UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft

FAKULTÄT FÜR
ELEKTROTECHNIK,
INFORMATIK UND
MATHEMATIK

Ressourceneffiziente Realisierung Pulscodierter Neuronaler Netze

Zur Erlangung des akademischen Grades

DOKTORINGENIEUR (Dr.-Ing.)

der Fakultät für Elektrotechnik, Informatik und Mathematik
der Universität Paderborn
vorgelegte Dissertation
von

M. Sc. Tim Kaulmann
Paderborn

Referent:	Prof. Dr.-Ing. U. Rückert
Korreferent:	Prof. Dr.-Ing. habil. R. Schüffny

Tag der mündlichen Prüfung: 05.10.2009

Paderborn, den 28.10.2009

Diss. EIM-E/256

Inhaltsverzeichnis

Einleitung **1**

1 Biologische Grundlagen neuronaler Netze **3**

1.1 Anatomie des menschlichen Gehirns 3

1.2 Physiologische Grundlagen der Zelle 5

1.2.1 Zelle und Zellkern 6

1.2.2 Zellmembran und Ionenkanäle 6

1.2.3 Energieumsatz in Nervenzellen, Natrium-Kalium-Pumpe 7

1.2.4 Energiegewinnung in der Zelle 9

1.2.5 Zelltypen: Purkinje Zelle und Körnerzelle 10

1.2.6 Betrachtung der Membrankapazität biologischer Neuronen 11

1.2.7 Membranpotential 13

1.3 Aktionspotential 14

1.4 Reizweiterleitung 16

1.5 Diskussion 17

2 Stand der Technik pulsender Neurone **19**

2.1 Technische Darstellungen von Neuronen 21

2.1.1 Spike Response Modell 21

2.1.2 Leaky Integrate and Fire Modell 22

2.2 Digitale Implementierungen 23

2.2.1 Die Schrauwen-Implementierung 24

2.2.2 Die Upegui-Implementierung 26

2.2.3 Die Torres-Huitzil-Implementierung 28

2.2.4 Die Johnston-Implementierung 30

2.2.5 Die Maya-Implementierung 31
2.2.6 Die Godin-Implementierung 31
2.3 Analoge Implementierungen 33
2.3.1 Die Matolin-Implementierung 34
2.3.2 Die van Schaik-Implementierung 34
2.3.3 Die Indiveri-Implementierung 34
2.3.4 Die Chicca-Implementierung 36
2.3.5 Die Wijekoon-Implementierung 37
2.4 Vergleich bestehender Implementierungsvarianten 38

3 Energetische Modellierung pulscodierter neuronaler Netze 41
3.1 Modellierung des Energieumsatzes: Biophysikalisches Grundmodell 42
3.1.1 Kanalströme - passiver Transport 42
3.1.2 Linearisiertes System 44
3.1.3 Pumpströme - aktiver Transport 47
3.1.4 Betrachtung der Natrium-Kalium-Pumpe als regelungstechnisches Problem 48
3.1.5 Stabilitätsprüfung des nichtlinearen, geregelten Systems mittels Ljapunov-Verfahren 51
3.1.6 Erweiterung des Grundmodells zu einem Modell für Synapse und Dendrit 52
3.1.7 Modellierung eines Aktionspotentials 56
3.1.8 Simulation eines Minimalsystems 58
3.2 Modellierung des Energieumsatzes mit elektrischen Schaltkreisen 62
3.2.1 Abschätzungen zum Energiebedarf biologischer Neurone am Ersatzschaltbild im steady-state 62
3.3 Modellierung eines LIAF Neurons mit elektrischen Ersatzschaltkreisen . . 63
3.3.1 Passives Entladen der Kapazität über Leckströme 65
3.3.2 Aktives Laden der Kapazität unter Berücksichtigung von Leckströmen 65
3.4 Verlustleistung 66
3.4.1 Gleichstrombetrieb 66
3.4.2 Herleitung der Übertragungskennlinie 68
3.5 Erweiterung der Betrachtung am LIAF Modell zum SRM 72
3.6 Diskussion 74

4 Ressourcenbedarf pulscodierter neuronaler Netze **75**
4.1 Analoge Implementierungen 76
4.1.1 Leaky Integrate and Fire Neuron 77
4.1.2 Statische Synapse 84
4.1.3 Ermittlung der äquivalenten Wortbreite 88
4.2 Digitale Implementierungen 91
4.2.1 Bitserielle Multiplikation 93
4.2.2 Digitale ultra-low-power Standardzellenbibliothek 95
4.2.3 Leaky Integrate and Fire Neuron 102
4.2.4 Vergleich digitaler Implementierungsvarianten 116
4.3 Analoger Testchip 117

5 Struktur und Funktion in pulscodierten neuronalen Netzen **125**
5.1 Fehlertoleranz neuronaler Assoziativspeicher 125
5.1.1 Struktur 126
5.1.2 Fehlertoleranz binärer neuronaler Assoziativspeicher 128
5.2 Einfluss der Pulscodierung auf die Funktion 141

Zusammenfassung **150**

A Mathematischer Anhang **155**
A.1 Herleitung von $u_{c,f}^{(N)}$ und $u_{c,T}^{(N)}$ 155
A.2 Herleitung des maximalen Gewichts zum fehlerfreien Abruf 157
A.3 Variation des Störabstandes in einer 90 nm ULP Standardzellenbibliothek 158

B Skalierungsregeln **161**

C Simulink Modelle **163**

Verzeichnis der verwendeten Abkürzungen und Formelzeichen **169**

Abbildungsverzeichnis **175**

Tabellenverzeichnis **181**

Literaturverzeichnis **183**

Eigene Publikationen **191**

Index **193**

Einleitung

Schon immer diente Wissenschaftlern die Natur mit ihren faszinierenden Lösungen als Vorbild für eigene technische Entwicklungen. Hierbei ziehen insbesondere Gehirne von Lebewesen aufgrund ihrer enormen Leistungsfähigkeit bei gleichzeitiger Robustheit die Aufmerksamkeit auf sich und liefern den Ansporn für die Entwicklung von technischen Systemen mit ähnlichen Eigenschaften. Um dieses Ziel zu erreichen, werden bestimmte Arbeitsprinzipien biologischer Gehirne mit sogenannten künstlichen neuronalen Netzen nachgeahmt. Dies geschieht zum einen in Software. Zum anderen kann in der Hardwareumsetzung insbesondere die moderne Mikro- und Nanoelektronik bezüglich der Robustheit der Schaltungen profitieren und zuverlässige integrierte Schaltkreise ermöglichen.

Mit der zunehmenden Verkleinerung von Strukturen in integrierten Schaltkreisen in den Bereich der Nanotechnologie bei gleichzeitiger Erhöhung der Anzahl von Bauelementen entstehen neue Herausforderungen beim Entwurf und dem späteren Betrieb von Mikrochips. Zum Einen nimmt die Wahrscheinlichkeit zu, dass Teile des Mikrochips durch Toleranzen bei der Herstellung nicht oder nur eingeschränkt funktionieren. Zum Anderen können schon geringe äußere Einflüsse im Betrieb des Mikrochips für Störungen seiner Funktion sorgen. Daher werden in aktuellen Forschungsvorhaben Umsetzungen integrierter Schaltkreise unter Nutzung von neuronalen Prinzipien untersucht, von denen man sich eine höhere Robustheit der integrierten Schaltung bei einer gleichzeitigen Verringerung des Energiebedarfs verspricht [107].

Besonders die Frage, wie sich Komponenten künstlicher neuronaler Netze ressourceneffizient, d. h. mit minimalem Energiebedarf und minimaler Chip-Fläche umsetzen lassen, ist für die Akzeptanz der Anwendung neuronaler Prinzipien in mikroelektronischen Schaltungen von größter Bedeutung. Dazu müssen neben der theoretischen Betrachtung elektrischer Modelle von Neuronen auch Schaltungen in aktuellen Halbleitertechnologien entworfen werden, mit denen die Funktion der Schaltungen und der Energiebedarf der einzelnen Komponenten ermittelt werden können.

Viele mikroelektronische Umsetzungen künstlicher neuronaler Netze in analoger Schaltungstechnik werden aufgrund der vorteilhaften elektrischen Eigenschaften oft in Technologien mit relativ großen Strukturgrößen von 350 nm umgesetzt. Die Eigenschaften der betrachteten Schaltungen und ihre Umsetzbarkeit in Strukturgrößen von 130 nm und darunter ist weitestgehend noch nicht betrachtet worden. Daher werden in dieser Arbeit Lösungen für die Höchstintegration neuronaler Netze für Technologien mit Strukturgrößen

von 130 nm und kleiner erarbeitet, da sich damit Netze bisher unerreichter Komplexität realisieren lassen. In diesem Bereich müssen schaltungstechnische Lösungen geschaffen werden, die den besonderen physikalischen und elektrischen Eigenschaften modernster Halbleiterprozesse gerecht werden.

Im Allgemeinen gelten der Standardzellenprozess und die digitale Schaltungstechnik in Technologien mit Strukturgrößen von 130 nm bis hinunter in den Bereich von 22 nm als beherrschbar, auch wenn für Strukturen von 90 nm und darunter zusätzliche Anforderungen an die Herstellbarkeit der Schaltungen gestellt werden. Diese Bedingungen können allerdings durch Werkzeugunterstützung und erweiterte Layoutregeln sowie zusätzliche Prüfschritte beim Schaltungsentwurf nach heutiger Einschätzung erfüllt werden. Aufgrund der neu eingeführten Arbeitsschritte in der Entwurfs- und Verifikationsphase gilt die digitale Schaltungstechnik auch bis weit unterhalb der Strukturgrößen von 130 nm als in hohem Maße zuverlässig. Daher sollen in dieser Arbeit auch die Möglichkeit der Umsetzung von Neuronen in digitaler Schaltungstechnik bewertet, und ressourceneffiziente Lösungen zur Implementierung von Neuronen aufgezeigt werden, welche die analogen Funktionen emulieren können.

Eine interessante Frage, die hier beantwortet werden soll ist, ob die digitale Umsetzung der Neuronen in kleineren CMOS-Technologien eine kleinere Fläche belegen wird, als ihr analoges Pendant, und wie sich die für die Funktion benötigte Energie in Bezug auf die benötigte Energie der analogen Variante verhält. Im Hinblick auf die von digitalen Systemen umgesetzte Verlustleistung sollen in dieser Arbeit aus anderen Bereichen bewährte Konzepte geprüft und neue Konzepte der schaltungstechnischen Umsetzung geschaffen und genutzt werden. Besonders das Einsparpotential für die umgesetzte Leistung bei der Nutzung von Standardzellenbibliotheken mit Elementen, die im Subschwellenbereich (engl. Subthreshold oder Sub-V_{TH}) arbeiten, soll an den digitalen Implementierungen der Neurone ermittelt werden. Während die Subschwellen-Schaltungstechnik im Bereich der analogen Schaltungen seit Jahren bekannt, wenn auch wenig genutzt ist, entwickelt sich dieser Arbeitsbereich zu einem aufstrebenden Forschungsgebiet im Bereich der digitalen Standardzellen.

Kapitel 1

Biologische Grundlagen neuronaler Netze

1.1 Anatomie des menschlichen Gehirns

Das menschliche Gehirn lässt sich in sechs Teile unterteilen [93]: Das Groß- oder Endhirn, das Kleinhirn, Zwischenhirn, Mittelhirn sowie die Brücke und daran anschließend das verlängerte Mark. Die letzteren drei Teile werden zusammengefasst als Hirnstamm bezeichnet. Während das Kleinhirn insbesondere für die motorischen Aufgaben zur Haltung und Bewegung sowie der Blickmotorik zuständig ist, hat das Großhirn weitreichende kognitive Fähigkeiten entwickelt. Die multifunktionale Veranlagung lässt sich auch durch seine komplexe Struktur erahnen. So ist jede Gehirnhälfte in vier Lappen zu unterteilen, den Frontallappen, Parietallappen, Temporallappen und Okzipitallappen. Daneben existieren zwei Bereiche, welche keinem dieser Lappen zugeordnet werden können und im Weiteren vernachlässigt werden. Das Großhirn beherbergt unter anderem das Kurzzeit- und Langzeitgedächtnis. Während das Kurzzeitgedächtnis nach heutigem Kenntnisstand eine Leistung des präfrontalen Kortex zu sein scheint, erfolgt die Langzeitspeicherung von Informationen unter Einbeziehung der Hörrinde, Sehrinde und motorischen Rinde in der gesamten Großhirnrinde.

Einer der am besten untersuchten Bereiche der Großhirnrinde ist das visuelle System der primären Sehrinde. In ihr werden die visuellen Informationen des Auges abgebildet und in höheren Schichten eine Auswertung der aufgenommenen Szenen vorgenommen. Bereits in frühen Arbeiten wurden grundlegende Funktionen des visuellen Systems erklärt, allerdings nicht durch den Ansatz der direkten Beobachtung einzelner Zellen in vivo, sondern durch den Abgleich von Antwortmustern technischer Filter mit den Antwortmustern größerer visueller Areale im Gehirn von Affen [53] oder visuellen Arealen im Gehirn von Katzen [58]. Die Struktur des Gehirns, insbesondere die Verbindungen zwischen Neuronen komplexerer Areale können erst heute langsam neue bildgebende Verfahren und Rekonstruktionsalgorithmen ermittelt werden [7, 37], so dass eine genaue Verifizierung der Ergebnisse früherer Veröffentlichungen möglich ist.

Während also die Struktur der neuronalen Netze im menschlichen Gehirn noch immer Gegenstand aktueller Forschung ist und Teile der Funktion begrenzter Areale verstanden werden, ist die Beschreibung der grundlegenden Zellen, die die kognitive Leistung erst ermöglichen, sehr weit fortgeschritten. Insbesondere die Struktur der verschiedenen Neurone, ihre einzelne Funktion und die elektrochemischen Eigenschaften sowie die Eigenschaften der Zellmembran wurden in vielen Veröffentlichungen behandelt und mit mathematischen Modellen nachgebildet. Bereits 1949 beschrieben Hodgkin und Huxley das Verhalten eines Neurons am Riesenaxon eines Tintenfischs ausführlich und erstellten ein Neuronenmodell auf Grundlage elektrischer Schaltkreise, das bis heute oft als Basis vieler weiterführender Arbeiten genutzt wird [48–52]. Neben den biologisch motivierten Modellen wurden früh vereinfachte mathematische Modelle für Neurone entworfen, welche eine Simulation größerer neuronaler Netze mit Software-Simulatoren auf Computern oder unter Einsatz spezialisierter Hardware erlauben. Die wichtigsten Modelle sollen in den Abschnitten zum Stand der Technik (siehe Kap. 2) aufgegriffen und erläutert werden.

Neben den rein mathematischen oder elektrischen Modellen entstanden mit der immer besseren Verfügbarkeit von Halbleitertechnologien und Entwurfswerkzeugen für Forschungseinrichtungen technische Realisierungen von Neuronen und Synapsen, um massiv parallele Systeme neuronaler Netze in anwendungsspezifischen integrierten Schaltkreisen (engl. Application Specific Integrated Circuis, ASIC) aufzubauen. Heute existieren technische Neurone, welche zum Teil die mathematischen Modelle exakt nachbilden, zum anderen Teil starke Vereinfachungen zur Beschleunigung der Schaltung vornehmen, aber die grundlegenden phänomenologischen Eigenschaften der biologischen Neurone nachahmen. Beide Varianten haben ihre Berechtigung wenn es um den Einsatz in spezialisierter Hardware geht. Die einfacheren technischen Modelle haben vor allem dann einen Vorteil, wenn große neuronale Netze und die sich in ihnen ausbildende Dynamik schnell simuliert werden sollen, wobei die Eigenschaften der komplexen Modelle nur eine untergeordnete Rolle spielen und durch die Vereinfachungen verloren gehen. Zum Zeitpunkt dieser Arbeit gibt es noch immer keine Klarheit darüber, welcher Anteil der Kommunikation zwischen Neuronen zur Informationsverarbeitung beiträgt [33]. Daher werden in System-Simulationen viele verschiedene Kommunikationsarten geprüft, z. B. Puls-Triplets, geordnete Abfolgen von Pulsen, und das *time-to-first-spike* Prinzip, also die zeitliche Korrelation zweier Pulse zueinander.

Ein Aspekt, der in den publizierten Neuronenmodellen bislang praktisch nicht zu finden ist, ist die Betrachtung des Energiebedarfs biologischer Neurone. Zwar existieren Publikationen, welche sich mit dem Umsatz von Energie in Nervenzellen grundsätzlich beschäftigen [8, 67], jedoch zielen diese eher auf Beschreibungen der chemischen Vorgänge ab, als auf die Abschätzung des Energieumsatzes bei der Informationsverarbeitung. Eine Übertragung der Ergebnisse auf den Energieumsatz von technischen Neuronenmodellen ist daher schwierig. Dass das Thema des Energieumsatzes bei der Betrachtung von Neuronenmodellen nicht vernachlässigt werden sollte wird durch den Hinweis auf die Größe des Energieumsatzes des menschlichen Gehirns in der aktuellen Literatur verständlich. Ein großer Teil der im menschlichen Körper umgesetzten Energie, die mit ungefähr 100 W angegeben wird, wird

für das Gehirn aufgewandt. Die Leistungsaufnahme des menschlichen Gehirns mit seinen ca. 10^{11} Neuronen wird im Mittel mit 20 W Freizeitumsatz [86] angegeben. Schmidt und Thews definieren bei ihrer Betrachtung des Energieumsatzes den Freizeitumsatz wie folgt:

> „Der „Freizeitumsatz“ (Energieumsatz eines nicht körperlich arbeitenden Menschen bei einer mehr kontemplativen Freizeitgestaltung) entspricht dem täglichen Gesamtumsatz weiter Bevölkerungskreise, die als „Schreibtischarbeiter“ und „Datenverwalter“ keine energetisch maßgeblichen körperlichen Aktivitäten entfalten.“

Diese Arbeit greift zur Abschätzung des Energieumsatzes biologischen Neurone das in [69] entworfene biophysikalische Neuronenmodell auf und modifiziert die Sicht auf den Ort des zentralen zellulären Energieumsatzes, der Natrium-Kalium Pumpe, zu einer regelungstechnischen Sicht. Das hier gewonnene biophysikalische Neuronenmodell erlaubt die Abschätzung des Energieumsatzes einzelner Neurone in Abhängigkeit von ihrer Erregung und der Informationsverarbeitung. Daneben beschäftigt sich diese Arbeit in der Hauptsache mit der ressourceneffizienten Umsetzung von Neuronenmodellen und dem Vergleich des Energiebedarfs sowie des Flächenbedarfs biologischer Nervenzellen und ihren technischen Umsetzungen in feldprogrammierbaren Gatteranordnungen (engl. field programmable gate array, FPGA) sowie anwendungsspezifischen integrierten Schaltkreisen (engl. application specific integrated circuit, ASIC) mit aktuellen Halbleitertechnologien mit Strukturgrößen von 130 nm und darunter.

Die Areale des menschlichen Gehirns bestehen aus verschiedenen Arten von Neuronen, den Purkinje-Zellen, Astrozyten, Körnerzellen (Granularzellen) und Stützgewebe, wozu vor allem die Gliazellen zählen. Alle aufgeführten Neuronenarten haben verschiedene Ausprägungen ihrer Form, der Länge und dem Vorhandensein eines Axons und der Anzahl der von ihnen eingegangenen Verbindungen mit weiteren Nervenzellen. Die Aufzählung der Neurone in diesem Abschnitt kann keinen Anspruch auf Vollständigkeit erheben, da die vollständige Beschreibung einerseits nicht zum Verständnis oder zur Entwicklung der hier gezeigten Modelle beiträgt und andererseits viele der gezeigten Parameter der komplexen Modelle ständigen Änderungen durch neue Erkenntnisse der neurobiologischen Forschung unterliegen.

1.2 Physiologische Grundlagen der Zelle

In diesem Abschnitt wird der grundlegende Aufbau einer Warmblüterzelle beschrieben. Dabei teilt sich die Betrachtung in die Energieträger erzeugenden Bestandteile und die Energie umsetzenden Bestandteile der Zelle auf. Obwohl sich Nervenzellen und Muskelzellen auf den ersten Blick in ihrer Struktur nicht unterscheiden, so existieren gerade bei den Nervenzellen hochspezialisierte Kanäle in der Zellmembran, die nur für eine eingeschränkte Gruppe von Nervenzellen gültig sind und zur elektrischen Funktion der Zellen beitragen.

Während zu Beginn dieses Abschnitts daher allgemein auf Zellen und ihren Aufbau eingegangen wird, werden im Laufe der Arbeit nur noch die speziellen Eigenschaften von Nervenzellen betrachtet, auch wenn im Text von Zellen statt von Nervenzellen gesprochen wird.

1.2.1 Zelle und Zellkern

Die Zellen von Warmblütern sind in sich abgeschlossene Räume, die von einer Membran aus Lipiden (Fetten) umgeben wird. In ihrem Inneren beherbergt eine Zelle das glatte sowie das raue endoplasmatische Retikulum (ER), den Golgi-Apparat und die mit eigenen Membranen vom Zellinneren abgeschlossenen Mitochondrien sowie den Zellkern. Das glatte ER spielt je nach Zelltyp eine Rolle bei Kohlenhydratstoffwechsel (z. B. in Leberzellen) und der Erzeugung von Fettsäuren und Steroiden. Dagegen sorgt das raue ER für die Synthese verschiedener Proteine [17]. Als weiterer Bestandteil der Zelle spielt das Mitochondrium eine besondere Rolle bei der Zellatmung. Es synthetisiert u. a. aus der Oxydation von Brenztraubensäure im Citrat-Zyklus den universellen Energieträger Adenosintriphosphat (ATP) und liefert der Atmungskette notwendige Zwischenprodukte zur ATP-Synthese.

1.2.2 Zellmembran und Ionenkanäle

Die Zellmembran aller Zellen besteht aus Lipiden (Fetten), Proteinen und Kohlehydraten [17]. Ihre Struktur wird durch die Lipide, insbesondere die Phospholipide erst ermöglicht. Phospholipide bestehen aus zwei Teilen, dem hydrophilen (wasserliebenden) Kopf und einem hydrophoben (wasserabweisenden) Rest. In wässriger Lösung besteht daher das Bestreben, nur den hydrophilen Kopf mit Wasser in Kontakt zu bringen, weshalb sich auf Wasseroberflächen eine künstliche Phospholipid-Einzelschicht erzeugen lässt [11, 28]. Sind die Lipide jedoch vollständig von Wasser umgeben, kann sich aus zwei Einzelschichten eine Phospholipid-Doppelschicht wie in Abb. 1.1 gezeigt, von 4 bis 5 nm Dicke [86] ausbilden. Diese bildet die Grundlage der heute verwendeten Modelle der Zellmembran, dem Flüssig-Mosaik-Modell [90], in dem die ebenfalls nachgewiesenen Proteine, wie z. B. die spannungsgesteuerten Ionenkanäle, in eine flüssige Doppelschicht aus Phospholipiden eingebettet sind, und dem dynamisch strukturierte Mosaikmodell [98], welches das Flüssig-Mosaik-Modell um eine Dynamik der Proteine innerhalb der flüssigen Lipiddoppelschicht erweitert.

Die Zellmembran ist besonders für ungeladene, polare Moleküle (auch Wasser) durchlässig, wogegen sie geladene Moleküle und Ionen nur schwer passieren lässt. Dennoch finden an der Zellmembran Ausgleichs- und Transportvorgänge von Ionen statt. Der passive Transport von Ionen durch die Zellmembran ist der erste Transportmechanismus. Dabei diffundieren Ionen entlang ihres Konzentrationsgradienten vom extrazellulären Raum durch die Lipiddoppelschicht in den intrazellulären Raum hinein oder vom intrazellulären Raum in den extrazellulären Raum heraus. Dieser Vorgang hält so lange an, bis auf

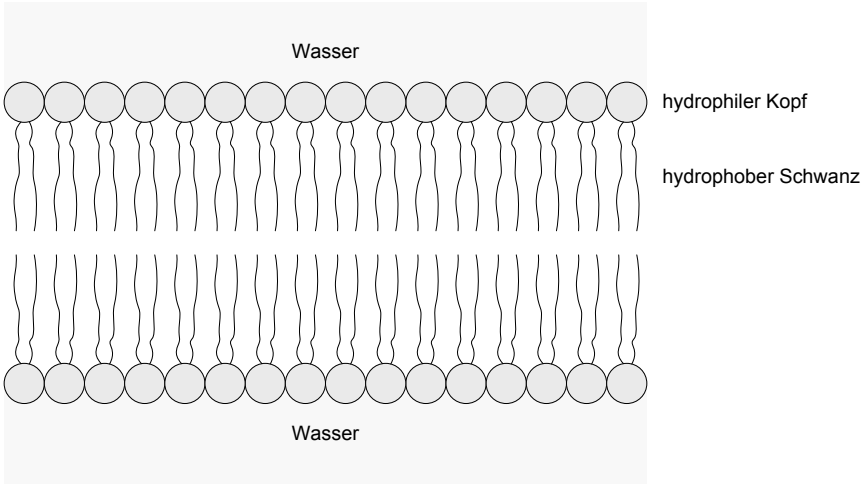


Abbildung 1.1: Schematische Darstellung einer ausgebildeten Zellmembran (nach [17]).

beiden Seiten der Membran die gleiche Konzentration vorliegt. Dem Diffusionsvorgang wirkt eine elektrische Kraft entgegen, die durch die Ladungstrennung an der Membran bei der Diffusion entsteht. Es stellt sich ein Gleichgewicht ein, bei dem die Diffusionskraft gleich der elektromotorischen Kraft der Teilchen im elektrischen Feld ist. Das elektrische Feld über der Zellmembran wird durch die Potentialdifferenz zwischen intrazellulärem Raum und extrazellulärem Raum hervorgerufen. Das Potential der extrazellulären Seite der Zellmembran in Bezug auf das Potential im Zellinneren wird als Membranpotential bezeichnet und stellt eine wesentliche Größe bei der Beschreibung von Funktionen an Zellen dar.

Einen weiteren passiven Transportmechanismus für Ionen stellen gesteuerte Kanäle dar, die in die Zellmembran eingelassen sind und welche spezifisch Ionen durch die Membran passieren lassen können. Man unterscheidet spannungsgesteuerte Ionenkanäle und transmittergesteuerte Ionenkanäle. Insbesondere spannungsgesteuerte Natrium- und Kaliumkanäle werden für das Auslösen eines Aktionspotentials (vgl. Kapitel 3.1.7) verantwortlich gemacht. Die Ionenkanäle stellen einen erleichterten Transportweg für den Austausch von Ionen durch die Zellmembran dar.

1.2.3 Energieumsatz in Nervenzellen, Natrium-Kalium-Pumpe

Der ständige Einstrom von Natriumionen in das Zellinnere und der Ausstrom von Kaliumionen aus dem Zellinneren führen zu einer Veränderung des Zellvolumens. Damit das Zellvolumen konstant bleibt und ein osmotisches Gleichgewicht hergestellt wird, wird der Ioneneinstrom und -ausstrom durch einen aktiven Transportmechanismus ausgeglichen [6]. Die Natrium-Kalium-Pumpe, auch NaK-ATPase genannt, befördert in einem Transportzyklus drei Natriumionen entgegen ihrem elektrochemischen Gradienten aus dem Zellinneren auf die extrazelluläre Seite und im Gegenzug zwei Kaliumionen von der extrazellulären Seite in das Zellinnere hinein. Dabei wird pro Pumpzyklus die Energie des

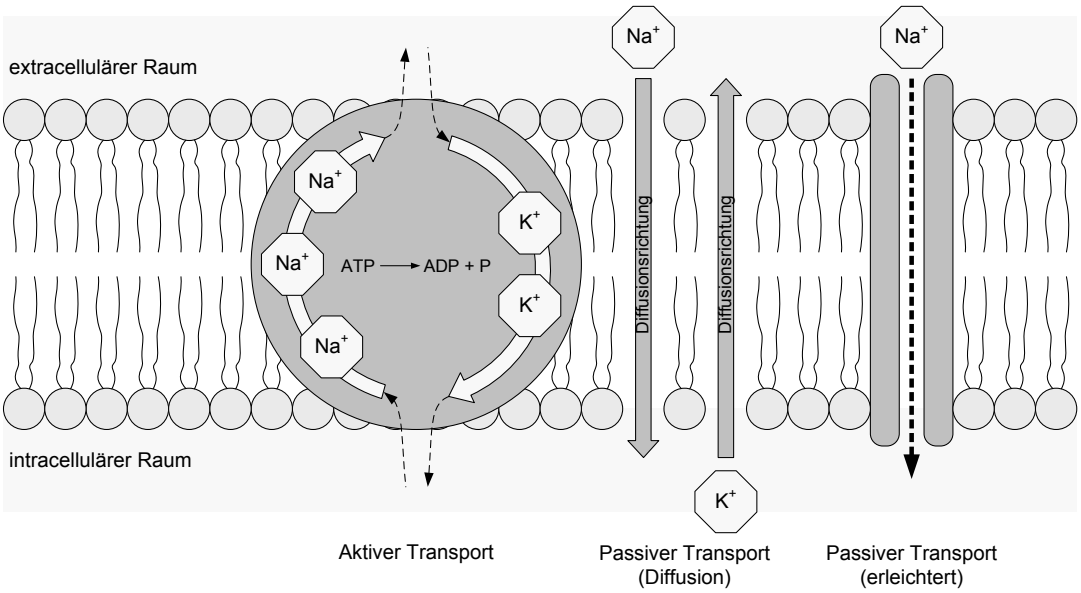


Abbildung 1.2: Zellmembran mit aktiven und passiven Transportmechanismen für die wichtigsten beteiligten Ionen.

Aufspaltens eines Moleküls Adenosintriphosphat (ATP) zu Adenosindiphosphat (ADP) und einem Phosphatrest für den Transport genutzt.

$$\mathrm{ATP} \xrightarrow{-46\,\mathrm{kJ/mol}} \mathrm{ADP} + \mathrm{P_i} \tag{1.1}$$

Die Natrium-Kalium-Pumpe sorgt so unter Energieeinsatz für die Aufrechterhaltung des Zellvolumens und des Membranpotentials. Die Natrium-Kalium-Pumpe ist mit einem Anteil von 30% am Gesamtumsatz als größter an der Informationsverarbeitung beteiligter Energieumsetzer der Zelle anzusehen [5] und wird daher in Kapitel 3 als Regler näher betrachtet. Für den Erhalt des Ruhepotentials wird in [8] ein Wert von $3,42 \cdot 10^8$ ATP-Molekülen angegeben, die pro Sekunde ausgeglichen werden müssen. Dabei werden die durch Diffusion in die Zelle einströmenden und ausströmenden Ionen ausgeglichen, um das Zellvolumen konstant zu halten. Der angegebene Wert entspricht einer Ruheleistung von ca. 26 pW. Das entspricht bei einer angenommenen Zahl von 10^{10} Neuronen, wenn man den Wert für den Freizeitumsatz von Schmidt und Thews [86] als ersten Anhaltspunkt für den Energiebedarf nimmt, kaum mehr als 1% des Gesamtumsatzes. Weitaus bedeutender für den Energieumsatz sind die aktiven Vorgänge, auf die im Verlauf der Arbeit eingegangen wird. Ein einfacher Querschnitt der Zellmembran mit den wichtigsten aktiven und passiven Transportmechanismen für Natrium- und Kaliumionen ist in Abb. 1.2 dargestellt.

Die Energie für den aktiven Pumpvorgang liefert die Hydrolyse des ATP-Moleküls, das in Abb. 1.3 abgebildet ist. Das ATP-Molekül besteht aus einer Pentose, genauer der Ribose als Zentralelement und einem Adenin-Rest auf der linken Seite dar Darstellung. Die drei angehängten Phosphatreste auf der rechten Seite der Darstellung bilden damit das Adenosintriphosphat. Das ATP kann durch Hydrolyse, d. h. Abspaltung eines Phosphatrests

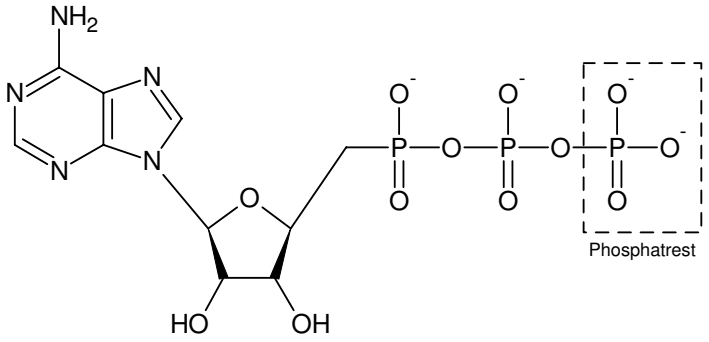


Abbildung 1.3: Molekül Adenosintriphosphat.

unter Aufnahme von Wasser zu Adenosindiphosphat (ADP) und Adenosinmonophosphat (AMP) abgebaut werden, welche den Energieträger gewinnenden Mechanismen wieder zur Verfügung gestellt werden.

Neben dem Na^+- und K^+-Transport laufen durch die Konzentrationsgradienten aller im flüssigen Medium anwesenden Ionen weitere Transportvorgänge ab. Der Verlust potentieller Energie durch den Einstrom von 3 Na^+-Ionen wird genutzt, um 1 Ca^{2+}-Ion aus der Zelle zu transportieren (Ca^{2+}-Na^+-Antiport). In gleicher Weise wird mit dem Einstrom eines Na^+-Ions ein Glucose-Molekül zur Energiegewinnung durch die Glycolyse in die Zelle transportiert (Glucose-Na^+-Symport).

1.2.4 Energiegewinnung in der Zelle

Um die im Folgenden beschriebenen Vorgänge einzuleiten, betrachten wir zunächst die Gewinnung der universellen zellulären Energieträger Adenosintriphosphat (ATP) und Nicotinamid-Adenin-Dinucleotid (NAD^+, oxidierte Form) bzw. seine reduzierte Form NADH. Während das ATP nur beim später betrachteten Energieumsatz an der Zellmembran eine Rolle spielen wird, sind beide Energieträger an der Erzeugung von ATP beteiligt.

Dabei wirken intrazellulär drei Mechanismen:

- Glycolyse: Oxidation von Glucose zu Pyruvat unter Freisetzung von ATP.
- Citratzyklus: Zyklische Umformung von Acetyl-CoA, einem enzymatischen Umwandlungsprodukt aus Pyruvat unter Freisetzung von ATP.
- Atmungskette und oxydative Phosphorylierung: An der inneren Mitochondrienmembran wird durch die ATP-Synthase aus ADP und einem Phosphatrest unter Einstrom von H^+-Ionen ATP synthetisiert. Die dort genutzten H^+-Ionen werden unter Oxidation von NADH zu $NAD^+ + H^+$ gegen ihren Konzentrationsgradienten durch die Membran bewegt und stehen für die Diffusion bei der ATP-Synthase zur Verfügung. Das genutzte NADH liefern Glycolyse und Citratzyklus. In der Atmungskette entsteht der Hauptanteil des zellulären Energieträgers ATP.

Abbildung 1.4: Purkinje Zelle nach [38]. a) Axon b) Kollaterale (Verzweigung des Axons im Zielgebiet) c) und d) Dendritenäste.

1.2.5 Zelltypen: Purkinje Zelle und Körnerzelle

Zu den größten Nervenzellen im menschlichen Gehirn gehören die Purkinje-Zellen im Cerebellum. Diese als pyramidenförmig oder birnenförmig beschriebene Zelle liegt in der mittleren Schicht der Kleinhirnrinde. Ihr Axon verlässt als einzige efferente Faser die Kleinhirnrinde [4]. In Abb. 1.4 ist die Darstellung einer Purkinje-Zelle nach Cajal [38] abgebildet. Diese Nervenzelle zeichnet sich durch ihre sehr große Anzahl afferenter (hinführender), dichter dendritischer Verbindungen aus und weist alle weiteren typischen Bestandteile einer Nervenzelle auf. So ist der große Zellkörper in der Mitte der Abbildung mit einem fortführenden langen Axon und einer Verzweigung des Axons, der sog. Kollaterale, zu sehen. Die Kollaterale ist eine Verzweigung des Axons in der Nähe eines empfangenden Neurons und verbindet sich über die Synapsen mit den Dendriten des empfangenden Axons.

Der Durchmesser des Zellkörpers der Purkinje-Zelle beträgt etwa $80\,\mu$m. Bei einer einfachen Modellvorstellung, dass der Zellkörper eine Kugel sei, ergibt sich so eine Zelloberfläche von $4\pi r^2 \approx 20106\,\mu\text{m}^2$. Drückt man die Zelle nun in der Modellvorstellung platt und ignoriert das mit weiteren Komponenten gefüllte Volumen, so lässt sich die Zelle auf eine Fläche von ca. $142\,\mu\text{m} \times 142\,\mu\text{m}$ bringen.

Die unter der Purkinje-Zellschicht (lat. Stratum purkinjense) liegende Schicht der Klein-

hirnrinde zeichnet sich durch eine große Anzahl besonders kleiner Zellen, den Körnerzellen oder Granularzellen (engl. granule cells), aus. Diese Schicht wird als Körnerschicht (lat. Stratum granulosum) bezeichnet und beinhaltet überwiegend kleine Körnerzellen mit einem Durchmesser von $4\,\mu$m bis $10\,\mu$m sowie große Körnerzellen mit einem Durchmesser von $11\,\mu$m bis $18\,\mu$m. Die Körnerzellen besitzen drei bis vier Dendriten und erstrecken ihr Axon in die darüber liegende Purkinje-Zellschicht, wo sie die Dendriten der Purkinje-Zellen kontaktieren [84]. Die Körnerzellen sind die einzigen erregenden Neurone der Kleinhirnrinde [93].

1.2.6 Betrachtung der Membrankapazität biologischer Neuronen

Um die Eigenschaften von biologischen Neuronen mit denen von technischen Implementierungen später vergleichen zu können, soll an dieser Stelle auf die Membrankapazität eines biologischen Neurons eingegangen werden. Die Betrachtung der Kapazität ist natürlich eine sehr technische Sicht, kann aber aufgrund der Ladungstrennung an der Zellmembran als Modell für die biologische Membran angenommen werden. Da die eingesetzten Kapazitäten später einen wesentlichen Flächenanteil der Gesamtfläche von technischen Umsetzungen von Integrate-and-Fire Neuronen ausmachen, soll die durch die Zellmembran gebildete Kapazität näher betrachtet werden. Aus der Literatur entnimmt man recht ungenaue Angaben zur flächenbezogenen Kapazität der Zellmembranen:

> „Ein typischer Wert für die Membrankapazität einer Nervenzelle ist $1\,\mu\mathrm{F/cm^2}$ Membranfläche.“ [61]

> „The generally agreed upon value for C_m is $1\,\mu\mathrm{F/cm^2}$.“ [65]

In [65] werden neben dem oben angegebenen Zitat allerdings auch weitere Angaben für die flächenbezogene Kapazität der Zellmembran zwischen $0,65\,\mu\mathrm{F/cm^2}$ und $0,90\,\mu\mathrm{F/cm^2}$ gemacht.

Aus der im vorhergehenden Abschnitt gemachten Annahme eines Durchmessers von $80\,\mu$m für die Purkinje-Zelle und der Angabe über die Membrankapazität ergibt sich eine Gesamtkapazität von ca. 201 pF pro Neuron. Dies entspricht bei einem mittleren Ruhepotential von $-80\,$mV – die Berechnung des Potentials erfolgt später – einer Ladungstrennung von $16,1 \cdot 10^{-12}\,$C bzw. $100 \cdot 10^6$ Ionen.

Zum Vergleich: Eine Kapazität der Größe 1 pF braucht in einer 350 nm Technologie eine Fläche von ca. $300\,\mu\mathrm{m^2}$. Der Nachbau der Membrankapazität der Purkinje-Zelle braucht in der gewählten CMOS Technologie somit etwa $0.06\,\mathrm{mm^2}$ Chipfläche, die etwa dreifache Fläche des biologischen Neurons. In einer aktuellen 130 nm CMOS Technologie braucht man nur noch $75,5\,\mu\mathrm{m^2}$ für eine Kapazität von 1 pF und damit etwa $0,017\,\mu\mathrm{m^2}$ Fläche für eine der Purkinje-Zelle äquivalente Kapazität. Damit ist die Fläche für die Kapazität des

Tabelle 1.1: Intra- und extrazelluläre Ionenkonzentrationen (aus [85]).

Ion	Intrazellulär	Extrazellulär
Na^+	12 mmol/l	145 mmol/l
K^+	155 mmol/l	4 mmol/l
Ca^{2+}	10^{-8} – 10^{-7} mol/l	2 mmol/l
		andere Kationen: 5 mmol/l
Cl^-	4 mmol/l	120 mmol/l
HCO_3^-	8 mmol/l	27 mmol/l
A^- (große Anionen)	155 mmol/l	

technischen Neurons erstmals unter der Zelloberfläche der Purkinje-Zelle, wenn zusätzliche Schaltkreise an dieser Stelle noch nicht berücksichtigt werden.

Die Gründe für diesen Unterschied der Kapazität bei der 350 nm Technologie liegen zum Einen in der Dicke der Zellmembran von ca. 2 nm bis 5 nm Dicke gegenüber einer Dicke des Gate-Oxids von etwa 7 nm bis 8 nm, zum Anderen in der Beschaffenheit des Dielektrikums. Die Dicke des Gate-Oxids aus Siliziumdioxid nimmt bis zur 130 nm Technologie mit jedem Technologieschritt ab, wird aber in Zukunft durch andere Materialien mit besseren Eigenschaften bezüglich Leckströmen ersetzt, wobei die Dicken nicht weiter abnehmen wird. Fasst man die biologische Zellmembran als Dielektrikum auf, wird dieses durch die Lipiddoppelschicht mit einer genäherten Dielektrizitätskonstante von $\varepsilon_{r,\text{Lipid}} = 2,1$ gebildet [11], bei CMOS Gate-Kapazitäten durch das Gate-Oxid – üblicherweise aus Siliziumdioxid (SiO_2) – mit etwa $\varepsilon_{r,\text{Gateoxid,SiO}_2} \approx 3,9$.

Einschub: Berechnung des $\varepsilon_{r,\text{Lipid}}$

Die Kapazität der Membran ist von der Dicke T_{Ox} des Dielektrikums und den Materialkonstanten abhängig.

$$T_{\text{Ox}} = \frac{\varepsilon_0 \cdot \varepsilon_r}{C_{\text{Ox}}} \tag{1.2}$$

Mit den aus [11] entnommenen Werten für die Dicke der Doppellipidschicht und die spezifische Kapazität ergibt sich mit

$$4 \cdot 10^{-9}\,\text{m} = \frac{8,85 \cdot 10^{-12}\,\text{F/m} \cdot \varepsilon_r}{\frac{1 \cdot 10^{-6}\,\text{F}}{1 \cdot 10^{-4}\,\text{m}^2}}$$

die mittlere Dielektrizitätskonstante der Zellmembran zu einem Wert von $\varepsilon_r \approx 2,1$.

1.2.7 Membranpotential

Das Membranpotential, also die Differenz der elektrischen Ladung zwischen dem intrazellulären Raum und dem extrazellulären Raum wird durch aktive und passive Mechanismen hervorgerufen. Die in der Zelle hoch konzentrierten K^+-Ionen können durch spezifische K^+-Kanäle in der Zellmembran aus der Zelle hinaus diffundieren. Durch den Verlust an positiver Ladung im Zellinnenraum wird das Zellinnere in Bezug auf das äußere Potential negativ aufgeladen. Die Aufladung wirkt der Diffusion von K^+ entgegen, und es wird ein Gleichgewicht erreicht, wenn die Kraft, die aus dem elektrischen Feld auf die Ionen wirkt, die Kraft durch den „Diffusionsdruck“ gerade aufhebt. Dieses Potential wird Gleichgewichtspotential genannt.

Das Gleichgewichtspotential eines bestimmten Ionentyps wird durch die Nernst-Gleichung bestimmt:

$$E = \frac{\mathrm{R}T}{z\mathrm{F}} \cdot \ln \frac{[\mathrm{Ion}]_{\text{außen}}}{[\mathrm{Ion}]_{\text{innen}}} \tag{1.3}$$

Das Potential ist also von der Gaskonstante R, der absoluten Temperatur T, der Ladungszahl z des Ions (negativ für Anionen), der Faradaykonstante F und der Ionenkonzentration [Ion] der betreffenden Ionen im Zellinnenraum und im extrazellulären Raum abhängig.

Bei Körpertemperatur (T=310 K) wird das K^+-Gleichgewichtspotential E_K mit den Werten aus Tab 1.1 zu

$$E_\mathrm{K} = \frac{8,314472 \cdot 310}{1 \cdot 96485,3383} \cdot \ln \frac{[\mathrm{K}^+]_{\text{außen}}}{[\mathrm{K}^+]_{\text{innen}}} = -97,7\,\mathrm{mV}. \tag{1.4}$$

Gleichzeitig diffundieren Natrium-Ionen aus dem extrazellulären Raum durch spezifische Ionenkanäle in das Zellinnere. Die Na^+-Permeabilität der ruhenden Zellmembran ist gering. In ihr sind nur wenige Na^+-Kanäle geöffnet. Durch den Konzentrationsgradienten und das Ruhepotential begünstigt, strömen jedoch Na^+-Ionen in die Zelle ein und stören das Gleichgewicht. Es ergibt sich ein Ruhepotential (engl. steady-state) aus den überlagerten Gleichgewichtspotentialen der beteiligten Ionentypen, welches in späteren Kapiteln durch die Goldmann-Gleichung beschrieben wird. Der Diffusion wirkt der aktive Mechanismus der NaK-Pumpe entgegen, der bereits in vorhergehenden Abschnitten behandelt wurde. Der aktive Pumpmechanismus verschiebt im ausgeregelten System den Gleichgewichtspunkt der Ionenkonzentrationen um einen kleinen Anteil zu den Gleichgewichtskonzentrationen $c_{\mathrm{K},0}$ und $c_{\mathrm{Na},0}$.

Durch Injektion von Ionen in die Zelle z. B. bei Erregung durch eine synaptische Verbindung, kann das Membranpotential vom Ruhepotential verschoben werden. Dabei spricht man bei Anhebung des negativen Ruhepotentials von Depolarisation der Membran, im Fall der weiteren Absenkung des Membranpotentials z. B. durch inhibitorischen Einfluss einer Synapse von Hyperpolarisation. Der Ausgleich der Ionenkonzentrationen und die damit verbundene Wiederherstellung des Ruhepotentials werden als Repolarisation bezeichnet.

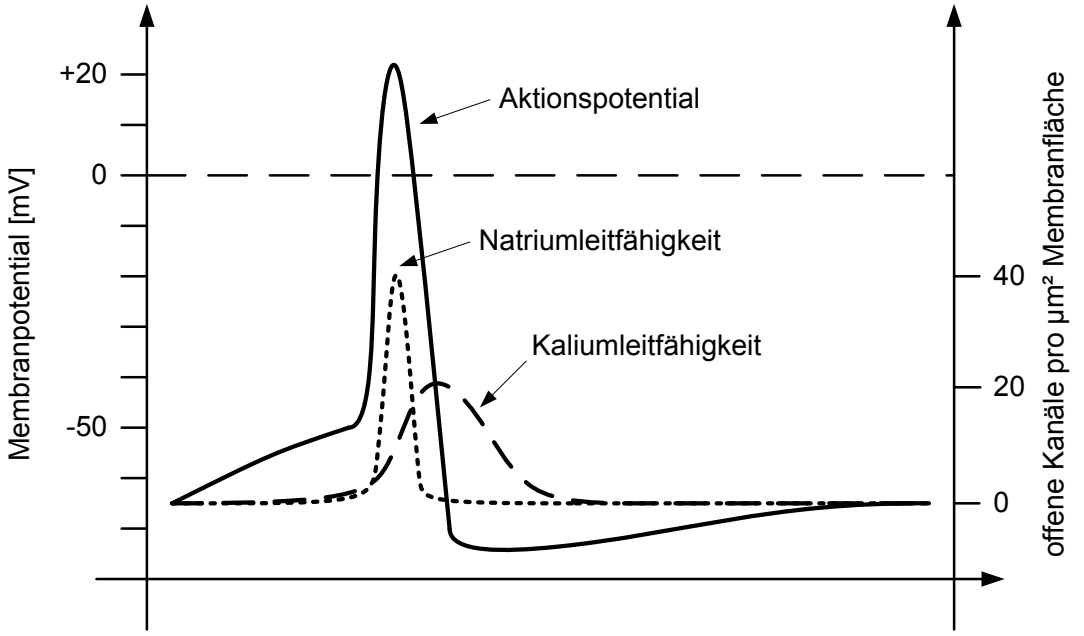


Abbildung 1.5: Zeitlicher Verlauf eines Aktionspotentials und der Permeabilität der Membran für bestimmte Ionentypen (nach [17]).

1.3 Aktionspotential

Wie bereits im vorhergehenden Abschnitt beschrieben, sind die Verteilung der Ionen und die Diffusion durch passive Transportkanäle für das Membranpotential verantwortlich. Daneben existieren weitere Arten von Ionenkanälen, z. B. die spannungsgesteuerten Kanäle. Aus der Literatur sind über 30 Kalium-Kanäle bekannt, welche an der Entstehung des Membranpotentials beteiligt sind, aber die Erzeugung eines Aktionspotentials lässt sich im Wesentlichen mit den potentialgesteuerten Natrium- und Kalium-Kanälen beschreiben [85]. Spannungsabhängige Kanäle wechseln Ihren Zustand von geschlossen zu offen oder umgekehrt in Abhängigkeit vom bestehenden Membranpotential. Erst durch Ihre Funktion wird die Entstehung eines definierten Signals möglich.

In Abb. 1.5 ist der zeitliche Verlauf eines Aktionspotentials und der Permeabilität der Zellmembran für bestimmte Ionentypen dargestellt. Erreicht das Membranpotential durch Einstrom von Natrium-Ionen in die Zelle einen bestimmten Wert, das sogenannte Schwellenpotential (im späteren Verlauf auch als Schwellenspannung oder Feuerschwelle bezeichnet), öffnen sich schlagartig bis zu dem Zeitpunkt inaktive (geschlossene), spannungsgesteuerte Natriumkanäle, welche für einen noch höheren Einstrom von Natrium-Ionen in die Zelle und für einen damit verbundenen abrupten Anstieg des Membranpotentials sorgen. Das Membranpotential erreicht dabei einen Spannungshub von 90 mV bis 100 mV. Bei anhaltender Depolarisation der Zellmembran schließen die Natrium-Ionenkanäle wieder und gehen in einen refraktären Zustand über, d. h., dass sich die Ionenkanäle auch bei höherer Depolarisation nicht wieder öffnen können. Dieser Zustand wird erst wieder bei Erreichen des Ruhemembranpotentials aufgehoben. Mit etwas Zeitversatz nach den Natriumkanälen öffnen spannungsinduziert auch die Kaliumkanäle (verzögerte K^+-Kanäle) und sorgen für

einen Ausstrom von K^+-Ionen. Dieses wirkt der Depolarisation des Membranpotentials entgegen und führt zur Repolarisation der Zellmembran. Da das Schließen der Kaliumkanäle langsam erfolgt, wird die Zellmembran nach Abbau des Aktionspotentials leicht hyperpolarisiert und anschließend durch den aktiven Transportmechanismus nach der Wiederherstellung der Ruhe-Ionenkonzentrationen wieder auf das Ruhepotential gebracht.

Neben den passiven Ionenkanälen und den spannungsgesteuerten Ionenkanälen existieren noch transmittergesteuerte Kanäle, wie z. B. von Calcium-Ionen abhängige K^+-Kanäle. Hier ist Calcium als Steuerstoff im engeren Sinne aufzufassen, dessen Anwesenheit direkten Einfluss auf die Permeabilität des jeweiligen Kanals hat. Diese Ionenkanäle werden hauptsächlich dafür verantwortlich gemacht, die Adaption der Pulsrate mit der Zeit an einen gleichmäßig erregten Eingang zu vollziehen (dieses zeigt sich in der Abnahme der Häufigkeit des Auftretens von Aktionspotentialen an konstant erregten Neuronen), oder ein Neuron mit schnellen und häufigen Aktionspotentialen (ein sog. Burst) nach einer bestimmten Zeit abrupt aus diesem Zustand zu bringen, und eine Ruhepause zu erzwingen.

Calcium scheint aber im Besonderen an Synapsen eingesetzt zu werden, an denen es nach verschiedenen Untersuchungen an Lernprozessen in der Langzeitpotenzierung (LTP) beteiligt ist [18, 72].

Nach [6] sind mehrere unterschiedliche Ionenkanäle an der Entstehung und der speziellen Ausprägung des Aktionspotentials beteiligt. Die für das Entstehen des Aktionspotentials verantwortlichen Kanäle wurden weiter oben schon behandelt, daneben gibt es aber noch weitere Typen von Kanälen, die die Eigenschaften der Aktionspotentiale beeinflussen. An Zellmembranen von Nervenzellen kann beobachtet werden, dass eine dauerhafte Erregung unterhalb einer bestimmten Schwelle nicht zum Auslösen eines Aktionspotentials führt. Dieses Verhalten, das sog. Unterschwellenverhalten, kann durch die oben beschriebenen beiden Kanäle nicht hervorgerufen werden, vielmehr sorgen die sog. spannungsgesteuerten frühen K^+-Kanäle für einen Ausgleich an Ladungsträgern, sobald die Zellmembran depolarisiert wird. Oberhalb einer bestimmten Schwelle werden die frühen K^+-Kanäle inaktiviert und die Na^+-Kanäle und die verzögerten K^+-Kanäle können wirken.

Eine weitere beobachtete Eigenschaft der Aktionspotentiale ist bei gleichbleibender Erregung der Membran die stetige Abnahme der Rate der Aktionspotentiale. Für diesen Vorgang der Adaption sind Ca^{2+}-gesteuerte K^+-Kanäle im Zusammenspiel mit spannungsgesteuerten Ca^{2+}-Kanälen verantwortlich. Bei der Depolarisation der Zellmembran mit jedem Aktionspotential strömt Calcium in den Zellinnenraum, womit die Calcium-Konzentration erhöht wird. Ca^{2+}-gesteuerte Kalium-Kanäle öffnen in Abhängigkeit von der Calcium-Konzentration und wirken der Depolarisation der Zellmembran dauerhaft entgegen. Der Abbau der Ca^{2+}-Konzentration erfolgt mit einer ATPase, ähnlich zum Transport bei der Natrium-Kalium-Pumpe.

In [8] wird der Energiebedarf für den Abbau eines Aktionspotentials mit $3,84 \cdot 10^8$ ATP-Molekülen angegeben. Dies entspricht einer aufzuwendenden Energie von ca. 29 pJ pro Aktionspotential.

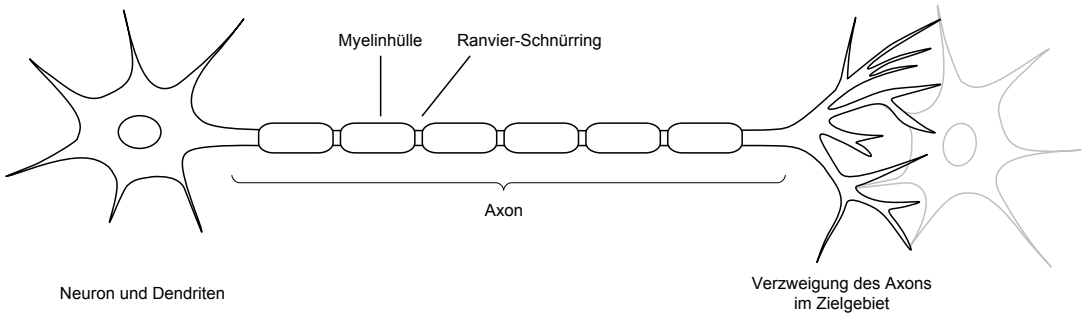


Abbildung 1.6: Neuron mit myelinisiertem Axon und Ranvier-Schnürringen zur schnellen Signalfortleitung. Weiter ist die Verzweigung des Axons im Zielgebiet anderer Neurone angedeutet.

1.4 Reizweiterleitung

Die Signalübertragung im menschlichen Körper dient im Wesentlichen zur Beeinflussung des Verhaltens von Zellen untereinander. So können einzelne Zellen das Verhalten von einzelnen bis ganzen Gruppen anderer, zum Teil weit entfernten Zellen steuern. Die dabei beteiligten Mechanismen nutzen Moleküle zur Kommunikation, welche an der empfangenden Zelle an auf der Zellmembran platzierten oder sich in der Zelle befindenden Rezeptoren binden [59] und über sekundäre Transmitter bestimmte Aktionen in der empfangenden Zelle auslösen können.

Hier bildet die für die Übertragung neuronaler Information (in Form des Aktionspotentials) synaptische Übertragung den wichtigsten Übertragungsweg. Das am Axonhügel des sendenden Neurons ausgebildete Aktionspotential wird über das Axon an zum Teil bis zu 1 m weit entfernte Empfängerneurone geleitet (siehe Abb. 1.6). Eine Besonderheit dieses Transports tritt bei myelinisierten, d. h. mit Myelin umhüllten Axonen auf. Diese Art von Axon weist regelmäßige Einschnürungen über seine Länge auf, die Ranvier-Schnürringe. Im myelinumhüllten Mark fällt das Potential mit der räumlichen Weiterleitung des Signals wie in marklosen Nervenfasern ab. Das Aktionspotential wird während seines passiven Transports entlang des Axons durch kapazitive und resistive Einflüsse in seiner Ausprägung abgeschwächt und verrundet. In den Bereichen der Einschnürungen (Ranvier Schnürringe) wird das Aktionspotential (AP) unter Energieumsatz regeneriert, da hier viele potentialabhängige Na^+ und K^+ Kanäle vorhanden sind. An den Ranvier-Schnürringen findet also eine Rekonstruktion des Aktionspotentials statt, indem die gleichen Vorgänge wie an der Zellmembran bei der Erzeugung des Aktionspotentials ablaufen. Myelinisierte Axone leiten die Signale daher mit weit größerer Geschwindigkeit zum Empfänger, als unmyelinisierte Axone. Diese sehr schnelle, sprunghafte Weiterleitung des Aktionspotentials wird saltatorische Erregungsleitung genannt und erreicht im Aα-Fasertyp eine mittlere Leitungsgeschwindigkeit von 100 m/s (Klassifikation nach Erlanger und Gasser, [86]).

Die Bewertung der Aktionspotentiale am empfangenden Neuron findet an den Synapsen statt, die sowohl exzitatorisch (erregend) als auch inhibitorisch (hemmend) am Neuron wirken können. Die Synapsen wandeln den elektrischen Puls in ein chemisches Signal um,

indem verschiedene Neurotransmitter, z. B. Acetylcholin, Glutamat, γ-Aminobuttersäure (GABA) und Glycin ausgeschüttet werden. Die erregenden Neurotransmitter Acetylcholin und Glutamat binden an spezifische Rezeptoren der Zellmembran des Neurons und öffnen transmittergesteuerte Natrium-Kanäle, welches einen Einstrom von Na^+ in die Zelle verursacht und zur Depolarisation der Zellmembran bis zum Auslösen eines Aktionspotentials führen kann. Dem entgegen stehen die inhibitorischen Neurotransmitter GABA und Glycin, welche an transmittergesteuerten Chlorid-Kanälen binden und diese öffnen, so dass durch den Einstrom der Cl^--Ionen die Membran repolarisiert oder hyperpolarisiert wird. Durch das Auftreten des elektrischen Aktionspotentials wird also über den Umweg der Ausschüttung chemischer Transmitter ein bestimmtes Potential auf der Zellmembran, das sogenannte postsynaptische Potential (PSP) erzeugt, welches einen charakteristischen Verlauf bei erregenden und hemmenden Signalen aufweist. In diesem Zusammenhang wird von erregendem postsynaptischen Potential (EPSP) und inhibitorischem postsynaptischen Potential (IPSP) gesprochen. Entlang des Dendriten des empfangenden Neurons können sich die PSP nicht nur räumlich sondern auch zeitlich überlagern, wie am technischen Modell in Kap. 2.1 für das dieses Verhalten nachbildende Spike-Response Modell noch gezeigt wird.

Treten ein EPSP und IPSP zur gleichen Zeit am gleichen Ort auf, so ergibt die Überlagerung beider Potentiale nicht die Summe der einzelnen PSP, sondern der geringere IPSP hemmt den weitaus größeren EPSP fast vollständig. Dieses ist durch den Einfluss der konzentrationsabhängig geöffneten Ionenkanäle zu erklären, zu deren näherer Erläuterung der Leser auf weiterführende Literatur [6] verwiesen wird.

1.5 Diskussion

In diesem Kapitel sind die Grundlagen der biologischen Zelle, insbesondere die Eigenschaften der Zellmembran und ihre Funktion an der Nervenzelle beschrieben worden. Es gibt einen Überblick über die grundlegenden Mechanismen, die in der Nervenzelle wirken, insbesondere den Ort des Energieumsatzes. Das hier verwendete Modell der Zellmembran ist Grundlage der späteren Modellierung.

Kapitel 2

Stand der Technik pulsender Neurone

Technische Umsetzungen von Neuronen- und Synapsenmodellen sowie komplexer neuronaler Netze wurden mit unterschiedlichen Methoden und Plattformen realisiert. So sind im Bereich der pulscodierten neuronalen Netze vier Zweige entstanden, die im Folgenden kurz gezeigt werden sollen. Auf die Besonderheiten einiger ausgewählter Umsetzungen der für diese Arbeit relevanten Zweige wird im Anschluss detaillierter eingegangen, um die in dieser Arbeit entwickelten Lösungen einordnen zu können.

Aufgrund des einfachen Zugangs sind viele Umsetzungen pulscodierter neuronaler Netze (PCNN) gerade aus dem Bereich der Computational Intelligence und Teilen der Informatik in Simulatoren auf herkömmlichen Personal Computern entstanden. Hier wurden auf der einen Seite die Modelle einzelner Neuronen detailliert modelliert, um das Verhalten biologisch plausibler Modelle statt in vivo oder in vitro zu untersuchen, direkt am PC untersuchen zu können [10]. Auf der anderen Seite wurden Simulatoren umgesetzt, die es erlauben, komplexe Netzwerke mit unterschiedlichen Neuronen- und Synapsentypen zu analysieren [12, 47]. Für eine ausführliche Betrachtung der zurzeit verfügbaren Simulatoren und ihre Gegenüberstellung wird an dieser Stelle auf eine aktuelle Veröffentlichung von Brette et al. [14] verwiesen.

Da die Simulation von großen neuronalen Netzen schnell an die Grenzen der vorhandenen Rechenleistung stößt, wurden Teile der Simulation auf spezielle Hardware ausgelagert, um die Rechenzeit zu verkürzen. Da die Umsetzung von komplexen, biologisch plausiblen Neuronen- und Synapsenmodellen auf analoge oder digitale Hardware schwierig ist, werden für diesen Zweck üblicherweise einfache elektrische Modelle der Zellmembran verwendet, die das Verhalten der biologischen Neurone nachahmen, detaillierte Vorgänge aber nicht berücksichtigen. Mit dem Argument, dass die kleinen Details das Systemverhalten nicht wesentlich beeinflussen und unter der Annahme, dass allein das Auftreten eines Aktionspotentials die Information in einem neuronalen Netz codiert, werden mit diesen Modellen komplexe Netze mit der Hilfe von Hardware-Beschleunigern simuliert. Dabei werden unter Anderem auch zeitliche und örtliche Zusammenhänge zwischen den Aktionspotentialen

betrachtet. Erst seit wenigen Jahren bemühen sich Forschergruppen um die detailliertere Modellierung der Neuronen- und Synapsenmodelle in spezialisierter Hardware, um die in langwierigen Simulationen ermittelten Ergebnisse schnell reproduzieren und nutzen zu können.

Die für die Simulation pulscodierter neuronaler Netze eingesetzte Hardware unterscheidet sich hauptsächlich durch die Realisierung als digitaler, analoger oder gemischt analog-digitaler Schaltkreis. Die digitalen Implementierungen wiederum teilen sich in Systeme, die wenige Verarbeitungseinheiten mit schnellen Integratoren und einer schnellen Speicheranbindung implementieren und es erlauben, eine große Anzahl virtueller Neurone zu simulieren, und in Systeme, die für jedes Neuron eine eigene Verarbeitungseinheit vorsehen, um alle Neurone zur gleichen Zeit verarbeiten zu können und das System in Echtzeit zu betreiben. Erstere Systeme sind, sofern sie Echtzeitanforderungen erfüllen sollen, durch ihre Speicherbandbreite, die Auslegung der Integratoren und der hinterlegten Mechanismen sowie den maximalen Arbeitstakt in der Anzahl der simulierten Neurone beschränkt [31]. Die anderen Systeme sind im Wesentlichen durch die Chipfläche bzw. die damit verbundenen Kosten oder die Anzahl an Logikzellen im FPGA begrenzt.

In den letzten Jahren sind vor allem durch vom Bundesministerium für Bildung und Forschung und der Europäischen Union geförderte Projekte aus Konsortien besetzt mit Vertretern der Universitäten, Forschungseinrichtungen und der Industrie viele Varianten von Neuronenmodellen entstanden, die sich mehr oder weniger für den Einsatz auf spezieller Hardware eignen. Einige in dieser Arbeit zitierten Veröffentlichungen stammen zum Teil aus den Projekten VisionIC [3], SpikeForce [2] und Facets [1], in denen pulscodierte neuronale Netze zur Informationsverarbeitung zum Einsatz kommen, oder aber Forschungsergebnisse der Neurobiologie durch Simulationen analysiert und verifiziert werden. Die aus diesen Projekten hervorgegangenen Publikationen werden aber nicht separat kenntlich gemacht.

Neben den bereits erwähnten softwaretechnischen Modellen und Simulatoren entstanden viele Ansätze zur Nutzung von reprogrammierbaren bzw. rekonfigurierbaren Bausteinen z. B. Feldprogrammierbaren Gatteranordnungen (FPGA), welche zum Teil die speziellen Bauelemente der FPGA ausnutzen, sich aber nur schwer oder mit hohen Kosten in einem Standardzellenprozess auf einem anwendungsspezifischen Baustein (ASIC) realisieren lassen. Dagegen stehen die Ansätze aus dem Bereich der analogen Schaltungstechnik, welche spezielle Implementierungen von Neuronenmodellen auf einem ASIC realisieren. Einige von diesen Realisierungen nutzen dabei allerdings Spezialverfahren der Chipfertigung aus, welche nicht in jedem Herstellungsprozess verfügbar sind oder aber die Herstellung teuer und somit für eine Massenproduktion unwirtschaftlich machen. In diesem Kapitel soll eine Übersicht über biologienahe, technisch realisierbare Neuronenmodelle gegeben werden. Dieses bedeutet auch, dass sich diese Arbeit im Folgenden nur noch mit zeitbehafteten, Pulse verarbeitenden Neuronen beschäftigt. Frühere Konzepte von zeitinvarianten, z. T. wertekontinuierlichen Modellvorstellungen wie z. B. das Perzeptron nach McCulloch und Pitts [75] oder nach Rosenblatt [81] sollen an dieser Stelle zur Vollständigkeit zwar erwähnt aber nicht im Detail diskutiert werden.

2.1 Technische Darstellungen von Neuronen

Im Folgenden werden die gebräuchlichsten Modelle für technische Realisierungen von Neuronen in pulscodierten neuronalen Netzen behandelt. Die Nomenklatur für die gezeigten Modelle ist aus [70] entnommen. Grundlage für diese Modelle bildet das (elektrisch) beobachtbare Verhalten von Neuronen, insbesondere das Verhalten bei Erregung der Neurone mittels konstanten Stroms oder mittels Pulsfolgen. Die Beobachtung des Zurücksetzens der gemessenen Spannung über der Membrankapazität und das Aussenden eines eigenen Pulses, dem Aktionspotential, beim Erreichen einer bestimmten Spannung über der Zellmembran, der sogenannten Feuerschwelle, wird hier ebenfalls berücksichtigt.

2.1.1 Spike Response Modell

Das Spike Response Modell [32, 64] wird vor allem durch Antwortfunktionen auf bestimmte Ereignisse beschrieben. So umfasst es eine zeitliche Antwort für die Erregung des Neurons durch Pulse präsynaptischer Neurone und eine zeitliche Antwort auf das eigene ausgelöste Aktionspotential. Der Zustand bzw. das Membranpotential des Neurons i zum Zeitpunkt t wird mit der Zustandsvariablen $u_i(t)$ (2.1) beschrieben.

$$u_i(t) = \sum_{t_i^{(f)} \in \mathcal{F}_i} \eta_i \left(t - t_i^{(f)}\right) + \sum_{j \in \Gamma_i} \sum_{t_j^{(f)} \in \mathcal{F}_j} w_{ij} \epsilon_{ij} \left(t - t_j^{(f)}\right) \tag{2.1}$$

Die Antwort des Neurons i auf eine Erregung durch Pulse präsynaptischer Neurone $j \in \Gamma_i$ wird durch die Antwortfunktion $\epsilon_{ij}(t)$ beschrieben. Diese enthält die Postimpulsantwort auf einen Reiz, der exzitatorisch (erregend) oder inhibitorisch (hemmend) wirken kann und zum Zeitpunkt $t_j^{(f)} \in \mathcal{F}_j$ auftritt. Dabei wird nicht die Form des eintreffenden Pulses, sondern nur dessen Auftreten berücksichtigt. Die Pulsantwort wird als *excitatory post-synaptic pulse* (EPSP) bzw. *inhibitory post-synaptic pulse* (IPSP) bezeichnet. Ein typischer Verlauf der Pulsantworten ist in Abb. 2.1 dargestellt, welcher dem „verrunden“ des Aktionspotentials durch die Wandlung der Art des Signals (elektrisch – chemisch – elektrisch) an der Synapse und dem Dendriten des empfangenden Neurons nachempfunden ist und die Form einer α-Funktion beschreibt. Beide Pulsformen, die exzitatorische und die inhibitorische, können unterschiedlich ausgeprägt sein. Die zeitliche und räumliche Überlagerung aller mit der synaptischen Stärke w_{ij} gewichteten Pulsantworten ergibt die Doppelsumme des rechten Terms von (2.1), welcher dem Membranpotential des Neurons ohne Erzeugung eines Aktionspotentials entspricht. Da das Neuron das Membranpotential beim Auslösen eines Aktionspotentials auf einen Initialwert zurücksetzt, wird dieses durch den zusätzlichen Term, der Antwort $\eta_i(t)$ auf das eigene Aktionspotential zum Zeitpunkt $t_i^{(f)}$ modelliert. Die Terme $\mathcal{F}_i$ sowie $\mathcal{F}_j$ bezeichnen alle Feuerzeitpunkte des betrachteten Neurons i und aller seiner präsynaptischen Neurone j.

Treffen nacheinander mehrere Aktionspotentiale präsynaptischer Neurone $j \in \Gamma_i$ am empfangenden Neuron i ein, überlagern sich die Postimpulsantworten des empfangenden

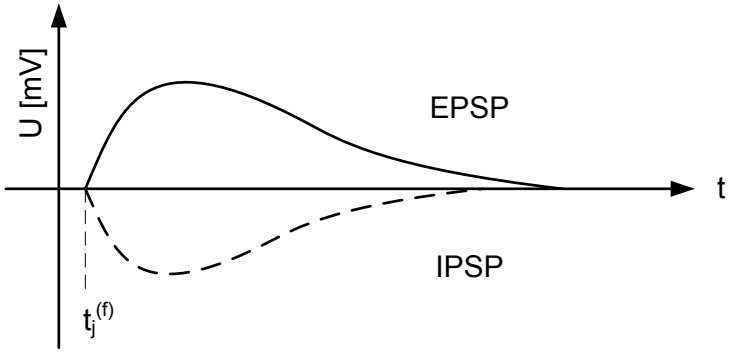


Abbildung 2.1: Beispielhafter Verlauf einer exzitatorischen und inhibitorischen postsynaptischen Antwort. Die exzitatorische und inhibitorische Antwort müssen nicht denselben Verlauf besitzen, sondern können separat modelliert werden.

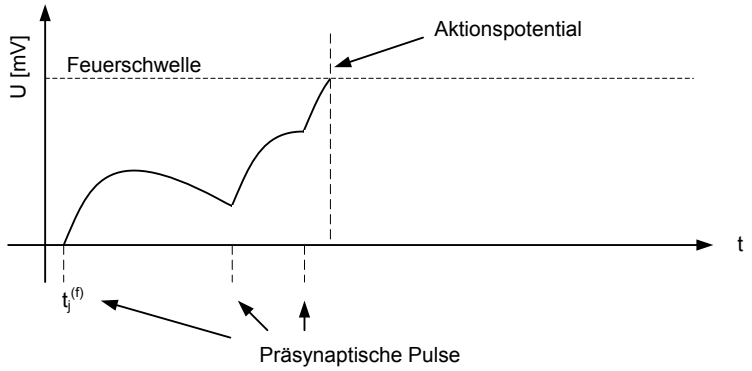


Abbildung 2.2: Auslösen eines Aktionspotentials durch zeitliche und räumliche Überlagerung exzitatorischer Pulse.

Neurons zeitlich zum Membranpotential $u_i(t)$ (Abb. 2.2). Erreicht das Membranpotential die eingestellte Feuerschwelle, wird ein eigenes Aktionspotential erzeugt und ausgesendet und das Membranpotential zurückgesetzt.

2.1.2 Leaky Integrate and Fire Modell

Das Integrate and Fire (IAF) Modell ist durch die Sicht des Neurons als elektrischer Ersatzschaltkreis geprägt. Das am biologischen Neuron beobachtbare elektrische Verhalten hat zur Modellvorstellung der Membrankapazität geführt, auf der einlaufende Strompulse über der Zeit integriert werden und welche durch passive Verlustterme – und im Fall eines Aktionspotentials – durch einen aktiven Mechanismus entladen wird. Die Betrachtung passiver Entladung des Neuronenmodells führt zur Bezeichnung Leaky Integrate and Fire Modell (LIAF oder LIF). Im Folgenden wird diese Unterscheidung in selbstentladende und Ladung erhaltende Membrane nicht mehr getroffen sondern von einer sich passiv entladenden Membran ausgegangen. Da die Bezeichnungen in der einschlägigen Literatur

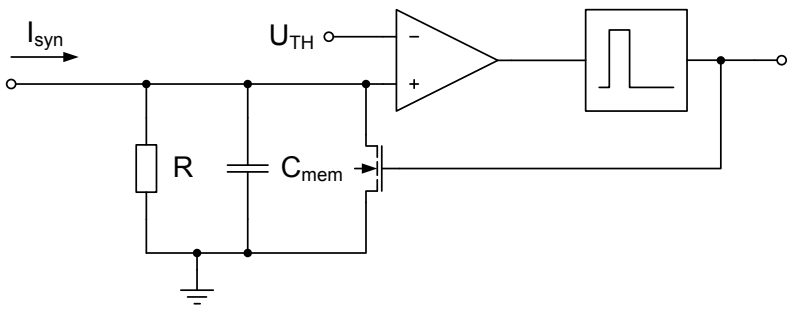


Abbildung 2.3: Modell eines Integrate and Fire Neurons aus der Modellvorstellung eines RC-Schaltkreises für die Membran. Nicht eingezeichnet sind zusätzliche notwendige Elemente für die Wandlung des Spannungspulses am Ausgang in den Eingangsstrom der nächsten Stufe (Synapse).

nicht konsistent geführt werden, wird im Text dieser Arbeit daher nur die Bezeichnung LIAF geführt. Ein einfaches Modell eines LIAF Neurons ist in Abb. 2.3 dargestellt. Es enthält die Membrankapazität C_{mem} mit parallel geschaltetem Widerstand R, welcher die passive Entladung der Membran modelliert. Daneben ist ein Vergleichselement dargestellt, welches das Membranpotential mit einem Schwellwert vergleicht. Übersteigt das Membranpotential den Schwellwert, wird ein Spannungspuls definierter Länge ausgesandt und das Membranpotential wird aktiv über den eingezeichneten Schalter, hier als MOS-Transistor ausgeführt, zurückgesetzt (die Membrankapazität wird schlagartig entladen).

Das Membranpotential $u(t)$ für das LIAF Neuron ergibt sich aus den erregenden und hemmenden Strömen I_{syn} und den parasitären Effekten der Bauelemente der Schaltung, so dass die Gleichung für den Fall der Aufladung bis zur Feuerschwelle mit

$$I_{\text{syn}}(t) = \frac{u(t)}{R} + C_{\text{mem}}\frac{du(t)}{dt} \quad \text{, wenn} \quad u(t) < U_{\text{TH}} \tag{2.2}$$

angegeben werden kann. Beim Erreichen der Feuerschwelle U_{TH} wird das Membranpotential zurückgesetzt und von neuem mit der Integration einlaufender Strompulse begonnen. Maass hat nachgewiesen [71], dass das Integrate and Fire Neuron ein Spezialfall des Spike Response Modells ist.

2.2 Digitale Implementierungen

Die Verwendung von FPGAs als Plattform für die Implementierung von neuronalen Netzen ist durch ihre Parallelität, Rekonfigurierbarkeit, die schnelle Entwurfszeit und den damit verbundenen geringen Kosten besonders geeignet. Für die Evaluierung von neuronalen Netzen sind aus den genannten Gründen Implementierungen weit verbreitet, die primär für das FPGA ausgelegt sind. Nachfolgend werden einige Implementierungen von digitalen neuronalen Netzen vorgestellt.

Die erste vorgestellte Implementierung von Schrauwen [87] beschäftigt sich mit der Realisierung eines kleinen neuronalen Netzes, das auf die Struktur paralleler Rechenelemente eines FPGAs abgebildet wird. Zusammen mit seriellen Verarbeitungseinheiten wird eine Systemstruktur aufgebaut, die die Verschaltung der Elemente Synapse und Neuron aus dem biologischen Vorbild nachahmt. Zudem erhöht die modulare Aufteilung des Systems in Komponenten die Modellvielfalt für die Synapsenmodelle und Membranmodelle. Die nachfolgende Arbeit von Upegui [96] stellt eine einfache Implementierung eines Neurons vor. Der Entwurf der Architektur berücksichtigt die dynamische Rekonfigurierbarkeit eines FPGAs. Die dritte vorgestellte Implementierung von Torres-Huitzil [92] arbeitet mit oszillierenden Neuronen in einer zweidimensionalen Verbindungsstruktur. Diese von der Struktur des visuellen Kortex abgeleitete Architektur wird zur Bildsegmentierung eingesetzt. Eine komparative Arbeit leistet Johnston [57] mit dem Vergleich von zwei klassischen und zwei pulscodierten neuronalen Netzen. Der Entwurf der in [57] vorgestellten Modelle erfolgt durchweg mit dem Xilinx System Generator. Eine Implementierung mit stochastischem Ansatz wird von S. Maya [74] vorgestellt. Dabei werden reelle Werte zwischen 0 und 1 durch eine zufällige Bitfolge repräsentiert, deren Anzahl gesetzter Bits im Verhältnis zur Länge des Vektors der reellen Zahl entspricht. Eine ASIC-Implementierung digitaler pulscodierter neuronaler Netze wird von Godin [35] vorgestellt, in der statische Neuronenmodelle mit einem intern pulsend arbeitenden Modell emuliert werden.

2.2.1 Die Schrauwen-Implementierung

Die Systemarchitektur der Schrauwen-Implementierung [87] ist in Abbildung 2.4a dargestellt. Die seriellen Daten mehrerer Eingänge werden über parallele Signalpfade durch Modelle der Synapsen und Dendriten nach und nach zusammengeführt und zum Modell der Membran geführt. In den jeweiligen Systemkomponenten werden die Daten bitseriell verarbeitet. Zur Durchsatzoptimierung sind Pipelinestufen zwischen den einzelnen Baumebenen eingefügt.

Eine zeitbasierte Simulation wird einer ereignisorientierten Simulation, die gerade bei kleinen Netzen einen höheren Ressourcenbedarf (durch den Mehraufwand der Listenverwaltung etc.) besitzt, vorgezogen. Die ereignisorientierte Simulation eignet sich insbesondere für die Simulation großer neuronaler Netze, da sich hier der Aufwand auf die Listenverwaltung der nächsten Ereignisse (z. B. die Erzeugung eines Aktionspotentials) beschränkt. Im zeitbasierten Simulator müssen alle Initialbedingungen aller Elemente eines Netzes zu jedem Zeitpunkt vorgehalten werden und die Integrationsschritte so klein gewählt werden, dass der Fehler bei der Integration minimal wird. Für die in diesem Entwurf betrachteten kleinen Netzwerke mit maximal 1000 Neuronen eignen sich daher zeitbasierte Simulationen. Zudem kann mit einer zeitbasierten Simulation die Echtzeitfähigkeit des Systems ohne Geschwindigkeitseinbußen, wie sie bei ereignisorientierten Netzen entstehen, garantiert werden. Voraussetzung ist natürlich die Möglichkeit der Abbildung des Netzes auf eine Hardware, welche die Echtzeitfähigkeit ermöglicht. Folgt man dem Signalpfad bis zur Membran, wird die Baumstruktur des Ansatzes deutlich.

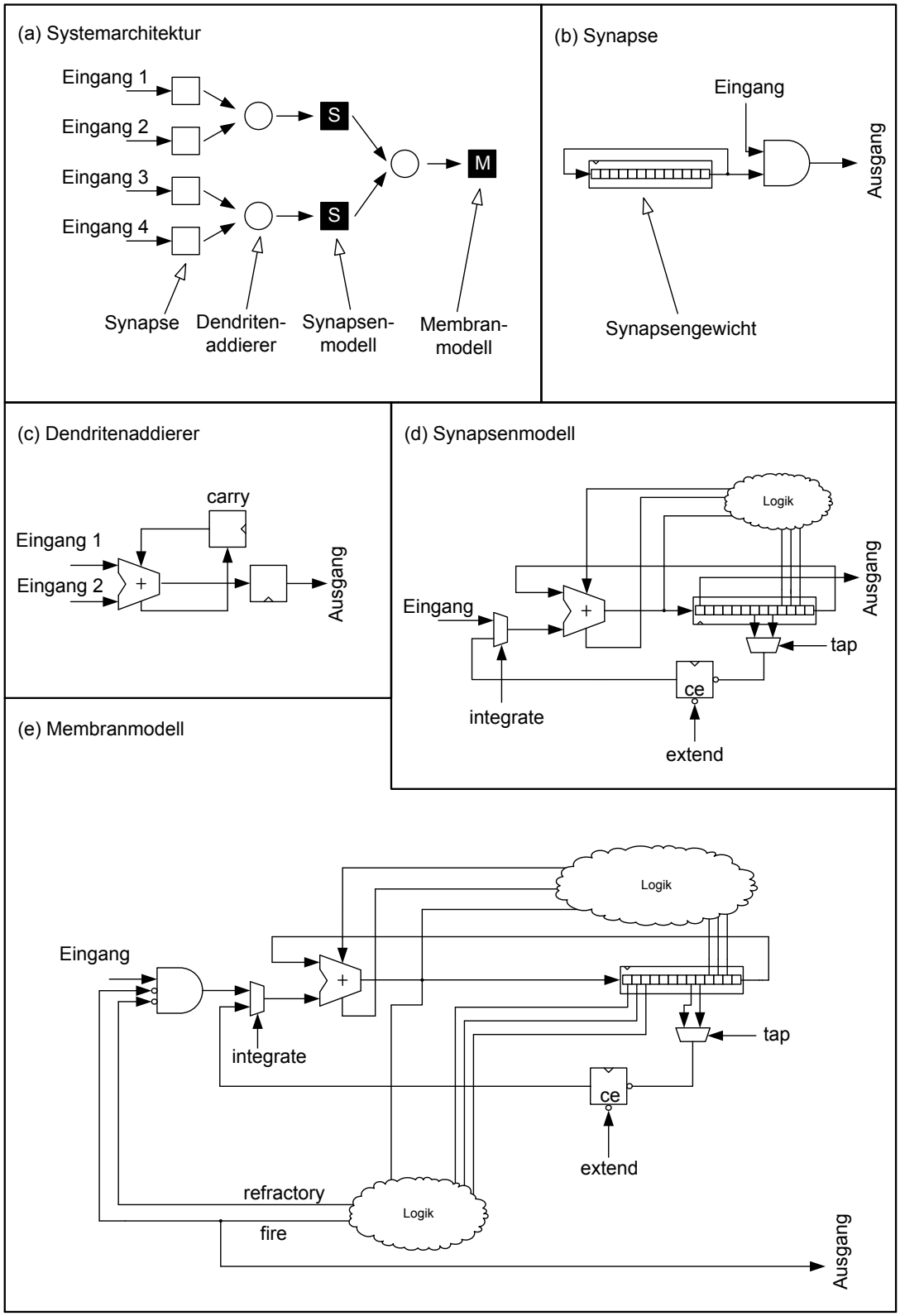


Abbildung 2.4: Vereinfachte Darstellung der Systemarchitektur und Systemkomponenten der Schrauwen-Implementierung (nach [87]).

Die Synapsenkomponenten bilden dabei die Blätter des Baums. Präsynaptische Pulse führen zur Freigabe der synaptischen Gewichte, die in der Synapse in einem Schieberegister gespeichert sind (Abb. 2.4b). Besitzen mehrere Synapsen dasselbe Modell, so bilden sie einen Teilbaum mit der Synapsenmodellkomponente als Wurzel. Die Zusammenführung der parallelen Signalpfade wird über das Dendritenmodell erreicht, welches aus einem Volladdierer (Abb. 2.4c) für die bitseriellen Signale besteht. Für die Realisierung des Synapsenmodells besteht die Synapsenmodellkomponente (Abb. 2.4d) aus einem Volladdierer mit nachgeschaltetem Schieberegister. Der Ausgang des Schieberegisters ist mit dem Eingang des Volladdierers verbunden. Mit dieser Verschaltung wird die Addition von einlaufenden Synapsengewichten mit dem internen Potential durchgeführt. Für den exponentiellen Zerfall des internen Synapsenpotentials wird das um einen festen Faktor verschobene Synapsenpotential als Subtrahend genutzt. Da die Werte im Zweierkomplement vorliegen, kann diese Subtraktion mit einem geringen Mehraufwand mit dem vorhandenen Addierer ausgeführt werden. Die Anzahl der Subtraktionen und die Verschiebungsanzahl des Subtrahenden können in diesem Modell frei gewählt werden. Diese Modellimplementierung findet sich in der Membrankomponente (Abb. 2.4e) wieder, in der die Integration und der Zerfall des Potentials in ähnlicher Weise implementiert sind. Zusätzliche Logik überprüft in der Membrankomponente das Erreichen der Potentialschwelle und generiert einen Puls als Aktionspotential. Anschließend wird das Membranpotential auf einen Initialwert zurückgesetzt und der Zustand des Neurons auf Refraktion gesetzt.

Die veröffentlichten Syntheseergebnisse der hier beschriebenen Struktur können aus der Tabelle 2.1 entnommen werden. Eine Besonderheit dieser Implementierung ist die Gestaltungsmöglichkeit komplexerer Modelle durch den modularen Systemaufbau. Es können Modelle, die sich aus der Kombination von exponentiellen Funktionen nachbilden lassen, realisiert werden. Als Anwendung werden die Steuerung von autonomen Robotern, Liquid-State-Machine-basierte Spracherkennung und eingebettete lernende Prozessoren angegeben.

2.2.2 Die Upegui-Implementierung

Die digitale Implementierung von Upegui [96] besteht im Wesentlichen aus dem detailliert in [94, 95] beschriebenen Neuron für die Implementierung auf rekonfigurierbaren FPGA. Neben dem weiter unten beschriebenen typischen Aufbau eines parallel arbeitenden digitalen Neurons betrifft eine Besonderheit in dieser Umsetzung die Behandlung von zwei simultan eintreffenden Pulsen an der Synapse. Im Gegensatz zu den SRM- oder LIAF-Modellen findet keine lineare Superposition der postsynaptischen Antworten statt, sondern eine einfache Addition der Synapsengewichte. Alle neuronalen Aktivitäten finden in einer zentralen Komponente statt. Die Synapsengewichte sind in der Neuronenkomponente gespeichert. Als Speicher dient eine Look-Up Table (LUT), deren Größe von der Anzahl der Synapseneingänge eines Neurons abhängig ist. Zusätzlich sind die Werte für die postsynaptische Antwort des Potentials und der Verlauf des Potentials nach einem ausgelösten Aktionspotential in der LUT gespeichert. Ein Neuron mit 30

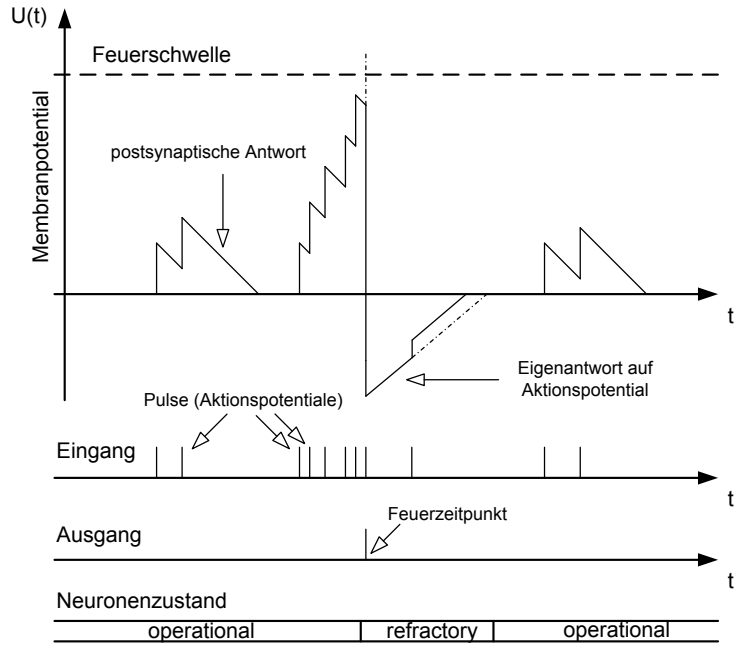


Abbildung 2.5: Beispiel des Verlaufs eines Membranpotentials für die Implementierung nach Upegui [96].

Eingängen und einer Datenbreite von 9 Bit benötigt eine 32×9 Bit große LUT, in der zusätzlich zu den Gewichten der Eingänge des Neurons die Steilheit der Zunahme oder des Abfalls des Membranpotentials festgehalten sind. Die zwei Zustände des Neurons *operational* und *refractory* werden mit einem Mooreschen Zustandsautomaten beschrieben. Im Zustand *operational* werden die Synapsengewichte zum internen Membranpotential addiert, wenn ein präsynaptischer Puls den Eingang erreicht (siehe Abb. 2.5). Bei Erreichen der Feuerschwelle wird am Ausgang des Neurons ein Puls generiert und das Neuron wechselt in den *refractory* Zustand, der das Verhalten eines Neurons während der Refraktärzeit beschreibt. Dabei wird das Membranpotential auf ein Potential unterhalb des Ruhepotentials zurückgesetzt. Solange das Membranpotential das Ruhepotential nicht wieder erreicht hat, werden Synapsengewichte nur stark gedämpft zum internen Potential addiert. Nach Erreichen des Ruhepotentials wechselt der Neuronenzustand wieder zum Zustand *operational*. Die Systemarchitektur besteht aus zu Blöcken gruppierten Neuronen. Innerhalb eines Blocks sind alle Neuronen miteinander verbunden. Weiter besitzt jedes Neuron Verbindungen zu den Neuronen im Vorgänger- und im Nachfolgerblock. Die Blockorganisation eignet sich für die dynamische Rekonfigurierung der Netzstruktur auf FPGAs. Dedizierte Kommunikationspunkte zwischen den Blöcken ermöglichen das einfache Hinzufügen von Blöcken in das bestehende Netzwerk zur Laufzeit.

Diese Implementierung zeichnet sich durch ihre kompakte Größe aus, die abhängig von der Anzahl der Eingänge des Neurons ist. Durch ihre Architektur mit zu Blöcken zusammengefassten Neuronen erlaubt sie im Zusammenspiel mit der dynamischen Rekonfigurierbarkeit der Zielplattform einen einfachen Austausch der Neuronen und eine Anpassung der Netztopologie im Betrieb.

2.2.3 Die Torres-Huitzil-Implementierung

Für die Bildverarbeitung haben sich lokal verbundene Neuronenfelder als besonders geeignet herausgestellt [99]. Das LEGION (Locally Excitatory Globally Inhibitory Oscillator Network) ist ein Feld mit oszillierenden Neuronen, in dem die Neuronen mit ihren unmittelbaren Nachbarn über exzitatorische Synapsen miteinander verbunden sind. Neuronenfelder, welche lokal exzitatorisch miteinander verbunden sind, zeigen Synchronisationseffekte und eine wellenartige Ausbreitung der Aktionspotentiale über das Neuronenfeld [43].

In dieser Implementierung sorgt ein globaler Inhibitor für ein Abklingen der sich wellenförmig im Neuronenfeld ausbreitenden Erregung (Desynchronisation der Neurone durch inhibitorische Pulse). Der Bereich V1 des visuellen Kortex im menschlichen Gehirn dient als biologisches Vorbild dieser Struktur. Bei Systemen dieser Art wird häufig ein aufgenommenes Bild auf darunter liegende Neurone mit lokaler Verschaltung abgebildet. Gemeinsame im Bild detektierte Merkmale, z. B. zusammenhängende Flächen ähnlicher Helligkeit oder Kanten mit bestimmter Ausrichtung werden durch eine synchrone Feueraktivität der beteiligten Neurone repräsentiert. Eine Beschreibung der Möglichkeiten dieser Systeme, wenn auch am Beispiel eines analogen Entwurfs findet sich in [44, 45].

Die Implementierung eines LEGION auf einem FPGA wird in der Arbeit von Torres-Huitzil [92] vorgestellt. Als Applikation wird in dieser Veröffentlichung die Bildsegmentierung aufgenommener Kamerabilder für einen autonomen Roboter vorgestellt. Das Neuronenmodell eines LEGION-Neurons wird mit der Differentialgleichung

$$\dot{x}_i = -x_i + I_i + \sum_{j \in N(i)} \frac{\alpha_j}{Z_i} P_j - G \tag{2.3}$$

beschrieben. Gleichung (2.3) beschreibt die Dynamik eines oszillierenden LIAF-Neurons i mit dem Membranpotential x_i, einem externen Stimulus I_i und mit der Summe der Erregungspulse P_j, die von der oszillierenden Nachbarschaft $N(i)$ erzeugt werden. Die Konstante α_j beschreibt die Verbindungsstärke der Nachbarn von i, welche durch die Summe der Nachbarn Z_i des Neurons geteilt wird. Das Neuron besteht aus vier Komponenten und besitzt Eingänge für die benachbarten Bildpunkte und Erregungsimpulse, einen Eingang für inhibitorische Pulse und weitere Eingänge zum Einstellen interner Parameter. Zur Bildsegmentierung eines Graustufenbildes werden die Helligkeitswerte der Bildpunkte als erregende externe Stimuli I_i auf die Neurone des LEGION abgebildet.

Ein Pixeldifferenztest ermittelt den Homogenitätsgrad der Umgebung von Neuron i. Dazu werden alle benachbarten Pixel j des aktuell ausgewählten Bildpunkts respektive Neurons betrachtet und die Differenz zum aktuellen Bildpunkt bestimmt. Sollte die Differenz $|p_i - p_j|$ des dem Neuron zugeordneten Bildpunktes p_i und des benachbarten Bildpunktes p_j unterhalb einer definierten Schwelle liegen, ist der Pixeldifferenztest für die zwei Bildpunktpaare bestanden. Die serielle Implementierung ist in Abbildung 2.6 dargestellt. Mit Hilfe des Multiplexers wird der zu testende benachbarte Bildpunkt selektiert. Die anschließende Subtraktionseinheit und der Komparator bestimmen die Differenz und

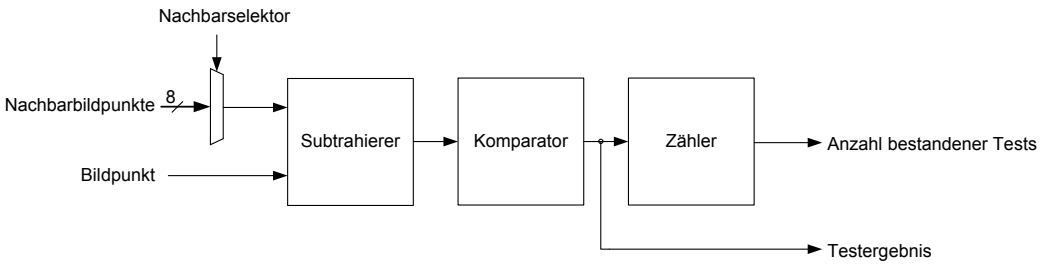


Abbildung 2.6: Pixeldifferenztestkomponente der Implementierung nach Torres-Huitzil [92].

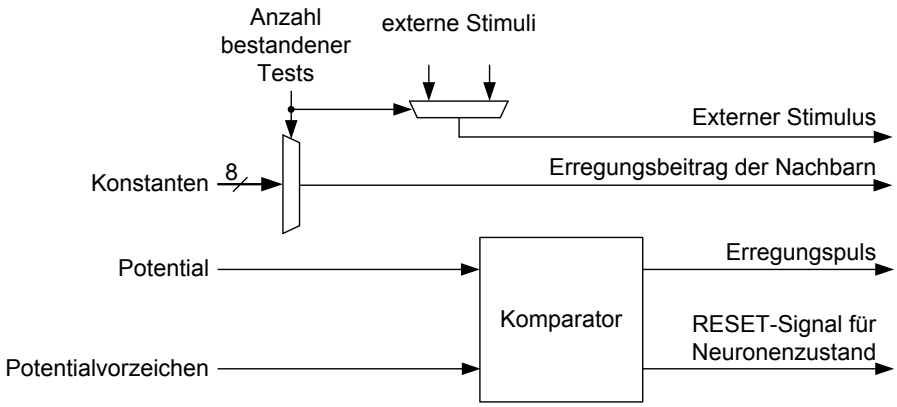


Abbildung 2.7: Komponente zur Bestimmung der Erregung des Zentralneurons in der Torres-Huitzil-Implementierung (nach [92]).

überprüfen die Homogenitäts-Bedingung. Der Zähler bestimmt die Anzahl der bestandenen Tests. Für eine weitere Auswertung werden die Anzahl der Nachbarneuronen und die Anzahl der bestandenen Tests an eine Komponente zur Bestimmung der Erregung des Zentralneurons weitergegeben.

Die Komponente zur Bestimmung der Erregung des Zentralneurons (Abb. 2.7) erzeugt exzitatorische Pulse für das Zentralneuron, indem der Beitrag der benachbarten Neurone zum Membranpotential des Zentralneurons sowie das eigene Membranpotential des Zentralneurons berücksichtigt werden. Weiter werden die externen Stimuli verarbeitet und der Erregung des Zentralneurons hinzugefügt, und das RESET-Signal für die Zurücksetzung des Membranpotentials erzeugt. Die Bedingung für die Erzeugung eines Erregungspulses und der Erzeugung des RESET-Signals wird mit dem Komparator überprüft. Die Kontributionsstärke der Nachbarpulse ist abhängig von der Anzahl der bestandenen Tests und wird als Konstante aus einem angeschlossenen Speicher ausgelesen. Ist die Anzahl der bestandenen Tests n größer als die Hälfte der Nachbaranzahl ($n > \frac{1}{2}|N(i)|$), wird der externe Stimulus auf das Zentralneuron mit $I_i = 1,25$ festgesetzt. Wenn kein Nachbarschaftstest erfolgreich war, ist $I_i = 0$. Für alle anderen Fälle wird der externe Stimulus mit $I_i = 0,95$ festgesetzt und das Membranpotential des Zentralneurons in die Nähe der Feuerschwelle gebracht. In diesem Zustand reicht eine geringe Anzahl und Stärke an Pulsen von benachbarten Neuronen aus, um das Zentralneuron zum Feuern zu bringen.

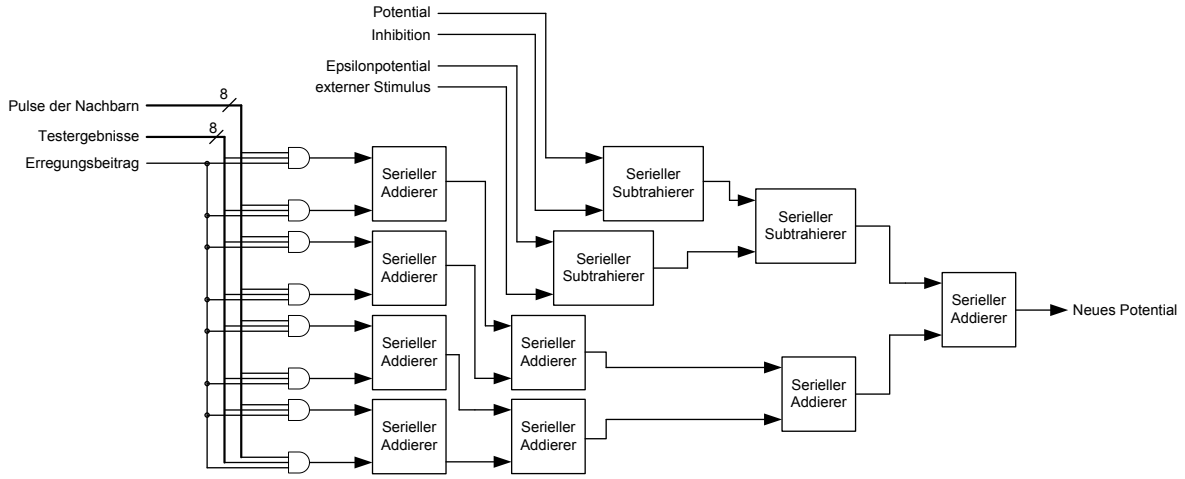


Abbildung 2.8: Arithmetikkomponente der Torres-Huitzil-Implementierung (nach [92]).

Die Arithmetikkomponente (Abb. 2.8) berechnet das dynamische Verhalten des Neurons nach Gleichung (2.3). Für die numerische Lösung der Differentialgleichung wird die Eulersche Methode mit einer Integrationsschrittweite von $\varepsilon = 0.25$ genutzt. Der serielle Addiererbaum berechnet den Beitrag der Nachbarimpulse zum Membranpotential des Zentralneurons. Die Beiträge der inhibitorischen Pulse und des externen Stimulus werden mit dem Subtrahiererbaum bestimmt und vom Membranpotential des Zentralneurons abgezogen.

Die entstandene Implementierung nutzt die vom genutzten FPGA zur Verfügung gestellten Ressourcen und deren zweidimensionale Anordnung. Die Multiplikationen und Divisionen wurden über Lookup-Tables realisiert. Für die Berechnung des Neuronenverhaltens wird serielle Arithmetik mit einer 12 Bit Festkommadarstellung im Zweierkomplementsystem verwendet. Zwei Bit kennzeichnen die Ganzzahlstellen und zehn Bit die Nachkommastellen.

2.2.4 Die Johnston-Implementierung

Eine Vorgehensweise mit einem auf FPGA abgestimmten Entwurfsablauf wird in [57] vorgestellt. Im Rahmen eines Vergleichs von Neuronen zweier klassischer Neuronaler Netze (Multilayer Perceptron und Radial Basis Function Network) und zwei pulscodierten Neuronenmodellen (LIAF und SRM) werden einfache neuronale Netze nach einer Evaluation in Matlab mit dem Xilinx System Generator (XSG) erzeugt. In der Evaluierungsphase wurde die Netzstruktur im Rahmen der Anlernphase bestimmt. Anschließend wurden die Neuronenmodelle mit Hilfe der Systemblöcke des XSG zusammengestellt. Das LIAF-Synapsenmodell wird mit einem Addierer, der die Synapsengewichte zum Membranpotential addiert, und einem Subtrahierer, der eine verschobene Version des Membranpotentials vom Membranpotential abzieht, realisiert. Die hier gewählte Subtraktion bildet den exponentiellen Abfall des Membranpotentials nach. Das SRM-Modell benötigt zusätzlich zu den Elementen des LIAF Neurons einen Multiplizierer, um die

Abhängigkeit der postsynaptischen Antwort von den präsynaptischen Pulsen zu realisieren. Die Syntheseergebnisse für Neurone und Synapsen mit einer Auflösung von 16 Bit sind in Tabelle 2.1 aufgeführt.

2.2.5 Die Maya-Implementierung

Ein Ansatz für ein bitserielles Neuron mit Verarbeitung stochastischer Pulsfolgen wird in der Implementierung von Maya [74] vorgestellt. Dabei wird ein Wert x für das Gewicht einer Synapse oder die Stärke eines Pulses durch eine pseudo-zufällige binäre Pulsfolge $x(n)$ dargestellt. Die Auftrittswahrscheinlichkeit einer 1 in $x(n)$ entspricht dem normalisierten Wert $\bar{x}$. Es gibt zwei Darstellungsformen für stochastische Modelle [105]. In der unipolaren Darstellungsform wird der reelle Wert $x \in [0, 1]$ durch eine Sequenz mit einer Auftrittswahrscheinlichkeit einer 1 von $p = x$ dargestellt. In dieser Darstellungsform wird für die Multiplikation von zwei Sequenzen nur ein AND-Gatter benötigt, eine Addition kann durch ein ODER-Gatter realisiert werden [15]. Die bipolare Darstellung bildet den Bereich $x \in [-1, 1]$ mit der Auftrittswahrscheinlichkeit einer 1 von $p = (x + 1)/2$ in eine Sequenz ab. Die Multiplikation kann in diesem Fall mit einem XOR-Gatter realisiert werden.

Das Modell für das in der Arbeit von Maya vorgestellte pulsende neuronale Netz besteht aus einem Modul für das Synapsenmodell und einem Modul für die Summation der einlaufenden Pulse sowie die Auswertung der Aktivierungsfunktion. Mit jedem eintreffenden Puls wird das Gewicht der aktivierten Synapse im Synapsenmodul akkumuliert. Tritt ein Überlauf des Zählers auf, wird abhängig vom Vorzeichen des Synapsengewichts ein exzitatorischer oder ein inhibitorischer Puls erzeugt und an das Summations-Aktivierungs-Modul übergeben. Das Summations-Aktivierungsmodul besteht aus einem exzitatorischen und einem inhibitorischen Schaltkreis, welche identisch aufgebaut sind. Die Schaltkreise dienen zur Zählung der eintreffenden exzitatorischen oder inhibitorischen Pulse. Ein Komparator entscheidet, welcher der beiden Zählerstände überwiegt und erzeugt einen ausgangsseitigen Puls, wenn der exzitatorische Anteil überwiegt.

2.2.6 Die Godin-Implementierung

Eine der wenigen digitalen ASIC-Implementierung wird in der Arbeit [35] vorgestellt. Implementiert wurde ein vorwärtsgerichtetes neuronales Netz bestehend aus dem Neuronenmodell SpikeCell. Das SpikeCell-Neuronenmodell erlaubt die Nachbildung von Neuronen mit beliebiger Aktivierungsfunktion $\varphi(h_i)$ mit Hilfe von pulsenden Neuronen. Der Eingang h_i der Aktivierungsfunktion muss linear vom Eingang x abhängig sein. Bei der Emulation wird der Pulscharakter der Implementierung selbst nicht aktiv für den Betrieb als pulsendes neuronales Netz genutzt. Zu den implementierten Modellen zählen Modelle mit Gaußfunktionen und Sigmoidfunktionen als nichtlineare Aktivierungsfunktionen. Distanzberechnende Modelle, wie das Radial Basis Function (RBF) Modell, werden von SpikeCell

nicht emuliert. Die Grundgleichungen eines statischen Neurons beschreiben den Ausgang y_i des Neurons i

$$y_i = \varphi(h_i) \tag{2.4}$$

und

$$h_i = \sum_j w_{ij} x_j \tag{2.5}$$

mit φ als Aktivierungsfunktion und h_i als gewichtete Summe der Eingangspulse x_j der präsynaptischen Neurone j mit den Verbindungsgewichten w_{ij}. Für eine flächeneffiziente Realisierung wurde die Multiplikation in Gleichung (2.5) durch die Beschränkung der Feuermenge eines Neurons auf die Pulswerte {-1,0,1} unnötig. Das SpikeCell-Modell kann stattdessen mit den Gleichungen

$$\delta_i^{(t+1)} = \begin{cases} -1, & \text{falls } \varphi(U_i^{(t)}) < \theta_i^{(t)} \\ 0, & \text{falls } \varphi(U_i^{(t)}) = \theta_i^{(t)} \\ 1, & \text{falls } \varphi(U_i^{(t)}) > \theta_i^{(t)} \end{cases} \tag{2.6}$$

als Bedingung für den Ausgangsimpuls,

$$\theta_i^{(t+1)} = \theta_i^{(t)} + d\delta_i^{(t+1)} \tag{2.7}$$

zur Berechnung der dynamischen Potentialschwelle, und

$$U_i^{(t+1)} = U_i^{(t)} + d\sum_j w_{ij}\delta_j^{(t+1)} \tag{2.8}$$

zur Berechnung des Membranpotentials beschrieben werden. Das neuronale Netz arbeitet dabei in einem Zyklus bestehend aus zwei Zeitschritten. Im ersten Zeitschritt werden anhand der Fallunterscheidung (2.6) die zu sendenden Pulse $\delta_i^{(t+1)}$ für jedes Neuron ermittelt. Dabei kennzeichnen $U_i^{(t)}$ das Membranpotential und $\theta_i^{(t)}$ die dynamische Potentialschwelle. Für die Kodierung der drei möglichen Pulswerte werden zwei Bit genutzt. Im zweiten Zeitschritt werden die Potentialschwelle (2.7) und das Membranpotential (2.8) des Neurons mit den im ersten Zeitschritt bestimmten Pulsen berechnet. Die Diskretisierungsschrittweite $\{d \mid 0 < d \leq 1\}$ ist ein Parameter für die Genauigkeit der Emulation und wird mit $d = \frac{1}{2^k}$ gewählt, wobei k die implementierte Wortbreite darstellt. Die Pulse vom präsynaptischen Neuron j werden mit δ_j beschrieben.

Diese Implementierung stellt ein einfaches Neuronenmodell dar, mit dem statische Neuronenmodelle emuliert werden können. Als Resultat erhält man ein Neuron, das nur ein

Tabelle 2.1: Übersicht über die Syntheseergebnisse der vorgestellten Implementierungen.

Referenz	Komponente	Anzahl Slices	Bemerkungen
Upegui [96]		FPGA:(XC2S200)	
	Neuron	17	14 Eingänge
	Neuron	23	30 Eingänge
	Neuron	46	60 Eingänge
Torres-Huitzil [92]		FPGA:(XC2V1500)	
	Neuron	41	8 Nachbarneurone
Johnston [57]		FPGA:(XC2V4000)	
	LIF Synapse	35	
	SRM Synapse	195	
	Soma	20	
Maya [74]		FPGA:(XV50)	
	Synapse	13	
	Neuron	47	

Referenz	Komponente	Anzahl LUTs	Bemerkungen
Schrauwen [87]		FPGA:(XC3S200) und (XC4VLX100)	
	Neuron	N(22+3I+10S)	N: Anzahl Neuronen I: Anzahl Eingänge S: Anzahl Synapsenmodelle
Torres-Huitzil [92]		FPGA:(XC2V1500)	
	Neuron	64	8 Nachbarneurone

Implementierung	Anzahl Zellen	Größe(mm^2)	Bemerkungen
Godin [35]	10	0,14	0,25 μm SOI-Technologie

Zehntel der Fläche eines mit Multiplizierern implementierten Neurons, aber eine sechsfache Berechnungszeit pro Rechenschritt benötigt.

In Tabelle 2.1 sind die wesentlichen Merkmale der hier vorgestellten digitalen Implementierungsvarianten noch einmal zusammengefasst. Da sich in der Literatur praktisch keine Angaben zur Leistungsaufnahme digitaler Neurone finden lassen, wurde in der Tabelle nur die benötigte Fläche in Form von Slices, Flipflops und Lookup-Tabellen angegeben.

2.3 Analoge Implementierungen

Neben der im letzten Abschnitt gezeigten Umsetzung der pulscodierten neuronalen Netze mittels einer digitalen Standardzellenbibliothek auf einen ASIC werden PCNN auch als

analoge Schaltkreise umgesetzt. Dabei kann eine im Vergleich zur digitalen Umsetzung große Flächenersparnis erzielt werden. Dem gegenüber stehen mit in kleiner werdenden Halbleiter-Technologien zunehmende Leckströme, die die längere Speicherung von Ladung auf integrierten Kapazitäten erschweren bzw. zu kleinen Zeitkonstanten führen. Diese Probleme müssen bei der Implementierung von PCNN in analoger Schaltungstechnik mit aktuellen Strukturgrößen von 130 nm und darunter besonders betrachtet werden. Im folgenden Abschnitt werden einige ausgewählte, überwiegend mit analoger Schaltungstechnik implementierte Neuronenmodelle gezeigt, um die später in dieser Arbeit entworfenen Umsetzungen einordnen zu können.

2.3.1 Die Matolin-Implementierung

In [73] implementiert Matolin ein einfaches pulsendes LIAF-Neuron. Die Schwellenüberschreitung des Membran-Potentials wird durch einen Schmitt-Trigger mit einstellbaren Trigger-Schwellen detektiert, die Membran-Entladezeit ist über eine Referenzspannung an einem als einstellbarer Leitwert genutzten Transistor festgelegt. In der Publikation wird ein neuronales Netz mit 64×64 Neuronen in einer $0,35\,\mu$m CMOS-Technologie realisiert, wobei jeweils benachbarte Neurone durch eine Synapse verknüpft sind. Das gesamte neuronale Netz zur Detektion zusammenhängender Flächen in einem aufgenommenen Kamerabild benötigt eine Fläche von $4,7\,\text{mm} \times 5,4\,\text{mm}$. Diese Arbeit ist im Rahmen des Eingangs bereits angesprochenen VisionIC Projekts [3] entstanden.

2.3.2 Die van Schaik-Implementierung

Bei dem in [97] von A. van Schaik vorgestellten Neuron wird ein Differenzverstärker zur Schwellenüberschreitung des Membran-Potentials verwendet. Über insgesamt sieben Stromquellen kann das Verhalten des Neurons parametrisiert werden. Zwei der Stromquellen dienen dabei zur Nachbildung der Ströme von Kalium- und Natrium-Ionen bei biologischen Neuronen. Daneben sind weitere Parameter, wie die Refraktärzeit des Neurons einstellbar. Die Umsetzung von 32 Neuronen auf einem ASIC benötigen nebst zusätzlicher Kommunikations-Schaltkreise eine Chipfläche von $1\,\text{mm} \times 2,5\,\text{mm}$ in einer $1\,\mu$m CMOS-Technologie.

2.3.3 Die Indiveri-Implementierung

In [54] stellt G. Indiveri ein LIAF-Neuron vor, das in Anlehnung an biologische Neurone eine Pulsraten-Adaption, eine einstellbare Refraktärphase sowie eine einstellbare Feuerschwelle bietet. Die Pulsraten-Adaption dient dazu, die Pulsrate bei permanenter starker eingangsseitiger Erregung zu drosseln. Bei biologischen Neuronen ist in diesem Fall eine ansteigende Konzentration von Calcium-Ionen für eine Verringerung der Feuerrate verantwortlich. Die Refraktärphase ist die Zeitspanne nach der Erzeugung eines Ausgangspulses, während der kein erneutes Feuern des Neurons möglich ist. Diese Phase tritt

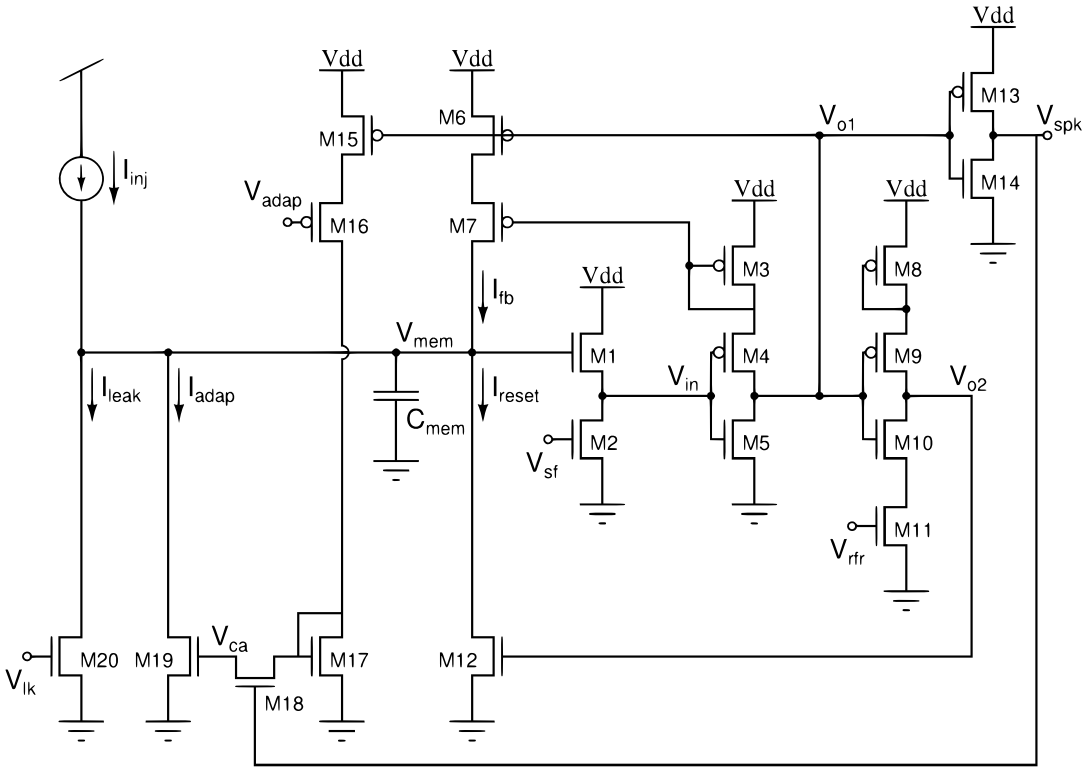


Abbildung 2.9: Integrate-and-Fire Neuron nach Indiveri [54].

auch bei biologischen Neuronen auf. Die Überschreitung der Feuerschwelle wird in dem LIAF-Neuron durch einen Inverter (Transistoren M4 und M5) mit positiver Rückkopplung über M7 auf das Membranpotential in Verbindung mit einem im Subschwellenbereich arbeitenden Source-Folger (Transistoren M1 und M2) detektiert (siehe Abb. 2.9). Durch die positive Rückkopplung wird ein schnelles Schalten des Inverters erreicht und damit die Verlustleistung der Schaltung minimiert. Diese Schaltung wurde angelehnt an den Axonhügel eines Neurons als *Axon-Hillock*-Schaltung bekannt. Die Dauer der Refraktärphase wird über die einstellbare Slew-Rate eines weiteren Inverters (M8–M11) gesteuert. Ein Stromspiegel-Integrator (M15–M19) realisiert die Pulsraten-Adaption. Dabei wird von dem in das Neuron injizierten Strom I_{inj} ein Adaptionsstrom I_{adap} subtrahiert, der mit jedem erzeugten Ausgangspuls erhöht wird, so dass die durchschnittliche Pulsrate mit der Zeit abnimmt und die Aktivität des Neurons begrenzt wird. Bei sich verringernder Erregung des Neurons geht auch die Adaption aufgrund von Leckströmen innerhalb des Stromspiegel-Integrators zurück. Für typische Betriebsbedingungen wird, bezogen auf eine $1,5\,\mu$m CMOS-Technologie, eine mittlere Verlustleistung von 300 nW sowie eine maximale Verlustleistung von $1,5\,\mu$W bei einer Feuerrate von 100 Hz angegeben. Dieses führt zu einem umgerechneten Energiebedarf von 3 nJ bis 15 nJ pro Aktionspotential. Weitere Untersuchungen der Eigenschaften, nicht jedoch Angaben über die Leistungsaufnahme des Neurons sind in [82] angegeben.

Eine der hier beschriebenen Schaltung ähnliche Schaltung wird in einer späteren Veröffentlichung von Indiveri angegeben [55]. Gegenüber der Schaltung aus [54] wurde die

Ausgangsschaltung so modifiziert, dass das *Address-Event Representation*-Protokoll (AER) unterstützt wird. Hierzu kommuniziert das Neuron über asynchrone Request- sowie eine Acknowledge-Leitung mit übergeordneten Schaltungsteilen. Für ein Neuron, implementiert in einem $0,35\,\mu$m Prozess, wird für optimale Betriebsbedingungen eine Energie von 900 pJ pro Ausgangspuls angegeben. Unter typischen Bedingungen beträgt die Verlustleistung je nach gewählter Triggerschwelle zwischen $10\,\mu$W und $120\,\mu$W. In einem $0,8\,\mu$m CMOS-Prozess benötigt eine Schaltung aus 32 Neuronen sowie 256 Synapsen eine Fläche von $1,6\,\text{mm}^2$. Für einen $0,35\,\mu$m Prozess wird für 32 Neuronen und 8000 Synapsen eine Fläche von weniger als $10\,\text{mm}^2$ angegeben.

Ähnliche Eigenschaften wie die Schaltung aus [54] weist auch das Neuron nach Liu [68] auf. Bei dieser Implementierung werden einzelne Module der Schaltung anders als bei Indiveri realisiert. Statt des Einsatzes eines Inverters als Schwellenspannungselement wird in der Arbeit von Liu ein Differenzverstärker verwendet.

Eine weitere ähnliche Implementierung eines pulscodierten Neurons wurde von Schulz und Jabri [88] publiziert. Die Besonderheit dieser Implementierung ist eine erste getrennte Modellierung von Membranpotential und Aktionspotential. Hierdurch wurde es möglich, das Aktionspotential, dessen Auftreten und dessen Form für die Anwendung physiologisch motivierter synaptischer Lernverfahren notwendig sind, unabhängig von der Behandlung des Membranpotentials zu machen und schafft so die Möglichkeit, das Aktionspotential in anderer Weise zu modellieren.

2.3.4 Die Chicca-Implementierung

In einer Publikation von Chicca, Badoni et al. [20] wird ebenfalls das Prinzip des Inverters als Schwellenelement genutzt, bei dem ein zusätzlicher Inverter und Koppelkapazität zurück auf den Eingang des Schwellenelements für eine Hysterese sorgen (Axon-Hillock Schaltung). Das Schaltbild des gesamten Neurons ist in Abb. 2.10 dargestellt und zeigt zusätzlich den aktiven Entlade-Pfad über die Transistoren M5 und M6, der bei Aussenden eines Aktionspotentials für eine schnelle Abnahme des Membranpotentials sorgt. Durch eine Referenzspannung an Transistor M5 wird die Dauer des Aktionspotentials eingestellt. Daneben existiert ein Zustands-Block, der in weiteren Schaltungsteilen ein Lernverfahren steuert. Der Eingang des Neurons ist mit zwei Stromquellen modelliert, welche bei Auftreten eines präsynaptischen Aktionspotentials für einen exzitatorischen oder einen inhibitorischen Eingangsstrom am empfangenden Neuron sorgen können.

In der Umsetzung eines Chips mit einer Fläche von $3,16\,\text{mm} \times 3,16\,\text{mm}$ wurden in der $0,6\,\mu$m Technologie 21 Neurone und 129 Synapsen aufgebaut. Trotzt detaillierter Beschreibung der Architektur werden in dieser Publikation keine Angaben zur Leistungsaufnahme des Neurons gemacht.

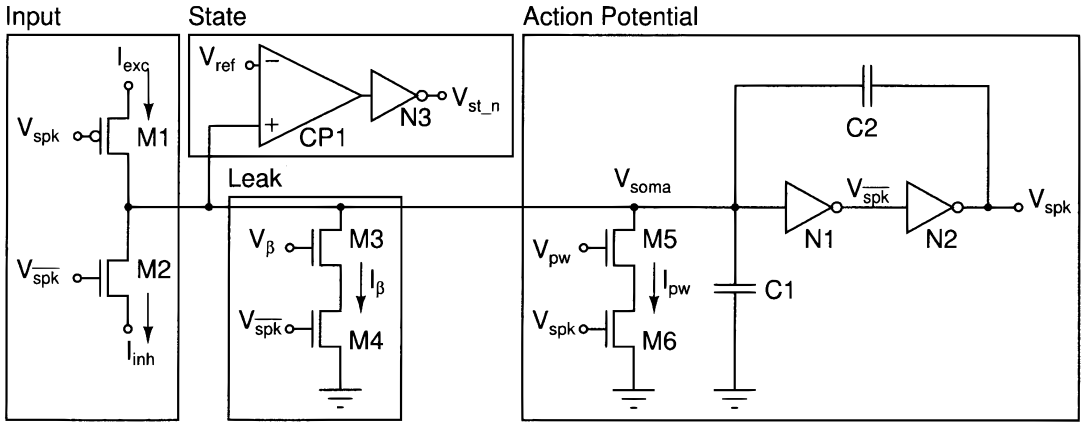


Abbildung 2.10: Integrate-and-Fire Neuron nach Chicca [20].

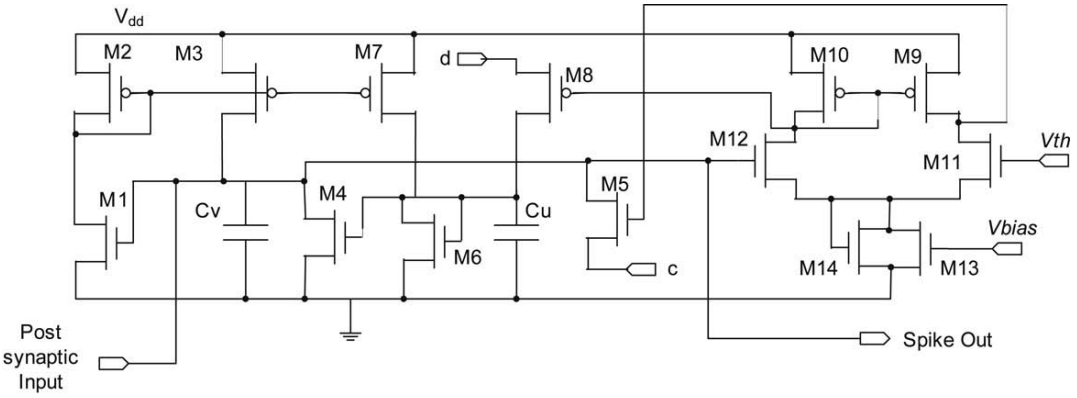


Abbildung 2.11: Integrate-and-Fire Neuron nach Wijekoon et al. [102].

2.3.5 Die Wijekoon-Implementierung

Eines der kleinsten und flexibelsten Neuronen ist von Wijekoon und Dudek [102] publiziert worden. Hier ist ein LIAF Neuron mit 14 Transistoren implementiert worden, welches auf dem einfachen Modell von Izhikevich [56] basiert. Izhikevich veröffentlichte eine einfache Beschreibung eines LIAF Neurons als zweidimensionales System, dessen Verhalten mit frei wählbaren Parametern gesteuert werden kann und verschiedene Klassen von Neuronen abbilden kann. Die von Wijekoon veröffentlichte Implementierung nutzt einen 350 nm Prozess und besteht aus einem Stromspiegel-Block (M1–M5) zur Bildung des Membranpotentials, einem zusätzlichen *Slow Variable*-Schaltkreis (M6–M8), der zur Nachbildung des in [56] beschriebenen Verhaltens notwendig ist, und einem Differenzverstärker (M9–M13) als Schwellwertelement (Abb. 2.11). Durch die Verwendung von Transistoren mit großen Abmessungen und Kapazitäten mit einer Kapazität von 1 pF belegt ein einzelnes Neuron eine Fläche von 2800 μm^2. Diese Implementierung arbeitet in einem Zeitbereich von Mikrosekunden für die Erzeugung eines Aktionspotentials (statt Millisekunden), welches eine Folge des Umgangs mit den in kleineren Technologien zunehmenden Leckströmen ist. Wijekoon bemerkt treffend, dass in der verfügbaren Literatur mit Ausnahme der Veröffentlichung von Indiveri 2003 [54] praktisch keine Angabe zur Leistungsaufnahme

der Neurone gemacht wird. Selbst gibt Wijekoon für die Implementierung des Neurons eine Energie von 8,5 pJ bis 9 pJ pro Aktionspotential an und schlägt vor, diese Angabe zum Vergleich verschiedener Neuronen-Implementierungen zu nutzen.

Tabelle 2.2 zeigt zusammengefasst die den aufgeführten Publikationen analoger Implementierungen von Neuronen entnommenen Daten. Eine Übersicht über den Energiebedarf der Implementierungen, normiert auf die Energie pro ausgesendetem Aktionspotential sowie der Energiebedarf biologischer Neurone sind in Tabelle 2.3 gegeben.

2.4 Vergleich bestehender Implementierungsvarianten

Die in diesem Kapitel betrachteten Realisierungsvarianten von digital und analog implementierten Neuronen stellen nur einen Ausschnitt der tatsächlich publizierten Varianten dar. Dennoch gibt dieses Kapitel einen Überblick und nennt die wichtigsten Referenzen. So sind in den letzten Jahren die Arbeiten von Schrauwen, u. A. [87] als Stand der Dinge in der Umsetzung digitaler pulscodierter neuronaler Netze zu sehen. Der in diesen Arbeiten durch die parallele Verarbeitung entstehende Flächenbedarf kann durch die Einführung von bitserieller Arbeitsweise, sei es ein stochastischer Ansatz, wie bei Maya [74], oder die Nutzung von bitseriellen Addierern und Multiplizierern, die in Kapitel 4.2 noch einmal aufgegriffen werden, stark reduziert werden. Durch die Nutzung von neuen Ansätzen wie z. B. speziellen Standardzellen-Bibliotheken für die Implementierung von digitalen Schaltungen auf ASICs mit besonders niedriger Versorgungsspannung kann der Energiebedarf der digitalen Neurone weiter gesenkt werden, was in späteren Kapiteln aufgegriffen wird.

Ein großer Vorteil digitaler Umsetzungen als ASIC ist, dass diese nur wenig den Prozess-Variationen während der Chipfertigung unterworfen sind. Dieser Einfluss äußert sich nur durch die Verlangsamung der Schaltungsteile, so dass die maximale Arbeitsfrequenz des digitalen Systems herabgesetzt wird. Dieses lässt sich durch ausreichende Reserven im Voraus bereits einplanen. Ein Nachteil der digitalen Umsetzungen ist jedoch, dass sie relativ viel Fläche in der ASIC-Umsetzung benötigen. Dieses wird auch noch in Kapitel 4 deutlich, in dem auch digitale Umsetzungen von LIAF Neuronen bewertet werden. Der Vergleich des Energiebedarfs der hier vorgestellten Implementierungen erweist sich als schwierig, da dieser unter anderem von der verwendeten Plattform (Typ des FPGAs oder ASICs) abhängig ist.

In Bereich der analogen Umsetzung pulscodierter Neurone in der CMOS-Technologie wurden nur ausgewählte Publikationen in die Betrachtung in diesem Kapitel aufgenommen, da der große Teil der Veröffentlichungen von Neuronen aus diesem Bereich älter als 10 Jahre ist, oder Bipolartransistoren nutzt. Die richtungsweisenden Implementierungen stammen von Indiveri [54, 55] mit der Axon-Hillock-Schaltung und Chicca [20], auch wenn hier nur wenige Angaben zu Flächenbedarf und Energiebedarf der Schaltungen zu finden sind. Die analogen Implementierungen beschränken sich jedoch meist auf CMOS-Technologien mit

Tabelle 2.2: Übersicht über vorgestellte analoge Implementierungen.

Referenz	Funktionen des Neurons	Implementierung	Fläche	Leistung
Matolin [73] (2004)	Detektion der Feuerschwelle: Schmitt-Trigger, einstellbare Feuerschwelle, einstellbare Refraktärphase	64x64 Neurone, je 4 Synapsen (350 nm)	$25,38\,\mathrm{mm}^2$	k. A.
van Schaik [97] (2001)	Detektion der Feuerschwelle: Differenzverstärker, Parameter über 7 Stromquellen einstellbar, davon bilden 2 Quellen Na^+ bzw. K^+-Ionen-Ströme nach, Membran-Leckentladung	32 Neurone mit Kommunikationsschaltkreisen ($1\,\mu\mathrm{m}$)	$2,5\,\mathrm{mm}^2$	k. A.
Indiveri [54] (2003)	Detektion der Feuerschwelle: Inverter mit positiver Rückkopplung, Pulsraten-Adaption, einstellbare Refraktärphase, einstellbare Feuerschwelle, Membran-Leckentladung	1 Neuron ($1,5\,\mu\mathrm{m}$)	k. A.	typ.: 300 nW max.: $1,5\,\mu$W
Indiveri et. al. [55] (2006)	wie zuvor, zusätzlich Unterstützung für AER-Protokoll	1 Neuron (350 nm)	k. A.	$10\,\mu$W–$120\,\mu$W
		32 Neurone, 256 Synapsen (800 nm)	$1,6\,\mathrm{mm}^2$	
		32 Neurone, 8000 Synapsen (800 nm)	$< 10\,\mathrm{mm}^2$	
Chicca et al. [20] (2003)	Detektion der Feuerschwelle: Rückgekoppelte Inverter	21 Neurone, 129 Synapsen ($0,6\,\mu\mathrm{m}$)	$10\,\mathrm{mm}^2$	k. A.
Kaulmann, Lütkemeier [112] (2007)	Detektion der Feuerschwelle: Differenzverstärker	1 Neuron, 5 Synapsen (130 nm)	$787\,\mu\mathrm{m}^2$	max. $24,16\,\mu$W Neuron: max. 675 nW Synapse: max. $4,697\,\mu$W
Wijekoon et al. [102] (2008)	Oszillierendes, zweidimensionales System nach [56], Detektion der Feuerschwelle: Differenzverstärker	1 Neuron (350 nm)	$2800\,\mu\mathrm{m}^2$	$8\,\mu$W–$40\,\mu$W

Tabelle 2.3: Energieumsatz technischer und biologischer Neuronen.

Referenz	Pulsrate [1/s][a]	Energie / Puls
Wijekoon [102]	10^6	8,5–9 pJ
Kaulmann [112]	$600 \cdot 10^3$	1,13 pJ
Indiveri [54]	100	3–15 nJ
Pyramidenzelle (Kortex) [8]		25,2–29,3 nJ[b]
Pyramidenzelle (Hippocampus) [8]		41,2 nJ[b]

[a] Die maximal erreichbare Pulsrate wurde zur Berechnung der Energie pro Puls herangezogen. Die Angaben der biologischen Neurone wurden ohne Berücksichtigung der Pulsrate in den angegebenen Publikationen ermittelt.

[b] Zur Erhaltung des Ruhepotentials wird eine aufzuwendende Energie von 26,1 pJ angegeben.

350 nm und darüber, in denen die Leckströme der heutigen Strukturgrößen von 130 nm und kleiner noch keinen nennenswerten Einfluss auf die Funktion der Schaltung haben. In Technologien von 90 nm und darunter wird sich dieser Trend weiter verstärken und die Leckströme durch neue Anteile, z. B. Gateströme verstärkt. Diese Einflüsse müssen zukünftig beim Entwurf der Schaltungen berücksichtigt werden. Die Umsetzung eines oszillierenden Modells von Wijekoon [102] nach dem Modell von Izhikevich [56] ist das derzeit flexibelste Modell eines LIAF Neurons und kann als Referenz für die weiteren Betrachtungen in dieser Arbeit dienen.

Um die in dieser Arbeit entwickelten Neuronen bezüglich ihres Flächenbedarfs und ihrer Leitungsaufnahme mit den referenzierten Publikationen vergleichen zu können, wurden die verfügbaren Daten und Eigenschaften der implementierten Neurone aus der einschlägigen Literatur in Tabelle 2.2 zusammengetragen. Da nur selten alle notwendigen Angaben zu Fläche und Verlustleistung veröffentlicht werden, ist die Übersicht unvollständig. Zum Vergleich der entwickelten Neurone eignen sich lediglich die Publikationen von Indiveri [54] und Wijekoon [102], auf deren Werte an geeigneter Stelle hingewiesen wird. Tabelle 2.3 enthält die auf die Energie für ein einzelnes Aktionspotential normierten Werte der vergleichbaren Publikationen. Die Angabe des Energieumsatzes in der Einheit J (Energie pro Puls), also bezogen auf die mittlere Pulsrate wird zum Vergleich in der gesamten Arbeit herangezogen.

Kapitel 3

Energetische Modellierung pulscodierter neuronaler Netze

In diesem Kapitel wird der Energiebedarf neuronaler Netze betrachtet. Dabei wird im Folgenden insbesondere die in technischen Systemen betrachtete Verlustleistung als beschreibende Größe verwendet. Ausgehend vom Umsatz biologischer Nervenzellen wird in späteren Kapiteln ein Vergleich mit digitalen synchronen und asynchronen sowie analogen Realisierungen pulscodierter neuronaler Netze möglich. Die in den technischen Realisierungen verwendeten Modelle wurden bereits in Kapitel 2.1 vorgestellt, die Mechanismen an der Zellmembran des biologischen Neurons wurden in Kap. 1.2 beschrieben und werden im Folgenden mathematisch gefasst.

Die von einem System aufgewendete Leistung lässt sich in einer ersten Betrachtung in zwei Teile trennen. Der erste Teil stellt die zur Aufrechterhaltung der grundlegenden Funktionen eines Systems notwendige minimale Leistung dar, welche im Folgenden mit dem Begriff Grundumsatz bezeichnet wird. Dieser betrifft in der biologischen Nervenzelle die Leistung, welche vor allem [5] von der Natrium-Kalium-Pumpe zum Aufrechterhalten des natürlichen Konzentrationsgefälles und zur Stabilisierung des intrazellulären Volumens erzeugt wird. Obwohl weitere Pumpen, z. B. die Calcium-Pumpe, mit einem geringen Energieumsatz an der Gesamtbilanz beteiligt sind, lässt sich die Funktion des Neurons, insbesondere das Aussenden eines Aktionspotentials, vollständig mit den Natrium- und Kalium-Kanälen beschreiben. Daher werden in dem in diesem Kapitel entwickelten Modell nur die wichtigsten Natrium- und Kalium-Kanäle betrachtet. Der Grundumsatz der biologischen Zelle wurde bereits in Kapitel 1.2.3 behandelt. In technischen Systemen kann der Grundumsatz als der Teil der Verlustleistung aufgefasst werden, der in digitalen Realisierungen im Ruhezustand, d. h. ohne Signalverarbeitung, durch das Schalten der Gatter in jedem Takt erzeugt wird. In analogen Realisierungen bezeichnet der Grundumsatz diejenige Leistung, die durch den Ausgleich von Leckströmen und Subschwellenströmen erzeugt wird (statische Verlustleistung).

Der zweite Teil der Leistung betrifft den Teil, der bei der Informationsverarbeitung und Informationsübertragung erzeugt wird. Dieser wird im Folgenden als Informationsum-

satz bezeichnet. Der Informationsumsatz kennzeichnet in biologischen Nervenzellen die Leistung, die durch das Erzeugen eines Aktionspotentials und den damit verbundenen aktiven und passiven Ionen-Ausgleichsvorgängen der Zelle entsteht. In analogen sowie digitalen Systemen beschreibt dieser Begriff die zusätzliche Verlustleistung, welche durch die Signalverarbeitung und Erzeugung eines Pulses in der Zelle entsteht (dynamische Verlustleistung).

$$P_{\text{Gesamt}} = P_{\text{Grundumsatz}} + P_{\text{Informationsumsatz}} \tag{3.1}$$

Die in technischen Realisierungen auftretende Verlustleistung durch Leckströme und Sperrströme wird als Grundumsatz betrachtet. Da sie jedoch bei immer kleiner werdenden Strukturen einen großen Anteil an der gesamten Verlustleistung annehmen kann, wird an entsprechender Stelle auf die jeweils abgeschätzte oder gemessene Größe verwiesen.

3.1 Modellierung des Energieumsatzes: Biophysikalisches Grundmodell

Das in diesem Kapitel entworfene biophysikalische Grundmodell betrachtet den Energieumsatz an der Membran einer Nervenzelle. Dabei spielen verschiedene Transportmechanismen eine Rolle, deren Beitrag zum Energieumsatz in diesem Kapitel ermittelt werden soll. Ursache für den Energieumsatz ist die Änderung der Konzentration vor allem der Kalium- und Natrium-Ionen. Die Konzentrationsänderung wird im Ruhezustand des Neurons durch einen passiven Transport der genannten Ionen durch ionenspezifische Kanäle hervorgerufen. Dem entstehenden Ungleichgewicht der Konzentration in Ruhe, vor allem aber der Änderung des Zellvolumens wird durch einen aktiven Pump-Prozess entgegengewirkt. Dieser Pump-Prozess arbeitet unter Umsatz von Energie. Während der Erzeugung eines Aktionspotentials wird das Gleichgewicht der Ionenkonzentrationen stark gestört und muss unter erhöhtem Energieumsatz wiederhergestellt werden. Im ersten Schritt werden der passive Transport und die grundlegenden Eigenschaften der Zellmembran mathematisch betrachtet. Anschließend wird der zum Erhalt des Zellvolumens und zum Ausgleich der Konzentration notwendige aktive Pumpmechanismus betrachtet und ein Grundmodell für eine biologische Nervenzellmembran erstellt. Darauf aufbauend ist eine Betrachtung der Erzeugung eines Aktionspotentials möglich, indem die Störung des geregelten Zell-Systems durch weitere, spannungsabhängige Ionenkanäle modelliert wird.

3.1.1 Kanalströme - passiver Transport

Durch unterschiedliche Konzentrationen eines Typs von Ionen im Zellinnen- und Zellaußenraum, wird die Diffusion von Ionen durch die permeable Zellmembran ausgelöst. Das sich dabei einstellende elektrische Potential wird durch das Nernstpotential beschrieben:

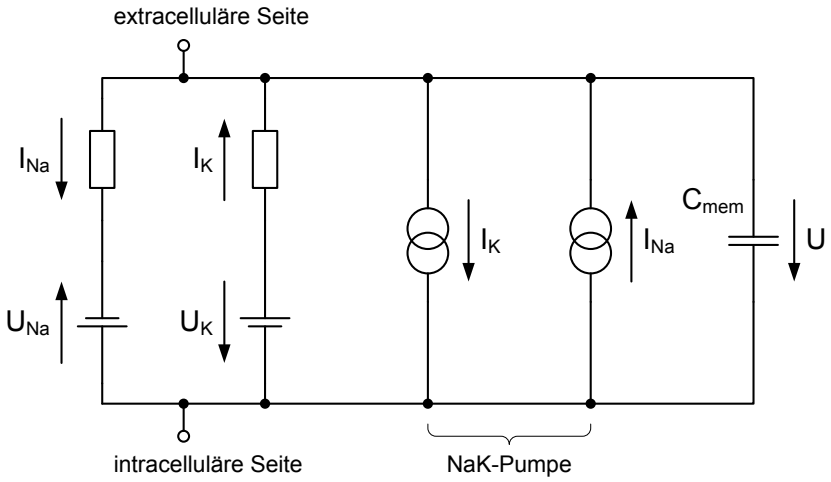


Abbildung 3.1: Elektrischer Ersatzschaltkreis für das Membranpotential im steady-state (angelehnt an [61]).

$$U_n = -\frac{\mathrm{R}T}{z\mathrm{F}} \ln\left(\frac{c_n}{c_{n,\mathrm{a}}}\right) \tag{3.2}$$

Der Parameter z beschreibt die Anzahl der durch das Ion beigetragenen Ladungen und hat im Folgenden bei der Betrachtung von K^+ und Na^+ den Wert 1. Die übrigen Konstanten sind die universelle Gaskonstante R und die Faradaykonstante F. Daneben geht die absolute Temperatur T, sowie die Konzentration des jeweiligen Ions im Zellinnenraum c_n sowie im Zellaußenraum $c_{n,\mathrm{a}}$ ein.

Die Überlagerung der Einzelpotentiale für K^+-Ionen und Na^+-Ionen ergibt das Gesamtpotential, bei dem zusätzlich die Permeabilität P der Zellmembran für den jeweiligen Ionentypen berücksichtigt werden muss (Goldman-Gleichung [36]).

$$U = -\frac{\mathrm{R}T}{\mathrm{F}} \ln\left(\frac{P_\mathrm{K} c_\mathrm{K} + P_\mathrm{Na} c_\mathrm{Na}}{P_\mathrm{K} c_{\mathrm{K,a}} + P_\mathrm{Na} c_{\mathrm{Na,a}}}\right) \tag{3.3}$$

Für die folgenden Rechnungen wird zur Vereinfachung die relative Permeabilität P für das Verhältnis $P_\mathrm{Na}/P_\mathrm{K}$ eingeführt:

$$U = -\frac{\mathrm{R}T}{\mathrm{F}} \ln\left(\frac{c_\mathrm{K} + P c_\mathrm{Na}}{c_{\mathrm{K,a}} + P c_{\mathrm{Na,a}}}\right) \tag{3.4}$$

Betrachtet man die Kanäle und auftretende Kanalströme als rein elektrische Vorgänge, so lässt sich unter der Annahme der Kanäle als Ohmsche Widerstände R_n folgende Gleichung angeben, welche den Stromfluss für Na^+ von der extrazellulären zur intrazellulären Seite, sowie den Stromfluss von K^+ von der intrazellulären zur extrazellulären Seite angibt (siehe Abb. 3.1). Im Folgenden wird die hier beschriebene Konvention für die

Stromrichtung angenommen: Der Einstrom von Natrium-Ionen in das Zellinnere wird durch einen positiven Strom I_{Na}, der Ausstrom von Kalium-Ionen aus dem Zellinneren in den extrazellulären Raum wird als negativer Strom I_{K} beschrieben.

$$I_n = \frac{\Delta U_n}{R_n} = -g_n \left(U - U_n\right) \tag{3.5}$$

Dabei beschreibt der Parameter U das Gesamtpotential auf der Membrankapazität, der Parameter U_n das Gleichgewichtspotential des jeweiligen Ions n.

Der Strom kann elektrochemisch als Änderung der Anzahl der Ladungsträger mit der Zeit bzw. als eine Konzentrationsänderung $\frac{d}{dt}c_n$ in einem definierten Zellvolumen V_i aufgefasst werden, so dass sich

$$I_n = q\mathrm{N_A}V_i \cdot \frac{d}{dt}c_n \tag{3.6}$$

ergibt. Die Konzentrationsänderung kann nun in Abhängigkeit von den durch die verschiedenen Ionen hervorgerufenen Potentialen beschrieben werden:

$$\frac{d}{dt}c_n = -\frac{g_n}{q\mathrm{N_A}V_i}\left(U - U_n\right) \tag{3.7}$$

3.1.2 Linearisiertes System

Im Folgenden soll das nichtlineare System (3.7) im Arbeitspunkt, d. h. um die Ruhekonzentrationen $c_{\mathrm{K},0}$ und $c_{\mathrm{Na},0}$ linearisiert werden. Dazu werden die Nernstpotentiale nach (3.2) mit Hilfe einer Taylorreihenentwicklung um den Konzentrationsarbeitspunkt $c_{n,0}$ des jeweiligen Ions linearisiert.

$$U_n(c_n) = U_n(c_{n,0}) + \frac{c_n - c_{n,0}}{1!}\frac{d}{dc_n}U_n(c_{n,0}) + \frac{(c_n - c_{n,0})^2}{2!}\frac{d^2}{{dc_n}^2}U_n(c_{n,0}) + \ldots \tag{3.8}$$

$$U_n(c_{n,0}) = -\frac{\mathrm{R}T}{\mathrm{F}}\ln\left(\frac{c_{n,0}}{c_{n,\mathrm{a}}}\right) \tag{3.9}$$

$$\frac{d}{dc_n}U_n(c_{n,0}) = -\frac{\mathrm{R}T}{\mathrm{F}}\frac{1}{c_{n,0}} \tag{3.10}$$

$$U(c_n) \approx -\frac{\mathrm{R}T}{\mathrm{F}}\ln\left(\frac{c_{n,0}}{c_{n,\mathrm{a}}}\right) - \frac{\mathrm{R}T}{\mathrm{F}}\frac{(c_n - c_{n,0})}{c_{n,0}} \tag{3.11}$$

In einem weiteren Schritt wird das Gesamtpotential nach (3.4) mit Hilfe einer Taylorreihe entwickelt:

$$U \approx -\frac{\mathrm{R}T}{\mathrm{F}} \ln\left(\frac{c_{\mathrm{K,0}} + P c_{\mathrm{Na,0}}}{c_{\mathrm{K,a}} + P c_{\mathrm{Na,a}}}\right) - \frac{\mathrm{R}T}{\mathrm{F}} \frac{1}{c_{\mathrm{K,0}} + P c_{\mathrm{Na,0}}} \cdot \left[(c_{\mathrm{K}} - c_{\mathrm{K,0}}) + P\,(c_{\mathrm{Na}} - c_{\mathrm{Na,0}})\right] \quad (3.12)$$

Zur Vereinfachung der Schreibweise werden im Folgenden die Taylorkoeffizienten angegeben:

$$\begin{aligned} U_{\mathrm{K}} &= U_{\mathrm{K,0}} + U_{\mathrm{K,1}} \cdot (c_{\mathrm{K}} - c_{\mathrm{K,0}}) \\ U_{\mathrm{Na}} &= U_{\mathrm{Na,0}} + U_{\mathrm{Na,1}} \cdot (c_{\mathrm{Na}} - c_{\mathrm{Na,0}}) \\ U &= U_0 + U_1 \cdot \left[(c_{\mathrm{K}} - c_{\mathrm{K,0}}) + P\,(c_{\mathrm{Na}} - c_{\mathrm{Na,0}})\right] \end{aligned} \quad (3.13)$$

$$\begin{aligned} U_{\mathrm{K,0}} &= -\tfrac{\mathrm{R}T}{\mathrm{F}} \ln\left(\tfrac{c_{\mathrm{K,0}}}{c_{\mathrm{K,a}}}\right) & U_{\mathrm{K,1}} &= -\tfrac{\mathrm{R}T}{\mathrm{F}} \tfrac{1}{c_{\mathrm{K,0}}} \\ U_{\mathrm{Na,0}} &= -\tfrac{\mathrm{R}T}{\mathrm{F}} \ln\left(\tfrac{c_{\mathrm{Na,0}}}{c_{\mathrm{Na,a}}}\right) & U_{\mathrm{Na,1}} &= -\tfrac{\mathrm{R}T}{\mathrm{F}} \tfrac{1}{c_{\mathrm{Na,0}}} \\ U_0 &= -\tfrac{\mathrm{R}T}{\mathrm{F}} \ln\left(\tfrac{c_{\mathrm{K,0}} + P c_{\mathrm{Na,0}}}{c_{\mathrm{K,a}} + P c_{\mathrm{Na,a}}}\right) & U_1 &= -\tfrac{\mathrm{R}T}{\mathrm{F}} \tfrac{1}{c_{\mathrm{K,0}} + P c_{\mathrm{Na,0}}} \end{aligned} \quad (3.14)$$

Dabei beschreibt der Koeffizient U_0 das Gleichgewichtspotential. Setzt man (3.13) und (3.14) in (3.7) ein, so erhält man die Differentialgleichungen für den potentialabhängigen zeitlichen Konzentrationsverlauf der beteiligten Ionen:

$$\begin{aligned} \frac{d}{dt} c_{\mathrm{K}} = -\frac{g_{\mathrm{K}}}{q \mathrm{N_A} V_i} \big(&U_0 + U_1 \cdot \left[(c_{\mathrm{K}} - c_{\mathrm{K,0}}) + P\,(c_{\mathrm{Na}} - c_{\mathrm{Na,0}})\right] \\ &- \left[U_{\mathrm{K,0}} + U_{\mathrm{K,1}} \cdot (c_{\mathrm{K}} - c_{\mathrm{K,0}})\right]\big) \end{aligned} \quad (3.15)$$

$$\begin{aligned} \frac{d}{dt} c_{\mathrm{Na}} = -\frac{g_{\mathrm{Na}}}{q \mathrm{N_A} V_i} \big(&U_0 + U_1 \cdot \left[(c_{\mathrm{K}} - c_{\mathrm{K,0}}) + P\,(c_{\mathrm{Na}} - c_{\mathrm{Na,0}})\right] \\ &- \left[U_{\mathrm{Na,0}} + U_{\mathrm{Na,1}} \cdot (c_{\mathrm{Na}} - c_{\mathrm{Na,0}})\right]\big) \end{aligned} \quad (3.16)$$

Mit der Darstellung der Konzentrationen als Vektor $\mathbf{c} = \left[c_{\mathrm{K}}\, c_{\mathrm{Na}}\right]^{\mathrm{T}}$ lässt sich eine einfache Darstellung des Gleichungssystems angeben:

$$\frac{d}{dt}\mathbf{c} = \mathbf{M} \cdot (\mathbf{c} - \mathbf{c_0}) + \mathbf{I_0} \quad (3.17)$$

$$\mathbf{M} = \begin{bmatrix} -\frac{g_{\mathrm{K}}}{q\mathrm{N_A}V_i}\left(U_1 - U_{\mathrm{K},1}\right) & -\frac{g_{\mathrm{K}}}{q\mathrm{N_A}V_i}\left(PU_1\right) \\ -\frac{g_{\mathrm{Na}}}{q\mathrm{N_A}V_i}\left(U_1\right) & -\frac{g_{\mathrm{Na}}}{q\mathrm{N_A}V_i}\left(PU_1 - U_{\mathrm{Na},1}\right) \end{bmatrix} \tag{3.18}$$

$$\mathbf{I_0} = \frac{1}{q\mathrm{N_A}V_i} \begin{bmatrix} -g_{\mathrm{K}}\left(U_0 - U_{\mathrm{K},0}\right) \\ -g_{\mathrm{Na}}\left(U_0 - U_{\mathrm{Na},0}\right) \end{bmatrix} \tag{3.19}$$

Der bislang noch offene Parameter P für die relative Permeabilität kann anhand der Bedingung für die Ströme durch die Ionenkanäle im Ruhezustand des Systems bestimmt werden. Das Verhältnis des passiven Ionenstroms der Natrium-Ionen zum Ionenstrom der Kalium-Ionen muss 3/2 betragen. Dieses ist das Verhältnis, in dem die Natrium-Kalium-Pumpe Ionen aktiv gegen die Flussrichtung der passiven Transportvorgänge durch die Membran transportiert. Im Ruhezustand sollten sich aktive und passive Transporte gerade ausgleichen. Aus der Bedingung für das Pumpverhältnis von 3 Natriumionen zu 2 Kaliumionen erhalten wir die Stromgleichung

$$3I_{\mathrm{K},0} + 2I_{\mathrm{Na},0} = 0. \tag{3.20}$$

Mit (3.20) und (3.5) erhalten wir eine weitere Gleichung (3.21) zur Bestimmung des Gleichgewichtspotentials U_0, dieses mal aus einer elektrischen Sicht.

$$U_0 = \frac{g_{\mathrm{K}}U_{\mathrm{K},0} + \frac{2}{3}g_{\mathrm{Na}}U_{\mathrm{Na},0}}{g_{\mathrm{K}} + \frac{2}{3}g_{\mathrm{Na}}} \tag{3.21}$$

Beide Lösungen für das Gleichgewichtspotential – (3.14) und (3.21) – können gleichgesetzt und nach dem Parameter der relativen Permeabilität P aufgelöst werden. Die relative Permeabilität ist damit hauptsächlich von den Leitwerten g_n der Ionenkanäle abhängig. Dieses eröffnet die Möglichkeit, die Erzeugung eines Aktionspotentials durch Modulation der Ionenkanal-Leitwerte zu modellieren und im weiteren Abschnitt spannungsgesteuerte Ionenkanäle in das Modell einzubringen.

$$P\left(\frac{g_{\mathrm{Na}}}{g_{\mathrm{K}}}\right) = -\frac{c_{\mathrm{K},0} - c_{\mathrm{K,a}} \cdot \exp(-\frac{U_0 F}{RT})}{c_{\mathrm{Na},0} - c_{\mathrm{Na,a}} \cdot \exp(-\frac{U_0 F}{RT})} \quad \text{mit} \quad U_0 = \frac{U_{\mathrm{K},0} + \frac{2}{3}\frac{g_{\mathrm{Na}}}{g_{\mathrm{K}}}U_{\mathrm{Na},0}}{1 + \frac{2}{3}\frac{g_{\mathrm{Na}}}{g_{\mathrm{K}}}} \tag{3.22}$$

3.1.3 Pumpströme - aktiver Transport

Durch den aktiven Transport von Ionen durch die Na^+-K^+-Pumpe unter Hydrolyse eines ATP-Moleküls wird ebenfalls eine Konzentrationsänderung in der Zelle hervorgerufen. Dabei werden aktiv 3 Na^+-Ionen aus der Zelle heraus und 2 K^+-Ionen in die Zelle hinein transportiert. Dieses lässt sich durch eine Konzentrationsänderung mit dem Term (3.23) beschreiben. Dabei ist auf die richtige Wahl der Vorzeichen zu achten. Der aktive Ausstrom von Natrium-Ionen durch die Natrium-Kalium-Pumpe wird vorzeichenrichtig mit negativem Vorzeichen beschrieben. Dieses ist durch die Abnahme der Na^+-Konzentration im Zellinneren gegeben.

$$\frac{d}{dt}\mathbf{c} = W \cdot (\mathbf{c} - \mathbf{c}_{0,\mathrm{NaK}}) \qquad \text{mit} \qquad W = \frac{k}{T}\begin{bmatrix} -2 & -2 \\ 3 & 3 \end{bmatrix} \tag{3.23}$$

Die Änderung der Konzentration von Na^+ und K^+ erfolgt an dieser Stelle in Abhängigkeit von der Gesamtkonzentration beider Ionenarten, um einerseits einen gleichbleibenden Austausch der Ionen im Verhältnis 3/2 zu gewährleisten, andererseits um die NaK-ATPase abhängig von der Konzentration der Ionen selbst zu machen. Diese Konzentrationsabhängigkeit wurde bereits in [6] beschrieben. Der zusätzliche Parameter $\mathbf{c}_{0,\mathrm{NaK}}$ ist noch in Abhängigkeit von der Gesamtbilanz zu wählen und beschreibt die Konzentration, bei der die NaK-ATPase beginnt zu arbeiten.

Berechnung der Konzentration $\mathbf{c}_{0,\mathrm{NaK}}$

Im Ruhezustand ($\mathbf{c} = \mathbf{c}_0$) muss der Ruhestrom $\mathbf{I}_0$ durch den aktiven Transport von Ionen durch die Natrium-Kalium-Pumpe ausgeglichen werden. Dazu ist die Bedingung

$$\frac{d}{dt}\mathbf{c} = \mathbf{0} = \mathbf{I}_0 + \mathbf{W} \cdot (\mathbf{c} - \mathbf{c}_{0,\mathrm{NaK}}) \tag{3.24}$$

zu erfüllen.

Durch Summation aller in diesem Kapitel betrachteten Ausgleichsvorgänge ergibt sich die Gesamtbilanz der Konzentrationsänderung zu

$$\frac{d}{dt}\mathbf{c} = \mathbf{M} \cdot (\mathbf{c} - \mathbf{c}_0) + \mathbf{I}_0 + \mathbf{W} \cdot (\mathbf{c} - \mathbf{c}_{0,\mathrm{NaK}})\,. \tag{3.25}$$

Was vorerst offen bleibt, ist die Bestimmung der Pumpzyklen der Matrix $\mathbf{W}$ aus (3.25), welche den Einfluss der Natrium-Kalium-Pumpe auf die Änderung der Gesamtkonzentration beschreibt. Im Weiteren soll dieser Teil durch Anwendung der Regelungstechnik auf das stabilisierende Element der Zelle – die Natrium-Kalium-Pumpe – behandelt werden.

3.1.4 Betrachtung der Natrium-Kalium-Pumpe als regelungstechnisches Problem

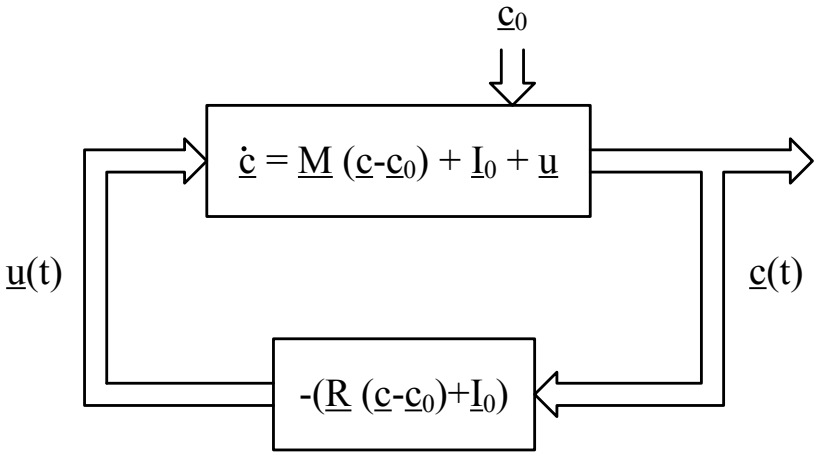
Konzentrieren wir uns auf die Natrium-Kalium-Pumpe als primären Mechanismus zur Aufrechterhaltung des Membranpotentials, stellt sich die Frage, wie diese Pumpe so modelliert werden kann, dass die Modellierung eine Aussage über den Energiebedarf von Nervenzellen erlaubt. In [69] wurden Erweiterungen des oben hergeleiteten Modells für ein Neuron vorgenommen, welche es erlauben, den aktiven Transport von Natrium- und Kalium-Ionen durch die NaK-ATPase als zusätzlichen Term des Gleichungssystems (3.17) zu beschreiben. Der zusätzliche Term unterliegt dabei besonderen Bedingungen, insbesondere Einschränkungen und Vorkehrungen bezüglich der Stabilität des Gesamtsystems. Die in [69] vorgenommenen Berechnungen bezüglich Stabilität des Systems beziehen sich dabei ausschließlich auf das linearisierte System. Prinzipiell kann der zusätzliche Term als Regler verstanden werden, der in dieser Arbeit mit Hilfe eines regelungstechnischen Ansatzes modelliert werden soll. Im Folgenden wird die Natrium-Kalium-Pumpe als Regler eines geschlossenen Systems (das Neuron) im klassischen Sinne betrachtet. Die Nomenklatur diese Abschnitts orientiert sich an dem Standardwerk von Föllinger [29] mit dem Unterschied, dass die Matrix $\mathbf{M}$ hier als Systemmatrix und nicht als Vorfilter genutzt wird (siehe Abb. 3.2). Das Neuron selbst weist dabei zwei Kontrollvariablen auf, den Natrium-Ionen- und Kalium-Ionen-Fluss in Form der zeitlichen Änderung der Ionenkonzentrationen. Das Mehrgrößensystem wird daher im Folgenden mit der Methode der Polvorgabe stabilisiert.

Der Regel-Vektor $\mathbf{R}$ kann direkt aus den Eigenwerten und Eigenvektoren der Systemmatrix $\mathbf{M}$ bestimmt werden:

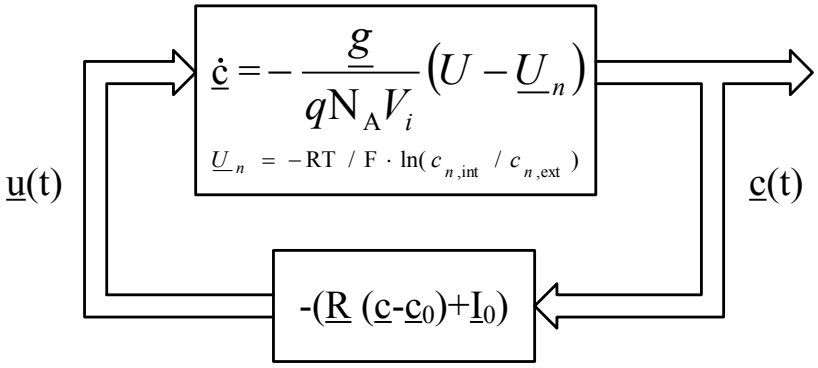
$$\mathbf{R} = \mathbf{V} \cdot (\mathrm{diag}(\lambda_1, \lambda_2) - \mathrm{diag}(\lambda_{R1}, \lambda_{R2})) \cdot \mathbf{V}^{-1} \tag{3.26}$$

Dabei beschreiben $\mathbf{V}$ die Matrix der Eigenvektoren der Systemmatrix $\mathbf{M}$ und λ_n die Eigenwerte von $\mathbf{M}$. Um das System zu stabilisieren, werden die dominanten Pole des Regelkreises durch im Allgemeinen freie Wahl der Eigenwerte des Regelvektors $\mathbf{R}$ in die linke komplexe Halbebene bewegt. Für die Polvorgabe in Mehrgrößensystemen ist die Wahl geeigneter Eigenwerte λ_{Rn} nur bei Vorliegen der Regelungsnormalform intuitiv möglich, für alle andere Fälle werden in der Regel geeignete Werte durch numerische Simulation bestimmt. Durch Simulation des Systems mit verschiedenen Werten für die Eigenwerte λ_{Rn} wurde ein optimaler Wert von $\lambda_n = -5 \cdot 10^4$ ermittelt. Bei dieser Wahl konvergiert die Ionenkozentration nach Auslenkung des Systems nach Auslösen eines Aktionspotentials innerhalb der physiologisch sinnvollen Zeit von 1 ms wieder gegen die jeweilige Ruhekonzentration.

Die Besonderheit an der hier gewählten Vorgehensweise ist, dass der Regler zwar am linearisierten System ausgelegt wird, später aber genauso das nichtlineare System ausregeln kann. Dieses wird in einem späteren Abschnitt anhand von Simulationen und

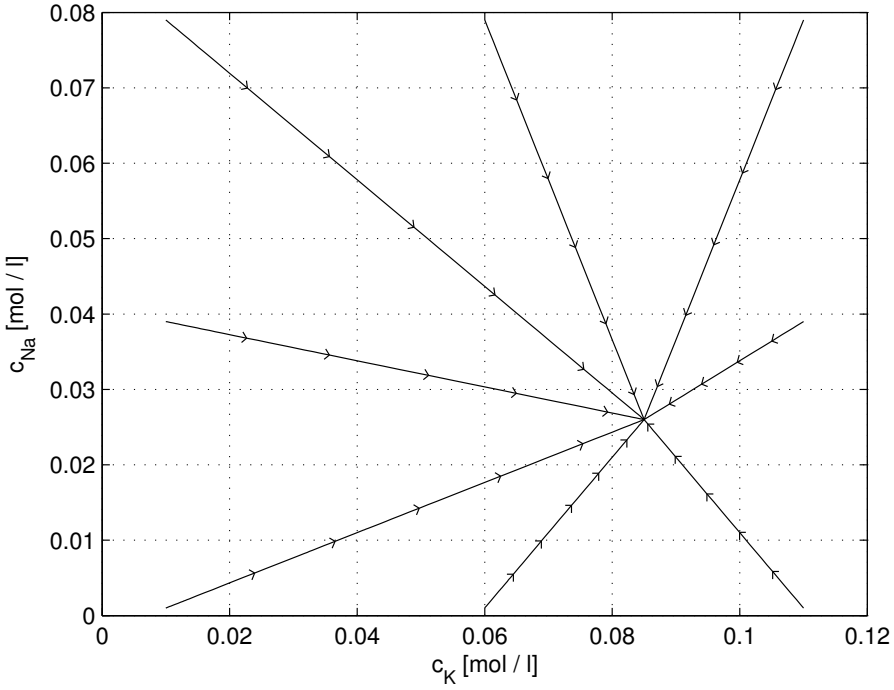


(a) Linearisiertes System.

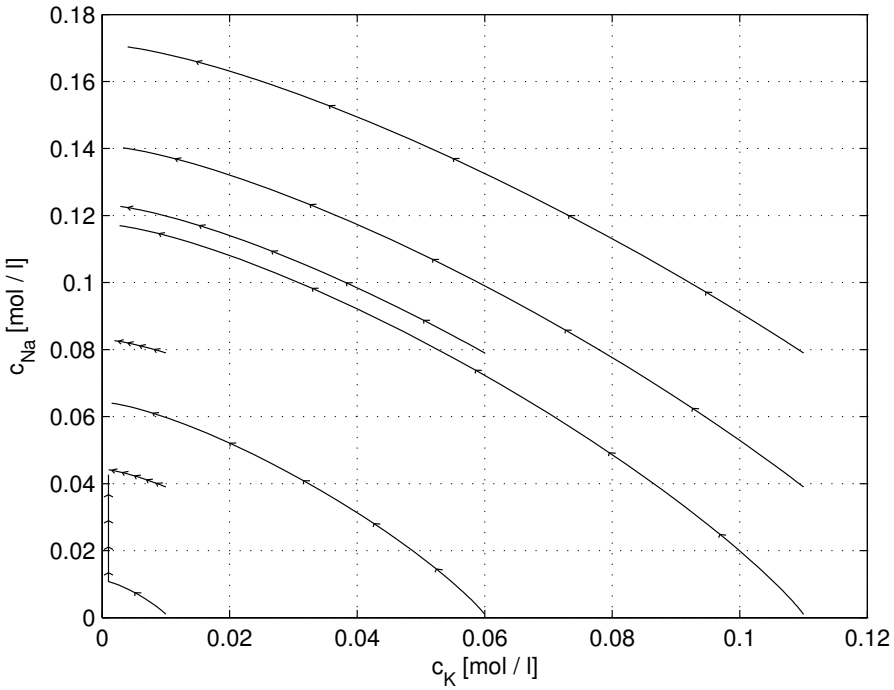


(b) Nichtlineares System.

Abbildung 3.2: Nomenklatur zum Reglerentwurf für das geschlossene regelungstechnische System.



(a)



(b)

Abbildung 3.3: Trajektorien des Verlaufs der Ionenkonzentrationen im einfachen nichtlinearen Neuronenmodell mit a) geschlossenem Regelkreis und b) offenem Regelkreis bei verschiedenen Anfangsbedingungen.

der Darstellung der Eigenbewegungen des Systems, d. h. der Darstellung der zeitlichen Verläufe der Ionenkonzentrationen in einem Phasenplot, gezeigt (siehe auch [111]).

In Abb. 3.3, sind die Eigenbewegungen des Neuronenmodells mit geschlossenem und offenem Regelkreis in einem Phasenplot dargestellt. Dabei wird der zeitliche Verlauf der Variablen des geregelten Systems, in diesem Fall der Natrium- und Kaliumionenkonzentration (c_{Na} und c_{K}) aufgetragen. Ausgehend von verschiedenen Anfangskonzentrationen eines physiologisch sinnvollen Konzentrationsbereiches stabilisiert der eingeführte Regler die Ionenkonzentrationen im Zellinneren des Neuronenmodells im physiologischen Arbeitspunkt $c_{\mathrm{K},0}$, $c_{\mathrm{Na},0}$ und lässt das System nach Auslenkung bzw. Start der Simulation in diesen zurückkehren. In Abb. 3.3a ist die Bewegung des Systems mit der Zeit bei geschlossenem Regelkreis dargestellt. In der daneben dargestellten Abbildung 3.3b sind die Eigenbewegungen des Systems nach Entfernung des Reglers dargestellt. Das ungeregelte System strebt darin zu einer Ruhelage $[c_{\mathrm{K,a}}, c_{\mathrm{Na,a}}]$, in der die Ionenkonzentrationen auf beiden Seiten der Zellmembran ausgeglichen sind.

3.1.5 Stabilitätsprüfung des nichtlinearen, geregelten Systems mittels Ljapunov-Verfahren

Im Folgenden soll die durch numerische Betrachtungen gezeigte Stabilität des geregelten, nichtlinearen Systems mathematisch mit der Methode nach Ljapunov verifiziert werden. Dazu wird das geregelte System in seiner zu erreichenden Ruhelage bei der Ruhekonzentration $\mathbf{c} = \mathbf{0}$ betrachtet. Die Analyse der Stabilität nach dem Verfahren von Ljapunov erfordert die Einführung des Begriffs der Definitheit.

Definition 1 *Eine stetige Funktion $V(\mathbf{c})$ heißt positiv definit in einer Umgebung $\mathfrak{U}$, falls gilt*

$$\begin{aligned} &V(\mathbf{0}) = \mathbf{0} \quad \text{und} \\ &V(\mathbf{c}) > \mathbf{0} \quad \forall\, \mathbf{c} \in \mathfrak{U},\, \mathbf{c} \neq \mathbf{0}. \end{aligned} \tag{3.27}$$

Eine stetige Funktion $V(\mathbf{c})$ ist negativ definit, wenn die Funktion $-V(\mathbf{c})$ positiv definit ist.

Definition 2 *Sei $\dot{\mathbf{c}} = f(\mathbf{c})$ ein dynamisches System mit Fixpunkt $\mathbf{c}_0 = \mathbf{0}$. Dann ist V eine strenge Ljapunov-Funktion des Systems in einer Umgebung $\mathfrak{U}$ von $\mathbf{c}_0$, falls gilt*

- *V ist positiv definit* $\quad \forall\, \mathbf{c} \in \mathfrak{U}$
- *$\dot{V}$ ist negativ definit* $\quad \forall\, \mathbf{c} \in \mathfrak{U}$
- $\dot{V}(\mathbf{c}) = \frac{\partial V}{\partial \mathbf{c}} f(\mathbf{c}) \quad \forall\, \mathbf{c} \in \mathfrak{U}$.

Existiert in einer Umgebung $\mathbf{c} = \mathbf{0}$ der betrachteten Ruhelage des geregelten Systems eine strenge Ljapunov-Funktion, ist diese Ruhelage asymptotisch stabil.

Gegeben ist diesem Fall das nichtlineare geregelte System (Abb. 3.2) in Form eines Differentialgleichungssystems in $\dot{\mathbf{c}} = f(\mathbf{c})$, dessen Ruhelage bei $\mathbf{c}_0 = [c_{\mathrm{K},0} \quad c_{\mathrm{Na},0}]^{\mathrm{T}}$ liegt. Da die Prüfung der Stabilität nach Ljapunov eine Ruhelage in $\mathbf{0}$ fordert, wird das gesamte System durch die Substitution von

$$\begin{aligned} c_{\mathrm{K}} &= c_{\mathrm{K},0} + \Delta c_{\mathrm{K}} \qquad \text{und} \\ c_{\mathrm{Na}} &= c_{\mathrm{Na},0} + \Delta c_{\mathrm{Na}} \end{aligned} \tag{3.28}$$

verschoben, so dass die Ruhelage in den Ursprung gebracht wird. Abschließend liegt das System als Differentialgleichungssystem in den neuen Koordinaten $\boldsymbol{\Delta}\mathbf{c}$ vor.

$$\dot{\boldsymbol{\Delta}}\mathbf{c} = -\frac{1}{q\mathrm{N_A}V_i} \begin{bmatrix} g_{\mathrm{K}}(U - U_{\mathrm{K}}) \\ g_{\mathrm{Na}}(U - U_{\mathrm{Na}}) \end{bmatrix} - (\mathbf{R}(\mathbf{c} - \mathbf{c}_0) + \mathbf{I}_0) \tag{3.29}$$

Zur Vereinfachung der Schreibweise wird im Folgenden der Vektor $\boldsymbol{\Delta}\mathbf{c}$ wieder durch $\mathbf{c}$ ersetzt. Wir haben nun ein System gegeben, das die Bedingungen $\dot{\mathbf{c}} = f(\mathbf{c})$ und $f(\mathbf{0}) = \mathbf{0}$ erfüllt und suchen eine dazu gehörende Ljapunov-Funktion $V(\mathbf{c})$.

Als Ljapunov-Funktion wird $V(\mathbf{c}) = \mathbf{c}^{\mathrm{T}}\mathbf{1}\mathbf{c}$ gewählt, deren Eigenwerte positiv sind. Damit wird $V(\mathbf{c})$ positiv definit.

Die Ableitung der Ljapunov-Funktion wird so zu:

$$\frac{\partial V(\mathbf{c})}{\partial \mathbf{c}} = 2[c_{\mathrm{K}} \quad c_{\mathrm{Na}}] \tag{3.30}$$

Um die Ljapunov-Stabilität nachweisen zu können, muss das Produkt aus der Ableitung und der Systembeschreibung um die Ruhelage negativ definit sein ($\dot{V}(\mathbf{c}) \leq 0$).

$$2[c_{\mathrm{K}} \quad c_{\mathrm{Na}}] \cdot \left[-\frac{1}{q\mathrm{N_A}V_i} \begin{bmatrix} g_{\mathrm{K}}(U - U_{\mathrm{K}}) \\ g_{\mathrm{Na}}(U - U_{\mathrm{Na}}) \end{bmatrix} - (\mathbf{R}\mathbf{c} + \mathbf{I}_0) \right] \leq 0 \tag{3.31}$$

Nach numerischer Auswertung ist die in (3.31) gegebene Bedingung für die gewählten Parameter des geregelten Systems innerhalb physiologisch sinnvoller Grenzen um die Ruhelage erfüllt (siehe Abbildung 3.4) und das System damit stabil.

3.1.6 Erweiterung des Grundmodells zu einem Modell für Synapse und Dendrit

Das in diesem Abschnitt vorgestellte Synapsenmodell beruht auf dem Grundmodell der Nervenzelle, welches in Kapitel 3.1 bereits ausführlich hergeleitet wurde. Die Ausarbeitung dieser Erweiterung wurde bereits von Löffler [69] vorgenommen und wird zum besseren Verständnis an dieser Stelle kurz wiederholt, da die Simulation des Minimalsystems aus

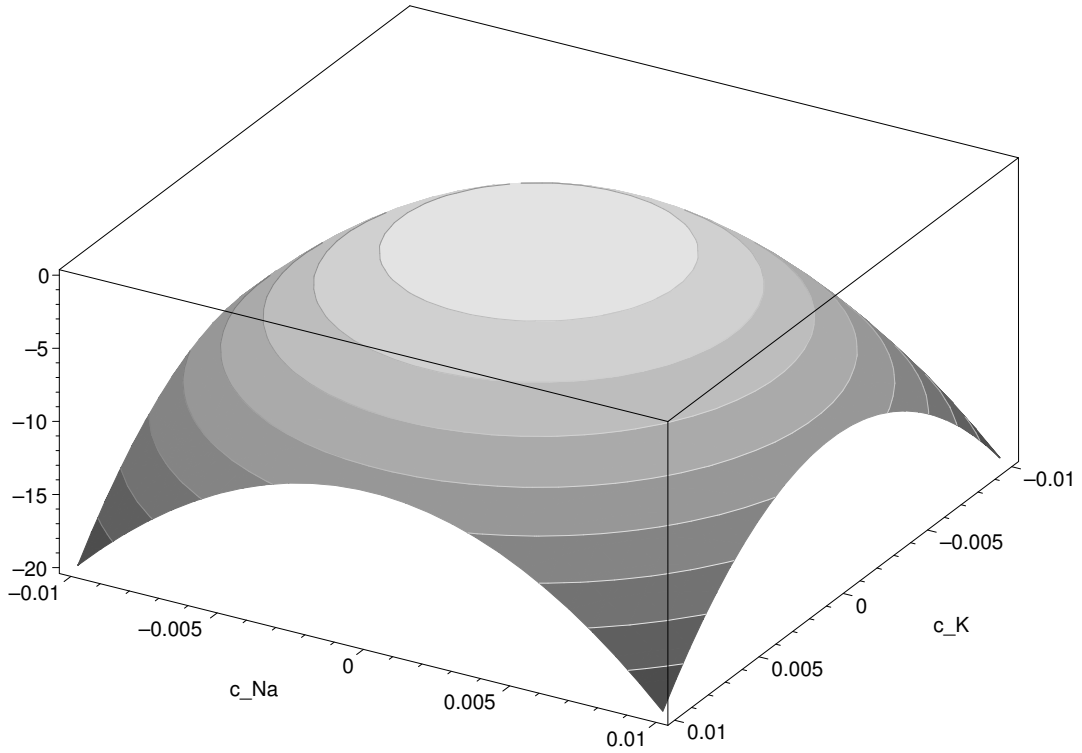


Abbildung 3.4: Darstellung von $\dot{V}(\mathbf{c})$ für das geregelte System. In der Umgebung der Ruhelage ist die Funktion negativ definit und die Ruhelage damit stabil.

dem Modell für das Neuron mit Erzeugung eines Aktionspotentials sowie dem Modell für die Synapse besteht.

Die Erregung der Synapse durch ein eintreffendes Aktionspotential ruft eine Erhöhung der Neurotransmitterkonzentration an den Rezeptoren des empfangenden Neurons hervor. Dieses hat den Effekt, dass sich die Leitwerte der transmittergesteuerten Natrium- und Kalium-Kanäle zeitlich ändern. Hier wird, statt einen Strom in das Neuron zu injizieren und damit ein bestimmtes Membranpotential hervorzurufen, direkt an den ionenspezifischen Leitwerten der Zellmembran angesetzt, wie in [13, 25] vorgeschlagen. Das Membranpotential ergibt sich in der Folge aus den geänderten Ionenkonzentrationen die durch die Änderung der Leitwerte der Zellmembran hervorgerufen werden. In dem hier vorliegenden Modell wird davon ausgegangen, dass die sich zusätzlich öffnenden Kanäle ionenunspezifisch sind und die Zunahme des Leitwerts auf Natrium- und Kalium-Ionenkanal den gleichen Einfluss hat. Diese Annahme kann mathematisch durch den Term

$$\begin{aligned} \hat{g}_{\mathrm{K}}(t) &= g_{\mathrm{K}} + G(t) \\ \hat{g}_{\mathrm{Na}}(t) &= g_{\mathrm{Na}} + G(t) \\ G(t) &= g_c A(t) \end{aligned} \tag{3.32}$$

ausgedrückt werden. Hier beschreibt der Parameter g_c den Leitwert eines einzelnen Kanals (im Sinne der Stärke der Synapse) und die Funktion $A(t)$ den zeitbehafteten Verlauf der Anzahl der zusätzlich geöffneten Ionenkanäle. Die Anzahl der zusätzlich geöffneten (oder geschlossenen) Kanäle hängt von der Konzentration der gebundenen (oder freien) Neurotransmitter ab. Die Dynamik der Transmitter(T)-Rezeptor(R)-Bindung kann mit der chemischen Reaktionsgleichung

$$\text{(freier Transmitter)} \quad R + T = B \underset{\beta}{\overset{\alpha}{\rightleftharpoons}} A = RT^* \quad \text{(gebundener Transmitter)} \tag{3.33}$$

beschrieben werden. Wir erhalten Differentialgleichungen für A und B, welche von den Übergangswahrscheinlichkeiten zwischen gebundenem oder freiem Transmitter abhängen:

$$\frac{d}{dt}A = \alpha B - \beta A, \quad \frac{d}{dt}B = \beta A - \alpha B \tag{3.34}$$

Die Übergangswahrscheinlichkeit α wird nun mit einem zeitabhängigen Neurotransmitter-Profil variiert. Dabei wird der Term α durch $\alpha \cdot q(t)$ ersetzt, wobei $q(t)$ den Zeitverlauf des Transmitterprofils beschreibt.

$$\frac{d}{dt}A = \alpha q(t)\,(A + B - A) - \beta A \tag{3.35}$$

Nach Multiplikation von (3.35) mit g_c und Anwendung des Terms $G(t) = g_c A(t)$ erhält man eine Differentialgleichung mit zeitabhängigen Koeffizienten für den Zusatzleitwert $G(t)$, der im Wesentlichen vom Neurotransmitter-Profil $q(t)$ abhängig ist.

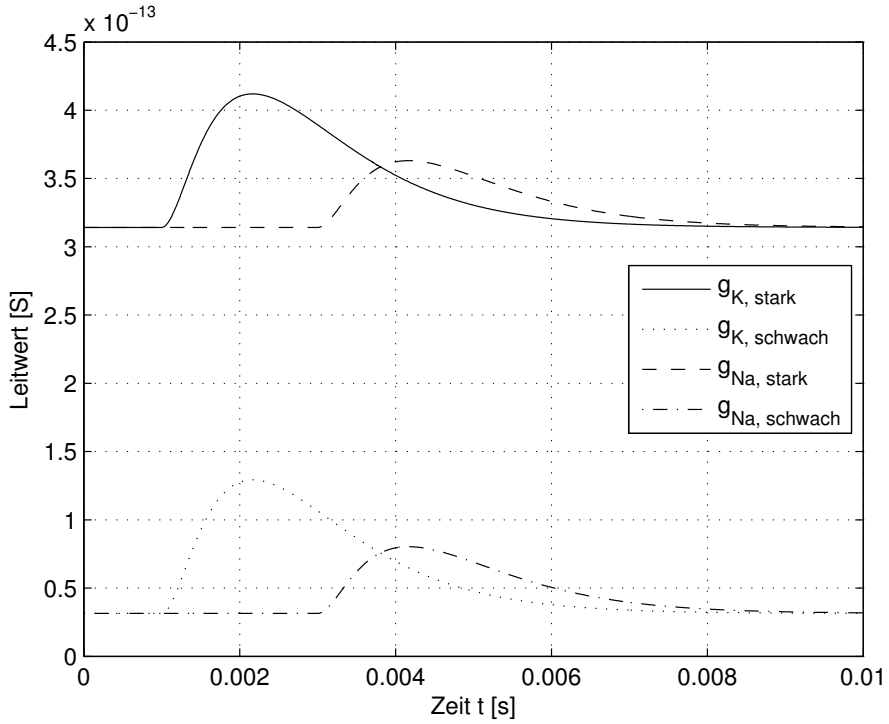


Abbildung 3.5: Zeitabhängige Leitwerte g_{K} und g_{Na} an der Synapse. Zum Zeitpunkt $t = 1\,\mathrm{ms}$ wird die starke Synapse erregt, zum Zeitpunkt $t = 3\,\mathrm{ms}$ wird die schwache Synapse erregt.

$$\frac{d}{dt}G(t) = \alpha q(t)\left(\underbrace{g_c\,(A+B)}_{G_S} - \underbrace{g_c A(t)}_{G(t)}\right) - \beta\,\underbrace{g_c A(t)}_{G(t)} \tag{3.36}$$

G_S ist ein Leitwert, der bei Sättigung auftritt, d. h. wenn alle Ionenkanäle geöffnet sind. Die Lösung dieser Differentialgleichung ist

$$\begin{aligned} G(t) = &\left[G_0 + \alpha G_S \int_0^t q(t'')\exp\left(\beta t'' + \alpha\int_0^{t''} q(t')dt'\right)dt''\right] \\ &\cdot\exp\left(-\left(\beta t + \alpha\int_0^t q(t')dt'\right)\right). \end{aligned} \tag{3.37}$$

In den später gezeigten Simulationen wird die α-Funktion als Form des Neurotransmitter-Profils $q(t)$ genutzt. Für eine schnelle Synapse nach [19] werden in den Simulationen die Parameter $\alpha = 2\,\mathrm{ms}^{-1}$ und $\beta = 1\,\mathrm{ms}^{-1}$ genutzt.

In Abb. 3.5 ist der Verlauf der zeitabhängigen Leitwerte der Natrium- und Kaliumkanäle bei Erregung der Zellmembran mit zwei verschieden starken Synapsen zu zwei verschiedenen Zeitpunkten dargestellt. Der Einfluss der Synapsen auf die Leitwerte der Zellmembran wird jeweils mit einer α-Funktion modelliert.

3.1.7 Modellierung eines Aktionspotentials

Um das Minimalmodell für eine Simulation zu komplettieren, muss an dieser Stelle noch die Erzeugung eines Aktionspotentials modelliert werden. Auch hier kann auf das Grundmodell zurückgegriffen werden. Das Aktionspotential wird durch eine abrupte Änderung der Leitfähigkeit erzeugt, welche vom Membranpotential durch die sogenannten spannungsabhängigen Kanäle (voltage-gated channels) abhängig ist. Daher werden im Folgenden zeitbehaftete ionenspezifische Leitwerte $\hat{g}_{\mathrm{K}}(t)$ und $\hat{g}_{\mathrm{Na}}(t)$ eingeführt, welche direkt aus einem Hodgkin-Huxley Modell [50] abgeleitet werden können. Tatsächlich beschreibt die nachfolgende Darstellung nur eine Approximation des Hodgkin-Huxley Modells, bewahrt jedoch dessen grundlegende Eigenschaften.

$$\begin{aligned} \hat{g}_{\mathrm{K}}(t) &= g_{\mathrm{K}} + G_{\mathrm{K}}(t) \quad \text{mit} \quad G_{\mathrm{K}}(t) = \bar{g}_{\mathrm{K}} n(t)^4 \\ \hat{g}_{\mathrm{Na}}(t) &= g_{\mathrm{Na}} + G_{\mathrm{Na}}(t) \quad \text{mit} \quad G_{\mathrm{K}}(t) = \bar{g}_{\mathrm{Na}} m(t)^3 h(t) \end{aligned} \tag{3.38}$$

Die Zeitverläufe der Koeffizienten $m(t)$, $n(t)$ und $h(t)$ können durch inhomogene Differentialgleichungen dargestellt werden, wie diese in der Arbeit von Löffler [69] hergeleitet wurden. An dieser Stelle soll noch einmal beispielhaft die Lösung für den Parameter $m(t)$ gezeigt werden. Wie schon im letzten Abschnitt wird bei der chemischen Reaktionsgleichung

$$1 - m \underset{\beta_m (U - U_{\mathrm{TH}})}{\overset{\alpha_m (U - U_{\mathrm{TH}})}{\rightleftharpoons}} m \tag{3.39}$$

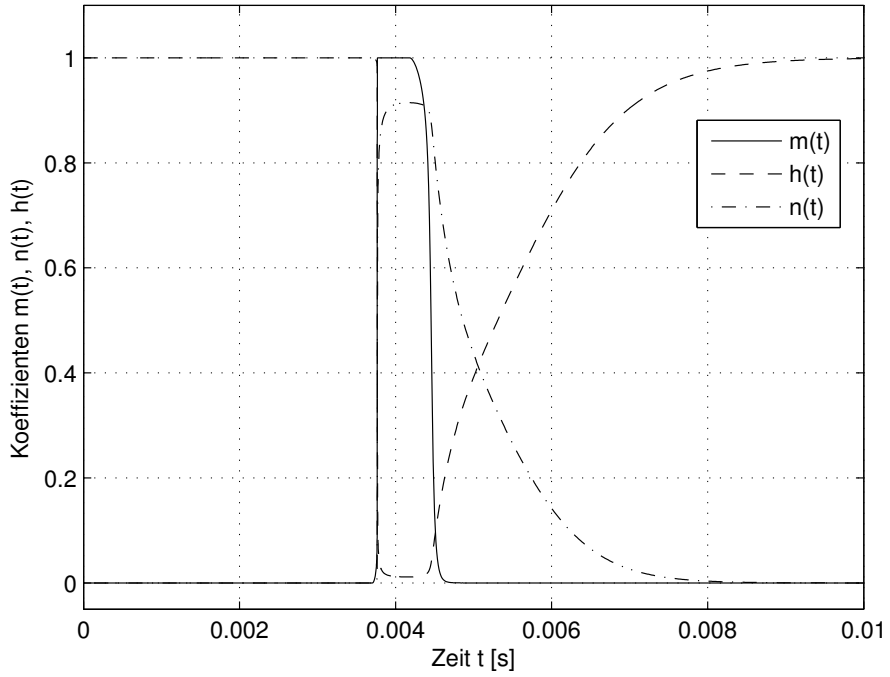
begonnen. Die Parameter α_m und β_m beschreiben die Wahrscheinlichkeit des Übergangs eines Ionenkanals vom offenen zum geschlossenen Zustand. Die Schwellenspannung ist mit U_{TH} angegeben, U beschreibt das Membranpotential. Diese Darstellung führt zu der inhomogenen Differentialgleichung

$$\frac{d}{dt} m = \alpha_m \left(U(t) - U_{\mathrm{TH}}\right) - m \left[\alpha_m \left(U(t) - U_{\mathrm{TH}}\right) + \beta_m \left(U(t) - U_{\mathrm{TH}}\right)\right] \tag{3.40}$$

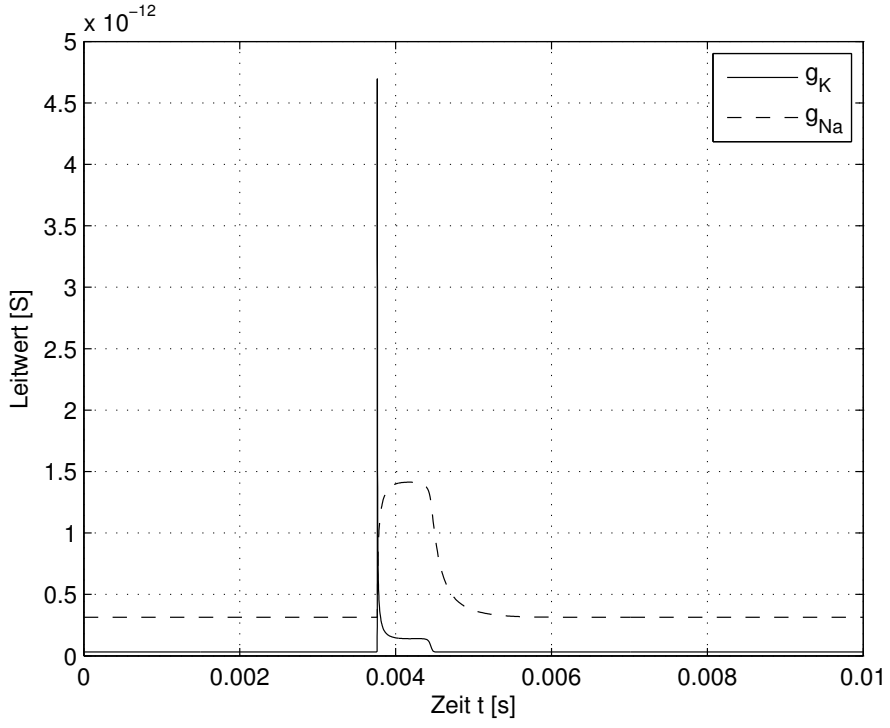
mit ihren stückweise definierten Koeffizienten α_m und β_m und der Lösung

$$m(t) = m_0 \exp\left(-\left(\alpha_m + \beta_m\right) t\right) + \frac{\alpha_m}{\alpha_m + \beta_m}. \tag{3.41}$$

Diese Gleichungen müssen partiell gelöst werden, da die Parameter α_m und β_m spannungsabhängig sind. Die Parameter, welche benötigt werden um die Funktionen von m, n und h zu bestimmen, wurden [24] entnommen. Ein typischer Zeitverlauf für die Funktionen $m(t)$, $n(t)$, und $h(t)$ ist in Abb. 3.6a dargestellt. Die daraus resultierende Leitfähigkeit der Kalium- und Natrium-Kanäle während eines Aktionspotentials ist in Abb. 3.6b dargestellt. Das Membranpotential bzw. Aktionspotential ergibt sich aus der veränderten Konzentration der Natrium- und Kalium-Ionen durch die zeitabhängig zusätzlich geöffneten Kanäle.



(a)



(b)

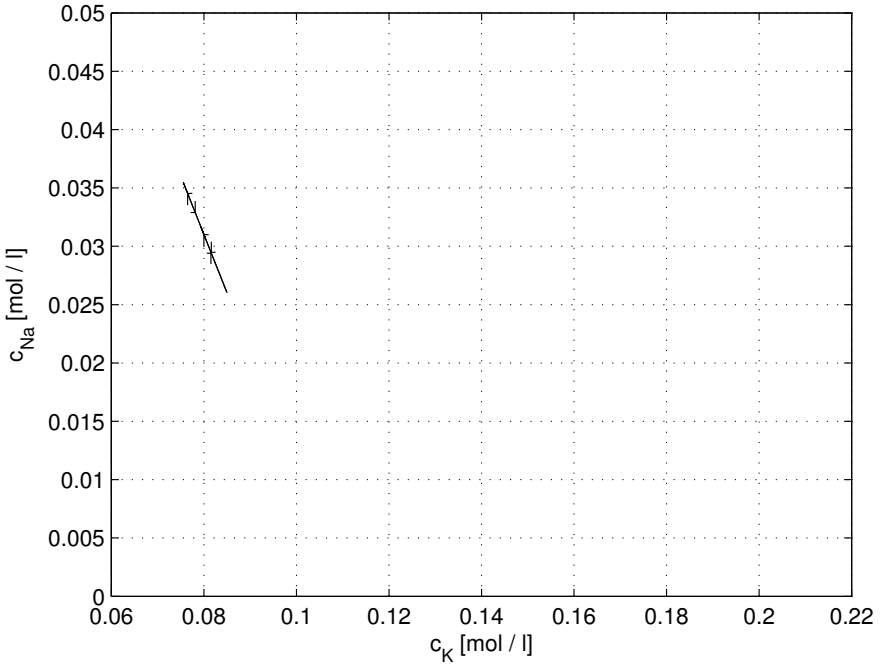
Abbildung 3.6: Zeitverlauf (a) der Koeffizienten $m(t)$, $n(t)$ und $h(t)$ und (b) der daraus resultierenden Leitwerte der Kalium- und Natrium-Kanäle für ein bei t=3,7 ms ausgelöstes Aktionspotential.

3.1.8 Simulation eines Minimalsystems

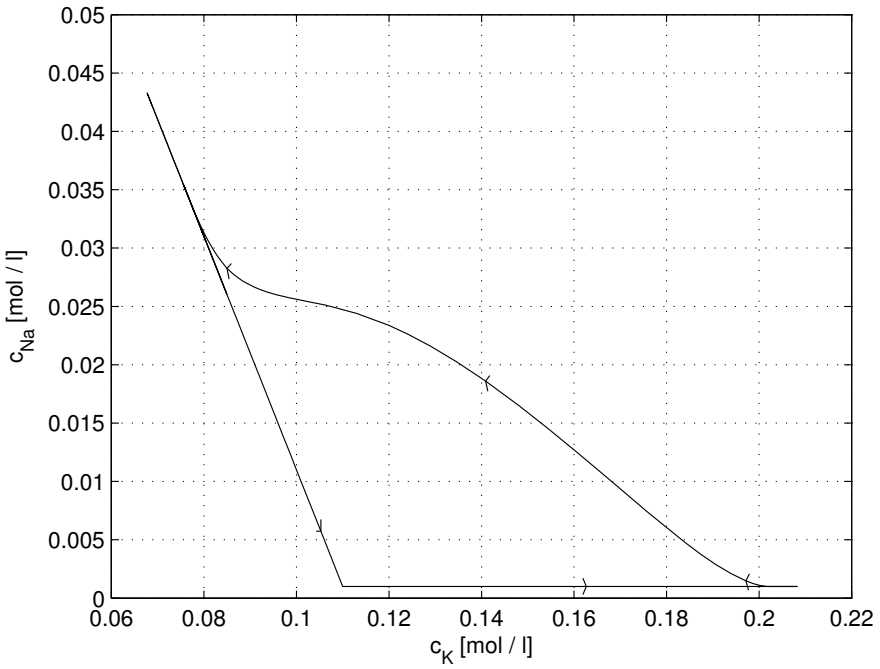
Das in den vorhergegangenen Kapiteln beschriebene biophysikalische Neuronenmodell wurde als Modell in der MATLAB/Simulink-Umgebung aufgebaut. Dazu wurde das grundlegende Zellmodell (Kapitel 3.1) mit dem vorgestellten Regler (Kapitel 3.1.4) als Modell für die Natrium-Kalium-Pumpe als aktivem Transportmechanismus aufgebaut. Zusätzlich wurden zwei Synapsen und ihre Antwort auf präsynaptische Aktionspotentiale integriert, um die Auswirkungen der spannungsgesteuerten Ionenkanäle und die Aktionspotentialerzeugung dieses Modells zu zeigen. Beide Synapsen wurden mittels eines einfachen Modells eines Dendrits an das postsynaptische Neuron angeschlossen. Die Stärke der Synapse kann durch den Leitwertparameter g_c für jede einzelne Synapse separat voreingestellt werden, so dass an dieser Stelle eine starke Synapse sowie eine schwache Synapse simuliert werden können. Alle in MATLAB/Simulink aufgebauten Blöcke basieren auf demselben Grundmodell mit *closed-loop* Regler. Die schematischen Darstellungen der Simulink-Blöcke des Zellmembranmodells, Axonhügels und der Aktionspotentialauslösung, sowie des Reglers und eine Übersicht des Minimalsystems sind im Anhang C aufgeführt. Im Folgenden sollen noch einmal die Eigenschaften des Grundmodells anhand von Simulationen überprüft werden und es soll gezeigt werden, dass der am linearisierten System entworfene Regler auch das nichtlineare System stabilisieren kann.

Der Phasenplot in Abbildung 3.7a zeigt die Trajektorie der Konzentrationen von Natrium und Kalium eines simulierten Neurons bei einer erhöhter Feuerschwelle. Bei der Erregung des Neurons mit externen Stimuli wird in dieser Simulation kein Aktionspotential ausgelöst und die Ionenkonzentrationen werden nach einer Auslenkung wieder im Arbeitspunkt stabilisiert. Dagegen zeigt der Phasenplot in Abb. 3.7b die Trajektorie der Konzentrationen für ein Neuron mit normaler Feuerschwelle, bei dem ein Aktionspotential durch spannungsinduzierte Konzentrationsänderung und das damit verbundene Überschreiten der Feuerschwelle ausgelöst wird. Das Aktionspotential stellt eine starke nichtlineare Störung des Systems dar, welche die Ionenkonzentrationen weiter auslenkt. Nach dem Abklingen des Aktionspotentials wird das System durch den Regler wieder im Arbeitspunkt stabilisiert.

In Abbildung 3.8a ist der Verlauf des Membranpotentials des simulierten Neuronenmodells mit Axon und kleinem Dendriten für eine Simulationsdauer von 10 ms dargestellt. Das Neuron wird über eine schwache Synapse und eine starke Synapse erregt. Dabei wird den Synapsen eine α-Funktion zu den Zeitpunkten $t_1 = 1\,\text{ms}$ und $t_2 = 3\,\text{ms}$ als Simulation eines Aktionspotentials von präsynaptischen Neuronen eingeprägt, welche den zeitlichen Verlauf an zusätzlichen geöffneten Natrium und Kalium-Kanälen beschreibt. Die Stärke der Synapse wird durch die Amplitude der gewichteten α-Funktion bestimmt. Das Überschreiten der Feuerschwelle am Neuron führt zum Auslösen eines Aktionspotentials durch die spannungsgesteuerten Ionenkanäle, wie in Kapitel 3.1.7 beschrieben. In Abbildung 3.8b ist die mit dem Potentialverlauf korrespondierende Leistungsaufnahme der NaK-ATPase abgebildet, welche sich direkt aus der Anzahl der Pumpzyklen des eingesetzten Reglers ergibt.

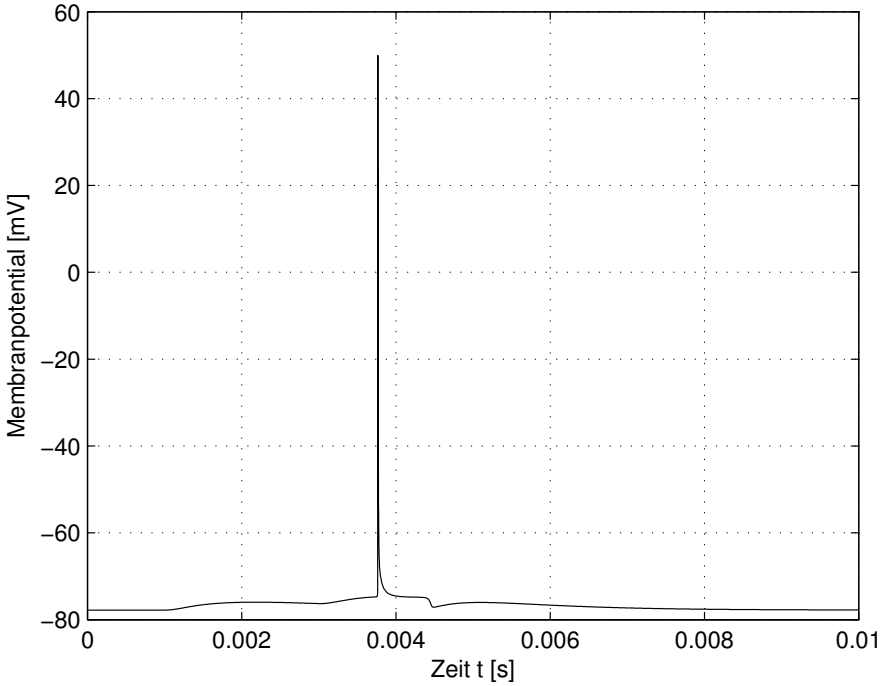


(a) System ohne Aktionspotentialerzeugung.

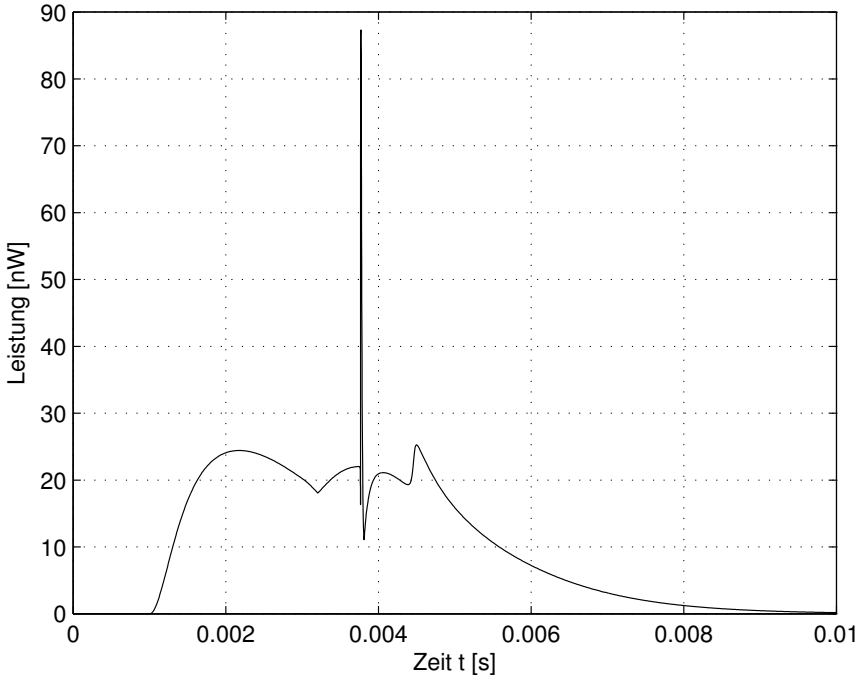


(b) System mit Aktionspotentialerzeugung.

Abbildung 3.7: Trajektorien der Natrium- und Kalium-Konzentration des simulierten Neuronenmodells (a) ohne Aktionspotentialerzeugung (überhöhte Feuerschwelle) und (b) mit Aktionspotentialerzeugung (normale Feuerschwelle).



(a) Membranpotential.



(b) Verlustleistung.

Abbildung 3.8: Simulationsergebnis des geregelten Systems mit zwei Synapsen (eine schwache Synapse, eine starke Synapse). Die Erregung des Neurons erfolgt zu den Zeitpunkten $t_1 = 1\,\mathrm{ms}$ und $t_2 = 3\,\mathrm{ms}$ über eine eingeprägte α-Funktion an den Synapsen.

Das Auslösen eines Aktionspotentials durch einen einzelnen Puls an einer einzelnen Synapse mag biologisch nicht plausibel erscheinen, jedoch lassen sich die hier verwendeten Beschreibungen der schwachen und starken Synapse auch als postsynaptisches Potential einer Gruppe von zum gleichen Zeitpunkt erregten Synapsen auffassen. Dieses wahrt die Konsistenz zu Beschreibungen aus den Neurowissenschaften.

Wie erwartet ergibt sich das Maximum des Energieumsatzes der NaK-ATPase zum Zeitpunkt der Auslösung des Aktionspotentials am Neuron, wenn der abrupte spannungsinduzierte Ioneneinstrom und Ionenausstrom von der Pumpe ausgeglichen werden muss. In Ruhe wird in diesem Neuronenmodell eine Leistung von 2.4 fW erzeugt und steigt beim Auslösen eines Aktionspotentials auf maximal 87.0 nW an. Die ermittelten Ergebnisse hängen allerdings direkt von den zu Grunde gelegten Leitwerten und den eingestellten elektrischen Eigenschaften und den geometrischen Abmessungen der Nervenzelle im Modell ab (z. B. Leitwerte und Zellvolumen, siehe (3.1.2)), und lassen sich somit leicht an neue biologische Erkenntnisse anpassen. Die für die hier dargestellten Simulationen verwendeten Parameter finden sich in Tab. 3.1.

Neben der dynamischen Erregung des Neurons über einzelne exzitatorische Aktionspotentiale wird in der Literatur häufig das Verhalten eines Neurons bei Erregung mit einem konstantem Eingangsstrom beschrieben. Dieses soll an dieser Stelle in einer Simulation nachgebildet werden, um die Grundfunktionalität des Modells zu zeigen. Für diesen Fall wurde der injizierte Strom über einen konstanten Aufschlag auf die Leitwerte der Ionenkanäle modelliert. Die Erregung des Neurons mit einem über längere Zeit konstanten Eingangsstrom führt zu einer in der Simulation beobachtbarer Aussendung von Aktionspotentialfolgen. Der Verlauf des Eingangsstroms und das Membranpotential sind in Abb. 3.9 dargestellt.

Um den aus den Simulationen ermittelten Energieumsatz des biophysikalische Neuronenmodells mit Natrium-Kalium-Regler mit den Angaben zum Energiebedarf biologischer Neurone aus der Literatur und dem Energiebedarf technischer Implementierungen vergleichen zu können, wurde die durch den Regler umgesetzte Energie während eines 1 ms dauernden Aktionspotentials sowie die Energie zur Erhaltung des Ruhepotentials ermittelt. Die Integration der momentanen Verlustleistung des Modells während der Erzeugung und des anschließenden Abbaus eines Aktionspotentials führt zu einem Wert von 34 pJ/Puls. Der ermittelte Wert liegt leicht über der Angabe in [8], welche den Zellkörper und das Axon getrennt mit unterschiedlichen Parametern betrachten, aber nur ein Gesamtergebnis angeben. Der in der Simulation mit einer Spitzenleistung von 84 nW auftretende starke Puls des Aktionspotentials benötigt nur einen Bruchteil der Zeit des gesamten Aktionspotentials von ca. 1 ms. Dagegen beträgt die für das Ruhepotential aufgewandte Energie des Grundmodells ohne Einbringen einer Störung je Sekunde lediglich 2,4 fJ. Gegenüber der Ruheleistung ist die pro Aktionspotential aufgewandte Energie groß, jedoch wird die während des Aktionspotentials umgesetzte Energie durch den Abbau der weiter laufenden synaptischen Erregung überlagert, so dass das der angegebene Wert die Summe aus beiden Prozessen darstellt. Beide Prozesse lassen sich natürlich nicht trennen, so dass in den weiteren Betrachtungen auf den angegebenen Wert zurückgegriffen wird.

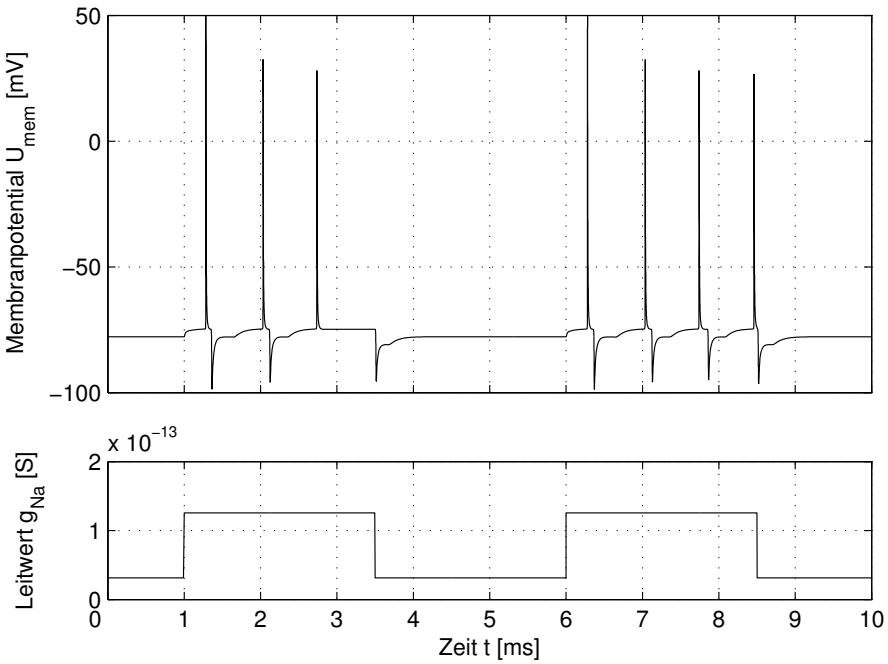


Abbildung 3.9: Erregung des Neurons mit konstantem Strom.

3.2 Modellierung des Energieumsatzes mit elektrischen Schaltkreisen

Im nachfolgenden Abschnitt wird vom Modell der Ionenpumpe des biologischen Neurons auf den Energieumsatz eines in CMOS-Technologie umgesetzten Neurons übergegangen. Dazu wird zuerst der Energiebedarf des biologischen Pendants im Ruhezustand betrachtet und anschließend das elektrische Modell eines LIAF-Neurons eingeführt. Durch Betrachtung der Vorgänge im elektrischen Modell des LIAF-Neurons wird eine Gleichung für das Übertragungsverhalten von in das einlaufenden Strompulsen auf den Ausgang des Neurons hergeleitet. Diese wird im Anschluss für die allgemeine Form des SRM-Neurons erweitert und wird im folgenden Kapitel zur Abschätzung der Verlustleistung des implementierten LIAF-Neurons genutzt.

3.2.1 Abschätzungen zum Energiebedarf biologischer Neurone am Ersatzschaltbild im steady-state

Im Folgenden wird der Energieumsatz einer Granularzelle und einer Purkinje-Zelle des menschlichen Gehirns im *steady-state*, dem Gleichgewichtsfall des Membranpotentials abgeschätzt. Der *steady-state* liegt dann vor, wenn der aktive Transportmechanismus der NaK-ATPase die passiven Ionenströme durch die Membran ausgleichen und so einen metastabilen Zustand schaffen. Wie im Abschnitt über die NaK-ATPase bereits diskutiert, erfolgt hier der Transport von 3 Na^+-Ionen aus dem intrazellulären Raum zum extrazellulären Raum und der Transport von 2 K^+-Ionen aus dem extrazellulären Raum in den

intrazellulären Raum unter Hydrolyse von ATP zu ADP und einem Phosphatrest. Im steady-state wird so der Einstrom von Natrium im Verhältnis von 3/2 zu Kalium durch die NaK-ATPase ausgeglichen. Damit gilt die Beziehung:

$$I_{\mathrm{Na},0} = -f \cdot I_{\mathrm{K},0} \quad \text{mit} \quad f = \frac{3}{2} \tag{3.42}$$

Nun kann mit den Beziehungen

$$I_{\mathrm{Na}} = -g_{\mathrm{Na}} \left(U_{\mathrm{mem}} - U_{\mathrm{Na}}\right) \quad , \quad I_{\mathrm{K}} = -g_{\mathrm{K}} \left(U_{\mathrm{mem}} - U_{\mathrm{K}}\right) \tag{3.43}$$

und den Nernst-Potentialen $U_{\mathrm{Na},0} = \frac{\mathrm{R}T}{\mathrm{F}} \ln \frac{c_{\mathrm{Na,a}}}{c_{\mathrm{Na},0}}$ und $U_{\mathrm{K},0} = \frac{\mathrm{R}T}{\mathrm{F}} \ln \frac{c_{\mathrm{K,a}}}{c_{\mathrm{K},0}}$ die Gleichgewichtsbedingung zum Ruhemembranpotential unter Berücksichtigung der NaK-ATPase umgestellt werden:

$$g_{\mathrm{Na}} \left(U_{\mathrm{mem}} - U_{\mathrm{Na},0}\right) + f \cdot g_{\mathrm{K}} \cdot \left(U_{\mathrm{mem}} - U_{\mathrm{K},0}\right) = 0 \tag{3.44}$$

$$U_{\mathrm{mem}} = \frac{g_{\mathrm{Na}} U_{\mathrm{Na},0} + f \cdot g_{\mathrm{K}} U_{\mathrm{K},0}}{g_{\mathrm{Na}} + f \cdot g_{\mathrm{K}}} \tag{3.45}$$

Mit diesem Ruhemembranpotential wird der Natrium-Strom berechnet, der durch die NaK-ATPase ausgeglichen wird. Die durchschnittliche Leistungsaufnahme eines einzelnen Neurons im Ruhezustand errechnet sich zu

$$P_{\mathrm{Neuron,steadystate}} = -46\,\mathrm{kJ/mol} \cdot \frac{A}{\mathrm{F}} \cdot \frac{g_{\mathrm{Na}}}{3} \cdot \left(U_{\mathrm{mem}} - U_{\mathrm{Na},0}\right), \tag{3.46}$$

wobei der hinzugekommene Parameter A die Mantelfläche der Zelle beschreibt.

Daraus ergibt sich für eine einzelne Granularzelle mit den Parametern nach Tabelle 3.1 eine Leistung von $1,08\,\mathrm{pW}$ und für eine Purkinje-Zelle eine durchschnittliche Leistung von $69,3\,\mathrm{pW}$.

Diese Ergebnisse liegen in der Größenordnung der Angaben aus [8] für den Grundumsatz einer Zelle zur Erhaltung des Ruhepotentials.

3.3 Modellierung eines LIAF Neurons mit elektrischen Ersatzschaltkreisen

Gegeben sei das in Abb. 3.10 dargestellte Modell der Zellmembran eines LIAF Neurons. Dieser Schaltkreis modelliert – wenn er auch auf den ersten Blick einfach erscheint – die Zellmembran mit ihren passiven elektrischen Eigenschaften gut. Der im Eingangspfad

Tabelle 3.1: Ionenkonzentrationen, Nernst-Potentiale und Leitwerte für verschiedene Ionenarten in Zellen im steady-state [86].

Parameter	Wert	Beschreibung
$c_{\mathrm{Na,0}}$	$12\,\mathrm{mmol/l}$	intrazelluläre Natriumkonzentration
$c_{\mathrm{Na,a}}$	$145\,\mathrm{mmol/l}$	extrazelluläre Natriumkonzentration
$c_{\mathrm{K,0}}$	$155\,\mathrm{mmol/l}$	intrazelluläre Kaliumkonzentration
$c_{\mathrm{K,a}}$	$4\,\mathrm{mmol/l}$	extrazelluläre Kaliumkonzentration
$U_{\mathrm{Na,0}}$	$64,42\,\mathrm{mV}$	Nernst-Potential für Natrium (T=300 K)
$U_{\mathrm{K,0}}$	$-95,54\,\mathrm{mV}$	Nernst-Potential für Kalium (T=300 K)
g_{Na}	$14,32\,\mu\mathrm{S/cm^2}$	Natriumleitwert im *steady-state*
g_{K}	$170,90\,\mu\mathrm{S/cm^2}$	Kaliumleitwert im *steady-state*
d_{Granular}	$10\,\mu\mathrm{m}$	Durchmesser einer Granularzelle (Körnerzelle)
d_{Purkinje}	$80\,\mu\mathrm{m}$	Durchmesser einer Purkinje-Zelle
U_{mem}	$-89,97\,\mathrm{mV}$	Gleichgewichtspotential nach (3.45) (T=300 $\mathrm{K^+}$)

liegende Widerstand beschreibt den Vorgang, der beim Eintreffen eines Aktionspotentials an der Synapse passiert, an welcher der Spannungshub des Aktionspotentials über die Freisetzung chemischer Botenstoffe zu einer Konformationsänderung der Ionenkanäle und damit zu einem Einstrom von Ionen in die Zelle führt. Genauso kann der Widerstand als sehr einfaches Modell des Dendriten aufgefasst werden, wobei für eine genauere Beschreibung des Dendriten auf die Lösung der Kabelgleichungen zurückgegriffen werden sollte.

Die in Abb. 3.10 eingezeichnete Kapazität mit parallel geschaltetem Leitwert beschreibt im ersten Schritt die passive Zellmembran, die durch den Einstrom von Ionen und dem damit verbundenen Ladungsunterschied zwischen intra- und extrazellulärem Raum auf ein bestimmtes Membranpotential aufgeladen wird. Gleichzeitig sorgt der Leitwert für eine Abnahme des Ladungsunterschieds und einen effektiven Ausstrom von Ionen mit i_{leak} durch die Natrium-Kalium-Pumpe und modelliert so praktisch einen aktiven Mechanismus der Zellmembran. Obwohl durch diesen Leitwert praktisch nur der Natrium-Austausch an der Zellmembran betrachtet wird, zeigt dieses elektrische Modell die Eigenschaften eines Neurons, da in erster Näherung angenommen werden kann, dass das Austauschverhältnis von Natrium und Kalium aneinander gekoppelt ist.

Weiterhin wird durch den Leitwert das Unterschwellenverhalten der Zellmembran (vgl. Kapitel 1.3) durch spannungsgesteuerte Kaliumkanäle nachgebildet, welche dafür sorgen, dass erst ein starker Eingangsstrom die Zellmembran stark depolarisieren und zum Auslösen eines Aktionspotentials führen kann. Dieses Verhalten wird auch in Kap. 3.4.2 aus dem einfachen elektrischen Modell abgeleitet.

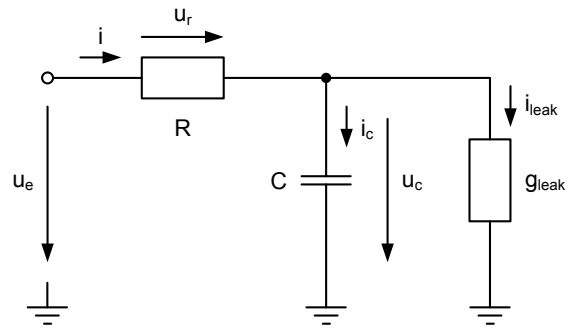


Abbildung 3.10: Vereinfachtes Modell der Zellmembran mit Berücksichtigung des Leckstroms.

3.3.1 Passives Entladen der Kapazität über Leckströme

Der Leitwert g_{leak} für die Entladung durch Leckströme wird durch die Abschätzung der ITRS für die obere Grenze der Leckströme $i_{\text{leak, max}}$ bei der maximal auf der Kapazität auftretenden Spannung $u_{\text{c, max}}$ bestimmt.

$$g_{\text{leak}} = \frac{i_{\text{leak, max}}}{u_{\text{c, max}}} = \frac{i_{\text{leak}}(t=0)}{u_{\text{dd}}} \tag{3.47}$$

Der so von der Spannung auf der Kapazität abhängige Leckstrom wird im ersten Schritt als alleiniger Grund für die Entladung der Kapazität betrachtet. Zum Zeitpunkt $t = 0$ sei die Kapazität auf einen Wert $u_{\text{c, max}}$ aufgeladen.

$$u_c(t) + \frac{C}{g_{\text{leak}}}\frac{du_c(t)}{dt} = 0, \qquad u_c(t=0) = u_{\text{c, max}} \tag{3.48}$$

Die Lösung der Differentialgleichung (3.48) lautet:

$$u_c(t) = u_{\text{c, max}} \cdot \mathrm{e}^{-\frac{t}{\tau}} \quad \text{mit} \quad \tau = \frac{C}{g_{\text{leak}}} \tag{3.49}$$

3.3.2 Aktives Laden der Kapazität unter Berücksichtigung von Leckströmen

Dem aktiven Aufladen der Kapazität C durch die hier idealisierte Spannungsquelle u_e wirken die Leckströme, berücksichtigt durch den Leitwert g_{leak}, entgegen. Als mathematische Beschreibung des Aufladevorgangs folgt mit der Randbedingung $u_{\text{c}}(t=0) = u_{\text{c,0}}$ aus der Differentialgleichung

$$u_c(t)\,(1 + R \cdot g_{\text{leak}}) + RC\frac{du_c(t)}{dt} = u_e \tag{3.50}$$

die Lösung

$$u_c(t) = \left(u_{\mathrm{c},0} - \frac{u_e}{1 + R \cdot g_{\mathrm{leak}}}\right) \cdot \mathrm{e}^{-\frac{t}{\tau}} + \frac{u_e}{1 + R \cdot g_{\mathrm{leak}}} \quad \text{mit} \quad \tau = \frac{RC}{1 + R \cdot g_{\mathrm{leak}}}. \tag{3.51}$$

Im Folgenden wird angenommen, der Eingang des Neurons sei mit einer idealen Stromquelle mit konstanten Strom I beschaltet. Dadurch reduziert sich die Differentialgleichung unter Vernachlässigung des Leitungswiderstands zu der Form

$$u_c(t) + \frac{C}{g_{\mathrm{leak}}} \frac{du_c(t)}{dt} = \frac{I}{g_{\mathrm{leak}}}. \tag{3.52}$$

Die Lösung dieses Terms ist unter der Bedingung $u_{\mathrm{c}}(t = 0) = u_{\mathrm{c},0}$ in (3.53) gegeben.

$$u_c(t) = u_{\mathrm{c},0} \cdot \mathrm{e}^{-\frac{t}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \cdot \left(1 - \mathrm{e}^{-\frac{t}{\tau}}\right) \quad \text{mit} \quad \tau = \frac{C}{g_{\mathrm{leak}}} \tag{3.53}$$

3.4 Verlustleistung

Die Verlustleistung des hier beschriebenen Modells der Membrankapazität ergibt sich durch Integration des Konstantstroms und der Spannung über der Kapazität über der Zeit, die bis zum Erreichen der Feuerschwelle U_{TH} vergeht. Der zweite Anteil der Verlustleistung wird durch die aktive Entladung der Kapazität über einen Schalttransistor erzeugt und separat betrachtet.

3.4.1 Gleichstrombetrieb

Bei Erregung des Neurons mit einem konstanten Strom der Stärke I lässt sich die Zeit zu Erreichen der Feuerschwelle leicht aus der Lösung der Differentialgleichung (3.52) ermitteln. Die Zeit bis zum ersten Puls t_{pulse} ergibt sich aus (3.53) zu:

$$t_{\mathrm{pulse}} = -\ln\left(\frac{U_{\mathrm{TH}} - \frac{I}{g_{\mathrm{leak}}}}{u_{\mathrm{c},0} - \frac{I}{g_{\mathrm{leak}}}}\right) \cdot \tau \tag{3.54}$$

Aus (3.53) ergibt sich eine notwendige direkte Forderung an den minimalen Eingangsstrom zum Erreichen der Feuerschwelle für $t \to \infty$. Dieser minimale notwendige Eingangsstrom ist von den Leckströmen, repräsentiert durch die Leitfähigkeit g_{leak}, und der Feuerschwelle U_{TH} abhängig:

$$I > U_{\mathrm{TH}} \cdot g_{\mathrm{leak}} \tag{3.55}$$

Die während des Ladens der Kapazität erbrachte Arbeit lässt sich nun mit

$$\begin{aligned} W_{\text{load}} &= \int_{t=0}^{t=t_{\text{pulse}}} I^2 \frac{1}{g_{\text{leak}}} \left(1 - \exp\left(-\frac{t}{\tau}\right)\right) dt + \int_{t=0}^{t=t_{\text{pulse}}} I \cdot u_{\text{c},0} \cdot \exp\left(-\frac{t}{\tau}\right) dt \\ &= \frac{I^2}{g_{\text{leak}}} \left[t_{\text{pulse}} + \tau \left(\exp\left(-\frac{t_{\text{pulse}}}{\tau}\right) - 1\right)\right] \\ &\quad + I \cdot u_{\text{c},0} \cdot \tau \left(1 - \exp\left(-\frac{t_{\text{pulse}}}{\tau}\right)\right) \end{aligned} \tag{3.56}$$

angeben. Während des Feuerns, d. h. dem Aussenden eines Pulses durch das Neuron wird die Kapazität aktiv entladen, so dass hier zusätzliche Arbeit entsteht. Dabei wird der passive Leitwert g_{leak} durch das Öffnen von Kanälen stark vergrößert. Der zusätzliche Leitwert wird in den folgenden Gleichungen mit $g'_{\text{leak}} = g_{\text{leak}} + g_{\text{discharge}}$ beschrieben.

$$\begin{aligned} W_{\text{reset}} &= g'_{\text{leak}} \cdot U_{\text{TH}}^2 \int_{t=0}^{t=t_{\text{fire}}} \exp^2\left(-\frac{t}{\tau'}\right) dt \\ &= g'_{\text{leak}} \cdot U_{\text{TH}}^2 \cdot \frac{\tau'}{2} \left(1 - \exp\left(-\frac{2t_{\text{fire}}}{\tau'}\right)\right) \\ &\text{mit} \quad \tau' = \frac{C}{g'_{\text{leak}}} \end{aligned} \tag{3.57}$$

Dieses ergibt für $t_{\text{fire}} \to \infty$ die Gesamtenergie auf einer Kapazität von $\frac{1}{2}CU^2$.

Da die Realisierung des Neurons in der Weise gewählt wurde, dass die Dauer des Pulses durch die aktive Entladung der Membrankapazität bestimmt wird, ist die Dauer der Entladung der Kapazität niemals kürzer als t_{fire} und der Term (3.57) immer gültig.

Da der erste Term von (3.56) durch einlaufende Pulse präsynaptischer Neurone verursacht wird, kann ein Teil der Arbeit dem Informationsumsatz zugerechnet werden, der Teil, der durch Leckströme verursacht wird, kann dem Grundumsatz zugerechnet werden.

Im Abb. 3.11 ist die Verlustleistung eines Neurons ohne Erzeugung eines Aktionspotentials und ohne Beschränkung der Spannung über der Membrankapazität dargestellt. Es zeigt den Verlauf der Verlustleistung, welche maximal durch das Umladen der Kapazität entsteht, die zusätzliche Verlustleistung durch den Leaky-Term sowie die Summe der Verlustleistung der gesamten Schaltung in Abhängigkeit von der Eingangspulsrate. Die Länge der Pulse wurde hier mit $1\,\mu$s angenommen, der Strom mit $I = 200\,$nA, der Leakage-Leitwert mit $g_{\text{leak}} = 1\,$nS und die Kapazität mit $C = 200\,$fF. Es zeigt sich, dass das Maximum der auf der Kapazität umgesetzten Verlustleistung bei einer Eingangspulsrate von etwa 650.000 Pulsen/s liegt. Oberhalb dieser Eingangspulsrate nimmt der Anteil der auf der Kapazität umgesetzten Energie, der dem Informationsumsatz zugerechnet wird, wieder ab. Daraus kann geschlossen werden, dass es sinnvoll ist, die Eingangspulsrate von Neuronen zu

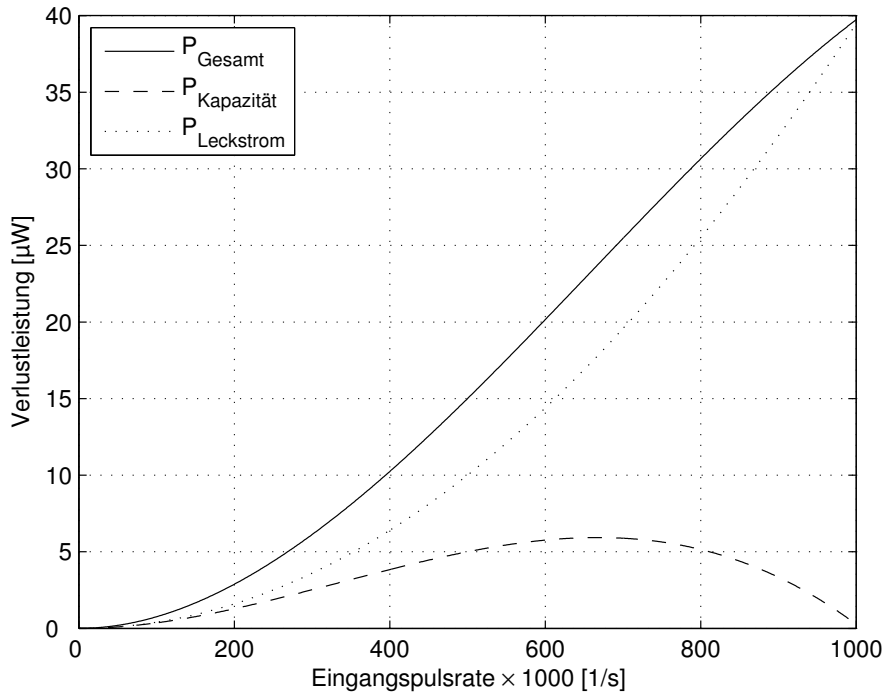


Abbildung 3.11: Verlustleistung eines Neurons ohne Erzeugung eines Aktionspotentials (Feuerschwelle heraufgesetzt) und ohne Beschränkung der Spannung über der Membrankapazität.

begrenzen, um den Anteil der umgesetzten Energie für den Informationsumsatz möglichst groß werden zu lassen.

Für eine Feuerrate am Eingang des empfangenden Neurons, welche sich der Grenze der Inversen der Feuerdauer eines Neurons annähert (hier: $1 \cdot 10^6\,\mathrm{s}^{-1}$), nähert sich die gesamte Verlustleistung der Verlustleistung an, welche nur noch durch einen konstanten Eingangsstrom und den Leitwert g_{leak} hervorgerufen wird (hier: $40\,\mu\mathrm{W}$). Auf der Kapazität wird in diesem Fall keine Energie mehr umgesetzt.

3.4.2 Herleitung der Übertragungskennlinie

Im folgenden Abschnitt soll die Übertragungskennlinie des LIAF Neurons unter Berücksichtigung der Eingangsulsrate, passiver Entladung und einstellbarer Feuerschwelle hergeleitet werden. Diese wurde von Maass [70] anhand von numerischen Betrachtungen beschrieben.

Der Grenzwert der Aufladung der Membrankapazität wird durch den Wert der Kapazität, den Leitwert der passiven Entladung sowie den Eingangsstrom und das Puls-Pausen Verhältnis der Eingangspulsrate bestimmt. Dabei können langsam einlaufende Pulse am Eingang des Neurons kein Aktionspotential auslösen, da die Feuerschwelle in diesem Fall niemals erreicht wird. Es ergibt sich das sog. Unterschwellenverhalten des LIAF Neurons (vgl. Kap. 1.3). Mathematisch lässt sich dieses nachweisen, indem eine Reihe für die Auf- und Entladevorgänge entwickelt wird.

Betrachtet wird ein Neuron mit der Dynamik nach (3.48) und (3.52), welches durch eine konstante Pulsrate der Frequenz $f = 1/T$ erregt wird. Die Periodendauer T setzt sich aus den Anteilen der Feuerdauer eines präsynaptischen Neurons t_{fire} und der Pausenzeit t_{relax} zusammen. Zunächst wird die Gleichung für das Aufladen des Neurons durch eine Konstantstromquelle betrachtet. Die Membrankapazität des Neurons sei zu Beginn der Betrachtung auf einen Wert u_0 aufgeladen. Nach (3.53) ergibt sich für die erste Aufladung der Membrankapazität

$$u_{c,f}^{(1)} = u_c(t = t_{\text{fire}}) = \left(u_0 - \frac{I}{g_{\text{leak}}}\right) \cdot \mathrm{e}^{-\frac{t_{\text{fire}}}{\tau}} + \frac{I}{g_{\text{leak}}} \quad \text{mit} \quad \tau = \frac{C}{g_{\text{leak}}}. \tag{3.58}$$

In der Pulspause klingt das Membranpotential nach (3.49) ab und erreicht nach einer gesamten Periode T den Wert

$$u_{c,T}^{(1)} = u_{c,f}^{(1)} \cdot \mathrm{e}^{-\frac{t_{\text{relax}}}{\tau}} = \left(u_0 - \frac{I}{g_{\text{leak}}}\right) \cdot \mathrm{e}^{-\frac{T}{\tau}} + \frac{I}{g_{\text{leak}}} \cdot \mathrm{e}^{-\frac{t_{\text{relax}}}{\tau}}, \tag{3.59}$$

welches die Initialbedingung für die nächste Aufladung $u_{c,f}^{(2)}$ ist.

Da nur während einer Aufladung die Feuerschwelle des Neurons von unten überschritten werden kann, ist zur Beantwortung der Frage, ob eine bestimmte Eingangspulsrate zum Auslösen eines Aktionspotentials ausreicht, die N-te Aufladung zu betrachten. Dazu wird der Wert $u_{c,f}^{(N)}$ bestimmt, der sich durch sukzessives Einsetzen der oben beschriebenen Auflade- und Entladevorgänge ergibt. Es ergibt sich ein Term (Herleitung siehe Anhang A.1), der nach Summenbildung über die einzelnen Glieder den Ausdruck für die Spannung $u_{c,f}^{(N)}$ auf der Membrankapazität nach der N-ten Aufladung ergibt:

$$u_{c,f}^{(N)} = u_0 \cdot \mathrm{e}^{-\frac{(N-1)T + t_{\text{fire}}}{\tau}} + \frac{I}{g_{\text{leak}}} \cdot \left(1 - \mathrm{e}^{-\frac{t_{\text{fire}}}{\tau}}\right) \cdot \sum_{n=0}^{N-1} \mathrm{e}^{-\frac{nT}{\tau}} \tag{3.60}$$

Für $N \to \infty$ konvergiert die Laurent-Reihe gegen die Lösung einer geometrischen Reihe. Mit der Annahme, dass die Kapazität zu Beginn auf $u_0 = 0\,\mathrm{V}$ aufgeladen ist, vereinfacht sich (3.60) zu:

$$u_{c,f}^{(\infty)} = \frac{I}{g_{\text{leak}}} \cdot \left(1 - \mathrm{e}^{-\frac{t_{\text{fire}}}{\tau}}\right) \cdot \frac{\mathrm{e}^{\frac{T}{\tau}}}{\mathrm{e}^{\frac{T}{\tau}} - 1} \tag{3.61}$$

Nun wird (3.61) gerade so gewählt, dass die Feuerschwelle U_{TH} nicht überschritten wird. Es ergibt sich die Ungleichung (3.62), welche eine Aussage darüber erlaubt, bei welcher Eingangspulsrate die Feuerschwelle gerade noch nicht erreicht wird.

$$\frac{I}{g_{\text{leak}}} \cdot \left(1 - \mathrm{e}^{-\frac{t_{\text{fire}}}{\tau}}\right) \cdot \frac{\mathrm{e}^{\frac{T}{\tau}}}{\mathrm{e}^{\frac{T}{\tau}} - 1} < U_{\text{TH}} \tag{3.62}$$

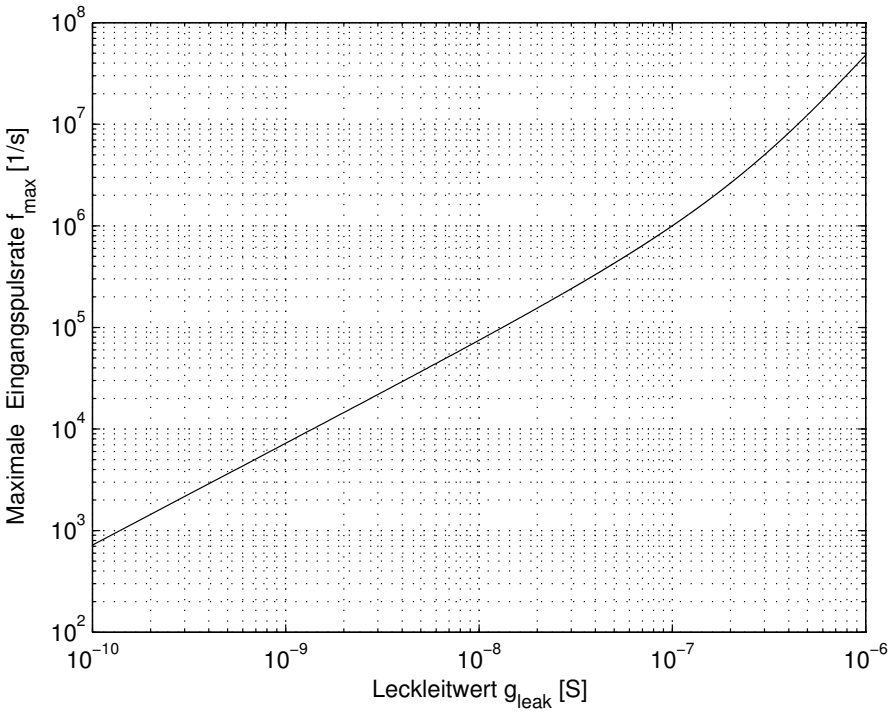


Abbildung 3.12: Maximale Eingangspulsrate, die ohne Auslösen eines Aktionspotentials möglich ist. Parameter $C = 200\,\mathrm{fF}$, $I = 200\,\mathrm{nA}$, $U_{\mathrm{TH}} = 2\,\mathrm{V}$, $t_{\mathrm{fire}} = 1\,\mu\mathrm{s}$.

Bei gegebenen Größen für die Membrankapazität, den Leitwert für die passive Entladung sowie den gepulsten Eingangsstrom lässt sich die Eingangspulsrate, bei der die Feuerschwelle gerade nicht erreicht wird ausdrücken als:

$$f < \left(\ln \left(\frac{U_{\mathrm{TH}}}{U_{\mathrm{TH}} - \frac{I}{g_{\mathrm{leak}}} \left(1 - \mathrm{e}^{-\frac{t_{\mathrm{fire}}}{\tau}} \right)} \right) \cdot \tau \right)^{-1} \tag{3.63}$$

Diese Ungleichung ist für ausgewählte Leitwerte g_{leak} in Abb. 3.12 dargestellt. Unter den hier angenommenen Bedingungen von $t_{\mathrm{fire}} = 1\,\mu\mathrm{s}$ kann eine Pulsrate von $10^6\,\mathrm{s}^{-1}$ jedoch niemals erreicht werden, da bei $T = 1\,\mu\mathrm{s}$ der Gleichstrombetrieb des Neurons beginnt. Damit die oben angegebene Gleichung Gültigkeit hat, muss zusätzlich die Bedingung

$$\frac{I}{g_{\mathrm{leak}}} < \frac{U_{\mathrm{TH}}}{1 - \mathrm{e}^{-\frac{t_{\mathrm{fire}}}{\tau}}} \tag{3.64}$$

erfüllt sein.

Es stellt sich die Frage, wie das Ausgangsverhalten des pulsenden Neurons in Abhängigkeit von der Eingangspulsrate beschrieben werden kann. Dazu wird (3.60) bis zum N-ten Glied betrachtet um festzustellen, ob der $(N-1)$-te präsynaptische Puls ein Aktionspotential auslösen kann. Die Summe $\sum_{n=0}^{N-1} \mathrm{e}^{-\frac{nT}{\tau}}$ kann zu einer geometrischen Reihe mit $N-1$

Gliedern umgeformt werden, deren Ergebnis

$$S_{N-1} = \frac{\mathrm{e}^{-N\frac{T}{\tau}} - 1}{\mathrm{e}^{-\frac{T}{\tau}} - 1} \tag{3.65}$$

ist.

Durch Grenzwertbetrachtung des Ergebnisses nach $\lim_{N\to\infty} S_{N-1} = \frac{1}{1-\mathrm{e}^{-\frac{T}{\tau}}}$ und Vergleich mit dem Ergebnis der Entwicklung der unendlichen Reihe lässt sich die Korrektheit der obigen Rechnung nachweisen. Die in (3.66) angegebene Ungleichung stellt die Bedingung für das Überschreiten der Feuerschwelle und das Auslösen eines Aktionspotentials beim N-ten präsynaptischen Puls dar

$$u_0 \cdot \mathrm{e}^{-\frac{(N-1)T+t_{\mathrm{fire}}}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \cdot \left(1 - \mathrm{e}^{-\frac{t_{\mathrm{fire}}}{\tau}}\right) \cdot \frac{\mathrm{e}^{-N\frac{T}{\tau}} - 1}{\mathrm{e}^{-\frac{T}{\tau}} - 1} \geq U_{\mathrm{TH}} \tag{3.66}$$

und kann nach der Anzahl der präsynaptischen Pulse N aufgelöst werden, bei dem ein Aktionspotential ausgelöst wird:

$$N = -\frac{\tau}{T} \ln\left(\frac{I \cdot \left(1 - \mathrm{e}^{-\frac{t_{\mathrm{fire}}}{\tau}}\right) + U_{\mathrm{TH}} \cdot g_{\mathrm{leak}} \cdot \left(\mathrm{e}^{-\frac{T}{\tau}} - 1\right)}{u_0 \cdot g_{\mathrm{leak}} \cdot \mathrm{e}^{\frac{t_{\mathrm{fire}}}{\tau}} \cdot \left(1 - \mathrm{e}^{\frac{T}{\tau}}\right) + I \cdot \left(1 - \mathrm{e}^{-\frac{t_{\mathrm{fire}}}{\tau}}\right)}\right) \tag{3.67}$$

Damit lässt sich das Verhältnis von Eingangs- und Ausgangspulsrate beschreiben. Damit diese Gleichung Gültigkeit besitzt, darf das Neuron nicht im Unterschwellenbetrieb gehalten werden, und (3.63) darf gerade nicht gelten. Durch Wahl der Eingangspulsrate kann sichergestellt werden, dass das Neuron nach einer bestimmten Anzahl von präsynaptischen Pulsen feuert. Die Übertragungsfunktion des LIAF Neurons ergibt sich so zu

$$f_{\mathrm{out}} = \frac{1}{N} \cdot f_{\mathrm{in}}. \tag{3.68}$$

Abbildung 3.13 zeigt das theoretische Maximum der Ausgangspulsrate eines LIAF Neurons bei Erregung mit einer konstanten Eingangspulsrate bei verschiedenen Leckströmen. Dabei zeigt sich, dass mit abnehmenden Leckströmen das Eingangs-Ausgangs Verhalten des Neurons nahezu linear wird, während bei hohen Leckströmen eine starke Nichtlinearität knapp über dem Überschreiten der Feuerschwelle zu beobachten ist. Hier wurde vorausgesetzt, dass die Feuerzeit t_{fire} am Eingang des Neurons die gleiche Zeit ist, wie am Ausgang des Neurons. Ist der Ausgangspuls kürzer, so ist zu erwarten, dass die Ausgangspulsrate im Vergleich zum oben angegebenen Zusammenhang ansteigt, während eine längere Pulsdauer am Ausgang zu einer Verringerung der berechneten Pulsrate führt. In der entwickelten Gleichung wurde ebenfalls nicht explizit die mögliche Überschneidung von Eingangs- und Ausgangspuls berücksichtigt. Diese sollte aber durch die Wahl eines reellen Ergebnisses (z. B. Auslösen eines Aktionspotentials nach $1,4$ Pulsen) kompensiert werden. Die hier gewonnenen theoretischen Ergebnisse für das Eingangs-Ausgangs-Verhalten von LIAF Neuronen sind konsistent mit den Ergebnissen aus numerischer Integration der Modellgleichungen eines Hodgkin-Huxley-Modells (siehe „gain function“, Kapitel 1.2.4.1 in [70]).

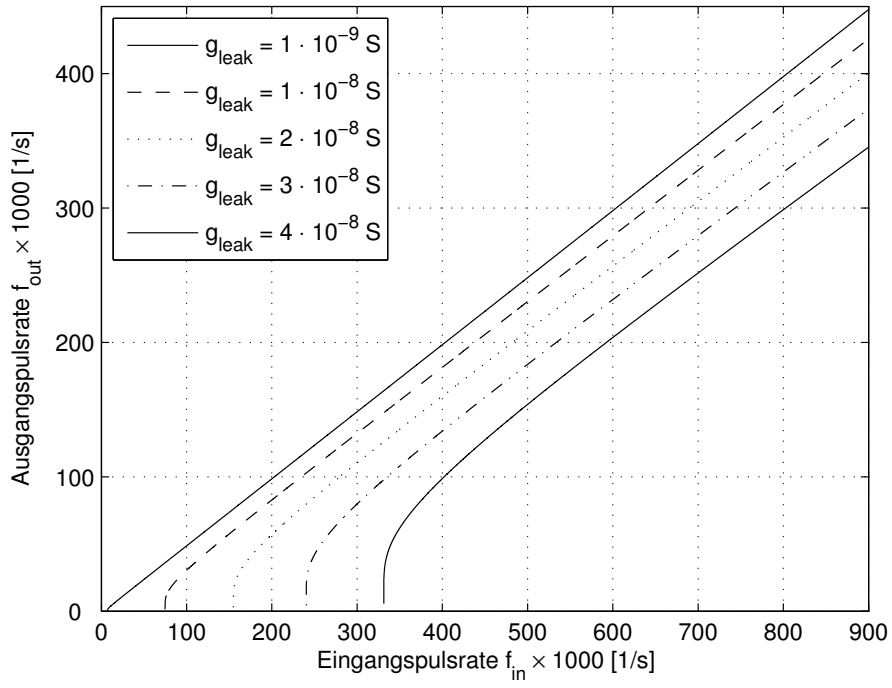


Abbildung 3.13: Darstellung der theoretisch erzielbaren Ausgangspulsrate eines Neurons gegenüber der Eingangspulsrate bei verschieden großen Leckströmen. $U_{\mathrm{TH}} = 2,0\,\mathrm{V}, I_{\mathrm{in}} = 200\,\mathrm{nA}, t_{\mathrm{fire}} = 1\,\mu\mathrm{s}$

3.5 Erweiterung der Betrachtung am LIAF Modell zur allgemeinen Form des Spike-Response Modells

Da das LIAF Modell als Sonderfall des Spike-Response Modells (SRM) angesehen werden kann [32, 70], soll an dieser Stelle eine Verknüpfung der gewonnenen Ergebnisse mit dem SRM geschaffen werden. Das SRM beschreibt die zeitliche Veränderung des Membranpotentials $u(t)$ eines Neurons aus der Summe zweier entscheidender Ereignisse (siehe (3.69). Der erste Term beschreibt mit η_i die Eigenantwort des Membranpotentials auf ein von sich selbst zum Zeitpunkt $t_i^{(f)}$ ausgelöstes Aktionspotentials. Die Doppelsumme im zweiten Term beschreibt die Post-Impuls-Antwort ε_{ij} des Membranpotentials beim Eintreffen von mit w_{ij} gewichteten Pulsen anderer Neurone zum Zeitpunkt $t_j^{(f)}$. Dabei beschreibt die Menge F_i die Menge aller Feuerzeitpunkte des Neurons selbst und F_j die Menge aller Feuerzeitpunkte von mit diesem Neuron verbundenen Neuronen der Menge Γ_i.

$$u(t) = \sum_{t_i^{(f)} \in F_i} \eta_i \left(t - t_i^{(f)}\right) + \sum_{j \in \Gamma_i} \sum_{t_j^{(f)} \in F_j} w_{ij} \varepsilon_{ij} \left(t - t_j^{(f)}\right) \tag{3.69}$$

Betrachtet wird im Folgenden der einfache Fall, in dem ein einzelnes Neuron mit dem empfangenden Neuron verbunden ist ($j = 1$). Mit einem kurzen Strompuls i der Dauer t_{fire}

des sendenden Neurons wird das Membranpotential des empfangenden Neurons erhöht und es stellt sich durch den zusätzlich wirkenden Verlustleitwert g_{leak} ein Membranpotential von

$$u(t_{\text{fire}}) = u_0 \exp\left(-\frac{t_{\text{fire}}}{\tau}\right) + \frac{i}{g_{\text{leak}}}\left(1 - \exp\left(-\frac{t_{\text{fire}}}{\tau}\right)\right) \quad \text{mit} \quad \tau = \frac{C}{g_{\text{leak}}} \tag{3.70}$$

ein.

Mit jedem eintreffenden Puls wird das Membranpotential des empfangenden Neurons um den zweitem Term aus (3.70) erhöht, während das Membranpotential ständig exponentiell mit dem ersten Term zerfällt. Daraus lassen sich für das SRM unter der Annahme, dass die Dauer des Aktionspotentials t_{fire} klein gegenüber der Zeit zwischen zwei Aktionspotentialen T ist, die folgenden Zuordnungen finden:

$$\begin{aligned} w_{ij} &= \frac{i_{ij}}{g_{\text{leak}}}\left(1 - \exp\left(-\frac{t_{\text{fire}}}{\tau}\right)\right) \quad \text{mit} \quad \tau = \frac{C}{g_{\text{leak}}} \\ \varepsilon_{ij}(t) &= \exp\left(-\frac{t}{\tau}\right) \end{aligned} \tag{3.71}$$

Die Frage aus dem letzten Abschnitt wird an dieser Stelle wiederholt und lautet: Wann beginnt das empfangende Neuron zu feuern? Durch diese Betrachtung kann die Eigenantwort η_i vernachlässigt werden und das Membranpotential wird zu:

$$u(t) = \sum_{t_j^{(f)} \in F_j} w_{ij} \exp\left(-\frac{t - t_j^{(f)}}{\tau}\right) \tag{3.72}$$

Mit Annahme einer konstanten Eingangspulsrate $t_j^{(f)} = nT$ mit $n \in \mathbb{N}^+$, kann das Membranpotential nach dem m-ten Eingangspuls bestimmt werden:

$$u(mT) = \sum_{n=0}^{m} w_{ij} \exp\left(-\frac{mT - nT}{\tau}\right) \quad \text{mit} \quad n \in \mathbb{N}^+ \,\text{und}\, t > nT \tag{3.73}$$

Mit der Substitution $k = m - n$ kann die Gleichung umgeschrieben werden. Die Vertauschung der Grenzen der Summe verändert das Ergebnis nicht und führt zur Form

$$\begin{aligned} u(mT) &= \sum_{k=0}^{m} w_{ij} \exp\left(-\frac{kT}{\tau}\right) \\ &= w_{ij} \frac{\exp\left(-\frac{(m+1)T}{\tau}\right) - 1}{\exp\left(-\frac{T}{\tau}\right) - 1}. \end{aligned} \tag{3.74}$$

Die Anwendung der Bedingung

$$u(mT) \geq \theta,$$

dass das Membranpotential die Feuerschwelle erreicht, führt zu einer ähnlichen Übertragungsfunktion des SRM, wie schon durch die rein elektrische Betrachtung am LIAF ermittelt wurde.

3.6 Diskussion

In diesem Kapitel wurden der Energieumsatz und das Verhalten von pulsenden Neuronen auf verschiedene Weise betrachtet. Die Modellierung der biologischen Zellmembran über ein mathematisches Modell der Ionenkanäle und der passiven Ausgleichsvorgänge wurde durch ein regelungstechnisches Modell des aktiven Ausgleichsvorgangs, der Natrium-Kalium-Pumpe erweitert. Die Natrium-Kalium-Pumpe ist dabei der zentrale Ort des Energieumsatzes, bei dem unter Freisetzung von Energie aus dem universellen Energieträger ATP die passiv durch die Zellmembran gedrungenen Ionen ausgeglichen werden. Durch den Ansatz des Ionen-Pumpmechanismus als Zustandsregler kann der Energiebedarf des Neurons sowohl in Ruhe als auch während des Aussendens von Aktionspotentialen beobachtet werden. Abschätzungen für den Energieumsatz sind hierfür bereits von Daut [23] und vor allem Attwell und Lauglin [8] anhand von einfachen Überlegungen der Ladungstrennung an der einer Zellmembran veröffentlicht worden. Zusätzlich ist mit dem in dieser Arbeit entwickelten Ansatz der Energieumsatz eines Neurons bei Erregung des Neurons im Unterschwellenbereich, also ohne Auslösung eines Aktionspotentials, möglich. Für diesen Bereich gibt es in der Literatur praktisch keine Angaben. Die im ersten Teil des Kapitels ermittelten Werte für den Energieumsatz des biologischen Neurons decken sich für die Erzeugung des Aktionspotentials mit den abgeschätzten Werten aus der einschlägigen Literatur. Die Abschätzung des Ruhepotentials ist dagegen zu niedrig.

Im zweiten Teil des Kapitels wurde das Verhalten des LIAF Neurons durch elektrische Schaltkreise beschrieben. Vor allem die Darstellung der Zellmembran als RC-Schaltung wurde für die Ermittlung des Energieumsatzes sowie der Beschreibung der Übertragungsfunktion von LIAF Neuronen genutzt. Durch Ermittlung der analytischen Lösung der Übertragungsfunktion ist es möglich, größere Systeme pulsender Neurone schnell zu simulieren, in dem das Eingangs-Ausgangs-Verhalten nachgebildet wird. Eine Abschätzung der dabei in diesen Systemen benötigten Energie ist durch die Anwendung der Ergebnisse dieses Kapitels möglich. Die Überprüfung der in diesem Kapitel theoretisch ermittelten Ergebnisse soll im folgenden Kapitel durch die Implementierung eines analogen LIAF Neurons vorgenommen werden. Der Vergleich der Implementierung mit den theoretischen Ergebnissen wird daher an geeigneter Stelle im folgenden Kapitel vorgenommen.

Kapitel 4

Ressourcenbedarf pulscodierter neuronaler Netze

Der Ressourcenbedarf pulscodierter neuronaler Netze (PCNN) ist einer der wesentlichen einschränkenden Faktoren bei der Umsetzung der Strukturen des biologischen Vorbilds in hochintegrierte mikroelektronische Schaltungen. Die Wahl, ob das PCNN in analoger oder digitaler Schaltungstechnik umgesetzt wird, hat direkten Einfluss auf den Flächenbedarf sowie die Verarbeitungsgeschwindigkeit und die Verlustleistung der Implementierung.

Rein digital umgesetzte Varianten von pulscodierten neuronalen Netzen können dabei in zwei Arten unterschieden werden:

- Auf FPGA-Strukturen optimierte Modelle, welche die auf heutigen FPGAs bereitgestellten Einheiten wie z. B. Multiplizierer oder DSP Blöcke optimal ausnutzen, aber bei der Synthese auf eine ASIC Technologie eine große Fläche erzeugen.
- Auf ASIC-Strukturen optimierte Modelle, welche typischerweise bitseriell umgesetzt werden, um eine kleine Fläche zu erzielen.

Letztere lassen bei Umsetzung auf FPGA die dort vorhandenen optimierten Blöcke weitgehend ungenutzt.

Rein analoge Schaltungen existieren in einer Chip-Umsetzung praktisch nicht. Häufig werden *Mixed-Signal* Systeme aufgebaut, in denen einzelne Komponenten des Systems, z. B. Neurone und Synapsen, analog implementiert werden und mit digitalen Übertragungssystemen kombiniert werden. Die digitalen Kommunikationssysteme werden dabei oft asynchron betrieben, was zu einer erheblichen Senkung des Energiebedarfs führt. Beispiele für beide Arten der Implementierung von Neuronen sind im Kapitel 2 zum Stand der Technik aufgeführt und diskutiert worden.

Der gemischt analog-digitale (*Mixed-Signal*) Entwurfs-Ansatz wirft in aktuellen CMOS-Technologien jedoch neue Fragen auf. Die Festlegung auf eine bestimmte Halbleiter-Technologie bestimmt die Randbedingungen für alle in ihr umgesetzten Schaltungen.

Während Standardzellen einer digitalen Standardzellenbibliothek in aktuellen CMOS-Technologien in der Regel[1] die minimale Strukturgröße ausnutzen können, sieht der klassische Entwurf von analogen Schaltungen wie z. B. einem LIAF Neuron oder dessen Elementen, einem Verstärker, Komparator etc. den Einsatz groß dimensionierter Schaltungsteile vor, um die Schaltung gegenüber internen und externen Einflüssen robust zu machen. Für die Höchstintegration eines pulscodierten neuronalen Netzes ist der überwiegend analoge Anteil pulscodierter neuronaler Netze damit störend.

Aus dieser Vorüberlegung ergeben sich Fragen, die in diesem Kapitel untersucht und beantwortet werden sollen:

- Welche Halbleiter-Technologie ist geeignet, um Mixed-Signal Schaltungen für pulscodierte neuronale Netze zu entwerfen?
- Können Schaltungen für LIAF Neurone in aktuellen CMOS-Technologien (130 nm und darunter) entworfen und gefertigt werden?
- Wie klein können analoge LIAF Neurone implementiert werden?
- Welche Schwierigkeiten treten beim Entwurf von gemischt analog-digitalen pulscodierten Systemen auf einem Chip (PCSoC) auf?
- Welche Schlussfolgerungen können aus dem Entwurf von PCSoC für die weitere technologische Entwicklung gezogen werden?

Um viele neuronale Elemente auf einem *Mixed-Signal* Chip unterbringen zu können, müssen die Neurone und Synapsen möglichst klein gehalten werden. Im Folgenden soll in diesem Kapitel der Flächenbedarf der verschiedenen Varianten (digital und analog) näher untersucht werden. Dazu werden im ersten Schritt die Details verschiedener Implementierungen vorgestellt und anschließend die Flächeninformationen aus Syntheseschritten (bei digitalen Implementierungen) bzw. der Flächenbedarf bei einem analogen Handentwurf ermittelt.

4.1 Analoge Implementierungen

Die analogen Implementierungen pulscodierter neuronaler Netze und ihrer Bauelemente, den pulsenden Neuronen und verschiedensten Arten von Synapsen, basieren auf einer Reihe von ähnlichen Bibliothekselementen, welche in fast jeder Realisierung wiederzufinden sind. Die für die in dieser Arbeit verwendeten Bauelemente sollen im Folgenden kurz vorgestellt und charakterisiert werden. An Stellen wo mehrere Lösungen möglich sind, werden die gewählten Lösungen mit den Alternativen verglichen und die Wahl begründet.

[1] Standardzellen werden unter Berücksichtigung der gewünschten Treiberleistung flächenoptimal ausgelegt. Spezielle Anwendungen, z. B. die später gezeigte Standardzellenbibliothek für den Subschwellenbereich können hiervon abweichen und auf andere Optimierungsziele hin ausgerichtet sein.

Die CMOS-Technologien, die für die folgenden Untersuchungen und Implementierungen zu Grunde gelegt wurden, sind zum einen eine im klassischen Entwurf analoger Schaltungen häufig genutzte 350 nm Technologie mit doppeltem Polysilizium, zum Anderen eine CMOS-Technologie mit einer minimalen Strukturgröße von 130 nm, welche bis auf spezielle Schritte für geschichtete Metall-Kapazitäten (*Metal-Stack*) keine besonderen Herstellungsoptionen bietet und somit als Prototyp für eine CMOS-Technologie dient. Die im Folgenden vorgestellten analogen Implementierungen zielen auf diese letztere Standard-CMOS-Technologie ab und bedienen sich keiner besonderen Prozessoptionen, um die Möglichkeit einer späteren Abbildung der Layouts auf CMOS-Technologien mit kleineren Strukturgrößen oder die Möglichkeit der Integration der analogen Elemente in einem *Mixed-Signal* SoC mit einem digitalen Standard-Prozess zu wahren. In der 130 nm-Technologie wurde daher auf die speziellen Metall-Kapazitäten verzichtet.

4.1.1 Leaky Integrate and Fire Neuron

Um das vorgestellte LIAF Neuron in einem späteren Abschnitt mit anderen Arbeiten vergleichen zu können, seien hier noch einmal kurz die relevanten verwandten Arbeiten von in analoger Schaltungstechnik implementierten Neuronen angegeben. Eine ausführliche Übersicht über für diese Arbeit relevante Implementierungen und Methoden ist bereits in Kap. 2 gegeben worden. Matolin [73] beschreibt ein Leaky Integrate and Fire (LIAF) Neuron mit einstellbaren Schaltschwellen für einen integrierten Schmitt-Trigger als Schwellenelement in einem System von 64×64 Neuronen, welche miteinander in einer nächster Nachbar-Beziehung verbunden sind. Indiveri [55] beschreibt ein LIAF Neuron mit Pulsratenadaption, einstellbarer Refraktärzeit und veränderlicher Feuerschwelle. Die Implementierung des Schwellenelements wurde durch einen Inverter mit positiver Rückkopplung über einen Source-Folger implementiert, welcher im Subschwellenbereich arbeitet. Durch die positive Rückkopplung wird eine besonders niedrige Verlustleistung des Neurons vor allem beim Schaltvorgang erreicht. Die Veröffentlichung von Indiveri [55] aufgreifend, ersetzt Liu [68] das Schwellenelement des Neurons durch einen Differenzverstärker.

In Abb. 4.1 ist der Schaltplan des in dieser Arbeit in einer 130 nm CMOS-Technologie implementierten LIAF Neurons dargestellt. Das Neuron besteht aus einem Komparator (Transistoren M4–M11) mit positiver Rückkopplung auf den Eingang, welche durch einen Koppelkondensator C_{back} realisiert wurde. Diese Rückkopplung wurde der bekannten Axon-Hillock-Schaltung [76] entlehnt. Der Komparator vergleicht den Wert des Membranpotentials mit der am Referenzeingang angelegten Feuerschwelle U_{TH} und erzeugt bei Überschreiten der Spannung am Referenzeingang ein Aktionspotential, d. h. der Ausgang des Komparators wird auf eine Spannung von U_{DD} gelegt. Das Membranpotential wird bis zu diesem Zeitpunkt durch die Integration einlaufender Ströme am Eingang V_{in} auf der Membrankapazität C_{mem} erzeugt und durch den Leckstrom über Transistor M16, welcher den Leckleitwert g_{leak} des technischen LIAF Neurons modelliert, begrenzt. Transistor M16 sorgt für die passive Abnahme des Membranpotentials mit der Zeit, wenn keine oder nur wenige Strompulse am Eingang einlaufen, und modelliert das Unterschwellenverhalten

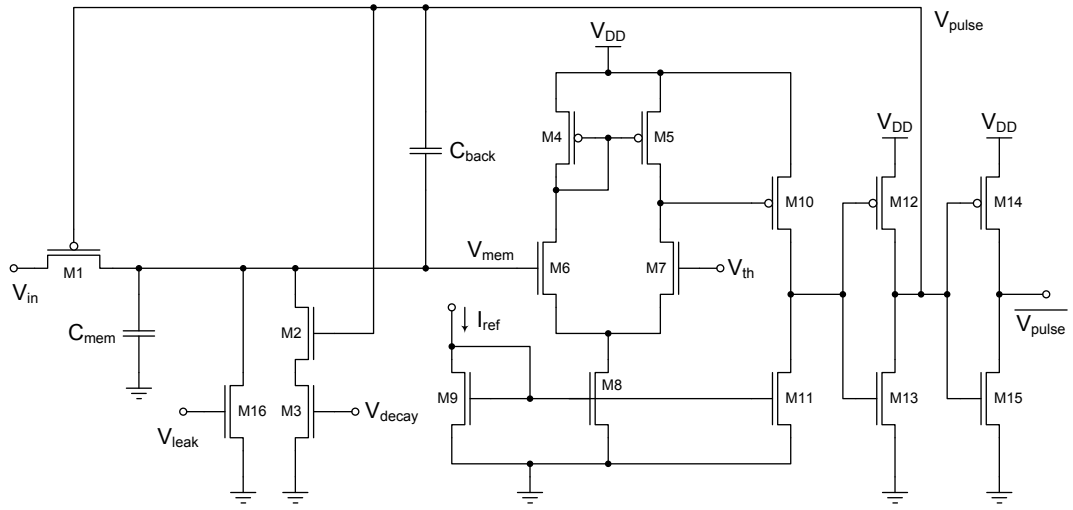


Abbildung 4.1: Schaltplan des LIAF Neurons in 130 nm Technologie.

des biologischen Neurons (vgl. Kap. 1.3). Wenn das Membranpotential die Feuerschwelle, dargestellt durch die Schwellenspannung U_{TH}, von unten kommend übersteigt, wird durch den Komparator ein Spannungspuls erzeugt, welcher dem Aktionspotential des biologischen Neurons entspricht. Der Komparator erzeugt eine Spannung von U_{DD} am Knoten V_{pulse} und sorgt durch die positive Rückkopplung über C_{back} für ein sicheres Schalten, indem die Spannung am Eingang des Komparators (Gate des Transistors M6) durch Ladungsverschiebung zwischen beiden Kapazitäten überhöht wird. Es stellt sich eine Membranspannung von

$$U_{\mathrm{mem}} = U_{\mathrm{TH}} + \frac{C_{\mathrm{back}}}{(C_{\mathrm{mem}} + C_{\mathrm{back}})} \cdot U_{\mathrm{DD}}$$

ein. Die invertierende Treiberstufe aus M14 und M15 sorgt für eine Entkopplung der kapazitiven Schaltung zur Erzeugung des Aktionspotentials von weiteren Stufen des Systems und erzeugt ein definiertes (wenn auch invertiertes) digitales Puls-Signal am Ausgang des Neurons.

Die Pulsbreite des Ausgangspulses wird hauptsächlich durch den Leitwert des Transistors M3 bestimmt, welcher durch externe Referenzspannung V_{decay} am Gate des Transistors eingestellt werden kann. In erster Näherung ist der Leitwert des Transistors M2 im Sättigungsbereich konstant und kann so einfach in die Betrachtung des aktiven Entladevorgangs über die Transistoren M2 und M3 einbezogen werden. Während der Erzeugung des Aktionspotentials wird der Eingang des Neurons durch den zusätzlichen im Eingangspfad liegenden PMOS-Transistor hochohmig geschaltet, um eine weitere Aufladung des Membranpotentials während der aktiven Entladung der Kapazität C_{mem} zu verhindern. Diese Maßnahme soll eine gleich bleibende Zeit für jedes Aktionspotential gewährleisten.

Wenn das Membranpotential während der Erzeugung des Aktionspotentials die Schwellenspannung U_{TH} durch aktive Entladung von oben kommend wieder unterschreitet, schaltet der Komparator erneut und legt an den Ausgang V_{pulse} das Massepotential an. Durch

die positive Rückkopplung über die Kapazität C_{back} auf die Membrankapazität wird das Membranpotential schlagartig auf den Wert

$$U_{\text{mem}} = U_{\text{TH}} - \frac{C_{\text{back}}}{(C_{\text{mem}} + C_{\text{back}})} \cdot U_{\text{DD}}$$

unterhalb der Schwellenspannung vermindert und sorgt so wieder für sicheres Schalten.

Beim Layout des Neurons in der 130 nm Technologie wurde besonders auf eine geringe benötigte Fläche Wert gelegt. Das Neuron sollte möglichst klein und kompakt aufgebaut werden, um eine sehr hohe Packungsdichte auf einem Chip erzielen zu können. Um das Layout möglichst kompakt zu halten, und um auf eine Standard-CMOS-Technologie ohne zusätzliche Prozess-Schritte abbilden zu können, werden in der hier gezeigten Variante die Kapazitäten nicht durch besondere Elemente der jeweiligen Technologie, wie doppelte Polysilizium-Lagen oder geschichtete Metall-Lagen realisiert, sondern durch das Gate-Oxid von MOS-Transistoren. Als Besonderheit dieses Entwurfs ist die Implementierung der Koppelkapazität als MOS-Kapazität eines PMOS-Transistors in einer n-Wanne mit gesteuertem Substratanschluss zu nennen, wodurch die benötigte Fläche im Vergleich zu einer Implementierung mit einer Metall-Metall Kapazität stark reduziert werden konnte. Durch Beschalten des Substratanschlusses mit dem Aktionspotential des Neurons bildet sich durch den Substratsteuereffekt und das niedrigere Potential am Gate des PMOS-Transistors eine Inversionsschicht aus, so dass die Kapazität über die gesamte Gatefläche ausgebildet ist. Die Nichtlinearität von MOS-Kapazitäten im Übergang zum Subschwellbereich der Transistoren kann in dieser Implementierung vernachlässigt werden, da die Elemente nach einer Initialisierungsphase alle in einem Arbeitsbereich oberhalb der Schwellenspannung arbeiten. In Abb. 4.2 ist das entstandene Layout zu sehen, in dem die gestreckten Kapazitäten zu einer insgesamt rechteckigen Struktur auf einer Fläche von $76,37\,\mu$m führen. Diese Anordnung erlaubt es, sehr leicht zusätzliche Synapsen um das Neuron anzuordnen, um größere Systeme und Netze aufzubauen.

Ein einschränkender Faktor für die Implementierung und weitere Verkleinerung dieses Neurons in zukünftigen Technologien ist die Zunahme der Leckströme der Transistoren und Tunnelströme durch dünne Gate-Oxide [89, Kapitel Process Integration, Devices, and Structures], welche sich bereits in der 130 nm Technologie zu insgesamt 100 pA aufsummieren. Diese Leckströme führen mit abnehmenden Strukturgrößen zu einer immer schnelleren Entladung der gefertigten Kapazitäten. In dieser Technologie erhält man für das Neuron bereits Zeitkonstanten von $\tau \approx 0.1\,$s für die Entladung der Kapazität, so dass Pulsraten im biologisch plausiblen Zeitbereich von Millisekunden kaum mehr verarbeitet werden können, da die gesamte eingelaufene Information schnell wieder „vergessen“, d. h. der Einfluss einzelner Pulse auf das Membranpotential abgebaut ist. Abhilfe schafft hier derzeit nur das Ausweichen auf einen Zeitskalenbereich, der mehrere Größenordnungen schneller ist, also die Nutzung von Pulsraten im Bereich von kHz bis MHz mit Anpassung der Pulsbreite des Aktionspotentials auf den Mikrosekundenbereich. Dieses Vorgehen hat Auswirkungen auf alle weiteren Elemente des neuronalen Netzes, deren Dynamik an die neue Zeitskala angepasst werden muss. Von einer Beschleunigung der Simulation

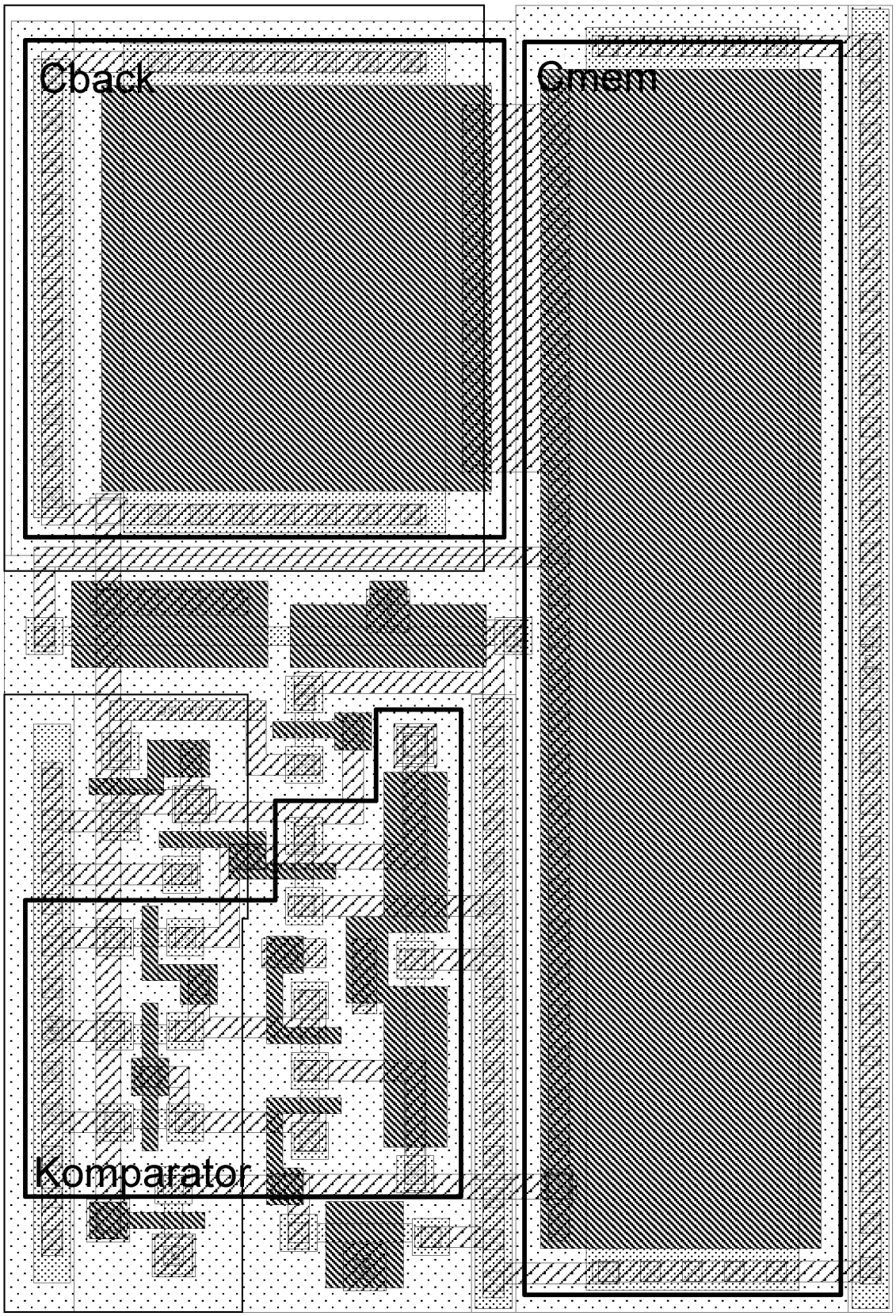


Abbildung 4.2: Layout des LIAF Neurons in 130 nm Technologie.

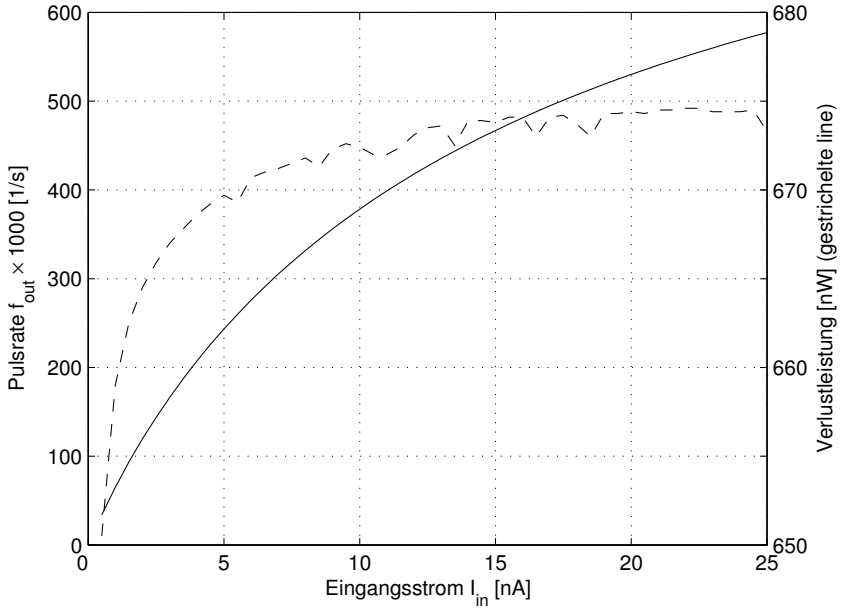


Abbildung 4.3: Ausgangspulsrate f_{out} (durchgezogene Linie) und Verlustleistung (gestrichelte Linie) aufgetragen über dem Eingangsstrom.

neuronaler Netze als Eigenschaft der vorgestellten Neuronenimplementierung kann in diesem Zusammenhang allerdings noch nicht gesprochen werden, der Wechsel der Zeitskala ist eher als Versuch anzusehen, die negativen Eigenschaften der Technologie für diese Schaltung zu umgehen.

Simulationen am RC-extrahierten Layout des beschriebenen LIAF Neurons zeigen eine zu erwartende Verlustleistung von 650 nW bis 675 nW über einen Pulsratenbereich des Ausgangs von 0 Pulsen/s bis $500 \cdot 10^3$ Pulsen/s (siehe Abb. 4.3). Dabei nimmt die ermittelte statische Verlustleistung des Komparators mit 650 nW den größten Teil der Verlustleistung ein. Für diese Simulationen wurde dem Neuron ein Strom von 0 nA bis 50 nA über einen einfachen p-Kanal Stromspiegel injiziert und die sich ergebende Ausgangspulsrate über den dem Stromspiegel eingeprägten Strom aufgetragen. Dieser Stromspiegel ist auch Hauptbestandteil der Implementierung statischer Synapsen. Die statische Verlustleistung des Neurons kann direkt nur durch die Wahl eines anderen Arbeitspunktes verringert werden. Gleichzeitig ist zu erwägen, ob alternative Vergleichselemente, z. B. Schmitt-Trigger bessere Eigenschaften bezüglich statischer Verlustleistung aufweisen. Da der einfache Stromspiegel den Referenzstrom in der Praxis nicht ideal gespiegelt in das Neuron injizieren kann, ist in Abb. 4.4 die Ausgangspulsrate des Neurons über dem korrigierten tatsächlichen Eingangsstrom für verschiedene Betriebsfälle (*typical*, *best analog* (FFA), *worst analog* (SSA)) aufgetragen. Es ergibt sich ein linearer Zusammenhang zwischen dem Eingangsstrom und der Ausgangspulsrate, wie bei kleinen passiven Leckleitwerten, die in dieser Simulation eingestellt waren, aus den theoretischen Überlegungen in Kap. 3.4.2 erwartet wird.

Abbildung 4.5a zeigt noch einmal das in Kapitel 3.3 ermittelte theoretische Maximum

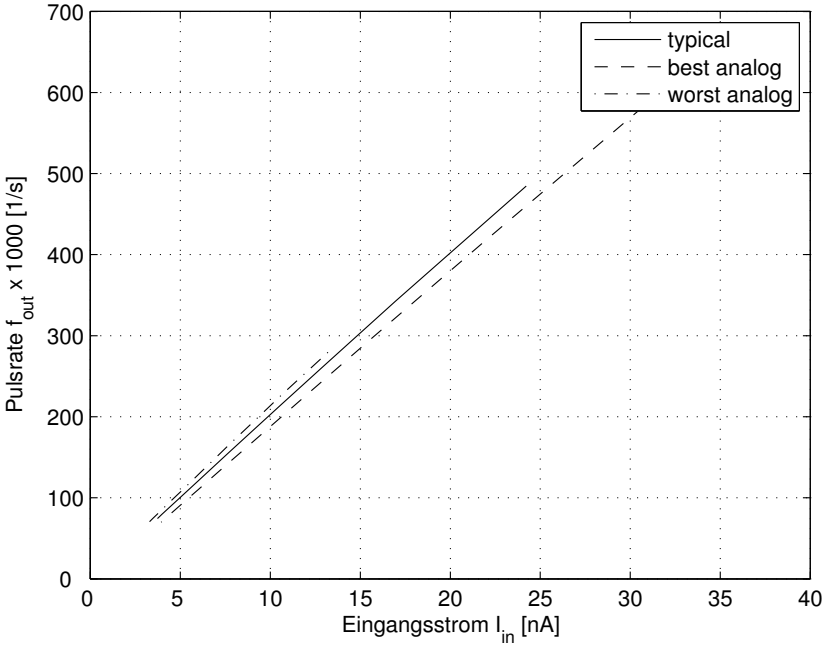
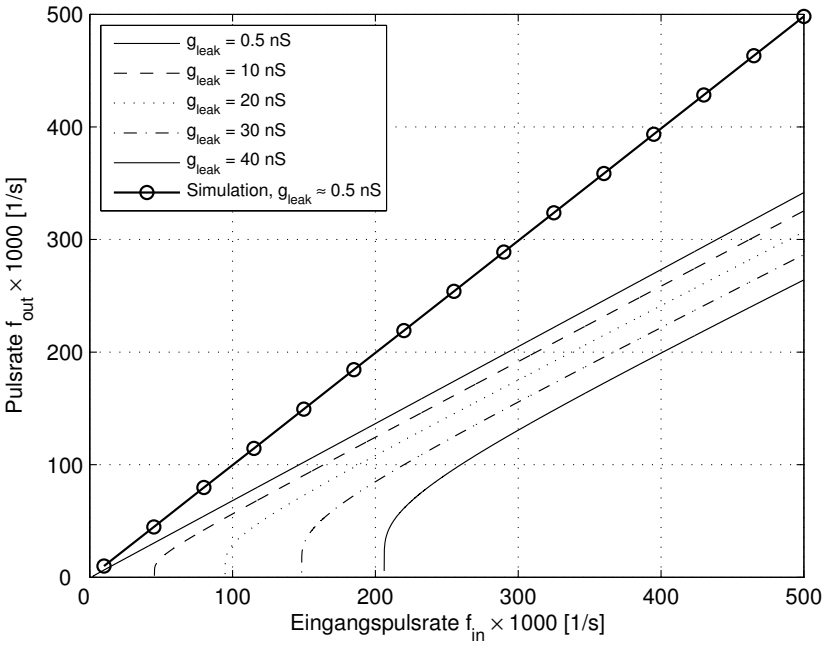


Abbildung 4.4: Ausgangspulsrate des Neurons in Abhängigkeit des Eingangsstroms für verschiedene Betriebsfälle.

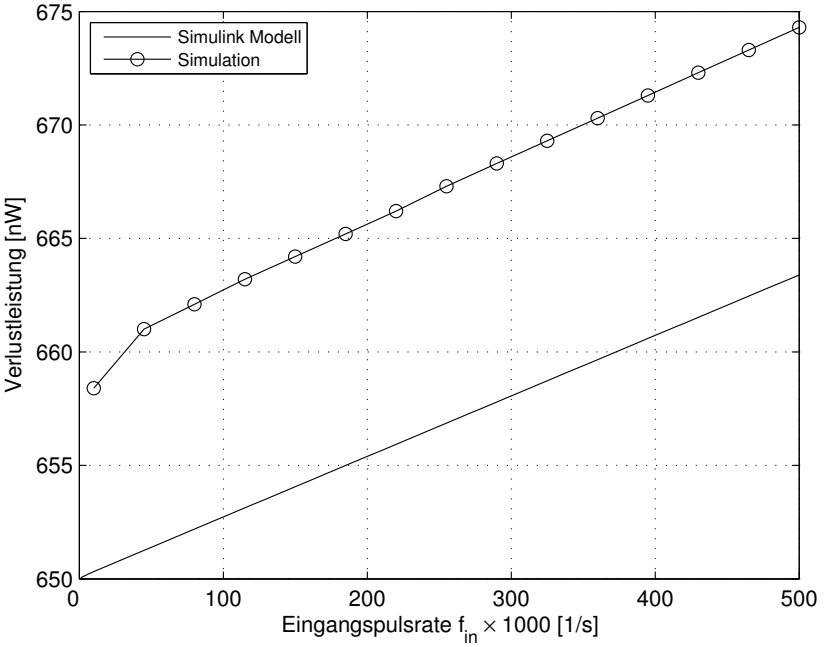
der Ausgangspulsrate f_{out} in Abhängigkeit von der Eingangspulsrate bei verschieden großen Leckleitwerten. Gleichzeitig ist das Ergebnis der Simulation am RC-extrahierten Layout des implementierten Neurons darüber gelegt. Im Eingang wurde sowohl in der theoretischen Betrachtung als auch in der Simulation eine konstante Pulsrate mit gleichbleibender Amplitude angelegt. Als Ergebnis lässt sich aus der Darstellung schließen, dass das Übertragungsverhalten (und damit die Übertragungsfunktion) des Neurons für geringe Leitwerte nahezu linear wird. Dieses lässt sich durch die Simulation der Schaltung verifizieren. Für größere Leitwerte wird die Übertragungsfunktion des Neurons hochgradig nichtlinear und nähert sich der bekannten „gain-function“ [71] mit dem charakteristischen Subschwellenverhalten eines biologischen Neurons an, wie in Kap. 3.4.2 gezeigt wurde.

Die Ausgangspulsrate des simulierten Neurons liegt oberhalb des abgeschätzten Wertes, welches sich auf eine unvollständige Entladung der Membrankapazität in der Simulation zurückführen lässt. Dadurch erreicht das Neuron in der Simulation die Feuerschwelle schneller, als das in den theoretischen Betrachtungen zugrunde gelegte vollständig entladene Neuron. Damit wird die Übertragungskennlinie steiler.

Um die dynamische Leistungsaufnahme des Neurons abzuschätzen, wurde ein Simulink-Modell der Schaltung erstellt, in welchem die gleichen Leitwerte wie in den implementierten Neuronen eingestellt wurden. In Abb. 4.5b sind die mit dem Modell ermittelte und die durch Simulationen am RC-extrahierten Layout ermittelte Verlustleistung übereinander gelegt. Das Ergebnis des Simulink-Modells stimmt mit dem Ergebnis aus der Simulation am RC-extrahierten Layout gut überein und kann somit als Modell für die Untersuchung an größeren Systemen dienen, welche bei Simulation der RC-extrahierten Layouts sehr zeitaufwändig werden.



(a) Ausgangspulsrate in Abhängigkeit der Eingangspulsrate für verschiedene passive Leckleitwerte.



(b) Verlustleistung in Abhängigkeit der Eingangspulsrate (Abschätzung der Verlustleistung enthält 650 nW statische Verlustleistung).

Abbildung 4.5: Abschätzung und Simulationsergebnis für die Ausgangspulsrate und die Verlustleistung des LIAF Neurons.

Tabelle 4.1: Flächenbedarf und Verlustleistung für Neuron und Synapsen in einer 130 nm CMOS-Technologie.

Neuron [μm^2]	Block [μm^2]	Pulsrate [1/s]	Leistung[d][nW]	Referenz
76,37	772,70[a]	$0 - 600 \cdot 10^3$	675	
	854,84[b]			Matolin et. al. [73][c]
			1473	Indiveri et. al. [55][c]
757,12		10^6	1190	Wijekoon et. al. [102][c]

[a] Ein Block besteht aus einem Neuron und 5 Synapsen.
[b] Die angegebene Fläche wird von einem Neuron und 4 Synapsen belegt.
[c] Strukturgrößen skaliert auf 130 nm Technologie mit 1, 2 V Versorgungsspannung.
[d] Leistung bei maximaler Pulsrate.

Das vorgestellte Neuron soll im Folgenden mit den zuvor erwähnten Modellen aus der Literatur verglichen werden. Dabei ergibt sich als besondere Schwierigkeit, dass die vorgestellten Implementierungen einerseits in anderen Technologien gefertigt wurden, andererseits auch für leicht unterschiedliche Funktionen optimiert wurden. Der folgende Vergleich ist daher möglicherweise nicht ganz fair – insbesondere was die Flächenangaben angeht, da ein Wechsel der Technologie immer Einfluss auf die Struktur der analogen Schaltung hat – zeigt aber das Gesamtergebnis tendenziell auf. Eine vergleichbare Angabe zum Flächenbedarf eines pulsenden Neurons ist nur in der Veröffentlichung von Wijekoon [102] zu finden. Diese ist nach Skalierung auf die CMOS-Technologie des in dieser Arbeit entwickelten analogen Neurons fast zehn mal größer. Das in [102] vorgestellte Neuron erlaubt dagegen die Emulation eine größeren Klasse von Neuronen. Die Verlustleistung des hier vorgestellten Neurons und des Neurons von Wijekoon sind dagegen praktisch gleich. In Tabelle 4.1 sind die Angaben für den Flächenbedarf und die Leistungsaufnahme des hier gezeigten Neurons angegeben. Daneben enthält die Tabelle auch auf eine 130 nm CMOS-Technologie und eine Versorgungsspannung von 1,2 V skalierte Versionen der vom Funktionsprinzip vergleichbaren Strukturen. Zur Skalierung der Strukturen auf eine gemeinsame Basis wurden die allgemeinen Skalierungsregeln aus Anhang B für CMOS-Schaltungen verwendet.

4.1.2 Statische Synapse

Die Synapse stellt das Bindeglied zwischen einzelnen Neuronen dar. Diese wandeln das digitale Signal des Aktionspotentials eines sendenden Neurons in einen postsynaptischen Strom für das empfangende Neuron um. Durch die Stärke des Stroms wird der Wert eines von einem präsynaptischen Neuron ausgesandten Aktionspotentials gewichtet. In der einfachsten Form der Informationsverarbeitung bedeutet ein hoher Strom eine große Relevanz des präsynaptischen Neurons für das postsynaptische Neuron. Der Begriff der statischen Synapse trifft die Eigenschaft der in dieser Arbeit entworfenen Schaltung nicht vollständig, da die Synapse einerseits während des Betriebs eine Konstantstromquelle darstellt, andererseits zu jeder Zeit von außen in ihrem Wert in diskreten Schritten

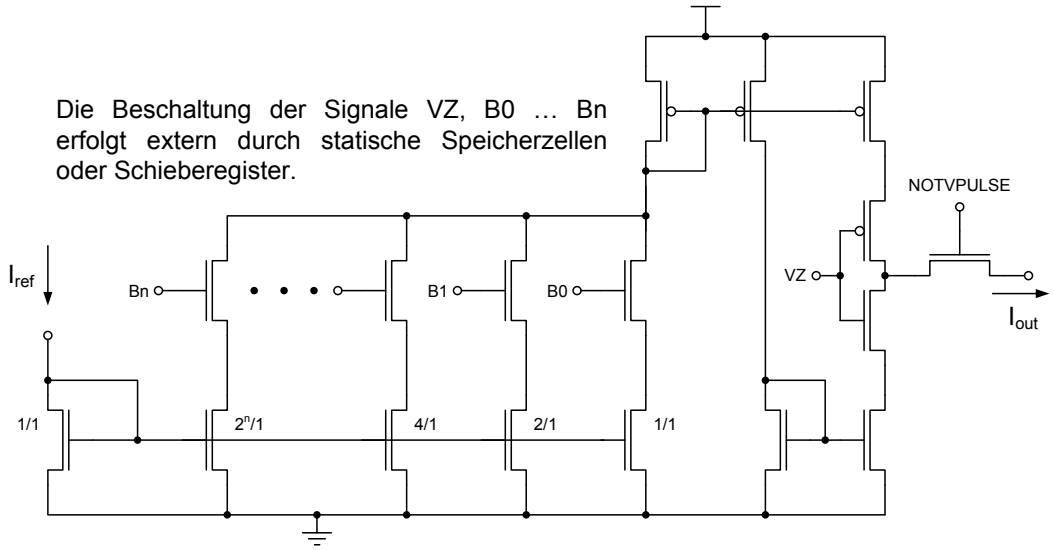


Abbildung 4.6: Schaltplan einer wahlweise exzitatorischen oder inhibitorischen n Bit Synapse in Stromschaltungstechnik.

verändert werden kann. Zur Abgrenzung vom Begriff der dynamischen Synapse seien die folgenden Definitionen gebraucht:

Definition 3 *Eine statische Synapse ändert den ihr zugewiesenen Wert nur durch externe Festlegung des Wertes (Benutzereingriff) und nicht durch eine Eigendynamik, also durch interne Zustände oder durch das Empfangen von Aktionspotentialen.*

Definition 4 *Eine dynamische Synapse ändert den ihr zugewiesenen Wert hauptsächlich durch eine gegebene Eigendynamik, welche sowohl den internen Zustand der Synapse als auch die empfangenen Aktionspotentiale berücksichtigen kann. Daneben kann durch externe Festlegung des Wertes (Benutzereingriff) ein initiales Gewicht voreingestellt werden.*

Die Untersuchung und Implementierung biologienaher dynamischer Synapsen ist eng verknüpft mit der Erforschung von Lernvorgängen um das Thema der *Spike-Time Dependend Plasticity* (STDP) und wird in dieser Arbeit nicht weiter behandelt. Eine Implementierungsvariante dynamischer Synapsen ist in [110] vorgestellt. Da sich diese Arbeit auf die Modellierung und Implementierung von LIAF Neuronen in PCNN beschränkt, ist der Einsatz von statischen Synapsen als Eingangselement für die Neurone an dieser Stelle ausreichend.

Schaltungstechnisch lässt sich eine statische Synapse effizient als Anordnung geschalteter Stromquellen realisieren. In Abb. 4.6 ist eine Realisierung der statischen Synapse mit Stromspiegeln dargestellt, bei denen die Weite eines Spiegeltransistors in jeder Stufe verdoppelt wird. Die Stromspiegel im unteren Teil der Synapse spiegeln den eingeprägten Referenzstrom I_{ref} abhängig von ihrer Dimensionierung zu einem Spiegelstrom der Stärke $2^n \cdot I_{\text{ref}}$. Dabei beschreibt n die Bitstelle der jeweils ausgewählten Stufe. Durch die Schalttransistoren B0 bis Bn können die Ströme zu einem binär codierten Gesamtstrom

zusammengefasst werden. Der so gebildete Strom wird wiederum gespiegelt und als Referenzstrom für je eine Stromquelle und eine Stromsenke genutzt. Durch Wahl des Signals für das Vorzeichen (VZ) kann ein exzitatorischer (positiver) bzw. inhibitorischer (negativer) Strom I_{out} erzeugt werden. Der Strom I_{out} wird mit Eintreffen eines Aktionspotentials freigegeben. In der abgebildeten Synapse wird dazu aufgrund der Implementierung des Neurons das negierte Aktionspotential *NOTVPULSE* ausgewertet.

Abb. 4.7 zeigt das Layout einer statischen exzitatorischen Synapse mit 5 Bit Auflösung in einer 130 nm CMOS-Technologie auf einer Fläche von 143 μm^2. Die Synapse besteht aus 5 SRAM-Speicherzellen, die den Wert der Synapse lokal für die Anordnung der Stromquellen bereitstellen und deren Wert durch externe Beschaltung verändert werden kann. Die SRAM-Zellen wurden in der schematischen Darstellung der Synapse ausgelassen und stellen dort den Wert B0 bis Bn zur Verfügung. Daneben sind im Layout die unterschiedlich dimensionierten Transistoren für die Skalierung des Stroms sowie die Schalttransistoren mit minimalen Abmessungen zu sehen. Die Abmessungen der Transistoren der Stromquellen wurden so angepasst, dass jede Stufe der Stromquellen den doppelten Strom der vorhergehenden Stufe bereitstellt. Die idealen Skalierungsregeln größerer CMOS-Technologien greifen an dieser Stelle durch die in der 130 nm CMOS-Technologie auftretenden *short-channel* und *narrow-channel* Effekte nicht mehr. Das hier gezeigte Layout der statischen Synapse ist für die Anordnung von 4 Synapsen um ein zentrales Neuron optimiert, eine Struktur, die bei der lokalen Verschaltung von Neuronen mit ihren nächsten Nachbarn häufig gewählt wird. In diesem Layout ist der Stromausgang I_{out} separat gekennzeichnet, der den exzitatorischen Strom bei angelegtem Massepotential am Gate des Schalttransistors den folgenden Neuronen zur Verfügung stellt. Für eine Beschaltung mit 5 Synapsen, bei der 4 Synapsen für den Aufbau einer Nächste-Nachbar Beziehung und eine Synapse für die Gewichtung externer Pulse vorgesehen ist, wurde ein optimiertes Layout entworfen, bei dem die SRAM-Zellen des Synapsenblocks an weitere Synapsenblöcke anreihbar sind.

Die für die Synapse verwendete statische Speicherzelle ist eine modifizierte 6-Transistor SRAM-Zelle. In Abb. 4.8a ist der schematische Aufbau der SRAM-Zelle mit 5 Transistoren dargestellt. Bei dieser Implementierung wurde im Gegensatz zur 6T-SRAM Zelle ein Auswahltransistor entfernt und der Knoten $U2$ stattdessen direkt an die Gate-Kapazität eines Auswahltransistors des Stromspiegels der in Abb. 4.6 gezeigten Synapse angeschlossen. Diese Maßnahme führt zu einer Flächeneinsparung in der Zelle. Gleichzeitig entfällt die Notwendigkeit der Bereitstellung eines invertierten Signals an Knoten $U2$ bei der Programmierung der Speicherzelle. Zur Programmierung der Zelle muss das invertierte Signal an Eingang BL bereitgestellt werden, der Knoten $U2$ bleibt vom Dateneingang entkoppelt, so dass die Kapazität keinen direkten Einfluss auf das Umladen des ersten Inverters der SRAM-Zelle hat. Die Längen und Weiten der Transistoren wurden überwiegend auf minimalen Maßen der 130 nm CMOS-Technologie belassen. Um die Robustheit der SRAM-Zelle gegenüber Störungen auf dem Knoten BL zu maximieren, wurde die Weite des Auswahltransistors $M0$ durch Simulation optimiert. Dabei wurde der statische Störabstand (engl. static Noise-Margin, SNM) aus der Übertragungskennlinie der SRAM-

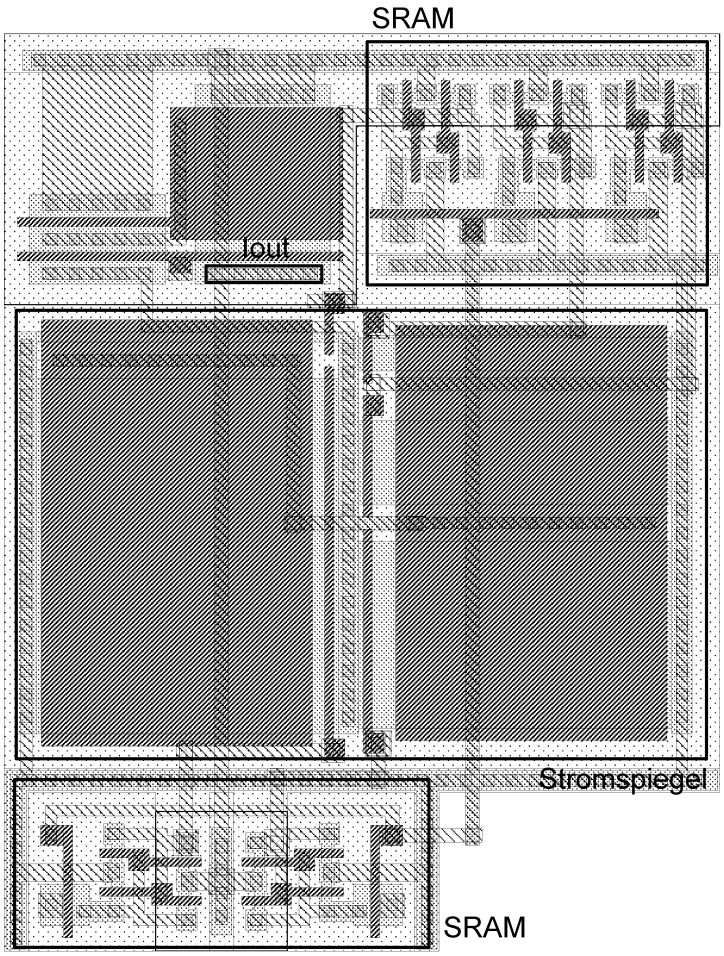
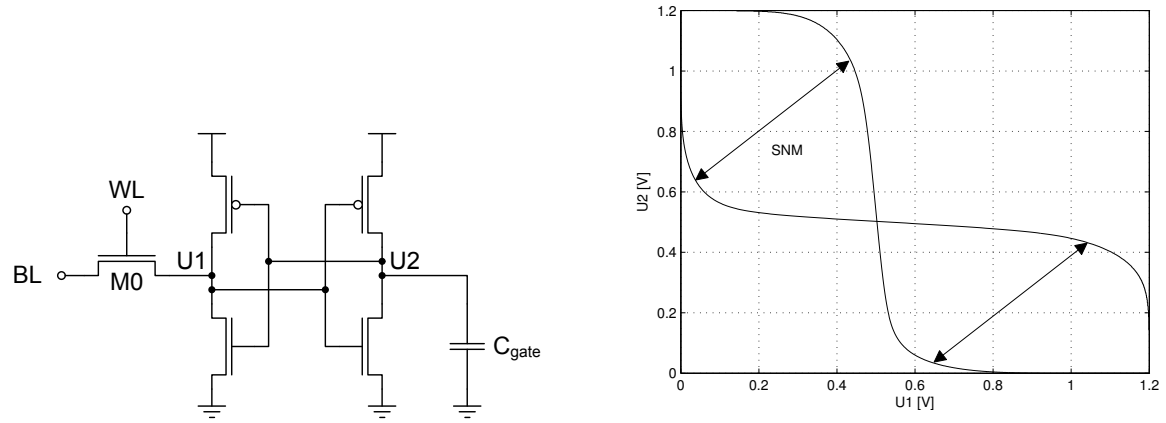


Abbildung 4.7: Layout einer exzitatorischen 5 Bit Synapse in einer 130 nm CMOS-Technologie.



(a) Schematische Darstellung der modifizierten SRAM-Zelle.

(b) Übertragungsfunktion der modifizierten SRAM-Zelle mit SNM.

Abbildung 4.8: Modifizierte SRAM-Zelle für lokale Gewichtsspeicherung.

Zelle durch Wahl der Transistorweite maximiert (siehe Abb. 4.8b zur Definition des SNM). Für ein maximales SNM wurde hier numerisch eine Weite von $0,6\,\mu$m ermittelt, womit auch die Forderung

$$U1\left(U2 = \frac{U_{\mathrm{DD}}}{2}\right) = \frac{U_{\mathrm{DD}}}{2}$$

für einen maximalen Störabstand annähernd erfüllt wird.

4.1.3 Ermittlung der äquivalenten Wortbreite

Um in der weiteren Arbeit die Flächenverhältnisse sowie die umgesetzte Leistung der digitalen Implementierungen mit den Werten der analogen Implementierungen vergleichen zu können, wird in diesem Abschnitt die benötigte Wortbreite für den Vergleich der digitalen Implementierungen ermittelt. Diese ergibt sich aus dem Auflösungsvermögen analoger Schaltungen, insbesondere der Fähigkeit, zwei Signale gerade eben noch voneinander unterscheiden zu können. Als unterste Grenze zur Unterscheidung von zwei Spannungs-Signalen gilt das Spannungs-Rauschen, auf dessen Auswirkungen im Folgenden eingegangen wird.

Zur Ermittlung der äquivalenten Wortbreite wird vom wichtigsten rauschbehafteten Element eines Neurons ausgegangen, dem Entscheidungselement. Dieses bestimmt im pulsenden Neuron, ob ein Aktionspotential generiert werden soll. Der zu diesem Zweck häufig eingesetzte Komparator besteht im Wesentlichen aus einem Differenzverstärker mit hoher Verstärkung; Dieser soll im Folgenden untersucht werden.

Der in Abb. 4.9 dargestellte Differenzverstärker besteht aus einer Stromquelle, die im Arbeitspunkt den Strom $I_{\mathrm{SS}}/2$ durch den linken und den rechten Zweig der Schaltung erzwingt. Dazu sind beide Zweige der Schaltung symmetrisch ausgelegt. Die Transistoren M3 und M4 werden durch die Bias-Spannung U_{B} im Sättigungsbereich betrieben und

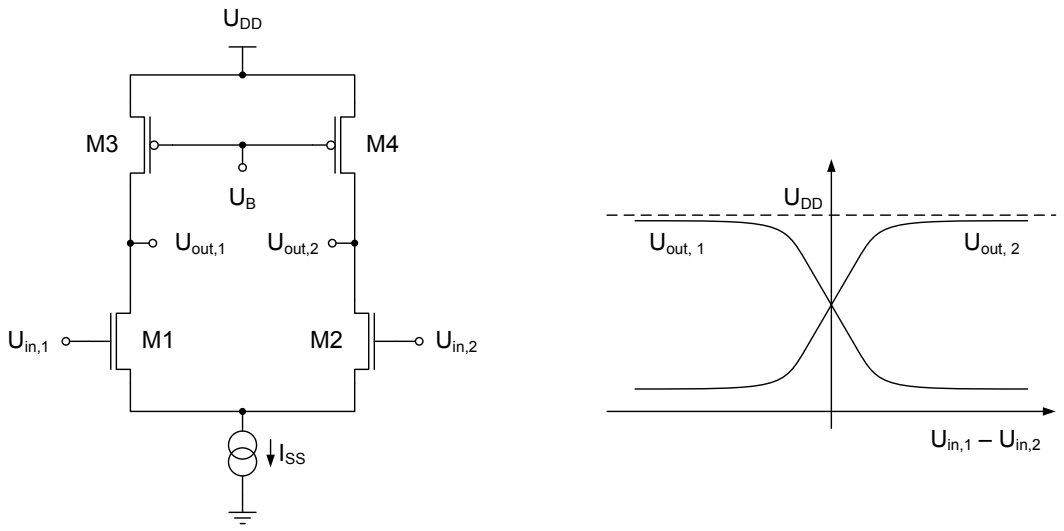


Abbildung 4.9: Differenzverstärker-Paar aus MOS-Transistoren und korrespondierende Übertragungskennlinie.

bilden so eine aktive Last für den Verstärker. Das Differenzpaar aus Transistoren $M1$ und $M2$ bildet die Differenz aus den angelegten Spannungen $U_{\text{in},1}$ und $U_{\text{in},2}$ und verstärkt diese zur differentiellen Ausgangsspannung aus $U_{\text{out},1}$ und $U_{\text{out},2}$. Der Verstärkungsfaktor A_v des gezeigten Differenzverstärkers ergibt sich aus der Betrachtung des Kleinsignalersatzschaltbildes im gewählten Arbeitspunkt (AP) der oben gezeigetn Schaltung mit der Nomenklatur nach [80] zu

$$A_v = \left.\frac{\Delta U_{\text{out}}}{\Delta U_{\text{in}}}\right|_{\text{AP}} = \left.\frac{U_{\text{out},2} - U_{\text{out},1}}{U_{\text{in},2} - U_{\text{in},1}}\right|_{\text{AP}} \approx -g_{m1}(r_{o1} \| r_{o3}), \tag{4.1}$$

mit der Gatesteilheit g_{m1} und dem Ausgangswiderstand r_{o1} des Transistors M1 und dem Ausgangswiderstand r_{o3} des Transistors M3.

Alle mikroelektronischen Schaltungen sind dem Einfluss von Rauschen unterworfen. Elektrisches Rauschen wird als „ungewollte“ Spannung oder äquivalent als „ungewollter“ Strom betrachtet, der zusätzlich zum Nutzsignal in Schaltungen auftritt [9]. Das Rauschen stellt die untere Grenze eines Signals dar, das von einer Schaltung sinnvoll verarbeitet werden kann und limitiert so das Auflösungsvermögen elektrischer Schaltungen [80]. Für eine Diskussion der Eigenschaften der verschiedenen Arten des Rauschens wird an dieser Stelle verzichtet und auf die bereits zitierten Quellen [9, 80] verwiesen, welche das Thema ausführlich behandeln.

Durch die verwendeten MOS-Transistoren treten in der Schaltung sowohl thermische Rauschquellen als auch 1/f-Rauschen (engl. Flicker-Noise) auf. Beide Rausch-Arten werden in der theoretischen Betrachtung verwendet und können als zusätzliche Spannungs- oder Stromquellen zwischen Gate und Source oder Drain und Source jedes einzelnen Transistors M1 bis M4 modelliert werden.

Um das Auflösungsvermögen des Differenzverstärkers zu bestimmen, muss das Signal-Rausch-Verhältnis (SNR) am Eingang der Schaltung bestimmt werden. Dazu wird im

ersten Schritt das Rauschleistungsdichtespektrum im Ausgang des Verstärkers bestimmt und anschließend daraus die Rauschspannung am Eingang ermittelt. Es wird angenommen, dass der Verstärker in seinen Elementen des linken und rechten Zweiges symmetrisch ist. Damit kann die Rauschspannung $\overline{V_n^2}$ am Eingang der Schaltung in Abhängigkeit von den Rauschquellen der Transistoren M1 und M3 ($\overline{V_{n1}^2}$ und $\overline{V_{n3}^2}$) bestimmt werden:

$$\overline{V_n^2} = 2\overline{V_{n1}^2} + \frac{2g_{m3}^2(r_{o1} \| r_{o3})^2}{A_v^2} \cdot \overline{V_{n3}^2} = \overline{V_{n1}^2} + \frac{2g_{m3}^2}{g_{m1}^2} \cdot \overline{V_{n3}^2} \tag{4.2}$$

Nach Einsetzen der Terme für die Rauschquellen und Integration über den betrachteten Frequenzbereich ergibt sich das Ergebnis für die Gesamtrauschspannung von

$$\overline{V_{n,\text{total}}^2} = \int_1^\infty 8kT \left(\frac{2}{3} g_{m1} + \frac{2}{3} \frac{g_{m3}}{g_{m1}} \right) + \frac{2K_N}{C_{\text{ox}}(WL)_1 f} + \frac{2K_P}{C_{\text{ox}}(WL)_3 f} \frac{g_{m3}^2}{g_{m1}^2} df. \tag{4.3}$$

Die hier eingeführten Parameter K_N und K_P beschreiben den Einfluss des 1/f-Rauschens und sind technologieabhängige Konstanten [80].

Die Rauschspannung am Eingang kann über den Zusammenhang

$$\text{SNR}\,|_{\text{dB}} = 20 \log \frac{V_{\text{signal}}}{V_{n,\text{total}}} \tag{4.4}$$

in ein äquivalentes Signal-Rausch-Verhältnis umgerechnet werden, aus dem sich über die Umformung

$$\text{SNR}\,|_{\text{dB}} \approx 6.02n + 1.76 \tag{4.5}$$

aus dem Signal-Rausch-Verhältnis die Auflösung in Bit (n) ermitteln lässt.

In der Praxis wird der Frequenzbereich bei Rechnungen nach oben begrenzt, da auch das Rauschen in Verstärkern bandbegrenzt ist. Im Fall des im LIAF Neuron implementierten Verstärkers wird die obere Integrationsgrenze mit $f_H = \frac{\pi}{2} f_{3\,\text{dB}} = 53\,\text{MHz}$ festgelegt. Die Grenze von $\frac{\pi}{2} f_{3\,\text{dB}}$ schließt das Rauschen in Einpolsystemen vollständig ein [80]. Obwohl es sich in diesem Fall tatsächlich um ein Mehrpolsystem handelt, stellt die Annahme eines Einpolsystems aufgrund des dominierenden ersten Pols und dem Unterschreiten der 0 dB Grenze vor Erreichen des zweiten Pols eine gute Näherung dar. Die Serienwiderstände der Quelle sowie des Eingangs an M1 sind unbekannt, daher wurde die 3 dB-Grenzfrequenz durch Simulation der Schaltung ermittelt. Mit den aus der betrachteten 130 nm Technologie entnommenen Werten für g_{m1} und g_{m3} lässt sich für die Rauschspannung am Eingang ein Wert von $V_{\text{in}} = 412{,}76\,\mu\text{V}$ errechnen. Durch Simulation der Schaltung wurde ein Wert von $V_{\text{in, Simulation}} = 365{,}22\,\mu\text{V}$ ermittelt. Unter der Annahme, dass das Nutzsignal einen Signalhub von maximal der Schwellenspannung $U_{\text{TH}} = 730\,\text{mV}$ aufweist, lässt sich eine äquivalente Auflösung von $n = 10\,\text{Bit}$ für das analog implementierte Vergleichselement angeben (vgl. [106]).

In dieser Betrachtung spielt die herstellungsabhängige Abweichung (Mismatch) von Bauelementen in einer Schaltung noch keine Rolle. Da die Bauteilabweichung mit kleiner werdenden Strukturen zunimmt, soll an dieser Stelle der Einfluss der Herstellungsabweichungen auf das Auflösungsvermögen des Komparators ermittelt werden. Angaben zur statistischen Abweichung der Bauelemente unterliegen in diesem Fall dem Betriebsgeheimnis des Technologieanbieters, so dass an dieser Stelle der Einfluss auf das Auflösungsvermögen durch Monte-Carlo Simulation mit Mismatch-behafteten Bauelement-Modellen untersucht wird.

Der Komparator wird in der Simulation mit einer Referenzspannung von $U_{\mathrm{TH}} = 730\,\mathrm{mV}$ betrieben, der Mess-Eingang für das Membranpotential wird von 0 V bis 1,2 V durchfahren. Dabei wird die Eingangsspannung U_{trip} ermittelt, bei der der Komparator-Ausgang von 0 V kommend den Wert von $U_{\mathrm{DD}}/2$ erreicht. Im Idealfall ohne Bauteil-Streuung ergibt sich ein Wert von $U_{\mathrm{trip}} = U_{\mathrm{TH}}$. Durch 100 Monte-Carlo Simulationen der implementierten Schaltung, in denen nur lokale Abweichungen der Schaltung betrachtet werden, wurden ein Mittelwert von $U_{\mathrm{trip}} = 726{,}5\,\mathrm{mV}$ und eine Standardabweichung von $s_{\mathrm{trip}} = 11{,}0\,\mathrm{mV}$ für den Komparator des analog implementierten Neurons ermittelt. Wird nun der Wert der Standardabweichung als minimal zu unterscheidende Spannung bzw. als der Wert des LSB aufgefasst, so ergibt sich daraus eine äquivalente Auflösung des Komparators von maximal $\lceil \log_2 (U_{\mathrm{trip}}/s_{\mathrm{trip}}) \rceil = 6\,\mathrm{Bit}$.

4.2 Digitale Implementierungen

Digitale Umsetzungen von neuronalen Netzen werden durch verschiedene Anforderungen getrieben. Auf der einen Seite gibt es hochgenaue, detaillierte Modelle biologischer neuronaler Komponenten, welche klassisch in Rechnersystemen simuliert werden. Diese Modelle sollen durch eine teilweise Umsetzung des Software-Modells auf digitale Hardware in Form von Coprozessoren oder spezialisierten Erweiterungskarten für PCs die Simulation großer neuronaler Netze beschleunigen (z. B. [46]). Auf der anderen Seite werden Modelle bei der Umsetzung in digitale Hardware möglichst einfach gehalten, um die Einzelkomponenten zu größeren, komplexeren Modellen verschalten zu können oder in Form von einem massiv parallelen Rechenfeld mit kleinen spezialisierten Verarbeitungseinheiten zu großen Netzen verschalten zu können. Für die erstere Form sei der Leser auf die Diskussion der vorgestellten digitalen Modelle in Kap. 2 verwiesen, die Umsetzung von einfachen Modellen für den Aufbau großer, paralleler Felder ist Gegenstand des folgenden Abschnitts, in dem ein einfaches LIAF Neuronenmodell auf die digitalen Zieltechnologien FPGA und ASIC abgebildet wird.

Da die jeweils gewählte Zieltechnologie charakteristische Eigenschaften und Einschränkungen aufweist, lassen sich digitale Implementierungen, welche speziell für ein FPGA entworfen wurden, nicht ohne weiteres direkt auf die ASIC Technologie abbilden. Dieses ist zwar eingeschränkt möglich, führt aber in vielen Fällen zu Ergebnissen, die weit unter dem optimalen Ergebnis eines speziell angepassten Entwurfs liegen. Anders herum lassen

sich speziell auf ASIC zugeschnittene digitale Systeme nicht einfach auf FPGAs umsetzen, da die vom FPGA zur Verfügung gestellten komplexeren Elemente, z. B. Multiplizierer und DSP-Blöcke, oft nicht optimal genutzt werden können[2].

Aus diesem Grund werden im Folgenden zwei alternative Implementierungen eines digitalen LIAF Neurons vorgestellt, bei der die erste auf typische FPGA-Strukturen zugeschnitten ist, während die zweite Implementierungsvariante als bitseriell arbeitendes Neuron speziell auf die ASIC-Synthese zugeschnitten ist. Die Auswirkungen der Synthese der gewählten Variante auf die jeweils andere Technologie werden an den entsprechenden Stellen vermerkt.

Um den Ressourcenbedarf der digitalen Implementierungen zu reduzieren, werden verschiedene Methoden genutzt. Für die Reduktion der Fläche hat sich, sofern ein Gesamtsystem oder Teile eines Systems langsam arbeiten können, die Implementierung von Teilen der Schaltung als bitserielle Schaltung bewährt. Dabei werden Rechenoperationen nicht in einem Schritt mit bitparallelen Operanden durchgeführt, sondern in mehrere Operationen mit Operanden kleinerer Wortbreite aufgeteilt und nacheinander durchgeführt. Im einfachsten Fall wird beispielsweise bei der Addition zweier binärer Zahlen äquivalent zur Handrechnung verfahren, indem von den niederwertigsten Bits der Operanden ausgehend die beiden Operanden stellenweise unter Berücksichtigung eines Übertrags zur nächsthöheren Bitstelle addiert werden. Bei zwei Operanden mit n Bitstellen ergibt sich aus diesem Verfahren eine Latenz von n Takten bzw. $n + 1$ Takten, wenn der Überlauf der höchsten Bitstelle mit ausgegeben werden soll. Im Gegensatz zur einfachen bitparallelen Addition mit n Volladdierern, werden für die bitserielle Umsetzung nur ein Volladdierer und ein zusätzliches Register benötigt. Ein Element, welches einen besonders hohen Flächenbedarf bei der bitparallelen Umsetzung aufweist, ist der Multiplizierer. Daher sollen im nächsten Abschnitt der bitserielle Multiplizierer und seine verschiedenen Implementierungsvarianten vorgestellt werden.

Die Reduktion der Verlustleistung ist ein weiteres Problem zukünftiger Implementierungen in CMOS-Technologien mit Strukturgrößen von 90 nm und darunter. Durch sehr kleine Abmessungen der Transistorgeometrie sowie dünne Isolationsschichten nimmt der Anteil der statischen Verlustleistung durch den Anstieg der Leckströme und Subschwellenströme zu. Zur Kompensation bieten sich verschiedene Verfahren an, z. B. die Nutzung von *Silicon-On-Insulator* (SOI) Technologie, bei der die Transistoren der Schaltung auf isoliertem Silizium-Substrat aufgebracht werden. Eine andere Möglichkeit zur Reduktion von Leck- und Tunnelströmen ist die Verwendung von dickeren Isolationsschichten aus neuartigen Materialien [103] mit hoher Dielektrizitätskonstante (sog. *high-k* Dielektrika). Neben diesen Herangehensweisen kann die Verlustleistung direkt durch Skalieren der Versorgungsspannung beeinflusst werden. In digitalen Systemen haben sich daher Methoden des *Clock-Gatings*, dem Anhalten des Taktes einzelner Schaltungsteile, des *Power-Gatings*, dem Abschalten von Schaltungsteilen, und erweiterter, zum Teil kombinierter Mechanismen, wie z. B. dem Absenken der Versorgungsspannung bei Anhalten des Takts, etabliert. Die Absenkung der Versorgungsspannung soll im übernächsten Abschnitt

[2]Die optimale Ausnutzung von Ressourcen auf FPGA und ASIC ist Gegenstand anhaltender Forschung und Entwicklung im Bereich der Synthesewerkzeuge.

Tabelle 4.2: Flächenbedarf und Verzögerungszeit paralleler Multiplizierer-Varianten (aus [79]).

	Iteratives Array	5-Counter
Transistoren[a]	$30n^2$	$30\,(n^2+3n-2)$
Gatter[b]	$13n^2$	$13\,(n^2+3n-2)$
Verzögerungszeit	$(2n-1)\,T_{\mathrm{FA}}$	$(n+1)\,t$[c]

	Booth-Algorithmus	Pekmestzi
Transistoren[a]	$21n^2+52n+31$	$21n^2+89n-73$
Gatter[b]	$10n^2+23n+13$	$8.5n^2+40.5n-34$
Verzögerungszeit	$\left(\frac{n}{2}+n\right)T_{\mathrm{FA}}$	$(n+1)\,T_{\mathrm{FA}}$

[a] Die Umrechnung der Gatter in eine äquivalente Anzahl an Transistoren erfolgt auf Grundlage von [101].
[b] NAND2-Äquivalente.
[c] t ist die Verzögerungszeit einer 5-Counter Zelle.

verfolgt werden, in welchem eine digitale Standardzellenbibliothek vorgestellt wird, bei der die Versorgungsspannung im laufenden Betrieb bis in den Bereich um 200 mV, den sogenannten Subschwellenbereich, abgesenkt werden kann.

4.2.1 Bitserielle Multiplikation

Ein grundlegendes Bauelement der digitalen LIAF Neurone ist der Multiplizierer, der oft zur Berechnung des zeitlichen Verlaufs des Membranpotentials oder zur Bewertung der Aktionspotentiale in Synapsen eingesetzt wird. Parallele Implementierungen des Multiplizierers können die Rechenoperation zwar in kürzester Zeit ausführen, belegen aber nach der Synthese eine große Fläche. Der Flächenbedarf und die Verzögerungszeit ausgewählter Umsetzungen von Multiplizierern mit n Bit Wortbreite sind in Tab. 4.2 angegeben. Dabei wird die Verzögerungszeit in Vielfachen der Verzögerungszeit T_{FA} eines einzelnen Volladdierers angegeben. Zur besseren Vergleichbarkeit wurde die Fläche bei allen Varianten auf die Anzahl an Transistoren umgerechnet. Diese Angabe gibt die Größe der Schaltung mit einem bestimmten Fehler an, da unterschiedlich dimensionierte Gatter gleichen Typs und unterschiedlich große Transistoren hier nicht berücksichtigt werden. Innerhalb der gleichen CMOS-Technologie ist der Fehler bei allen Schaltungsvarianten gleich, so dass sich die Größen der einzelnen Implementierungen vergleichen lassen.

Während FPGAs zum großen Teil heute feste Multiplizierer-Blöcke bereitstellen, und durch Auslassen dieser Elemente kein Flächenvorteil entsteht, sollten diese Elemente aufgrund der Verzögerungszeit und angebotenen Wortbreiten von bis zu 18 Bit bei einer Umsetzung von digitalen LIAF Neuronen auf FPGAs genutzt werden.

Dagegen lässt sich durch den Einsatz bitserieller Multiplizierer bei der Umsetzung von

Tabelle 4.3: Flächenbedarf und Verzögerungszeit bitserieller Multiplizierer-Varianten (aus [60]).

	Ienne	Kalivas Typ I	Kalivas Typ II
Transistoren	$146n + 10$	$116n + 36$	$144n + 60$
Gatter[a]	—	—	—
Verzögerungszeit[b]	$2,5T_{\mathrm{FA}}$	$T_{\mathrm{MUX}} + T_{\mathrm{FA}}$	$T_{\mathrm{MUX}} + T_{\mathrm{FA}}$

[a] Keine Angaben zu NAND2-Äquivalenten.
[b] Verzögerungszeit pro Bit.

LIAF Neuronen auf einem ASIC eine große Fläche einsparen. Bitserielle Multiplizierer arbeiten häufig nach dem Prinzip der Zerlegung zweier Faktoren

$$\begin{aligned} A &= (a_n + \ldots + a_1 + a_0); \qquad a_n \in \{0, 2^n\} \quad \text{und} \\ B &= (b_n + \ldots + b_1 + b_0); \qquad b_n \in \{0, 2^n\} \end{aligned}$$

in einen parallelen Multiplikator und einen seriell zugeführten Multiplikanden, so dass sich die Rechenvorschrift

$$\begin{aligned} (a_n + \ldots + a_1 + a_0) \cdot (b_n + \ldots + b_1 + b_0) = {} & (a_n + \ldots + a_1 + a_0) \cdot b_n \\ & + \ldots \\ & + (a_n + \ldots + a_1 + a_0) \cdot b_1 \\ & + (a_n + \ldots + a_1 + a_0) \cdot b_0 \end{aligned} \tag{4.6}$$

ergibt, bei der sich jede dargestellte Addition seriell ausführen lässt. Für die Implementierung der Rechenvorschrift ist zu beachten, dass der Summand nach jedem Rechenschritt um eine Bitstelle nach links verschoben werden muss. Anschließend wird der mit dem Inhalt der jeweiligen Bitstelle UND-verknüpfte Multiplikator zum Zwischenergebnis addiert. Es ergibt sich eine Schaltung mit n Volladdierern und einem n Bit Register zur Speicherung des Zwischenergebnisses in jedem Rechenschritt. Zusätzlich benötigt der bitserielle Multiplizierer ein weiteres n Bit Register zum Vorhalten des parallel zugeführten Operanden.

Literaturangaben zu benötigter Fläche und der Verzögerungszeit pro berechneter Bitstelle von Umsetzungen bitserieller Multiplizierer sind in Tab. 4.3 zusammengefasst. Die benötigte Fläche wurde hier auf die Anzahl der Transistoren umgerechnet, um den Vergleich zu parallelen Implementierungen ziehen zu können. Die Verzögerungszeit bis zum Erscheinen der ersten Bitstelle am Ausgang setzt sich aus der Verzögerungszeit T_{FA} eines Volladdierers und der Verzögerung T_{MUX} der eingesetzten Multiplexer zusammen. Zusätzlich wird in diesen Umsetzungen Zeit für das bitserielle Zuführen der Operanden benötigt. Die in der Zahl benötigter Transistoren angegebene Fläche in Abhängigkeit von der Wortbreite ist in Abb. 4.10 dargestellt. Beim Vergleich des Flächenbedarfs der parallelen Varianten mit den bitseriellen Varianten der Multiplizierer ergibt sich ein Flächenvorteil der kleinsten parallelen Implementierung gegenüber der kleinsten bitseriellen Implementierung bis zu

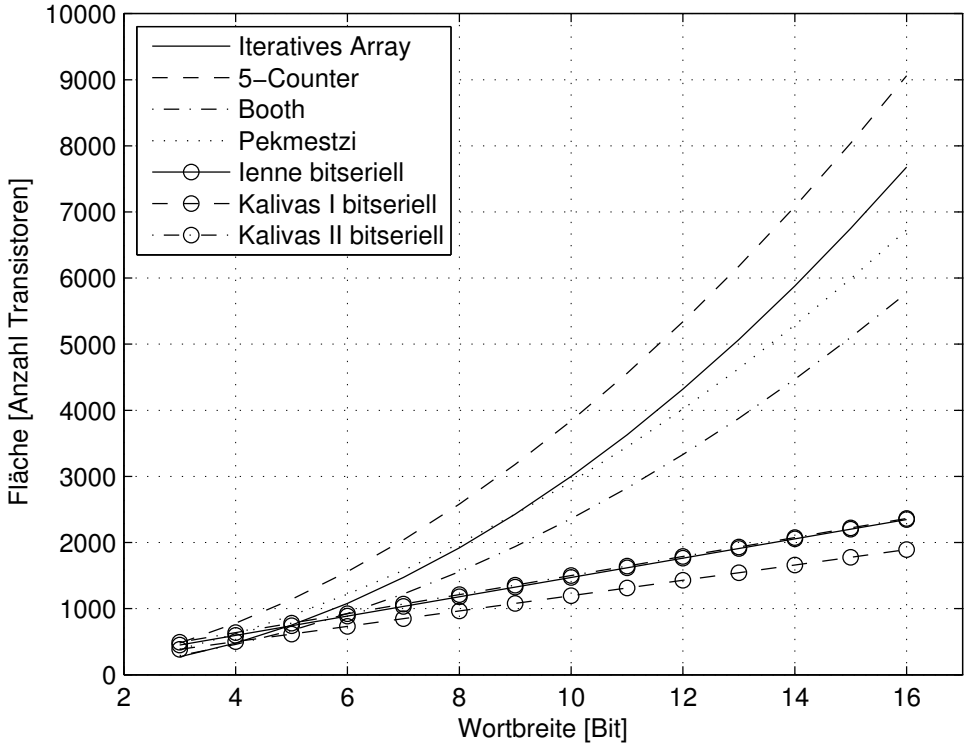


Abbildung 4.10: Flächenbedarf ausgewählter bitparalleler und bitserieller Multiplizierervarianten über der Wortbreite.

einer Wortbreite von 5 Bit. Da das hier betrachtete System jedoch eine Auflösung von mindestens 6 Bit aufweisen soll (vgl. Berechnung der äquivalenten Wortbreite), wurde die Entscheidung für die Untersuchung einer ASIC-Implementierung mit bitseriellen Multiplizierern getroffen. Die später vorgestellte FPGA-Umsetzung nutzt im Gegensatz dazu die von der Zielplattform bereitgestellten parallelen Multiplizierer.

4.2.2 Digitale ultra-low-power Standardzellenbibliothek

Bei der Integration von digitalen Systemen mit CMOS-Technologien, die eine Strukturgröße von 130 nm in Richtung der Nanoelektronik unterschreiten, wird die Betrachtung sowohl der steigenden statischen Verlustleistung als auch der Robustheit gegenüber Variation von Prozess-Parametern während der Herstellung einer Schaltung zunehmend wichtiger. Während bisher auf einem Chip vor allem weit auseinander liegende, gleiche Schaltungsteile voneinander abwichen, erreicht die Abweichung in CMOS-Technologien von 90 nm und darunter den lokalen Bereich um einen Transistor. In diesem Fall sind die klassischen Methoden des analogen Layouts, das *Matching* und *Common-Centroid* Layout-Technik [9, 41], für die Gewährleistung der Robustheit einer Schaltung nicht mehr ausreichend. Für digitale Gatter bedeutet dies, dass bereits zwei gleiche, nebeneinander liegende Gatter unterschiedliche Verzögerungs- und Schaltzeiten aufweisen können.

Diese negative Eigenschaft muss für zukünftige digitale Schaltungen berücksichtigt und kompensiert werden. Mit der zunehmenden Verkleinerung der Strukturgrößen geht auch die Zunahme der statischen Verlustleistung einher, wie später noch in den Randnotizen der Syntheseergebnisse anhand einer 350 nm Technologie und einer 130 nm Technologie gesehen werden kann. Ursache hierfür ist vor allem die Abnahme der Schichtdicken der Dielektrika, höhere Dotierungen sowie die kleinen geometrischen Abmessungen der Transistoren, die für erhöhte Leckströme, Unterschwellenströme und Tunnelströme verantwortlich sind. Die Kompensation der Tunnelströme erfordert neue Materialien, die als Dielektrikum eingesetzt werden können und eine höhere Schichtdicke bei Erhalt des elektrischen Feldes erlauben (high-k Dielektrika). Dieser Ansatz ist jedoch zur Reduktion der statischen Verlustleistung in bestehenden Technologien ungeeignet. Eine direkte Möglichkeit, die statische und dynamische Verlustleistung von Implementierungen in bestehenden CMOS-Technologien zu senken, ist das Herabsetzen der Versorgungsspannung der Standardzellenbibliothek. Die Methoden des *Clock-Gating* und des *Power-Gating* sind heute Stand der Technik und werden bereits von vielen Halbleiterherstellern unterstützt. Eine Reduktion der Verlustleistung um Größenordnungen ist mit dem Absenken der Versorgungsspannung in den Bereich der Subschwellenspannung möglich, in dem der Stromtransport im MOS-Transistor durch Diffusion (ähnlich dem Transport im Bipolar-Transistor) statt durch freie Ladungsträger in der Inversionsschicht erfolgt. Dabei zeigt sich die typische exponentielle Abhängigkeit des Stroms von der Drain-Source-Spannung, die bereits seit einiger Zeit in analogen Subschwellen-Schaltungen genutzt wird [76]. Im Folgenden wird das EKV-Modell [27] für die Beschreibung des Drainstroms eines nMOS- und eines pMOS-Transistors im Subschwellenbereich genutzt:

$$I_{\mathrm{D,\,n}} = 2n_{\mathrm{n}}U_{\mathrm{T}}^2\mu_{\mathrm{n}}C'_{\mathrm{ox}}\frac{W}{L}\exp\left(\frac{U_{\mathrm{G}}-U_{\mathrm{TH0,n}}}{n_{\mathrm{n}}U_{\mathrm{T}}}\right)\left(\exp\left(\frac{U_{\mathrm{S}}}{U_{\mathrm{T}}}\right)-\exp\left(-\frac{U_{\mathrm{D}}}{U_{\mathrm{T}}}\right)\right) \tag{4.7}$$

$$I_{\mathrm{D,\,p}} = 2n_{\mathrm{p}}U_{\mathrm{T}}^2\mu_{\mathrm{p}}C'_{\mathrm{ox}}\frac{W}{L}\exp\left(-\frac{U_{\mathrm{G}}-U_{\mathrm{TH0,p}}}{n_{\mathrm{p}}U_{\mathrm{T}}}\right)\left(\exp\left(-\frac{U_{\mathrm{S}}}{U_{\mathrm{T}}}\right)-\exp\left(\frac{U_{\mathrm{D}}}{U_{\mathrm{T}}}\right)\right) \tag{4.8}$$

Dabei beschreibt W die Kanalweite des Transistors, L die Kanallänge, U_{G} das Potential am Gate, U_{S} das Potential an Source und U_{D} das Potential am Drainanschluss des Transistors. Das Substrat ist der Bezugspunkt für alle Potentiale. Die Schwellenspannungen der Transistoren sind mit U_{TH0} beschrieben, die Ladungsträgerbeweglichkeiten mit μ. Die Temperaturspannung $U_{\mathrm{T}} = \mathrm{k_B T/q}$ wird mit 26 mV bei Raumtemperatur angenommen. Der Parameter C'_{ox} beschreibt die spezifische Kapazität des Gate-Oxids, Parameter n den *Slope-Faktor* (exponentieller Zusammenhang zwischen I_{D} und U_{G} im Subschwellenbereich).

Die Elemente der Subschwellenbibliothek wurden auf eine Versorgungsspannung von 200 mV ausgelegt. Dieser Wert erweist sich als eine gute Wahl bezüglich des relativen Fehlers $\sigma_{\mathrm{NM}}/\mu_{\mathrm{NM}}$ bei der Betrachtung der Streuung des Störabstands einzelner Gatter in Monte-Carlo Simulationen [113]. Unterhalb der Versorgungsspannung von 200 mV steigt der relative Fehler des Störabstands stark an.

Um die Zellen auf Robustheit zu optimieren, muss der Störabstand der Gatter maximiert werden. Dieses kann durch eine symmetrische Übertragungskennlinie erreicht werden, wenn gilt:

$$U_{\mathrm{out}}\left(U_{\mathrm{in}} = U_{\mathrm{DD}}/2\right) = U_{\mathrm{DD}}/2 \tag{4.9}$$

Die folgenden Betrachtungen werden für den Fall eines CMOS-Inverters durchgeführt. Komplexere Gatter in CMOS-Technik lassen sich für eine ähnliche Betrachtung auf diese Struktur zurückführen.

Aus der Gleichheit der Drainströme (4.7) und (4.8) eines Inverters mit den Transistoren M1 und M2 im obigen Arbeitspunkt folgt, dass die Bedingung (4.9) erfüllt werden kann, wenn gilt:

$$\frac{W_{\mathrm{p}}}{L_{\mathrm{p}}} = s \cdot \frac{W_{\mathrm{n}}}{L_{\mathrm{n}}} \quad \text{mit} \quad s = \frac{n_{\mathrm{n}}\mu_{\mathrm{n}} \exp\left(\frac{U_{\mathrm{DD}}/2 - U_{\mathrm{TH0,n}}}{n_{\mathrm{n}}U_{\mathrm{T}}}\right)}{n_{\mathrm{p}}\mu_{\mathrm{p}} \exp\left(\frac{U_{\mathrm{DD}}/2 + U_{\mathrm{TH0,p}}}{n_{\mathrm{p}}U_{\mathrm{T}}}\right)}.$$

Da die Prozessparameter der verwendeten CMOS-Technologie dem Betriebsgeheimnis des Herstellers unterliegen, wurde der Faktor s anhand von Simulationen mit $s \approx 2{,}3$ ermittelt. Daraus lässt sich die Dimensionierung jedes einzelnen Gatters unter Berücksichtigung der Symmetriebedingung und der einzuhaltenden Schaltzeiten ableiten.

Eine weitere Anforderung an die Dimensionierung der Transistoren ergibt sich aus der Minimierung der Streuung der Schaltungsparameter. Die Streuung der Schwellenspannung ist nach [22] umgekehrt proportional zur Wurzel aus der Gatefläche

$$\sigma\left(U_{\mathrm{TH}}\right) \propto \frac{1}{\sqrt{WL}},$$

so dass durch eine Vergrößerung der Transistorabmessungen die Parameterstreuung vermindert werden kann. Durch Verdoppelung der Transistorfläche wird die Streuung der Schwellenspannung halbiert. Um ein möglichst flächeneffizientes Layout zu erreichen, bietet es sich an, statt nur der Weite der Transistoren gleichzeitig auch die Länge der Transistoren in ungenutzten Bereichen der Standardzellen und damit auch die Robustheit zu erhöhen. Es kann gezeigt werden, dass mit der Variation der Schwellenspannung der Störabstand eines Gatters im gleichen Maße abnimmt, wie bei herkömmlichen Standardzellen (siehe Anhang A.3). Positiv wirkt an dieser Stelle der *reverse short-channel effect*, ein sekundärer Kurzkanaleffekt, der bei Verlängerung des Gates für eine Absenkung der Schwellenspannung und eine damit verbundene Erhöhung des Drainstroms sorgt. Damit nimmt neben der Robustheit des Gatters bei Verlängerung des Transistorgates gleichzeitig auch die Treiberstärke des Gatters in begrenzem Maße zu. Es ergibt sich ein Optimum für die Länge von Transistoren in Bezug auf die Treiberstärke von Gattern im Subschwellbereich, das bereits von Kim [63] beschrieben wurde.

Unter Berücksichtigung der genannten Bedingungen wurde eine Standardzellenbibliothek für die Synthese und das Platzieren und Verdrahten (engl. *Place and Route*) (PAR) von robusten digitalen Schaltungen im Subschwellenbereich entworfen. Die Bibliothek umfasst 11 Zellen mit verschiedenen Treiberstärken: Inverter, Treiber, Tristate-Treiber, NAND, NOR, AND-OR-INVERT, D-Flipflop, D-Latch, sowie Aufwärts-Level-Shifter von 0,2 V auf 1,0 V und Abwärts-Level-Shifter von 1,0 V auf 0,2 V als Interface zu Padzellen der verwendeten 90 nm CMOS-Technologie. Dieser Satz von Standardzellen umfasst die Minimalanforderung des Synthesewerkzeugs Synopsys Design Compiler, erweitert um ein NAND mit 3 Eingängen zur Implementierung von robusten Flipflops. Zusätzlich wurde eine AND-OR-INVERT Zelle bereitgestellt, welche die häufig auftretenden logischen Ausdrücke in disjunktiver Normalform

$$f(X) \bigvee_i \bigwedge_j (\neg) x_{ij}$$

abbilden kann. Die Entscheidung für die Umsetzung des Terms als AND-OR-INVERT Gatter in der Form

$$Z = \neg(AB + CD)$$

resultiert aus Betrachtungen zur Optimierung des Umfangs der Synthesebibliothek und der Analyse häufig in digitalen Schaltungen auftretender Gatterkombinationen. Die Bibliotheken für Synthese und Layout wurden manuell anhand von Simulationen an aus den erstellten Layouts extrahierten Netzlisten charakterisiert. Zur Verifikation der Ergebnisse wurde ein Testchip mit vier 32 Bit ALUs zur Analyse eines kleinen Systems, einzelnen Gattern zur Prüfung der grundlegenden Funktionalität und einem Ringoszillator zur Bestimmung des maximalen Takts automatisiert aufgebaut. Der Chip soll nach der Charakterisierung der Bibliothek mit einer Versorgungsspannung von mindestens 200 mV bis zu einer Versorgungsspannung von 1,0 V fehlerfrei funktionieren.

In Abb. 4.11 ist das auf einer Fläche von 1 mm^2 gefertigte Layout des Chips mit den großen IO-Pad-Zellen zu sehen. Der Chip besteht aus den vier eingezeichneten 32 Bit ALUs. Jede ALU besitzt zwei 32 Bit breite Register, den Akkumulator A und den Operanden B, der durch die angelegte Instruktion auf den Akkumulator abgebildet wird. Die einzelnen Register wurden als Scan-Register-Kette aufgebaut. Der Eingang der 4 Scan-Register-Ketten ist auf ein gemeinsames Eingangs-Pad des Chips geführt, um allen ALUs denselben Wert für nachfolgende Tests einzuprägen. Die Ausgänge der 4 Scan-Register-Ketten wurden auf eigene Ausgangs-Pads geführt, um den Ausfall einzelner ALUs im Test feststellen zu können. Im besten Fall entspricht das Ausgangssignal aller vier ALUs dem gleichen, korrekten Wert. Die ALU unterstützt 8 verschiedene Operationen: NOP, bitweises NAND, NOR und XOR, ADD, SUB, SHL (*shift left*) und ROL (*rotate left*). Für einen einfachen Test der grundlegenden Elemente des Chips wurden die Grundgatter NAND2, NOR2 und AND-OR-INVERT über eine eigene 4 Bit lange Scan-Register-Kette aufgebaut. Die Ausgänge der Grundgatter wurden über Level-Shifter direkt auf Ausgangs-Pads gelegt und erlauben eine direkte Beurteilung der Funktionsfähigkeit dieser Gatter. Zur

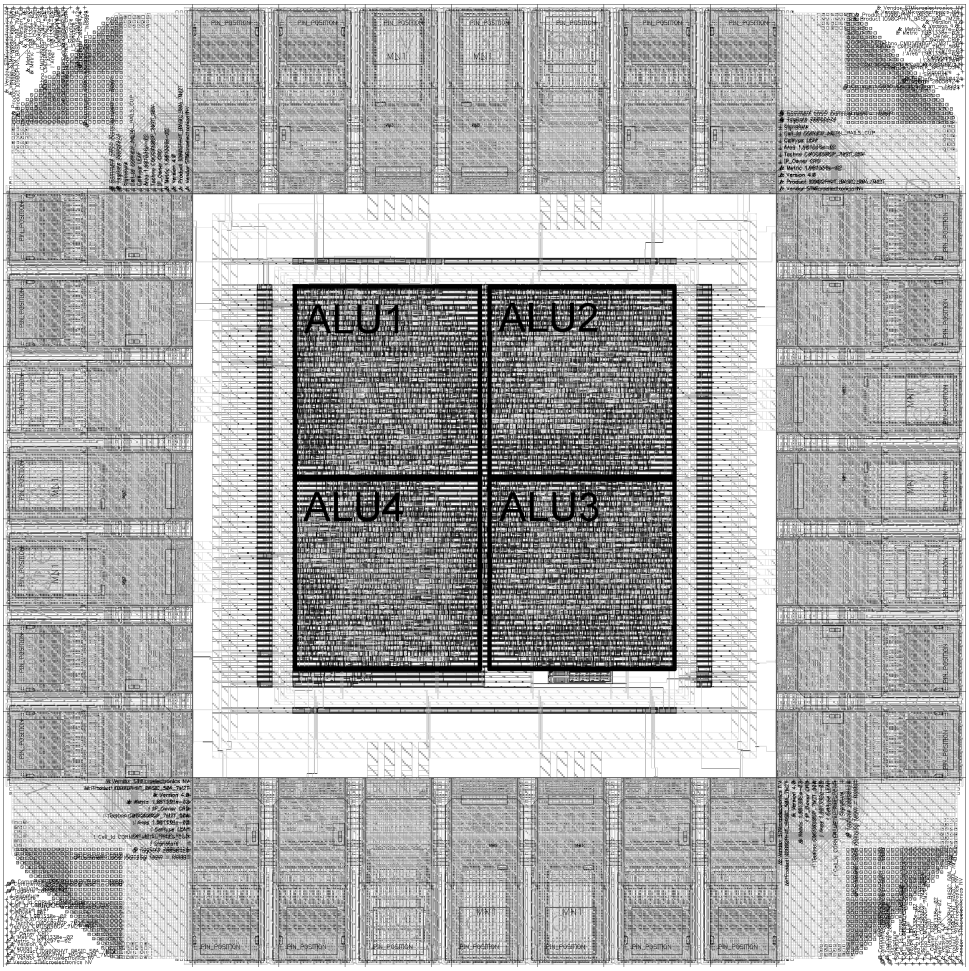


Abbildung 4.11: Layout des Testchips mit 4 ALUs in einer 90 nm CMOS-Technologie mit Versorgungsspannung im Subschwellenbereich.

ersten Bestimmung des maximalen Takts, mit dem die Schaltung arbeiten kann, wurde unterhalb der ALU Nr. 3 ein Ringoszillator mit 33 Stufen und einem Teiler aus Flipflops im Verhältnis von $f_{\max}/256 = f_{\text{osc}}$ implementiert. Der maximal mögliche Takt lässt sich so durch Messung des erzeugten Takts am Ausgang f_{osc} bestimmen. Der Takt-Teiler wurde notwendig, da die eingesetzten IO-Zellen nur für Signale mit einer Bandbreite von 80 MHz spezifiziert sind. Als Besonderheit ist anzumerken, dass auf Seiten des Chips an den Eingangs- und Ausgangs-Pads Level-Shifter zur Umsetzung der Signale auf dem Chip von einem Spannungswert von 200 mV bis 1,0 V auf eine Spannung für die IO-Pads von 1,0 V eingebracht wurden. Durch diese Maßnahme ist der exakte Verlauf der Spannungen der einzelnen Gatter zwar nicht mehr direkt ermittelbar, da keine Möglichkeit der Messung auf dem Chip besteht, jedoch wird erwartet, dass ein Betrieb des Chips mit Eingangssignalen von 1,0 V zu einer deutlich höheren Robustheit des Gesamtsystems führt, da sich Störungen durch Rauschen auf 1,0 V Signale weniger auswirken, als auf Signale mit einem Hub von 200 mV. Gleichzeitig wird angenommen, dass die Rauschquellen auf dem Chip selbst deutlich schwächer ausgeprägt sind als in der Testumgebung, so dass ein fehlerfreier

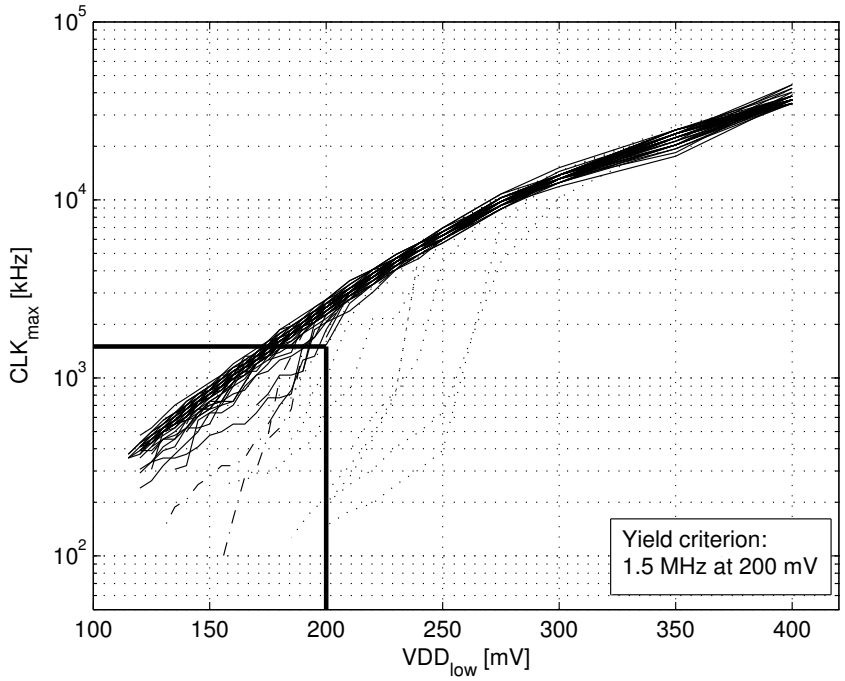


Abbildung 4.12: Maximaler Takt der ALUs über der Versorgungsspannung.

Betrieb mit 200 mV Versorgungsspannung für den Kernbereich des Chips begünstigt wird. Als Versorgungsspannungen für den Chip wurde die Spannung für die IO-Pads VDD von der Versorgungsspannung für den *Low-Power* Bereich $\mathrm{VDD_{core}}$ getrennt. Zur Reduktion des Rauschens wurden auch die Masse-Anschlüsse des IO-Bereichs und des *Low-Power* Bereichs voneinander getrennt und erst extern zusammengeführt.

Für den Test der ALU wurden zwei 32 Bit Wörter mit einem festen, niedrigen Takt in die Register der vier ALUs geschrieben. An dieser Stelle wurde ein relativ niedriger Takt gewählt, da nicht die maximale Arbeitsfrequenz der Level-Shifter getestet werden sollte, sondern der maximal mögliche Takt der ALUs. Der Scan-Modus der Register wurde anschließend abgeschaltet, und die ALUs wurden für 10^6 Zyklen mit frei einstellbarem Takt betrieben. Während der Ausführung wurde die Operation ADD durchgeführt, bei der alle 32 Bit des Akkumulators ihren Zustand wechselten und mehrfache Überläufe des Akkumulators auftraten. Während der Addition wird der längste Pfad in der ALU genutzt, welcher den maximal möglichen Takt des Systems bestimmt. Nach Durchführung der Additionen wurde das Ergebnis aus Akkumulator A und Register B mit niedrigem Takt ausgegeben und verifiziert. Der Testdurchlauf startete mit einer Versorgungsspannung für den Kern von $\mathrm{VDD_{core}} = 400\,\mathrm{mV}$ und einem Takt von $\mathrm{CLK_{max}} = 62{,}5\,\mathrm{MHz}$. Der Takt wurde verringert, bis alle ALUs korrekte Ergebnisse für die Berechnung zurücklieferten. Im nächsten Schritt wurde die Versorgungsspannung des Kerns herabgesetzt und wieder die maximale Frequenz bestimmt. Im Subschwellbereich unter 250 mV wurden die Abstufungen der Versorgungsspannung feiner gewählt, da erwartet wurde, dass die Prozessvariation unterhalb dieser Spannung stärker in der Funktion der Schaltung sichtbar wird.

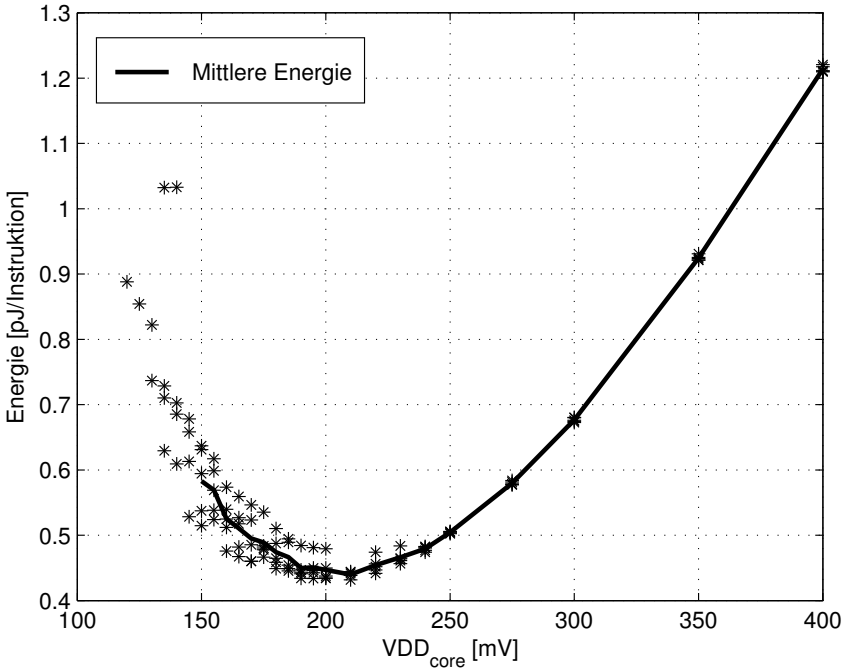


Abbildung 4.13: Energieumsatz der ALU pro Instruktion über der Versorgungsspannung.

In Abb. 4.12 ist der maximal erreichte Takt der ALUs über der Versorgungsspannung der ALUs aufgetragen. Als Kriterium für eine funktionierende ALU wurde aus dem vorherigen Syntheseergebnis die zu überschreitende Grenze von 1,5 MHz bei einer Versorgungsspannung von 200 mV gewählt. ALUs, die diese Grenze nicht überschreiten, wurden in der Abbildung mit gestrichelter Linie dargestellt und fallen aus den weiteren Betrachtungen heraus. Es gibt wenige Muster, die das Auswahlkriterium erfüllen, kurz darunter jedoch schnell in der Funktion nachlassen. Die zwei Exemplare wurden ebenfalls für die weitere Betrachtung der Energieaufnahme vernachlässigt. Unter Berücksichtigung des Ausbeutekriteriums ergibt sich eine Ausbeute von funktionierenden ALUs von 88,3%. Die ALUs wurden bis zu einer Versorgungsspannung von 110 mV getestet, wobei die zu erzielende Versorgungsspannung von 200 mV von nur einer ALU nicht erreicht werden konnte. Der durchschnittlich zu erreichende Takt bei der Zielgröße einer Versorgungsspannung von 200 mV liegt bei 2,5 MHz. Unabhängig vom maximal möglichen Takt konnten 75% aller gefertigten ALUs bei 120 mV Versorgungsspannung betrieben werden. Die ALU mit der niedrigsten Versorgungsspannung funktionierte noch bei 115 mV [113]. Im einleitenden Abschnitt zur Standardzellenbibliothek mit Subschwellenspannung als Versorgungsspannung wurde bereits auf die Möglichkeit der Reduktion der Verlustleistung hingewiesen. Um vergleichbar zu aktuellen relevanten Veröffentlichungen zu sein, wird im Folgenden die pro Instruktion umgesetzte Energie betrachtet. Die Energieaufnahme wurde sowohl in Ruhe (statische Verlustleistung) als auch bei maximal möglichem Takt (dynamische Verlustleistung) für die ALUs ermittelt und auf eine einzelne Instruktion normiert.

Abbildung 4.13 zeigt die aufgenommene Energie für die als funktionierend klassifizierten ALUs über einen Versorgungsspanungsbereich von 120 mV bis 400 mV. Die einzelnen Messergebnisse sind als Sterne dargestellt. Darübergelegt ist die durchschnittlich aufgenommene Energie der betrachteten ALUs mit einem klaren Minimum bei einer Versorgungsspannung von $\mathrm{VDD_{core}} = 210\,\mathrm{mV}$, sehr nahe der Versorgungsspannung, auf die während des Layouts der Bibliothek optimiert wurde. Unterhalb von 150 mV wurde die Streuung der Messwerte aufgrund des abnehmenden Stichprobenumfangs sehr groß, so dass an dieser Stelle die Auswertung abgebrochen wurde. Alle Messungen wurden bei einer Raumtemperatur von 25°C durchgeführt. Das beobachtete Minimum des Energieumsatzes pro Instruktion liegt bei 0,45 pJ bei einer Versorgungsspannung von $\mathrm{VDD_{core}} = 210\,\mathrm{mV}$ und einem Takt von 3 MHz. Als Vergleich des erreichten Werts werden an dieser Stelle relevante Veröffentlichungen der letzten Jahre herangezogen. In [16] zeigen Calhoun und Chandrakasan einen 32 Bit Kogge-Stone Addierer in einer 90 nm CMOS-Technologie mit einer minimalen Energie von 0,1 pJ pro Addition bei einer Versorgungsspannung von $\mathrm{VDD} = 330\,\mathrm{mV}$ und einem maximalen Takt von 50 kHz. In [39] wird ein 8 Bit Sub-Threshold Prozessor in einer 130 nm CMOS-Technologie mit 3,5 pJ pro Instruktion bei einer Versorgungsspannung von $\mathrm{VDD} = 350\,\mathrm{mV}$ und einem maximalen Takt von 354 kHz gezeigt. Zhai beschreibt in [104] einen 8 Bit CISC-Prozessor mit 0,85 pJ pro Instruktion bei einer Versorgungsspannung von $\mathrm{VDD} = 280\,\mathrm{mV}$ in einer 130 nm CMOS-Technologie. Eine Angabe zum maximal möglichen Takt gibt es in dieser Veröffentlichung nur für eine Versorgungsspannung von 260 mV und wird mit 84,7 kHz angegeben.

4.2.3 Leaky Integrate and Fire Neuron

Integrate and Fire Neurone können in digitalen Systemen auf unterschiedliche Weise umgesetzt werden. Für die Umsetzung auf FPGAs wird ein Ansatz gewählt, der auf viele gleiche Elemente setzt, die durch die Verbindungs-Struktur des FPGAs flexibel miteinander verschaltet werden können und so größere komplexe Modelle abbilden können. Dieses wird am Beispiel eines in [21] veröffentlichten LIAF Modells gezeigt.

FPGA optimierter Entwurf (SIRENS)

Die Dynamik des Membranpotentials eines LIAF Neurons kann nach [21] mit

$$C\frac{dV}{dt} = I - g_L V + \sum_l A\left(t - t_l\right) \tag{4.10}$$

$$A(t) = \begin{cases} g_L V_A \mathrm{e}^{\xi t}, & 0 < t \leq \Delta \\ -C(V_T - V_0 + V_M)\delta(t - \Delta), & \Delta < t \end{cases} \tag{4.11}$$

ausgedrückt werden. Diese Darstellung entspricht der bekannten Struktur für LIAF Neurone (vgl. Kap. 2.1.2), erweitert um die Beschreibung des Verhaltens des Membranpotentials

bei der Aktionspotentialerzeugung. Dabei beschreibt der Parameter C die Membrankapazität des Zellkörpers, I den postsynaptischen Strom präsynaptisch feuernder Neurone, von denen das empfangende Neuron die Eingangssignale erhält. Der Parameter g_L ist der Leitwert, der die passiven Leckströme und aktiven Transportströme durch die Zellmembran modelliert, $A(t)$ beschreibt das Verhalten des Neurons, wenn das Membranpotential die Feuerschwelle V_T überschreitet.

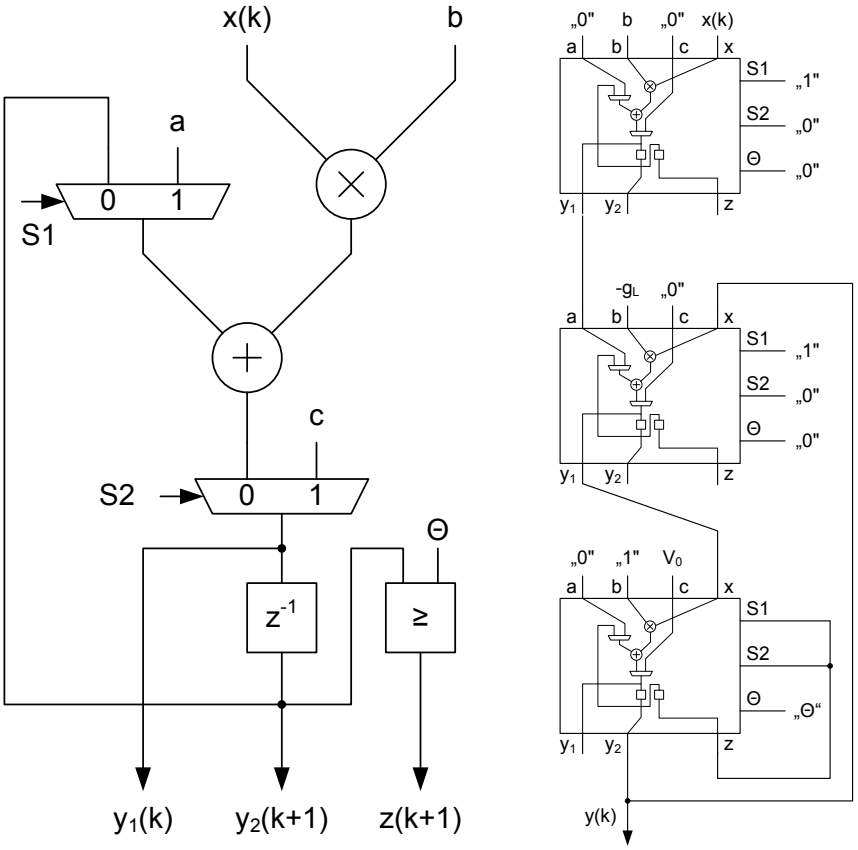
Die Parameter ξ und V_A beschreiben die Geschwindigkeit des Anstiegs und das Maximum des Aktionspotentials, welches vom Neuron nach dem Überschreiten der Feuerschwelle erzeugt wird. Der Parameter Δ beschreibt die Dauer des Aktionspotentials. V_M ist das Maximum des Aktionspotentials, nach dessen Erreichen das Membranpotential auf ein Ruhepotential von V_0 zurückgesetzt wird. Mit $\delta(\cdot)$ ist der Dirac-Impuls bezeichnet, mit dem das Membranpotential zurückgesetzt wird.

Dieses Modell ist zwar ein zeitkontinuierlicher Entwurf eines LIAF Neurons, soll aber der Ausgangspunkt für das im Folgenden vorgestellte zeitdiskrete digitale Modell sein, welches aus vielen gleichen Untereinheiten zusammengesetzt werden kann.

Im Folgenden Abschnitt wird die Implementierung einer digitalen Struktur zum Aufbau pulscodierter neuronale Netze vorgestellt. Das umgesetzte Neuronenmodell basiert auf einem LIAF Modell, dessen Funktion in Kapitel 2.1.2 beschrieben wurde. Das digitale Neuron stellt eine Approximation der Gleichung (4.10) mittels eines spezialisierten Rechenelements, der Prozesseinheit (PE) dar. Die Auflösung der PE beträgt im folgenden Beispiel 16 Bit in Festkommadarstellung, von denen 13 Bit für die Nachkommastellen, 2 für die Vorkommastellen und ein Bit für das Vorzeichen vorgesehen sind. In einem früheren Abschnitt ist zum Vergleich mit analogen Neuronen die minimale Wortbreite für diese Architektur bereits hergeleitet worden. Die Erweiterung der Wortbreite von 6 Bit auf 16 Bit wird durch die für die Funktion des Neurons wichtigen einzustellenden Werte notwendig.

Aufbau

Die Basis der SIRENS Einheit (Simple Reconfigurable Neural Hardware Structure) [108] bildet die Prozesseinheit, deren Aufbau in Abb. 4.14a dargestellt ist. Die PE besteht aus zwei Multiplexern, welche den Multipliziererblock, einen Addierer und einen Vergleichselement mit vorgeschaltetem Register zu unterschiedlichen Strukturen verschalten können. Die abgebildete Struktur stellt klassisch eine *multiply-and-accumulate* (MAC) Einheit dar, welche sich ideal auf heutige FPGA bzw. Digitale Signalprozessor-Blöcke (DSP) abbilden lässt. Der Ausgang des Registers ist über einen Multiplexer auf das Register zurückgeführt, um die Funktionalität der Integration schon in der Prozesseinheit zu implementieren. Die Wahl dieser Struktur erlaubt es, die PE sowohl als klassischen Funktionsapproximator zu nutzen, wie in [26] detailliert beschrieben wird, als auch durch die Erweiterung um das Vergleichselement zum Aufbau von pulsenden LIAF Neuronen zu nutzen. Dazu werden drei gleiche Prozesseinheiten in der in Abb. 4.14b gezeigten Weise verschaltet und bilden so ein komplexeres zeitdiskretes, pulsendes Neuron.



(a) Grundelement. (b) LIAF Neuron.

Abbildung 4.14: Grundelement eines einzelnen Neurons zur zeitdiskreten Approximation kontinuierlicher Funktionen (a) und Verschaltung dreier Grundelemente zur Funktion eines Leaky Integrate-and-Fire Neurons (b).

Diskrete Gleichung und Übertragungsfunktion

Das in Abb. 4.14a dargestellte digitale System ist durch die Möglichkeit der unterschiedlichen Beschaltung der Multiplexer S1 und S2 durch die in (4.12) bis (4.15) aufgeführten Gleichungen gegeben. Im Folgenden wird zunächst nur der verzögerte Ausgang $y_2(k)$ betrachtet. Zur Vollständigkeit ist der unverzögerte Ausgang $y_1(k)$ aufgeführt, welchem erst bei der Erweiterung des IAF Neurons zum LIAF Neuron eine Bedeutung zukommt.

$$y_{2,1}(k) = a + bx(k-1) \qquad \text{falls S1=1} \wedge \text{S2=0} \qquad (4.12)$$

$$y_{2,2}(k) = y_{2,2}(k-1) + bx(k-1) \qquad \text{falls S1=0} \wedge \text{S2=0} \qquad (4.13)$$

$$y_{2,3}(k) = c \qquad \text{falls S2=1} \qquad (4.14)$$

$$z(k) = \text{sign}(y_{2,(1,2,3)}(k) - \Theta) \qquad (4.15)$$

Für die Funktion als IAF Neuron ist die Nutzung von (4.13) und (4.14) notwendig,

alternativ kann auch eine Kombination aus (4.12) und (4.13) genutzt werden. Für die erste Variante wird Ausgang $z(k)$ an den Eingang des Multiplexers S1 angeschlossen.

In den folgenden Betrachtungen wird aus Gründen der Übersichtlichkeit nur noch der verzögerte Ausgang $y_2(k)$ angegeben und wird als Ausgang des Grundelements mit $y(k)$ bezeichnet.

Ein System mit der oben beschriebenen Rückkopplung des Ausgangs $z(k)$ auf den Multiplexer S2 ergibt eine Differenzengleichung von:

$$\begin{aligned} y(k) = {} & bx(k-1) + y(k-1) \\ & + (c - y(k-1)) \cdot \delta(k-n) \\ & + (c - y(k-1)) \cdot \delta(k-2n) \\ & + \dots \\ & + (c - y(k-1)) \cdot \delta(k-mn) \end{aligned} \tag{4.16}$$

Der neu eingeführte Parameter n bestimmt den Zeitpunkt, zu dem das System auf einen Wert c (oder in der zweiten möglichen Variante auf den Wert a) zurückgesetzt wird. Der Zeitpunkt n berechnet sich aus der Feuerschwelle Θ nach der Vorschrift

$$n = \left(\left\lceil \frac{\Theta}{b} \right\rceil - 1 \right) \cdot V_{\text{tast}} + 1. \tag{4.17}$$

Dabei wird durch den Parameter V_{tast} das mittlere Puls-Perioden-Verhältnis einer einlaufenden Bitfolge beschrieben (z. B. 1 Puls alle 2 Takte $\rightarrow V_{\text{tast}} = \frac{1}{2}$).

Zusammengefasst ergibt sich die Funktion des Ausgangs zu

$$y(k) = bx(k-1) + y(k-1) + c \cdot \sum_{m=1}^{\infty} \delta(k-mn) - y(k-1) \cdot \sum_{m=1}^{\infty} \delta(k-mn). \tag{4.18}$$

Eine einzelne Prozesseinheit kann bereits ein IAF Neuron nachbilden, wenn der Leaky-Term des nachzubildenden Neuronenmodells vernachlässigt werden kann. Für die Berücksichtigung des Leaky-Terms müssen mehrere PE verschaltet werden, wie in Abb. 4.14b dargestellt ist. Der Ausgang $y(k)$ der Struktur aus drei Grundelementen weist in der angegebenen Konfiguration zum Zeitpunkt k den Wert

$$y(k) = y(k-1) + \hat{b}x(k) - \hat{g}_L y(k-1) \tag{4.19}$$

auf, solange die Feuerschwelle Θ nicht erreicht wird. Bei Überschreiten der Feuerschwelle wird $y(k)$ auf den Wert von V_0 zurückgesetzt. Die Überführung der Parameter des

Tabelle 4.4: Relativer Diskretisierungsfehler der Parameter des diskreten Systems.

Auflösung [Bit]	11	12	13	14	15	16
Relativer Fehler $\hat{b}$	15,9%	15,9%	5,4%	0,1%	0,1%	0,1%
Relativer Fehler $\hat{g}_L$	8,9%	8,9%	8,9%	8,9%	3,3%	0,4%

kontinuierlichen Modells in die Parameter des diskreten Modells wird durch einen Koeffizientenvergleich von (4.10) und (4.19) unter Berücksichtigung der Integrationsschrittweite des diskreten Systems von $T_{\text{cyc}} = 1/f_{\text{clk}}$ und einem zum Wert 1 gewählten Eingang $\hat{I}$ ermittelt. Für die Parameter des diskreten Systems ergeben sich die Koeffizienten zu

$$\hat{b} = \frac{I}{C}\hat{I}T_{\text{cyc}} \tag{4.20}$$

$$\hat{g}_L = \frac{g_L}{C}T_{\text{cyc}}. \tag{4.21}$$

Die Erweiterung der Wortbreite von 6 Bit auf 16 Bit liegt in der Minimierung des Fehlers bei der Überführung der Parameter des kontinuierlichen Systems in das diskrete System. Bei der Darstellung des Wertes für $\hat{b}$ reichen an dieser Stelle erst 11 binäre Nachkommastellen aus, um den relativen Diskretisierungsfehler unter 1% zu bringen, für den Wert von $\hat{g}_L$ werden dazu 13 Nachkommastellen benötigt (siehe Tab. 4.4).

Mit den in (4.20) und (4.21) überführten Werten wurde eine Simulation des digitalen Systems mit einem Systemtakt von 100 Hz durchgeführt und die Ausgabe der Simulation dem zeitkontinuierlichen Modell gegenübergestellt. Das Simulationsergebnis des pulsenden LIAF Neurons ist in Abb. 4.15 dargestellt und emuliert das zeitkontinuierliche Neuronenmodell in ausreichender Weise. Für die Nachbildung des exponentiellen Anstiegs des Aktionspotentials sind weitere PE notwendig, welche die Exponentialreihe $\exp(t) = \sum_{n=0}^{\infty} \frac{t^n}{n!}$ approximieren. Für die Nutzung der vorgestellten Struktur als Funktionsapproximator sei an dieser Stelle auf die Arbeit [26] verwiesen.

Das vorgestellte flexible Konzept eines Neurons für Approximationsaufgaben sowie der Möglichkeit, komplexere Modelle wie z. B. ein LIAF Neuron zu erzeugen, wurde mit verschiedenen Synthesewerkzeugen auf unterschiedliche Zielplattformen synthetisiert. Dabei wurden Abschätzungen für die durchschnittliche Leistungsaufnahme der Schaltung sowie für den Flächenbedarf ermittelt. Zu den Ergebnissen der Synthese auf eine ASIC Zielplattform lässt sich anmerken, dass die Synthese nur der erste Schritt in der Implementierung der beschriebenen Schaltung auf dem ASIC ist. Die während der Synthese ermittelten Größen geben nur – wenn auch relativ gute – Richtwerte für die tatsächlichen Größen wieder. Die endgültigen Ergebnisse lassen sich erst nach dem Platzieren und Verdrahten (PAR) ermitteln und liegen üblicherweise leicht höher, als die in der Synthese ermittelten Werte.

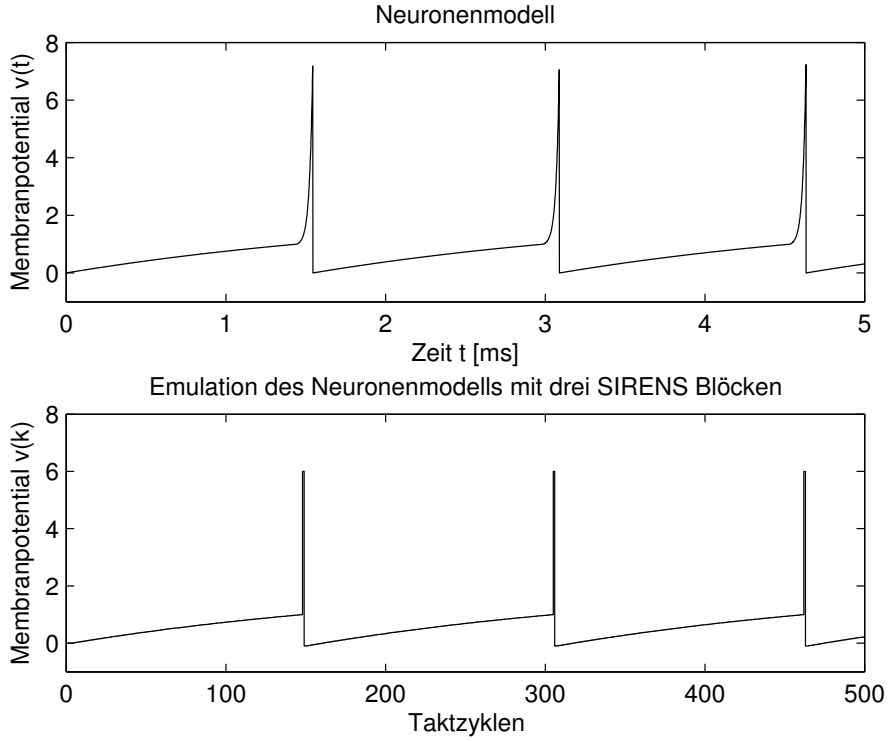


Abbildung 4.15: Approximation der Funktion des kontinuierlichen pulscodierten Neurons nach (4.10) mit der SIRENS Struktur. Der Verlauf des Membranpotentials des kontinuierlichen LIAF Neurons wurde mit den Modellparametern $\xi = 50; \Delta = 0{,}1; v_A = 1; I = 1{,}3; g_L = 0{,}6; C = 1{,}4$ erzeugt, das Simulationsergebnis des digitalen pulscodierten Neurons aus drei Grundelementen mit den Parametern: $\hat{I} = 1{,}0; \hat{b} = 0{,}929{\cdot}10^{-2}; \hat{g}_L = 0{,}429{\cdot}10^{-2}$.

In Tabelle 4.5 sind die Syntheseergebnisse für den Flächenbedarf der vorgestellten SIRENS Struktur in verschiedenen CMOS-Technologien dargestellt. Diese wurden in allen folgenden Synthesen für drei Arbeitsbereiche der Schaltungen analysiert, welche mit den Bezeichnungen *best*, *typical* und *worst* gekennzeichnet werden und sich in den Parametern für Versorgungsspannung, Temperatur und Skalierung der Strukturen unterscheiden. Eine Übersicht der relevanten Parameter der verschiedenen Arbeitsbereiche sind in Tabelle B.2 im Anhang zu finden. Die Synthese wurde für eine Taktfrequenz von 1 MHz mit Optimierung auf geringe dynamische Verlustleistung mit einer anschließenden Optimierung der Chipfläche durchgeführt. Die vorliegende Implementierung nutzt 16 Bit breite Register zur Verarbeitung von vorzeichenbehafteten Fixpunkt-Zahlen mit 2 Bit für den ganzzahligen Anteil und einem zusätzlichem Vorzeichen-Bit. Diese Wortbreiten sind notwendig, um die Berechnungen des Neurons mit ausreichender Genauigkeit (verglichen mit dem mathematischen Modell) durchführen zu können. Obwohl es sich bei dieser Struktur um eine sehr flexible und kompakte Variante für digitale Neurone handelt, machen sich Nachteile durch den Einsatz von Addierer- und Multiplizierer-Strukturen, welche das Ergebnis in einem Taktzyklus bereitstellen können, bemerkbar. Diese Tatsache führt

Tabelle 4.5: Flächenbedarf und dynamische Verlustleistung der SIRENS Struktur mit 16 Bit Wortbreite bei Synthese auf 1 MHz Takt.

Technologie	Fläche [μm^2]	Verlustleistung [μW]		
		best	typical	worst
350 nm Technologie	265.390,6	363,5	349,2	353,3
130 nm Technologie B	25.647,7	13,7[a]	10,0[a]	8,0[a]
130 nm Technologie A	25.755,6	19,6[a]	14,8[a]	12,4[a]
	16.851,7[b]	17,7[b]	13,4[b]	11,3[b]
90 nm Sub-V_{TH}	25.258,0[b,c]		0,283[b,c]	
Xilinx FPGA	41 Slices		580,0[d]	
Virtex2 Pro (130 nm)	3 MULT18X18			

[a] Der zusätzliche Anteil der Verlustleistung durch Leckströme bewegt sich mindestens in der Größenordnung der hier angegebenen dynamischen Verlustleistung.

[b] Die Hierarchie der Implementierung wurde aufgelöst. Ungenutzte Elemente (z. B. konstante beschaltete Blöcke) wurden während der Synthese identifiziert und entfernt. Dieses Syntheseergebnis ist daher nur noch als LIAF Neuron einsetzbar.

[c] Maximale dynamische Verlustleistung nach Abbildung auf eine handentworfene Standardzellenbibliothek mit 200 mV Versorgungsspannung. Angaben zur verwendeten Bibliothek finden sich unter Kapitel 4.2.2. Das Sub-V_{TH} System wurde auf einen Takt von 500 kHz synthetisiert. Die statische Verlustleistung liegt in der Größenordnung der dynamischen Verlustleistung.

[d] Xilinx ISE 9.2i gibt in der Power-Analyse 206,72 mW statische Verlustleistung an.

zu dem Ergebnis, dass die digitale Schaltung zwar mit einer niedrigen Taktrate von bis zu 1 MHz betrieben werden kann, jedoch eine große Fläche belegt. Es wird ersichtlich, dass diese Struktur für den Einsatz auf FPGA mit speziellen *full-custom* Blöcken (Multipliziereinheit, Volladdierer, DSP) optimiert ist. Die in der ersten Synthese und in Tabelle 4.5 gezeigten Werte für die Verlustleistung beruhen auf der Annahme von Schaltwahrscheinlichkeiten der einzelnen Gatter, welche von der Taktfrequenz abgeleitet werden. Da die genauen Werte für die Schaltwahrscheinlichkeiten aufwendig zu ermitteln sind, wurde im nächsten Schritt die Verlustleistung anhand von Simulationen und daraus gewonnenen Schalthäufigkeiten (im *Backannotation*-Verfahren) der einzelnen Gatter in Abhängigkeit von der Ausgangstaktrate des Neurons für einen 130 nm Prozess ermittelt. Das digitale System wurde mit Synopsys Design Compiler Z-2007.03-SP4 synthetisiert. Beim Übergang der Synthese von der 350 nm Techologie auf die 130 nm Technologien zeigt sich, dass die allgemeinen Skalierungsregeln der CMOS-Technologie (siehe Anhang B) greifen. Die Abnahme der Verlustleistung um einen Faktor von ungefähr 23 ergibt sich aus dem Quadrat des Verhältnisses der minimalen Strukturgrößen von 130 nm zu 350 nm, dem Quadrat des Verhältnisses der Versorgungsspannung von 3,3 V zu 1,2 V und der Reduktion des Gateoxids um einen Faktor von ca. 130/350. Gleichzeitig wird deutlich, dass die Ergebnisse für die Verlustleistung stark vom jeweiligen Technologieanbieter abhängen. So liegt die Verlustleistung bei Technologie A je nach Betriebsfall zwischen 40% und 50% höher, als bei Technologie B eines anderen Anbieters. Durch die Auflösung der flexiblen Struktur der SIRENS Implementierung und die Festlegung auf die Funktion von jeweils drei Funktionsblöcken zu einem LIAF Neuron lässt sich die Fläche um fast

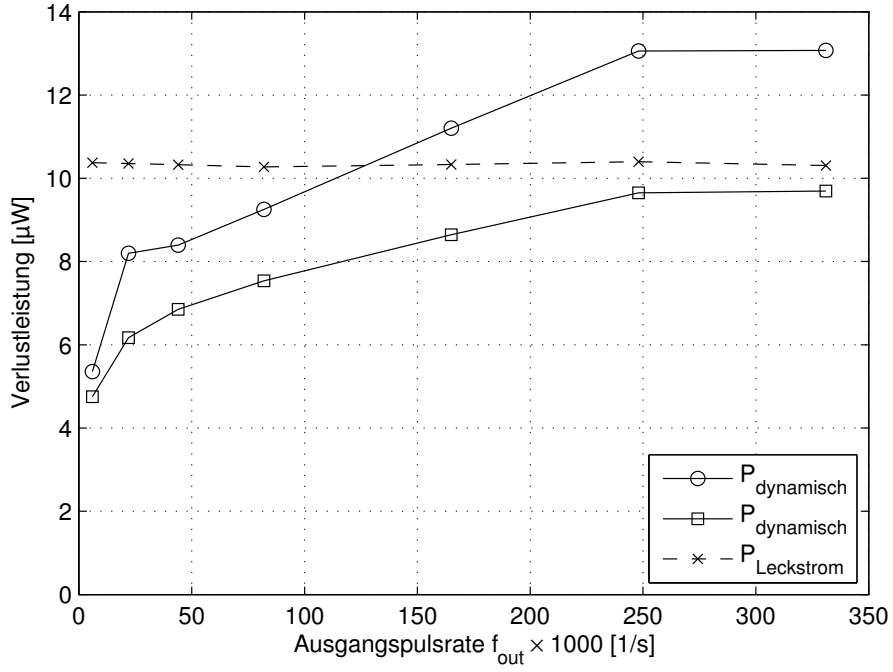


Abbildung 4.16: Dynamische Verlustleistung (durchgezogene Linien) und Verlustleistung durch statische Leckströme (gestrichelte Linie) der SIRENS-Struktur in einem 130 nm Prozess.

35% verringern. Die Verlustleistung verringert sich dabei aber um weniger als 30% im typischen Betriebsfall, in den Randbereichen um weniger als 10%.

Die Synthese der SIRENS Struktur auf eine Standardzellenbibliothek für den Subschwellenbetrieb in einer 90 nm CMOS-Technologie ergibt fast die gleiche benötigte Fläche für einen flexiblen Block aus drei SIRENS Elementen, wie die Synthese auf eine 130 nm Technologie. Dieses wird durch den erhöhten Flächenbedarf der Elemente der Standardzellenbibliothek für den Subschwellenbetrieb hervorgerufen. Die Elemente wurden zur Steigerung der Robustheit der Bibliothek nicht mit minimalen Maßen ausgelegt sondern vergrößert. Zudem enthält diese Bibliothek nur grundlegende Gatter und verzichtet auf Gatter mit mehr als 3 in Reihe geschalteten Transistoren. Die angegebene Verlustleistung wurde für eine Versorgungsspannung von 200 mV und einem Takt von 500 kHz ermittelt. Bei dieser Versorgungsspannung erreicht die Schaltung die vorgegebene Geschwindigkeit von 1 MHz nicht mehr.

In Abb. 4.16 ist die Verlustleistung des Neurons basierend auf der SIRENS-Struktur in Abhängigkeit von der Ausgangstaktrate des Neurons dargestellt. Es existieren zwei reproduzierbare Syntheseergebnisse für die dynamische Verlustleistung, bei denen die höhere Verlustleistung (Kreise) möglicherweise durch die Simulation des *Clock*-Netzwerkes als nicht ideales Netz zustande kommt und im Gegensatz dazu die niedrigere Verlustleistung (Quadrate) durch die Simulation des *Clock*-Netzwerkes als ideales Netzwerk (welches selbst keine Stromaufnahme hat). Auffällig ist der große Anteil der statischen Verlustleistung durch Leckströme, welche sich im Bereich von 10 μW bewegt und damit bei niedrigen

Pulsraten höher ist, als die Verlustleistung durch die Funktion der Schaltung selbst. Mit der Verlustleistung im Bereich von 5 μW bis 13 μW und der zusätzlichen Verlustleistung durch die Leckströme liegt die digitale Implementierung bei ähnlichen Eigenschaften für das Erreichen der Ausgangspulsrate mit in der Summe bis zu 25 μW Leistungsaufnahme etwa 37 mal höher als die vorgestellte analoge Variante des LIAF Neurons. Der Flächenbedarf des Neurons ist etwa 450 mal größer, als das in analoger Schaltungstechnik implementierte Gegenstück. Ein interessanter Punkt ist die Abschätzung der ersten Synthese ohne *Backannotation*-Schritt, welche die dynamische Verlustleistung mit 14 μW abgeschätzt hat. Dieses Ergebnis wird für eine hohe Ausgangspulsrate des Neurons mit den Backannotation-Daten fast erreicht und kann somit als Abschätzung nach oben dienen. Die Abbildung der bitparallel arbeitenden SIRENS-Struktur auf eine ASIC-Technologie ohne speziell dafür angefertigte Logikblöcke führt nach Tab. 4.5 zu einem relativ hohen Flächenbedarf. Aus diesem Grund wurden weitere Strukturen entworfen, welche einerseits auf FPGAs synthetisierbar, speziell aber auf die Synthese in einer ASIC-Technologie zugeschnitten sind. Für die Implementierung als ASIC wurde das Neuron in vollständig bitserieller Arbeitsweise umgesetzt.

ASIC optimierter Entwurf, bitseriell

Für den Entwurf von digitalen Systemen, welche auf einem ASIC integriert werden sollen, sind besondere Randbedingungen zu beachten. Da es im Entwurfsprozess durchaus üblich ist, die in einer Hardware-Beschreibungs-Sprache beschriebene Schaltung in einem FPGA-Baustein zu testen, ist darauf zu achten, bei der Beschreibung des Systems spezielle FPGA-Blöcke oder vom FPGA-Anbieter zur Verfügung gestellte Hardware-Beschleuniger nicht zu benutzen. Die Synthesewerkzeuge können die beschriebene Schaltung während der Synthese auf die angegebene Zieltechnologie durchaus so optimieren, dass sie auf spezielle Blöcke der FPGAs abgebildet werden. Werden diese Blöcke aber händisch in der Hardware-Beschreibungs-Sprache instanziert, so ist eine Synthese der Schaltung auf einen ASIC-Prozess oft schwierig, wenn die entsprechenden Makros dort vom jeweiligen Anbieter nicht vorliegen. Ebenso lässt sich aus der auf einem FPGA funktionierenden Schaltung nicht ableiten, dass die Schaltung auch auf einem ASIC fehlerfrei arbeitet, da jeder Technologie-Prozess eigene Besonderheiten aufweist, auf die bei der Synthese der Schaltung speziell eingegangen werden muss. Daher ist die Vorgehensweise bei der Implementierung einer Schaltung auf einem ASIC, dass alle Schaltungsteile unter Ausschluss von Makros wie z. B. mit Hilfe des XILINX CORE Generator oder des ALTERA MegaWizard aufgebaut werden.

Die im vorhergehenden Abschnitt beschriebene Architektur eines digitalen Neurons ist für die Umsetzung auf einen ASIC aufgrund der in jedem Block verwendeten parallelen Multiplizierer in Hinblick auf die benötigte Chipfläche ungeeignet, zumal die Multiplizierer für die Berechnung des Ergebnisses in einem Takt ausgelegt wurden, was zu einem sehr hohen Flächenbedarf führt (siehe Tabelle 4.5). Handentworfene Multipliziererblöcke bzw. optimierte IP-Cores, wie auch im FPGA vorhanden sind, verringern den Flächenaufwand

Tabelle 4.6: Flächenbedarf und dynamische Verlustleistung der SIRENS Struktur mit bitseriellen 16 Bit Multiplizierern bei Synthese auf 20 MHz Takt.

Technologie	Fläche [μm^2]	Verlustleistung [μW]		
		best	typical	worst
350 nm Technologie	174.668,8	2.422,2	2.333,8	2.264,2
130 nm Technologie B	18.140,8	122,8[a]	92,7[a]	63,6[a]
130 nm Technologie A	17.567,8	179,8[a]	131,2[a]	109,5[a]
	15.385,2[b]	174,6[b]	127,4[b]	106,7[b]
90 nm Sub-V_{TH}	9.914,6[b,c]		0,077[b,c]	

[a] Der zusätzliche Anteil der Verlustleistung durch Leckströme bewegt sich mindestens in der Größenordnung der hier angegebenen dynamischen Verlustleistung.

[b] Die Hierarchie der Implementierung wurde aufgelöst. Ungenutzte Elemente (z. B. konstante beschaltete Blöcke) wurden während der Synthese identifiziert und entfernt. Dieses Syntheseergebnis ist daher nur noch als LIAF Neuron einsetzbar.

[c] Maximale dynamische Verlustleistung nach Abbildung auf eine handentworfene Standardzellenbibliothek mit 200 mV Versorgungsspannung. Angaben zur verwendeten Bibliothek finden sich unter Kapitel 4.2.2. Das Sub-V_{TH} System wurde auf einen Takt von 1 MHz synthetisiert. Die statische Verlustleistung liegt in der Größenordnung der dynamischen Verlustleistung.

zwar, brauchen aber auf einem ASIC immer noch eine große Fläche. Das vorrangige Ziel ist daher, eine Umsetzung eines digitalen Neurons mit möglichst kleinem Flächenbedarf auf einem ASIC zu finden. Die Vorgehensweise beim Entwurf eines bitseriellen Multiplizierers wurde bereits in einem vorangegangenen Kapitel behandelt.

Nach der Identifikation des Multiplizierers als Element mit dem größten Flächenbedarf wurde das vorgestellte SIRENS Modell mit Hilfe von bitseriellen Multiplizierern aufgebaut. Dazu wurde nach der Analyse verschiedener bitseriell arbeitender Varianten ein optimierter Block aus den „DesignWare-Komponenten“ des Synopsys Design Compilers genutzt. Dieser lässt sich sowohl für die Synthese auf einem ASIC nutzen, als auch für die Synthese auf eine FPGA Plattform. Im Vergleich mit den theoretisch ermittelten Werten aus Abb. 4.10 unterschreitet die Fläche der bitseriellen Implementierung des LIAF Neurons aus SIRENS Blöcken erst ab einer Wortbreite von 9 Bit statt 6 Bit die Fläche der parallelen Implementierung (siehe auch Abb. 4.20). Dieses wird durch den zu Grunde liegenden Addierer hervorgerufen, der vor allem auf eine hohe Verarbeitungsgeschwindigkeit ausgelegt ist. Zusätzlich erhöhen die mit Registern versehenen Ein- und Ausgänge des bitseriellen Multiplizierers im Vergleich mit den theoretischen Ergebnissen den Flächenbedarf. Die gesamte Schaltung mit bitseriellem Multiplizierer der Wortbreite n muss zum Erreichen der gleichen Übertragungsfunktion wie die parallele Implementierung mit einem Takt von $(n+4)$ MHz betrieben werden. Für die Verlustleistung wurde im Fall $n = 16$ aus der Synthese eine Verlustleistung von ca. 130 μW ermittelt. Die Zunahme der Verlustleistung bei der Verwendung der bitseriellen DesignWare Komponenten ist durch den notwendigen höheren Takt von 20 MHz zu begründen. Aus Tab. 4.6 wird ersichtlich, dass die Verlustleistung letztlich von der verwendeten Zieltechnologie bzw. dem Technologieanbieter abhängig ist.

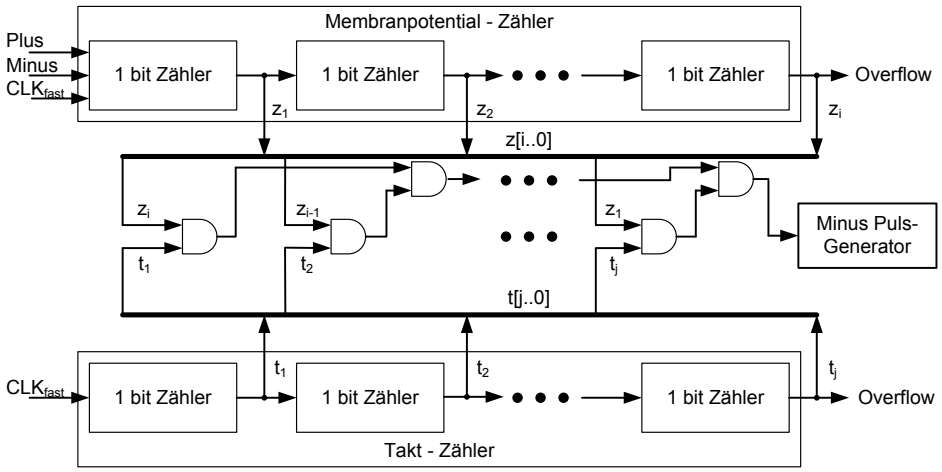


Abbildung 4.17: Blockschaltbild eines auf Zählern basierenden LIAF Neuronenmodells.

Die für die SIRENS Struktur benötigte Fläche sinkt durch den Einsatz bitserieller Multiplizierer um mehr als 30%. Gleichzeitig wird ersichtlich, dass die Auflösung der drei SIRENS Funktionseinheiten zu einem gemeinsamen LIAF Neuron unter Verlust der Flexibilität nur noch eine Flächenabnahme von 12,5% bringt, während sich die Verlustleistung kaum verändert. Im Vergleich mit der bitparallelen Implementierung muss die bitserielle Struktur zum Erreichen der gleichen Ausgangscharakteristik um den Faktor 20 schneller getaktet werden. Dabei steigt die Verlustleistung nur um den Faktor 8 an. Die Synthese auf die Standardzellenbibliothek für den Subschwellenbereich mit einer Versorgungsspannung von 200 mV konnte an dieser Stelle nur noch auf einen Takt von 1 MHz statt 20 MHz durchgeführt werden. Dadurch ergibt sich eine deutlich kleinere Fläche und eine geringe Verlustleistung. Wird der für die Synthese gewählte Takt maximiert, erhöht sich die Fläche auf ungefähr den Wert der Implementierungen in der 130 nm CMOS-Technologie, da fast ausschließlich Zellen mit hoher Treiberleistung und damit großer Fläche genutzt werden.

ASIC optimierter Entwurf, zählerbasiert

Aus der beobachteten hohen Energieaufnahme des Systems mit bitseriellen Multiplizierern wird als zweite Möglichkeit für einen auf einen ASIC optimierten Entwurf eine auf Zählern basierende Umsetzung gewählt, die im Folgenden detailliert beschrieben wird. Trotz der möglichen Flächeneinsparung durch die Nutzung von bitseriellen Multiplizierern (siehe Tab. 4.3) wurde versucht, in diesem Ansatz vollständig auf Multiplikationen zu verzichten und die phänomenologisch beobachtbaren Eigenschaften eines LIAF Neurons über andere Mechanismen zu emulieren. Grundlage des digitalen Neurons ist ein Block, der im nachfolgenden Text als Soma (Zellkörper) bezeichnet wird. Dieser Block emuliert die Integration von Pulsen auf der Zellmembran sowie die Abnahme des Membranpotentials über der Zeit. Die in Abb. 4.17 dargestellte Soma [109] besteht aus einem synchronen n Bit Aufwärts- und Abwärts-Zähler für das Membranpotential, der in jedem Takt durch ein gesetztes Bit am *Plus* Eingang um ein Bit erhöht oder durch ein gesetztes Bit am *Minus* Eingang um

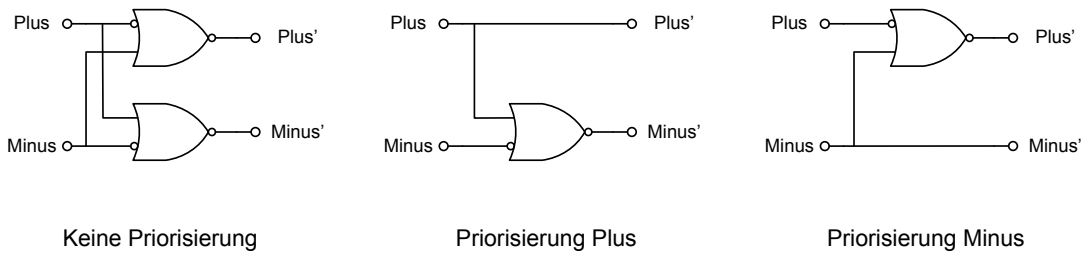


Abbildung 4.18: Schaltungen zur Priorisierung eines Eingangs.

ein Bit vermindert werden kann. Liegt gleichzeitig an beiden Eingängen ein gesetztes Bit an, heben sich das exzitatorische und das inhibitorische Signal auf und der Zählerstand bleibt unverändert. Eine alternative Implementierung, bei der das exzitatorische oder das inhibitorische Signal priorisiert wird, ist ebenfalls denkbar. Die für beide Möglichkeiten der Priorisierung verwendeten Schaltungen (Abb. 4.18) benötigen den gleichen Flächenbedarf. Die an den *Plus*- und *Minus*-Eingängen verarbeiteten Informationen werden mit einem schnellen Takt *CLKfast* verarbeitet. Hier laufen Informationen über ein Aktionspotential von vier Synapsen im Zeitmultiplexverfahren ein. Die Pulsrate in jedem einer Synapse zugeteilten Zeitschlitz entspricht einem durch die Synapse gewichteten Aktionspotential. Der Ausgang des höchstwertigen Flip-Flops des Zählers kann zur Detektion der Überschreitung der Feuerschwelle genutzt werden, wenn der höchste Wert, den der Zähler annehmen kann, als Feuerschwelle angenommen wird. Daneben kann an jedem Bit des Zählers abgegriffen werden oder – unter Nutzung zusätzlicher Fläche – ein Komparator zum Vergleich des Zählerstandes mit einem gespeicherten Wert implementiert werden. Für den Zerfall des Membranpotentials sorgt ein weiterer n Bit Zähler. Dieser Takt-Zähler genannte Schaltungsteil ist ein Abwärts-Zähler, der beginnend mit dem Wert der Feuerschwelle mit jedem Takt um eins vermindert wird. Der dargestellte Komparator aus UND-Gattern vergleicht die Zählerstände beider Zähler bitweise miteinander und erzeugt bei Gleichheit beider Zähler ein Steuersignal für einen Schaltungsteil, der inhibitorische Pulse erzeugt, welche auf den *Minus*-Eingang des Membranpotential-Zählers zurückgeführt werden. Diese Implementierung führt zu einer hohen Rate von im Neuron erzeugten inhibitorischen Pulsen, wenn sich das Membranpotential in der Nähe der Feuerschwelle befindet. Je weiter das Membranpotential unterhalb der Feuerschwelle liegt, umso weniger inhibitorische Pulse werden erzeugt. Für das Membranpotential bedeutet dieses einen exponentiellen Zerfall mit der Zeit, wenn das Neuron nicht erregt wird.

Das Gesamtsystem arbeitet mit Pulsraten zur Beschreibung der Signalstärke eines Aktionspotentials. Dazu werden einzelne vom Neuron ausgesandte Aktionspotentiale durch eine Synapse in einen mit dem Wert der Synapse korrespondierenden Ratencode umgewandelt. Der Ratencode wird in einem der Synapse zugeteilten Zeitschlitz an das Neuron übertragen. Das hier beschriebene Neuron erhöht oder vermindert mit jedem ihn erreichenden Puls den Zähler für das Membranpotential und sendet bei Erreichen der Feuerschwelle einen einzelnen Puls der Länge eines Taktes aus. Anschließend wird der Membranzähler auf einen Initialwert zurückgesetzt.

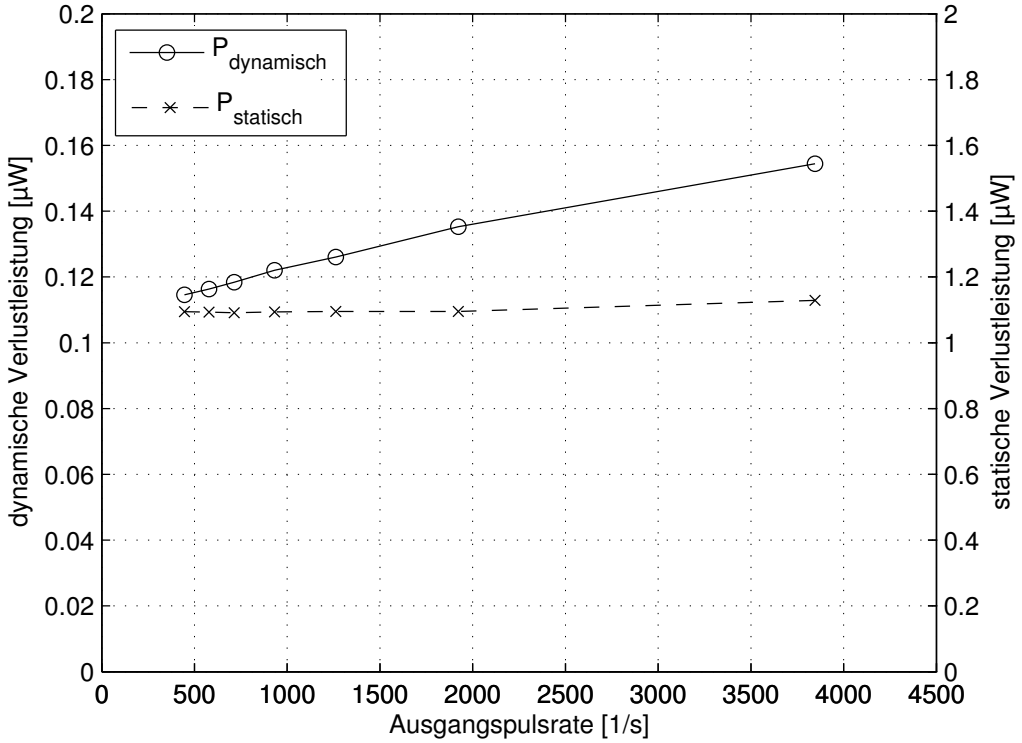


Abbildung 4.19: Dynamische Verlustleistung (durchgezogene Linie) und Verlustleistung durch statische Leckströme (gestrichelte Linie) eines bitseriellen, zählerbasierten Neurons mit 6 Bit Auflösung und 250 kHz Takt in einem 130 nm CMOS-Prozess.

In Tabelle 4.7 sind die Syntheseergebnisse für den Flächenbedarf und die zu erwartende Verlustleistung der hier mit einem Takt von 250 kHz beschriebenen Schaltung angegeben. Das synthetisierte bitserielle Neuron benötigt in der 6 Bit Variante (6 Bit ist die äquivalente Auflösung des analog implementierten Neurons) eine um den Faktor 4 kleinere Fläche in der verwendeten 130 nm CMOS-Technologie, als eine bitparallele Implementierung mit 6 Bit Wortbreite. Gleichzeitig konnte die Verlustleistung gesenkt werden, was vor allem durch den niedrigen Takt des Systems ermöglicht wird. Die Verlustleistung über der Ausgangspulsrate ist in Abb. 4.19 dargestellt und zeigt einen nahezu konstanten statischen Anteil der Verlustleistung von 1,15 μW, während die dynamische Verlustleistung ein Maximum von 154 nW bei einer Ausgangspulsrate von etwa 3850 Pulsen/s erreicht. Umgerechnet ergibt sich so eine Energie von 40 pJ pro ausgesandtem Aktionspotential.

Bereits im vorangegangenen Abschnitt konnte gezeigt werden, dass der Flächenbedarf eines Neurons durch den Einsatz eines bitseriellen Arbeitsprinzips stark reduziert werden kann. Dieser Flächengewinn wird dabei jedoch durch den Betrieb mit einem schnelleren Takt als dem der bitparallelen Implementierung durch eine auf die belegte Fläche bezogene, hohe dynamische Verlustleistung erkauft. In der hier vorgestellten zählerbasierten Variante können die Fläche und die Verlustleistung weiter reduziert werden. Es ergibt sich eine Verlustleistung im Bereich von 100 nW bei einer Ausgangspulsrate von bis zu 4000 Pulsen/s. Das Verhältnis von erreichter Ausgangspulsrate und Verlustleistung kommt dem Verhältnis

Tabelle 4.7: Flächenbedarf und dynamische Verlustleistung der zählerbasierten bitseriellen Struktur mit 6 Bit Auflösung bei 250 kHz Takt.

Technologie	Fläche [μm^2]	Verlustleistung [nW]		
		best	typical	worst
350 nm Technologie	15.002,6	2.478,3[a]	2.479,6[a]	2.580,7[a]
130 nm Technologie B	1.429,0	132,8[b]	99,7[b]	69,5[b]
130 nm Technologie A	1.397,9	152,9[b]	111,9[b]	93,5[b]
90 nm Sub-V_{TH}	1.564,1		2,5[c]	

[a] Die zusätzliche statische Verlustleistung durch Leckströme liegt mit ca. 10 nW mehrere Größenordnungen unter der dynamischen Verlustleistung.

[b] Die zusätzliche statische Verlustleistung durch Leckströme liegt mit 1–2 uW um eine Größenordnung über der dynamischen Verlustleistung. Die Synthese wurde auf eine geringe statische und dynamische Verlustleistung hin optimiert.

[c] Maximale dynamische Verlustleistung nach Abbildung auf eine handentworfene Standardzellenbibliothek mit 200 mV Versorgungsspannung. Angaben zur verwendeten Bibliothek finden sich unter Kapitel 4.2.2. Die statische Verlustleistung liegt mit ca. 32 nW um eine Größenordnung über der dynamischen Verlustleistung.

der bitparallelen SIRENS Implementierung sehr nahe und lässt den Schluss zu, dass die zählerbasierte Implementierung eine sehr gute Umsetzung ist, wenn besonders wenig Fläche genutzt werden soll. Die dynamische Verlustleistung der bitseriellen zählerbasierten Variante beträgt im Fall der Umsetzung in einer 130 nm Technologie etwa den Wert der analogen Variante des LIAF Neurons in gleicher Technologie, dagegen ist die statische Verlustleistung mit 1 μW bis 2 μW fast doppelt so hoch wie die statische Verlustleistung der analogen Implementierung mit 650 nW. Durch die höhere Ausgangspulsrate des analogen Neurons ergibt sich eine bessere Energiebilanz für die analoge Implementierung. Die Fläche des bitseriellen Neurons belegt dabei etwa das 15-fache der Fläche des analog implementierten Neurons.

Eine weitere Absenkung des Energiebedarfs des zählerbasierten Neurons kann durch den Einsatz spezialisierter Standardzellenbibliotheken, wie der später noch vorgestellten Bibliothek in Subschwellen-Schaltungstechnik mit 200 mV Versorgungsspannung erreicht werden. Diese Bibliothek beinhaltet Standardzellen, die besonders auf Robustheit gegenüber Parameter-Variation im Herstellungsprozess ausgelegt sind und eine größere Fläche belegen, als vergleichbare Standardzellen, die mit höheren Versorgungsspannungen arbeiten. Obwohl die Subschwellenbibliothek eine 90 nm CMOS-Technologie nutzt, ergibt sich aus der Optimierung auf hohe Robustheit der in Tab. 4.7 angegebene leicht höhere Flächenbedarf im Vergleich mit der bisher verwendeten 130 nm CMOS-Technologie. Durch den Einsatz der Subschwellen-Bibliothek kann in der vorliegenden Implementierung die dynamische Verlustleistung im Vergleich mit der Synthese auf eine 130 nm Standardzellenbibliothek um den Faktor 32 verringert werden, während die statische Verlustleistung um den Faktor 34 abnimmt.

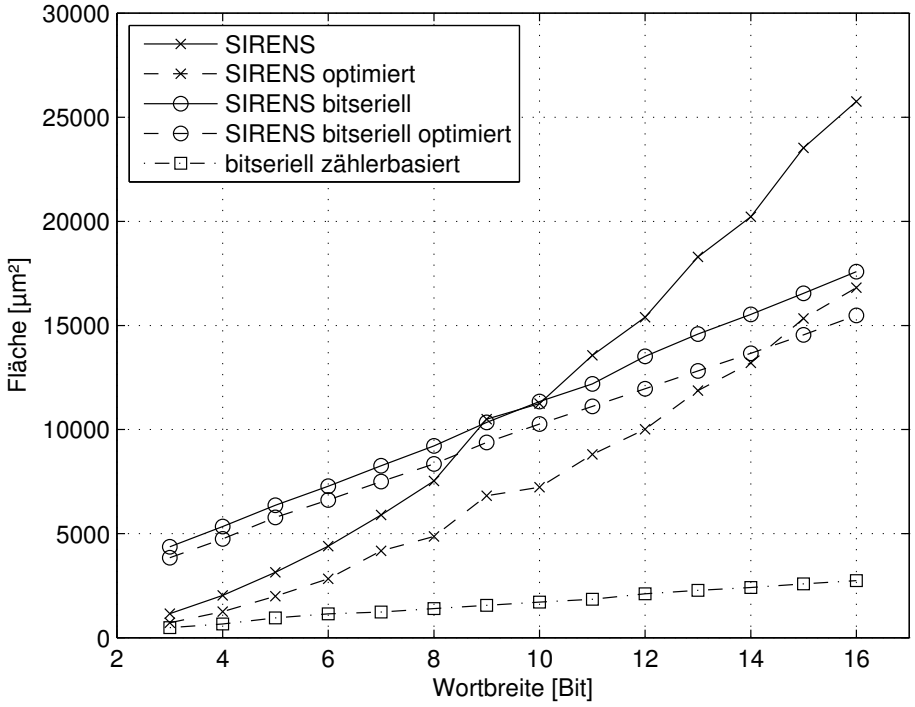


Abbildung 4.20: Flächenbedarf digitaler Implementierungsvarianten über der Wortbreite.

4.2.4 Vergleich digitaler Implementierungsvarianten

Der Einsatz von digitalen Umsetzungen pulsender Neurone hängt im Wesentlichen von den Anforderungen an die Verarbeitungsgeschwindigkeit und die belegte Fläche ab. Während die bitparallele Umsetzung des LIAF Neurons in Form der SIRENS Struktur Pulsraten im Bereich bis 1 MHz erlaubt, wird durch die bitparallele Implementierung der Multiplizierer auf einen ASIC dabei eine große Fläche belegt. Die Umsetzung ist, wenn eine FPGA-Plattform eingebettete Multiplizierer bietet, vor allem für die Umsetzung auf ein FPGA geeignet. Der Wechsel zu einer bitseriellen Arbeitsweise kann die benötigte Fläche besonders für hohe Wortbreiten reduzieren, erhöht aber in gleichem Maße die Latenz jeder einzelnen Stufe der SIRENS Struktur. Zum Erhalt der gleichen Übertragungsfunktion muss das bitserielle System daher mit deutlich höherem Takt betrieben werden, was zu einer Erhöhung der Verlustleistung führt. Der experimentell ermittelten Erhöhung der Verlustleistung um den Faktor 10 steht eine Flächenabnahme von etwa 1/3 gegenüber. In Abb. 4.20 ist der Flächenbedarf der einzelnen Varianten des LIAF Neurons in einer 130 nm CMOS-Technologie dargestellt. Die bitseriellen Varianten sind durch die charakteristische lineare Zunahme der Fläche mit der Wortbreite zu sehen. Diese schneiden die quadratisch wachsende Fläche der bitparallel umgesetzten Varianten frühestens ab einer Wortbreite von etwa 9 Bit. Im Abschnitt über die bitparallele SIRENS Struktur wurde jedoch deutlich, dass für das gewählte Neuronenmodell eine Umsetzung mit 16 Bit erforderlich ist. Bei den als optimierte Implementierungen bezeichneten Umsetzungen

wurde die flexible SIRENS Struktur durch das Synthese-Werkzeug aufgelöst und zusammengefasst. Es ergibt sich daher ein geringerer Flächenbedarf aufgrund der Entnahme ungenutzter Elemente der SIRENS Struktur. Diese Einheiten aus drei SIRENS-Blöcken können nur noch für die Funktion eines LIAF Neurons genutzt werden und haben ihre Flexibilität verloren. Im Gegensatz zu den SIRENS Varianten erlaubt eine auf Zählern basierende Umsetzung die flächenminimale Implementierung eines LIAF Neurons in digitaler Schaltungstechnik. Neben der linearen Skalierung der Fläche mit der Wortbreite bietet diese Umsetzung ein ähnliches Verhältnis zwischen Verlustleistung und Ausgangspulsrate wie die bitparallele SIRENS Struktur. Für eine Implementierung von LIAF Neuronen in digitaler Schaltungstechnik ist wegen der kleinen belegten Fläche und der im Vergleich mit der bitseriellen Umsetzung relativ geringen Verlustleistung die zählerbasierte Schaltung besonders geeignet.

4.3 Analoger Testchip

Zur Verifikation der in den vorigen Abschnitten dieses Kapitels erarbeiteten Eigenschaften von in analoger Schaltungstechnik integrierten pulsenden Neuronen wurde ein Testchip in einer 130 nm CMOS-Technologie entworfen, der die in Kapitel 4.1 entworfenen LIAF Neurone, deren Layout in Abbildung 4.2 dargestellt ist, enthält. Insbesondere die Frage, welches Verhalten diese Neurone mit Minimal-Layout nach der Fertigung aufweisen, ist nur durch die Messung an gefertigten Chips zu klären, da diese allen Prozess-Schwankungen während der Herstellung unterworfen sind. Diese Vorgehensweise soll Aufschluss über notwendige Modifikationen der vorgeschlagenen Implementierungsvariante geben. Der in dem implementierten Neuron genutzte Strombereich ist mit Strömen von wenigen Nanoampere schwer zu handhaben, da die Ströme in diesem Bereich stark von der Umgebungstemperatur – auch der lokalen Umgebungstemperatur anderer Elemente auf dem Chip – beeinflusst werden können.

Um den Messbereich der aufgenommenen Leistung sowie den in die Schaltung zu injizierenden Strom auf handhabbare Werte zu erhöhen, wurden viele Kopien des gleichen Neurons auf dem Testchip vorgesehen, welche parallel betrieben werden. In vier identischen Arealen wurden jeweils 260 LIAF Neurone in der Weise verschaltet, dass sie zur gleichen Zeit mit dem gleichen Eingangsstrom getrieben werden. Das Betreiben der Neurone eines Areals mit dem gleichen Strom sollte bei enger Nachbarschaft, auf die im Areal ausdrücklich hin gearbeitet wurde, zum Auslösen der gleichen Ausgangspulsrate in allen Neuronen führen. Dadurch sind Rückschlüsse auf das Verhalten und die Leistungsaufnahme eines einzelnen Neurons des Areals möglich, wenn man voraussetzt, dass alle Neurone bei gleichem Stimulus dasselbe Verhalten aufweisen. Neben der Erhöhung der Mess- und Betriebsbereiche kommt die homogene Struktur der Neuronen-Felder der Robustheit der Schaltung entgegen, indem durch die durch das Layout erzwungene homogene Struktur Einflüsse auf die Schaltung während der Herstellung (z. B. kleinste Geometrieänderungen) gleichmäßig auf das homogene Feld verteilt und so minimiert werden.

Tabelle 4.8: Versorgungs- und Referenzspannungen des Testchips

Spannung	Wert
VDD1-4	1,2 V
VDDamp	1,2 V
VDDring	1,2 V
VREF1U	0,351 V
Vleak	0,0 V
Vdecay	0,240 V
VTH	0,730 V

Zur Messung des Membranpotentials und der ausgelösten Aktionspotentiale wurden aus jeweils einem Areal aus 260 Neuronen der Ausgang eines einzelnen Neurons im Randbereich des Feldes beobachtbar gemacht und nach außen auf einen Anschluss-Pin geführt. Die zusätzliche Lastkapazität von ca. 100 pF, die durch den Anschluss-Pin am Ausgang des Neurons eingebracht wurde, und welche zusätzlich vom Neuron umgeladen werden muss, trägt nur unwesentlich zur erwarteten Leistungsaufnahme des gesamten Neuronen-Feldes von 176 μW bei und kann mit einem Anteil von wenigen Nanowatt vernachlässigt werden. Neben der Beobachtung des Aktionspotentials kann an zwei Arealen (Areal 1 und Areal 3) auch das Membranpotential während des Betriebs beobachtet werden. Da die Lastkapazität der analogen Pad-Zellen im Vergleich zur als Membrankapazität genutzten MOS-Kapazität von 100 fF erheblich größer ist, wurde an diesen Arealen zusätzlich ein Verstärker als Impedanzwandler vorgesehen, da die Funktion des Neurons durch die veränderte Kapazität stark verändert oder zerstört worden wäre. Mit Hilfe der anhand von Simulationen am RC-extrahierten Layout des Verstärkers ermittelten Übertragungskennline des implementierten invertierenden Verstärkers kann das Membranpotential aus dem gemessenen analogen Spannungspegel der Pad-Zelle rekonstruiert werden.

Die Stromversorgung der Verstärker wird über separate Pad-Zellen für die Versorgungsspannung gewährleistet, um die Leistungsaufnahme der einzelnen Areale messen zu können. Die aus Simulationen entnommenen Versorgungs- und Referenzspannungen sind in Tab. 4.8 aufgeführt. Für die separate Versorgung des Pad-Rings und der Verstärker sind die Versorgungsspannungen VDDring und VDDpad separat herausgeführt. Die Spannungsversorgung der einzelnen Areale wird über VDD1 bis VDD4 vorgenommen. Die Spannung VREF1U ist eine Referenzspannung für den Komparator der Neurone. Alle Neurone werden global parametriert, indem die Stärke des Abklingterms über Vleak, die Dauer eines Aktionspotentials über Vdecay und die Feuerschwelle mit VTH eingestellt werden. Mit den in der Tabelle angegebenen Parametern wirkt nur der Subschwellenstrom der 130 nm Technologie als passiver Abklingterm, die Dauer eines Aktionspotentials wurde auf 1 ms festgelegt. In Abbildung 4.21 ist das zur Fertigung gegebene Chiplayout des Testchips mit den beschriebenen vier Arealen und dem Pad-Zellen-Ring zu sehen. Für das Layout des Chips wurde die minimale zu fertigende Fläche von 1 μm^2 vollständig ausgenutzt.

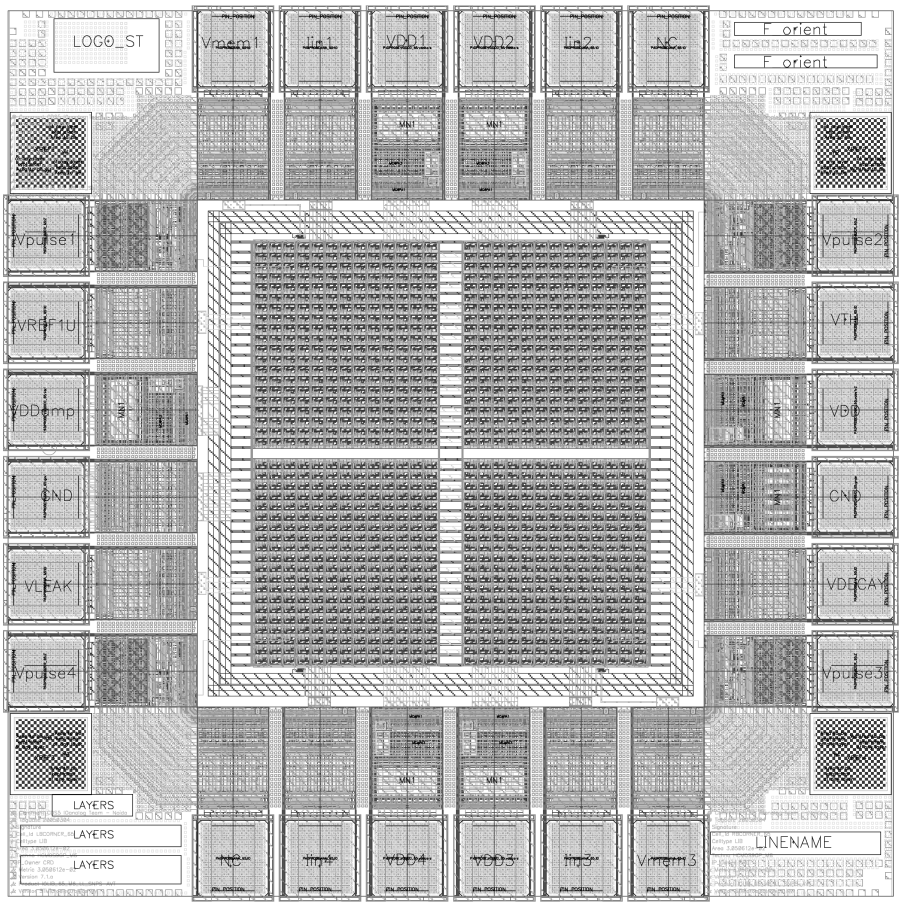


Abbildung 4.21: Layout eines Testchips mit 1040 LIAF Neuronen in einer 130 nm Technologie. Erkennbar sind vier separate ansteuerbare Felder mit je 260 Neuronen des vorgestellten LIAF-Modells und der Anschluss-Pad-Ring im äußeren Bereich.

Im Folgenden sind die gefertigten Areale mit ihren Nummern benannt. Das erste Areal befindet sich oben links in der Abbildung, die weiteren Areale werden im Uhrzeigersinn nummeriert. Zusätzlich zu den Pad-Zellen für die Versorgungsspannung und die Masse des Pad-Zellen-Rings selbst sind die Versorgungs-Pad-Zellen der Verstärker sowie die Pad-Zellen für die Referenzspannungen zu sehen. Daneben sind für jedes Areal eine unabhängige, separate Spannungsversorgung und ein Eingang für den zu injizierenden Strom vorgesehen. Alle vier Areale besitzen einen eigenen digitalen Ausgang, an dem die Erzeugung eines Aktionspotentials beobachtet werden kann. Zusätzlich besitzen Areal 1 und Areal 3 eine analoge Ausgangs-Pad-Zelle für die Messung des Membranpotential-Verlaufs.

Mit der Inbetriebnahme der Testchips wird geprüft, welche der gefertigten Chips eine Funktion aufweisen. Dabei sollen mögliche schwerwiegende Layout- oder Herstellungsfehler erkannt und die funktionierenden Chips identifiziert werden. Mit Anlegen der in den Simulationen bestimmten Referenzspannungen werden die Chips auf die Erzeugung von Aktionspotentialen getestet. Dazu werden im ersten Schritt die Schwellenspannung VTH und die Referenzspannung für den Leaky-Term Vleak niedrig gehalten und ein Strom

in die Schaltung injiziert. Da der Leaky-Term klein und somit der Abfluss der Ladung von der Membrankapazität gering ist, ist zu erwarten, dass ein funktionierendes Neuron nach kurzer Zeit feuert, wenn der injizierte Strom größer ist als in der verwendeten Halbleitertechnologie auftretende Subschwellenströme. Durch diese Vorgehensweise werden die aktiven Aktionspotential-Ausgänge und die funktionierenden Areale bestimmt. Anschließend werden alle Referenzspannungen an die aus Simulationen ermittelten Werte angeglichen und die Übertragungsfunktionen der Neurone durch automatisierte Mess-Schritte ermittelt. Bei den durchgeführten Messungen konnte festgestellt werden, dass die Ausgangspulsrate der Areale der einzelnen Chips größeren Schwankungen von Testchip zu Testchip unterworfen war. Weiter konnte ermittelt werden, dass die vier Areale eines Chips untereinander leicht abweichende Ausgangspulsraten aufwiesen und dass Areal 1 und Areal 4 nahezu identische Messergebnisse liefern. Von diesen weichen Areal 2 und Areal 3 leicht ab, sind in ihrem Verhalten zueinander aber wieder ähnlich. Die starke Streuung der Ausgangspulsraten der einzelnen Chips im Bereich von 90.000 Pulsen/s bis 160.000 Pulsen/s am unteren Messbereich ist auf die Produktion auf einem Multi-Project-Wafer (MPW) zurückzuführen, bei der mehrere verschiedene Layouts unterschiedlicher Entwickler auf einem Wafer gleichzeitig gefertigt werden. Dieses Vorgehen führt zu inhomogenen Strukturen auf dem MPW, welche für die einzelnen Bearbeitungsschritte des Chips zu größeren Streuungen führen können. Zusätzlich kann es durch die Anordnung der Layouts auf einem MPW zu relativ großen Abständen zwischen den Layouts der gleichen Chips auf einem Wafer kommen. Diese Bedingungen erklären die Schwankungen der gemessenen Ausgangspulsraten, die von Chip zu Chip festgestellt werden konnten, während die vier auf einem Chip angeordneten Areale aufgrund ihrer räumlichen Nähe ähnliche Ausgangspulsraten liefern. An dieser Stelle machen sich zusätzlich die kleinen Strukturgrößen stark bemerkbar, insbesondere die Größe des Minimal-Neuronen-Layouts, welche zu großen Streuungen bei der Fertigung der Chips führen. Der Hersteller kann bei den MPW Projekten keine Aussage über die Lage der einzelnen Chips machen, was für eine Beurteilung des Einflusses der Herstellung und der Lage des Layouts auf die Funktion eines einzelnen Chips notwendig gewesen wäre. Als bester Chip der 15 gefertigten Muster wurde Chip Nr. 5 identifiziert, auf den sich die folgenden Betrachtungen beziehen.

Die Funktion des implementierten LIAF-Neurons konnte mit der Messung der ausgesandten Aktionspotentiale sowie der Aufnahme des Membranpotentials gezeigt werden (siehe Abb. 4.22 der Messergebnisse von Chip Nr. 5). In der Abbildung ist das Aussenden eines Aktionspotentials zum Zeitpunkt $2,5\,\mu$s zu sehen (Low-Pegel des digitalen Ausgangs), welches durch das Membranpotential, das in diesem Moment die Feuerschwelle überschreitet, ausgelöst wird. Am Verlauf des Membranpotentials wird deutlich, dass die erwartete Hysterese über die eingebrachte Koppelkapazität funktioniert, da eine Spannungsüberhöhung beim Aussenden des Aktionspotentials auf dem Membranpotential stattfindet. Während des Aussendens eines Aktionspotentials wird die Membrankapazität entladen und die Koppelkapazität sorgt für eine weitere Absenkung des Membranpotentials beim Zurückschalten des Ausgangsinverters des Neurons, der das ausgesandte Aktionspotential anzeigt.

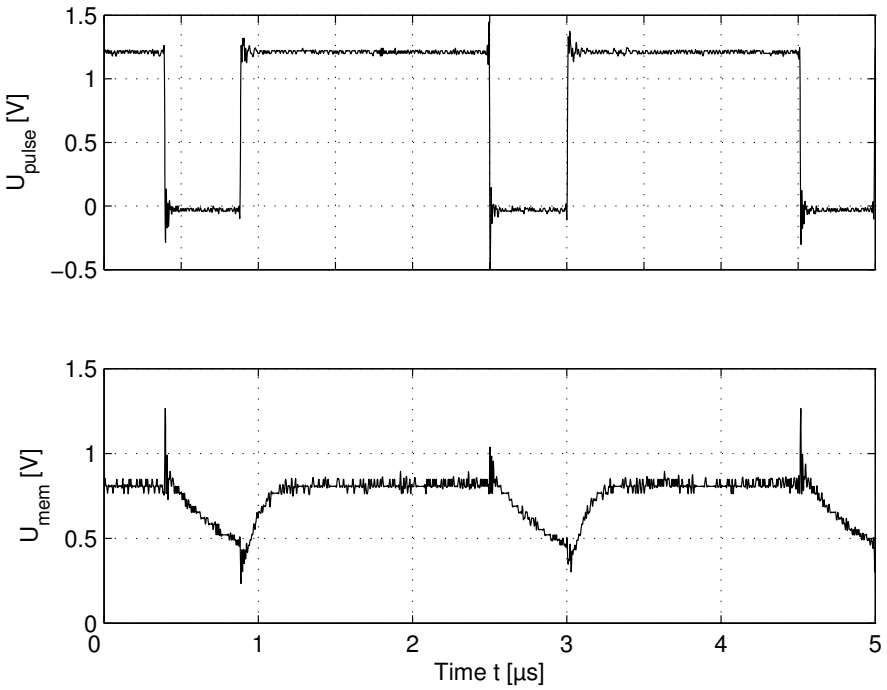


Abbildung 4.22: Gemessene Aussendung eines Aktionspotentials an U_{pulse} (negiertes Signal) und rekonstruiertes, korrespondierendes Membranpotential U_{mem} für einen injizierten Strom von $I_{\text{in}} = 4/260\,\mu\text{A}$ ($U_{\text{TH}} = 730\,\text{mV}$, $U_{\text{leak}} = 0\,\text{V}$).

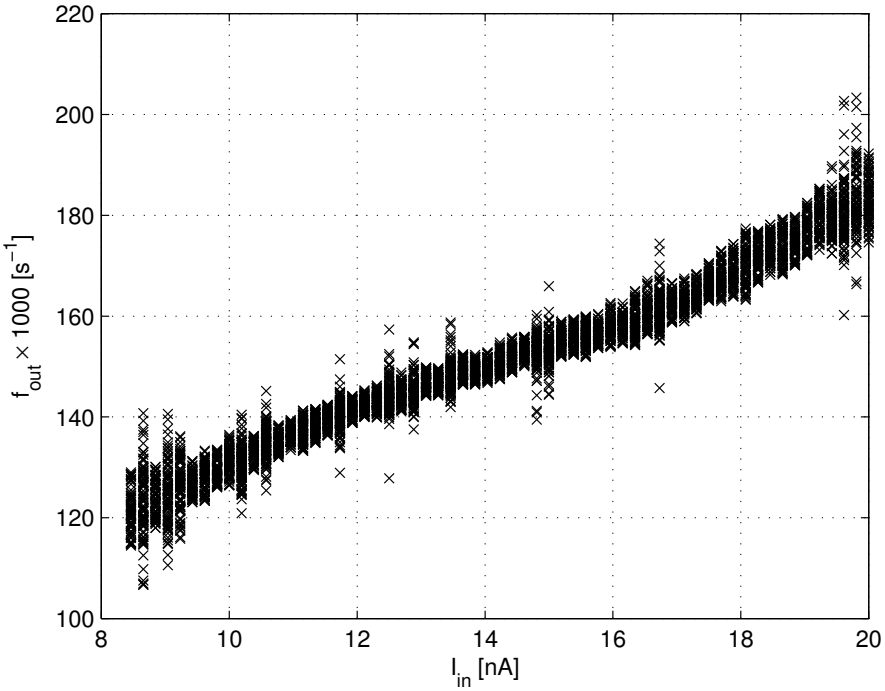


Abbildung 4.23: Übertragungskennlinie eines HW LIAF Neurons.

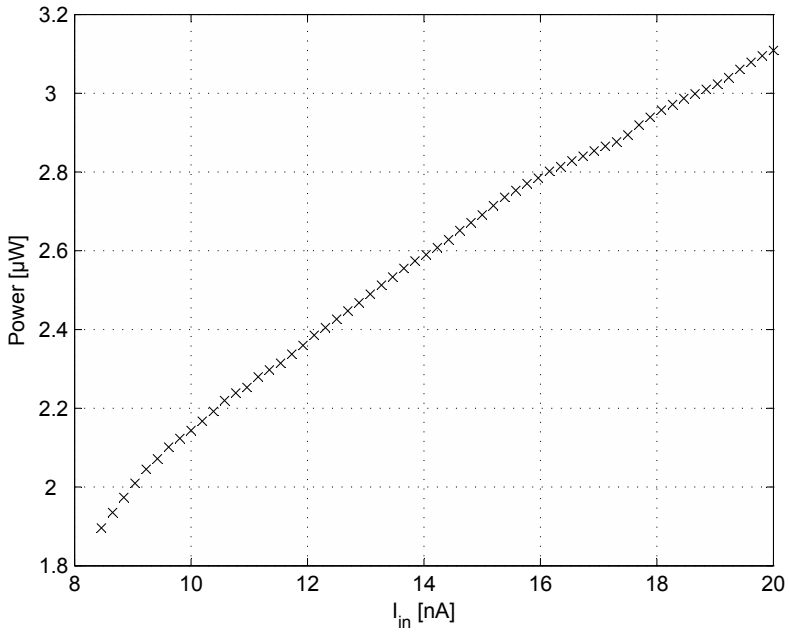


Abbildung 4.24: Leistungsaufnahme eines LIAF Neurons.

Der Komparator zeigte auch bei dem in den Messungen festgestellten Rauschen des Membranpotentials eine robuste Funktion, ist aber empfindlich gegenüber Änderungen an der Referenzspannung seiner Stromquelle, so dass für eine Erhöhung der Robustheit dieses Elements über alternative Vergleichselemente, z. B. den Einsatz eines Schmitt-Triggers, nachgedacht werden muss. In Abb. 4.23 ist die ermittelte Übertragungsfunktion eines Neurons aus Areal 1 des Testchips Nr. 5 dargestellt, bei der die erzeugte Ausgangspulsrate über dem injizierten erregenden Strom aufgetragen ist. Der injizierte Strom kann ebenfalls als mittlerer Strom, der durch eine angelegte konstante Eingangspulsrate erzeugt wird, aufgefasst werden. Dieses erlaubt den Vergleich der Messung mit den theoretischen Ergebnissen für die Übertragungsfunktion des LIAF Neurons aus Kapitel 3.3. In der Abbildung sind zu jedem injizierten erregenden Strom die Ergebnisse aus jeweils 100 Einzelmessungen aufgetragen. Die sichtbare Streuung der Messergebnisse lässt sich durch verschiedene Einflüsse während der Messung erklären. Zum Einen ist die Pulsrate der gemessenen Neurone nicht konstant sondern ist einer leichten zeitlichen Variation unterworfen, so dass zu unterschiedlichen Zeitpunkten leicht unterschiedliche Periodendauern ermittelt wurden. Zum Anderen wurde während der Messung die Umgebungstemperatur nicht konstant gehalten, so dass dieser Einfluss direkte Auswirkungen auf das Messergebnis hatte. Ein weiterer Einflussfaktor ist die Verwendung von Konstantspannungsquellen, deren Ausgangsspannungen leichten, aber messbaren Schwankungen unterworfen waren.

Neben der Ausgangspulsrate wurde über die Konstantstromquelle während der Messung der von den Neuronen eines Areals aufgenommene Strom integriert und daraus die mittlere aufgenommene Leistung ermittelt. Aus den aufgenommenen Werten kann der zum Vergleich des Neurons mit anderen Implementierungen oft gebrauchte Graph der

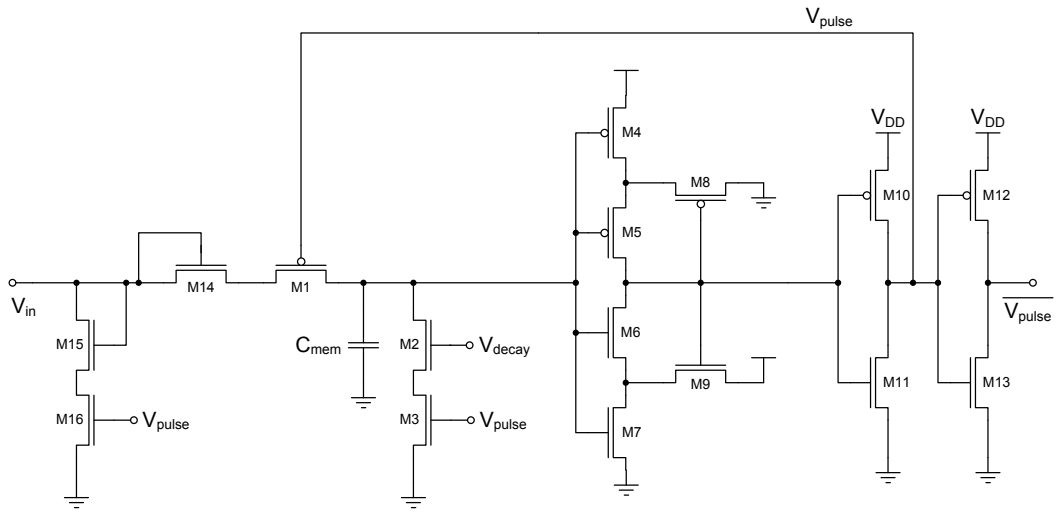


Abbildung 4.25: Auf eine robuste Funktion hin modifiziertes Neuron in 130 nm CMOS-Technologie.

Leistungsaufnahme über dem injizierten Strom aufgetragen werden. In Abb. 4.24 ist die von einem Neuron in Areal 1 des Chips Nr. 5 aufgenommene Leistung dargestellt. Diese liegt mit einem Bereich von $2\,\mu$W bis $3\,\mu$W, deutlich über der in der Simulation ermittelten Leistungsaufnahme von 650 nW bis 750 nW, zeigt aber den für LIAF Neurone typischen abflachenden Verlauf bei größeren erregenden Strömen.

Während der Messungen fiel auf, dass mit zunehmender Stärke des Eingangsstroms die Dauer des Aktionspotentials zunahm. Dieser Effekt kann darauf zurückgeführt werden, dass Transistor M1 (siehe Abb. 4.1) während des Aussendens eines Aktionspotentials nicht ideal sperrt. Durch die Beschaltung des Eingangs mit einer Konstantstromquelle wird trotz gesperrtem Transistor M1 weiter Strom auf die Membrankapazität aufgebracht, so dass die Entladedauer der Kapazität über die Transistoren M2 und M3 zunimmt. Der Einfluss des injizierten Stroms muss bei einem neuen Entwurf des Neurons bedacht werden und es müssen schaltungstechnische Maßnahmen getroffen werden, um den injizierten Strom bereits vor M1 abzuleiten. Eine mögliche Variante des beschriebenen Neurons, welches alle zuvor angeregten Änderungen einbezieht, ist in Abb. 4.25 dargestellt. Es enthält einen Schmitt-Trigger als Schwellenelement, wobei in der vorliegenden Form keine Einstellung der Feuerschwelle mehr möglich ist. Zusätzlich wurden die Transistoren M14 bis M16 hinzugefügt. Transistor M14 implementiert eine Diode, die als optional anzusehen ist. Durch M14 wird der Abfluss von Ladung von der Membrankapazität über den zusätzlichen Pfad durch M15 und M16 verringert, wenn M1 nicht ideal sperrt. Während des Aussendens eines Aktionspotentials kann von der Quelle injizierter Strom über den Pfad M15 und M16 abfließen, so dass kein zusätzlicher Strom während der Entladephase der Membrankapazität von außen auf die Kapazität aufgebracht wird und die Entladezeit, respektive die Dauer eines Aktionspotentials erhöht.

Kapitel 5

Struktur und Funktion in pulscodierten neuronalen Netzen

In diesem Kapitel soll die Anwendung von pulsenden Neuronen am Beispiel eines assoziativen Speichers gezeigt werden. Die klassisch mit kontinuierlich anliegenden Werten arbeitenden Schaltungen werden hier mit pulsenden Neuronen betrieben. Die durch den Pulsbetrieb veränderten Eigenschaften und die notwendige Anpassung des assoziativen Speichers werden diskutiert und anhand von Simulationen verifiziert.

5.1 Fehlertoleranz neuronaler Assoziativspeicher

Für die Datenspeicherung in integrierten Schaltungen sind verschiedene Konzepte implementiert worden. So existieren neben herkömmlichen statischen und dynamischen Speichern, in denen Daten wahlfrei geschrieben und gelesen werden können auch solche Varianten, die zusätzliche Eigenschaften aufweisen. Dazu gehören Speicher, die eine Adressierung über den Inhalt – Content Addressable Memory (CAM) – erlauben, indem über Abbildungsregeln von einem angelegten Eingangsvektor (Schlüssel), der auch fehlerbehaftet sein kann, ein zugehöriger Ausgangsvektor (Daten) abgerufen wird. Eine Erweiterung des CAM stellen die Assoziativspeicher dar, welche unter den Namen Lernmatrix [91], Correlation Matrix [66], Sparse Distributed Memory [78] oder Associative Neural Memory [40, 62] veröffentlicht wurden. Während im CAM zu jeder Adresse genau ein Datensatz gespeichert werden kann und so ein fehlerfreier Abruf jedes gespeicherten Datensatzes möglich ist„ können im Assoziativspeicher auch Daten abgelegt werden, bei denen eine Speicherstelle mehrfach benutzt wird. Dieses Verfahren erlaubt die Speicherung einer größeren Anzahl von Daten, führt jedoch, je nach Füllgrad des Speichers, zu einem Anstieg der Fehler beim Abruf der Daten, der kompensiert werden muss.

Im Allgemeinen kann man Assoziativspeicher nach der Art der Musterabbildung in zwei Klassen unterscheiden: autoassoziativ und heteroassoziativ abbildende Speicher. Wird während der Programmierphase ein Eingangsvektor $\mathbf{x}$ auf einen davon verschiedenen Aus-

gangsvektor $\mathbf{y}$ abgebildet ($\mathbf{x} \rightarrow \mathbf{y}$), so nennt man diese Musterabbildung heteroassoziativ. Bildet man stattdessen in der Programmierphase einen Eingangsvektor $\mathbf{x}$ auf sich selbst ab ($\mathbf{x} \rightarrow \mathbf{x}$), wird die Speicherung autoassoziativ genannt.

Wird nach der Programmierphase ein zum Eingangsvektor $\mathbf{x}$ genügend ähnlicher Eingangsvektor $\mathbf{x}'$ an den Assoziativspeicher angelegt, ruft dieser bei der Heteroassoziation wieder den Ausgangsvektor $\mathbf{y}$ ab, bei der Autoassoziation wird der Originalvektor $\mathbf{x}$ am Ausgang rekonstruiert.

5.1.1 Struktur

Im Folgenden wird der spezielle Typ des neuronalen Assoziativspeichers, der binäre neuronale Assoziativspeicher (engl. binary neural associative memory) (BiNAM) betrachtet. Beim BiNAM handelt es sich um einen Assoziativspeicher, dessen Eingangsvektor $\mathbf{x}$ über eine Gewichtsmatrix $\mathbf{W}$ mit dem Ausgangsvektor $\mathbf{y}$ des Speichers assoziiert wird. Diese Gewichtsmatrix $\mathbf{W}$ speichert mit den Gewichten $w_{ij} \in [0;1]$ an den Schnittpunkten von Eingang x_j und Ausgang y_i eine binäre Relation zwischen zwei binären Vektoren. Eine schematische Darstellung eines Assoziativspeichers ist in Abb. 5.1 dargestellt.

Um die Assoziationsfähigkeit des Speichers zu erreichen, muss die leere Gewichtsmatrix des Assoziativspeichers programmiert werden. Dazu können verschiedene Lernregeln wie z. B. die Hebb-Regel [42] genutzt werden:

$$\Delta w_{ij} = \eta x_j y_i \tag{5.1}$$

Der Parameter η stellt hier eine konstante Lernrate dar, x_j repräsentiert den Wert des Eingangs j und y_i repräsentiert den Wert des Ausgangs i.

Da der hier betrachtete binäre neuronale Assoziativspeicher nur Eingänge und Gewichte mit den diskreten Werten 0 und 1 verarbeitet, muss die Lernregel (5.1) zur geklippten Hebb-Regel [70] modifiziert werden. Dabei wird in (5.1) $\eta = 1$ gesetzt. Bei dieser Lernregel gibt es im BiNAM entweder nur positive Gewichtsveränderungen oder keine.

Beim binären Assoziativspeicher der Form $m \times n$ seien $\mathbf{x} = (x_0, x_1, ..., x_{m-1}) \in [0;1]$ und $\mathbf{y} = (y_0, y_1, ..., y_{n-1}) \in [0;1]$. Die Gewichtsmatrix $\mathbf{W}$ mit $w_{ij} \in [0;1]$ sei zu Beginn des Lernvorgangs eine Null-Matrix. Der Vorgang des Speicherns einer Assoziation verändert die Elemente w_{ij} der Gewichtsmatrix nach

$$w_{ij} = w_{ij} \vee (x_j \wedge y_i). \tag{5.2}$$

Die Gewichtsmatrix $\mathbf{W}$ wird durch die geklippte Hebb-Regel in (5.2) programmiert. Die Matrix wird an den Schnittpunkten gleichzeitig aktivierter Eingänge und Ausgänge auf Eins gesetzt. Ein einmal auf Eins gesetztes Gewicht kann nicht wieder zurückgesetzt werden.

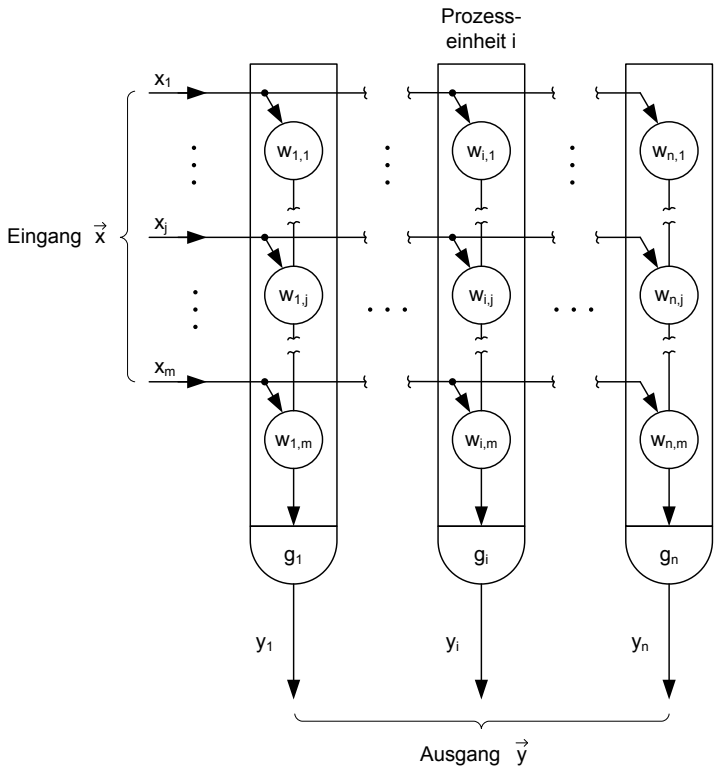


Abbildung 5.1: Allgemeine Struktur eines Assoziativspeichers.

Je mehr Muster gespeichert werden und je größer die Anzahl aktiver Elemente in den zu lernenden Mustern ist, umso stärker wird die Matrix mit Einsen besetzt. Dieses kann bei einer großen Anzahl gespeicherter Abbildungen zur „Überbesetzung“ der Assoziativspeichermatrix und zu Fehlern bei der späteren Assoziation führen.

Die in Abb. 5.1 dargestellte Aktivierungs- bzw. Bewertungsfunktion g_i kann durch Sigmoide oder eine Schwellwertfunktion mit der Schwelle Θ beschrieben werden. Im Folgenden wird für die Aktivierungsfunktion ein Schwellwertentscheider mit

$$g_i(s_i) = \begin{cases} 1 & \text{, falls } s_i \geq \Theta \\ 0 & \text{, sonst} \end{cases} \tag{5.3}$$

angenommen.

Die Bewertung des Musters $\mathbf{v}$ am Eingang erfolgt spaltenweise durch die Gewichtsmatrix w_{ij}. Der Eingangsvektor wird mit den korrespondierenden Elementen der Gewichtsmatrix gewichtet und aufsummiert, so dass für die Schwellwertfunktion die Summe

$$s_i = \sum_j w_{ij} v_j \tag{5.4}$$

sichtbar wird. Der Term (5.3) beschreibt hier die Funktion eines zu jeder Spalte i gehörenden Neurons: Die durch den Eingangsvektors ausgewählten Spalten werden durch die Synapsen bewertet. Die bewerteten Eingänge werden summiert und einem Entscheidungselement zugeführt.

Tabelle 5.1: Fehlerklassen im Assoziativspeicher. Der Fehler wird als Variation des Originalwerts mit $(1 + \delta)$ betrachtet.

Fehlerort / Fehlerklasse	Parameter
Eingangsmuster v_j^μ	$v_j^\mu \left(1 + \delta_{v_j}\right)$
Gewicht w_{ij}	$w_{ij} \left(1 + \delta_{w_{ij}}\right)$
Schwellenelement Θ_i	$\Theta_i \left(1 + \delta_{\Theta_i}\right)$
Summation s_i	$s_i \left(1 + \delta_{s_i}\right)$

5.1.2 Fehlertoleranz binärer neuronaler Assoziativspeicher

Im folgenden Abschnitt soll die Fehlertoleranz neuronaler Assoziativspeicher ermittelt werden. Dabei wird zuerst die Fehlertoleranz eines einzelnen Neurons der Spalte i des Speichers betrachtet. Im Anschluss werden die Ergebnisse zur Fehlertoleranz der einzelnen Spalte auf den gesamten Speicher erweitert.

Beim Assoziativspeicher können an verschiedenen Stellen Fehler auftreten, welche im Folgenden als Fehlerklassen aufgefasst werden. Tabelle 5.1 gibt die hier betrachteten Fehlerklassen und den Einflussort des Fehlers an. Einige von diesen Fehlerklassen weisen eine ähnliche Verteilung auf und können zusammengefasst werden. Darauf wird an den entsprechenden Stellen hingewiesen. Im Anschluss werden mögliche Kombinationen auftretender Fehler analysiert.

Für die Berechnung der Fehlertoleranz wird zunächst der Fall der binären Gewichte und Muster $w_{ij},\ v_j \in [-1; 1]$ betrachtet. Der Spezialfall für Gewichte und Muster $w_{ij},\ v_j \in [0; 1]$ wird anschließend daraus abgeleitet.

Fehler im Eingabevektor Nimmt man einen Fehler δ_j in den Zeilen des Eingabemusters $\mathbf{v}$ an, so erzeugt dieser einen Fehler im Summenvektor $\mathbf{s}$ des Assoziativspeichers. Die fehlerhafte Spaltensumme der Spalte i ergibt sich unter Berücksichtigung von δ_j zu

$$s_{\text{error},i} = \sum_j w_{ij} v_j \left(1 + \delta_j\right) = s_i + \sum_j w_{ij} v_j \delta_j \tag{5.5}$$

Ein in der Summe auftretender Fehler ist von den im Speicher gespeicherten Mustern abhängig und dann tolerierbar, wenn dieser kleiner ist, als der kleinste geometrische Abstand aller Muster zu einer trennenden Hyperebene im Darstellungsraum. Dieses wird in Abb. 5.2 durch Reduktion auf den zweidimensionalen Fall und eine Trennlinie zwischen zwei Klassen veranschaulicht. Der kleinste geometrische Abstand der Muster zur Trennlinie ist $\Delta_{i,\mu}$. Jeder Fehler im Eingabevektor, der größer ist als $\Delta_{i,\mu}$ führt zur falschen Klassifizierung des fehlerhaften Musters.

Der maximal erlaubte Fehler lässt sich also als minimaler Abstand aller Muster $\mathbf{v}^\mu$ zum

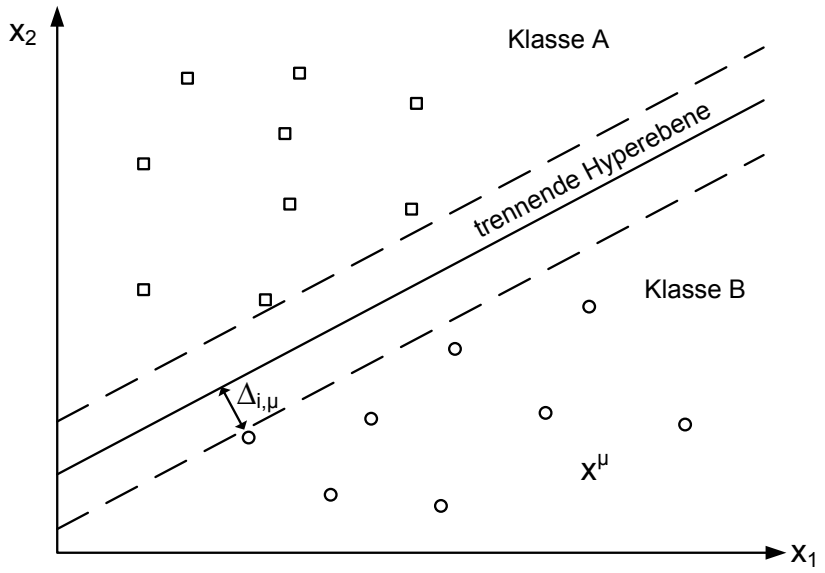


Abbildung 5.2: Geometrische Interpretation der Mustertrennung durch einen Schwellwert. Der minimale Abstand zum nächsten Element eines Musters ist mit $\Delta_{i,\mu}$ dargestellt.

Schwellwert auffassen. Dieses wird durch den minimalen Abstand der Summe s_i der ausgewählten Zeile zum Schwellwert Θ ausgedrückt.

$$\Delta_{i,\mu} = \min_{\mu} |s_i - \Theta| \tag{5.6}$$

Der rechte Teil von Gleichung (5.5) muss danach kleiner als $\Delta_{i,\mu}$ sein.

$$\Delta_{i,\mu} > \left| \sum_j w_{ij} v_j^\mu \delta_j \right| \tag{5.7}$$

Im „worst case“, d. h. die Fehler in jedem Element des Eingabemusters wirken sich gleich aus (d. h. alle Fehler haben entweder ein positives oder ein negatives Vorzeichen), verändert sich diese Bedingung zu

$$\Delta_{i,\mu} > \sum_j \left| w_{ij} v_j^\mu \right| |\delta_j| \,. \tag{5.8}$$

Nimmt man an, dass der Fehler in allen Eingängen gleich gemacht wird, z. B. hervorgerufen durch Prozessvariation bei der Herstellung oder einen Offset beim Anlegen des Musters, kann man den Fehler zu

$$\delta := |\delta_j| \quad \forall \quad j$$

umschreiben.

Unter Einbeziehung von 5.6 kann man nach dem Fehler δ auflösen:

$$\delta < \min_{\mu} \frac{\left| \left(\sum_j w_{ij} v_j^{\mu} \right) - \Theta \right|}{\sum_j \left| w_{ij} v_j^{\mu} \right|} \quad \forall \quad w_{ij},\, v_j^{\mu} \in [-1; 1] \tag{5.9}$$

Für den Spezialfall, dass alle Werte positiv sind und im Intervall $[0; 1]$ liegen ergibt sich

$$\delta < \min_{\mu} \left| 1 - \frac{\Theta}{\sum_j \left| w_{ij} v_j^{\mu} \right|} \right| \quad \forall \quad w_{ij},\, v_j^{\mu} \in [0; 1]\,. \tag{5.10}$$

Fehler in der Gewichtsmatrix Nimmt man einen Fehler δ_j in den Gewichten w_{ij} der i-ten Spalte der Gewichtsmatrix $\mathbf{M}$ an, so erzeugt dieser einen Fehler im Summenvektor $\mathbf{s}$ des Assoziativspeichers. Die fehlerhafte Spaltensumme ergibt sich unter Berücksichtigung von δ_j zu

$$s_{\text{error},i} = \sum_j w_{ij} \left(1 + \delta_j\right) v_j = s_i + \sum_j w_{ij} v_j \delta_j \tag{5.11}$$

Dieses entspricht der Form der fehlerhaften Summe, die bei der Betrachtung des Fehlers im Eingangsvektor ermittelt wurde. Für diesen Fall lässt sich mit der Verallgemeinerung $\delta := \delta_j \,\forall\, j$ die Lösung direkt angeben:

$$\delta < \min_{\mu} \frac{\left| \left(\sum_j w_{ij} v_j^{\mu} \right) - \Theta \right|}{\sum_j \left| w_{ij} v_j^{\mu} \right|} \quad \forall \quad w_{ij},\, v_j^{\mu} \in [-1; 1] \tag{5.12}$$

Fehler im Schwellenelement Der Summenterm, der den Fehler des Schwellenelements beinhaltet, lässt sich mit

$$\begin{aligned} s_i + \Theta_{\text{error},i} &= \sum_j w_{ij} v_j - \Theta \left(1 + \delta_i\right) \\ &= \left(s_i - \Theta\right) - \Theta \delta_i \end{aligned} \tag{5.13}$$

beschreiben. Die in diesem Abschnitt öfter angewandte Ungleichung für die untere Schranke des Fehlers ergibt in diesem Fall

$$\Delta_{i,\mu} \geq \left| -\Theta \delta_i \right|.$$

Mit der Verallgemeinerung, dass der Fehler δ_i in allen Schwellenelementen in gleicher Weise auftritt,

$$\delta := |\delta_i| \quad \forall \quad i,$$

lässt sich der maximal zu tolerierende Fehler umschreiben zu:

$$\delta < \min_{\mu} \frac{\left| \left(\sum_j w_{ij} v_j^{\mu} \right) - \Theta \right|}{|\Theta|} \quad \forall \quad w_{ij},\, v_j^{\mu} \in [-1;1] \tag{5.14}$$

Fehler in der Summenbildung

$$\begin{aligned} s_{\text{error},i} &= \sum_j w_{ij} v_j^{\mu} \,(1+\delta_i) \\ &= s_i \,(1+\delta_i) \\ &= s_i + \delta_i \sum_j w_{ij} v_j^{mu} \end{aligned} \tag{5.15}$$

$$\Delta_{i,\mu} \geq \left| \delta_i \sum_j w_{ij} v_j^{mu} \right| \tag{5.16}$$

Im „worst case“ ergibt sich für den maximalen Fehler die Forderung:

$$\Delta_{i,\mu} \geq |\delta_i| \sum_j \left| w_{ij} v_j^{mu} \right| \tag{5.17}$$

Mit der Annahme

$$\delta := |\delta_i| \quad \forall \quad i$$

lässt sich der maximal erlaubte Fehler δ für eine korrekte Entscheidung abschätzen:

$$\delta < \min_{\mu} \frac{\left| \left(\sum_j w_{ij} v_j^{\mu} \right) - \Theta \right|}{\sum_j \left| w_{ij} v_j^{\mu} \right|} \quad \forall \quad w_{ij},\, v_j^{\mu} \in [-1;1] \tag{5.18}$$

Kombinationsfehler

Kombinationsfehler treten immer dann auf, wenn in mehreren Schaltungsteilen Fehler in gleicher Weise auftreten. Dieses kann z. B. durch Parametervariationen in der Herstellung einer integrierten Schaltung hervorgerufen werden.

Fehler im Eingangsvektor und in der Gewichtsmatrix Nimmt man einen Fehler δ_{v_j} in den Zeilen des Eingabemusters $\mathbf{v}$, sowie einen weiteren Fehler δ_{w_j} in den Zeilen der Spalte i der Gewichtsmatrix an, so erzeugt dieser einen Fehler im Summenvektor $\mathbf{s}$ des Assoziativspeichers. Die fehlerhafte Spaltensumme ergibt sich zu

$$\begin{aligned} s_{\text{error},i} &= \sum_j w_{ij} \left(1 + \delta_{w_j}\right) v_j \left(1 + \delta_{v_j}\right) \\ &= \sum_j w_{ij} v_j \left(\delta_{w_j}\delta_{v_j} + \delta_{w_j} + \delta_{v_j} + 1\right) \\ &= s_i + \sum_j w_{ij} v_j \left(\delta_{w_j}\delta_{v_j} + \delta_{w_j} + \delta_{v_j}\right) \end{aligned} \tag{5.19}$$

Der hier auftretende Fehler ist in Abschnitt 5.1.2 bereits diskutiert worden und dann tolerierbar, wenn der er kleiner ist, als der kleinste geometrische Abstand aller Muster zur trennenden Hyperebene im Darstellungsraum.

$$\Delta_{i,\mu} = \min_{\mu} |s_i - \Theta| \tag{5.20}$$

Der rechte Teil von Gleichung (5.19) muss danach kleiner als $\Delta_{i,\mu}$ sein.

$$\Delta_{i,\mu} > \left| \sum_j w_{ij} v_j^{\mu} \left(\delta_{w_j}\delta_{v_j} + \delta_{w_j} + \delta_{v_j}\right) \right| \tag{5.21}$$

Im „worst case“ verändert sich diese Bedingung zu:

$$\Delta_{i,\mu} > \sum_j \left| w_{ij} v_j^{\mu} \right| \left(\left|\delta_{w_j}\delta_{v_j}\right| + \left|\delta_{w_j}\right| + \left|\delta_{v_j}\right| \right) \tag{5.22}$$

Die Annahme, die im Folgenden getroffen wird, muss mit Bedacht gewählt werden. Für eine Abhängigkeit zwischen dem Fehler des Eingangs und einem Fehler in der Gewichtsmatrix müssen diese praktisch dicht nebeneinander liegen und ähnlichen Prozessvariationen ausgesetzt sein. Dieses sei hier vorausgesetzt.

$$\delta := \left|\delta_{w_j}\right| = \left|\delta_{v_j}\right| \quad \forall \quad j \tag{5.23}$$

Die Ungleichung kann zu δ umgestellt werden:

$$\Delta_{i,\mu} > \sum_j \left| w_{ij} v_j^{\mu} \right| \left(\delta^2 + 2\delta\right) \tag{5.24}$$

$$\delta^2 + 2\delta < \min_{\mu} \frac{\sum_j w_{ij} v_j^\mu - \Theta}{\sum_j \left| w_{ij} v_j^\mu \right|} \tag{5.25}$$

$$\sqrt{(\delta+1)^2} < + \sqrt{\min_{\mu} \frac{\sum_j w_{ij} v_j^\mu - \Theta}{\sum_j \left| w_{ij} v_j^\mu \right|} + 1} \tag{5.26}$$

$$\delta < + \sqrt{\min_{\mu} \frac{\sum_j w_{ij} v_j^\mu - \Theta}{\sum_j \left| w_{ij} v_j^\mu \right|} + 1} - 1 \quad \forall \quad w_{ij},\, v_j^\mu \,\in\, [-1;1] \tag{5.27}$$

Fehler in der Gewichtsmatrix und im Schwellwertelement Wir nehmen an, dass wir eine einzelne Spalte i des Speichers betrachten, welche durch ein Eingabemuster v^μ erregt wird. Wenn die Parameter dieser Einheit, d. h. die Gewichte, die Übertragungsfunktion oder die Amplitude der angelegten Signale fehlerhaft sind, kann die Entscheidung des Neurons fehlerhaft sein. Gründe für Abweichungen in den Parametern können temporäre Störungen durch Rauschen oder permanente Einflüsse aus der Herstellung der Schaltung sein.

Wir betrachten wieder den maximalen Fehler, unter dessen Einfluss das Neuron noch eine richtige Entscheidung treffen kann (5.8). Die Parameter $\delta_j, \delta_\Theta \in [-1;1] \subset \mathbb{R}$ seien geringe Abweichungen von $w_{ij} v_j^\mu$ sowie dem Schwellenwert Θ. Damit erhalten wir die gestörte Übertragungsfunktion:

$$\begin{aligned} s_{\text{error},i} + \Theta_{\text{error}} &= \sum_j w_{ij} v_j^\mu \left(1 + \delta_j\right) - \Theta \left(1 + \delta_\Theta\right) \\ &= \left(s_i - \Theta\right) + \sum_j w_{ij} v j^\mu \delta_j - \Theta \delta_\Theta \end{aligned} \tag{5.28}$$

Das Neuron wird die korrekte Entscheidung treffen, wenn die Bedingung

$$\Delta_{i,\mu} > \left| \sum_j w_{ij} v_j^\mu \delta_j - \Theta \delta_\Theta \right| \tag{5.29}$$

erfüllt ist. Wie bereits beschrieben tritt der größte Fehler auf, wenn die rechte Seite von (5.29) maximiert wird:

$$\Delta_{i,\mu} > \sum_j \left| w_{ij} v_j^\mu \right| \left| \delta_j \right| + \left| \Theta \right| \left| \delta_\Theta \right| \tag{5.30}$$

Mit der Annahme $\delta := |\delta_{ij}| = |\delta_\Theta|$ oder $\delta := \max(|\delta_{ij}|, |\delta_\Theta|)$ für alle j, ergibt sich:

$$\delta < \min_\mu \frac{\left|\left(\sum_j w_{ij} v_j^\mu\right) - \Theta\right|}{\sum_j \left|w_{ij} v_j^\mu\right| + |\Theta|} \quad \forall \quad w_{ij},\, v_j^\mu \in [-1;1] \subset \mathbb{R} \tag{5.31}$$

Fehlertoleranz gegenüber stuck-at-Fehlern

Stuck-at-Fehler treten häufig als Erscheinung von Prozessparametervariationen bei der Fertigung integrierter Schaltungen auf, z. B. durch Kurzschluss benachbarter Leitungen. Genauso können sie durch Elektromigration hervorgerufen werden, bei der Leitungen im besten Fall hochohmig werden. Kennzeichen dieser Art von Fehlern bei Speicherstrukturen ist das Verweilen eines Speicherelements auf dem logischen Null-Wert (Stuck-at-0, ST0) oder dem logischen Eins-Wert (Stuck-at-1, ST1). Die Auswirkung des Auftretens dieser Fehler in der Assoziativspeichermatrix auf den Informationsgehalt des Speichers soll an dieser Stelle rekapituliert und präzisiert werden, da diese Betrachtungen bereits in [83] durchgeführt und anhand von Simulationen verifiziert wurden.

Zunächst werden für die folgende Rechnung die grundlegenden Ereignisse angegeben:

Es sei A das Ereignis „Matrixgewicht besetzt“. Die Wahrscheinlichkeit für das Eintreten dieses Ereignisses steigt mit der Anzahl aktivierter Elemente l im Eingangsvektor der Größe m und der Anzahl aktivierter Elemente k im Ausgangsvektor der Größe n sowie der Anzahl trainierter Muster z und ergibt sich nach [77] zu:

$$p(A) = p_{\text{on}} = 1 - \left(1 - \frac{kl}{mn}\right)^z \tag{5.32}$$

Die Gegenwahrscheinlichkeit für das Ereignis $\bar{A}$ „Matrixgewicht unbesetzt“ ist dementsprechend $p(\bar{A}) = p_{\text{off}} = 1 - p_{\text{on}}$. Palm [78] gibt auch die Speichereffizienz I_h für heteroassoziative Abbildungen in Assoziativspeichern an.

$$I_h = \sum_{\mu=1}^{z} \left[\log_2 \binom{n}{k} - \log_2 \binom{k + N_{k1}}{k}\right] \tag{5.33}$$

Dabei ist die Zahl N_{k1} die Anzahl zusätzlicher Einsen im Ausgabevektor $\mathbf{y}^k$, die fälschlich durch Überlagerung von gespeicherten Mustern im Speicher hervorgerufen werden. Der Binomialkoeffizient $\binom{n}{k}$ stellt die Anzahl der Möglichkeiten dar, mit der k aktivierte Ausgänge (Einsen) an n möglichen Positionen im Ausgabevektor auftreten können. Daher ist $\log_2 \binom{n}{k}$ der maximal mögliche Informationsgehalt eines Ausgabevektors

unter der Annahme, dass alle Muster dieselbe Wahrscheinlichkeit des Auftretens haben. Der Binomialkoeffizient $\begin{pmatrix} k+N_{k1} \\ k \end{pmatrix}$ beschreibt die Anzahl der Möglichkeiten, die k korrekt aktivierten Ausgänge in den $k+N_{k1}$ aktivierten Ausgängen unterzubringen. $\log_2 \begin{pmatrix} k+N_{k1} \\ k \end{pmatrix}$ ist der Informationsgehalt, der durch das Auftreten der zusätzlichen Einsen im Ausgabevektor verloren geht. Der gesamte Informationsgehalt setzt sich aus der Summe aller gespeicherten Muster $\mathbf{x}^\mu \rightarrow \mathbf{y}^\mu$ und dem Verlust durch zusätzlichen Codierungsaufwand zur Eliminierung der zusätzlichen aktivierten Ausgänge zusammen.

Mit der Annahme, dass

$$E(N_{k1}) = (n-k) \cdot p_{on}^{l} \tag{5.34}$$

eine gute Approximation des Erwartungswerts für N_{k1} ist, sowie der Voraussetzung, dass die Einsen in der Gewichtsmatrix zufällig verteilt sind, erhält man einen Erwartungswert für I_h [77]:

$$E(I_h) \geq -z \cdot \sum_{i=0}^{k-1} \log_2 \left(\frac{E(N_{k1}) + k - i}{n - i} \right) \tag{5.35}$$

Stuck-at-1 Fehler

Nun sei B das Ereignis „Matrixgewicht ist stuck-at-1 (ST1)“ mit der Wahrscheinlichkeit $p(B) = p_{\text{st1}}$. Die Wahrscheinlichkeit $p_{\text{on}}^{'}$, dass bei zufälliger Auswahl eines Elements der Gewichtsmatrix dieses durch ein programmiertes Muster oder einen stuck-at-1 Fehler besetzt ist, ergibt sich zu:

$$\begin{aligned} p_{\text{on}}^{'} &= p(A \vee B\bar{A}) \\ &= p_{\text{on}} + (1 - p_{\text{on}})\, p_{\text{st1}} \\ &= 1 - \left(1 - \frac{kl}{mn}\right)^z + \left(1 - \frac{kl}{mn}\right)^z p_{\text{st1}} \\ &= \left(1 - \left(1 - \frac{kl}{mn}\right)^z\right)(1 - p_{\text{st1}}) + p_{\text{st1}} \end{aligned} \tag{5.36}$$

Mit einem Anstieg für die Wahrscheinlichkeit $p_{\text{on}}^{'}$, dass ein Matrixgewicht gesetzt ist, steigt auch der Erwartungswert für zusätzliche Einsen nach (5.35). In Abb. 5.3a sind der Informationsgehalt einer Gewichtsmatrix sowie die Anzahl zusätzlicher fehlerhafter Einsen aufgetragen. Die Ergebnisse wurden durch numerische Auswertung von (5.34) und (5.35) für eine Speichermatrix der Größe 4096×4096 Gewichten mit $k = 13$ aktivierten Eingängen und $l = 3$ aktivierten Ausgängen ermittelt. Diese theoretisch gewonnenen Ergebnisse können durch Simulationsergebnisse aus [83] untermauert werden.

Die Annahme von 1% defekten Gewichtselementen scheint für heutige VLSI Technologie-Prozesse sehr hoch gewählt zu sein, vor allem da die ITRS Roadmap [89] für heutige Prozesse eine Defekt-Rate von 3500 Fehlern pro m^2 angibt. Es ist jedoch zu erwarten, dass diese Defekt-Rate mit Einführung zukünftiger nanoelektronischer Architekturen wieder zunimmt und daher fehlertolerante Systeme zum Einsatz kommen müssen.

Stuck-at-0 Fehler

Stuck-at-0 Fehler in der Gewichtsmatrix führen zu fehlenden Einsen im Ausgabevektor. Die fehlenden Einsen können durch den Term N_{k0} beschrieben werden. Zur gleichen Zeit sinkt die Anzahl der fehlerhaften zusätzlichen Einsen N_{k1}.

Wir vereinfachen die folgenden Rechnungen durch die Annahme eines bekannten N_{k0}. Der Informationsgehalt I_h eines Speichers ist dann durch

$$I_h = \sum_{\mu=1}^{z} \left[\log_2 \begin{pmatrix} n \\ k \end{pmatrix} - \log_2 \begin{pmatrix} N_{k1} + k - N_{k0} \\ k - N_{k0} \end{pmatrix} - \log_2 \begin{pmatrix} n - N_{k1} - k + N_{k0} \\ N_{k0} \end{pmatrix} \right] \quad (5.37)$$

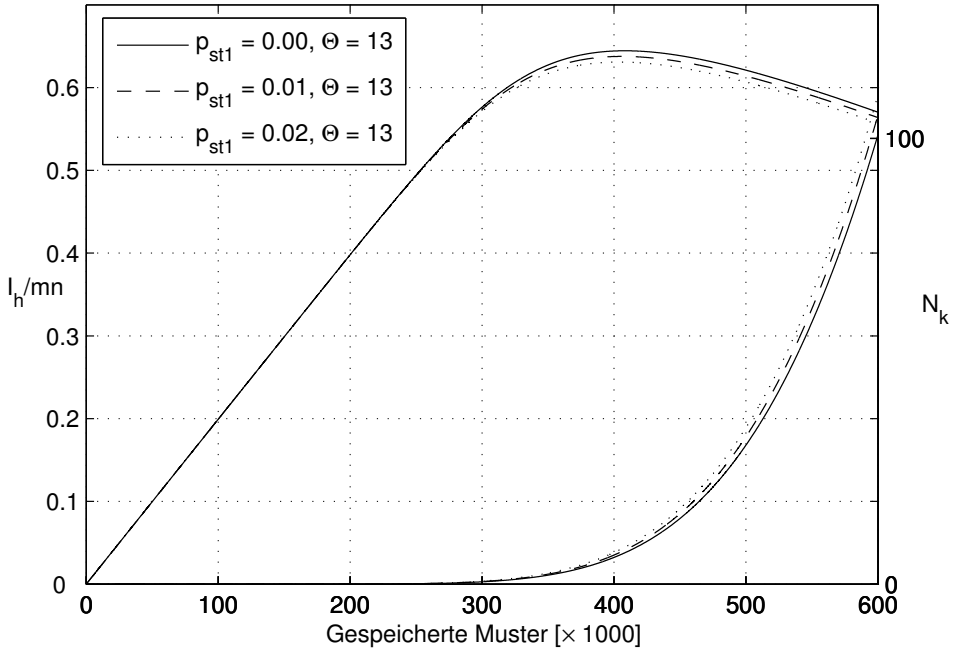
gegeben. Der Informationsgehalt eines einzelnen gespeicherten Musters $\log_2 \begin{pmatrix} n \\ k \end{pmatrix}$ wird durch den Informationsgehalt reduziert, der zur Bestimmung der zusätzlichen Einsen in Ausgangsvektor nötig ist $\log_2 \begin{pmatrix} N_{k1} + k - N_{k0} \\ k - N_{k0} \end{pmatrix}$. Zusätzlich wird der Informationsgehalt durch die Bestimmung der fehlenden Einsen N_{k0} in den $n - N_{k1} - k + N_{k0}$ Nullen des Ausgangsvektors reduziert. Daher muss noch $\log_2 \begin{pmatrix} n - N_{k1} - k + N_{k0} \\ N_{k0} \end{pmatrix}$ abgezogen werden.

Eine Konsequenz der spärlichen Codierung ist die große Anzahl an Nullen verglichen mit der Anzahl an Einsen im Ausgabevektor. Daher hat der dritte Term von (5.37) großen Einfluss auf die Reduktion des Informationsgehalts der Speichermatrix. Um die Wahrscheinlichkeit zu vermindern, dass Einsen fehlen, kann daher die Schwelle Θ herabgesetzt werden.

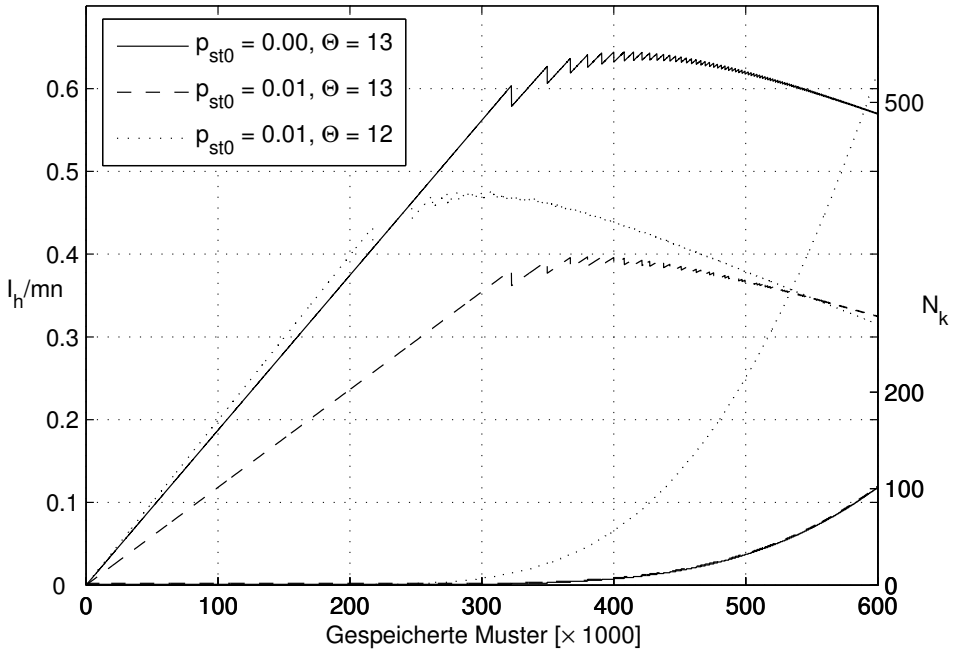
In Abbildung 5.3b stellt die gepunktete Linie mit $p_{\text{st0}} = 0{,}01, \Theta = 12$ das theoretische Maximum des Informationsgehalts mit abgesenkter Feuerschwelle dar. Der minimale Informationsgehalt liegt zwischen dieser Kurve und der Kurve mit $p_{\text{st0}} = 0{,}01, \Theta = 13$. Das Bedeutet, dass obwohl man erst einmal durch die abgesenkte Schwelle eine größere Anzahl an Fehlern im Ausgang in Kauf nimmt, der Nutzen durch die auf dem Wert 0 festsitzenden Gewichte wieder zunimmt und man am Ende einen höheren Informationsgehalt der Speichermatrix erhält.

Berechnungen bezüglich N_{k1} and N_{k0}

Um die Anzahl der aktivierten Ausgänge zu bestimmen, die auftritt, wenn die Feuerschwelle zur Kompensation der stuck-at-0 Fehler abgesenkt wird, nehmen wir die Wahrscheinlich-



(a) Auswirkung von Stuck-at-1 Fehlern bei $p_{\mathrm{st1}} = 0$; 0,01; 0,02.



(b) Auswirkung von Stuck-at-0 Fehlern mit $p_{\mathrm{st0}} = 0$; 0,01; $N_k = N_{k0} + N_{k1}$ und mit $p_{\mathrm{st0}} = 0{,}01$ und abgesenkter Schwelle Θ (gepunktete Linie).

Abbildung 5.3: Auswirkung der stuck-at Fehler auf den Informationsgehalt I_h/mn des Assoziativspeichers (n, m = 4096; l = 13; k = 3).

keit, dass ein Gewicht auf Eins gesetzt ist und multiplizieren diese mit der Größe des Eingangsvektors. Diese Abschätzung gibt an, wie viele Elemente in einer Spalte nach dem Programmieren von z Mustern gesetzt sind. Es gibt $\begin{pmatrix} \lceil m \cdot p_{on} \rceil \\ \Theta \end{pmatrix}$ Möglichkeiten, ein auf Eins gesetztes Gewicht mit einem Eingangsmuster mit Θ aktiven Elementen zu treffen. Da die Anzahl aktivierter Elemente k größer ist als die Feuerschwelle Θ, die abgesenkt wurde, beträgt die Wahrscheinlichkeit mindestens Θ gesetzte Gewichte zu treffen (unter Summation aller Möglichkeiten zwischen Θ und k aktivierten Eingängen)

$$p_{out1} = \sum_{i=\Theta}^{k} \frac{\begin{pmatrix} \lceil m \cdot p_{on} \rceil \\ i \end{pmatrix} \begin{pmatrix} \lceil m \cdot (1 - p_{on}) \rceil \\ k - i \end{pmatrix}}{\begin{pmatrix} m \\ k \end{pmatrix}}. \tag{5.38}$$

Letztendlich ist der Erwartungswert für zusätzliche Einsen $E(N_{k1})$:

$$E(N_{k1}) = n \cdot p_{out1} \tag{5.39}$$

Weiter wird der Erwartungswert für fehlende Einsen $E(N_{k0})$ nach

$$E(N_{k0}) = k \cdot p_{st0} \tag{5.40}$$

abgeschätzt, wobei p_{st0} die Wahrscheinlichkeit für einen stuck-at-0 Fehler ist.

Fehlertoleranz gegenüber Eingabefehlern

Eingabefehler zeichnen sich dadurch aus, dass nur ein Teil des trainierten Musters korrekt angelegt wird. Der Rest ist durch zusätzliche aktivierte Elemente (zusätzliche Einsen im Muster) oder deaktivierte Elemente (fehlende Einsen im Muster) verfälscht. Die folgenden Betrachtungen werden vor der üblichen Wahl der Schwelle Θ gemacht, welche anhand der Zahl der aktivierten Elemente im Eingangsvektor gewählt wird. Es sei nicht verschwiegen, dass es andere Möglichkeiten der Schwellenwahl gibt. Der Einfluss anderer Schwellen auf den Informationsgehalt der Speichermatrix wird später diskutiert.

Fehlende Einsen im Eingabevektor Es sei der Parameter l' die Anzahl der Einsen im Eingabevektor $\mathbf{v}^{\mu}$. Weiterhin gelte die Bedingung, dass die Anzahl aktivierter Elemente im Eingabevektor kleiner der Anzahl aktivierter Elemente der Trainingsvektoren sei ($l' < l$). Da die Schwelle Θ nach der Anzahl vorhandener Einsen im Eingabevektor gewählt wird ($\Theta \stackrel{!}{=} l'$), wird durch die aktivierten Zeilen das gespeicherte Muster – fehlerfreie Speicherung vorausgesetzt – weiterhin korrekt abgerufen. In den Spalten, die nicht zum gespeicherten Muster gehören, können mit der Besetzungswahrscheinlichkeit p_{on} auch Einsen gespeichert

sein, welche zu zusätzlichen Einsen im Ausgabevektor führen können. Der Erwartungswert $E(N_{k1})$ für die Anzahl zusätzlicher Einsen ergibt sich mit der kleineren Anzahl l' Einsen im Eingabevektor zu:

$$E(N_{k1}) = (n-k) \cdot p_{\text{on}}^{l'} \tag{5.41}$$

Mit abnehmender Anzahl durch den Eingabevektor aktivierter Zeilen steigt die Anzahl zusätzlicher Einsen im Ausgabevektor, so dass der Informationsgehalt I_h der Speichermatrix nach (5.42) sinkt (Abb. 5.4a).

$$I_h = \sum_{\mu=1}^{z} \left[\log_2 \left(\begin{array}{c} n \\ k \end{array} \right) - \log_2 \left(\begin{array}{c} k+N_{k1} \\ k \end{array} \right) \right] \tag{5.42}$$

Zusätzliche Einsen im Eingabevektor Neben den fehlenden Einsen im Eingabemuster können auch zusätzliche Einsen im Eingabemuster auftreten. Es ist üblich, die Schwelle Θ so zu wählen, dass sie der Anzahl Einsen im Eingabemuster entspricht. Diese Vorgehensweise führt jedoch zu einer starken Abnahme der Informationskapazität der Assoziativspeichermatrix, sobald nur ein Bitfehler zugelassen wird. Dieses soll im Folgenden verdeutlicht werden.

Es sei l die Anzahl Einsen in den fehlerfreien Trainingsvektoren. Für den Abruf betrachten wir $l' > l$ Einsen, so dass sich $l - l'$ zusätzliche Einsen im Eingang ergeben. Die Wahl der Schwelle wird mit $\Theta \stackrel{!}{=} l'$ getroffen.

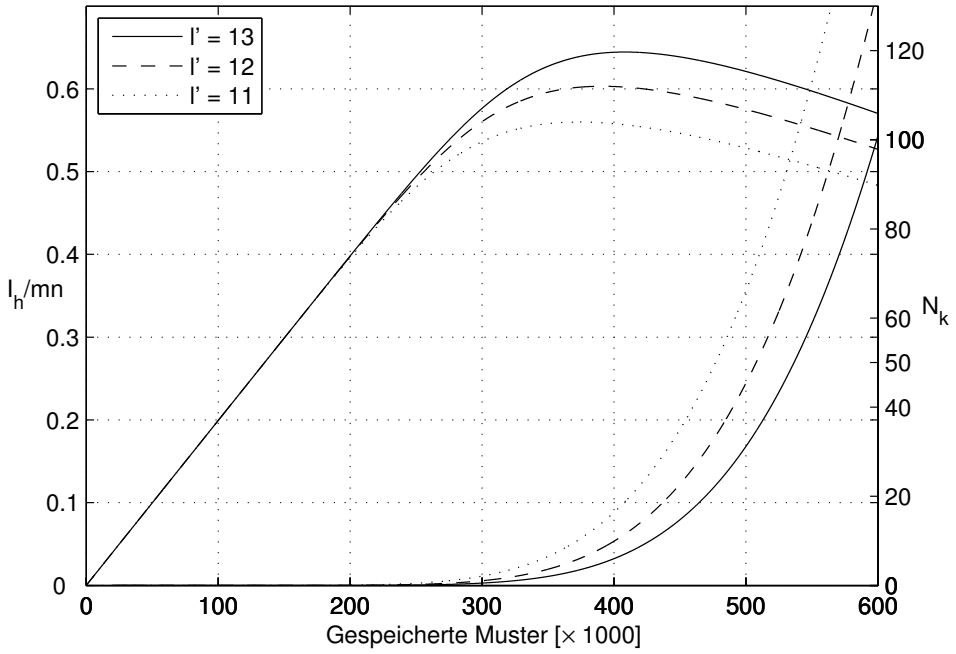
Ein Effekt, den man allein aus Überlegungen zur Wahl der Schwelle vorhersagen kann, ist, dass bei wenigen gespeicherten Mustern im Assoziativspeicher und zusätzlichen Einsen im Eingabemuster keines der gespeicherten Muster abgerufen werden kann, da ohne eine bestimmte Belegung der Matrix die Schwelle Θ nicht erreicht werden kann. Neben zusätzlichen Einsen im Ausgangsvektor müssen nun also auch fehlende Einsen des Originalmusters in die Berechnung des Informationsgehalts mit einbezogen werden.

Der Erwartungswert für zusätzliche Einsen im Ausgabevektor ist wie im Falle fehlender Einsen:

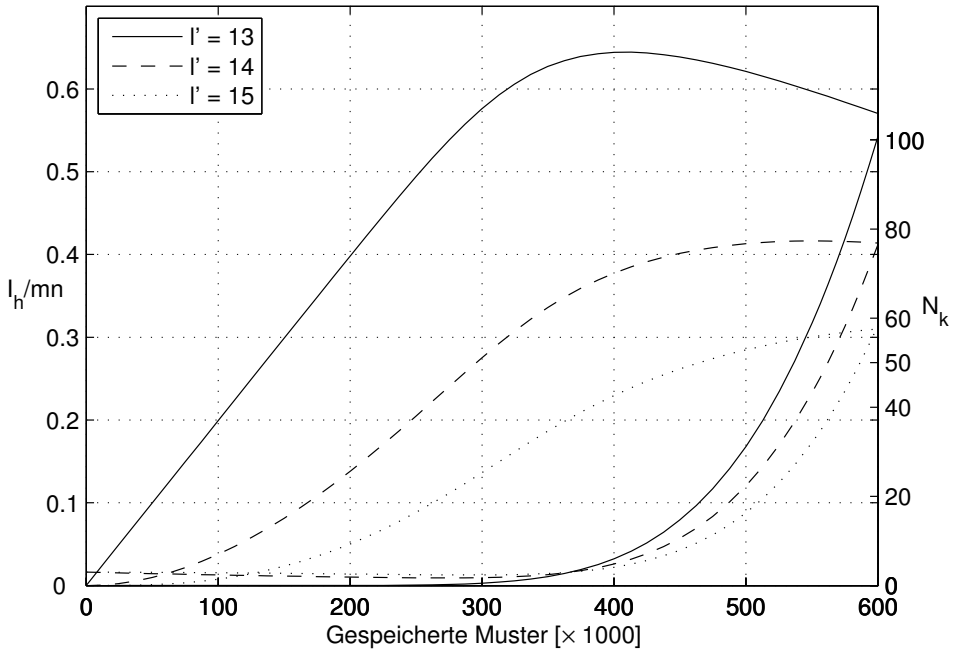
$$E(N_{k1}) = (n-k) \cdot p_{\text{on}}^{l'} \tag{5.43}$$

Die Berechnung der fehlenden Einsen des abzurufenden Musters berechnet sich etwas aufwändiger. Zunächst muss bestimmt werden, wie hoch die Wahrscheinlichkeit ist, dass neben den l bereits ausgewählten Zeilen des richtigen Musters mit den $l' - l$ zusätzlichen Einsen im Eingabemuster ein gesetztes Matrixgewicht getroffen wird.

$$p_{\text{zusätzlich},w=1} = \frac{p_{on} \cdot m - l}{m - l} \tag{5.44}$$



(a) Informationsgehalt des Speichers bei fehlenden Einsen im Eingabevektor ($l' = 13, 12, 11$).



(b) Informationsgehalt des Speichers bei zusätzlichen Einsen im Eingabevektor ($l' = 13, 14, 15$; $N_k = N_{k0} + N_{k1}$).

Abbildung 5.4: Auswirkung von fehlenden Einsen und zusätzlichen Einsen im Eingabemuster auf den Informationsgehalt I_h/mn der Assoziativspeichermatrix (m, $n = 4096$; $l = 13$; $k = 3$; $\Theta = l'$).

Der Erwartungswert für fehlende Einsen der k erwarteten Einsen im Ausgabevektor kann nun mit

$$E(N_{k0}) = k \cdot \left(1 - \left(\frac{p_{on} \cdot m - l}{m - l}\right)^{l'-l}\right) \tag{5.45}$$

abgeschätzt werden und dient als Grundlage für die Berechnung des Informationsgehalts der Assoziativspeichermatrix:

$$I_h = \sum_{\mu=1}^{z} \left[\log_2 \left(\begin{array}{c} n \\ k \end{array}\right) - \log_2 \left(\begin{array}{c} N_{k1} + k - N_{k0} \\ k - N_{k0} \end{array}\right) - \log_2 \left(\begin{array}{c} n - N_{k1} - k + N_{k0} \\ N_{k0} \end{array}\right)\right] \tag{5.46}$$

Die Abnahme des Informationsgehalts I_h bei zusätzlichen Einsen im Eingabevektor ist in Abb. 5.4b dargestellt. Es wird deutlich, dass bei Wahl der Schwelle in Abhängigkeit von der Aktivität des Eingangs zusätzliche Einsen im Eingabevektor einen sehr viel größeren Einfluss auf die Speicherkapazität des Assoziativspeichers besitzen, als fehlende Einsen. Dieses Ergebnis ist entgegengesetzt zu den Ergebnissen der Analyse von stuck-at Fehlern, bei denen stuck-at-0 Fehler in der Gewichtsmatrix größeren Einfluss auf die Speicherkapazität hatten, als stuck-at-1 Fehler.

5.2 Einfluss der Pulscodierung auf die Funktion

Im Folgenden soll der bisher betrachtete Assoziativspeicher um die Eigenschaft der Verarbeitung von pulscodierten Eingangsdaten erweitert werden. Zusätzlich zu den bekannten fehlerkorrigierenden Eigenschaften aus den vorherigen Abschnitten stellt die Verarbeitung von pulscodierten Mustern weitere Anforderungen an den Speicher.

Am grundlegenden Verhalten des Assoziativspeichers ändert sich bei der Umstellung auf die Verarbeitung von Pulsen nichts, jedoch wird das statische Schwellenelement durch ein LIAF Neuron mit statischer Schwelle ersetzt. Diese Ersetzung führt dazu, dass nach Abschluss der Lernschritte die Gewichte der Gewichtsmatrix in einer Weise angepasst werden müssen, dass nur ein korrekt angelegtes Pulsmuster das Membranpotential der erregten Neurone über die Feuerschwelle anheben kann.

Es kann gezeigt werden, dass die bisherigen Betrachtungen für den statischen Assoziativspeicher auch für einen mit pulsenden Neuronen aufgebauten Assoziativspeicher (PCNN-AM) gültig sind. Dazu muss vorausgesetzt werden, dass auch beim PCNN-AM „perfekte“, d. h. fehlerfreie Muster anliegen und die Schwelle (hier die Feuerschwelle) geeignet gewählt ist. Dieses kann analog zum statischen Speicher erfolgen. Weiter sei festgelegt, dass ein Abruf eines Musters aus dem PCNN-AM mit dem ersten Auftreten eines Pulsmusters am Eingang erfolgt. Während im klassischen, statischen AM die Schwelle

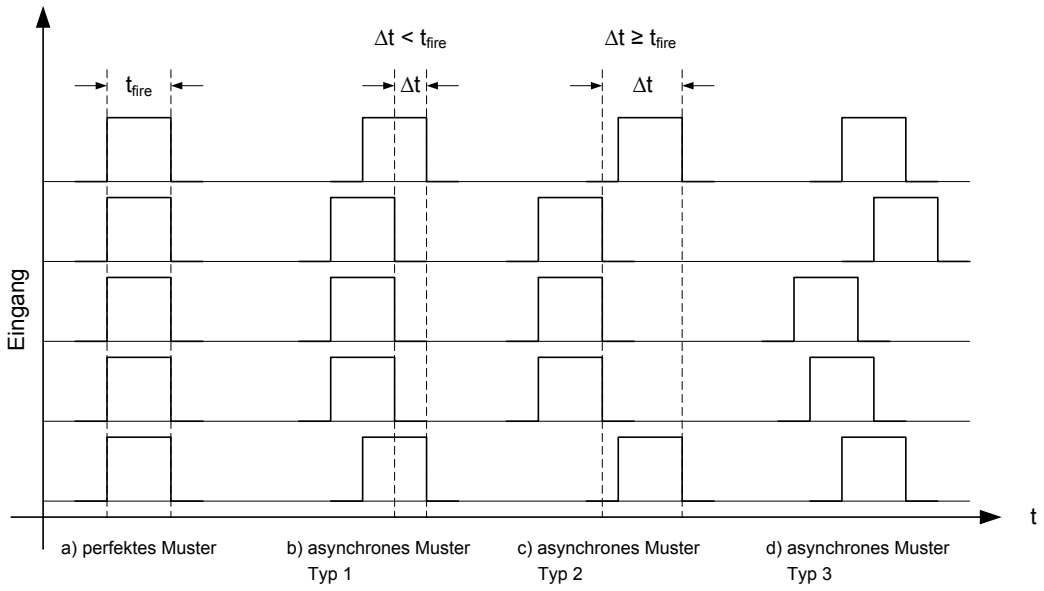


Abbildung 5.5: Arten von Synchronisationszuständen angelegter Muster.

in Abhängigkeit von der Anzahl der aktiven Elemente des Eingangs $\Theta = l'$ gewählt wird, gilt für die Wahl der Feuerschwelle des PCNN-AM unter Zuhilfenahme von (3.54)

$$U_{\mathrm{TH}} = l' \cdot \left[i \cdot \frac{1 - \exp\left(-\frac{t_{\mathrm{fire}}}{\tau}\right)}{g_{\mathrm{leak}}} \right] \tag{5.47}$$

mit den in Kapitel 3.3 beschriebenen Parametern und einem frei wählbaren Grundstrom i, der von einem einzelnen Gewicht verstärkt wird.

Die lineare Abhängigkeit der Feuerschwelle des PCNN-AM führt zu den gleichen Zusammenhängen in Bezug auf das Verhalten bei allen bisher betrachteten Fehlern des Assoziativspeichers.

Wenn im Folgenden von einem korrekt angelegten Muster gesprochen wird, bedeutet dieses, dass das Muster korrekt, d. h. ohne fehlende oder zusätzliche Information, auf den zuvor trainierten Eingängen angelegt wird. Die Korrektheit bezieht sich hier also auf die örtliche Verteilung der Pulse. Dagegen kann ein angelegtes Muster durchaus eine bestimmte zeitliche Verteilung annehmen, im besten Fall laufen allerdings alle Pulse zur exakt gleichen Zeit ein.

In Abb. 5.5 sind verschiedene zeitliche Verläufe eines einlaufenden Pulsmusters dargestellt. Die fünf übereinander liegenden Pulse müssen nicht örtlich beieinander liegen, sondern stellen beliebige, aber zu einem Muster gehörende Eingänge dar, welche in einer bestimmten Zeit erregt werden müssen, um ein Aktionspotential am Ausgang zu erzeugen und ein Muster aus dem Speicher abzurufen. Der erste Fall (Abb. 5.5a) zeigt das synchrone Einlaufen eines vollständigen Pulsmusters. Dieser Fall ist in Bezug auf die Wahl der Parameter des Speichers leicht zu lösen und beschränkt sich auf eine einfache Wahl der Gewichte w, um die Feuerschwelle der LIAF Neurone zu erreichen. Wir wählen l als

Anzahl der aktiven Eingänge im Eingangsvektor $\mathbf{x}$ der Größe m. Grundsätzlich muss zum Erreichen der Feuerschwelle eine Bedingung erfüllt sein, welche von der Anzahl der aktiven Eingänge l in einem korrekten Muster, dem dadurch erzeugten Ladestrom $(w \cdot i)$ und dem Verlustterm durch Leckströme g_{leak} im LIAF Neuron abhängt (5.48).

$$\frac{wli}{g_{\text{leak}}} \geq U_{\text{TH}} \quad \text{wobei} \quad l \in \mathbb{N}^* \tag{5.48}$$

Dabei beschreibt i den Grundstrom, den jede Synapse erzeugen kann und w die Stromverstärkung in der Synapse, die zum Erreichen der Feuerschwelle im empfangenden LIAF Neuron gewählt werden muss. Eine weitere Bedingung stellt die zeitliche Ausprägung der Pulse: Die Feuerschwelle muss innerhalb der Feuerzeit t_{fire} des Eingangs erreicht werden. Um dieses zu erreichen, muss der Verstärkungsfaktor w, der im Folgenden als Gewicht bezeichnet wird, der Bedingung

$$w \geq \frac{U_{\text{TH}} - u_{c,0} \cdot \exp(-\frac{t_{\text{fire}}}{\tau})}{1 - \exp(-\frac{t_{\text{fire}}}{\tau})} \cdot \frac{g_{\text{leak}}}{li} \qquad \text{mit} \qquad \tau = \frac{C}{g_{\text{leak}}} \tag{5.49}$$

genügen. Der Parameter $u_{c,0}$ stellt das zu Beginn der Betrachtung vorhandene Membranpotential dar, welches bei genügend großen zeitlichen Abständen zwischen einlaufenden Pulsmustern gegen den Wert 0 V tendiert.

Der zweite Fall eines einlaufenden Pulsmusters (Abb. 5.5b)zeigt eine leichte Asynchronität, bei der die Pulse mit einer Verzögerung von maximal Δt bezogen auf den ersten einlaufenden Puls bzw. die erste einlaufende Pulsgruppe am Eingang einlaufen, sich aber zeitlich noch überlagern. Das zum Erreichen der Feuerschwelle notwendige Gewicht wird im folgenden Abschnitt ermittelt. Dabei bezeichnet der Parameter l die Anzahl aller zu einem Muster gehörenden ausgewählten Eingänge. Der Parameter l_2 beschreibt die Anzahl Eingänge, die in Bezug auf die erste eintreffende Pulsgruppe l_1 eines Musters um die Zeit Δt verzögert sind. Dabei sind zwei Fälle zu unterscheiden: Im ersten Fall überlagern die eintreffenden verspäteten Pulse die zuerst einlaufenden Pulse um die Zeit $t_{\text{fire}} - \Delta t$ und es existiert ein Zeitraum, in dem das empfangende Neuron durch alle Eingänge gleichzeitig erregt wird. Im zweiten Fall ist die Verzögerung der verspäteten Pulsgruppe größer als die Feuerzeit der ersten Pulsgruppe, so dass hier bereits eine Entladung des empfangenden Neurons zwischen dem Eintreffen der beiden Pulsgruppen stattfinden kann.

Zunächst ist zu prüfen, ob ein Aktionspotential von nur l_2 verspäteten Elementen aus l aktiven Eingängen ausgelöst werden kann. Ist dieses der Fall, muss die gesamte Zeit $t_{\text{fire}} + \Delta t$ des Einlaufens aller Pulse betrachtet werden. Die Integration des Eingangsstroms über die Zeit von $t = 0 \ldots \Delta t$ für die ersten l_1 Pulse, über $t = \Delta t \ldots t_{\text{fire}}$ für das gemeinsam feuernde Muster und $t = t_{\text{fire}} \ldots t_{\text{fire}} + \Delta t$ für die l_2 einlaufenden verspäteten Pulse ergibt eine Bestimmungsgleichung für w:

$$
\begin{aligned}
w \geq U_{\mathrm{TH}} \cdot \frac{g_{\mathrm{leak}}}{i} \cdot \Bigg[& (l - l_2) \cdot \left(1 - \exp\left(-\frac{\Delta t}{\tau}\right)\right) \cdot \exp\left(-\frac{t_{\mathrm{fire}}}{\tau}\right) \\
& + l_2 \cdot \left(1 - \exp\left(-\frac{\Delta t}{\tau}\right)\right) \\
& + l \cdot \left(\exp\left(-\frac{\Delta t}{\tau}\right) - \exp\left(-\frac{t_{\mathrm{fire}}}{\tau}\right)\right) \Bigg]^{-1}
\end{aligned}
\tag{5.50}
$$

Dieses stellt das minimal einzustellende Gewicht für den besten Fall, die eine Verzögerung annehmen sollte, dar. Kann jedoch mit den verzögerten Pulsen alleine kein Aktionspotential ausgelöst werden, muss die möglicherweise sehr kurze Zeit, in der das gesamte Muster anliegt ausreichen, um die Feuerschwelle des Neurons zu erreichen. Daher wird das Gewicht nach der Integration über den Zeitraum von $t = 0 \ldots t_{\mathrm{fire}}$ bestimmt. Für den zweiten genannten Fall ergibt sich ein Gewicht, das größer ist, als das eben hergeleitete Gewicht (5.50).

$$
\begin{aligned}
w \geq U_{\mathrm{TH}} \cdot \frac{g_{\mathrm{leak}}}{i} \cdot \Bigg[& (l - l_2) \cdot \left(1 - \exp\left(-\frac{\Delta t}{\tau}\right)\right) \cdot \exp\left(-\frac{t_{\mathrm{fire}} - \Delta t}{\tau}\right) \\
& + l \cdot \left(1 - \exp\left(-\frac{t_{\mathrm{fire}} - \Delta t}{\tau}\right)\right) \Bigg]^{-1}
\end{aligned}
\tag{5.51}
$$

Diese Betrachtung ist gültig, solange die verzögerten Pulse allein schon ein Aktionspotential auslösen können. Im allgemeinen Fall ergibt sich diese Eigenschaft erst nach der Wahl der Gewichte, so dass die gesamte Betrachtung mit den ermittelten Gewichten erneut durchgeführt werden muss, und man sich schrittweise der Lösung nähert. Es hat sich allerdings gezeigt, dass es ausreichend ist, für den allgemeinen Fall das Gewicht für beide Fälle zu ermitteln und das Gewicht zu wählen, das am kleinsten ist.

Die erhöhte Robustheit gegenüber asynchron einlaufenden Mustern wird mit einer erhöhten Fehleranfälligkeit bei synchron einlaufenden gestörten Mustern mit zusätzlichen aktiven Eingängen erkauft. Dagegen wird aber die Assoziationsleistung bei unvollständig anliegenden Mustern erhöht.

In Abb. 5.6 sind die Konturlinien für die Wahl des minimalen Gewichts in Abhängigkeit von der Anzahl verzögerter Pulse eines Eingangsmusters mit fünf aktiven Eingängen und einer Verzögerung zwischen 0,1 μs und 1 μs dargestellt. Dabei entspricht die Verzögerung von fünf Pulsen einem ungestörten Pulsmuster. Das Gewicht w muss nur leicht nach oben angepasst werden, um asynchron einlaufende Pulsmuster zu berücksichtigen. Für den Fall, dass die verzögerten Pulse alleine ein Aktionspotential auslösen können (dieses ist gegeben, wenn der Grundstrom i hoch genug und der Verlustterm g_{leak} niedrig genug ist), zeigt sich eine besondere Eigenschaft der Gewichte. Die Berücksichtigung von wenigen

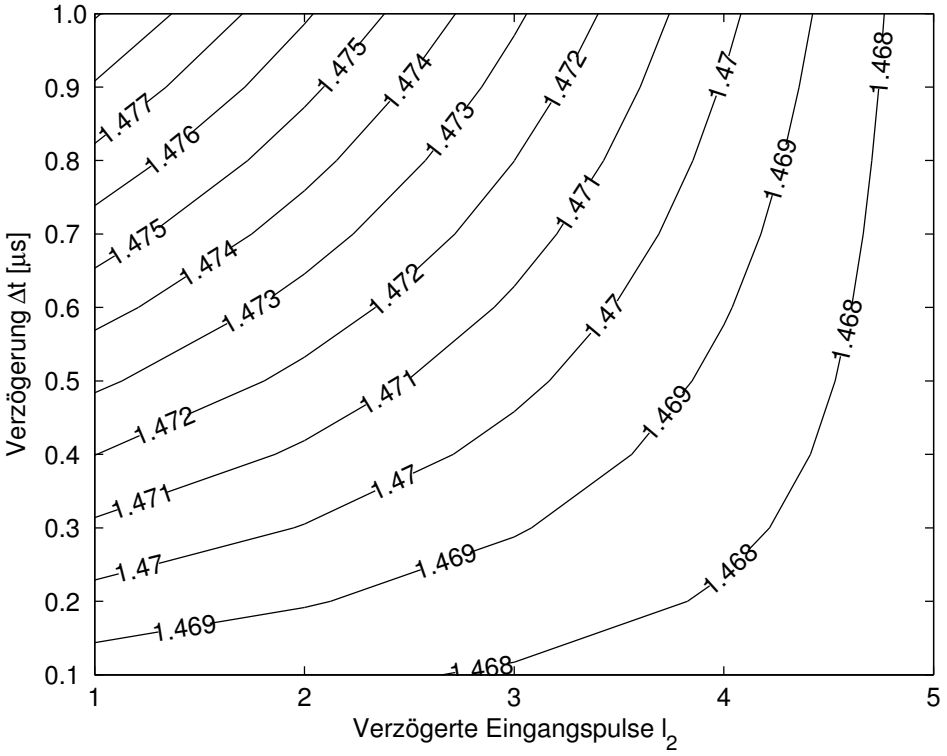


Abbildung 5.6: Konturlinien zur Wahl des minimalen Gewichts für asynchrone Eingangsmuster. Angelegt wurde ein Muster mit fünf aktiven Eingängen. Parameter: $g_{\text{leak}} = 10^{-9}\,\text{S}$, $t_{\text{fire}} = 1\,\mu\text{s}$, $C = 100\,\text{fF}$, $i = 10\,\text{nA}$, $U_{\text{TH}} = 730\,\text{mV}$

verzögerten Pulsen erfordert hier ein höheres Gewicht, als die Berücksichtigung von mehreren verzögerten Pulsen.

Problematisch ist ein erhöhtes Gewicht, wenn statt asynchroner Muster nun perfekte Muster einlaufen. Im Folgenden soll das maximal einstellbare Gewicht ermittelt werden, so dass gerade kein Muster fehlerhaft abgerufen wird, wenn ein korrektes Eingangsmuster synchron anliegt. Aus der Bedingung, dass mit l aktivierten Eingängen und $l-1$ besetzten Gewichten gerade kein Neuron zum Feuern gebracht wird, kann das maximale Gewicht $w_{\max}$ mit

$$w_{\max} \leq U_{\text{TH}} \cdot \frac{g_{\text{leak}}}{i} \cdot \frac{1}{(l-1)\left(1-\exp\left(-\frac{t_{\text{fire}}}{\tau}\right)\right)} \tag{5.52}$$

ermittelt werden.

Es kann gezeigt werden, dass das Gewicht nach (5.51) in allen Fällen kleiner ist, als das maximal erlaubte Gewicht $w_{\max}$. Für das Gewicht nach (5.50) ergibt sich daraus die Forderung

$$\Delta t < -\tau \ln\left(1 - \frac{1}{1-l}\right) \tag{5.53}$$

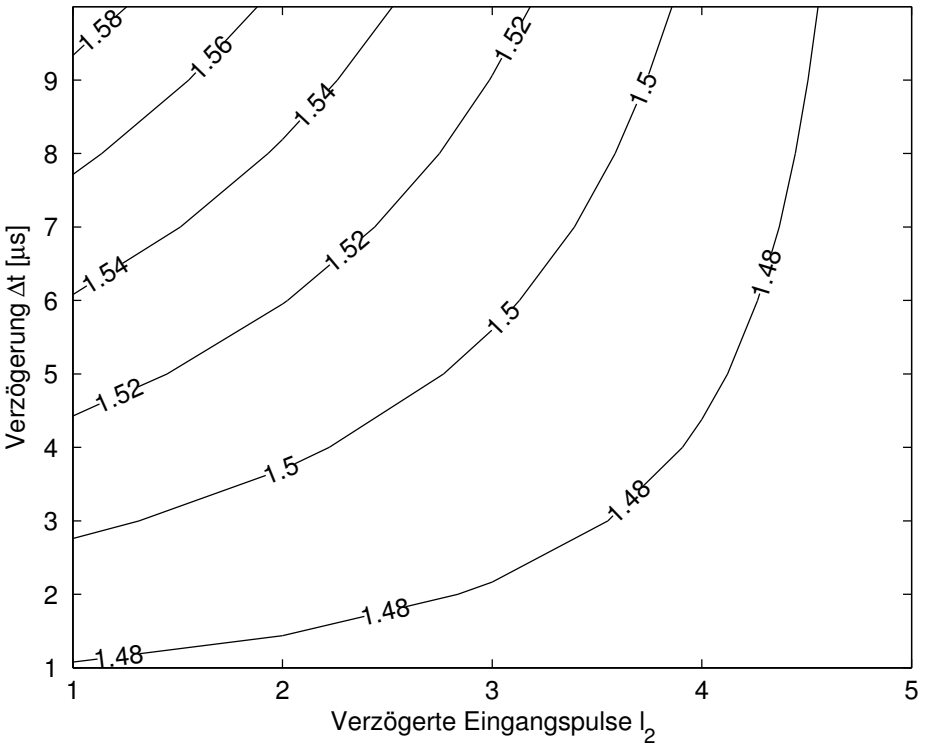


Abbildung 5.7: Konturlinien zur Wahl des minimalen Gewichts für asynchrone Eingangsmuster. Angelegt wurde ein Muster mit fünf aktiven Eingängen und einer Verzögerung der verspäteten Pulse von $\Delta t > t_{\text{fire}}$. Parameter: $g_{\text{leak}} = 10^{-9}\,\text{S}$, $t_{\text{fire}} = 1\,\mu\text{s}$, $C = 100\,\text{fF}$, $i = 10\,\text{nA}$, $U_{\text{TH}} = 730\,\text{mV}$

an die Verzögerungszeit, die in Anhang A.2 hergeleitet wird.

Große Verzögerungen im Eingangsmuster Wird die Verzögerung zwischen den ersten einlaufenden Pulsen l_1 und der verzögerten Menge l_2 über die Dauer eines Aktionspotentials hinaus erhöht ($\Delta t \geq t_{\text{fire}}$), muss der in der Pulspause entstehende Abklingterm des Membranpotentials berücksichtigt werden. Um das Aktionspotential nicht mit den ersten einlaufenden Pulsen auszulösen, sondern erst mit der zweiten Welle, darf das gewählte Gewicht nicht zu groß sein. Es gilt:

$$w < U_{\text{TH}} \cdot \frac{g_{\text{leak}}}{i} \cdot \left[(l - l_2) \cdot \left(1 - \exp\left(-\frac{t_{\text{fire}}}{\tau} \right) \right) \right]^{-1} \tag{5.54}$$

Gleichzeitig muss das Gewicht so groß gewählt werden, dass mit dem Einlaufen der zweiten Welle von Pulsen das Aktionspotential im empfangenden Neuron überschritten wird. Voraussetzung ist natürlich, dass die zweite einlaufende Menge an Pulsen ein Aktionspotential auslösen kann, d. h. $\frac{l_2 w i}{g_{\text{leak}}} > U_{\text{TH}}$ erfüllt ist. Es gilt:

$$w \geq U_{\mathrm{TH}} \cdot \frac{g_{\mathrm{leak}}}{i} \cdot \left[\left(1 - \exp\left(-\frac{t_{\mathrm{fire}}}{\tau}\right)\right) \cdot \left[l \cdot \exp\left(-\frac{\Delta t}{\tau}\right) + l_2 \cdot \left(1 - \exp\left(-\frac{\Delta t}{\tau}\right)\right)\right]\right]^{-1} \tag{5.55}$$

Beide Bedingungen für das Gewicht sind zu erfüllen und ergeben eine Anforderung an die Anzahl der „verspäteten“ Pulse l_2, welche von der Verzögerung Δt abhängig ist. In der Praxis sollte die Verzögerung Δt nicht größer als $t_{\mathrm{fire}} + 2\tau$ sein, da das Membranpotential in dieser Zeit bereits wieder auf weniger als 1/7 des maximal erreichbaren Wertes (der Feuerschwelle) zerfallen ist. In diesem Fall kann man nicht mehr von einer Bindung der später einlaufenden Pulse an die ersten eingelaufenen Pulse sprechen, sondern sie sollten als eigenständiges, fehlerbehaftetes Muster angesehen werden. Die betrachteten Neurone besitzen also ein zeitbehaftetes Gedächtnis des letzten Pulses, welches bereits von Gerstner [34] als *short-term memory* beschrieben wurde.

Die notwendige Bedingung für das Auslösen eines Aktionspotentials ergibt sich aus (5.54) und (5.55):

$$l_2 \geq \left\lceil l \cdot \frac{\left(1 - \exp\left(-\frac{\Delta t}{\tau}\right)\right)}{\left(2 - \exp\left(-\frac{\Delta t}{\tau}\right)\right)} \right\rceil \tag{5.56}$$

In Abb. 5.7 ist das minimal zu wählenden Gewicht für 5 Bit-Pulsmuster mit um mehr als eine Zeit von t_{fire} verzögerten Elementen dargestellt. Für große Leckleitwerte, z. B. $g_{\mathrm{leak}} = 10^{-6}\,\mathrm{S}$ ergibt sich der Fall, dass die Anpassung der Gewichte nicht mehr ausreicht und das asynchrone Muster nicht synchronisiert werden kann. Gleichung 5.48 muss in jedem Fall erfüllt werden. Auch ohne Beachtung dieser Bedingung ist sofort klar, dass – wenn eine bestimmte Anzahl von Pulsen kein Aktionspotential auslösen konnte – eine geringere Anzahl von später auftretenden Pulsen ebenfalls kein Aktionspotential auslösen wird. Daher ist für diesen Fall die Anzahl der „verspäteten“ Pulse mit mindestens $\lceil l/2 \rceil$ zu wählen, um zu einer Lösung zu kommen.

Abb. 5.8 zeigt den theoretischen Informationsgehalt des BiNAM sowie den Erwartungswert zusätzlicher, im Ausgangsmuster auftretender Einsen, die den Informationsgehalt nach oben begrenzen. Demgegenüber ist der aus Simulation eines BiNAM mit pulsenden LIAF Neuronen ermittelte Informationsgehalt sowie die auftretenden zusätzlichen Einsen über der Anzahl gespeicherter Muster für ein System mit 100×100 Gewichten und 5 aktiven Eingängen, sowie 2 aktiven Ausgängen dargestellt. Die Wahl der Feuerschwelle wird anhand der Anzahl aktiver Eingänge mit 5 festgelegt. Es wird ersichtlich, dass sich der Informationsgehalt des LIAF-BiNAM dem theoretischen Informationsgehalt bei zufälligen gespeicherten Mustern annähert. Durch die relativ geringe Größe des betrachteten Systems und der pseudo-zufälligen Erzeugung der Eingangsmuster liegt der durch Simulation ermittelte Informationsgehalt unterhalb des theoretischen Ergebnisses, bei dem von

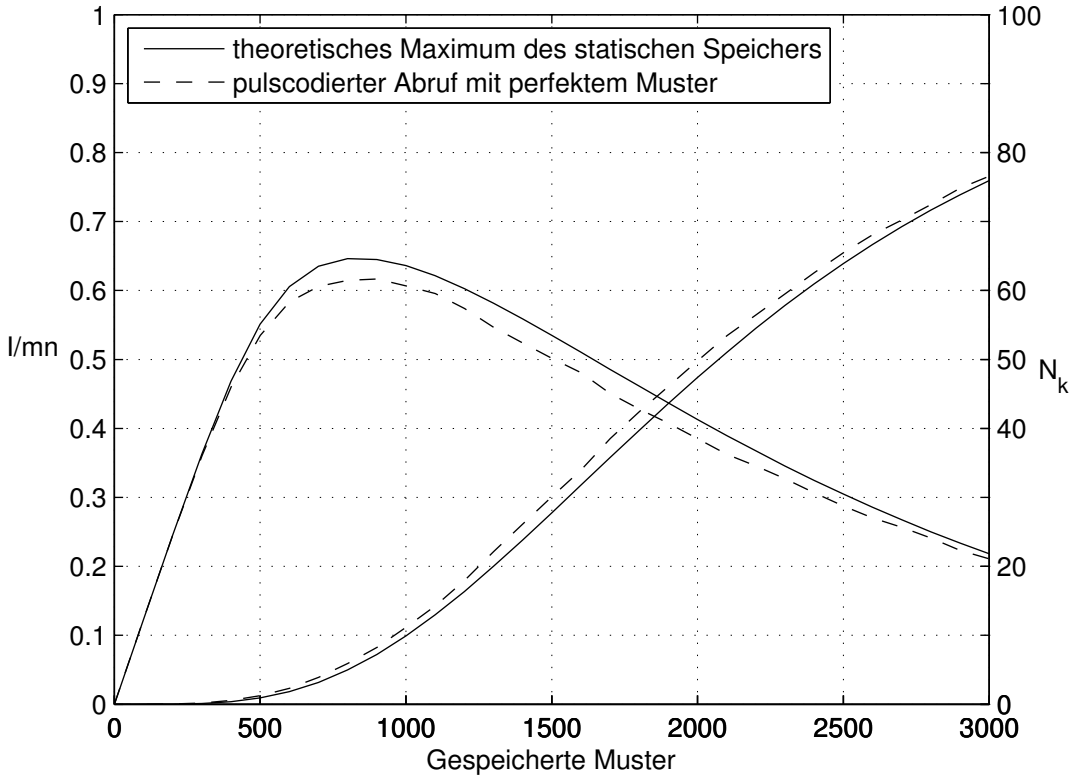


Abbildung 5.8: Vergleich des theoretischen Informationsgehalts des BiNAM und dem durch Simulation ermittelten Informationsgehalts des PCNN-BiNAM mit perfekten Mustern im Abruf (m, $n = 100$; $l = 5$; $k = 2$; $\Theta = l$).

zufällig verteilten Mustern ausgegangen wird. Es kann gezeigt werden, dass sich das am PCNN-BiNAM ermittelte Ergebnis für den Informationsgehalt bei Betrieb mit perfekten Mustern für große Systeme dem theoretischen Ergebnis des statischen BiNAM annähert. Dazu muss die Wahl der Gewichte der Gleichung (5.49) entsprechen und es müssen perfekte Muster für den Abruf vorausgesetzt werden. Mit der vorangegangenen Wahl der Gewichte ergibt sich nach Einlaufen eines Musters gerade ein Membranpotential, das der Feuerschwelle des Neurons entspricht, wenn alle durch den Eingangsvektor ausgewählten Zeilen einer Spalte (vgl. Abb. 5.1) mit einem nach (5.49) gewählten Gewicht ungleich 0 besetzt sind. Entscheidend für die Aktivierung eines Ausgangs sind in diesem Fall nur noch die Anzahl der ausgewählten Zeilen und die Besetzungswahrscheinlichkeit der Speichermatrix, so dass die Gleichungen zum des statischen Speichers auf den pulsenden Assoziativspeicher anwendbar werden. Dazu ist noch eine weitere Bedingung zu erfüllen. Die Neurone, die nicht zum Feuern gebracht werden, und somit nicht zu einem Muster gehören, bestimmen die maximal mögliche Frequenz des Musterabrufs. Ein Neuron, das nicht zum Feuern angeregt wurde, muss vor der erneuten Erregung das Membranpotential von

$$u_{\mathrm{mem}} = \frac{l-1}{l} \cdot U_{\mathrm{TH}}$$

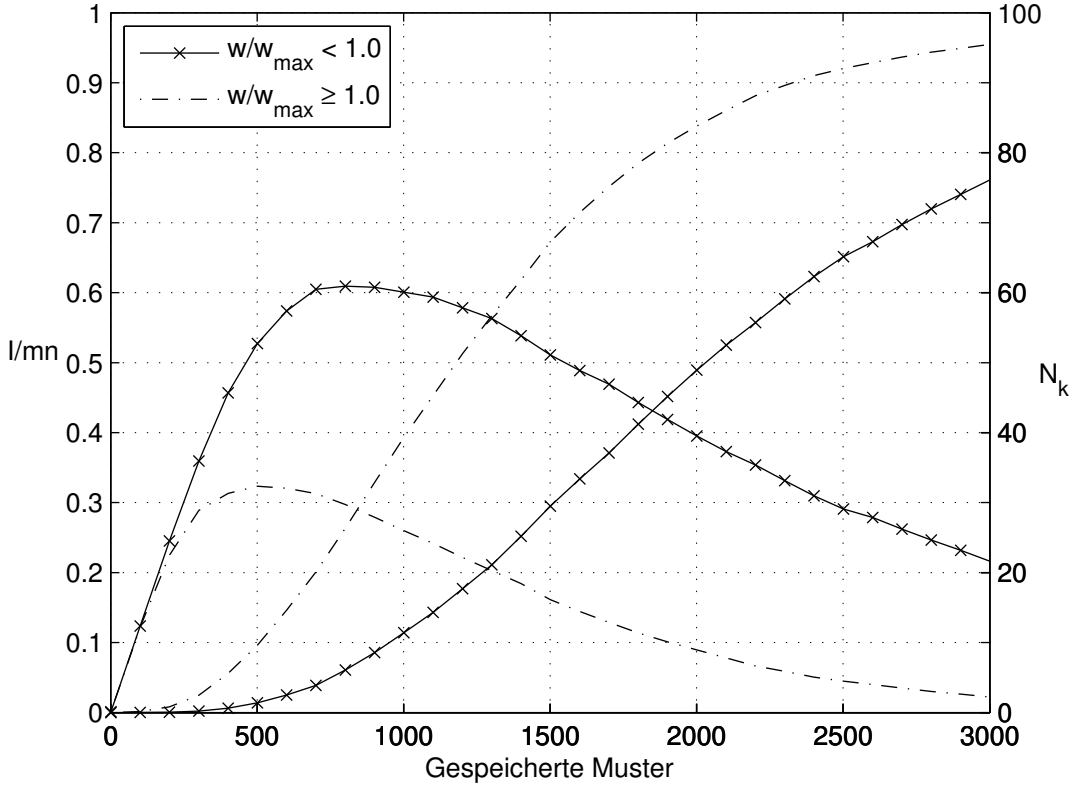


Abbildung 5.9: Informationsgehalt des PCNN-BiNAM ohne Korrekturfaktor und mit Korrekturfaktor bei perfekten Mustern (m, $n = 100$; $l = 5$; $k = 2$; $\Theta = l$).

abbauen, um im nächsten Abruf keinen Fehlerhaften Abruf zu erzeugen. Ein Zerfall auf 1% der Feuerschwelle ist an dieser Stelle ausreichend, so dass sich eine minimale Zeit zwischen zwei abgerufenen Mustern von

$$t_{\text{pause}} = -\tau \ln\left(0,01 \cdot \frac{l}{l-1}\right)$$

ergibt.

Zuletzt soll der Einfluss der in den vorhergegangenen Abschnitten ermittelten notwendigen Korrektur der Gewichte zur Verarbeitung von asynchronen Mustern betrachtet werden. Als Beispiel sei hier wieder ein System mit 100 Eingängen und 100 Ausgängen zu Grunde gelegt, bei welchem Muster mit 5 aktiven Eingängen und 2 aktiven Ausgängen betrachtet werden. Aus Abb. 5.6 wird für ein Pulsmuster mit einem maximal 250 ns verzögertem Anteil von 3 Pulsen bei 1 μs Pulsdauer der einlaufenden Pulse ein minimal für jedes Gewicht einzustellender Wert von 1,47 ermittelt. Der maximale Wert, der eingestellt werden kann, ohne zusätzliche Fehler beim Abruf zu erzeugen, liegt nach (5.52) bei einem Wert von 1,8341. Praktisch kann man sich die Korrektur als Einbringen zusätzlicher Einsen in die Gewichtsmatrix vorstellen, jedoch mit der Schwierigkeit, dass die zusätzlichen Einsen nicht in jeder Spalte auftreten, wie bei der Betrachtung des stuck-at-1

Fehlers modelliert, sondern vielmehr bereits stark mit Einsen besetzte Spalten durch zusätzliche Einsen weiter gefüllt werden. Im schlimmsten Fall treffen nun alle Pulse eines Musters zur gleichen Zeit ein, welches bei schwach besetzter Gewichtsmatrix zum korrekten Abruf führt. Bei einer stark mit Einsen besetzter Gewichtsmatrix werden jedoch auch die Neurone aktiviert, deren korrespondierende Spalte schwächer besetzt ist. Der Vorgang ist in seiner Auswirkung vergleichbar mit der Anpassung der Feuerschwelle in vorangegangenen Abschnitten, jedoch wirken hier durch die Gewichte und das zusätzlich betrachtete Zeitverhalten des Systems andere Mechanismen. In Abb. 5.9 ist dargestellt, wie sich der Informationsgehalt des Assoziativspeichers bei Anpassung der Gewichte zur Kompensation von asynchron einlaufenden Mustern reduziert. Die Reduktion des Informationsgehalts erfolgt bei Überschreiten eines durch (5.52) gegebenen Grenzwerts für die Verstärkung sprunghaft. Die Simulation wurde mit einem System der Größe 100×100 mit 5 aktiven Eingängen und 2 aktiven Ausgängen durchgeführt. Die dabei zu Grunde liegenden Neurone wurden mit den Parametern $g_{\mathrm{leak}} = 10^{-9}\,\mathrm{S}$, $t_{\mathrm{fire}} = 1\,\mu\mathrm{s}$, $C = 100\,\mathrm{fF}$, $i = 10\,\mathrm{nA}$, $U_{\mathrm{TH}} = 730\,\mathrm{mV}$ wie schon für die Berechnung für Abbildung 5.6 gewählt. Die Gewichte des ungestörten System wurden aus dieser Betrachtung nach (5.49) mit einem Wert von w=1,4673 versehen. Der maximal einstellbare Wert für eine Synapse für einen fehlerfreien Abruf von Mustern ergibt sich mit den angegebenen Parametern nach (5.52) zu w_{max} =1,8341. Darüber ergeben sich Fehler im Abruf, welche den Informationsgehalt des Speichers reduzieren. Für die weiteren Simulationen wurde der Gewichtswert des ungestörten Systems mit einem um einen Faktor von 1,249 und 1,25 verstärken Gewicht durchgeführt, um sich der Grenze der Informationsgehaltsreduktion zu nähern. Die um den Faktor von 1,25 verstärkten Gewichte überschreiten den Maximalwert, der nach (5.52) bestimmt wurde. Nach der Programmierung der Speichermatrix wurden die zuvor gespeicherten zufälligen Muster mit gespeicherten, perfekten Mustern abgerufen, was den schlechtesten Fall für das System mit verstärkten Gewichten darstellt. Bis zu einer Verstärkung des Gewichts um einen Wert von knapp unter 1,25 ($w/w_{\mathrm{max}} < 1{,}0$) verändert sich der Informationsgehalt des Speichers nicht. Oberhalb dieses Werts ($w/w_{\mathrm{max}} \geq 1{,}0$) geht der maximale Informationsgehalt für dieses System sprungartig auf fast die Hälfte zurück. Das bedeutet, dass es eine maximale zeitliche Ausdehnung der Muster gibt, die ohne Fehler bzw. ohne Reduktion des Informationsgehalts des Assoziativspeichers mit pulsenden Neuronen durch Adaption der Gewichte kompensiert werden kann.

Zusammenfassung

Diese Arbeit befasst sich mit der ressourceneffizienten Implementierung pulscodierter neuronaler Netze, insbesondere der Umsetzung der einzelnen Komponenten von PCNN in aktuellen CMOS-Technologien. Als Ressourcen wurden hier die durch das Layout festgelegte Fläche in einem ASIC bzw. Logikzellen auf einem FPGA sowie die im Betrieb auftretende Verlustleistung in Form von statischer Verlustleistung, die der Ruheleistung zugeordnet wurde, und der dynamischen Verlustleistung, die zur umgesetzten Energie pro Puls umgerechnet wurde, betrachtet. Ausgangspunkt ist die Betrachtung der Leistung des menschlichen Gehirns, das mit seinen vielen Verarbeitungseinheiten unter dem Einsatz von vergleichsweise wenig Energie komplexe Probleme lösen kann. Besonders die hohe Robustheit von Teilen des Gehirns, das trotz Ausfall von kleineren Teilbereichen eine weiterhin korrekte Funktion bietet, ist ein Vorbild für die Integration von robusten mikroelektronischen Schaltungen in immer kleiner werdenden CMOS-Technologien.

Ausgehend von der Motivation des Einsatzes neuronaler Prinzipien in mikroelektronischen Schaltungen wurde in dieser Arbeit eine Einführung in die Grundlagen der biologischen Zelle, insbesondere der Zellmembran und der dort wirkenden Transportmechanismen von Ionen gegeben und der Ort des maßgeblichen Energieumsatzes in einer Nervenzelle beschrieben. Der Ausgleichsmechanismus der Natrium-Kalium-Pumpe zum Erhalt des intrazellulären Volumens wurde besonders betrachtet. Hier wird der Hauptteil der Energie aus dem universellen Energieträger ATP zum Transport von Ionen aus der Zelle heraus und in die Zelle hinein genutzt, um ein auftretendes Ungleichgewicht der Ionenkonzentrationen beteiligter Ionen auszugleichen und das Zellvolumen stabil zu halten. Um den Energieumsatz eines biologischen Neurons abschätzen zu können, wurde in Kap. 3 ein mathematisches Modell des Ionentransports an der Zellmembran aufgegriffen und um eine regelungstechnische Beschreibung der Natrium-Kalium-Pumpe als klassischer Zustands-Regler erweitert. Mit Hilfe des Reglers und der Anwendung der Methode der Polvorgabe konnten die passiven Ausgleichsvorgänge, die durch das mathematische Modell der Zellmembran beschrieben werden, ausgeregelt werden und ein stabiler Arbeitspunkt bei biologisch plausiblen Konzentrationen erreicht werden. Aus den Angaben des Reglers zu den notwendigen Pumpzyklen zum Ausgleich des passiven Ionentransports kann die umgesetzte Energie direkt errechnet werden, so dass diese Modellierung es erlaubt, den Energieumsatz einzelner Neurone und größerer Systeme durch Simulation zu ermitteln. Weiter ist es möglich, auch den Energieumsatz der Ausgleichsvorgänge im Unterschwellenverhalten, also vor dem Auslösen eines Aktionspotentials einzelner Neurone zu ermitteln.

In der Literatur finden sich zum Energieumsatz im Unterschwellenverhalten des Neurons keine Angaben, dort wird vereinzelt die Energie für ein einzelnes Aktionspotential abgeschätzt. Die Stabilität des geregelten nichtlinearen Systems der Zellmembran wurde anhand von Simulationen verifiziert und durch die Betrachtung der Stabilitätsbedingung nach Ljapunov nachgewiesen.

Die Akzeptanz und die Anwendung neuronaler Prinzipien in mikroelektronischen Schaltungen ist direkt von der Möglichkeit abhängig, die Komponenten ressourcenschonend, d. h. mit einer möglichst geringen Fläche und einer möglichst geringen Verlustleistung im Betrieb in CMOS-Technologien von 130 nm und darunter zu implementieren. Die grundsätzlich zu klärenden Fragen, welche Zielplattform, FPGA oder ASIC, und welche Technologie zum Einsatz kommen soll, wurden durch theoretische Betrachtungen, Analyse von Syntheseergebnissen und durch die Implementierung der in dieser Arbeit vorgestellten Strukturen beantwortet. Zur Hinführung in die unterschiedlichen Implementierungsvarianten wurden in Kap. 2 veröffentlichte Umsetzungen für digitale Systeme auf FPGAs, vereinzelt ASICs sowie Beispiele für die in dieser Arbeit verwendeten Techniken (bitparallel und bitseriell) diskutiert. Die relevanten Implementierungen analoger LIAF Neurone wurden gezeigt und analysiert. Bei der Recherche wurde deutlich, dass Vergleichswerte für Flächenbedarf und Verlustleistung nur in den seltensten Fällen angegeben werden, so dass ein Vergleich der erreichten Werte mit bestehenden Systemen schwierig ist.

In Kap. 4 dieser Arbeit konnte gezeigt werden, dass analog implementierte Neurone in einer aktuellen 130 nm CMOS-Technologie für eine flächeneffiziente Umsetzung von pulsenden Neuronen geeigneter sind, als ihre digitalen Pendants. Das Auflösungsvermögen des Entscheidungselements der analogen Umsetzung wurde durch Betrachtung der Prozessvariation im 130 nm CMOS-Prozess mit maximal 6 Bit ermittelt. Davon ausgehend wurden die digitalen Systeme, sofern möglich, bei einer äquivalenten Auflösung von 6 Bit betrachtet und der analogen Implementierung gegenübergestellt. Um die Skalierung der digital implementierten Neurone zu überprüfen, wurden zusätzliche Synthesen auf Wortbreiten von 3 Bit bis 16 Bit durchgeführt. Die vorgestellte analoge Umsetzung benötigt gegenüber der kleinsten digitalen Umsetzung nur etwa 1/18 der Fläche, womit sich die analoge Umsetzung für die Integration sehr vieler LIAF Neurone auf einem ASIC anbietet. Die im analogen Neuron durch Messung an gefertigten Chips ermittelte umgesetzte Energie pro Puls von 17 pJ steht der umgesetzten Energie von 40 pJ des zählerbasierten digitalen Neurons gegenüber. Die statische Verlustleistung bei beiden Implementierungen liegt mit 1,9 μW beim analogen Neuron über der statischen Verlustleistung von ca. 1,1 μW des digitalen zählerbasierten Neurons. Der Hauptunterschied bei beiden Implementierungen liegt in der zu erreichenden Ausgangspulsrate. Durch die Normierung auf die Energie pro Puls wurde dieser Unterschied berücksichtigt. Durch die in Kapitel 4.3 vorgeschlagenen Anpassungen des analogen LIAF Neurons kann die Verlustleistung weiter abgesenkt und die Robustheit der Schaltung erhöht werden.

Die digitalen Neurone zeichnen sich gegenüber den analogen LIAF Neuronen, in denen immer wiederkehrende Komponenten zu finden sind, durch einen erhöhten Variantenreichtum in der Umsetzung aus. Die Umsetzung von bitparallel arbeitenden Elementen für ein

Tabelle 5.2: Energie-, Flächenbedarf und Eigenschaften der Varianten der Neurone

Implementierung	Energie pro AP	Ruheleistung	Fläche
Biologisches Neuron (Purkinje Zelle)[a]	25 pJ – 40 pJ	26 pW	20.196 μm^2
Biophysikalisches Neuronenmodell	34 pJ	2,4 fW[b]	
Digitales Neuron (parallel, 130 nm)	49 pJ	10 μW	25.756 μm^2
Digitales Neuron (bitseriell, 130 nm)	386 pJ	5,5 μW	17.568 μm^2
Digitales Neuron (Zähler, 130 nm)	40 pJ	1,1 μW	1.398 μm^2
Analoges Neuron (130 nm)	17 pJ	1,9 μW	76 μm^2

[a] Als Fläche wird die Oberfläche einer kugelförmigen Zelle angenommen. Diese Größe dient nur dem ungefähren Vergleich und ist stellt keine Angabe über die tatsächliche Oberfläche der Purkinje Zelle dar.
[b] Die angegebene Ruheleistung liegt im Vergleich mit der Energie pro Aktionspotential zu niedrig.

LIAF Neuron kann zwar gut auf die Strukturen eines FPGAs abgebildet werden, ist jedoch für die Umsetzung auf einem ASIC durch den hohen Flächenbedarf bitparalleler Multiplizierer ungeeignet. Die Verringerung der Fläche durch das Einbringen von bitseriellen Elementen, vor allem dem Ersatz der Multiplizierer, kann bei den vorgestellten Implementierungen praktisch erst ab einer Wortbreite von 6 Bit erreicht werden. Darunter benötigt der bitparallele Multiplizierer eine kleinere Fläche. Gleichzeitig muss für die bitserielle Umsetzung der Takt der Schaltung erhöht werden, um die gleiche Ausgangscharakteristik, vor allem eine gleiche Ausgangspulsrate zu erreichen, wie die bitparallele Umsetzung. Der Energiebedarf der verschiedenen Implementierungen im Vergleich mit der biologischen Zelle ist in Tab. 5.2 zusammengefasst. Vor allem die analoge Implementierung erlaubt aufgrund ihrer kleinen Fläche die Integration vieler LIAF Neurone auf einem Chip. Die angegebene umgesetzte Energie pro Aktionspotential unterschreitet in fast allen Fällen die vom biologischen Neuron benötigte Energie. Die besonders niedrige Ruheleistung des biologischen Vorbilds kann jedoch von den vorgestellten Implementierungen nicht erreicht werden. Die weitere Absenkung der Verlustleistung konnte durch den Aufbau und die Verwendung einer speziellen Synthesebibliothek erreicht werden, die mit Versorgungsspannungen im Subschwellenbereich von 200 mV bis hinauf zu 1 V arbeiten kann. Die Bibliothek wurde in Kap. 4.2.2 kurz vorgestellt, und die einzelnen Systeme wurden zum Vergleich darauf abgebildet.

Abschließend wurde der Einsatz von pulsenden Neuronen am Beispiel eines robusten Speichers gezeigt. Die Verwendung des binären neuronalen Assoziativspeichers als fehlerkorrigierender Speicher für zukünftige Implementierungen von Speichern in CMOS-Technologien mit Strukturen von 130 nm und darunter wurde durch Betrachtung seiner Eigenschaften und seiner Robustheit gegenüber Parameterschwankungen und Fehlern motiviert. Ein Wechsel vom kontinuierlich betriebenen Speicher zu einem Speicher mit pulsenden Neuronen stellt neue Herausforderungen an den Speicher. Der Abruf der Daten wird durch nicht vollständig synchron einlaufende Pulse am Eingang des Speichers erschwert. Es wurde untersucht, welchen Einfluss asynchron einlaufende Pulse auf den Abruf von Daten aus dem Speicher haben, und wie die Gewichte des Speichers angepasst werden

müssen, um eine korrekte Funktion des Speichers zu gewährleisten. Unter Berücksichtigung der in Kap. 5.2 angegebenen unteren Grenzen der Gewichte ist die Anwendung der zuvor für den statischen Speicher ermittelten Abschätzungen für den Informationsgehalt des Speichers möglich. Da die in Kap. 5 gezeigte Anpassung der Gewichte Einfluss auf den Informationsgehalt des Speichers hat, wurde der Einfluss der Gewichtsanpassung zur Verarbeitung von asynchronen Pulsen anhand von Simulationen ermittelt und dargestellt. Gleichzeitig wurde eine obere Grenze für die Gewichtsanpassung hergeleitet, unterhalb deren Wert der Informationsgehalt des Speichers nicht verringert wird. Die Eignung von pulsenden Neuronen für die Implementierung eines BiNAM konnte in diesem letzten Kapitel erfolgreich gezeigt werden.

Anhang A

Mathematischer Anhang

In diesem Anhang sind Rechnungen aufgeführt, welche den Lesefluss der vorliegenden Arbeit unnötig unterbrochen hätten. Sie sind zur Darstellung der gewonnenen Ergebnisse im Hauptteil dieser Arbeit nicht zwingend erforderlich, tragen aber zum Verständnis und zur Nachvollziehbarkeit der angegebenen Gleichungen bei. An den entsprechenden Stellen im Text wurde auf den jeweiligen Anhang referenziert.

A.1 Herleitung von $u_{c,f}^{(N)}$ und $u_{c,T}^{(N)}$

Grundlage für die Berechnung des Potentials einer Membrankapazität nach unendlich vielen Aufladungen bilden die Differentialgleichungen eines IAF Neurons mit der Dynamik nach (3.48) und (3.52). Mit der Annahme, dass dieses Neuron mit einer konstanten Pulsrate der Frequenz $f = 1/T$ erregt wird, lässt sich die nachfolgende Rechnung ausführen. Diese Bedingung ist vor dem Hintergrund der präsynaptischen Erregung des empfangenden Neurons dann erfüllt, wenn vorausgesetzt wird, dass die präsynaptischen Neurone uniform, d. h. gleichmäßig feuern, und nur eine Pulsfolge mit mittlerer Pulsrate zum Auslösen eines Aktionspotentials am postsynaptischen Neuron führen kann. Die Periodendauer T der Pulse setzt sich aus den zwei Anteilen der Pulsrate zusammen, der Feuerdauer eines präsynaptischen Neurons t_{fire} und der Pausenzeit t_{relax}. An dieser Stelle wird die Aufladung einer Membrankapazität durch einen konstanten Strom betrachtet, welcher im Falle eines präsynaptischen Aktionspotentials in das postsynaptische Neuron injiziert wird. Die Membrankapazität des Neurons sei zu Beginn der Betrachtung auf einen Wert $u_c(t = 0) = u_0$ aufgeladen. Nach (3.53) ergibt sich nach dem ersten Puls auf der Membrankapazität ein Potential von:

$$u_{c,f}^{(1)} = u_c(t = t_{\text{fire}}) = \left(u_0 - \frac{I}{g_{\text{leak}}}\right) \cdot \mathrm{e}^{-\frac{t_{\text{fire}}}{\tau}} + \frac{I}{g_{\text{leak}}} \quad \text{mit} \quad \tau = \frac{C}{g_{\text{leak}}} \tag{A.1}$$

In der Pause zwischen konsekutiven präsynaptischen Pulsen klingt das Membranpotential

nach (3.49) durch passive Entladevorgänge ab und erreicht nach einer gesamten Periode T den Wert:

$$u_{c,T}^{(1)} = u_{c,f}^{(1)} \cdot \mathrm{e}^{-\frac{t_{\mathrm{relax}}}{\tau}} = \left(u_0 - \frac{I}{g_{\mathrm{leak}}}\right) \cdot \mathrm{e}^{-\frac{T}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \cdot \mathrm{e}^{-\frac{t_{\mathrm{relax}}}{\tau}} \tag{A.2}$$

Dieser Wert ist nun wieder als Startwert für $u_{c,f}^{(2)}$ einzusetzen. Nach dem zweiten Puls ergibt sich also:

$$\begin{aligned} u_{c,f}^{(2)} &= \left(\left(u_0 - \frac{I}{g_{\mathrm{leak}}}\right) \cdot \mathrm{e}^{-\frac{T}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \cdot \mathrm{e}^{-\frac{t_{\mathrm{relax}}}{\tau}} - \frac{I}{g_{\mathrm{leak}}}\right) \cdot \mathrm{e}^{-\frac{t_{\mathrm{fire}}}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \\ &= \left(u_0 - \frac{I}{g_{\mathrm{leak}}}\right) \cdot \mathrm{e}^{-\frac{T+t_{\mathrm{fire}}}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \cdot \mathrm{e}^{-\frac{T}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \left(1 - \mathrm{e}^{-\frac{t_{\mathrm{fire}}}{\tau}}\right) \end{aligned} \tag{A.3}$$

Dieses Ergebnis ist nun wieder für $u_{c,T}^{(2)}$ einzusetzen, usw.

$$\begin{aligned} u_{c,T}^{(2)} &= \left(\left(u_0 - \frac{I}{g_{\mathrm{leak}}}\right) \cdot \mathrm{e}^{-\frac{T+t_{\mathrm{fire}}}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \cdot \mathrm{e}^{-\frac{T}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \left(1 - \mathrm{e}^{-\frac{t_{\mathrm{fire}}}{\tau}}\right)\right) \cdot \mathrm{e}^{-\frac{t_{\mathrm{relax}}}{\tau}} \\ &= \left(u_0 - \frac{I}{g_{\mathrm{leak}}}\right) \cdot \mathrm{e}^{-\frac{2T}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \cdot \mathrm{e}^{-\frac{T+t_{\mathrm{relax}}}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \left(1 - \mathrm{e}^{-\frac{t_{\mathrm{fire}}}{\tau}}\right) \cdot \mathrm{e}^{-\frac{t_{\mathrm{relax}}}{\tau}} \end{aligned} \tag{A.4}$$

$$\begin{aligned} u_{c,f}^{(3)} = &\left(u_0 - \frac{I}{g_{\mathrm{leak}}}\right) \cdot \mathrm{e}^{-\frac{2T+t_{\mathrm{fire}}}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \cdot \mathrm{e}^{-\frac{2T}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \left(1 - \mathrm{e}^{-\frac{t_{\mathrm{fire}}}{\tau}}\right) \cdot \mathrm{e}^{-\frac{T}{\tau}} \\ &- \frac{I}{g_{\mathrm{leak}}} \cdot \mathrm{e}^{-\frac{t_{\mathrm{fire}}}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \end{aligned} \tag{A.5}$$

Dieser Term lässt sich nun zusammenfassen, so dass im Folgenden die Reihenentwicklung deutlich wird:

$$\begin{aligned} u_{c,f}^{(3)} = &u_0 \cdot \mathrm{e}^{-\frac{2T+t_{\mathrm{fire}}}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \left(1 - \mathrm{e}^{-\frac{t_{\mathrm{fire}}}{\tau}}\right) \cdot \mathrm{e}^{-\frac{2T}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \left(1 - \mathrm{e}^{-\frac{t_{\mathrm{fire}}}{\tau}}\right) \cdot \mathrm{e}^{-\frac{T}{\tau}} \\ &+ \frac{I}{g_{\mathrm{leak}}} \cdot \left(1 - \mathrm{e}^{-\frac{t_{\mathrm{fire}}}{\tau}}\right) \end{aligned} \tag{A.6}$$

Nur während einer Aufladung kann die Feuerschwelle von unten überschritten werden, deshalb wird an dieser Stelle nur die Aufladung nach dem N-ten präsynaptischen Puls betrachtet. Dazu wird der Wert $u_{c,f}^{(N)}$ bestimmt, der sich durch sukzessives Einsetzen der oben beschriebenen Auflade- und Entladevorgänge ergibt. Die Exponentialterme lassen

sich zu einer Summe zusammenfassen, so dass sich für das Potential nach dem N-ten präsynaptischen Aktionspotential ein Membranpotential von

$$u_{c,f}^{(N)} = u_0 \cdot \mathrm{e}^{-\frac{(N-1)T+t_{\mathrm{fire}}}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \cdot \left(1 - \mathrm{e}^{-\frac{t_{\mathrm{fire}}}{\tau}}\right) \cdot \sum_{n=0}^{N-1} \mathrm{e}^{-\frac{nT}{\tau}} \tag{A.7}$$

ergibt.

Für $N \to \infty$ lässt sich diese Summe in den Grenzwert für eine geometrische Reihe entwickeln, welcher sich mit $\frac{\mathrm{e}^{\frac{T}{\tau}}}{\mathrm{e}^{\frac{T}{\tau}}-1}$ angeben lässt. Der erste Term der Gleichung A.7 verschwindet in diesem Fall für endliche Werte von u_0.

In gleicher Weise lässt sich der Wert für $u_{c,T}^{(N)}$ ermitteln, der sich wie folgt ergibt:

$$\begin{aligned} u_{c,T}^{(N)} &= u_0 \cdot \mathrm{e}^{-\frac{NT}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \cdot \left(\mathrm{e}^{+\frac{t_{\mathrm{fire}}}{\tau}} - 1\right) \cdot \sum_{n=0}^{N-1} \mathrm{e}^{-\frac{(n+1)T}{\tau}} \\ &= u_0 \cdot \mathrm{e}^{-\frac{NT}{\tau}} + \frac{I}{g_{\mathrm{leak}}} \cdot \left(\mathrm{e}^{+\frac{t_{\mathrm{fire}}}{\tau}} - 1\right) \cdot \mathrm{e}^{-\frac{T}{\tau}} \cdot \frac{\mathrm{e}^{-N\frac{T}{\tau}} - 1}{\mathrm{e}^{-\frac{T}{\tau}} - 1} \end{aligned} \tag{A.8}$$

A.2 Herleitung des maximalen Gewichts zum fehlerfreien Abruf

Gleichung 5.50 kann umgeschrieben werden zu:

$$w \geq U_{\mathrm{TH}} \cdot \frac{g_{\mathrm{leak}}}{i} \cdot \left[(l - l_2) \cdot \left(1 - \exp\left(-\frac{t_{\mathrm{fire}}}{\tau}\right)\right) + l_2 \cdot \left(1 - \exp\left(-\frac{t_{\mathrm{fire}} - \Delta t}{\tau}\right)\right)\right]^{-1} \tag{A.9}$$

Gleichung 5.51 kann umgeschrieben werden zu:

$$w \geq U_{\mathrm{TH}} \cdot \frac{g_{\mathrm{leak}}}{i} \cdot \left[l \cdot \left(1 - \exp\left(-\frac{t_{\mathrm{fire}}}{\tau}\right)\right) \cdot \exp\left(-\frac{\Delta t}{\tau}\right) + l_2 \cdot \left(1 - \exp\left(-\frac{\Delta t}{\tau}\right)\right)\right]^{-1} \tag{A.10}$$

Gleichzeitig muss das Gewicht der Bedingung (5.52) genügen:

$$w_{\max} \leq U_{\mathrm{TH}} \cdot \frac{g_{\mathrm{leak}}}{i} \cdot \left[(l-1)\left(1 - \exp\left(-\frac{t_{\mathrm{fire}}}{\tau}\right)\right)\right]^{-1} \tag{A.11}$$

In Kapitel 5.2 wurde bereits darauf eingegangen, dass mit der Wahl von $l_2 = 1$ das maximale Gewicht ermittelt wird. Durch direkten Vergleich der Ungleichungen (A.9) und

(A.11) ergibt sich, dass $w < w_{\max}$ durch den zusätzlichen Term $l_2 \cdot \left(1 - \exp\left(-\frac{t_{\text{fire}} - \Delta t}{\tau}\right)\right)$ in der Ungleichung (A.10) für alle Δt gegeben ist.

Für das Gewicht nach (A.9) muss ausgewertet werden, ob die Bedingung

$$\begin{aligned}&(l - l_2) \cdot \left(1 - \exp\left(-\frac{t_{\text{fire}}}{\tau}\right)\right) + l_2 \cdot \left(1 - \exp\left(-\frac{t_{\text{fire}} - \Delta t}{\tau}\right)\right) \\ &\geq (l-1) \cdot \left(1 - \exp\left(-\frac{t_{\text{fire}}}{\tau}\right)\right)\end{aligned} \tag{A.12}$$

erfüllt ist.

Diese Ungleichung kann nach

$$\Delta t \leq -\tau \ln\left(1 - \frac{1}{l-1}\right) \tag{A.13}$$

aufgelöst werden.

A.3 Variation des Störabstands in einer 90 nm ultra-low-power Standardzellenbibliothek

Da MOS-Transistoren im Subschwellenbereich und in kleinen Strukturgrößen höheren Prozessvariationen unterliegen, soll im Folgenden der Einfluss der Variation auf die Bibliothekszellen untersucht werden. Die Betrachtung wird beispielhaft am Inverter durchgeführt, lässt sich aber direkt auf andere Zellen übertragen.

Die Robustheit der Standardzellenbibliothek soll maximiert werden. Als Maß für die Robustheit eines Gatters und die Analyse der Ausbeute einer Schaltung kommt der Störabstand (Noise Margin, NM) zum Einsatz. Als Grenze für die Ausbeute wird ein erlaubter Störabstand von 20% U_{DD} angenommen.

Die Berechnungen in diesem Abschnitt erfolgen nach dem EKV Modell [27], durch das die Drainströme eines nMOS-Transistors und eines pMOS-Transistors durch die Gleichungen

$$I_{\text{D, n}} = 2 n_{\text{n}} U_{\text{T}}^2 \mu_{\text{n}} C'_{\text{ox}} \frac{W}{L} \exp\left(\frac{U_{\text{G}} - U_{\text{TH0,n}}}{n_{\text{n}} U_{\text{T}}}\right) \left(\exp\left(\frac{U_{\text{S}}}{U_{\text{T}}}\right) - \exp\left(-\frac{U_{\text{D}}}{U_{\text{T}}}\right)\right) \tag{A.14}$$

$$I_{\text{D, p}} = 2 n_{\text{p}} U_{\text{T}}^2 \mu_{\text{p}} C'_{\text{ox}} \frac{W}{L} \exp\left(-\frac{U_{\text{G}} - U_{\text{TH0,p}}}{n_{\text{p}} U_{\text{T}}}\right) \left(\exp\left(-\frac{U_{\text{S}}}{U_{\text{T}}}\right) - \exp\left(\frac{U_{\text{D}}}{U_{\text{T}}}\right)\right) \tag{A.15}$$

beschrieben werden.

Der Drainstrom des nMOS-Transistors ist durch die physikalischen Parameter W_n und L_n (Weite und Länge) des Transistors, die Technologiekonstanten n_{n} und die Ladungsträgerbeweglichkeit μ_{n}, die Temperaturspannung $U_{\text{T}} = \text{k}_{\text{B}} T/\text{q}$, die Schwellenspannung $U_{\text{TH0,n}}$ und

die Drain-, Source- und Gate-Potentiale gegenüber dem Substratpotential gegeben. Der Drainstrom des pMOS-Transistors ist über die korrespondierenden Technologieparameter gegeben.

Nimmt man an, dass die Kanallänge des nMOS-Transistors und des pMOS-Transistors gleich gewählt wird, kann der Skalierungsfaktor s für die Weite des pMOS-Transistors aus der Forderung nach einer symmetrischen Übertragungskennlinie bestimmt werden:

$$W_\mathrm{p} = s \cdot W_\mathrm{n} \quad \text{, dass} \quad U_\mathrm{out}(U_\mathrm{in} = U_\mathrm{DD}) = U_\mathrm{DD}/2$$

Es kann gezeigt werden, dass eine symmetrische Kennlinie zu einem maximalen Störabstand führt. Der Faktor s hängt von den Technologieparametern ab und ergibt sich zu

$$s = \frac{n_\mathrm{n}\mu_\mathrm{n} \exp\left(\frac{U_\mathrm{DD}/2 - U_\mathrm{TH0,n}}{n_\mathrm{n} U_\mathrm{T}}\right)}{n_\mathrm{p}\mu_\mathrm{p} \exp\left(\frac{U_\mathrm{DD}/2 + U_\mathrm{TH0,p}}{n_\mathrm{p} U_\mathrm{T}}\right)}. \tag{A.16}$$

Wir nehmen an, dass die Weite des pMOS-Transistors entsprechend dieses Zusammenhangs gewählt, und so der Störabstand maximiert wurde. Nun wird ermittelt, welchen Einfluss eine Änderung der Schwellenspannung ΔU_TH auf den Störabstand hat. Mit (A.14) und (A.15) kann die Übertragungsfunktion $U_\mathrm{out}(U_\mathrm{in})$ bestimmt werden:

$$\begin{aligned} U_\mathrm{out} = \; & \tfrac{U_\mathrm{DD}}{2} - U_\mathrm{T}\ln 2 + U_\mathrm{T}\ln\Bigg(\exp\left(\tfrac{U_\mathrm{DD}}{2U_\mathrm{T}}\right) \\ & - \exp\left(\tfrac{3U_\mathrm{DD} - 4U_\mathrm{in} + 4\Delta U_\mathrm{TH}}{2U_\mathrm{T}}\right) \\ & + \Big[\exp\left(\tfrac{U_\mathrm{DD}}{U_\mathrm{T}}\right) - 2\exp\left(\tfrac{2(U_\mathrm{DD} - U_\mathrm{in} + \Delta U_\mathrm{TH})}{U_\mathrm{T}}\right) \\ & + \exp\left(\tfrac{3U_\mathrm{DD} - 4U_\mathrm{in} + 4\Delta U_\mathrm{TH}}{U_\mathrm{T}}\right) \\ & + 4\exp\left(\tfrac{U_\mathrm{DD} - 2U_\mathrm{in} + 2\Delta U_\mathrm{TH}}{U_\mathrm{T}}\right)\Big]^{-1/2}\Bigg) \end{aligned} \tag{A.17}$$

Der Störabstand dieses Inverters ist durch die charakteristischen Punkte der Funktion

$$\frac{dU_\mathrm{in}}{dU_\mathrm{out}} = -1.$$

gegeben. Die Lösung dieser Gleichung führt zu den Punkten $U_\mathrm{in,l}$ and $U_\mathrm{in,h}$ für die Eingangsspannung (A.18), an denen die Übertragungskennlinie des Inverters Richtung Masse mit einer Steigung von -1 abfällt:

$$\begin{aligned} U_\mathrm{in,l/h} = \; & \Delta U_\mathrm{TH} + \tfrac{1}{2}\ln\Bigg(-\tfrac{8}{3} + \tfrac{5}{3}\left(\exp\left(\tfrac{U_\mathrm{DD}}{2U_\mathrm{T}}\right)\right)^2 \\ & \mp \tfrac{4}{3}\sqrt{4 - 5\left(\exp\left(\tfrac{U_\mathrm{DD}}{2U_\mathrm{T}}\right)\right)^2 + \left(\exp\left(\tfrac{U_\mathrm{DD}}{2U_\mathrm{T}}\right)\right)^4}\Bigg) \end{aligned} . \tag{A.18}$$

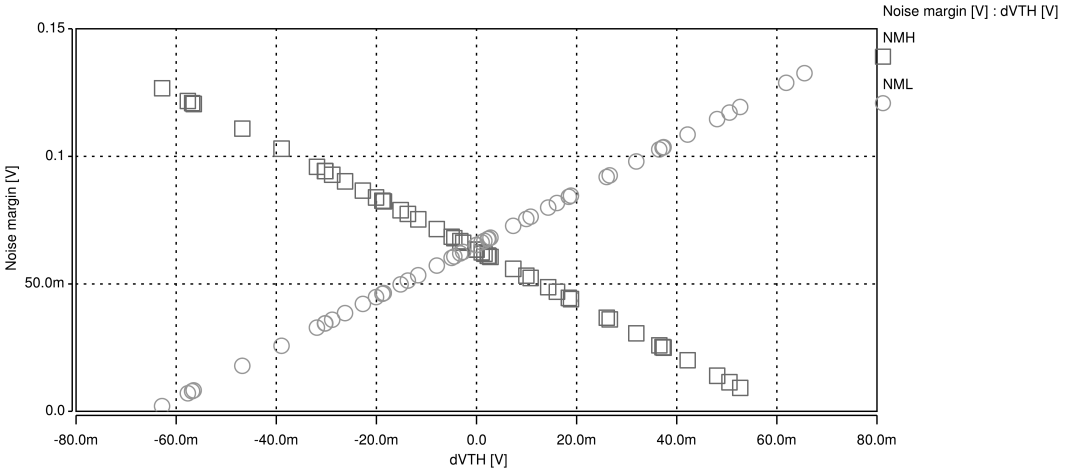


Abbildung A.1: Störabstand in Abhängigkeit des ΔU_{TH}.

Daraus ergibt sich der Störabstand für den High-Level $\mathrm{NM_H}$ und für den Low-Level $\mathrm{NM_L}$ durch Auswertung der Übertragungsfunktion an den Punkten $U_{\mathrm{in,l}}$ and $U_{\mathrm{in,h}}$:

$$U_{\mathrm{out,h}} = U_{\mathrm{out}}(U_{\mathrm{in,l}}) \quad \text{und} \quad U_{\mathrm{out,l}} = U_{\mathrm{out}}(U_{\mathrm{in,h}}),$$

$$NM_L = U_{\mathrm{in,l}} - U_{\mathrm{out,l}} \quad \text{und} \quad NM_H = U_{\mathrm{out,h}} - U_{\mathrm{in,h}}.$$

Aus (A.18) wird eine lineare Abhängigkeit von der Schwellenspannung ΔU_{TH} ersichtlich. Unter der Voraussetzung, dass sich die Versorgungsspannung U_{DD} nicht ändert, sind die logarithmischen Ausdrücke konstant, und der Einfluss von $U_{\mathrm{in,l/h}}$ auf $U_{\mathrm{out,h/l}}$ kann ermittelt werden. Damit wird auch der Einfluss von ΔU_{TH} auf den Störabstand bestimmbar. Die Auswertung von (A.17) an der Stelle $U_{\mathrm{in,l/h}}$ ergibt, dass ΔU_{TH} in den Exponentialtermen immer eliminiert wird, und diese keinen Einfluss auf den Störabstand haben. Dies bedeutet, dass sich die Form der Übertragungsfunktion mit ΔU_{TH} nicht ändert, und wir immer den selben Wert für die Auswertung der Übertragungsfunktion an der Stelle $U_{\mathrm{in,l/h}}$ erhalten. Vielmehr wird die Übertragungsfunktion nur um ΔU_{TH} auf der Ordinate verschoben. Damit variiert der Störabstand ebenfalls nur um ΔU_{TH}:

$$NM_L^* = NM_L + \Delta U_{\mathrm{TH}} \quad \text{und} \quad NM_H^* = NM_H - U_{\mathrm{TH}} \tag{A.19}$$

Zur Verifikation des Ergebnisses wurden Monte-Carlo Simulationen an einem Inverter mit Versorgungsspannung im Subschwellbereich durchgeführt. Um eine technologieunabhängige Auswertung vorzunehmen, wurden dazu *predictive technology models* (PTM) [100] genutzt, die mit einer Normalverteilung mit einem Sigma von 30 mV für die Schwellenspannung U_{TH} versehen wurden. Das Simulationsergebnis des Störabstands in Abhängigkeit von der Abweichung der Schwellenspannung vom Nominalwert ist in Abb. A.1 dargestellt. Hier wird die lineare Abhängigkeit von $\mathrm{NM_H}$ und $\mathrm{NM_L}$ von ΔU_{TH} direkt ersichtlich.

Anhang B

Skalierungsregeln

Zum Vergleich von Implementierungen mit verschiedenen CMOS-Technologien und zur Abschätzung des Ressourcenbedarfs von bestehenden Implementierungen in anderen CMOS-Technologien werden Skalierungsregeln für die Veränderung charakteristischer Größen herangezogen. In Tabelle B.1 sind die in dieser Arbeit genutzten Regeln angegeben (abgeleitet von [30]), wobei sich die Faktoren α_l für die Gatelänge, α_u für die Versorgungsspannung sowie $\alpha_{\mathrm{T_{ox}}}$ für die Dicke des Gate-Dielektrikums aus der Änderung dieser Werte zwischen zwei Technologieschritten ergeben. Ändern sich die Gate-Länge, die Dicke des Dielektrikums und die Versorgungsspannung in gleichem Maße, spricht man vom „constant-field scaling“, mit einem gemeinsamen Skalierungsfaktor α für alle Skalierungsfaktoren.

Als Beispiel wird der Skalierungsfaktor (SF) für die Gatelänge α_l für den Wechsel von einer 350 nm Technologie zu einer 130 nm Technologie zu

$$\alpha_\mathrm{l} = \frac{L_{\mathrm{Gate,\ min,\ 350\,nm}}}{L_{\mathrm{Gate,\ min,\ 130\,nm}}} \approx \frac{350\,\mathrm{nm}}{130\,\mathrm{nm}} \approx 2,6923.$$

In dieser Arbeit werden die Begriffe *worst case*, *typical case* und *best case* im Zusammenhang mit Arbeitsbereichen von in CMOS-Technologien entworfenen Schaltungen

Tabelle B.1: Allgemeine Skalierungsregeln für MOS-Technologie (abgeleitet von [30]).

Größe	Skalierungsfaktor	Konstant-Feld Skalierung
Gatelänge	$1/\alpha_\mathrm{l}$	$1/\alpha$
Gatekapazität	$\alpha_{\mathrm{T_{ox}}}/\alpha_\mathrm{l}^2$	$1/\alpha$
Spannung	$1/\alpha_\mathrm{u}$	$1/\alpha$
Frequenz	α_f	α
Fläche	$1/\alpha_\mathrm{l}^2$	$1/\alpha^2$
Leistung	$(\alpha_\mathrm{f}\alpha_{\mathrm{T_{ox}}}) \,/\, (\alpha_\mathrm{l}^2\alpha_\mathrm{u}^2)$	$1/\alpha^2$

Tabelle B.2: Parameter der Arbeitsbereiche verwendeter CMOS-Technologien.

Technologie	best	typical	worst
350 nm	3,60 V, 0°C, SF 0,64	3,30 V, 25°C, SF 1,00	3,00 V, 75°C, SF 1,4
130 nm	1,32 V, 0°C, SF 0,80	1,20 V, 25°C, SF 1,00	1,08 V, 85°C, SF 1,2

verwendet. Die Tabelle B.2 gibt die wichtigsten Parameter der Arbeitsbereiche, die Versorgungsspannung, die Temperatur und den Skalierungsfaktor für die verwendete Technologie und den jeweiligen Arbeitsbereich an. Der Skalierungsfaktor gibt an, wie die geometrischen Strukturen des Layouts einer Schaltung von den entworfenen Strukturen abweichen. Dieses hat direkten Einfluss auf die entstehenden parasitären Kapazitäten.

Anhang C

Simulink Modelle

In diesem Kapitel sind die für die Simulation des Minimalsystems eines Neurons verwendeten Simulink Modelle aufgeführt. Die Modelle umfassen das Modell der Zellmembran mit dem passiven Transport von Ionen durch die Ionenkanäle (Abb. C.2). Daneben sind die Natrium-Kalium-Pumpe in Abb. C.1 sowie das Modell des Axonhügels in Abb. C.3 dargestellt. Die Auslösung des Aktionspotentials wurde mit dem Modell nach Hodgkin und Huxley in Abb. C.4 nachgebildet. Das Gesamtsystem zur Simulation des Minimalmodells ist in Abb. C.5 zur Übersicht gegeben.

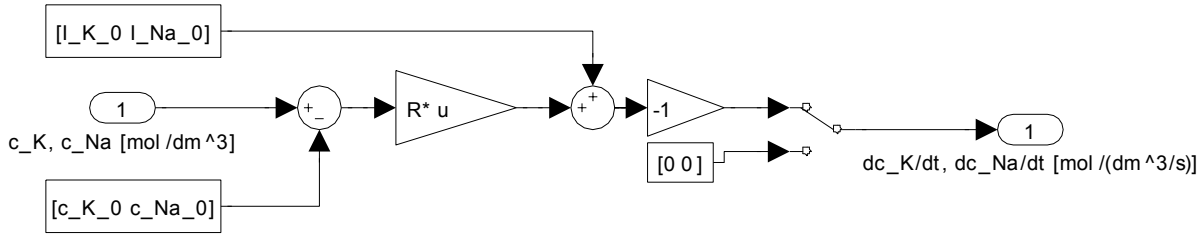


Abbildung C.1: Simulink Modell des Zustandsreglers der Natrium-Kalium-Pumpe.

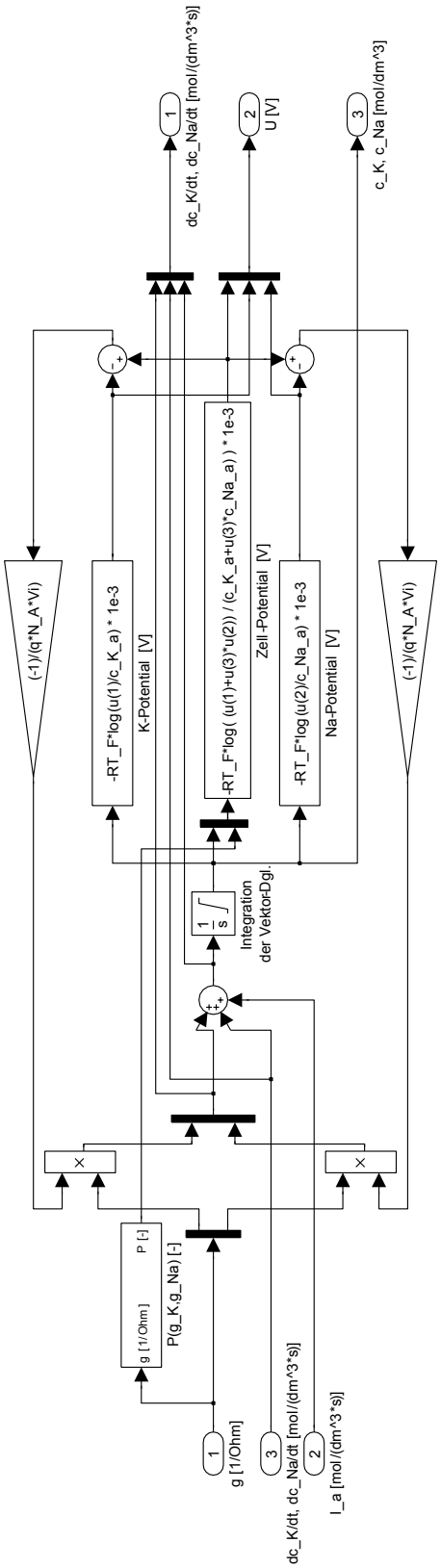


Abbildung C.2: Simulink Modell der Zellmembran (nichtlineares Ionenkanalmodell).

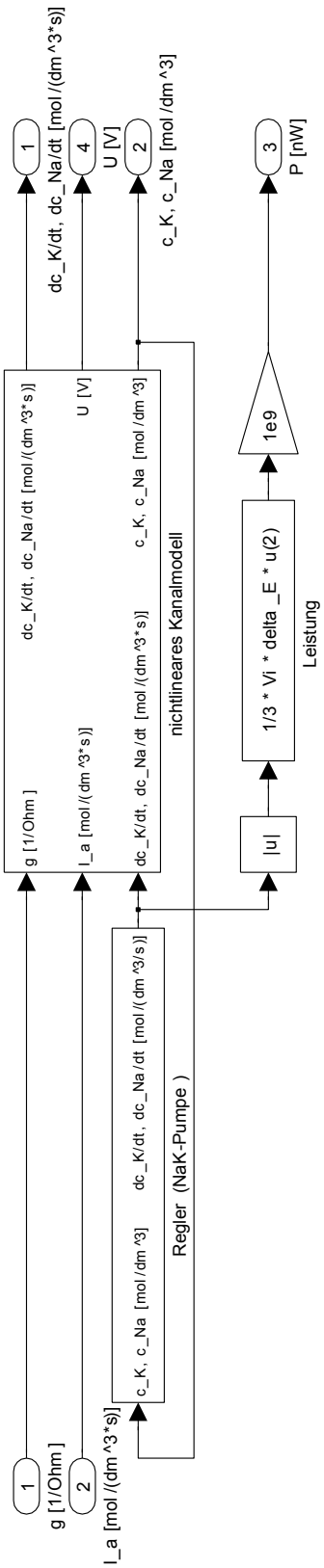


Abbildung C.3: Simulink Modell des Axonhügels.

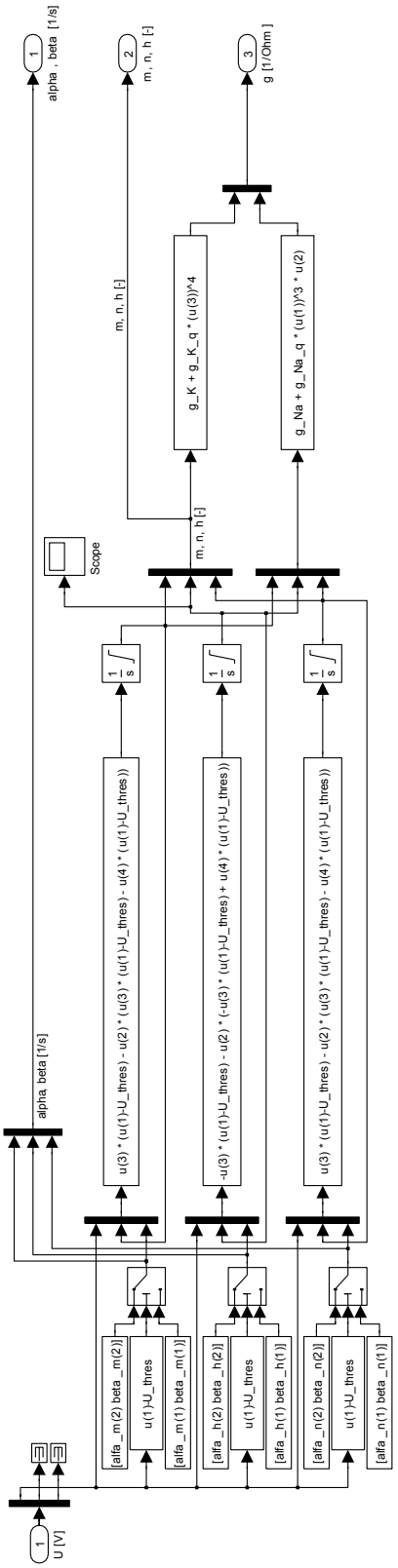


Abbildung C.4: Simulink Modell der axonalen Erregung nach dem Hodgkin-Huxley Modell.

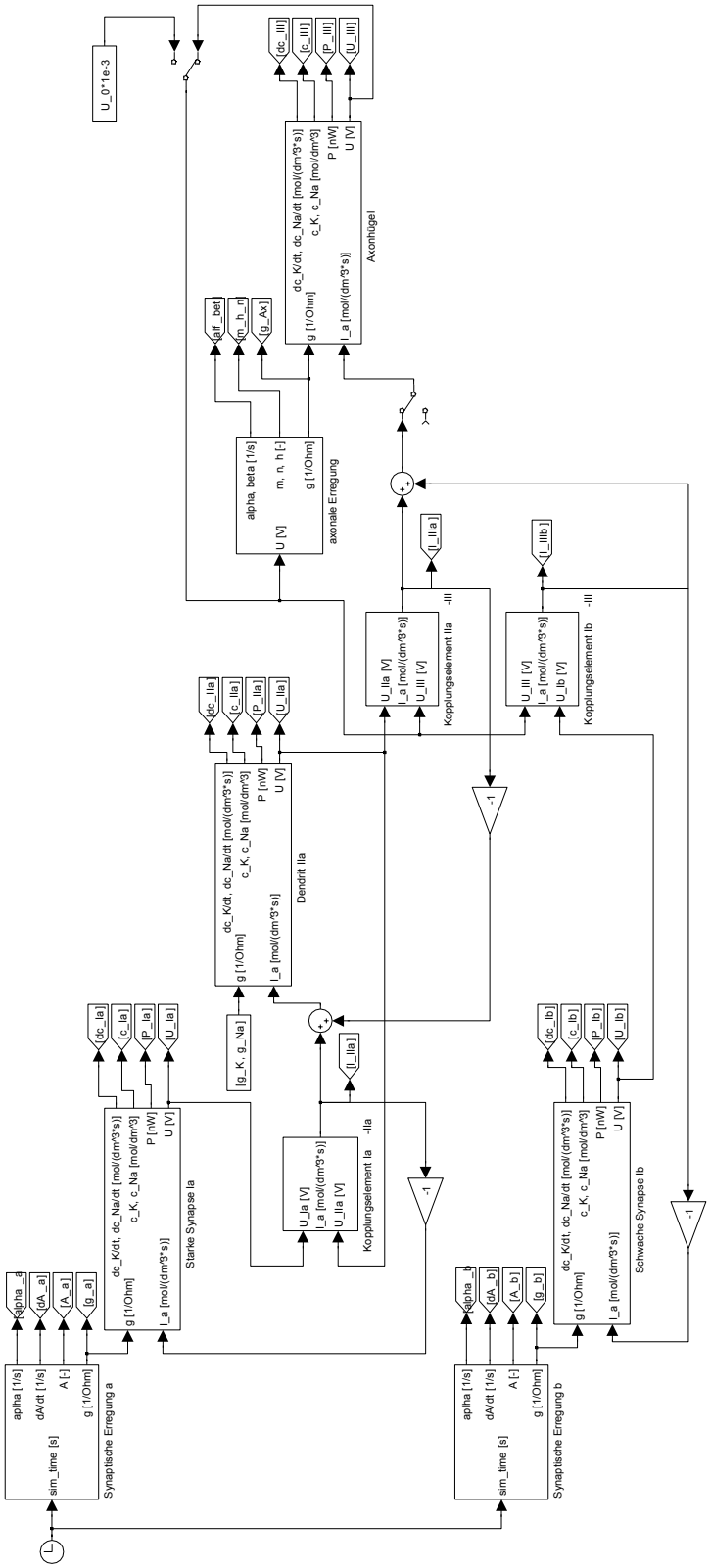


Abbildung C.5: Gesamtmodell von Neuron und zwei unterschiedlich starken Synapsen, verbunden über eine dendritische Verbindung.

Verzeichnis der verwendeten Abkürzungen und Formelzeichen

Abkürzungen

cAMP	Cyclisches AdenosinMonoPhosphat
ALU	Arithmetic Logic Unit
AM	Associative Memory – Assoziativspeicher (Eine Form des CAM)
AMP	AdenosinMonoPhosphat
ADP	AdenosinDiPhosphat
AER	Address Event Representation (Protocol)
AP	AktionsPotential
ASIC	Application Specific Integrated Circuit
ATP	AdenosinTriPhosphat
BINAM	BInary Neural Associative Memory
CAM	Content Addressable Memory – inhaltsadressierbarer Speicher
CNN	Cellular Neural Network
DSP	Digital Signal Processor
EPSC	Excitatory PostSynaptic Current
EPSP	Excitatory PostSynaptic Potential
ER	Endoplasmatisches Retikulum
FPGA	Field Programmable Gate Array
GABA	γ-Aminobuttersäure
IPSC	Inhibitory PostSynaptic Current
IPSP	Inhibitory PostSynaptic Potential
IAF	Integrate And Fire Neuron
LEGION	Locally Excitatory Globally Inhibitory Oscillator Network
LIAF	Leaky Integrate And Fire Neuron
LIF	Leaky Integrate and Fire Neuron
LUT	Look-Up Table

LSB	Least Significant Bit
MAC	Multiply and ACcumulate
MSB	Most Significant Bit
PCNN	Pulse Coded Neural Network
PE	ProzessElement
PAR	Place And Route
PSP	PostSynaptic Potential
SNR	Signal Noise Ratio
ST0	Stuck-at Zero
ST1	Stuck-at One
SRM	Spike Response Model

Konstanten

F	Faradaykonstante (F = 96485,3383 C/mol)
R	Gaskonstante (R = 8,314472 J/mol K)
k_B	Boltzmann-Konstante ($k_B = 1{,}3806505 \cdot 10^{-23}$ J/K)
N_A	Avogadrozahl, Anzahl der Teilchen in einer Stoffmenge von 1 mol ($N_A = 6{,}0221415 \cdot 10^{23}$)

Lateinische Buchstaben

A	Zelloberfläche eines Neurons
$A(t)$	Zeitbehafteter Verlauf der Anzahl zusätzlich geöffneter Ionenkanäle
A_v	Spannungsverstärkung
C_{back}	Koppelkapazität des analogen LIAF Neurons
C_{mem}	Membrankapazität der Zellmembran eines Neurons
C_{ox}	Spezifische Kapazität von Siliziumdioxid
$G(t)$	Erhöhung der Membranleitwerte durch synaptische Übertragung
$\mathbf{I}_0$	Vektor der Ruheströme des biophysikalischen Grundmodells
I_{AM}	normierter Informationsgehalt einer Assoziativspeichermatrix
I_D	Drainstrom eines MOS-Transistors
I_h	Informationsgehalt eines Assoziativspeichers
I_{K}	Kalium-Ionenstrom
I_{Na}	Natrium-Ionenstrom
$I(\mathbf{W})$	Informationsgehalt einer Assoziativspeichermatrix
$I(\mathbf{y})$	Informationsgehalt eines Ausgangsvektors
K_P	Parameter des 1/f-Rauschens bei MOS-Transistoren

K_N	Parameter des 1/f-Rauschens bei MOS-Transistoren
$\mathbf{M}$	Systemmatrix des ungeregelten Systems des biophysikalischen Grundmodells
N_{k0}	Erwartungswert für fehlerhafte, fehlende Einsen im Abrufvektor eines Assoziativspeichers
N_{k1}	Erwartungswert für fehlerhafte, zusätzliche Einsen im Abrufvektor eines Assoziativspeichers
NM_H	Noise-Margin High-Pegel
NM_L	Noise-Margin Low-Pegel
P	Relative Permeabilität
P_{K}	Permeabilität von Kalium-Ionen
P_{Na}	Permeabilität von Natrium-Ionen
R	Allgemeiner (ohmscher) Widerstand
$\mathbf{R}$	Regel-Vektor des biophysikalischen Grundmodells
RT^*	Rezeptor-Transmitter Bindung
$R+T$	Rezeptor und Transmitter
T	Periodendauer wiederkehrender Feuerereignisse
T_{cyc}	Periodendauer eines Taktsignals bei digitalen Systemen
T_{FA}	Verzögerungszeit eines Volladdierers
T_{MUX}	Verzögerungszeit eines Multiplexers
U_0, U_1	Potentiale des linearisierten Zellmembranmodells
U_{D}	Drain-Potential eines MOS-Transistors
U_{G}	Gate-Potential eines MOS-Transistors
U_{K}	Nernst-Potential von Kalium
U_{Na}	Nernst-Potential von Natrium
$U_{\mathrm{K,0}}$	Nernst-Potential von Kalium im *steady-state*
$U_{\mathrm{Na,0}}$	Nernst-Potential von Natrium im *steady-state*
U_{S}	Source-Potential eines MOS-Transistors
U_{T}	Temperaturspannung
U_{TH}	Feuerschwelle eines LIAF Neurons
U_{trip}	Schaltspannung eines Komparators
$\mathbf{V}$	Eigenvektoren der Systemmatrix $\mathbf{M}$
V_0	Ruhepotential der SIRENS Implementierung
V_i	Zellvolumen eines Neurons
$\bar{V}_n^2$	Rauschspannung
V_{tast}	Tastverhältnis der Eingangspulsrate der SIRENS Implementierung
$\mathbf{W}$	Matrix der Pumpzyklen des biophysikalischen Grundmodells in Kap. 3.1.3
$\mathbf{W}$	Assoziativspeichermatrix in Kapitel 5

W_{load}	Arbeit beim Aufladen einer Zellmembran
W_{reset}	Arbeit beim aktiven Entladen einer Zellmembran
$\mathbf{c}$	Vektor der Ionenkonzentrationen
c_{K}	Konzentration von Kalium-Ionen
c_{Na}	Konzentration von Natrium-Ionen
$c_{\text{K,0}}$	Konzentration von Kalium-Ionen im *steady-state*
$c_{\text{Na,0}}$	Konzentration von Natrium-Ionen im *steady-state*
$c_{\text{K,a}}$	Konzentration von Kalium-Ionen im extrazellulären Raum
$c_{\text{Na,a}}$	Konzentration von Natrium-Ionen im extrazellulären Raum
f	Pumpverhältnis von Natriumionen zu Kaliumionen (typ. 3/2)
f_{3dB}	3 dB Grenzfrequenz
f_{clk}	Frequenz des Takteingangs bei digitalen Neuronen
f_{in}	Eingangspulsrate
f_{osc}	Frequenz des Ringoszillators des ultra-low-power Chips
f_{out}	Ausgangspulsrate
$g_{\text{discharge}}$	Leitwert bei aktiven Entladeströmen
g_{leak}	Leitwert bei passiven Leckströmen
g_m	Gatesteilheit eines MOS-Transistors
g_{K}	Leitwert der Kalium-Ionenkanäle
g_{Na}	Leitwert der Natrium-Ionenkanäle
$h(t)$	Funktion des Hodgkin-Huxley Modells
i_{syn}	Synaptischer Strom
i_{leak}	Strom bei passiver Entladung
k	Anzahl an Einsen im Ausgangsvektor der Größe n
k_1	Anzahl fehlerhafter Einsen im Ausgangsvektor der Größe n
l	Anzahl an Einsen im Eingangsvektor der Größe m
m	Größe eines Eingangsmusters in Bit
$m(t)^3$	Funktion des Hodgkin-Huxley Modells
n	Größe eines Ausgangsmusters in Bit
$n(t)^4$	Funktion des Hodgkin-Huxley Modells
p_{on}	Wahrscheinlichkeit, dass ein Matrixelement eines Assoziativspeichers auf Eins gesetzt ist
p_{off}	Wahrscheinlichkeit, dass ein Matrixelement eines Assoziativspeichers auf Null gesetzt ist
p_{st0}	Wahrscheinlichkeit für einen Stuck-at 0 Fehler
p_{st1}	Wahrscheinlichkeit für einen Stuck-at 1 Fehler
$q(t)$	Zeitbehaftetes Neurotransmitterprofil, z. B. eine α-Funktion

r_o	Ausgangswiderstand eines MOS-Transistors
t_{fire}	Dauer eines Aktionspotentials eines LIAF Neurons
t_{pulse}	Zeit bis zum Auslösen eines Aktionspotentials
t_{relax}	Relaxionszeit eines Neurons mit Leaky-Term in einer Feuerpause
$u_{c,f}$	Membranpotential nach Aufladung der Zeit t_{fire}
$u_{c,T}$	Membranpotential nach Entladung der Zeit t_{relax}
$\mathbf{v}$	Abrufvektor (Eingangsvektor) in einen Assoziativspeicher, auch fehlerbehaftet
w_{kl}	Gewicht einer synaptischen Verbindung
$w_{\max}$	Maximum eines Gewichts einer synaptischen Verbindung
$w_{\min}$	Minimum eines Gewichts einer synaptischen Verbindung
$\mathbf{x}$	Eingangsvektor in einen Assoziativspeicher
$\mathbf{y}$	Ausgangsvektor aus einem Assoziativspeicher
z	Anzahl gespeicherter Muster im Assoziativspeicher
$z(k)$	Pulsausgang (AP) der SIRENS Implementierung

Griechische Buchstaben

Θ	Feuerschwelle eines Neurons
$\alpha(t)$	α-Funktion
α	Übergangswahrscheinlichkeit des freien Transmitter zum gebundenen Transmitter
β	Übergangswahrscheinlichkeit des gebundenen Transmitter zum freien Transmitter
δ_j	Fehler in einer Zeile j einer Spalte des Assoziativspeichers
$\Delta_{i,\mu}$	Geometrischer Abstand eines Musters μ zur idealen Trennlinie des Hyperraums
η	Lernrate der allg. Hebb-Regel
μ_n	Ladungsträgerbeweglichkeit eines nMOS-Transistors
μ_p	Ladungsträgerbeweglichkeit eines pMOS-Transistors
τ	Zeitkonstante

Abbildungsverzeichnis

1.1 Schematische Darstellung einer ausgebildeten Zellmembran (nach [17]). . 7

1.2 Zellmembran mit aktiven und passiven Transportmechanismen für die wichtigsten beteiligten Ionen. 8

1.3 Molekül Adenosintriphosphat. 9

1.4 Purkinje Zelle nach [38]. a) Axon b) Kollaterale (Verzweigung des Axons im Zielgebiet) c) und d) Dendritenäste. 10

1.5 Zeitlicher Verlauf eines Aktionspotentials und der Permeabilität der Membran für bestimmte Ionentypen (nach [17]). 14

1.6 Neuron mit myelinisiertem Axon und Ranvier-Schnürringen zur schnellen Signalfortleitung. Weiter ist die Verzweigung des Axons im Zielgebiet anderer Neurone angedeutet. 16

2.1 Beispielhafter Verlauf einer exzitatorischen und inhibitorischen postsynaptischen Antwort. 22

2.2 Auslösen eines Aktionspotentials durch zeitliche und räumliche Überlagerung exzitatorischer Pulse. 22

2.3 Modell eines Integrate and Fire Neurons aus der Modellvorstellung eines RC-Schaltkreises für die Membran. Nicht eingezeichnet sind zusätzliche notwendige Elemente für die Wandlung des Spannungspulses am Ausgang in den Eingangsstrom der nächsten Stufe (Synapse). 23

2.4 Vereinfachte Darstellung der Systemarchitektur und Systemkomponenten der Schrauwen-Implementierung (nach [87]). 25

2.5 Beispiel des Verlaufs eines Membranpotentials für die Implementierung nach Upegui [96]. 27

2.6 Pixeldifferenztestkomponente der Implementierung nach Torres-Huitzil [92]. 29

2.7 Komponente zur Bestimmung der Erregung des Zentralneurons in der Torres-Huitzil-Implementierung (nach [92]). 29

2.8 Arithmetikkomponente der Torres-Huitzil-Implementierung (nach [92]). . 30

2.9 Integrate-and-Fire Neuron nach Indiveri [54]. 35

2.10 Integrate-and-Fire Neuron nach Chicca [20]. 37

2.11 Integrate-and-Fire Neuron nach Wijekoon et al. [102]. 37

3.1 Elektrischer Ersatzschaltkreis für das Membranpotential im steady-state (angelehnt an [61]). 43

3.2 Nomenklatur zum Reglerentwurf für das geschlossene regelungstechnische System. 49

3.3 Trajektorien des Verlaufs der Ionenkonzentrationen im einfachen nichtlinearen Neuronenmodell mit a) geschlossenem Regelkreis und b) offenem Regelkreis bei verschiedenen Anfangsbedingungen. 50

3.4 Darstellung von $\dot{V}(\mathbf{c})$ für das geregelte System. In der Umgebung der Ruhelage ist die Funktion negativ definit und die Ruhelage stabil. 53

3.5 Zeitabhängige Leitwerte g_{K} und g_{Na} an der Synapse. Zum Zeitpunkt $t = 1\,\mathrm{ms}$ wird die starke Synapse erregt, zum Zeitpunkt $t = 3\,\mathrm{ms}$ wird die schwache Synapse erregt. 55

3.6 Zeitverlauf (a) der Koeffizienten $m(t)$, $n(t)$ und $h(t)$ und (b) der daraus resultierenden Leitwerte der Kalium- und Natrium-Kanäle für ein bei t=3,7 ms ausgelöstes Aktionspotential. 57

3.7 Trajektorien der Natrium- und Kalium-Konzentration des simulierten Neuronenmodells (a) ohne Aktionspotentialerzeugung (überhöhte Feuerschwelle) und (b) mit Aktionspotentialerzeugung (normale Feuerschwelle). . . . 59

3.8 Simulationsergebnis des geregelten Systems mit zwei Synapsen (eine schwache Synapse, eine starke Synapse). Die Erregung des Neurons erfolgt zu den Zeitpunkten $t_1 = 1\,\mathrm{ms}$ und $t_2 = 3\,\mathrm{ms}$ über eine eingeprägte α-Funktion an den Synapsen. 60

3.9 Erregung des Neurons mit konstantem Strom. 62

3.10 Vereinfachtes Modell der Zellmembran mit Leckstrom 65

3.11 Verlustleistung eines Neurons ohne Erzeugung eines Aktionspotentials (Feuerschwelle heraufgesetzt) und ohne Beschränkung der Spannung über der Membrankapazität. 68

3.12 Maximale Eingangspulsrate, die ohne Auslösen eines Aktionspotentials möglich ist. Parameter $C = 200\,\mathrm{fF}$, $I = 200\,\mathrm{nA}$, $U_{\mathrm{TH}} = 2\,\mathrm{V}$, $t_{\mathrm{fire}} = 1\,\mu\mathrm{s}$. . 70

3.13 Darstellung der theoretisch erzielbaren Ausgangspulsrate eines Neurons gegenüber der Eingangspulsrate bei verschieden großen Leckströmen. $U_{\mathrm{TH}} = 2,0\,\mathrm{V}, I_{\mathrm{in}} = 200\,\mathrm{nA}, t_{\mathrm{fire}} = 1\,\mu\mathrm{s}$ 72

4.1 Schaltplan des LIAF Neurons in 130 nm Technologie. 78

4.2 Layout des LIAF Neurons in 130 nm Technologie. 80

4.3 Ausgangspulsrate f_{out} (durchgezogene Linie) und Verlustleistung (gestrichelte Linie) aufgetragen über dem Eingangsstrom. 81

4.4 Ausgangspulsrate des Neurons in Abhängigkeit des Eingangsstroms für verschiedene Betriebsfälle. 82

4.5 Abschätzung und Simulationsergebnis für die Ausgangspulsrate und die Verlustleistung des LIAF Neurons. 83

4.6 Schaltplan einer wahlweise exzitatorischen oder inhibitorischen n Bit Synapse in Stromschaltungstechnik. 85

4.7 Layout einer exzitatorischen 5 Bit Synapse in einer 130 nm CMOS-Technologie. 87

4.8 Modifizierte SRAM-Zelle für lokale Gewichtsspeicherung. 88

4.9 Differenzverstärker-Paar aus MOS-Transistoren und korrespondierende Übertragungskennlinie. 89

4.10 Flächenbedarf ausgewählter bitparalleler und bitserieller Multiplizierervarianten über der Wortbreite. 95

4.11 Layout des Testchips mit 4 ALUs in einer 90 nm CMOS-Technologie mit Versorgungsspannung im Subschwellenbereich. 99

4.12 Maximaler Takt der ALUs über der Versorgungsspannung. 100

4.13 Energieumsatz der ALU pro Instruktion über der Versorgungsspannung. . 101

4.14 Grundelement eines einzelnen Neurons zur zeitdiskreten Approximation kontinuierlicher Funktionen (a) und Verschaltung dreier Grundelemente zur Funktion eines Leaky Integrate-and-Fire Neurons (b). 104

4.15 Approximation der Funktion des kontinuierlichen pulscodierten Neurons nach (4.10) mit der SIRENS Struktur. Der Verlauf des Membranpotentials des kontinuierlichen LIAF Neurons wurde mit den Modellparametern $\xi = 50; \Delta = 0{,}1; v_A = 1; I = 1{,}3; g_L = 0{,}6; C = 1{,}4$ erzeugt, das Simulationsergebnis des digitalen pulscodierten Neurons aus drei Grundelementen mit den Parametern: $\hat{I} = 1{,}0; \hat{b} = 0{,}929 \cdot 10^{-2}; \hat{g}_L = 0{,}429 \cdot 10^{-2}$. 107

4.16 Dynamische Verlustleistung (durchgezogene Linien) und Verlustleistung durch statische Leckströme (gestrichelte Linie) der SIRENS-Struktur in einem 130 nm Prozess. 109

4.17 Blockschaltbild eines auf Zählern basierenden LIAF Neuronenmodells. . . 112

4.18 Schaltungen zur Priorisierung eines Eingangs. 113

4.19 Dynamische Verlustleistung (durchgezogene Linie) und Verlustleistung durch statische Leckströme (gestrichelte Linie) eines bitseriellen, zählerbasierten Neurons mit 6 Bit Auflösung und 250 kHz Takt in einem 130 nm CMOS-Prozess. 114

4.20 Flächenbedarf digitaler Implementierungsvarianten über der Wortbreite. . 116

4.21 Layout eines Testchips mit 1040 LIAF Neuronen in einer 130 nm Technologie. Erkennbar sind vier separate ansteuerbare Felder mit je 260 Neuronen des vorgestellten LIAF-Modells und der Anschluss-Pad-Ring im äußeren Bereich. 119

4.22 Gemessene Aussendung eines Aktionspotentials an U_{pulse} (negiertes Signal) und rekonstruiertes, korrespondierendes Membranpotential U_{mem} für einen injizierten Strom von $I_{\text{in}} = 4/260\,\mu\text{A}$ ($U_{\text{TH}} = 730\,\text{mV}$, $U_{\text{leak}} = 0\,\text{V}$). 121

4.23 Übertragungskennlinie eines HW LIAF Neurons. 121

4.24 Leistungsaufnahme eines LIAF Neurons. 122

4.25 Auf eine robuste Funktion hin modifiziertes Neuron in 130 nm CMOS-Technologie. 123

5.1 Allgemeine Struktur eines Assoziativspeichers. 127

5.2 Geometrische Interpretation der Mustertrennung durch einen Schwellwert. Der minimale Abstand zum nächsten Element eines Musters ist mit $\Delta_{i,\mu}$ dargestellt. 129

5.3 Auswirkung der stuck-at Fehler auf den Informationsgehalt I_h/mn des Assoziativspeichers (n, m = 4096; l = 13; k = 3). 137

5.4 Auswirkung von fehlenden Einsen und zusätzlichen Einsen im Eingabemuster auf den Informationsgehalt I_h/mn der Assoziativspeichermatrix (m, n = 4096; $l = 13$; $k = 3$; $\Theta = l'$). 140

5.5 Arten von Synchronisationszuständen angelegter Muster. 142

5.6 Konturlinien zur Wahl des minimalen Gewichts für asynchrone Eingangsmuster ($g_{\text{leak}} = 10^{-9}\,\text{S}$). 145

5.7 Konturlinien zur Wahl des minimalen Gewichts für asynchrone Eingangsmuster ($g_{\text{leak}} = 10^{-9}\,\text{S}$). 146

5.8 Vergleich des theoretischen Informationsgehalts des BiNAM und dem durch Simulation ermittelten Informationsgehalts des PCNN-BiNAM mit perfekten Mustern im Abruf (m, $n = 100$; $l = 5$; $k = 2$; $\Theta = l$). 148

5.9 Informationsgehalt des PCNN-BiNAM ohne Korrekturfaktor und mit Korrekturfaktor bei perfekten Mustern (m, $n = 100$; $l = 5$; $k = 2$; $\Theta = l$). . 149

A.1 Störabstand in Abhängigkeit des ΔU_{TH}. 160

C.1 Simulink Modell des Zustandsreglers der Natrium-Kalium-Pumpe. 163

C.2 Simulink Modell der Zellmembran (nichtlineares Ionenkanalmodell). . . . 164

C.3 Simulink Modell des Axonhügels. 165

C.4 Simulink Modell der axonalen Erregung nach dem Hodgkin-Huxley Modell. 166

C.5 Gesamtmodell von Neuron und zwei unterschiedlich starken Synapsen, verbunden über eine dendritische Verbindung. 167

Tabellenverzeichnis

1.1 Intra- und extrazelluläre Ionenkonzentrationen (aus [85]). 12

2.1 Übersicht über die Syntheseergebnisse der vorgestellten Implementierungen. 33
2.2 Übersicht über vorgestellte analoge Implementierungen. 39
2.3 Energieumsatz technischer und biologischer Neuronen. 40

3.1 Ionenkonzentrationen, Nernst-Potentiale und Leitwerte für verschiedene Ionenarten in Zellen im steady-state [86]. 64

4.1 Flächenbedarf und Verlustleistung für Neuron und Synapsen in einer 130 nm CMOS-Technologie. 84
4.2 Flächenbedarf und Verzögerungszeit paralleler Multiplizierer-Varianten (aus [79]). 93
4.3 Flächenbedarf und Verzögerungszeit bitserieller Multiplizierer-Varianten (aus [60]). 94
4.4 Relativer Diskretisierungsfehler der Parameter des diskreten Systems. . . 106
4.5 Flächenbedarf und dynamische Verlustleistung der SIRENS Struktur mit 16 Bit Wortbreite bei Synthese auf 1 MHz Takt. 108
4.6 Flächenbedarf und dynamische Verlustleistung der SIRENS Struktur mit bitseriellen 16 Bit Multiplizierern bei Synthese auf 20 MHz Takt. 111
4.7 Flächenbedarf und dynamische Verlustleistung der zählerbasierten bitseriellen Struktur mit 6 Bit Auflösung bei 250 kHz Takt. 115
4.8 Versorgungs- und Referenzspannungen des Testchips 118

5.1 Fehlerklassen im Assoziativspeicher. Der Fehler wird als Variation des Originalwerts mit $(1 + \delta)$ betrachtet. 128
5.2 Energie-, Flächenbedarf und Eigenschaften der Varianten der Neurone . . 153

B.1 Allgemeine Skalierungsregeln für MOS-Technologie (abgeleitet von [30]). 161

B.2 Parameter der Arbeitsbereiche verwendeter CMOS-Technologien. 162

Literaturverzeichnis

[1] *Facets: Fast Analog Computing with Emergent Transient States in Neural Architectures.* – Verfügbar unter `http://cordis.europa.eu`, Projekt Referenz 015879

[2] *SpikeForce: Real-time Spiking Networks for Robot Control.* – Verfügbar unter `http://cordis.europa.eu`, Projekt Referenz IST-2001-35271

[3] *VisionIC: intelligente Vision-Plattform für den Massenmarkt.* – Verfügbar unter `http://tiborder.gbv.de`, Förderkennzeichen BMBF 01 M 3127 A

[4] *Pschyrembel. Klinisches Wörterbuch.* 259. Auflage. de Gruyter Verlag, 2002

[5] Alberts, B. ; Bray, D. ; Hopkin, K. ; Johnson, A. ; Lewis, J. ; Raff, M. ; Roberts, K. ; Walter, P. ; Nover, L. (Hrsg.) ; von Koskull-Döring, P. (Hrsg.): *Lehrbuch der Molekularen Zellbiologie.* 3. Auflage. Wiley Verlag, 2005

[6] Alberts, B. ; Bray, D. ; Lewis, J. ; Raff, M. ; Roberts, K. ; Watson, J. ; Jaenicke, L. (Hrsg.): *Molekularbiologie der Zelle.* 3. Auflage. Garland Science, 1995

[7] Amorim, A. ; Picanço-Diniz, C.: Horizontal projections of area 17 in Cebus monkeys: metric features, and modular and laminar distribution. In: *Brazilian Journal of Medical and Biological Imaging* 30 (1997), S. 1489–1501

[8] Attwell, D. ; Laughlin, S.: An Energy Budget for Signaling in the Grey Matter of the Brain. In: *Journal of Cerebral Blood Flow and Metabolism* 21 (2001), Nr. 10, S. 1133–1145

[9] Baker, R.: *CMOS: Circuit Design, Layout, and Simulation.* 2. Auflage. IEEE Press, 2005

[10] Baxter, D. ; Byrne, J.: Simulator for neural networks and action potentials (SNNAP): Description and application. In: Crasto, C. (Hrsg.): *Methods in Molecular Biology: Neuroinformatics.* Humana Press, 2007

[11] Benz, R. ; Fröhlich, O. ; Läuger, P. ; Montal, M.: Electrical capacity of black lipid films and of lipid bilayers made from monolayers. In: *Biochimica et Biophysica Acta* 394 (1975), S. 323–334

[12] Bower, J. ; Beeman, D.: *The Book of GENESIS: Exploring Realistic Neural Models with the GEneral NEural SImulation System.* 2. Auflage. Springer-Verlag, 1998

[13] Brette, R. ; Gerstner, W.: Adaptive Exponential Integrate-and-Fire Model as an Effective Description of Neuronal Activity. In: *Journal of Neurophysiology* 94 (2005), November, Nr. 5, S. 3637–3642

[14] Brette, R. ; Rudolph, M. ; Carnevale, T. ; Hines, M. ; Beeman, D. ; Bower, J. ; Diesmann, M. ; Morrison, A. ; Goodman, P. ; Harris Jr., F. ; Zirpe, M. ; Natschläger, T. ; Pecevski, D. ; Ermentrout, B. ; Djurfeldt, M. ; Lansner, A. ; Rochel, O. ; Vieville, T. ; Muller, E. ; Davison, A. ; El Boustani, S. ; Destexhe, A.: Simulation of networks of spiking neurons: A review of tools and strategies. In: *Journal of Computational Neuroscience* 23 (2007), December, Nr. 3, S. 349–398

[15] *Kapitel* Stochastic Bit-Stream Neural Networks. In: Burge, P. ; van Daalen, M. ; Rising, B. ; Shawe-Taylor, J.: *Pulsed Neural Networks.* MIT Press, 1999, S. 337–350

[16] Calhoun, B. ; Chandrakasan, A.: Ultra-dynamic voltage scaling using sub-threshold operation and local voltage dithering in 90nm CMOS. In: *ISSCC Dig. Tech. Papers*, 2005, S. 300–301

[17] Campbell, N.: *Biologie.* Spektrum Akad., 1997

[18] Castellani, G. ; Quinlan, E. ; Bersani, F. ; Cooper, L. ; Shouval, H.: A model of bidirectional synaptic plasticity: From signaling network to channel conductance. In: *Learning & Memory* 12 (2005), S. 423–432

[19] Chapeau-Blondeau, F. ; Chambet, N.: Synapse Models for Neural Networks: From Ion Channel Kinetics to Multiplicatice Coefficient w_{ij}. In: *Neural Computation* 7 (1995), S. 713–734

[20] Chicca, E. ; Badoni, D. ; Dante, V. ; D'Andreagiovanni, M. ; Salina, G. ; Carota, L. ; Fusi, S. ; Del Guidice, P.: A VLSI Recurrent Network of Integrate-and-Fire Neurons Connected by Plastic Synapses With Long-Term Memory. In: *IEEE Transactions on Neural Networks* 14 (2003), Nr. 5, S. 1297–1307

[21] Chow, C. ; Kopell, N.: Dynamics of Spiking Neurons with Electrical Coupling. In: *Neural Computation* 12 (2000), Nr. 7, S. 1643–1678

[22] Croon, J. ; Decoutere, S. ; Sansen, W. ; Maes, H.: Physical modeling and prediction of the matching properties of MOSFETs. In: *Proceeding of the 34th European Solid-State Device Research conference (ESSDERC 2004)*, 2004, S. 193–196

[23] Daut, J.: The living cell as an energy-transducing machine. A minimal model of myocardial metabolism. In: *Biochimica et Biophysica Acta (BBA) - Reviews on Bioenergetics* 895 (1987), Nr. 1, S. 41–62

[24] Destexhe, A.: Conductance-based integrate-and-fire models. In: *Neural Computation* 9 (1997), Nr. 3, S. 503–514

[25] Destexhe, A. ; Rudolph, M. ; Paré, D.: The high-conductance state of neocortical neurons in vivo. In: *Nature Reviews Neuroscience* 4 (2003), Nr. 9, S. 739–751

[26] Eickhoff, R. ; Rückert, U. (Hrsg.): *Fehlertolerante Neuronale Netze zur Approximation von Funktionen*. Heinz Nixdorf Institut, 2007 (HNI-Verlagsschriftenreihe 214)

[27] Enz, C. ; Krummenacher, F. ; Vittoz, E.: An analytical MOS transistor model valid in all regions of operation and dedicated to low-power and low-current applications. In: *Analog Integrated Circuits and Signal Processing* 8 (1995), Nr. 1, S. 83–114

[28] Fettiplace, R. ; Andrews, D. ; Haydon, D.: The Thickness, Composition and Structure of Some Lipid Bilayers and Natural Membranes. In: *Journal of Membrane Biology* 5 (1971), Nr. 2, S. 277–296

[29] Föllinger, O.: *Regelungstechnik*. 8. Auflage. Hüthig, 1994

[30] Frank, D. ; Dennard, R. ; Nowak, E. ; Solomon, P. ; Taur, Y. ; P., Hon-Sum: Device scaling limits of Si MOSFETs and their application dependencies. In: *Proceedings of the IEEE* 89 (2001), Nr. 3, S. 259–288

[31] Frank, G. ; Hartmann, G. (Hrsg.): *Ein digitales Hardwaresystem zur echtzeitfähigen Simulation biologienaher neuronaler Netze*. Heinz Nixdorf Institut, 1997 (HNI-Verlagsschriftenreihe 26)

[32] Gerstner, W.: Time structure of the activity in neural network models. In: *Physical Review E* 51 (1995), Nr. 1, S. 738–758

[33] *Kapitel* Spiking Neurons. In: Gerstner, W.: *Pulsed Neural Networks*. MIT Press, 1999, S. 3–53

[34] Gerstner, W. ; van Hemmen, J. ; Cowan, J.: What matters in neuronal locking. In: *Neural Computation* 8 (1996), S. 1653–1676

[35] Godin, C. ; Gordon, M. ; Muller, J.: SpikeCell: a deterministic spiking neuron. In: *Neural Networks* 15 (2002), Nr. 7, S. 873–879

[36] Goldman, D.: Potential, impedance, and rectification in membranes. In: *Journal of general physiology* 27 (1943), Nr. 1, S. 37–60

[37] Gomes-Leal, W. ; Silva, S. ; Oliveira, R. ; Picanço-Diniz, C.: Computer-assisted morphometric analysis of intrinsic axon terminals in the supragranular layers of cat striate cortex. In: *Anat. Embryol.* 205 (2002), S. 291–300

[38] Gray, H.: *Anatomy of the human body.* Lea & Febiger, 1918

[39] Hanson, S. ; Zhai, B. ; Seok, M. ; Cline, B. ; Zhou, K. ; Singhal, M. ; Minuth, M. ; Olson, J. ; Nazhandali, L. ; Austin, T. ; Sylvester, D. ; Blaauw, D.: Performance and Variability Optimization Strategies in a Sub-200mV, 3.5pJ/inst, 11nW Subthreshold Processor. In: *Proc. of IEEE Symposium on VLSI Circuits*, 2007, S. 152–153

[40] Hassoun, M. (Hrsg.): *Associative Neural Memories.* Oxford University Press, 1993

[41] Hastings, A.: *The Art of Analog Layout.* 2. Auflage. Prentice Hall, 2005

[42] Hebb, D.: *Organization of Behaviour.* Wiley, 1949

[43] Heittmann, A. ; Ramacher, U.: An Analog VLSI Pulsed Neural Network for Image Segmentation Using Adaptive Connection Weights. In: Dorronsoro, José R. (Hrsg.): *Artificial Neural Networks – Proc. of ICANN 2002* Bd. 2415, Springer-Verlag, 2002 (Lecture Notes in Computer Science). – ISBN 3–540–44074–7, S. 1293–1298

[44] Heittmann, A. ; Ramacher, U.: An Architecture for Feature Detection Utilizing Dynamic Synapses. In: *47th IEEE International Midwest Symposium on Circuits and Systems, Hiroshima, Japan*, 2004, S. II–160–II–488

[45] Heittmann, A. ; Ramacher, U.: Pulsed Neural Networks for Feature Detection Using Dynamic Synapses. In: *EIS 2004*, 2004, S. 160–488

[46] Hellmich, H. ; Klar, H.: An FPGA based Simulation Acceleration Platform for Spiking Neural Networks. In: *Proc. of the 47th IEEE Midwest Symposium on Circuits and Systems (MWSCAS)*, 2004, S. 389–392

[47] Hines, M. ; Carnevale, N ; Johnston, D.: The NEURON simulation environment. In: *Neural Computation* 9 (1997), Nr. 6, S. 1179–1209

[48] Hodgkin, A. ; Huxley, A.: Currents Carried by Sodium and Potassium Ions through the Membrane of the Giant Axon of Loligo. In: *Journal of Physiology* 116 (1952), S. 449–472

[49] Hodgkin, A. ; Huxley, A.: Measurement of Current-Voltage Relations in the Membrane of the Giant Axon of Loligo. In: *Journal of Physiology* 116 (1952), S. 424–448

[50] Hodgkin, A. ; Huxley, A.: A quantitative description of membrane current and its application to conduction and excitation in nerve. In: *Journal of Physiology* 117 (1952), S. 500–544

[51] Hodgkin, A. ; Huxley, A.: The Components of Membrane Conductance in the Giant Axon of Loligo. In: *Journal of Physiology* 116 (1952), S. 473–496

[52] Hodgkin, A. ; Huxley, A.: The Dual Effect of Membrane Potential on Sodium Conductance in the Giant Axon of Loligo. In: *Journal of Physiology* 116 (1952), S. 497–506

[53] Hubel, D. ; Wiesel, T.: Receptive fields and functional architecture in monkey striate cortex. In: *Journal of Physiology* 195 (1967), S. 215–243

[54] Indiveri, G.: A low-power adaptive integrate-and-fire neuron circuit. In: *Proc. IEEE International Symposium on Circuits and Systems* Bd. 4, 2003, S. 820–823

[55] Indiveri, G. ; Chicca, E. ; Douglas, R.: A VLSI Array of Low-Power Spiking Neurons and Bistable Synapses With Spike-Timing Dependent Plasticity. In: *IEEE Transactions on Neural Networks* 17 (2006), Nr. 1, S. 211–221

[56] Izhikevich, E.: Simple Model of Spiking Neurons. In: *IEEE Transactions on Neural Networks* 14, No. 6 (2003), S. 1569 – 1572

[57] Johnston, S. ; Prasad, G. ; Maguire, L. ; M., McGinnity: Comparative Investigation into Classical and Spiking Neuron Implementations on FPGAs. In: Duch, W. (Hrsg.) ; Kacprzyk, J. (Hrsg.) ; Oja, E. (Hrsg.) ; Zadrozny, S. (Hrsg.): *Artificial Neural Networks: Biological Inspirations – Proc. of ICANN 2005* Bd. 3696, Springer-Verlag, 2005 (Lecture Notes in Computer Science), S. 269–274

[58] Jones, J. ; Palmer, L.: An Evaluation of the Two-Dimensional Gabor Filter Model of Simple Receptive Fields in Cat Striate Cortex. In: *Journal of Neurophysiology* 58 (1987), S. 1233–1258

[59] Kahn, C.: Membrane receptors for hormones and neurotransmitters. In: *Journal of Cell Biology* 70 (1979), August, Nr. 2, S. 261–286

[60] Kalivas, P. ; Pekmestzi, K. ; Bougas, P. ; Tsirikos, A. ; Gotsis, K.: Low-Latency and High-Efficiency bit Serial-Serial Multipliers. In: *Proc. of XII. European Signal Processing Conference (EUSIPCO)*, 2004, S. 1345–1348

[61] Kandel, E. ; Schwartz, J. ; Jessell, T.: *Principles of neural science.* 4. Auflage. McGraw-Hill, 1999

[62] Kanerva, P.: *Sparse Distributed Memory.* MIT Press, 1988

[63] Kim, T. ; Keane, J. ; Eom, H. ; Kim, C.: Utilizing reverse short-channel effect for optimal subthreshold circuit design. In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 15 (2007), Nr. 7, S. 821–829

[64] Kistler, W. ; Gerstner, W. ; Hemmen, J. van: Reduction of the Hodgkin-Huxley Equations to a Single-Variable Threshold Model. In: *Neural Computation* 9 (1997), Nr. 5, S. 1015–1045

[65] Koch, C.: *Biophysics of Computation: Information Processing in Single Neurons.* Oxford University Press, 1999

[66] Kohonen, T.: *Associative Memory: A System-Theoretical Approach.* Springer-Verlag, 1977

[67] Laughlin, S. ; de Ruyter van Steveninck, R. ; Anderson, J.: The metabolic cost of neural information. In: *Nature Neuroscience* 1 (1998), Nr. 1, S. 36–41

[68] Liu, S. ; Kramer, J. ; Indiveri, G. ; Delbrück, T. ; Douglas, R.: Orientation-selective aVLSI Spiking Neurons. In: *Neural Networks* 14 (2001), S. 629–643

[69] Löffler, A. ; Rückert, U. (Hrsg.): *Energetische Modellierung neuronaler Signalverarbeitung.* Heinz Nixdorf Institut, 2000 (HNI-Verlagsschriftenreihe 72)

[70] Maass, W. (Hrsg.) ; Bishop, C. (Hrsg.): *Pulsed Neural Networks.* MIT Press, 1998

[71] Maass, W. ; Natschläger, T.: Emulation of Hopfield networks with spiking neurons in temporal coding. In: Bower, J. M. (Hrsg.): *Computational Neuroscience: Trends in Research.* Plenum Press, 1998, S. 221–226

[72] Madison, D. ; Malenka, R. ; Nicoll, R.: Mechanisms underlying long-term potentiation of synaptic transmission. In: *Annu. Rev. Neurosci.* 14 (1991), S. 379–397

[73] Matolin, D. ; Schreiter, J. ; Getzlaff, S. ; Schüffny, R.: An Analog VLSI Pulsed Neural Network Implementation for Image Segmentation. In: *Proc. of International Conference on Parallel Computing in Electrical Engineering*, 2004, S. 51–55

[74] Maya, S. ; Reynoso, R. ; Torres, C. ; Arias-Estrada, M.: Compact Spiking Neural Network Implementation in FPGA. In: Hartenstein, R.W. (Hrsg.) ; Grünbacher, H. (Hrsg.): *FPL '00: Proceedings of the The Roadmap to Reconfigurable Computing, 10th International Workshop on Field-Programmable Logic and Applications*, Springer-Verlag, 2000 (Lecture Notes in Computer Science 1896), S. 270–276

[75] McCulloch, W. ; Pitts, W.: A logical calculus of the ideas immanent in nervous activity. In: *Bulletin of Mathematical Biophysics* 5 (1943), S. 115–133

[76] Mead, C.: *Analog VLSI and Neural Systems.* Addison-Wesley, 1989

[77] Palm, G.: Neural Associative Memories. In: *Biological Cybernetics* 36 (1980), S. 19–36

[78] Palm, G.: *Neural Assemblies.* Springer-Verlag, 1982

[79] Pekmestzi, K.: Multiplexer-Based Array Multipliers. In: *IEEE Transactions on Computers* Vol. 48 (1999), Nr. 1, S. 15–23

[80] Razavi, B.: *Design of Analog CMOS Integrated Circuits.* McGraw-Hill, 2001

[81] Rosenblatt, F.: The perceptron : a probabilistic model for information storage and organization in the brain. In: *Psychological Reviews* 65 (1958), Nr. 6, S. 386–408

[82] Rubin, D. ; Chicca, E. ; Indiveri, G.: Characterizing the Firing Properties of an Adaptive Analog VLSI Neuron. In: Ijspeert, A. (Hrsg.) ; Murata, M. (Hrsg.) ; Wakamiya, N. (Hrsg.): *Proc. of BioADIT 2004* Bd. 3141, Springer-Verlag, 2004 (Lecture Notes in Computer Science), S. 189–200

[83] Rückert, U. ; Suhrmann, H.: Tolerance of a Binary Associative Memory Towards Stuck-at-Faults. In: Kohonen, T. (Hrsg.): *Artificial Neural Networks* Bd. 2. Elsevier, 1991, S. 1195–1198

[84] Sajonski, H. ; Smollich, A.: *Mikroskopische Anatomie.* Hirzel, 1972

[85] Schmidt, R.: *Neuro- u. Sinnesphysiologie.* Springer-Verlag, 1998

[86] Schmidt, R. ; Thews, G.: *Physiologie des Menschen.* 29. Auflage. Springer-Verlag, 2005

[87] Schrauwen, B. ; Van Campenhout, J.: Parallel hardware implementation of a broad class of spiking neurons using serial arithmetic. In: Verleysen, M. (Hrsg.): *Proceedings of the European Symposium on Artificial Neural Networks (ESANN)*, 2006, S. 623–628

[88] Schultz, S. ; Jabri, M.: Analogue VLSI 'integrate-and-fire' neuron with frequency adaptation. In: *Electronic Letters* 31 (1995), Nr. 16, S. 1357–1358

[89] Semiconductor Industry Association: *International Technology Roadmap for Semiconductors.* 2007

[90] Singer, S. ; Nicolson, G.: The fluid mosaic model of the structure of cell membranes. In: *Science* 175 (1972), S. 720–731

[91] Steinbuch, K.: Die Lernmatrix. In: *Kybernetik* 1 (1961), S. 36–45

[92] Torres-Huitzil, C. ; Girau, B.: Massively Distributed Digital Implementation of a Spiking Neural Network for Image Segmentation on FPGA. In: *Neural Information Processing – Letters and Reviews* 10, Nos. 4–6 (2006), S. 105–114

[93] Trepel, M.: *Neuroanatomie: Struktur und Funktion.* Elsevier, 2004

[94] Upegui, A. ; Peña-Reyes, C. ; Sanchez, E.: A functional spiking neuron hardware oriented model. In: Mira, J. (Hrsg.) ; Álvarez, J. R. (Hrsg.): *Computational Methods in Neural Modeling I* Bd. 2686, Springer-Verlag, 2003 (Lecture Notes in Computer Science), S. 136–143

[95] Upegui, A. ; Peña-Reyes, C. ; Sanchez, E.: A methodology for evolving spiking neural-network topologies on line using partial dynamic reconfiguration. In: *ICCI - International Conference on Computational Intelligence*, 2003

[96] Upegui, A. ; Peña-Reyes, C. ; Sanchez, E.: A Hardware Implementation of a Network of Functional Spiking Neurons with Hebbian Learning. In: Ijspeert, A. (Hrsg.) ; Murata, M. (Hrsg.) ; Wakamiya, N. (Hrsg.): *Proc. of BioADIT 2004* Bd. 3141, Springer-Verlag, 2004 (Lecture Notes in Computer Science), S. 399–409

[97] van Schaik, A.: Building blocks for electronic spiking neural networks. In: *Neural Networks* 14 (2001), S. 617–628

[98] Vereb, G. ; Szöllősi, J. ; Matkó, J. ; Nagy, P. ; Farkas, T. ; Vigh, L. ; Mátyus, L. ; Waldmann, T. ; Damjanovich, S.: Dynamic, yet structured: The cell membrane three decades after the Singer-Nicolson model. In: *Proc. Natl. Acad. Sci. USA* 100 (2003), Nr. 14, S. 8053–8058

[99] Wang, D.: A Comparison of CNN and LEGION Networks. In: *Proc. of IJCNN 2004*, 2004, S. 1735–1740

[100] Wei, Z. ; Yu, C.: Predictive Technology Model for Nano-CMOS Design Exploration. In: *ACM Journal on Emerging Technologies in Computing Systems* 3 (2007), Nr. 1, S. 1–17

[101] Weste, N. ; Eshragian, K.: *Principles of CMOS VLSI Design.* Addison-Wesley, 1985

[102] Wijekoon, J. ; Dudek, P.: Compact silicon neuron circuit with spiking and bursting behaviour. In: *Neural Networks* 21 (2008), Nr. 2–3, S. 524–534

[103] Wilk, G. ; Wallace, R. ; Anthony, J.: High-κ gate dielectrics: Current status and materials properties considerations. In: *Jounal of Applied Physics* 89 (2001), Nr. 10, S. 5243–5275

[104] Zhai, B. ; Nazhandali, L. ; Olson, J. ; Reeves, A. ; Minuth, M. ; Helfand, R. ; Pant, S. ; Blaauw, D. ; Austin, T.: A 2.60 pJ/Inst Subthreshold Sensor Processor for Optimal Energy Efficiency. In: *Proc. of IEEE Symposium on VLSI Circuits*, 2006, S. 154–155

[105] Zhao, J. ; Shawe-Taylor, J. ; Daalen, M. van: Learning in Stochastic Bit Stream Neural Networks. In: *Neural Networks* 9 (1996), S. 991–998

Eigene Publikationen

[106] Eickhoff, R. ; Kaulmann, T. ; Rückert, U.: Impact of Shrinking Technologies on the Activation Function of Neurons. In: *Proc. of International Conference on Artificial Neural Networks (ICANN)* Bd. 4668, Springer-Verlag, 2007 (Lecture Notes in Computer Science), S. 501–510

[107] Eickhoff, R. ; Kaulmann, T. ; Rückert, U.: Neural Inspired Architectures for Nanoelectronics. In: *Proc. of International Work-Conference on Artificial Neural Networks (IWANN)* Bd. 4507, Springer-Verlag, 2007 (Lecture Notes in Computer Science), S. 414–421

[108] Eickhoff, R. ; Kaulmann, T. ; Rückert, U.: SIRENS: A Simple Reconfigurable Neural Hardware Structure for artificial neural network implementations. In: *Proc. of International Joint Conference on Neural Networks (IJCNN)*, 2006, S. 2830–2837

[109] Kaulmann, T. ; Dikmen, D. ; Rückert, U.: A Digital Framework for Pulse Coded Neural Network Hardware with Bit-Serial Operation. In: *Proc. of the 7th International Conference on Hybrid Intelligent Systems (HIS)*, IEEE Computer Society, 2007, S. 302–307

[110] Kaulmann, T. ; Ferber, M. ; Witkowski, U. ; Rückert, U.: Analog VLSI Implementation of Adaptive Synapses in Pulsed Neural Networks. In: Cabestany, J. (Hrsg.) ; Prieto, A. (Hrsg.) ; Sandoval, D. (Hrsg.): *Proceedings of the 8th International Work-Conference on Artificial Neural Networks (IWANN)* Bd. 3512, Springer-Verlag, 2005 (Lecture Notes in Computer Science), S. 455–462

[111] Kaulmann, T. ; Löffler, A. ; Rückert, U.: A Control Approach to a Biophysical Neuron Model. In: *Proc. of International Conference on Artificial Neural Networks (ICANN)* Bd. 4668, Springer-Verlag, 2007 (Lecture Notes in Computer Science), S. 529–538

[112] Kaulmann, T. ; Lütkemeier, S. ; Rückert, U.: IAF Neuron implementation for Mixed-Signal PCNN Hardware. In: *Proc. of International Work Conference on Artificial Neural Networks (IWANN)* Bd. 4507, Springer-Verlag, 2007 (Lecture Notes in Computer Science), S. 447–454

[113] Lütkemeier, S. ; Kaulmann, T. ; Rückert, U.: A Sub-200mV 32bit ALU with 0.45pJ/instruction in 90nm CMOS. In: *Proc. of Semiconductor Conference Dresden (SCD)*, 2009. – eingeladener Vortrag

Index

ADP, 9
Aktionspotential, 14, 56
Antwortfunktion, 21
Assoziativspeicher, 125–141
    BiNAM, 126
Atmungskette, 6, 9
ATP, 7
Axon-Hillock-Schaltung, 35
Axonhügel, 16

Citratzyklus, 9

Depolarisation, 13
Dielektrikum, 12
Diffusion, 7, 13, 42
Diffusionsdruck, 13

Energieumsatz, 42
EPSC, 17
EPSP, 17, 21

Fehlerklassen, 128
Fehlertoleranz, 128–141
Fehlertoleranz von Assoziativspeichern
    Fehler im Eingabevektor, 128
    Fehler im Schwellenelement, 130
    Fehler in der Gewichtsmatrix, 130
    Fehler in der Summenbildung, 131
    Kombinationsfehler, 131
    Stuck-at Fehler, 134–141
Feuerschwelle, 14, 21
Feuerzeitpunkt, 21

Gewichtsmatrix, 126
Gleichgewichtspotential, 13
Glycolyse, 9
Goldmann-Gleichung, 43
Granularzelle, 10
Grundmodell, 42
Grundumsatz, 41

Hyperpolarisation, 13

Implementierung
    analog, 76, 117
    digital, bitparallel, 103
    digital, bitseriell, 110
    digital, zählerbasiert, 112
Informationsumsatz, 42
Ionenkanal, 6
IPSC, 17
IPSP, 17, 21

Körnerzelle, 10

Leaky Integrate and Fire Neuron, 77
Lernrate, 126
Lernregel, 126
    Hebb'sche Regel, 126
LIAF Neuron
    analoge Implementierung, 76
    ASIC optimierter Entwurf, 110
    Definition, 22
    digitale Implementierung, 102
    FPGA optimierter Entwurf, 102
Lipid, 6
Lipiddoppelschicht, 6
Ljapunov-Funktion, 51

Membran, 6
Membrankapazität, 11
Membranpotential, 13, 23

NADH, 9
NaK-ATPase, Natrium-Kalium-Pumpe, 7
Nernst-Gleichung, 13, 42

Nernstpotential, 42
Nervenzelle, 10
Neuron, 10
Neuronenmodell
    Leaky Integrate and Fire Modell, 22
    mathematische Betrachtungen, 63
    Spike Response Modell, 21
Neurotransmitter, 17

PCNN, 19
Phasenplot, 51, 58
Phosphatrest, 8
Pumpzyklus, 7
Purkinje Zelle, 10

Ranvier-Schürring, 16
Reizweiterleitung, 16
Repolarisation, 13
Ressourcenbedarf, 75
Rezeptor, 54
Ruhepotential, 13

Schwellenpotential, 14
Simulator, 19
Simulink, 163
SIRENS, 102, 103
Skalierungsregeln, 84, 161
SRAM, 86
SRM Neuron
    Definition, 21
Standardzellen, 98
steady-state, 13
Subschwellen-Bibliothek, 95
Subschwellenverhalten, 82
Synapse, 84
Synapsenmodell, 52

Transmitter, 16, 54
Transportmechanismus
    aktiver Transport, 47
    Natrium-Kalium-Pumpe, 48
    passiver Transport, 42

Übertragungskennlinie, 68
Unterschwellenverhalten, 15
Unterschwellverhalten, 64

Zellatmung, 6
Zelle, 6
Zellkern, 6
Zellmembran, 6