Ioneneinstrom hervorgerufen. Nach einiger Zeit schließen die Kanäle wieder und andere Kanäle werden geöffnet, bis das Ruhepotenzial wiederhergestellt ist.

Teil **C** zeigt den Bruch der Linearität bei Überschreitung einer Schwellspannung  $V_{th}$ . Das Neuron feuert, die postsynaptischen Potenziale werden nicht weiter aufsummiert, stattdessen wird ein elektrischer Puls generiert, der zwei Größenordnungen größer ist als ein einzelnes postsynaptisches Potenzial. Dies ist das Aktionspotenzial, oder der *Spike*, der über das Axon läuft und dann über Synapsen auf andere Neuronen koppelt. Nach der Erzeugung des *Spikes* ist das Membranpotenzial für die sogenannte *Refraktärzeit* hyperpolarisiert, ein sofortiges erneutes Feuern wird verhindert.

In der Realität sind meist viel mehr postsynaptische Potenziale nötig, um einen *Spike* auszulösen, als in Abbildung 3.3 gezeigt. Die Amplitude eines einzelnen liegt im Bereich von 1 mV, die Schwellspannung bei 20 – 30 mV über dem Ruhepotenzial. Form und Amplitude von *Spikes* verschiedener Neuronen unterscheiden sich kaum, der Feuerzeitpunkt allein beschreibt das Verhalten der Neuronen.

## 3.2 Neuron- und Netzwerkmodelle

Es gibt viele Möglichkeiten, das Verhalten von Neuronen zu modellieren und in Gleichungen auszudrücken. Versucht man die verschiedenen Modelle zu kategorisieren, so kann man das von Wolfgang Maass entworfene Generationenmodell [19] verwenden.



**Abbildung 3.3:** Reaktion des Membranpotenzials eines postsynaptischen Neurons auf exzitatorische präsynaptische Aktionspotenziale. Entnommen aus [11].

### 3.2.1 Neuronenmodelle der ersten und zweiten Generation

In die erste Generation nach Maass fallen die McCulloch-Pitts Neuronen [20], die digitale Ein- und Ausgänge verwenden. Die Eingänge werden mit einem Gewichtungsfaktor multipliziert, addiert und mit einem Schwellwert verglichen. Abhängig davon wird der Ausgang auf logisch eins oder null geschaltet. Ein bekanntes Neuronales Netzwerk, das auf McCulloch-Pitts Neuronen beruht, ist das von Frank Rosenblatt eingeführte Perzeptron-Modell[21, 22]. Der von der *Electronic Vision(s)* Gruppe entwickelte Netzwerk-Chip HAGEN beruht auf dem Perzeptron-Modell und verfügt über 256 Neuronen mit jeweils 128 Synapsen.

Die Neuronen der zweiten Generation verfügen über analoge Ein- und Ausgänge, die Höhe der Ausgangsspannung repräsentiert die Feuerrate. Modelle, die auf Feuerraten basieren, werden als *ratenkodierte* Modelle bezeichnet. Mit Feuerraten lassen sich viele Verhaltensmuster erklären, ein

klassisches Beispiel von 1926 zeigt, dass die Feuerfrequenz eines Neurons in einer Muskelspindel mit zunehmendem Druck auf den Muskel ansteigt [23].

Feuerraten lassen sich über zeitliche Mittelung der Feuerfrequenz bestimmen. Ein einzelner *Spike* trägt kaum zur Feuerrate bei. Dies steht jedoch im Widerspruch zu der Geschwindigkeit, mit der Lebewesen auf äußere Reize reagieren können. Es konnte gezeigt werden, dass Menschen in wenigen hundert Millisekunden optische Szenen erkennen und darauf reagieren können [24]. Da aber mehrere Schritte notwendig sind, in denen die Szene erkannt und dann daraufhin Muskelbewegungen eingeleitet werden, kann man davon ausgehen, dass die Neuronen keinen Mittelungsprozess durchführen, sondern schon nach wenigen *Spikes* reagieren.

### 3.2.2 Ein Modell aus der dritten Generation: Leaky Integrate-and-Fire

Modelle die nicht auf Feuerraten, sondern auf den Zeitpunkten von einzelnen *Spikes* beruhen, heißen *pulskodiert* und bilden nach Maass die dritte Generation von Neuronenmodellen. Ein oft benutztes Modell aus dieser Generation ist das *leaky Integrate-and-Fire* Modell. Ein sehr einfaches Modell mit *strombasierten Synapsen* ist in Abbildung 3.4 gezeigt.

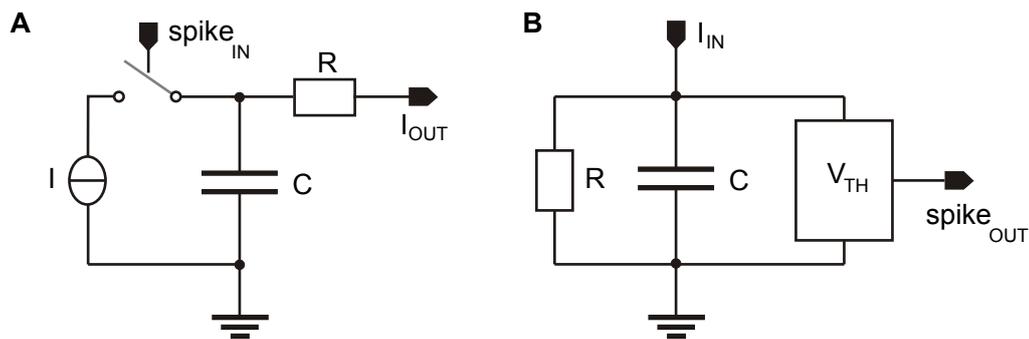


Abbildung 3.4: Synapsen- und Neuronschaltung eines einfachen *Integrate-and-Fire* Modells

In der Synapse wird zum Feuerzeitpunkt  $t^f$  kurzzeitig der Schalter geschlossen. Der entstehende Strompuls bekommt durch den aus Widerstand und Kondensator gebildeten Tiefpass eine Form, die gut mit einem postsynaptischen Potenzial übereinstimmt. Die Strompulse werden an die Neuronschaltung weitergeleitet, die aus einem Widerstand  $R$ , einem dazu parallel geschalteten Kondensator  $C$  sowie einer Resetschaltung besteht. Die über  $C$  anliegende Spannung  $V(t)$  repräsentiert die Auslenkung des Membranpotenzial aus der Ruhelage,  $R$  simuliert die Ionenkanäle und -pumpen, die das Membranpotenzial wieder in die Ruhelage zurückbringen und führt zu dem Begriff „leaky“. Überschreitet  $V(t)$  eine gewählte Schwelle  $V_{th}$ , so wird  $V(t)$  auf einen Wert  $V_r < V_{th}$  zurückgesetzt und am Ausgang ein *Spike* erzeugt. Wie beim McCulloch-Pitts Neuron muss der Einfluss verschiedener Synapsen auf ein Neuron nicht gleich groß sein. Das *Synapsengewicht*, die Stärke der Kopplung, lässt sich beispielsweise über den Ausgangsstrom der Stromquelle festlegen. Ein geringer Strom erzeugt ein kleines postsynaptisches Potenzial und entspricht einem niedrigen Synapsengewicht, eine großer Strom einem hohen.

Der Strom  $I(t)$ , die Summe der einzelnen Strompulse von allen Synapsen, teilt sich in zwei Teilströme auf:  $I(t) = I_R(t) + I_C(t)$ . Für den Strom durch den Widerstand gilt das ohmsche Gesetz:

$I_R(t) = V(t)/R$ . Mit der Definition der Kapazität  $C = Q/V$  und  $I(t) = dQ/dt$  folgt für den Strom durch den Kondensator  $I_C = C \cdot dV/dt$ . Damit gilt für den Gesamtstrom:

$$I(t) = \frac{V(t)}{R} + C \frac{dV}{dt} \quad (3.1)$$

Benutzt man die Zeitkonstante des RC-Glieds  $\tau = R \cdot C$ , so resultiert daraus folgende Differentialgleichung für das Membranpotenzial.

$$\tau \frac{dV}{dt} = -V(t) + R \cdot I(t) \quad (3.2)$$

Gleichung 3.2 gilt nur zwischen zwei Feuerzeitpunkten, den kompletten Zeitverlauf beschreibt folgende Gleichung:

$$\frac{dV}{dt} = -\frac{V(t)}{\tau} + \frac{I(t)}{C} - (\vartheta - V_r) \cdot \delta(t - t^f) \quad (3.3)$$

Ein Modell mit strombasierten Synapsen vernachlässigt allerdings die aktuelle Höhe des Membranpotenzials, die postsynaptischen Potenziale sind immer gleich groß. Beim biologischen Neuron stellt man dagegen eine starke Abhängigkeit vom Membranpotenzial fest. Eine inhibitorische Synapse wirkt hyperpolarisierend auf das Membranpotenzial. Sie lässt negativ geladene Ionen einströmen, die das Membranpotenzial absenken. Dieser Einstrom von Ionen ist aber abhängig von zwei Faktoren, dem Konzentrationsunterschied und der Membranspannung zwischen dem Neuroninneren und der Umgebung. Liegt das Membranpotenzial im Bereich der Ruhelage, so ist die Membranspannung bereits stark negativ. Ein präsynaptisches Aktionspotenzial induziert in diesem Fall nur ein sehr kleines postsynaptisches Potenzial, da die große negative Ladung im Inneren des Neurons den durch den Konzentrationsunterschied bewirkten Ioneneinstrom stark abschwächt.

### 3.2.3 Leitfähigkeits-basierte Synapsen

Wurde das Membranpotenzial dagegen von exzitatorischen Synapsen aus der Ruhelage ausgelenkt und ist nicht mehr so negativ, resultiert dies in einem größeren Ioneneinstrom, das postsynaptische Potenzial ist größer. Die größte Auswirkung haben inhibitorische Synapsen, wenn das Membranpotenzial in der Nähe der Schwellspannung liegt. Inhibitorische Synapsen nahe am Soma können das Feuerverhalten stark beeinflussen und unterdrücken, man spricht von *shunting inhibition* [25].

Das Membranpotenzial, bei dem effektiv keine Ladungen ein- oder ausströmen, da sich die von Potenzial- und Konzentrationsunterschied hervorgerufenen Ströme gegenseitig aufheben, nennt man *Umkehrpotenzial*. An diesem Punkt kehrt sich der Ionenstrom durch die geöffneten Kanäle gerade um. Inhibitorische Synapsen zeichnen sich durch ein Umkehrpotenzial  $E_i$  aus, das unter dem Ruhepotenzial  $E_r$  liegt. Dagegen liegt das exzitatorische Umkehrpotenzial  $E_x$  meist weit oberhalb der Schwellspannung, bei der das Neuron feuert. Oftmals werden für die Potenziale folgende Werte angenommen [10]:  $E_x = 0$  mV,  $E_i = -80$  mV,  $E_r = -70$  mV, sowie  $V_{th} = -55$  mV.

Modellieren kann man dieses Verhalten durch die Verwendung von *leitfähigkeitsbasierten Synapsen*. Die Schaltung der Synapsen aus Abbildung 3.4 muss sich dabei nicht zwangsläufig ändern, wohl aber die des Neurons. Abbildung 3.5 zeigt schematisch eine Neuronschaltung, die auf Leitfähigkeit beruht. Die Strompulse der Synapsen koppeln nicht mehr direkt den den Kondensator, der die Membranspannung speichert, sondern an einstellbare Widerstände. Für inhibitorische und exzitatorische

Synapsen gibt es jeweils einen Widerstand. Der zweite Anschluss des Widerstandes ist jeweils an eine Spannungsquelle angeschlossen, an der das entsprechende Umkehrpotenzial anliegt. Der Strom durch den Widerstand, der dem postsynaptischen Potenzial entspricht, ist abhängig vom aktuellen Membranpotenzial und dem anliegenden Umkehrpotenzial. Die durch das Gegenspiel von Konzentrationsunterschied und Membranspannung verursachte Abhängigkeit vom Membranpotenzial wird durch dieses Modell also gut nachgebildet.

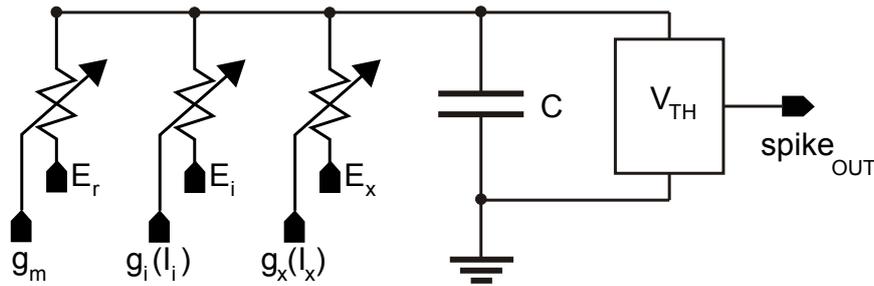


Abbildung 3.5: Integrate-and-Fire Modell: Neuronschaltung für Leitfähigkeits-basierte Synapsen

Statt Gleichung 3.2 ergibt sich:

$$C_m = g_m(V - E_r) + \sum_k p_k(t)g_k(t)(V - E_x) + \sum_l p_l(t)g_l(t)(V - E_i) \quad (3.4)$$

Das Produkt aus Leitfähigkeit und Spannung ergibt einen Strom, die Summe über  $k$  ergibt die Summe der von den exzitatorischen Synapsen induzierten Ströme, die über  $l$  die Ströme der inhibitorischen.  $g_m(V - E_r)$  repräsentiert den Leckstrom, der das Membranpotenzial wieder auf den Ruhewert zurückführt.  $C_m$  ist die Membrankapazität, mit der das Membranpotenzial gespeichert wird.  $p_k(t)$  und  $p_l(t)$  geben die Wahrscheinlichkeit an, mit der eine Synapse einen *Spike* auf das postsynaptische Neuron weiterleitet [26]. Die Leitfähigkeiten  $g_k(t)$  und  $g_l(t)$  sind proportional zu den synaptischen Gewichten.

## 3.3 Lernen und Plastizität

### 3.3.1 Lernmodelle

Am Anfang des Kapitels wurde gesagt, dass das Gehirn weniger anfällig gegenüber Fehlern ist als ein Von-Neumann-Computer. Weiterhin kann es viele Aufgaben, die schwer als Algorithmus zu formulieren sind, schneller erledigen. Ein weiterer wichtiger Punkt, der Gehirn und Von-Neumann-Computer unterscheidet, ist die Lernfähigkeit. Das Gehirn kann sich selbstständig an neue Umgebungen anpassen und sich neue Fähigkeiten aneignen. Während die elektrischen Schaltungen und Verbindungen in einem Prozessor statisch sind, können sich die Kopplungsstärken von Synapsen mit der Zeit verändern. Es wird angenommen, dass diese synaptische *Plastizität* über längere Zeiträume

der Lernfähigkeit zu Grunde liegt und sie ermöglicht.

In der Theorie der neuronalen Netze unterscheidet man zwischen verschiedenen Lernmethoden. Beim *überwachten Lernen* gibt es eine Zielfunktion für das Netzwerk, die aus einem gewählten Satz von Eingabeparametern Ausgabewerte erzeugt. Die gewünschten Ausgabewerte liegen dem Netz dabei vor. Es gibt einen Lehrer, der die Ausgabe des Netzes analysiert. Er kann direkt die Synapsengewichte manipulieren, um der Zielfunktion und der gewünschten Ausgabe näher zu kommen. Diese Art von Lernen als Form unmittelbarer Manipulation von Synapsen kommt jedoch in der Natur nicht vor, da kein Lehrer Zugriff auf die Synapsengewichte hat. Mit dieser Methode kann vielmehr abgeschätzt werden, ob ein Netzwerk überhaupt in Lage ist, eine gewünschte Funktion zu erfüllen.

Eine abgeschwächte Form von überwachtem Lernen ist das *Bekräftigungslernen*. Der Lehrer hat keinen Zugriff auf die Synapsengewichte, sondern bewertet das Verhalten des Netzes und kann es mit Belohnungs- und Strafmechanismen beeinflussen. Dies ist ein Verfahren, das oft in Tierversuchen verwendet wird. Je nachdem, ob und wie das Tier eine Aufgabe erfüllt, bekommt es eine Belohnung, den Weg dahin muss es jedoch selbst finden.

Es ist jedoch auch Lernen ohne Lehrer möglich. Beim *unüberwachten Lernen* organisiert das Netzwerk sich selbst und verändert die Synapsengewichte selbstständig. Im Gegensatz zum überwachten Lernen kennt das Netz die gewünschten Ausgabewerte nicht. Es gibt keine zielgerichtete Entwicklung, die Synapsengewichte werden nach gewissen Regeln abhängig von den Eingabewerten verändert.

### 3.3.2 Spike Time Dependent Plasticity

Im Jahr 1949 formulierte Donald Hebb seine berühmte Lernregel. Sie besagt, dass das Synapsengewicht zwischen zwei Neuronen verstärkt wird, wenn das präsynaptische Neuron oft zum Feuern des postsynaptischen beigetragen hat. Hepp verfasste seine Lernregel basierend auf theoretischen Vorhersagen, später konnte sie experimentell verifiziert und ergänzt werden. Das Synapsengewicht wird abgeschwächt, wenn das präsynaptische Neuron nicht zum Feuern der postsynaptischen beiträgt, wenn ein präsynaptisches Aktionspotenzial also zum Beispiel kurz nach dem Feuern des postsynaptischen Neurons eintrifft. Eine quantitative Formulierung dieses Prinzips ist die *Spike Time Dependent Plasticity* (STDP), genauer beschrieben in [27]. Die Hebb'sche Lernregel wird durch Plastizität, die auf den Feuerzeitpunkten der Neuronen beruht, umgesetzt. Verstärkung des Synapsengewichts tritt ein, wenn das präsynaptische Neuron kurz vor dem postsynaptischen feuert, wenn das übermittelte postsynaptische Potenzial also effektiv zur Überschreiten der Schwellspannung beigetragen hat.

### 3.3.3 Short Term Plasticity

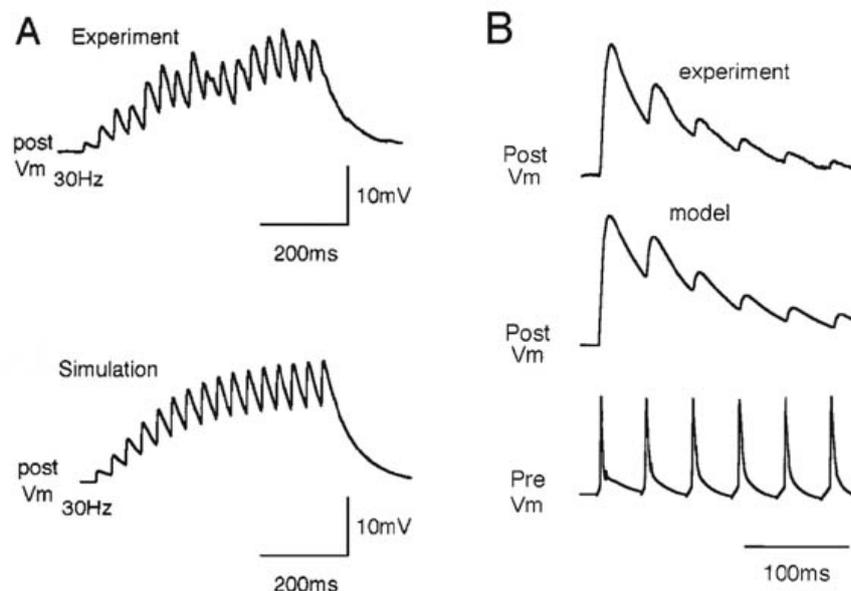
Mit STDP lassen sich Effekte beschreiben, die länger als einige Minuten anhalten. Es gibt jedoch auch kurzzeitige Veränderungen der Synapsengewichte: *Short Term Plasticity* (STP). Mit *Short Term Plasticity* wird ein Adaptionsmechanismus von Synapsen beschrieben, bei dem kurz aufeinander folgende präsynaptische Aktionspotenziale unterschiedliche Auswirkungen auf das Membranpotenzial haben. Bei *Short Term Depression*, zu deutsch: Kurzfristige Senkung, reagiert das Membranpotenzial immer weniger auf anhaltende, hochfrequente präsynaptische Aktionspotenziale. Die Ursachen können vielfältig sein. Die Übertragung des Aktionspotenzial über eine Synapse ist abhängig von mehreren biochemischen Vorgängen, die mitunter eine Erholungszeit zwischen zwei *Spikes* erfordern. Zum

Beispiel kann der Vorrat an synaptischen Vesikeln erschöpft sein. Weiterhin ist es möglich, dass die Neurotransmitter des vorherigen *Spikes* noch nicht vollständig an den Rezeptoren abgebaut sind [28].

Das Gegenteil bewirkt *Short Term Facilitation*, kurzzeitige Verstärkung: Schnell aufeinander folgende Aktionspotenziale resultieren in immer stärkeren Reaktionen der Membranspannung. Die Ursachen sind nicht genau bestimmt.

Ob und welche *Short Term Plasticity*-Effekte auftreten ist abhängig davon, welche Art von Neuronen die Synapse verbindet. Von Pyramiden-Zellen auf Interneurone konnte *Facilitation*, zwischen Pyramiden-Zellen *Depression* gemessen werden [29]. Nach einiger Zeit ohne Aktivität klingen beide Effekte wieder ab.

Beispiele von *Short Term Plasticity* werden in Abbildung 3.6 gezeigt.



**Abbildung 3.6:** Beispiele von *Short Term Facilitation* (A) und *Depression* (B). Entnommen aus [30] bzw. [31]

Ein einfaches Modell um *Short Term Plasticity* nachzubilden benutzt *synaptische Wirksamkeiten* (*efficacies*) [31, 30, 32]. Im Fall von *Depression* erfährt ein einzelnes Aktionspotenzial ohne vorherige Aktivität die *maximale Wirksamkeit*  $A_{SE}$ , verbraucht dabei aber die Wirksamkeit  $U_{SE}$  (*used*). Für folgende Aktionspotenziale spaltet sich die maximale in die *inaktive Wirksamkeit*  $X_I$  und die *wiederhergestellte Wirksamkeit*  $X_R$  (*recovered*) auf, es gilt  $X_R + X_I = 1$ , wobei nach jedem Aktionspotenzial instantan die Wirksamkeitsmenge  $U_{SE} \cdot X_R$  von  $X_R$  nach  $X_I$  transportiert wird. Die nachfolgenden Aktionspotenziale erfahren nur noch den wiederhergestellten Anteil von  $A_{SE}$ , wiederhergestellt deshalb, weil  $X_R$  kontinuierlich zulasten der inaktiven Wirksamkeit wiederhergestellt wird. Die Zeitkonstante für diesen Prozess heißt  $\tau_{rec}$ , der Zeitpunkt, zu dem ein Aktionspotenzial die

Synapse erreicht  $t_{AP}$ . Drückt man dies als Formel aus, so ergibt sich:

$$\frac{dX_I}{dt} = -\frac{X_I}{\tau_{rec}} + U_{SE} \cdot X_R \cdot \delta(t - t_{AP}) \quad (3.5)$$

Für die synaptische Leitfähigkeit  $g^{\max}$  gilt:

$$g^{\max} = A_{SE} \cdot X_R \quad (3.6)$$

Das einfachste Modell für *Facilitation* ergibt sich durch vertauschen von  $X_R$  und  $X_I$ , statt  $\tau_{rec}$  wird  $\tau_{facil}$  benutzt.

Mit beiden Plastizitätseffekten zusammen ergibt sich für die synaptischen Leitfähigkeiten aus Gleichung 3.4:

$$g_{k,l}(t) = w_{k,l}(t) \cdot g_{k,l}^{\max}(t) \quad (3.7)$$

*Spike Time Dependant Plasticity* verändert langfristig die Gewichtungsfaktoren  $w_{k,l}(t)$ ,  $g_{k,l}^{\max}(t)$  wird von der *Short Term Plasticity* Schaltung gesteuert.

### 3.4 High-Conductance State

Vergleicht man die Ergebnisse von *in-vitro* und *in-vivo Patch-Clamp*-Messungen, so stellt man fest, dass sich Membranpotenzial und Leitfähigkeit stark unterscheiden. Im Wachzustand unterliegt das Membranpotenzial starken Schwankungen, es liegt im Bereich von -60 bis -65 mV und damit deutlich höher als bei *in-vitro* Messungen mit -70 bis -80 mV [33]. Strominjektionen haben bei *in-vitro* Messungen stärkere Auswirkungen auf das Membranpotenzial als bei *in-vivo* Experimenten im Wachzustand. Daraus lässt sich schließen, dass die Leitfähigkeit im Wachzustand deutlich größer ist als im inaktiven Fall.

Das fluktuierende, depolarisierte Membranpotenzial und die erhöhte Leitfähigkeit im Wachzustand haben ihren Ursprung in der dauerhaft vorhandenen Aktivität im Netzwerk. Ein einzelnes Neuron empfängt ständig *Spikes*, und löst auch selbst spontan *Spikes* aus. Dieser Zustand wird *High-Conductance State* genannt. In [10] wird gezeigt, dass die Lage des Membranpotenzials in erster Näherung von dem Verhältnis der exzitatorischen zu der inhibitorischen Leitfähigkeit abhängig ist.



# Kapitel 4

## Spikey-Chip

Spikey ist ein VLSI<sup>8</sup> Mikrochip, auf dem ein neuronales Netzwerk implementiert ist. Er dient nicht dazu, eine Simulation von einem Netzwerk in Software zu beschleunigen, sondern beherbergt Schaltungen, die die Neuronen und Synapsen selbst repräsentieren. Für viele wichtige Parameter gibt es im Chip Äquivalenzen in der gleichen physikalischen Größe, zum Beispiel wird das Membranpotenzial eines Neurons auch durch ein Potenzial dargestellt. Der Chip wird in einem 180 nm Prozess hergestellt und beinhaltet 384 Neuronen mit jeweils 256 Synapsen, zusammen also annähernd  $10^5$  Synapsen. Der große Vorteil der Realisierung eines neuronalen Netzwerks in Hardware ist die Geschwindigkeit. Die Zeitkonstanten können im Vergleich zum biologischen Vorbild um den Faktor  $10^5$  reduziert werden, für die Simulation von 1 s in biologischer Zeit benötigt der Chip nur  $10 \mu\text{s}$ . Es gibt mittlerweile zwei Versionen des Chips, in der zweiten wurden einige Fehler beseitigt.

### 4.1 Neuron-Modell

Das implementierte Netzwerk ist pulskodiert, die Feuerzeitpunkte der Neuronen bestimmen das Netzwerkverhalten. Die Grundlage für die Neuronen bildet das *leaky Integrate-and-Fire* Modell, welches jedoch in einigen Punkten erweitert wurde. So ist der Feuerzeitpunkt nicht nur vom Membranpotenzial, sondern auch von dessen Ableitung abhängig. Eine spezielle Schaltung überwacht das Membranpotenzial und löst einen digitalen *Spike* aus, sobald die Schwellwertbedingungen erfüllt sind. Die *Spiques* werden an Synapsen des gleichen Chips weitergeleitet, können aber auch an andere Chips versendet werden.

Die Synapsen sind leitfähigkeitsbasiert, die postsynaptischen Potenziale werden jedoch aufwendiger generiert, als in Abschnitt 3.2.3 beschrieben. Die verwendete Schaltung erlaubt die Erzeugung von realistischeren postsynaptischen Potenzialen, zusätzlich können *Short Term Plasticity* Effekte verwirklicht werden. Gleichzeitig wird auch der *High-Conductance State* nachgebildet.

Auch die zweite in Abschnitt 3.3 beschriebene Plastizitätsvariante ist im Chip integriert: STDP. Die Synapsen messen die Korrelation zwischen prä- und postsynaptischen Aktionspotenzialen, ihre Gewichte können dann entsprechend verändert werden. Die Speicher für die Synapsengewichte sind lokal bei den Synapsen untergebracht. In weiteren Speichern ist es möglich, annähernd 3000 analoge Parameter zu speichern, mit denen sich das Verhalten des Netzwerks vielfältig einstellen lässt. Die analogen Parameter werden anhand der digital gespeicherten Werte von einem internen DAC<sup>9</sup>

---

<sup>8</sup>VLSI: very large scale integration

<sup>9</sup>DAC: Digital to Analog Converter

erzeugt. Auf diese Weise ist es möglich, viele verschiedene Neuronarten mit dem Chip zu simulieren. Die Aktualisierung der Parameter interferiert nicht mit der Arbeitsweise des Netzwerks selbst, es ist möglich, sie im laufenden Betrieb durchzuführen.

Durch den Zusammenschluss von mehreren Spikey-Chips ist es möglich größere Netzwerke zu realisieren. Zur Kommunikation mit anderen Chips stehen zwei unidirektionale 16 Bit LVDS<sup>10</sup> Verbindungen zur Verfügung, mit einer Datenrate von jeweils 1,3 GBytes/s.

Dass die *Spikes* digital sind, erleichtert die Verbindung von mehreren Spikey-Chips erheblich. Sie können direkt mit einem Transportprotokoll an andere Netzwerkteile verschickt werden. Das bei VLSI basierten neuronalen Netzwerken oft benutztes Protokoll AER<sup>11</sup> [34] kann in diesem Fall allerdings nicht benutzt werden. Das AER-Protokoll transportiert die Ereignisse im neuronalen Netz, die *Events*, zeitkontinuierlich, Laufzeitunterschiede über die AER-Verbindungen machen sich direkt als Laufzeitunterschiede im neuronalen Netz bemerkbar. Aufgrund der hohen Geschwindigkeit des Spikey-Chips und den unterschiedlich langen Verbindungen zwischen mehreren Chips würden sich Laufzeitunterschiede stark bemerkbar machen.

Statt eines zeitkontinuierlichen Transportes werden die *Events* mit einem Zeitstempel versehen. Das Problem der unterschiedlich langen Laufzeiten über eine Verbindung reduziert sich also auf das Unterschreiten einer maximalen Verzögerung. Gelangen die *Events* innerhalb der maximalen Verzögerung an ihr Ziel, können sie anhand der Zeitstempel sortiert und in das Netzwerk gespeist werden. Die maximale Verzögerung ergibt sich aus der Länge des Axons im biologischen Vorbild und der Zeitspanne, die ein *Spike* vom Soma bis zum Ende des Axons benötigt. Je nach zugrunde liegendem zu simulierendem Neuron entspricht dies einer maximalen Verzögerung von 50 bis 500 ns [35], Zeitvorgaben, die sich gut mit den LVDS Verbindungen erreichen lassen.

## 4.2 Chipaufbau

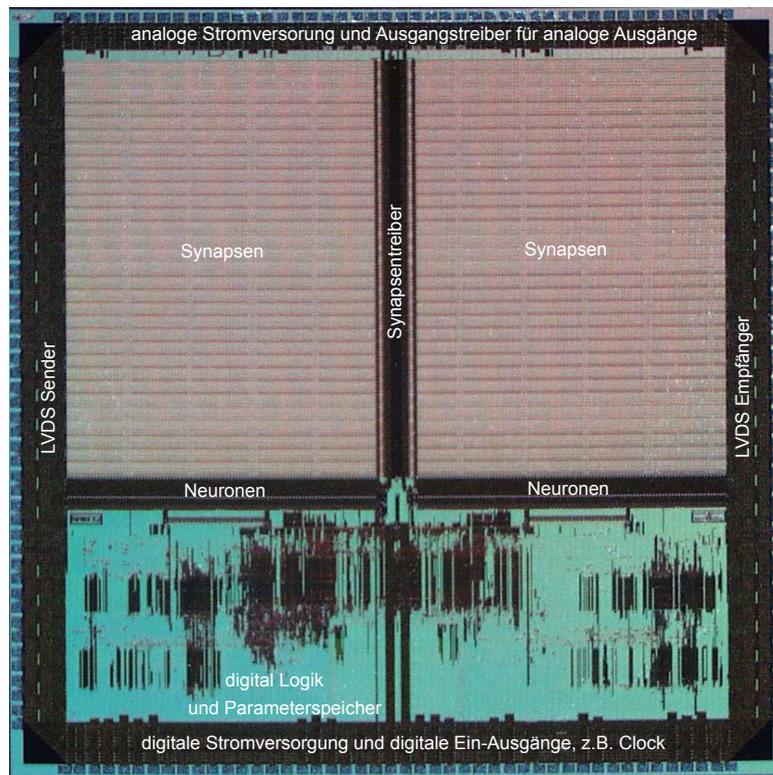
Das neuronale Netz selbst nimmt etwa 2/3 der Chipfläche ein, im übrigen Bereich befinden sich Digitalschaltungen, die für die Kommunikation mit anderen Chips und für die Speicherung und Erzeugung der 3000 analogen Parameter zuständig sind. Den größten Anteil an der Fläche haben die beiden Synapsenblöcke, die jeweils aus einer Matrix von 256 Zeilen  $\times$  192 Spalten bestehen. Unter den Zeilen liegen die 192 Neuronen pro Block. Die fast 100000 Synapsen beanspruchen den größten Anteil der Chipfläche und sind aus diesem Grund möglichst einfach aufgebaut. Jeweils 192 Synapsen teilen sich einen *Synapsentreiber*, der die Erzeugung der postsynaptischen Potenziale vorbereitet.

Abbildung 4.1 zeigt eine Fotografie des Spikey 2 Chips. Den Digitalteil im unteren Drittel kann man gut von den beiden darüber liegenden Synapsenblöcken unterscheiden. In den schwarzen Bereichen unter den Synapsenblöcken liegen die Neuronschaltungen, zwischen den Synapsen liegen die Synapsentreiber. An der rechten Außenseite des Chips sind die LVDS Empfänger platziert, an der linken die LVDS Sender.

Die Kernkomponenten des neuronalen Netzwerkes und ihre Verschaltung werden in Abbildung 4.2 gezeigt. Die Synapsentreiber generieren aus den digitalen *Spikeevents* einen dreieckförmigen Spannungsverlauf, der an die Synapsen weitergeleitet wird. Für jeden Synapsentreiber ist einstellbar, ob er die *Events* von einem zugeordneten Neuron aus dem eigenen Block, dem Nachbarblock oder von

<sup>10</sup>LVDS: low voltage differential signaling; Schnittstellen-Standard für Hochgeschwindigkeits-Datenübertragung

<sup>11</sup>AER: address event representation;



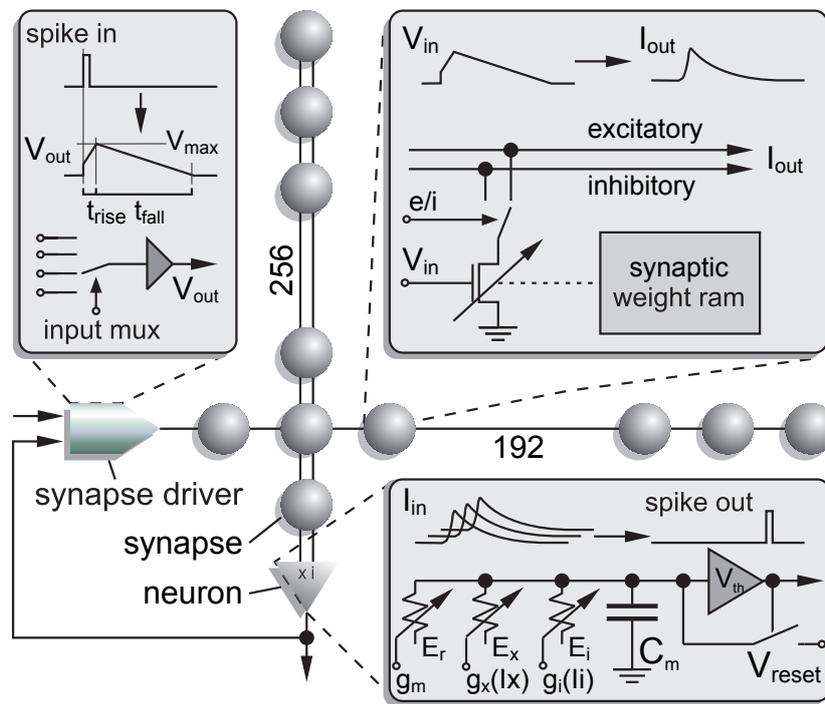
**Abbildung 4.1:** Fotografie des Spikey 2 Chips

einem externen Eingang empfangen soll. Weiterhin findet die Erzeugung der *Short Term Plasticity* in den Synapsentreibern statt, in Abschnitt 6.3.1 wird genauer auf die Synapsentreiber eingegangen und der Schaltungsaufbau diskutiert.

Die Synapsen basieren auf 15 als Stromsenken verwendeten NMOS-Transistoren, an deren Eingänge der Synapsentreiber angeschlossen ist. Das Synapsengewicht, das in einem 4 Bit statischem RAM gespeichert ist, bestimmt, wie viele der Stromsenken an den gemeinsamen Ausgang angeschlossen sind. Die NMOS-Transistoren befinden sich im Bereich der schwachen Inversion, der Ausgangstrom ist deshalb gemäß Gleichung 2.5 exponentiell abhängig von der *Gate*-Spannung. Der dreieckförmige Spannungsverlauf wird deshalb in einen Stromverlauf umgewandelt, der dem biologischen postsynaptischen Potenzial sehr nahe kommt. Mit Hilfe einer weiteren Steuerleitung kann die Art der Synapse eingestellt werden. Alle exzitatorischen Synapsen einer Spalte koppeln auf denselben Ausgang, der die Summe der Ströme  $I_x$  auf die zugehörige Neuronschaltung weiterleitet. Selbiges gilt für die inhibitorischen Synapsen und den Strom  $I_i$ .

Die Neuronen basieren auf der in Abbildung 3.5 gezeigten Schaltung.  $I_x$  und  $I_i$  werden in die Leitfähigkeiten  $g_x(I_x)$  und  $g_i(I_i)$  umgerechnet, die den beiden Summentermen aus Gleichung 3.4 entsprechen. Damit ergibt sich als Differenzialgleichung für das Membranpotenzial:

$$C_m = g_m(V - E_r) + g_x(I_x)(V - E_x) + g_i(I_i)(V - E_i) \quad (4.1)$$



**Abbildung 4.2:** Kernkomponenten des neuronalen Netzwerkes: Synapsentreiber, Synapsen und Neuronen. Entnommen aus [32]

Der Strom, der das Membranpotenzial in die Ruhelage  $E_r$  zurückführt, fließt durch  $g_m$ , und kann für jedes Neuron separat eingestellt werden. Die Umkehrpotenziale  $E_r$ ,  $E_i$  und  $E_x$ , sowie die Schwellspannung  $V_{th}$  und die Resetspannung  $V_{reset}$  lassen sich für Gruppen von 96 Neuronen einstellen.

Ein Komparator vergleicht das Membranpotenzial mit  $V_{th}$  und löst gegebenenfalls einen *Spike* aus. Der Komparator kann so eingestellt werden, dass nicht nur der Wert des Membranpotenzials, sondern auch die Ableitung Auslösekriterien sind. Die endliche Geschwindigkeit des Vergleichvorgangs ist dafür verantwortlich, sie kann über einen Biasstrom eingestellt werden.

Der *Spike* gelangt über den aufgespaltenen Axon des Neurons an vier Ziele. Dies ist im Blockdiagramm des Spike-Chips zu sehen, das in Abbildung 4.3 gezeigt wird. In der Mitte befinden sich die beiden Synapsenblöcke, die von den Synapsentreibern links und rechts davon ihre Eingaben bekommen. Die Ausgänge der Synapsen gehen an die darunter liegenden Neuronen. Das Neuron schickt seine *Spike*s an den zugehörigen Synapsenblock, zwei bestimmte Synapsentreiber und an den *Asynchronous Priority Encoder*. Ein fünfter Pfeil, der ebenfalls den Neuronblock verlässt, symbolisiert keinen Transport von *Spike*s, über diese Verbindung können die Membranpotenziale von außen gemessen werden.

- Im Synapsenblock wird der *Spike* gebraucht, damit die Synapsen die Korrelation zwischen prä- und postsynaptischen Aktionspotenzialen messen können.

- Jedem Neuron sind zwei Synapsentreiber mit den zugehörigen Synapsenzeilen zugeordnet, einer für jede Synapsenblock. Ist der Ausgang des Neurons nur mit Neuronen auf demselben Chip verbunden, werden die *Events* ausschließlich über diese Verbindungen weitergeleitet.
- Für Verbindungen zu anderen Chips wird der *Spike* in ein *Event* kodiert und dann versendet. Die Kodierung beginnt im *Asynchronous Priority Encoder*.

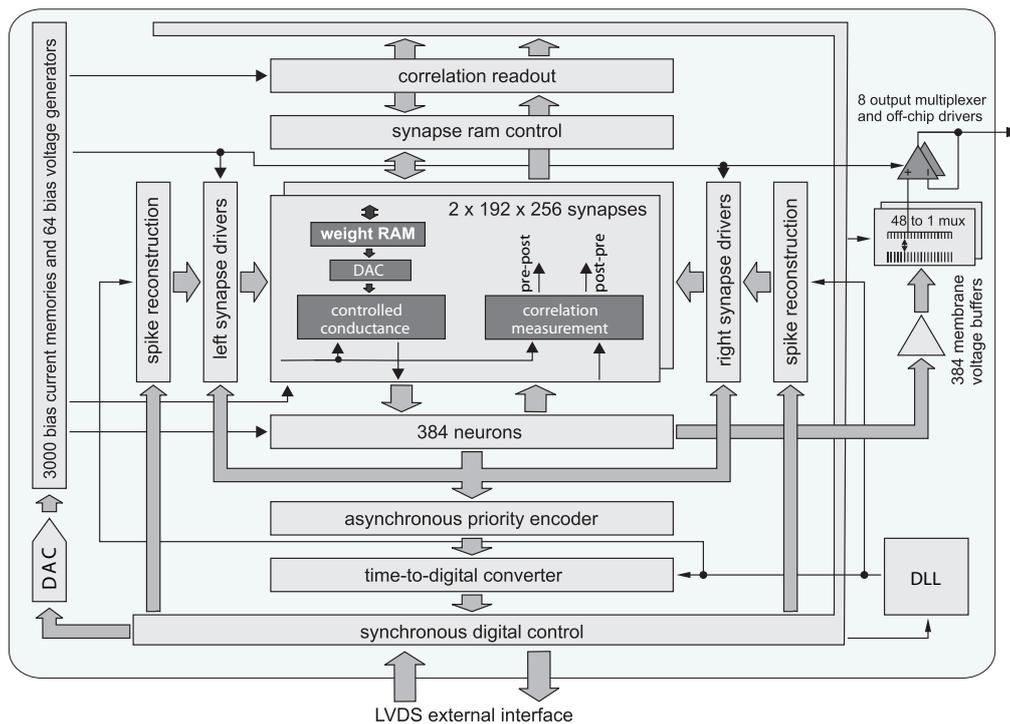


Abbildung 4.3: Blockdiagramm des Spikey-Chips. Entnommen aus [35]

Der Asynchronous Priority Encoder detektiert feuernde Neuronen und schickt die entsprechende Neuronnummer an die nächste Stufe. Feuern mehrere Neuronen gleichzeitig, wird die Neuronnummer mit höchster Priorität ausgewählt und weitergeleitet, anschließend der Reihe nach jeweils das mit der nächst höchsten Priorität. Hierbei werden die *Events* geringfügig verzögert. Die nächste Stufe bildet der *Time-to-Digital Converter*, dieser versieht das *Event* mit einem Zeitstempel. Die Taktfrequenz, mit der der Spikey-Chip betrieben wird, wird zur Erhöhung der Präzision in 16 Subtakte aufgeteilt. Ein vollständiger Zeitstempel besteht aus einer Takt- und einer Subtaktnummer.

Um nicht Gefahr zu laufen, zu viele *Events* zu verzögern, gibt es sechs *Priority Encoder* und *Time-to-Digital Converter*, die jeweils für Blöcke bestehend aus 64 Neuronen zuständig sind.

Anschließend wird das *Event* in das Netzwerk gespeist. Eine externe Logik [36] ist dafür zuständig, die *Events* dem richtigen Chip und Neuron zuzustellen, zusätzlich bekommen sie einen neuen Zeitstempel, der den Feuerzeitpunkt im Zielchip angibt.

Im Zielchip werden die *Events* an die *Spike Rekonstruktionseinheiten* weitergeleitet, die in Abbildung 4.3 benachbart zu den Synapsentreibern angeordnet sind. Diese warten bis zum Feuerzeitpunkt und lassen dann von den Synapsentreibern die postsynaptischen Potenziale generieren.

# Kapitel 5

## Test-Hardware

### 5.1 Backplane und Nathan

Zu Beginn der Diplomarbeit wird ein Messaufbau hergestellt. Zur Verfügung stehen eine *Backplane* sowie mehrere *Nathan*-Platinen [37]. Die *Nathan*-Platinen, in Kurzform: *Nathan*, sind die Träger des jeweiligen neuronalen Netzwerk-Chips, ein FPGA<sup>12</sup> dient zur Ansteuerung und Kommunikation. Weiterhin stellt *Nathan* Speicherressourcen zur Verfügung, zwei SRAM Chips mit je 512 KB, sowie eine Speicherbank, die DDR-SDRAM Module mit einer Größe von maximal 2 GB aufnehmen kann.

Eine *Backplane* hat 16 Steckplätze für *Nathan*-Platinen aufnehmen und stellt die Verbindung zwischen den *Nathans* bzw. deren Netzwerkchips her. Über ein *SCSI*<sup>13</sup>-Kabel wird die *Backplane* mit einem PC verbunden, mit dessen Hilfe die Hardware konfiguriert und die Experimente durchgeführt werden können. Die Verbindung zum PC wird über eine speziell entwickelte PCI-Steckkarte hergestellt [39]. Abbildung 5.1 zeigt eine *Backplane* mit vier eingesteckten *Nathan*-Platinen.

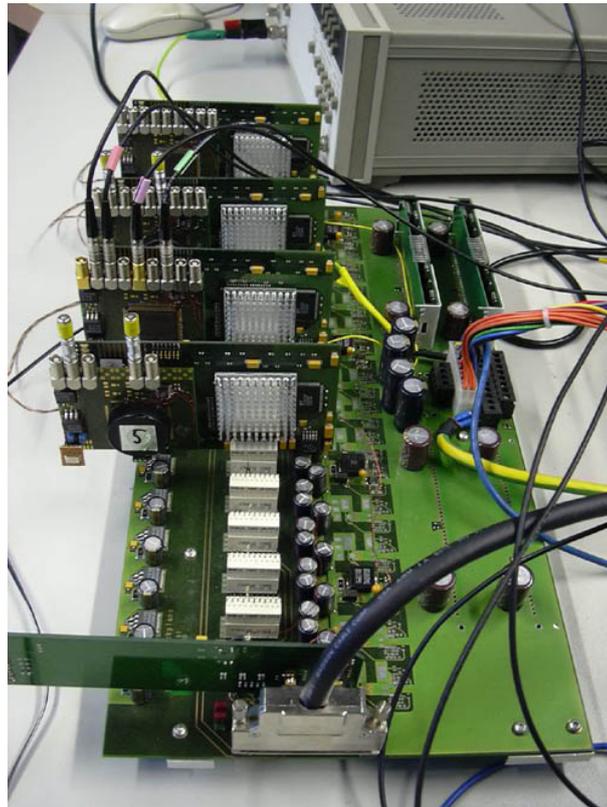
Die zur Verfügung stehenden *Nathans* müssen zunächst modifiziert werden. Der bisher verwendete *HAGEN*-Chip kann direkt in den auf *Nathan* verbauten Sockel eingesetzt werden, der zu untersuchende *Spikey*-Chip benötigt jedoch zusätzliche Spannungen und Signalleitungen. Des Weiteren wären die großen parasitären Kapazitäten des Sockels störend für die schnelleren Signalleitungen des *Spikey*-Chips. *Spikey* wird auf eine Zusatzplatine montiert, die sich auf *Nathan* aufstecken lässt. Diese Tochterplatine (*Recha*) stellt die zusätzlichen Spannungen zur Verfügung, sowie weitere Ausleseelektronik. In Abbildung 5.2 wird eine *Nathan*-Platine mit bereits entferntem *HAGEN*-Sockel gezeigt.

Da auf der Rückseite der *Recha*-Platine Bauteile untergebracht sind, muss der *HAGEN*-Sockel auf *Nathan* ausgelötet werden. Ein erstes Vorgehen scheint Erfolg versprechend: Ein Miniaturkran zieht den *HAGEN*-Sockel aus der *Nathan*-Platine, während diese im Lötöfen erhitzt wird. Der Miniaturkran wird auf *Nathan* gestellt, der Ausleger an den *HAGEN*-Sockel geklebt. Eine Sprungfeder wird den Ausleger beim Löten anheben, zunächst ist dieser allerdings noch mit einem Kunststoffstift fixiert, um mechanische Spannungen und etwaiges Verkannten zu verhindern.

---

<sup>12</sup>benutzter FPGA: Virtex II-pro, Typ XC2VP7 [38]

<sup>13</sup>SCSI: Small Computer System Interface; eine standardisierte parallele Schnittstelle zur Datenübertragung zwischen Geräten auf einem Computer-Bus, in diesem Fall werden jedoch nur das Kabel und die Steckverbinder benutzt, jedoch nicht das Protokoll



**Abbildung 5.1:** Backplane mit vier eingesteckten Nathans.

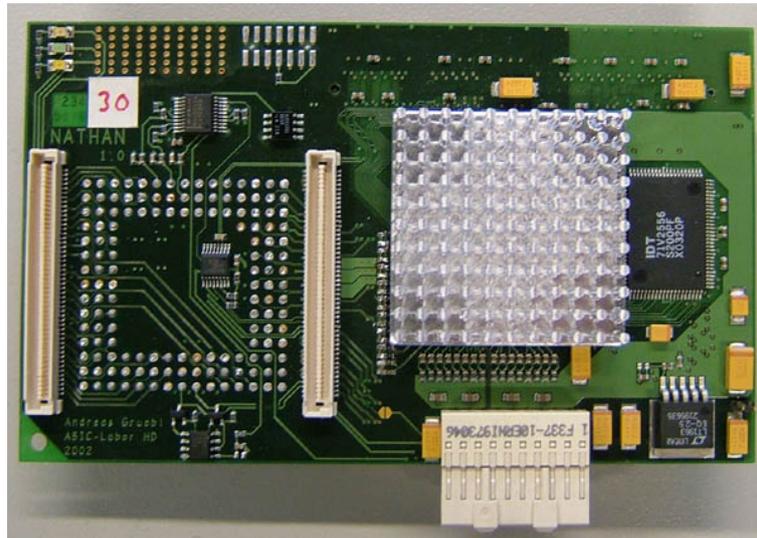
Im Lötoven wird eine spezielle, teflonhaltige Flüssigkeit verdampft. Der Dampf steigt vom Boden des Ofens auf, bis er das Nathan erreicht und dort kondensiert. Die Platine wird dabei auf den Siedepunkt der Flüssigkeit aufgeheizt. Dieser liegt über dem Schmelzpunkt des Lötzinns, die Bauteile und der HAGEN-Sockel sind nun locker, an der Unterseite der Platine haften die Bauteile nur noch aufgrund der Adhäsion an der Platine. Der Dampf steigt nun weiter auf und erhitzt den Kran, samt Fixierstift. Dieser wird weich, die Kraft der Sprungfeder genügt nun, um ihn zu verformen. Der Ausleger hebt sich und zieht den HAGEN-Sockel aus Nathan. Abbildung 5.3 zeigt eine Nathan-Platine mit aufgesetztem Entlöt-Kran und ausgelötetem HAGEN-Sockel.

Leider erweist sich die Methode als nicht ganz zuverlässig. Die *Ball-Grid*<sup>14</sup> Verbindung von FPGA und Nathan scheint das erneute Löten nicht zu verkraften. Beim zweiten Nathan, dem auf diese Weise der HAGEN-Sockel entfernt wird, ist eine der *Ball-Grid* Verbindungen unterbrochen.

Da noch eine ganze Reihe von Nathans umgerüstet werden sollen, kommt das Auslöten per LötKolben und Entlöt-pumpe nicht in Frage. Es wird daraufhin ein Lötstempel gefertigt, eine übergroße LötKolbenspitze, die alle Pins des Sockels gleichzeitig erhitzt.

---

<sup>14</sup>Ball-Grid-Array: eine platzsparende Verbindungsart, bei der die Anschlüsse des elektronischen Bauteils nicht seitlich, sondern unter dem Bauteil liegen.



**Abbildung 5.2:** Nathan mit FPGA und Steckverbindern für die Tochterplatine Recha. Der zwischen den Steckverbindern liegende HAGEN-Sockel ist bereits entfernt.

## 5.2 Recha

Die von Andreas Grübl entworfene Recha-Platine [36] ist Träger für Spikey, welcher entweder direkt auf die Platine *gebondet*<sup>15</sup> oder, in einem Gehäuse untergebracht, verlötet werden kann. Ein DAC [40] und zwei Festspannungsregler [41] erzeugen zusätzliche Spannungen, wobei die Stromversorgung für den digitalen und den analogen Teil des Chips getrennt sind.

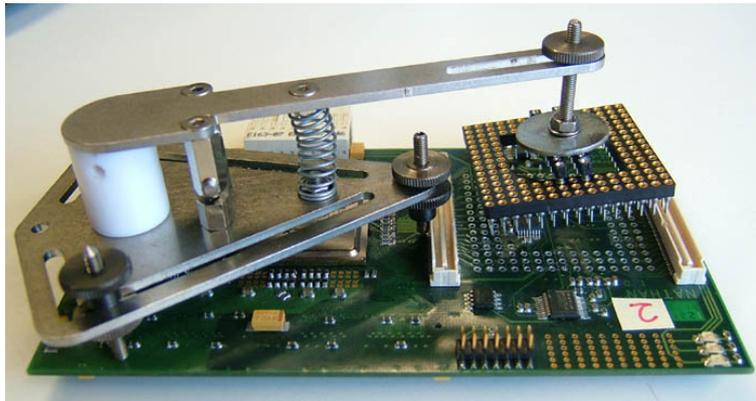
Neun Leitungen sind vom Chip heraus direkt an LEMO-Buchsen geführt. Acht sind mit den Neuronenschaltungen verbunden und können zur Betrachtung der Membranspannungen mit einem Oszillographen benutzt werden, an der neunten, der *ibtest*-Leitung, ist zur exemplarischen Untersuchung einer der Stromspeicher angeschlossen. Die internen Ausgangstreiber haben eine Impedanz von  $50\ \Omega$ , zur Vermeidung von Signalreflexionen werden entsprechende Terminierungswiderstände im Oszillographen vorausgesetzt.

Die Leitungen zu den LVDS-Verbindungen auf Spikey bestehen aus differentiellen Paaren mit einer Impedanz von  $100\ \Omega$ . Auf Nathan sind auf auf gleiche Weise zum FPGA fortgesetzt. Die Recha-Platine besteht aus vier Kupferlagen. Die beiden mittleren Lagen dienen der Spannungsversorgung, dies hat den Vorteil, dass die differentiellen Leitungen auf den Außenlagen ein konstantes Referenzpotenzial erfahren. Die Leitungen selbst sind meanderförmig verlegt, auf diese Weise können die Längen der Leitungen und damit die Signallaufzeiten aneinander angepasst werden.

Bestückt werden die Platinen mit dem hauseigenen, halbautomatischen Bestücker, gelötet wird im Lötöfen. Eine bestückte Recha-Platine mit direkt *gebondetem* Spikey ist in Abbildung 5.4 zu sehen.

Es stellt sich heraus, dass die Chip-interne Analogspannungserzeugung nicht wie gewünscht funktioniert. Um dies genauer untersuchen zu können, werden nachträglich einige Bauteile hinzugefügt und mit Fädeldraht angeschlossen. Zum einen sind dies Dioden, am unteren Rand in Abbildung 5.4 zu sehen, zum anderen ein *Analog-Digital-Converter* (ADC) [42], der „huckepack“ auf den DAC ge-

<sup>15</sup>Beim *Bonden* wird ein feiner Draht auf die Kontaktflächen (Pads) gepresst und durch Ultraschall mit diesen verschweißt



**Abbildung 5.3:** Nathan mit aufgesetztem Entlöt-Kran und ausgelötetem HAGEN-Sockel.

klebt und oben rechts in der Abbildung zu sehen ist. Die Dioden dienen zum Schutz des Chips vor elektrostatischer Entladung, zum Beispiel durch Kontakt mit der Hand. Sie sind so verschaltet, dass Überspannungen an die Versorgungsspannung, bzw. an die Masse des Chips abgeleitet werden. Auf diese Weise wird ein Spannungsdurchbruch und eine Zerstörung der Transistoren an den Ein- bzw. Ausgängen verhindert. Normalerweise sind alle *IO-Pads*<sup>16</sup> Chip-intern mit *ESD*<sup>17</sup>-Schutzdioden versehen. In diesem Fall sollen jedoch zwei der intern generierten Analogspannungen untersucht werden, die an ungeschützten *Probepads* anliegen. *Probepads* werden im Normalfall nur zu Testzwecken in einem *Wafer-Prober* mit einer Nadelkarte<sup>18</sup> angeschlossen, da die Spannungen jedoch im laufendem Betrieb gebraucht werden, kommt ein *Wafer-Prober* nicht in Frage. Die *Probepads* werden also *gebondet*, da sie jedoch viel kleiner als die Standard IO-Pads sind, gestaltet sich dies als schwierig und gelingt nicht immer. Abbildung 5.5 zeigt ein *gebondetes Probepad*, das vollkommen vom *Bonddraht* verdeckt ist. Auch in Abbildung 5.4 Teil B sind die *Bondverbindungen* zu den *Probepads* zu erkennen sowie die Drähte, die zu den externen ESD-Schutzdioden geführt sind.

Die *Probepads* sind nur auf zwei *Rechas* angeschlossen, während alle Platinen über einen ADC verfügen. An zwei der vier Eingänge sind die Neuronschaltungen angeschlossen, die auch mit den LEMO-Buchsen verbunden sind. Man kann damit also die Membranspannungen messen. Bei den Platinen mit angeschlossenen *Probepads* sind diese mit den anderen beiden Eingängen des ADCs verbunden.

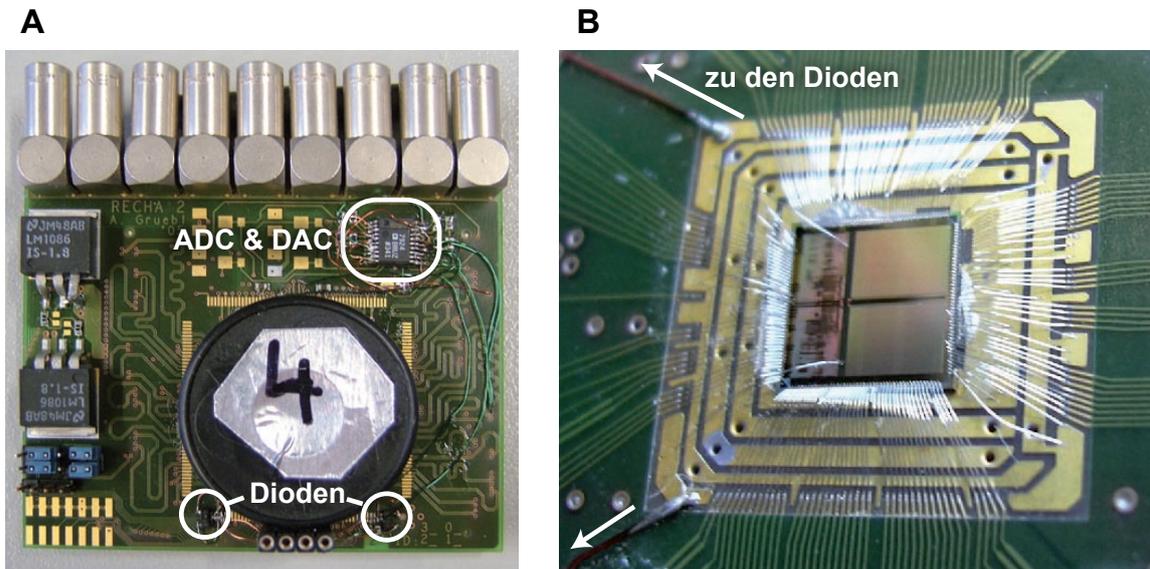
### 5.3 Recha V2

Im Zuge der Umrüstung auf die neue Version Spikey 2 wird auch die Recha-Platine überarbeitet, die neue Version heißt Recha V2.

<sup>16</sup>IO-Pads: Input & Output-Pads

<sup>17</sup>ESD: Electrostatic Discharge

<sup>18</sup>Mit Hilfe des *Wafer-Probers* werden können Chips auf unzerteilten *Wafers* untersucht werden, über dünne Nadeln wird der Kontakt mit dem Chip hergestellt. Um Signale untersuchen zu können, die nicht an den für den normalen Betrieb des Chips nötigen Ausgängen anliegen, werden *Probepads* verbaut. Aus Platzgründen sind diese meistens kleiner, sind nicht über alle Metalllagen mit Vias gestützt und haben keine ESD-Dioden.



**Abbildung 5.4:** Recha-Platine, mit direkt *gebondetem* Spikey-Chip unter einem Schutzdeckel (A) bzw. ohne Deckel (B).

Der ADC ist nun fest vorgesehen und bekommt seinen eigenen Platz. Er erhält außerdem eine eigene Referenzspannungsquelle für 1,8 Volt [43]. Bisher wurde die des DACs mitbenutzt, welche nur 1,25 Volt liefert und den Messbereich des ADCs begrenzt.

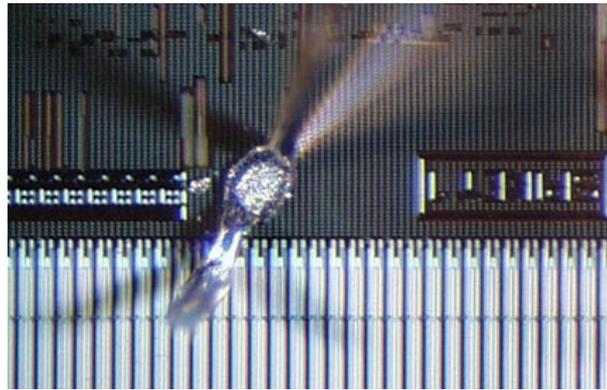
Die Membranspannungen aller Neuronen lassen sich über die acht LEMO-Buchsen mit einem Oszillographen untersuchen, jeweils 48 über eine Buchse. Für das FACETS Project wurden Oszillographen gekauft, die über das Netzwerk angesteuert und ausgewertet werden können. Dies ermöglicht es anderen Arbeitsgruppen über das Internet Experimente durchzuführen. Allerdings ist es unzuweckmäßig, dass ein Spikey allein acht Oszillograph-Kanäle benötigt, um alle Membranspannungen betrachten zu können. Aus diesem Grund verfügt Recha V2 über einen Multiplexer, der einzelne Neuronen auf die neunte LEMO-Buchse schalten kann.

An dieselbe Buchse ist auch ein hinzugekommener OP-AMP [44] angeschlossen. Er wird als Ausgangstreiber für die kombinierte *ibtest/Analog-Readout*-Leitung benötigt. *Analog-Readout* ist eine neue Funktion von Spikey 2, man kann nun alle intern erzeugten Analogspannungen über diese Leitung messen oder alternativ wie bei Spikey 1 den exemplarischen Stromspeicher *ibtest*. Diese *ibtest/Analog-Readout*-Leitung ist auch an einen Eingang des ADCs geführt, die Spannungswerte können somit von der Software ausgelesen werden.

Die *Probepads* sind bei Spikey 2 nicht mehr vorhanden, da sie nun überflüssig sind.

Um einen verwendbaren Multiplexer zu finden werden zunächst die Anforderungen an die Bandbreite und die *Slew Rate*<sup>19</sup> berechnet. Am stärksten ändert sich die Membranspannung, wenn ein *Spike* ausgelöst wird. Dies wird in Abbildung 5.6 gezeigt. Die Membranspannung wird an einer der acht LEMO-Buchsen gemessen, die direkt mit den Neuronschaltungen verbunden sind. Gleich-

<sup>19</sup>Slew Rate: Die maximale Anstiegs- oder Abfallgeschwindigkeit der Ausgangsspannung einer Verstärker- oder Treiberschaltung



**Abbildung 5.5:** Ein *Probepad* auf Spikey 1, vollkommen verdeckt vom *Bonddraht*.

zeitig zeigt die Abbildung auch die gemessene Membranspannung bei Verwendung des schließlich ausgewählten Multiplexers, diese Kurve ist jedoch noch von Interesse, sie wird später diskutiert. Der Verlauf der Membranspannung entsteht durch vier gleichzeitig feuernde externe Eingänge, die über exzitatorische Synapsen auf das betrachtete Neuron koppeln. Da dessen Schwellspannung sehr niedrig gewählt ist, wird hierdurch ein *Spike* ausgelöst, das Membranpotenzial fällt rapide auf den Wert von  $V_{\text{reset}}$  ab. Anhand dieses Abfalls kann man die Anforderungen an die Bandbreite und die *Slew Rate* berechnen. Für die *Slew Rate*  $SR$  misst man im steilsten Stück des Graphen  $SR \approx 320 \text{ V}/\mu\text{s}$ . Um die maximale Frequenz des Signals zu ermitteln wird folgende Faustformel benutzt [45], die darauf beruht, dass die Zeitspanne für den Abfall der Amplitude von 90% auf 10% ungefähr einem Drittel der Periode einer Sinusfunktion entspricht:

$$\nu \approx \frac{0.35}{\tau_{90\%-10\%}} \quad (5.1)$$

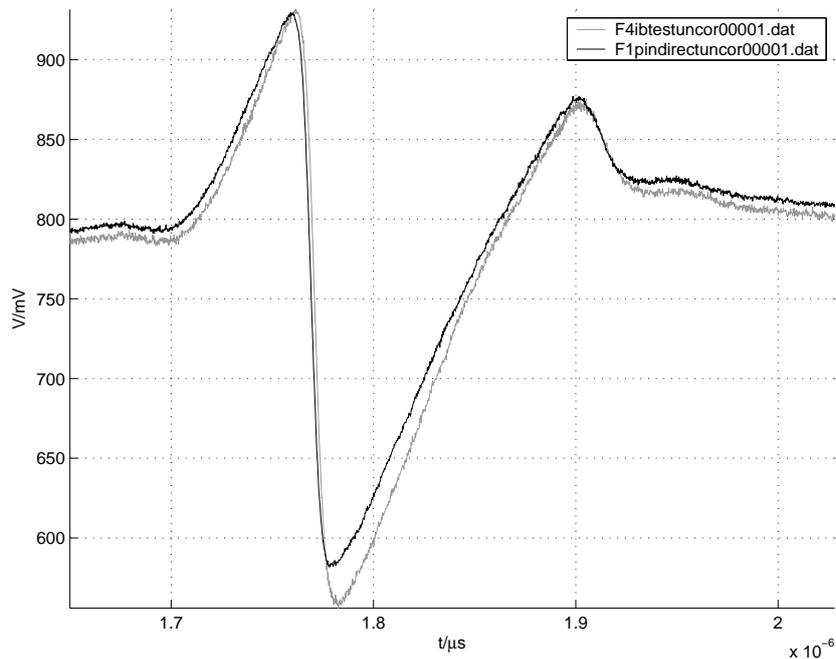
Mit  $\tau_{90\%-10\%} \approx 10 \text{ ns}$  ergibt sich eine Bandbreite von mindestens 35 MHz.

Es lassen sich nicht viele Multiplexer finden, die diese Anforderungen erfüllen. Die Spannungsversorgung auf Recha schränkt die Suche weiter ein, es stehen nur positive Spannungen zur Verfügung, es bleiben also nur *single-supply* Ausführungen. Allerdings sind die 3,3 Volt vom Recha zu wenig für die noch verbliebene Auswahl an Multiplexern, 5 Volt werden benötigt. Deshalb ist eine zusätzliche Spannungsversorgung erforderlich. Das ATX-Netzteil, welches die Backplane mit Strom versorgt, stellt zwar auch 5 Volt zu Verfügung, allerdings von begrenzter Stabilität. Sicherer ist es, einen eigenen Spannungsregler [46] zu verwenden, und diesen mit 12 Volt vom Netzteil zu betreiben.

Weiterhin muss der Multiplexer in der Lage sein, eine  $50 \Omega$ -Leitung zu treiben, da der Ausgang über eine LEMO-Buchse und ein Koaxialkabel mit dem Oszillographen verbunden werden soll.

Der verwendete Multiplexer MAX4312 von Maxim ist eigentlich für Video-Anwendungen gedacht. Nur in diesem Anwendungsbereich findet man Ausführungen mit genügend hoher Bandbreite und *Slew Rate*, die ohmsche Lasten treiben können. Laut Datenblatt [47] hat er eine Bandbreite von 265 MHz und eine *Slew Rate* von  $345 \text{ V}/\mu\text{s}$ .

Wie in Abbildung 5.6 zu sehen ist, weicht der direkt gemessene Wert vom hinter dem Multiplexer gemessenen Wert ab. Durch eine lineare Korrektur lassen sich die beiden Kurven jedoch bis auf einen Zeitversatz in Deckung bringen, dargestellt in Abbildung 5.7. Die gemessenen Unterschiede lassen sich durch Abweichungen der  $50 \Omega$  Ausgangswiderstände erklären. In den Spannungsteilern, siehe

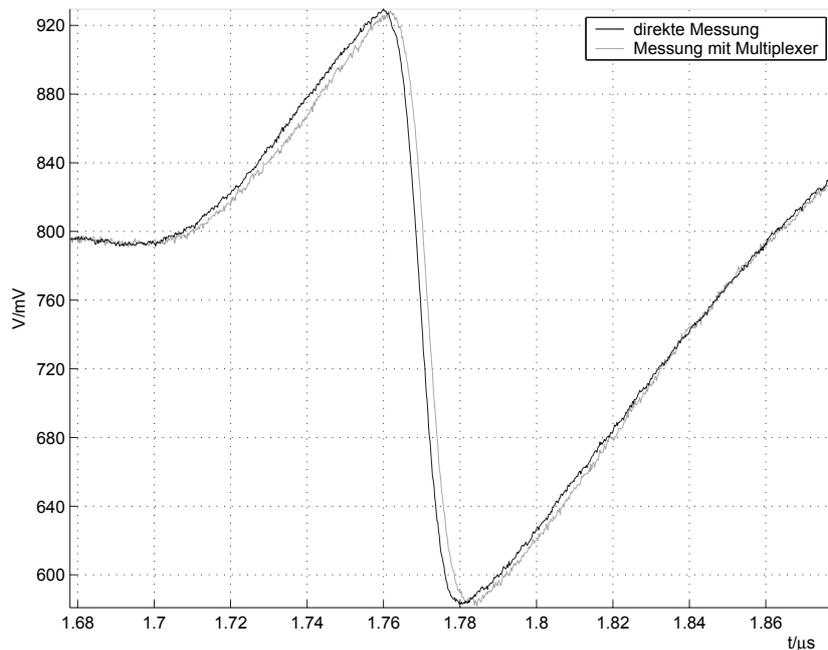


**Abbildung 5.6:** Ein Membranpotenzial überschreitet die Schwellspannung und löst einen *Spike* aus. Dargestellt sind die direkte Messung und die Messung nach Durchleitung durch den Multiplexer. Der Spannungsteiler, bestehend aus Ausgangs- und Terminierungswiderstand, halbiert die Amplitude des Signals der direkten Messung. Am Eingang des Multiplexers liegt nur noch die halbierte Amplitude an, hinter dem Multiplexer wird diese wiederum vom Spannungsteiler halbiert. Die dargestellten Messwerte sind aus diesem Grund verdoppelt, bzw. vervierfacht worden.

die Erläuterung zu Abbildung 5.6, wird die Spannung nun nicht mehr genau halbiert, dies führt zu linearen Abweichungen. In zukünftigen Messungen müssen diese linearen Abweichungen und der Zeitversatz von 2 ns berücksichtigt werden.

Die Spannungen an den digitalen Eingängen des Multiplexers müssen an die Versorgungsspannung von 5 Volt angepasst werden. Ein NMOS-Transistor und ein *pull-up* Widerstand dienen als Pegelwandler, wobei das Signal vom FPGA invertiert wird. Dies wird ausgenutzt, um eine Leitung zum FPGA zu sparen. Der Multiplexer und der OP-AMP, der die *Analog-Readout*-Leitung treibt, teilen sich das *enable*-Signal, jedoch wird es vor dem Multiplexer einmal invertiert. So ist immer nur eine Baugruppe eingeschaltet, es wird verhindert, dass die aktiven Ausgänge gegeneinander treiben. Abbildung 5.8 zeigt die verwendete Schaltung.

Um dem gewachsenen Platzanspruch auf der Platine gerecht zu werden, werden die alten Spannungsregler durch kleinere Bauteile [48] ersetzt und weiter zusammengeschoben. Multiplexer und ADC finden auf einer Fläche Platz, die vorher vier optionalen Potentiometern zur Verfügung stand. Diese wurden allerdings nicht genutzt, da die Alternative, der DAC, bequemer und vielseitiger einsetzbar ist.



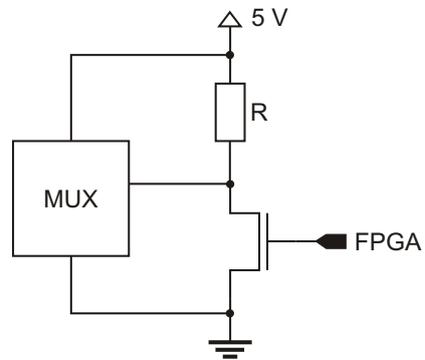
**Abbildung 5.7:** Das gleiche Membranpotenzial wie in Abbildung 5.6, jedoch mit mit linearen Korrekturen in den hinter dem Multiplexer gewonnenen Messwerten: Es wurde ein Offset von 65 mV addiert, die Messwerte wurden nicht mit dem Faktor 4, wie in Abbildung 5.6, sondern mit 3,7 multipliziert.

Auch der *Bondbereich* wird überarbeitet. Es wird ein *Footprint*<sup>20</sup> für Spikey erstellt. Nun stimmen Schaltplan und Layout überein. Vorher endeten die Leitungen im Layout zwar an der richtigen Stelle, hatten aber keinen echten Abschluss, was zu vielen Fehlermeldungen führte. Der *Footprint* erleichtert das Erstellen von neuen Platinen, wie zum Beispiel dem geplanten *Multi-Spikey-Board*<sup>21</sup>. In Absprache mit der Produktionsfirma Würth Elektronik [49] wird die Breite der *Bondpads* vergrößert, der Abstand im Gegenzug verringert, die Pads lassen sich nun besser *Bonden*.

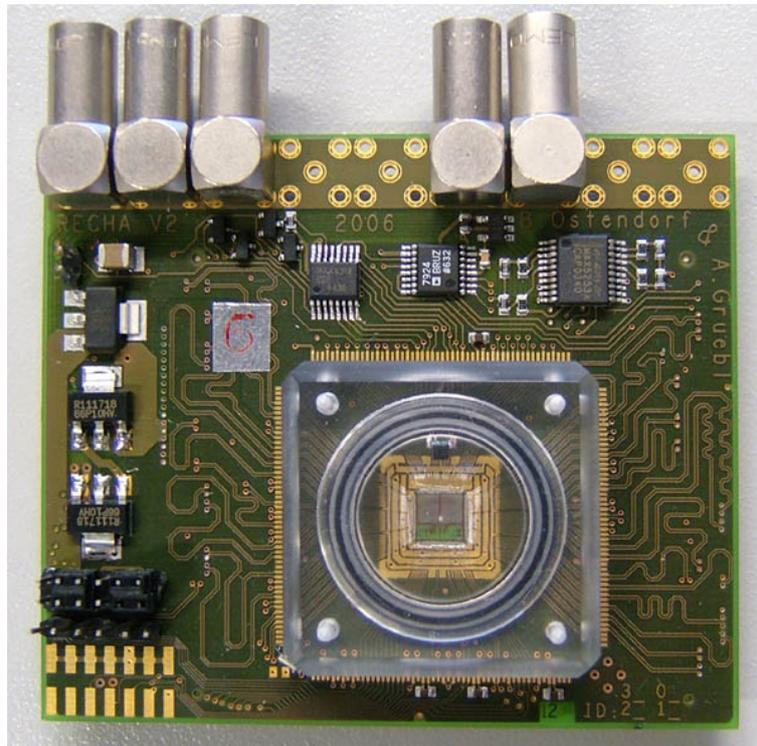
Da in Zukunft alle Chips direkt *gebondet* werden sollen, wird ein Schutzdeckel entworfen. Bisher wurden abgeschnittene Filmdosen als Deckel verwendet, dies ist aber unschön und unsauber. In der hauseigenen Feinmechanikwerkstatt wird ein Deckel aus Plexiglas gefertigt. Recha V2 verfügt über vier Bohrungen, an denen dieser festgeschraubt werden kann, ein Dichtungsring schließt den Chip luftdicht ab. Auf Recha V2 sind weiterhin Anschlüsse für LEDs vorhanden, mit denen Deckel und Chip bei Präsentationen beleuchtet werden können. Die neue Version der Recha-Platine ist in Abbildung 5.9 gezeigt.

<sup>20</sup>footprint: der „Fussabdruck“ eines Bauteils, definiert die Anschlussflächen auf der Platine, auf denen die Anschlüsse des Bauteils verlötet werden

<sup>21</sup>Multi-Spikey-Board: wie Recha eine Aufsteckplatine, jedoch mit mehreren Spikeys



**Abbildung 5.8:** Pegelwandler an den digitalen Eingängen des Multiplexers (MUX)



**Abbildung 5.9:** Recha V2-Platine mit direkt gebonetem Spiky unter Plaxiglasdeckel.



# Kapitel 6

## Messungen

Um Tests mit dem Spikey-Chip durchzuführen müssen die Testprogramme auf dem PC mit Spikey kommunizieren können. Hardwareseitig findet diese Kommunikation über eine PCI-Steckkarte, ein SCSI-Kabel und die Backplane statt. Die verwendete Software ist in der objektorientierten Sprache C++ geschrieben und besteht aus mehreren Abstraktionsschichten. Die unterste Schicht, genannt *Slow-Control* [50, 51], implementiert ein *Token-Ring* [52] ähnliches Netzwerk und ist dafür zuständig Daten zwischen den FPGAs auf Nathan und dem PC zu transportieren. Darauf aufbauend ermöglicht die nächste Schicht der Software Zugriffe auf den FPGA und den Speicher auf Nathan. Über Prozeduren im FPGA ist es somit auch möglich mit Spikey-Funktionen zu kommunizieren. Auf diese Weise lassen sich die Grundfunktionen von Spikey testen und analysieren. Es ist dabei möglich direkt auf den Chip zuzugreifen (Zugriff über *Slow-Control*), oder die Anweisungen gebündelt in das DDR-RAM auf Nathan zu kopieren sie dann von dort nach und nach an den Spikey-Chip zu schicken (Zugriff über *Playback Memory*). Weitere Schichten fügen weitere Funktionalität hinzu, die es ermöglichen, Konfigurationen in den Chip zu laden oder *Spikes* zu senden und zu empfangen.

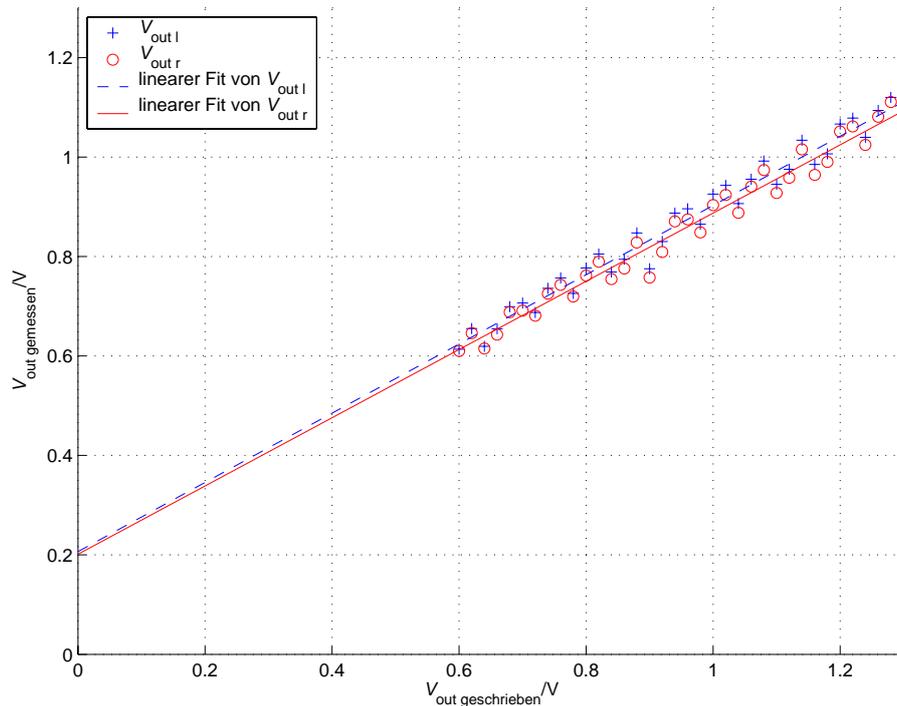
Die zugrunde liegende Software wurde von Johannes Schemmel und Andreas Grübl entwickelt, die zu den Messungen verwendeten Routinen bauen darauf auf.

### 6.1 Analogspannungsquellen Spikey 1

#### 6.1.1 Messungen

Um das in Abschnitt 3.2.2 beschriebene Neuronmodell korrekt nachbilden zu können, werden im Spikey-Chip diverse analoge Spannungen generiert. So lassen sich zum Beispiel die verschiedenen Umkehrpotenziale oder die Schwellspannung einstellen. Einige Parameter gelten global für einen der beiden Synapsenblöcke, zum Teil gibt es aber auch feinere Unterteilungen. Pro Synapsenblock ist jeweils eine Analogspannungsquelle, im folgenden mit *Vout* bezeichnet, mit den *Probepads* verbunden. Bei zwei Spikey-Chips sind die *Probepads* über *Bonddrähte* angeschlossen und mit dem ADC auf Recha verbunden, die Messwerte können von der Software gelesen und weiterverarbeitet werden. In Abbildung 6.1 werden die an den *Probepads* gemessenen Ausgangsspannungen mit den Vorgabewerten verglichen. Die Messwerte des anderen vermessbaren Spikeys sehen ähnlich aus.

Vergleicht man die geschriebenen mit den gelesenen Spannungswerten, so fallen zwei verschiedene Fehler auf. Zum einen weichen die Messwerte mitunter stark von der Ausgleichsgeraden ab, außerdem geht die Ausgleichsgerade nicht durch den Nullpunkt, und hat eine Steigung ungleich von



**Abbildung 6.1:** Analogspannungsquellen: Vorgabe- und Messwerte. Die Indizes 'l' und 'r' stehen für den linken bzw. rechten Synapsenblock.

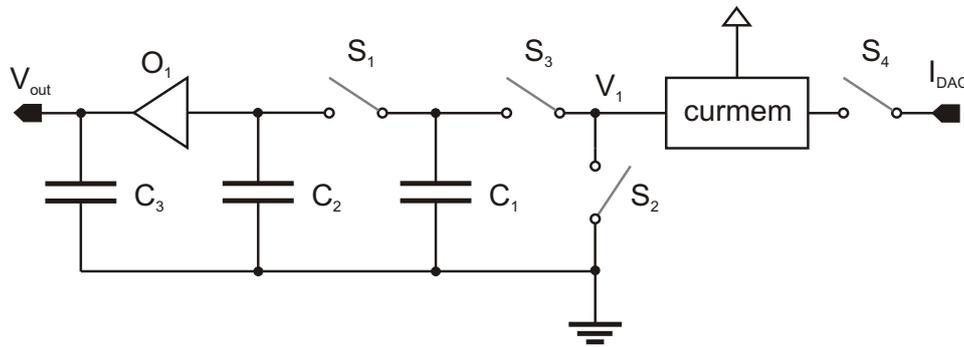
eins. 600 mV ist die untere Grenze, an der die  $V_{out}$ -Schaltungen noch gut funktionieren, nach oben hin ist der Messbereich durch die Referenzspannung von 1,25 V des ADCs begrenzt. Im folgenden Abschnitt werden die Ursachen der Fehler diskutiert.

### 6.1.2 Schaltungsaufbau der Analogspannungsquellen

Der erste Fehler wird durch Übersprechen hervorgerufen. In Abbildung 6.2 ist die  $V_{out}$ -Schaltung gezeigt, deren Aufgabe es ist, aus einem Strom eine Spannung zu generieren. Abhängig von der zu schreibenden Spannung  $V_{out}$  erzeugt ein zentraler DAC den Strom  $I_{DAC}$ . Dieser wird in einem Stromspeicher, in Abbildung 6.2 mit *curmem*<sup>22</sup> bezeichnet, des entsprechenden  $V_{outs}$  gespeichert. Insgesamt gibt ca. 3000 dieser Stromspeicher, die die Bias-Ströme für die Synapsentreiber und die Neuronschaltungen speichern. Die Reihenfolge mit der die Stromspeicher programmiert werden ist einstellbar. Am günstigsten ist es, diese nach der Größe der zu schreibenden Ströme zu sortieren, so muss der DAC keine großen Sprünge machen und die Zeitspanne, nach der der gewünschte Wert am Ausgang anliegt, ist geringer.

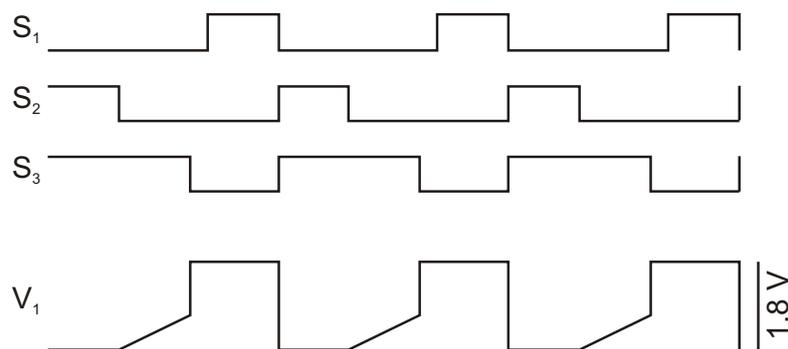
Der DAC wird für die 20 Analogspannungen mitbenutzt, da er aber Ströme generiert, müssen diese in den  $V_{out}$ -Zellen in Spannungen umgewandelt werden. Diese Umwandlung findet in einem

<sup>22</sup>curmem: current memory



**Abbildung 6.2:** Vereinfachtes Schaltbild der Analogspannungsquellen

sich ständig wiederholenden Zyklus statt. Zunächst wird  $C_1$  komplett entladen, dann auf die gewünschte Spannung aufgeladen und schließlich wird diese Spannung auf  $C_2$  übertragen.  $O_1$  dient als Ausgangsverstärker,  $C_3$  als Glättungskondensator, die Spannung über  $C_2$  entspricht also  $V_{\text{out}}$ . Bei konstantem  $I_{\text{DAC}}$  verändert sich die Spannung an  $C_1$  ständig. Die an  $C_2$  dagegen ändert sich nicht. Wird ein anderer Strom angelegt, strebt sie so langsam gegen den neuen Wert, dass keine ungewollten Effekte durch sprunghafte Änderung von  $V_{\text{out}}$  auftreten. Der genaue Ladezyklus ist Abbildung 6.3 zu entnehmen. Die Schalter  $S_2$  und  $S_3$  sind zunächst geschlossen, der Kondensator  $C_1$  ist vollständig entladen. Nach dem Öffnen von  $S_2$  fließt der Strom nicht mehr nach Masse, sondern auf  $C_1$ . Der Strom wird so gewählt, dass am Ende des Ladevorgangs die gewünschte Spannung  $V_{\text{out}}$  am Kondensator anliegt. Durch Schließen von  $S_1$  gleichen sich die Spannungen über  $C_1$  und  $C_2$  an, nach einigen Zyklen liegt auch an  $C_2$   $V_{\text{out}}$  an. Als Schalter werden die in Abschnitt 2.3 vorgestellten kompensierten *Transmission-Gates* verwendet, um *Charge Injection* zu vermeiden.



**Abbildung 6.3:** Zyklus der  $V_{\text{out}}$ -Schaltung: Spannung am Knoten  $V_1$  sowie Schaltzeiten, ein hoher Pegel symbolisiert einen geschlossenen Schalter

Der Stromspeicher ist eine Stromquelle, d.h. solange die Ausgangsspannung kleiner als die Versorgungsspannung ist, liefert er einen konstanten Strom, für den gilt  $I = Q/t$ .  $Q$  ist die absolute Ladungsmenge, die zum Zeitpunkt  $t$  aus dem Stromspeicher geflossen ist. Mit der Formel für die Spannung über einem Kondensator  $V = Q/C$  lässt sich dann die Spannung am Knoten  $V_1$  während des Ladevorgangs berechnen:

$$V_1 = \frac{I_{\text{DAC}} \cdot t}{C} \quad (6.1)$$

Sie folgt dem in Abbildung 6.3 dargestellten Verlauf. Zunächst liegt sie auf dem Massepotenzial, dann steigt sie gemäß Gleichung 6.1 linear auf  $V_{\text{out}}$  an und springt schließlich beim Öffnen von  $S_3$  auf die Versorgungsspannung von 1,8 V, da der Stromspeicher nun keine Last mehr hat. Am Anfang des nächsten Zyklus springt sie wieder auf Masse zurück.

Diese Sprünge machen sich als Übersprechen bemerkbar. Sie koppeln auf den Stromspeicher *curmem*, welcher in Folge dessen nicht den gewünschten Strom  $I_{\text{DAC}}$  speichert, sondern einen geringfügig anderen. Der *Refresh*-Zyklus des Stromspeichers und der Zyklus der *Vout*-Schaltung haben eine zufällige Phasenlage, abhängig von dieser Lage macht sich das Übersprechen unterschiedlich stark bemerkbar. Dies ist ein Grund für die zufälligen Abweichungen der  $V_{\text{out}}$  Werte, der sich allerdings korrigieren lässt. Nicht korrigieren lässt sich dagegen, dass auch die Adressleitungen der *Vout*-Schaltungen auf die Stromspeicher koppeln.

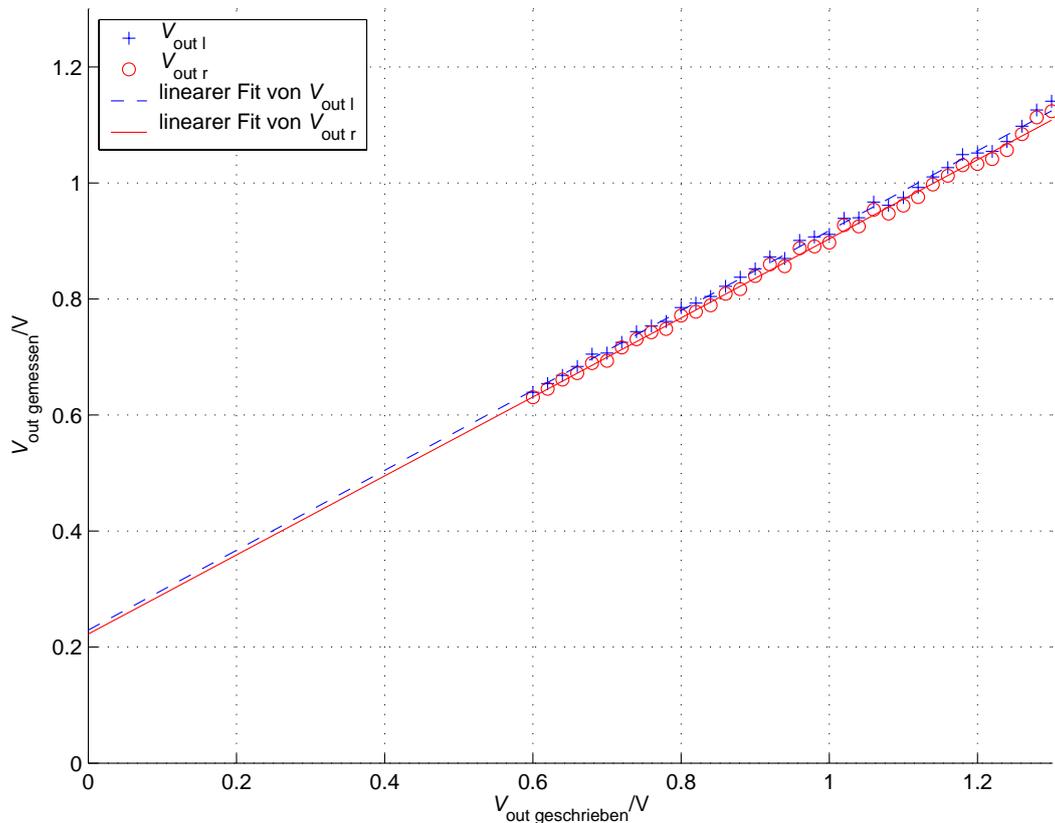
### 6.1.3 Erste Kalibrierung: Beseitigung der scheinbar zufälligen Fehler

Schreibt man denselben Strom oft hintereinander in eine *Vout*-Zelle, so stellt man fest, dass die generierten Ausgangsspannungen nicht komplett zufällig auftreten, sondern dass es eine Verteilung mit einem Maximum gibt. Man kann die zufälligen Abweichungen unterdrücken, indem man nicht einmal, sondern wiederholt den gleichen Wert schreibt. Die Verteilung der Ausgangsspannungen wird aufgenommen und das Maximum ermittelt. Anschließend wird wiederum solange der gleiche Wert geschrieben, bis die gemessene Spannung im Maximum der Verteilung liegt. Es ist zwar nicht garantiert, dass die so generierte Spannung genau der gewünschten entspricht, aber sie ist reproduzierbar. Eine Kalibrierungsroutine, die diese Methode benutzt wurde von Johannes Schemmel geschrieben.

In Abbildung 6.4 werden wiederum die Messwerte mit den geschriebenen Werte verglichen, diesmal jedoch nach Anwendung der Kalibrierungsroutine. Man erkennt, dass die Abweichungen von der Ausgleichsgeraden durch Anwenden der Kalibrierungsroutine deutlich schwächer ausfallen. Damit die Kalibrierung auf möglichst allen Recha-Platinen zur Verfügung steht, wird zur Messung die Membranspannung zweier spezifischer Neuronen herangezogen, die sich diese auf allen Rechas messen lassen.

### 6.1.4 Zweite Kalibrierung: Korrektur der Steigung

Im Folgenden soll der Spikey-Chip nun so kalibriert werden, dass die Ausgleichsgerade eine Steigung von eins hat und durch den Nullpunkt geht, dass also die gemessenen Werte mit den geschriebenen übereinstimmen. Dies ist nur bei den beiden Spikeys mit angeschlossenen *Probepads* möglich, da



**Abbildung 6.4:** Analogspannungsquellen: Ausgangsspannungen bei Anwendung einer Kalibrierungsroutine zur Verminderung der zufälligen Abweichungen

dort die Ausgänge der *Vout*-Schaltungen verfügbar sind. Die Membranspannungen enthalten zu viele Konstituenten, als dass sich daraus genaue Parameter bestimmen ließen. Die Hoffnung ist, dass sich die einzelnen Schaltungen innerhalb eines Blockes gleich verhalten, und die Kalibrierung der *Vouts* an den *Probepads* auch für die übrigen Zellen gültig ist. Überprüfen lässt sich dies jedoch nicht.

Ziel ist es, die bestehende Software möglichst unverändert zu lassen, d.h. dass sie sowohl mit kalibrierten als auch unkalibrierten Spikeys funktioniert. Sie wird um zwei Komponenten erweitert. Zum einen um die *SpikeyVoutCalib*-Klasse. Diese verfügt über Routinen, mit denen die Kalibrierungswerte aus einer XML-Datei<sup>23</sup> gelesen und dorthin geschrieben werden können, als auch über eine *AutoCalib*-Routine, die die Kalibrierungswerte automatisch bestimmt.

Zum anderen wird die *SpikeyCalibratable*-Klasse hinzugefügt. Diese ist abgeleitet von der Klasse *Spikey*. Die *Spikey*-Klasse ist die oberste Hardware-Abstraktionsschicht, baut auf niedrigeren Abstraktionsklassen auf und stellt die zum Betrieb des Chips nötigen Grundfunktionen bereit. In *SpikeyCalibratable* wird die Funktion überladen, die die gewünschten Spannungswerte in den Strom umrechnet, der schließlich im Chip vom DAC erzeugt und in den *curmem*-Zellen gespeichert wird. Die neue Umrechnungsfunktion benutzt die von *SpikeyVoutCalib* bereitgestellten Kalibrierungswerte. Die Umrechnung erfolgt transparent für den Benutzer, er muss nur die Spikey-Nummer angeben. Die Software benutzt dann die zugehörigen Werte aus der xml-Datei. Die Kalibrierungsparameter

<sup>23</sup> xml: Extensible Markup Language; Eine Sprache, in der Daten in einer Baumstruktur gespeichert werden.

bestehen pro Synapsenblock aus einem Faktor, der die Steigung korrigiert, und einem Offset. In der *AutoCalib*-Routine werden beide Werte solange angenähert, bis die Änderung zum vorhergehenden Iterationsschritt einen vorgegebenen Grenzwert unterschreitet.

Die Umrechnungsfunktionen in der *Spikey*-Klasse:

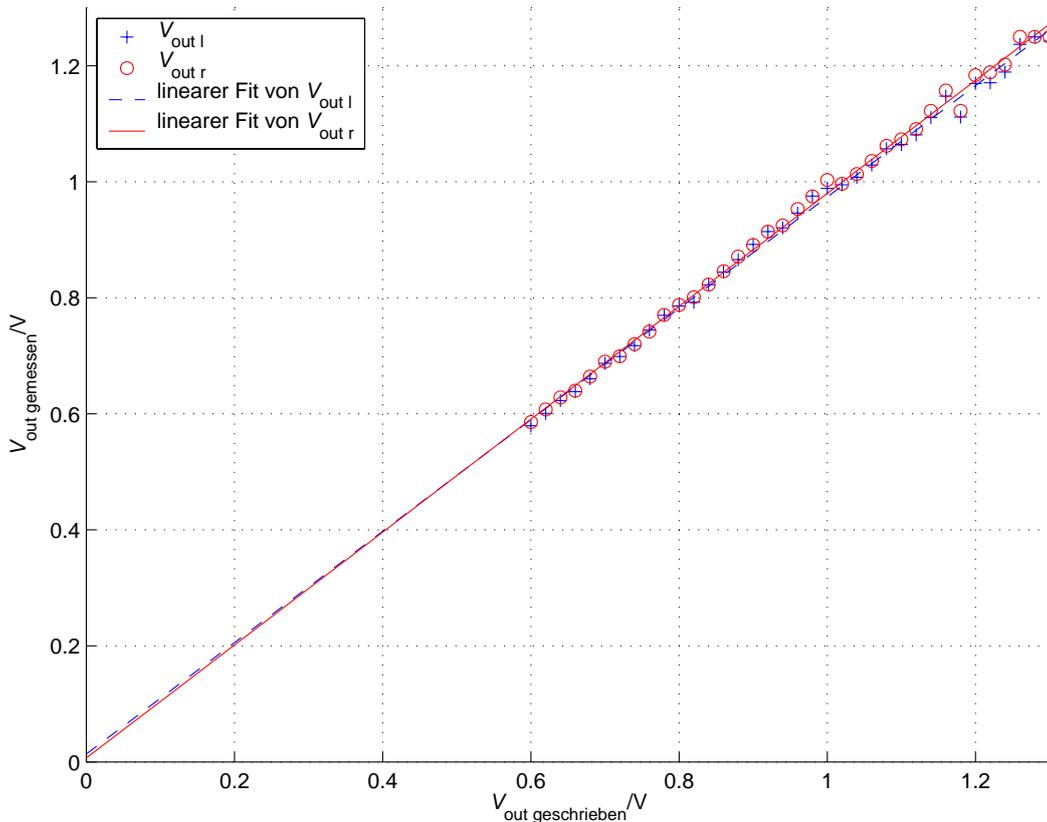
$$I_{\text{DAC}} = \frac{V_{\text{out}} \cdot c}{n_{\text{periods}} \cdot T_{\text{clock}}} \quad (6.2)$$

sowie in der *SpikeyCalibratable*-Klasse:

$$I_{\text{DAC}} = \frac{\left( \frac{V_{\text{out}}}{K_{\text{slope}}} - K_{\text{off}} \right) \cdot c}{n_{\text{periods}} \cdot T_{\text{clock}}} \quad (6.3)$$

In der Konstante  $c$  sind die Kapazität des Kondensators und ein Umrechnungsfaktor auf den Maximalstrom, den der DAC liefern kann, zusammengefasst.  $T_{\text{clock}}$  ist die Periode des Taktes, mit dem der Chip betrieben wird und  $n_{\text{periods}}$  entspricht der Anzahl der Takte, in denen der Kondensator geladen wird. Der Korrekturfaktor für die Steigung ist  $K_{\text{slope}}$  für den Offset  $K_{\text{off}}$ .

Abbildung 6.5 zeigt das Verhalten der  $V_{\text{out}}$ -Schaltungen nach vollständiger Kalibrierung. Man erkennt, dass die gemessenen Werte nun gut mit den gewünschten übereinstimmen.



**Abbildung 6.5:** Ausgänge der Analogspannungsquellen: Spannungswerte nach vollständiger Kalibrierung

## 6.2 Analogspannungsquellen Spikey 2

### 6.2.1 Änderungen der Schaltung

Da der Stromspeicher *curmem* bei der ersten Version des Spikey-Chips sehr empfindlich gegenüber Kopplungseffekten ist, kommt er bei Spikey 2 an dieser Stelle nicht mehr zur Verwendung. Anstatt einen Strom eine feste Zeitspanne lang auf einen Kondensator fließen zu lassen, wird nun der stromabhängige Spannungsabfall an einem Widerstand zur Spannungsgenerierung genutzt. Die neue Schaltung ist in Abbildung 6.6 dargestellt.

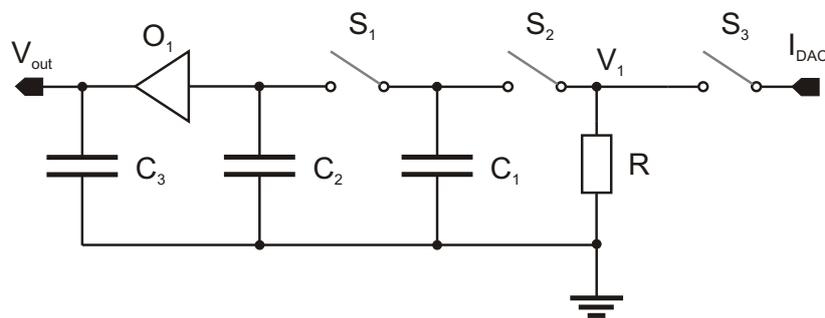


Abbildung 6.6: Vereinfachtes Schaltbild der überarbeiteten Analogspannungsquellen in Spikey 2

Der zentrale DAC, der die Ströme in die verschiedenen *curmem*-Speicher schreibt, wird auch benutzt, um  $I_{DAC}$  für die einzelnen  $V_{out}$ -Schaltungen zu generieren. Ist eine  $V_{out}$ -Schaltung an der Reihe, werden dort gleichzeitig die Schalter  $S_2$  und  $S_3$  geschlossen. Gemäß dem ohmschen Gesetz bildet sich eine Spannung über dem Widerstand  $R$ , welche auf dem Kondensator  $C_1$  gespeichert wird. Damit die Ausgangsspannung  $V_{out}$  bei einer Änderung des gewünschten Wertes nicht springt, sondern langsam gegen den neuen Wert strebt, sind zusätzlich noch  $C_2$  und  $S_1$  vorhanden. Wenn  $C_1$  auf die gewünschte Spannung geladen ist, werden  $S_2$  und  $S_3$  geöffnet und  $S_1$  geschlossen, die Spannungen auf  $C_1$  und  $C_2$  gleichen sich an.  $O_1$  dient weiterhin als Buffer,  $C_3$  als Glättungskondensator.

Der DAC liefert im Vergleich zu Spikey 1 einen zehnfach höheren Strom. Diese Änderung kommt der Genauigkeit der Spannung zu Gute. So kann der Widerstand  $R$  relativ klein dimensioniert werden, und damit verbunden auch dessen Abweichungen, die in Halbleiter-Prozessen immer auftreten.

### 6.2.2 Auswirkungen der Änderungen

Die Änderungen an der Schaltung und der Wegfall des *curmem* Stromspeichers führen allerdings auch zu ungewolltem Verhalten. Der Stromspeicher in Spikey 1 ermöglicht, dass  $I_{DAC}$  ständig zur Verfügung steht und dass ein interner *Refresh*-Zyklus die Spannung an  $C_2$  konstant hält. Ohne den Stromspeicher fällt der interne Zyklus weg, es gibt nur noch den langsameren Zyklus, in welchem der DAC an die einzelnen  $V_{out}$ -Zellen geschaltet wird. Des Weiteren wird der Knoten  $V_1$  über  $R$  auf Masse gezogen, wenn die Zelle nicht programmiert wird. Dies verursacht im Vergleich zu Spikey 1

viel stärker Leckströme, die die Kondensatoren entladen. Verstärkt wird dieser Effekt noch durch den langsameren *Refresh*-Zyklus.

Die Leckströme fließen zum größten Teil durch den Kanal des NMOS-Transistors des *Transmission-Gates*, welches Schalter  $S_2$  implementiert. Beide Transistoren befinden sich im *Subthreshold*-Bereich, die Ströme sind exponentiell abhängig von der *Gate-Source*-Spannung  $V_{GS}$ , vergleiche Formel 2.5. Da  $V_{out}$ , die *Source*-Spannung des PMOS-Transistors, immer kleiner ist als dessen *Gate*-Spannung  $V_{DD}$ , ist  $V_{GS}$  negativ und der Strom verschwindend gering. Beim NMOS dagegen liegen sowohl  $V_1$ , die *Source*-Spannung, als auch das *Gate* auf Masse. Es fließt also ein nicht vernachlässigbarer Strom durch den NMOS über  $R$  nach Masse. Durch Übersprechen von benachbarten Leitungen auf das *Gate* des NMOS-Transistors kann dessen  $V_{GS}$  sogar positiv werden, der Leckstrom wird stärker.

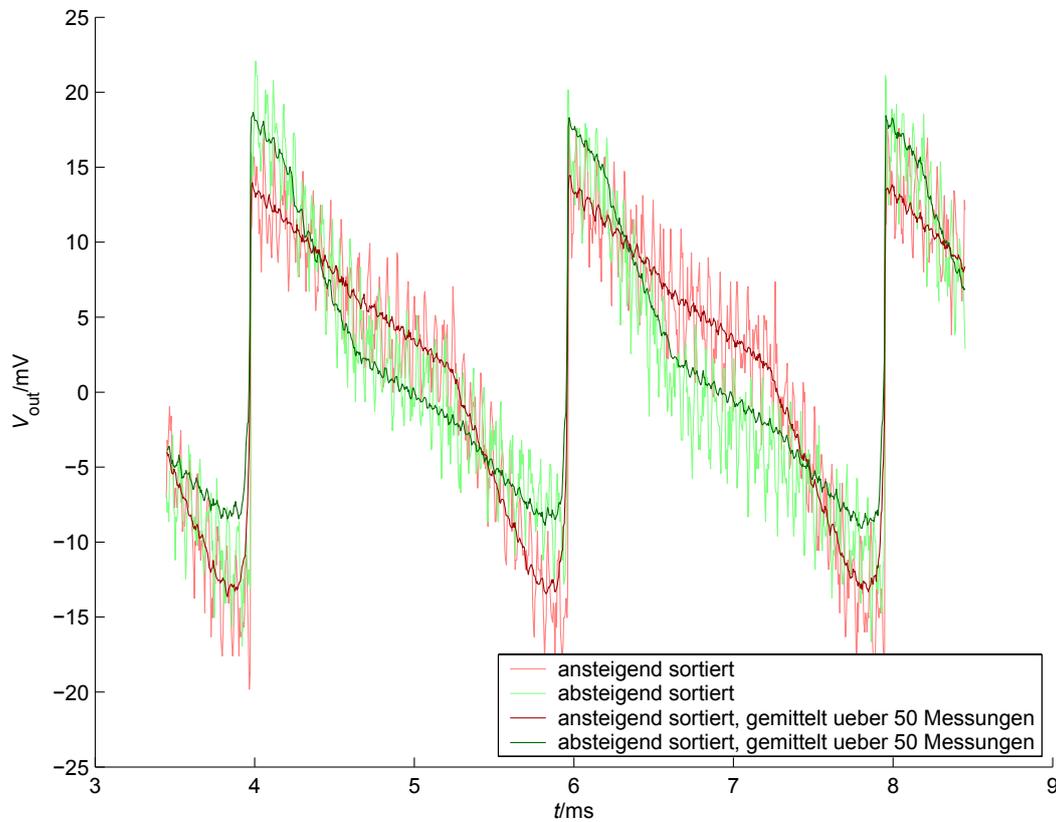
Dieses Verhalten ist gut in Abbildung 6.7 zu erkennen, die den Wechselspannungsanteil der Ausgangsspannung einer *Vout*-Schaltung zeigt. Die beiden dargestellten Kurven unterscheiden sich in der Reihenfolge, mit der die Stromspeicher und die *Vout*-Schaltungen vom DAC beschrieben werden. Einmal wurden die Zellen nach aufsteigendem Strom sortiert, einmal nach absteigendem. Dies hat Auswirkungen auf den Zeitpunkt, an dem die Nachbarzellen beschrieben werden, und somit auf das Übersprechen auf das *Gate* des NMOS-Transistors. Die Amplitude der entstehenden Sägezahn-Kurve ist von Zelle zu Zelle unterschiedlich und auch abhängig von  $V_{out}$ : Je größer  $V_{out}$ , desto größer die Amplitude des Sägezahns. Letzteres resultiert aus dem ohmschen Gesetz, je größer die Spannung über  $R$ , desto größer ist auch der Leckstrom. Die Unterschiede zwischen den Zellen beruhen wahrscheinlich auf geringfügig unterschiedlichen *Gate-Source*-Spannungen, die aufgrund der exponentiellen Abhängigkeit zu deutlich unterschiedlichen Leckströmen führen. Die in Abbildung 6.7 gezeigte Zelle hat einen vergleichsweise stark ausgeprägten Sägezahn-Effekt und ist auf 1,6 Volt eingestellt, die Abbildung zeigt also die maximalen Abweichungen.

Falls eine weitere *Spikey*-Version entwickelt wird, kann dieser Fehler leicht beseitigt werden. Zum Beispiel indem man an den Knoten  $V_1$  eine positive Spannung anlegt, wenn die Zelle nicht programmiert wird. Hierdurch ist  $V_{GS}$  vom NMOS negativ und die Leitfähigkeit des Transistors geht gegen null. Ein anderer Ansatz, der keinen neuen Chip benötigt, ist das Verringern der Periode des *Refresh*-Zyklus. Soll die Ausgangsspannung einer Zelle konstant bleiben, so ist die benötigte Zeit, um den Kondensator  $C_1$  auf den richtigen Wert zu laden, sehr klein, es muss nur die durch Leckstrom verursachte Differenz ausgeglichen werden. Soll eine Zelle jedoch umprogrammiert werden, so muss  $I_{DAC}$  länger zur Verfügung stehen. Bisher ist die *Refresh*-Zeit für alle Zellen gleich lang. Wäre diese jedoch dynamisch und abhängig vom zu schreibenden Wert, so sollte sich die Periode des *Refresh*-Zyklus stark verringern.

### 6.2.3 Kalibrierung der Analogspannungsquellen

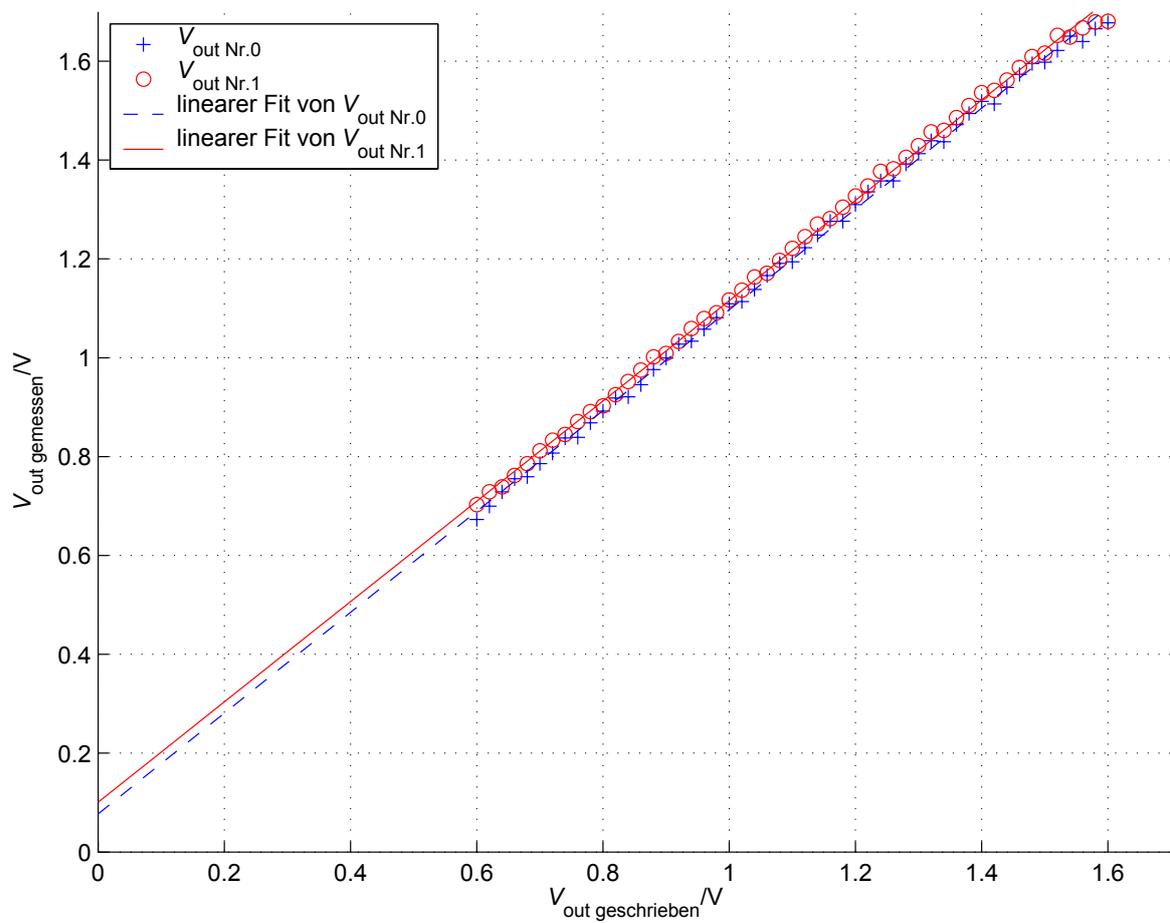
Dank der neuen *Analog-Readout* Schaltung in *Spikey 2* fällt die Vermessung der *Vout*-Schaltungen deutlich leichter als bei *Spikey 1*. Man kann jede *Vout*-Zelle auslesen und muss nicht vorher die *Probepads* anschließen. Abbildung 6.8 zeigt, dass die neuen Schaltungen schon ohne Kalibrierung besser funktionieren als diejenigen in *Spikey 1*. Es fällt auf, dass die Kurven von benachbarten Zellen nicht genau übereinstimmen. Allerdings sind die Abweichungen zwischen zwei Zellen geringer, als die Abweichung vom Idealwert.

Die schon bei *Spikey 1* verwendeten Kalibrierungsroutinen werden angepasst. Es werden für jeden *Vout* zwei Kalibrierungsparameter bestimmt und in der XML-Datei gespeichert. In Abbildung 6.9 ist

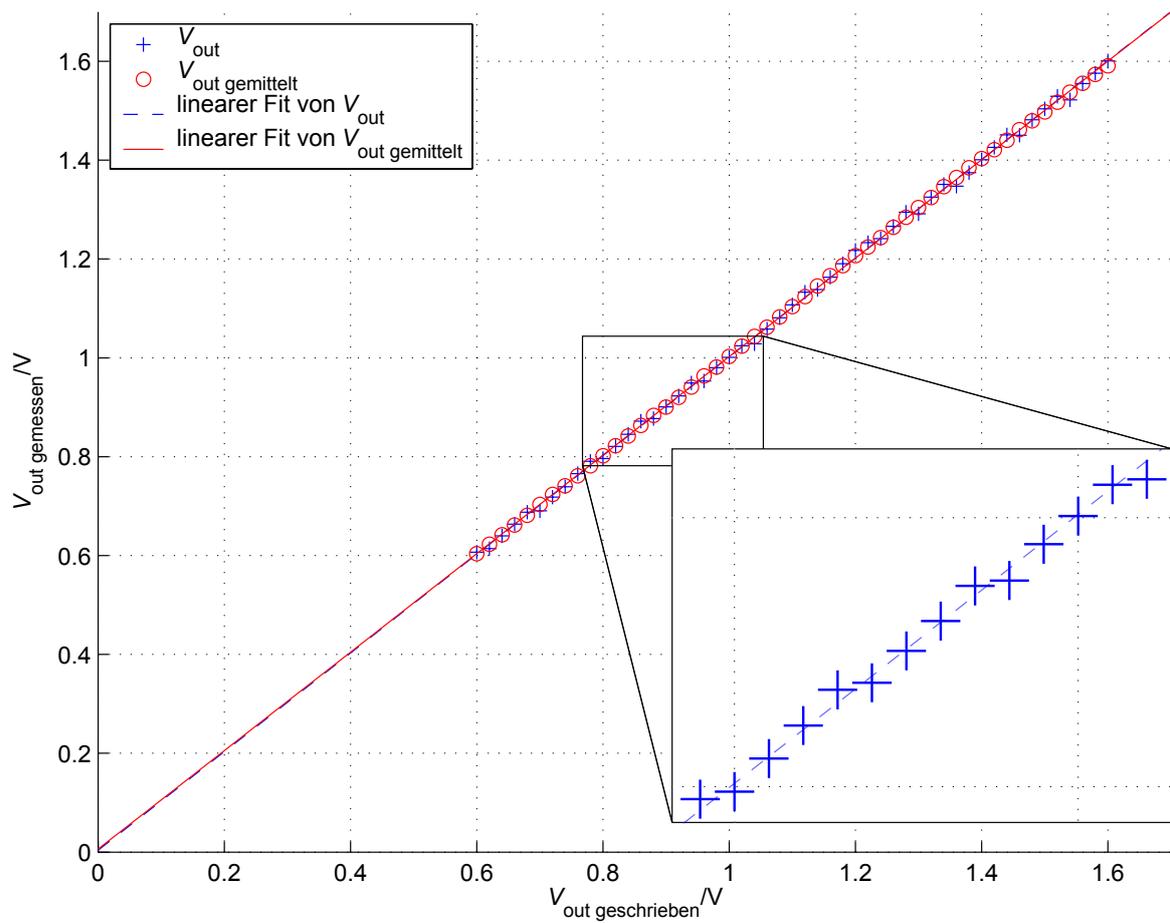


**Abbildung 6.7:** Wechselspannungsanteil des Ausgangs einer  $V_{out}$ -Schaltung, gut zu erkennen: *Refresh*-Zyklus und Abhängigkeit von Reihenfolge des *Refreshes*

das erfreuliche Ergebnis zusehen, wieder sind die Messwerte über die geschriebenen Werten aufgetragen. In der Vergrößerung kann man erkennen, dass sich die Abweichungen in der einfach-Messung sehr gut mit dem durch die Leckströme verursachten Sägezahn-Effekt erklären lassen. Die Frequenzen des *Refresh*-Zyklus und der Auslese stimmen nicht überein. Man misst nicht immer zur gleichen Phase des *Refresh*-Zyklus, sondern wandert auf dem Sägezahn entlang. Durch mehrfach-Messung mittelt sich dieser Effekt heraus.



**Abbildung 6.8:** Ausgangsspannungen zweier benachbarter, unkalibrierter  $V_{\text{out}}$ -Schaltungen

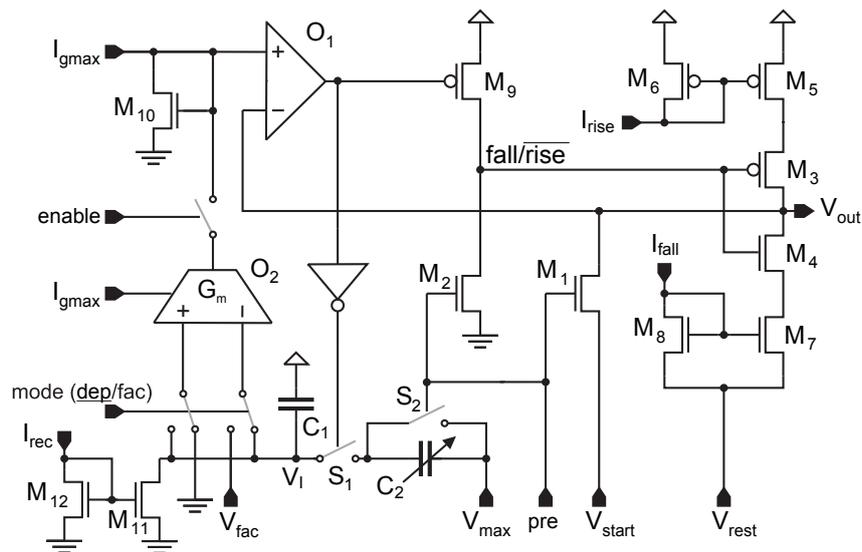


**Abbildung 6.9:** Ausgangsspannung einer kalibrierten  $V_{\text{out}}$ -Schaltung, einmal bzw. 100mal mit dem ADC gelesen

## 6.3 Short Term Plasticity

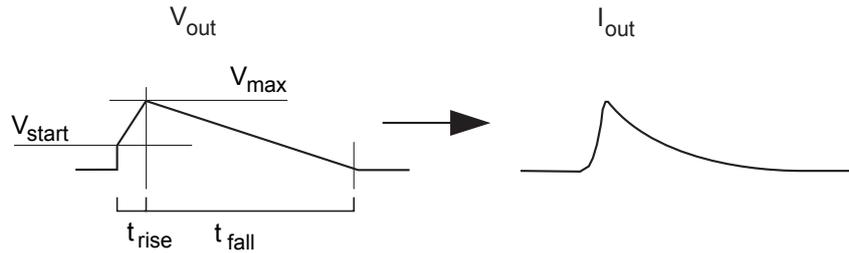
### 6.3.1 Schaltungsaufbau der Synapsentreiber

Die Plastizität wird in den Synapsentreibern erzeugt und ist damit für alle 192 Synapsen dieser Zeile gleich, vergleiche Abbildung 4.2. Für jede Zeile lässt sich einstellen, ob und welche Form der Plastizität auftreten soll sowie wie stark ein sich Aktionspotenzial auf den Effekt auswirken soll, repräsentiert durch  $U_{SE}$  aus dem in Abschnitt 3.3.3 eingeführtem Modell.  $\tau_{rec}$  sowie weitere Parameter, die das Short Term Verhalten beeinflussen wie zum Beispiel die maximale Stärke des Effekts lassen sich für Blöcke bestehend aus 128 Synapsentreibern festlegen. Den Schaltplan eines Synapsentreivers Abbildung 6.10. Die Erzeugung der Short Term Effekte findet im unteren linken Teil der Abbildung statt, zunächst wird jedoch der rechte Teil betrachtet. Hier wird die Grundlage für ein dem natürlichen nachgebildetes Aktionspotenzial geschaffen: Ein Spannungsverlauf, der später von den Stromsenken in den Synapsen zum gewünschten Aktionspotenzialverlauf umgewandelt wird, da der Strom im Bereich der schwachen Inversion exponentiell von  $V_{out}$  abhängig ist. Der Verlauf der Ausgangsspannung der Synapsentreiber sowie des Ausgangsstroms der Synapsen ist in Abbildung 6.11 dargestellt.



**Abbildung 6.10:** Schaltplan der Synapsentreiber mit integrierter Erzeugung des Short Term Verhaltens. Entnommen aus [32]

Ein Stromspiegel bestehend aus  $M_5$  und  $M_6$  erzeugt die ansteigende Rampe, die Steigung ist abhängig von  $I_{rise}$ , so kann die Zeit  $t_{rise}$  eingestellt werden. Die fallende Rampe wird analog dazu von  $M_7$  und  $M_8$  generiert, über  $I_{fall}$  wird  $t_{fall}$  kontrolliert. Der Übergang zwischen beiden Rampen wird durch das Umschalten der  $fall/rise$ -Leitung erreicht. Nach der Erzeugung eines Aktionspotenzials befindet sich diese Leitung auf dem Niveau der Versorgungsspannung,  $V_{out}$  wird über  $M_4$  und den unteren Stromspiegel auf das Ruhepotenzial  $V_{rest}$  gezogen. Im Ruhezustand bleibt die  $fall/rise$ -Leitung auf dem Potenzial Versorgungsspannung, im Folgenden als *high*-Zustand bezeichnet. Sowohl  $M_9$  als auch  $M_2$  sind zwar logisch „aus“, sie befinden sich im hochohmigen Bereich, allerdings handelt



**Abbildung 6.11:** Generierung eines Aktionspotenzials, Ausgang eines Synapsentreibers  $V_{out}$  und Ausgang einer Synapse  $I_{out}$

es sich bei  $M_9$  um einen Transistor mit reduzierter Schwellspannung, er hat also einen kleineren Widerstand als  $M_2$  und hält die Leitung auf  $V_{DD}$ .

Gelangt ein präsynaptisches Aktionspotenzial an die Synapse, wird die *pre*-Leitung 5 ns lang auf *high* gesetzt, was zunächst zwei Folgen hat. Zum einen springt  $V_{out}$  auf den Wert von  $V_{start}$ , zum anderen wird die *fall/rise*-Leitung über  $M_2$  auf Masse gezogen. Der obere Stromspiegel ist nun über den geschlossenen  $M_3$  mit  $V_{out}$  verbunden, die Rampenerzeugung beginnt. Der Sprung auf  $V_{start}$  ist nötig, da das in den Synapsen generierte Aktionspotenzial sonst zu langsam ansteigen würde.

$M_2$  wird zwar nach 5 ns wieder geschlossen, aber der Strom durch  $M_9$  ist so gering, dass sich das Potenzial der *rise/fall*-Leitung nur sehr langsam ändert. Der obere Stromspiegel sorgt weiterhin für ein Ansteigen der Ausgangsspannung, bis  $V_{max}$  erreicht ist. Danach wird auf den unteren Stromspiegel umgeschaltet, der  $V_{out}$  wieder auf  $V_{rest}$  zieht. Für das Umschalten sorgt der Komparator  $O_1$ , welcher *fall/rise* auf *high* setzt, wenn  $V_{out} > V_{max}$  übersteigt.  $V_{max}$  wird durch  $I_{gmax}$  und  $M_{10}$  generiert,  $M_{10}$  ist baugleich zu den Stromsenken in den Synapsen, es fällt bei gleichem Strom also die gleiche Spannung über ihm ab. Über  $I_{gmax}$  lässt sich somit  $V_{max}$  einstellen.

Die Plastizität wird im Wesentlichen in vier Komponenten erzeugt und über die *enable*-Leitung aktiviert.

- $C_1$  speichert die wiederhergestellte Wirksamkeit  $X_R$ . Diese ist gleich eins, wenn über  $C_1$  die Versorgungsspannung abfällt. In Kombination mit  $C_2$  sowie den Schaltern  $S_1$  und  $S_2$  kann die benutzte Wirksamkeit  $U_{SE}$  entnommen werden, die über den einstellbaren Wert von  $C_2$  festgelegt wird.
- Der Stromspiegel bestehend aus  $M_{11}$  und  $M_{12}$  sorgt für die Wiederherstellung von  $X_R$ ,  $I_{rec}$  bestimmt die Zeitkonstante  $\tau_{rec}$ .
- Über  $O_2$  findet die Kopplung der Wirksamkeiten an die schon beschriebene Erzeugung des Aktionspotenzials statt.
- Mithilfe der *mode*-Leitung wird *Depression* oder *Facilitation* ausgewählt.

Die Spannung am Knoten  $V_1$  repräsentiert die inaktive Wirksamkeit. Ohne Aktivität ist sie gleich null, da  $V_1$  über  $M_{11}$  auf Masse gezogen wird.  $C_1$  ist voll geladen,  $X_R$  ist eins. Zuvor wurden nur zwei Folgen eines präsynaptischen Aktionspotenzials erwähnt, die dritte Folge ist, dass  $C_2$  kurzgeschlossen und somit komplett entladen wird. Analog zum Verhalten ohne Plastizität steigt nun  $V_{out}$ , bis sie

$V_{\max}$  erreicht und  $O_1$  umschaltet. Dieser schließt zusätzlich auch den Schalter  $S_1$ , was dazu führt, dass Ladung vom voll geladenen  $C_1$  auf den entladenen  $C_2$  fließt. Die Größe von  $C_2$  ist einstellbar und zwar in 4 Schritten von  $1/8 \cdot C_1$  bis  $7/8 \cdot C_1$ . Zusammen mit  $V_{\max}$  bestimmt er die Menge der ausgetauschten Ladung und damit  $U_{SE}$ . Ist  $V_{\max}$  null, fließt gar keine Ladung,  $V_1$  bleibt auf dem Massepotenzial. Die größtmögliche  $U_{SE}$  erreicht man dagegen bei maximaler Größe von  $C_2$  und  $V_{\max}$  auf dem Potenzial der Versorgungsspannung. Ersatzschaltbilder für beide Fälle sind in Anhang E.1 abgebildet. In Formel 6.4 wird dies quantifiziert, sie beschreibt die Änderung der Spannung  $V_1$  vom  $n$ -ten auf das  $(n + 1)$ -te Aktionspotenzial [32]. Dies entspricht der Änderung der inaktiven Wirksamkeit und somit dem rechten Teil von Formel 3.5. Der linke Teil, die kontinuierliche Wiederherstellung von  $X_R$  in Anhängigkeit von  $\tau_{rec}$  kann über den Strom  $I_{rec}$  eingestellt werden, der dafür sorgt, dass sich  $C_1$  wieder auflädt.

$$V_{I,n+1} - V_{I,n} = \frac{C_2}{C_1 + C_2} (V_{\max} - V_{I,n}) \quad (6.4)$$

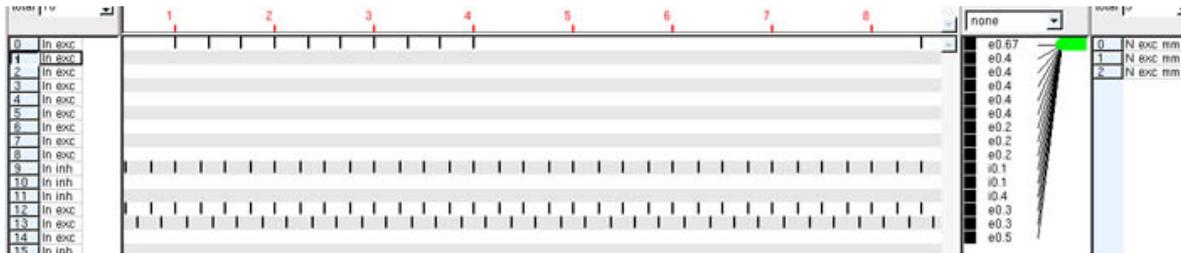
Eingekoppelt wird der Plastizitätseffekt durch den Operational Transconductance Amplifier (OTA)  $O_2$ . Dies ist ein Differenzverstärker, der eine Spannungsdifferenz am Eingang in einen Ausgangstrom überführt. Im Fall von *Depression*, wie in Abbildung 6.10 gezeigt, liegt Masse am nicht-invertierenden Eingang und  $V_1$  am invertierenden Eingang an. Hat ein vorheriges Aktionspotenzial dafür gesorgt, dass  $V_1$  positiv ist, so wirkt  $O_2$  als Stromsenke.  $I_{g\max}$  fließt nicht mehr komplett über  $M_{10}$ , sondern zum Teil über  $O_2$ . Dies hat zur Folge, dass  $V_{\max}$  und damit auch die Amplitude des Aktionspotenzials geringer sind als ohne *Depression*.  $O_2$  ist in der Lage,  $I_{g\max}$  komplett aufzunehmen, so dass  $V_{out}$  nach dem Sprung auf  $V_{start}$  sofort wieder auf den Ruhewert  $V_{rest}$  zurückgeht.

Im Fall von *Facilitation* ist  $V_1$  mit dem nicht-invertierenden Eingang verbunden, am invertierenden ist eine Referenzspannung  $V_{fac}$  angeschlossen. Im Gegensatz zur *Depression* ist der Ausgangstrom von  $O_2$  nicht mehr proportional zu  $-V_1$ , sondern zu  $V_1 - V_{\max}$ . Ist  $V_{fac}$  gleich null, so findet immer *Facilitation* statt,  $O_2$  wirkt als Stromquelle. Die Spannung über  $M_{10}$  steigt und somit erreicht auch das Aktionspotenzial höhere Werte, bevor  $O_1$  auf die fallende Rampe umschaltet. Erhöht man  $V_{fac}$ , so wird der *Facilitation*-Effekt geringer und schlägt dann in *Depression* um, für niedrigfrequente Aktionspotenziale früher, für hochfrequente später, da  $V_1$  frequenzabhängig ist.

### 6.3.2 Messergebnisse

Die Messungen zur *Short Term Plasticity* werden mit der HFacets Software [53] durchgeführt, die auf dem HANNEE-Framework<sup>24</sup> aufsetzt. Sie stellt eine grafische Benutzeroberfläche zur Verfügung, mit der man die Verbindungen der Neuronen untereinander, aber auch mit den externen Eingängen festlegen kann. Weiterhin werden Zeitleisten die Zeitpunkte dargestellt, an denen die Neuronen feuern. Über einen zweiten Block von Zeitleisten kann man die Feuerzeiten der externen Eingänge festlegen. Ein Ausschnitt der Software ist in Abbildung 6.12 gezeigt, es ist das im Folgenden beschriebene Setup zu sehen. Im linken Feld kann für jeden Synapsentreiber die Quelle der Spikes eingestellt werden. Das große Feld in der Mitte zeigt die Zeitleisten, auf denen man die Zeitpunkte für die Eingangsspiques festlegen kann, wenn man den externen Eingang als Quelle ausgewählt hat. In den beiden rechten Fenstern können die synaptischen Verbindungen und die zugehörigen Gewichte konfiguriert werden.

<sup>24</sup>HANNEE: Heidelberg Analog Neural Network Evolution Environment [54]



**Abbildung 6.12:** Ausschnitt aus der HFacets-Software, der das zur Messung der *Short Term Plasticity* verwendete Setup zeigt.

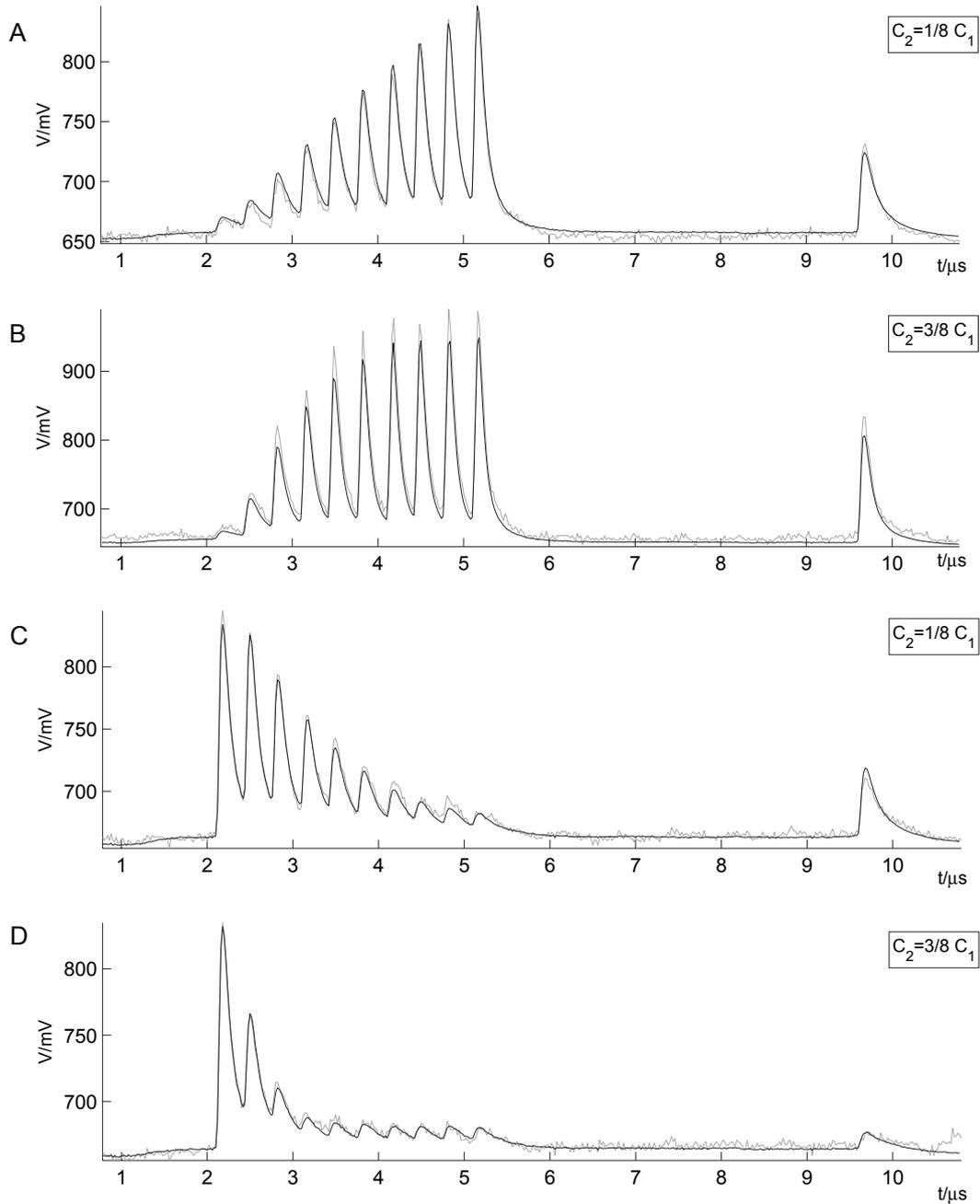
Mit der Software wird nun ein Setup aufgesetzt, bei dem nur externe Eingänge benutzt werden, es gibt keine Kopplung der Neuronen untereinander. Ein Neuron wird mit dem Oszillographen betrachtet, dessen Schwellspannung aufs Maximum gestellt wird, da nur der Verlauf der Membranspannung von Interesse ist, es sollen keine *Spikes* ausgelöst werden. Für eine einzelne Synapse wird die *Short Term Plasticity* aktiviert und das Synapsengewicht relativ groß gewählt. Um den Arbeitspunkt der Transistoren in der Neuronschaltung richtig einzustellen, ist es nötig, zusätzlich einige Synapsen zu aktivieren. Damit sich das Neuron von Anfang an im High Conductive State befindet und gehalten wird, feuern drei der externen Eingänge periodisch. Die Zeiten  $t_{\text{rise}}$  und  $t_{\text{fall}}$  sind sehr lang gewählt, die entstehenden Aktionspotenziale sind flach und gestreckt. Zusätzlich sind die Synapsengewichte klein. Dies bewirkt, dass die einzelnen Aktionspotenziale auf der Membranspannung nicht sichtbar sind, sondern eine gleichmäßige Anhebung bewirken. Ohne diese Hintergrundaktivität werden einzelne bzw. das erste Aktionspotenzial einer Kette stark unterdrückt.

Abbildung 6.13 zeigt die entstehenden Verläufe des Membranpotenzials. Bei ca.  $1,4 \mu\text{s}$  erkennt man den Anstieg der Membranpotenziale, der durch das Einsetzen der Hintergrundaktivität entsteht. Wie gewünscht sind die einzelnen Aktionspotenziale nicht zu erkennen.

In Teil **A** und **B** von Abbildung 6.13 wird *Facilitation* gezeigt, zu beachten sind einerseits die unterschiedliche Skala und andererseits die Verhältnisse von  $C_1$  zu  $C_2$ . Ein kleineres Verhältnis entspricht einer kleineren benutzten Wirksamkeit  $U_{\text{SE}}$ . Während bei Teil **A** die Amplitude der Aktionspotenziale kontinuierlich zunimmt, gibt es bei Teil **B** einen Bereich der Sättigung, die Amplitude wächst nicht mehr. In diesem Bereich ist  $V_1$  schon nahe bei  $V_{\text{max}}$ , die Menge der Ladung, die von  $C_1$  auf  $C_2$  transportiert wird, ist nur noch sehr klein und wird durch den Strom durch  $M_{11}$  bis zum nächsten Aktionspotenzial wieder ausgeglichen. In beiden Fällen erkennt man, dass der Plastizitätseffekt mit der Zeit wieder abklingt, das einzelne Aktionspotenzial am Ende des Experiments hat eine deutlich geringere Amplitude als die vorhergehenden.

Die Teile **C** und **D** zeigen *Depression*.  $g_{\text{max}}$  ist größer gewählt als in den Versuchen zur *Facilitation*, um den Effekt besser untersuchen zu können. Man erkennt in beiden Fällen deutlich die Abnahme der Amplitude mit zunehmender Anzahl von Aktionspotenzialen.

Vergleicht man die beiden Kurven mit dem größeren  $U_{\text{SE}}$ : **B** und **D**, so stellt man fest, dass der Sättigungsbereich bei *Depression* viel schneller erreicht ist. Dies hat zwei Ursachen. Zum einen sorgt im Fall der *Facilitation* die Spannung  $V_{\text{fac}}$  für eine Abschwächung des Effekts. Zum anderen wird bei jedem Aktionspotenzial kurzzeitig  $V_{\text{start}}$  an  $V_{\text{out}}$  gelegt. Die minimale Amplitude der Aktionspotenziale ist also durch  $V_{\text{start}}$  begrenzt, selbst wenn die *Depression* dafür sorgt, dass  $V_{\text{max}}$  kleiner ist. Dies erklärt auch die kleine Amplitude des einzelnen Aktionspotenzials in Teil **D**. Obwohl die Amplituden des vorhergehenden Aktionspotenzials in **C** und **D** gleich groß sind, hat sich die des einzelnen in **C**



**Abbildung 6.13:** Ergebnisse der *Short Term Plasticity* Messungen. Jeweils eine Einzelmessung ist exemplarisch in grau gezeigt, die schwarze Linie ist der Mittelwert aus 500 Einzelmessungen. **A** und **B** zeigen *Facilitation*, bei unterschiedlichem  $C_2$ , **C** und **D** zeigen *Depression*.

deutlich stärker vom *Depression*-Effekt erholt. Dies liegt daran, dass  $V_{\text{start}}$  die Amplitude des vorhergehenden Aktionspotenzials kontrolliert, und nicht  $V_1$ . Die Spannung  $V_1$  ist in  $\mathbf{D}$  größer, es dauert länger, bis sie vom Strom durch  $M_{11}$  wieder abgesenkt wird.

Die Messungen haben gezeigt, dass sich das in Abbildung 3.6 gezeigte und von den Formeln 3.5 und 3.6 beschriebene Verhalten gut reproduzieren lässt.



## Kapitel 7

# Zusammenfassung und Ausblick

In der Arbeit wurde erläutert, welche Wege beschritten werden, um die Funktionsweise des Gehirns zu ergründen. Das Verhalten einzelner Neuronen kann mittlerweile gut von Modellen beschrieben werden, Schwerpunkt der Forschungen sind deshalb Netzwerke mit vielen Neuronen. Neben Messungen am lebendigen Gehirn bietet sich die Verwendung von künstlichen Neuronalen Netzen an. In diesen werden Neuronmodelle simuliert, die sich im Detailgrad, mit dem das biologische Vorbild modelliert wird, stark unterscheiden können. Neuronale Netzwerke können entweder in Software simuliert werden, oder physikalisch in einen Mikrochip implementiert werden, wie im HAGEN und Spikey-Chip geschehen. Hierdurch erlangt man einen erheblichen Geschwindigkeitsvorteil von mehreren Größenordnungen. Durch Kombination von mehreren Chips lässt sich das Netzwerk vergrößern.

Das *Integrate-and-Fire* Modell, welches beim Spikey-Chip verwendet wird, beschreibt Neuronen phänomenologisch. Es modelliert das Membranpotenzial, nicht aber die verschiedenen Ionenarten, die es hervorrufen. Aktionspotenziale zeichnen sich nur durch den Zeitpunkt aus, an dem sie entstehen, die Form ist bei allen gleich und damit uninteressant. Gelangt ein Aktionspotenzial an eine Synapse, so wird beim postsynaptischen Neuron ein postsynaptisches Potenzial erzeugt, das das Membranpotenzial kurzfristig verändert. Überschreitet das Membranpotenzial dabei eine Schwellenspannung, so feuert das Neuron, es wird ein Aktionspotenzial erzeugt.

Der Spikey-Chip, das Untersuchungsobjekt dieser Arbeit hat ein pulskodiertes Netzwerk implementiert, die Feuerzeitpunkte der Neuronen sind entscheidend für das Netzwerkverhalten, nicht die Feuerraten. Zusammen mit einem erweiterten *Integrate-and-Fire* Modell und Leitfähigkeits-basierten Synapsen lässt sich das Verhalten des biologischen Vorbilds gut nachbilden, auch wenn die genauen inter- und intrazellulären Vorgänge vernachlässigt werden. Der Chip kann die Korrelation zwischen prä- und postsynaptischen Aktionspotenzialen messen, durch Veränderung der Synapsengewichte lässt sich somit *Spike Time Dependent Plasticity* realisieren. Ein zweiter integrierter Plastizitätsmechanismus modelliert *Short Term Plasticity*, ein Verhalten bei dem sich die Übertragungsfunktion einer Synapse ändert, wenn diese viele *Spikes* in einem kurzen Zeitraum empfängt.

Um den Spikey-Chip mit dem vorhandenen Test-System betreiben zu können, müssen die Nathan-Platinen modifiziert werden. Um das Entlöten eines störenden Sockels zu erleichtern wurde eine spezielle Lötspitze hergestellt, die alle Kontakte gleichzeitig erhitzen kann.

Die Träger-Platine für Spikey, Recha, wurde überarbeitet. Es können nun alle Bauteile direkt verlötet werden, es ist kein Fädeldraht mehr notwendig. Hinzugekommen sind zudem noch eine Re-

ferenzspannungsquelle, die Messungen mit dem ADC über den gesamten Spannungsbereich hinweg erlaubt, sowie ein Multiplexer samt Spannungsversorgung. Der Multiplexer ermöglicht die Beobachtung aller Neuronen über einen Oszillographen-Kanal, wobei allerdings immer nur eines zur Zeit dargestellt werden kann.

Für den hauseigenen *Autobonder* wurde ein Programm erstellt, mit welchem das direkte *Bonden* des Spikey-Chips auf die Recha-Platine zügig durchgeführt werden kann. Zum Schutz des Chips kann die neue Recha Version mit einer Schutzkappe versehen werden.

Die Messungen am Spikey 1 haben gezeigt, dass die Analogspannungsquellen nicht wie gewünscht funktionieren. Die Anwendung zweier Kalibrierungsroutinen hat die Genauigkeit der erzeugten Spannungen deutlich verbessert, allerdings sind nur zwei Ausgänge direkt messbar.

An dieser Stelle könnte noch weiter untersucht werden, ob die bei der Kalibrierung ermittelten Verteilungen von Ausgangsspannungen zwischen zwei Spikey-Chips gleich sind. Damit könnte bestimmt werden, ob der Fehler systematisch ist, oder von kleinen Abweichungen beim Herstellungsprozess bestimmt wird.

In Anhang A wird der Versuch zur Kalibrierung der Neuronen vorgestellt. Dabei hat sich herausgestellt, dass die Ausgangsspannungen der Analogspannungsquellen auch innerhalb eines Synapsenblocks voneinander abweichen, eine Kalibrierung einer exemplarischen Spannungsquelle pro Block ist also ungenügend. Es gestaltet sich problematisch, die Neuronen zu kalibrieren, da es viele Parameter gibt, die das Feuerverhalten beeinflussen. Eine Kalibrierung der Synapsengewichte mit statischen Parametern ist sehr grobkörnig und nur für gleichbleibende Eingangsfeuertaten gültig. An dieser Stelle ist noch viel Raum für Optimierung und alternative Ansätze. Weiterhin sollte untersucht werden, ob die Schwankungen der Durchschnittsfeuertaten von Neuronen mit grader und ungerader Nummerierung bei Spikey 2 ähnlich stark ausfallen.

Die Schaltung zur Erzeugung der *Short Term Plasticity* funktioniert sehr gut. Es konnte ein Parametersatz gefunden werden, bei dem das gemessene Verhalten dem des biologischen Vorbilds zufriedenstellend nahe kommt.

Auch die Analogspannungsquellen bei Spikey 2 zeigen nicht ganz das gewünschte Verhalten. Wie bei Spikey 1 besteht eine lineare Abweichung zwischen gewünschter und gemessener Spannung, die sich aber leicht korrigieren lässt. In diesem Fall kann sogar jede einzelne Spannungsquelle kalibriert werden. Bedingt durch den selteneren *Refresh* der Spannungswerte macht sich jedoch ein Leckstrom-Effekt bemerkbar, der einen Sägezahn-förmigen Verlauf der Spannungen verursacht.

Falls eine weitere Version vom Spikey-Chip entwickelt wird ist der Leckstrom-Effekt leicht korrigierbar, ein Vorschlag wird in Kapitel 6.2.2 gegeben. An der gleichen Stelle wird auch eine weitere Lösungsmöglichkeit gegeben, die auf einer Änderung der zum Schreiben der analogen Parameter verwendeten Software beruht.

Bisher wurden nur neuronale Netzwerke innerhalb eines Spikey-Chips simuliert. In naher Zukunft soll die Vernetzung mehrerer Chips ermöglicht werden, erforderlich ist dazu die Fertigstellung des *Event-Routings* [36]. Es sollte dann möglich sein kleinere, biologisch realistische Neuronenverbunde zu simulieren. Besonders interessant ist dabei die zeitliche Entwicklung des Synapsenverbindungen, welche durch die Plastizitätsmechanismen ermöglicht wird. Durch die hohe Geschwindigkeit der Spikey-Chips ist es dann in kurzer Zeit möglich, den Einfluss verschiedener Parametern zu untersuchen.

Eine dritte Spikey-Version erscheint zum jetzigen Zeitpunkt unwahrscheinlich, die Electronic Vision(s) Gruppe arbeitet schon an der nächsten Generation von Neuronaler Hardware. Die Verbindung zwischen zwei Netzwerkchips soll dabei so kurz wie möglich ausfallen. Die kürzest mögliche Ver-

bindung ist die zwischen zwei Chips auf einem unzerlegten *Wafer*. Man spricht von *Wafer-Scale-Integration*, der *Wafer* mit den Netzwerkchips wird nicht zersägt, sondern als ganzes in Betrieb genommen. Angestrebt sind Verbindungen mit Abständen im Mikrometer- Bereich, die mit Fotolithografie nachträglich auf dem *Wafer* erstellt werden. Über Hochgeschwindigkeits-Verbindungen soll die Vernetzung von mehreren *Wafers* realisiert werden.



# Anhang A

## Kalibrierung der Neuronen

Vergleicht man die Feuerraten unterschiedlicher Neuronen miteinander, so fällt auf, dass sie mitunter stark unterschiedlich sind, trotz gleicher Eingangsraten. Dies wird in Abbildung A.1 gezeigt, ein Balken stellt für jedes Neuron die Feuerrate dar. Die Messung wurde mit Spikey 1 durchgeführt, bei dem aufgrund einer nicht angeschlossenen Adressleitung nur jeweils die Hälfte der Neuronen pro Block adressierbar ist. Die Neuronen in der linken Hälfte liegen in einem Block, die Neuronen ab Nummer 97 im anderen Block. In der Messreihe wird jedes Neuron einzeln vermessen. Die Neuronen bekommen periodisch über exzitatorische Synapsen jeweils vier Eingangsspiques gleichzeitig. Da die Sensitivität der Neuronen allgemeingültig und unabhängig von den gewählten Synapsen gemessen werden soll, werden die vier Synapsentreiber nach jedem Eingangsspike gewechselt.

Auf den ersten Blick sind in Abbildung A.1 deutliche Unterschiede in den durchschnittlichen Feuerraten pro Block erkennbar. Insbesondere in der rechten Hälfte fällt jedoch auch ein Unterschied zwischen Neuronen mit gerader und mit ungerader Nummerierung auf. Die Umkehrpotenziale und die Schwellspannung sind jeweils pro Block für gerade und ungerade Neuronen separat einstellbar. Die gemessenen Abweichungen lassen sich also durch unterschiedliche analoge Spannungen erklären. Der verwendete Spikey-Chip ist vorher wie in Abschnitt 6.1.1 beschrieben kalibriert worden, offensichtlich genügt aber die Kalibrierung von einer Analogspannungsquelle pro Block nicht. Diese Messung sollte unbedingt mit Spikey 2 wiederholt werden, nach Kalibrierung aller Analogspannungsquellen.

Der erste Schritt zur Kalibrierung der Neuronen besteht in einer Angleichung der Durchschnittsfeuerraten pro Block und Parität der Neuronnummer untereinander. Dies wird erreicht durch Veränderung der Schwellspannungen. Dies ist jedoch nur eine Möglichkeit, die exzitatorischen Umkehrpotenziale oder die Ruhepotenziale haben ebenso Auswirkungen auf die Feuerraten. Es wird ein Algorithmus geschrieben, der die Schwellspannungen solange verändert, bis die Abweichungen der Durchschnittsfeuerraten einen gewissen Schwellwert unterschreiten. Das Ergebnis ist in Abbildung A.2 zu sehen, welche auf diese Weise kalibrierte Feuerraten als graue Balken zeigt. Die schwarzen Punkte werden später diskutiert. Die Verteilung der Messwerte sieht nun deutlich homogener aus, jedoch treten immer noch starke Schwankungen auf.

Es gibt nur wenige Parameter, die für jedes Neuron getrennt einstellbar sind. Dies sind der Strom, der das Neuron auf das Ruhepotenzial zurück bringt, die Geschwindigkeit, mit der der Komparator das Membranpotenzial mit der Schwellspannung vergleicht, und die Synapsengewichte. Da die Synapsengewichte den stärksten Einfluss auf die Feuerrate haben, setzt hier der nächste Schritt der Kalibrierung an. Die Synapsengewichte werden für jedes Neuron solange verändert, bis die Feuer-

rate, umgerechnet in die biologische Zeitskala, möglichst nah bei 2 Hz liegt. Dies ist ein Wert, der im Wachzustand bei exzitatorischen Synapsen gemessen werden konnte [55]. Die Eingangsfeuerrate beträgt in biologischer Zeitskala 100 Hz. Die Messergebnisse sind als schwarze Punkte in Abbildung A.2 zu sehen.

Die Unterschiede zwischen den Feuerraten können auf diese Weise deutlich vermindert werden. Allerdings ist die Modifikation der Synapsengewichte mitunter sehr grob. Bei der Erhöhung des Synapsengewichtes um eine Stufe springen die Feuerraten zum Teil von 0 Hz auf 7 Hz.

Eine Kalibrierung könnte nun erstellt werden, wenn das mittlere Synapsengewicht ermittelt, und dann für jedes einzelne Neuron die Differenz zum Mittelwert gespeichert wird. In Zukunft würden die Synapsengewichte dann um diese Differenz modifiziert werden.

Problematisch ist jedoch, dass diese Differenz von der Eingangsfeuerrate abhängig ist. Wird die Messung mit der halbierten Eingangsfeuerrate wiederholt, so liegt das mittlere Synapsengewicht entsprechend höher, und auch die Differenzen vom Mittelwert werden größer. Es ist also nicht möglich eine Kalibrierung mit statischen Werten zu erreichen, die Kalibrierungsparameter müssen an das Experiment angepasst werden. Störend kommt die Grobkörnigkeit dieser Art der Kalibrierung hinzu. Dies ist insbesondere bei sehr kleinen Synapsengewichten auffallend, einige Neuronen feuern schon beim kleinstmöglichen Synapsengewicht mit Raten, die andere Neuronen erst mit mehrere Stufen stärkeren Gewichten erreichen. Es ist bei diesen stark feuernden Neuronen nicht möglich, die Rate über die Synapsengewichte zu senken. Aufgrund dieser Einschränkungen und der Vielzahl an Abhängigkeiten wird die Suche nach einem Kalibrierungsalgorithmus nicht weiter verfolgt.

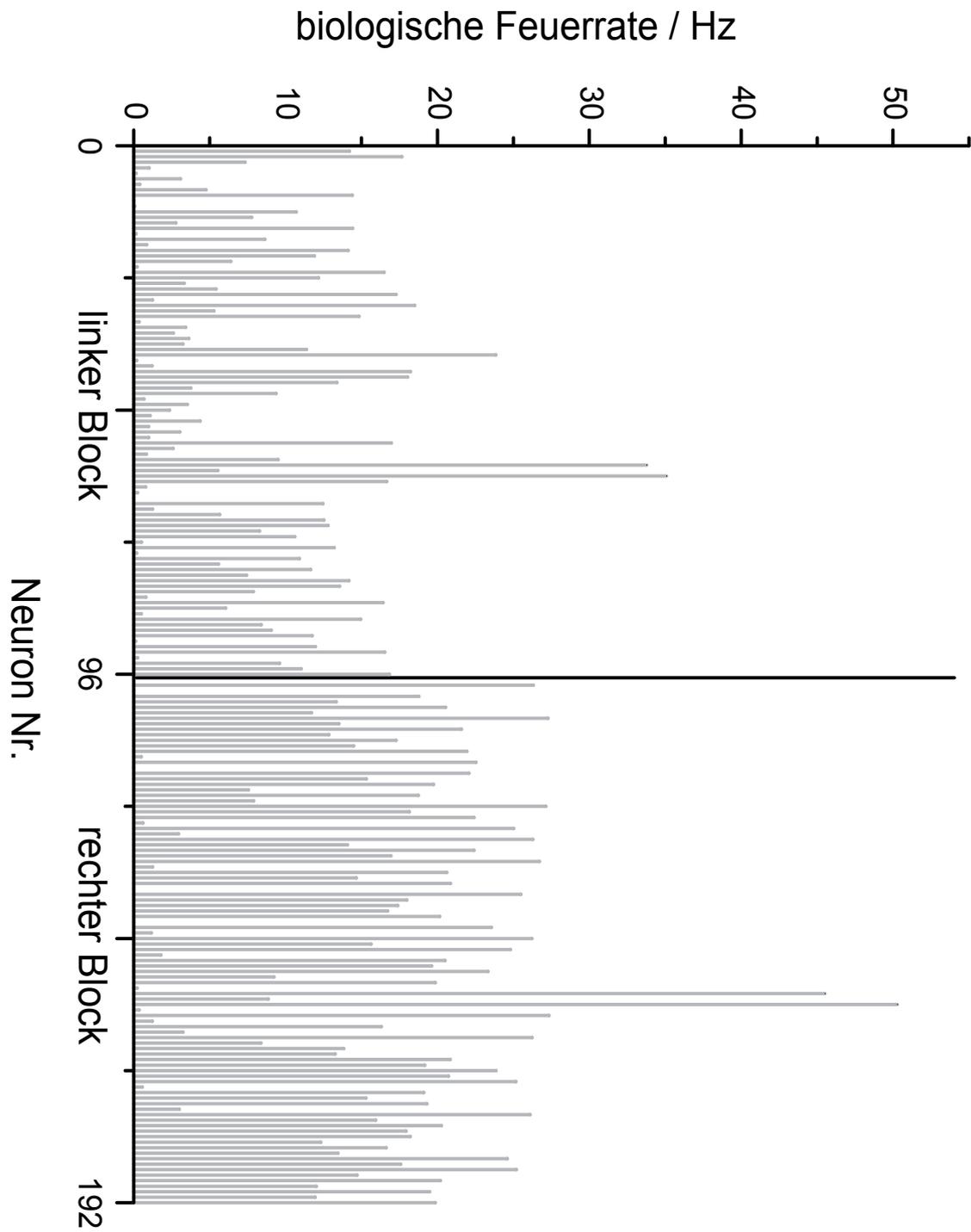
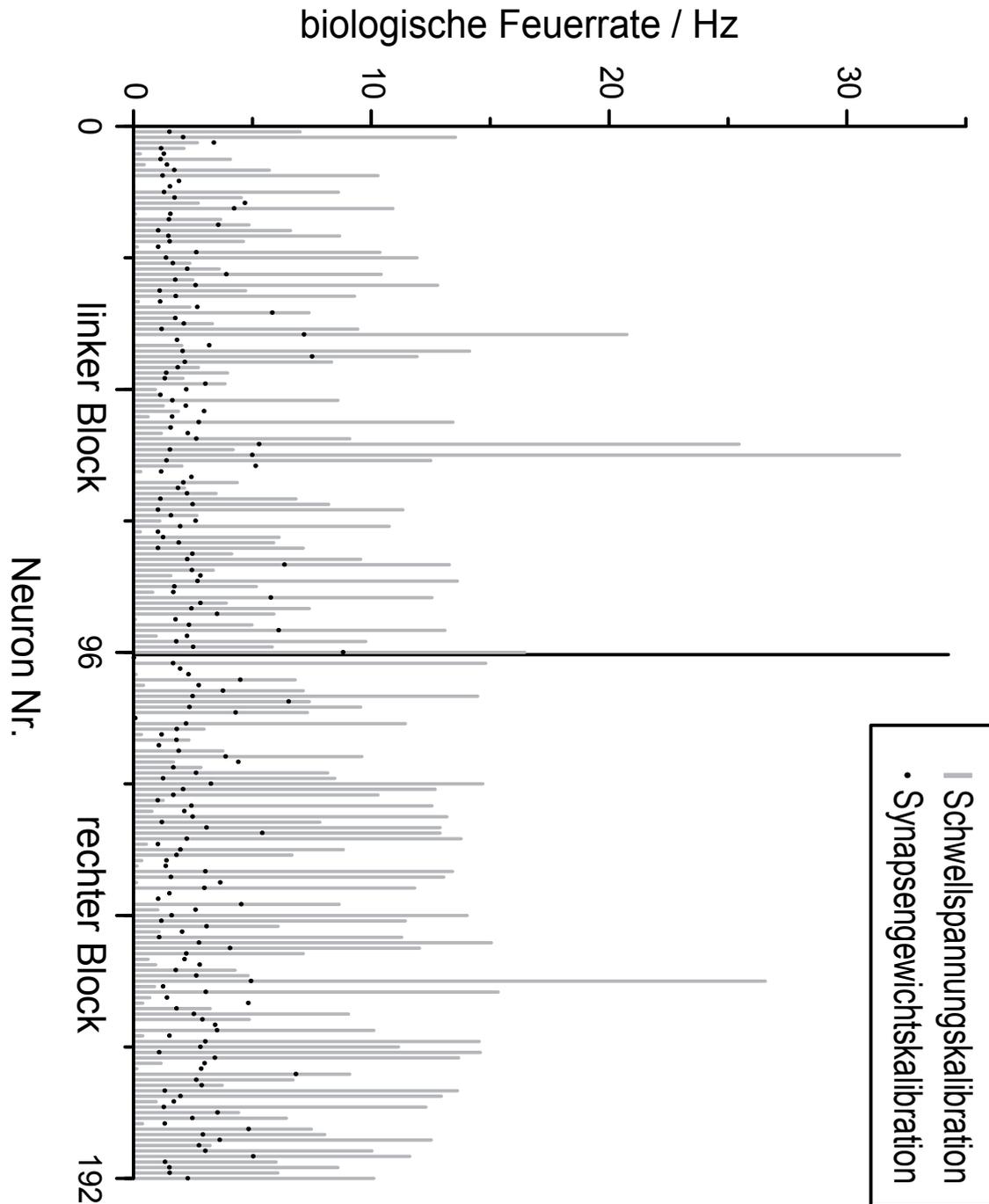


Abbildung A.1: Vergleich der Feuerraten der Neuronen.



**Abbildung A.2:** Vergleich der Feuerraten der Neuronen nach Kalibrierung der Schwellspannungen bzw. der Synapsengewichte.

## **Anhang B**

# **Schaltpläne**

Diese Seite bleibt leer. Die Schaltpläne der Recha V2-Platine befinden sich auf den folgenden Seiten.

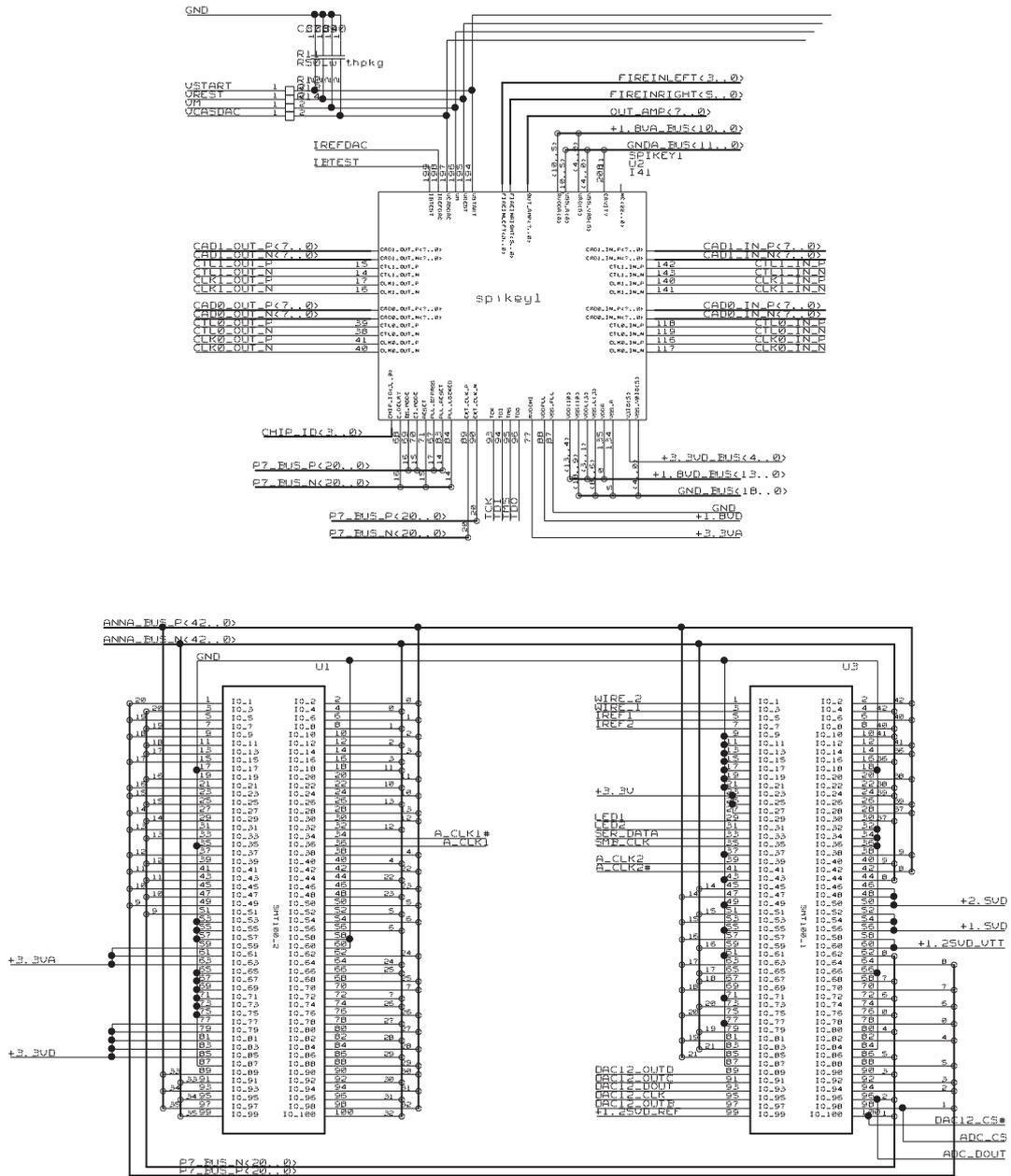
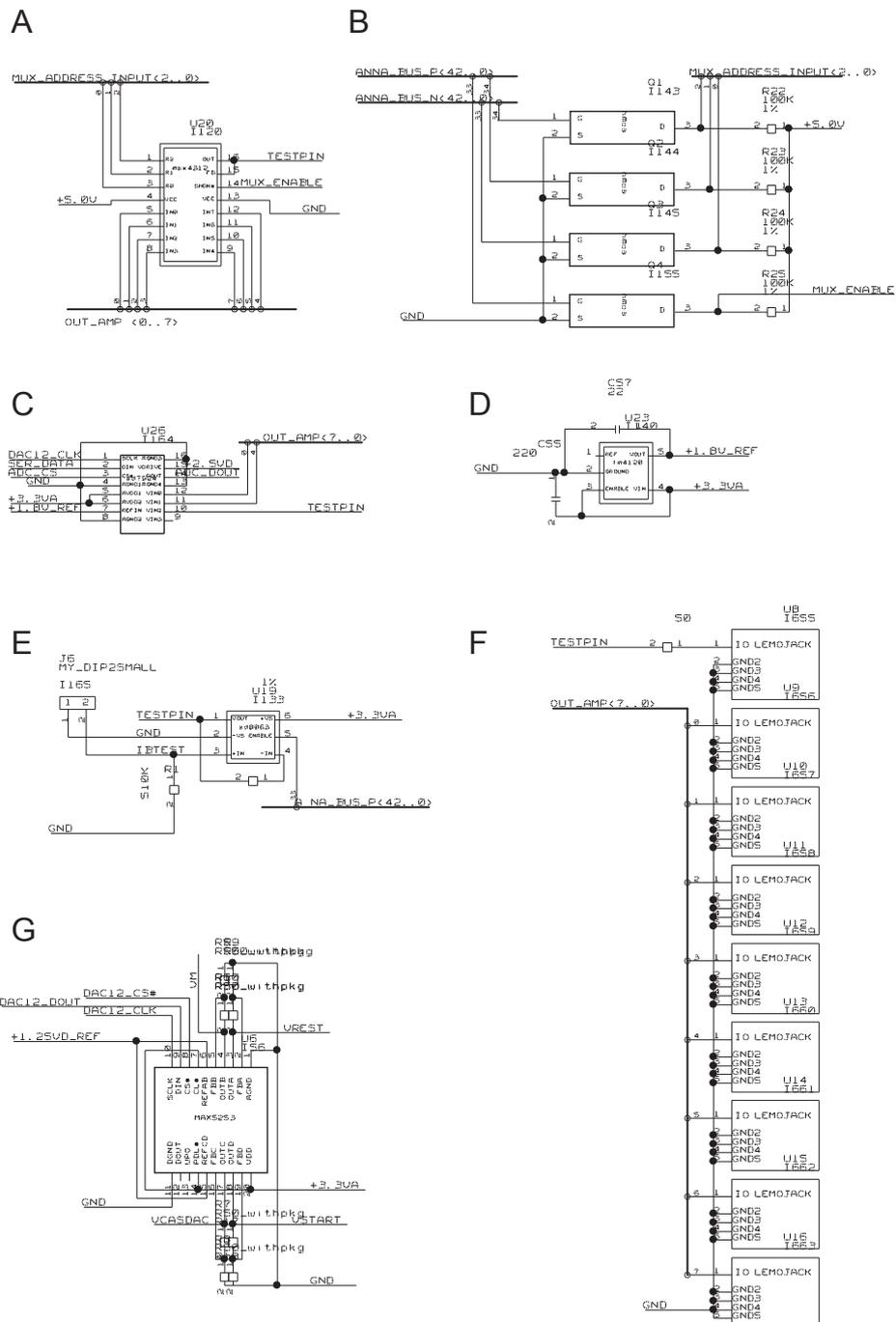
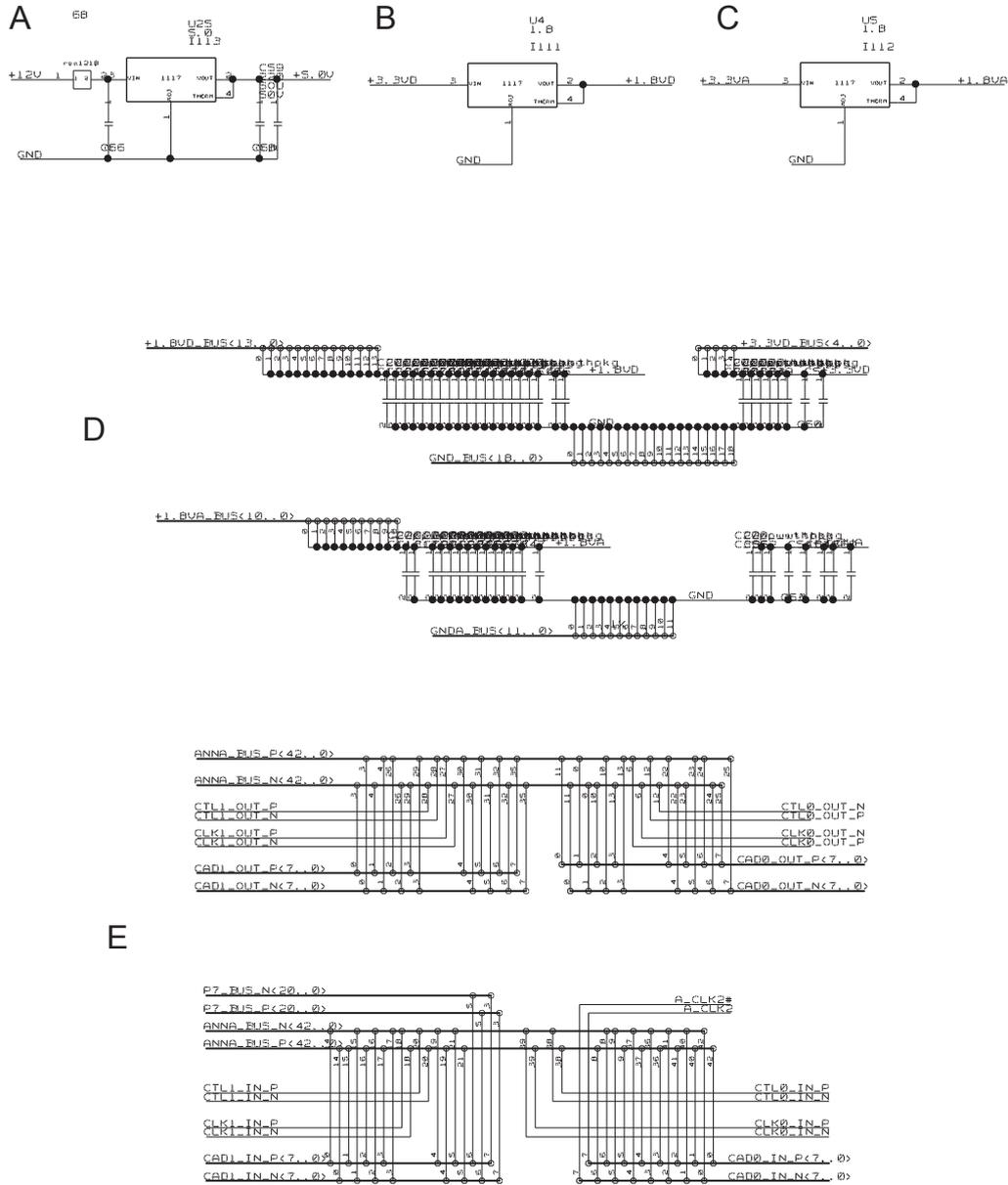


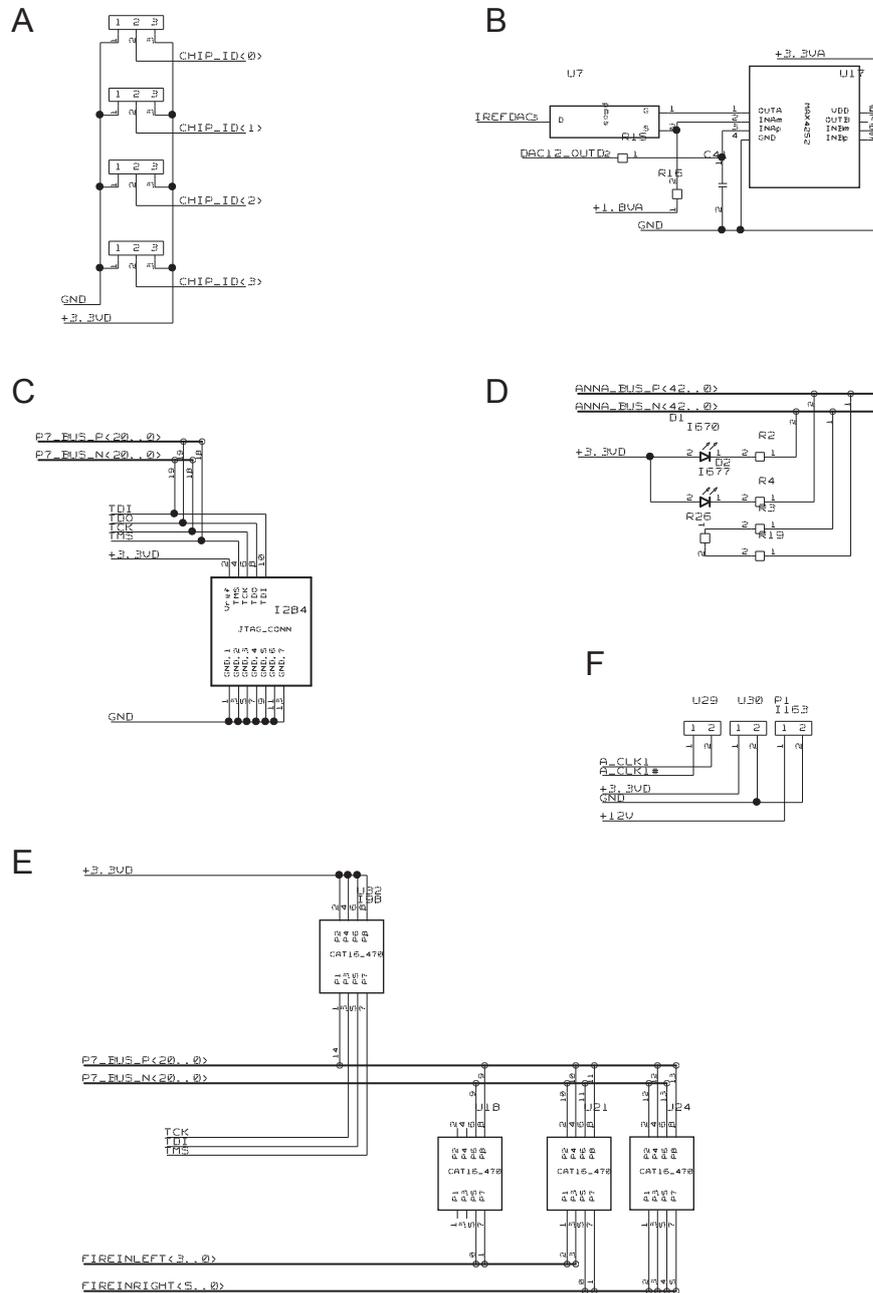
Abbildung B.1: Blatt 1: Spiky und SMT-Verbinder zur Nathan-Platine.



**Abbildung B.2:** Blatt 2: Multiplexer (A), Pegelwandler (B), ADC (C), Referenzspannungsquelle (D), OP-AMP (E), LEMO-Buchsen (F) und DAC (G).



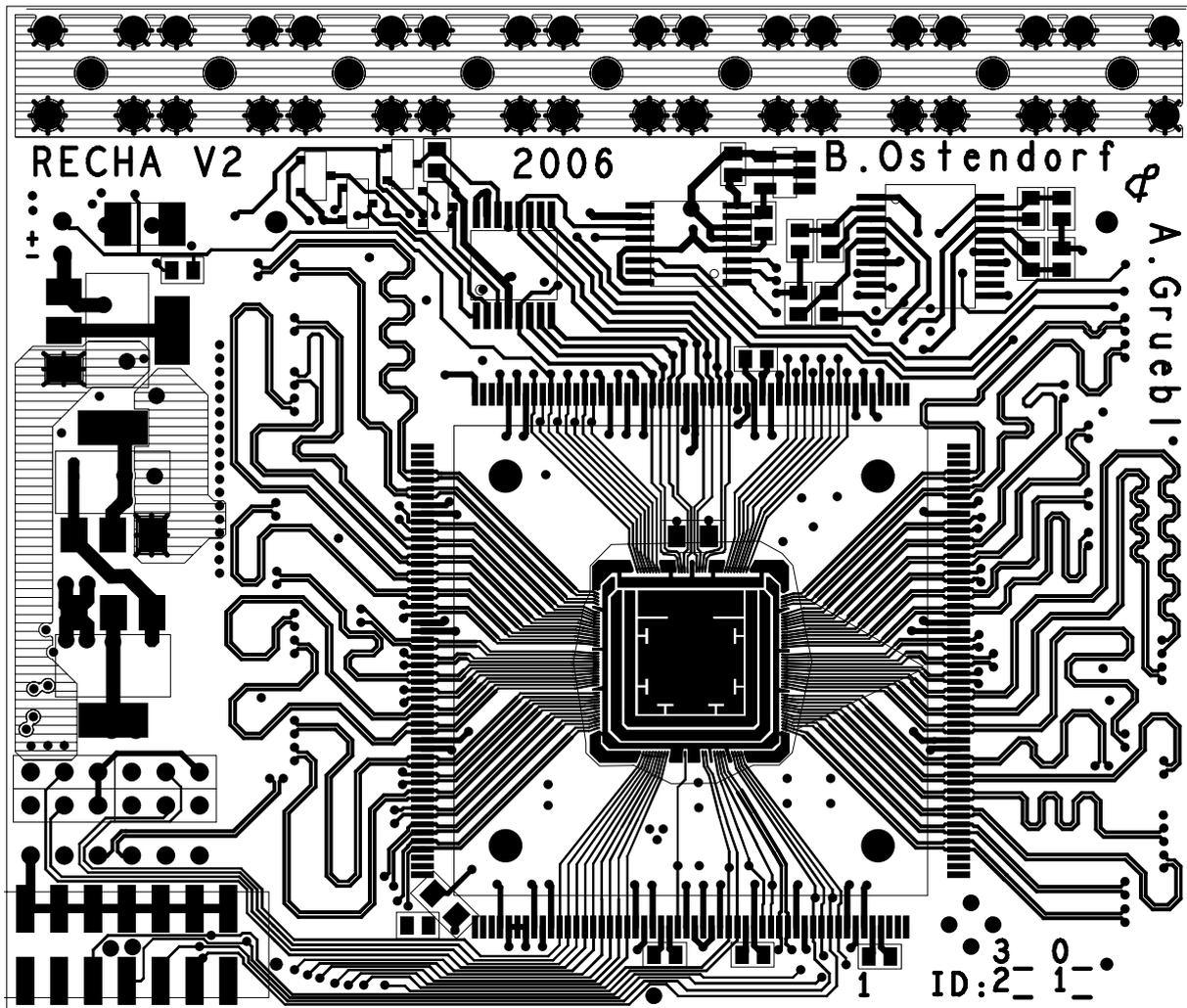
**Abbildung B.3:** Blatt 3: Spannungsregler: 5 V (A), 1,8 V digital (B), 1,8 V analog (C), Blockkondensatoren (D) und Signalbusse (E).



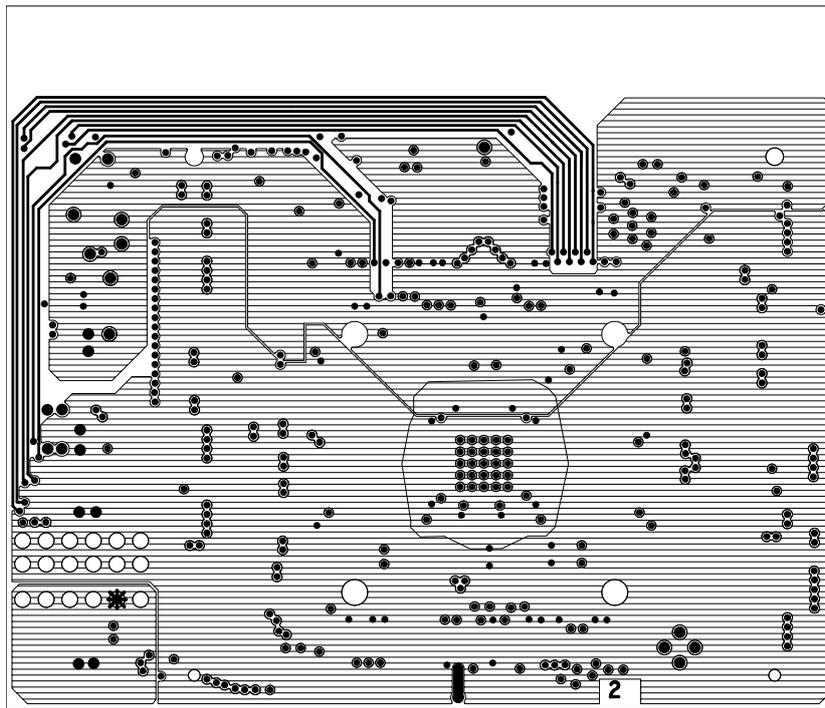
**Abbildung B.4:** Blatt 4: Chip-ID Selektion (A), OP-AMP (B), JTAG-Verbinder (C), LEDs (D), Anschlüsse für Takt und 12 V Spannungsversorgung (E) sowie Fire-In Leitungen (F).

## **Anhang C**

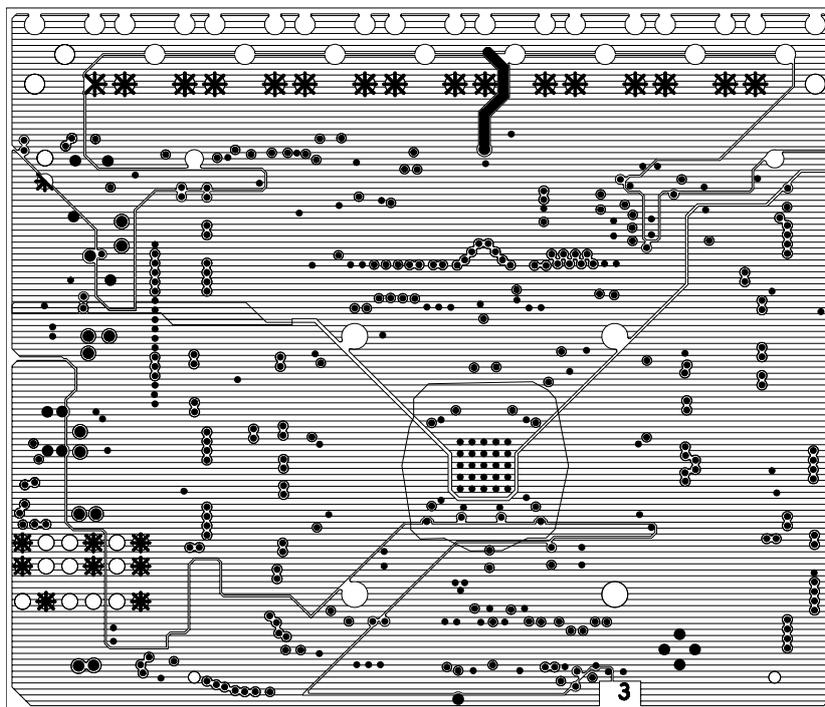
# **Lagenpläne**



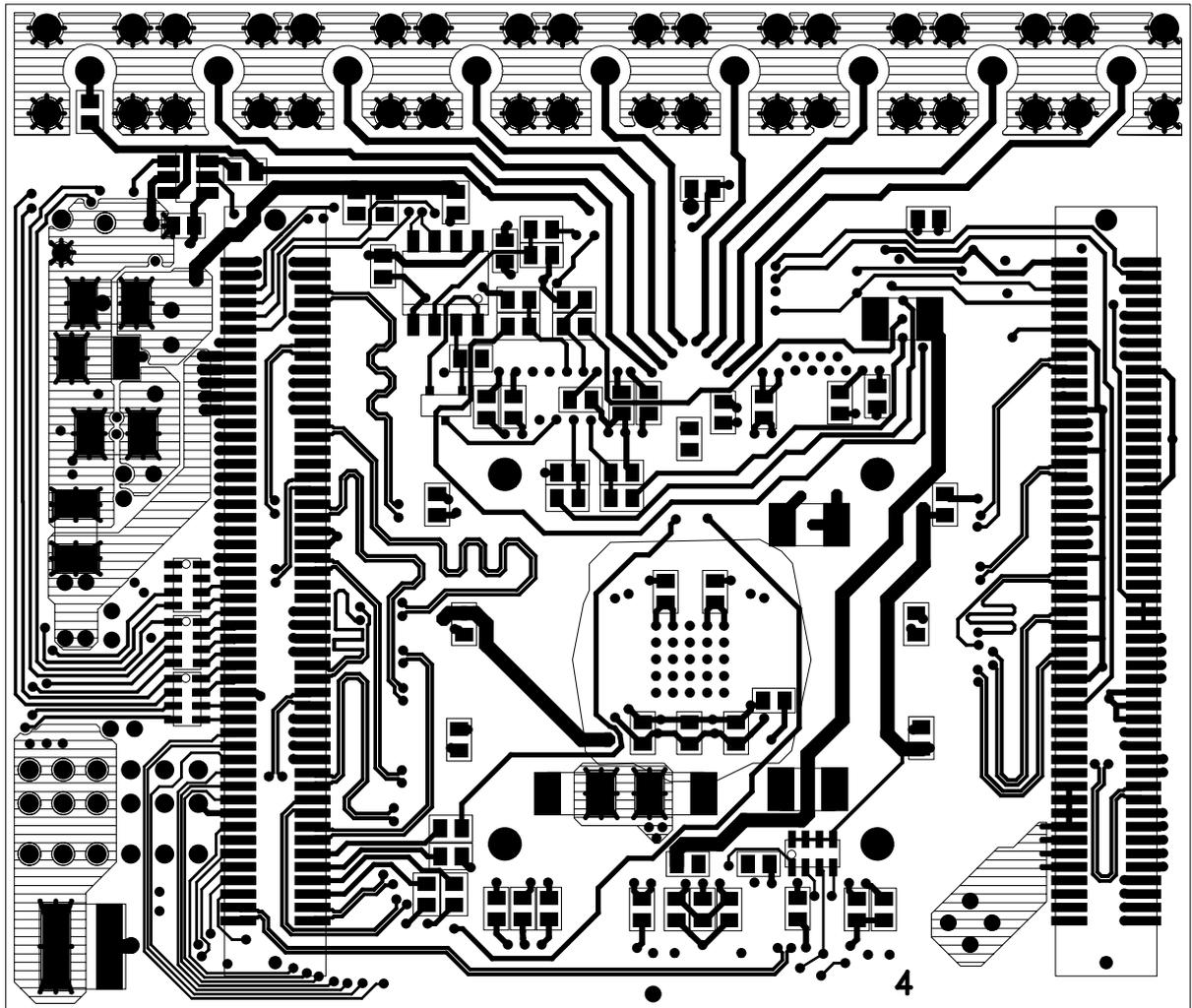
**Abbildung C.1:** Lage 1: Spikey mit Pads zum direkten *Bonden* und zum Auflöten im Package (Mitte), meanderförmige LVDS-Leitungen links und rechts von Spikey, Stromversorgung (links), LEMO-Buchsen (oben) sowie von links nach rechts: Pegelwandler, Multiplexer, ADC, Referenzspannungsquelle und DAC (zwischen Spikey und den LEMO-Buchsen).



**Abbildung C.2:** Lage 2: Spannungsversorgung: 1,8 V digital (unten), 1,8 V analog (oben), 12 V (unten links) und Fire-In Leiterbahnen (oben links).



**Abbildung C.3:** Lage 3: Spannungsversorgung: Masse-digital (untere Hälfte), Masse-analog (obere Hälfte), 3,3 V digital (unten links) und 3,3 V analog (oben).



**Abbildung C.4:** Lage 4: SMT-Steckverbinder zur Verbindung mit Nathan, analoge Leitungen zu den LEMO-Buchsen (oben) und jede Menge Blockkondensatoren.

## **Anhang D**

# **Bestückung**

Diese Seite bleibt leer. Der Bestückungsplan der Platine Recha V2 befindet sich auf den folgenden Seiten.

## Bestückungsliste Recha V2

Ref	Wert	Gehäuse	Menge	Best.-Nr.	Lieferant
C1-C36,C42-C44,C49, C52,C55,C59,C60	220 nF, X7R, 50 V	0603	44	Sortiment	
C37-C41,C57	22 nF, X7R, 50 V	0603	6	1301890	Farnell
C48,C50,C53,C56,C58	47 $\mu$ F, X7R, 6,3 V	1206	5	Sortiment	
C45-C47,C51,C54	10 $\mu$ F, X7R, 16 V	1206	5	9527729	Farnell
J1-J4,J6-J9	2mm Stiftleiste	2mm Raster	10	672105	Farnell
J5	JTAG-Connector	SMT14	1	-	-
D1,D2	Superhelle LEDs	0805	4	-	-
R1	510 k $\Omega$ , 1%	0603	1	Sortiment	
R2-R4,R19	Vorwiderstand LEDs	0603	4	Sortiment	
R5,R8,R9,R18,R22-R25	100 k $\Omega$ , 1%	0603	4	9330402	Farnell
R6,R7,R10,R17	43 k $\Omega$ , 1%	0603	1	9331204	Farnell
R11-R15	10 $\Omega$ , 1%	0603	5	9330429	Farnell
R16	5, 1 k $\Omega$ , 1%	0603	1	Sortiment	
R20,R21	50 $\Omega$ , 0,1%	0603	2	-	-
R26	unbestückt	-	-	-	-
U1,U3	SMT-Stecker	SMT100	2	361-6715	RS
U2	Spikey	direkt gebondet	1	-	-
U27	Spikey	CQFP208	1	-	-
U4,U5	REG1117A-1.8	SOT223	2	9778209	Farnell
U6	4-Channel DAC	SSOP20	1	MAX5253ACAP	MAXIM
U7	P-MOS BSS84	SOT23	1	436-8091	RS
U8-U16	LEMO-Buchsen, gewinkelt	-	9	Sortiment	
U17	AD8692ARZ	SOIC8	1	9425837	Farnell
U18,U21,U24	Widerstandsnetzwerk 4 $\times$ 220 $\Omega$	CAT16	3	241-9298	RS
U32	Widerstandsnetzwerk 4 $\times$ 1 k $\Omega$	CAT16	1	522-5406	RS
U19	AD8063	SOT23-6	1	523-6147	RS
U20	8-Channel MUX	QSOP16	1	MAX4312EEE	MAXIM
U22	RES 68 $\Omega$ , 5%, 1W	1218	1	9239944	Farnell
U23	LM4120AIM5-1.8	SOT23-6	1	9778772	Farnell
U25	LM1117MP-5.0	SOT223	1	9778209	Farnell
U26	AD7924BRUZ	TSSOP16	1	8397759	Farnell
U5	Temperatursensor	QSOP16	1	MAX1617	Maxim
Q1-Q4	N-MOS PMV45EN	SOT23	2	1081483	Farnell

Tabelle D.1: Bestückungsliste der Platine Recha V2.

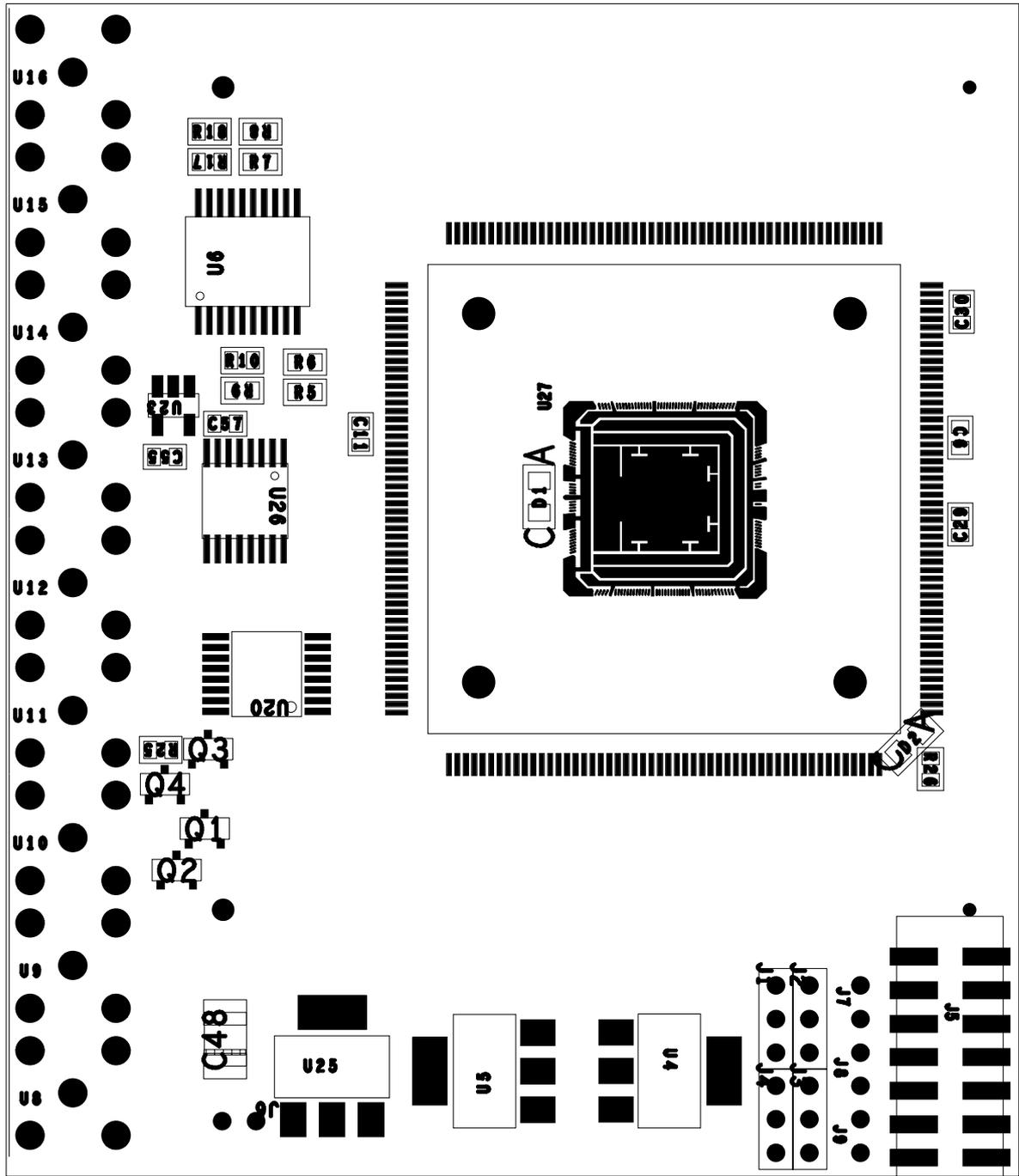


Abbildung D.1: Bestückungsplan der Recha V2-Platine, Oberseite.

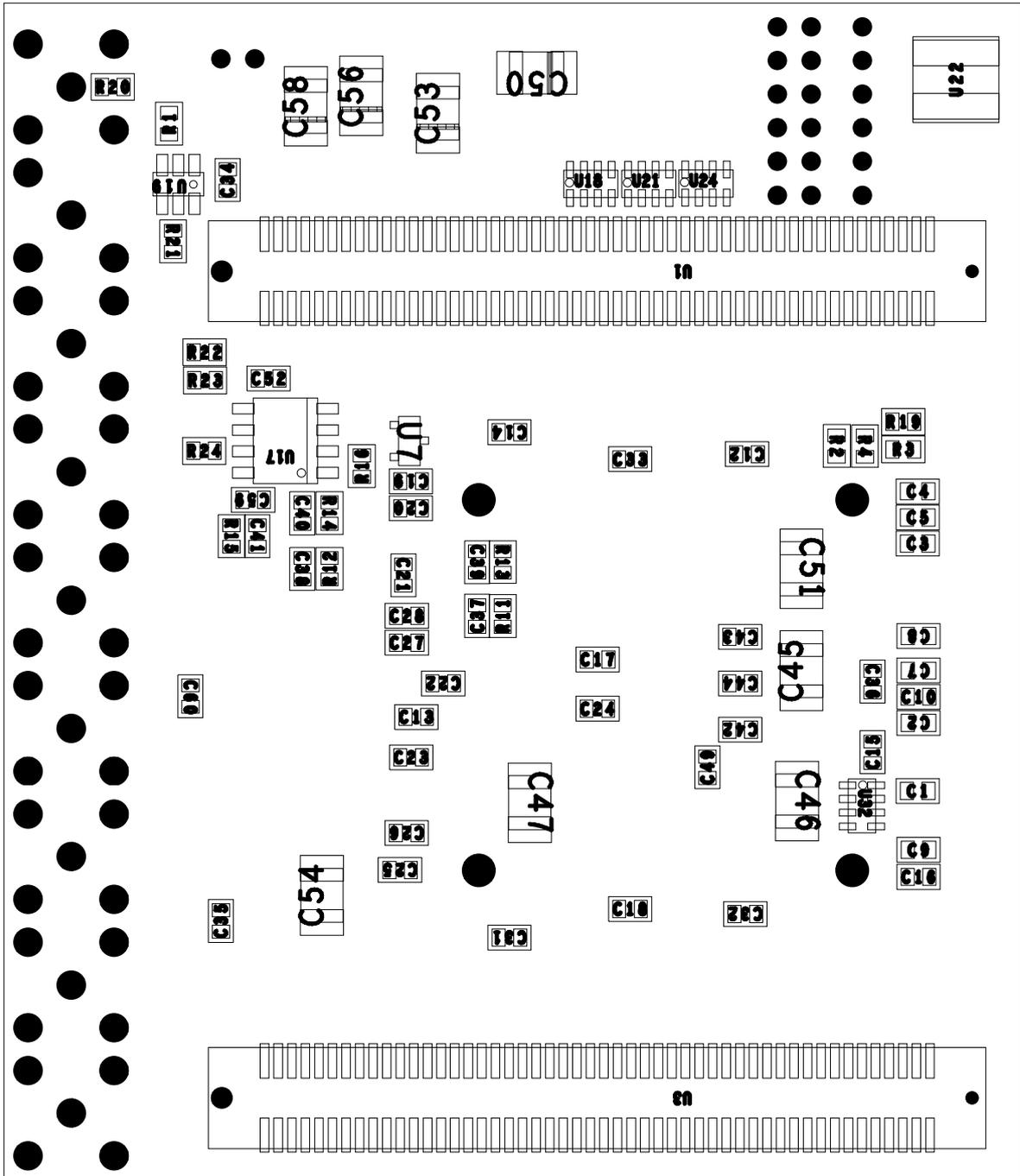
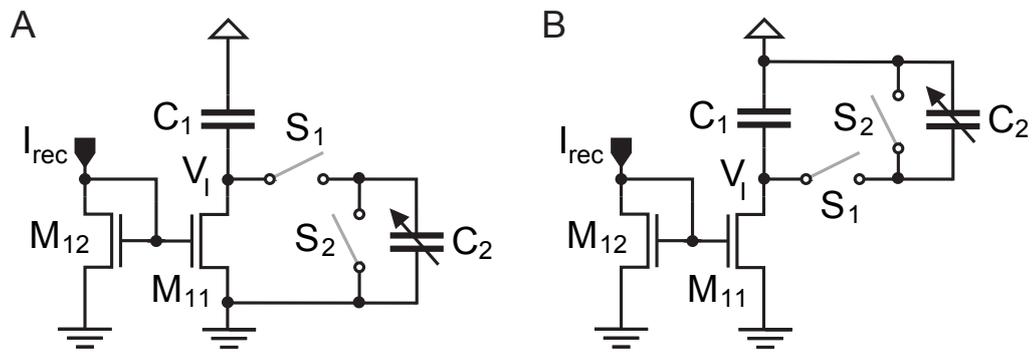


Abbildung D.2: Bestückungsplan der Recha V2-Platine, Unterseite.

# Anhang E

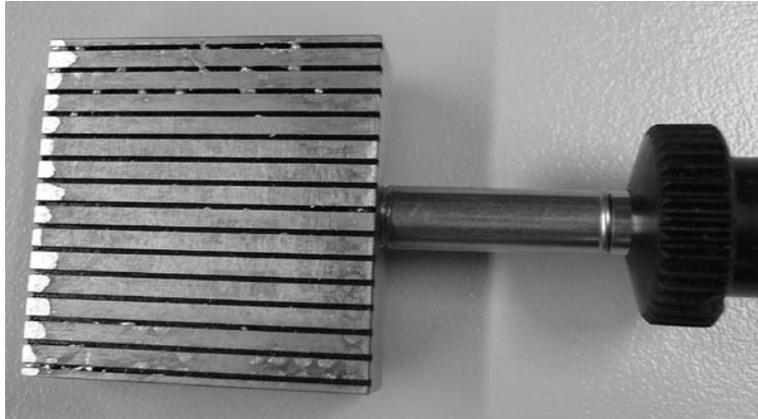
## Ergänzende Abbildungen

### E.1 Ersatzschaltbilder zur Short Term Plasticity Erzeugung



**Abbildung E.1:** Ersatzschaltbilder zum *Short Term Plasticity* Teil des Synapsentreibers. In Teil A wird der Fall mit  $V_{max} = 0\text{ V}$  gezeigt, in Teil B der mit  $V_{max} = V_{dd}$ .

## E.2 Lötstempel



**Abbildung E.2:** Lötstempel zum Auslöten des HAGEN-Sockels.

## **Anhang F**

# **Konfigurationsdateien zur Short Term Plasticity Messung**

Diese Seite bleibt leer. Die Konfigurationsdateien befinden sich auf den folgenden Seiten.

**Konfiguration für die Messungen zur Depression:**

```

SpikeyConfig
ud_chip 1
tsense 150 tpcsec 30 tpcorperiod 360
ud_dac 1
irefdac 25 vcasdac 1.6 vm 0 vstart 0.2 vrest 0.05
ud_param 1
synapse 0 drviout 2 adjdel 0.5 drvifall 1 drvirise 2
synapse 242 drviout 2 adjdel 0.5 drvifall 0.1 drvirise 0.1
synapse 244 drviout 2 adjdel 0.5 drvifall 1 drvirise 2
synapse 246 drviout 2 adjdel 0.5 drvifall 0.1 drvirise 0.1
synapse 247 drviout 2 adjdel 0.5 drvifall 1 drvirise 2
synapse 255 drviout 2 adjdel 0.5 drvifall 2 drvirise 1
synapse 256 drviout 2 adjdel 0.5 drvifall 1 drvirise 2
synapse 498 drviout 2 adjdel 0.5 drvifall 0.1 drvirise 0.1
synapse 500 drviout 2 adjdel 0.5 drvifall 1 drvirise 2
synapse 502 drviout 2 adjdel 0.5 drvifall 0.1 drvirise 0.1
synapse 503 drviout 2 adjdel 0.5 drvifall 1 drvirise 2
synapse 511 drviout 2 adjdel 0.5 drvifall 2 drvirise 1
end
neuron 0 ileak 2 icb 0.6
end
biasb 2 2 2 2 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.1 0.4 0.6
outamp 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 1
vout      0.6 0.6 0.8 0.8 0.6 0.6 1.4 1.2 1.55 1.55 0.89 0.8 0.5 0.5 0.5 1 1.8 1.8 1.6 1.2
          0.7 0.7 0.6 0.6 0.3 0.3 1.7 1.7 1.55 1.55 0.89 0.8 1.1 1.1 1.6 1.6 1.1 1.15 1.6 1.1
voutbias  0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
          0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
probepad  0.8
          0.8
probebias 0.2
          0.2
ud_rowconfig 1
synapse 0 config 00000000
synapse 241 config 00000100
synapse 244 config 00001000
synapse 247 config 00000100
synapse 255 config 01110100
synapse 256 config 00000000
end
ud_colconfig 1
neuron 0 config 0110
neuron 3 config 0000
end
ud_weight 1
w 0 Gewicht 0 bis 240 sind alle 0, 241 bis 255: 7 4 4 6 1 1 3 3 3 6 6 6 6 6 a
w 1 Gewicht 0 bis 240 sind alle 0, 241 bis 255: 0 0 0 3 3 3 3 3 3 3 3 3 3 3 a
    
```

w 2 Gewicht 0 bis 240 sind alle 0, 241 bis 255: 7 1 1 6 1 1 6 6 6 6 6 6 6 a  
end EndSpikeyConfig

**Konfiguration für die Messungen zur Facilitation:**

```

SpikeyConfig
ud_chip 1
tsense 150 tpcsec 30 tpcorperiod 360
ud_dac 1
irefdac 25 vcasdac 1.6 vm 0 vstart 0.2 vrest 0.05
ud_param 1
synapse 0 drviout 2 adjdel 0.5 drvifall 1 drvirise 2
synapse 242 drviout 2 adjdel 0.5 drvifall 0.1 drvirise 0.1
synapse 244 drviout 2 adjdel 0.5 drvifall 1 drvirise 2
synapse 246 drviout 2 adjdel 0.5 drvifall 0.1 drvirise 0.1
synapse 247 drviout 2 adjdel 0.5 drvifall 1 drvirise 2
synapse 255 drviout 2 adjdel 0.5 drvifall 2 drvirise 1
synapse 256 drviout 2 adjdel 0.5 drvifall 1 drvirise 2
synapse 498 drviout 2 adjdel 0.5 drvifall 0.1 drvirise 0.1
synapse 500 drviout 2 adjdel 0.5 drvifall 1 drvirise 2
synapse 502 drviout 2 adjdel 0.5 drvifall 0.1 drvirise 0.1
synapse 503 drviout 2 adjdel 0.5 drvifall 1 drvirise 2
synapse 511 drviout 2 adjdel 0.5 drvifall 2 drvirise 1
end
neuron 0 ileak 2 icb 0.6
end
biasb 2 2 2 2 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.1 0.4 0.6
outamp 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 1
vout      0.6 0.6 0.8 0.8 0.6 0.6 1.4 1.2 1.55 1.55 0.89 0.8 0.5 0.5 1 1 1.8 1.8 1.6 1.2
          0.7 0.7 0.6 0.6 0.3 0.3 1.7 1.7 1.55 1.55 0.89 0.8 1.1 1.1 1.6 1.6 1.1 1.15 1.6 1.1
voutbias  0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
          0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
probepad  0.8
          0.8
probebias 0.2
          0.2
ud_rowconfig 1
synapse 0 config 00000000
synapse 241 config 00000100
synapse 244 config 00001000
synapse 247 config 00000100
synapse 255 config 01010100
synapse 256 config 00000000
end
ud_colconfig 1
neuron 0 config 0110
neuron 3 config 0000
end
ud_weight 1
w 0 Gewicht 0 bis 240 sind alle 0, 241 bis 255: 7 4 4 6 1 1 3 3 3 6 6 6 6 6 a
w 1 Gewicht 0 bis 240 sind alle 0, 241 bis 255: 0 0 0 3 3 3 3 3 3 3 3 3 3 3 a

```

w 2 Gewicht 0 bis 240 sind alle 0, 241 bis 255: 7 1 1 6 1 1 6 6 6 6 6 6 6 a  
end EndSpikeyConfig

# Abbildungsverzeichnis

2.1	Strom-Spannungskennlinie einer pn-Diode. Entnommen aus [8]	5
2.2	Schnitt durch einen NMOS-Transistor	5
2.3	<i>Drainstrom</i> eines MOS-Transistors in Abhängigkeit von $V_{DS}$	7
2.4	Übergang zwischen den verschiedenen Arbeitsbereichen eines MOS-Transistors. Entnommen aus [9].	7
2.5	Ein Schalter realisiert durch einen einzelnen NMOS (A) bzw. PMOS (B) sowie ein <i>Transmission-Gate</i> (C)	8
2.6	Ein <i>Fully Compensated Transmission-Gate</i> zur Vermeidung von <i>Charge Injection</i>	9
2.7	Ein einfacher Stromspiegel	10
2.8	Ein <i>Push-Pull</i> -Inverter, Schaltplan (A) und Ersatzschaltbild (B).	10
3.1		14
3.2	Zeichnung einiger Neuronen aus dem Visuellen Kortex einer Katze von Ramón y Cajal [18].	15
3.3	Reaktion des Membranpotenzials eines postsynaptischen Neurons auf exzitatorische präsynaptische Aktionspotenziale. Entnommen aus [11].	17
3.4	Synapsen- und Neuronschaltung eines einfachen <i>Integrate-and-Fire</i> Modells	18
3.5	<i>Integrate-and-Fire</i> Modell: Neuronschaltung für Leitfähigkeits-basierte Synapsen	20
3.6	Beispiele von <i>Short Term Facilitation</i> (A) und <i>Depression</i> (B). Entnommen aus [30] bzw. [31]	22
4.1	Fotografie des Spikey 2 Chips	27
4.2	Kernkomponenten des neuronalen Netzwerkes: Synapsentreiber, Synapsen und Neuronen. Entnommen aus [32]	28
4.3	Blockdiagramm des Spikey-Chips. Entnommen aus [35]	29
5.1	Backplane mit vier eingesteckten Nathans.	32
5.2	Nathan mit FPGA und Steckverbindern für die Tochterplatine Recha. Der zwischen den Steckverbindern liegende HAGEN-Sockel ist bereits entfernt.	33
5.3	Nathan mit aufgesetztem Entlöt-Kran und ausgelötetem HAGEN-Sockel.	34
5.4	Recha-Platine, mit direkt <i>gebondetem</i> Spikey-Chip unter einem Schutzdeckel (A) bzw. ohne Deckel (B).	35
5.5	Ein <i>Probepad</i> auf Spikey 1, vollkommen verdeckt vom <i>Bonddraht</i> .	36

5.6	Ein Membranpotenzial überschreitet die Schwellspannung und löst einen <i>Spike</i> aus. Dargestellt sind die direkte Messung und die Messung nach Durchleitung durch den Multiplexer. Der Spannungsteiler, bestehend aus Ausgangs- und Terminierungswiderstand, halbiert die Amplitude des Signals der direkten Messung. Am Eingang des Multiplexers liegt nur noch die halbierte Amplitude an, hinter dem Multiplexer wird diese wiederum vom Spannungsteiler halbiert. Die dargestellten Messwerte sind aus diesem Grund verdoppelt, bzw. vervierfacht worden. . . . .	37
5.7	Das gleiche Membranpotenzial wie in Abbildung 5.6, jedoch mit mit linearen Korrekturen in den hinter dem Multiplexer gewonnenen Messwerten: Es wurde ein Offset von 65 mV addiert, die Messwerte wurden nicht mit dem Faktor 4, wie in Abbildung 5.6, sondern mit 3,7 multipliziert. . . . .	38
5.8	Pegelwandler an den digitalen Eingängen des Multiplexers (MUX) . . . . .	39
5.9	Recha V2-Platine mit direkt <i>gebondetem</i> Spikey unter Plaxiglasdeckel. . . . .	39
6.1	Analogspannungsquellen: Vorgabe- und Messwerte. Die Indizes 'l' und 'r' stehen für den linken bzw. rechten Synapsenblock. . . . .	42
6.2	Vereinfachtes Schaltbild der Analogspannungsquellen . . . . .	43
6.3	Zyklus der <i>Vout</i> -Schaltung: Spannung am Knoten $V_1$ sowie Schaltzeiten, ein hoher Pegel symbolisiert einen geschlossenen Schalter . . . . .	43
6.4	Analogspannungsquellen: Ausgangsspannungen bei Anwendung einer Kalibrierungsroutine zur Verminderung der zufälligen Abweichungen . . . . .	45
6.5	Ausgänge der Analogspannungsquellen: Spannungswerte nach vollständiger Kalibrierung . . . . .	46
6.6	Vereinfachtes Schaltbild der überarbeiteten Analogspannungsquellen in Spikey 2 . . . . .	47
6.7	Wechselspannungsanteil des Ausgangs einer <i>Vout</i> -Schaltung, gut zu erkennen: <i>Refresh</i> -Zyklus und Abhängigkeit von Reihenfolge des <i>Refreshes</i> . . . . .	49
6.8	Ausgangsspannungen zweier benachbarter, unkalibrierter <i>Vout</i> -Schaltungen . . . . .	50
6.9	Ausgangsspannung einer kalibrierten <i>Vout</i> -Schaltung, einmal bzw. 100mal mit dem ADC gelesen . . . . .	51
6.10	Schaltplan der Synapsentreiber mit integrierter Erzeugung des Short Term Verhaltens. Entnommen aus [32] . . . . .	52
6.11	Generierung eines Aktionspotenzials, Ausgang eines Synapsentreibers $V_{out}$ und Ausgang einer Synapse $I_{out}$ . . . . .	53
6.12	Ausschnitt aus der HFacets-Software, der das zur Messung der <i>Short Term Plasticity</i> verwendete Setup zeigt. . . . .	55
6.13	Ergebnisse der <i>Short Term Plasticity</i> Messungen. Jeweils eine Einzelmessung ist exemplarisch in grau gezeigt, die schwarze Linie ist der Mittelwert aus 500 Einzelmessungen. <b>A</b> und <b>B</b> zeigen <i>Facilitation</i> , bei unterschiedlichem $C_2$ , <b>C</b> und <b>D</b> zeigen <i>Depression</i> . . . . .	56
A.1	Vergleich der Feuerraten der Neuronen. . . . .	iii
A.2	Vergleich der Feuerraten der Neuronen nach Kalibrierung der Schwellspannungen bzw. der Synapsengewichte. . . . .	iv
B.1	Blatt 1: Spikey und SMT-Verbinder zur Nathan-Platine. . . . .	vi
B.2	Blatt 2: Multiplexer (A), Pegelwandler (B), ADC (C), Referenzspannungsquelle (D), OP-AMP (E), LEMO-Buchsen (F) und DAC (G). . . . .	vii

B.3	Blatt 3: Spannungsregler: 5 V (A), 1,8 V digital (B), 1,8 V analog (C), Blockkondensatoren (D) und Signalbusse (E). . . . .	viii
B.4	Blatt 4: Chip-ID Selektion (A), OP-AMP (B), JTAG-Verbinder (C), LEDs (D), Anschlüsse für Takt und 12 V Spannungsversorgung (E) sowie Fire-In Leitungen (F). . .	ix
C.1	Lage 1: Spikey mit Pads zum direkten <i>Bonden</i> und zum Auflöten im Package (Mitte), meanderförmige LVDS-Leitungen links und rechts von Spikey, Stromversorgung (links), LEMO-Buchsen (oben) sowie von links nach rechts: Pegelwandler, Multiplexer, ADC, Referenzspannungsquelle und DAC (zwischen Spikey und den LEMO-Buchsen). . . . .	xi
C.2	Lage 2: Spannungsversorgung: 1,8 V digital (unten), 1,8 V analog (oben), 12 V (unten links) und Fire-In Leiterbahnen (oben links). . . . .	xii
C.3	Lage 3: Spannungsversorgung: Masse-digital (untere Hälfte), Masse-analog (obere Hälfte), 3,3 V digital (unten links) und 3,3 V analog (oben). . . . .	xiii
C.4	Lage 4: SMT-Steckverbinder zur Verbindung mit Nathan, analoge Leitungen zu den LEMO-Buchsen (oben) und jede Menge Blockkondensatoren. . . . .	xiv
D.1	Bestückungsplan der Recha V2-Platine, Oberseite. . . . .	xvii
D.2	Bestückungsplan der Recha V2-Platine, Unterseite. . . . .	xviii
E.1	Ersatzschaltbilder zum <i>Short Term Plasticity</i> Teil des Synapsentreibers. In Teil A wird der Fall mit $V_{\max} = 0 \text{ V}$ gezeigt, in Teil B der mit $V_{\max} = V_{\text{dd}}$ . . . . .	xix
E.2	Lötstempel zum Auslöten des HAGEN-Sockels. . . . .	xx

# Literaturverzeichnis

- [1] SCHEMMEL, J. ; HOHMANN, SG. ; MEIER, K. ; SCHÜRMAN, F.: A Mixed-Mode Analog Neural Network using Current-Steering Synapses. In: *Analog Integrated Circuits and Signal Processing* 38 (2004), S. 233–244
- [2] SCHÜRMAN, F. ; HOHMANN, SG. ; MEIER, K. ; SCHEMMEL, J.: Interfacing Binary Networks to Multi-valued Signals. In: KAYNAK, O. (Hrsg.) ; ALPAYDIN, E. (Hrsg.) ; OJA, E. (Hrsg.) ; XU, L. (Hrsg.): *Supplementary Proceedings of the Joint International Conference ICANN/ICONIP*, 2003, S. 430–433
- [3] HOHMANN, SG. ; SCHEMMEL, J. ; SCHÜRMAN, F. ; MEIER, K.: Predicting Protein Cellular Localization Sites with a Hardware Analog Neural Network. In: *Proceedings of the Int. Joint Conf. on Neural Networks*, IEEE Press, July 2003, S. 381–386
- [4] HOHMANN, SG. ; FIERES, J. ; MEIER, K. ; SCHEMMEL, J. ; SCHMITZ, T. ; SCHÜRMAN, F.: Training Fast Mixed-Signal Neural Networks for Data Classification. In: *Proceedings of the 2004 International Joint Conference on Neural Networks (IJCNN'04)*, IEEE Press, July 2004, S. 2647–2652
- [5] SCHÜRMAN, F. ; MEIER, K. ; SCHEMMEL, J.: Edge of Chaos Computation in Mixed-Mode VLSI - „A Hard Liquid“. In: SAUL, LK. (Hrsg.) ; WEISS, Y. (Hrsg.) ; BOTTOU, L. (Hrsg.): *Advances in Neural Information Processing Systems 17*, MIT Press, Cambridge, 2005, S. 430–433
- [6] BRÜDERLE, D.: *Implementing Spike-Based Computation on a Hardware Perceptron*. 2004. – Diplomarbeit
- [7] *FACETS: Fast Analog Computing with Emergent Transient States*. – facets.kip.uni-heidelberg.de, gefördert von der EU unter IST-2005-15879
- [8] HUNKLINGER, S. *Festkörperphysik, Skript zur Vorlesung, Universität Heidelberg, SS 04*
- [9] ALLEN, PE. ; HOLBERG, DR.: *CMOS Analog Circuit Design*. Oxford University Press, 1987
- [10] SHELLEY, M. ; MCLAUGHLIN, D. ; SHAPLEY, R. ; WIELAARD, J.: States of High Conductance in a Large-Scale Model of the Visual Cortex. In: *Journal of Computational Neuroscience* 13 (2001), June, S. 93–109
- [11] GERSTNER, W. ; KISTLER, W.: *Spiking Neuron Models*. Cambridge University Press, 2002
- [12] WHITE, P. ; BISKUP, B. ; ELZENGA, J. ; HOMANN, U. ; THIEL, G. ; WISSING, F. ; MAATHUIS, F.: Advanced patch-clamp techniques and single-channel analysis. In: *Journal of Experimental Botany* 50 (1999), June, S. 1037–1054

- [13] FERZOU, I. ; BOLEA, Sonia ; PETERSEN, C.: Visualizing the Cortical Representation of Whisker Touch: Voltage-Sensitive Dye Imaging in Freely Moving Mice. In: *Neuron* Bd. 50, 2006, S. 617–629
- [14] MARKRAM, H.: The Blue Brain Project. In: *Nature Reviews Neuroscience* 7 (2006), February, S. 153–160
- [15] SCHOENAUER, T. ; ATASOY, S. ; MEHRTASH, N. ; KLAR, H.: Simulation of a digital neurochip for spiking neural networks. In: *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks* 4 (2000), S. 490–495
- [16] SONG, S. ; MILLER, K. ; ABBOTT, L.: PCNN Neurocomputers – Event Driven and Parallel Architectures. In: *European Symposium on Artificial Neural Networks (ESANN)* (2002), April, S. 331–336
- [17] BEAR, M. ; CONNORS, B. ; PARADISO, M.: *Neuroscience: Exploring the Brain*. 3rd. Lippincott Williams & Wilkins, 2007
- [18] CAJAL, R.: Textura de la corteza visual del gato. In: *Archivos de Neurobiología* 2 (1921), S. 338–368
- [19] MAASS, W.: Networks of spiking neurons: the third generation of neural network models. In: *Neural Networks* 10 (1997), S. 1659–1671
- [20] MCCULLOCH, W. S. ; PITTS, W. H.: A Logical Calculus of the Ideas Immanent in Neural Nets. In: *Bulletin of Mathematical Biophysics* 5 (1943), S. 115–133
- [21] ROSENBLATT, F.: The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. In: *Psychological Review* 65 (1958), S. 386–408
- [22] ROSENBLATT, F.: Perceptron Simulation Experiments. In: *Proceedings of the IRE* 48 (1958), S. 301–309
- [23] ADRIAN, E.: The Impulses Produced by Sensory Nerve Endings. In: *The Journal of Physiology* 61 (1926), March, S. 49–72
- [24] THORPE, S. ; FIZE, D. ; MARLOT, C.: Speed of Processing in the Human Visual System. In: *Nature* 381 (1996), June, S. 520–522
- [25] FRÉGNAC, Y. ; MONIER, C. ; CHAVANE, F. ; BAUDOT, P. ; GRAHAM, L.: Shunting inhibition, a silent step in visual cortical computation. In: *Journal of Physiology* 97 (2003), S. 441–451
- [26] DAYAN, P. ; ABBOTT, L.: *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. The MIT Press, 2001
- [27] SONG, S. ; MILLER, K. ; ABBOTT, L.: Competitive Hebbian learning through spike-timing-dependent synaptic plasticity. In: *Nature Neuroscience* 3(9) (2000), September, S. 919–926
- [28] DESTEXHE, A. ; MAINEN, ZF.: Synthesis of Models for Excitable Membranes, Synaptic Transmission and Neuromodulation Using a Common Kinetic Formalism. In: *Journal of Computational Neuroscience* 1 (1994), S. 195–230

- [29] THOMSON, AM.: Activity-dependent properties of synaptic transmission at two classes of connections made by rat neocortical pyramidal axons in vitro. In: *Journal of Physiology* 502 (1997), S. 131–147
- [30] MARKRAM, H. ; WANG, Y. ; TSODYKS, M.: Differential signaling via the same axon of neocortical pyramidal neurons. In: *Proceedings of the national academy of science USA* 95 (1998), April, S. 5323–5328
- [31] TSODYKS, M. ; MARKRAM, H.: The neural code between neocortical pyramidal neurons depends on neurotransmitter release probability. (1997), January
- [32] SCHEMMEL, J. ; BRÜDERLE, D. ; MEIER, K. ; OSTENDORF, B.: *A Software Framework for Tuning the Dynamics of Neuromorphic Silicon Towards Biology*. 2007. – Veröffentlichung auf dem International Symposium on Circuits and Systems 2007 (ISCAS 2007)
- [33] DESTEXHE, A. ; RUDOLPH, M. ; PARÉ, D.: The high-conductance state of neocortical neurons in vivo. In: *Nature Reviews Neuroscience* 4 (2003), S. 739–751
- [34] MORTARA, A. ; VITTOZ, E.: A communication architecture tailored for analog VLSI artificial neural networks: intrinsic performance and limitations. In: *IEEE Transactions on Neural Networks* 5(3), May, S. 459–466
- [35] SCHEMMEL, J. ; GRUEBL, A. ; MEIER, K. ; MUELLER, E.: Implementing Synaptic Plasticity in a VLSI Spiking Neural Network Model. In: *Proceedings of the 2006 International Joint Conference on Neural Networks (IJCNN 2006)*, IEEE Press, 2006
- [36] GRÜBL, A.: *VLSI Implementation of a Spiking Neural Network*. May 2007. – Doktorarbeit, Universität Heidelberg
- [37] GRÜBL, A.: *Eine FPGA-basierte Plattform für neuronale Netze*. 2003. – Diplomarbeit
- [38] Xilinx, Inc.: *Virtex-II Pro Platform FPGAs: Introduction and Overview*. DS083. 2002. – [www.xilinx.com](http://www.xilinx.com)
- [39] BECKER, J.: *Ein FPGA-basiertes Testsystem für gemischt analog/digitale ASICs*. 2001. – Diplomarbeit
- [40] Maxim Integrated Products: *+3V, Quad, 12-Bit Voltage-Output DAC with Serial Interface*. MAX5253. September 1996. – [www.maxim.com](http://www.maxim.com)
- [41] National Semiconductor Corporation: *1.5A Low Dropout Positive Regulators*. LM1086. August 2001. – [www.national.com](http://www.national.com)
- [42] Analog Devices: *4-Channel, 1 MSPS, 8-/10-/12-Bit ADCs with Sequencer in 16-Lead TSSOP*. AD7924. 2000. – [www.analog.com](http://www.analog.com)
- [43] National Semiconductor: *Precision Micropower Low Dropout Voltage Reference*. LM4120. March 2005. – [www.national.com](http://www.national.com)
- [44] Analog Devices: *Low Cost, Single with Disable, 300 MHz Rail-to-Rail Amplifier*. AD8063. December 2005. – [www.analog.com](http://www.analog.com)

- [45] INSTRUMENTS, National: *Tutorial: Bandwidth, Sample Rate, and Nyquist Theorem*. February 2006. – [zone.ni.com/devzone/cda/tut/p/id/2709](http://zone.ni.com/devzone/cda/tut/p/id/2709)
- [46] National Semiconductor: *800mA Low-Dropout Linear Regulator*. LM1117. June 2005. – [www.national.com](http://www.national.com)
- [47] Maxim Integrated Products: *High-Speed, Low-Power, Single-Supply Multichannel, Video Multiplexer-Amplifiers*. MAX4312. December 2002. – [www.maxim.com](http://www.maxim.com)
- [48] Burr-Brown Products from Texas Instruments: *800mA and 1A Low Dropout Positive Regulator*. REG1117. July 2004. – [www.ti.com](http://www.ti.com)
- [49] Würth Elektronik. – [www.wuerth-elektronik.de](http://www.wuerth-elektronik.de)
- [50] PHILIPP, S.. – Doktorarbeit in Vorbereitung, Universität Heidelberg
- [51] FIERES, J. ; GRÜBL, A. ; PHILIPP, S. ; MEIER, K. ; SCHEMMEL, J. ; SCHÜRMAN, F.: Platform for Parallel Operation of VLSI Neural Networks. In: *Proceedings of the Brain Inspired Cognitive Systems (BICS2004)* NC4.3 (2004), S. 1–7
- [52] *Token Ring*. – IEEE Spezifikation 802.5
- [53] BRÜDERLE, D. ; GRÜBL, A. ; MEIER, K. ; MUELLER, E. ; SCHEMMEL, J.: *Modeling Synaptic Plasticity within Networks of Highly Accelerated I&F Neurons*. 2007. – Veröffentlichung auf der International Work-Conference on Artificial Neural Networks 2007 (IWANN2007)
- [54] HOHMANN, S.: *Stepwise Evolutionary Training Strategies for Hardware Neural Networks*. 2005. – Doktorarbeit, Universität Heidelberg
- [55] DESTEXHE, A. ; PARÉ, D.: Impact of Network Activity on the Integrative Properties of Neocortical Pyramidal Neurons In Vivo. In: *Journal of Physiology* 81(4) (1999), April, S. 1531–1547

# Danksagung

Ich möchte mich bei allen bedanken, die zum Gelingen dieser Arbeit beigetragen haben. Insbesondere sind dies:

Herrn Prof. Dr. Karlheinz Meier für die freundliche Aufnahme in die Arbeitsgruppe und die Ermöglichung dieser Diplomarbeit.

Herrn Prof. Dr. Norbert Herrmann für die Übernahme der Zweitkorrektur.

Dr. Johannes Schemmel für die umfassende Betreuung, stets kompetente Hilfe und konstruktive Kritik.

Andi Grübl für die umfassende Betreuung und Geduld bei allen aufgetretenen Problemen. Insbesondere für die Bereitschaft auch während des Verfassens seiner Doktorarbeit noch mit Rat und Tat dabei zu sein.

Daniel Brüderle für diverse Tips und Tricks beim Programmieren, für das Korrekturlesen des Manuskripts sowie auch für das gemeinsame Bezwingen widerspenstiger Hard- und Software.

Tillmann Schmitz und Andre Srowig für konstruktive Vorschläge und ermunternde Gespräche.

Allen Mitgliedern der Vision(s) Gruppe für die Hilfsbereitschaft und stets angenehme und freundschaftliche Arbeitsatmosphäre.

Meinen Eltern für die Ermöglichung meines Studiums und ihre Unterstützung.

Mareike Schwarze für unsere gemeinsame Zeit, insbesondere die letzten zwei Wochen.

## Erklärung:

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

Heidelberg, den 01.05.2007

.....  
(Unterschrift)