Conception and First Implementation of Novel Sensory Signal Conditioning and Digital Conversion Electronics Based on Spiking Neuron Ensembles for Inherently Robust Processing in Aggressively Scaled Integration Technologies

Konzeptentwurf und erste Implementierung einer neuartigen Elektronik für Sensorsignalkonditionierung und Digitalwandlung basierend auf pulsenden Neuronenstrukturen für inhärent robuste Verarbeitung in Integrationstechnologien extrem reduzierter Strukturgrößen

vom
Fachbereich Elektrotechnik und Informationstechnik
der Technischen Universität Kaiserslautern
zur Verleihung des akademischen Grades

Doktor der Ingenieurwissenschaften (Dr.-Ing.)

genehmigte Dissertation

von
Abhaya Chandra Kammara Subramanyam, M.Sc
geb. in Tirupathi (Indien)

D 386

Eingereicht am: 2. Dezember 2015
Tag der mündlichen Prüfung: 17. Juli 2017
Dekan des Fachbereichs: Prof. Dr.-Ing. Ralph Urbansky

Promotionskommission
Vorsitzender: Jun.-Prof. Dr.-Ing. Daniel Görges
1. Berichterstattender: Prof. Dr.-Ing. Andreas König
2. Berichterstattender: Prof. Dr.-Ing. habil. Roland Werthschützky

Erklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit selbst angefertigt und verfasst habe und alle benutzen Hilfsmittel in der Arbeit angegeben habe. Die vorliegende Dissertation oder Teile hiervon würden noch nicht als Prüfungsarbeit für eine staatliche oder andere wissenschaftliche Prüfung eingereicht.

Kaiserslautern, den 2. Dezember 2015

Abhaya Chandra Kammara Subramanyam

Abstract

"In contemporary electronics 80% of a chip may perform digital functions but the 20% of analog functions may take 80% of the development time." [1]. Aggravating this, the demands on analog design is increasing with rapid technology scaling. Most designs have moved away from analog to digital domains, where possible, however, interacting with the environment will always require analog to digital data conversion. Adding to this problem, the number of sensors used in consumer and industry related products are rapidly increasing. Designers of ADCs are dealing with this problem in several ways, the most important is the migration towards digital designs and time domain techniques. Time to Digital Converters (TDC) are becoming increasingly popular for robust signal processing. Biological neurons make use of spikes, which carry spike timing information and will not be affected by the problems related to technology scaling. Neuromorphic ADCs still remain exotic with few implementations in sub-micron technologies Table 2.7. Even among these few designs, the strengths of biological neurons are rarely exploited. From a previous work [2], LUCOS, a high dynamic range image sensor, the efficiency of spike processing has been validated. The ideas from this work can be generalized to make a highly effective sensor signal conditioning system, which carries the promise to be robust to technology scaling.

The goal of this work is to create a novel spiking neural ADC as a novel form of a Multi-Sensor Signal Conditioning and Conversion system, which

- Will be able to interface with or be a part of a System on Chip with traditional analog or advanced digital components.
- Will have a graceful degradation.
- Will be robust to noise and jitter related problems.
- Will be able to learn and adapt to static errors and dynamic errors.
- Will be capable of self-repair, self-monitoring and self-calibration

Sensory systems in humans and other animals analyze the environment using several techniques. These techniques have been evolved and perfected to help the animal survive. Different animals specialize in different sense organs, however, the peripheral neural network architectures remain similar among various animal species with few exceptions. While there are many biological sensing techniques present, most popularly used engineering techniques are based on intensity detection, frequency detection, and edge detection. These techniques are used with traditional analog processing (e.g., color

sensors using filters), and with biological techniques (e.g. LUCOS chip [2]). The localization capability of animals has never been fully utilized.

One of the most important capabilities for animals, vertebrates or invertebrates, is the capability for localization. The object of localization can be predator, prey, sources of water, or food. Since these are basic necessities for survival, they evolve much faster due to the survival of the fittest. In fact, localization capabilities, even if the sensors are different, have convergently evolved to have same processing methods (coincidence detection) in their peripheral neurons (for e.g., forked tongue of a snake, antennae of a cockroach, acoustic localization in fishes and mammals). This convergent evolution increases the validity of the technique. In this work, localization concepts based on acoustic localization and tropotaxis are investigated and employed for creation of novel ADCs.

Unlike intensity and frequency detection, which are not linear (for e.g. eyes saturate in bright light, loose color perception in low light), localization is inherently linear. This is mainly because the accurate localization of predator or prey can be the difference between life and death for an animal.

Figure 1 visually explains the ADC concept proposed in this work. This has two parts. (1) Sensor to Spike(time) Conversion (SSC), (2) Spike(time) to Digital Conversion(SDC). Both of the structures have been designed with models of biological neurons. The combination of these two structures is called SSDC.

To efficiently implement the proposed concept, a comparison of several biological neural models is made and two models are shortlisted. Various synapse structures are also studied. From this study, Leaky Integrate and Fire neuron (LIF) is chosen since it fulfills all the requirements of the proposed structure. The analog neuron and synapse designs from Indiveri et. al. [3], [4] were taken, and simulations were conducted using cadence and the behavioral equivalence with biological counterpart was checked. The LIF neuron had features, that were not required for the proposed approach. A simple LIF neuron stripped of these features and was designed to be as fast as allowed by the technology.

The SDC was designed with the neural building blocks and the delays were designed using buffer chains. This SDC converts incoming Time Interval Code (TIC) to sparse place coding using coincidence detection. Coincidence detection is a property of spiking neurons, which is a time domain equivalent of a Gaussian Kernel. The SDC is designed to have an online reconfigurable Gaussian kernel width, weight, threshold, and refractory period. The advantage of sparse place codes, which contain rank order coding was

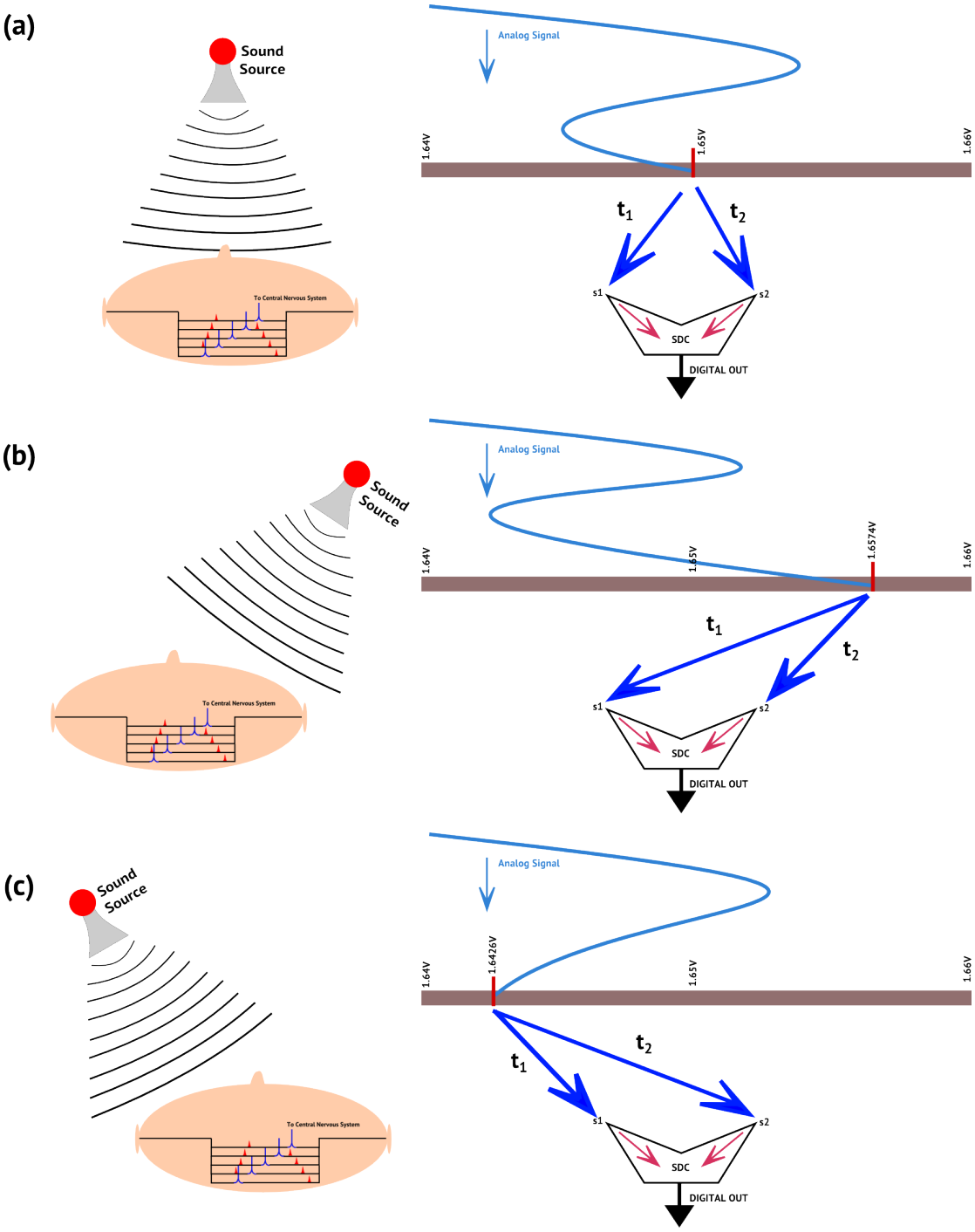


FIGURE 1: ADC as a localization problem (right), Jeffress model of sound localization visualized (left). The values t_1 and t_2 indicate the time taken from the source to s1 and s2 respectively.

described in our work [5]. A time based winner take all circuit with memory was created based on a previous work [6] for reading out of sparse place codes asynchronously.

The SSC was also initially designed with the same building blocks. Additionally, a differential synapse was designed for better SSC. The sensor element considered was

a Wheatstone full bridge AMR sensor AFF755 from Sensitec GmbH. A reconfigurable version of the synapse was also designed for a more generic sensor interface.

The first prototype chip SSDCα was designed with 257 modules of coincidence detectors realizing the SDC and the SSC. Since the spike times are the most important information, the spikes can be treated as digital pulses. This provides the capability for digital communication between analog modules. This creates a lot of freedom for use of digital processing between the discussed analog modules. This advantage is fully exploited in the design of SSDCα. Three SSC modules are multiplexed to the SDC. These SSC modules also provide outputs from the chip simultaneously. A rising edge detecting fixed pulse width generation circuit is used to create pulses that are best suited for efficient performance of the SDC. The delay lines are made reconfigurable to increase robustness and modify the span of the SDC. The readout technique used in the first prototype is a relatively slow but safe shift register. It is used to analyze the characteristics of the core work. This will be replaced by faster alternatives discussed in the work. The area of the chip is 8.5 mm^2. It has a sampling rate from DC to 150 kHz. It has a resolution from 8-bit to 13-bit. It has 28,200 transistors on the chip. It has been designed in 350 nm CMOS technology from ams. The chip has been manufactured and tested with a sampling rate of 10 kHz and a theoretical resolution of 8 bits. However, due to the limitations of our Time-Interval-Generator, we are able to confirm for only 4 bits of resolution.

The key novel contributions of this work are

- Neuromorphic implementation of AD conversion as a localization problem based on sound localization and tropotaxis concepts found in nature.
- Coincidence detection with sparse place coding to enhance resolution.
- Graceful degradation without redundant elements, inherent robustness to noise, which helps in scaling of technologies
- Amenable to local adaptation and self-x features.

Conceptual goals have all been fulfilled, with the exception of adaptation. The feasibility for local adaptation has been shown with promising results and further investigation is required for future work. This thesis work acts as a baseline, paving the way for R&D in a new direction. The chip design has used 350 nm ams hitkit as a vehicle to prove the functionality of the core concept. The concept can be easily ported to present aggressively-scaled-technologies and future technologies.

Kurzfassung

"In der gegenwärtigen integrierten Elektronik sind typisch 80 % der Funktionalität eines Chips digital und nur ca. 20 % analog, aber diese 20 % der analogen Funktionalität können bis zu 80 % der Entwicklungszeit in Anspruch nehmen." [1]. Erschwerend kommt dabei hinzu, dass die Anforderungen an den Analogentwurf mit zunehmender Technologieskalierung immer höher werden. Die meisten Entwürfe haben, soweit möglich, analoge zugunsten von digitalen Bereichen ersetzt, jedoch in der Wechselwirkung mit der Umgebung bleibt u.a. die Erfassung analoger Daten und deren Konvertierung in digitale Repräsentation unverzichtbar. Darüber hinaus steigt auch die Anzahl der in Verbraucher- und Industriegütern eingesetzten Sensoren rapide. Entwerfer von ADCs müssen sich mit diesem Problem in mehrfacher Hinsicht auseinandersetzen. Die wichtigste Problemstellung ist die Umstellung auf dominant digitalen Entwurf und Zeitbereichsverfahren. Zeit-zu-Digital-Wandler (Time-to-Digital-Converter, TDC) werden immer beliebter für eine robuste Signalverarbeitung. Biologische Neuronen nutzen Pulse oder Spikes, die im spezifischen zeitlichen Auftreten Informationen tragen und nicht durch die oben angesprochenen Probleme der Technologieskalierung gravierend berührt werden. Neuromorphe ADCs erscheinen immer noch als exotische Ansätze mit wenigen Implementierungen in Sub-Mikrometer-Technologien (Tabelle 2.7). Selbst unter diesen wenigen Entwürfe sind die Stärken von biologischen Neuronen und deren technischer Modelle kaum erschlossen. Aus einer früheren Arbeit [2] zu einem Bildsensor hohen Dynamikbereichs konnten wertvolle Erkenntnisse zur technischen Nutzung der Spike-Verarbeitung in der Sensorik gewonnen werden. Die Ideen aus dieser Arbeit können verallgemeinert werden, um eine höchst effektive Sensorsignalverarbeitung zu realisieren, die verspricht, auch robust gegen Technologieskalierung zu sein.

Das Ziel dieser Arbeit ist es, einen neuartigen 'spikenden' neuronalen ADC als innovative Form eines Multi-Sensor Signalkonditionierungs- und Wandlungssystems zu erschaffen, welches

- in der Lage sein wird, mit herkömmlichen analogen oder fortgeschrittenen digitalen Komponenten unmittelbar oder als Teil eines Systems-on-Chip zusammen zu wirken.
- Graceful Degradation aufweist.
- robust gegen Rauschen und Jitter Probleme sein wird.
- in der Lage sein wird, zu lernen und sich im Hinblick auf Kompensation statischer und dynamischer Fehler zu adaptieren.

- in der Lage sein wird, Selbstreparatur-, Selbstüberwachungs- und Selbstkalibrierungsmechanismen etc. zu implementieren.

Sensorische Systeme bei Menschen und Tieren analysieren die Umwelt mittels verschiedener Techniken. Diese Techniken wurden ent- wickelt und perfektioniert, um das Überleben des Tieres sicher zustellen. Verschiedene Tiere bzw. Tierarten haben sich zwar hinsichtlich verschiedener Sinnesorgane spezialisiert , weisen jedoch mit wenigen Ausnahmen hinsichtlich der die peripheren neuronalen Netzwerkarchitekturen große Ähnlichkeiten auf. Während viele biologische Akquisitionstechniken vorliegen, sind die am häufigsten verwendeten Techniken auf Intensitätsdetektion, Frequenzdetektion, und Kantenerkennung basierend. Diese Techniken werden mit traditioneller Analogverarbeitung (z.B. Farbsensoren mit Filtern) und mit biologisch motivierten Techniken (z.B. LUCOS chip [2]) eingesetzt. Die Lokalisierungsfähigkeit der Tiere wurde bislang noch nie vollständig technisch ausgenutzt.

Eine der wichtigsten Funktionen für Tiere, Wirbeltiere oder Wirbellose, ist die Fähigkeit zur Lokalisierung. Die Aufgabe der Lokalisierung kann die Findung von Räubern, Beute, Quellen von Wasser oder Nahrung sein. Da es sich dabei um Grundfertigkeiten zum Überleben handelt, entwickeln diese sich viel schneller durch das Prinzip des Überlebens des Stärkeren. Tatsächlich haben sich Lokalisierungsfunktionen, auch wenn die Sensoren unterschiedlich sind, konvergent in Richtung gleicher Verarbeitungsverfahren entwickelt, wie Koinzidenzerfassung in ihren peripheren Neuronen (für z.B. die gespaltene Zunge einer Schlange, die Antennen einer Kakerlake, oder die akustische Lokalisierung in Fischen und Säugetieren). Diese konvergente Evolution erhöht die Wirksamkeit des Verfahrens. In dieser Arbeit werden Lokalisierungskonzepte auf Basis von akustischer Lokalisierung und Tropotaxis untersucht und für die Schaffung von neuartigen ADCs eingesetzt.

Im Gegensatz zu Intensitäts- und Frequenzerkennung, die nicht linear sind (zum Beispiel erreichen die Augen in hellem Licht eine Sättigungsgrenze, oder verlieren die Farbwahrnehmung bei schwachem Licht), ist die Lokalisierung von Natur aus linear. Dies ist vor deshalb, da die genaue Lokalisierung von Räuber und Beute den Unterschied zwischen Leben und Tod für ein Tier kann. Bild 2 veranschaulicht visuell das in dieser Arbeit vorgeschlagene ADC-Konzept. Dieses hat zwei Teile. (1) Sensor nach Spike (Zeit) Wandlung (SSC), (2) Spike (Zeit) nach Digital-Wandlung (SDC) . Beide Strukturen wurden anhand von Modellen biologischer Neuronen entworfen. Die Kombination dieser beiden Strukturen wird als SSDC bezeichnet.

Zur effizienten Durchführung des vorgeschlagenen Konzepts wird ein Vergleich von mehreren biologischen neuronalen Modellen vorgenommen und zwei Modelle in die engere

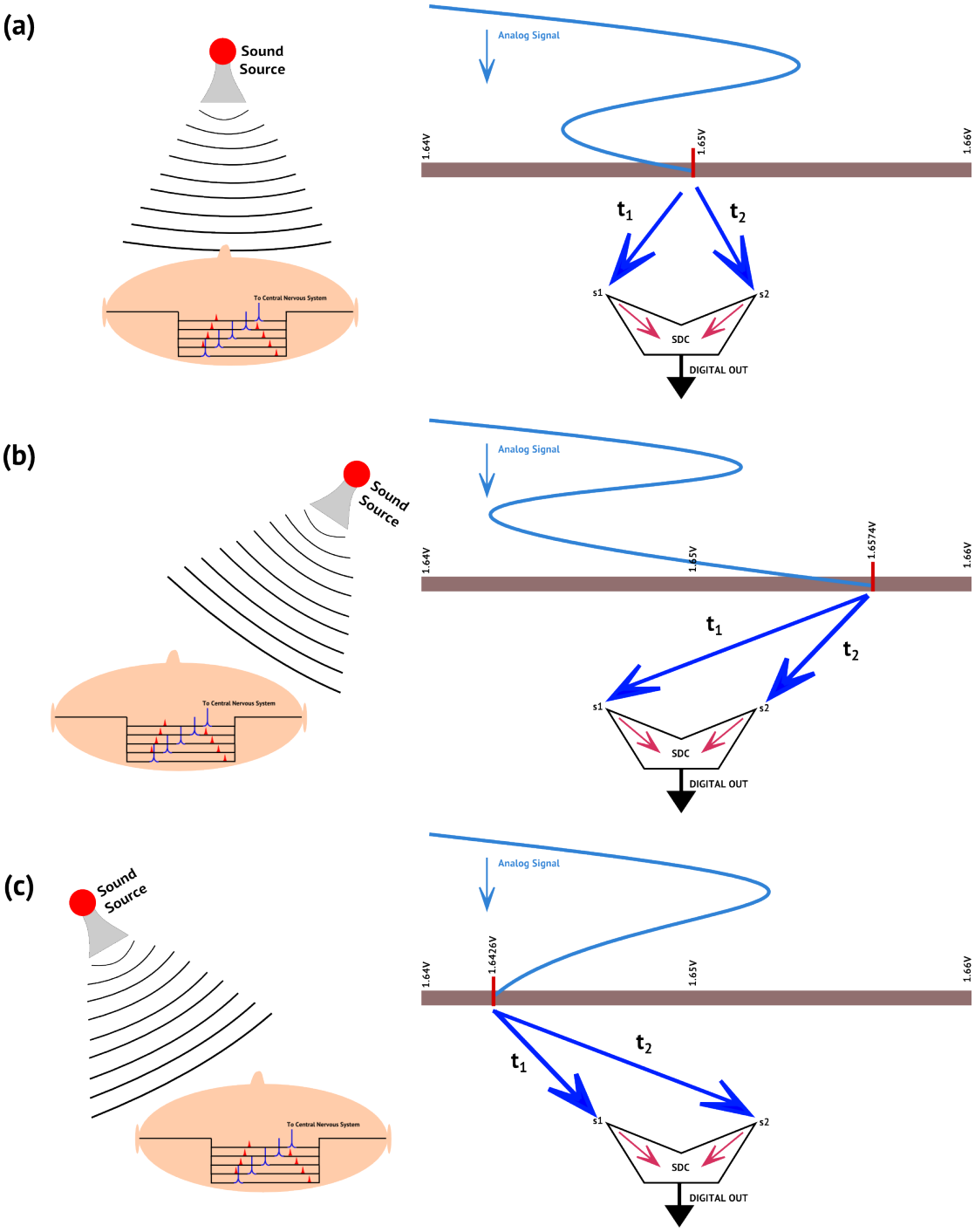


FIGURE 2: ADC als Lokalisierungsproblem (rechts), Jeffress Modell zur akustischen Lokalisierung (links). Die Werte t_1 und t_2 repräsentieren die Laufzeit von der Quelle zu s1 bzw. s2.

Wahl gebracht. Verschiedene Synapsenstrukturen werden ebenfalls untersucht. In dieser Studie wurde das Leaky Integrate and Fire Neuron (LIF) gewählt, da es alle Anforderungen der vorgeschlagenen Struktur erfüllt. Die analogen Neuronen und Synapse von Indiveri et. al. [3], [4] wurden für den Entwurf herangezogen, Simulationen wurden mit Cadence durchgeführt und die Verhaltensäquivalenz mit dem biologischen Mo-

dell überprüft. Die ursprünglichen LIF Neuron hatten Merkmale, die nicht für den vorgeschlagenen Ansatz erforderlich waren. Ein vereinfaches LIF Neuron wurde entwickelt, und hinsichtlich seiner Verarbeitungsgeschwindigkeit bis an die Grenzen der konkret verwendeten Technologie optimiert.

Der SDC wurde mit den neuronalen Bausteinen konzipiert und die Verzögerungen wurden als Pufferketten ausgelegt. Das SDC konvertiert eingehende Time Interval Code (TIC) zu spärlichen Ortskodierung mit Koinzidenzerfassung. Koinzidenzerfassung ist eine Eigenschaft der spikenden Neuronen, die ein Zeitbereichsäquivalent eines Gauß-Kerns darstellt. Der SDC wurde entworfen, um online rekonfigurierbare Gauß-Kern-Breite, Gewicht, Schwelle und Refraktärzeit zu haben. Der Vorteil der spärlichen Ortskodierung, die Rangordnungskodierung enthalten, wurde in unserer Vorarbeit [5] beschrieben. Ein zeitbereichsbasierter Gewinner-nimmt-Alles-Mechanismus mit Speicher wurde auf der Grundlage einer früheren Arbeit [6] zum asynchronen Auslesen der spärlichen Ortskodierung geschaffen .

Der SSC wurde zunächst mit den gleichen Bausteinen entwickelt. Zusätzlich wurde mit einer Differenzsynapse eine weitere, bessere SSC gestaltet. Das betrachtete Sensorelement war eine Wheatstone-Vollbrücke mit einem AMR-Sensor AFF755 von Sensitec GmbH. Eine rekonfigurierbare Version der Synapsen wurde ebenfalls in Richtung einer generischen Sensorschnittstelle entwickelt.

Der erste Prototyp-Chip SSDCα wurde mit 257 Modulen der Koinzidenzdetektoren zur Realisierung der SDC und des SSC konzipiert. Da das Spike-Timing und nicht die Pegel die wichtige Informationen beinhalten, können diese Spikes als digitale Impulse behandelt werden. Dies stellt die Fähigkeit zur digitalen Kommunikation zwischen den Analogmodulen her. Das schafft viel Freiraum für die Verwendung digitalen Verarbeitung zwischen den diskutierten Analogmodulen. Dieser Vorteil wurde vollständig in dem Design des SSDCα ausgenutzt. Drei SSC-Module werden an die SDC gemultiplext. Diese SSC-Module bieten gleichzeitig auch Ausgänge auf dem Chip. Eine Schaltung zur Deketion steigender Flanken wird zur Erzeugung fester Impulsbreiten verwendet, um optimierte Impulse für den SDC zu schaffen. Die Verzögerungsleitungen sind rekonfigurierbar, um die Robustheit zu erhöhen und die Spanne des SDC anzupassen. Das Ausleseverfahren in der ersten Prototyp ist über ein relativ langsames, aber sicheres Schieberegister realisiert. Dieses wird verwendet, um die Eigenschaften der Kernzellen zu analysieren. Dies wird durch schnellere Alternativen, wie in der Arbeit diskutiert, ersetzt werden. Die Fläche des Chips beträgt 8,5 mm ^{2}. Er hat eine Abtastrate von DC bis 150 kHz und eine Auflösung von 8 Bit bis zu 13 Bit. Es sind 28.200 Transistoren auf dem Chip integriert, welcher in der 350 nm-CMOS-Technologie von ams entwickelt wurde. Der Chip wurde hergestellt und getestet, mit einer Abtastrate von 10 kHz

und einer theoretische Auflösung von 8 Bits. Aufgrund der Einschränkungen unseres Time-Interval-Generators können wir jedoch nur 4 Bits Auflösung bestätigen.

Die wichtigsten neuen Beiträge dieser Arbeit sind

- Neuromorphische Umsetzung der AD-Wandlung als Lokalisierungsproblem auf der Grundlage in der Natur zu findender akustischer Lokalisierung und Tropotaxis-konzepten
- Koinzidenzerfassungs mit spärlicher Ortskodierung zur Auflösungsverbesserung.
- Graceful Degradation ohne redundante Elemente, inhärente Robustheit gegenüber Rauschen, und den Herausforderungen durch die fortschreitende Technologieskalierung
- Zugänglichkeit des Ansatzes für lokale Adaption und Self-x Funktionalität

Die gesteckten konzeptionellen Ziele wurden alle erfüllt, mit Ausnahme der Adaption. DDie prinzipielle Machbarkeit für lokale Adaption ist mit ersten, vielversprechenden Ergebnissen jedoch gezeigt worden. Weitere Untersuchungen sind in künftigen Arbeit erforderlich.Diese Doktorarbeit dient als Ausgangsbasis und Ebnung des Weges für eine neue Richtung in Forschung und Entwicklung. Das erste Chip-Design dient dabei als Vehikel, um die Funktionalität des Grundkonzepts beweisen. Das vorgestellte Konzept verspricht insbesondere eine einfache Übertragbarkeit auf stark skalierte gegenwärtige und kommende Integrationstechnologien.

Acknowledgements

I would like to thank Prof. Dr.-Ing. Andreas König for suggesting the thesis topic and his guidance throughout my research work. Our discussions on the thesis and other scientific topics helped me become the researcher I am today. I am deeply indebted to him for his interest in my work, his encouragement when results were scarce, providing crucial suggestions which helped in shaping the work.

I would like to thank my second reviewer Prof. Dr.-Ing. habil. Roland Werthschützky for showing interest in my work.

I would like to thank Thomas Gräf for designing and assembling the shield boards required for the chip and in his help with German language issues.

I would like to thank my wife for her immense support during this journey. I would like to thank my parents for their support and encouragement.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Micro and nano technologies have become ubiquitous in the last decade, and we have been seeing a rapid increase in the growth of sensory technologies. Sensors had an unpredicted growth of 222% in the period between 2007 and 2012 [61]. The main reasons for this can be attributed to the introduction of IPhone and Wii. The mobile market is still growing at a rapid pace and with each device carrying at least six sensors, the projected growth is not unexpected. In addition to these we have the recent burst in interest for IoT (Internet of Things), which is basically wirelessly connected sensors embedded in devices, Driver assistance systems, Driver-less cars etc. While this explains about the growth of sensors in consumer sector, the industrial sector has also been moving towards process automation using sensors for the last few decades. The German government has provided a vision for Industrie 4.0 [62], which encompasses IOT, Cyber-physical systems, Internet of Services, which could lead to smart manufacturing. United States of America has a similar initiative called Smart (SMLC) [63]. The Trillion sensors road map has provided a vision that there will be trillion sensors within the next decade [61].

The ITRS roadmap 2013 [64] discusses about the various predicted changes for integrated chips in the next decades. The main focus of this discussion is about the rapid scaling of technologies, which would reach 5 nm in the next 10 years, This will be the fundamental limit of 2D scaling. The discussion also focuses on fundamental changes to the transistor design including the recent introduction of Tri-gate(FinFet) transistors by intel[65]. 3D design of ICs will be the focus for power scaling. These changes inevitably increase the vulnerability of the devices to hard and soft defects, which compromise the dependability and accuracy of the measurement system both statically and dynamically.

The roadmap also discusses about the sensor trends for handheld devices, which would contain more than 20 sensors embedded in the devices, The article also discusses about wearable technologies, which are dependent on sensors. These clearly show the rising importance of robust sensor conditioning systems.

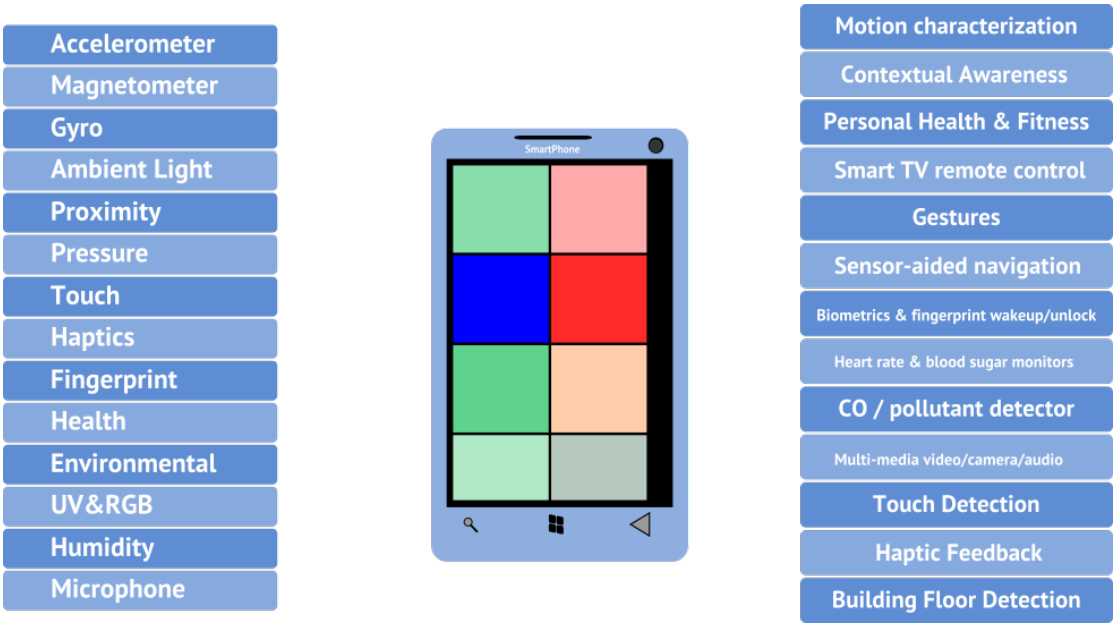


FIGURE 1.1: ITRS Sensor Trends for Handsets (taken from presentation of P.Gargini for IEEE March 11, 2014)

1.1.1 Conventional Signal Conditioning Systems

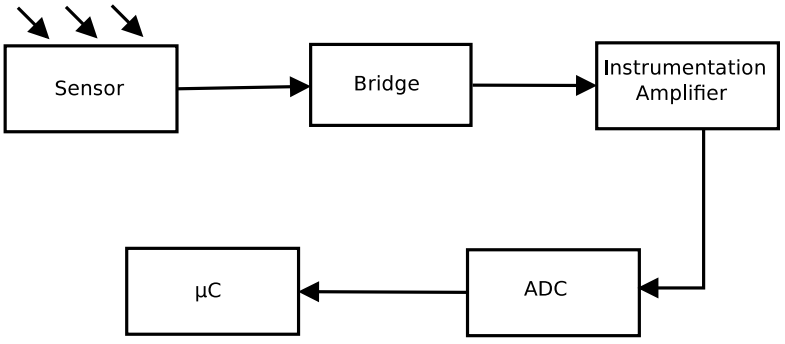


FIGURE 1.2: Conventional Signal Conditioning Systems

The conventional signal conditioning systems typically make use of instrumentation amplifiers, which are highly stable signal processing amplifiers. They are capable of reducing static errors and offset errors using auto zeroing and chopping techniques [66]. There are instrumentation amplifiers, which are capable of self calibration to compensate the static and dynamic deviations of the system. However, these systems are amplitude coded and hence more vulnerable to the rapid scaling of technologies [67] [68].

1.1.2 Self-x Signal Conditioning Systems

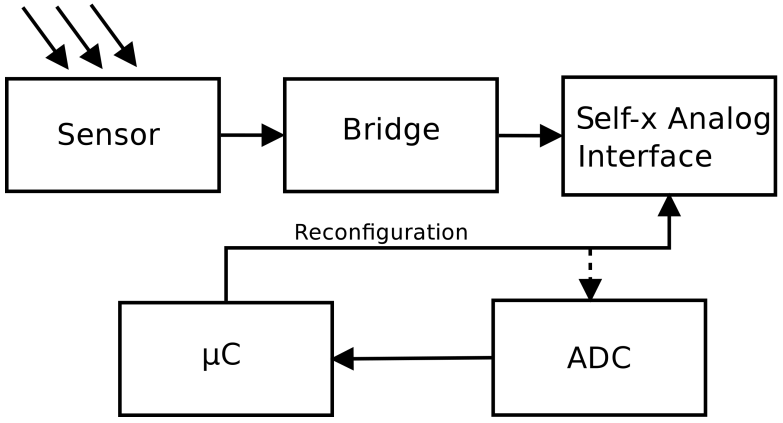


FIGURE 1.3: Self-x Signal Conditioning Systems

Reconfigurable and self-x systems make use of programmable components where components can be resized as required to manipulate the supply voltage, resolution and other specifications of the system, They also have the capability for learning and adaptation. ICs like PGA309 from Texas instruments, Zooming ADC from Semtech etc have various features like providing calibration for zero, span, zero drift, span drift, sensor linearization errors etc. These methods are relatively less vulnerable, however, they still use amplitude coded techniques, which are inherently vulnerable.

1.1.3 Problems and Current Solutions

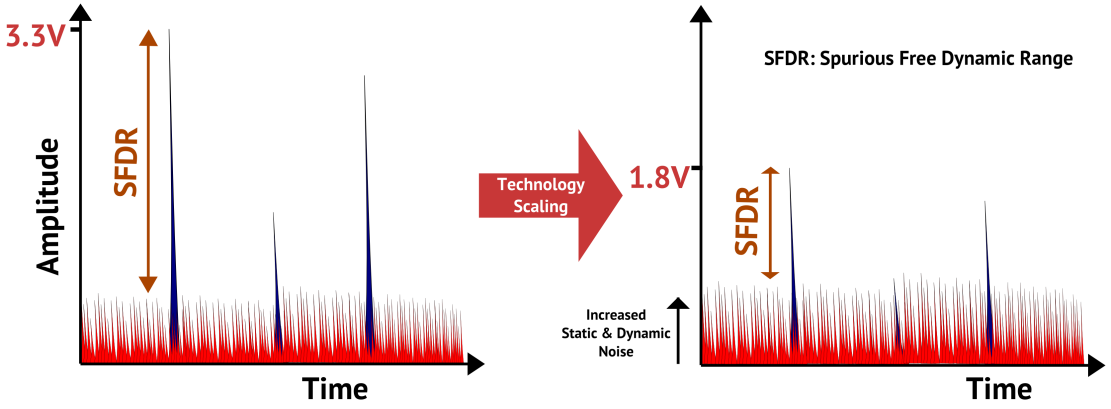


FIGURE 1.4: Problems in Amplitude Domain Signal Processing

Figure 1.4 shows one of the major problems of working in amplitude domain. The technology scaling is drastically increasing the signal-to-noise-ratio (SNR). This is one of the major reasons for the discontinuation of voltage scaling with technology scaling. As these challenges increase, the designers are moving towards digital design and time

domain. Figure 1.5 shows the inherent advantages of working with time-delay based time domain architectures.

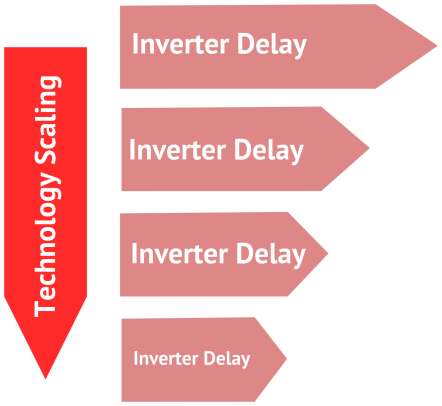


FIGURE 1.5: Advantages of Time Domain Signal Processing

1.1.4 Time Domain Systems

There are many pulse, time and frequency (rate) coded approaches, which are relatively more robust to noise. Figure 1.6 shows the architecture of a basic delay line TDC. These simple structures offer more robust solutions at submicron technologies than amplitude domain techniques. It is not surprising that these approaches have become recently popular in the industry [69].

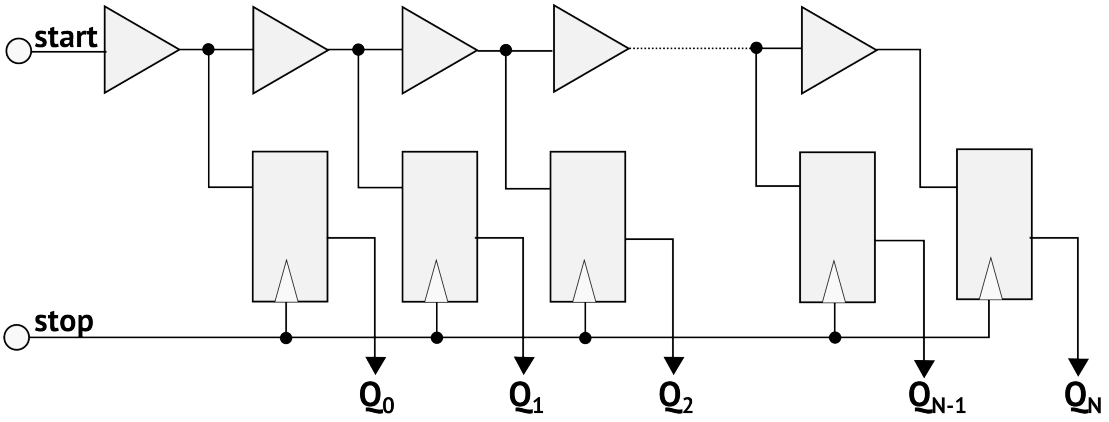


FIGURE 1.6: Basic Architecture of Delay-Line TDCs

1.2 Goals of the Thesis

The Figure 1.7 shows the ISE vision for the design of automated intelligent sensor systems. This work will focus on providing a novel architecture, which focuses on various goals described below. This architecture will replace the "Reconfigurable Electronics"

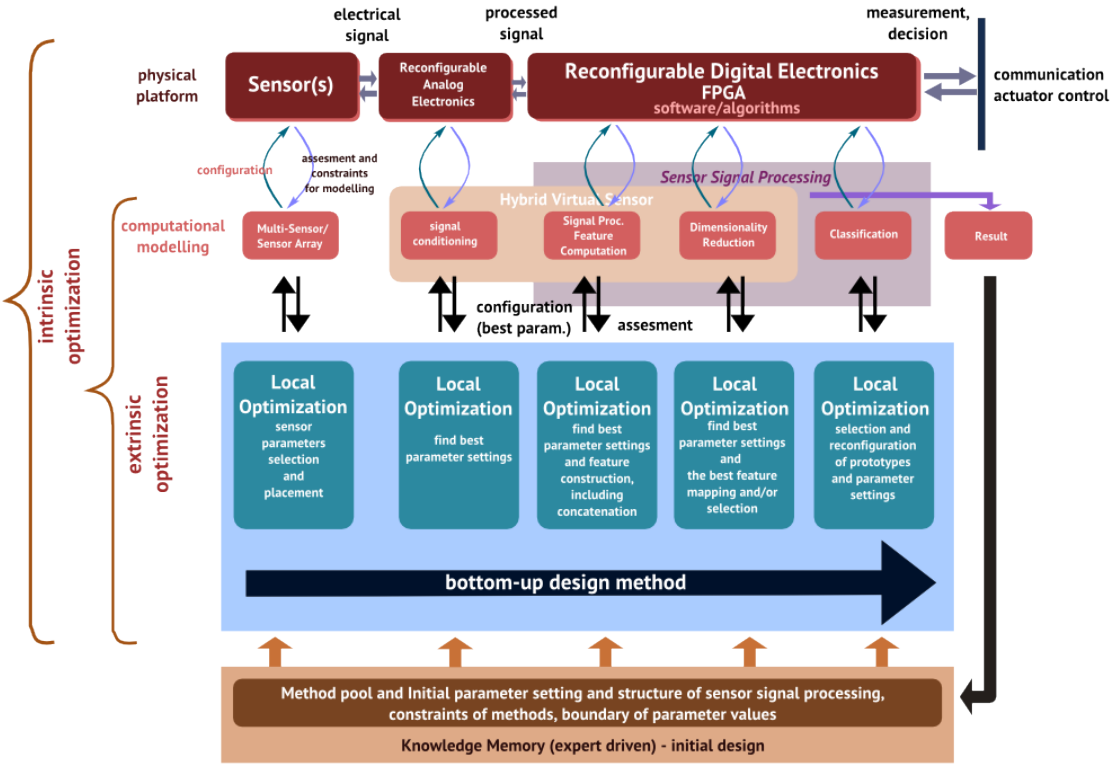


FIGURE 1.7: ISE Ecosystem: Intelligent Sensor System Design

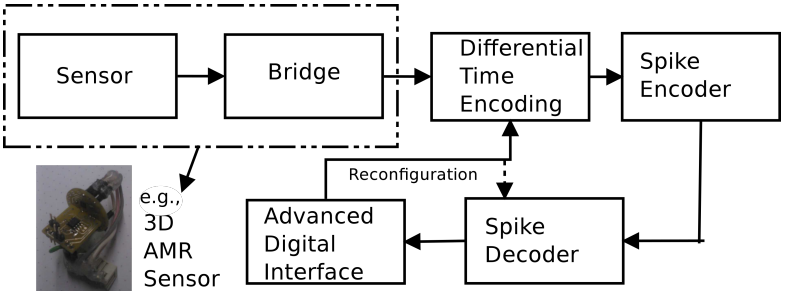


FIGURE 1.8: Neural Signal Conditioning Systems

with "Bio-mimetic Neural Information Processing" with signal conditioning in the time domain rather than analog and digital domains as shown in Figure 1.7 and as described in page 25 of AMA technology roadmap [70].

Sensory systems in living beings have evolved over a long period of time. These systems have become robust to various environmental problems. Biological models of neural networks called spiking neural networks, which use time based codes like rank coding, place coding, and rate coding. These neurons with the sensory techniques from central and peripheral nervous systems can be used to create robust signal conditioning systems. Amplitude domain neural approaches have focused on similar goals, however, they face the same problems as the systems described above. A detailed analysis will be provided in the next chapter. Analog sub-micron spiking neural networks have also been explored significantly in the last decade, however, the main goals of these approaches differ vastly

from our goals. They mainly focus on understanding the human brain. This is the goal of human brain project [71], which was the result of two previous projects BrainScaleS [72] and Facets [38]. Our research group, ISE, has worked on spike based LUCOS [2], which is a high dynamic range, high resolution pixel sensor. This work focuses on developing signal conditioning systems, which can replace or interface with the current systems.

The vision of this thesis work is to create a novel way of sensing, conditioning and conversion by taking cues from biological processes. To achieve this, this work will focus on researching concepts based on such processes, exploring various neural architectures, developing behavioral and physical implementations and assessing the viability of this approach and its efficacy against conventional competition and the problems predicted by ITRS. These results will then be optimized to deliver the best results as a physical demonstrator. This thesis work would fall under "More than Moore" design as a separate chip, however, the design will be beneficial to "More Moore" designs.

The goal is to create a novel ADC, a Multi-Sensor Signal Conditioning and Conversion system, which

- Will be able to interface with or be a part of a System on Chip with traditional analog or advanced digital components.
- Will be able to learn and adapt to static errors and dynamic errors.
- Will have a graceful degradation.
- Will be robust to noise and jitter related problems.
- Will be capable of self-repair, self-monitoring and self-calibration

1.3 Document Structure

The thesis is organized as follows

Chapter 2 will provide a comprehensive history of amplitude based neural ADCs. Then spiking neural models and networks will be introduced. This is followed by state of the art ADCs, which use spiking neurons and time coded ADCs.

Chapter 3 will provide an engineering perspective to biological sensory systems and introduce the proposed architecture

Chapter 4 will provide an introduction to biological neural networks, their electronic counterparts and also discuss the state of the art spiking neuromorphic systems, which make use of these electronic structures.

Chapter 5 will show the design and implementation of the building blocks of the architecture. The simulations will also show that these models are behaviorally similar to their biological counterparts.

Chapter 6 will introduce the spike to digital conversion concept, its design and implementation.

Chapter 7 will introduce the sensor to spike conversion concept, its design and implementation.

Chapter 8 will show the complete physical design, discuss its features, advantages and improvements that could be made.

Chapter 9 will discuss the testing and characterization methods used for the planned chip

Chapter 10 will provide a summary of the work, the novelties of the present work and the vision for the future work.

Chapter 2

State-of-the-Art Analog to Digital Conversion Techniques

This chapter will give a brief overview of conventional, reconfigurable, and Time-to-Digital Converter (TDC) architectures. Evolution of amplitude-based neural ADCs will be discussed and then the state-of-the-art spiking ADCs will be reviewed.

ADC	Analog/ Mixed Signal	Digital	Amplitude Domain	Time Domain	Neuromorphic
Flash	●		●		
Pipeline	●		●		
ΔΣ	●	●	●		
SAR	●		●		
Dual Slope	●	●		●	
Delay Line		●		●	
Sub Gate Delay		●		●	
Hopfield	●		●		●
Spiking Neural	●	●		●	●

TABLE 2.1: ADC Types and Differences

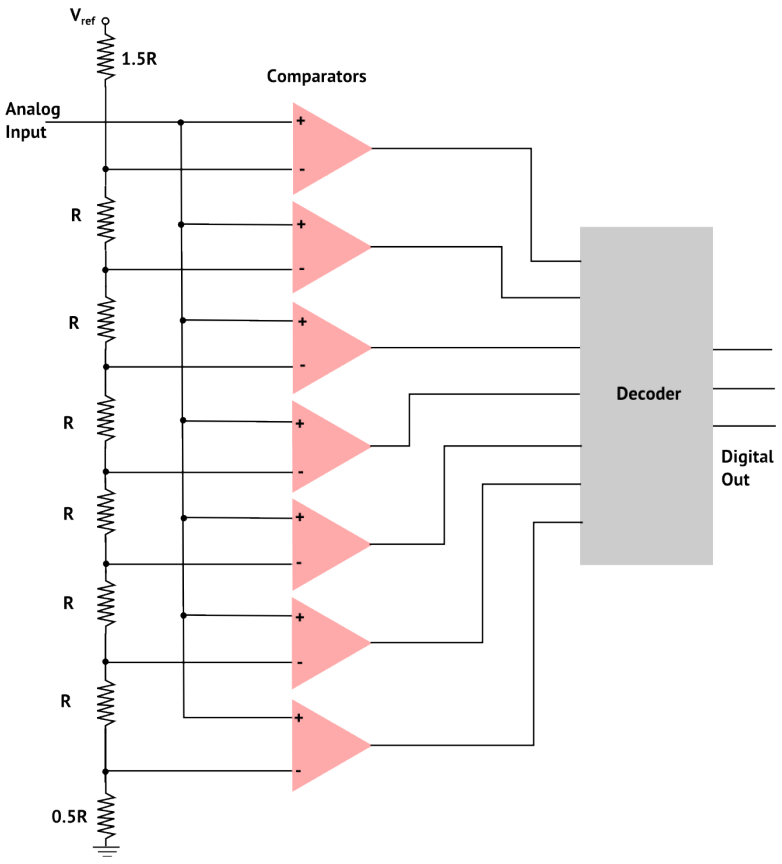


FIGURE 2.1: Architecture of a Flash ADC

2.1 Conventional ADCs

2.1.1 Flash ADC

Flash ADCs make use of resistive ladders in combination with comparators to convert analog signals to digital. While this technique (as shown in Figure 2.1) is extremely fast, it consumes a large area and is extremely power hungry. The input voltage passes through the set of resistors and is later compared to reference voltages using comparators. The number of comparators increases greatly with the number of bits required. An n bit Flash ADC will require $2^n - 1$ comparators. One of the main problems generally not mentioned with this ADC is the accuracy of reference voltages. The importance of accuracy for voltage references is described here [73]. Even if the initial accuracy is acceptable, the voltage references may degrade/drift due to aging or temperature effects. The noise in voltage references is also generally overlooked.

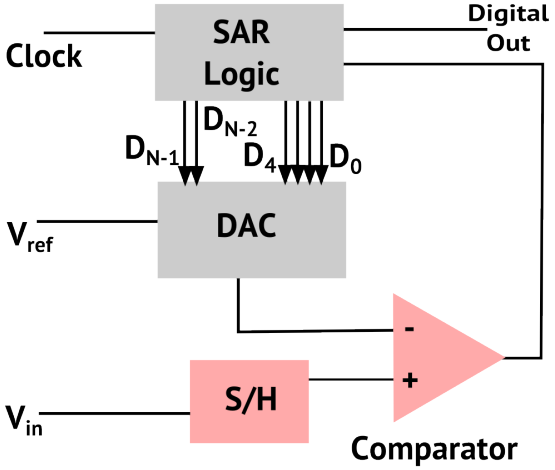


FIGURE 2.2: Architecture of a SAR ADC

2.1.2 Successive Approximation Register ADC

The Successive Approximation Register ADC works by performing a binary search on the input signal. In every step of the ADC, bits beginning from the MSB are identified one bit at a time. The MSB is first set to 1. An internal DAC generates the voltage for Vref/2, which is then compared with the input signal. the MSB is changed according to the output of the comparator and then moves to the next bit. This is performed n times where n is the number of bits required. Since it requires N periods to perform an n bit comparison, there is a relationship between maximum resolution one can obtain for a given sampling rate. This is the best choice for low to medium resolution applications. As technology scales, it is possible to use this architecture for higher resolutions.

2.1.3 Pipeline ADC

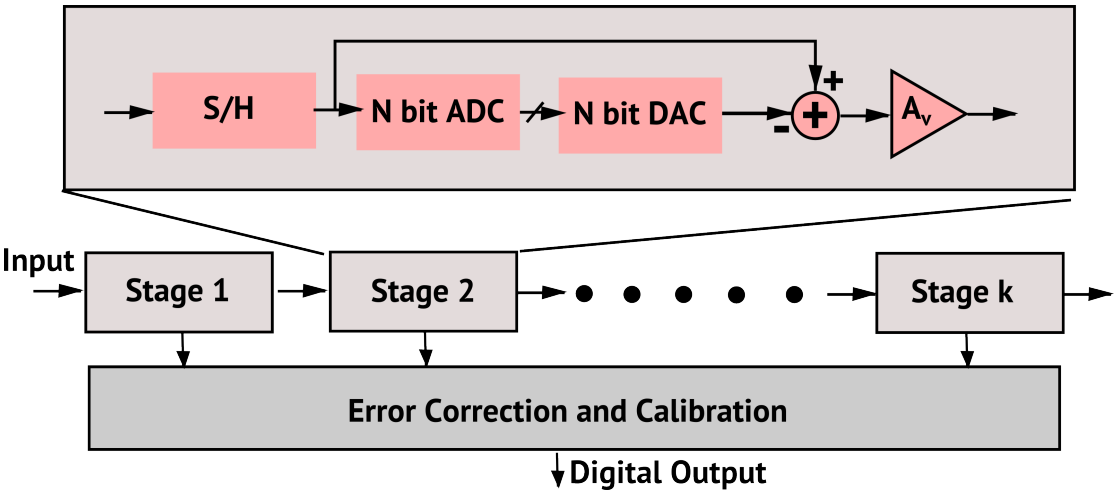


FIGURE 2.3: Architecture of a Pipeline ADC [11]

Pipeline ADC is primarily used in places where SAR cannot be used due to speed requirements. It uses more space than SAR, but nowhere as much as a Flash ADC for the same resolution. Its working can be explained based on the Figure 2.3. The incoming analog signal is converted by a coarse n bit ADC where n is usually 1 or 2 bits of resolution, this value is sent to the output. A DAC converts the output of this ADC and the analog output is subtracted from the original signal. The resultant signal is amplified and sent to the next stage of the pipeline where the process is repeated. The output bits are sent through a correction logic before going to the output.

2.1.4 Sigma Delta ADCs

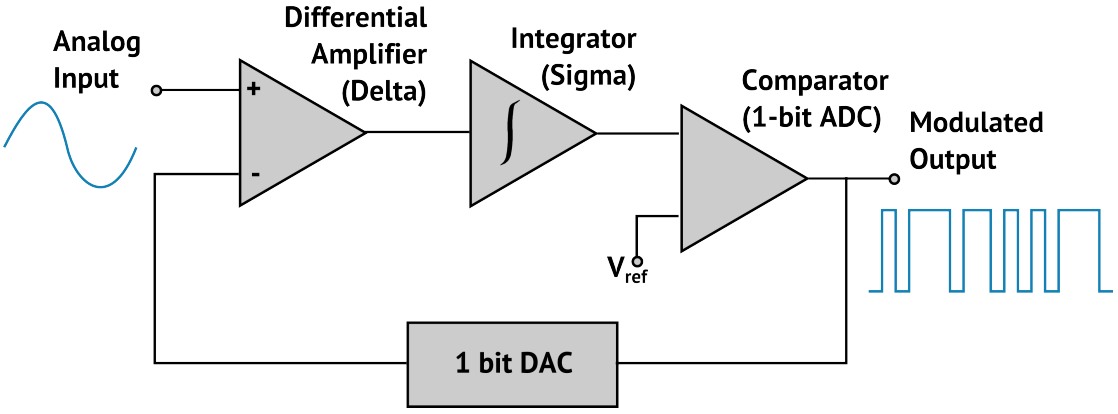


FIGURE 2.4: Block Diagram of Sigma Delta Modulator

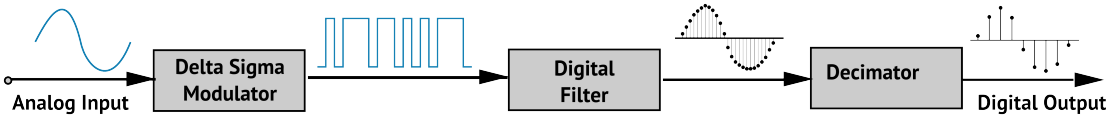


FIGURE 2.5: Architecture of an Sigma Delta ADC (adapted from [12])

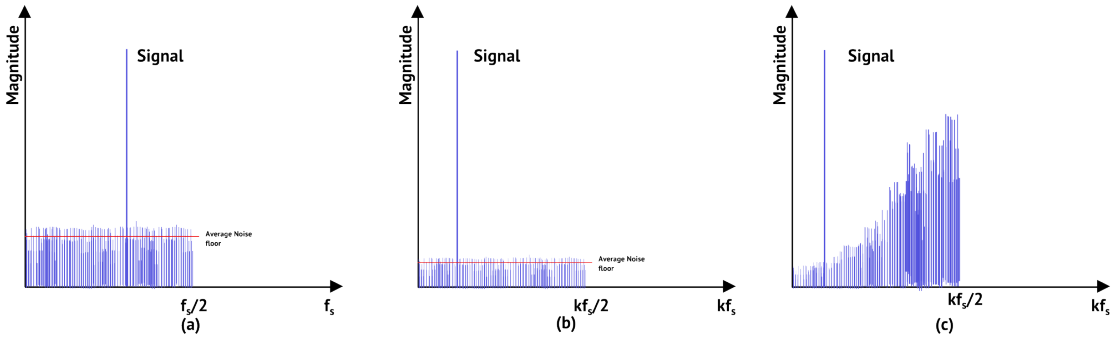


FIGURE 2.6: (a)Typical Output Spectrum of a Normal ADC with Quantization Noise, (b) Output Spectrum of an Oversampled ADC,(c) Typical Output after Noise Shaping

Delta-Sigma or Sigma-Delta ADCs have become the ADCs of choice for applications requiring high resolutions. One of the main advantages of this ADC is that most of the design is digital. The analog part of the design consists of the sigma delta modulator

shown in Figure 2.4. The working of this ADC can be explained by the Figure 2.6. It has been observed that oversampling, i.e., sampling at a rate much higher than the Nyquist rate, reduces the noise floor. This can be used to extract more bits of data from the ADC. When combined with noise shaping, as shown in Figure 2.6c and a digital decimation filter, more bits can be extracted from the ADC. There is an interesting similarity between sigma delta modulator and a biological neuron, which can be observed in the later sections.

2.1.5 28 nm ADC

Since we deal with technology scaling, here, we look at the various ADCs designed with 28 nm technology. The ADCs here are clearly moving towards SAR architecture, for e.g., Synopsys [51], which used a pipeline architecture in the previous technology step has completely moved to SAR architecture, while others are moving towards hybrid architectures with SAR, as seen in the table above. Another observation is the gradual removal of amplifiers from the ADC structures. Most of the ADCs in the Table 2.2 and Table 2.3 are moving towards such structures. Comparators are becoming the most important part of the "detection" in scaled ADC structures.

References	Dautriche et.al.	Craninckx et.al.	Kwan et. al.	Craninckx et.al.	Huang et. al.	Bult et. al.
Company	STMicroelectronics	IMEC	Broadcom Corp	imec	Broadcom Corp	Broadcom
Year	2014	2014	2014	2013	2013	2014
Bits	6	14	N.A.	11	12	N.A.
Sampling Rate	10 GS/s	200 MS/s	100 MS/s	410 MS/s	5.4 GS/s	1.9GS/s
Frequency	20GHz	N.A.	>50MHz	N.A.	2.6GHz	N.A.
ENOB	5.3	N.A.	11.5	N.A.	N.A.	N.A.
Energy	81fJ/conv. step	N.A.	N.A.	6.5fJ/conv. step	N.A.	9.1fJ/conv. step
SFDR	N.A.	N.A.	89dB	N.A.	N.A.	N.A.
SNR	N.A.	N.A.	N.A.	N.A.	61dB	N.A.
SNDR	N.A.	70.7dB	N.A.	59.8dB	N.A.	N.A.
THD	N.A.	N.A.	N.A.	N.A.	57dB	N.A.
Power	32mW	2.3mW	8mW	2.1mW	500mW	N.A.
Supply	1V	0.9V	N.A.	N.A.	N.A.	N.A.
Area	0.4 sq mm	0.009sq.mm	N.A.	0.1sq.mm	N.A.	N.A.
Technology	UTBB FDSOI	Digital CMOS	CMOS	Digital CMOS	CMOS	CMOS
Type	Time Interleaved SAR	2X Interleaved Pipelined SAR	SAR	2X Interleaved Pipelined SAR	2 Way Interleaved Pipeline	2X Interleaved Pipelined SAR

TABLE 2.2: A Comparison of Conventional 28 nm ADCs Part I (citations in order are [43] [44] [45] [46] [47] [48])

References	Brandolini et. al.	Bellasi et.al.	Azeredo-Leme et. al.	Schüffny et. al.	Ytterdal et. al.	
Company	Broadcom	ETH Zurich	Synopsis	TU Dresden	Norwegian University	
Year	2015	2014	2013	2014	2012	
Bits	10	4	12	12	9	
Sampling Rate	N.A.	5 GS/s	80MS/s	4.0/3.2MS/s	50MS/s	
Frequency	N.A.	3.8GHz	N.A.	N.A	N.A	
ENOB	N.A.	N.A.	N.A.	10.1/9.9	8.84	
Energy	N.A.	N.A.	N.A.	26/12.5fJ/conv.step	2.01fJ/conv.step	
SFDR	N.A.	N.A.	N.A.	N.A	N.A	
SNR	N.A.	N.A.	N.A.	N.A	N.A	
SNDR	N.A.	N.A.	N.A.	N.A	N.A	
THD	N.A.	N.A.	N.A.	N.A	N.A	
Power	N.A.	5.4mW	N.A.	115/38 µW	45 µW	
Supply	N.A.	N.A.	N.A.	1.0/0.7V	1.0V	
Area	N.A.	0.1sq.mm	N.A.	0.016sq.mm	N.A	
Technology	CMOS	CMOS	CMOS	CMOS	CMOS	
Type	Pipeline/SAR hybrid	Flash	SAR	SAR	SAR	

TABLE 2.3: A Comparison of Conventional 28 nm ADCs Part II (citations in order are [49] [50] [51][52] [53])

2.2 Reconfigurable ADCs

Reconfigurable ADCs have been popular in the last decade with since they are able to provide more flexibility to the signal conditioning process. These structures are capable of changing their resolution, sampling rate etc depending on the requirements. There have been reconfigurable versions of ADCs for all the structures that have been discussed in the previous chapter. A list of such ADCs has been shown in Table 2.4. There has not been a detailed investigation of using reconfiguration to improve robustness of an ADC. The two works, which deal with this are described below. a detailed discussion about self-calibrating ADCs can also be found in [74].

	Architecture	Resolution(Bits)	Speed (MSPS)	Power(mW)	Process
[75]	Flash	4	10˜1000	9.6˜10.6	0.18μm
[76]	SAR	9	0˜50	0˜0.7	90 nm
[77]	SAR	8/12	0-0.1/0-0.2	0˜0.025	0.18μm
[78]	Pipeline	12	20˜130	21˜110	0.18μm
[79]	Pipeline	10	0.11˜50	0.015˜35	0.18μm
[80]	Pipeline	14	10/20/30/40	19/34/51/73	0.18μm
[81]	Pipeline	10	25˜120	10˜36	90 nm
[82]	Time-interleave Pipeline	5/7	550/1100	13/30/46	90 nm
[83]	Time interleave Pipeline	10/11	11/44	14.8/20.2	0.25μm BiCMOS
[84]	Hybrid Pipeline/Cyclic	6˜10	2.5/5/10/80	30.2˜93.7	0.18μm
[85]	Hybrid Pipeline/Sigma-Delta	6˜16	0˜10	2˜24.6	0.6μm
[86]	Hybrid Pipeline/Sigma-Delta	10/11/12	0.2/4/20 *	15/37	0.18μm
[87]	Sigma-Delta	11˜15	0.1˜10 *	2.9˜20.5	0.13μm
[88]	Sigma-Delta	9˜14	0.2˜10 *	1.44˜7	90 nm
[89]	Sigma-Delta	12.5˜15	0.135˜1.92 *	2.6˜3.7	65 nm
[90]	Sigma-Delta	9˜12	20*	3.9˜20.1	90 nm
[91]	Sigma-Delta	11˜14.5	0.2/1/3. 84*	3.4˜6.8	90 nm

TABLE 2.4: Comparison of Reconfigurable ADCs by [54]

2.2.1 Fault Tolerant Reconfigurable ADC Circuit

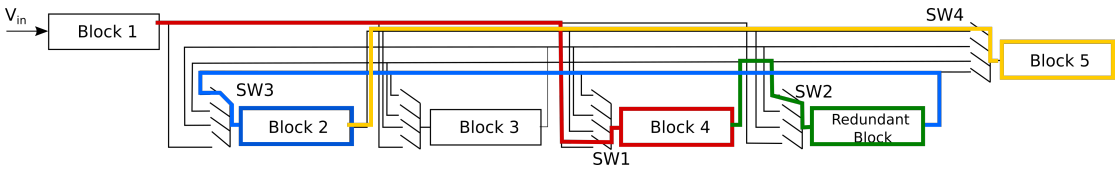


FIGURE 2.7: Reconfiguration for Fault Tolerance [13]

In the structure shown in Figure 2.7 form [13] the order of stages in the pipeline can be changed. This capability allows the users to put more accurate stages in front and the others at the back. The differences in accuracy occur due to process variations during manufacturing and degradation of the components over time.

2.2.2 Reconfigurable Time-to-Digital Converter with Online Background Calibration

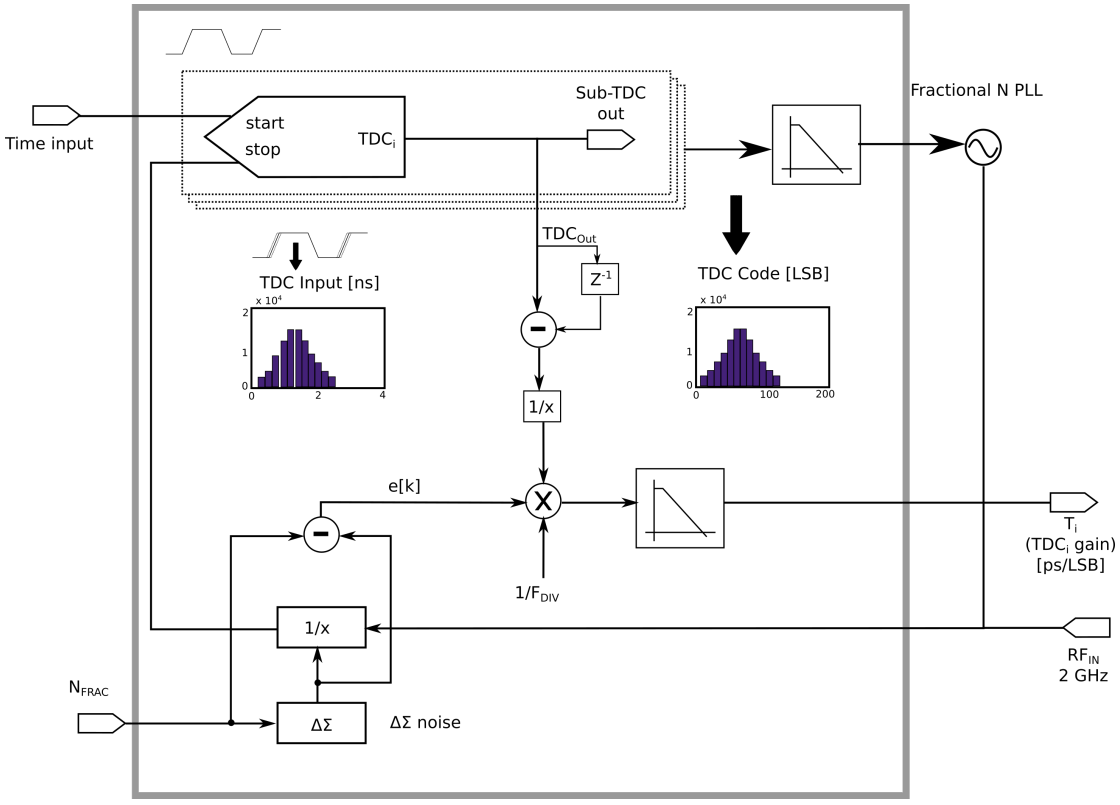


FIGURE 2.8: Online Calibration Scheme [14]

In this work, a background self-calibration has been set up using a Fractional N PLL setup shown in the Figure 2.8. In this structure, there exists a relationship between the frequency emitted by the PLL and the input time. Using this relationship the error can be calculated and the TDC can be calibrated effectively.

2.2.3 Summary

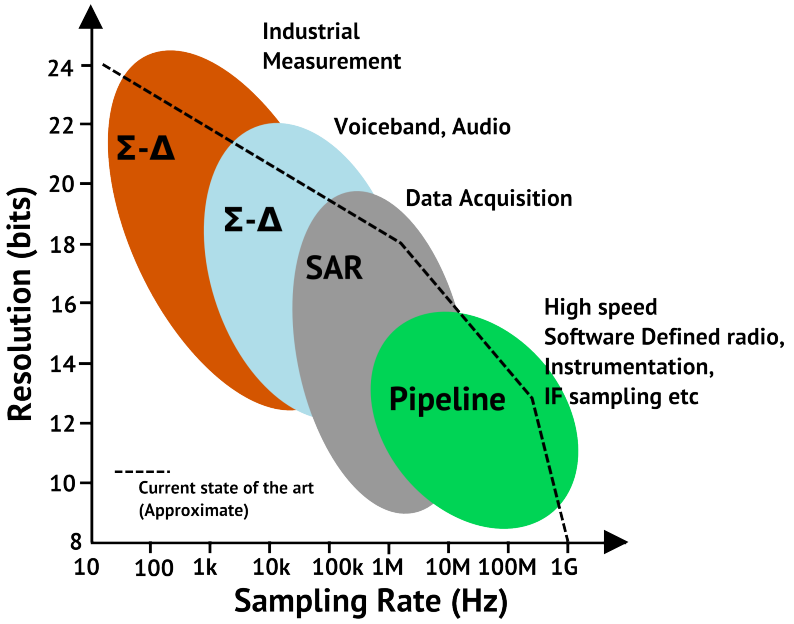


FIGURE 2.9: Comparison of Different Conventional ADC Structures [15]

From Figure 2.9 we can see the places, where the different ADC structures are being used. An important point to notice is the presence of sigma-delta ADC for high resolutions, where the sampling rates are sacrificed. This shows the role of noise in amplitude-based techniques. While the noise is combated by various schemes in reconfigurable methods, it is much better to completely drastically reduce the effect of such noise by using time domain based techniques. These techniques are described in the next section 2.3.

2.3 Time to Digital Converters (TDCs)

Since the Spiking Neural Networks work in time domain, It would also be essential to look at the progress in conventional TDCs.

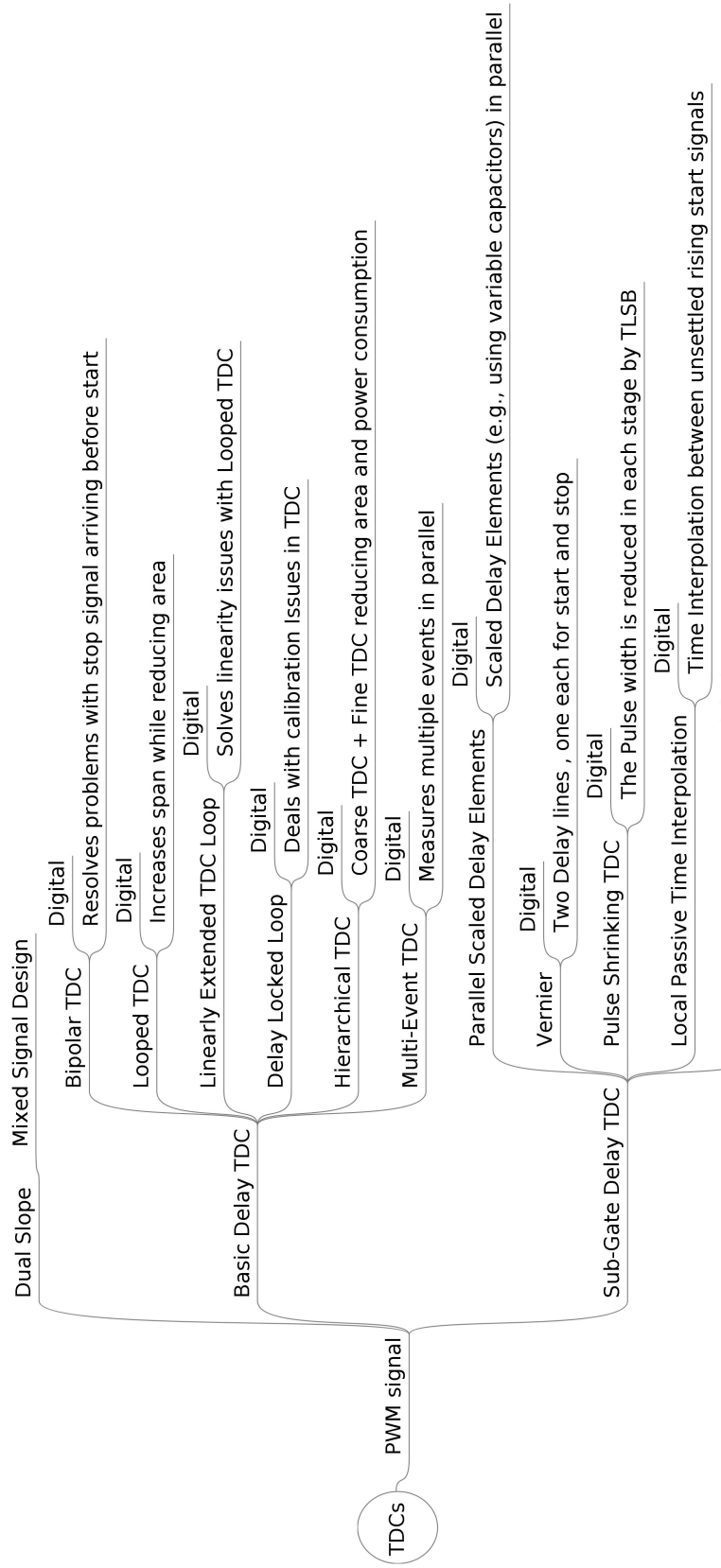


FIGURE 2.10: TDC Taxonomy based on [16]

2.3.1 First Generation TDCs

The first generation of TDCs make use of analog techniques to convert time interval to digital signal. The main examples are the single slope and dual slope converters. However, the main goals of the TDCs is to make use of scaled CMOS technologies and become generic mixed signal building blocks for various architectures. As long as TDCs use analog techniques, the above mentioned goals will not be reachable since they cannot make use of robustness provided by digital technologies. So, we move to the second generation TDCs [16].

2.3.2 Second Generation TDCs

The second generation TDCs are fully digital [16]. They do not have the disadvantages of the first generation TDCs. The main problem they face is the problem of speed, which is limited by the clock frequency. For faster TDCs they have to make use of specialized oscillators. However, if they create relative clocks by making use of delay elements, then the speed of the TDC is only limited by the delay element. An example of such a delay line TDC is shown in Figure 2.11. In this example, the start signal moves through the delay line producing delayed signals, which are then compared to the stop signal. This creates a thermometer code, which can be sent to a digital processor for further processing. There are various modifications to this basic structure, which reduces the delay (Inverter delay line TDC), Extended Loop TDC, where the delay structure is reused in a loop, Hierarchical ADC etc.

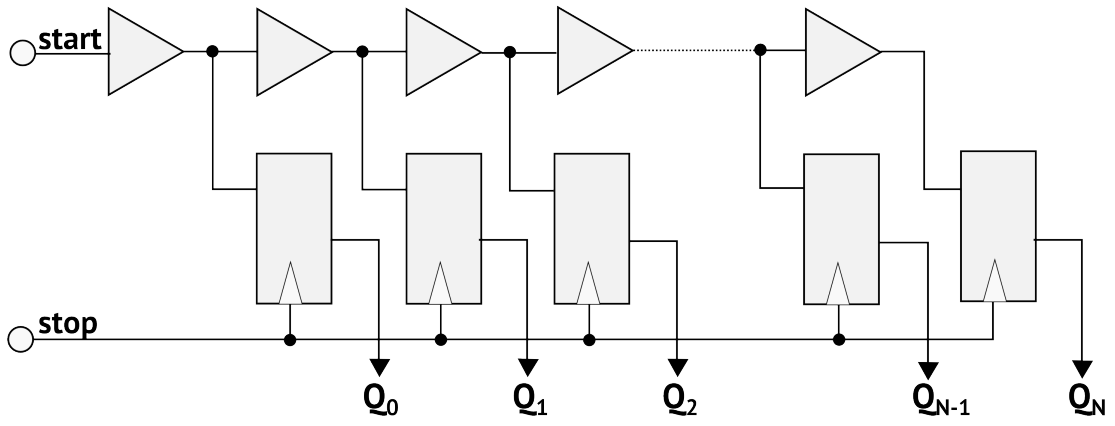


FIGURE 2.11: Implementation of Basic Delay-Line based TDC [16]

2.3.3 Third Generation TDCs

The third Generation of TDCs focus on being much faster than their predecessors, i.e., their limit is not the speed of the delay element. They make use of parallel scaled delay elements to obtain resolutions less than the delay of one gate (Inverter). This is called Sub Gate Delay Resolution [16]. One of the basic examples of this type is a Vernier Delay TDC shown in Figure 2.12. This works similar to a Vernier calipers. There are two delays td_1 and td_2. One of the delays is slightly less than the other. The start and stop signals move through these different delays providing an output that would be more accurate, than a simple inverter delay line TDC. There are many other techniques like pulse shrinking TDC, Local Passive Interpolation TDC, Gated Ring Oscillator TDC. The Gated Ring Oscillator TDC is used by acam messtechnik GmbH. [69].

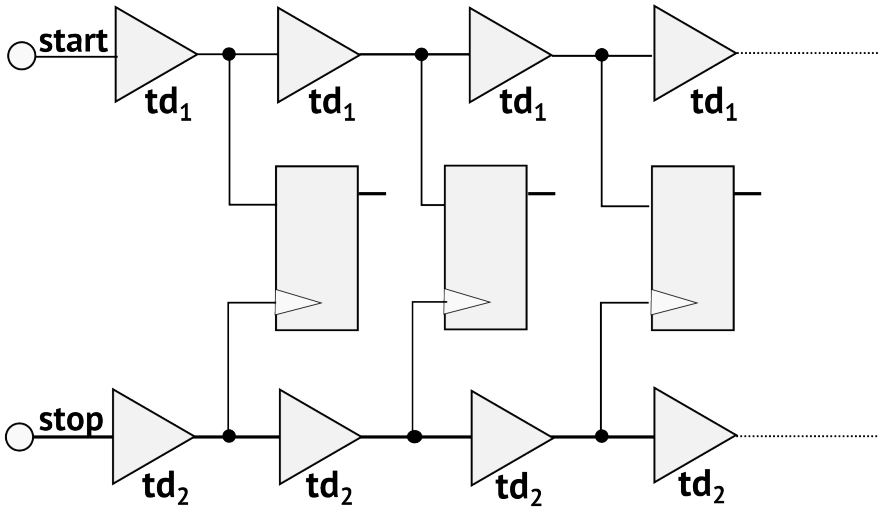


FIGURE 2.12: Implementation of Vernier Delay-Line based TDC [16]

2.3.4 Problems with TDCs

The following problems are the main focus of the TDC community

- The speed limit of the TDCs. The second generation TDCs were limited by the speed of the delay element. However, this problem is partially solved by the third generation TDCs.
- Typical TDCs will have problems, if the stop signal arrives before the start signal. Bipolar TDCs are used to solve this issue by using a reverse TDC along with the forward TDC.
- High dynamic range requires a much larger area, however, looped TDC solves this issue by looping over the start signal until the stop signal arrives.

- The basic linear TDC has good linearity for a small dynamic range. For larger dynamic range with good linearity linearly extended TDCs are used.

To summarize, there are many TDCs, however, according to [16] each TDC is optimized to its requirement. Further it is stated in [16] that no single TDC can solve all these issues.

2.3.5 Self-Calibrated Stochastic TDC Architecture [7]

While there are a few TDC architectures, which focus on self-calibration, this is one of the more interesting examples. It consists of an encoder, which ensures the monotonicity of the TDC. There is a self-calibration module, when activated, calculates the DNL and INL errors and stores them in memory. In normal operation mode the inverse function of INL is used to correct and produce linear output characteristics. The presence of this self-calibration module creates an interesting advantage. Since the TDC can be self-calibrated using this technique, the delay elements need not be an inverter or buffer. They can be replaced by process variations in flip flop circuits. This delay due to process variations are much finer than those of any delay elements. This helps in obtaining sub picosecond resolution for the structure.

2.4 Amplitude Based Neural ADCs

2.4.1 Hopfield ADC and Modifications

Neural architectures are fundamentally different from the traditional digital architectures. Most problems in real life can be described as the search for best solutions. The most prominent problems among these are pattern recognition or classification problems and the ADC problem is one of them. In the 1980s neural networks became popular with the introduction of backpropogation and Hopfield networks. Hopfield found that a symmetric feedback from the output to the inputs of a neural network created an energy function, which generally moved towards the minimum value [92]. These are highly parallel and can be optimized to solve many problems.

Hopfield in his work [17] discusses about using analog VLSI neural networks to solve such problems. He begins this with the problem of AD conversion. He makes use of analog amplifiers as neurons. These structures have feedback and high inter connectivity. The resistors act as weights.

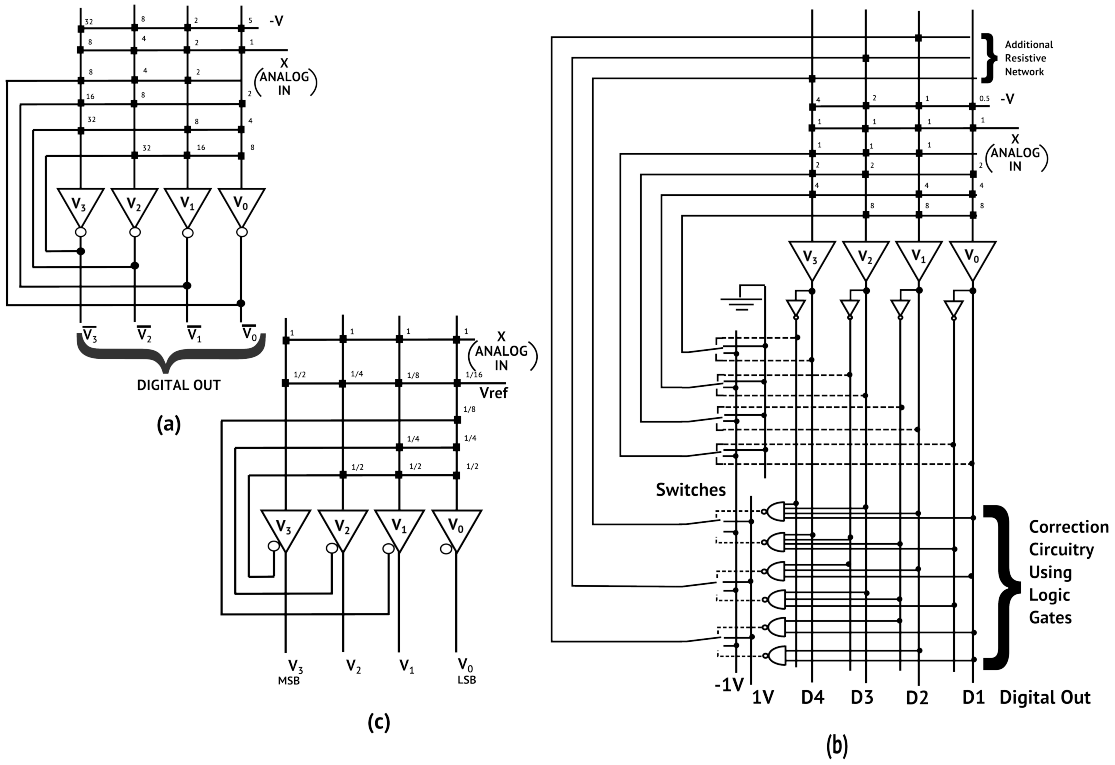


FIGURE 2.13: (a) Hopfield 4-bit ADC neural network [17]. X is the analog input, $V_3V_2V_1V_0$ is the digital output with each value being 0 or 1. (b) Hopfield ADC modified for removing local minima by Sheu et. al. [18] (c) Lower Triangular Matrix to remove local optima by Manetti et. al.,[19][20]

The Hopfield 4-bit ADC is shown in the Figure 2.13. It consists of four inverting amplifiers (One per bit). The amplifiers are assumed to have insignificant time constants, however, the inputs are connected to a resistor leading to a reference ground and a capacitor. The integrative analog summation of input currents coming from other parts of the network. The network has attractor states (energy surface), which will reach the required states after an ideal start. The authors have found that the convergence is pretty quick, however, it required perfect starting conditions or it got stuck at local minima. Regardless of these disadvantages, this neural ADC was completely parallel so all bits of the digital word are obtained at the same time. The ADC is also capable of adaptivity, since the conductances (resistances) of the device can be changed. This can be used for compensation of device mismatches and compensation of long term drifts [18]. This structure was the first popular neural ADC, which was later improved by various other researchers.

Sheu et. al., [18] added a correction logic to the Hopfield ADC so that the local minima are removed from the ADC. This correction logic used simple digital logic gates fed back to the inputs. The circuit can be seen in Figure 2.13b. The improvement cause by the addition of these gates can be seen in Figure 2.14.

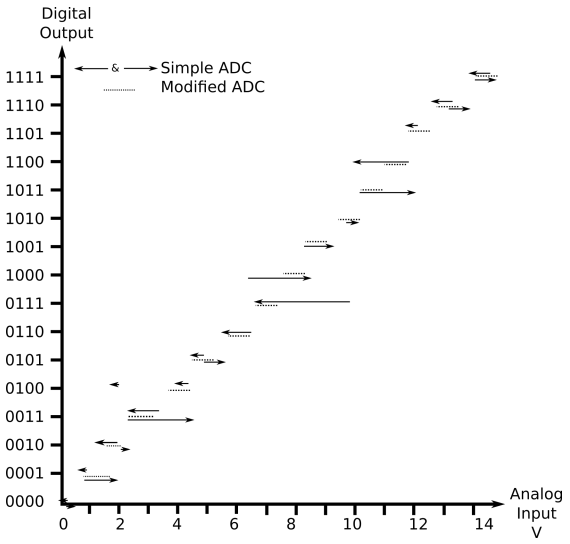


FIGURE 2.14: Transfer Characteristics of Hopfield ADC vs. Sheu ADC [18]

Manetti et. al., in [19][20] found that a lower triangular matrix (Non symmetrical network) also is capable of removing the local optima. This also removes the problem of bad initial conditions. They found that the resistances in the network can be replaced by R/2R ladder. This provides a lot of advantages for the resistance spread of the entire network. Changing the V_{ref} as shown in Figure 2.13c changes the code obtained to BCD. Similarly it is also possible to obtain Gray codes. This method is very useful to create n-bit ADC. The information can be made to cycle through the ADC and a shift register to store the values. This will create a cyclical cascade to extract any number of bits. This method would, however, loose the advantage of parallelism provided by neural networks.

Newcomb et. al., modified Sheu et. al's [18] work in [93] by using multi-level neurons i.e. neurons which could have more than two states. With this they were able to use just 3 neurons and 6 weights for a 6 bit ADC, while former required 6 neurons and 30 weights to accomplish the same. They also make use of transistors as weights since they consume less area than resistors. This method, however, produces higher nonlinearities than other techniques described above. Tanaka et. al., [94] modified Hopfield neural ADC by making use of inverters for both neurons and weights and implemented the complete circuit in conventional digital CMOS process.

2.4.1.1 Summary

While Hopfield's ADC was the first ADC using neural approach to become popular, its recurrent structure created a lot of problems because of local minima. The ADC requires proper initial conditions to reach the right attractor states. While these problems have

been circumvented in the modified circuits, they are more complex, have low resolution, are non linear, and in some cases loose the parallelism, which has been the strength of the structure. The strength of these structures like the capability for adaptivity and compensation of manufacturing and drift related issues makes the approach relevant for our work.

2.4.2 Synthesis of Neural ADCs

Perfetti et. al., studied about different ways of synthesizing neural ADCs in [8]. They use three established techniques to synthesis neural ADCs. These are

2.4.2.1 "Synthesis by Superposition of Sigmoidal Functions" [8]

The ADC problem can be considered as a classification problem, hence a multi-layer perceptron can be trained as an ADC. Due to the problems with traditional perceptron training using backpropogation techniques, here the authors use the superposition of sigmoidal functions to model the ADC. Each function will model one state of the ADC, which is trained using an optimization algorithm. In their first attempt they synthesize a 3 bit ADC using this approach. The result can be seen in Figure 2.15. This structure appears to be very similar to Flash ADC. It also requires the same number of comparators, however, the decoding circuit can be realized easily in this structure as compared to a Flash ADC.

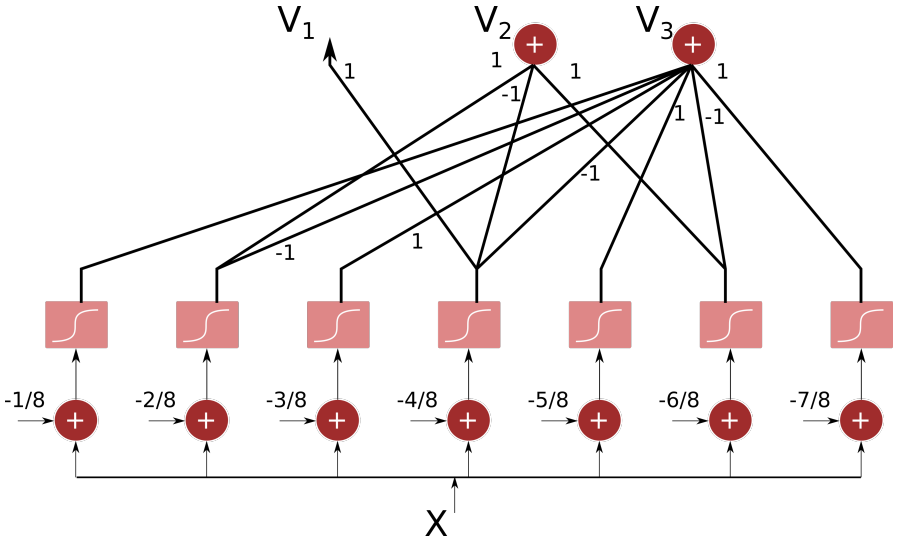


FIGURE 2.15: Binary Coded ADC Generated by Synthesis (superposition of sigmoidal functions) [8]

2.4.2.2 "Synthesis by N-layer Perceptrons"

In this approach, the weighted nature of the code is exploited to synthesize the ADC. The digits of the output can be determines in descending order from MSB to LSB. Several techniques have been used to synthesize using this type of a weighted code structure. The authors show that these techniques synthesize structures similar to different conventional ADCs like equilibrium encoder, algorithm ADC and Successive Approximation ADC shown in Figure 2.16

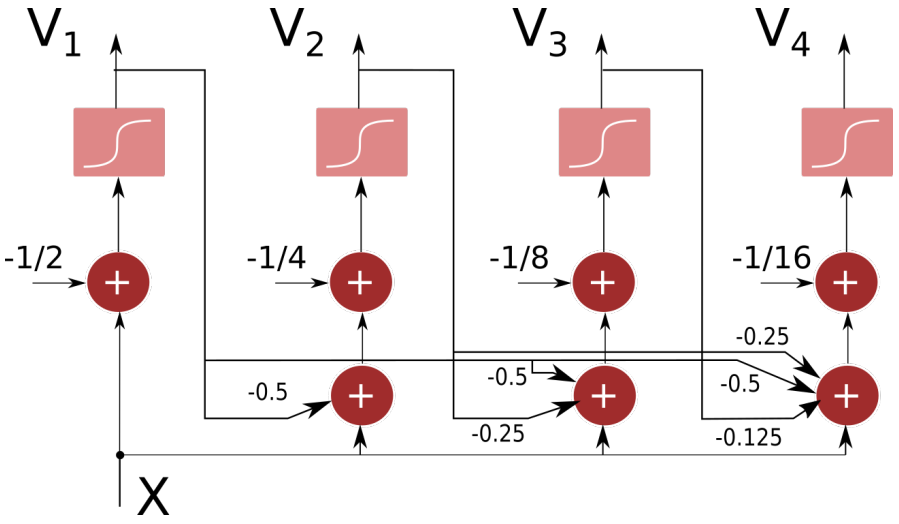


FIGURE 2.16: Binary Coded ADC Generated by Synthesis (N layer Perceptron)[8]

2.4.2.3 "Synthesis using Neural Nets with Lower Triangular Interconnecting Structure" [8]

This technique works on Hopfield networks. It makes use of the network connections and the transition states for each connection matrix with different threshold. The weights and thresholds are modified until the required transition states are reached. One of the obtained outputs, while synthesizing a 4 bit ADC is shown in the Figure 2.17

2.4.2.4 Summary

Three types of synthesis are described in the work. This comparison is shown in Table 2.5. This table describes the different structures that they were able to synthesize. These structures appeared to be roughly equivalent to the analog structures mentioned in Table 2.5. Other than these structures, some more novel structures have been identified by the authors. This analysis of synthesis techniques shows that neural networks can be used to create new novel high-level architectures.

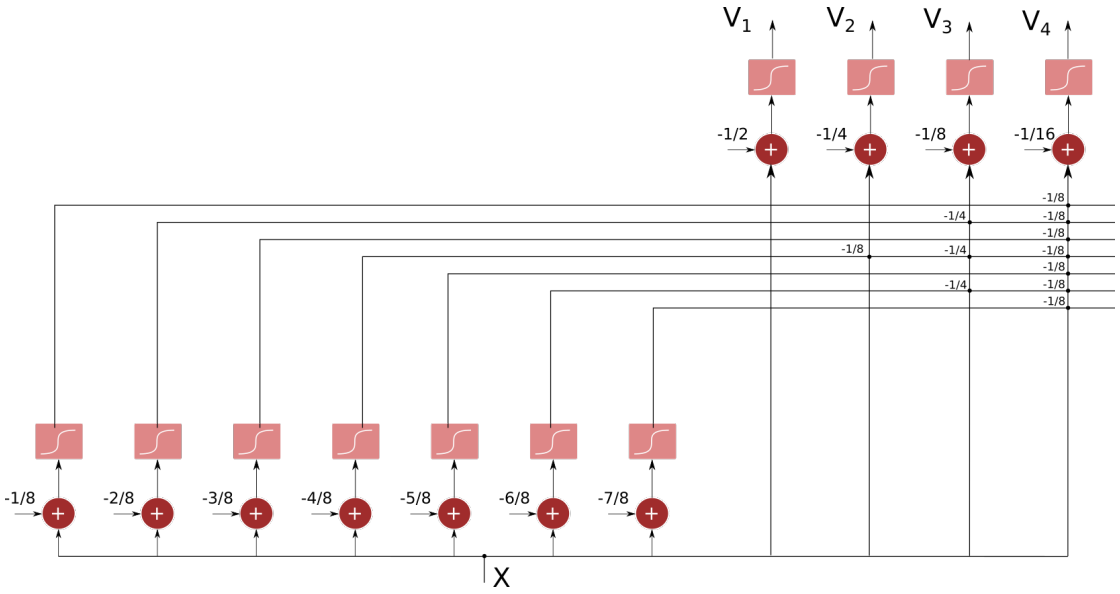


FIGURE 2.17: Binary Coded ADC generated by synthesis (lower triangular interconnect) [8]

Scheme	No. comp.	No. of conn.	Conv. tim.	Analog eqv.
Sigmoid Superposition	$2^{(N-1)}$	$2^{(N+1)}$-N-2	tau	Flash
N-layer perceptron v1	N	2N-3	Ntau	SAR
N-layer perceptron v2	N	N(N-1)/2	<tau >Ntau	SAR
Lower triangular matrix	$2^{(N-1)}$+ N -1	2^{N} -N -1	2 tau	2 step flash

TABLE 2.5: Comparison between Neural and Traditional Analog A/D Structures by Perfetti et. al., [8]

2.4.3 A Neural ADC similar to Pipeline ADC Architecture

Grimaldi et. al., have worked on various works related to the neural AD converters. In their first work [95], as shown in Figure 2.18 they designed a neural AD converter, which makes use of a comparator and subtracting amplifier for each stage. Each stage measures the difference between a reference signal and the input value and propagates the difference. They have modified this system in [21] and [96] to create cascadable analog blocks (Neurons), which increase the resolution.

2.4.4 Other Approaches

In [22] Grimaldi et. al. propose methods for ADC neural modeling as shown in the Figure 2.19. Here a precise ADC is used as a reference from, which the network would learn. The network obtained could have the same precision, but it might be able to operate at much higher speeds.

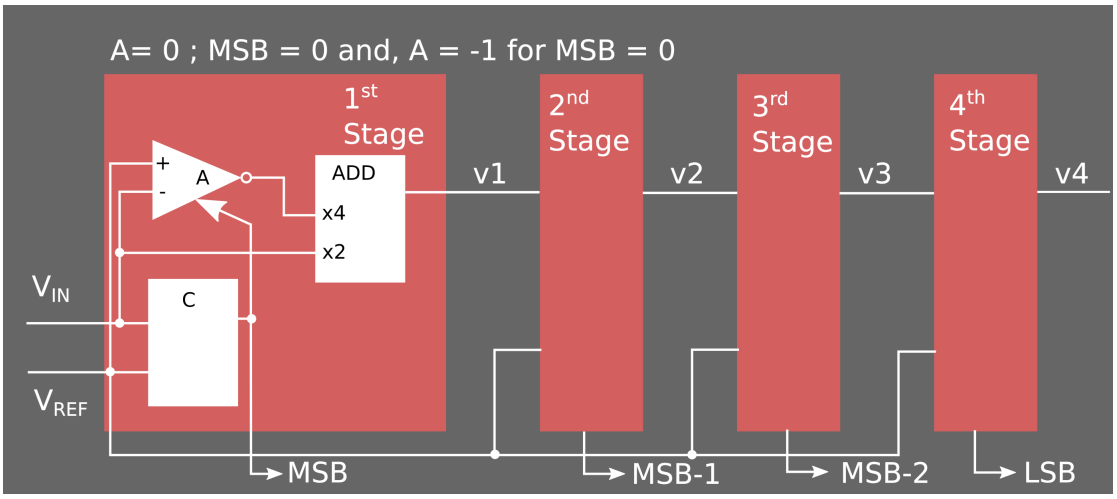


FIGURE 2.18: Block Diagram of a Neural Pipeline ADC [21]

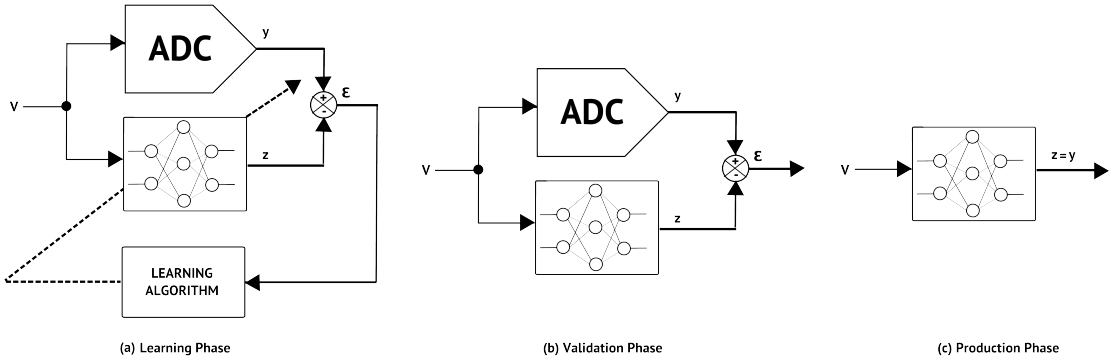


FIGURE 2.19: ADC Neural Modeling by Grimaldi et. al. [22].

Chanal et. al. [23] proposed a method to correct errors in an ADC using neural networks shown in Figure 2.20 based on the ADC neural modeling approach in Figure 2.19 Pearce et. al., describe an ADC [97] designed by making use of noise. It makes use of a technique called Suprathreshold Stochastic Resonance (SSR). where a population of comparators with different noise levels are used to obtain an ADC. The area of such a circuit is much larger than a regular flash ADC, however, the design is simple according to the authors.

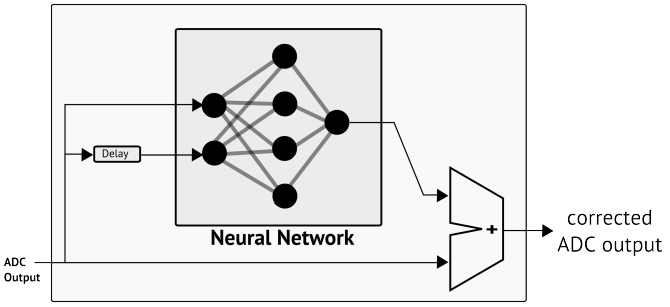


FIGURE 2.20: ADC Correction by Neural Networks proposed by Chanal et. al. [23]

2.4.5 Summary

From the amplitude based neural ADCs we can clearly see the advantages of neural approaches especially, when it comes to design of novel architectures, which are adaptive in nature. These qualities, when used with time domain neural networks, can be used to create robust ADCs.

Work	Neuron Type	Neuron Circuit Type	Synapse Ckt Type	Num. of bits	Notes	INL	DNL	ENOB
Hopfield et. al.	Sigmoid	Analog Amplifier	Resistors	4	Perfect Initial Conditions needed, Local minima		5LSB	
Sheu et. al.	Sigmoid	Analog Amplifier	Resistors	4	Uses Correction logic to reduce above problems		0.5LSB	
Manetti et. al.	Sigmoid	Analog Amplifier	R2R ladder	4	1-bit ADC in pipeline	-	-	
Marini et. al.	Sigmoid	Analog Amplifier	Resistors	4	Non Symmetrical Connection Metric to get unique Equilibrium points			
Newcob et. al.	Multilevel Sigmoid	Custom Neuron	Bi-directional Current Mirrors	6	Required only 3 neurons and 6 weights for 6 bits , However, higher non-linearity			
Tanaka et. al.	Sigmoid	CMOS Inverters	CMOS Inverters	4	Completely implemented in Digital CMOS.			
Perfetti et. al.	Perceptron	-	-	-	Synthesis of Neural ADC using different Techniques			
Grimaldi et. al., 1	Sigmoid	Analog Comparator	Amplifier	8	Pipeline Converters	0.8LSB	0.3LSB	
Grimaldi et. al., 2,3	Sigmoid	Comparator	Amplifier, Adding Block	8	Uses Gray Code output	1LSB	0.1LSB	
Grimaldi et. al., 4	Sigmoid	-	-	-	ADC Neural Modeling methods	-	-	
Chanal et al.,	-	-	-	-	Error Compensation	-	-	
Pearce et. al.,	-	-	-	-	ADC from Noise based on Suprathreshold Stochastic Resonance	-	-	

TABLE 2.6: Comparison of Amplitude-Based Neural ADC papers

2.5 Spiking Neural Networks

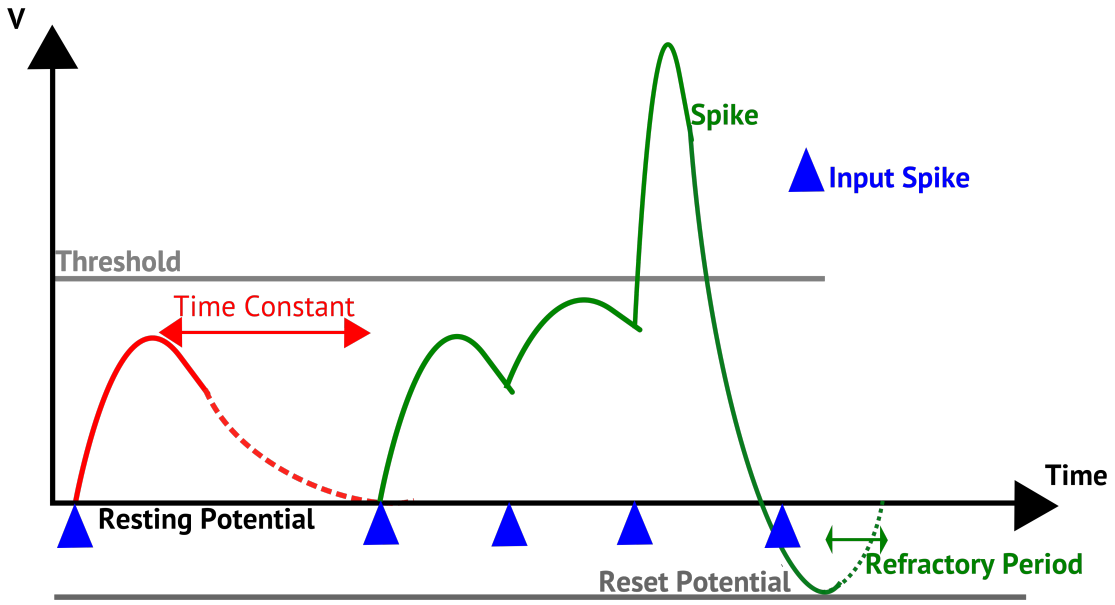


FIGURE 2.21: Properties of a Spiking Neuron

Spiking Neural Networks are closer to biological neurons than perceptrons and other artificial neurons. They have more properties to control and the outputs are spikes. There are various coding techniques that can be used with spikes, however, the most prominent natural techniques are "Time to first spike" and Rate Coding.

The working of a spiking neuron can be seen in the Figure 2.21. The neuron has a threshold similar to artificial neurons. The time constant of the neuron can be controlled. If the time constant is low, then the input spikes have to be closer to each other to create an output spike. Normally the membrane potential (V) of the neuron returns to the "Resting Potential", if it does not spike. If there is a spike, then the membrane potential goes to a much lower value called "Reset Potential". Until it recovers and reaches the resting potential it will not respond to any inputs. The time taken to reach the resting potential from reset is called the refractory period of the neuron. Increasing this will prevent immediate bursts of spikes in the output.

The spiking neurons have been studied a lot with analog implementations in the past decade. One of the main advantages is, that the code is in time domain, which is much more robust than amplitude domain. There have been a few implementations of ADCs making use of these neurons.

2.5.1 Spiking Neural ADCs

2.5.1.1 A Bio-Inspired Low Power ADC

The earliest spiking neural ADC approach is the Ultra-Energy efficient Spiking ADC by Sarpeshkar et. al. [24].This work is similar to a conventional sequential pipeline converter. The incoming voltage charges a capacitor and the discharge time is calculated. This calculation provides the MSB of the ADC. The residual output is then converted from time to voltage and amplified by a factor of 2 and fed back to the inputs. This is able to create a design, where the resolution is linearly dependent on time. They also make use of synchronizing clocks to provide more accuracy and compensate errors. An 8 bit design with layout is presented, which consumes 950 nW of power with 1.2 V voltage source. It runs at a speed (sampling rate) of 45 kHz with an area of 0.021 mm^2 with an ENOB of 7.4 (SNR 47dB). The focus of this work is on bio-medical applications, which require very low power.

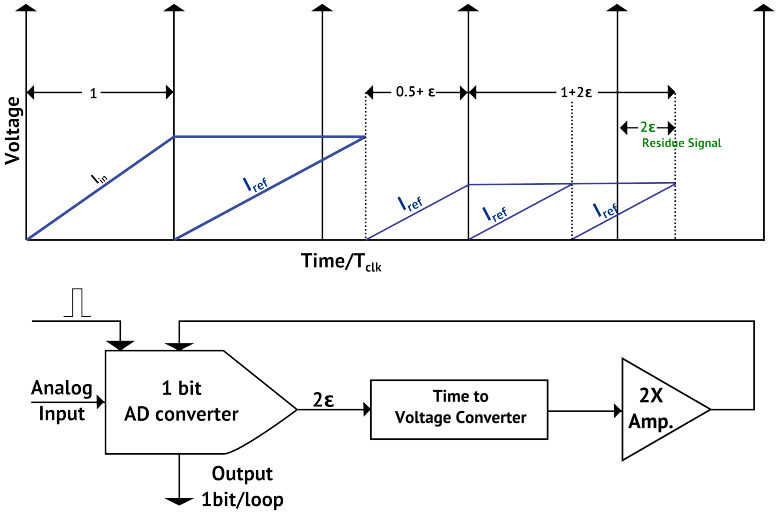


FIGURE 2.22: Working of Spiking ADC by Sarpeshkar et. al.[24]

2.5.1.2 Time Interval Mapping based Spiking Neural ADC

The next work is by Torikai et. al. [25], where they create an ADC by making use of spike interval times. Each analog input has a unique spike pattern in this approach. It requires an operating time of (l+1) to obtain l bits of digital information. There are two methods of obtaining the pattern, by mapping of spike positions and by mapping of spike intervals. In this work, the authors experimentally show that Spike Interval Maps are more useful than Spike Position Maps. A simple circuit model of their work is shown in Figure 2.22. Here U(t) and $B_1(t)$ are two voltage sources synchronized with

a clock signal. A complete circuit has been mentioned as part of future work, however, no related work has been found.

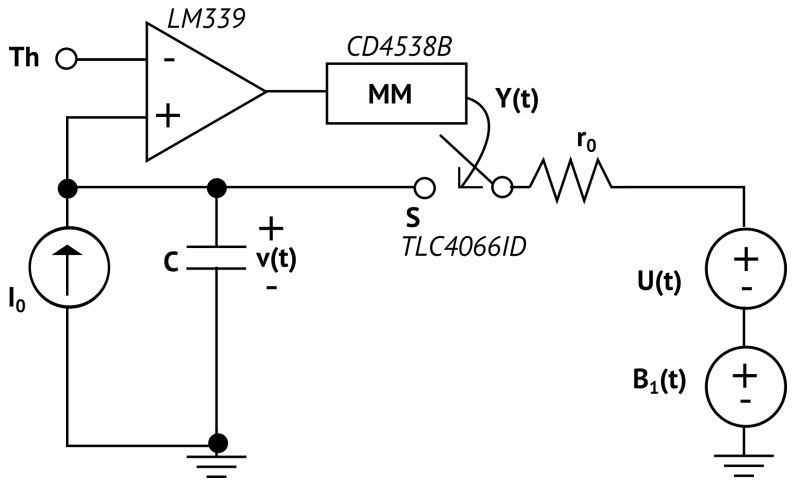


FIGURE 2.23: Spiking ADC [25], MM is the Monostable Multivibrator

2.5.1.3 A Spiking Neural Network ADC Concept in C++

Cios et. al., introduce a population-based spiking ADC system [26] to decipher images. Each neuron consists of a circuit, which makes it sensitive to one particular intensity, by making use of several sub-circuits of such neurons for different intensities an ADC can be created. This is mainly done using a combination of inhibitory and excitatory neurons as shown in the Figure 2.24. This works based on a population of neurons where several neurons detecting the same intensity are present. This redundancy helps in increasing the accuracy. This ADC has been simulated in C++ with VSSN (Very Simple Spiking Neurons). No chip implementation was found. The problem with this approach is that precision and drift of the intensity detection neuron. While this can be reduced by using a population of neurons (redundancy), it will greatly increase the circuit area. The circuit can be trained only after manufacturing.

2.5.1.4 A Parallel Spiking ADC Architecture

In this work [27], a proof of concept Spiking ADC is discussed. It makes use of "on" and "off" spikes as shown in the Figure 2.25. Each of these pathways makes use of lateral inhibition to inhibit the other neurons from spiking. The spike frequency, and phase codes are used in this work. The spike rate increases linearly with intensity of the input signal. An implementation is created with off-the-shelf components to prove that it can work as an ADC.

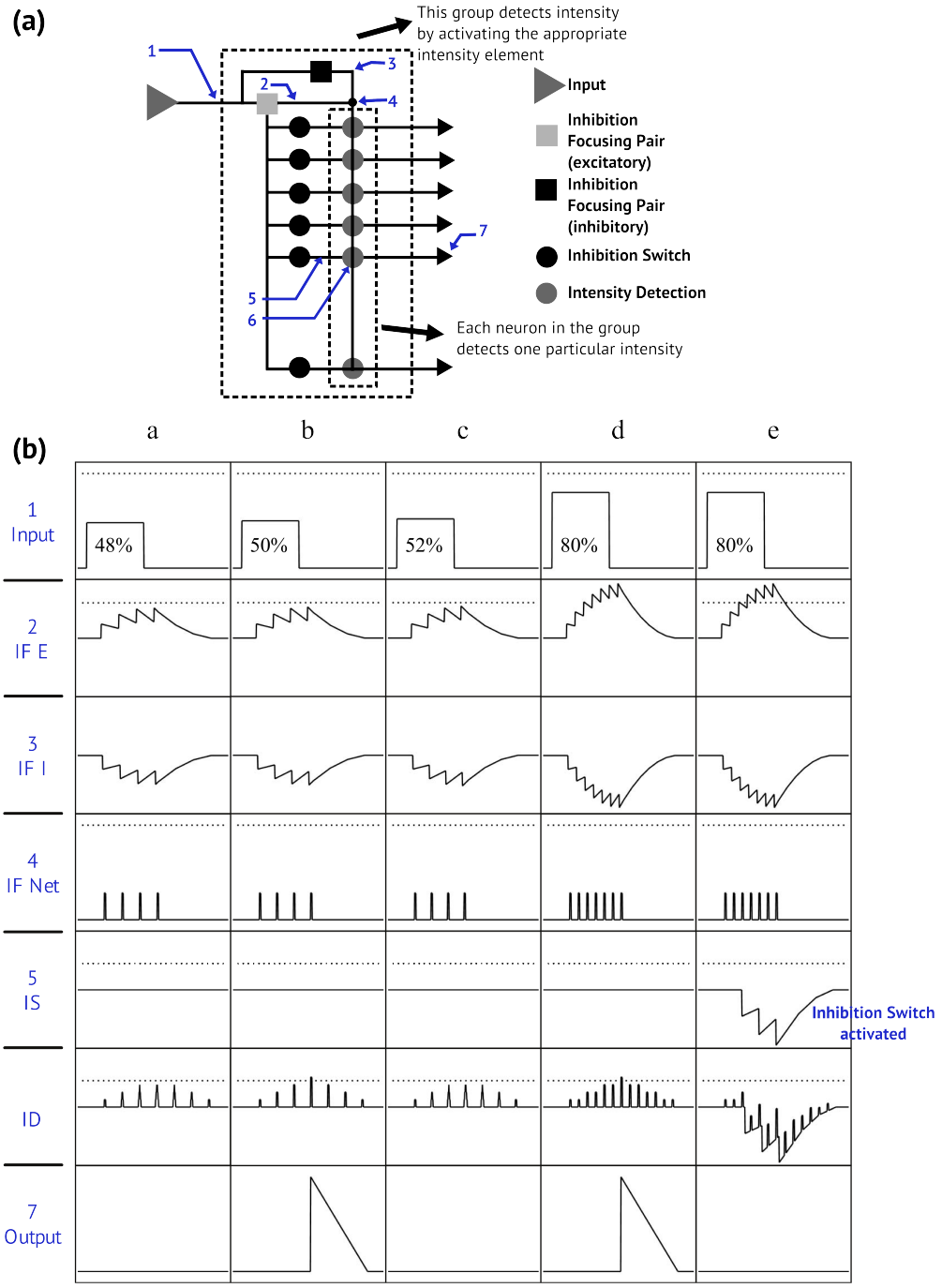


FIGURE 2.24: (a) Spiking ADC concept by Cios et. al., [26] (b) Rows show recorded potentials at the locations in (a). The columns show different inputs. Inhibition switch neuron is disabled in column d and enabled in column e.

2.5.1.5 Reconfigurable ADC using NEF

Mayr et. al. make use of the Neural Engineering Framework (NEF) as the base for creating their ADC. NEF is a generic method for implementation of different algorithms using biological spiking neurons [98]. They make use of the variations in frequency with respect to the input signal. The variations in the property of the neurons create

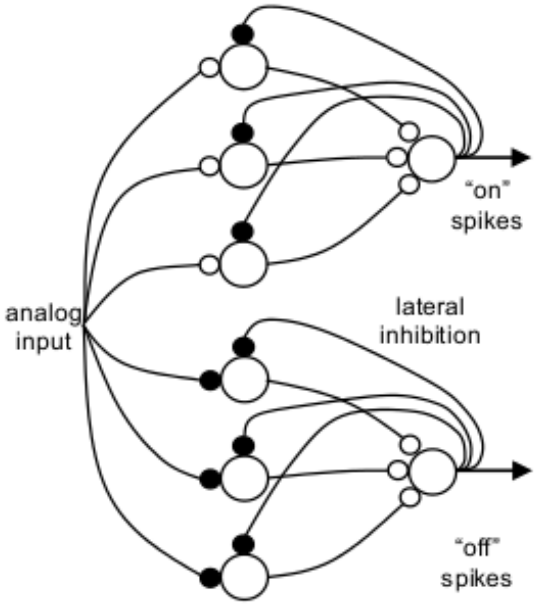


FIGURE 2.25: Spiking ADC [27]

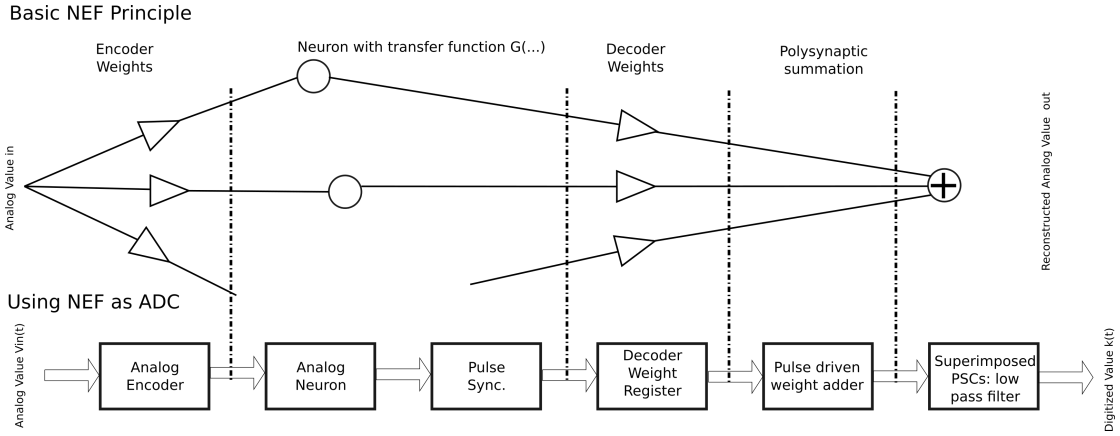


FIGURE 2.26: Spiking ADC based on NEF Neuron [28]

changes to the frequency output curve. In this work, these changes are called "Encoder weights". A related set of linear decoder weights should be found to return the required output value. An exponential kernel is applied to the weighted spike summation for each time step. In the implementation the parameter variation is accomplished through the manufacturing process, however, this creates the need for individual decoder weights for each ADC that is manufactured.

Reference	Sarpshekar et. al.	Torikai et. al.	Cios et. al.	Schaik et. al.	Mayr et. al.
Domain	Amplitude & Time	Time	Time	Time	Time
Coding Scheme	Time Interval	Time Interval Mapping	Population Code	Spike Rate and Phase	Encoder and Decoder weights
Parallel/ serial	Serial	Serial	Parallel	Parallel	Parallel
Area & Power	Extremely Low Power and Area	No Data	No Data	No Data	High Power & Large Area
Reconfigurable/ adaptable	No	No	No	No	Rate & Resolution
Capable of Sensor Fusion	No	No data	Yes	No data	Yes
Resolution	8 bit	No data	No data	No data	7.67 bit
Implementation	0.18 µm subthreshold CMOS	Off the Shelf Components	C++ & XML	Off the Shelf Components	0.18 µm CMOS

TABLE 2.7: Comparison of Neuromorphic Spiking ADCs

2.6 Discussion

ADCs have applications in various essential fields from sensor signal conditioning to software defined radios. The ADC problem has been tackled by various methods on different fronts. The present conventional ADCs still work in the amplitude domain, however, this path is moving towards a dead end. The main reason for this is the effect of technology scaling on the signal-to-noise ratio of the circuits. This also affects the dynamic range of the ADCs. The effect of technology scaling can be clearly observed in 28 nm ADC structures 2.1.5 where the amplifiers are noticeable by their relative absence, and the clear movement towards SAR ADCs. These ADCs also have reconfigurable structures present to combat the problems with manufacturing. One of the major problems faced by these approaches is that the properties of the ADCs will become worse with technology scaling.

TDCs unlike ADCs have taken advantage of technology scaling with circuits becoming better with the scaling and not vice versa. The second and third generation TDCs have also moved towards digital technology for their designs, which are more robust. The dynamic range of time domain signal processing will not be affected by technology scaling. With these advantages TDCs are set to capture the market of conventional ADCs. However, these TDCs work with pulse width modulated signals as inputs and do not explore other possibilities. The TDCs also face problems due to manufacturing

as they depend on accurate delays. These problems reduce the robustness of these architectures.

The sensory system in animals is extremely important for the survival of the species. The natural process of evolution has tuned these sensory systems of both predator and prey to survive. This provides a huge knowledge base of ideas and inspirations for us to exploit. Although there have been numerous neural architectures described in the previous sections, they focus on inspirations from central nervous system. These structures working both in amplitude and spike domain have been discussed in the previous sections. Peripheral nervous systems have several features, which will become more and more important in the near future like, multi-sensor fusion, local and global adaptation, low power consumption, robust signal processing etc. The structure of these sensory systems must be studied in detail and can be used to design ADCs or signal conditioning systems, which can compete with living organisms.

Refining the goals of this work, described in section 1.2

- A thorough analysis of biological sensory systems from an ADC perspective
- Selection of the most suitable sensory model for an ADC, which can be implemented.
- Study of various neural models to find the most viable model.
- Study of state-of-the-art analog VLSI spiking neural models to find the most suited model to implement the concept
- Modifying the selected neuron models to have the advantages of the biological counterparts, while removing their disadvantages (for e.g., not limiting our-self to the speed of biological neurons).
- Study of various coding schemes, and comparing the selected coded schemes with conventional schemes.
- Design of electronic equivalent of the selected biological concept and testing it on a schematic level
- Physical design of the corresponding architecture
- Manufacturing and testing of the first chip in CMOS technology.

Chapter 3

Biological Sensory Systems

Sensory systems in living beings perform tasks identical to signal conditioning systems discussed in the last chapter. There have been several adaptations of parts of sensory systems in various real life problems. We will have a look at various transduction processes that occur in human body and some special cases in other living beings.

3.1 Human Sensory System

Human Sensory System (HSS) acts as a nice starting point to understand the sensory systems in living beings. In Figure 3.1 a simple description of the sensory units are provided as a mind map. Here, it can be observed that each sensing unit preforms multiple tasks. The nervous system including the spinal cord and the brain process the information, which includes sensor fusion (for e.g., effect of smell on the taste of the food, Pavlovian reflex etc.). The transduction of signals in each of these systems and the signal processing by the nervous system are both essential, when designing a signal conditioning system. Let us take a look at each of the sensory units and their unique properties from an engineering perspective before going into the transduction and signal conditioning methods.

Eyes

The eyes are the primary sensory units of the human vision system. Their structure is very similar to that of a camera. It has a huge array of several types of sensing units which are capable of performing different tasks. In humans there are three types of cones and one type of rods, which help in color and night vision, respectively. [99]

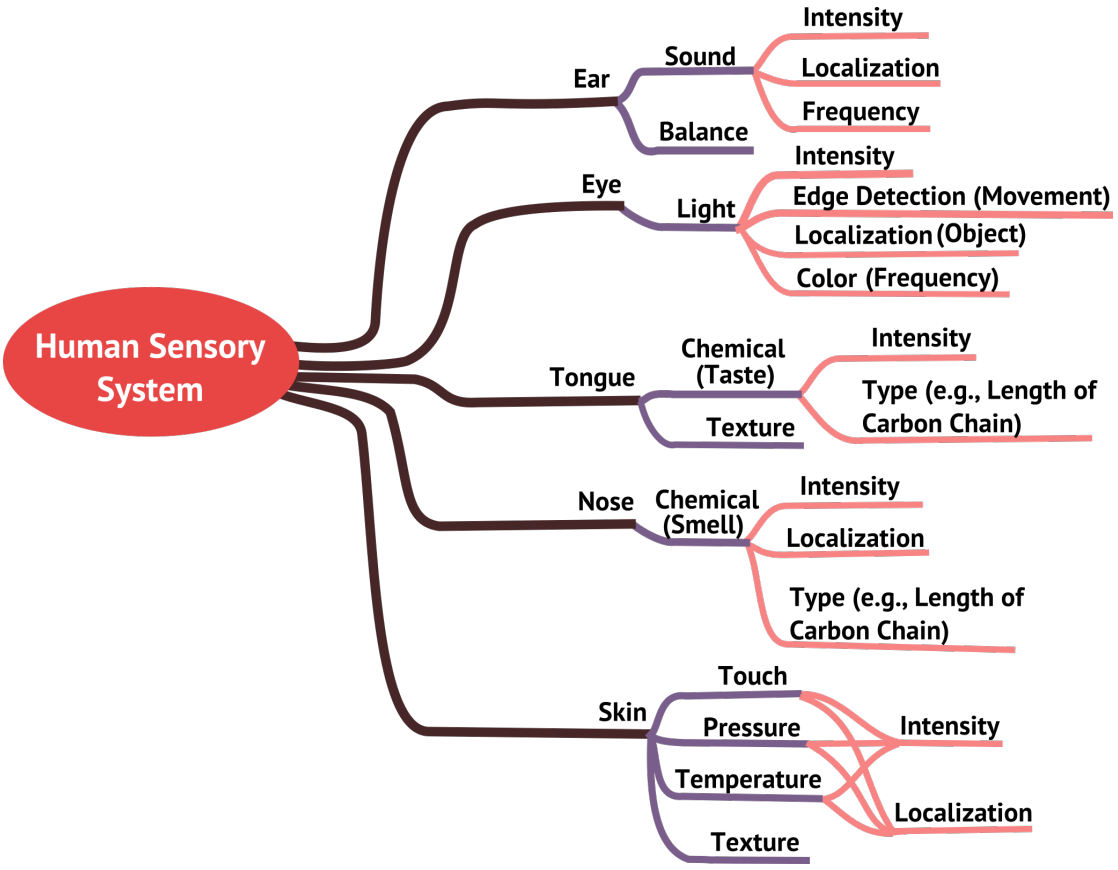


FIGURE 3.1: A Mind Map of Human Sensory System

Skin

The skin has a large area of "sensors" which can detect touch, pressure and temperature. The sensitivity and density of these "sensors" vary at different parts of the body depending on their importance.

Tongue

The tongue is a modification of skin, which is able to sense different chemical signatures and classify them. It consists of several detectors, which are tuned to detect specific chemical structures. Tongue is used for other purposes in a few reptiles (For e.g., Snakes use their forked tongue to detect vibrations and locate prey).

Nose

The nose is similar to the tongue, since it detects chemical signatures too, although it uses "non-contacting" principle. It is a multi sensor system similar to other sensory units described above.

Ears

The ears consist of bones, membranes, liquids, haircells etc., which detect vibrations. In humans they are placed on the sides of the head and are almost impossible to move independently. These bones are also used to maintain the balance of the body. Unlike the previous sensory units, the ear has the least number of "sensors" which is similar to many sensing units we use practically.

Others

The sensory systems in other living beings are also extremely interesting like the infrared sensors in snakes, magnetic sensors in pigeons, foxes, antennae of insects, whiskers of mice etc. A few of these will also be discussed later in this chapter.

3.1.1 Intensity Detection

FIGURE 3.2: A Simplified Illustration Light Intensity Detection in Human Eye [29].

Every one of the sensory units described above can detect the intensity of their respective signals [29]. Each "detector" unit is sensitive to the signal, however, this sensitivity is non-linear. The intensity detection can be represented by different codes in engineering, for e.g., increase in voltage with respect to an increase in input signal, increase in pulse

width with respect to increase in the input signal etc. In the biological neural networks two codes are considered to be the most common, (1) Rate Code where the frequency of the spikes is proportional to the input signal and (2) Time to First Spike, where the time to spike is inversely proportional to the input signal [33]. These codes can be seen in the Figure 3.2. A detailed comparison of the various codes will be presented in later chapters.

The pulse width encoding is the expected encoding technique in conventional TDCS [16]. The spiking neural techniques mentioned in the previous chapter mostly make use of these transduction techniques.

3.1.2 Edge Detection

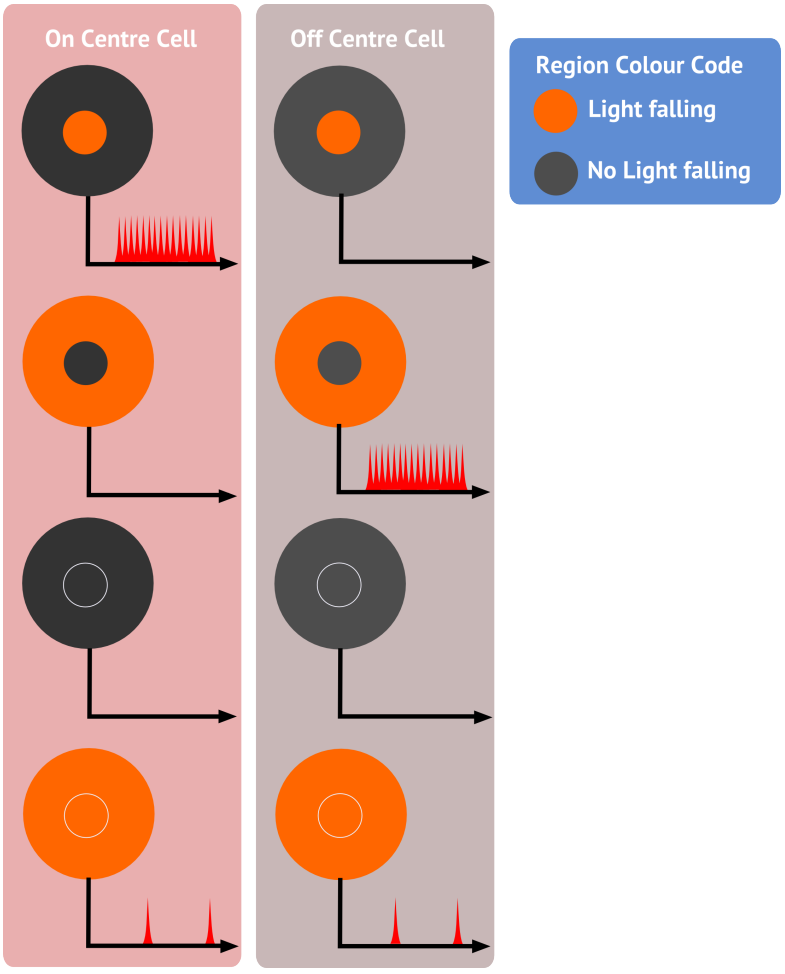


FIGURE 3.3: Edge Detection Ganglion Cells of Human Eye [30].

Edge detection makes use of receptive fields with both on- and off-centers to detect the edges of objects [100]. This helps in fast movement detection. The use of on- and off-center receptive fields is very effective. Schaik et. al. [27] use a similar structure in their ADC, however, they do not mention an inspiration from this structure.

3.1.3 Wavelength Detection

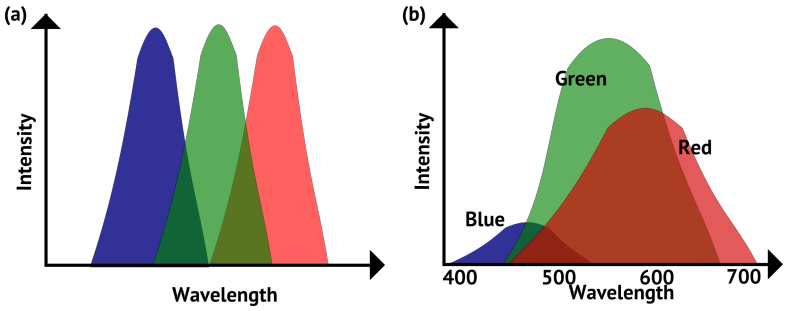


FIGURE 3.4: (a) Ideal Sensitivity (b) Actual Sensitivity to Wavelengths in Human Eye [31]

Wavelength or spectral detection is performed in eyes for color vision [101] and ears (In the nose, tongue, and skin it would be similar to chemical and texture detection). It is commonly performed by having a set of detectors sensitive to different wavelengths. The region, where this "detector" is able to bond to the signal is its receptive field. Receptive fields on their own are not very interesting from an engineering perspective since they would be similar to a flip-flop or a latch. The resolution of such a sensory unit would be equal to the number of receptive fields that are present. When there is an overlap between these receptive fields, then these coarse "detectors" can be used to detect much finer changes in the signal. Figure 3.4 (a) shows an ideal representation of receptive fields, while Figure 3.4 (b) shows the sensitivity of cones in the human eye responsible for color vision.

3.1.4 Localization

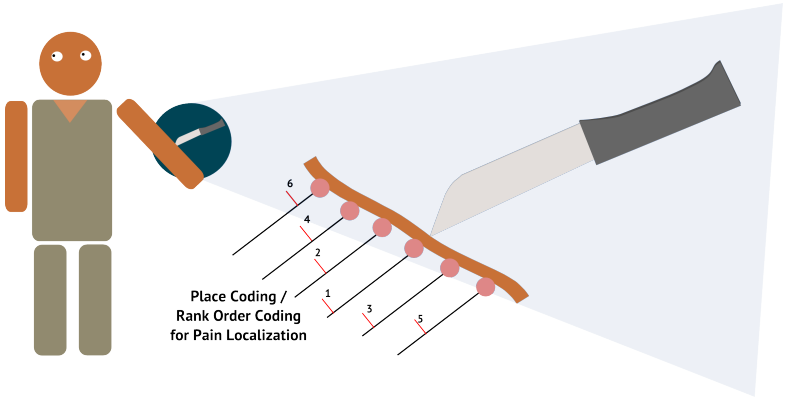


FIGURE 3.5: Pain Localization in Skin

Localization of different signals or the answer to the question "where?" is one of the most important senses for any predator or prey. Every sensory unit has the capability for localization. The use of place coding to find the location of pain is shown in the

Figure 3.5. It can be observed that multiple "detectors" are required for this type of localization [102]. A similar technique is used for object recognition in the human eye using Rank Order Coding / Place Coding [103]. These techniques make use of data from several sensors, the density of sensors is directly proportional to the precision of localization, for e.g., hand has higher density of mechanoreceptors than a less important region like the arm. Although the use of several sensors and sensor fusion is becoming more common in conventional signal processing systems, typically data from a single sensor is used. Many animals from insects to mammals have paired sensory units, which use data from only two sensors which are paired with each other for localization and Tropotaxis. These use the minimum number of sensors (i.e., two sensors) for localization. These paired sensors seem to have unique advantages in localization of objects. ADC as a localization problem has not been studied in the past, this might be a good starting point to study in detail in the next section.

3.1.5 Signal Processing in Central Nervous System

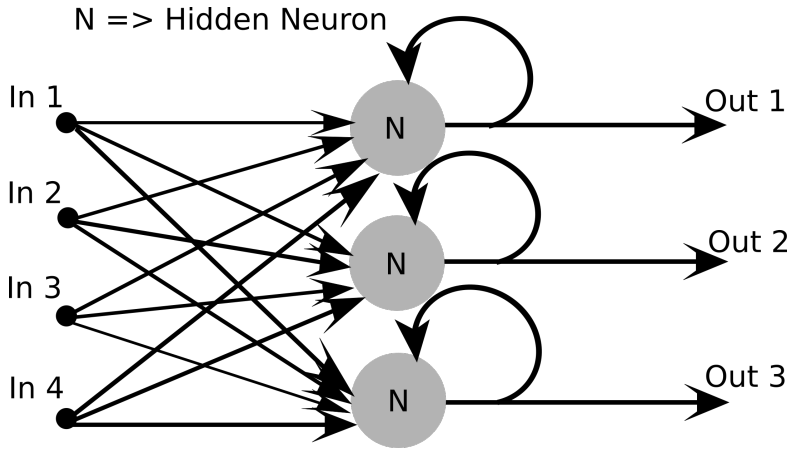


FIGURE 3.6: A Simple Recurrent Neural Network [32]

Signal processing in the central nervous system is still not completely understood, however, there are many types of neural information processing techniques that have been proposed. The most popular methods are forward networks, convolutional neural networks, recurrent neural networks, Deep Learning, Hebbian learning, Kohonen SOM , RBF neural networks etc. While RBF neural networks are based on frequency detection schemes using receptive fields described above, most other networks are based on the working of neurons in central nervous system. Hopfield networks, which are now categorized under recurrent networks have been used extensively in the design of amplitude domain neural ADCs as described in the previous chapter.

3.1.6 Summary

In this section, an engineering perspective of human peripheral and central nervous system and their uses in signal conditioning have been discussed. Many neural ADC approaches tend to make use of the concepts inspired from central nervous system like the Hopfield ADC and its derivatives. The concepts like multi-sensor fusion are inherent to neural networks, the advantages of this is not yet completely tapped. Although localization seems to be a promising approach, it requires multiple sensors, however, paired sensory units for localization might provide a good starting point to study this approach.

3.2 Localization using Paired Sensory Units

3.2.1 Tropotaxis

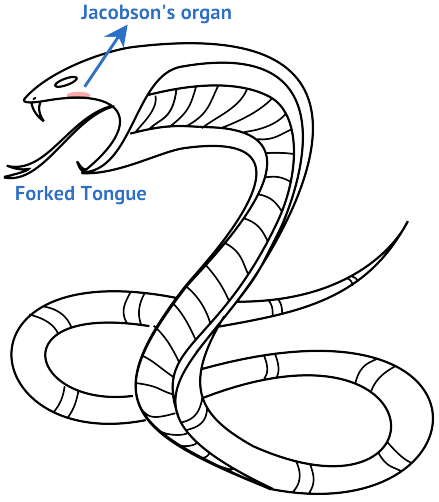


FIGURE 3.7: Forked Tongue of a Snake used for Smell Localization

The presence of paired sensory units is almost ubiquitous among animals. The best example to show the advantage of paired sensory units is the forked tongue of a snake. Snakes use their forked tongue in combination with the Jacobson's organ located at the top of their mouth for smell localization Figure 3.7. The forked tongue collects the data, while the Jacobson's organ acts as the sensory element. This "feature" is not present in humans and other mammals. The evolution of a forked tongue for such localization shows the importance of paired sensors. Accurate localization is impossible with the presence of one sensory unit. Binocular vision for better depth perception also requires paired sensory units. Most predators have binocular vision for accurate targeting, while prey have peripheral vision for easily locating a predator. There is significant research

in the use of antennae of cockroaches for tactile localization, among many other sensory purposes [104]

3.3 Jeffress Model for Sound Localization

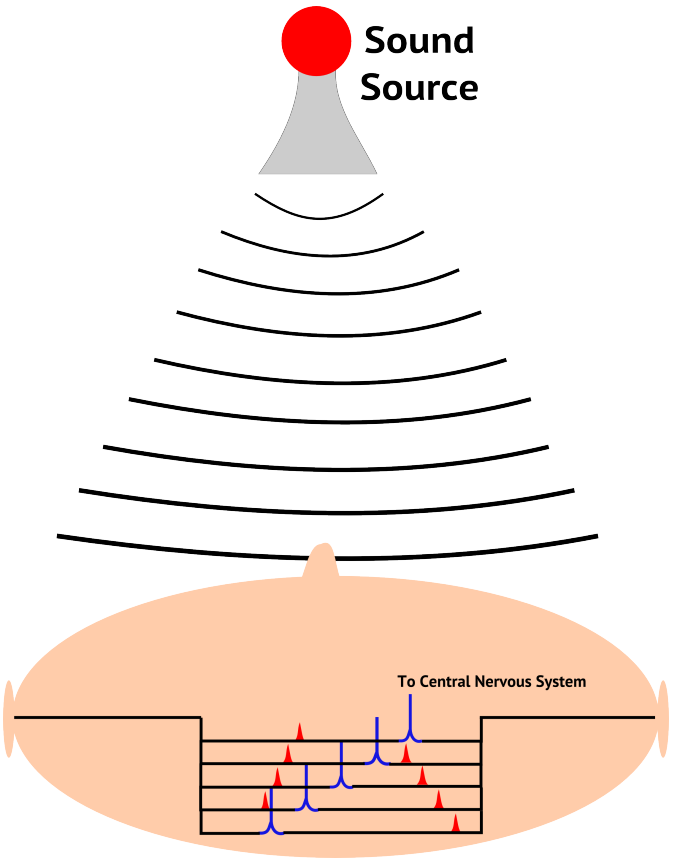


FIGURE 3.8: Jeffress Model of Sound Localization

The main example of paired sensors in humans are the ears. Sound localization in human ears was theorized by Jeffress in 1948 [105] as shown in the Figure 3.8. The time delay between different branches are used as the base for localization. The signals "meeting" each other through these time delayed paths are checked for coincidence. Depending on the paths that are active for coincidence, the localization can be achieved. Each of these coincidence detectors act as virtual sensors, thus creating the effect of a population of sensors. The output from these is very similar to the output from skin localization shown in Figure 3.5. It is important to note that these coincidence detectors act as receptive fields based in time instead of frequency as described in frequency detection. These detectors produce a place code instead of a 1-of-n code. The advantages of place codes will be described in detail in the following chapters.

3.4 ADC as a Localization Problem

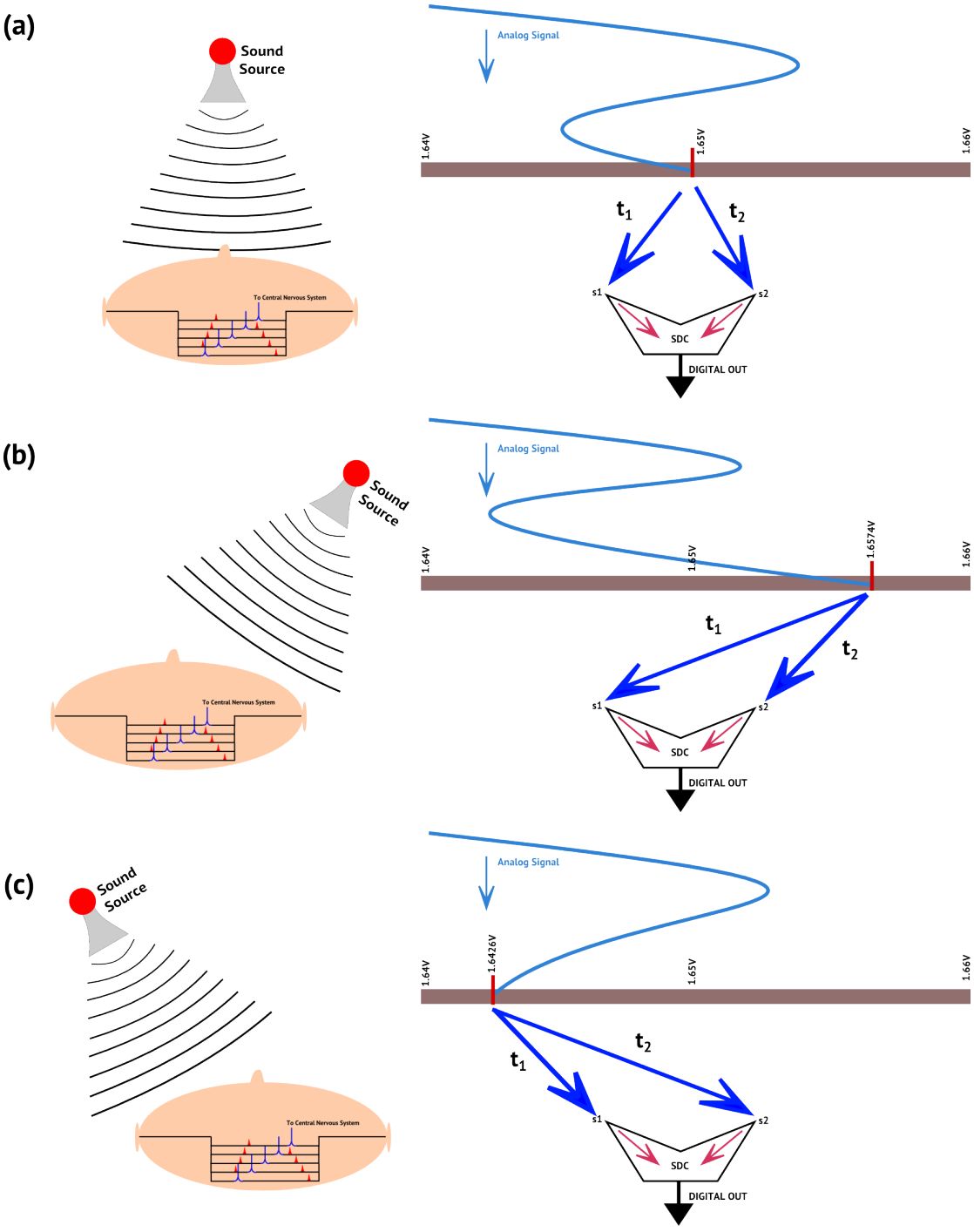


FIGURE 3.9: ADC as a Localization Problem (right), Jeffress Model of Sound Localization Visualized (left). (The values t_1 and t_2 indicate the time taken from the source to s1 and s2 respectively.)

Localization is an inherently linear problem, unlike, intensity detection for instance, where an exponential transduction has more advantages. This makes it a natural inspiration for an ADC. Figure 3.9 describes the concept of an ADC based on auditory

localization described in the previous section, where s1 and s2 are the two paired sensors. If we construct an ADC based on Figure 3.8, we would get the structure shown in Figure 3.10. The inputs and outputs here are in spike codes, converting spike time intervals to spatial codes. This structure can be constructed in any technology and tested to see if it can function as theorized. The amplitude to spike conversion also can be performed by neurons similar to the neurons used as coincidence detectors. The second architecture is based on "Tropotaxis" found in many insects and reptiles, including snakes [106]. It has been observed, that the animals using Tropotaxis move their heads or bodies in the direction of the source until the signals from the paired receptors coincide. This phenomenon has many advantages since it is capable of self-calibration. We can simulate the movement of the head by several methods. One of the methods is described in Figure 3.11. Here the head movement is "controlled" by changing the delays of the delay elements. The degree of change of the delay elements provides the required digital output. Figure 3.12 shows another way of implementing Tropotaxis based ADC.

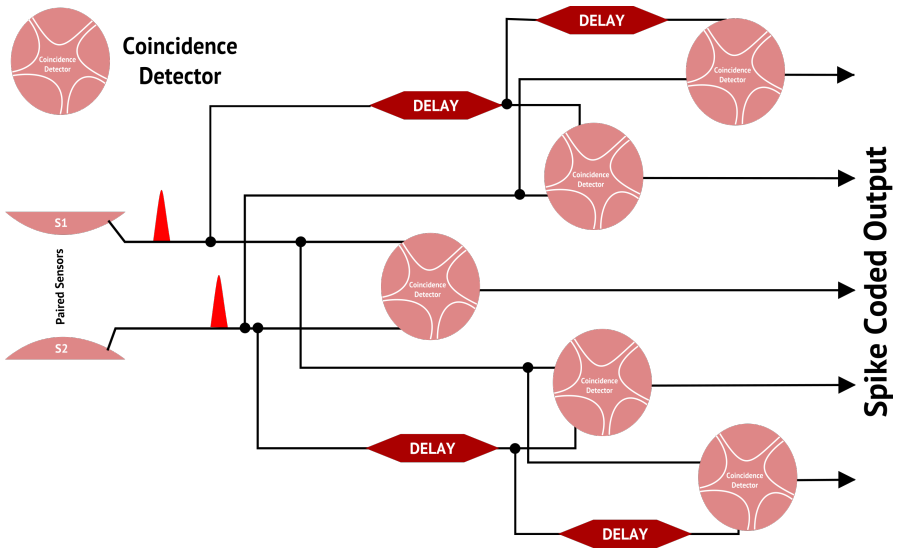


FIGURE 3.10: Architecture 1: ADC based on Jeffress Model

3.4.1 TDC and SDC Architecture

A comparison between Figure 3.13 with Figure 3.10, and Figure 3.11 shows the similarities and differences between the conventional TDC architectures and the presented architecture (which will be referred to as Spike to Digital Converter or SDC from now on.). The following advantages can be clearly observed.

- The TDCs use pulse width modulation as the input, while this structure uses spike interval codes. This means that the TDC inputs can work only in one half of the span, i.e., the start signal should always be before the stop signal. This is not

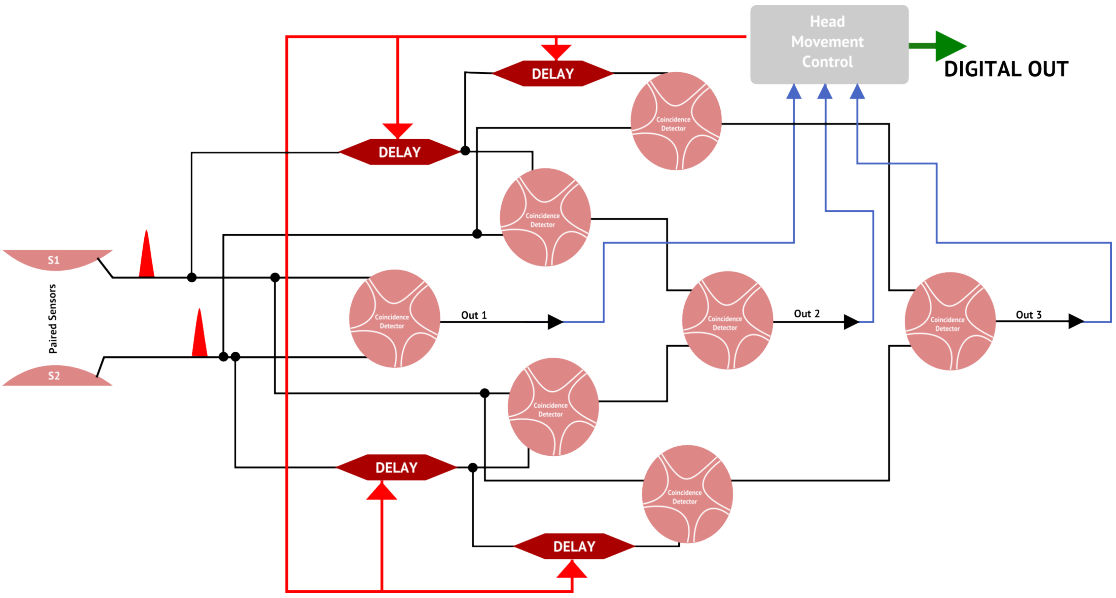


FIGURE 3.11: Architecture 2: ADC based on Tropotaxis

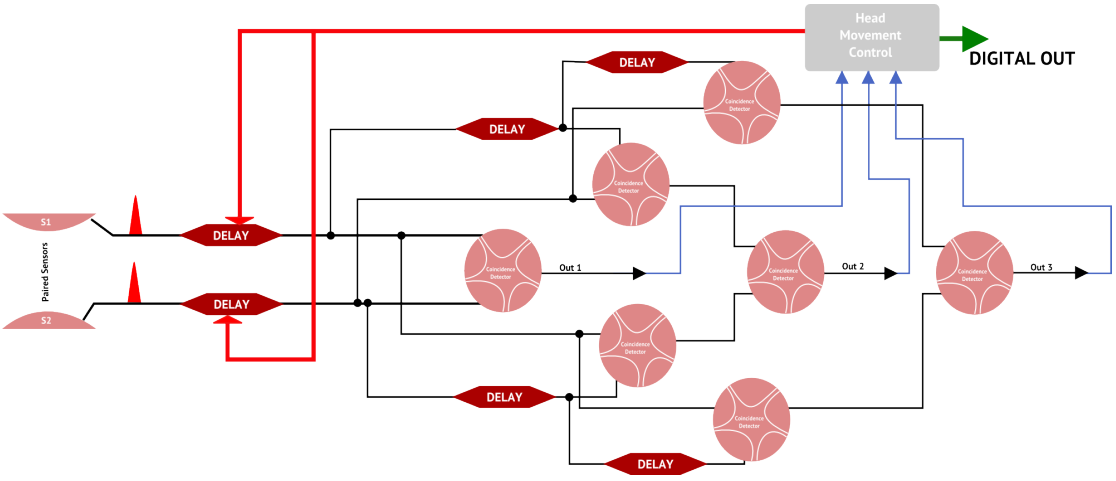


FIGURE 3.12: Architecture 3: ADC based on Tropotaxis v 2.0

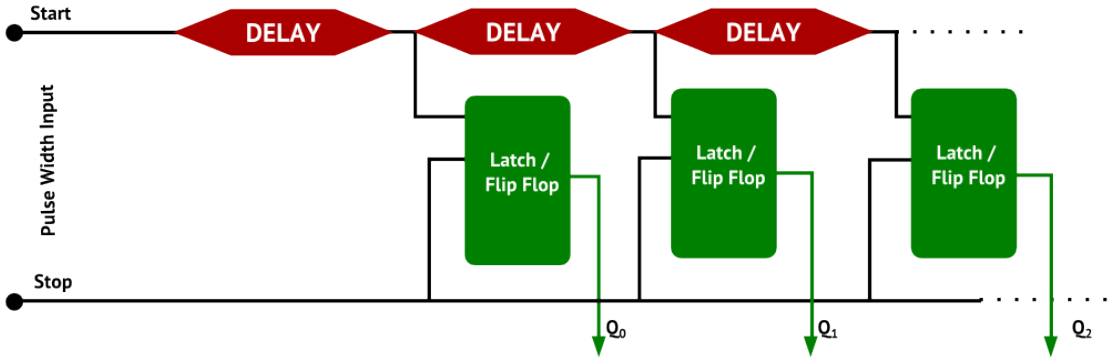


FIGURE 3.13: Simple TDC Architecture

the case with spike interval coding so twice the amount of information can be represented by this code as compared to pulse width modulation.

- The SDC architecture produces a set of spikes for zero input. This information is crucial for self-calibration of the structure.
- The symmetrical structure of the SDC provides an advantage, that the mismatch in manufactured components can be detected inherently.
- Coincidence detection offers more information than a flip-flop/latch. This will be discussed in detail in Chapter 6.

3.5 Time Interval Code vs. Pulse Width vs. Time to first Spike vs. Rate Coding

Figure 3.14 shows four different popular time domain coding schemes. Time to first spike coding scheme is found in the nervous system of animals, mainly proven to be used for object recognition by the eyes [103]. The main advantage of this scheme is the speed of processing, unlike rate coding where a window of spikes is checked for the frequency. On the other hand, rate coding is robust. Pulse width modulation is the standard coding technique used in TDCs. In terms of advantages, it is similar to rate coding for processing, however, its robustness when compared to rate coding must be studied. Time Interval Coding has the advantage of at least having twice the range of pulse width modulation in the same time scale. It also has two pulses, which are active for a "zero" input. This is different from the other codes discussed here. Since two pulses carry the information, it can be considered quite robust compared to other schemes, as any spurious spike can be easily avoided during decoding.

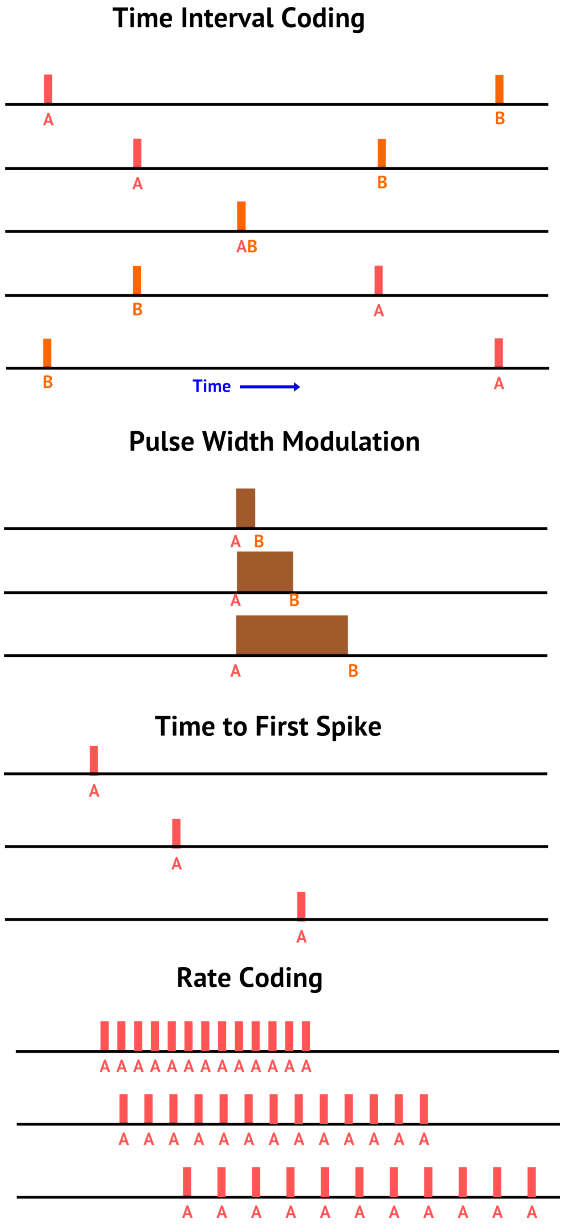


FIGURE 3.14: Visualization of Four Major Time Domain Coding Schemes

3.6 Discussion

Sensory System and the peripheral nervous system have a significant evolutionary history of sensor signal conditioning and processing. There are several interesting concepts here that can be used in various electronic systems for, e.g., cold blooded animals can be studied for sensors with energy harvesting and low power. While vision in humans and other animals has been explored in research, the other sensory systems are not in the forefront of research yet. In this chapter, an abstract study of the sensory system from an engineer's perspective has been presented. Since most of the sensory units use a population of sensors, human auditory system and other sensory systems with paired

sensors were selected as the first step to design an ADC. Two architectures based on sound localization and Tropotaxis have been presented and compared to conventional works. The similarity with TDC structures show, that these concepts can be easily converted to manufacturable electronics. It can also be observed that these structures are inherently robust, and capable of self-calibration. Local adaptation by making use of hybrid architectures will be discussed in Chapter 8.

Chapter 4

Neuromorphic Computing

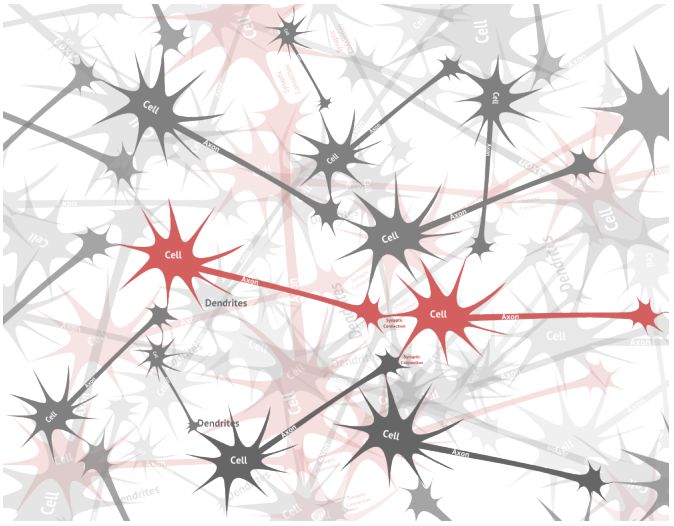


FIGURE 4.1: Biological Neurons

While Neuromorphic Computing is still not technically mainstream, people are slowly becoming aware of its advantages. The dream of parallel computing with noisy adaptive modular silicon neurons, which are naturally capable of powerful pattern recognition, has been a part of research for more than three decades. There are several reasons why neurons are being modeled in silicon.

Some of them are

- Software neural models do not run efficiently on von-Neuman systems.
- Parallel computing with noisy neurons
- Understanding the human brain to imitate or to cure diseases.[72] [38]
- Interacting with biological neurons [107]

- Replacing traditional analog with neural modules.
- Create autonomous robots which can learn like children [55]

4.1 Biological Neuron Dynamics

Understanding the functioning of the nervous system was an integral part of the creation of the initial biological models. The models have various abstraction levels from extremely complex and realistic models to point neuron models. Regardless of the abstraction level of the neural model, the basic neuron dynamics works as shown in the Figure 4.2. It should be noted that artificial neural models like multi-layer perceptrons do not posses such dynamics.

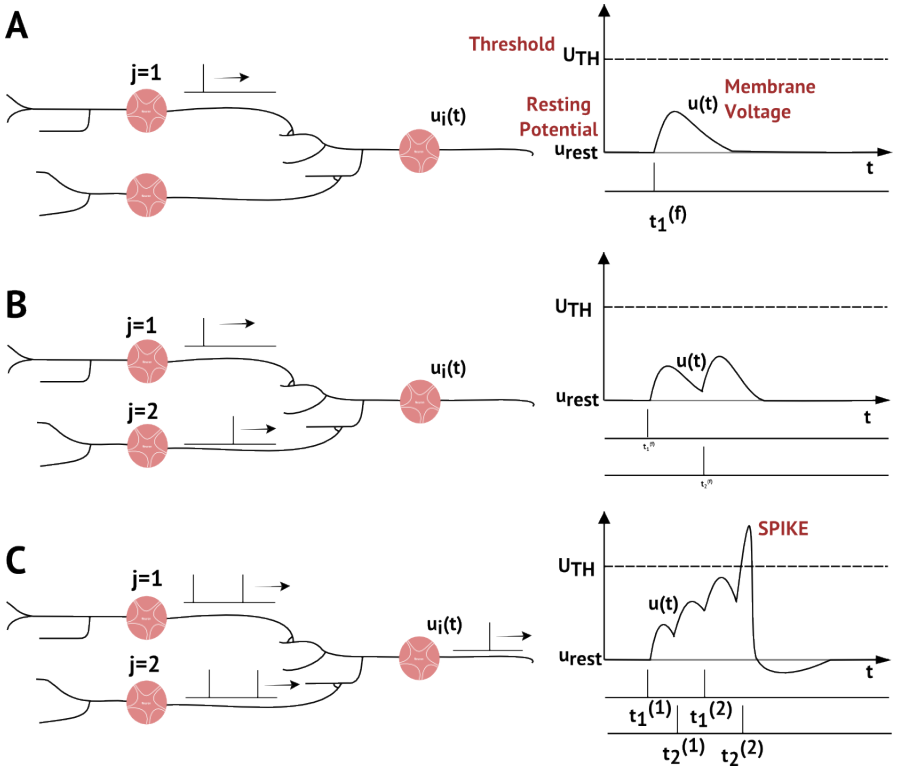


FIGURE 4.2: Neural Dynamics [33]

The neuron has several properties as shown in Figure 2.21

- Threshold: The neuron spikes, when the membrane potential is greater than this value.
- Resting Potential: The resting state membrane potential
- Reset Potential: The neuron moves to the reset potential after the spike and gradually returns to resting potential

- Time Constant (τ): The rate of discharge of membrane potential to resting potential.
- Refractory period: The time taken for the neuron to return to resting state from the reset state.

In Figure 4.2 A we can see a single pre-synaptic spike moving towards the target neuron. In the corresponding plot we can observe the characteristics of the neuron. This spike is not able to provide enough charge to create a post-synaptic spike. In Figure 4.2 B there are two pres-synaptic spikes from two different neurons, however, a post synaptic spike is still not created. In Figure 4.2 C we can see the presence of the post-synaptic spike. From this series of images we can also derive an interesting effect. The total amount of charge going into the target neuron is inversely proportional to the time taken for the spike (Time-to-First-Spike Coding). It is also completely valid to say that this charge is also proportional to the frequency or rate of the output spikes (Rate Coding). In this work, we will focus on Time-to-First-Spike codes since they are much faster than rate codes, and rate codes have been proven biologically-impossible in many scenarios [103].

4.1.1 Coincidence Detection

Figure 4.3 and Figure 4.4 show how the neuron dynamics described in Figure 4.2 are capable of producing temporal Gaussian kernels. These kernels are the main modules, which work in both Jeffress Model and Tropotaxis described in Chapter 3. The time constant of the neuron described by u(t) in the figures above plays a major role in manipulating the width of the Gaussian kernel. Amplitude based Gaussian kernels, a part of RBF networks are extremely popular in pattern recognition and artificial neural networks.

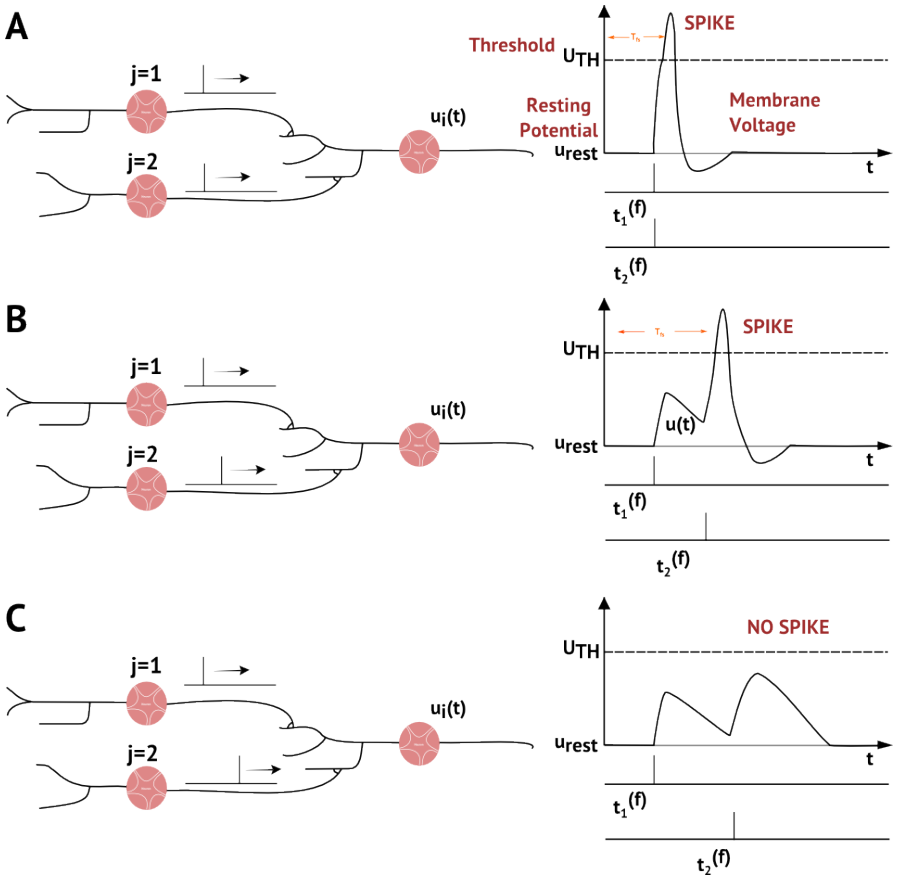


FIGURE 4.3: Neuron Dynamics Creating the Gaussian Kernel

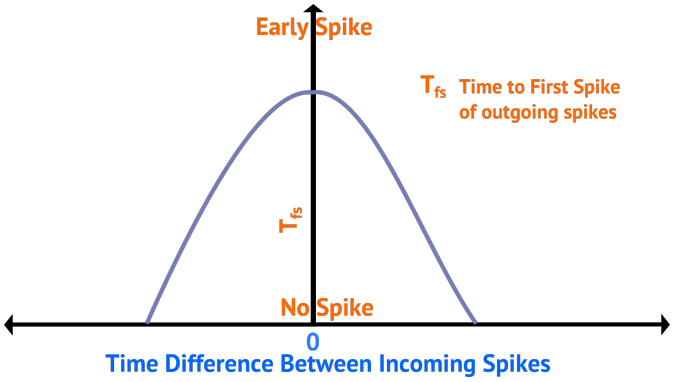


FIGURE 4.4: Coincidence Detection Creating Time Domain Gaussian Kernel based on Neural Dynamics from Figure 4.3

4.2 Biological Neuron Models

There are several biological models of neurons beginning from the simple integrate and fire model in 1907 to custom neural models developed by IBM for its True North Chip. A comparison of various neural models taken from [34] is shown in the Figure 4.5 and in Table 4.2. In Table 4.2 we can see several properties each model is capable of, while it

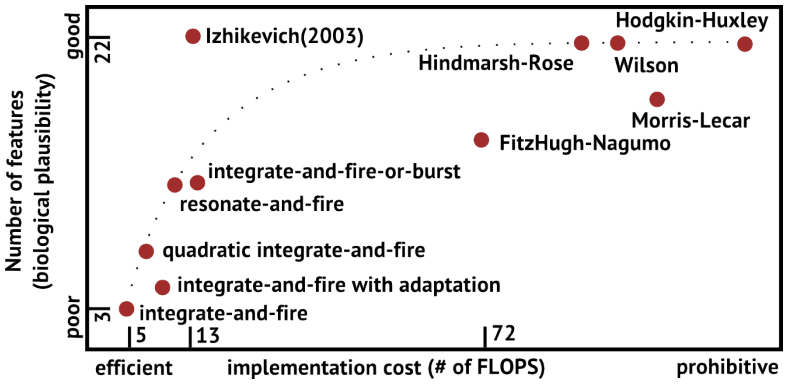


FIGURE 4.5: Comparison of Neural Models [34]

is optimal to find the model with most properties and the least relative implementation cost we should also look at the features required for our proposed circuit.

In our proposed work, we require

- Time-to-first-spike inversely proportional to the incoming charge.
- Coincidence Detection
- To act as a time delay neural network

These basic properties can be implemented with any spiking model described in Section 4.3. From the Table 4.2 and the Figure 4.5, we can shortlist Integrate and Fire based models and Izhikevich models to look for their integrated counterparts.

Models	biophysically meaningful	tonic spiking	phasic spiking	tonic bursting	phasic bursting	mixed mode	spike frequency adaptation	class 1 excitable	class 2 excitable	spike latency	subthreshold oscillations	resonator	integrator	rebound spike	rebound burst	threshold variability	bistability	DAP	accomodation	inhibition-induced spiking	inhibition-induced bursting	chaos	# of FLOPS
integrate-and-fire	-	+	-	-	-	-	-	+	-	-	-	-	+	-	-	-	-	-	-	-	-	-	5
integrate-and-fire with adapt.	-	+	-	-	-	-	+	+	-	-	-	-	+	-	-	-	-	+	-	-	-	-	10
integrate-and-fire-or-burst	-	+	+		+	-	+	+	-	-	-	-	+	+	+	-	+	+	-	-	-		13
resonate-and-fire	-	+	+	-	-	-	-	+	+	-	+	+	+	+	-	-	+	+	+	-	-	+	10
quadratic integrate-and-fire	-	+	-	-	-	-	-	+	-	+	-	-	+	-	-	+	+	-	-	-	-	-	7
Izhikevich (2003)	-	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	13
FitzHugh-Nagumo	-	+	+	-		-	-	+	-	+	+	+	-	+	-	+	+	-	+	+	-	-	72
Hindmarsh-Rose	-	+	+	+			+	+	+	+	+	+	+	+	+	+	+	+	+	+		+	120
Morris-Lecar	+	+	+	-		-	-	+	+	+	+	+	+	+		+	+	-	+	+	-	-	600
Wilson	-	+	+	+			+	+	+	+	+	+	+	+	+	+		+	+				180
Hodgkin-Huxley	+	+	+	+			+	+	+	+	+	+	+	+	+	+	+	+	+	+		+	1200

TABLE 4.1: Comparison between Neuron Models [34]

4.3 Izhikevich Spiking Neuron Model

From our analysis, Izhikevich model is chosen as one of the shortlisted structures. Izhikevich in his work [34] was able to simplify complex multidimensional neural models like the Hodgkin Huxley model to two differential equations using bifurcation methodologies as shown in the equations 4.1, 4.2, and 4.3 below.

$$v' = 0.04v^2 + 5v + 140 - u + I \tag{4.1}$$

$$u' = a(bv - u) \tag{4.2}$$

with the spike resets

$$if\ v >= 30\ mV,\ then\ v \leftarrow c, u \leftarrow u + d \tag{4.3}$$

Here v represents the membrane potential (dimensionless), u represents the membrane recovery after a spike (also dimensionless). The model has the max spike potential set at 30 mV, resting potential between -60 and -70 mV and threshold varying between -55 mV to -40 mV. These parameters are only considered in functionality for the analog models.

The parameter a describes the refractory period, b describes the sensitivity of u , c describes the reset value, d describes the reset of the variable u. Different choices of these parameters will result in different types of firing patterns found in biological neurons like regular spiking, bursting, chattering, fast spiking, cortical spiking etc.

4.3.1 Analog Model

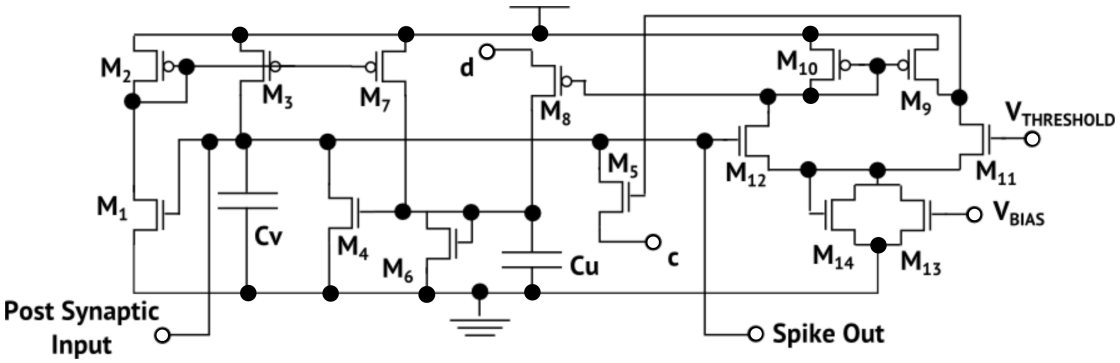


FIGURE 4.6: Izhikevich Neuron [35]

One example of the Izhikevich model has been implemented by Wijekoon et. al. [35]. The circuit is shown in the Figure 4.6. The proposed circuit functionally replicates the

mathematical model. It has 14 transistors and 2 capacitors with configurable threshold, bias, c, and d values.

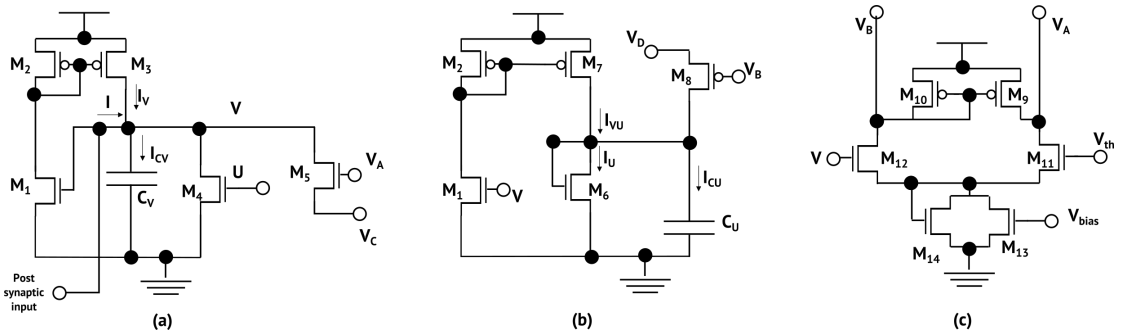


FIGURE 4.7: Izhikevich Neuron Sub Circuits (a) Membrane Circuit (b) "u" Circuit (c) Comparator Circuit [35]

The working of the circuit can be explained by its three sub-circuits as shown in the Figure 4.7.

Membrane Circuit: The current mirror with M_2 and M_3 act as a positive feedback loop for the incoming spikes. M_4 is the leakage transistor controlled by the variable "u". The sum of these currents are integrated by the capacitor C_V. When the spike is generated by the comparator circuit (described below), it is fed back to M_5, which discharges the membrane potential, thus, resetting the spike.

"u" Circuit: This circuit is similar to the previous one, but it is relatively slower. The transistor M_7 discharges smaller current than its counterpart M_3 and the capacitor C_u is larger than C_V. The transistor M_8 acts as the discharge transistor, however, the transistor does not completely discharge the "u" after the spike. The residual voltage keeps increasing with output spikes, creating "Spike Frequency Adaptation" effect present in biological neurons.

Comparator Circuit: The circuit compares the membrane potential V with the threshold potential V_{th} to create the spike. When a spike is created, it sends the spike to V_A and V_B (V_B is an inverted spike). M_{14} increases the comparator current during the spike.

4.4 Leaky Integrate-and-Fire Model

Since the working of integrate-and-fire model is explained in Figure 4.2 and in Figure 2.21, the VLSI models are directly considered.

4.4.1 Analog Model

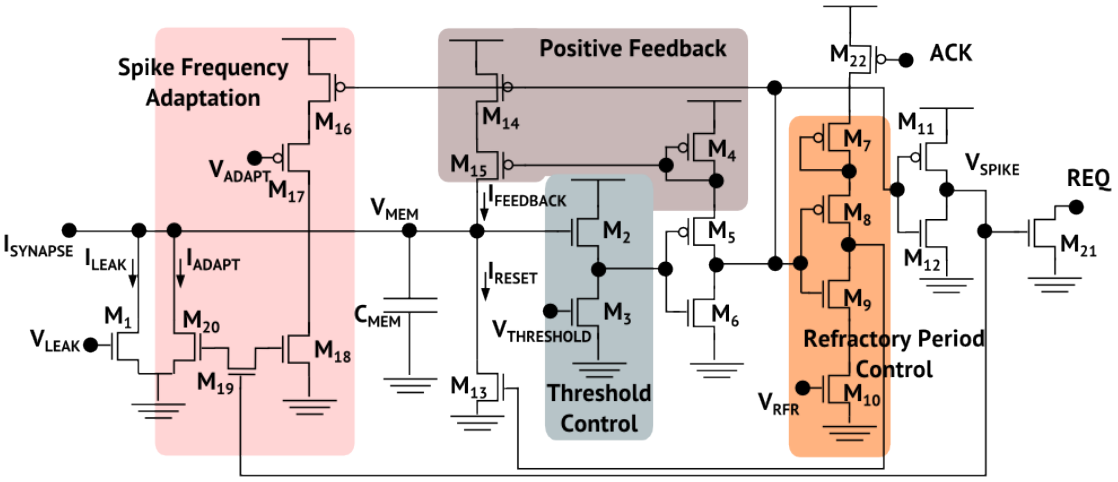


FIGURE 4.8: Integrate and Fire Neuron [4]

An implementation of Leaky Integrate-and-Fire neuron by Indiveri et. al. [4]. This is a modified model with several other properties similar to quadratic integrate-and-fire model. The circuit consists of 21 transistors and one capacitor. The working of the circuit is described below

Membrane Circuit: This circuit consists of the transistors M_1,M_{14}, inverters M_4,M_5 and M_{12},M_{13} and the membrane capacitor C_{MEM}. The current from the synapse charges the capacitor C_{MEM}. This goes through the threshold control circuit to the inverter. When the voltage is greater than the crossing point of the inverter, then it activates the second inverter. The same circuit also activates the refractory period control circuit, which discharges the membrane potential through M_{13}. Meanwhile the transistor M_1 is leaking the membrane potential based on V_{LEAK}.

Threshold Control: It is a simple voltage follower circuit, which gives more options to control the threshold of the neuron.

Refractory Period Control: This circuit controls the transistor M_{13} to discharge the membrane potential, and continue doing it as long as it is required by the voltage V_{rfr}.

Positive Feedback: This circuit is designed for low power consumption. The positive feedback from the first inverter increases the membrane potential faster.

Spike Frequency Adaptation: The output spike is fed back to M_{18}, which controls I_{ADAPT} based on V_{ADAPT}. It reduces the spike rate for continuous inputs. This is a biological property of neurons to adapt to environmental noise.

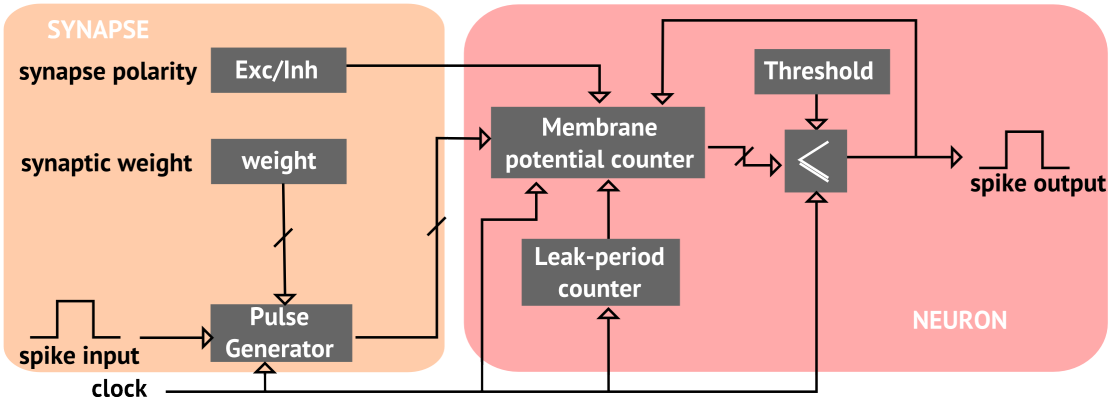


FIGURE 4.9: Digital Leaky Integrate and Fire Neuron [36]

4.4.2 Digital Model

The Leaky Integrate-and-Fire model is the most commonly used neural model, The digital version of the circuit is shown in Figure 4.9. This circuit is simple, It has counters (7-bit) to replicate the increase of membrane potential and the leakage. A comparator, which compares, if the membrane potential is greater than the threshold to reset and create the spike. The synapse consists of a pulse generator with configurable weight. It is also possible to make the synapse excitatory or inhibitory.

4.4.3 Optical Model

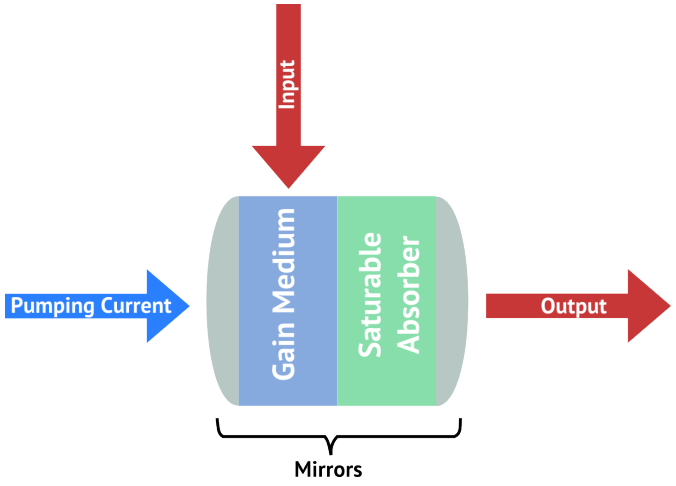


FIGURE 4.10: Optical Integrate and Fire Neuron based on Saturable Absorber (SA) Laser to be Implemented in Vertical Cavity Surface Emitting Laser (VCSEL) [37]

A laser integrate-and-fire neuron has been proposed by Nahmias et.al. [37]. It is based on saturable absorber laser shown in Figure 4.10. This structure has a gain medium, which can be selectively perturbed by inputs either electrically or optically. A network of such

lasers implemented in VCSEL technology can act similar to network of leaky integrate-and-fire neurons, which are billion times faster than their biological counterparts.

4.5 The aVLSI Synapse

While the digital and optical neurons has synapses built in to their neuron bodies, the analog models discussed above require synapse circuits for processing. The synapse plays an important role in the circuit by providing weight to the incoming spike. However, this is not the only task for the synapse. It must produce enough current to charge the neuron, and must be linear to mimic various biological circuits. the design of the synapse and its evolution are shown in Figure 4.11 and Figure 4.12 [3].

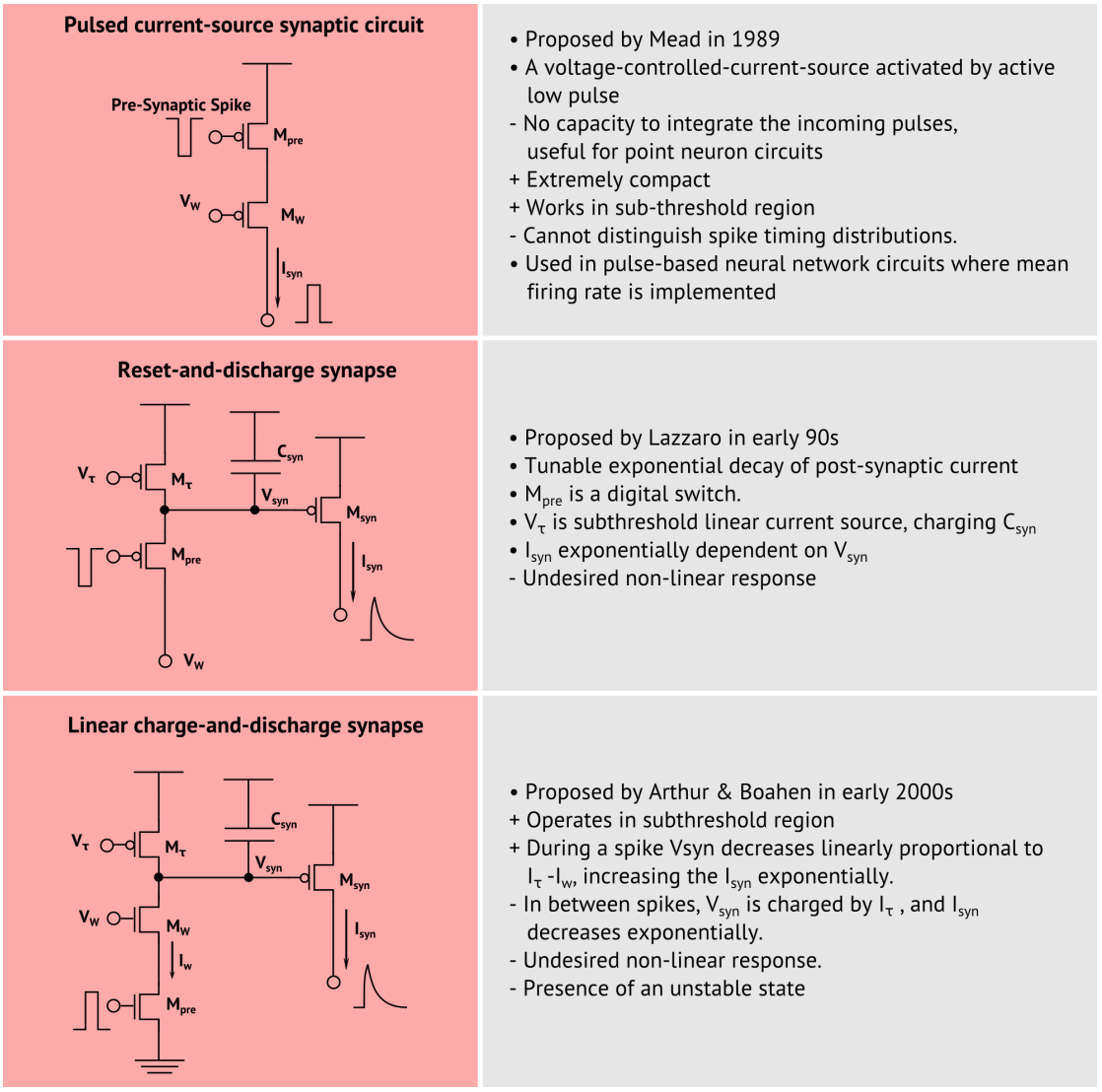


FIGURE 4.11: Evolution of Synapse - Part I [3]

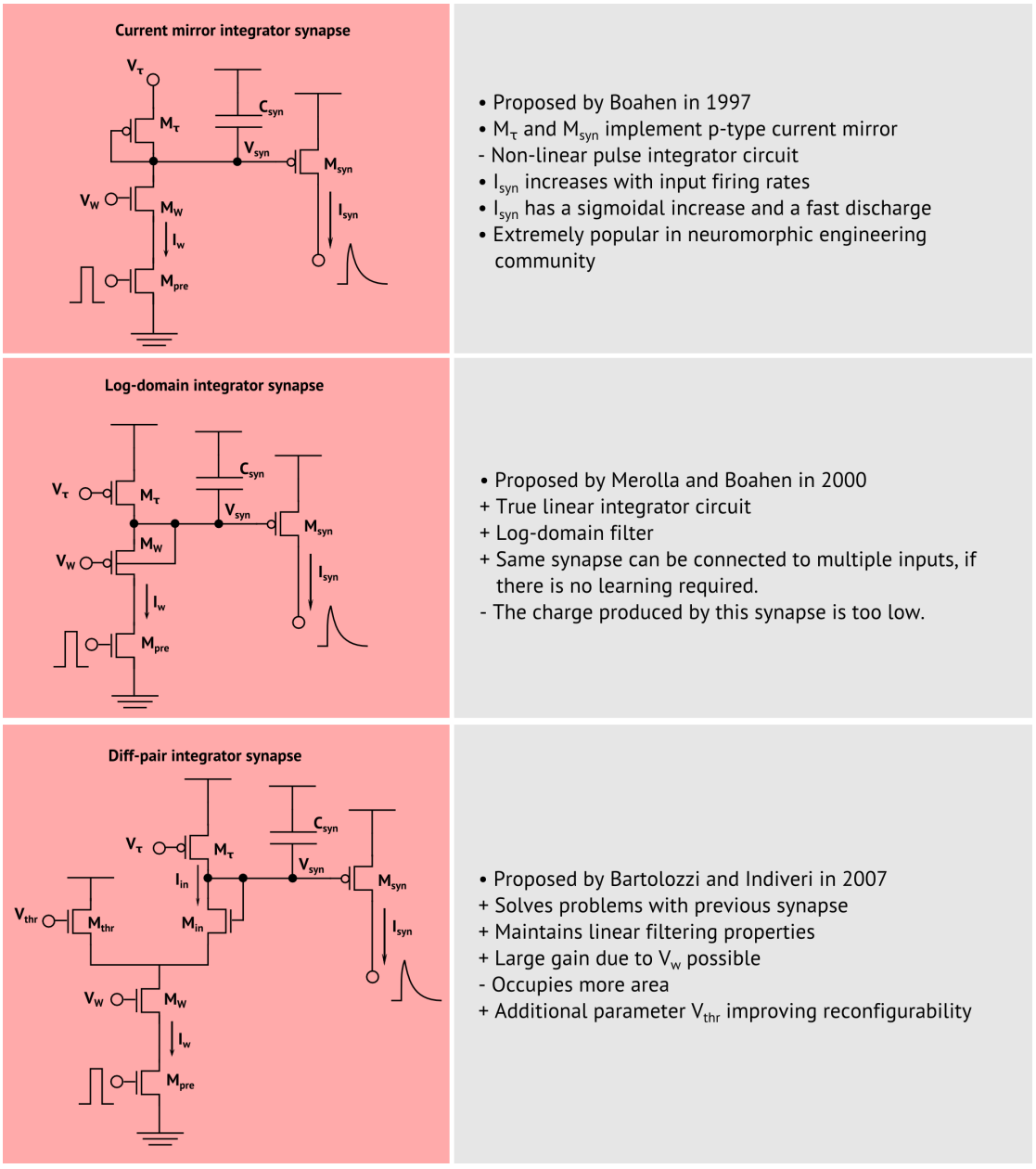


FIGURE 4.12: Evolution of Synapse - Part II [3]

4.6 Brain Scale Neuromorphic Chips

While the long time research of ISE group has been towards applications of neural approaches in sensor signal conditioning systems, understanding of large scale neuromorphic chips built recently might provide an overview and comparison with the approach used in this thesis work. The essence of neuromorphic computing was started by Carver Mead in the early 80s, now there are more than a few architectures that are working towards brain scale computing like IBM, DARPA, HRL and universities like Heidelberg, Manchester, Stanford etc. The main goals of these structures are to mimic brain like

capabilities to create solutions to a variety of problems, which do not include signal conditioning and processing.

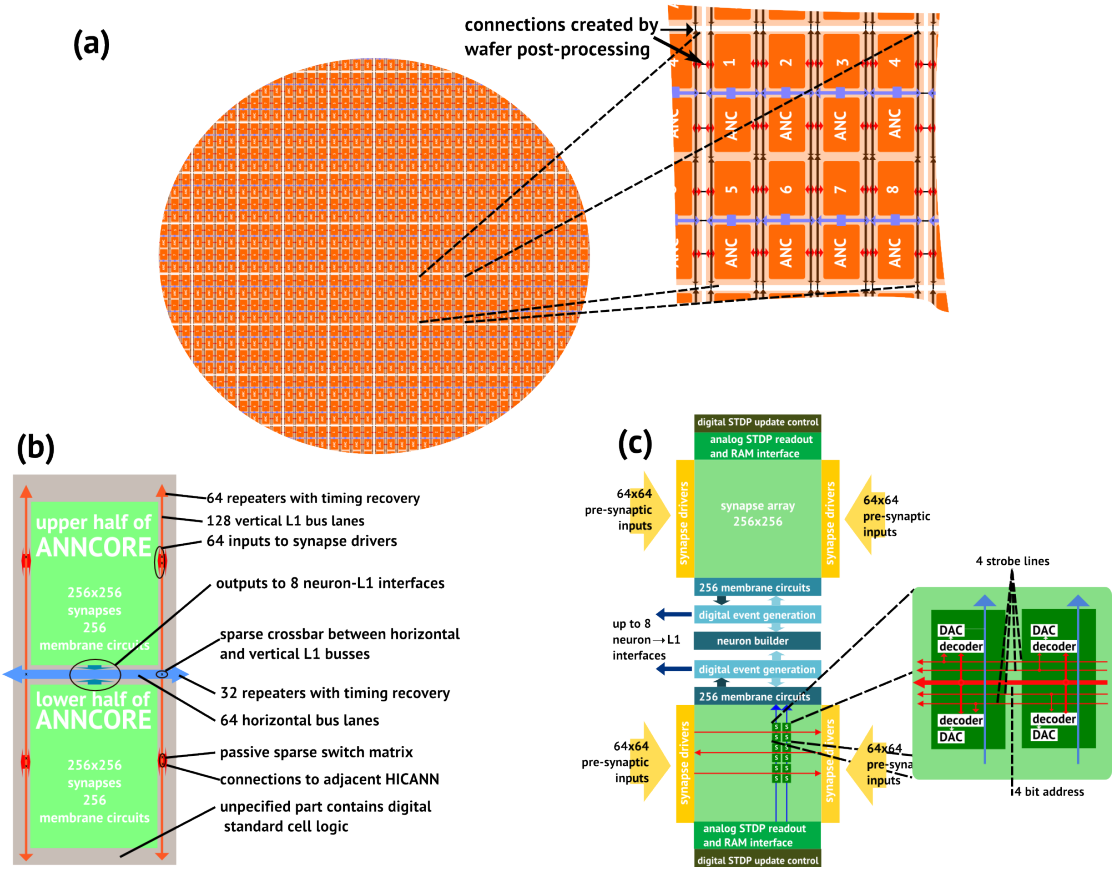


FIGURE 4.13: (a) Arrangement of HICANN Chips on Wafer (b) Main Functional Blocks of the HICANN chip (c) Main Elements of the ANNCORE [38]

These large scale architectures make use of various technologies to achieve their goals, which include mems, digital and analog circuits. The neuromorphic chips from HRL (Huge Research Labs) make use of memristors on CMOS to provide synaptic connections with re-configurable weights [56] . Since the memristors occupy a large area, the spiking neurons are designed to be 10,000 times faster and time-division-multiplexing is used to modify the synaptic connections as required. This creates a much larger set of virtual synapses. The goal of the work is in autonomous robotics where the robot can learn and mature over time. They create large neural networks by connecting all the chips on a wafer and creating interconnecting wafers.

Another group that integrates several wafers is from Uni-Heidelberg on their HICANN chip [38]. This is shown in detail in Fig. 4.13. The neuron core is made of two "chips", eight of these analog cores together form one HICANN chip. The chips are then connected by post processing on the wafer. At present 6 wafers have been connected to create the large scale neural network. This was a part of the FACETS project, which has now been extended to BrainScaleS project [72]. The neurons on these chips are also

several times faster than their biological counterparts. The goal is to create a brain, which can simulate a day of human learning in a few minutes. This can help in learning a lot about plasticity and learning in humans.

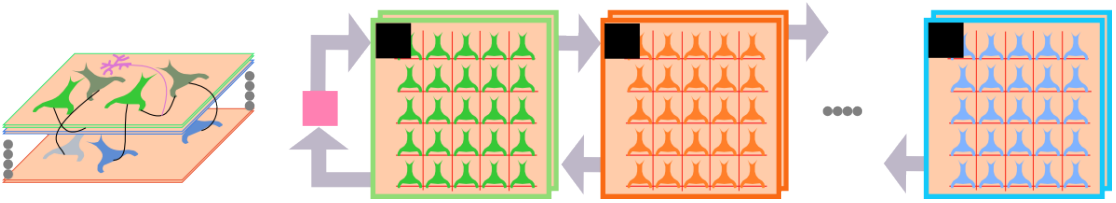


FIGURE 4.14: Neurogrid Platform.(The cortical neurons shown on the left are functionally mapped on to the chips as shown in the right.[39])

Kwabena Boahen et.al. from Mead's lab have created the neurogrid project [57], which is capable of simulating 65,536 neurons per chip with 500 million synapses. The neurons are modeled based on Hodgkin Huxley models, which increases the complexity of each neuron by several orders more than the other approaches here. This project was built to mimic brain and is now focused on connecting such chips to human body for providing bionic extensions to challenged people. The power consumption of this structure can only be beaten by the True North chip of IBM.

The True North chip was created by IBM as part of DARPA's SyNAPSE project. This makes use of a custom digital neuron model, which is comparable to the Izhikevich neuron model discussed in the previous section [58]. This model has one million neurons with 256 million synapses and is the largest neural chip available today. While there is no wafer integration, 16 chips are integrated to create a large scale neural network. The goal is to make neural structures available for mobile sensors and cloud computing.

SpiNNaker is designed by Uni. Manchester as part of the Human Brain project [59]. The interesting part of the chip is the neuron model. They make use of ARM cores to create several different neural models, which can be selected by the user. Each chip can contain up to 16,000 neural models depending on users choice. The present implementation has 1,152 chips connected to form a large scale neural model. The main focus of the project is to understand, simulate and learn more about diseases in human brain.

	True North	Neurogrid	HRL neuro- chips	SpiNNaker	HICANN
Designed by	IBM, DARPA Synapse	Stanford Uni.	HRL labs, DARPA SyNAPSE	Uni Manchester, Human Brain Proj.	Uni Heidelberg, Human Brain Proj.
Neurons	1 million	65,536	576	Upto 16,000	512
Synapses	256 million	500 million	73,000	16 million	114,688
Power Density	20 mW/cm^2	50 mW/cm^2	120 mW/cm^2	1,000 mW/cm^2	3,000 mW/cm^2
Analog/Digital	Digital	Hybrid	Hybrid	Digital	Hybrid
Tech.	28 nm	180 nm	90 nm	130 nm	180 nm
Wafer Integration	No	No	4 Wafers	No	6 Wafers
Chip Integration	16 Chips	16 Chips	N.A.	1,152 Chips	N.A.
Target	Mobile Sensors, Cloud Computing	Autonomous robots	Autonomous robots	Brain disease modelling	Brain learning, plasticity
Neuron Model	Custom	Hodgkin Huxley	Integrate & Fire	Multiple Models	Integration of eng. & neuroscience models

TABLE 4.2: Comparison between Brain-Like Neuromorphic Chips [55][39][56][57][58][59]

4.7 Neuromorphic Architectures for Sensing

As discussed in Chapter 3, neuromorphic architectures would provide several advantages in sensing and signal processing. The goal of this work is to make a generic ADC, which can replace traditional ADCs without any changes to the overall architecture of the systems. The generic ADC should be capable of working with any type of sensor. Many groups are focused on creating sensors mimicking human senses. A few examples are shown below.

4.7.1 Silicon Retinae [9]

There are a few groups working on creating silicon retinae. Research has shown that human eyes are extremely fast at detecting faces [103]. This has led to the creation of Dynamic Vision Sensor, which became a part of the Robotic Goalie project. The robot was able to accurately predict and stop most of the balls going towards the goal. They also make use of stereopsis mentioned in the previous chapter to attain depth perception.

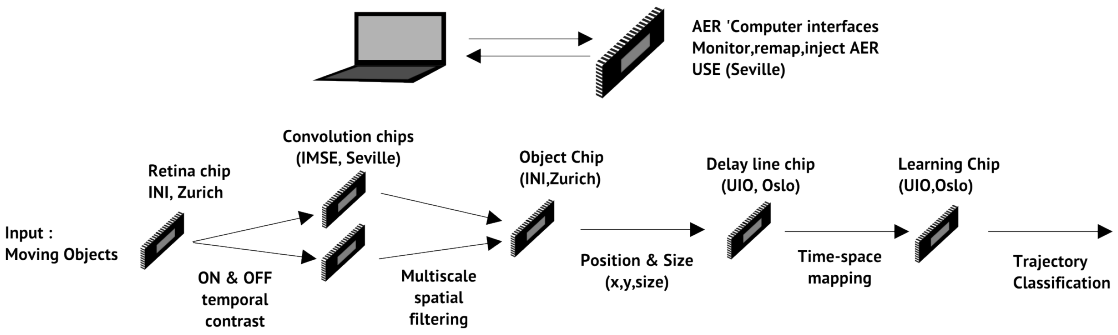


FIGURE 4.15: CAVIAR Multi-Chip Vision System [9]

CAVIAR is a much larger project with several universities working to create massively parallel hardware for high speed object recognition and tracking, capable of handling 2 million events per second. The architecture of the system is shown in Figure 4.15. Each chip performs one step of the process and the chips are connected to each other using address event representation, which will be discussed in the next section. It contains about 45,000 spiking neurons and five million synapses with spike and rate based learning combined with recurrent learning techniques. The connections to the synapses are implemented using convolutional chips. The goal of the work is to create complex real time real world artificial neural systems.

4.7.2 Silicon Cochlea [10]

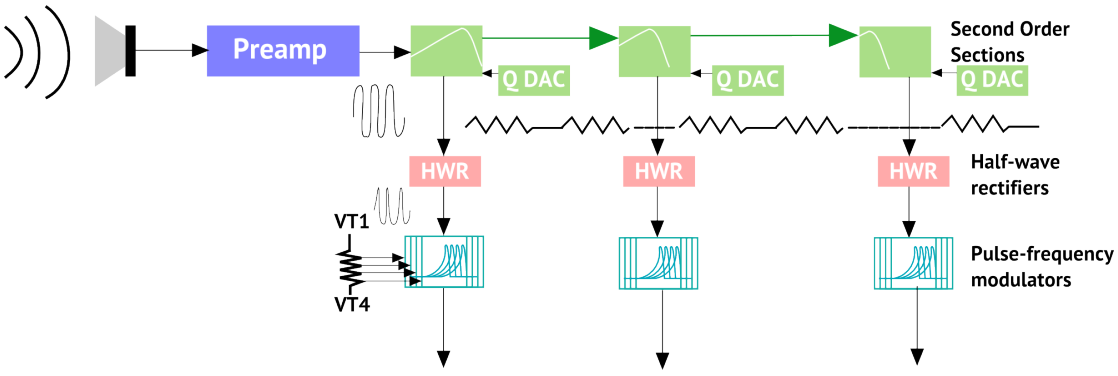


FIGURE 4.16: AER EAR2 Silicon Cochlea

AER-EAR project works on silicon cochlea. They create circuits, that model mammalian cochlea, hair, and the initial neural system. They make use of address event representation (Hence the name AER-EAR) to create the system with several communicating chips. They work with spike frequency processing instead of spike timing. They are able to simulate the sound localization discussed in the previous chapter by cross correlation of frequencies from two microphones. They do not try to emulate biological neurons in terms of speed, their system works at a maximum rate of 50 kHz. The architecture of

the system is shown in Figure 4.16. The future goal is to try and emulate biological models in a more realistic manner.

4.8 Inter-Chip Communication in Neuromorphic Circuits

In the previous sections we have seen various multi-chip and multi-wafer circuits. This seems to be common in neuromorphic systems. Is this required for a relatively simple single chip or a shared chip ADC ? In the proposed concept, coincidence detection has been discussed and the proposed architecture implemented for a 16 bit system will have several hundred neurons. A regular chip cannot support outputs on a large scale, so it must be converted to more appropriate digital format. The most popular approach as seen in the previous section is Address-Event-Representation (AER). The working has been discussed in more detail below.

4.8.1 Address Event Representation

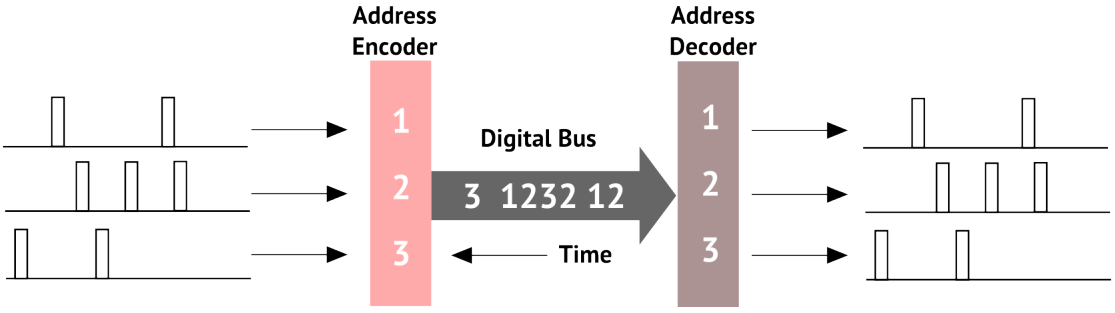


FIGURE 4.17: Address Event Representation [40]

Figure 4.17 shows a very concise description of AER. The goal of the encoder is to encode the address of the neuron that arrives. This system is perfect for our task. However, the implementations of this technique have been only in rates conforming to biological neurons. The system was also designed for chips working on populations of neurons so in many situations the accurate time or position of the spike is statistically unimportant. Many systems make use of arbiters to decide the order of spikes, which occur at the same time. Typically, these arbiters arbitrate for two neurons and are designed in a tree like structure [39]. The result of these arbiters are sent to address encoders. The system is usually designed in a two dimensional fashion with rows and columns to increase the speed of encoding.

4.9 Summary

In this chapter, neural building blocks and larger neuromorphic chips and architectures have been discussed. Based on the analysis in the first section on neural models, the advantage of the Integrate-and-Fire model is clearly observed. The model has been implemented in several technologies as discussed above. Additionally, there are also quantum implementations of the model. The architecture proposed in Chapter 3 is viable in all of these technologies. The technologies accessible for this thesis work are analog and digital. Which technology should be pursued ? This is answered by Joubert et. al. [36]. In their work they study and compare the analog and digital versions of the neuron. While the digital neuron is relatively easier to implement, the power consumption and area is much higher. The digital version can beat the analog version only if the technology is smaller by several orders of magnitude. In their analysis they state that the analog version would have power advantage until the digital version reaches the 22 nm technology node. They implemented both the analog and digital neurons in ST Microelectronics CMOS 65 nm technology. This work can probably achieve more by using simpler neuron design among many other design considerations.

In the second part of the chapter, several large scale neuromorphic architectures have been discussed. While various groups focus on mimicking mammalian senses or human brain, this thesis work and the scope of ISE research will pragmatically examine the building blocks and architectures best suited for creating a practical and viable ADC, which can compete with commercial ADCs and TDCs.

.

Chapter 5

Building Blocks of the SSDC

This chapter focuses on the building blocks of the architecture discussed in Chapter 3. The main goal of this chapter is to show that the analog models work behaviorally similar to their biological counterparts. While the sizing of the structures in this chapter has been found by investigation of sensitive components by simulation of worst case corners, temperature corners, Monte-Carlo analysis, and reducing the sensitivities, these sizes are not universal and will be modified in the following chapters for application specific purposes. The investigation methodology of this thesis work is described at the end of this chapter.

5.1 Leaky Integrate and Fire Neuron

The working of this Leaky Integrate and Fire model has been discussed in detail in the previous chapter (Section 4.3). In this section, the results of the simulations are shown and discussed. The sizing of the transistors is shown in Table 5.1. The sizes are optimized for producing the spiking neural properties with highest variability for various programmable voltages. These values can be further optimized depending on the application scenario with more specific goals.

In the simulations shown below we make use of the parameters shown in Table 5.2 with variations of the parameters explicitly mentioned in the simulation result. The simulations are performed for various pulse width values of input current pulses. The goal of these simulations is to show the properties of the neuron, which can be used in the present and future works.

In Figure 5.1 and Figure 5.2 , we can see that the spikes occur regardless of the frequency of the pulsed input. The main advantage of this is, that the spikes will not vary a lot for

Components	Sizing (w/l) µm/µm	Components	Sizing (w/l) µm/µm
M_1	10/1	M_2	1/10.5
M_3	3/1	M_4	6/1
M_5	6/1	M_6	3/1
M_7	6/3	M_8	6/1
M_9	3/1	M_{10}	12/1
M_{11}	6/1	M_{12}	3/1
M_{13}	6/1	M_{14}	6/4
M_{15}	6/4	M_{16}	1/1
M_{17}	1/1	M_{18}	3/1
M_{19}	10/1	M_{20}	3/1
M_{21}	3/1	M_{22}	6/1
C_{MEM}	400 pF	V_{DD}	3.3 V

TABLE 5.1: Final Component Sizing

Parameters Used	
$I_{SYNAPSE}$	25µA
V_{LEAK}	500mV
V_{ADAPT}	2.75V
$V_{THRESHOLD}$	450mV
V_{RFR}	400mV

TABLE 5.2: Component Parameters used in the Following Experiments. (Variations are mentioned wherever used. These parameters were identified by combinatorial parametric analysis.)

noisy sources for the sensing elements. The neurons will also work for spiking sources. While the speed of the first spike occurrence shown in the simulations is in the order of 500 ns, this can be easily modified and made to work much faster. These are shown later in this chapter.

Figure 5.3 [1] shows how the time for first spike varies with the amplitude of the input current. This shows that the current can be directly converted to spike time data. We

[1] $I_{SYNAPSE}$ and I_{SYN} are used alternatively throughout the text

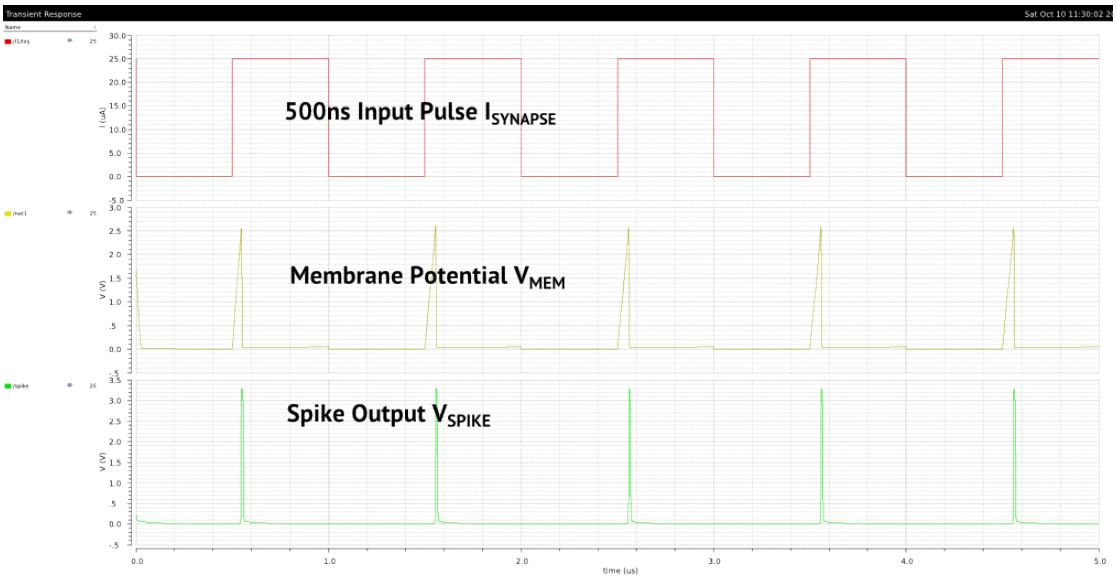


FIGURE 5.1: Simulation of Spikes with 500 ns Input Pulse Showing the Change in Membrane Potential

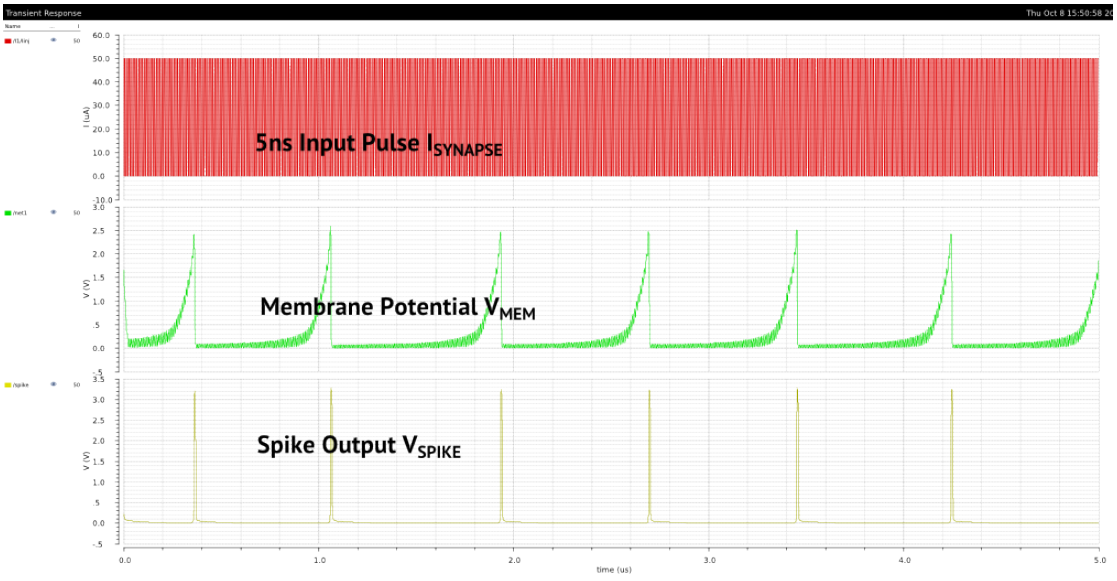


FIGURE 5.2: Simulation of Spikes with 5 ns Input Pulse Showing the Change in Membrane Potential

can also see that the frequency of spikes are proportional to the amplitude of the input current. However, the advantage of using time to first spike codes over frequency codes in terms of speed and power have been discussed in [103] and [5].

Figure 5.4 shows the change in the refractory period of the neuron. The transistor M_{10} is meant to work in sub-threshold region. This feature is extremely useful since the number of spikes in a fixed time period can be easily controlled. Figure 5.5 shows the change of spiking with the change in the threshold voltage of the neuron. While this

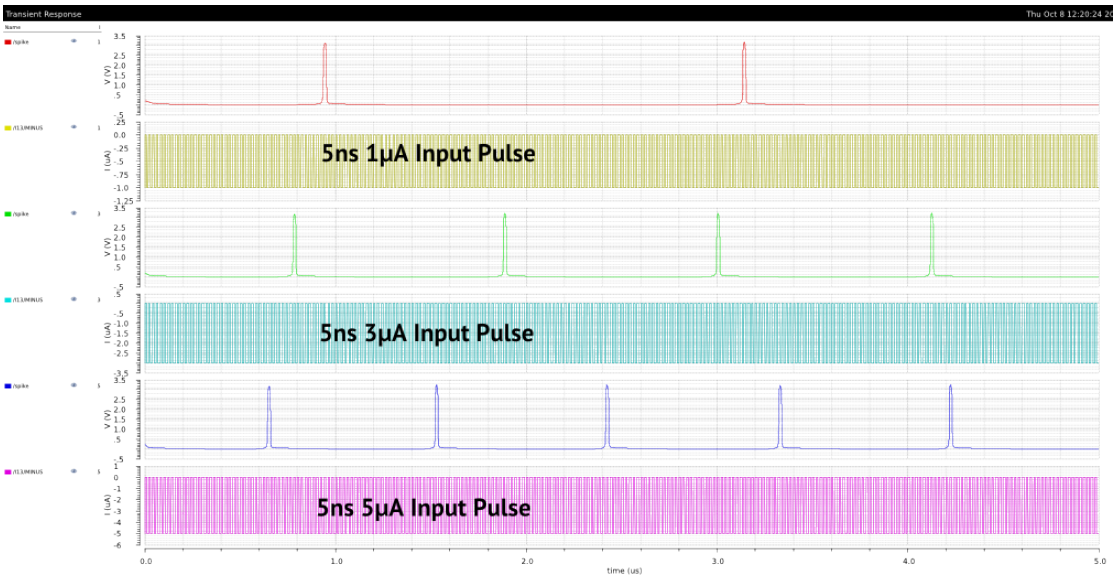


FIGURE 5.3: Simulation of Spikes with Different Input Currents $I_{SYNAPSE}$

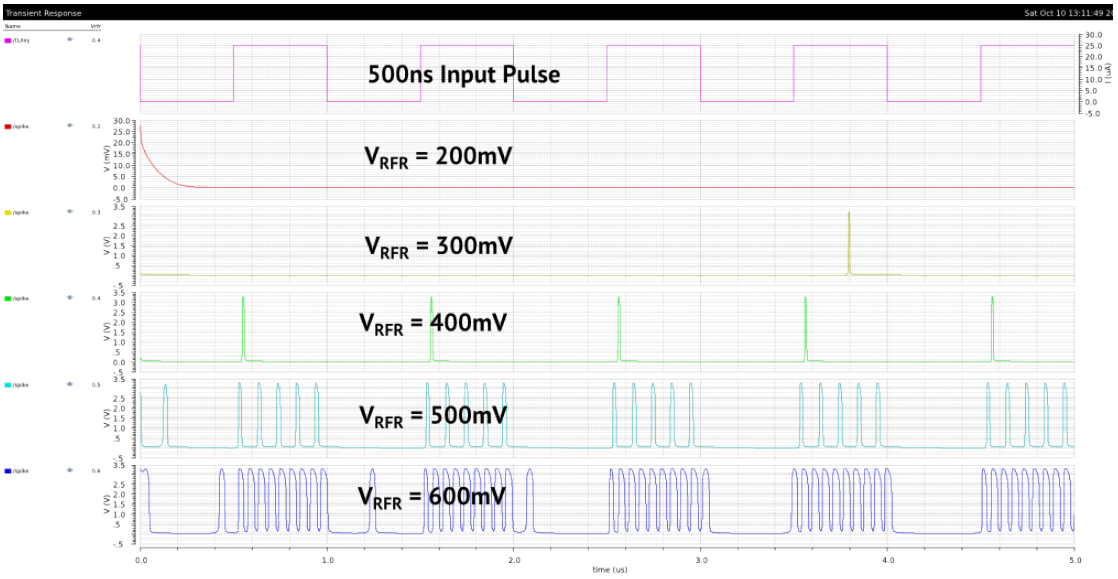


FIGURE 5.4: Simulation of Changes to Spikes with Change in Voltage Controlling the Refractory Period

might be useful for many applications, In this work, the threshold provided by CMOS transistors might get the required results.

Figure 5.6 shows the effect of short term potentiation. The frequency of the spikes gradually reduces over time for the same signal as shown in Figure 5.7. This memory is removed, when the input signal changes. This can be very useful for creating sensors, which will stop reacting to environmental noise. However, this feature will not be a part of this thesis work as it increases complexity. This feature will be pursued in future research work.

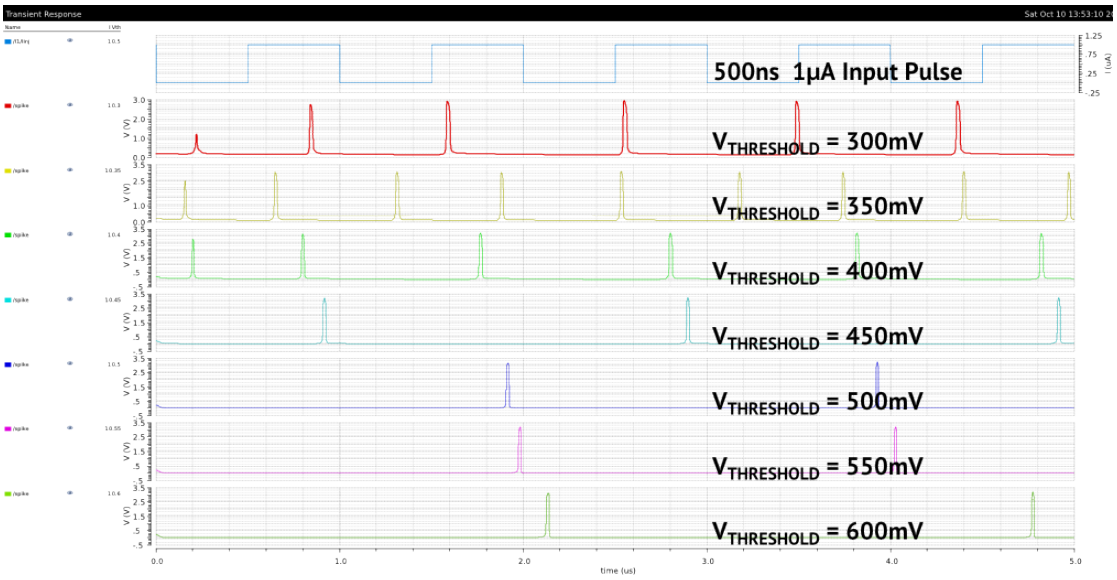


FIGURE 5.5: Simulation of Changes to Spikes with Change in Voltage Controlling the Threshold

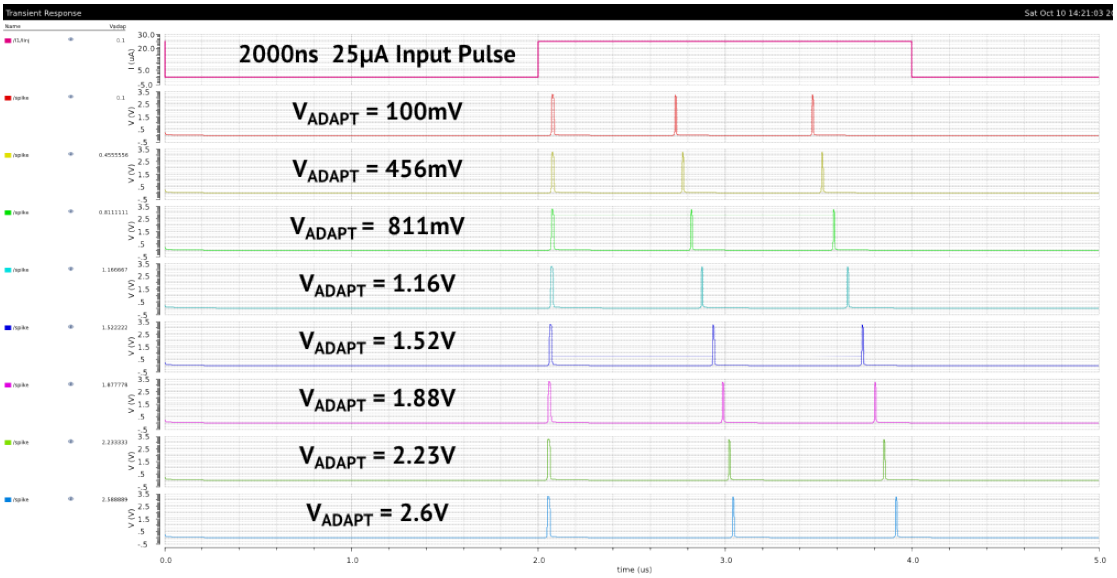


FIGURE 5.6: Simulation of Spike Frequency Adaptation

5.1.1 Discussion

In this section, the properties of analog LIF neuron shortlisted from the previous chapter have been shown. While the neuron seems to be extremely flexible, the focus of this design was to imitate a biological neuron [4]. This results in slower operating speed. Since this work is not bound to the limitations of biological nervous system, but the limits of the technology, this neuron has to be simplified and made to perform as fast as the technology allows. In the first step a simple LIF neuron without biological adaptation methods would be useful to create the foundation for the planned SSDC.

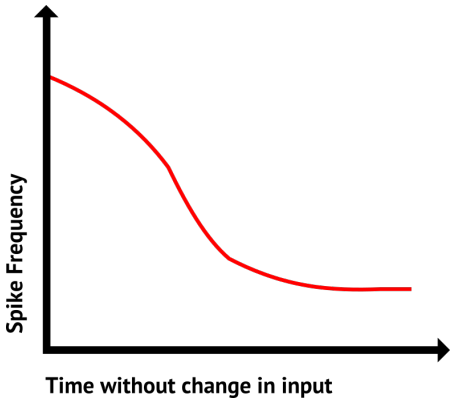


FIGURE 5.7: Spike Frequency Adaptation Curve (showing the change in spike frequency over time, if there is no change to the inputs. This "memory" is lost, when there is a change to the input values.)

5.2 Simplified Leaky Integrate and Fire Neuron

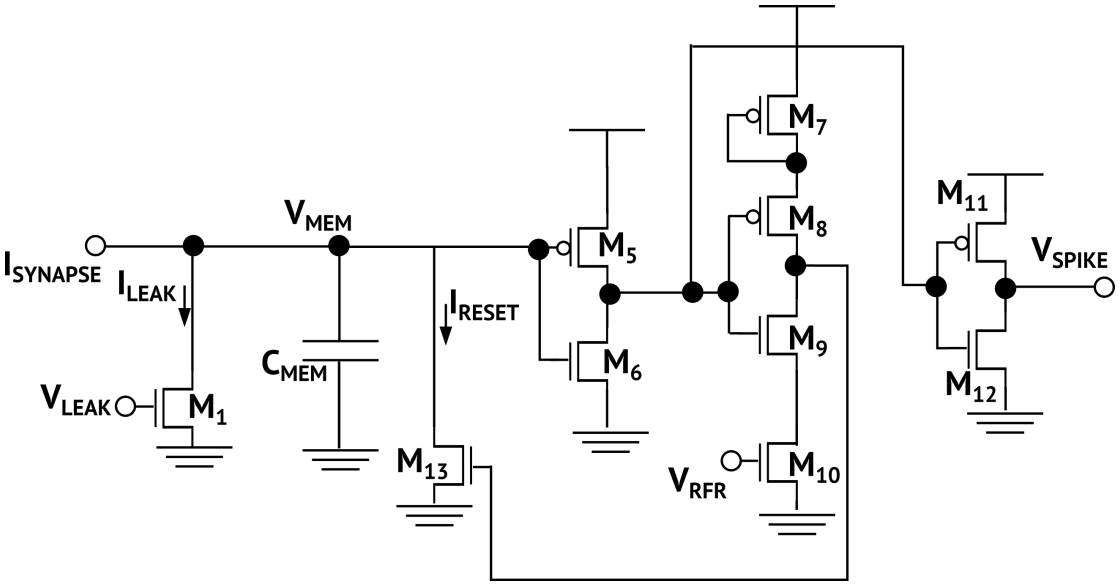


FIGURE 5.8: Leaky Integrate and Fire Neuron (Simplified from Indiveri's neuron)

In Figure 5.8, a simplified LIF neuron is presented, which has been deprived of many components unnecessary for the target of this work. Another goal is to make the neuron reach the speed limits of the technology. The modified structure has a membrane capacitor with M_1 leaking a fixed current depending on V_{LEAK} voltage. As the membrane capacitance increases above the crossing point of the first inverter, the inverter output changes. The output of this inverter is sent to two components. One is an inverter constituting of transistors M_{11} and M_{12}, which produces the output value. The other component is a current starved inverter, which discharges the membrane potential V_{MEM} by controlling the transistor M_{13} to reduce the membrane Voltage and effectively producing the spike. Depending on the Voltage V_{RFR}, this transistor keeps discharging

the membrane potential for a proportional period of time. The initial experiments shown here will use the same component sizes as the previous neuron to show the properties. The other parameters will also remain the same as Table 5.2 unless specified in the simulation figures present in this section. These values will then be optimized depending on the specific application (for, e.g., Coincidence Detection requires a membrane capacitance C_{MEM} of negligible size, while SSC requires a much larger C_{MEM}).

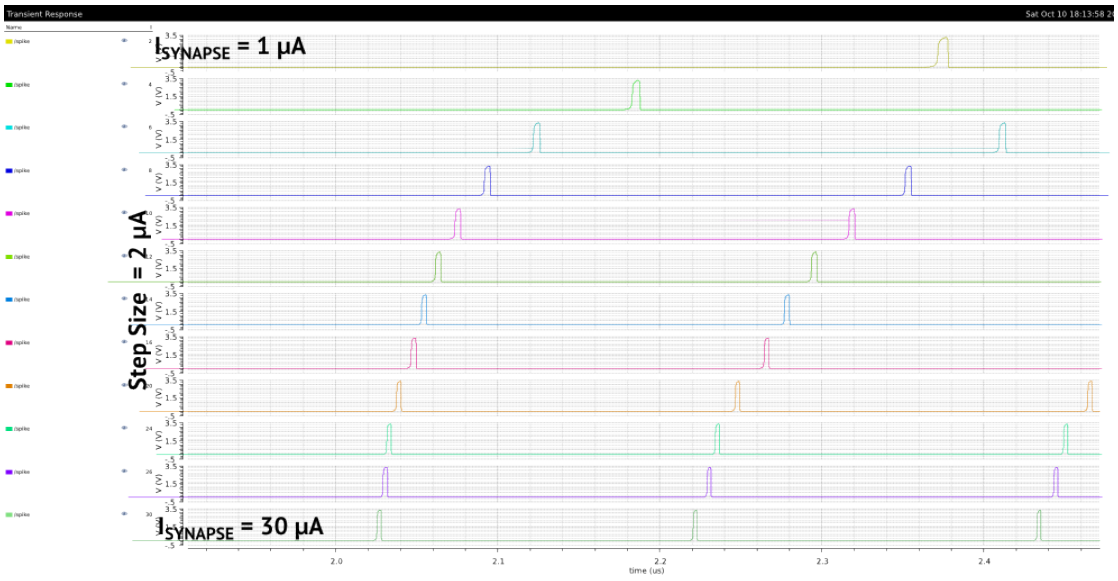


FIGURE 5.9: Simulation of Spikes for Different Input Currents.

Figure 5.9 shows the spikes for $I_{SYNAPSE}$ currents in the range mentioned. The non-linearity of the spike response is the result of leakage by the transistor M_1.

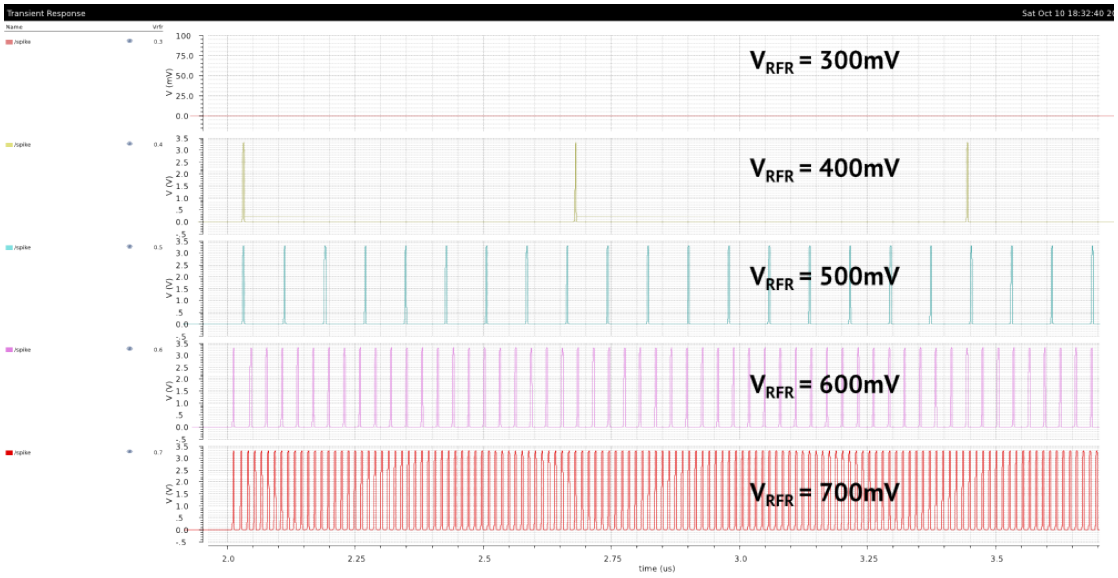


FIGURE 5.10: Simulation of the Effect of V_{RFR} on Spikes in the Simplified Integrate and Fire neuron

Figure 5.10 shows the change in the refractory period of the neuron, depending on the voltage V_{RFR}. The most effective span is between 400 and 600 mV.

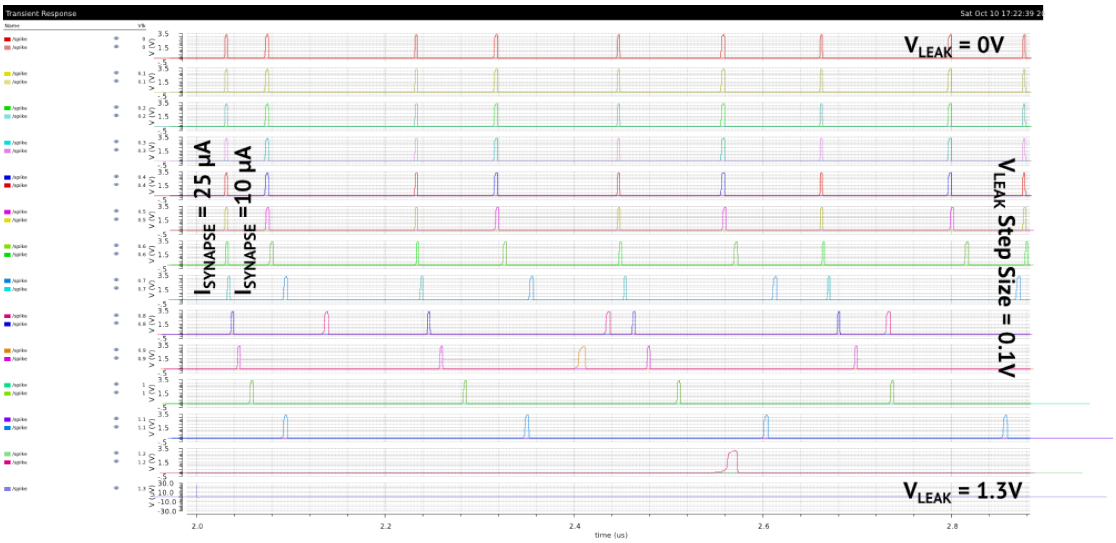


FIGURE 5.11: Effect of V_{leak} Voltage on Simplified Integrate and Fire Neuron

Figure 5.11 shows the simulation of V_{LEAK} voltages for two different current inputs.

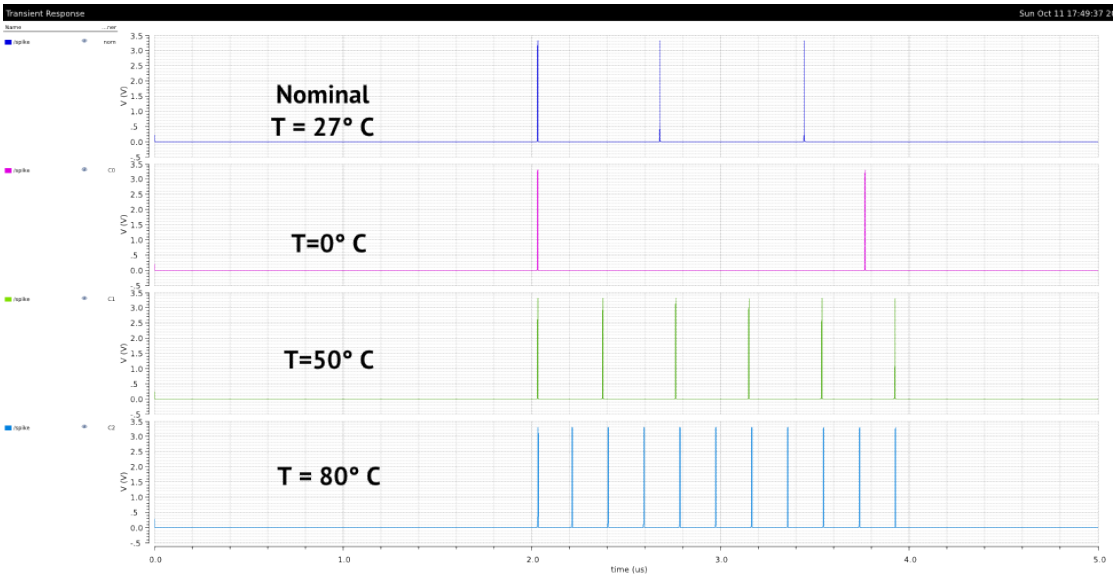


FIGURE 5.12: Effect of Temperature on the Neuron

Figure 5.12 shows the effect of temperature on the neuron. The transistor discharging the membrane potential M_{13} and the refractory period control circuit seem to be the most sensitive part of this design. This can be observed by the fact that the time to first spike did not vary at different temperatures. As discussed before, since we focus on the time to first spike rather than the frequency of spiking, the effect of temperature seemed to be negligible for this circuit. However, this is only true if the current feeding the circuit is ideal. Its behavior will be different, when it is fed by the non-ideal synapse

circuit. It can be concluded that the neuron is behaving as required and the focus should move to the synapse to provide close to ideal inputs.

5.2.1 Discussion

This simplified neuron is capable of the tasks required by our proposed architecture. It is also much faster than the original circuit. The sizing of the capacitors in the circuit can be further modified as required by the application at hand (for, e.g., Coincidence Detection requires a membrane capacitance C_{MEM} of negligible size, while SSC requires a much larger C_{MEM}).

5.3 Differential Pair Integrator Synapse

While the neuron model provides the required behavior, it performs only the task of the neuron body. The current fed to the neuron should be generated ideally by one or more synapses. Each neuron can be connected to several other neurons, and also have inputs from several other neurons. The synapses make these connections with the weights, thresholds, and time constants required for each input value. We have discussed various synapses in the previous chapter.

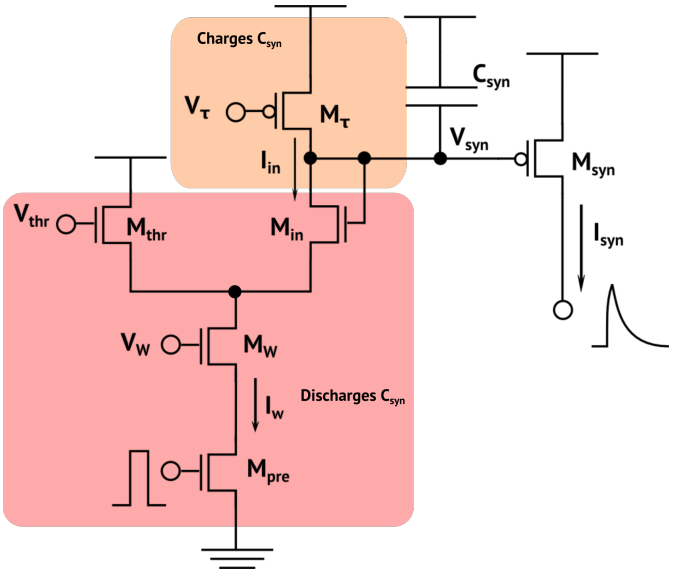


FIGURE 5.13: Differential Pair Integrator Synapse

Figure 5.13 shows the Differential Pair Integrator synapse. The working of this synapse is based on the capacitor C_{syn} and the PMOS transistor M_{syn}. The capacitor C_{syn} is charged by the transistor M_τ. The charging speed depends on V_τ. The other transistors

play a role in discharge the synapse current. When a pulse arrives, the transistor M_{pre} starts discharging the synapse current depending on the voltages V_w and V_{thr}. Depending on the size of C_{syn} and V_τ there is a slowly reducing current at the output. This replicates the required neuron dynamics.

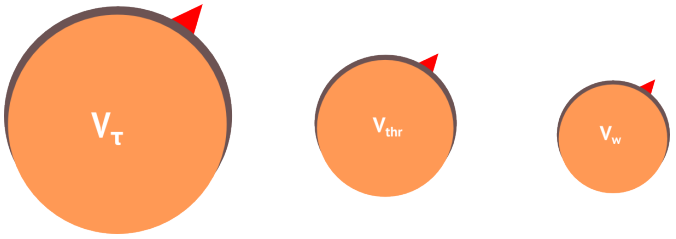


FIGURE 5.14: Controlling Capabilities of each Voltage Exemplified by the Radius of the Knob. (It visualizes the effect of change in voltage to the change in $I_{SYNAPSE}$)

From the point of view of analog design, Figure 5.14 shows the effectiveness of different voltages in control of the output current. These have the additional capability of correcting for process, manufacturing, and temperature related deviations. The effect of these changes can be seen in the simulations below.

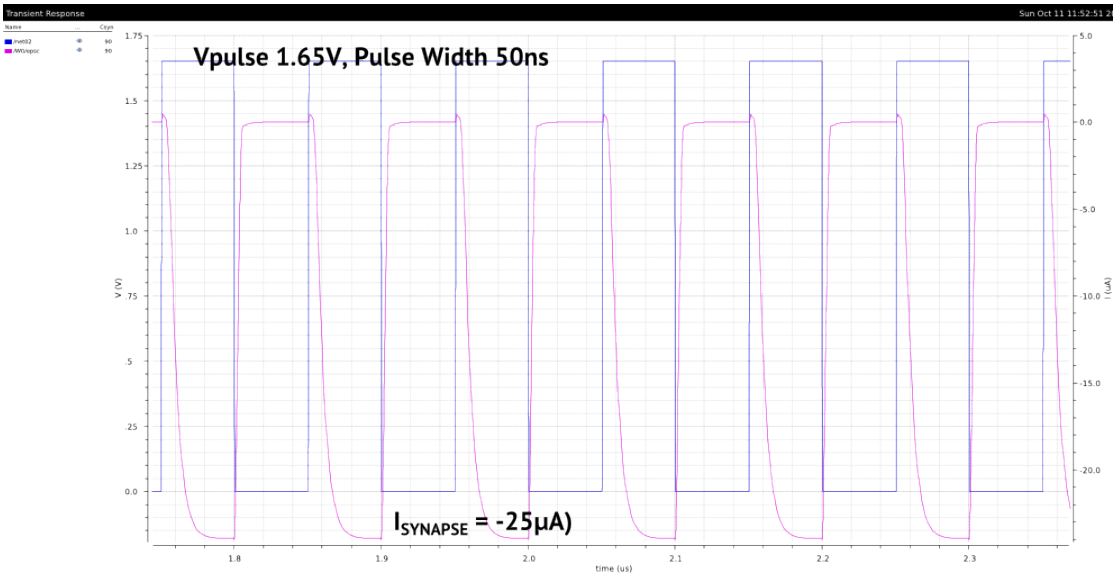


FIGURE 5.15: Output Current $I_{SYNAPSE}$ from the DPI Synapse

In Figure 5.15 and Figure 5.16, the effect of the pulse width and period (normally there is only one input pulse in our application), for the configured parameters, on the output current $I_{SYNAPSE}$ can be observed. The current is negative because the direction of the current measured is moving out of the transistor M_{syn}.

Figure 5.17 shows the control of time constant of the neuron by changing the voltage V_τ. When V_τ is high (3.3 V), the synapse converts the neuron into a simple delay element and looses all the characteristics of a neuron.

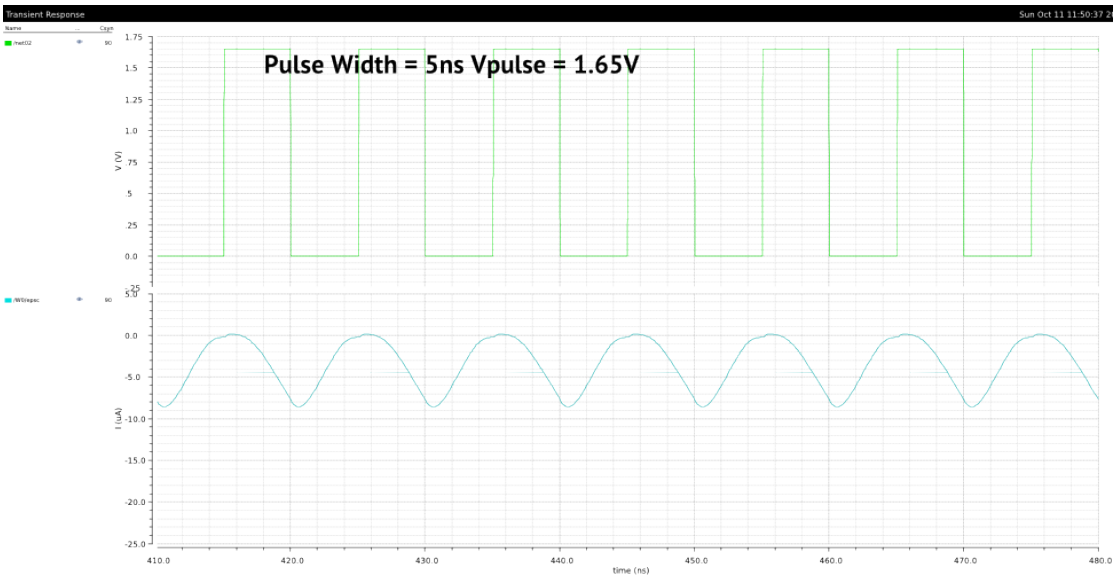


FIGURE 5.16: Output Current $I_{SYNAPSE}$ from the DPI Synapse for 5ns Pulse

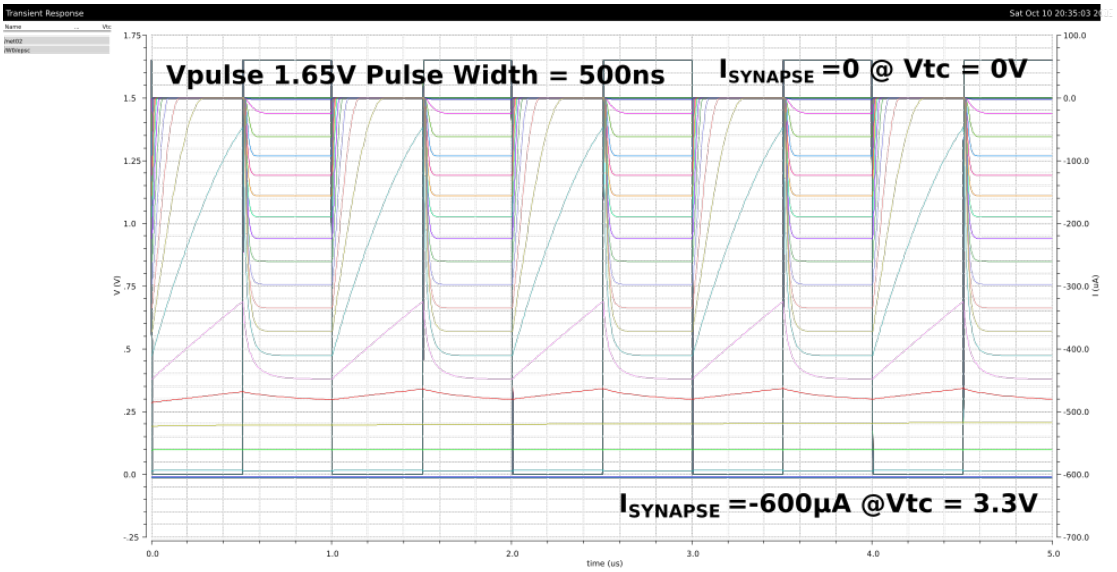


FIGURE 5.17: Effect of Changes to V_τ on DPI Synapse

Figure 5.18 shows the effect of the threshold voltage. The linearity of the change in output current to the change in threshold can be observed. This is different from Figure 5.19, where the changes in output current with respect to the changes in V_w is not linear. This observation is very useful, when adding adaptation to these neural circuits.

Figure 5.20 and Figure 5.21, show that the change in output current with respect to the change in input pulse is linear. This is very important for replication of coincidence detection properties of the neuron.

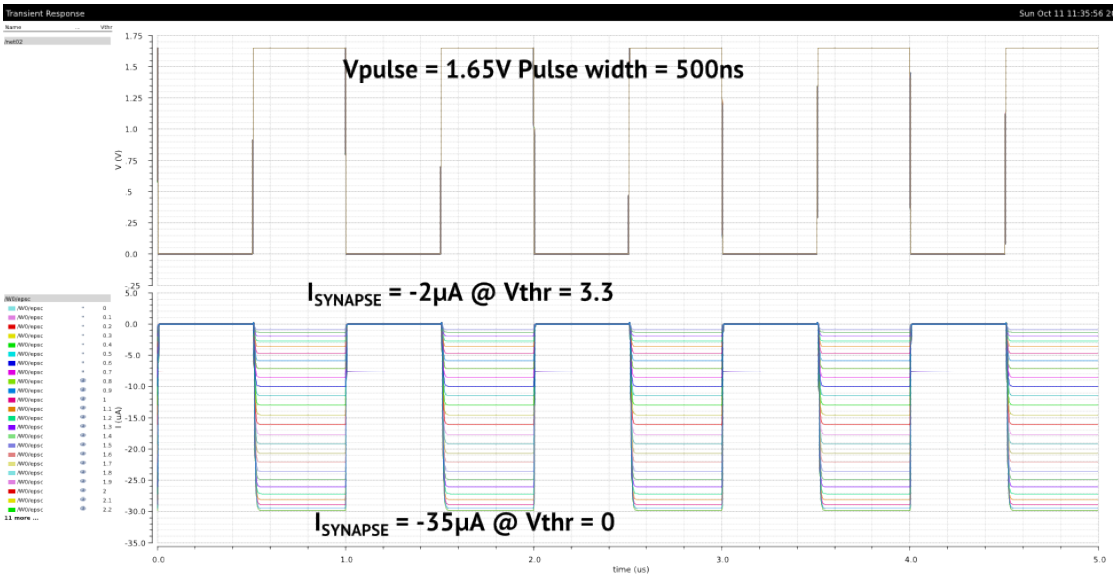


FIGURE 5.18: Effect of Changes to Threshold (V_{thr}) to Current Output of DPI Synapse

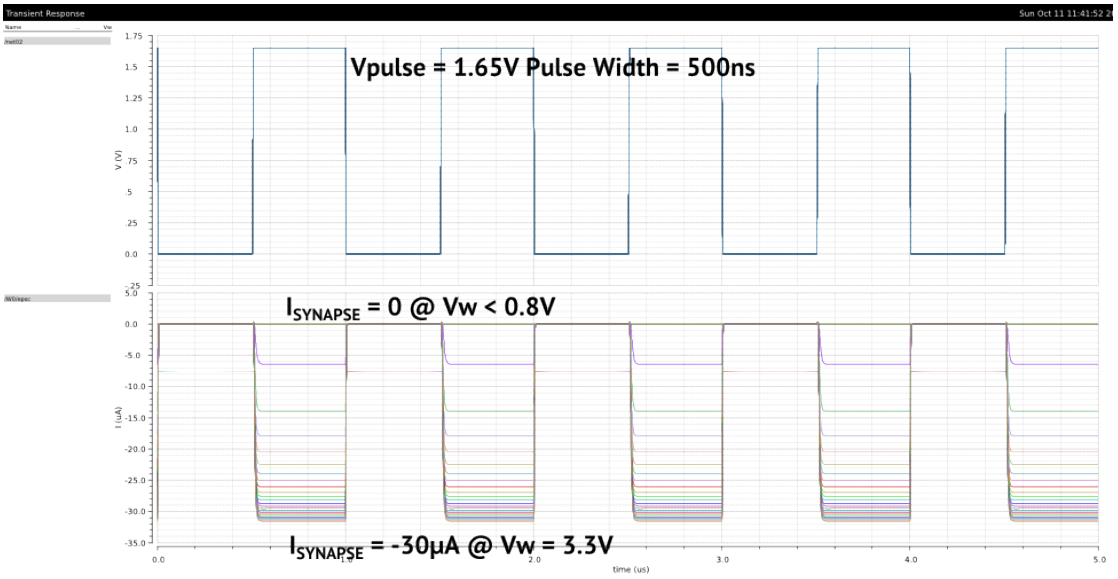


FIGURE 5.19: Effect of Changes to Weight (V_w) to Current Output of DPI Synapse

5.3.1 Discussion

This section showed the structure of the synapse, which has 3 different control voltages. These control voltages can be used to modify the behaviors of the neuron or for adaptation of the analog structure to deviations due to process, temperature, and aging.

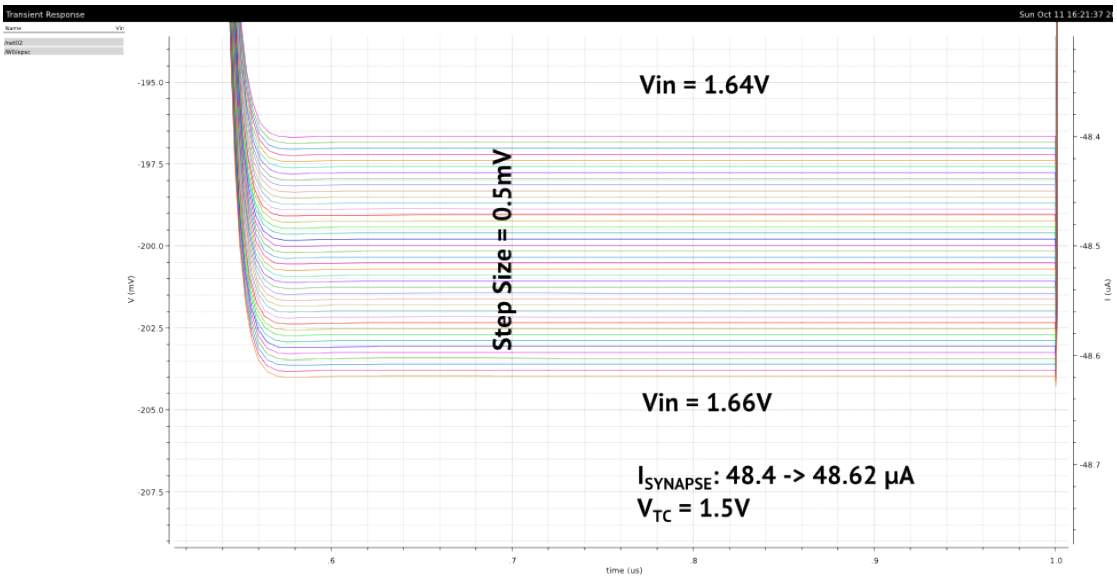


FIGURE 5.20: Current Output $I_{SYNAPSE}$ Change for Different Vin Values at V_τ of 1.5 V

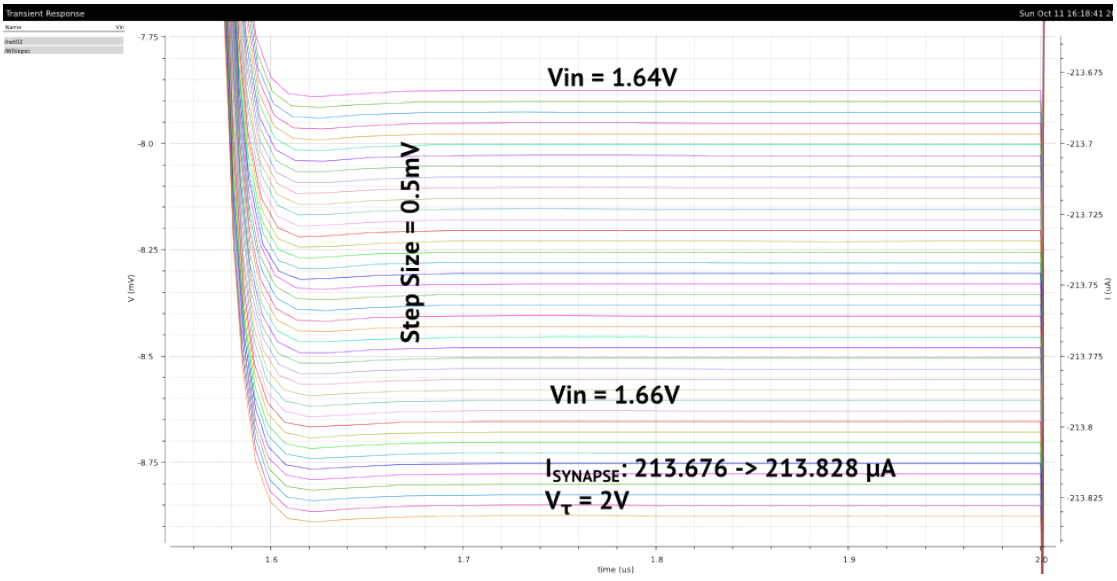


FIGURE 5.21: Current Output $I_{SYNAPSE}$ Change for Different Vin values at V_τ of 2.0 V

5.4 Implementation Methodology

One of the advantages of the biological approach is the presence of only a few building blocks, in our case only the neuron and the synapse. This provides an incentive for complete checking and design of the basic building blocks. The first steps of the design is to find the "hot spots" in the circuit and size them to more robust values. This first order checking of the sizing provides a good baseline for the circuit. This structure is then checked using Monte-Carlo analysis described below.

In this example, the DPI synapse module is chosen since (1) it will be the most used module in the final design, (2) it has fewer elements so it is easier to explain the design methodology, (3) it is the most sensitive part of the design. The Table 5.3 shows the sizing of the DPI synapse for the coincidence detection module described in the next chapter. The size of C_{syn} of the working of coincidence detection was found to range from 74 pF to 108 pF, so 90 pF has been chosen as the final design value. The sizing of the transistors was made to be the minimum working for the technology, and then each transistor size was increased to check if there were lesser deviations in the important characteristics in Monte-Carlo analysis of 100 runs. Coincidence Detection has been described in Section 4.1.1. The most important aspect of this module is the rate of discharge as this is essential in maintaining the width of the Gaussian kernel. The amplitude of the current $I_{SYNAPSE}$ plays a comparatively minor role. Here, we will look at glimpses of the search for optimum values to show the process behind the implementation methodology, which has been used for all the modules in this work.

Components	Sizing (w/l) µm/µm	Components	Sizing (w/l) µm/µm
M_{thr}	1/3	M_W	6/3
M_{pre}	3/1	M_{dif}	1/3
M_τ	4/10	M_{syn}	2/11
C_{syn}	90 pF		

TABLE 5.3: DPI Synapse Sizing

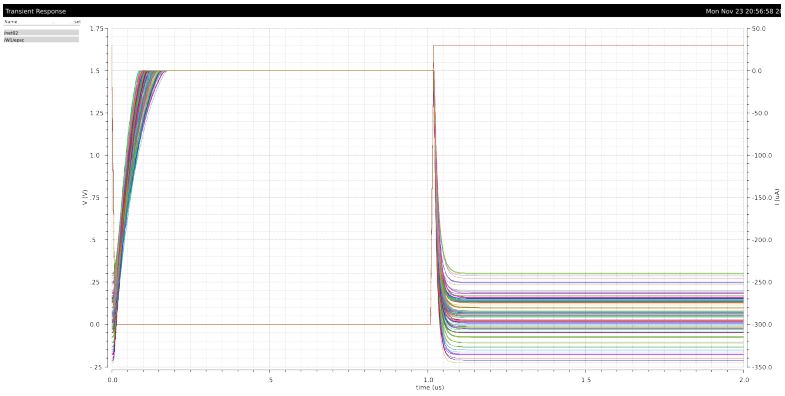


FIGURE 5.22: Monte-Carlo Analysis of 100 Runs (with all sizings as per Table 5.3)

Figure 5.22 shows the results of Monte-Carlo analysis of 100 runs for the values in Table 5.3. In Figure 5.23 the values of the differential pair of the DPI synapse, the transistors M_{thr}, M_{dif} have been changed to another working value. We can observe, that there is a higher deviation in the rate of discharge of the module. This effect is repeated, when we change the sizing of the other transistors M_W and M_{syn} to different degrees. By repetition of this approach for different working sizings of the transistors,

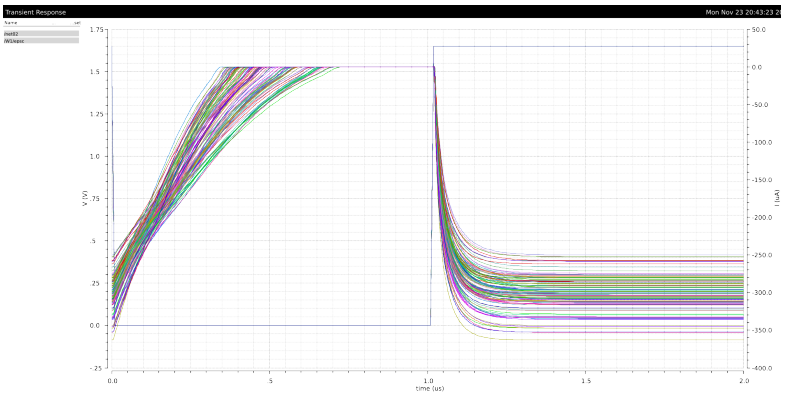


FIGURE 5.23: Monte-Carlo Analysis of 100 Runs (with all sizings according to Table 5.3 except M_{dif} and M_{thr} transistors, which have size w/l 3/9 μm/μm)

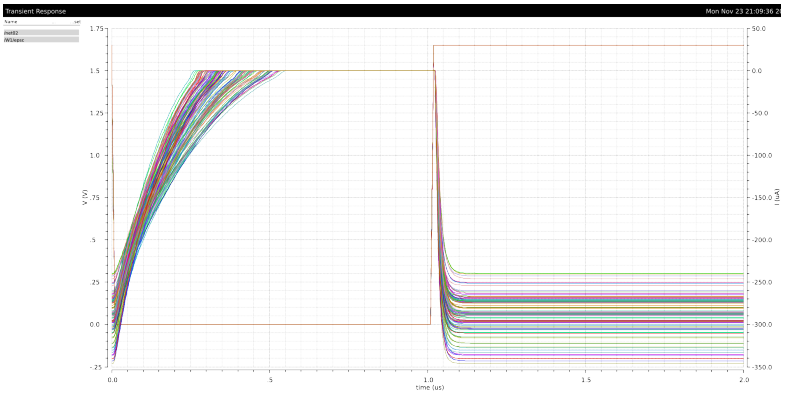


FIGURE 5.24: Monte-Carlo Analysis of 100 Runs (with all sizings according to Table 5.3 except M_w transistor with size w/l 9/4.5 μm/μm)

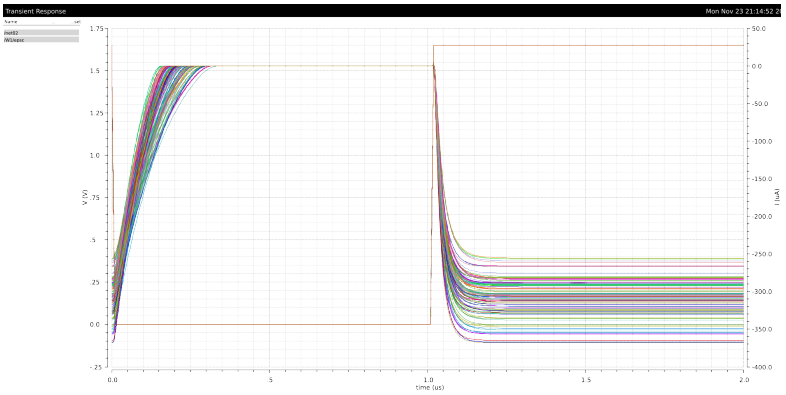


FIGURE 5.25: Monte-Carlo Analysis of 100 Runs (with all sizes according to Table 5.3 except M_{syn} transistor with size w/l 11/2 μm/μm)

we are able to find the sizes, which (1) work as intended (2) Minimum deviations in the most important properties required by the module. The Monte-Carlo analysis of 100 runs is simulated for process, mismatch and temperature settings.

5.5 Summary

This chapter provides the overview for the basic structures, which will be used in the following chapters. The simplified neuron spikes, when the membrane voltage crosses the crossing point of the inverter. The DPI synapse is able to provide the required current proportional to the input pulse. We have voltages V_{LEAK}, V_{RFR} as controlling voltages in the neuron body, V_{τ}, V_{thr} and V_{w} as controlling voltages in the synapse. These control voltages are easily able to move the neuron into working region after changes due to process, temperature, and aging. The previous section describes the implementation strategy used for the design of cells used in this work. The advantage of this approach is that it can be easily optimized automatically by synthesis tools like ABSYNTH [108], which is an analog block synthesis tool developed at ISE.

Chapter 6

SDC - Spike to Digital Converter using Coincidence Detection

The operating principle of coincidence detection has been described in chapter 4.1.1. This can now be implemented with the building blocks from the previous chapter. The resulting structure is shown in the Figure 6.1. It has two DPI synapses for the two inputs and their outputs are connected to the input of the neuron body.

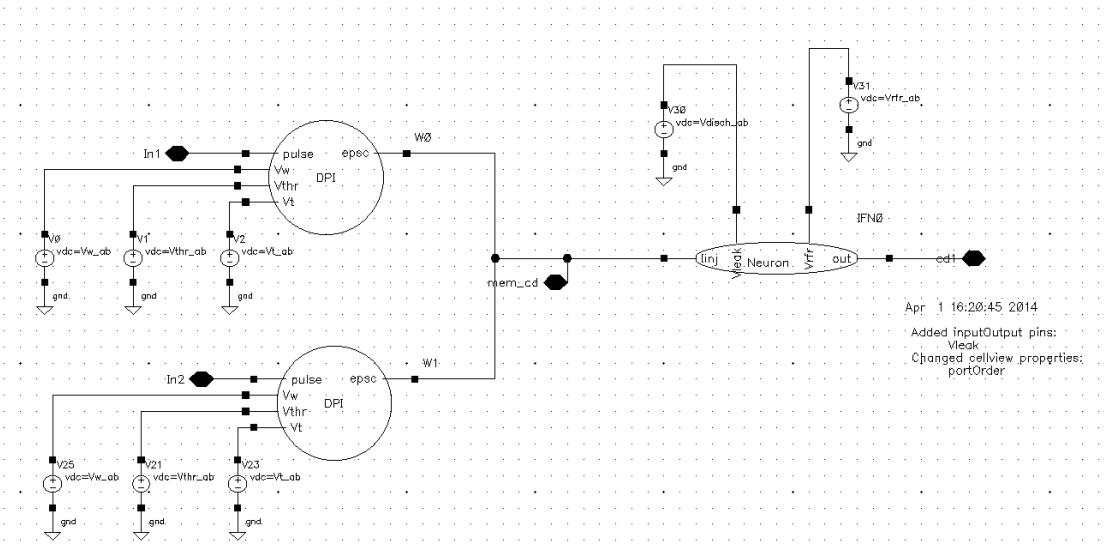


FIGURE 6.1: Coincidence Detection Setup

Figure 6.2 shows the test bench with the input pulses. The transistor present is a reset transistor used to drain the membrane capacitance after every run. The structure of neurons and synapses with reset transistors will be shown later in this chapter.

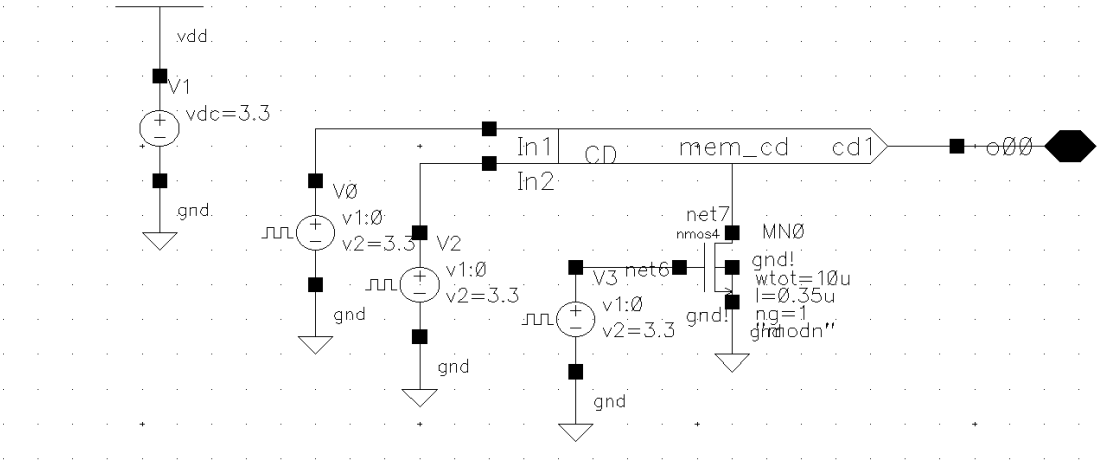


FIGURE 6.2: Coincidence Detection Test Bench

6.1 Sensitivity of Coincidence Detection

The sensitivity of coincidence detection depends on the width of the temporal Gaussian kernel that can be created with the building blocks. Since the time constant of the neuron plays a major role in the creation of the kernel, V_τ is the main controlling voltage in this scenario. Table 6.1 shows the different values of Gaussian kernel widths (CD_{RANGE}) that are possible with the configurations. At higher V_τ the Gaussian kernel becomes rectangular or flat. Rectangular kernels can be useful for robust design, however, we loose the advantage of place coding.

Properties	Config. 1	Config. 2	Config. 3	Config. 4
$C_{SYNAPSE}$	90 fF	90 fF	90 fF	90 fF
V_{LEAK}	1.6 V	1.6 V	1.6 V	1.6 V
V_{RFR}	0.4 V	0.4 V	0.4 V	0.4 V
V_τ	1.92 V	2.15 V	2.1 V	1.92 V
$V_{THRESHOLD}$	1.65 V	1.65 V	1.65 V	1.65 V
V_W	1.5	1.5	1.5	1.5
C_{MEM}	100 fF	20 fF	20 fF	20 fF
CD_{RANGE}	4 ns	50 ns	25 ns	4 ns
Pulse Width	5 ns	5 ns	5 ns	5 ns

TABLE 6.1: Configuration Table

Figure 6.3 shows how the place coding looks, if input pulses arrive at the same time, under the assumption that one branch of the input is delayed by 5 ns compared to the previous coincidence detector. As the difference between the input pulses increases the

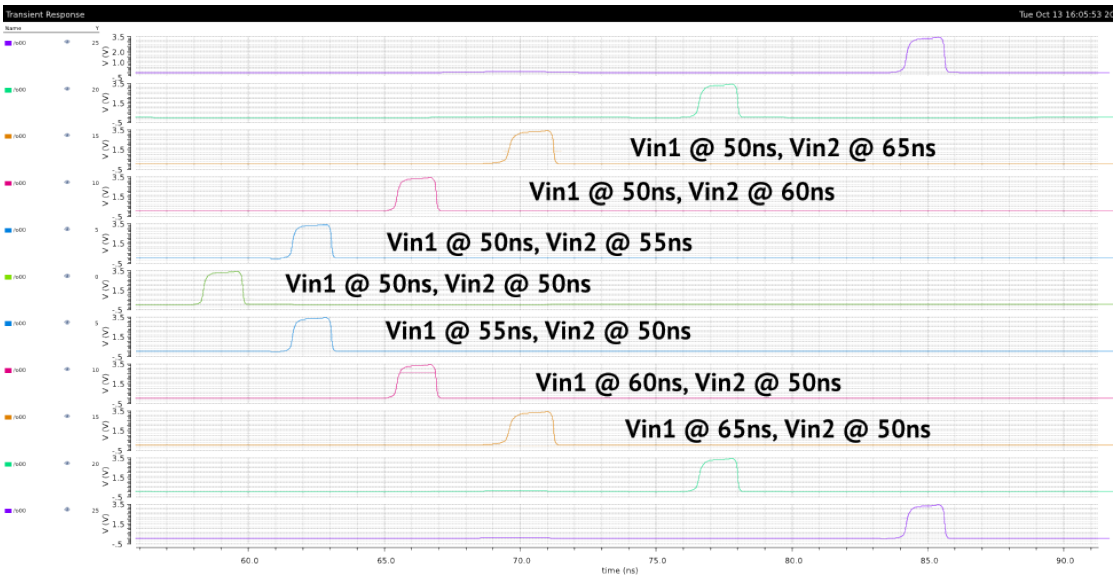


FIGURE 6.3: Simulation showing Coincidence Detection with Configuration 2

delay makes the input pulses at the other branch take priority since they will appear at the same time at the next branch. It is possible to modify the width of the Gaussian kernel using V_{tc}. This is shown in the following figures from Fig. 6.4 to 6.7

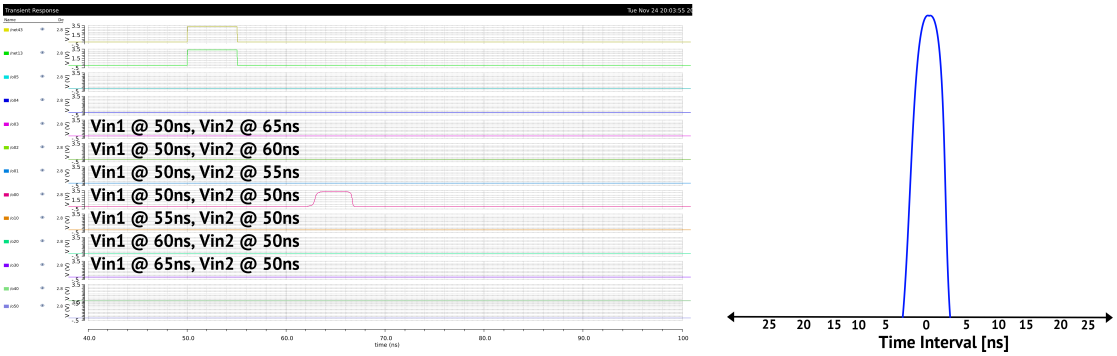


FIGURE 6.4: Simulation showing Coincidence Detection (with minimum Gaussian kernel width) (left), A Sketch of the corresponding Gaussian Kernel (right)

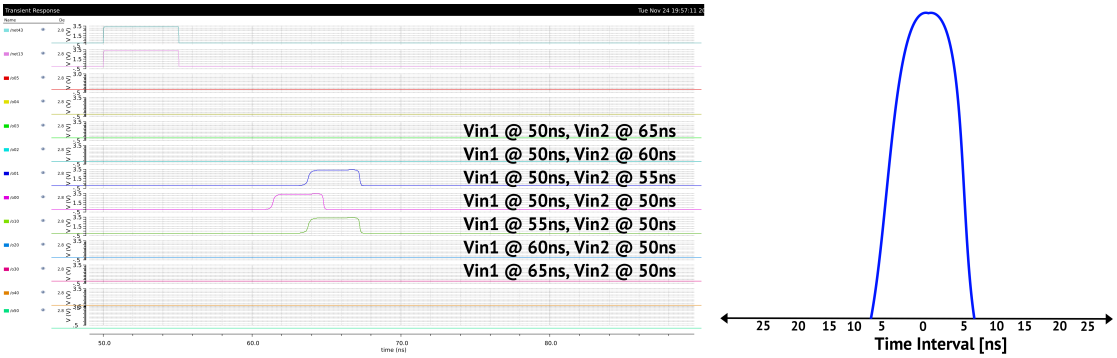


FIGURE 6.5: Simulation showing Coincidence Detection (with small Gaussian kernel width) (left), A Sketch of the corresponding Gaussian Kernel (right)

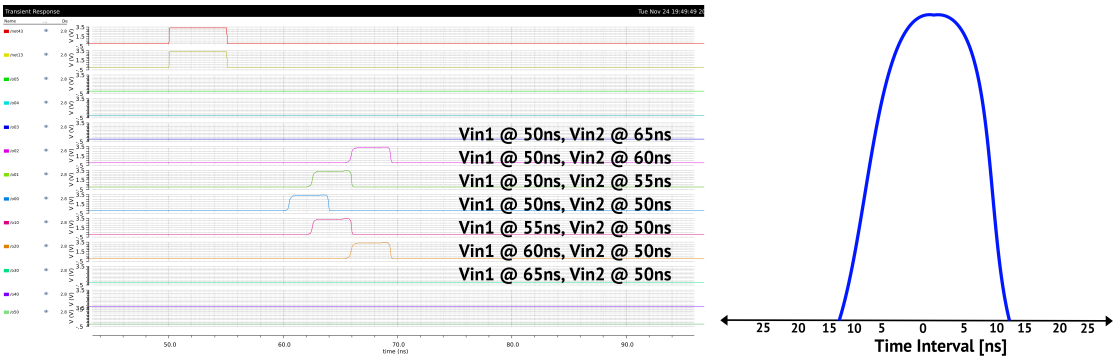


FIGURE 6.6: Simulation showing Coincidence Detection (with medium Gaussian kernel width (left)), A Sketch of the corresponding Gaussian Kernel (right)

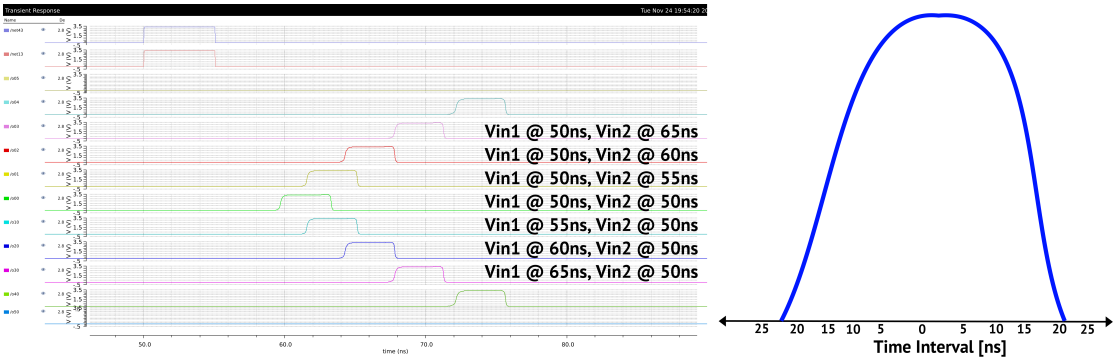


FIGURE 6.7: Simulation showing Coincidence Detection (with large Gaussian kernel width) (left), A Sketch of the corresponding Gaussian Kernel (right)

6.2 Place Coding

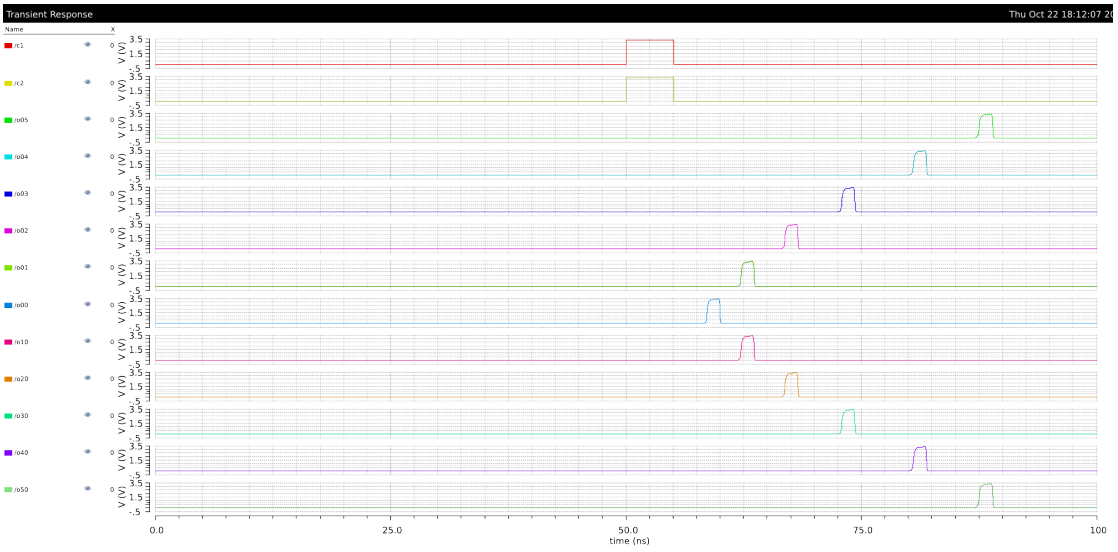


FIGURE 6.8: Place Coding - pulses for time difference of 0 ns of incoming pulses

Figure 6.8 to Figure 6.14 show the simulations of Place Coding for time differences 0 ns to 30 ns. The understanding of Place Coding is not as intuitive as simple coincidence detection with the width of the Gaussian close to 0. In simple coincidence detection,

FIGURE 6.9: Place Coding - pulses for time difference of 5 ns of incoming pulses

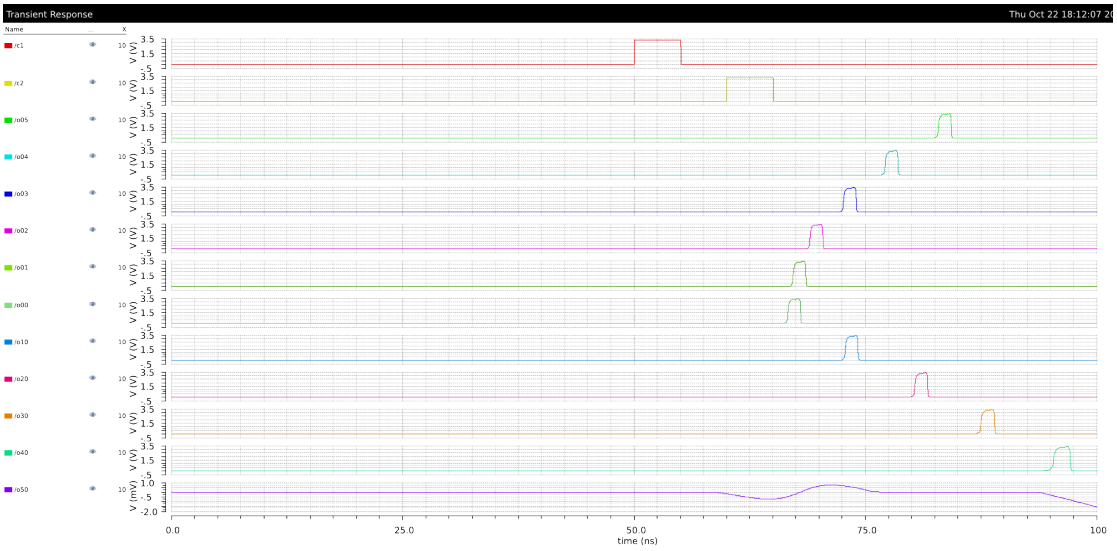


FIGURE 6.10: Place Coding - pulses for time difference of 10 ns of incoming pulses

there is one spike exactly as the delay is programmed similar to a flip flop. However, in place coding there is a time shift, that occurs because some of the detectors are active earlier than the others. This effect reduces the span of the coincidence detection, however, increases the resolution. The effect is enhanced, if the delay between the coincidence detectors is high, and, it is almost negligible at very low delays. Example of place coding enhancing the resolution can be found in earlier works [109], and [110].

FIGURE 6.11: Place Coding - pulses for time difference of 15 ns of incoming pulses

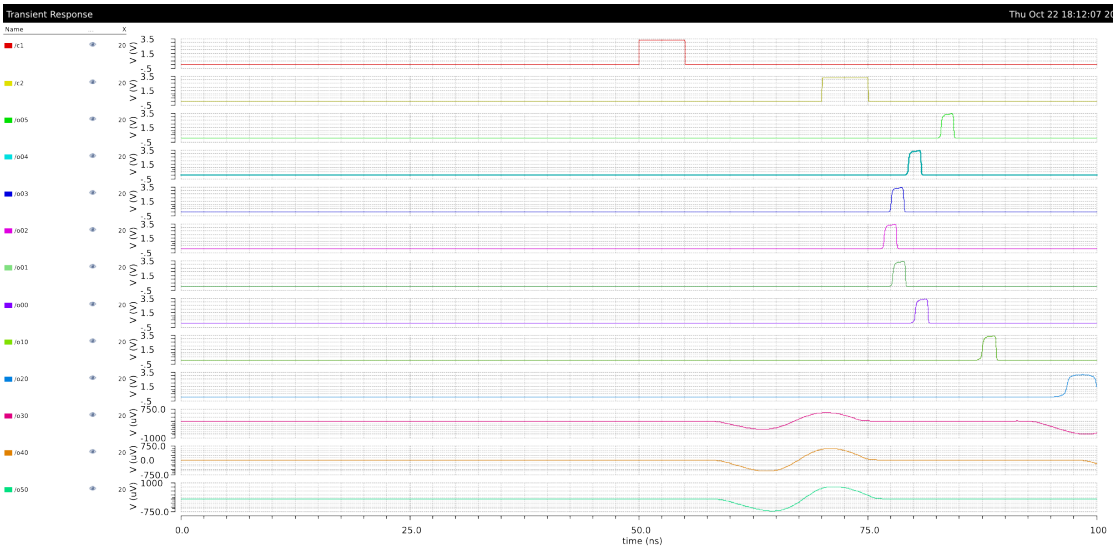


FIGURE 6.12: Place Coding - pulses for time difference of 20 ns of incoming pulses

FIGURE 6.13: Place Coding - pulses for time difference of 25 ns of incoming pulses

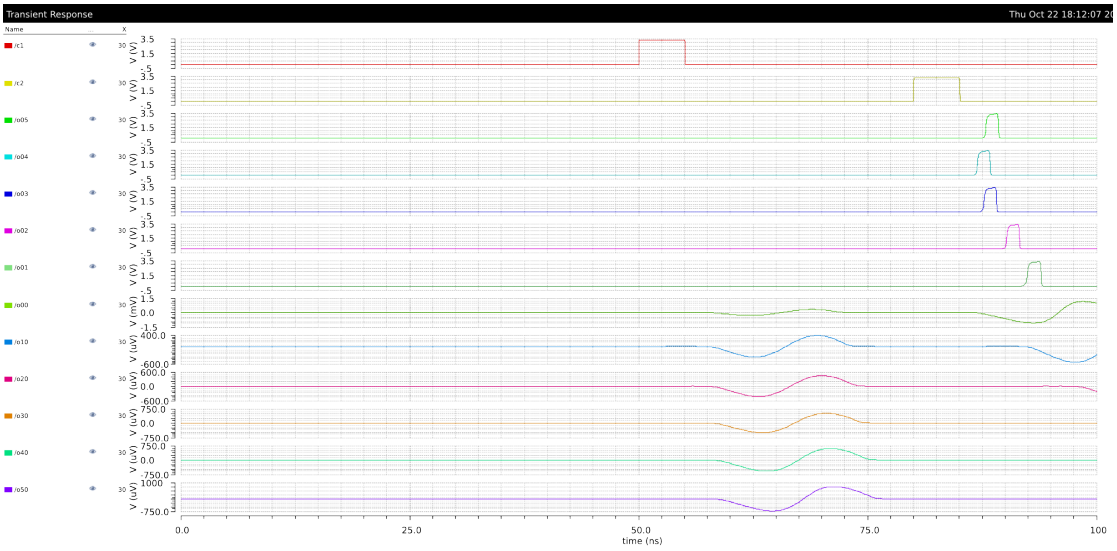


FIGURE 6.14: Place Coding - pulses for time difference of 30 ns of incoming pulses

6.3 Delay Circuits

There are quite a few different methods for creating delays from traditional RC circuits to buffer chains. Circuits of two different delay structures are shown in the Figure 6.15, and Figure 6.16. The third structure is a simple buffer chain, DEL42 cell taken from CORELIBD library of ams hitkit 4.10. A comparison between different delay structures is shown in the Figure 6.17, Figure 6.18, and Figure 6.19. From the first comparisons, we can see that buffer chain is the most robust among these choices, because,

- Current starved inverter has changes in both delay and pulse-width,
- Neural delay varies in delay, pulse-width, and amplitude, while,
- Buffer delay only has small changes to its delay.

The advantage of current starved inverter is the possibility to modify the delay with the voltage V_{del}. However, in the first implementation for simplicity, a buffer chain is considered. The other structures have to be studied further in future work.

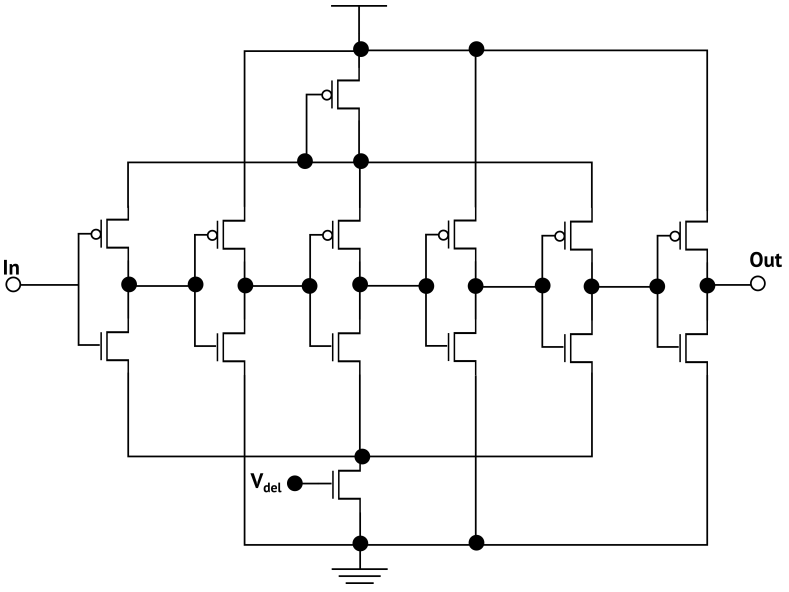


FIGURE 6.15: Current Starved Inverter Chain

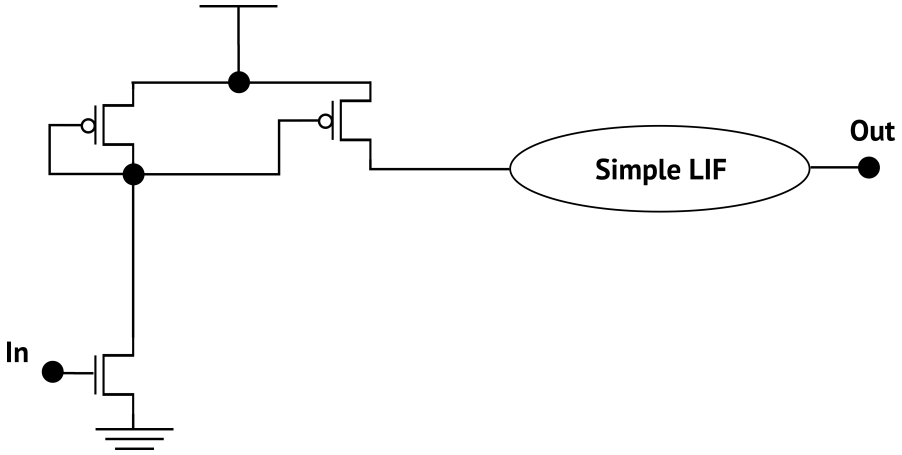


FIGURE 6.16: Neural Delay Element, Simple LIF shown in Figure 5.8

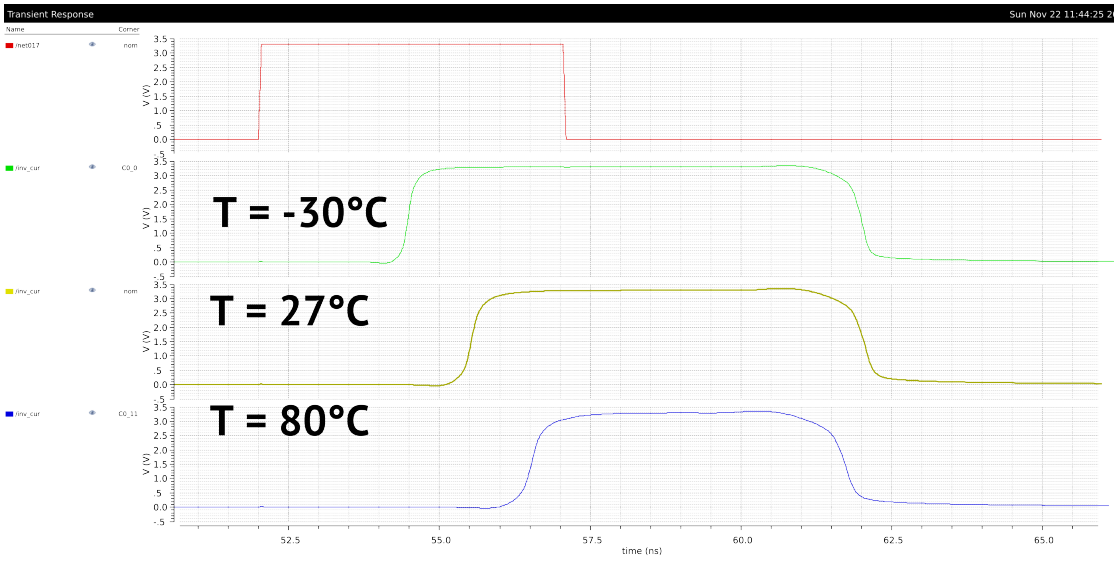


FIGURE 6.17: Current Starved Inverter Simulation with Temperature Corners

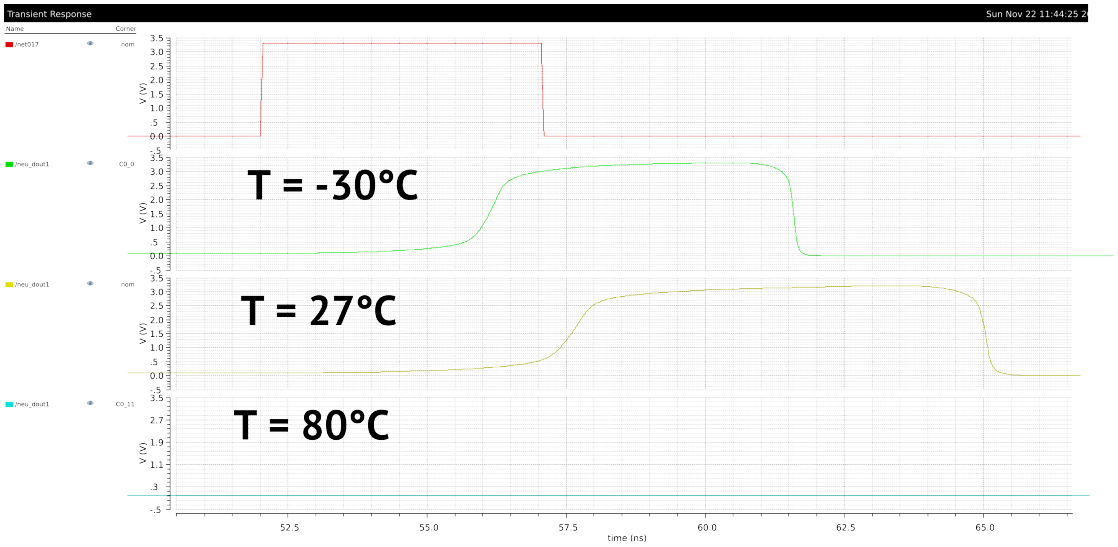


FIGURE 6.18: Neural Delay Circuit Simulation with Temperature Corners

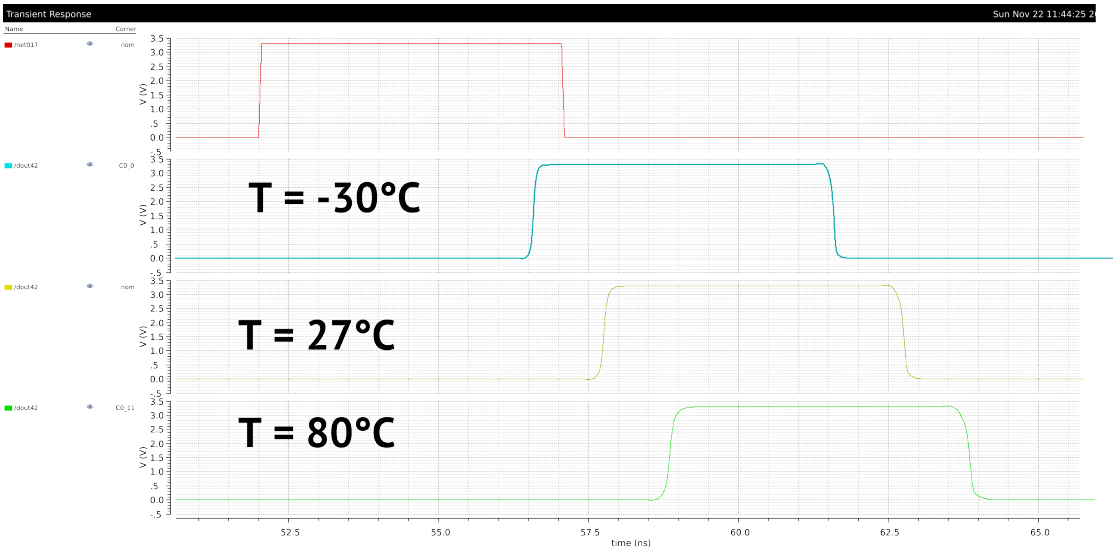


FIGURE 6.19: Buffer Chain Delay Simulation with Temperature Corners

6.4 Sparse Place Coding to Digital

The output of the coincidence detection layer are sparse rank coded spikes. If the number of lines is small, then they can be read out from a simple integrated chip directly. However, this is not the case in this work. To have a viable ADC application, we would need a lot of output lines from coincidence detectors, much more than are viable for a simple chip. While there are techniques like Address Event Encoding, they are designed for speeds of biological neurons and are mostly not viable for identifying the neurons, which spike close to each other in the range of 1-2 ns. While this can be read by making use of a priority encoder, which scans the output lines of the coincidence detectors at high data rates of at least 500 MHz, this approach will consume a lot of power and moves towards clock rates at radio frequency. This creates additional complexity in design, which deviates from the main goal of having a simple fool-proof proof-of-principle-chip.

6.4.1 Time Domain Winner Take All with Memory

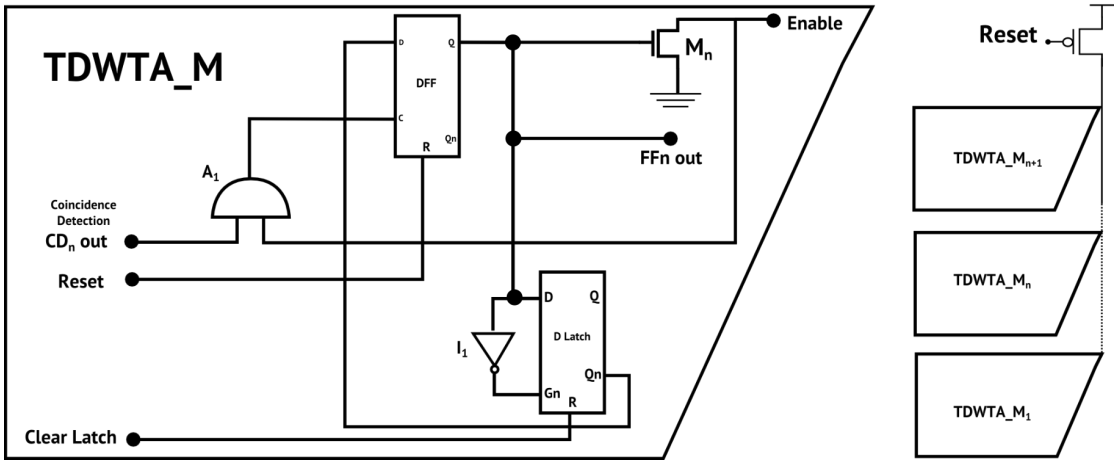


FIGURE 6.20: Time Domain Winner-Take-All with Memory Module (left), Three Module Circuit for Three Input Lines (right)

The Figure 6.20 shows the schematic of the time domain winner take all circuit with memory. This circuit uses a wired OR circuit technique for the enable signal similar to [6]. This signal can be drained to "0" by the NMOS transistor M_1, disabling all the other lines. When there is an input from the coincidence detector and the enable signal is "1", then the negative output of the D Latch passes to the NMOS transistor M_1. In the initial state, the negative output of the D Latch is "1", so "1" passes through the DFF. This brings the enable signal down to zero. The output of the DFF goes to the D Latch, which requires an inverted enable signal. This is also provided by the output of DFF, bringing the latch value to 1. Now regardless of the input values the output of the this DFF will not change until it is reset. Even then the DFF will not switch on until

the latch is cleared. It is capable of identifying spikes occurring with a time difference of 200ps.

The described readout mechanism has been inspired [111] by the implementation of part of the ARAMYS-II chips [6] functionality. This neural chip was conceived to carry out massively parallel distance metric computation and winner(loser)-takes-all determination. In the winner(loser)-takes-all circuit, also wired-or by open-collector (drain) technique can be found. To implement a multiple winner or k-nearest-neighbor computation ability, the ARAMYS-II chip includes an inhibit memory in each neuron, which is reset in the beginning. In case a neuron is the winner of a competition, this inhibit latch can be set, preventing the participation of this neuron in the next competition round. Effectively, one neighbor after the other can thus be sequentially be determined. In case of multiple detected neighbors of the same distance, the order in the array is employed as tie breaking rule. This mechanism has been abstracted and employed for the successive determination of one pulse after the other in repeated read-outs of the sensor, which, of course, makes mild assumptions on the stationarity of the sensory stimulation during the test chip application.

It is possible to unfold this into one larger converter similar to an approach shown in [112] although that approach requires much larger area. The output of this circuit can be read by a set priority encoders to obtain a dense digital code. In this work, for the physical realization, we will use a simple shift register to read out the code. This structure is easier to read, occupies much smaller area, however, it requires repeated sensor read-out, which assumes stationarity of the measurand.

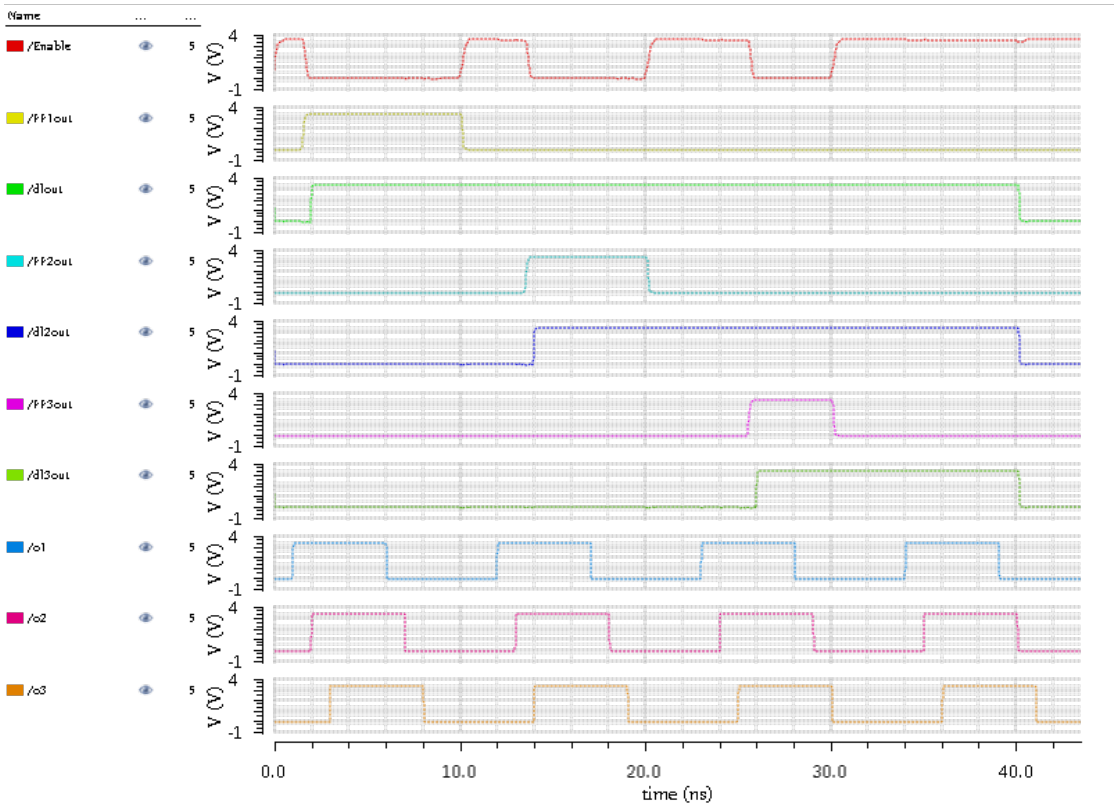


FIGURE 6.21: Simulation of Time Domain WTA with Memory.(This uses three lines, and so three copies of the structures shown in Figure 6.20)

6.5 Physical Design of SDC Coincidence Detector Module

The physical design of one module for the SDC has 4 components, the neuron, synapse, delay, the time domain winner-take-all circuits. The delay and the time domain WTA circuits use components from ams hitkit core libraries and the layouts are taken from these libraries. The most important part is the coincidence detection module. This module requires reset capability. The modified circuits with additional reset transistors are found in Figure 6.22, and Figure 6.23. The final sizes of these components are found in the Table 6.2, and Table 6.3. These values have been found by a search for best coincidence detection properties. It should be noted, that these sizes are not the best sizes, but the most robust sizes, where there is a compromise between area and sensitivity to deviations due to process, temperature, aging etc. have been considered.

The physical design uses matched capacitors for the two synapses, which are protected using Faraday shields (Capacitors placed in n-wells connected to Vdd). The most sensitive path between the synapses and neuron is also designed to be least affected by parasitics. The final layout of the module can be seen in Figure 6.24. The post layout simulations show negligible differences to the schematic simulations.

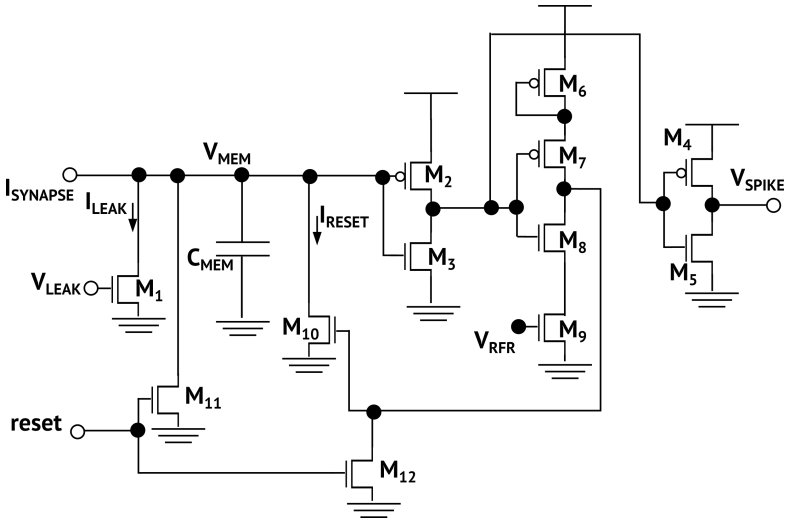


FIGURE 6.22: Simple LIF Neuron with Reset

Components	Sizing (w/l) µm/µm	Components	Sizing (w/l) µm/µm
M_1	1/1	M_2	3/1
M_3	1/1	M_4	3/1
M_5	1/1	M_6	1/3
M_7	3/1	M_8	1/1
M_9	1/1	M_{10}	6/1
M_{11}	1/1	M_{12}	1/1
C_{MEM}	20 pF		

TABLE 6.2: Final sizing of Simple LIF Neuron with Reset

Components	Sizing (w/l) µm/µm	Components	Sizing (w/l) µm/µm
M_{thr}	1/3	M_W	6/3
M_{pre}	3/1	M_{dif}	1/3
M_τ	4/10	M_{syn}	2/11
C_{syn}	90 pF		

TABLE 6.3: Final Sizing of DPI Synapse with Reset

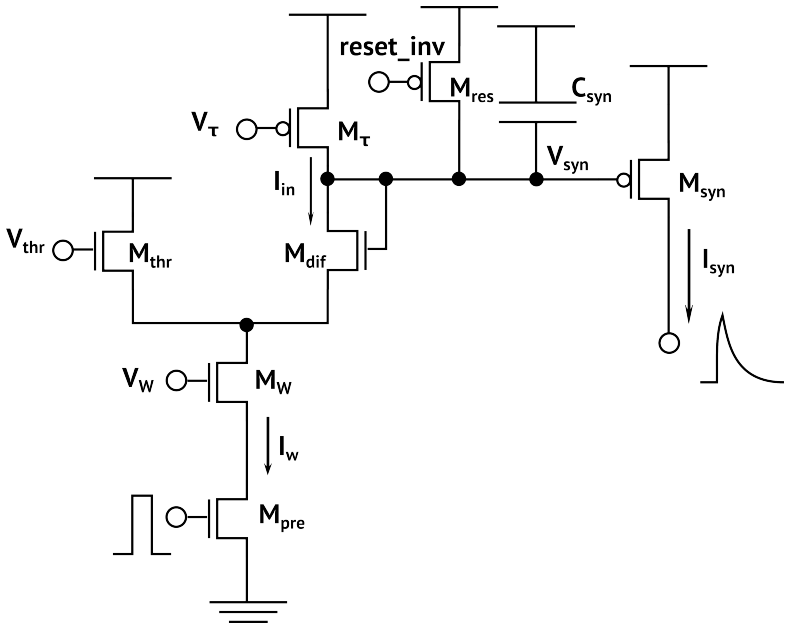


FIGURE 6.23: DPI Synapse with Reset

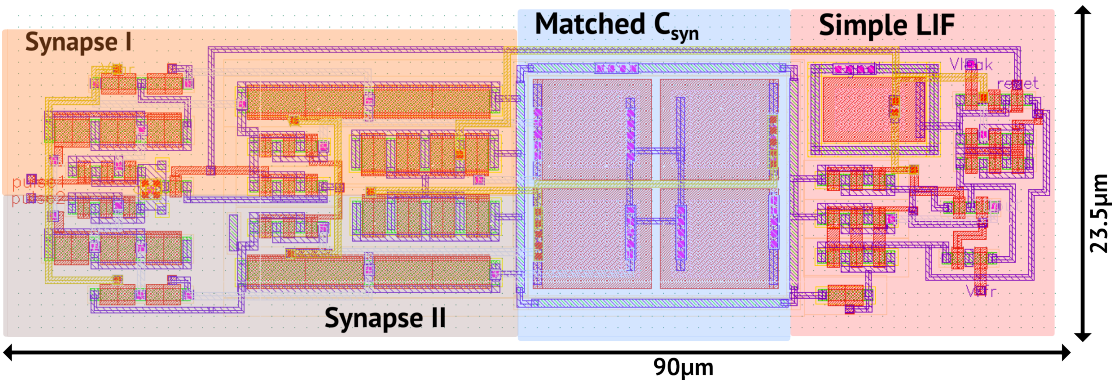


FIGURE 6.24: SDC Coincidence Detection Layout

FIGURE 6.25: Post-Layout Simulation (showing coincidence detection with minimum Gaussian kernel width. It is the counterpart of the Figure 6.4)

FIGURE 6.26: Post-layout Simulation (showing coincidence detection with minimum Gaussian kernel width. It is the counterpart of the Figure 6.6)

6.6 Inherent Robustness of SDC

There are several advantages inherently found in this architecture. The most important one is that the architecture provides an output for pulses occurring at the same time. This is extremely useful for calibration of the SDC. Another advantage of this architecture is graceful degradation. This is shown in Figure 6.27 with the faulty detector coloured red. Detectors that do not work, (stuck at zero, stuck at one, spike at fixed (wrong) time, erroneous spikes that cannot be fixed by changing control voltages of DPI synapse etc.) can be easily identified because of the neighborhood effect of place coding. Every spike's location can be linked to the location of one of its neighborhood spikes. This helps in predicting static and dynamic errors at the coincident detectors. This effect can also be drastically reduced by input reversal described in section 6.6.1

Any spurious spikes at input will not produce any spikes at the output as shown in the Figure 6.28 unless they occur almost simultaneously in both the input lines.

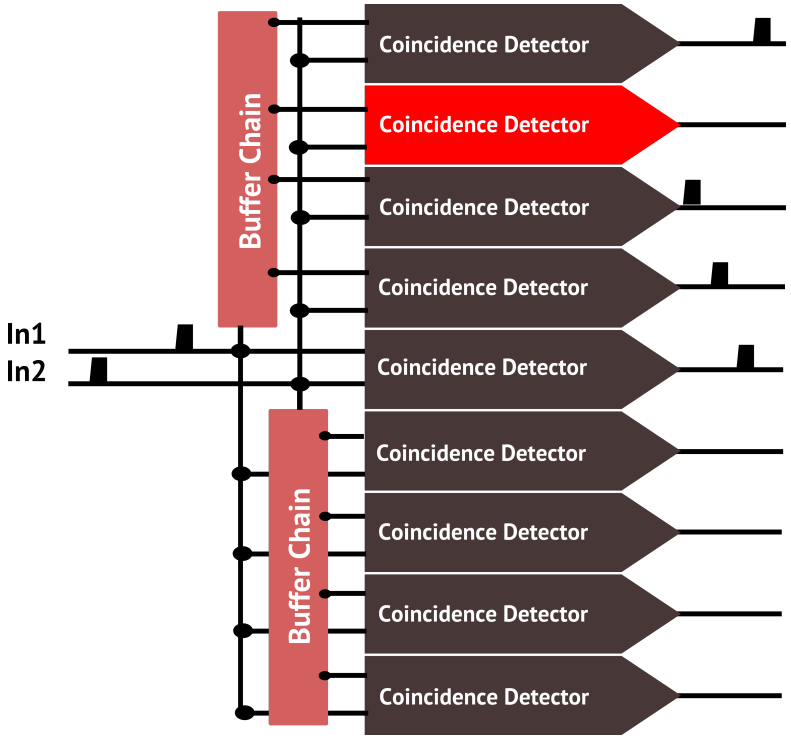


FIGURE 6.27: Graceful Degradation of the SDC

6.6.1 Input Reversal

The inputs to the SDC can be swapped (In1 to In2 and vice versa) to get the same output at the other branch of the SDC. This is possible because the coincidence detectors are mirrored around a central coincidence detector ("zero" detector). This technique can

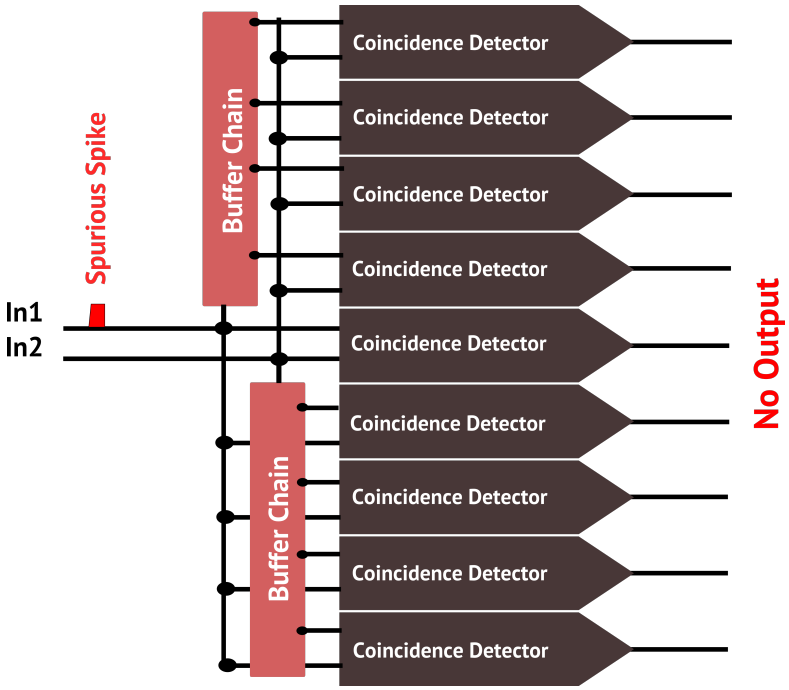


FIGURE 6.28: Spurious Spikes having No Effect on Outputs

also be used to reduce problems due to mismatch and process variations, effectively reducing the INL and DNL errors.

6.6.2 Comparison of Comparators, Latches, and Coincidence Detectors

From Figure 3.10, it can be observed, that coincidence detectors are doing tasks similar to a Flip-Flop or Latch in TDCs. These components can be compared to comparators in ADCs. A comparison between these three "detector" elements is shown in Figure 6.29.

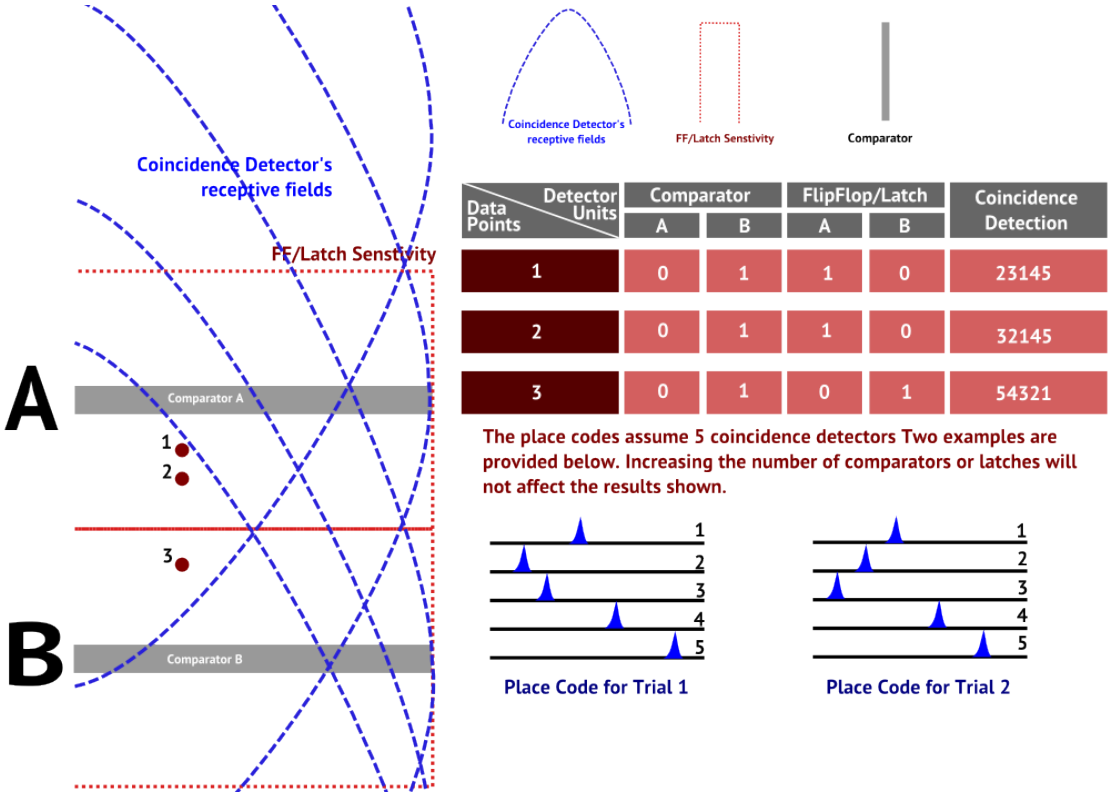


FIGURE 6.29: Comparison Between "Detector" Units

6.7 Decoding of Place Codes

The creation of place codes and their relationship with the Gaussian kernel width has already been discussed. If the Gaussian kernel width is minimum, then simple 1-of-N codes can be used for decoding. However, the decoding of place codes and how they can enhance the simple 1-of-N codes requires more explanation.

Figure 6.30 shows, how 4 values can be extracted by place coding, where, 1-of-N codes extract one value. A simple decoding algorithm can be used to decode these place codes as shown in Algorithm 1. As the width of the Gaussian kernel is increased, the number of bits will also be increased. However, the effect of time shift for higher number of place codes has to be studied. The time shift effect is greatly reduced, if the delay between two detectors is minimum.

6.7.1 Time Shift Effect

Figure 6.31 shows the time shift problem with place coding. This occurs because some coincidence detectors are evaluated earlier than the others. in Figure 6.31a the bottom detector (output E) gets evaluated first, the next detector starts evaluation 5 ns later and so on. This means that the output at E occurs 5 ns earlier than in ideal case. This

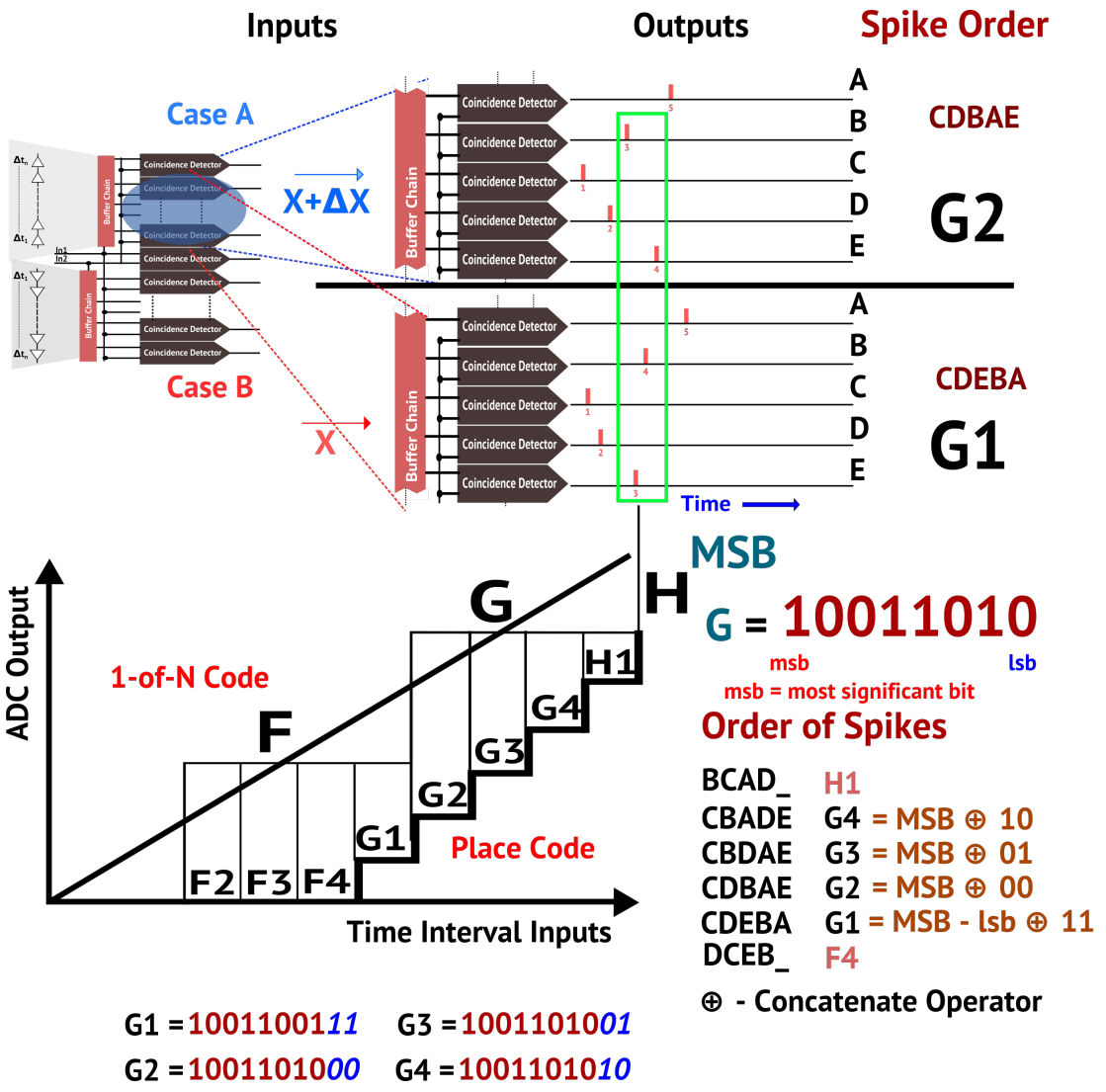


FIGURE 6.30: Example showing 2 bit Increase by Place Coding. (Here 4 values are obtained instead of 1 value. This is using a smaller Gaussian kernel width, which can be increased to much higher values.)

creates a limitation on place coding, if the delays between the detector are higher. In Figure 6.31 the time shifts would cause a reduction of resolution by one bit. This will not change with increase in the width of the Gaussian kernel, i.e., with increase in spike pairs used for place coding (in Figure 6.31 two pairs of spikes are used to get 2-bit increase as discussed in Figure 6.30). The effect reduces with the reduction in delays between the detectors, mainly because the Gaussian slope does not change. Since this is a predicted effect, it can be compensated during post-processing of the code or by a local adaptation scheme. This post-processing possibility is not available in our current readout implementation as the time information is removed by the time domain Winner-Take-All circuit, however, local adaptation will make it feasible and will be studied as part of the future research.

Algorithm 1: Decoding for Place Codes based on Figure 6.30.

Input: Place Codes with A,B,C,D,E

Output: Place Coded output

```
Find instance of the coincidence detector providing MSB(most significant byte) (In our example, "G")
lsb is the least significant bit
if D earlier than B then
    if E earlier than B then
        output = MSB − lsb ⊕ 11 ("G1" in our example)
    else
        output = MSB ⊕ 00 ("G2" in our example)
else
    if A earlier than D then
        output = MSB ⊕ 10 ("G4" in our example)
    else
        output = MSB ⊕ 01 ("G3" in our example)
return output
```

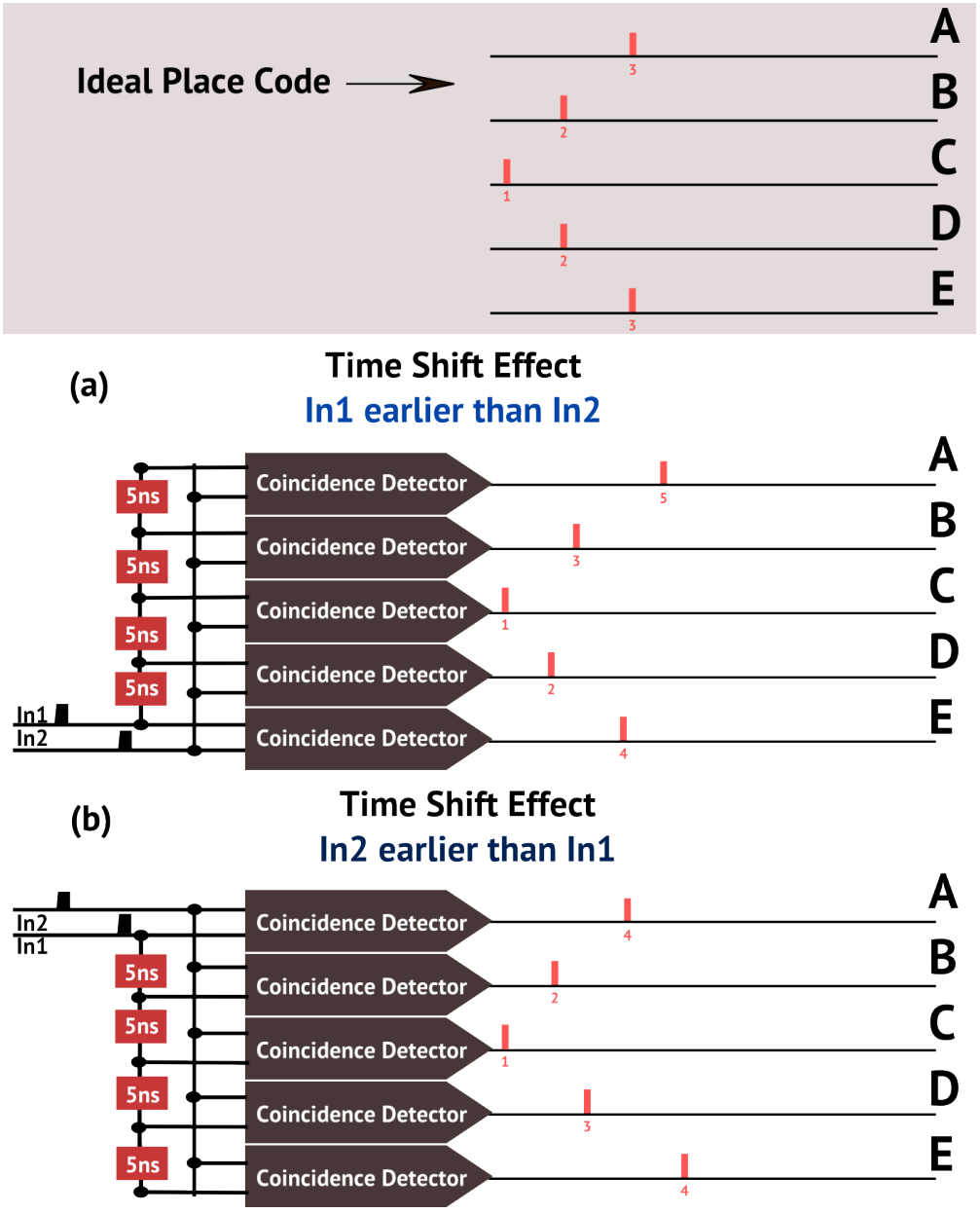


FIGURE 6.31: Time Shift Effect Visualized

6.8 Analysis of the SDC

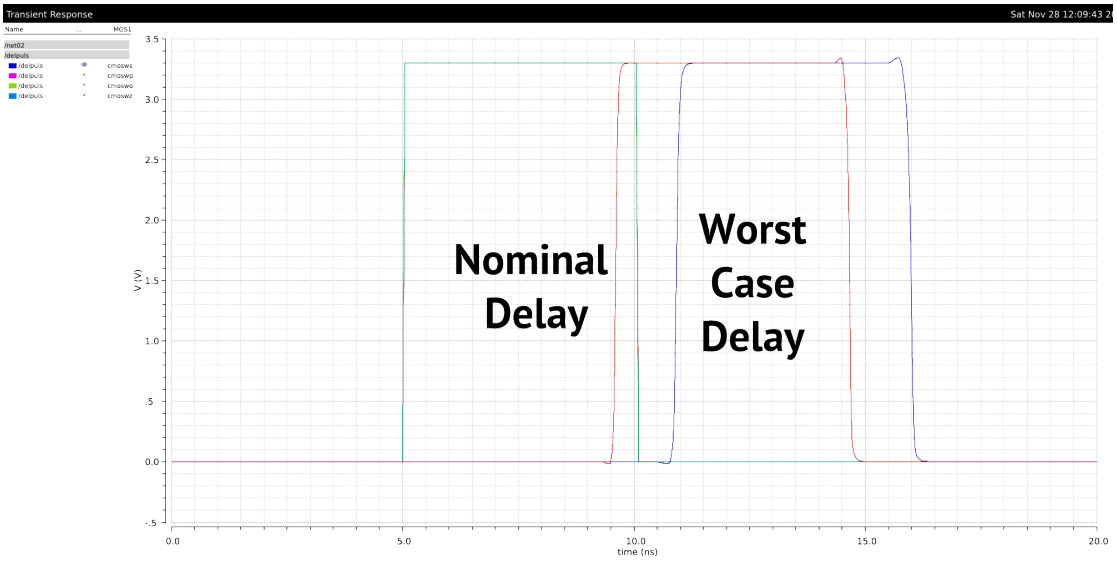


FIGURE 6.32: Worst Case Delay for the Buffer Chain

FIGURE 6.33: Worst Case Delay for the Coincidence Detector.

Figure 6.32 and Figure 6.33 show the worst case delays of the buffer delay and the coincidence detector respectively. It can be observed from these figures that the worst case delay for the buffer chain is 1.2ns and the worst case delay for the coincidence detector is 9ns. How will these values affect the quality of the SDC in terms of INL and DNL ?

The delay variations of the coincidence detectors do not affect the quality of the SDC to a significant extent. This is mainly because of place coding. The neighborhood spike times can be predicted for a given Gaussian kernel width. This problem is similar to

the missing spike problem explained in section 6.6. The effect of delay variation of each spike is reduced by the population of spikes by place coding effect.

The delay variations of the buffer chain have a larger effect in the INL and DNL of the SDC. If the delay variation is uniform throughout the design, then it will not have any effect on the INL and DNL, although, it will increase or decrease the span of the SDC. The worst case situation is if few of the buffer chains have the worst case delay, while the others remain nominal, as the delay is propagated to the rest of the chain.

A Monte-Carlo analysis can provide deviations that can be used to find the worst-case INL and DNL values by simulation. However, the digital models used in the layout, taken from the ams core libraries (standard cells from PDK) do not work with Monte-Carlo analysis at present, so a python based alternative was used in this scenario.

There are two ways to reduce this issue. (1) Delay calibration by using configurable delay chains. This is explained later in this work in section 8.4.2. (2) Reversing the input nodes to make use of the other branch of the SDC chain as discussed in section 6.6.1. This is possible with the current planned physical design as discussed later in chapter 8. Further, it should be noted that the delays can also be corrected using the various control voltages of the synapse.

6.9 Discussion

In this chapter, the Spike-to-Digital conversion using coincidence detection has been elaborated. The differences occurring due to the change in the width of the Gaussian kernels adds another dimension of reconfigurability to the SDC. Since this can be controlled by the voltage V_τ, there is immense freedom in adapting the SDC to various configurations, as described in section 6.1. The time domain winner-take-all with memory needs repeated measurements to get the place code values in the present proof-of-principle design. This can be unfolded to remove the need for repeated measurements, in future, in more generous chip designs. This is not required for the first prototype since the goal is to analyze the place coding and its properties on chip. An initial part of this work was presented at [113]

Chapter 7

SSC - Sensor to Spike Conversion

In this chapter, the focus is on conversion of signals from the sensors to time interval signals required by the SDC. There are several ways to approach this problem. For the first physical realization we focus on the Wheatstone full bridge based AMR sensor.

7.1 AMR sensor

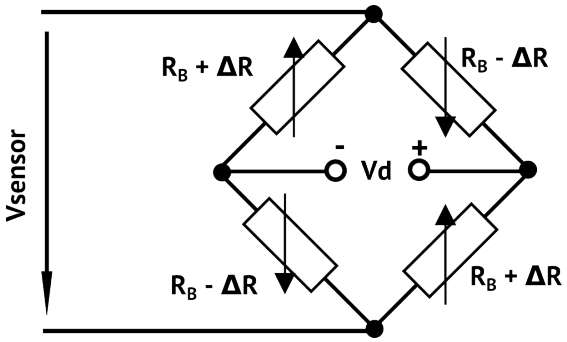


FIGURE 7.1: Wheatstone Full Bridge (AMR sensor)

There are various sensor concepts, which are single ended and differential. Wheatstone full bridge is a differential sensing concept, which is commonly used in many sensors. Our department ISE has a magnetic localization demonstrator shown in Figure 7.2 and described in [42]. This demonstrator is used to localize sensor nodes in industrial containers and related indoor scenarios. It makes use of a 3D AMR sensor module developed in our group shown in the Figure 7.3. The reasons described above make the AMR sensor very attractive as the test-bed, for our first prototype SSC as described in [114]. The properties of the AMR sensor are shown in the Table 7.1.

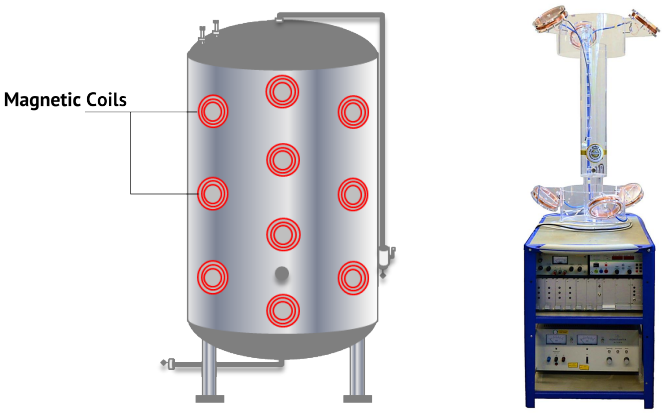


FIGURE 7.2: (left) Planned Industrial Setup of the Localization System (right) Experimental Demonstrator Exhibited at Hannover Messe 2013. (The images have been taken from [41])

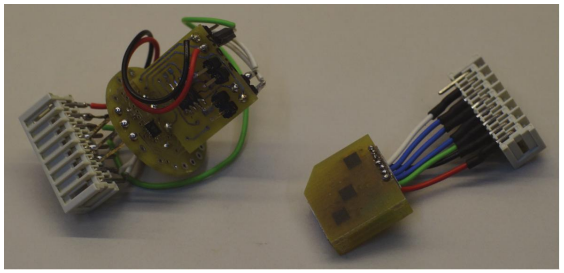


FIGURE 7.3: 3D AMR Sensor Module based on Conventional PCB Node with AFF755B (left) and Advanced AML Technology Node with AFF756 (right). (image taken from [42])

Symbol	Parameter	Typical Value	Units
Vsensor	Supply Voltage	5*	V
R_B	Bridge Resistance	2.5	kΩ
S	Sensitivity	15	(mV/V)/(kA/m)
SR	Linear Sensitivity Range	+/-160	A/m

Max ΔR for linear sensitivity range:
15(mV/V)/(kA/m) = 2.4 (mV/V)/(160 A/m)
= 7.92 (mV/3.3 V)/(160 A/m) (@Vsensor = 3.3V)
Vd/2 @ ΔR = 12 Ω = 7.92 mV
* Vsensor of 3.3 V is used in this work

TABLE 7.1: Properties of AFF755B AMR Sensor from Sensitec GmbH [60]

7.2 Time Interval Coding

In Chapter 3, various sensory mechanisms, that exist in humans and animals have been discussed. The first architecture, that was shortlisted was Jeffress model of sound localization. The analogon for the type of code, that is required for this technique, effectively, the SDC described in the earlier chapter is Time Interval Coding. Time interval coding is shown in the Figure 7.8. In later simulations, we will calculate the time difference and check for linearity over a range of values using the Wheatstone full-bridge. This is accomplished using cadence ocean scripts and python, mainly because this simulation cannot yet be automated by cadence. The ocean scripts are exported from cadence, then using an interface between python and ocean complex simulations can be performed and the related plots can be easily generated.

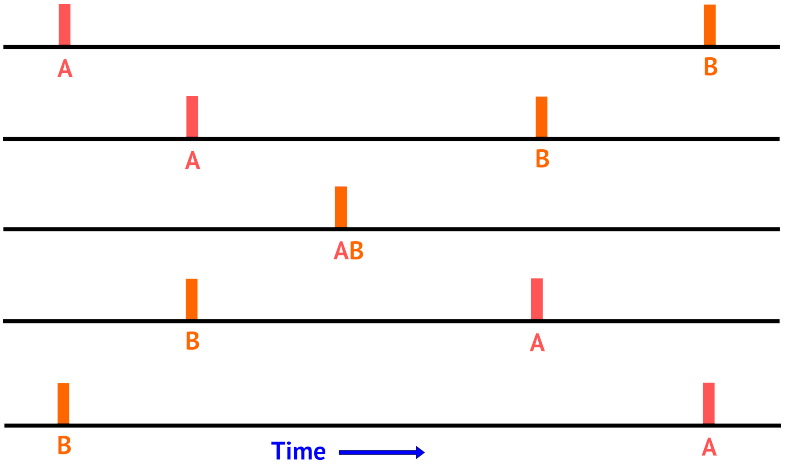


FIGURE 7.4: Time Interval Coding

7.3 Investigation with AMR Model, Simple LIF, and DPI Synapse

Figure 7.5 shows the sensor to spike conversion module with two synapses and two neurons. Each pair connects to different pins of the diagonal voltage V_d of the bridge. The Wheatstone full bridge uses the bridge resistance of the AMR sensor described in section 7.1. In Figure 7.8, a simulation of the setup for a range of -255 Ω to 245 Ω is shown. This is much higher than the linear range of the AMR sensor, which is 12 Ω, if the V_{sensor} is 3.3 V. The component sizes used for the synapse are shown in the Table 7.7. The size of the components for the neuron remain unchanged from the previous chapter with the membrane capacitance as an exception.

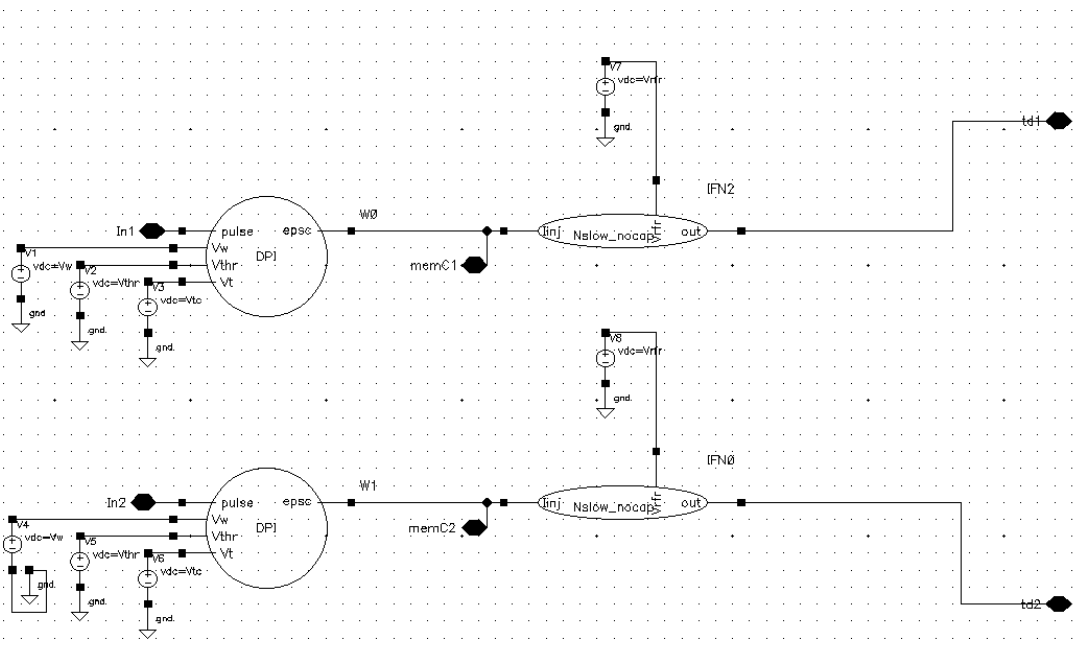


FIGURE 7.5: The SSC Setup

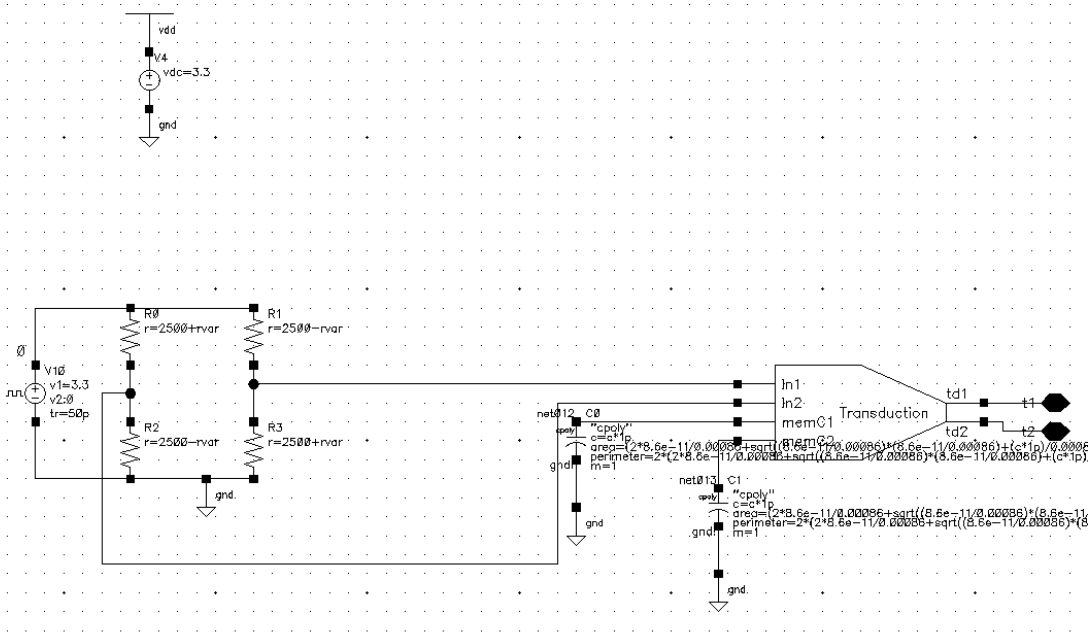


FIGURE 7.6: SSC Test Bench with AMR Sensor

7.3.1 Membrane Capacitor Sizing

While the simulation in Figure 7.8 shows the different aspects of time interval coding, the span, that can be achieved for the linear range of AMR sensor seems to be extremely small. It is around 20 ns. The SDC designed in the previous chapter can only provide 4-bits of information for such a small span. Our goal is to increase the variance of this output to a much higher value. This can be achieved by increasing the capacitance size

Components	Sizing (w/l) µm/µm	Components	Sizing (w/l) µm/µm
M_{thr}	1/3	M_W	6/3
M_{pre}	3/1	M_{dif}	1/3
M_τ	1.7/2.2	M_{syn}	2.2/13.5
C_{syn}	90 pF		

FIGURE 7.7: Component Sizes for DPI Synapse

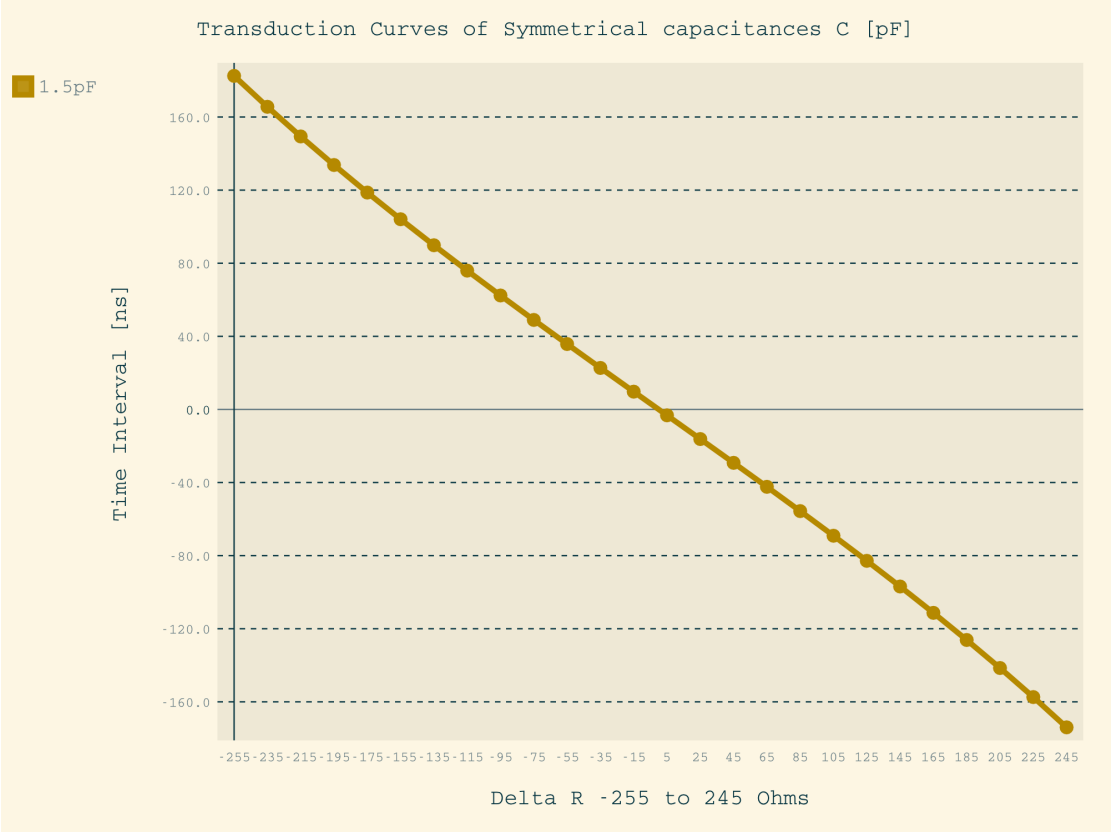


FIGURE 7.8: Time Interval Coding and Linearity

to a certain extent, as described below. This parametric simulation has been shown in the Figure 7.9. This has also been studied in detail in our earlier work [115]

In Figure 7.9 we can see that the the highest variance occurs at 5 to 6 pF. Two of these values are chosen and checked for linearity in Figure 7.10 and Figure 7.11. We can see that 6 pF membrane capacitance provides a linear output over the complete tested range. This will be useful for various sensing elements, however, the AMR sensor will still see only a variation of 50 ns, which is not sufficient for obtaining high resolution by the SDC.

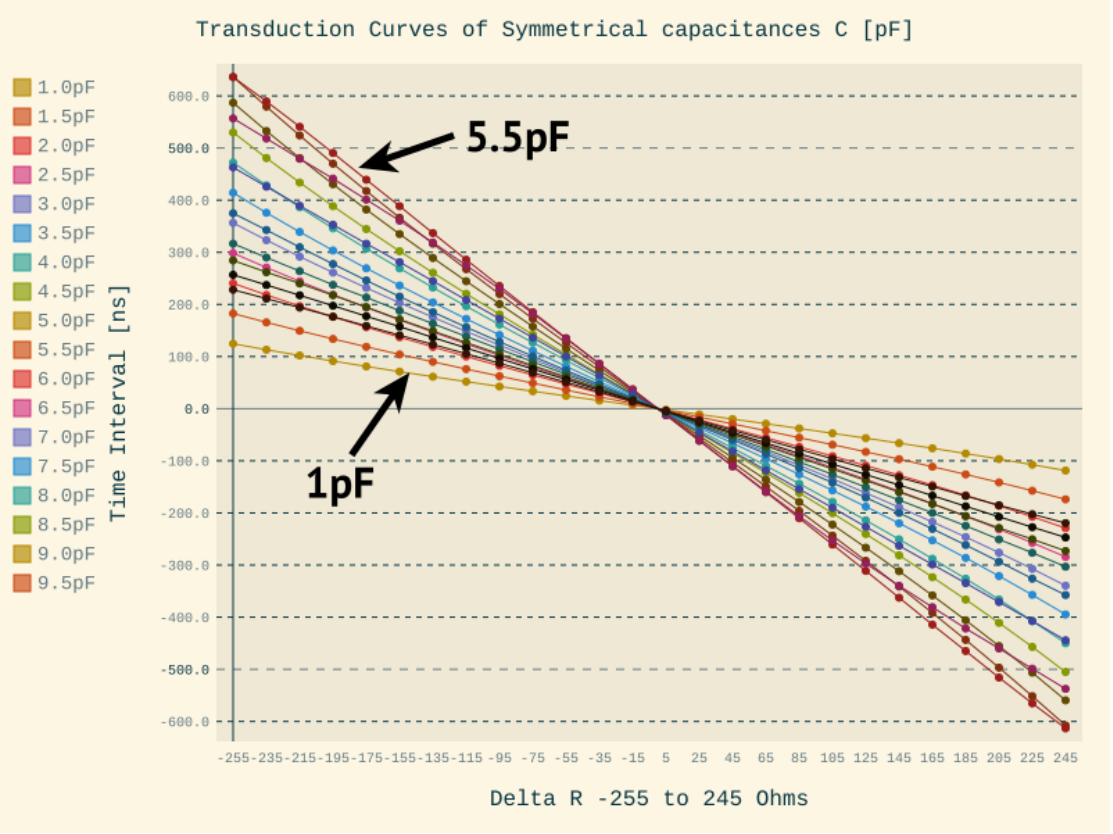


FIGURE 7.9: Time Delay Plot for Different Capacitances to find the Highest Variance

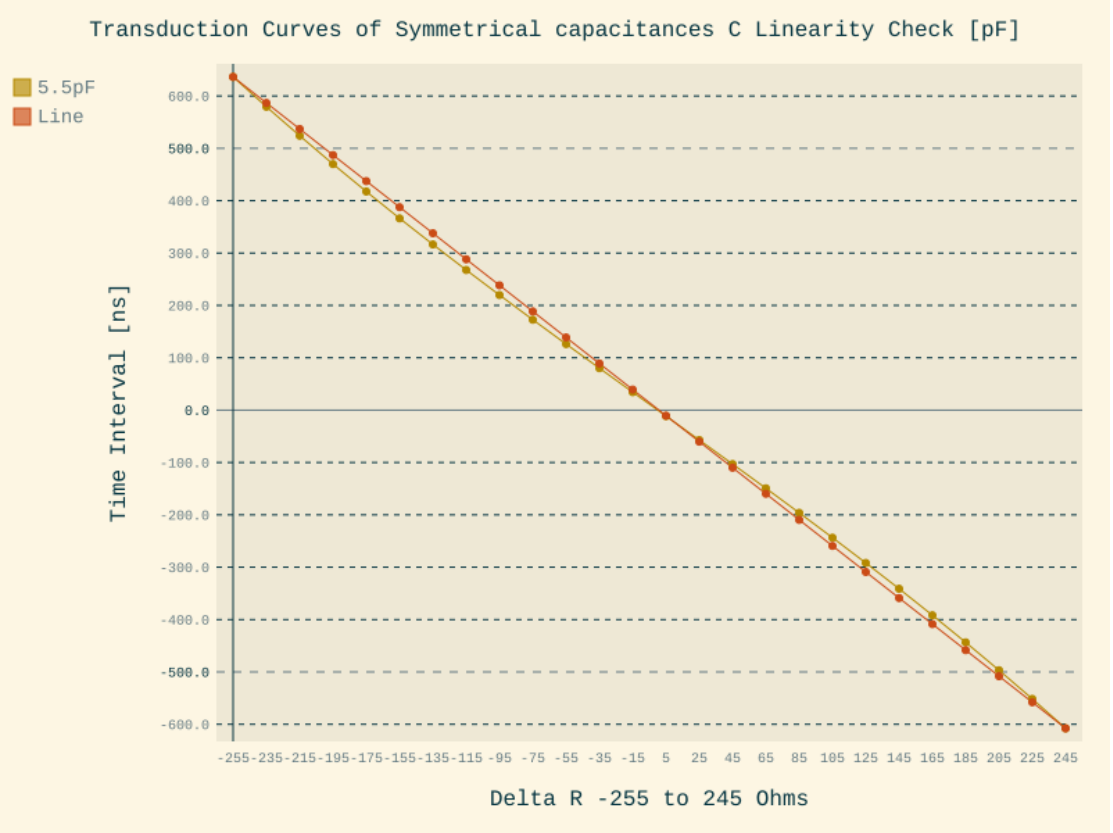


FIGURE 7.10: Checking the Linearity of SSC for a Membrane Capacitance of 5.5 pF

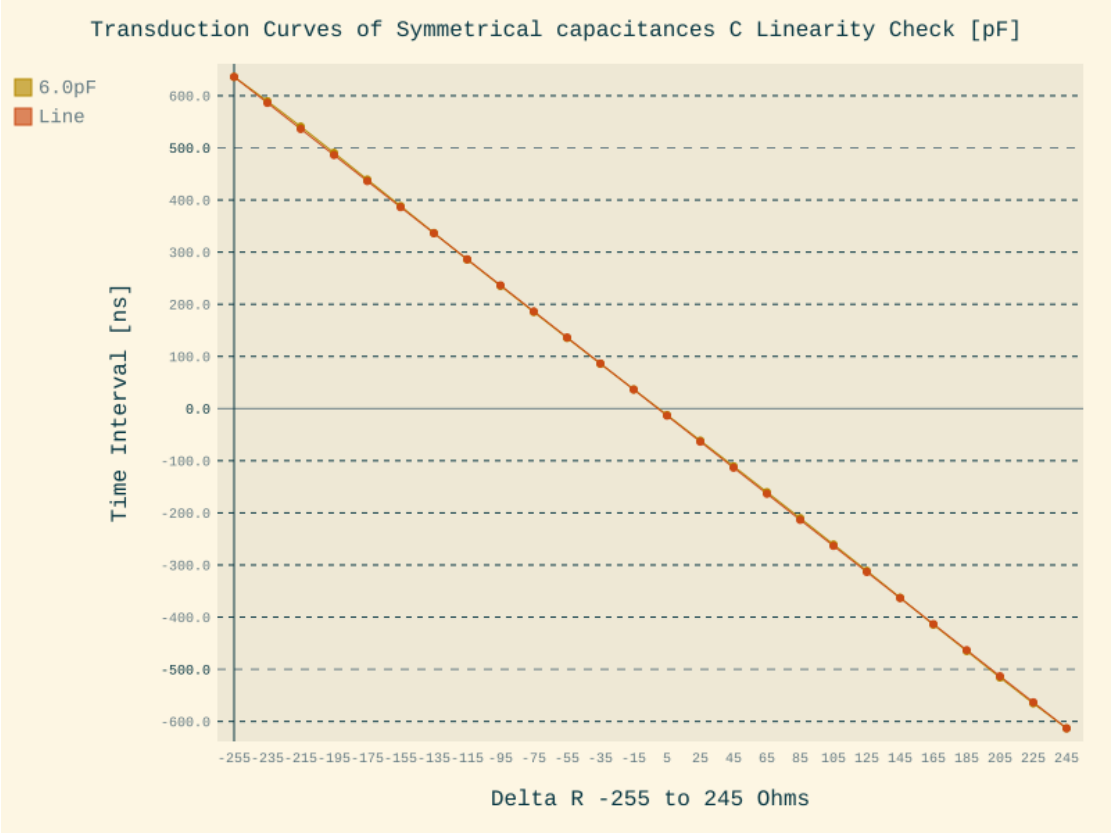


FIGURE 7.11: Checking the Linearity of SSC for Membrane Capacitance of 6 pF

7.4 Differential Synapse

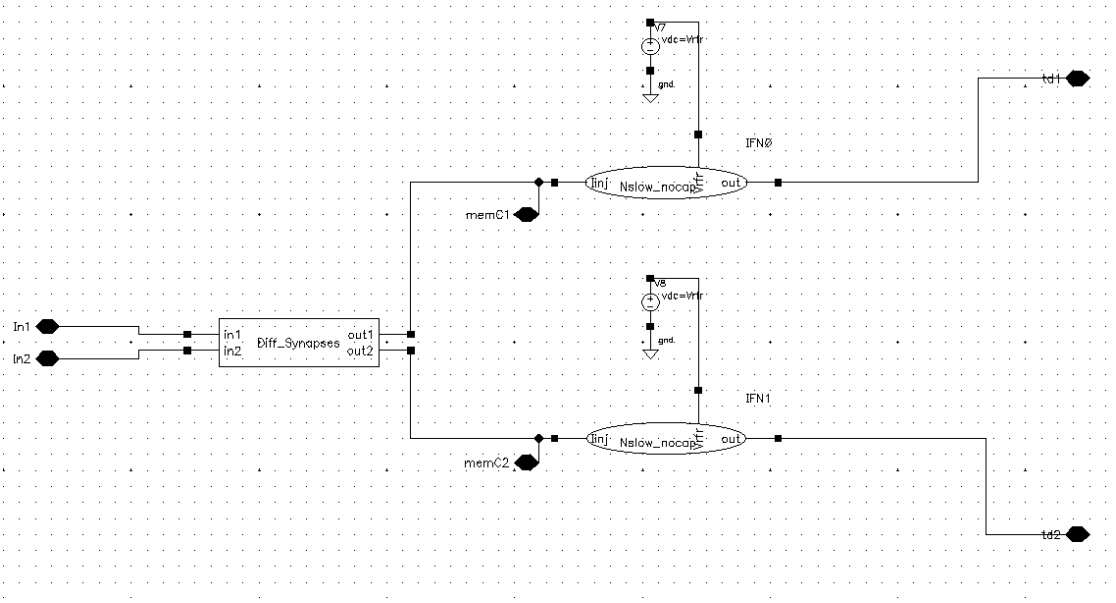


TABLE 7.2: New SSC Circuit

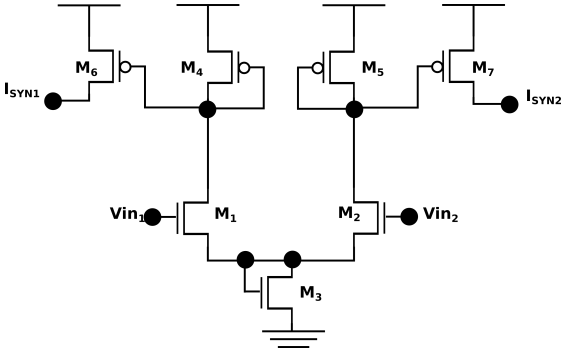


FIGURE 7.12: Differential Synapse for SSC

Since the previous approach was insufficient, a differential current source with tail current controlled by a voltage source was suggested [116]. After testing the basic suggested circuit, it was found that a diode-connected transistor produced better linearity than the tail current source and this was then used as shown in the Figure 7.12. All the transistors are sized in the w/l of 1/1 μm/μm.

The first test to be performed was for linearity of the current output from the new synapse. This is shown in the Figure 7.13. In addition to the simple linearity test, the figure also shows the drift due to temperature. The interesting part is that the difference between the currents remain equal, which means that the time interval will not be affected by the temperature related drifts.

Figure 7.14 shows the time interval outputs for the same range as the previous test set up. One of the main observations, is that it is non-linear. However, this non-linearity

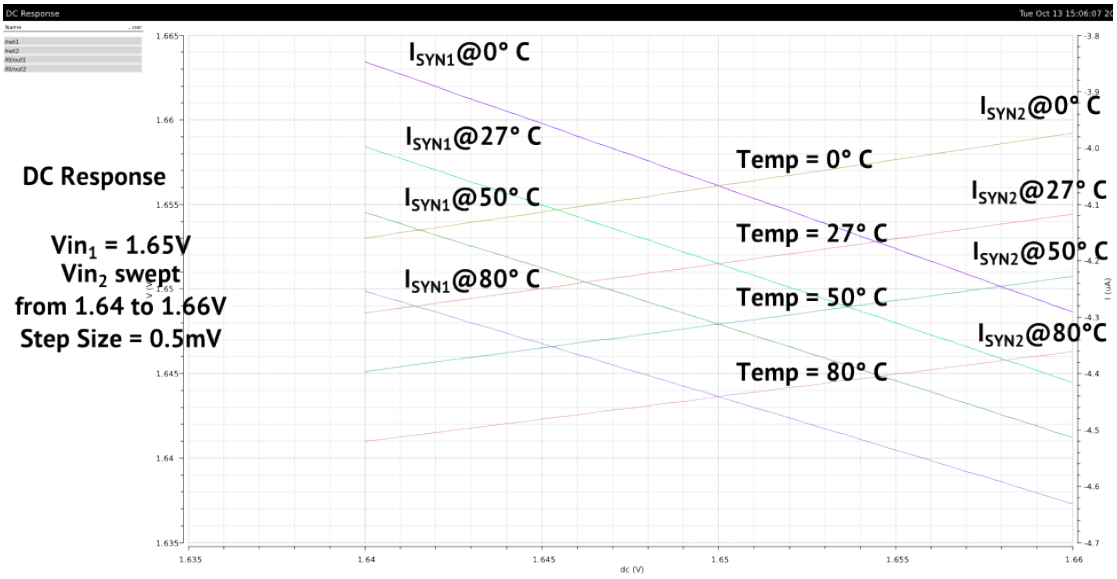


FIGURE 7.13: Drift Due to Temperature in the Differential Synapse. (The difference in current is not affected.)

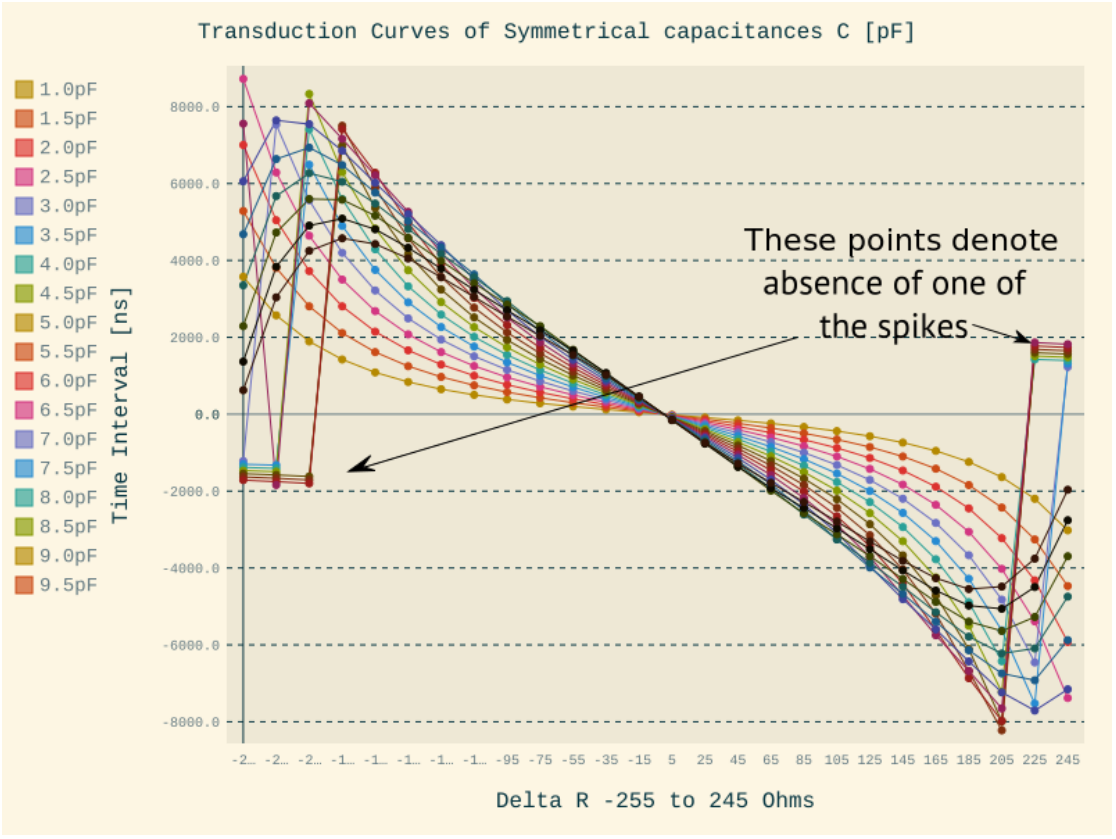


FIGURE 7.14: Simulation to Find the Best Membrane Capacitance Using the Differential Synapse

is attributed to the neuron and not to the synapse. Although this output is non-linear in the range pictured, it is linear in the range of the AMR sensor. This also increases the variance from 50 ns of the previous setup, upto 2 μs. This can be seen in the

Figure 7.15. This massive amplification provides the freedom required for the SDC to work. Figure 7.16 shoes a comparison the linearity for different transistor sizes with the membrane capacitance showing maximum variance. While the worst-case analysis shows deviations, the size of the capacitance is robust enough that, it never fails or becomes non-linear in these conditions.

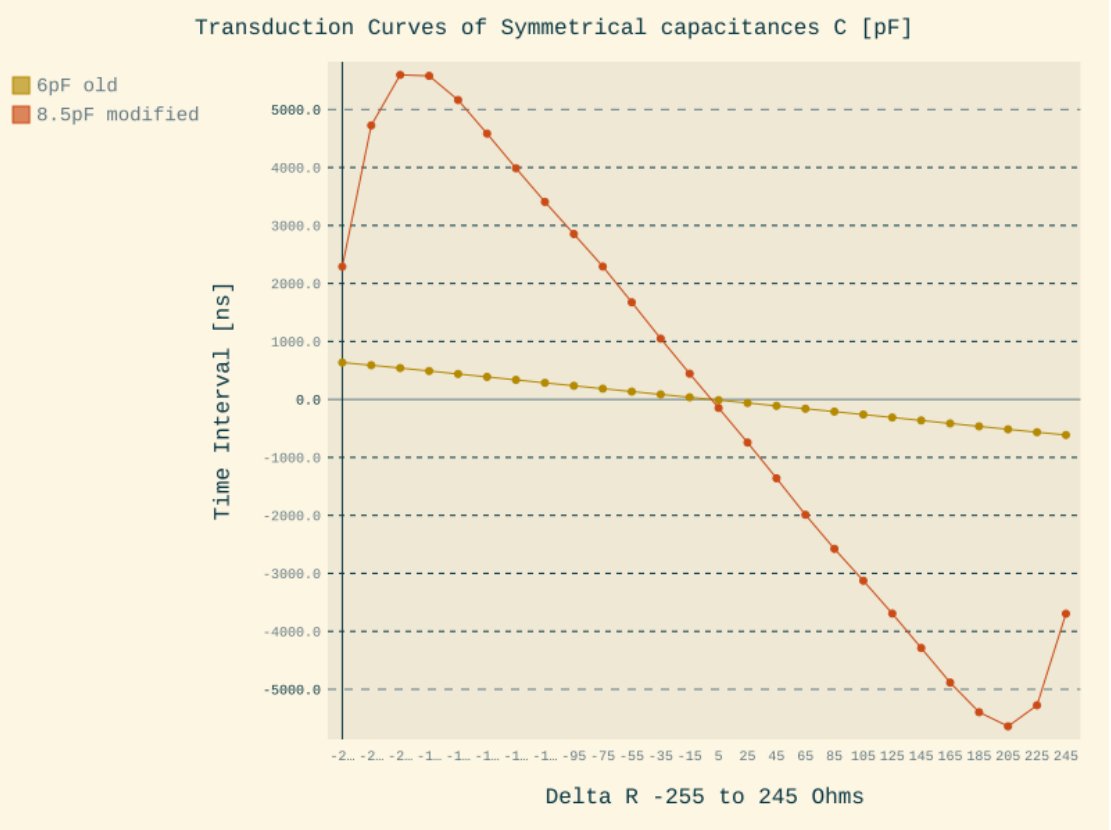


FIGURE 7.15: Comparison of the Curves for the DPI Synapse and the Differential Synapse

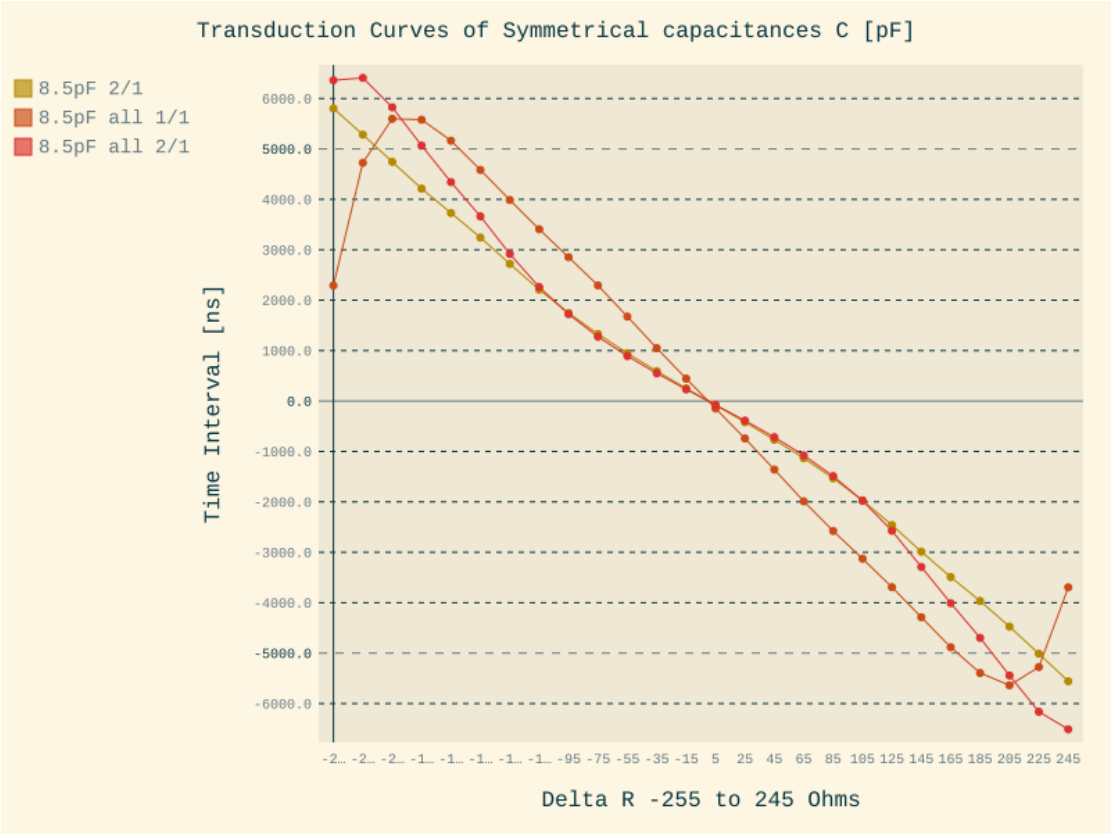


FIGURE 7.16: Comparison of the Membrane Capacitance with highest variance for different transistor sizes. The most linear plot with highest variance is 8.5 pF, when all transistors are sized for W/L of 1/1 μm/μm.

7.5 Physical Design

The general simulations of the SSC with the differential synapses were performed in the previous section. In Table 7.4 the final sizes that are used for physical design are shown. The goal of the sizing is to bring the linear range of the AMR sensor to the full range possible by the intended physical design of the SDC. This is approximately +/-500 ns. This is reflected in the component sizes. The size of membrane capacitance is chosen to be 5 pF. Compared to the size of the planned SDC and the 5 pF membrane capacitance, the sizes of the transistors are negligible. So, more freedom with area is taken. Since the dynamic range does not play a major role in the first prototype, the diode connected transistor is scaled to a higher size, increasing its tolerance.

Figure 7.17 shows the layout of the differential synapse. All considerations for matching using interleaved structures has been used. Figure 7.18 shows the layout of the two neurons. The neurons are matched so that they could drift without affecting the time interval outputs.

Components	Sizing (w/l) µm/µm	Components	Sizing (w/l) µm/µm
M_1	30/1	M_2	30/1
M_3	10/10	M_4	30/1
M_5	30/1	M_6	25/1
M_7	25/1		

TABLE 7.3: Component Sizes of the Differential Synapse

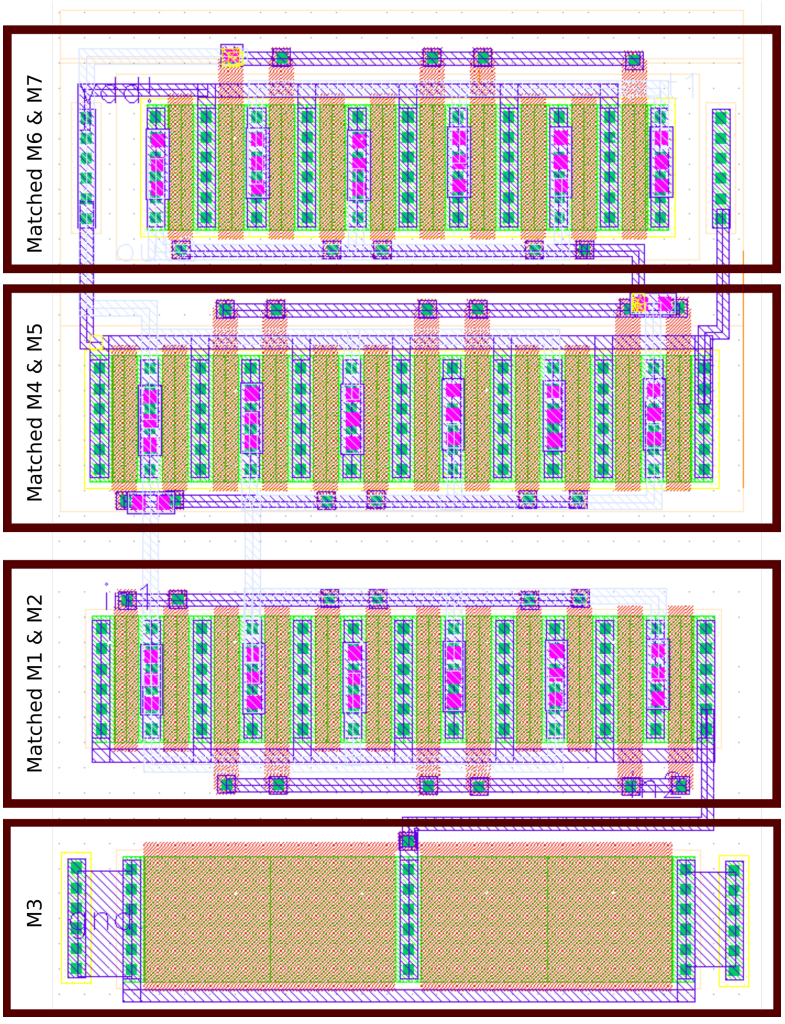


FIGURE 7.17: Layout of the Differential Synapse of Figure 7.12

Figure 7.19 shows the design of the complete SSC. The capacitors are designed with Faraday shielding again. Figure 7.20 shows the comparison between schematic and post-layout simulations for the considered sizes. They are identical for all requirements of the application.

A reconfigurable version of the differential synapse has been designed to increase the flexibility of the SSC for different sensor elements. This uses 8 bit scalable NMOS

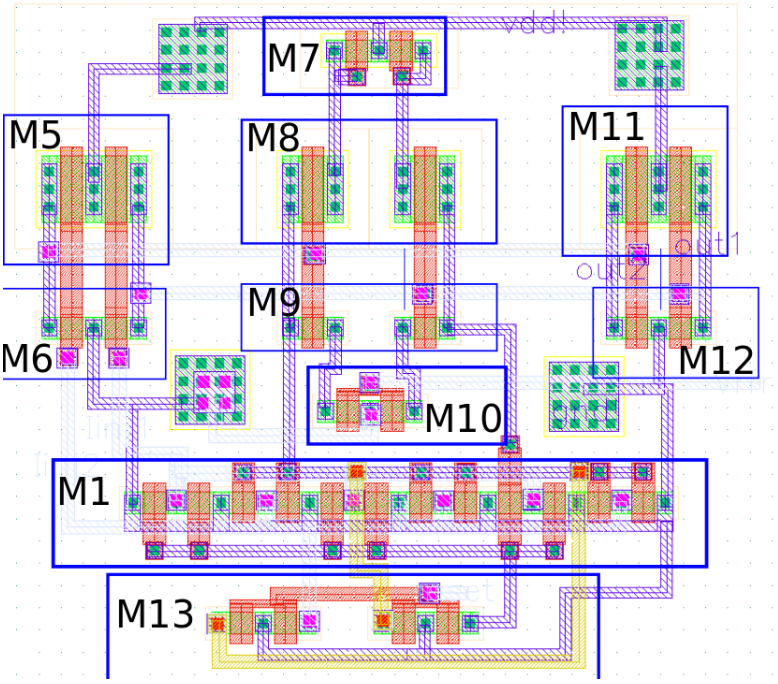


FIGURE 7.18: Layout of the Neuron Pair as shown in Figure 7.5. (The transistor names represent both neurons as they are mostly matched.)

Components	Sizing (w/l) µm/µm	Components	Sizing (w/l) µm/µm
M_1	(1to128)/1	M_2	(1to128)/1
M_3	10/10	M_4	(1to128)/1
M_5	(1to128)/1	M_6	(1to128)/1
M_7	(1to128)/1		

TABLE 7.4: Component sizes of the Reconfigurable Differential Synapse

transistors, which have been designed by Freier in [41]. 8 bit scalable PMOS transistors have been designed based on this scalable NMOS design. The shift register to store the programmed bit pattern has been adapted from [41]. This reconfigurable synapse has all transistors except diode connected transistors M_3 as scalable elements. This provides flexibility to control the span of the SSC and to remove deviations due to process and manufacturing, and drifts due to temperature.

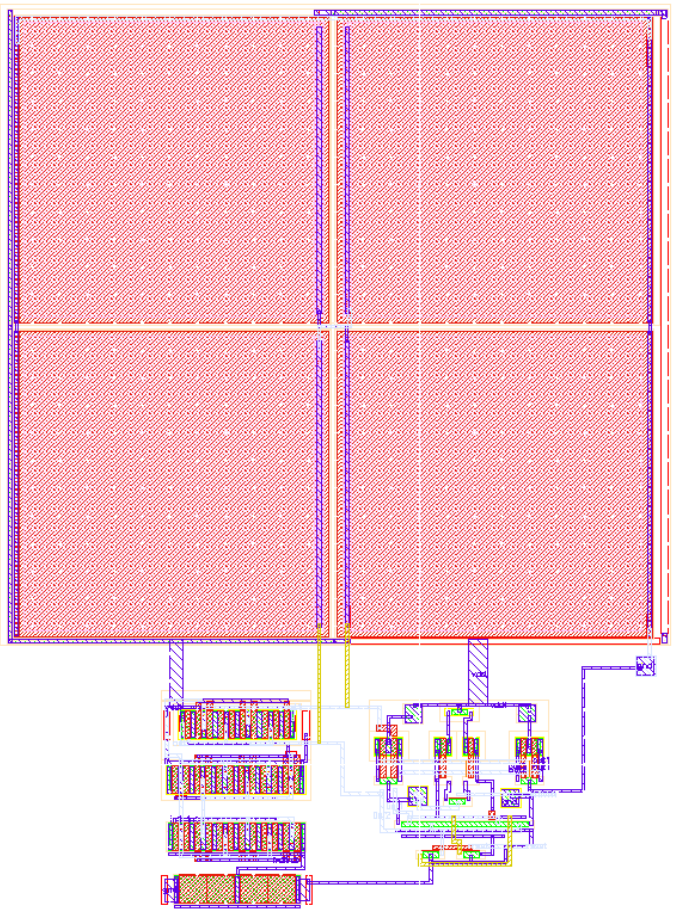


FIGURE 7.19: Layout of the SSC

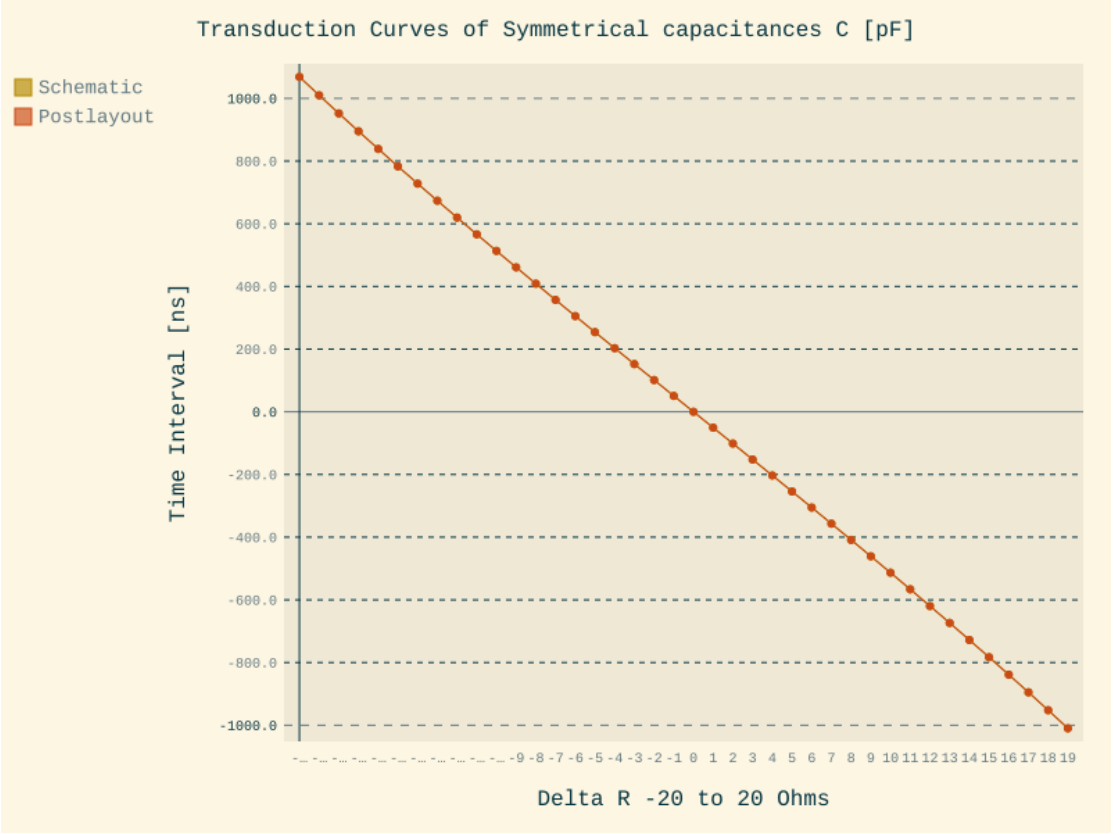


FIGURE 7.20: Comparing the Schematic and Post Layout Simulations for the Final Design

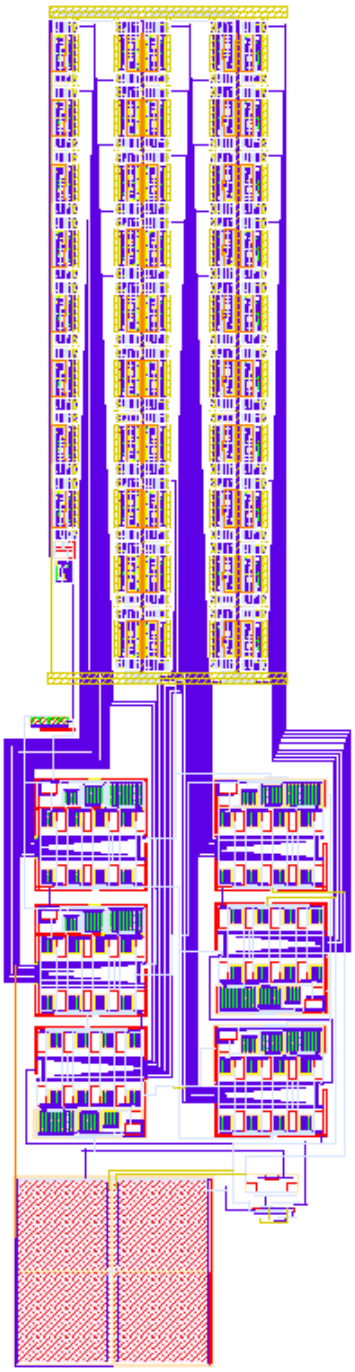


FIGURE 7.21: Layout of the SSC with Reconfigurable Synapse, with Shift Register to Program the Scalable PMOS and NMOS Transistors. (The shift register and scalable NMOS transistors were taken from [41])

7.6 Analysis of the SSC

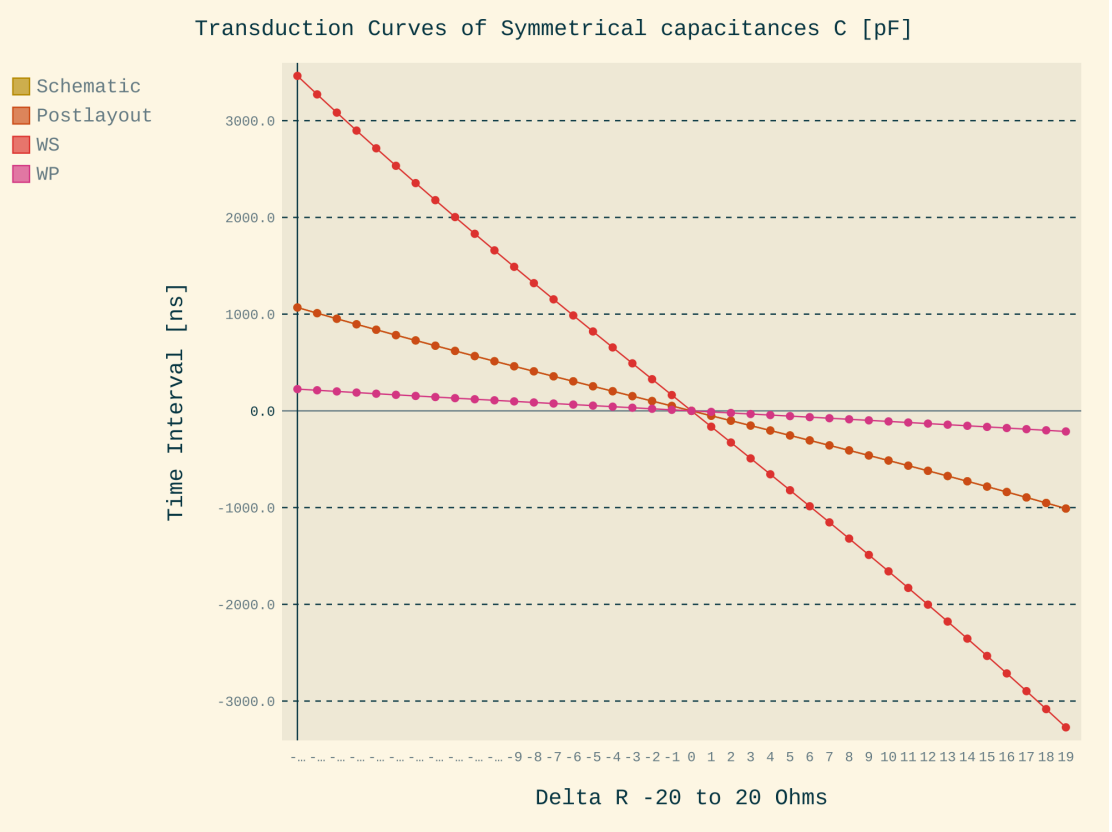


FIGURE 7.22: Schematic, Post Layout Plot shown in Figure 7.20 Compared with Worst Case Speed and Worst Case Power.

Figure 7.22 shows the worst case scenarios for the time interval codes at outputs of the SSCs. As discussed in section 7.4, the deviations due to manufacturing do not affect the linearity, only create a drift in the values. This can also bee seen from this figure. Although there is a deviation from the nominal value, which increases or reduces the total span required from the SDC, there is no problem with the linearity.

The effect of changes to span can be easily counteracted by reducing the supply voltage to the sensor. A better way to do it is to reconfigure the transduction element according to the required span. This can be done easily with the reconfigurable version of the synapse discussed earlier.

From this analysis, we can conclude that the adaptation is mainly required at the SDC outputs to reduce the deviations due to delay elements. The linearity of the SSDC is not majorly affected by the SSC module.

7.7 Discussion

In this chapter, one sensor to spike conversion method has been studied and discussed. Since the capability of the DPI synapse was not able to represent the sensitivity of the AMR sensor, so a new synapse was introduced and implemented. While this differential synapse was implemented specifically targeting the AMR sensor, a more generic reconfigurable synapse was also designed. This synapse can be programmed for any Wheatstone bridge based sensor interface. It can also be used for single-ended sensors by having the other one at voltage equal to half of the sensor range. The use of multiple SSC techniques feeding one SDC without loosing information will be part of the next chapter.

Chapter 8

SSDCα - Sensor to Spike to Digital Converter

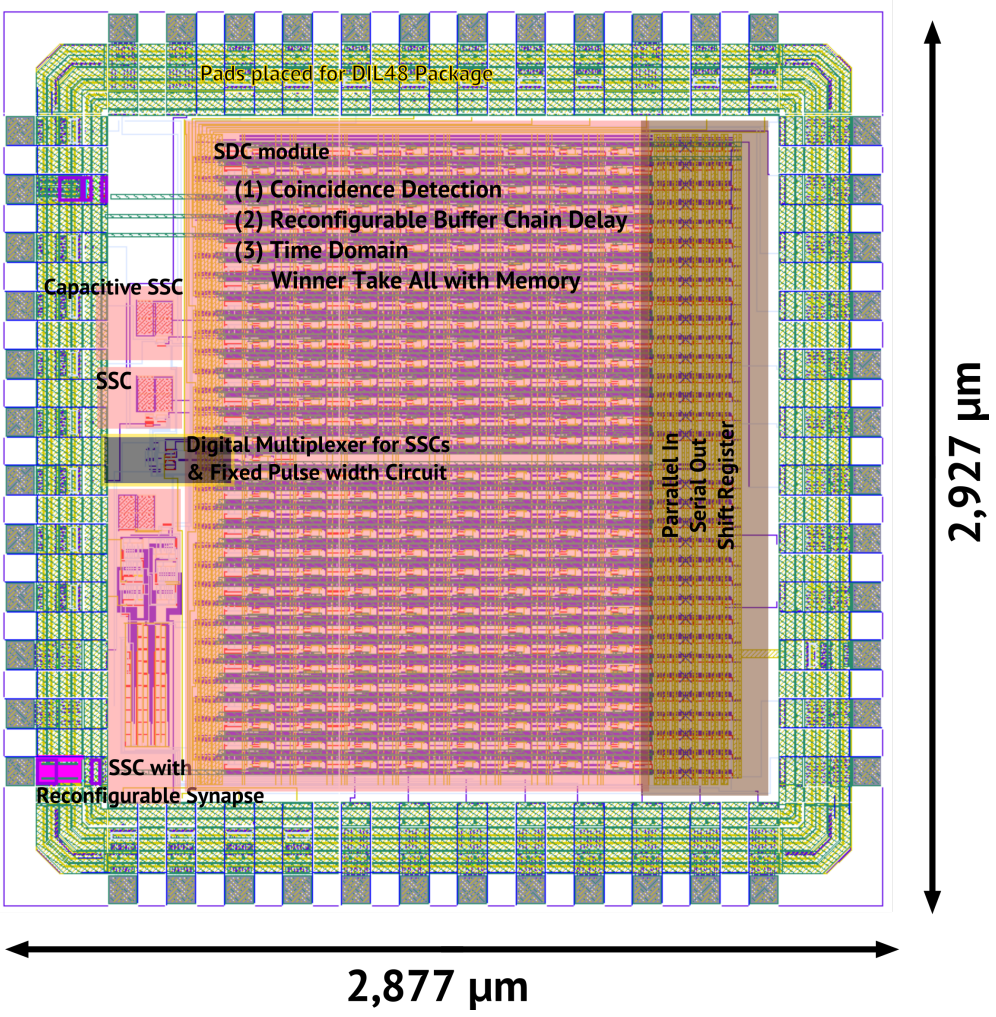


FIGURE 8.1: Layout of the First SSDCα for 8-13 bit A/D Conversion in 350 nm AMS Technology.

Figure 8.1 shows the final design of the chip with the modules highlighted. Each module has features and choices, which have not been discussed in earlier chapters.

8.1 Analog Processing & Digital Communication

The most important and interesting feature of the spike processing is that the shape of the spike is not very important for the processing method. While this is true with biological neurons and the models used in the work, these features can be improved to suit the task at hand. In this work, all the communication between modules is treated as digital. This can be implemented mainly because the time of the spikes is the only relevant information that is required, the spike amplitude, spike width etc. are not part of the processing.

From the explanation above, it can be deduced that only analog communication and processing occurs within the two main modules (1) SSC, (2) Coincidence Detection module. All communication between these modules are digital. This gives us the freedom to perform digital processing between these modules.

The following features were a direct consequence of this advantage.

- Multiplexing several SSC units as input to the SDC.
- Providing the SSC outputs as outputs from chip.
- Providing inputs to the SDC from outside the chip.
- Using a fixed pulse-width generator based on rising edge of input spikes to make the coincidence detection predictable.
- Multiplexing different delays, so that the span of the SDC can be changed, creating a zooming effect.
- Using the flip-flop based readout circuit for SDC described in Chapter 6.

8.2 Spike to Digital Converter

The SDC module is designed with $128 * 2 + 1$ individual modules as shown in the Figure 6.24. Each of these modules is connected to a reconfigurable buffer chain and the readout circuit described in Chapter 6, Section 6.4.1.

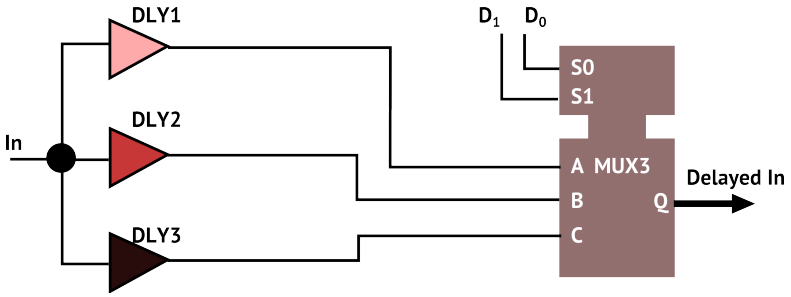


FIGURE 8.2: Reconfigurable Buffer Delay

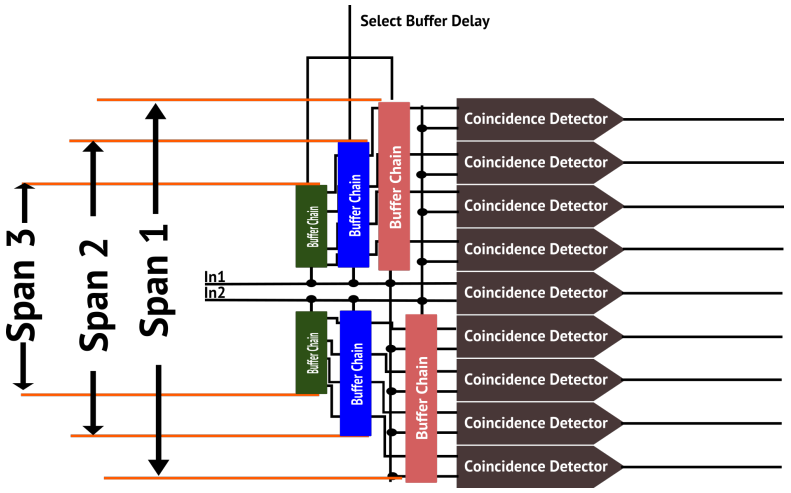


FIGURE 8.3: Span Change by Reconfigurable Delay

8.2.1 Reconfigurable Delay - Zooming Effect

A reconfigurable delay using a multiplexer could be created by having delays in series or in parallel. Having delays in series will reduce the area since multiple delays can be combined to form a larger delay. In this work, parallel delays have been chosen.

The buffer chain is the most important part of the SDC. Failure of one buffer will disable all the buffers after it, making the SDC fail. By having delays in parallel, this problem is resolved by reducing the probability of failure. One buffer in each line has to fail to make the SDC fail completely. Thus a parallel set of delays might be safer than delays in series in this scenario. The multiplexer is still one point of failure, however, these layouts have been taken from ams hitkit libraries, which have been tested and proven after several manufacturing runs [117], so the possibility of failure is still much less than for unproven designs.

8.2.2 Priority Encoder vs. Shift Register Readout

The 257 output lines from the SDC could be read out in two simple ways (not considering the use of packages like BGA to reduce costs) (1) 2 Priority Encoders with opposing priorities reducing the 257 output lines to 18. (2) Parallel in Serial Out Shift Register.

The use of priority encoders have several advantages like speed and power consumption, since only one clock cycle is required to read out the data. However, in the first test chip we can efficiently make use of the chip area, if we use DIL48 package. Any larger package might result in a lot of empty area on the chip. To reduce the pads and to efficiently study the output codes, the shift register was chosen for this work. This reduces the pads required from 18 to 2 for the readout. This will not be the case for future works.

Another advantage of using the shift register is the recording of every spike that is triggered at the same time. The time domain winner take all circuit that was designed has a sensitivity of 150 ps, all spikes arriving within this time will be triggered. The priority encoders can only read the two edges of the spikes in the scenario, while a shift register can be used to get every spike triggered. This helps in analyzing the first prototype in detail. Table 8.1 provides a comparison between three different readout methods. The

Property\Technique	WTA + Shift Register	WTA + Priority Encoder	Unfolded WTA
Area	Low	Low	High
Read out speed	Low	Medium	High
Power Consumption	Medium	Low	High
Sensor Measurments	Multiple	Multiple	Single
Robustness	High	Low	Medium
Pads Required	2	N_{bit}^2+1	$(N_{bit}^2+1)xN_{folds}$

TABLE 8.1: Comparing different Readout Techniques

most interesting is the unfolded WTA, which required larger area, however, it needs only a single sensor measurement unlike our present moderate chip implementation effort. As mentioned before, the present technique is only used to analyze the behavior of the first prototype chip.

8.3 SSC

There are three SSC modules designed for the chip. They are

- SSC tuned for AMR sensor node

- Reconfigurable SSC for various sensor nodes
- Capacitive Sensing Prototype SSC

8.3.1 Multiplexing several SSC Units

The output of the SSC units are spikes with spike timing information. These spikes can be treated as digital pulses and sent through several digital multiplexers without loosing any information. Although analog multiplexers can also be used in a similar way with general sensor conditioning techniques, there could be a loss of information, when the sensitive sensor data passes through transmission gates. This problem does not exist with spike processing. The simple structure used is described in Figure 8.4

8.3.2 Fixed Pulse Width Generation

The Spike-to-Sensor-Converter circuits produce spikes of varying pulse widths. These circuits model biological neurons, which make use of both intensity information (as pulse widths or rates) and spike timing information. In this work, we do not use the pulse width information. The coincidence detection capability of the SDC varies with the pulse width, it can be tuned to be more efficient a certain pulse widths. When the pulse width is high, it usually reduces the efficiency of coincidence detection, as the width of the pulse will effect the spike timing information at the synapse.

A simple flip flop triggered by the rising edge of the spike and reset after a fixed time delay is used to create pulses at approximately 5 ns, as shown in the Figure 8.6. This was also the pulse width used during the experiments shown in Chapter 6.

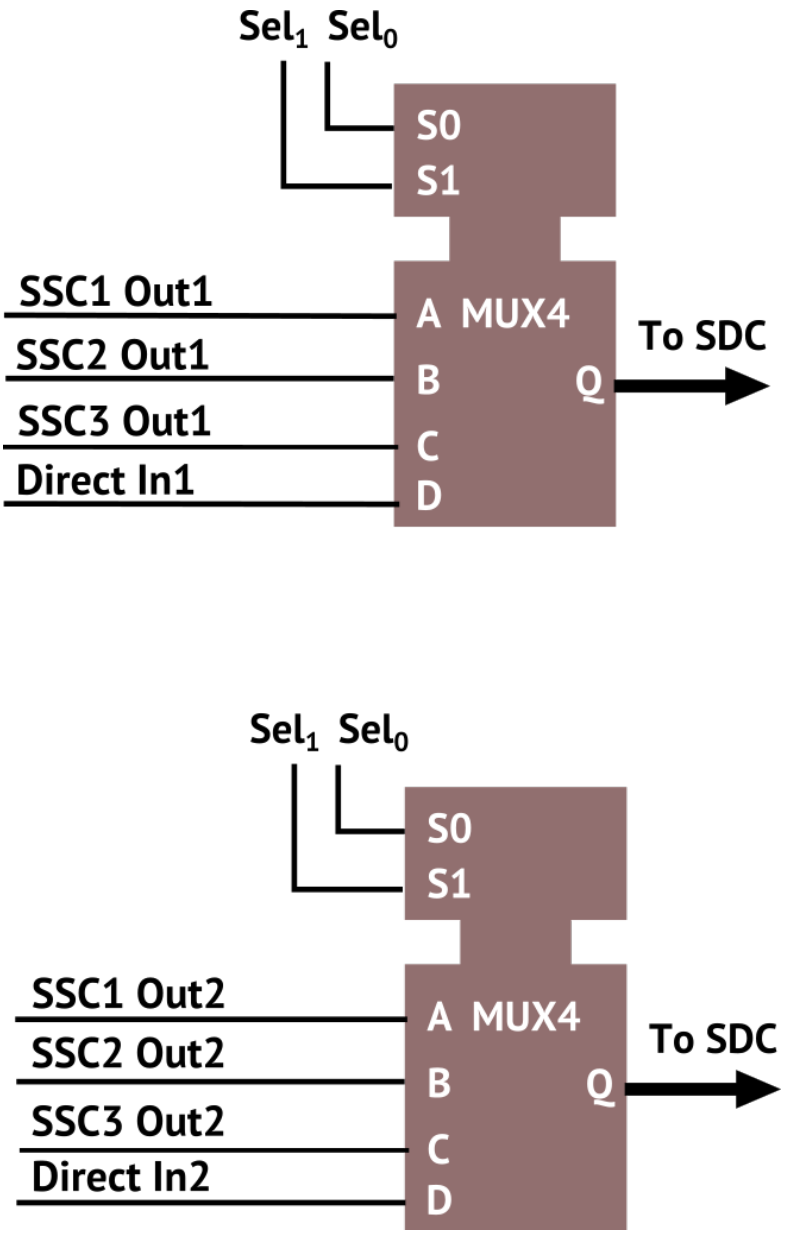


FIGURE 8.4: SSC to SDC Connection

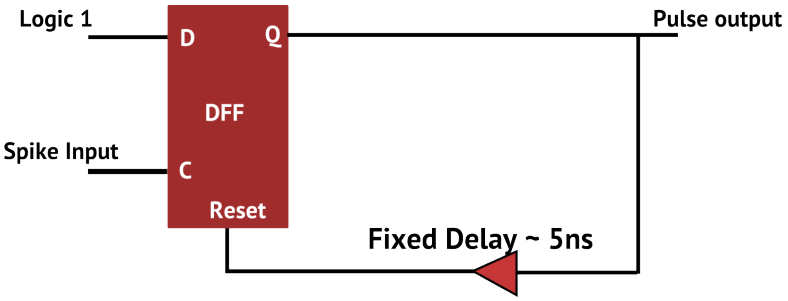


FIGURE 8.5: Fixed Pulse Width Generation

8.4 Adaptation

Adaptation is one of the biggest motivations for approaching biological techniques. The sensory system must be extremely adaptive for an organism to survive. In this work, the synapses have been designed with several control voltages, which can be used for adaptation like V_W, V_{thr}, V_τ. However, for the first prototype these voltages have been made common to all the detectors except for V_τ. The SDC has been divided into 8 blocks based on the location on chip to have 8 different V_τ inputs. This gives us some room to adapt the neurons to deviations during manufacturing.

There are different ways in which these can be designed for every neuron to be programmable.

- Analog Capacitor Memory: A capacitor for every programmable voltage, which is charged by a DAC (internal or external). The DAC charges every capacitor in a cyclic scheme where every capacitor is recharged to the programmed voltage before it deviates from the programmed value.
- Floating Gates: Analog floating gates can be used to store the programmed values for each control voltage in the neuron and synapse. This is more area efficient and has been used in synaptic structures before.
- Memristors: In HRL [56] neuromorphic chips memristors have been used for storing the synaptic weights. Many neurons share these weights in a time multiplexed approach.

8.4.1 Hybrid Adaptation

Although the current physical design of does SSDC does not have traditional neural adaptation methods like STDP (Spike Time Dependent Plasticity) yet, it has the following means of adaptation.

- Reconfigurable SSC for intrinsic optimization using tools like ABSYNTH
- Reconfigurable span, by reconfiguration of delays.
- Adaptation of SDC using the control voltages. Although many control voltages are common to all coincidence detectors, they can be programmed for the best performance of the SDC.

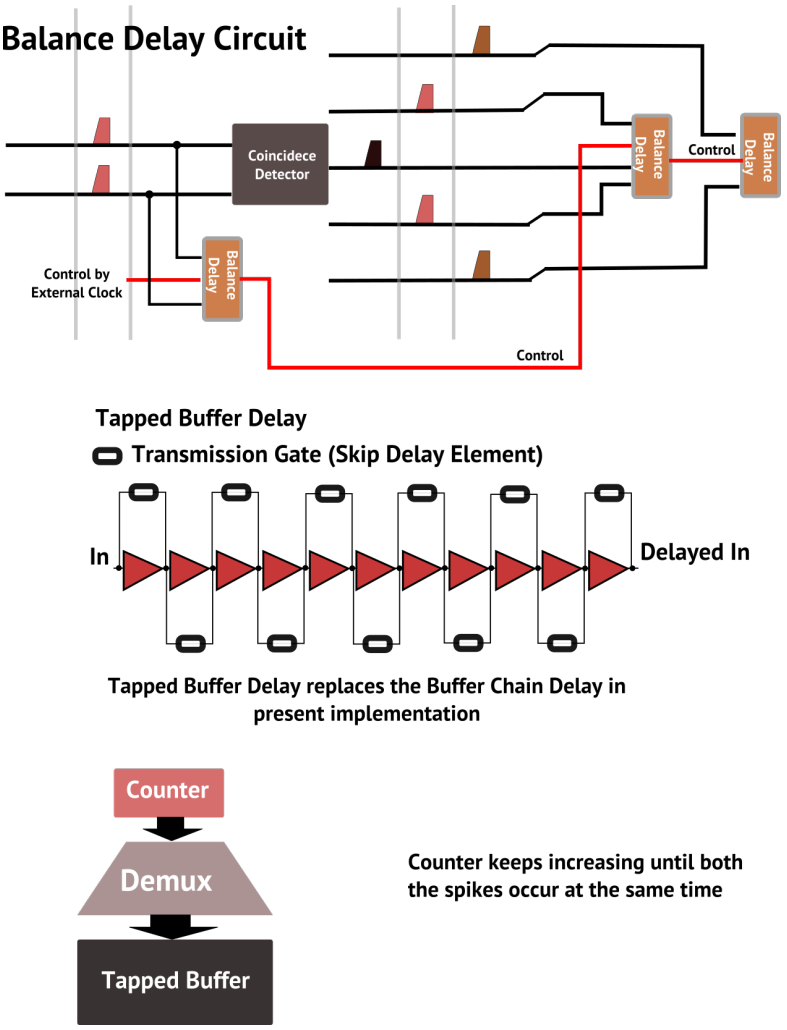


FIGURE 8.6: Proposed Adaptation

8.4.2 Feasibility for Local Adaptation

Local adaptation and self calibration are extremely important to solve the issues with technology scaling and produce more working chips per run. Figure 3.11 in Chapter 3 showed a tropotaxis based ADC architecture. This architecture can be used for self calibration of the SSDC.

The SSDC produces output for zero input, i.e., when both the pulses arrive at the same time to the SDC. Another interesting aspect is that, increasing the Gaussian kernel width to maximum, i.e., increasing the V_τ to 3.3V converts the coincidence detectors to constant delay elements. All the detectors produce outputs corresponding to the delays at this configuration. This property can be used for the adaptation of the delay chain for deviations.

The working of this concept is shown in Figure 8.6. The self-calibration starts at the center and the delays get adapted from their neighbor as the calibration moves from the

center outwards, similar to the blooming of a flower. The first calibration is controlled by an external clock as shown in the Figure 8.6.

This concept has been implemented on cadence and tested by a student [118].

8.5 Speed of SSDCα

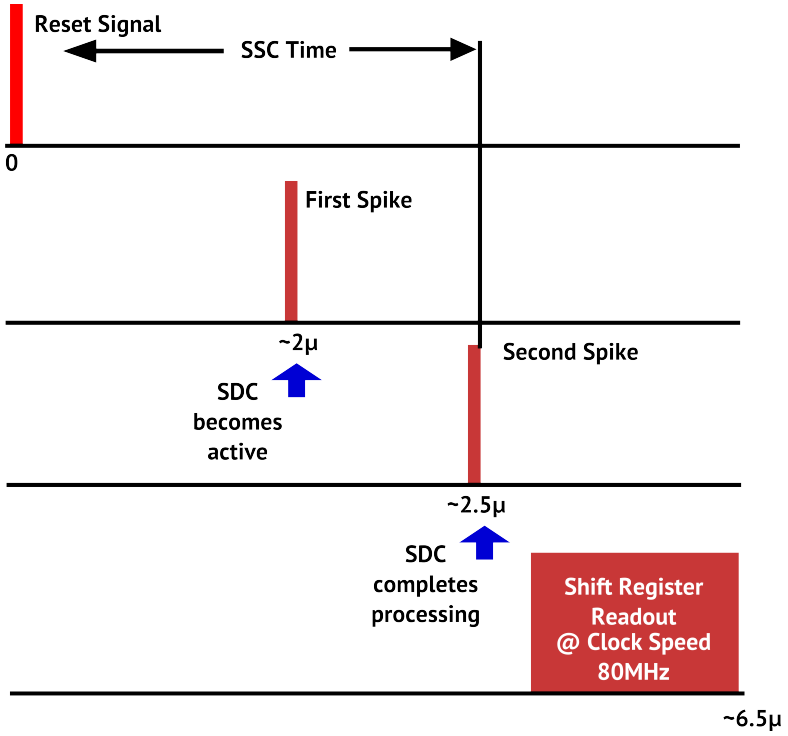


FIGURE 8.7: Image Visualizing the Time Taken by Different SSDC Modules

The SSDCα chip has a speed from DC to 150 kHz. This value might not give a clear view of the speed of its internals. The shift register, which is specific to this implementation takes 50% of the time. The speed of SSC is at 350kHz range. This is the time required to process the signal. The SDC starts processing at the first spike and finishes its processing at the second spike. It is running at a speed of 200 MHz in the chip. Since it is also possible to feed inputs directly to the SDC, this speed should be considered important. This provides a clear view of the capability of each module to obtain a better understanding of the SSDCα.

8.6 Area of SDC Module

From the Figure 8.1, we can observe that the SDC module occupies a very large area. This area can be reduced a lot by integrating the shift register or any readout circuit

as part of the SDC coincidence detector module, thereby reducing the interconnect area, which is currently 40% of the total area of the SDC module. However, the major advantage of the SDC module is that it scales with the technology and improves in resolution. The 350 nm ams hitkit technology is used just as a vehicle to demonstrate the basic functionality of the SSDC concept. The SDC should theoretically work better at smaller technology nodes like 28 nm and with new transistor types like 3D finfets from intel. This will be one of the goals of the future research on this topic.

8.7 Properties of SSDCα

Property	Value
Number of Transistors	28,200
Area	8.5 mm^2
Sampling Rate	DC- 150 kHz
Resolution	8-13 bit

FIGURE 8.8: Properties of SSDCα Chip

8.8 Discussion

This chapter discussed the design and features of the SSDCα chip. It was sent for manufacturing on Nov.16 2015 to Europractice and arrived in the first week of March 2016. The first prototype chip has many features, that make the information processing unique, robust, adaptive, and evolvable. The next steps would be the addition of local adaptation mechanisms, which can self calibrate as described in the subsection 8.4.2.

Chapter 9

Testing and Characterization of SSDCα

This chapter deals with testing of the SSDC chip that has been manufactured in AMS 350 nm technology. The bonding diagram and chip are shown in Figure 9.1 and Figure 9.2 respectively. Four of forty manufactured chips were tested and their results are discussed in this chapter

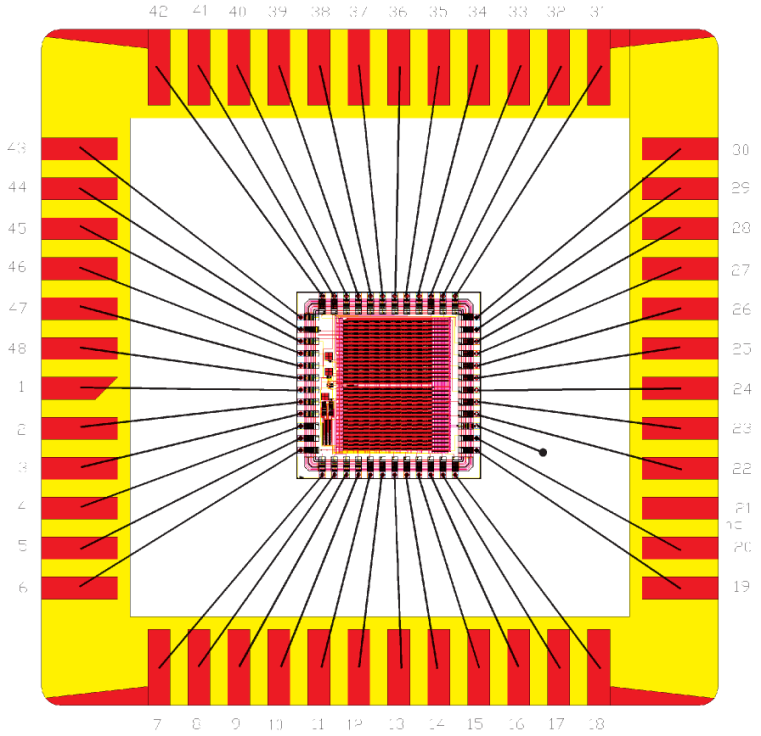


FIGURE 9.1: Bonding Diagram of SSDCα

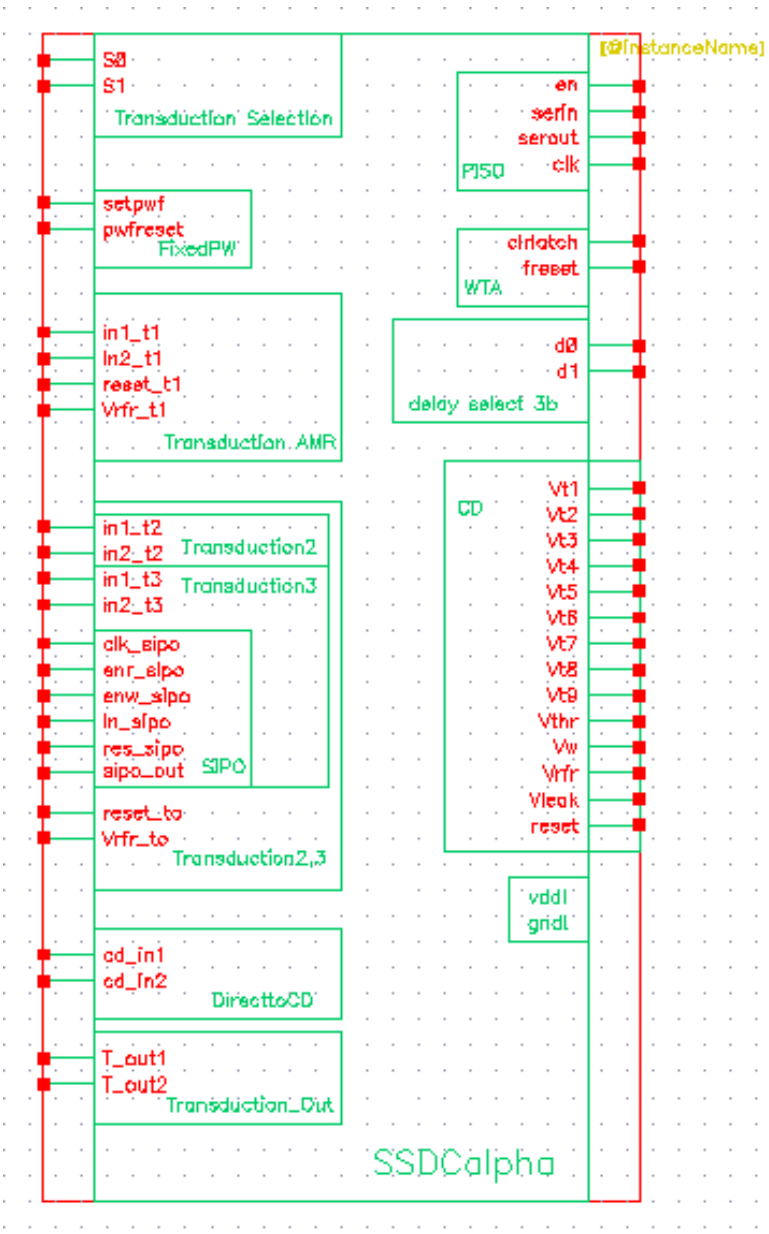


FIGURE 9.2: Symbol of the SSDCα Chip with Pin Information

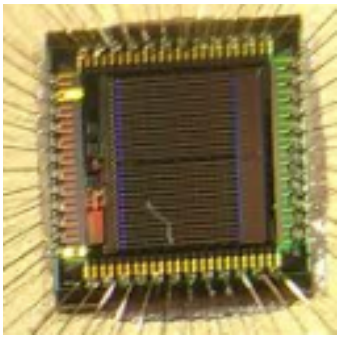


FIGURE 9.3: SSDCα Chip Image

9.1 Test Setup

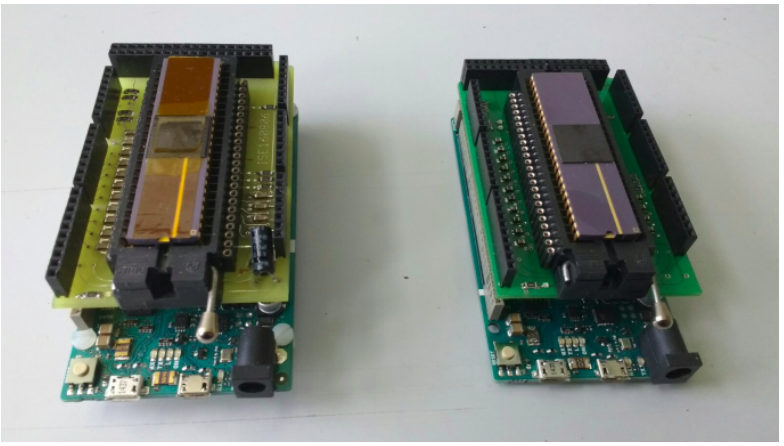


FIGURE 9.4: Arduino Duo with Shield for SSDCα Chip. Test Bench V1 (left), Test Bench V2 (right)

As discussed in the earlier chapters, SSDC has two important components. The SSC is the transduction element, which converts the incoming differential voltage into Time-Interval Code. This code is then used by the SSDC to convert it into I-of-N or Place Code. It is then vital to test these to modules separately to clearly understand how they perform, and their respective weaknesses to better develop the next generation of SSDC chips. In the first step of testing, one of the three SSC modules and the SSDC module will be used, so that the full flow can be tested. The first SSC module to be tested will be the SSC designed for resistive bridges, without the capability for reconfiguration.

Figure 9.2 shows the inputs required to run the SSDC. These inputs are control voltages that have to be generated by a DAC. There are 15 Analog voltages required to control the neurons and synapses. A shield was designed [119] for Arduino Due with required DACs. Among the 15 voltages, there are 3 that require voltages outside the range of the DAC. A Control-Voltage-Converter was used in the shield with an op-amp. Details of the shield design are provided in Appendix B. The shield plugged to the Arduino and the chip can be seen in Figure 9.4. The manufactured chip can be seen in Figure 9.3.

9.2 SSC Testing

9.2.1 Potentiometer

The SSC is tested with a setup shown in Figure 9.5. The two potentiometers are set to the desired diagonal voltages and the output is observed and measured in an oscilloscope.

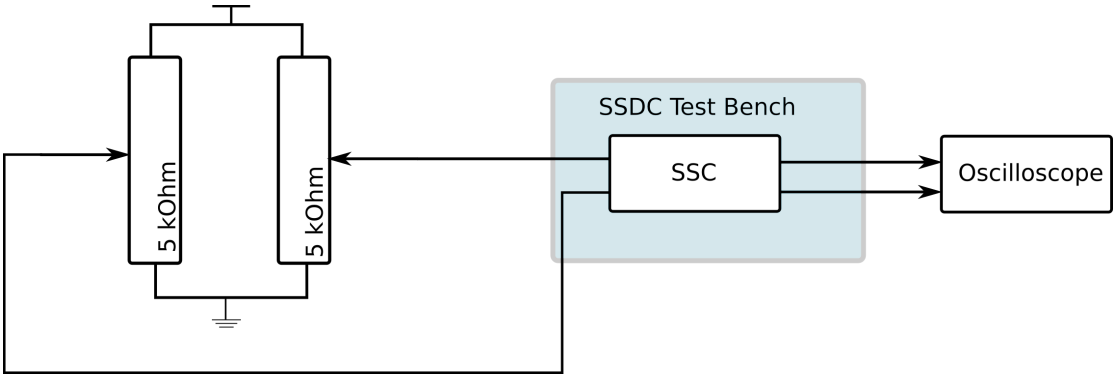


FIGURE 9.5: SSC Testing Setup

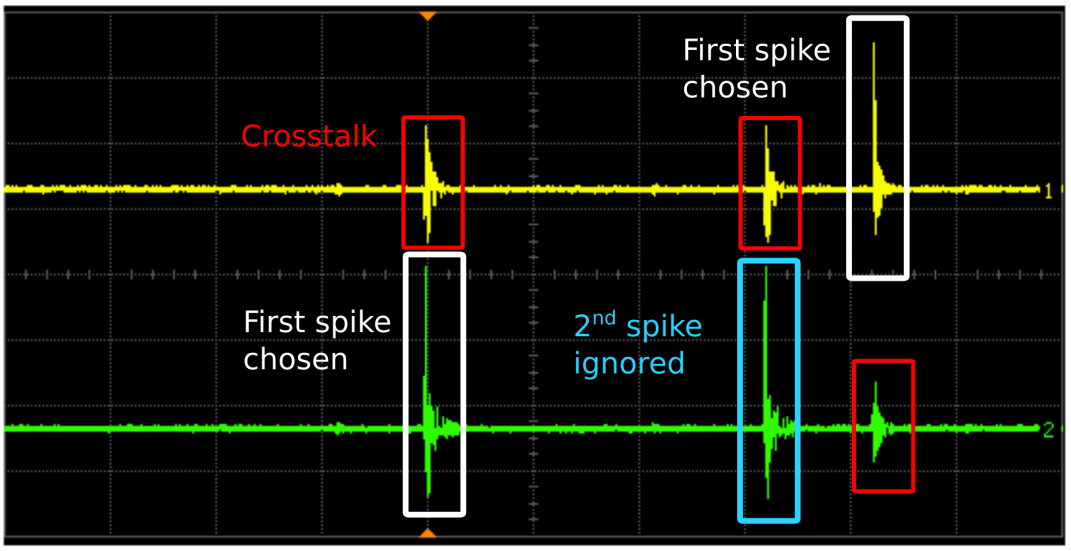


FIGURE 9.6: SSC Transduction Outputs at Oscilloscope with Explanations

The Figure 9.6 shows an example of time interval outputs taken from the oscilloscope. The crosstalk plays a major role in the outputs, as seen in the measurement results. There is also a large negative voltage that is being generated at the spikes. These problems have to be reduced in the next design. However, the time interval generation of the spikes works as intended.

A set of diagonal voltages were varied between 1.6 V and 1.7 V with ΔV of 0.02 V per step. The common mode is maintained at 1.65 V in this scenario. The results for this test are shown in Figure 9.7. All the oscilloscope outputs can be found in Appendix A. The results obtained from the chip deviate a little from the ideal expectations, but it is well within the expected range.

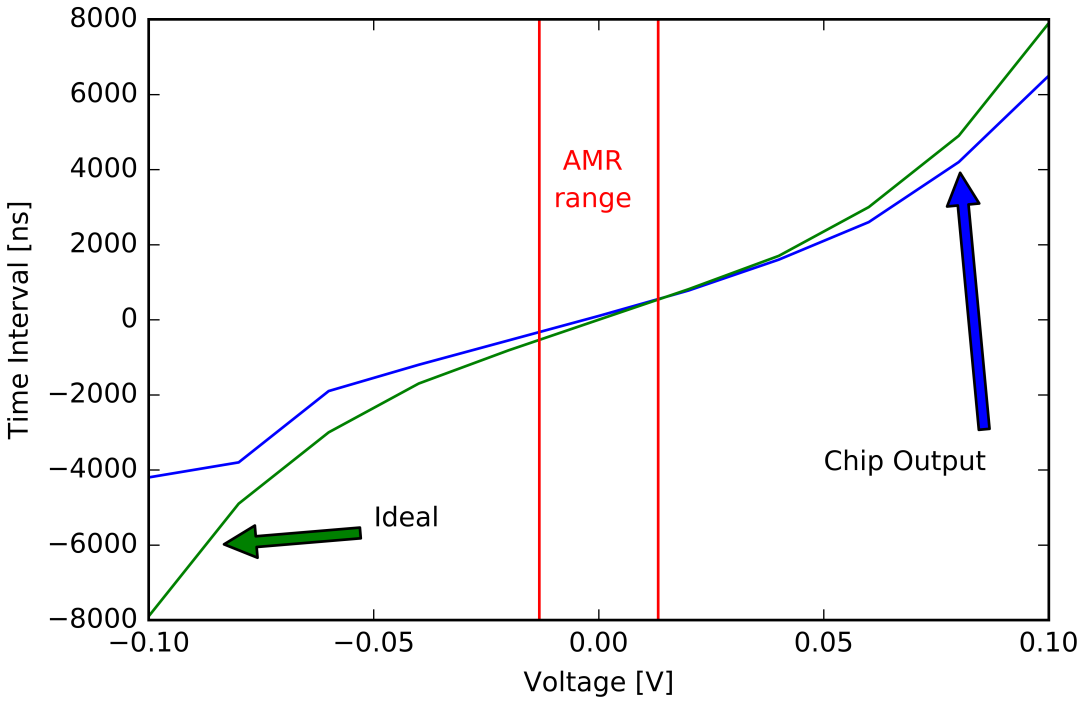


FIGURE 9.7: SSC Transduction Outputs, Ideal vs. Obtained (with ideal common mode voltage of 1.65 V)

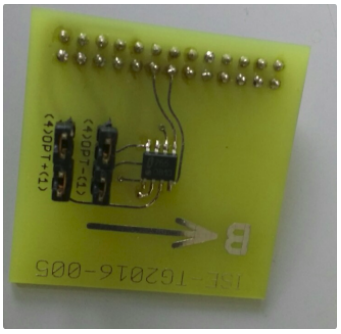


FIGURE 9.8: AMR Sensor PCB

9.2.2 AMR Sensor

In a second test the SSC was tested with an AMR sensor shown in Figure 9.8. The setup for the test is shown in Figure 9.9. A very basic test result with spike for presence and absence of a permanent magnet within the AMR's detection range is shown in Figure 9.10 and Figure 9.11. This test provides qualitative results of the SSC working with AMR sensors.

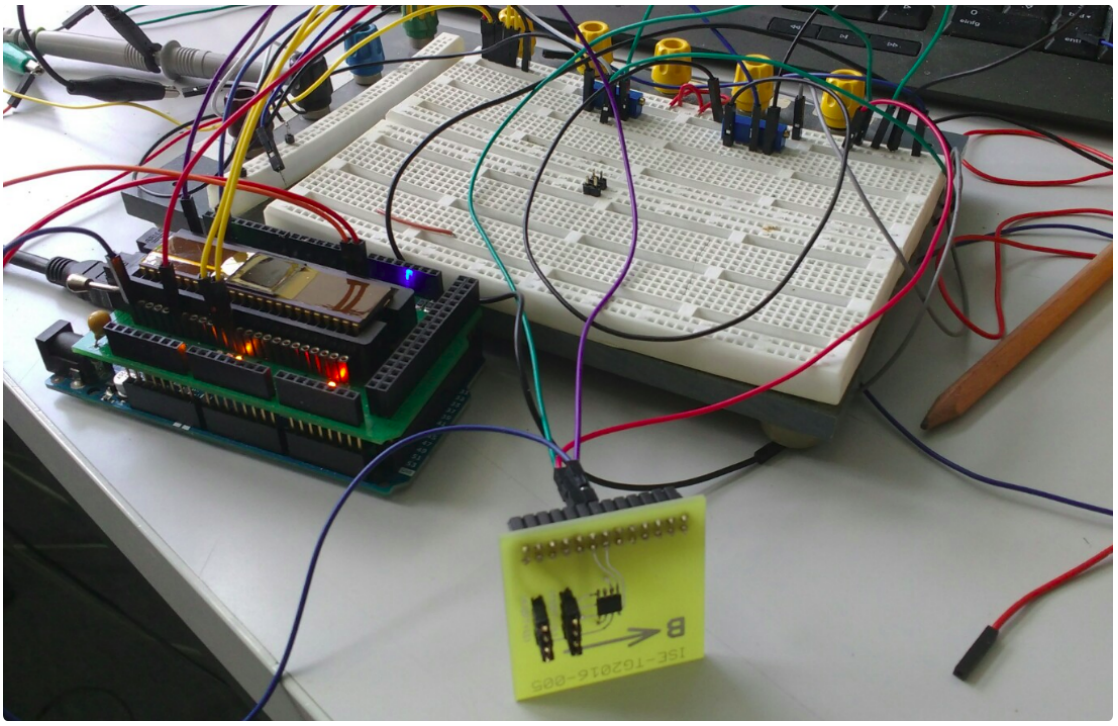


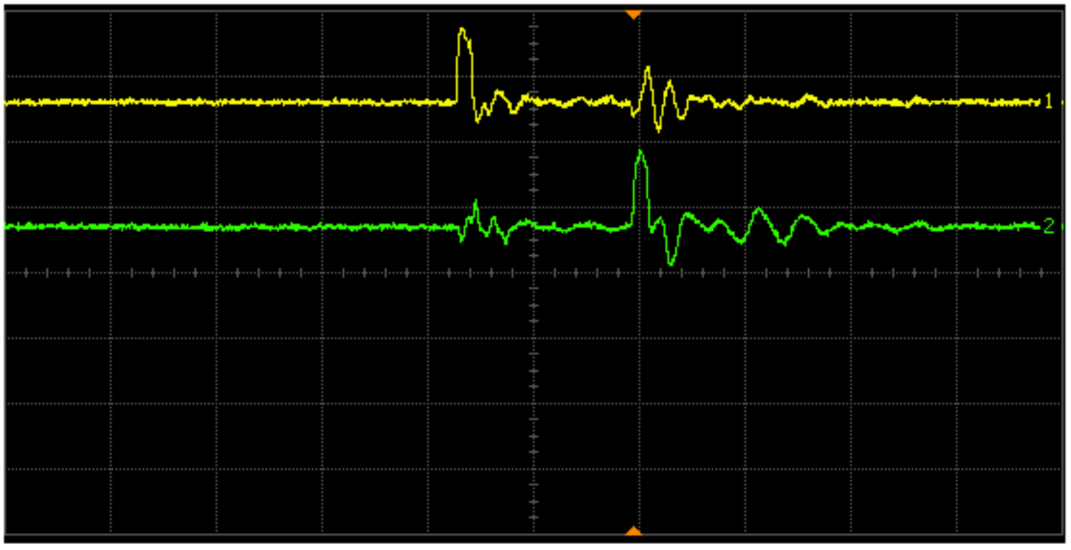
FIGURE 9.9: AMR Sensor Test Setup

9.2.3 Sensitivity to Common Mode

Another issue that is important for the SSC is the sensitivity to common mode. During the design the differential synapse was used to remove most of the common mode effects, however, there is still a small effect with a change in common mode in Figure 9.12. This has to be studied and improved in the next design phase. Figure 9.13 shows the comparison between ideal and chip test results for this scenario. All the measurement results by oscilloscope for these experiments can also be found in Appendix A.

9.2.4 Discussion

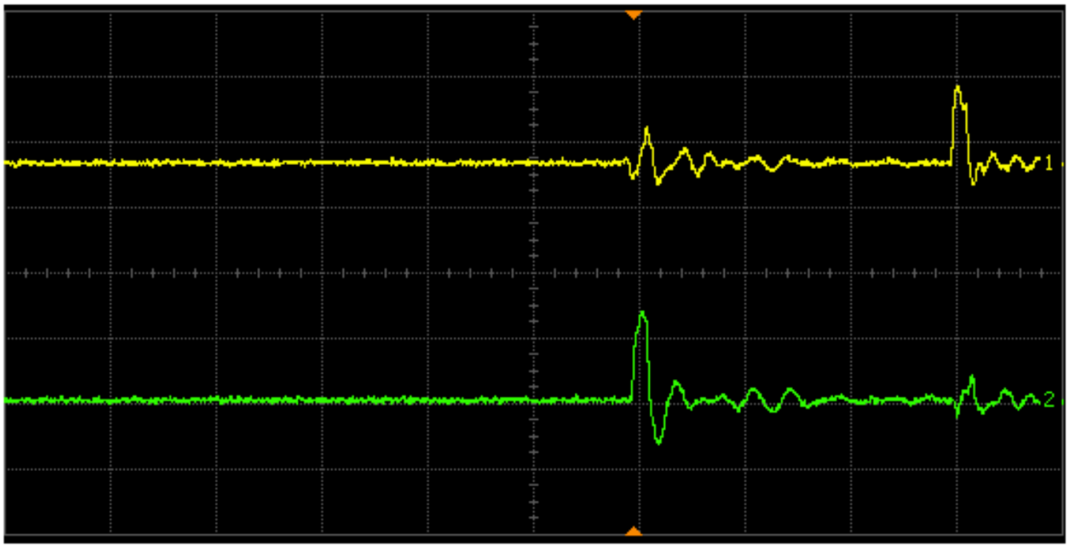
From the tests above, we can see that the SSC works as intended, although the problems with crosstalk and the high negative voltages has to be reduced. The reduction of common mode in these time based designs should also be improved in the next design.



Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 4.00 V/ Offset -10.58 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 4.00 V/ Offset -2.90 V Coupling DC Impedance 1M Ohm
Time base	Scale 100 ns/ Position -94.6400 ns Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Coupling DC Source channel 2 Trigger level 1.10 V Slope rising

Inputs
AMR Sensor
No Field
Applied

FIGURE 9.10: Output with No Field Applied to AMR Sensor



Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 4.00 V/ Offset -6.88 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 4.00 V/ Offset 7.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 100 ns/ Position -94.6400 ns Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Coupling DC Source channel 2 Trigger level 1.10 V Slope rising

Inputs
AMR Sensor
In
Magnetic Field

FIGURE 9.11: Output with Magnetic Field applied to AMR Sensor

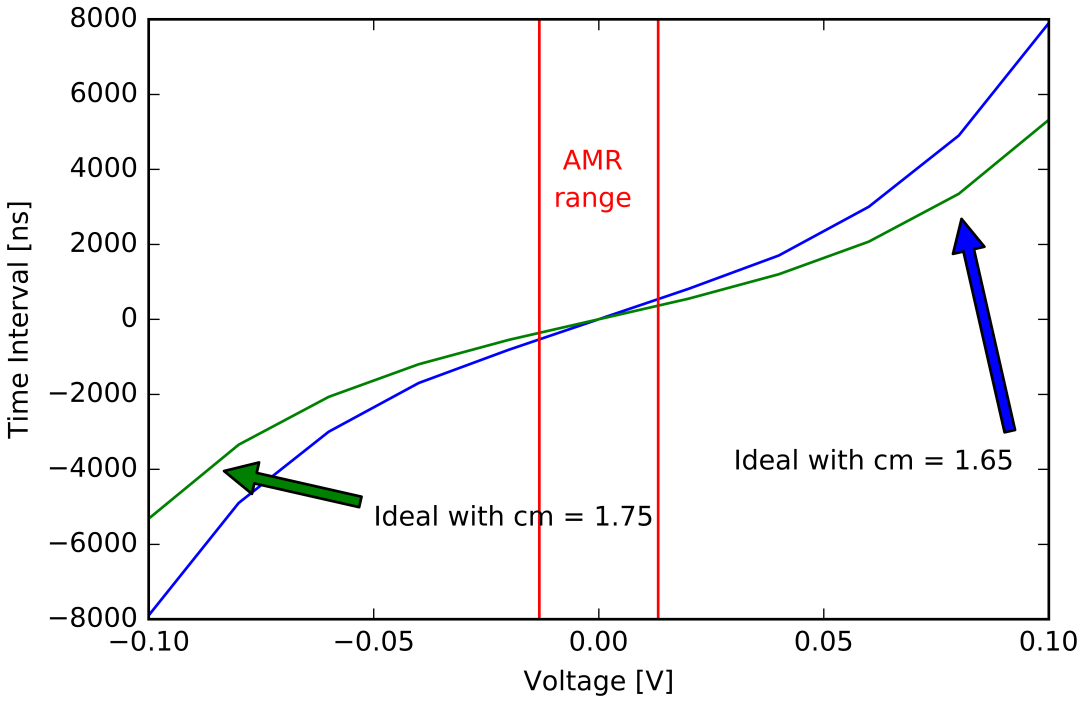


FIGURE 9.12: Common Mode Sensitivity

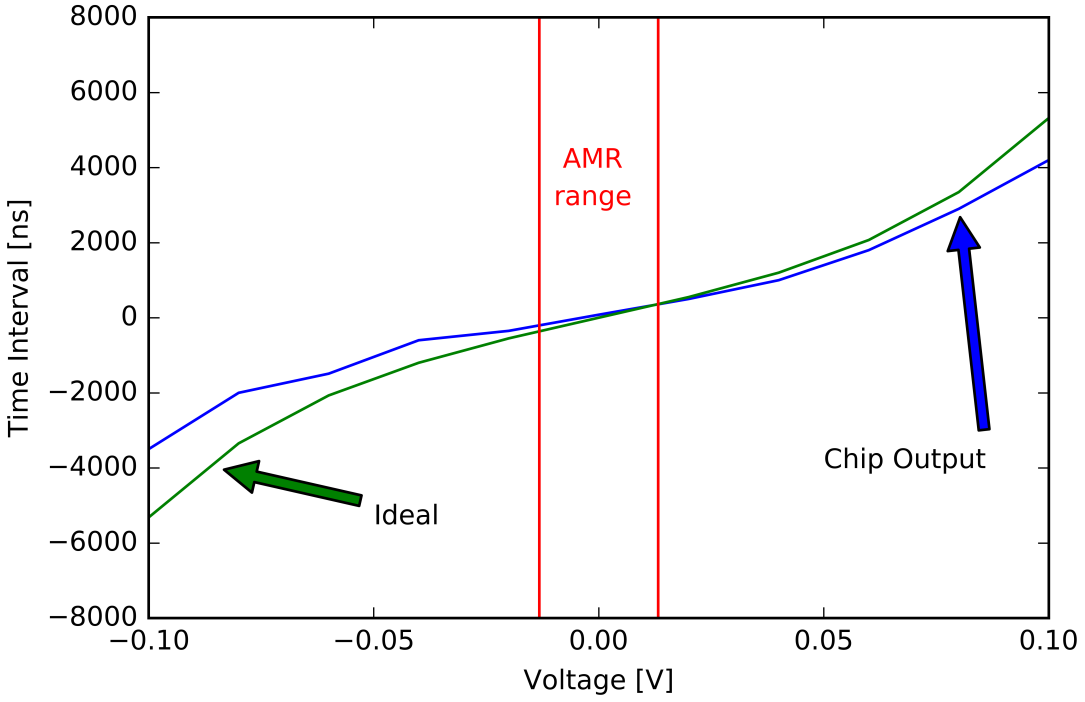


FIGURE 9.13: SSC Transduction Outputs, Ideal vs. Obtained with a Common Mode Voltage of 1.75 V

9.3 SDC Testing

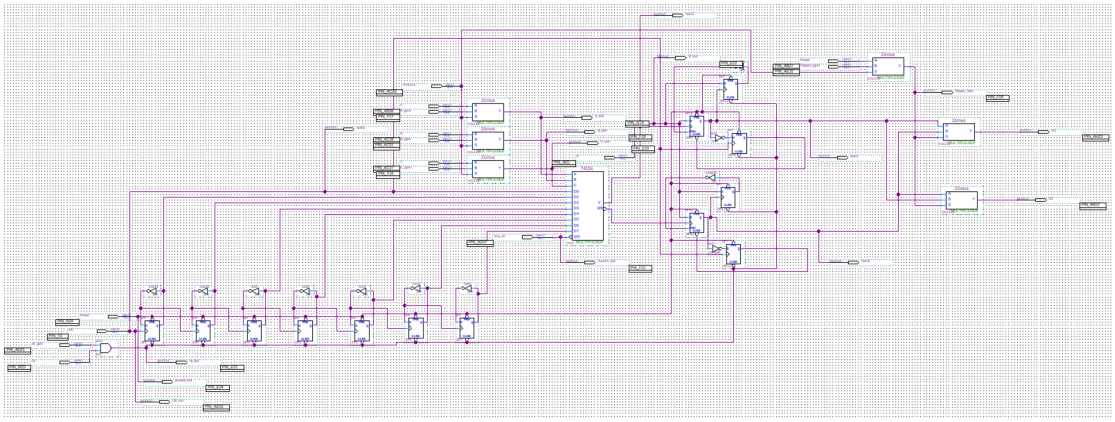


FIGURE 9.14: Quartus II Schematic of Time Interval Generator

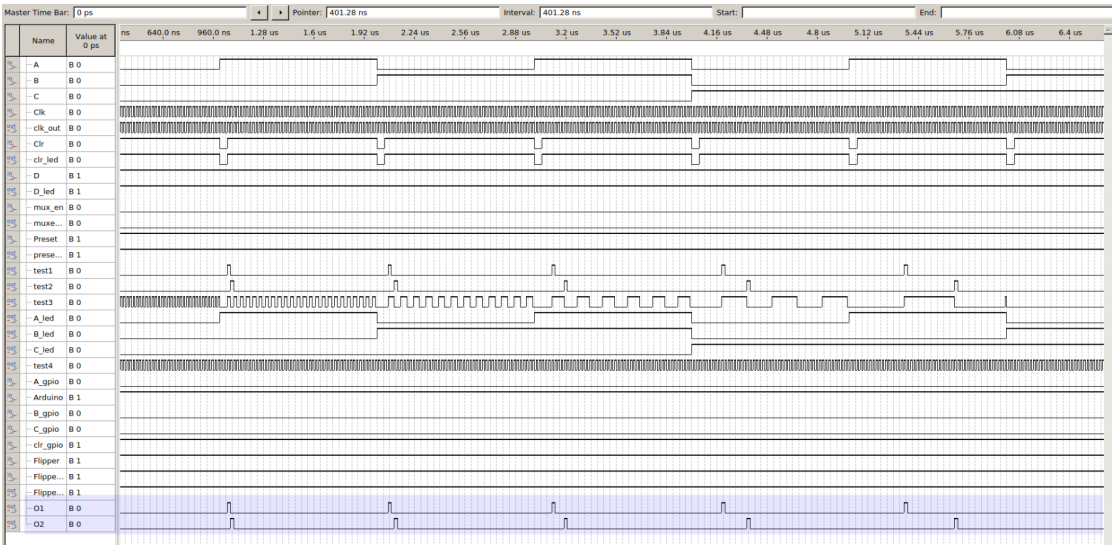


FIGURE 9.15: Timing Diagram of Time Interval Generator

The SDC is tested by using an FPGA with 50 MHz clock generating our time intervals. However, since frequency division is used to generate the intervals, it does not give the required resolution. Since this is the best available solution, it was used in this testing process. Figure 9.14 and Figure 9.15 show the schematic and timing diagram for the time interval generator. The outputs are recorder by the Arduino and plotted as shown in Figure 9.16. Figure 9.16 shows the spike outputs of the SDC. Here the SDC does not seem to work as intended, since there are at least 50 coincidence detectors spiking at the same time. This could be due to reduced sensitivity of the Time-Domain-WTA circuit, which was used only as a cost effective approach in the first design. Another problem is because the neurons are not self-adaptive or externally reconfigurable (individually) in the current chip. The reduced reconfigurability was a naive approach to reduce the manufacturing costs. This reduced the capability to control the coincidence detectors

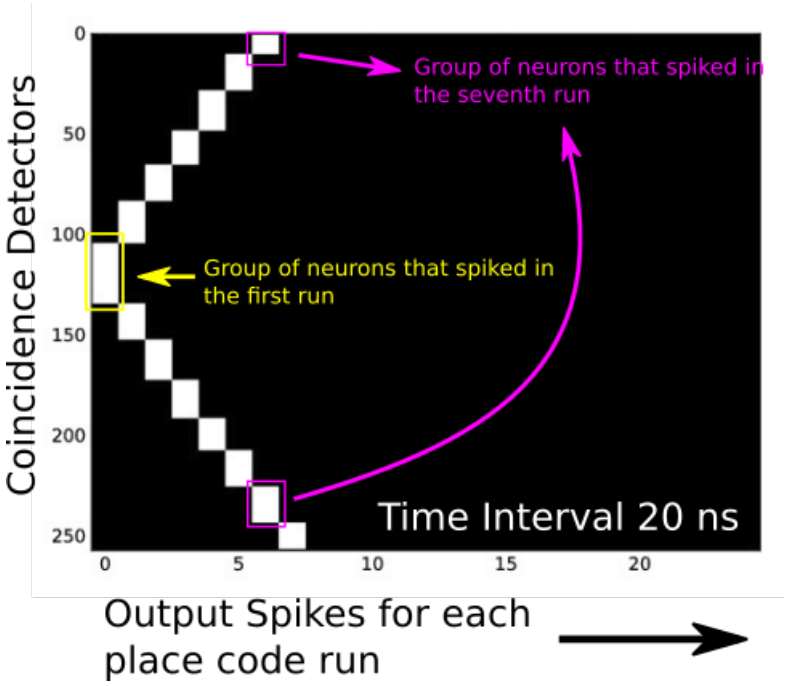


FIGURE 9.16: SDC Outputs Explanation

as needed. Although the capability of control voltages of individual neurons was absent, the available control voltages were tuned using particle swarm optimization [120], where the cost function was defined to increase the sensitivity of the SDC. Although these results are not as expected, using place coding is still effective for making the chip work.

Figure 9.17, Figure 9.18, and Figure 9.19 show the SDC outputs for the inputs generated by the FPGA. Although several neurons spike instead of a few as expected, the place code algorithm 1 can still be used by considering each group of spikes as a single spike. This simple heuristic can theoretically get at least 8 bits from the SDC outputs, although more complex heuristics with group sizes can be used to obtain more bits of information. Using the time interval generator, only 16 time intervals spaced unequally can be generated. So, this can only show the 4 bits of unequally spaced outputs for the 16 inputs. This also does not resolve to our required criteria, so it is impossible to obtain INL and DNL characteristics from these experiments.

9.3.1 Discussion

Although the SDC did not work as intended, an acceptable resolution was achieved using place coding. The place coding reduces the sampling rate with our current cost effective implementation as each "position" for the group of spikes requires one run, so to get 8 bits of data, the sampling rate would be 10 kHz. This proves that gradual degradation is a feature of this architecture. It also shows the importance of self adaptation in such

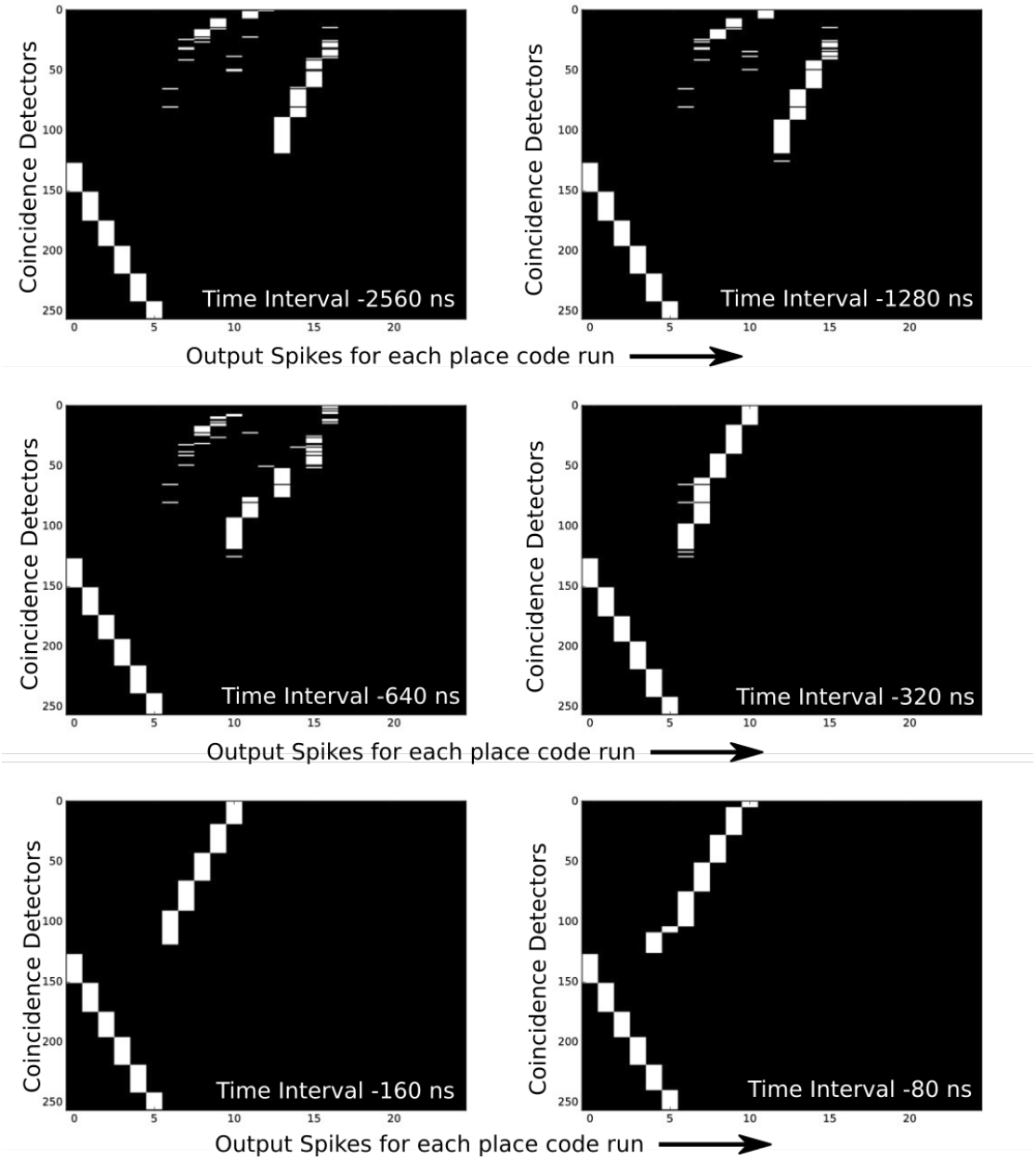


FIGURE 9.17: SDC Outputs

neural designs. Adaptation techniques should be implemented as described in previous chapters and more in the next design, to exploit all of the advantages of this architecture.

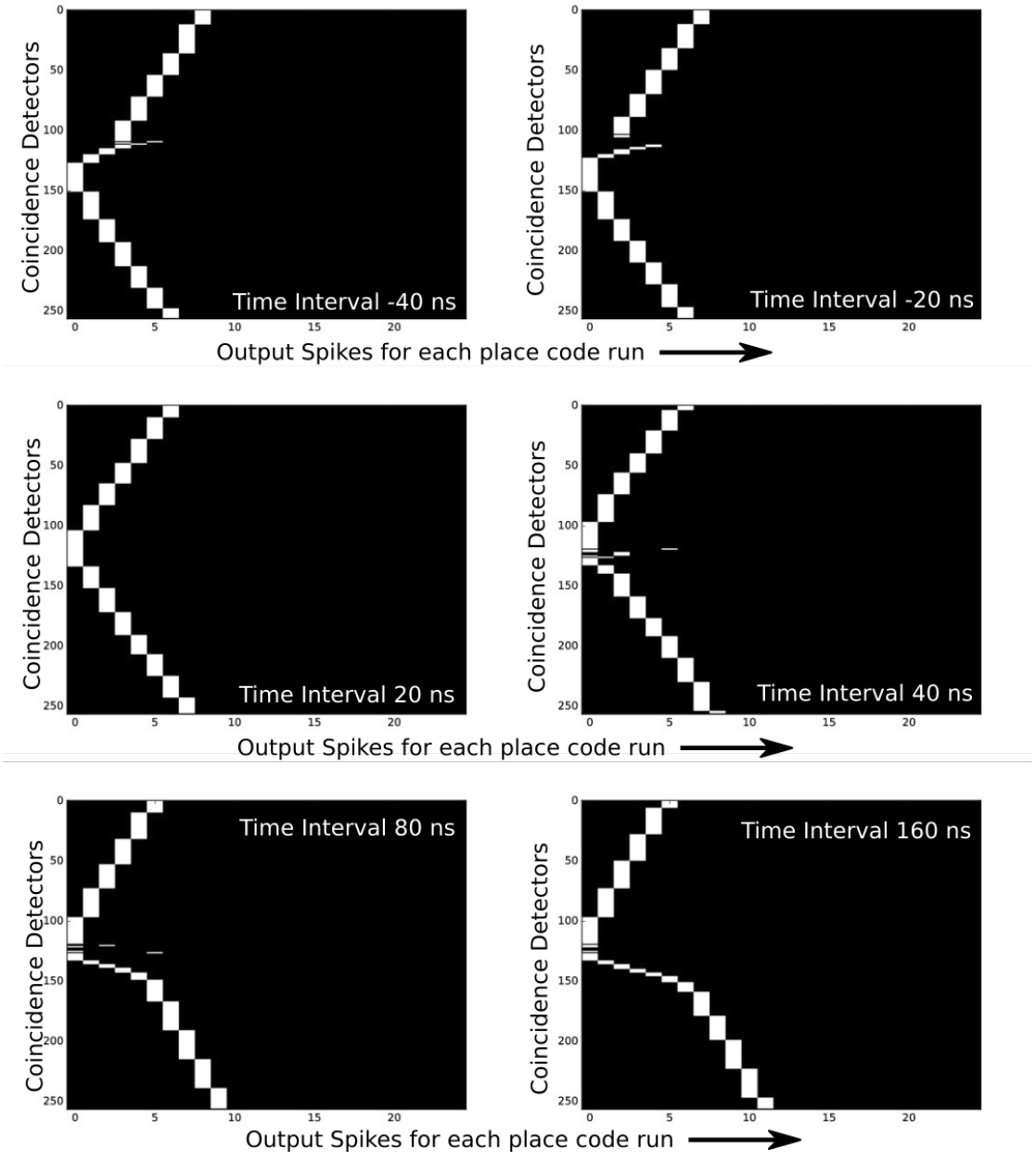


FIGURE 9.18: SDC Outputs

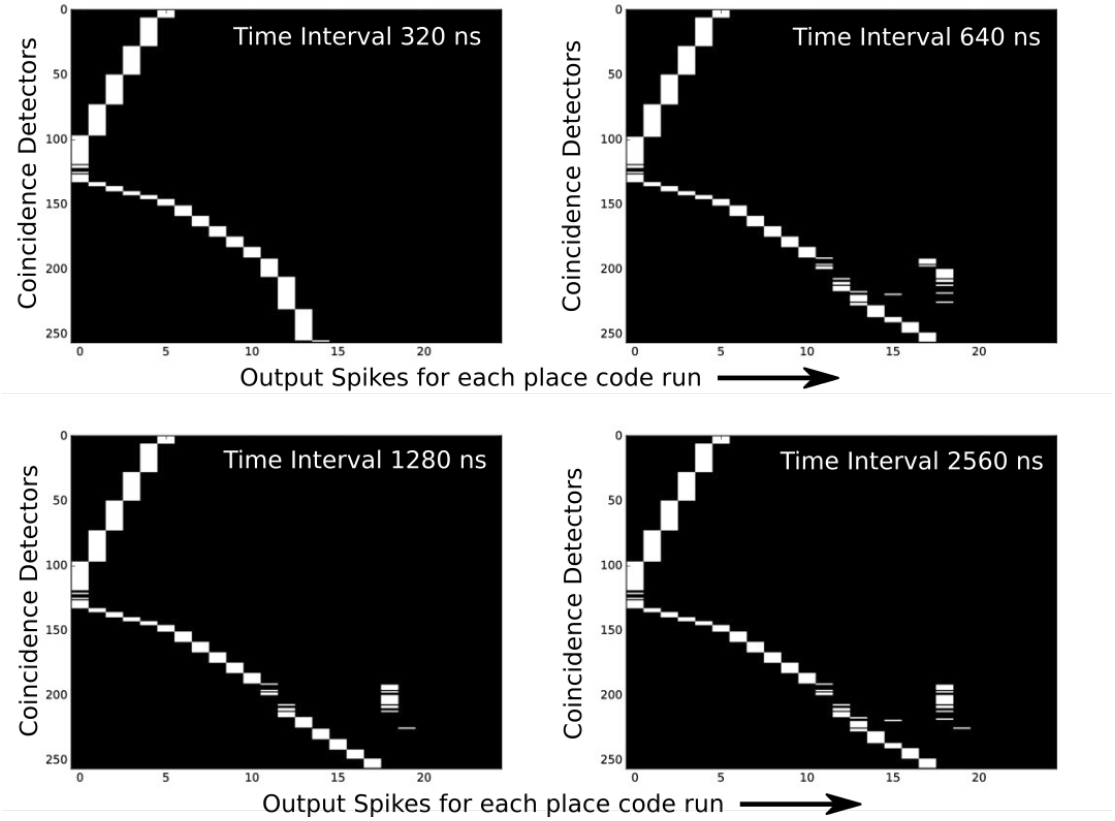


FIGURE 9.19: SDC Outputs

9.4 Summary

In this chapter, the fundamental flow of the SSDC concept has been tested. Four out of forty chips were used for testing. The first chip (SSDC001) did not work due to manufacturing defects. The other chips (SSDC002, SSDC003, SSDC 007) worked similar to the results shown earlier in the chapter. As described earlier, One of the three SSC modules on the Chip was thoroughly tested. The SSC measurement results fall in line with the expectations, while the SDC results were not as expected. While the results of the SDC were not as intended, using place coding techniques mentioned earlier, the working of the SDC for at least 4 bits was confirmed. The coincidence detection sub-circuit of the SDC was analyzed with Monte-Carlo simulations, however the complete SDC was not amenable to comprehensive statistical analysis, due to its rapidly increasing simulation time with the number of coincidence detection units. Although most of these variations, found in testing, are in the predicted range, they could and shall be reduced by reconfiguration or adaptation.

During the design phase, the neurons and synapses were designed to be individually reconfigurable, however, for the final design all the neurons were connected to the same control voltages, to reduce manufacturing costs. This reduced the capability for correcting for manufacturing variations to a large extent. This can be avoided by the addition of self adaptation to future designs.

As the first proof-of-principle chip, it is more than able to provide the base for future research. As discussed earlier, the issues with SSC and SDC have to be reduced and the focus should also be placed in designing the components in more aggressively scaled technologies.

Chapter 10

Conclusion

Rapid scaling of technologies is creating more demands from analog designs.This is pushing the designs towards digital domain, wherever possible. The recent rise of all-digital TDCs, which work with pulse width modulation techniques, is just one such example. The past research work of the ISE group, with spiking image sensor chips like LUCOS [2], which is a high dynamic range, highly resolving sensor has shown the effectiveness of spike processing. This motivated the direction of this work, which was to create a highly effective sensor signal conditioning system, which carries the promise to be robust to technology scaling.

This research work started with an overview of the various standard ADC structures including the ADCs designed in the latest 28 nm technology. This discussion was followed by an overview of reconfigurable ADCs in the amplitude domain. An overview of state-of-the-art TDC structures was provided and their advantages were discussed. This was then followed by an overview of neural ADC structures beginning from the amplitude domain neural ADCs to the neuromorphic spiking ADC structure.

A detailed study of the biological sensory systems, from an engineering perspective provided the baseline for the bio-mimetic ADC concept implemented in this work and several viable concepts for future research. Sensor concepts in humans and animals, which use intensity, frequency, edge detection, and localization are presented. Localization based on tropotaxis and acoustic localization in animals are shortlisted for the work, with 3 different architectures presented and the first architecture is the focus of the first prototype physical design, while the other architectures will be part of the future work. The proposed ADC consisted of two parts (1) Spike to Digital Conversion (SDC), (2) Sensor to Spike Conversion (SSC), which use spiking neurons as their building blocks.

The biological spiking neurons, the neural concepts, the state-of-the-art behavioral models, and the state-of-the-art neuromorphic electronics, which make use of these models has been presented. The shortlisted neuron and synapse were checked for behavioral similarity to the biological counterparts using cadence simulations. These modules were modified pragmatically to work at the limits of the technology, while having the minimum features required for the concept. These modules were carefully sized with the help of several runs of Monte-Carlo analysis to find the most robust sizes. The SDC and SSC were designed using these building blocks. Additionally a differential synapse was introduced in this work to improve the results of the SSC. A reconfigurable SSC was also designed in the process of moving towards a generic sensor interface.

The first prototype chip, SSDCα was designed as a proof of concept and hence the focus was on making a safe and inexpensive chip, where many features of the concept cannot be fully exploited yet. This experimental chip has several features, which include

- Three SSC modules
  - SSC tuned for AMR sensor module.
  - Reconfigurable SSC.
  - Experimental capacitive SSC.
- SSC outputs simultaneously to SDC and to output pins.
- Direct input pins to SDC.
- Reconfigurable delays to change the span of the SDC.

This prototype chip SSDCα was designed in 350 nm ams technology. This technology is used as a research vehicle for this thesis work. The chip has an area of 8.5 mm^2. It has a sampling rate from DC to 150 kHz. It has a resolution from 8-bit to 13-bit. It has 28,200 transistors on the chip.

The features which are not present in the present prototype are discussed, implemented and checked on simulation level. These include the local adaptation for delay correction, priority encoding techniques described in chapter 8. Many advantages of the concept will be observable in much smaller technologies, where conventional systems might fail.

The testing and characterization of the SSDC was shown in Chapter 9. The results are promising and also show the importance of adaptation in future designs. The complete implementation with additional techniques using Tropotaxis have been filed as a patent [121]. The current proof-of-principle chip is able to theoretically achieve at least 8 bits

at a sampling rate of 10 kHz. However, due to the limitations of our Time-Interval-Generator, we are able to confirm for only 4 bits of resolution.

Conceptual goals have all been fulfilled, with the exception of adaptation. The feasibility for local adaptation has been shown with promising results and further investigation is required for future work. This thesis work acts as a baseline, paving the way for R&D in a new direction. The chip design has used 350 nm ams hitkit as a vehicle to prove the functionality of the core concept. The concept can be easily ported to present aggressively-scaled-technologies and future technologies.

10.1 Novel Contributions of this Work

The key novelties of this work can be listed as follows

- Neuromorphic implementation of AD conversion as a localization problem based on sound localization and tropotaxis concepts found in nature.
- Coincidence detection with sparse place coding to enhance resolution.
- Graceful degradation without redundant elements, inherent robustness to noise, apt to perform in aggressively scaled technologies.
- Amenable to local adaptation and self-x features
- Technology agnostic architecture.

10.2 Comparison with State-of-the-Art

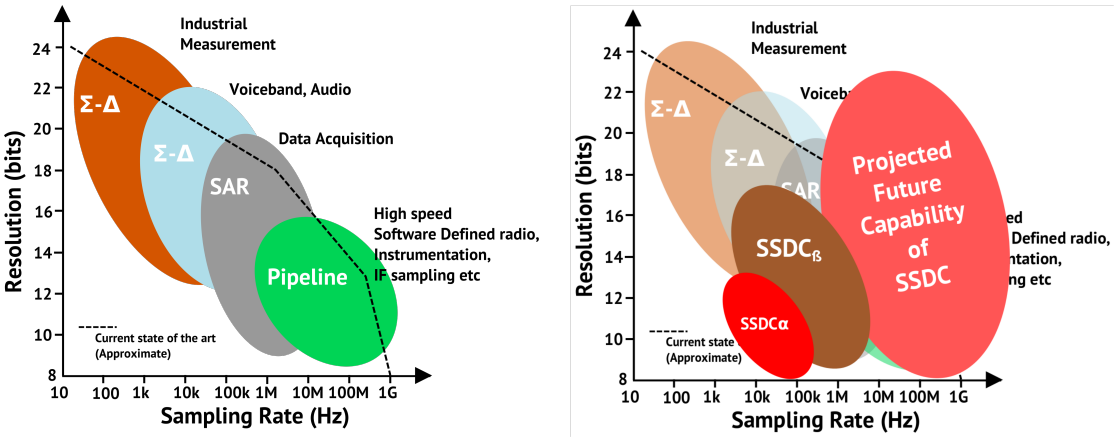


FIGURE 10.1: Positioning the Present and Future Work in Comparison to State-of-the-Art Conventional ADCs

Reference	Sarpshekar et. al.	Torikai et. al.	Cios et. al.	Schaik et. al.	Mayr et. al.	SSDCα
Domain	Amplitude & Time	Time	Time	Time	Time	Time
Coding Scheme	Time Interval	Time Interval Mapping	Population Code	Spike Rate and Phase	Encoder and Decoder weights	Time Interval & Sparse Place Codes
Parallel/ serial	Serial	Serial	Parallel	Parallel	Parallel	Parallel
Area & Power	Extremely Low Power and Area	No Data	No Data	No Data	High Power & Large Area	Average Power & Large Area
Reconfigurable/ adaptable	No	No	No	No	Rate & Resolution	Resolution, Kernel Width, Span etc.
Capable of Sensor Fusion	No	No data	Yes	No data	Yes	Yes
Resolution	8 bit	No data	No data	No data	Rate Dependent	8-13 bit
Implementation	0.18 µm subthreshold CMOS	Off the Shelf Components	C++ & XML	Off the Shelf Components	0.18 µm CMOS	0.35 µm CMOS
Capabillity for Mass Manufacturing	Yes	Yes	No Data	Yes	Requires calibration for every chip	Yes

TABLE 10.1: Comparison with Neuromorphic ADCs

Figure 10.1 shows the position of SSDCα, when compared to the state-of-the-art conventional ADCs. It also shows the projected positioning of the next version of the SSDC, SSDC$_{\beta}$, and the projected capability of the SSDC. Table 10.1 shows the comparison between neuromorphic ADCs and the SSDCα.

10.3 Future Work

- At present the SSDC is capable of handling differential and single-ended inputs, capacitive sensing, however, the SSDC can be developed as a universal generic sensor interface similar to [41].
- Investigation and implementation of local adaptation.
- Exploration of self-calibration, self-monitoring and self-repair capabilities of the concept.
- While the present application "magnetic localization" is chosen as the vehicle for proof of working of the SSDC, future work will focus on bringing these concepts to fields like IoT, Industrie 4.0, and Cyber Physical Systems.

Abbreviations

IoT	**I**nternet **o**f **T**hings
LIF	**L**eaky **I**ntegrate (and) **F**ire
DPI	**D**ifferential **P**air (and) **I**ntegrater
ISE	**I**ntegrated **SE**nsor Systems
AER	**A**ddress **E**vent **R**epresentation
RBF	**R**adial **B**asis **F**unctions
ADC	**A**nalog (to) **D**igital **C**onverter
SSDC	**S**ensor (to) **S**pike (to) **D**igital **C**onverter
SDC	**S**pike (to) **D**igital **C**onverter
SSC	**S**ensor (to) **S**pike **C**onverter
STDP	**S**pike **T**ime **D**ependent **P**lasticity
WTA	**W**inner **T**ake **A**ll
BGA	**B**all **G**rid **A**rray
TDC	**T**ime (to) **D**igital **C**onverter

Symbols

I_{SYN} **or** $I_{SYNAPSE}$	Synapse Current
V_{MEM}	Membrane Potential
V_{SPIKE}	Spike
V_{RFR}	Control Voltage: Refractory Period
V_{THR} **or** $V_{THRESHOLD}$	Threshold Voltage
V_{ADAPT}	Control Voltage: Spike Frequency Adaptation
C_{MEM}	Membrane Capacitance
V_{LEAK}	Membrane Potential
V_W	Control Voltage: Synapse Weight
V_{TC} **or** V_τ	Membrane Potential
C_{SYN}	Synapse Capacitance
V_{IN}	Input Voltage
V_{PRE}	Pre-Synaptic Voltage

Bibliography

[1] Education of Analog Design. `http://sscs.ieee.org/ieeetv.html`. Accessed: 2015-11-30.

[2] J. Döge, G. Schonfelder, G.T. Streil, and A. König. An HDR CMOS image sensor with spiking pixels, pixel-level ADC, and linear characteristics. *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, 49(2): 155–158, Feb 2002. ISSN 1057-7130. doi: 10.1109/TCSII.2002.1002518.

[3] C Bartolozzi and G Indiveri. Synaptic Dynamics in Analog VLSI. *Neural Computation*, 19(10):2581–2603, Oct 2007. ISSN 0899-7667. doi: 10.1162/neco.2007.19.10.2581.

[4] G. Indiveri. A low-power adaptive integrate-and-fire neuron circuit. In *Circuits and Systems, 2003. ISCAS '03. Proceedings of the 2003 International Symposium on*, volume 4, pages IV–820–IV–823 vol.4, May 2003. doi: 10.1109/ISCAS.2003.1206342.

[5] Kammara, A.C., and Hornberger,J. and König, A. Inherently Robust ADC Concepts with Biologically Inspired Spiking Neural Networks using Rank Order Coding - A Case Study. *AHMT*, 2010.

[6] A. König, P. Windirsch, and M. Glesner. ARAMYS - A Bit-Serial SIMD-Processor for Fast Parallel Nearest Neighbor Search and Associative Processing. In *Proceedings of the International Conference on Solid State Devices and Materials SSDM'94, Symposium Neurodevices and Neurochips*, pages 394–396. August 1994.

[7] S. Ito, S. Nishimura, H. Kobayashi, S. Uemori, Y. Tan, N. Takai, T. J. Yamaguchi, and K. Niitsu. Stochastic TDC architecture with self-calibration. In *2010 IEEE*

Asia Pacific Conference on Circuits and Systems, pages 1027–1030, Dec 2010. doi: 10.1109/APCCAS.2010.5774740.

[8] Martinelli, G. and Perfetti, R. Synthesis of feedforward neural analogue-digital convertors. *Circuits, Devices and Systems, IEE Proceedings G*, 138(5):567–574, Oct 1991. ISSN 0956-3768.

[9] R. Serrano-Gotarredona, M. Oster, P. Lichtsteiner, A. Linares-Barranco, R. Paz-Vicente, F. Gomez-Rodriguez, L. Camunas-Mesa, R. Berner, M. Rivas-Perez, T. Delbruck, Shih-Chii Liu, R. Douglas, P. Hafliger, G. Jimenez-Moreno, A.C. Ballcels, T. Serrano-Gotarredona, A.J. Acosta-Jimenez, and B. Linares-Barranco. CAVIAR: A 45k Neuron, 5M Synapse, 12G Connects/s AER Hardware Sensory, Processing, Learning, Actuating System for High-Speed Visual Object Recognition and Tracking. *Neural Networks, IEEE Transactions on*, 20(9):1417–1438, Sept 2009. ISSN 1045-9227. doi: 10.1109/TNN.2009.2023653.

[10] V. Chan, Shih-Chii Liu, and A. van Schaik. AER EAR: A Matched Silicon Cochlea Pair With Address Event Representation Interface. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 54(1):48–59, Jan 2007. ISSN 1549-8328. doi: 10.1109/TCSI.2006.887979.

[11] V. Kledrowetz and J. He. Basic Block of Pipelined ADC Design Requirements, Radioengineering. 2011(1):234–238, 2011.

[12] Demystifying Delta-Sigma ADCs. https://www.maximintegrated.com/en/app-notes/index.mvp/id/1870. Accessed 11-08-2017.

[13] Yueran Gao and Haibo Wang. A Reconfigurable ADC Circuit with Online-Testing Capability and Enhanced Fault Tolerance. In *Defect and Fault Tolerance in VLSI Systems, 2009. DFT '09. 24th IEEE International Symposium on*, pages 202–210, Oct 2009. doi: 10.1109/DFT.2009.31.

[14] Kameswaran Vengattaramane, Jonathan Borremans, Michiel Steyaert, and Jan Craninckx. A standard cell based all-digital Time-to-Digital Converter with reconfigurable resolution and on-line background calibration. In *ESSCIRC*, pages 275–278. IEEE, 2011. ISBN 978-1-4577-0703-2. URL `http://dblp.uni-trier.de/db/conf/esscirc/esscirc2011.html#VengattaramaneBSC11`.

[15] Which ADC Architecture Is Right for Your Application? http://www.analog.com/library/analogdialogue/archives/39-06/architecture.html. Accessed: 2017-07-21.

[16] Henzler Stephan. *Time-to-Digital Converters*, volume 29 of *Springer Series in Advanced Microelectronics*. Springer Netherlands, 1 edition, 2010. ISBN 978-90-481-8628-0.

[17] Tank, D. and Hopfield, J.J. Simple 'neural' optimization networks: An A/D converter, signal decision circuit, and a linear programming circuit. *Circuits and Systems, IEEE Transactions on*, 33(5):533–541, May 1986. ISSN 0098-4094. doi: 10.1109/TCS.1986.1085953.

[18] Lee, B.W. and Sheu, B.J. Design of a neural-based A/D converter using modified hopfield network. *Solid-State Circuits, IEEE Journal of*, 24(4):1129–1135, Aug 1989. ISSN 0018-9200. doi: 10.1109/4.34101.

[19] Avitabile, G. and Manetti, S. Some structures for neural based A/D conversion. *Electronics Letters*, 26(18):1516–1517, Aug 1990. ISSN 0013-5194. doi: 10.1049/el:19900973.

[20] Avitabile, G. and Forti, M. and Manetti, S. and Marini, M. On a class of nonsymmetrical neural networks with application to ADC. *Circuits and Systems, IEEE Transactions on*, 38(2):202–209, Feb 1991. ISSN 0098-4094. doi: 10.1109/31.68298.

[21] Daponte, P. and Grimaldi, D. and Michaeli, L. Gray code ADC based on an analog neural circuit. *Radioengineering*, 4(1):7–12, Apr 1995.

[22] Bernieri, A. and Daponte, P. and Grimaldi, D. ADC neural modeling. *Instrumentation and Measurement, IEEE Transactions on*, 45(2):627–633, Apr 1996. ISSN 0018-9456. doi: 10.1109/19.492800.

[23] Chanal, H. Hardware Implementation of an ADC Error Compensation Using Neural Networks. *International Workshop on ADC Modelling, Testing and Data Converter Analysis and Design and IEEE 2011 ADC Forum*, 2011.

[24] Yang, H.Y. and Sarpeshkar, R. A Bio-Inspired Ultra-Energy-Efficient Analog-to-Digital Converter for Biomedical Applications. *Circuits and Systems I: Regular*

Papers, IEEE Transactions on, 53(11):2349–2356, Nov 2006. ISSN 1549-8328. doi: 10.1109/TCSI.2006.884463.

[25] Torikai, H. and Tanaka, A. and Saito, T. . Artificial Spiking Neurons and Analog-to-Digital-to-Analog Conversion. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E91-A(6):1455–1462, 06 2008.

[26] Lovelace, J.J. and Rickard, J.T. and Cios, K.J. A spiking neural network alternative for the analog to digital converter. In *Neural Networks (IJCNN), The 2010 International Joint Conference on*, pages 1–8, July 2010. doi: 10.1109/IJCNN.2010.5596909.

[27] J. Tapson and A. van Schaik. An asynchronous parallel neuromorphic ADC architecture. In *Circuits and Systems (ISCAS), 2012 IEEE International Symposium on*, pages 2409–2412, May 2012. doi: 10.1109/ISCAS.2012.6271783.

[28] Christian G Mayr, Johannes Partzsch, Marko Noack, and Rene Schüffny. Configurable Analog-Digital Conversion Using the Neural Engineering Framework. *Frontiers in Neuroscience*, 8(201), 2014. ISSN 1662-453X. doi: 10.3389/fnins.2014.00201. URL `http://www.frontiersin.org/neuromorphic_engineering/10.3389/fnins.2014.00201/abstract`.

[29] Christophe Micheyl, Paul R. Schrater, and Andrew J. Oxenham. Auditory Frequency and Intensity Discrimination Explained Using a Cortical Population Rate Code. *PLoS Comput Biol*, 9(11):e1003336, 11 2013. doi: 10.1371/journal.pcbi.1003336. URL `http://dx.doi.org/10.1371%2Fjournal.pcbi.1003336`.

[30] Wikipedia. Receptive field, 2015. URL `https://en.wikipedia.org/wiki/Receptive_field`. Accessed 13-10-2015.

[31] Colour Vision. `http://hyperphysics.phy-astr.gsu.edu/hbase/vision/colcon.html`. Accessed: 2015-10-13.

[32] Wim De Mulder, Steven Bethard, and Marie-Francine Moens. A survey on the application of recurrent neural networks to statistical language modeling. *Computer Speech and Language*, 30(1):61 – 98, 2015. ISSN 0885-2308. doi: http://dx.doi.org/10.1016/j.csl.2014.09.005. URL `http://www.sciencedirect.com/science/article/pii/S088523081400093X`.

[33] Wulfram Gerstner and Werner Kistler. *Spiking Neuron Models: An Introduction*. Cambridge University Press, New York, NY, USA, 2002. ISBN 0521890799.

[34] E.M. Izhikevich. Which model to use for cortical spiking neurons? *Neural Networks, IEEE Transactions on*, 15(5):1063–1070, Sept 2004. ISSN 1045-9227. doi: 10.1109/TNN.2004.832719.

[35] J.H.B. Wijekoon and P. Dudek. Simple Analogue VLSI Circuit of a Cortical Neuron. In *Electronics, Circuits and Systems, 2006. ICECS '06. 13th IEEE International Conference on*, pages 1344–1347, Dec 2006. doi: 10.1109/ICECS.2006.379731.

[36] A. Joubert, B. Belhadj, O. Temam, and R. Heliot. Hardware spiking neurons design: Analog or digital? In *Neural Networks (IJCNN), The 2012 International Joint Conference on*, pages 1–5, June 2012. doi: 10.1109/IJCNN.2012.6252600.

[37] M.A. Nahmias, B.J. Shastri, A.N. Tait, and P.R. Prucnal. A Leaky Integrate-and-Fire Laser Neuron for Ultrafast Cognitive Computing. *Selected Topics in Quantum Electronics, IEEE Journal of*, 19(5):1–12, Sept 2013. ISSN 1077-260X. doi: 10.1109/JSTQE.2013.2257700.

[38] The FACETS Project. `https://facets.kip.uni-heidelberg.de/`. Accessed: 2015-02-17.

[39] Rae Silver, Kwabena Boahen, Sten Grillner, Nancy Kopell, and Kathie L. Olsen. Symposium Neurotech for Neuroscience: Unifying Concepts, Organizing Principles, and Emerging Tools, 2007.

[40] K.A. Boahen. Point-to-point connectivity between neuromorphic chips using address events. *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, 47(5):416–434, May 2000. ISSN 1057-7130. doi: 10.1109/82.842110.

[41] Robert Freier. *Ein universelles und dynamisch rekonfigurierbares Interface für eingebettete und intelligente Multi-Sensor-Systeme mit Self-x Eigenschaften, Forschungsberichte Integrierte Sensorsysteme Band 04, Herausgegeben von Prof. Dr.-Ing. Andreas König, ISBN 978-3-943995-81-7, 2015*. PhD thesis.

[42] Dennis Groben, Kittikhun Thongpull, Abhaya Chandra Kammara, and Andreas König. Neural Virtual Sensors for Adaptive Magnetic Localization of Autonomous Dataloggers, Advances in Artificial Neural Systems. 2014(17), 2014.

[43] S. Le Tual, P.N. Singh, C. Curis, and P. Dautriche. A 20GHz-BW 6b 10GS/s 32mW time-interleaved SAR ADC with Master TH in 28nm UTBB FDSOI technology. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International*, pages 382–383, Feb 2014. doi: 10.1109/ISSCC.2014.6757479.

[44] B. Verbruggen, K. Deguchi, B. Malki, and J. Craninckx. A 70 dB SNDR 200 MS/s 2.3 mW dynamic pipelined SAR ADC in 28nm digital CMOS. In *VLSI Circuits Digest of Technical Papers, 2014 Symposium on*, pages 1–2, June 2014. doi: 10.1109/VLSIC.2014.6858451.

[45] M. Inerfield, A. Kamath, Feng Su, J. Hu, Yu Xinyu, V. Fong, O. Alnaggar, Fang Lin, and T. Kwan. An 11.5-ENOB 100-MS/s 8mW dual-reference SAR ADC in 28nm CMOS. In *VLSI Circuits Digest of Technical Papers, 2014 Symposium on*, pages 1–2, June 2014. doi: 10.1109/VLSIC.2014.6858453.

[46] B. Verbruggen, M. Iriguchi, M. de la Guia Solaz, G. Glorieux, K. Deguchi, B. Malki, and J. Craninckx. A 2.1 mW 11b 410 MS/s dynamic pipelined SAR ADC with background calibration in 28nm digital CMOS. In *VLSI Circuits (VLSIC), 2013 Symposium on*, pages C268–C269, June 2013.

[47] Jiangfeng Wu, A. Chou, Cheng-Hsun Yang, Yen Ding, Yen-Jen Ko, Sha-Ting Lin, Wenbo Liu, Chi-Ming Hsiao, Ming-Hung Hsieh, Chun-Cheng Huang, Juo-Jung Hung, Kwang Young Kim, M. Le, Tianwei Li, Wei-Ta Shih, A. Shrivastava, Yau-Cheng Yang, Chun-Ying Chen, and Hung-Sen Huang. A 5.4GS/s 12b 500mW pipeline ADC in 28nm CMOS. In *VLSI Circuits (VLSIC), 2013 Symposium on*, pages C92–C93, June 2013.

[48] F. van der Goes, C. Ward, S. Astgimath, H. Yan, J. Riley, J. Mulder, S. Wang, and K. Bult. 11.4 A 1.5mW 68dB SNDR 80MS/s interleaved SAR-assisted pipelined ADC in 28nm CMOS. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International*, pages 200–201, Feb 2014. doi: 10.1109/ISSCC.2014.6757399.

[49] M. Brandolini, Young Shin, K. Raviprakash, Tao Wang, Rong Wu, H.M. Geddada, Yen-Jen Ko, Yen Ding, Chun-Sheng Huang, Wei-Ta Shin, Ming-Hung Hsieh, Wei-Te Chou, Tianwei Li, A. Shrivastava, Yi-Chun Chen, Juo-Jung Hung, G. Cusmai, Jiangfeng Wu, M.M. Zhang, G. Unruh, A. Venes, Hung Sen Huang, and Chun-Ying Chen. 26.6 A 5GS/S 150mW 10b SHA-less pipelined/SAR hybrid ADC in 28nm CMOS. In *Solid- State Circuits Conference - (ISSCC), 2015 IEEE International*, pages 1–3, Feb 2015. doi: 10.1109/ISSCC.2015.7063129.

[50] D. Bellasi, L. Bettini, T. Burger, Qiuting Huang, C. Benkeser, and C. Studer. A 1.9 GS/s 4-bit sub-Nyquist flash ADC for 3.8 GHz compressive spectrum sensing in 28 nm CMOS. In *Circuits and Systems (MWSCAS), 2014 IEEE 57th International Midwest Symposium on*, pages 101–104, Aug 2014. doi: 10.1109/MWSCAS.2014.6908362.

[51] Carlos Azeredo-Leme, Pedro Figueiredo, and Manuel Mota. White paper: Scaling ADC architectures for mobile & multimedia socs at 28-nm & beyond. Technical report, Synopsis, January 2013.

[52] S. Haenzsche, S. Hoppner, G. Ellguth, and R. Schuffny. A 12-b 4-MS/s SAR ADC With Configurable Redundancy in 28-nm CMOS Technology. *Circuits and Systems II: Express Briefs, IEEE Transactions on*, 61(11):835–839, Nov 2014. ISSN 1549-7747. doi: 10.1109/TCSII.2014.2345301.

[53] Tuan-Vu Cao, S. Aunet, and T. Ytterdal. A 9-bit 50MS/s asynchronous SAR ADC in 28nm CMOS. In *NORCHIP, 2012*, pages 1–6, Nov 2012. doi: 10.1109/NORCHP.2012.6403105.

[54] Heng Zhang. *High performance RF and baseband A/D interface for multi-standard/wideband applications*. PhD thesis, Texas A&M University, Dec 2010.

[55] Tom Simonite. Thinking in Silicon. `http://www.technologyreview.com/featuredstory/522476/thinking-in-silicon/`. Accessed: 2015-10-13.

[56] Jose M Cruz-Albrecht, Timothy Derosier, and Narayan Srinivasa. A scalable neural chip with synaptic electronics using CMOS integrated memristors. *Nanotechnology*, 24(38):384011, 2013. URL `http://stacks.iop.org/0957-4484/24/i=38/a=384011`.

[57] The Board: Neurogrid. `https://web.stanford.edu/group/brainsinsilicon/neurogrid.html`. Accessed: 2015-10-13.

[58] Andrew S. Cassidy, Paul Merolla, John V. Arthur, Steve K. Esser, Bryan Jackson, Rodrigo Alvarez-icaza, Pallab Datta, Jun Sawada, Theodore M. Wong, Vitaly Feldman, Arnon Amir, Daniel Ben dayan Rubin, Emmett Mcquinn, William P. Risk, and Dharmendra S. Modha. Cognitive computing building block: A versatile and efficient digital neuron model for neurosynaptic cores. In *in International Joint Conference on Neural Networks (IJCNN). IEEE*, 2013.

[59] Andrea Calimera, Enrico Macii, and Massimo Poncino. The human brain project and neuromorphic computing. *Funct Neurol*, 28:191–196, 2013.

[60] AMR Sensors. `http://www.sensitec.com/english/products/magnetic-field/aff755.html`. Accessed: 2015-11-30.

[61] Tsensor Roadmap. `http://www.tsensorssummit.org/Resources/Why%20TSensors%20Roadmap.pdf`. Accessed: 2015-02-17.

[62] Industrie 4.0. `http://www.plattform-i40.de/`. Accessed: 2015-02-17.

[63] Smart Manufacturing Leadership Coalition. `https://smartmanufacturingcoalition.org/`. Accessed: 2015-02-17.

[64] Gargini, P. A Roadmap to Success: 2013 ITRS Update, accessed 11-05-2014, currently replaced by itrs 2015.

[65] Multigate Devices. `http://en.wikipedia.org/wiki/Multigate_device`. Accessed: 2015-02-17.

[66] C. Menolfi and Qiuting Huang. A low-noise CMOS instrumentation amplifier for thermoelectric infrared detectors. *Solid-State Circuits, IEEE Journal of*, 32(7): 968–976, Jul 1997. ISSN 0018-9200. doi: 10.1109/4.597287.

[67] Tsung-Heng Tsai, P.J. Hurst, and S.H. Lewis. Correction of Mismatches in a Time-Interleaved Analog-to-Digital Converter in an Adaptively Equalized Digital Communication Receiver. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 56(2):307–319, Feb 2009. ISSN 1549-8328. doi: 10.1109/TCSI.2008.2002114.

[68] Seung-Hoon Lee and Bang-Sup Song. Digital-domain calibration of multistep analog-to-digital converters. *Solid-State Circuits, IEEE Journal of*, 27(12):1679–1688, Dec 1992. ISSN 0018-9200. doi: 10.1109/4.173093.

[69] ACAM Time to Digital Converters. `http://www.acam.de/products/time-to-digital-converters/`. Accessed: 2015-04-21.

[70] AMA Sensor Trends. `http://www.ama-sensorik.de/fileadmin/Pubikationen/AMA_Study_Sensor_Trends%5B1%5D.pdf`. Accessed: 2015-02-17.

[71] The Human Brain Project. `https://www.humanbrainproject.eu/`. Accessed: 2015-02-17.

[72] Brain Scales Project. `http://brainscales.kip.uni-heidelberg.de/`. Accessed: 2015-02-17.

[73] Voltage References. http://www.analog.com/media/en/training-seminars/tutorials/MT-087.pdf. Accessed: 2015-07-21.

[74] Peter Messiha Mehanny Tawdross. *Bio-Inspired Circuit Sizing and Trimming Methods for Dynamically Reconfigurable Sensor Electronics in Industrial Embedded Systems, Forschungsberichte Integrierte Sensorsysteme Band 01, Herausgegeben von Prof. Dr.-Ing. Andreas König, ISBN 978-3-941438-62-0, 2007.* PhD thesis. URL `http://nbn-resolving.de/urn/resolver.pl?urn:nbn:de:hbz:386-kluedo-21516`.

[75] J. Ryckaert, M. Verhelst, M. Badaroglu, S. D'Amico, V. De Heyn, C. Desset, P. Nuzzo, B. Van Poucke, P. Wambacq, A. Baschirotto, W. Dehaene, and G. Van der Plas. A Ultra-Wideband Receiver for Low Data-Rate Communication. *Solid-State Circuits, IEEE Journal of*, 42(11):2515–2527, Nov 2007. ISSN 0018-9200. doi: 10.1109/JSSC.2007.907195.

[76] J. Craninckx and G. Van der Plas. A 65fJ/Conversion-Step 0-to-50MS/s 0-to-0.7mW 9b Charge-Sharing SAR ADC in 90nm Digital CMOS. In *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, pages 246–600, Feb 2007. doi: 10.1109/ISSCC.2007.373386.

[77] N. Verma and A.P. Chandrakasan. An Ultra Low Energy 12-bit Rate-Resolution Scalable SAR ADC for Wireless Sensor Nodes. *Solid-State Circuits, IEEE Journal of*, 42(6):1196–1205, June 2007. ISSN 0018-9200. doi: 10.1109/JSSC.2007.897157.

[78] T.N. Andersen, B. Hernes, A. Briskemyr, F. Telsto, J. Bjornsen, T.E. Bonnerud, and O. Moldsvor. A cost-efficient high-speed 12-bit pipeline ADC in 0.18- μm digital CMOS. *Solid-State Circuits, IEEE Journal of*, 40(7):1506–1513, July 2005. ISSN 0018-9200. doi: 10.1109/JSSC.2005.847519.

[79] I. Ahmed and D.A. Johns. A 50-MS/s (35 mW) to 1-kS/s (15 μW) power scalable 10-bit pipelined ADC using rapid power-on opamps and minimal bias current variation. *Solid-State Circuits, IEEE Journal of*, 40(12):2446–2455, Dec 2005. ISSN 0018-9200. doi: 10.1109/JSSC.2005.856289.

[80] H. Matsui, M. Ueda, M. Daito, and K. Iizuka. A 14bit digitally self-calibrated pipelined ADC with adaptive bias optimization for arbitrary speeds up to 40MS/s. In *VLSI Circuits, 2005. Digest of Technical Papers. 2005 Symposium on*, pages 330–333, June 2005. doi: 10.1109/VLSIC.2005.1469398.

[81] G. Geelen, E. Paulus, D. Simanjuntak, H. Pastoor, and R. Verlinden. A 90nm CMOS 1.2V 10b power and speed programmable pipelined ADC with 0.5pJ/conversion-step. In *Solid-State Circuits Conference, 2006. ISSCC 2006. Digest of Technical Papers. IEEE International*, pages 782–791, Feb 2006. doi: 10.1109/ISSCC.2006.1696118.

[82] Cheng-Chung Hsu, Chen-Chih Huang, Ying-Hsi Lin, Chao-Cheng Lee, Z. Soe, T. Aytur, and Ran-Hong Yan. A 7b 1.1GS/s Reconfigurable Time-Interleaved ADC in 90nm CMOS. In *VLSI Circuits, 2007 IEEE Symposium on*, pages 66–67, June 2007. doi: 10.1109/VLSIC.2007.4342768.

[83] Bo Xia, A. Valdes-Garcia, and E. Sanchez-Sinencio. A 10-bit 44-MS/s 20-mW configurable time-interleaved pipeline ADC for a dual-mode 802.11b/Bluetooth receiver. *Solid-State Circuits, IEEE Journal of*, 41(3):530–539, March 2006. ISSN 0018-9200. doi: 10.1109/JSSC.2005.864131.

[84] M. Anderson, K. Norling, A. Dreyfert, and J. Yuan. A reconfigurable pipelined ADC in 180 nm CMOS. In *VLSI Circuits, 2005. Digest of Technical Papers. 2005 Symposium on*, pages 326–329, June 2005. doi: 10.1109/VLSIC.2005.1469397.

[85] G. Gielen and E. Goris. Reconfigurable front-end architectures and A/D converters for flexible wireless transceivers for 4G radios. In *Emerging Technologies: Circuits and Systems for 4G Mobile Wireless Communications, 2005. ETW '05. 2005 IEEE 7th CAS Symposium on*, pages 13–18, June 2005. doi: 10.1109/EMRTW.2005.195670.

[86] K. Gulati and Hae-Seung Lee. A low-power reconfigurable analog-to-digital converter. In *Solid-State Circuits Conference, 2001. Digest of Technical Papers. ISSCC. 2001 IEEE International*, pages 54–55, Feb 2001. doi: 10.1109/ISSCC.2001.912543.

[87] Thomas Christen, T. Burger, and Qiuting Huang. A 0.13 μm CMOS EDGE/UMTS/WLAN Tri-Mode Delta; Sigma; ADC with -92dB THD. In *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, pages 240–599, Feb 2007. doi: 10.1109/ISSCC.2007.373383.

[88] S. Ouzounov, R. van Veldhoven, C. Bastiaansen, K. Vongehr, R. van Wegberg, G. Geelen, L. Breems, and A. van Roermund. A 1.2V 121-Mode CT Modulator for Wireless Receivers in 90nm CMOS. In *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, pages 242–600, Feb 2007. doi: 10.1109/ISSCC.2007.373384.

[89] B. Putter. A 5th-order CT/DT Multi-Mode Modulator. In *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, pages 244–245, Feb 2007. doi: 10.1109/ISSCC.2007.373385.

[90] P. Malla, H. Lakdawala, K. Kornegay, and K. Soumyanath. A 28mW Spectrum-Sensing Reconfigurable 20MHz 72dB-SNR 70dB-SNDR DT ADC for 802.11n/WiMAX Receivers. In *Solid-State Circuits Conference, 2008. ISSCC 2008. Digest of Technical Papers. IEEE International*, pages 496–631, Feb 2008. doi: 10.1109/ISSCC.2008.4523274.

[91] L. Bos, Gerd Vandersteen, J. Ryckaert, P. Rombouts, Y. Rolain, and G. Van der Plas. A multirate 3.4-to-6.8mW 85-to-66dB DR GSM/bluetooth/UMTS cascade DT M in 90nm digital CMOS. In *Solid-State Circuits Conference - Digest of Technical Papers, 2009. ISSCC 2009. IEEE International*, pages 176–177,177a, Feb 2009. doi: 10.1109/ISSCC.2009.4977365.

[92] J. J. Hopfield. Neurocomputing: Foundations of Research. chapter Neural Networks and Physical Systems with Emergent Collective Computational Abilities, pages 457–464. MIT Press, Cambridge, MA, USA, 1988. ISBN 0-262-01097-6. URL `http://dl.acm.org/citation.cfm?id=65669.104422`.

[93] Yuh, J.-D. and Newcomb, R.W. A multilevel neural network for A/D conversion. *Neural Networks, IEEE Transactions on*, 4(3):470–483, May 1993. ISSN 1045-9227. doi: 10.1109/72.217190.

[94] Cong-Kha Pham and Tanaka, M. and Shono, K. A simple neural-based A/D converter employing CMOS inverters. In *Neural Networks, 1994. IEEE World Congress on Computational Intelligence., 1994 IEEE International Conference on*, volume 4, pages 2093–2096 vol.4, Jun 1994. doi: 10.1109/ICNN.1994.374537.

[95] Cennamo, F. and Daponte, P. and Grimaldi, D. and Loizzo, E. Testing the performances of neural A/D converter. In *Instrumentation and Measurement Technology Conference, 1994. IMTC/94. Conference Proceedings. 10th Anniversary. Advanced Technologies in I amp; M., 1994 IEEE*, pages 899–902 vol.2, May 1994. doi: 10.1109/IMTC.1994.351963.

[96] Daponte, P. and Grimaldi, D. and Michaeli, L. A full neural Gray-code-based ADC. *Instrumentation and Measurement, IEEE Transactions on*, 45(2):634–639, Apr 1996. ISSN 0018-9456. doi: 10.1109/19.492801.

[97] McDonnell,M.D. and Abbott, D. and Pearce, C.E. Neural mechanisms for analog-to-digital conversion. *SPIE Proceedings*, 5275:278–286, March 2004.

[98] Terrence C. Stewart. A Technical Overview of the Neural Engineering Framework. Technical report, Centre for Theoretical Neuroscience, 2012.

[99] The Rods and Cones of the Human Eye. `http://hyperphysics.phy-astr.gsu.edu/hbase/vision/rodcone.html`. Accessed: 2015-11-30.

[100] D. H. Hubel and T. N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, 160 (1):106, January 1962. URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1359523/`.

[101] Wikipedia. Color Vision, 2015. URL `https://en.wikipedia.org/wiki/Color_vision`. Accessed: 13-10-2015.

[102] RS Johansson and I Birznieks. First spikes in ensembles of human tactile afferents code complex spatial fingertip events. *Nature Neuroscience*, 7(5):170–7, 2004-02-01 00:00:00.001. ISSN 1097-6256. doi: 10.1038/nn1177.

[103] Simon Thorpe and Jacques Gautrais. Rank Order Coding. In James M. Bower, editor, *Computational Neuroscience*, pages 113–118. Springer US, 1998. ISBN 978-1-4613-7190-8. doi: 10.1007/978-1-4615-4831-7_19. URL `http://dx.doi.org/10.1007/978-1-4615-4831-7_19`.

[104] J. Okada. Cockroach antennae. 4(10):6842, 2009. revision 149368.

[105] Go Ashida and Catherine E. Carr. Sound localization: Jeffress and beyond. *Current opinion in neurobiology*, 21(5):745, October 2011. doi: 10.1016/j.conb.2011.05.008. URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3192259/`.

[106] Tristram Wyatt. *Pheromones and animal behaviour communication by smell and taste*. Cambridge University Press, Cambridge, UK New York, 2003. ISBN 9780521485265.

[107] Jeremy Hsu. How IBM Got Brainlike Efficiency From the TrueNorth Chip. `http://spectrum.ieee.org/computing/hardware/how-ibm-got-brainlike-efficiency-from-the-truenorth-chip`. Accessed: 2015-10-13.

[108] ABSYNTH - Analog Design Automation. `https://www.eit.uni-kl.de/koenig/gemeinsame_seiten/projects/ABSYNTH.html`. Accessed: 2015-11-30.

[109] A. C. Kammara and A König. Contributions to Integrated Adaptive Spike Coded Sensor Signal Conditioning and Digital Conversion in Neural Architecture. In *Proc. of Sensors and Measuring Systems 2014; 17. ITG/GMA Symposium*, page 1–6. 2014.

[110] A. C. Kammara and A. König. *Increasing the Resolution of an Integrated Adaptive Spike Coded Sensor to Digital Conversion Neuro-Circuit by an Enhanced Place*

Coding Layer. In: Tagungsband des XXVIII Messtechnisches Symposium des Arbeitskreises der Hochschullehrer für Messtechnik e. V. (AHMT), pp., Saarbrücken, Sept, 2014.

[111] ARAMYS. Personal commnunication with Prof. Andreas König, Location: Gebaude 12, Raum 447, 29-06-2015, .

[112] Philipp Häfliger and ElinJørgensen Aasebø. A Rank Encoder: Adaptive Analog to Digital Conversion Exploiting Time Domain Spike Signal Processing. *Analog Integrated Circuits and Signal Processing*, 40(1):39–51, 2004. ISSN 0925-1030. doi: 10.1023/B:ALOG.0000031432.51988.17. URL `http://dx.doi.org/10.1023/B%3AALOG.0000031432.51988.17`.

[113] Abhaya Chandra K., Beverly Bwali, Juergen Hornberger, and A. König. *Revisiting Neural A/D-Conversion, ITG-Fachgruppensitzung "Mikroelektronik Neuronaler Netze", CITEC, Universität Bielefeld, Bielefeld.* 25, November 2010.

[114] A. König, A. C. Kammara D. Groben, and K. Thongpull. Dynamically Reconfigurable Integrated Sensor Electronics for Magnetic Localisation of Distributed Autonomous Sensor Nodes. In *AMA Conferences SENSOR 2013*, pages 334–339. 2013.

[115] A. C. Kammara and A. König. *Robust ADCs for Dependable Integrated Measurement Systems based on Adaptive Neuromorphie Spiking Realization.* In: Tagungsband des XXVIII Messtechnisches Symposium des Arbeitskreises der Hochschullehrer für Messtechnik e. V. (AHMT), pp., Ilmenau, Sept, 2015.

[116] Differential Current Source. Personal commnunication with Prof. Andreas König, Location: Gebaude 12, Raum 447, 25-08-2015, .

[117] Design Rule Check 350 nm CMOS c35. `http://asic.ams.com/cgi-sbin/hk_download_documents.cgi?all`. Accessed: 2015-11-30.

[118] Jan Lappas. Investigation of Self-Calibration Options for a Spiking ADC. 2015, Neurocomputing Project, ISE, TU Kaiserslautern.

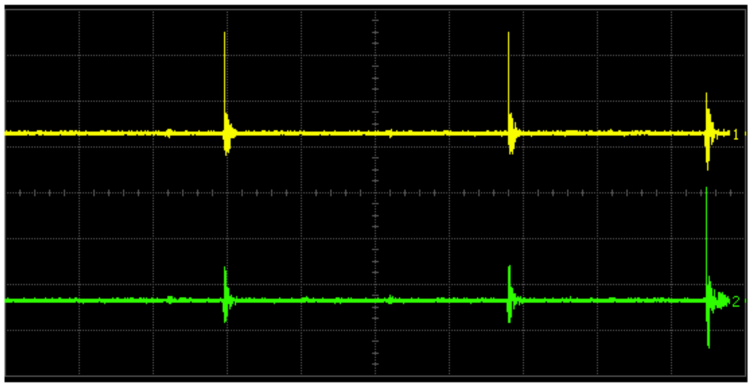
[119] Thomas Gräf. Design of SSDC Shield. ISE, TU Kaiserslautern.

[120] J. Kennedy and R. Eberhart. Particle Swarm Optimization. In *Neural Networks, 1995. Proceedings., IEEE International Conference on*, volume 4, pages 1942–1948 vol.4, Nov 1995. doi: 10.1109/ICNN.1995.488968.

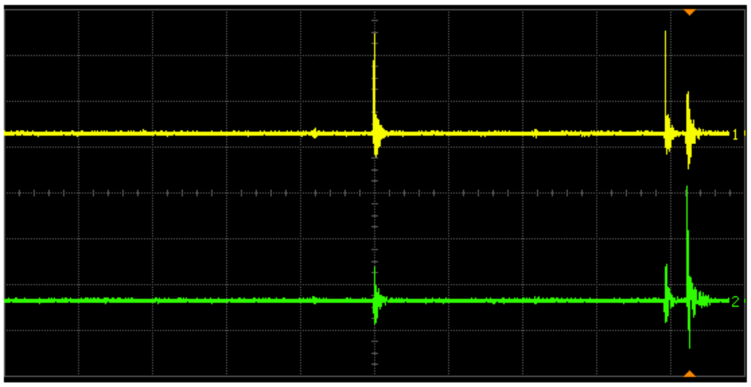
[121] Abhaya Chandra Kammara und Andreas König. Verfahren und Vorrichtung zur Konditionierung und Umwandlung von Analogsignalen in Digitalsignale. Deutsche Patentanmeldung 10 2016 116 492.5, eingereicht 02. September 2016.

A. SSC Measurement Results

The figures below show the measurement results of the SSC which can be understood based on Figure 9.6. These measurement results form the basis for Figure 9.7 and Figure 9.13

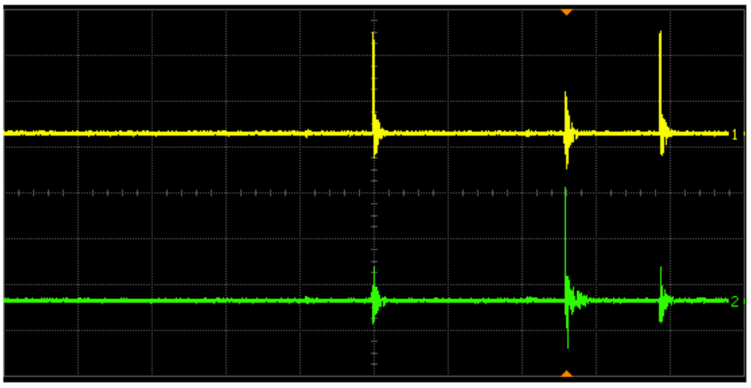


Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on	Inputs Vin1 = 1.70 V Vin2 = 1.60 V
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm	
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm	
Time base	Scale 1.00 µs/ Position -420.1615600 µs Reference center	
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1	



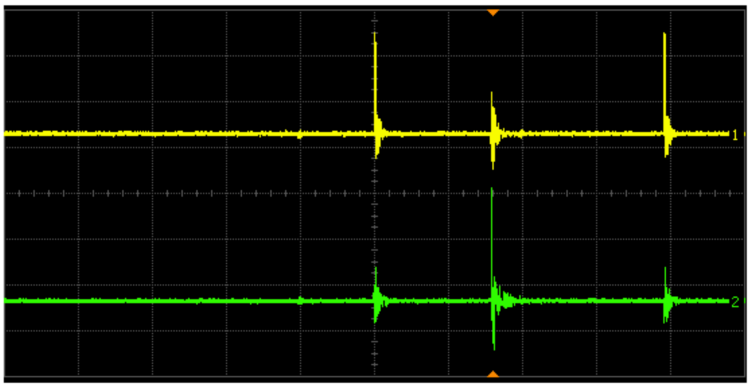
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 1.00 µs/ Position -4.2399000 µs Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.69 V
Vin2 = 1.61 V



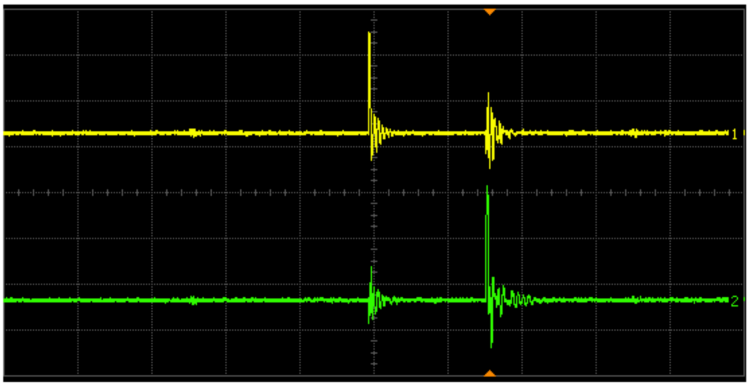
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 1.00 µs/ Position -2.6019000 µs Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.68 V
Vin2 = 1.62 V



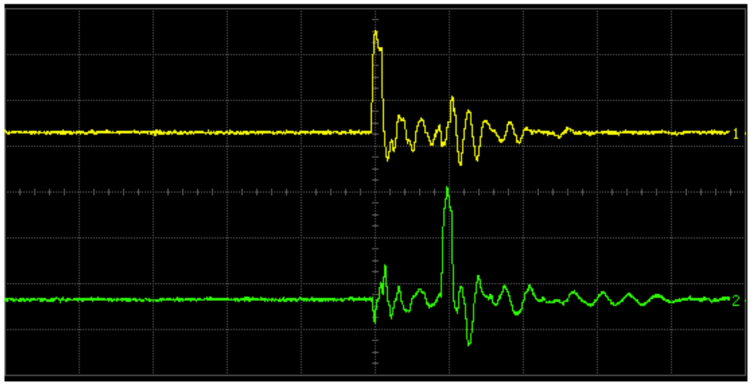
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 1.00 µs/ Position -1.6009000 µs Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.67 V
Vin2 = 1.63 V



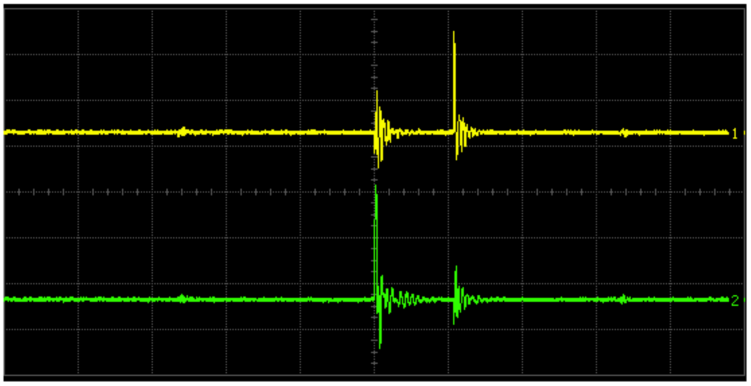
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 500 ns/ Position -781.9000 ns Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.66 V
Vin2 = 1.64 V



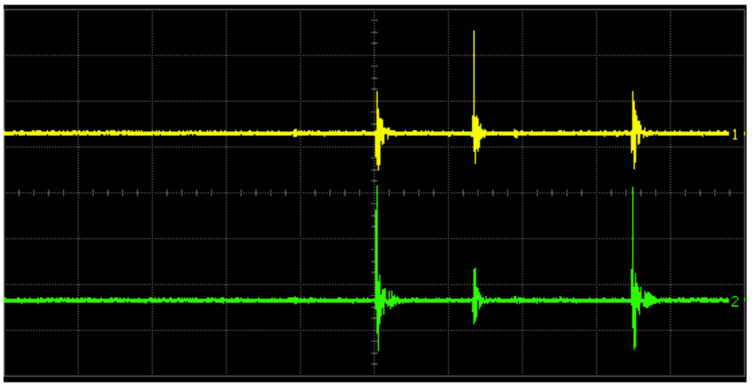
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 100 ns/ Position -72.0913200 µs Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.65 V
Vin2 = 1.65 V



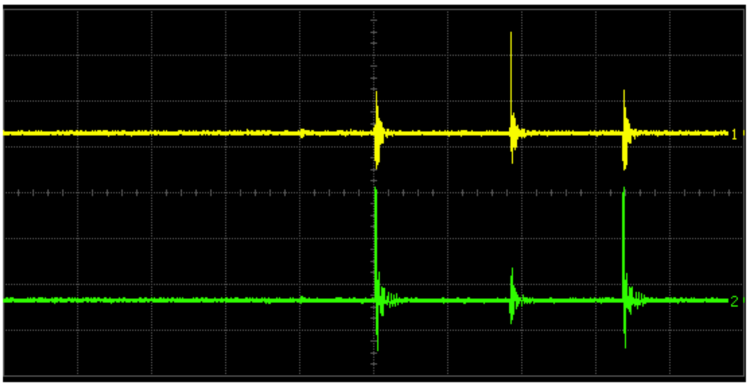
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 500 ns/ Position -32.6883200 µs Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.64 V
Vin2 = 1.66 V



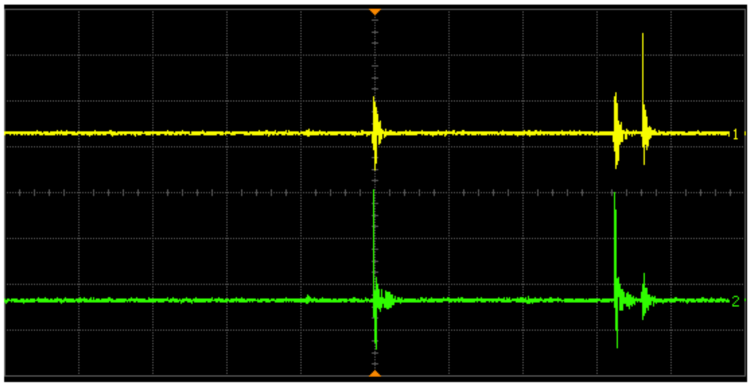
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 1.00 µs/ Position -768.5143200 µs Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.63 V
Vin2 = 1.67 V



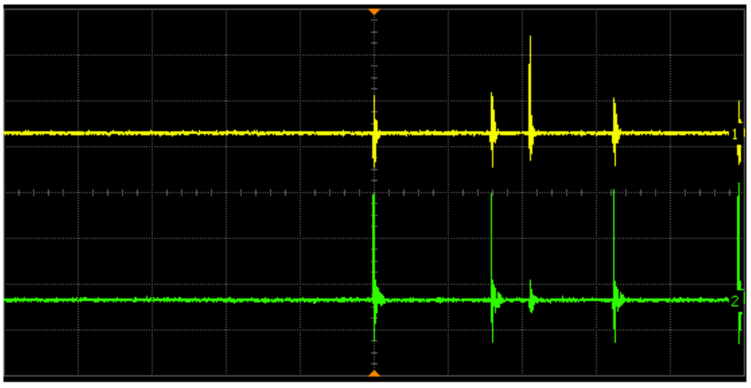
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 1.00 µs/ Position -2.9455255000 ms Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.62 V
Vin2 = 1.68 V



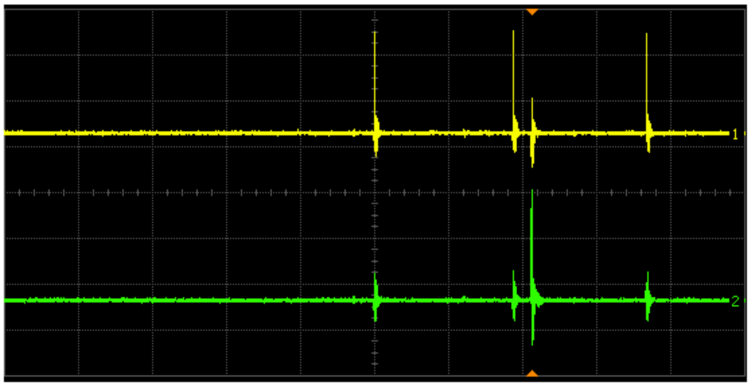
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 1.00 µs/ Position 0.0 s Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.61 V
Vin2 = 1.69 V



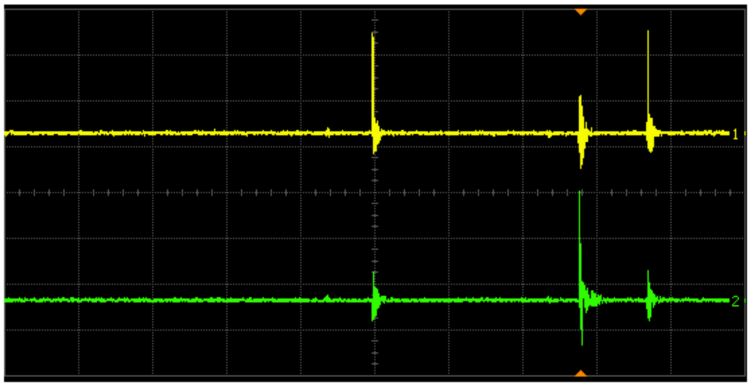
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 2.00 µs/ Position 0.0 s Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.60 V
Vin2 = 1.70 V



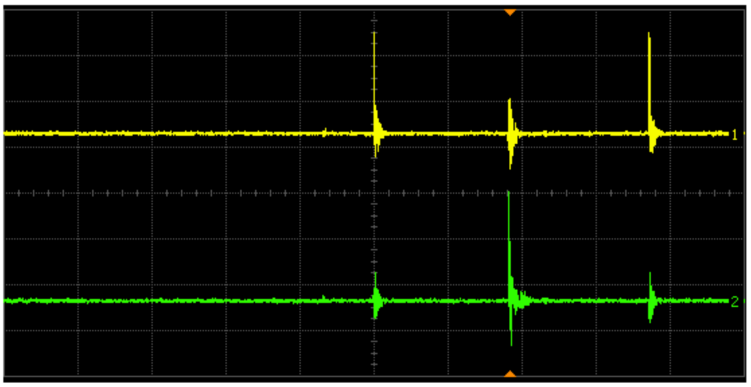
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 2.00 µs/ Position -4.2584000 µs Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.80 V
Vin2 = 1.70 V



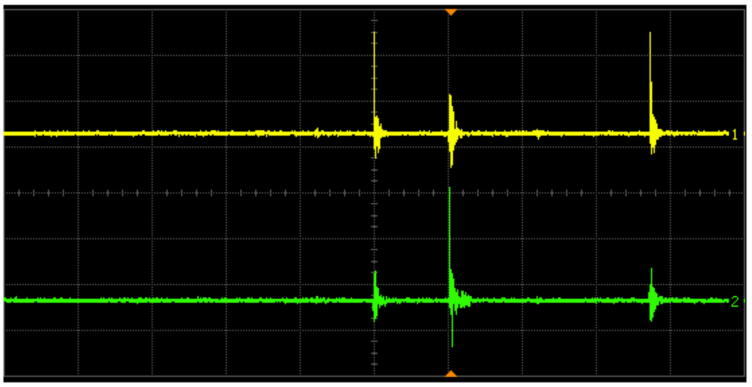
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 1.00 µs/ Position -2.7842000 µs Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.79 V
Vin2 = 1.71 V



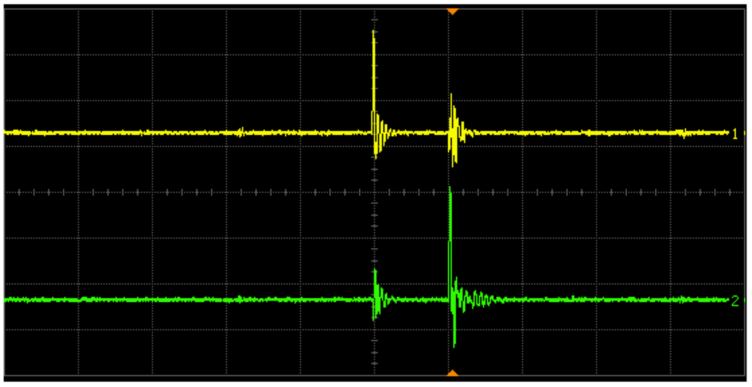
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 1.00 µs/ Position -1.8387000 µs Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.78 V
Vin2 = 1.72 V



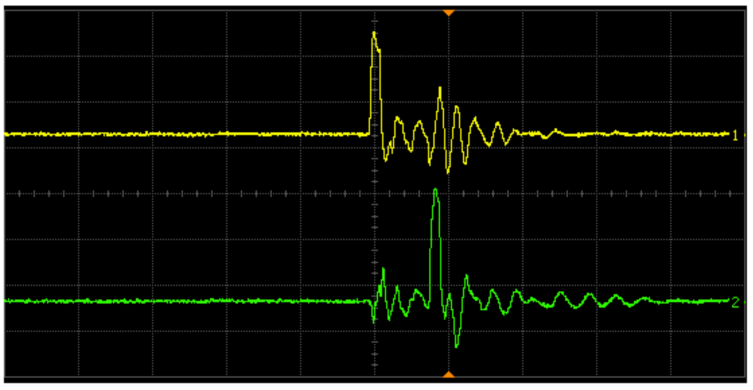
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 1.00 µs/ Position -1.0379000 µs Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.77 V
Vin2 = 1.73 V



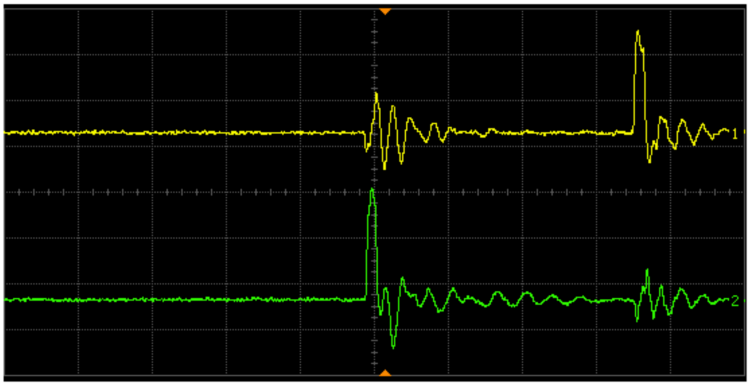
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 500 ns/ Position -528.2600 ns Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.76 V
Vin2 = 1.74 V



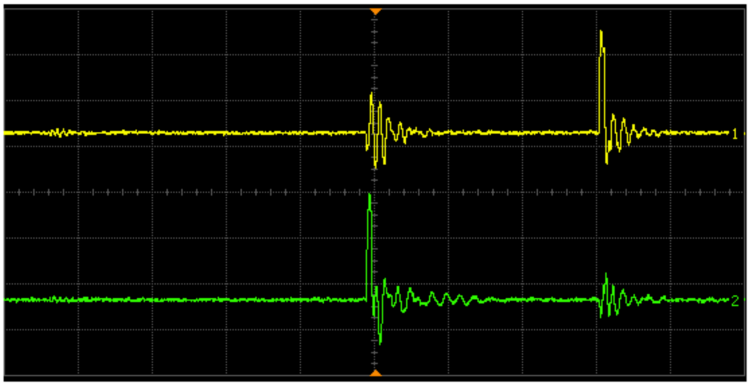
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 100 ns/ Position -99.2200 ns Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.75 V
Vin2 = 1.75 V



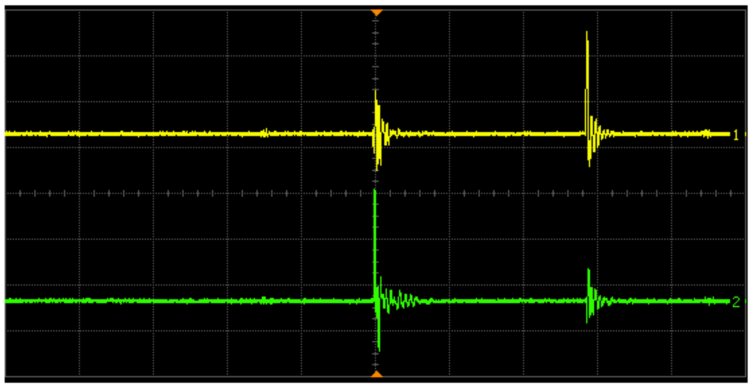
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 100 ns/ Position -14.5600 ns Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.74 V
Vin2 = 1.76 V



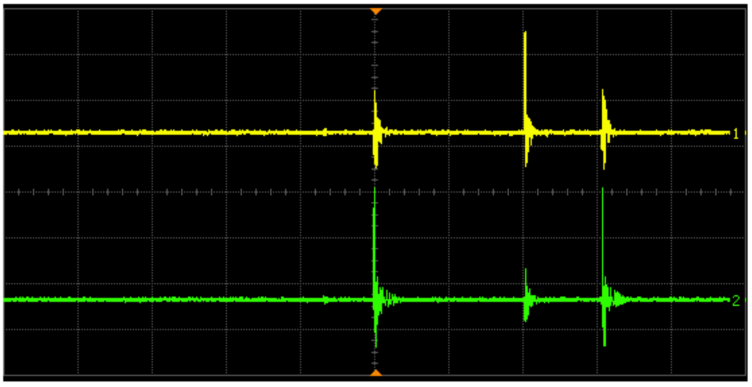
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 200 ns/ Position -3.6400 ns Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.73 V
Vin2 = 1.77 V



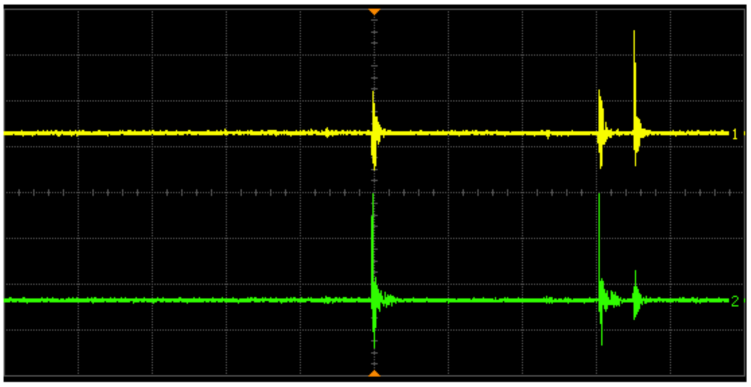
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 500 ns/ Position -12.7300 ns Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

Inputs
Vin1 = 1.72 V
Vin2 = 1.78 V



Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 1.00 µs/ Position -12.7300 ns Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

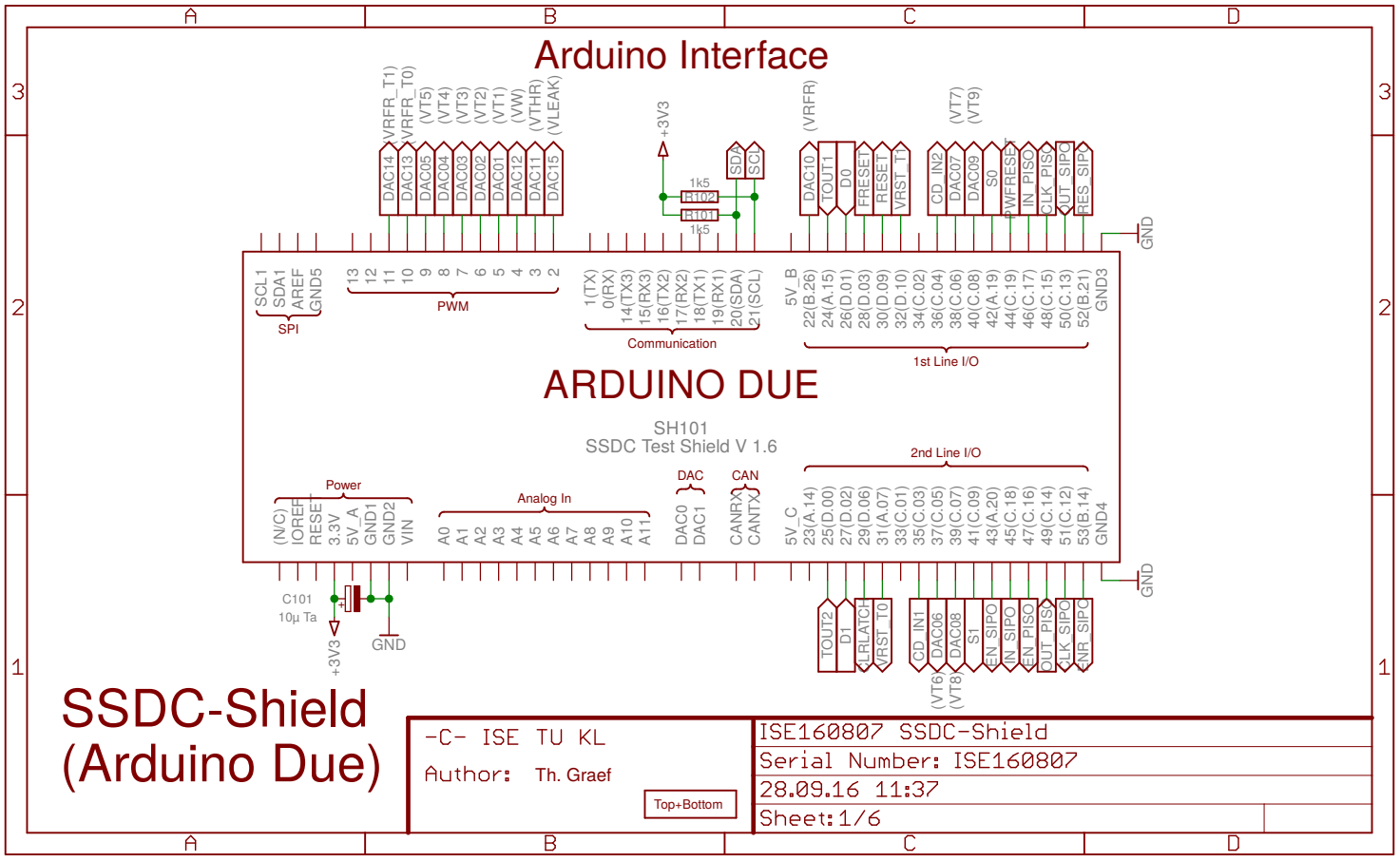
Inputs
Vin1 = 1.71 V
Vin2 = 1.79 V

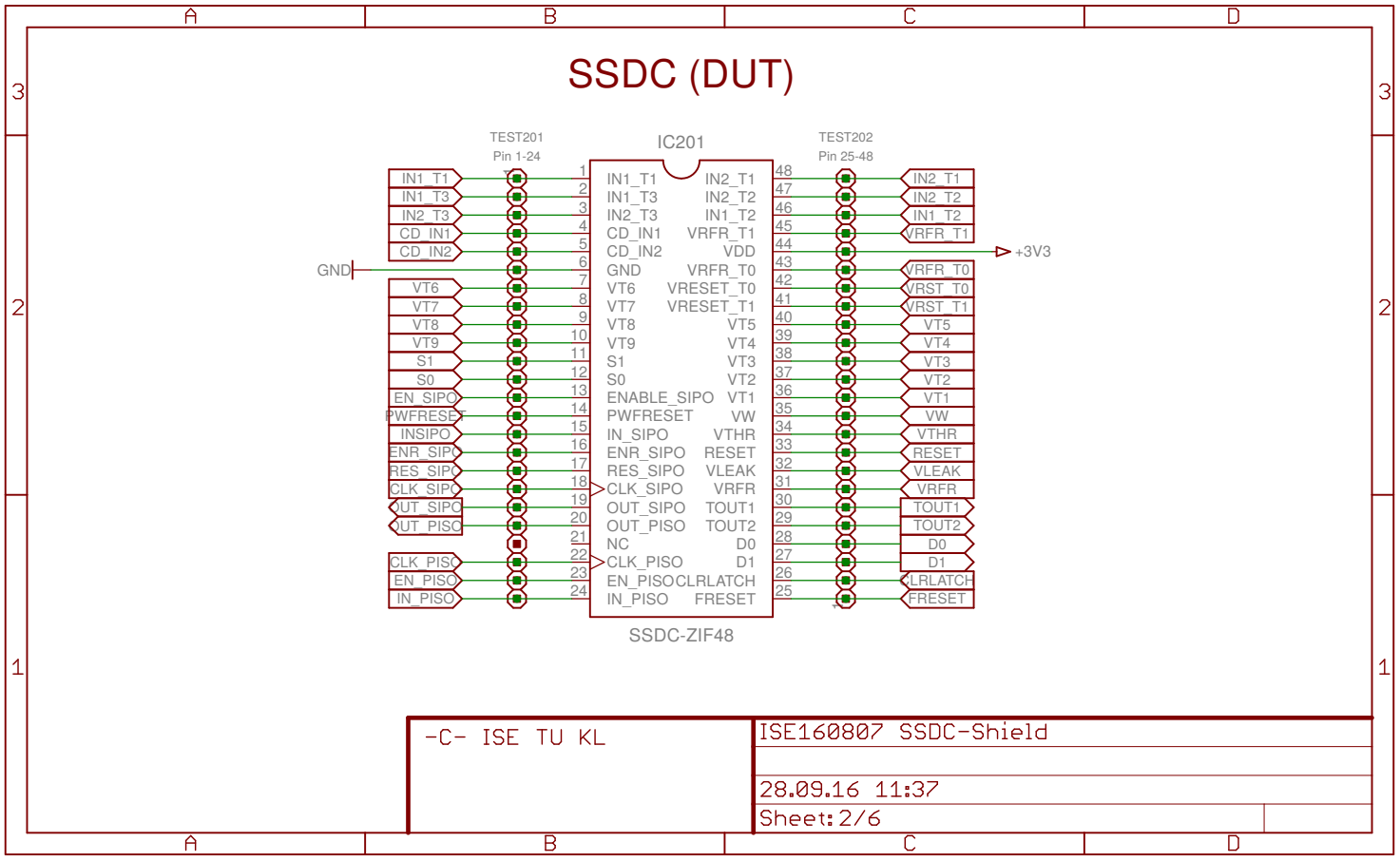


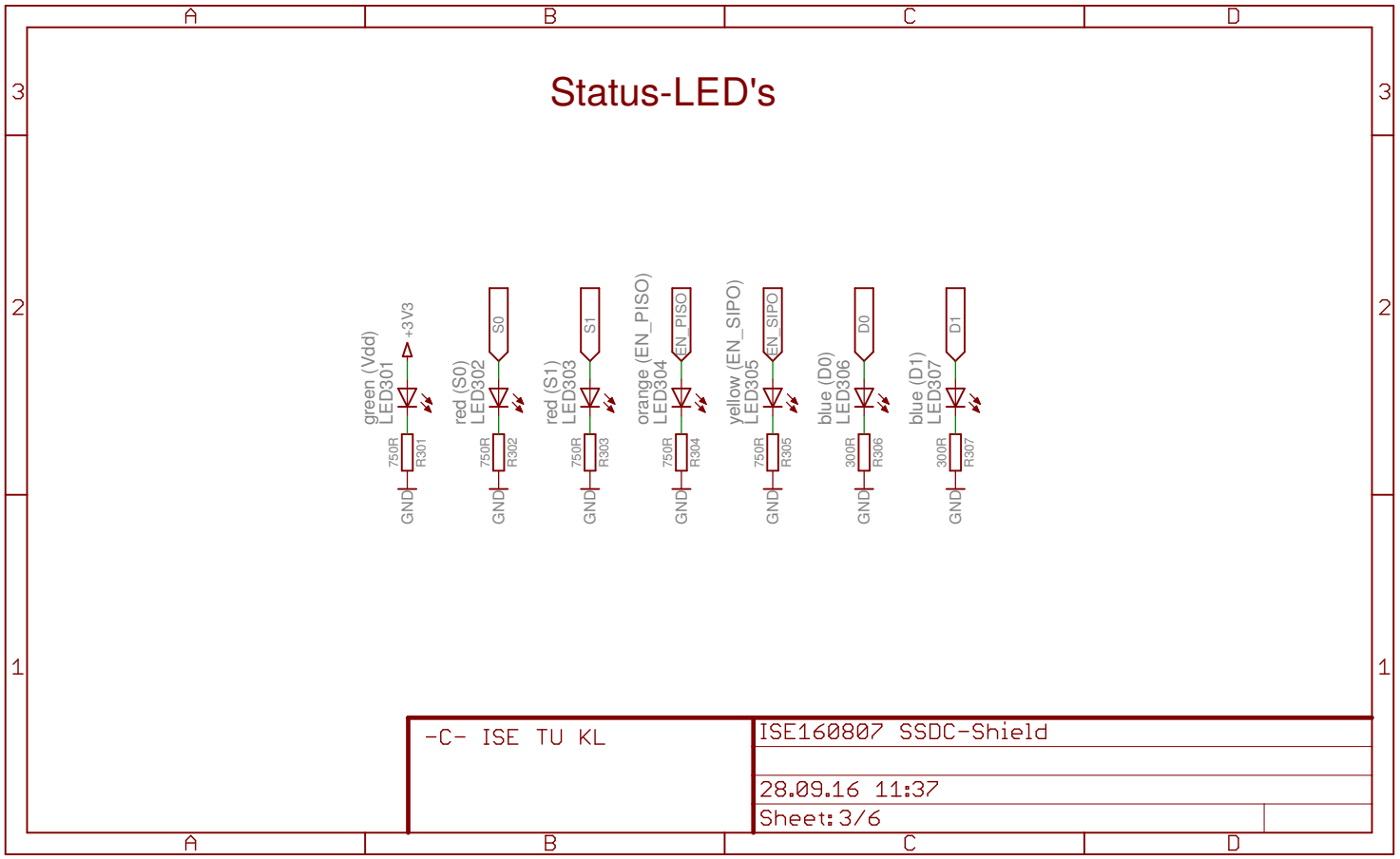
Acquisition	Sampling mode real time Normal Memory depth automatic 8200000 pts Sampling rate automatic Sampling rate 2.00 GSa/s Averaging off Interpolation on
Channel 1	Scale 2.00 V/ Offset -2.59 V Coupling DC Impedance 1M Ohm
Channel 2	Scale 2.00 V/ Offset 4.70 V Coupling DC Impedance 1M Ohm
Time base	Scale 1.00 µs/ Position 5.4700 ns Reference center
Trigger	Mode edge Sweep single Sensitivity normal Holdoff time 50 ns Source digital 1

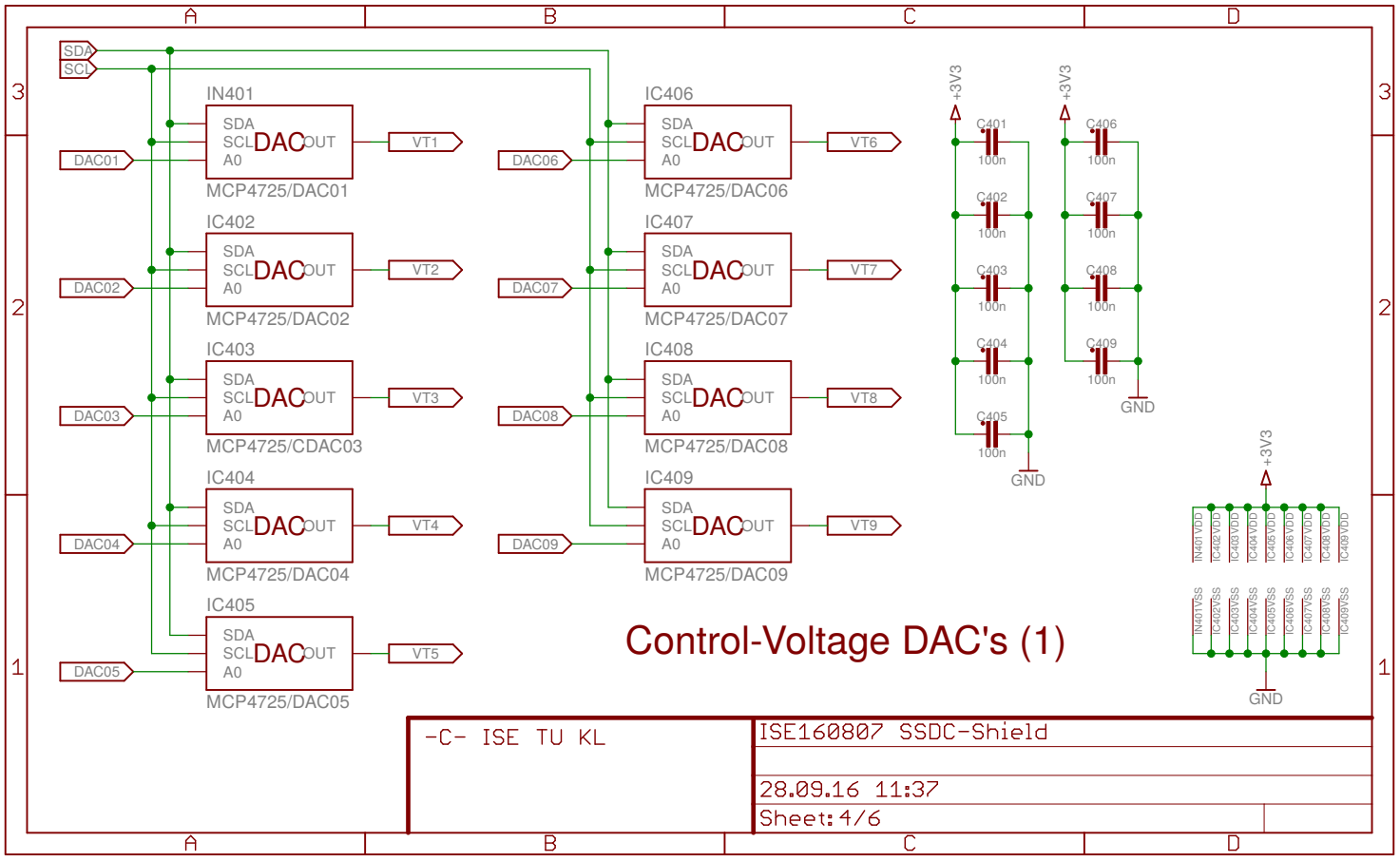
Inputs
Vin1 = 1.70 V
Vin2 = 1.80 V

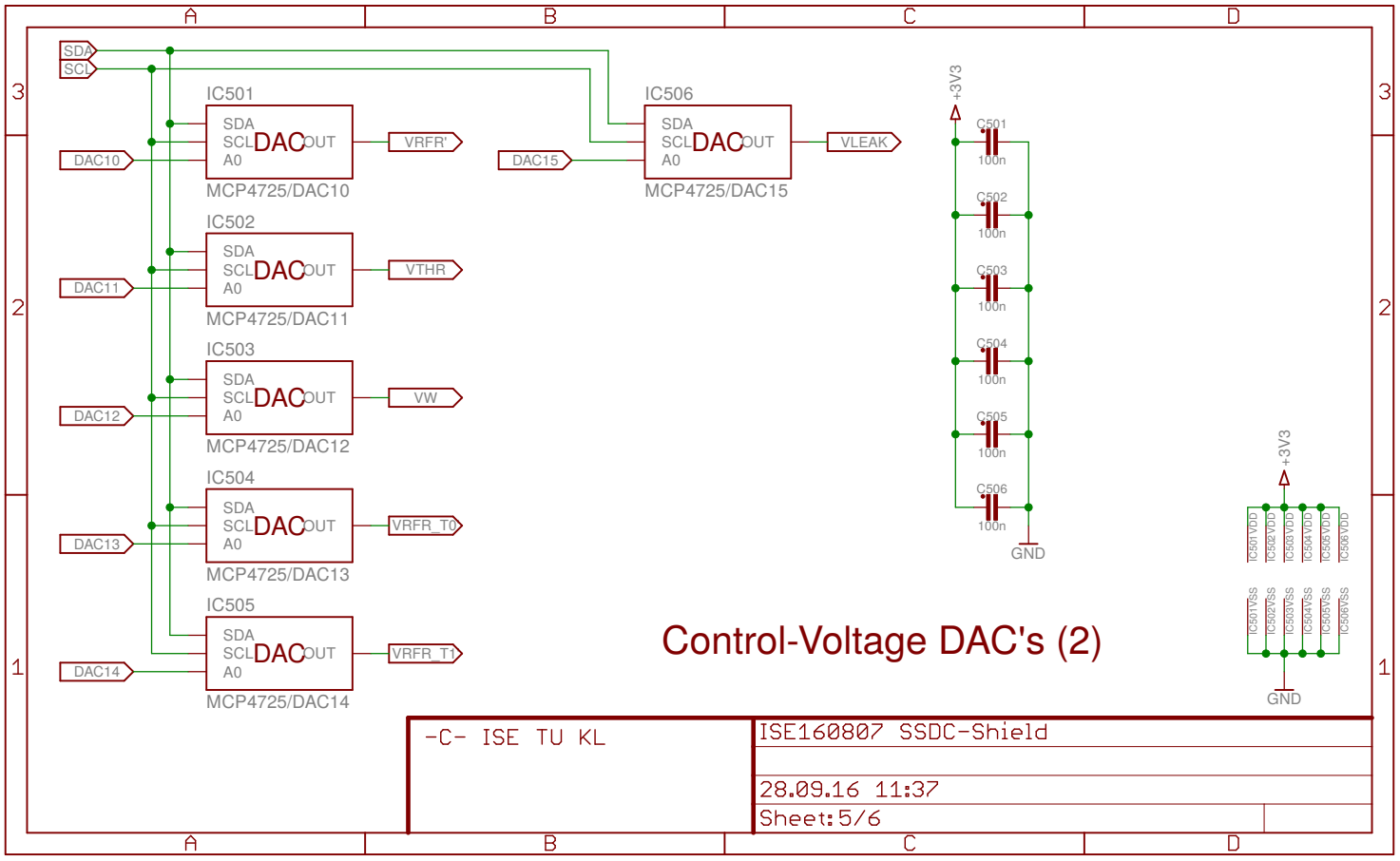
B. SSDC Shield Eagle Schematics

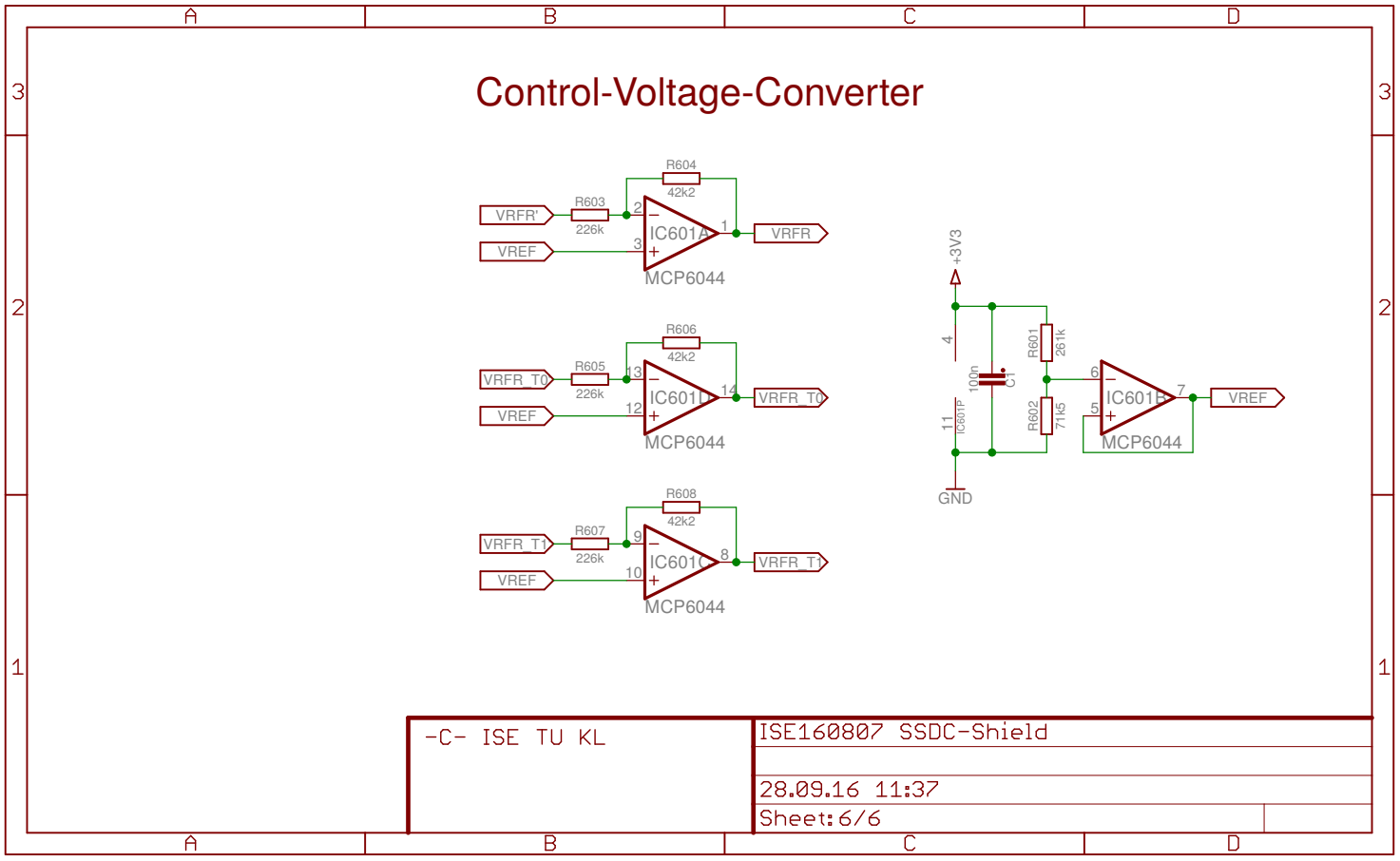

Arduino Interface
ARDUINO DUE
SH101
SSDC Test Shield V 1.6
SSDC-Shield
(Arduino Due)
-C- ISE TU KL
Author: Th. Graef
Top+Bottom
ISE160807 SSDC-Shield
Serial Number: ISE160807
28.09.16 11:37
Sheet:1/6
SPI
PWM
Communication
1st Line I/O
2nd Line I/O
Power
Analog In
DAC
CAN
GND
+3V3
C101
10µ Ta


SSDC (DUT)
TEST201
Pin 1-24
IC201
TEST202
Pin 25-48
IN1_T1
IN1_T3
IN2_T3
CD_IN1
CD_IN2
GND
VT6
VT7
VT8
VT9
S1
S0
EN_SIPO
PWFRESET
INSIPO
ENR_SIPO
RES_SIPO
CLK_SIPO
OUT_SIPO
OUT_PISO
CLK_PISO
EN_PISO
IN_PISO
IN2_T1
IN2_T2
IN1_T2
VRFR_T1
VDD
VRFR_T0
VRESET_T0
VRESET_T1
VT5
VT4
VT3
VT2
VT1
VW
VTHR
RESET
VLEAK
VRFR
TOUT1
TOUT2
D0
D1
CLRLATCH
FRESET
+3V3
SSDC-ZIF48
-C- ISE TU KL
ISE160807 SSDC-Shield
28.09.16 11:37
Sheet: 2/6


A
B
C
D
1
2
3
Status-LED's
green (Vdd)
LED301
+3V3
750R
R301
GND
red (S0)
LED302
S0
750R
R302
GND
red (S1)
LED303
S1
750R
R303
GND
orange (EN_PISO)
LED304
EN_PISO
750R
R304
GND
yellow (EN_SIPO)
LED305
EN_SIPO
750R
R305
GND
blue (D0)
LED306
D0
300R
R306
GND
blue (D1)
LED307
D1
300R
R307
GND
-C- ISE TU KL
ISE160807 SSDC-Shield
28.09.16 11:37
Sheet: 3/6


A
B
C
D
1
2
3
SDA
SCL
IN401
IC406
IC402
IC407
IC403
IC408
IC404
IC409
IC405
DAC01
DAC02
DAC03
DAC04
DAC05
DAC06
DAC07
DAC08
DAC09
SDA
SCL
A0
DAC
OUT
VT1
VT2
VT3
VT4
VT5
VT6
VT7
VT8
VT9
MCP4725/DAC01
MCP4725/DAC02
MCP4725/CDAC03
MCP4725/DAC04
MCP4725/DAC05
MCP4725/DAC06
MCP4725/DAC07
MCP4725/DAC08
MCP4725/DAC09
+3V3
C401
C402
C403
C404
C405
C406
C407
C408
C409
100n
GND
Control-Voltage DAC's (1)
-C- ISE TU KL
ISE160807 SSDC-Shield
28.09.16 11:37
Sheet: 4/6


A
B
C
D
1
2
3
SDA
SCL
IC501
SDA
SCL
DAC
OUT
A0
MCP4725/DAC10
DAC10
VRFR'
IC506
MCP4725/DAC15
DAC15
VLEAK
IC502
MCP4725/DAC11
DAC11
VTHR
IC503
MCP4725/DAC12
DAC12
VW
IC504
MCP4725/DAC13
DAC13
VRFR_T0
IC505
MCP4725/DAC14
DAC14
VRFR_T1
+3V3
C501
C502
C503
C504
C505
C506
100n
GND
Control-Voltage DAC's (2)
-C- ISE TU KL
ISE160807 SSDC-Shield
28.09.16 11:37
Sheet:5/6


Control-Voltage-Converter
R604
42k2
R603
226k
VRFR'
VREF
IC601A
MCP6044
VRFR
R606
42k2
R605
226k
VRFR_T0
VREF
IC601B
MCP6044
VRFR_T0
R608
42k2
R607
226k
VRFR_T1
VREF
IC601C
MCP6044
VRFR_T1
+3V3
R601
261k
C1
100n
IC601F
R602
71k5
GND
IC601D
MCP6044
VREF
-C- ISE TU KL
ISE160807 SSDC-Shield
28.09.16 11:37
Sheet:6/6

Curriculum Vitae

M.Sc. Abhaya Chandra Kammara Subramanyam

1988 - 2002:	Alpha Matriculation Higher Secondary School, Chennai 600038
2002 - 2006:	Bachelor of Engineering, Electronics and Instrumentation, R.M.K Engg. College, Anna University, Chennai.
Sep. 2006 - Aug. 2007:	Assistant Systems Engineer, Tata Consultancy Services.
Aug. 2007 - Jan. 2010:	M.Sc. Electrical Engineering, TU Kaiserslautern, Kaiserslautern.
Feb. 2010 - Present:	Wissenschaftlicher Mitarbeiter, ISE, TU Kaiserslautern.