Graphene-based neuromorphic computing
Artificial spiking neural networks

Wang, H.

DOI
10.4233/uuid:1b81527f-c493-482c-af9f-cd46adb32729

Publication date
2021

Document Version
Final published version

Citation (APA)
Wang, H. (2021). *Graphene-based neuromorphic computing: Artificial spiking neural networks*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:1b81527f-c493-482c-af9f-cd46adb32729

Graphene-based Neuromorphic Computing: Artificial Spiking Neural Networks

Graphene-based Neuromorphic Computing: Artificial Spiking Neural Networks

Dissertation

for the purpose of obtaining the degree of doctor
at Delft University of Technology
by the authority of the Rector Magnificus, Prof.dr.ir. T.H.J.J. van der Hagen,
chair of the Board for Doctorates
to be defended publicly on
Monday 11 October 2021 at 10:00 o'clock

by

He Wang

Master of Engineering in Computer Science and Technology,
National University of Defense Technology, China
born in Shangqiu, Henan, China

This dissertation has been approved by the promotors:

promotor: dr. S.D. Cotofana
promotor: dr. ir. J.S.S.M. Wong,

Composition of the doctoral committee:

Rector Magnificus	chairperson
Dr. S.D. Cotofana	Delft University of Technology, promotor
Dr.ir. J.S.S.M. Wong	Delft University of Technology, promotor
Independent members:	
Prof.dr.ir. G.J.T. Leus	Delft University of Technology
Prof.dr. H. Corporaal	Eindhoven University of Technology, The Netherlands
Prof.dr. A. Rubio	Polytechnic University of Catalonia, Spain
Prof.dr. G.C. Sirakoulis	Democritus University of Thrace, Greece
Dr. Y. Wang	National University of Defense Technology, China
Prof.dr.ir. S. Hamdioui	Delft University of Technology, reserve member

Keywords: Neuromorphic Computing, Graphene, Spiking Neural Network, Synaptic Plasticity, Spiking Neuron

Printed by: ProefschriftMaken, the Netherlands

Front & Back: designed by He Wang, Amie Zheng

ISBN 978-94-6384-265-5

An electronic version of this dissertation is available at
`http://repository.tudelft.nl/`.

Dedicated to my family.

SUMMARY

The human brain is a natural high-performance computing system with outstanding properties, e.g., ultra-low energy consumption, highly parallel information processing, suitability for solving complex tasks, and robustness. As such, numerous attempts have been made to devise neuromorphic systems able to achieve brain-akin computation abilities, which can aid in understanding the complex human brain functionality and can be utilized to solve complex problems, e.g., pattern recognition and data mining. However, the fact that human brain comprises billions of neurons, which are the fundamental information processing units, and trillions of synapses that interconnect them makes the design and implementation of large-scale brain-inspired computing systems quite a challenging task. Graphene appears to be a promising candidate for scalable neuromorphic implementations as it exhibits a wealth of outstanding properties, e.g., ballistic transport, ultimate thinness, flexibility, and graphene devices are capable of emulating complex nonlinear functions and can be readily tuned to provide various conduction dynamics while preserving low energy operation and small footprint. Moreover, graphene is biocompatible, which offers perspectives for graphene-based neuromorphic bio-interfaces. This thesis aims to investigate graphene's potential to enable scalable and energy effective neuromorphic computing. To this end, we first introduce an atomistic-level simulation model for calculating graphene electronic transport properties, that captures the hysteresis effects induced by interface charges trapping/detrapping phenomena. Second, we propose a generic graphene based synapse, which can be tailored to emulate different synaptic plasticity types by properly modifying its Graphene NanoRibbon (GNR) shape and contacts topology, as well as applying external voltages. Subsequently, we introduce a compact graphene-based integrate-and-fire spiking neuron that mimics the basic spiking neuronal dynamics. We further propose a basic Spiking Neural Network (SNN) unit, which can be utilized to implement complex graphene-based SNN structures. Finally, we introduce a reconfigurable graphene-based SNN architecture and a training methodology for obtaining the initial SNN synaptic weight values. We demonstrate the feasibility of the synaptic weights training methodology and the practical capabilities of the proposed SNN architecture by applying them to solve character recognition and edge detection problems. Our experiments clearly indicate that the proposed graphene-based neuromorphic approach enables low energy operation at small chip real estate footprint, which are enabling factors for the realization of scalable energy-efficient SNN implementations.

SAMENVATTING

Het menselijk brein is een natuurlijk, krachtig computersysteem met buitengewone eigenschappen, zoals een ultralaag energieverbruik, gelijktijdige verwerking van veel verschillende informatiestromen, uitvoering van complexe taken, en robuustheid. Er zijn dan ook talrijke pogingen ondernomen om neuromorfische systemen te ontwerpen die in staat zijn hersen-achtige rekencapaciteiten te bereiken, kunnen helpen bij het begrijpen van de complexe functionaliteit van het menselijk brein, en kunnen worden gebruikt om complexe problemen op te lossen zoals patroonherkenning en datamining. Echter, het feit dat het menselijk brein miljarden neuronen bevat, de fundamentele informatieverwerkende bouwstenen, en triljoenen synapsen die hen onderling verbinden, maakt het ontwerp en de implementatie van grootschalige door het brein geïnspireerde computersystemen een nogal uitdagende taak. Grafeen blijkt een veelbelovende kandidaat te zijn voor schaalbare neuromorfische systemen omdat het over tal van uitstekende eigenschappen beschikt, zoals ballistisch transport, extreme dunheid, en flexibiliteit, en omdat grafeen devices complexe niet-lineaire functies kunnen nabootsen en gemakkelijk kunnen worden ingesteld om verschillende geleidingsdynamieken te bieden met behoud van laag energieverbruik en een kleine voetafdruk. Bovendien is grafeen biocompatibel, wat perspectieven biedt voor op grafeen gebaseerde neuromorfe bio-interfaces. Dit proefschrift onderzoekt het potentieel van grafeen om schaalbare en energie-efficiënte neuromorfische computers te maken. Hiertoe introduceren we eerst een simulatiemodel op atomair niveau voor de berekening van de elektronische transporteigenschappen van grafeen, inclusief de hysterese-effecten veroorzaakt door interfaceladingen. Ten tweede stellen we een generieke op grafeen gebaseerde synaps voor, die kan worden ingesteld om verschillende synaptische plasticiteitstypes na te bootsen door aanpassing van de vorm van de Graphene NanoRibbon (GNR) en de topologie van de contacten , alsook door toepassing van externe spanningen. Vervolgens introduceren we een compact grafeen-gebaseerd integratie-en-vuur neuron dat de basis spiking neuronale dynamica nabootst. Verder stellen we een basis Spiking Neural Network (SNN) eenheid voor, die kan gebruikt worden om complexe grafeen-gebaseerde SNN structuren te implementeren. Tenslotte introduceren we een herconfigureerbare grafeen-gebaseerde SNN architectuur en een trainingsmethodologie voor het verkrijgen van de initiële SNN synaptische gewichtswaarden. We demonstreren de haalbaarheid van de trainingsmethodologie voor synaptische gewichten en de praktische mogelijkheden van de voorgestelde SNN architectuur door ze toe te passen op het oplossen van karakterherkenning- en randdetectieproblemen. Onze experimenten tonen duidelijk aan dat de voorgestelde grafeen-gebaseerde neuromorfische aanpak een laag energieverbruik met een kleine chip voetafdruk mogelijk maakt ten behoeve van schaalbare energie-efficiënte SNN implementaties.

ACKNOWLEDGEMENTS

Time flies by. Four years of Ph.D. study are finally coming to an end. When I recall the experience during this period, it was like a unique interesting and challenging adventure. I would like to express my sincere gratitude to everyone who helped me complete the adventure successfully.

First of all, I would like to express my deepest gratitude to my promotor Dr. Sorin Cotofana. Thank you for providing me the opportunity to pursue my Ph.D. thesis under your guidance. Sorin, thank you for instructing me on how to do research, write a scientific paper, present at a conference, and collaborate in a research team, for letting me know the importance of independent and critical thinking, and for supporting me to develop my ideas. Your responsibility, optimism, humor and wisdom have profoundly influenced my attitude towards work and life. Thank you very much for all your efforts, patience and help during my Ph.D. study.

I would like to especially thank Dr. Nicoleta Cucu Laurenciu for her continuous help during my Ph.D. study. Nicoleta, it is one of the luckiest things to work with you in the past four years. I am pretty sure that our hard work, fruitful discussions, and cheerful chats will be my precious memories. I would like to thank my colleague Yande Jiang for the valuable suggestions and the pleasant office time.

I would like to express my appreciation to Prof. Geert Leus, Prof. Henk Corporaal, Prof. Antonio Rubio, Prof. Georgios Ch. Sirakoulis, Prof. Said Hamdioui and Dr. Yao Wang for accepting to be my committee members, reading this dissertation, and providing feedback, thank you for all your efforts.

It is my honor to study in Quantum Computer and Engineering (QCE) department. I would like to thanks all the QCE members for the pleasant working environment. I would like to thank Prof. Said Hamdioui, as the head of the QCE department, for offering us a wonderful and friendly atmosphere. I would like to thank Dr.ir. Stephan Wong for signing my doctoral programme related documents. I would like to thank Lidwina, Joyce, Laura, Trisha, and Paul for managing paperwork, administration matters, and other secretary-related tasks. In addition, I would like to thank Paul for translating my thesis summary into Dutch. I am thankful to Erik for providing technical support related to cluster usage.

I would like to extend my appreciation to my friends in Delft. They are Lei Xie, Jintao Yu, Shanshan Ren, Jian Fang, Xiang Fu, LingLing Lao, Na Chen, Lizhou Wu, Yu Zhang, Hai Zhu, Sihang Qiu, Xiaohui Wang, Meng Zhang, Min Jiang, and Langzi Chang. It is my pleasure to have a good time with you. I also would like to thank my friends in Changsha for helping me dealing with personal affairs. They are Difu Sun, Xiang Xing, Hao Shi, Panpan Zhao, Yemao Xu and Shuo Tian.

I would like to thank Dr. Yao Wang for recommending me as a Ph.D. student to TU Delft. Without you, I will have no chance to study neuromorphic computing in TU Delft. I would like to express my appreciation to my mentor in China, Prof. Weixia Xu, thank you for supporting me all the time.

Last but not least, I would like to express my deepest thanks to my family, in particular my parents, grandparents and little brother. Your consistent support has strengthened my motivation to move forward and given me the courage to overcome all kinds of difficulties. There are no words that can fully express my gratitude towards you. Love you all forever.

He Wang
Delft, September, 2021

CONTENTS

1

INTRODUCTION

Human brain is a natural high performance computing system, which due to its unique and attractive properties, e.g., energy efficiency, real-time reaction, robustness, suitability for complex task solving and highly parallel information processing ability, spurred the development of a disruptive computing paradigm, the Neuromorphic Computing (NC). In the past decades much effort has been made to develop brain-inspired computation paradigms and biologically-inspired neuromorphic systems based on artificial neurons and synapses. The NC development promoted research aiming at understanding brain's fundamental operational principles and achieving human brain comparable computation abilities. However, the fact that human brain comprises billions of neurons, which are the fundamental information processing units, and trillions of synapses that interconnect them, makes the design and implementation of large-scale brain-inspired computing systems quite a challenging task. Software simulation of neural networks on Turing-von Neumann computing platforms offers flexible support for a wide range of neuromorphic models and have been widely utilized to address machine learning issues and aid neuroscience research. However, this is a power hungry approach and the fundamental incompatibilities between conventional Turing-von Neumann architectures and the human brain, e.g., separate memory and processing, limit the feasibility horizon of neuromorphic simulations. On the other hand, much effort has been made to implement neuromorphic systems in hardware. However, in most state-of-the-art neuromorphic systems, neurons and synapses are implemented with complex CMOS circuitry, which results in high energy consumption and limited scalability and integration density. More recently, emerging technologies, e.g., phase change memory, memristive and graphene devices, proved to be attractive candidates for the design and implementation of neuromorphic systems. Among them, graphene appears to be the most promising candidate as it exhibits a wealth of outstanding properties, e.g., ballistic transport, high current density and good electrical conductivity. Additionally, graphene devices are capable of emulating complex nonlinear functions and can be readily tuned to provide various conduction dynamics while preserving low energy operation and small footprints that make them ideal candidates for neuromorphic implementations. Moreover, when compared with other emerging technologies, graphene is a biocompatible material, which offers perspectives for graphene-based neuromorphic bio-interfaces. In view of this, in this thesis we investigate graphene's potential to enable scalable and energy effective neuromorphic computing.

1.1. Motivation

For decades, devising a platform capable of processing information with brain-like computation capacity has been a computing driving force. Software simulations of neural models on Turing-von Neumann paradigm inspired computation platforms, e.g., GPUs and multi/many cores, are widely utilized in various application scenarios, e.g., recognition of text [1–3], image [4–6], and speech [7–9], and neuroscience research [10–12]. However, substantial differences exist between the Turing-von Neumann computation model and the human brain in terms of, e.g., organizational structure, power requirements, and computation capability, which impede the realization of fast and energy efficient neural systems software implementations. As a result, investigations on the possibility of developing alternative architectures based on brain-inspired models have been initiated towards the end of the previous millennium, and Neuromorphic Computing (NC) has emerged as a promising approach to obtain brain-akin processing ability with artificial neural computation systems. Carver Mead invented the term "Neuromorphic Computing" in 1990 [13], and at that time, NC was referring to Very Large Scale Integration (VLSI) with analog components that mimicked biological neural systems. More recently, the term has evolved to encompass various artificial systems that are based on biologically-inspired artificial neural networks [14–16]. These neuromorphic systems are notable for their high parallelism and the ability to execute complex computations faster, more energy-efficiently, and with a smaller footprint than traditional Turing-von Neumann counterparts. Furthermore, neuromorphic systems can be utilized to investigate the operating principles of the human brain in neuroscience research [17, 18] and to implement machine learning algorithms to solve practical tasks [19, 20]. While intriguing on its own merits, NC has received greater attention due to, e.g., increasing power limitations associated with Dennard scaling, Moore's law approaching end, and the von Neumann bottleneck [21], and various neuromorphic systems have been designed and implemented [22–24].

Much of the early NC work aimed to design computation platforms capable of performing extremely parallel data processing [25–27], inspired by the parallelism observed in the human brain. Neuromorphic systems comprise simple basic processing units, known as neurons, and dense interconnections between them, known as synapses, and by their very architecture exhibit intrinsic parallelism. Some early NC work also targeted the implementation of neural network task specific accelerators [28–30], as due to their natural parallelism and custom hardware support they can perform neural computation much faster than conventional architectures. The study of neuromorphic systems with natural parallelism and fast computing prompted the development and implementation of bio-inspired systems with real-time performance [31–33]. Currently, aside from enabling parallelism, fast neural computation, and real-time performance, researchers pay special attention on devising artificial neural systems that consume low energy [17, 20, 23] and creating small footprint devices capable of delivering neural style behaviors [34–36], as they can open the road towards the implementation of large-scale energy-efficient neuromorphic systems with brain-akin computation ability. To date, neuromorphic systems have been utilized in a wide range of applications, e.g., brain-machine interfaces [37–40], image [41–44] and speech recognition [45–48], and robotics [49–52].

NC encompasses a wide range of research areas, including materials science, electrical

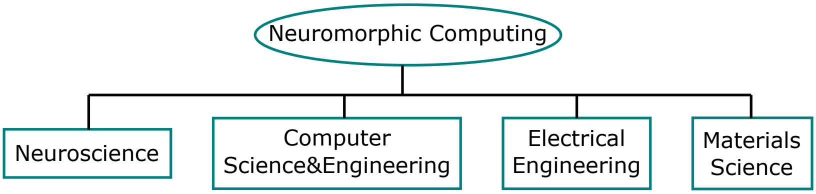


Figure 1.1: Research Areas Involved in Neuromorphic Computing.

engineering, computer science and engineering, and neuroscience, as depicted in Figure 1.1. Materials researchers investigate and develop new neuromorphic device tailored materials, with special emphasis on obtaining characteristics that make them suitable for mimicking biological neural behavior. Electrical and computer engineers work on new neuromorphic computing devices, and devise non-traditional circuitry and architecture for neuromorphic systems. Computer scientists and engineers focus on developing new neural network models that can be utilized for biological and machine learning tasks, and of the associated learning and training schemes. Neuroscience researchers investigate the human brain functionality and develop bio-inspired models, and also make use of neuromorphic systems to aid their studies. The aforementioned research areas are clearly interdependent and their synergy opens unexpected avenues towards the creation of novel NC systems.

When developing new neuromorphic systems, the key question is: Which neural model to utilize? Inspired from the biological brain, neural network models comprise neurons, which are basic information processing units, and synapses, which are junctions connecting them. Based on the neurone type (spiking or non-spiking), neural networks can be categorized into Spiking Neural Networks (SNNs) [53–55], and non-spiking neural networks, which are called Artificial Neural Networks (ANNs) [56]. Generally speaking, spiking neuron models capture more biological neuron features than the non-spiking ones. A broad range of neuron models, from complicated biologically plausible to simple computation-oriented ones, have been proposed [57–63]. A qualitative comparison [14] of different neuron models in terms of biological plausibility and complexity is presented in Figure 1.2. McCulloch-Pitts models are derivatives of the original McCulloch-Pitts neuron [57], which is non-spiking and is utilized in most computation-oriented ANNs. Integrate-and-Fire (I&F) models [53, 58] are a category of simple biologically-inspired spiking neuron models, which capture the essential neuron functionalities with low complexity. Furthermore, higher complexity models that emulate more biological neuron features include Izhikevich [59], Fitzhugh-Nagumo [60], and Hindmarsh-Rose [61] models. The Hodgkin-Huxley (HH) model [62] is the most popular biologically plausible neuron model. It is a rather sophisticated neuron model and it is commonly utilized in neuromorphic systems that try to accurately emulate biological neural systems. Morris Lecar model [63] is simpler than HH, yet it has a similar biological plausibility as HH. When determining the appropriate model for a neuromorphic system, one should consider the specific target, e.g., McCulloch-Pitts for computation-oriented applications and Hodgkin-Huxley for biologically plausible applications, while making the best tradeoff between biological plausibility

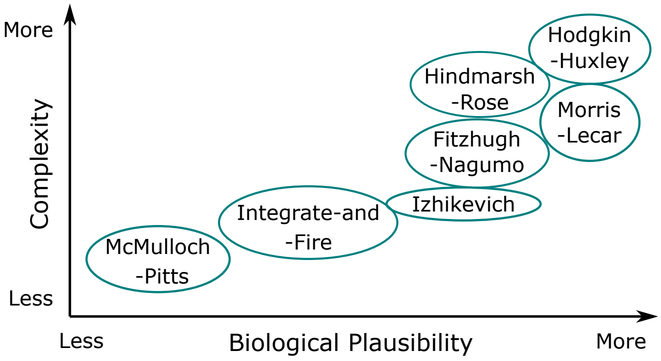


Figure 1.2: A Qualitative Comparison of Neuron Models [14].

and complexity.

SNNs are particularly appealing for NC research, as they capture the essential spiking neuron features with low complexity. To illustrate SNN's functionality, Figure 1.3 depicts a small SNN example consisting of 3 spiking neurons (N_i, N_j, and N_k) that are connected via 2 synapses (Synapse$_{\text{ik}}$ and Synapse$_{\text{jk}}$). Neuron N_k collects input spikes (S_i and S_j) from the other neurons, and generates an output spike S_k when the effect of the cumulated input reaches its firing threshold. Immediately after the firing event, N_k enters a refractory period, during which it doesn't react to incoming spikes. Synapses, while essentially known as signal transmission between adjacent neurons media, assume a processing role as well since their transmission efficiency (synaptic weight), governed by the so-called Synaptic Plasticity (SP) process, can either enhance or inhibit the transported signals. SP is believed to play a crucial role in human brain learning and memorizing processes [64, 65].

Various hardware technologies have been utilized to implement the aforementioned neural network models, which can be categorized at the device and material level as conventional, i.e., CMOS, and emerging, e.g., memristor [66], phase change memory [67], and graphene [68]. As synapses and neurons are the fundamental neural network components, devising appropriate artificial synapses and neurons is the primary focus of any neuromorphic implementation. However, since the human brain comprises billions of neurons and trillions of synapses, designing and implementing large-scale brain-inspired computing systems is a huge challenge.

In general, the implementation of high complexity neuron models (depicted in Figure 1.2) and synapses with versatile plasticity require very complex circuitry, which is the case for most state-of-the-art CMOS neuromorphic systems, e.g., [17, 19, 69, 70]. These systems suffer from high energy consumption and limited scalability. Besides, CMOS designs rely on external control signals and additional circuitry to enable versatile synaptic and neuronal functionalities, e.g., different SP types and I&F dynamics. More recently, emerging technologies have been utilized for neuromorphic system implementations as they exhibit electronic properties that are more appropriate for SNN emulation. However, neuromorphic systems implemented with, e.g., memristor and phase change memory, are hy-

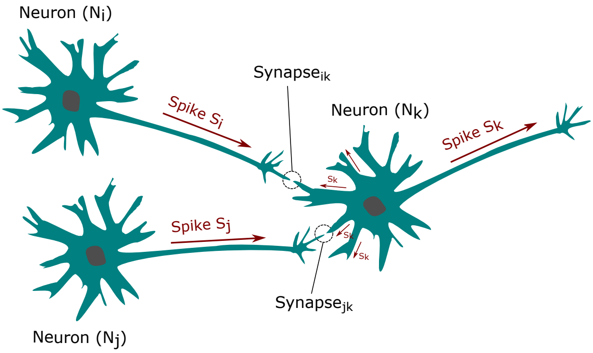


Figure 1.3: Spiking Neural Network Illustration.

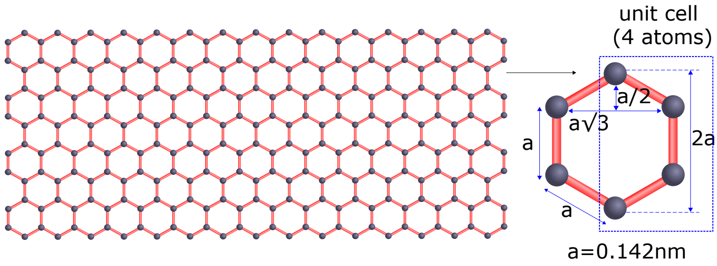


Figure 1.4: Graphene Structure.

brid, i.e., make use of emerging devices to simplify the designs of individual synapses and neurons [71–74], and of additional CMOS circuitry to enable complete synaptic and neuronal functionalities [75–78], which impedes their utilization for scalable energy-efficient implementations.

Another emerging technology with great potential for NC implementations is graphene, which is a two-dimensional carbon atom honeycomb lattice. An example of the crystal structure of a graphene layer is illustrated in Figure 1.4, where the carbon atoms occupy the hexagon vertices and the distance between two adjacent carbon atoms is 0.412 nm. Graphene exhibits outstanding electrical and mechanical properties [79–83], e.g., ballistic carrier transport, high current density, good electrical conductivity, ultimate thinness, and flexibility. Graphene's remarkable characteristics make it an appealing option for a wide range of applications, e.g., spintronics [84, 85], sensors [86–88], biomedicine [89–91], and electronics [92–94]. Specifically, previous work on graphene-based Boolean logic gates [95–98] indicates that graphene-based devices with simple small footprint structure and operating under low power supply voltage can exhibit complex nonlinear functionalities. Besides, graphene-based devices are quite versatile in the sense that different conduction dynamics can be achieved by carving the graphene sheet geometry and altering the contacts topology [99]. Moreover, graphene is a biocompatible material, which offers perspectives for bio-interface applications. These properties make it a particularly promising candidate for scalable energy-efficient neuromorphic implementations. Some preliminary research on graphene-based synapses [100, 101] demonstrated that graphene devices can be utilized in neuromorphic implementations, and that synaptic plasticity can be emulated with a single graphene-based device. However, these designs operate at high voltage and can only provide limited synaptic plasticity types while the feasibility of implementing artificial neurons and SNNs with graphene devices has not been investigated. Therefore, in this thesis we make use of graphene's benefits to design and implement graphene-based synapses, spiking neurons, and SNNs.

1.2. OPPORTUNITIES AND CHALLENGES

Due to its exceptional characteristics, graphene has enormous potential for the implementations of large-scale energy-efficient neuromorphic systems, and graphene devices can be utilized as fundamental building blocks for constructing artificial synapses and neurons. The graphene benefits can be stated from the standpoint of neuromorphic implementations as follows: (i) capability to emulate complex non-linear functionalities, (ii) capability to deliver various conduction dynamics by carving the graphene sheet shape and adjusting the contacts topology of the device, (iii) low energy operation and small footprint, and (iv) biocompatibility, which offers perspectives for neural interfaces. However, despite the fact that graphene exhibits excellent properties for implementing neuromorphic systems, state-of-the-art research on graphene-based neuromorphic computing was mainly concentrated on synapses, while the feasibility of graphene-based neurons and all graphene-based SNNs has not been investigated. Therefore, further research into the possibility to implement SNNs with graphene devices is required to take full advantage of graphene's aforementioned potential to enable scalable energy-efficient neuromorphic computing. Thus, en route to graphene-based neuromorphic systems, some challenges must be addressed.

From the artificial synapses perspective, previous work on synapses [100, 101] demonstrated that synaptic plasticity can be emulated by a single graphene device. However, these designs have many limitations. In [100], the proposed graphene-based synapses operate with prohibitively large back-gate (20 V and 40 V) and input signal voltages (2 V), which make them power hungry and inappropriate for energy effective implementations. In [100] and [101], only restricted synaptic plasticity types are provided and the obtained synaptic transmission efficiency change is small (< 0.01% in [100] and < 2% in [101]). Therefore, further investigation on graphene-based synapses is required to harness graphene's great characteristics. To implement scalable energy-efficient neuromorphic systems with graphene devices, the synapses should exhibit low energy operation and small footprints. To enable the utilization of graphene-based neuromorphic systems for diverse application scenarios, the synapses should be programmable and able to emulate versatile plasticity, i.e., the same synapse can be programmed to deliver different plasticity types.

From the artificial neuron perspective, no prior work on graphene-based neurons has been reported. The basic spiking neuron functionality, e.g., I&F dynamics, is too complex to be achievable with one single graphene device, thus a circuitry comprised of multiple devices is required. The key challenge in developing graphene-based neurons relates to the identification of appropriate devices, e.g., with properly adjusted graphene sheet ge-

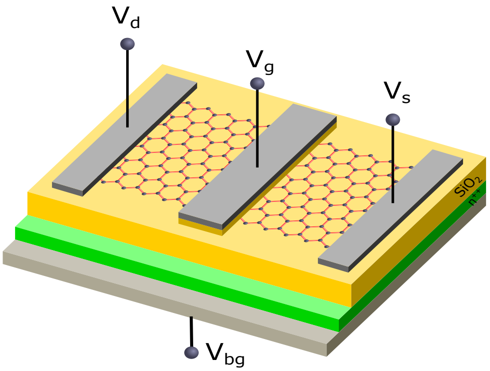


Figure 1.5: Generic Graphene-based Device.

ometry and contacts topology, that, when assembled together, can jointly mimic the neuronal functionality. Furthermore, the resulting graphene-based neuron should operate at low voltage and have a small footprint.

From the neural network perspective, the input-output compatibility of graphene-based synapses and neurons must be assured in order to implement SNNs, as only in this case, artificial synapses and neurons can be directly interconnected to form application tailored SNN structures.

This thesis aims to address the aforementioned graphene-based neuromorphic implementations associated challenges by providing answers to the research questions stated in the following section.

1.3. RESEARCH QUESTIONS

In its most general form, the research question addressed by this thesis can be expressed as:

- **Can graphene pave the way towards scalable energy-efficient neuromorphic computing that goes beyond the CMOS and other emerging technologies horizon?**

To answer this fundamental question, we conduct a complex study, in which we address six related questions that are critical in relation to the general one.

When employing graphene for neuromorphic computing, the first issue is how to use graphene devices, as basic building blocks for artificial synapse and neuron implementations, to obtain the required nonlinear conduction dynamics. Such a generic graphene-based device structure is depicted in Figure 1.5. It relies on a Graphene NanoRibbon (GNR), which serves as a conduction channel when the device is subjected to a drain-to-source bias voltage $V_d - V_s$. The device conduction profile is determined by the nanoribbon geometry and contacts topology, while the actual conductance value is modulated by exerting external voltages on the top and/or back gates. To enable complex synaptic and neuronal behaviors, different nonlinear functionalities are required, and for every such functionality, a GNR device with a different shape and contacts topology ought to be devised. Thus, the first research question can be formulated as:

- **Can we identify appropriate graphene structures, which conduction can emulate given synaptic and neuronal functionalities, while maintaining the potential for scalable energy-efficient neuromorphic implementations?**

To address this question we should be able to identify the nonlinear functionalities required for emulating synaptic and neuronal behaviors, and perform a Design Space Exploration (DSE) by changing GNRs geometry, contacts topology, and bias voltages, to obtain a set of devices capable to deliver the targeted behaviors. Moreover, to enable scalable energy-efficient implementation, the obtained GNR devices should operate at low voltage and have small footprints.

To facilitate the appropriate GNR devices identification, and to enable the design and evaluation of graphene-based neuromorphic systems, a simulation framework able to accurately model GNR specific physical phenomena, as well as to perform SPICE circuit-level simulations is required. Moreover, to fully comprehend GNRs behavior and their potential benefit in circuits, such a framework ought to be able to preserve high accuracy GNR modelling during circuit-level evaluations. Thus, the next research question that needs to be addressed can be formulates as:

- **Can we devise a graphene-based neuromorphic systems evaluation framework able to combine high accuracy physical-level simulation of graphene-based devices with fast circuit level SPICE simulation?**

In such a simulation framework, the GNR device electronic transport properties calculation should be carried out by using an atomistic-level graphene model to ensure high precision and to support the simulation of a wide range of GNR shapes and topologies, which are needed to emulate complex synaptic and neuronal functionalities.

Equipped with the GNR device identification method and the simulation framework we can now proceed with the design and implementation of fundamental neural network components, i.e., synapses and neurons.

The synapses should be able to emulate basic synaptic functionalities, i.e., Spike-Timing-Dependent-Plasticity (STDP) and Long-Term Plasticity. To satisfy the requirements for diverse application scenarios, GNR synapses should be programmable and have versatile synaptic plasticity, i.e., the same device should be able to emulate different plasticity types. Moreover, given that synapses are the most abundant element in neural networks, they should operate at low voltage and have small footprints to enable scalable energy-efficient implementations. Thus, the next research question formulates as:

- **Can we devise single device programmable graphene-based synapses with versatile plasticity while preserving their suitability for scalable neuromorphic system implementations?**

The main idea behind this investigation is to incorporate complex synaptic functionalities into an operating voltage and footprint confined single device, which potentially benefits graphene neuromorphic implementation energy efficiency and scalability. Furthermore, GNR synapses should be programmable by adjusting the external voltages applied to the device, e.g., via top/back gates.

The graphene neuron ought to be able to emulate the fundamental spiking neuron dynamics, i.e., collect input spikes from other neurons, generate an output spike when the cumulated signal exceeds the firing threshold, and exhibit a refractory period after a firing event. Since neuronal functionalities are too complex to be achieved with a single device, GNR-based circuits are required when devising graphene-based spiking neurons. The neuron should combine multiple GNR devices and be able to emulate the aforementioned neuronal dynamics. In this context, the next research question can be formulated as:

- **Can we devise compact graphene spiking neurons, by means of multiple GNR devices, that emulate the complex neuronal functionality, while exhibiting low energy consumption and requiring low area?**

The main idea behind such a investigation is to identify appropriate GNR devices with certain nonlinear functionalities, which can be combined to build graphene-based circuitry that can mimic the complex neuronal behavior. Moreover, graphene neurons should be able to operate at low voltage and be compact in order to enable scalable and energy efficient SNN implementations.

Following the identification of graphene synapses and neurons, the next stage of research focuses on how to interconnect them in order to create SNNs. Since the plasticity emulated by graphene synapses have a direct impact on SNN operations, a comprehensive investigation on the effect of the graphene enabled plasticities and the acquired unsupervised learning ability is necessary to demonstrate the feasibility of graphene-based SNNs in practical applications. In view of this, the next research question formulates as:

- **Can we interconnect GNR synapses and neurons to construct application specific graphene-based SNNs with unsupervised learning capabilities?**

To positively answer this research question, we need to assure input-output compatibility between synapses and neurons, such that they can be directly interconnected to form SNN structures. Afterwards, the SNN operations and its unsupervised learning ability can be explored by means of the aforementioned mixed atomistic-circuit simulation framework.

Given that the SNN structure is dependent on the practical task it has to perform, SNN implementations have to be tailored to the application they execute. To diminish the design overhead and facilitate the utilization of graphene-based neuromorphic systems in various practical applications, a versatile SNN architecture capable of providing hardware support for various SNN structures is necessary. Besides, investigations are required to reveal the potential capabilities of graphene-based neuromorphic systems to solve complex practical problems, which raises the following research question:

- **Can we devise a versatile graphene-based SNN architecture that provides hardware support for various applications while being area and energy efficient?**

Such an architecture ought to be reconfigurable to map different neural network structures. The comprising synapses should be programmable in terms of plasticity type and weight value. Moreover, an associated training methodology is needed to determine the initial synaptic weight values. The architecture area and energy efficiency are obviously determined by the figures of merit of the proposed graphene synapses and neurones.

In answering the above research questions, this thesis explores and demonstrates the graphene potential to pave the way towards scalable energy-efficient neuromorphic computing that goes beyond the CMOS and other emerging technologies horizon.

1.4. THESIS CONTRIBUTIONS

This section summarizes the contributions of this thesis towards GNR device simulation and design and implementation of graphene-based synapses, neurons, and SNNs, as follow:

- We introduce an atomistic-level graphene device simulation model, which can calculate the electric transport properties of GNR devices with rectangular or non-rectangular GNR geometries while capturing hysteretic effects caused by interface charges trapping/detrapping phenomena. Specifically, we make use of the tight binding Hamiltonian matrix to describe the interactions between carbon atoms and external potentials, the Non-Equilibrium Green Function (NEGF) formalism to solve the Schrödinger equation, and the Landauer-Büttiker formula to derive the GNR current and conductance [102]. The GNR potential distribution profile is obtained by solving a 3D Poisson equation self-consistently and the trapping/detrapping phenomena are accounted for by calculating the equivalent voltage shift induced by interface trapped charges. We apply the model on a rectangular graphene shape and validate the results against experimentally measured data. Moreover, we demonstrate model's versatility by considering and investigating the hysteretic behavior of two non-rectangular GNRs. The experiments indicate a good agreement between simulated and measured data, which qualifies the model appropriate for traps-aware evaluation of the conduction of graphene-based devices and circuits.

- We develop a hybrid simulation framework that embeds the high accuracy of physical level modelling of graphene conductance within the SPICE environment. Specifically, we make use of a Verilog-A graphene device generic model [103], which in order to enable time effective SPICE simulation of graphene circuits relies on GNR topology specific precomputed lookup tables containing graphene conduction simulation data obtained with the aforementioned atomistic level simulation method. This hybrid framework is utilized to provide circuit level evaluations for graphene-based neuromorphic implementations, i.e., synapses, neurons, and SNNs.

- We present a methodology to emulate synapse and neuron functionalities by means of graphene device conduction dynamics. For synapses, to identify a GNR topology able to provide support for a targeted plasticity we perform a Design Space Exploration (DSE) by changing GNR dimensions, shape, top-gates widths and positions, and back-gate voltage. The synaptic transmission efficiency changes are mirrored on the GNR conductance modifications and the DSE process continues until the targeted functionality is achieved. We utilize the same methodology to identify appropriate GNR devices for the neuron implementation. As multiple GNR devices are required to emulate the neuronal functionality, we perform the DSE process for each of the constituent GNR devices. We note that this methodology is generic and can be utilized to achieve other synaptic and neuronal functionalities with GNR devices, potentially beyond the ones considered in the thesis.

- We propose generic one- and two- top gates graphene-based synapse structures, which can emulate the fundamental synaptic functionalities, i.e., Spike-Timing-Dependent Plasticity (STDP) and Long-Term Plasticity. Additionally, they are programable by means of back-gate bias voltage and the same device can exhibit both Long-Term Potentiation and Long-Term Depression. Our simulations indicate that the one-top-gate synapse can achieve the 100% plasticity change provided by natural synapses. The two-top-gates synapse exhibits STDP with spike duration dependent potentiation/depression time scale while achieving a maximum of 30% synaptic weight change and a potentiation/depression time scale range from $[-1.5\,\text{ms}, 1.1\,\text{ms}]$ to $[-32.2\,\text{ms}, 24.1\,\text{ms}]$. Furthermore, we explore the effect of two-top-gates synapse at the SNN level by performing NEST [10] based simulations. Our experiments indicate a strong corelation between the synaptic plasticity type, i.e., Hebbian and anti-Hebbian, and the number of firing events occurring within the network and that the number of SNN output firing events monotonously varies with respect to the input spikes frequency. For Hebbian STDP and a spike duration of 20 ms we obtain an SNN behavior similar with the one provided by the same SNN with biological STDP. Given that the proposed graphene-based synapses have small footprints ($30\,\text{nm}^2$), operate in the hundred millivolt range, and are versatile from the synaptic behavior point of view, we strongly believe that they can be outstanding candidates for implementing scalable energy efficient neuromorphic systems.

- We propose an entirely graphene-based ultra-compact and low voltage neuron, which is able to emulate the essential features of spiking neurons, including the membrane potential accumulation, the firing event, the refractory interval, and the output spike generation. The proposed neuron operates at voltage ranges akin to those of biological neurons, which makes it a good candidate for biologically plausible utilization scenarios. It consists of 6 GNR-based devices controlled via top-gate voltages, one of them emulating the membrane potential dynamics, and the remaining 5 generating the necessary control signals as well as the output spikes. We validate, by means of SPICE simulations, the basic nonlinear Leaky Integrate-and-Fire (LIF) neuron functionality under periodic input spike trains and noisy stochastic inputs. Our results indicate robustness to neuronal signals variability, and regular output firing rate statistics with a slowly decreasing trend and < 1 interspike interval variation coefficient, when increasing the input firing rate from 20 to 200 spikes per second. For all simulation, we use spike duration and amplitude of 2 ms and 100 mV, respectively, which are comparable to those observed in biological neurons. Note that, the low area footprint (GNR-based device area of max. $36\,\text{nm}^2$) and low operating voltage (200 mV supply voltage) prove the suitability of our proposal for large-scale integration.

- We introduce a basic SNN unit, which comprises a graphene-based synapse and a spiking neuron with input-output compatibility, and can be utilized for the implementation of complex SNNs. An extension approach is provided to accommodate larger than one fan-in, i.e., multiple synapses connected to the same neuron. We first demonstrate the proper operation of the graphene SNN unit by relying on the aforementioned hybrid simulation framework. Subsequently, we analyze the way synaptic plasticity affects SNN behavior by making use of a 2-layer SNN example consisting of 6 neurons. Our results indicate that LTP significantly increases the number of SNN firing events while LTD is diminishing them, as expected. To get inside on the plausibility of the 2-layer SNN reaction to input stimuli we also simulate its behavior by means of NEST, a well established SNN simulation framework. Our experiments indicate that the SPICE obtained reaction is in close agreement with the one reported by means of NEST based simulation, which clearly suggests that the proposed design exhibit a proper behavior. Further, we demonstrate unsupervised learning capabilities by considering a 2-layer SNN consisting of 30 neurons meant to recognize vowel characters (and variations of them). The simulation results indicate that the graphene SNN is able to perform unsupervised learning and that the enabled recognition ability is robust to input character variations. Finally, we note that the proposed SNN unit requires a small real-estate footprint (max. 30 nm^2 are required by one graphene-based device) and operates at 200 mV supply voltage, which suggest its suitability for the design of scalable energy-efficient computing systems.

- We propose a reconfigurable graphene-based SNN architecture and an associated training methodology for initial synaptic weight values determination. Specifically, the reconfigurable SNN architecture comprises a synaptic array (consisting of graphene-based programmable synapses) and a neuronal array (consisting of graphene-based spiking neurons), onto which application dependent network structures can be mapped. To reconfigure the proposed graphene-based platform for a practical application, two ingredients are required: an SNN topology and an initial SNN state, e.g., initial synaptic weights. To demonstrate the validity of the synaptic weights training methodology and the suitability of the proposed SNN architecture for practical utilization, we consider 2 applications, i.e., character recognition and edge detection. We map on the generic graphene-based platform a 2-layer SNN tailored for vowel characters recognition and demonstrate by means of SPICE simulations that it can achieve up to 94.5 % recognition accuracy for the considered training and evaluation datasets, which is very close to the one achieved by a functionally equivalent ANN counterpart. Further, we map and evaluate a 3-layer SNN to perform edge detection on Lena and Cameraman images and demonstrate that the edge detection result quality matches and even outperforms the one obtained with classical edge detection operators. Our results suggests the feasibility and flexibility of the proposed approach for various application purposes. Moreover, the utilized graphene-based synapses and neurons operate at low supply voltage (200 mV), consume low energy per spike for both neuron (43 pJ and 5.2×10^{-7} pJ at 200 Hz and 20 GHz spike frequency scale, respectively) and synapse (5.1 pJ and 6.0×10^{-8} pJ at 200 Hz and 20 GHz spike frequency scale, respectively), and a synapse occupies an active area of ≈45 nm^2 (2 GNR devices) and a neuron an active area of ≈176 nm^2 (6 GNR devices), which are desired properties for scalable energy-efficient implementations.

1.5. Thesis Organization

The remainder of the thesis is organized as follows:

In Chapter 2, we present the atomistic-level hysteresis-aware electron transport model. We apply the model on a rectangular graphene shape and validate it by comparing its outcome with experimentally measured data. Moreover, we demonstrate the versatility of the model by considering two non-rectangular GNRs and investigate their hysteretic behaviour.

In Chapter 3, we propose generic one- and two- top gates graphene-based synapse structures and investigate their capabilities to emulate various plasticity types. We further investigate the two-top-gates synapse capability to achieve spike duration dependent potentiation/depression time scale and explore the two-top-gates synapses' effect at SNN level by performing NEST based simulations. Moreover, we look into the potential area and energy consumption of the graphene-based synapses.

In Chapter 4, we propose an ultra-compact, entirely graphene-based nonlinear Leaky Integrate-and-Fire spiking neuron. We validate its basic functionality and investigate its output response under stochastic noisy input spike trains with a variable firing rate by means of SPICE simulations.

In Chapter 5, we propose a basic graphene-based SNN unit that comprises a graphene synapse and a graphene neuron, which can potentially be utilized to implement complex SNNs. An extension approach is provided to accommodate larger than one fan-in situation, i.e., multiple synapses connected to the same neuron. We demonstrate the proper operation of the SNN unit and analyze the way the synaptic plasticity affects the behavior of a 2-layer SNN example by relying on the hybrid simulation framework introduced in Chapter 2. We further assess the plausibility of the graphene SNN reaction to input stimuli by comparing SPICE with NEST simulation results. Moreover, we demonstrate unsupervised learning capabilities of the proposed design when utilized for vowel characters recognition.

In Chapter 6, we propose a reconfigurable graphene-based SNN architecture and a training methodology for initial synaptic weight values determination. We consider 2 applications, i.e., character recognition and edge detection, to demonstrate the validity of the synaptic weights training methodology and the suitability of the proposed SNN architecture for practical utilization. In each case, we evaluate the reaction of the reconfigured graphene-based platform by means of SPICE simulations.

In Chapter 7, we conclude the thesis and discuss future research directions.

2

Graphene Structures Electron Transport Model

Hysteretic behavior has been experimentally observed in graphene-based structures and has a major influence on graphene surface potential and gate field modulation ability. Thus, a graphene electronic transport modelling methodology, which incorporates hysteresis effects is crucial in order to properly assess gated-controlled graphene structures response and performance. To this end, we propose an atomistic-level electronic transport model, which is non restricted to rectangular graphene geometries and captures hysteretic effects caused by interface charge trapping/detrapping phenomena. We apply the model on a rectangular graphene shape and validate our results against experimentally measured drain current vs. top gate voltage hysteresis curves. Moreover, to demonstrate model's versatility we consider two non-rectangular Graphene NanoRibbons (GNRs) and investigate their hysteresis behaviour. Our experiments indicate good agreement between simulated and measured results, which qualifies the model appropriate for traps-aware evaluation of the conduction of graphene-based devices and circuits.

The content of this chapter is based on the following paper:

H. Wang, N. Cucu Laurenciu, Y. Jiang, and S.D. Cotofana, "Atomistic-level Hysteresis-aware Graphene Structures Electron Transport Model", *IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1-5, 2019.

2.1. Introduction

Hysteretic behavior in the transfer characteristics (drain current with respect to gate voltage) is frequently encountered [81], [104], [105], [106] for graphene-based structures, which can be ascribed to, e.g., charge injection into trap sites situated at the interface between the graphene sheet and the gate dielectric layer. Such hysteretic behavior changes the graphene surface potential and affects the gate field modulation ability. Thus, a theoretical treatment of the electronic transport properties incorporating hysteresis effects in the context of experimental measured data, benefits the investigation and a deeper understanding level of graphene gated structures operation. Up to date, there are very few models able to evaluate graphene's conduction while taking into account interface traps caused hysteresis. Moreover they are restricted to rectangular graphene shapes and rely on a high level transport modelling, e.g., [106], [100], [107], which provides limited accuracy. The preponderant part of previous work is focused either on hysteresis analysis based on in-field measured data [104], [108], or on hysteresis modelling relying on a rather large amount of graphene and trap related measurements [109], [110].

In view of the previous discussion and because non-rectangular graphene topologies appear to have interesting potential [111], [112] we propose an atomistic-level model which can simulate the electronic transport properties of graphene-based structures while taking into account the hysteretic effects caused by charge traps situated at the interface between graphene and the oxide layer. Our model is able to simulate configurations with different graphene geometries (e.g., rectangular or non-rectangular) and contact topologies (e.g., gate width and position relative to the source/drain contacts, multiple gates). We use the Tight Binding (TB) Hamiltonian to model the system, the Non-Equilibrium Green Function (NEGF) and Landauer-Büttiker formalism to model the transport properties, and a 3D Poisson solver to compute the graphene potential self-consistently. The effects of the interface charge traps are reflected as a shift of the top gate voltage, which we compute by relying on solely two trap profile related parameters: the trapping/detrapping time constant and the density of interfacial traps. The two traps parameters are typically extracted from in-field measurement data. To validate our model, we consider the top-gated graphene FET in [113], which we subject to similar set-up conditions and simulate with the proposed methodology. Our experiments indicate that the drain current versus gate voltage characteristics obtained by means of simulation are in good agreement with the reference counterparts reported in [113]. To demonstrate the versatility of our approach we further investigate the hysteretic behavior of two non-rectangular GNR structures with geometric properties that enable a wider energy bandgap, which benefits the "ON"/"OFF" current and power ratio, while subjected to similar trapping conditions.

The remaining of the chapter is structured as follows: Section 2.2 is devoted to the description of the proposed model. Section 2.3 entails the experimental validation results, and finally, Section 2.4 concludes the chapter.

2.2. Hysteresis Model Formalism

In this chapter we deal with the following problem statement: Given a graphene-based structure with a specified stack of materials, graphene geometry (which can also be non-rectangular), and interface traps profile (specified by, e.g., trapping/detrapping time con-

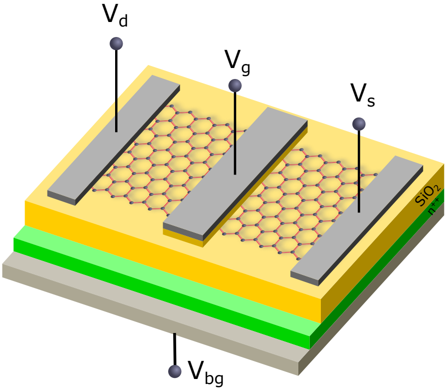


Figure 2.1: Generic Graphene-based Device Structure.

stant, and interface traps capacitance or density of interface states), derive its electronic properties (e.g., current, conductance), while taking into account the effects - degraded graphene surface potential, hysteretic behavior of the current - of near-interfacial defects. To this end, we present subsequently the underlying graphene-based structure, followed by the simulation model of the electronic transport properties, while accounting for trap-caused hysteresis.

Figure 2.1 presents a schematic cross-sectional view of a typical graphene-based structure that can be utilized as basic circuit building block. The channel consists of monolayer graphene on top of an insulating layer, with a doped substrate serving as backgate. The current flow in the graphene channel is induced by applying a bias voltage (i.e., $V_d - V_s$) between the two end-point contacts (source and drain) situated on top of the graphene sheet, and is modulated by an input voltage (i.e., V_g) applied via the top gate.

For modelling the graphene electronic transport, we make use of the atomistic Tight-Binding (TB) approach to represent the system Hamiltonian, the Non-Equilibrium Green Function (NEGF) quantum transport model for solving the Schrödinger equation coupled with the three-dimensional (3D) Poisson equation in a self-consistent manner, and the Landauer-Büttiker formalism for deriving the graphene current [111], [102].

Specifically, the graphene is described by a Hamiltonian matrix $H = H_0 + U$, which models the interactions between neighbor carbon atoms (via H_0) and incorporates all internal and external potentials (e.g., top and back gate voltages) via U.

$$H_0 = \sum_{i,j} t_{i,j} \left|i\right\rangle \left\langle j\right|, \tag{2.1}$$

$$\text{where } t_{i,j} = \begin{cases} \tau, & \text{if atoms } i \text{ and } j \text{ are adjacent} \\ 0, & \text{otherwise.} \end{cases} \tag{2.2}$$

In our simulation we account for first nearest-neighbor (1NN) interactions, with hopping energy between atoms $\tau = -2.7$eV [102]. The potential distribution matrix U is determined self-consistently as the solution of the 3D Poisson equation

$$\nabla \cdot [\epsilon(\mathbf{r})\,\nabla U(\mathbf{r})] = -\frac{\rho(\mathbf{r})}{\epsilon_0}, \tag{2.3}$$

where $\mathbf{r} = x\hat{\mathbf{x}} + y\hat{\mathbf{y}} + z\hat{\mathbf{z}}$ is a position vector in space, ρ is the net charge density distribution, ϵ_0 is the vacuum permittivity, and $\epsilon(\mathbf{r})$ is the dielectric permittivity of the materials at position $\mathbf{r}$. The Poisson equation is numerically solved by making use of the finite difference method [114].

Along the transport direction, on the graphene channel two sides, reside the drain and source contacts with different electrochemical potentials $q \cdot V_s$, and $q \cdot V_d$, which sustain the conduction. The interactions between the two contacts and the graphene channel are modelled via the contact self-energy matrices Σ_1 and Σ_2, respectively. Having computed H and $\Sigma_{1,2}$, the transmission function $T(E)$, which models the probability of one electron being transmitted between the source and the drain contacts, is derived as a function of energy as:

$$T(E) = \text{Trace}\left[\Gamma_1\, G_R\, \Gamma_2\, G_R^\dagger\right], \tag{2.4}$$

where

$$G_R(E) = [EI - H - \Sigma_1 - \Sigma_2]^{-1}, \tag{2.5}$$

$$\Gamma_{1,2} = i[\Sigma_{1,2} - \Sigma_{1,2}^\dagger]. \tag{2.6}$$

The current then writes:

$$I = \frac{q}{h}\int_{-\infty}^{+\infty} T(E) \cdot \big(f_0(E-\mu_1) - f_0(E-\mu_2)\big)\,\mathrm{d}E, \tag{2.7}$$

where $f_0(E)$ denotes the Fermi-Dirac distribution function at temperature T, and $\mu_{1,2}$ represent the source and drain contacts electrochemical potential.

As a graphene structure with a few thousand atoms translates to a Hamiltonian with size in the order of 10^6, in order to reduce the G_R high computational complexity, we exploit on one hand the block tri-diagonal structure of H for the calculation of matrix inversion, and on the other hand the fact that only a single block from G_R is needed for the computation of $T(E)$, as exemplified in Equation (2.8) for a very small graphene divided into 3 columns along the transport direction, with each column containing N atoms.

$$\begin{aligned}
\text{Trace}\left[\Gamma_1\, G_R\, \Gamma_2\, G_R^\dagger\right] &= \text{Trace}\left(\begin{bmatrix}\gamma_1 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}* & * & \circledcirc\\ * & * & *\\ * & * & *\end{bmatrix}\right.\\
&\left.\begin{bmatrix}0 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & \gamma_2\end{bmatrix}\begin{bmatrix}* & * & *\\ * & * & *\\ \circledcirc^\dagger & * & *\end{bmatrix}\right) = \text{Trace}\begin{bmatrix}\boxdot & * & *\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\\
&= \text{Trace}\left(\boxdot\right).
\end{aligned} \tag{2.8}$$

The $*$ blocks denote non-zero blocks which are irrelevant for the trace computation. For deriving the G_R block $\circledcirc$, we rely on a computationally efficient method [115], which

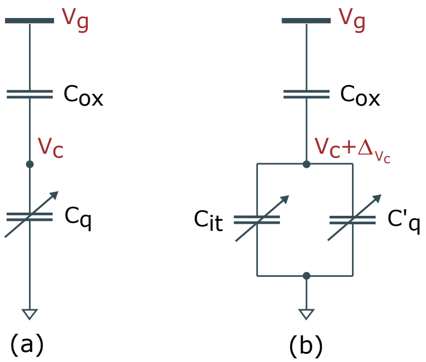


Figure 2.2: Equivalent Capacitive Circuit (a) without Interface Traps, and (b) with Interface Traps.

operates solely on blocks (e.g., size $N \times N$ in Equation (2.8)) of the original matrix, instead of performing the full matrix inverse. Trace(⊡) is then derived based solely on 3 blocks (i.e., γ_1, γ_2, and ⊚ blocks).

Thus far, we have presented the framework for deriving the transport properties. However, as the charge transfer to/from graphene due to near-interfacial traps will cause an equivalent shift of the gate voltage, denoted thereafter as Δ_{V_g}, we have to compute it and update the potential profile U with $V_g + \Delta_{V_g}$ prior to deriving the current/conductance characteristics.

Figure 2.2 depicts the equivalent capacitive circuit of the structure described in Figure 2.1, in the absence (a) and presence (b) of near-interfacial defects, where C_{it} is the capacitance caused by interface traps, C_q and C'_q are the quantum capacitances without and with traps, respectively, and C_{ox} is the gate oxide capacitance. In order to obtain the gate voltage shift Δ_{V_g}, we first compute the voltage drop across the graphene layer, V_c, in the absence of interface traps, in a self-consistent manner using:

$$\begin{cases} C_q(V_c) & = q^2 \cdot \int_{-\infty}^{+\infty} D(E) \cdot \left(- \frac{\partial f_0(E - E_F)}{\partial E} \right) dE, \\ V_c & = V_g \cdot C_{ox}/(C_{ox} + C_q). \end{cases} \tag{2.9}$$

where $D(E)$ denotes the density of states and $E_F = V_c \cdot q$ is the graphene Fermi energy [105]. Once C_q and V_c are computed, Q_q is derived as $Q_q = V_c \cdot C_q$. As concerns the interface traps charge, Q_{it}, it is computed as a fraction α of the net charge in graphene Q_q, while taking into account the gate voltage sweep rate via current time moment t and the trapping/detrapping time constant τ_{trap}:

$$Q_{it} = \alpha \cdot Q_q \cdot e^{-\frac{t}{\tau_{trap}}}. \tag{2.10}$$

In the presence of traps (Figure 2.2 b)), since $Q_{ox} = Q_{it} + Q_q$, ΔV_c follows from:

$$\Delta V_c = V_g - V_c - (Q_{it} + Q_q)/C_{ox}. \tag{2.11}$$

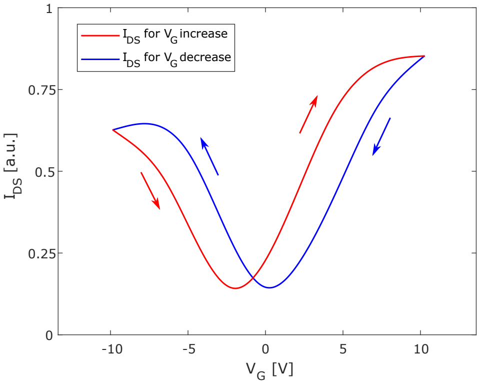


Figure 2.3: I_{DS} vs. V_G for Proposed Model.

Based on $Q_{it} = C_{it} \cdot (V_c + \Delta V_c)$ and Equation (2.11), the value of α as a function of the interface traps capacitance C_{it}, becomes:

$$\alpha(C_{it}) = \frac{C_{it} \cdot (V_g C_{ox} - Q_q)}{Q_q \cdot (C_{it} + C_{ox})}. \tag{2.12}$$

Given that the interface traps capacitance C_{it} (or the density of charge traps) is known and typically extracted from in-field, e.g., capacitance-voltage (C-V) measurements [109], [113], [116], Δ_{V_c} translates to a variation of the gate voltage Δ_{V_g} equal to:

$$\Delta_{V_g} = (V_c + \Delta_{V_c}) \cdot \frac{C_{ox} + C_{it} + C_q}{C_{ox}} - V_g, \tag{2.13}$$

which completes our model.

2.3. SIMULATION RESULTS

To ascertain the validity of proposed hysteresis-aware model, we consider the top-gated Graphene FET (GFET) structure in [113], similar the one depicted Figure 2.1.

The interface traps profile is characterized by a trapping/detrapping time constant of 100 ms and an interface state density $D_{it} = 1.875 \cdot 10^{12}\,\text{cm}^{-2}(\text{eV})^{-1}$ [113]. Based on D_{it}, the interface traps capacitance C_{it} is then obtained as $C_{it} = q^2 \cdot D_{it}$. In the simulation we bias the GFET at 0.1 V and linearly sweep the top gate voltage V_g from -10 V to 10 V, with a rate of 0.1 Vms^{-1} to mimic the measurement conditions reported in [113]. Based on this setup, we construct the GFET system Hamiltonian and first compute the density of states $D(E)$ by using the NEGF-Poisson framework described in Section 2.2. The shift of the top gate voltage Δ_{V_g} is then derived. NEGF-Poisson is run to calculate the new graphene potential profile U with the updated gate voltage $V_g + \Delta_{V_g}$. Finally, the drain current is obtained. Figure 2.3 presents the drain current-gate voltage characteristic in the presence of traps, obtained with the proposed model while the reference hysteresis curves from [113] are

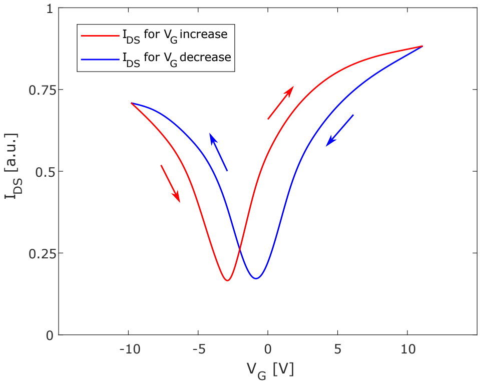


Figure 2.4: I_{DS} vs. V_G for Reference Model [113].

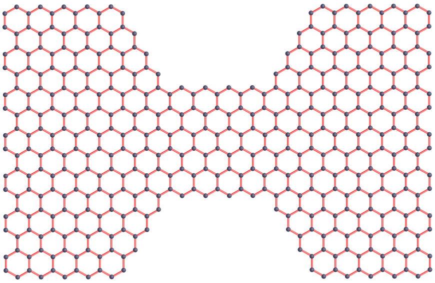


Figure 2.5: Non-rectangular GNR Shape #1.

depicted in Figure 2.4. The Figures clearly indicate that the simulation results are in good agreement with the reference data (i.e., we obtain a shift in Fermi level $\Delta V_{Dirac} = 2.2$ V vs. 2.07 V in experimental measurements), which confirms the correctness of the proposed model.

Contrary to conventional high-level hysteresis-based approaches, the proposed model is able to simulate structures with different graphene geometries and contacts topologies. To demonstrate this we consider two non-rectangular GNR channel shapes, illustrated in Figure 2.5 and Figure 2.6, respectively, which trapezoidal shape can potentially open a wider energy bandgap. Both GNRs have a length of 5 nm and a width of 3 nm. We assume a density of interface traps $D_{it} = 2.5 \cdot 10^{-12}$ cm^{-2}(eV)$^{-1}$ and trapping/detrapping time constant $\tau_{trap} = 70$ ms, and subject the GNRs to a bias $V_d - V_s = 0.1$ V. As we would like to investigate the non-rectangular shapes current behaviour in the presence of interface traps, we change the top gate voltage in increments of 100 mV/ms, in a range which might be of potential interest when using such shapes for building, e.g., logic circuits, case in which negative voltages are of little use while smaller positive ones are. To this end and in order to investigate different voltage ranges we choose a top gate voltage sweep range of 0.7 V to 1.6 V, and of 1.2 V to 2.8 V, for the non-rectangular GNR #1 and #2, respectively.

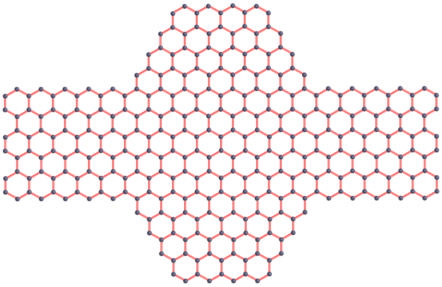


Figure 2.6: Non-rectangular GNR Shape #2.

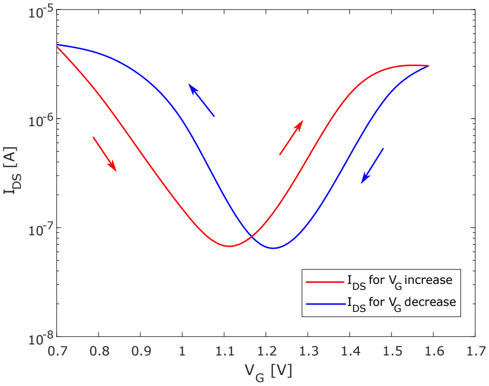


Figure 2.7: Non-rectangular GNR #1 $I-V$ Transfer Curves.

We note that the curves can be shifted along Ox axis by back-gate bias, which allows for the investigation of other input voltage ranges.

Figure 2.7 graphically illustrates the drain current vs. top gate voltage transfer curve for the non-rectangular GNR shape #1. To gain some insight into the interface traps effect, we note that the gate voltage corresponding to the minimum current value experiences a voltage shift of 0.3 V in the presence of interface traps. Thus, without traps, the drain current estimation may suffer a loss of accuracy, which can be regarded proportional to the V_g magnitude shift corresponding to the minimum current point in the figure.

The drain current for the non-rectangular GNR #2 is depicted in Figure 2.8. The rather different hysteresis loops relative to the conventional ones (around the Dirac point) can be attributed to the heavy influence of GNR dimension and geometry on its conductance behavior. We observe that the hysteresis minimum points (and maximum ones) - between increasing curve and decreasing one - shift with 0.19 V (and with 0.22 V). If the traps are ignored, a small shift of 200 mV, can translate into a current estimation error of up to 2 orders of magnitude. We note that the ability to capture such phenomena is of great importance when designing low-frequency circuits (where the hysteresis effects are more pronounced), e.g., for neuromorphic computing, which due to graphene's biocompatibil-

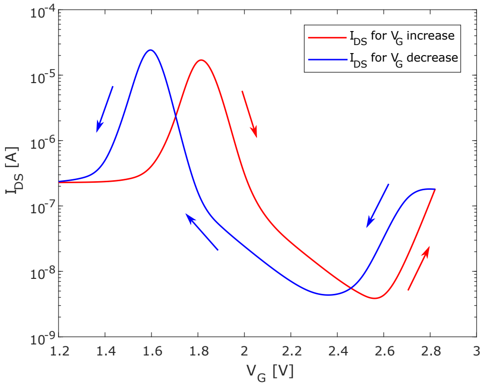


Figure 2.8: Non-rectangular GNR #2 $I - V$ Transfer Curves.

ity could be of great research interest.

2.4. CONCLUSIONS

In this chapter, we proposed an atomistic level simulation model for graphene electronic transport properties able to capture hysteresis effects caused by near-interfacial charge traps. We demonstrated the capability of our model to accurately capture the traps-caused hysteresis effects by comparing its outcome with actual experimental measurements performed on a Graphene FET device. As opposed to state of the art counterparts which are only applicable for rectangular topology the proposed approach can simulate hysteresis effects on structures with various graphene geometries and we exemplified this for two non-rectangular Graphene Nanoribbons. Our experiments suggested that the model can be utilized for novel graphene-based structures conduction properties investigations in the presence of traps, which require an accuracy degree closer to physical level not achievable by other state-of-the-art hysteresis models.

3

Graphene-based Synapses with Versatile Plasticity

In this chapter we investigate the feasibility of graphene-based synapses to emulate various synaptic plasticity behaviors and look into their potential area and energy consumption for large-scale implementations. We propose generic one- and two- top gates graphene-based synapse structures, which can emulate the fundamental synaptic functionalities, i.e.,Spike-Timing-Dependent Plasticity (STDP) and Long-Term Plasticity. Additionally, they are programable by means of back-gate bias voltage and the same device can exhibit both Long-Term Potentiation and Long-Term Depression. Our simulations indicate that the one-top-gate synapse can achieve the plasticity change of 100% provided by natural synapses. The two-top-gates synapse exhibits STDP with spike duration dependent potentiation/depression time scale while achieving a maximum of 30% *synaptic weight change and potentiation/depression time scale range from* $[-1.5\,\mathrm{ms}, 1.1\,\mathrm{ms}]$ *to* $[-32.2\,\mathrm{ms}, 24.1\,\mathrm{ms}]$. *Furthermore, we explore the effect of two-top-gates synapse at the SNN level by performing NEST based simulations. Our experiments indicate a strong corelation between the synaptic plasticity type, i.e., Hebbian and anti-Hebbian, and the number of firing events in the network and that the number of SNN output firing events monotonously varies with respect to the input spikes frequency. For Hebbian STDP and a spike duration of* 20 ms *we obtain an SNN behavior similar with the one provided by the same SNN with biological STDP. Given that the proposed graphene-based synapses have small footprints (*$30\,\mathrm{nm}^2$*), operate in the hundred millivolt range, and are versatile from the synaptic behavior point of view, we strongly believe that they can be outstanding candidates for implementing scalable energy efficient neuromorphic systems.*

The content of this chapter is based on the following papers:

H. Wang, N. Cucu Laurenciu, Y. Jiang, and S.D. Cotofana, "Graphene Nanoribbon-Based Synapses with Versatile Plasticity", *IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH)*, pp. 1-6, 2019.

H. Wang, N. Cucu Laurenciu, Y. Jiang, and S.D. Cotofana, "Graphene-based Artificial Synapses with Tunable Plasticity", *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, vol. 17, pp. 1-21, 2021.

3.1. Introduction

The synapse, which is the most abundant components in neural systems, is a junction connecting two neurons and sustaining the information transmission between them. The synaptic transmission efficiency is variable and appears as the potentiation or depression of the transmitted signals[117, 118]. The synaptic transmission efficiency strengthening or weakening is known as synaptic plasticity, which is believed to be the basis of learning and memory in human brain. From synaptic behavior point of view, an artificial synapse ought to implement two basic functionalities: (i) Spike-Timing-Dependent Plasticity (STDP), which changes the transmission efficiency depending on the relative timing difference between the pre-synaptic and post-synaptic spikes [119], and (ii) Long-Term Plasticity (LTP) in its two flavours, i.e., Long-Term Potentiation (LTP) and Long-Term Depression (LTD), which is a persistent synaptic transmission efficiency change [120].

Since there are trillions of synapses in the nervous system that are essential for supporting the human brain complex functionality, designing and implementing artificial synapses for any large-scale biological-inspired computation systems received massive attention. In most state-of-the-art neuromorphic systems, the artificial synapses are implemented with complex CMOS circuitry [18, 22, 121, 122]. However, as CMOS scaling is approaching atomic feature size, which results in high power consumption and low reliability, CMOS-based synapses bring limitations to scalability, energy efficiency, and integration density of large-scale neuromorphic systems. Besides, CMOS-based artificial synapses cannot efficiently mimic the analog synaptic behavior. Recently, emerging resistive switching memory devices [66] attracted interest and have been used to implement artificial synapses [34, 35, 72, 123]. The obtained artificial synapses exhibit outstanding properties, e.g., inherently analog behavior, simple structure (one single or a few resistive switching memory devices for one synapse), and good scalability potential. However, they suffer from temporal and spatial variability of the resistive states and undesired stochastic behavior, which may cause neuromprphic system instability. They also need to be operated at relatively high voltage, which precludes the implementation of energy efficient computation systems. Previous work on graphene-based synapses demonstrated that graphene-based devices can emulate synaptic plasticity. In [100], the authors demonstrated various synaptic plasticities, however, their proposed artificial synapses operate with quite large back-gate voltage (20 V, 40 V) and input signals (2 V), which makes them power hungry and inappropriate for energy effective implementations. Moreover, the obtained synaptic transmission efficiency change is extremely small ($< 0.01\%$) and only restricted synaptic plasticity types are provided. In [101], a graphene-based synapse is proposed that relies on changing the Li ion concentration between graphene layers to control the device conductance. This synapse endows low-power switching ability, low variability, and is potentially suitable for large-scale implementations. However the reported conductance change is small ($< 2\%$) and the considered spikes timing difference around 1000 ms is significantly different from the one observed in natural synapses (≈ 80 ms).

In this chapter we propose two generic graphene-based artificial synapse structures, which are implemented by graphene-based devices with one top-gate and with two top-gates, respectively. Both structures consist of a single-layer Graphene Nanoribbon (GNR) on top of an insulating material with a doped substrate serving as back-gate. The current flow in the GNR channel is induced by applying a drain-to-source bias voltage. The synaptic transmission efficiency, which is reflected as the conductance of the proposed synapses, can be modulated by means of external voltages, e.g., via the top-gate(s) and back-gate. To mimic the synaptic behavior, we consider two fundamental synapse functionalities, i.e., STDP and Long-Term Plasticity. By carving the GNR geometries and contact topologies of the proposed structures, we successfully obtain various synaptic plasticity including balanced Hebbian STDP, potentiation biased Hebbian STDP, potentiation biased anti-Hebbian STDP, Long-Term Potentiation (LTP),and Long-Term Depression (LTD) for both structures and additionally obtain balanced anti-Hebbian STDP for two-top-gates synapse. We also demonstrate that the same device can emulate both LTP and LTD by simply changing the back-gate voltage. The GNR device is biased at 0.2 V and operates on 110 mV inputs, which is consistent with measured biological synapses data and makes it operation compatible with natural neural matter. For the one-top-gate synapse,the simulations indicate that the plasticity change of 100% provided by natural synapses can be achieved. Besides, by applying input spikes with different spike duration, the two-top-gates synapses can emulate STDP with varying potentiation/depression time scale without affecting the obtained amplitude of the synaptic transmission efficiency change. We obtained a maximum 30% synaptic weight change and potentiation/depression time scale range from $[-1.5\,\text{ms}, 1.1\,\text{ms}]$ to $[-32.2\,\text{ms}, 24.1\,\text{ms}]$. Furthermore, we explore the effect of two-top-gates synapse at the Spiking Neural Network (SNN) level by performing NEST [124] based simulations of a small SNN implemented with 5 leaky-integrate-and-fire neurons connected via GNR-based synapses. Our experiments indicated a strong connection between the synaptic plasticity type, i.e., Hebbian and anti-Hebbian, and the number of firing events in the network. Moreover the number of SNN output firing events monotonously varies with respect to the input spikes frequency. For graphene-based Hebbian STDP and spike duration of 20 ms we obtained an SNN behavior relatively similar with the one provided by the SNN with biological STDP.

The proposed design and synaptic plasticity emulation methodology is generic and our simulation results suggest that by changing the GNR shape and contact topologies various synaptic plasticities can be obtained, potentially beyond the 6 reported cases. Given that the proposed graphene-based synapses have small footprints ($30\,\text{nm}^2$), operate in the hundred millivolt range, and are versatile from the synaptic behavior point of view, we strongly believe that they can be outstanding candidates for implementing large-scale energy efficient neuromorphic systems.

The remaining of this chapter is structured as follows: Section 3.2 presents the underlying concepts about synapse and synaptic plasticity, and introduces the proposed graphene-based synapse structures. In Section 3.3 we describe the simulation model for graphene electronic transport properties calculation, and the simulation setup and employed methodology. In Section 3.4 and Section 3.5 we present the obtained simulation results, investigate the impact of different spike duration on synaptic plasticity, and explore the effect of synaptic plasticity on neural networks behavior. Section 3.6 concludes the chapter.

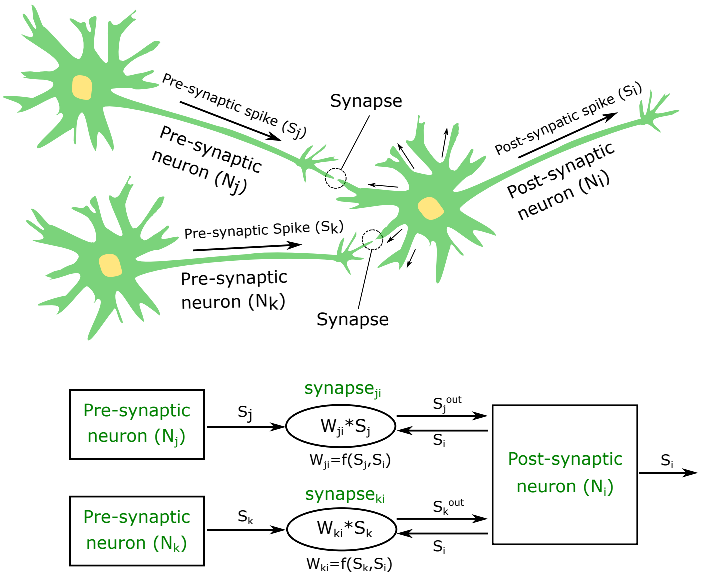

Pre-synaptic spike (S_j)
Synapse
Post-synpatic spike (S_i)
Pre-synaptic neuron (N_j)
Post-synaptic neuron (N_i)
Pre-synaptic Spike (S_k)
Pre-synaptic neuron (N_k)
Synapse
Pre-synaptic neuron (N_j)
S_j
synapse$_{ji}$
$W_{ji}*S_j$
$W_{ji}=f(S_j,S_i)$
S_j^{out}
S_i
Post-synaptic neuron (N_i)
S_i
Pre-synaptic neuron (N_k)
S_k
synapse$_{ki}$
$W_{ki}*S_k$
$W_{ki}=f(S_k,S_i)$
S_k^{out}
S_i

Figure 3.1: Synapse-based Signal Transmission.

3.2. Synaptic Plasticity and Graphene Synapse

In this section we present the fundamental concepts underlying synaptic plasticity and then introduce the proposed generic graphene-based synapses.

3.2.1. Synapse and Synaptic Plasticity

Synapse is the most abundant component in human brain, which serves as the junction connecting two neurons. In order to explain how a synapse affects the information transmission between neurons, a small Neural Network (NN) is depicted in Figure 3.1. This NN consists of three spiking neurons connected via two synapses. The post-synaptic neuron N_i collects signals (pre-synaptic spikes S_j and S_k) from pre-synaptic neurons N_j and N_k, and when the cumulated signals exceed a certain firing threshold, neuron N_i generates an output signal (post-synaptic spike S_i), which transmits through all its terminations. From the synapse functionality point of view (consider the synapse connecting neurons N_j and N_i), there are: (i) two input signals: pre-synaptic spike S_j, which is output spike of neuron N_j and post-synaptic spike S_i, which is the output spike of neuron N_i, and (ii) one output signal S_j^{out}, which will be transmitted to neuron N_i. The synaptic transmission efficiency (synaptic weight) is a function of the two input spikes, denoted as $W_{ji} = f(S_j, S_i)$. In general, the transmission efficiency is plastic and can either strengthen or weaken the signals transmitted via the synapse. This property is known as synaptic plasticity which is fundamental synaptic functionality and is believed to be the basis of learning and memory in human brain. There are two types of basic synaptic plasticity: Spike-Timing-Dependent Plasticity (STDP) and Long-Term Plasticity (including Long-Term Potentiation (LTP) and Long-Term Depression (LTD)). STDP is a widely used Hebbian synaptic learning rule [125],

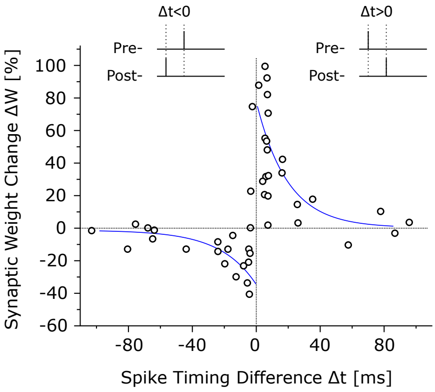


Figure 3.2: Biological Synapse STDP Measured Data [126].

which suggests that synaptic weight changes according to the relative timing difference between pre-synaptic spike and post-synaptic spike. When the pre-synaptic spike arrives at the synapse shortly before the post-synaptic spike, the synaptic weight increases, and this may lead to a persistent efficiency increase (LTP); otherwise, the synaptic weight decreases, and this may lead to a persistent efficiency decrease (LTD). When the two spikes are very close in time, i.e., very small timing difference, a large synaptic efficiency change occurs. Figure 3.2 depicts a standard STDP behavior based on biological measured data [126]. ΔW denotes the synaptic weight change and Δt denotes the spike timing difference. We denote by t_{pre} and t_{post}, the arrival time of pre-synaptic spike and post-synaptic spike, respectively. Thus the spike timing difference is calculated as $\Delta t = t_{post} - t_{pre}$. Even though the biological synaptic weight change behavior illustrated in Figure 3.2 (a) exhibits stochasticity, a widely accepted STDP interpolating model is as follows:

$$\Delta W(\Delta t) = \begin{cases} A_+ \cdot exp(-\Delta t/\tau_+), & \text{for} \quad \Delta t > 0 \\ -A_- \cdot exp(\Delta t/\tau_-), & \text{for} \quad \Delta t < 0, \end{cases} \tag{3.1}$$

where A_+ and A_- are parameters affecting the amplitude of synaptic weight change, and τ_+ and τ_- are time constants reflecting the time scale in which the potentiation and depression occurs.

3.2.2. GRAPHENE-BASED GENERIC SYNAPSE STRUCTURES

The proposed one-top-gate and two-top-gates graphene-based synapse generic structures are illustrated in Figure 3.3 and Figure 3.4 (a), respectively. They both consist of a single-layer Graphene Nanoribbon (GNR) located on top of an insulating material and a doped substrate serves as back-gate. When a drain-to-source bias voltage $V_d - V_s$ is applied, the GNR constitutes a conduction channel. By shaping the GNR sheet geometry and contact topologies as well as by applying external voltages (e.g., via top-gate(s) and back-gate), the GNR conductance G can be modulated. Figure 3.4 (b) depicts a conductance

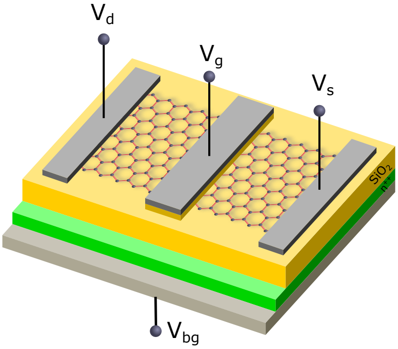


Figure 3.3: One-top-gate Graphene-based Synapse Structure.

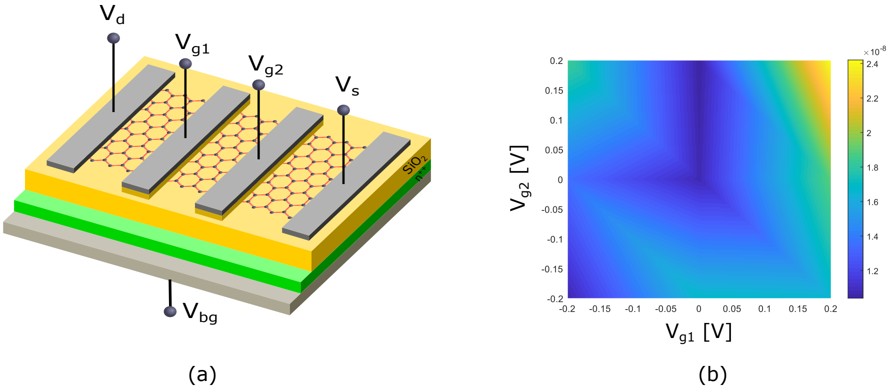


Figure 3.4: (a) Two-top-gates Graphene-based Synapse Structure and (b) its Conductance Map.

map (conductance vs. applied voltage) example of the proposed two-top-gates graphene-based synapse generic structure.

From the synaptic behavior point of view, the top-gate(s) is used for applying synaptic input spikes, e.g., for two-top-gates synapse, pre-synaptic spike is applied to top-gate-1 and post-synaptic spike to top-gate-2, corresponding to V_{g1} and V_{g2} in Figure 3.4 (b), respectively. The synaptic plasticity is reflected by the induced GNR conductance change, and the output spike current is represented by the drain-to-source current, which corresponds to the synaptic output spike S_j^{out} in Figure 3.1.

A key element in our proposal is the fact that, as experimentally observed, graphene-based device inherently exhibits interface traps [81], which are usually caused by defects in the top-gate oxide, and charges can be trapped from graphene to the interface or released. These trapping and deptrapping phenomena affect the top-gates conductance modulation ability, and as such, when applying an input spike the graphene-based device conductance and output current will depend on the cumulated previous activities in the artificial synapse. This dependence can be naturally utilized to implement time varying synaptic plasticity, e.g., STDP, LTP, and LTD. By carving the GNR geometry, the synapse conductance can be modulated such that it reflects a certain functionality. Thus, for every GNR topology a different conductance map can be obtained. As STDP weight change is reflected by the change of the device conductance, with different conductance maps we can have different STDP types. Furthermore, the GNR topology affects as well the trapping mechanism, which is fundamental in inducing plasticity.

In Section 3.5 we demonstrate that by: (i) Shaping the GNR into non-rectangular forms as well as changing the contact topologies, various synaptic plasticities can be obtained, which is not the case for previously proposed rectangular graphene-based synapses [100] (ii) Changing the back-gate voltage, we can emulate both excitatory and inhibitory synaptic behavior with the same graphene-based device, and (iii) Applying input spikes with different duration, the graphene-based synapse can emulate STDP with varying potentiation/depression time scale without affecting the amplitude of synaptic weight change.

3.3. SIMULATION FRAMEWORK

In this section we briefly present the simulation model for computing electronic transport properties of the proposed graphene-based synapses, and describe the simulation setups and employed methodology to emulate the desired synaptic plasticity (the GNR conductance change).

3.3.1. GNR ELECTRONIC TRANSPORT SIMULATION MODEL

In order to compute the GNR electronic transport properties, we make use of the atomistic-level hysteresis-aware graphene structures simulation model presented in Chapter 2. Specifically, we utilize the Tight-Binding approach to represent the system Hamiltonian, the Non-Equilibrium Green Function (NEGF) quantum transport model to solve the Schrödinger equation, and the Landauer-Büttiker formalism to derive the GNR current and conductance [102, 127]. The GNR potential distribution profile is obtained by solving a 3D Poisson equation self-consistently. Additionally, by calculating the equivalent voltage shift induced by interface trapped charges we account for the trapping/detrapping phenomena influence on the GNR device operation [128].

3.3.2. SIMULATION SETUP AND METHODOLOGY FOR ONE-TOP-GATE SYNAPSE

In order to apply the input spikes to the one-top-gate graphene synapse, we employ a single-input scheme, as exemplified in Figure 3.5. The signal applied as input to the synapse is computed as a superposition of the pre- and post-synaptic spikes (i.e., the voltage difference between the two spikes). We define T_{overlap} as the arrival time of the secondly arriving spike. To perform biologically plausible simulations, we considered data consistent with measured data from brain synapses: $-50\,\text{mV}$ to $50\,\text{mV}$ pre- and post-synaptic spikes voltage range, and $-60\,\text{ms}$ to $60\,\text{ms}$ Δt range (which covers the general time range for biological LTP and LTD) [129], [126].

As concerns the GNR for the one-top-gate synapse, we define its topology in Figure 3.6.

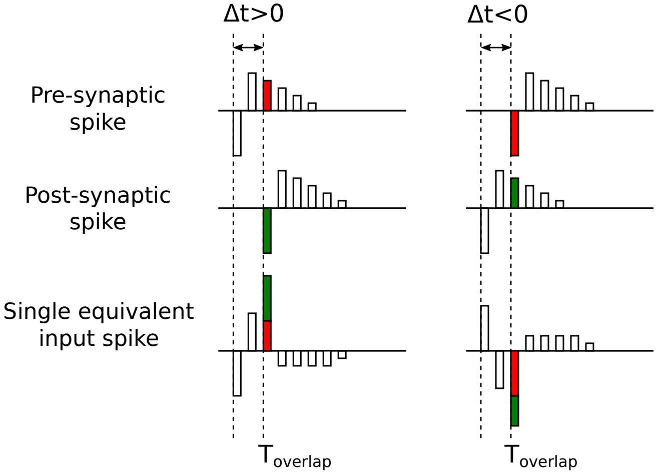


Figure 3.5: One-top-gate Synapse Single Input Spike Scheme.

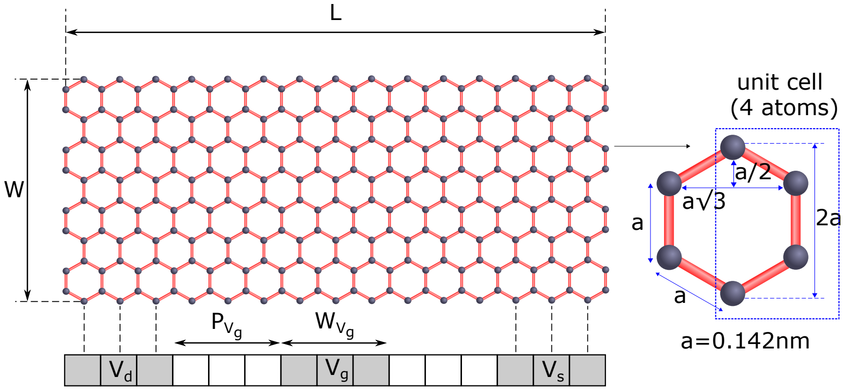


Figure 3.6: GNR Dimensions and Contacts Topology for One-top-gate Synapse.

In particular, W and L represent the GNR width and length, P_{V_g} signifies the distance between the top-gate and the drain contact, and W_{V_g} denotes the width of the top-gate contact. In our simulation, we considered multiple non-rectangular GNRs with different shapes but the same overall $W = 39a$ and $L = 28\sqrt{3}a$. For the top-gate contact we set $P_{V_g} = 8\sqrt{3}a$ and $W_{V_g} = 6\sqrt{3}a$, where a is 0.142 nm. Concerning the traps induced hysteresis, we assume a density of interface traps of $2.5 \times 10^{12} cm^{-2}(eV)^{-1}$, and we set a trapping/detrapping time constant of 20 ms [108, 113].

Subsequently, we present the overall design and simulation methodology. For a desired plasticity behavior, we first determine a potentially appropriate GNR geometry and drain-to-source and back-gate voltages. Subsequently, we subject the graphene synapse to a

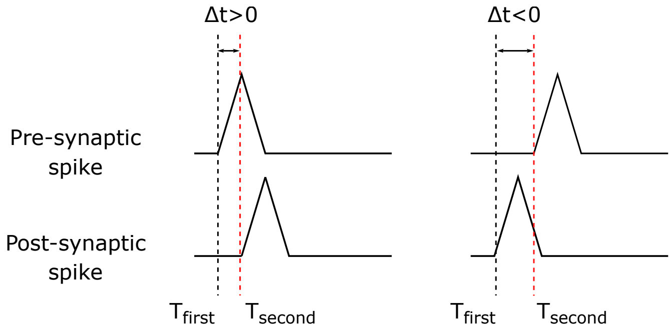


Figure 3.7: Spikes Timing Difference Indication for Two-top-gates Synapse.

train of spikes applied via the top-gate, one spike for each Δt in the considered range. Corresponding to each input spike, we then measured the synaptic weight change ΔW (the difference between the GNR conductance values at two consecutive time moments, i.e., T_{overlap} and the immediately previous time moment), and asses its compliance with the desired $\Delta W(\Delta t)$ plasticity curve. If results are not according with the desired plasticity we change the GNR geometry and bias voltages.

3.3.3. SIMULATION SETUP AND METHODOLOGY FOR TWO-TOP-GATES SYNAPSE

In order to emulate STDP behaviors with the two-top-gates synapse, the pre-synaptic spike and post-synaptic spike are applied to V_{g1} and V_{g2}, respectively. As exemplified in Figure 3.7, when the pre-synaptic spike arrives before the post-synaptic spike, the spike timing difference $\Delta t > 0$, and $\Delta t < 0$ when they arrive in inverse order. We define T_{first} as the arrival time of the first arriving spike and T_{second} as the arrival time of the second arriving spike. We assume that the applied input spikes lay into 70 mV to 180 mV voltage range and a spike duration of 2 ms. When no spike arrives the two top-gates are subjected to 70 mV voltage, mimicking the rest potential in biological spike trains. Thus for different spikes timing difference, the voltages applied on two top-gates varies at specific time moment, causing a time varying conductance change. Additionally, the long-term plasticity behavior is naturally captured by charge trapping/detrapping phenomena.

In Figure 3.8, the GNR dimension and contacts topologies are graphically defined for the two-top-gates synapse. Specifically, W and L represent the GNR sheet width and length, PV_{g1} indicates the distance between top-gate-1 and the drain contact, PV_{g2} indicates the the distance between top-gate-2 and the source contact, while WV_{g1} and WV_{g2} signify the width of top-gate-1 and top-gate-2, respectively. Note that $a = 0.142\,\text{nm}$ denotes the distance between two adjacent carbon atoms. Concerning the traps caused hysteresis, we assume a density of interface traps $2.363 \times 10^{13} cm^{-2}(eV)^{-1}$, and the trapping/detrapping time constant is set to 1 ms [108, 113].

To identify a GNR topology able to provide support for a targeted plasticity we perform a Design Space Exploration (DSE) by changing GNR dimension, shape, widths and positions

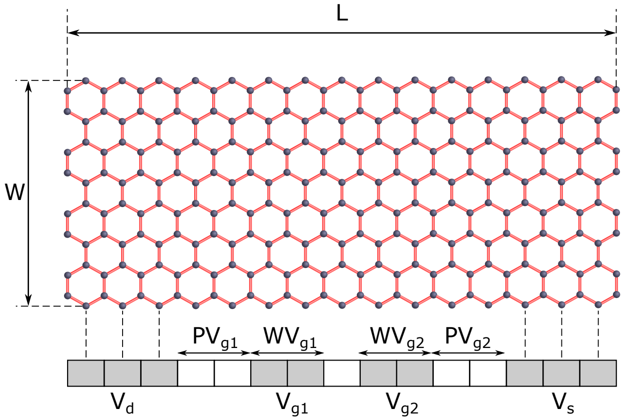


Figure 3.8: GNR Dimensions and Contacts Topology for Two-top-gates Synapse.

of the two top-gates, and back-gate voltage. For each relevant Δt value, we apply a pair of spike trains (pre-synaptic spike and post-synaptic spike) to the two top-gates and measure the synaptic weight change ΔW (the difference between GNR conductance values at time moments T_{first} and T_{second}, where $T_{\text{second}} - T_{\text{first}} = |\Delta t|$ as illustrated in Figure 3.7). We assess the resemblance of the obtained synaptic weight change with the desired $\Delta W(\Delta t)$ plasticity and if this is not satisfactory we continue DSE by changing the GNR topology parameters.

3.4. ONE-TOP-GATE GRAPHENE-BASED SYNAPSE EVALUATION

To evaluate the capabilities of proposed one-top-gate graphene synapse, we target 3 common plasticity types underlying balanced and potentiation dominated learning [130], [131]: Hebbian STDP with balanced LTD and LTP (Figure 3.9 (a)), LTP-biased Hebbian STDP (Figure 3.9 (d)), and LTP-biased Anti-Hebbian STDP (Figure 3.9 (g)).

Figure 3.9 (b) depicts the GNR synapse shape we obtained for the Hebbian STDP with balanced LTD and LTP scenario, biased at V_d =0.2 V and back-gate voltage V_{back} =0.2 V. The simulated synaptic weight change (conductance change) (Figure 3.9 (c)) indicates a good resemblance with the Hebbian STDP with balanced LTD and LTP weight change trend. In biological models, there is a certain randomness in the synapse reaction. We seek a synaptic reaction tendency closer to the plasticity models. When fitting the simulated conductance change with the canonical model in Equation 3.1, we obtained τ_+ =23 ms and τ_- =37 ms. Since for a biologically plausible input, we obtain an amplitude of the conductance change around 100%, which is consistent with biological synapse measured data shown in Figure 3.2, the proposed graphene synapse can enable potentially biologically plausible implementations (artificial synapses which can be interfaced with biological neurons in the context of, e.g., neural prosthetics).

Figure 3.9 (e) and (h) illustrate the obtained GNR synapse shapes for LTP-biased Hebbian STDP and LTP-biased anti-Hebbian STDP, respectively. The drain voltage V_d is set

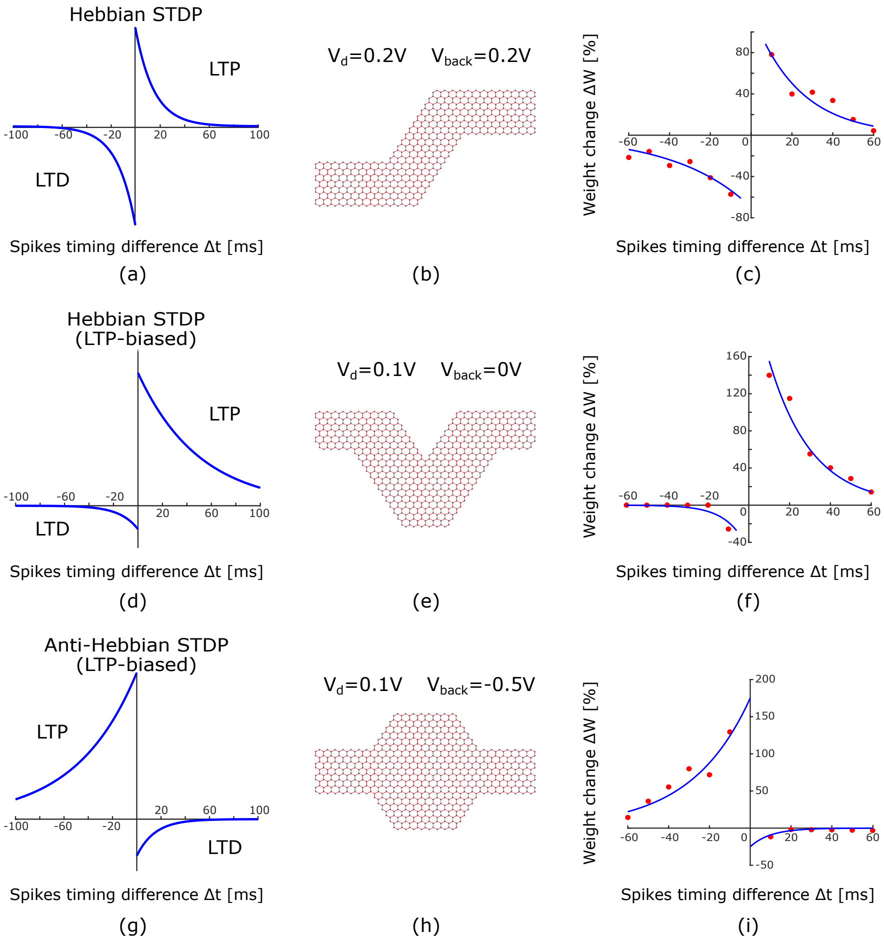


Figure 3.9: GNR Synapse Shapes and Corresponding STDP.

to 0.1 V for both shapes, while the applied back-gate voltage is 0 V and −0.5 V, respectively. The simulated synaptic weight change (conductance change) in Figure 3.9 (f) and (i) is temporally asymmetric, being dominated by (LTP) potentiation for both graphene synapses. When fitted with the model in Equation 3.1, we obtained $\tau_+ = 21$ ms and $\tau_- = 10$ ms for the LTP-biased Hebbian synapse, and $\tau_+ = 19$ ms and $\tau_- = 15$ ms for the LTP-biased Anti-Hebbian synapse.

A synapse can either exhibit excitatory behavior (i.e., synaptic weight potentiation when the pre-synaptic spike arrives before post-synaptic spike) or inhibitory behavior (i.e., synaptic weight depression when the pre-synaptic spike arrives before post-synaptic spike). Traditionally, 2 artificial synapses are employed, but we are able to obtain both excitatory and inhibitory behaviors with a single synapse, which is beneficial from the area and energy standpoints for large-scale integrations. For instance, the GNR synapse shape illustrated

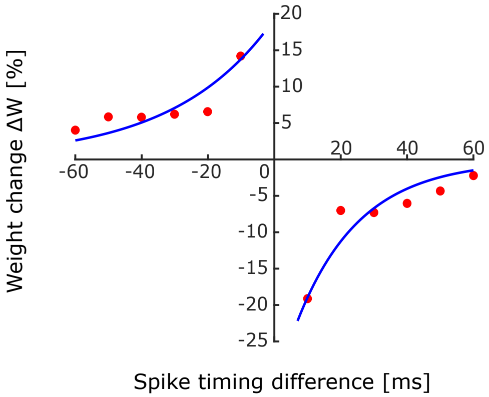


Figure 3.10: Anti-Hebbian STDP for GNR Synapse Shape from Figure 3.9 (b).

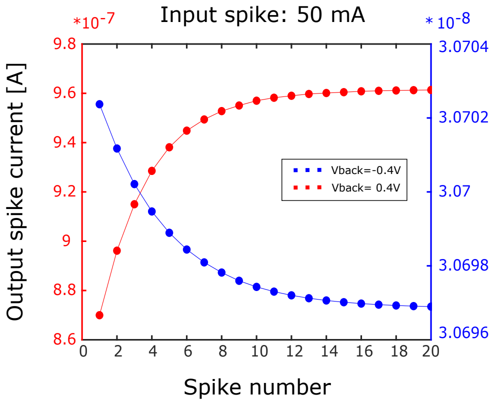


Figure 3.11: Output Spike Current for Long-term Plasticity.

in Figure 3.9 (b) exhibits an excitatory behavior but by simply changing the biasing gate voltage V_{back} from 0.2 V to 0.5 V, while the other GNR applied voltages (V_d and V_g) are identical the inhibitory counterpart is obtained, as depicted in Figure 3.10.

Apart from STDP, Long-Term Plasticity is a fundamental synaptic functionality, dominant for how the brain stores information, which is obtained when applying an identical spike consecutively. In our experiments we considered the GNR synapse shape from Figure 3.9 (h) and applied 50 mV input spikes with an intermission period between the spikes of 1 s. For each spike, we measured the GNR drain to source current, which represents the current of the output spike generated by the graphene synapse (e.g., S_j^{out} in Figure 3.1). The long lasting potentiation and depression are successfully emulated for the considered time range with positive and negative back-gate voltage, respectively, as illustrated in Figure 3.11.

Table 3.1: Two-top-gates GNR synapses topologies

Plasticity Type	W [a]	L [a]	PV_{g1} [a]	WV_{g1} [a]	PV_{g2} [a]	WV_{g2} [a]
Hebbian STDP	23	$25\sqrt{3}$	$9\sqrt{3}$	$3\sqrt{3}$	$2\sqrt{3}$	$6\sqrt{3}$
LTP-biased Hebbian STDP	29	$30\sqrt{3}$	$3\sqrt{3}$	$3\sqrt{3}$	$10\sqrt{3}$	$3\sqrt{3}$
anti-Hebbian STDP	29	$30\sqrt{3}$	$3\sqrt{3}$	$6\sqrt{3}$	$7\sqrt{3}$	$6\sqrt{3}$
LTP-biased anti-Hebbian STDP	23	$25\sqrt{3}$	$9\sqrt{3}$	$3\sqrt{3}$	$2\sqrt{3}$	$3\sqrt{3}$

3.5. Two-top-gates Graphene-based Synapse Evaluation

In this section we evaluate the capability of the proposed two-top-gates graphene-based synapses to emulate various plasticity types and investigate how input spike duration affects the achieved synaptic plasticity. Finally, we explore the effect of the obtained synaptic plasticity on the behavior of an example spiking neural network.

3.5.1. Spike-Timing-Dependent Plasticity and Long-Term Plasticity

In order to evaluate the ability of the proposed graphene-based synapse to emulate various plasticity types, we considered 4 different STDP types underlying balanced and potentiation dominated learning [130, 131]: Hebbian STDP with balanced LTP and LTD (Figure 3.12 (a)), LTP-biased Hebbian STDP (Figure 3.12 (d)), anti-Hebbian STDP with balanced LTP and LTD (Figure 3.13 (a)) and LTP-biased anti-Hebbian STDP (Figure 3.13 (d)). The GNR topologies (overall GNR width and length as well as the widths and positions for two top-gates) for the considered STDP types as determined by means of DSE are summarized in Table 3.1, where all values are expressed in term of $a = 0.142\,\text{nm}$, which is the distance between adjacent carbon atoms in a graphene unit cell. The GNR shapes for the 4 considered cases are depicted in Figure 3.12 (b) and (e), and Figure 3.13 (b) and (e), which capture their actual dimensions in term of carbon atoms.

Figure 3.12 (b) depicts the obtained GNR shape for Hebbian STDP with balanced LTP and LTD plasticity, with drain-to-source bias voltage $V_d = 0.2\text{V}$ and back-gate voltage $V_{back} = 0\text{V}$. The simulated synaptic weight change (conductance change) as presented in Figure 3.12 (c) has a good resemblance with the ideal Hebbian STDP with balanced LTD and LTP behavior trend (Figure 3.12 (a)). One can observe that the range of the obtained synaptic plasticity change is around 30%. To properly evaluate the obtained synaptic behavior, we define the STDP potentiation and depression time scale (t_+ and t_-, respectively) as the spikes timing difference Δt range in which the corresponding synaptic weight change ΔW is significant, i.e., larger than 0.1%. Thus, when for $t_- < \Delta t < t_+$ the synaptic weight change $|\Delta W| > 0.1\%$. For the simulated Hebbian STDP, we obtain a potentiation time scale $t_+ = 2.6\,\text{ms}$ and depression time scale $t_- = -2.5\,\text{ms}$, which indicates

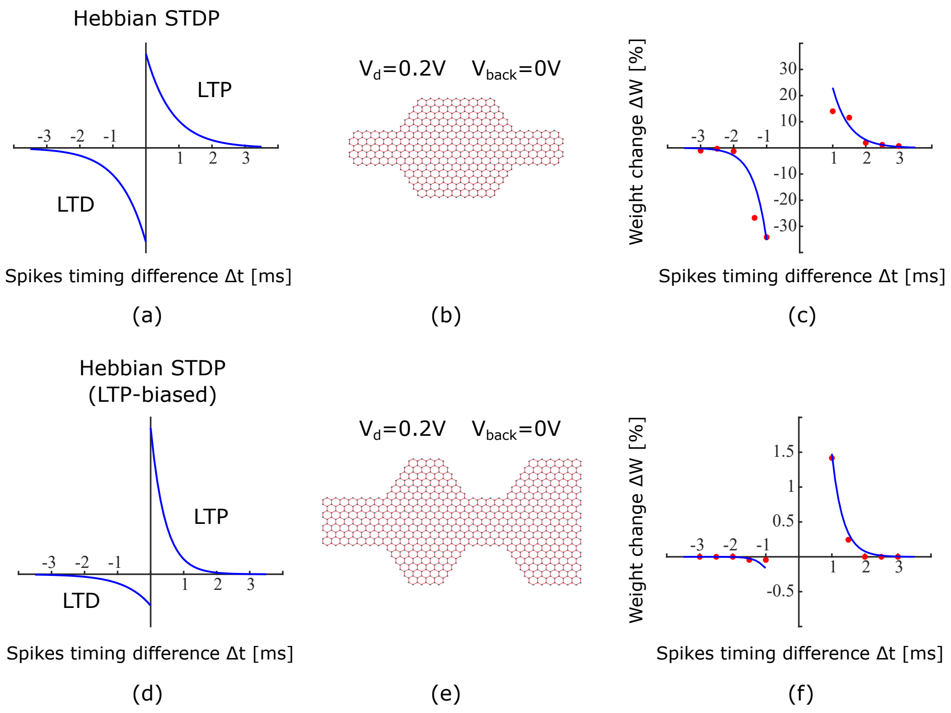


Figure 3.12: GNR Synapse Shapes and Obtained Plasticity Corresponds to Hebbian STDP and LTP-biased Hebbian STDP.

that the synaptic weight change is tiny when the spikes timing difference is beyond this range($[-2.5\,\mathrm{ms}, 2.6\,\mathrm{ms}]$). We further investigate how different input spikes duration affects the obtained STDP potentiation/depression time scale in Section 3.5.2.

Figure 3.12 (e), Figure 3.13 (b), and Figure 3.13 (e) illustrate the obtained GNR shapes for LTP-biased Hebbian STDP, anti-Hebbian STDP with balanced LTP and LTD, and LTP-biased anti-Hebbian STDP, respectively. In all these cases, the drain-to-source bias voltage is $V_d = 0.2\,\mathrm{V}$ and the back-gate voltage $V_{back} = 0\,\mathrm{V}$. For LTP-biased Hebbian STDP, the simulated synaptic plasticity in Figure 3.12 (f) is temporally asymmetric, and we observe that the amplitude of synaptic weight potentiation is about 3 times larger than synaptic weight depression, which results in a LTP-biased behavior. Compared with the amplitudes obtained for Hebbian STDP with balanced LTP and LTD, the amplitudes in this case are relatively smaller. For obtained anti-Hebbian STDP with balanced LTP and LTD in Figure 3.13 (c), the synaptic weight potentiation amplitude is around 2.5% while the depression amplitude is around 1.5%, which is approximately symmetric and exhibits a good resemblance with the ideal behavior depicted in Figure 3.13 (a). As for the simulated LTP-biased anti-Hebbian STDP in Figure 3.13 (f), the synaptic weight potentiation amplitude is about 30% while the depression amplitude is about 5%, thus exhibiting the obvious LTP-biased

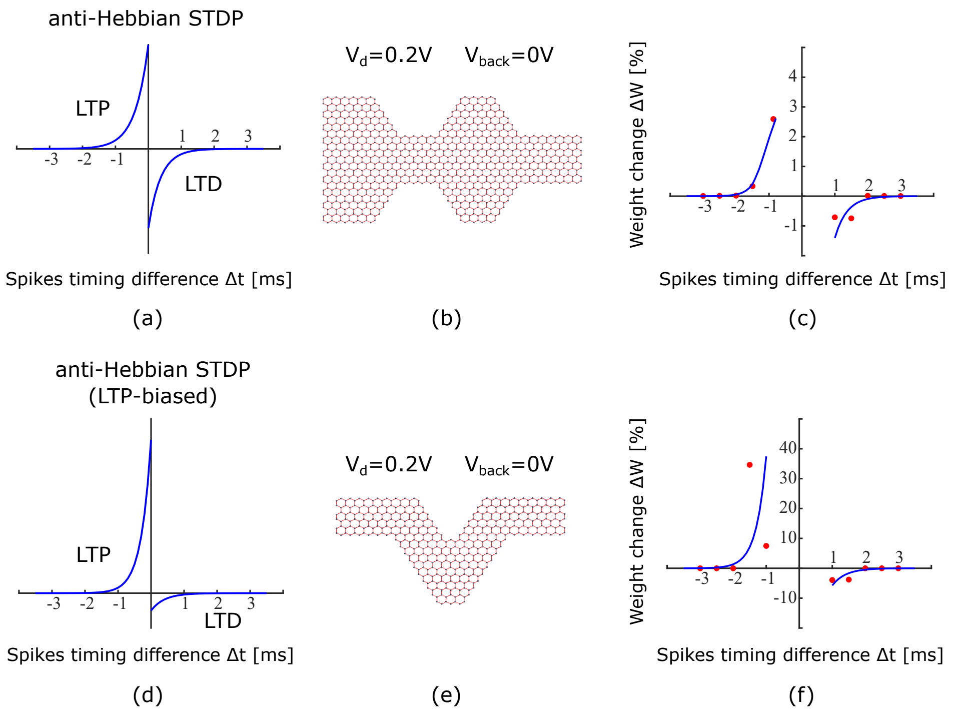


Figure 3.13: GNR Synapse Shapes and Obtained Plasticity Corresponds to Anti-Hebbian STDP and LTP-biased Anti-Hebbian STDP.

behavior. Note that for these 3 cases (LTP-biased Hebbian STDP, balanced anti-Hebbian STDP and LTP-biased anti-Hebbian STDP), the obtained plasticity has similar STDP potentiation/depression time scale which is around $[-2\,\text{ms}, 2\,\text{ms}]$.

With the proposed two-top-gates graphene-based synapses, we are able to obtain both excitatory and inhibitory synaptic behaviors with the same graphene-based device by changing the back-gate voltage. For instance, the GNR synapse shape illustrated in Figure 3.12 (b) exhibits an excitatory synaptic behavior. By simply changing the back-gate voltage from 0 V to −0.15 V while keeping the GNR shape, contact topologies, and applied voltages identical, the inhibitory synaptic behavior is obtained as depicted in Figure 3.14 (a). Similarly, the GNR shape depicted in Figure 3.13 (e) exhibits an inhibitory synaptic behavior. By simply changing the back-gate voltage from 0 V to −0.05 V, the corresponding excitatory synaptic behavior is obtained as illustrated in Figure 3.14 (b).

To obtain Long-Term Plasticity, an identical spike is applied to the top-gate-1 consecutively while a constant voltage (the rest potential) is applied to top-gate-2. The applied spikes and the voltage mimicking the rest potential are consistent with the ones used in STDP simulations. In our experiment we considered the GNR synapse shapes from Figure 3.12 (b) and Figure 3.13 (e) and apply an input spike train consisting of identical spike

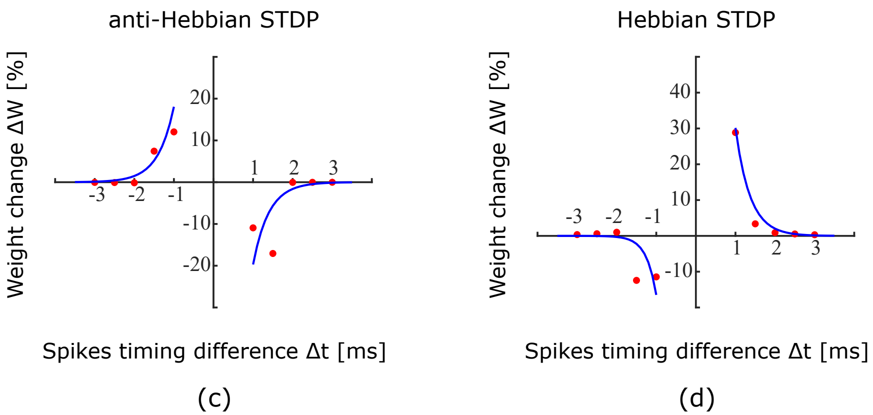


Figure 3.14: (a) Corresponding Inhibitory Synaptic Behavior for Synapse Shape Depicted in Figure 3.12 (b), and (b) Corresponding Excitatory Synaptic Behavior for Synapse Shape Depicted in Figure 3.13 (e).

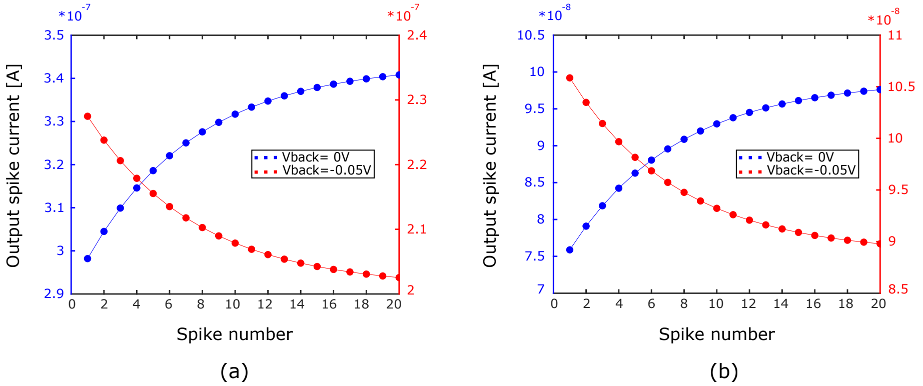
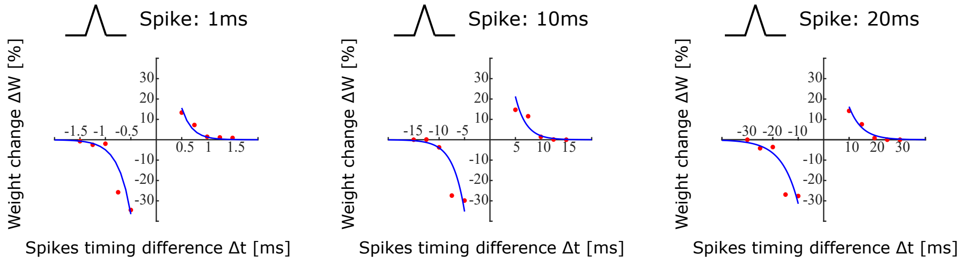


Figure 3.15: Output Spike Current for Long-term Plasticity. (a) Graphene-based Synapse Illustrated in Figure 3.12 (b). (b) Graphene-based Synapse Illustrated in Figure 3.13 (e).

with inter-spike period 1 ms. For each spike, we measure the GNR drain to source current, which represents the output spike current generated by the synapse (e.g., S_j^{out} in Figure 3.1). The long term potentiation and depression are obtained by properly changing the back-gate voltage for the two GNR synapse shapes, respectively. For both GNR synapse shapes in Figure 3.12 (b) and Figure 3.13 (b), LTP is obtained with back-gate voltage $V_{back} = 0\,\text{V}$ and LTD is obtained with back-gate voltage $V_{back} = -0.05\,\text{V}$, as illustrated in Figure 3.15 (a) and (b), respectively.

Our simulation results demonstrate the capabilities of the proposed artificial synapses to emulate various types of synaptic plasticity. The design and simulation methodology



(a) (b) (c)

Figure 3.16: Simulated STDP with Input Spike Duration (a) 1 ms, (b) 10 ms, and (c) 20 ms.

is generic, more synaptic plasticity types can be potentially obtained beyond the aforementioned ones. The proposed graphene-based synapses have small area (max. 30 nm^2) and operate with low operating voltage (0.2 V drain-to-source bias voltage and max. 0.18 V input spike voltage), which are desired properties for large-scale neuromorphic computation systems. To get further inside into our proposed graphene-based synapses, we investigate in the following subsection how input spikes with different spike duration affect the obtained synaptic plasticity.

3.5.2. SPIKE-TIMING-DEPENDENT PLASTICITY TIME SCALE VARIATION

To investigate the effect of varying input spike duration on obtained STDP, we consider the GNR synapse in Figure 3.12 (b) and apply input spike with duration of 1 ms, 10 ms, and 20 ms. For each case input spikes with the specific duration are applied to the graphene-based synapse and the resulting plasticity are recorded. The GNR geometry, contact topologies, drain-to-source biased voltage and back-gate voltage are identical with the previous STDP simulation. The obtained STDPs are depicted in Figure 3.16. The simulated STDP in Figure 3.16 (a) corresponds to spike duration 1 ms. The obtained potentiation and depression time scale $t_+ = 1.1\,\text{ms}$ and $t_- = -1.5\,\text{ms}$, which is smaller than the one obtained with a spike duration of 2 ms. We observe the amplitude of synaptic weight potentiation and depression is around 20% and −30%, respectively, and are almost identical with the ones obtained with a spike duration of 2 ms. The obtained STDP for 10 ms and 20 ms spikes are illustrated in Figure 3.16 (b) and (c), respectively. In both 10 ms and 20 ms cases, the amplitude of the synaptic weight change is around 30%, while the STDP time scale is $[-13.3\,\text{ms}, 11.5\,\text{ms}]$ and $[-32.2\,\text{ms}, 24.1\,\text{ms}]$, respectively. The simulation results suggest that the input spike duration can affect the obtained STDP potentiation/depression time scale, while having little effect on the synaptic weight change amplitude.

Figure 3.17 statistically illustrates the relation between the input spike duration and obtained STDP potentiation/depression time scale, when considering [1 ms, 2 ms, 5 ms, 10 ms, 15 ms, 20 ms] spikes. One can observe an approximately linear relation between the input spike duration and obtained STDP potentiation/depression time scale. The synaptic weight change amplitudes in all cases are identical. Thus we conclude that by changing the input spike

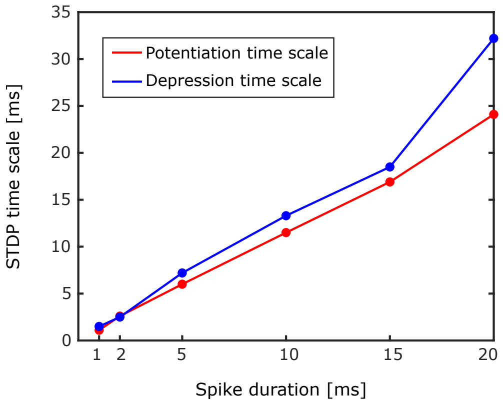


Figure 3.17: STDP Potentiation and Depression Time Scale vs. Input Spike Duration.

duration, the proposed synapse can achieve STDP with different potentiation/depression time scale without affecting the amplitudes of synaptic weight change.

3.5.3. Spiking Neural Network Impact

To get some preliminary inside on the implication of our proposal at the higher level we consider a small Spiking Neural Network (SNN) with GNR-based synapses and simulate its behavior by means of the NEST simulator [124]. As illustrated in Figure 3.18 (a), this SNN has a fully connected network topology, and consists of five leaky-integrate-and-fire neurons connected via synapses. Since in a given neural network the spikes are alike, the form of a single spike doesn't carry any information, but the number and timing of spikes matter [53], we concentrate on investigating how the proposed graphene-based synapse affects the SNN's firing behavior. As a thorough analysis of any synaptic plasticity influence on neural network's behavior is out of the scope of this chapter, we restrict the investigation to the cases when the simulated SNN is constructed with the biological synapses (as illustrated in Figure 3.2) [126] and proposed graphene-based synapses. The plasticity model described in Equation (3.1) is utilized as standard STDP model in the NEST simulator and we specify synaptic behaviors in SNNs by fitting plasticity data (e.g., biological measured data and simulation results with graphene-based synapses) with the STDP model for different cases. For simplicity, we call the SNN with synaptic behavior specified by biological measured data as SNN with biological STDP, and the SNN with synaptic behavior specified by graphene-based synapses simulation data as SNN with graphene-based STDP (e.g., SNN with graphene-based Hebbian STDP, SNN with graphene-based anti-Hebbian STDP). For the SNN simulation, an input spike train (with spike times sampled from a Poisson distribution) with a firing frequency of 10 kHz is applied to all neurons, and the firing events are recorded. Figure 3.18 depicts the inter-spike interval distribution of the input spike train. Every simulation is performed with one specific synaptic behavior, while the other settings are keep consistent. The overall simulation time is set to 500 ms.

We consider SNN with biological STDP as baseline and then perform simulations for

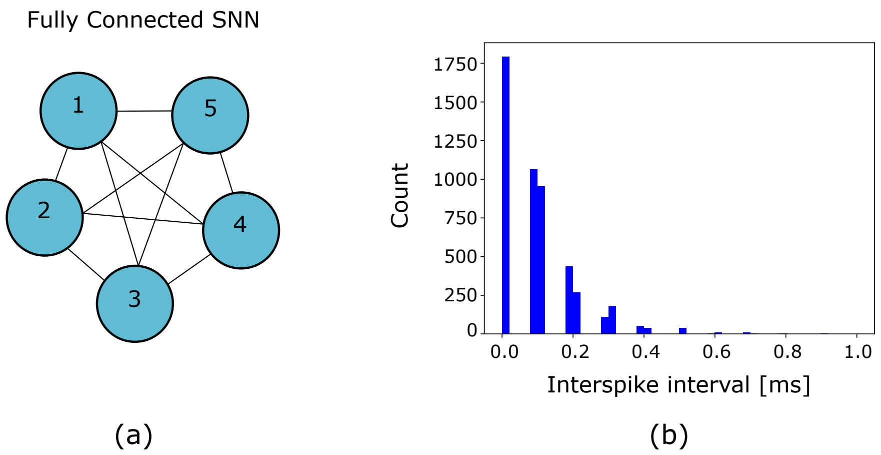


Figure 3.18: The Simulated Fully Connected SNN and The Inter-spike Interval Distribution of The Input Spike Train

Table 3.2: SNN Total Output Firing Number

Synaptic Plasticity	Number[#]
STDP with biological STDP	220
Graphene-based Hebbian STDP (2 ms)	820
Graphene-based anti-Hebbian STDP (2 ms)	80
Graphene-based Hebbian STDP (1 ms)	290
Graphene-based Hebbian STDP (20 ms)	200

SNN with graphene-based Hebbian STDP (in Figure 3.12 (c)) and anti-Hebbian STDP (in Figure 3.13 (c)), which correspond to excitatory and inhibitory synaptic behaviors, respectively. Furthermore, graphene-based Hebbian STDPs obtained with different input spike durations (e.g.,spike duration 1 ms in Figure 3.16 (a) and spike duration 20 ms in Figure 3.16 (c)) are also utilized to perform the simulation. Table 3.2 summarizes the obtained total number of firing events for all cases. The graphene-based synaptic plasticity types are indicated by the applied spike duration in previous GNR simulations, e.g., graphene-based Hebbian STDP (2 ms) indicates the corresponding plasticity obtained with input spike duration of 2 ms.

Figure 3.19 illustrates the simulation results for SNN with biological STDP. The top panel shows a raster plot for firing events belonging to the five neurons. Each dot in the plot indicates the occurrence time of one firing event and all dots in the same row belong to the same neuron. The histogram plot in the bottom panel represents the SNN firing rate at each time moment. As the simulated SNN has fully connected topology and inputs are applied to all neurons, the firing events for all neurons are identical, which can be observed through the same firing events distribution for the five neurons. As for the firing

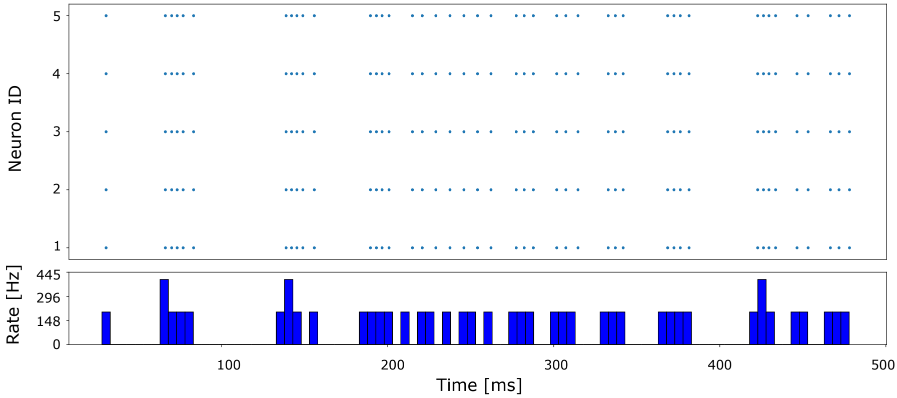


Figure 3.19: SNN Firing Events with Biological STDP.

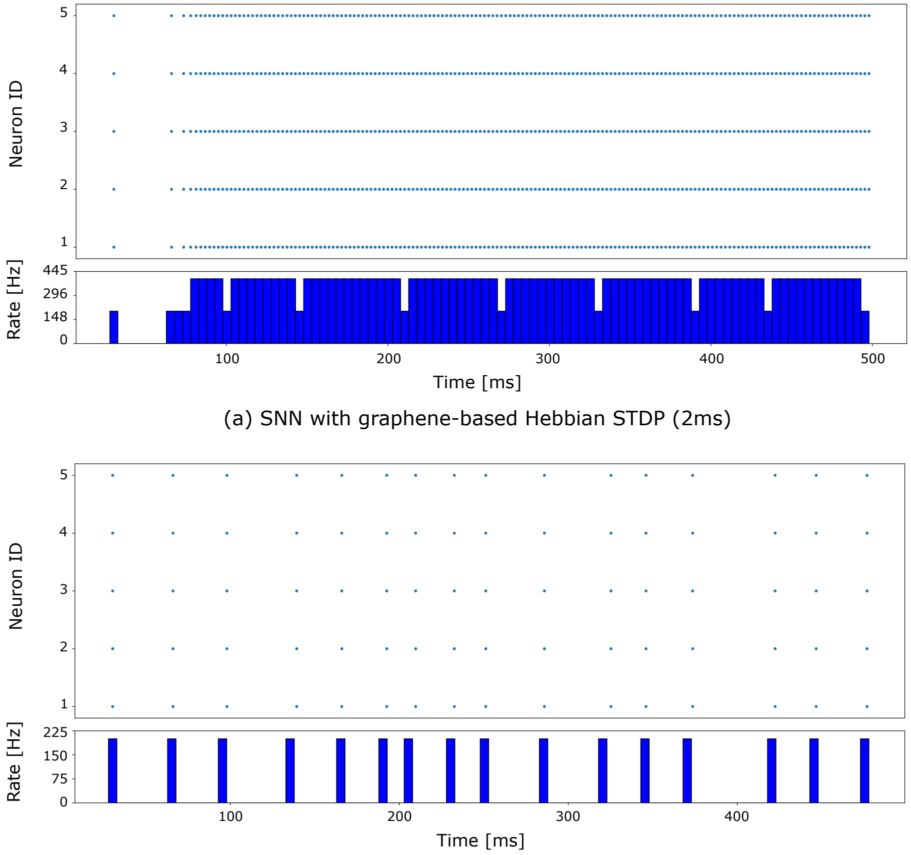
rate in the histogram plot it exhibits a sparse distribution and there is no obvious tendency to increase or decrease. The total firing events number in this case is 220.

Figure 3.20 (a) depicts the simulation results for SNN with graphene-based Hebbian STDP (2 ms). One can observe a dramatic increase of the firing rate during the first 100 ms and then the SNN keeps a relatively high firing rate until the end. The total firing events number is 820, which is 3.7 times larger than the one obtained in SNN with biological STDP, which suggests that the graphene-based Hebbian STDP is able to exhibit significant strengthening effect on the SNN's firing events.

The simulation results for SNN with graphene-based anti-Hebbian STDP (2 ms) is illustrated in Figure 3.20 (b). One can observe that the general firing rate is smaller than the aforementioned two SNN simulations, and it exhibits a sparser firing events distribution. The total firing events number is 80, which suggests that proposed graphene-based anti-Hebbian synapse can properly emulate inhibitory synaptic behavior and suppress the SNN's firing events.

To evaluate the implications of the spike length we perform simulations also for SNNs with graphene-based Hebbian STDP (1 ms) and with graphene-based Hebbian STDP (20 ms), which results are illustrated in Figure 3.21 (a) and (b), respectively. We observe that for the SNN with graphene-based Hebbian STDP (1 ms), there is a gradually increase of the firing rate, and the total firing events number is 290. Compared with SNN with biological STDP, this graphene-based synapse can exhibit strengthening effect, but is weaker than the one observed in Figure 3.20 (a). As for the SNN with graphene-based Hebbian STDP 20 ms, the firing events have sparse distribution and don't exhibit any strengthening or weakening tendency, which is similar to the response of the SNN with biological STDP. As presented in Section 3.5.2, graphene-based Hebbian STDPs with spike duration (1 ms) and 20 ms have identical synaptic weight change amplitude and the potentiation/depression time scales are different. The firing events observed in two cases indicate that the potential/depression time scale has an obvious influence on the SNN's behaviors.

In order to investigate how input spike frequency affects the SNN output firing events, we consider 3 different STDP types, i.e., biological STDP, graphene-based Hebbian STDP



(a) SNN with graphene-based Hebbian STDP (2ms)

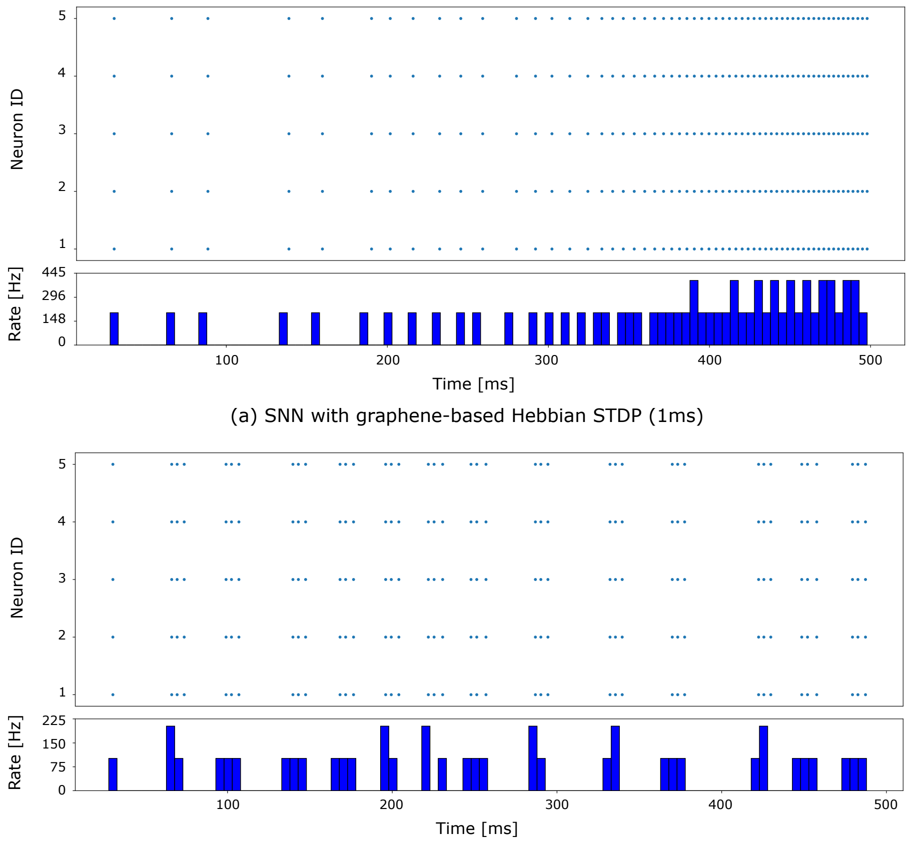
(b) SNN with graphene-based anti-Hebbian STDP (2ms)

Figure 3.20: SNN Firing Events with Proposed Hebbian STDP and Anti-Hebbian STDP

(2 ms), and graphene-based Hebbian STDP (20 ms), and vary the SNN input spike frequency from 10 kHz to 30 kHz. Simulation results are presented in Figure 3.22 and indicate that the number of SNN output firing events for all 3 STDP cases monotonously varies with respect to the input spikes frequency. Specifically, we observe: (i) an increase of 13% in the SNN output firing rate for the graphene-based Hebbian STDP (2 ms), and (ii) ≈ 2× increases for both biological STDP and graphene-based Hebbian STDP (20 ms)). Furthermore, we notice a close resemblance between the SNN firing rates for the biological and the Hebbian 20 ms STDP cases for all input spike frequencies, which is consistent with the results reported in aforementioned simulations.

3.5.4. Synapses Implementations in Current Technologies

To have a better view of the synapse designs landscape, and investigate in this context the potential of using graphene-based synapses for large scale neuromorphic systems, we



(a) SNN with graphene-based Hebbian STDP (1ms)

(b) SNN with graphene-based Hebbian STDP (20ms)

Figure 3.21: SNN Firing Events with Proposed Hebbian STDP Obtained with Different Spike Duration.

consider different technologies, i.e., CMOS, memristor, and evaluate comparatively the synaptic implementations summarized in Table 3.3.

From the large-scale implementations' suitability point of view, we look at the synapse footprint and operating voltage. Area-wise the synapse designs based on emerging technologies, i.e., memristors [132, 133] and graphene [101], have generally compact implementations, and are thus better equipped than CMOS-based counterparts [69, 134] for a high density of integration. The proposed GNR synapse design has 30 nm^2 footprint, which is 2 orders and 5 orders of magnitude smaller than memristor and CMOS designs, respectively. Furthermore, our GNR synapse operates at low voltage (0.2 V), at least 5× smaller than memristor and CMOS based counterparts, which is essential when striving for brain-akin energy efficiency envelopes. From the functionality point of view, to emulate abundant enough neural network dynamics, synapse designs require flexibility for mimicking an enriched repertoire of synaptic plasticities. Memristor and CMOS synapse

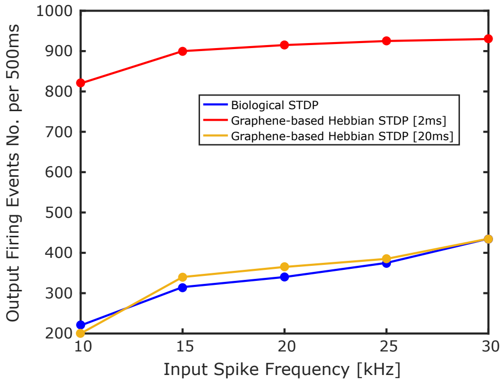


Figure 3.22: SNN Output Firing Events Number with Different Input Spike Frequencies.

designs, in order to adapt to different STDP types, rely on external control signals and additional circuitry that generates for each STDP type, input spikes with the required shapes [69, 135]. Differently from the aforementioned adaptation approaches, the proposed GNR-based synapse structure can accommodate various STDP types within a single graphene-based device simply by carving a different GNR geometry for each STDP type. So, for different plasticity types, we have different devices, i.e., the same single graphene-based synapse structure but with another GNR geometry. The fact that the STDP adaptation ability is obtained with the same input spike shapes for all STDP types and that the entire synaptic functionality can be encapsulated within a single device, makes the proposed GNR-based synapse structure a versatile modular plug-in component for neural network implementations. When compared to existing graphene-based synapse designs [100, 101], our proposal outperforms its counterparts by requiring 30× to 5 orders of magnitude less area and by straightforward emulation of different plasticity types. The comparison clearly indicates that the proposed GNR-based synapse, by its low real-estate requirements, small operating voltage, as well as adaptation versatility to different synaptic plasticity types, exhibits a high potential for large scale and functionally diverse neuromorphic computing platform implementations.

3.6. CONCLUSIONS

In this chapter we proposed generic one- and two- top gates graphene-based synapse structures. We demonstrated that by properly changing the GNR shape and contact topologies, and applying external voltages, the proposed graphene-based synapses are capable of mimicking various synaptic plasticity types. We successfully emulated two fundamental synaptic functionalities: Spike-Timing-Dependent Plasticity (STDP) and Long-Term Plasticity, including Long-Term Potentiation (LTP) and Long-Term Depression (LTD), which are foundational for human brain learning and remembering capabilities. Moreover, the same graphene-based synapse can emulate both LTD and LTP by simply changing its back-gate voltage. Given that we relied on a generic methodology to identify the appropri-

Table 3.3: Synapse Implementations with Different Technologies

Synapse Type	Operating Voltage	Area	STDP Type Adaptation
Biological [126]	Spike: $[-40, 70]\,mV$	-	-
CMOS [69]	Spike: 1 V	-	Yes
CMOS [134]	Supply:1 V	$\approx 1.3 \times 10^7\,nm^2$	No
Memristor [132]	Spike: $[-1, 1]\,V$	-	No
Memristor [133]	Spike: $[-5, 1.5]$V	$\approx 1.0 \times 10^4\,nm^2$	No
Graphene [100]	Supply:0.1 V, Spike: 2 V	$\approx 9.0 \times 10^6\,nm^2$	No
Graphene [101]	Supply:0.1 V	$\approx 9.0 \times 10^2\,nm^2$	No
Proposed Design	Supply:0.2 V, Spike: 0.18 V	$\approx 3.0 \times 10^1\,nm^2$	Yes

ate GNR topology for a desired synaptic plasticity our proposal is by no means restricted to the 4 STDP types and 2 LTP types considered in the chapter. Our simulations indicate that the one-top-gate synapse can achieve the plasticity change of 100% provided by natural synapses. The two-top-gates synapse exhibits STDP with spike duration dependent potentiation/depression time scale without changing the obtained synaptic weight change amplitude while achieving a maximum of 30% synaptic weight change and potentiation/depression time scale range from $[-1.5\,\text{ms}, 1.1\,\text{ms}]$ to $[-32.2\,\text{ms}, 24.1\,\text{ms}]$. This property makes the proposed synapses versatile in emulating various plasticity types suitable for different application scenarios. Furthermore, we explored the effect of two-top-gates synapse at the SNN level by performing NEST based simulations. Our experiments indicated a strong corelation between the synaptic plasticity type, i.e., Hebbian and anti-Hebbian, and the number of firing events in the network and that the number of SNN output firing events monotonously varies with respect to the input spikes frequency. For Hebbian STDP and a spike duration of 20 ms we obtained an SNN behavior similar with the one provided by the same SNN with biological STDP. The proposed graphene-based synapses have small area (30 nm^2), operate in the 100 mV bias and input range, and can emulate various plasticity types, which are making them very promising candidates for scalable energy-efficient neuromorphic system implementations.

4

Graphene-based Nonlinear Leaky Integrate-and-Fire Spiking Neuron

In this chapter, we propose an ultra-compact, all graphene-based nonlinear Leaky Integrate-and-Fire spiking neuron. We validate, by means of SPICE simulations, the basic nonlinear Leaky Integrate-and-Fire (LIF) neuron functionality under periodic input spike trains and noisy stochastic input. Our results indicate robustness to neuronal signals variability, and regular output firing rate statistics with a slowly decreasing trend and < 1 *interspike interval variation coefficient, when increasing the input firing rate from* 20 *to* 200 *spikes per second. For all simulation, we used spike duration and amplitude of* 2 ms *and* 100 mV*, respectively, which are comparable to those observed in biological neurons. Moreover, the low area footprint (GNR-based device area of max.* 36 nm^2*) and low operating voltage (*200 mV *supply voltage) prove the suitability of our proposal for large-scale integration.*

The content of this chapter is based on the following paper:

H. Wang, N. Cucu Laurenciu, Y. Jiang, and S.D. Cotofana, "Ultra-compact, Entirely Graphene-based Nonlinear Leaky Integrate-and-Fire Spiking Neuron", *IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1-5, 2020.

4.1. Introduction

Since the nervous system, which supports the human brain complex functionality, comprises billions of neurons, it makes the design and implementation of large-scale neuromorphic computing systems an extremely challenging task. State-of-the-art CMOS-based artificial neurons use complex CMOS circuitry and have a relatively high power consumption [136], [137], which limit the complexity, scalability, and energy efficiency of achievable neuromorphic system implementations. Besides, CMOS-based neurons cannot intrinsically mimic the analog behavior of biological neurons. Recently, emerging resistive switching memory devices [66] attracted interest and have been utilized in spiking neurons implementations [138], [71], due to their analog behavior, ability to restore the state memory, and good scalability. However, they suffer from resistive state temporal and spatial variability and undesired stochastic behavior, which may cause neuromorphic systems instability. Artificial neurons based on the phase-change devices were also proposed as an alternative for scalable neuromorphic systems [36], [139] as their accumulation property can provide a proper electronic mimicry of spiking neurons membrane potential dynamics. However, phase change neuron implementations require additional CMOS circuitry to emulate the neuron functionality and rely on externally generated auxiliary signals that control the basic functionality of phase-change devices. They also operate at relatively high voltages, which impede the implementation of energy efficient neuromorphic systems.

In this chapter we investigate graphene's potential towards low cost and energy effective implementations of spiking neurons. Specifically, we propose an all graphene-based ultra-compact and low voltage neuron, which is able to emulate the essential features of spiking neurons, including the membrane potential accumulation, the firing event, the refractory effect, and the output spike generation. The proposed neuron is operated with voltage ranges akin to those of biological neurons, which makes it a good candidate for biologically plausible utilization scenarios. The neuron consists of 6 GNR-based devices controlled via top-gate voltages, one of them emulating the membrane potential dynamics, and the remaining 5 generating the necessary control signals as well as the output spikes. We validate the basic nonlinear Leaky Integrate-and-Fire (LIF) neuron functionality with periodic input spike trains. We further evaluate the neuron output spike response when subjected to noisy stochastic input. All experiments are carried out by means of SPICE simulation. The obtained results indicate robustness to neuronal signals variability, and regular output firing rate statistics with a slowly decreasing trend and < 1 interspike interval variation coefficient, when increasing the input firing rates from 20 to 200 spikes per second. For all simulation, we used spike duration and amplitude of 2 ms and 100 mV, respectively, which are comparable to those observed in biological neurons. Note that, the low area footprint (GNR-based device area of max. 36 nm^2) and low energy consumption (200 mV supply voltage) prove the suitability of our proposal for large-scale integration.

The remaining of this chapter is organized as follows: Section 4.2 explains the basic concepts of nonlinear leaky integrate-and-fire neuron, and introduces the basic building block for graphene-based neurons. In Section 4.3 we describe the design of the proposed graphene-based neuron and explain its operation principle. Section 4.4 presents simulation results and Section 4.5 concludes the chapter.

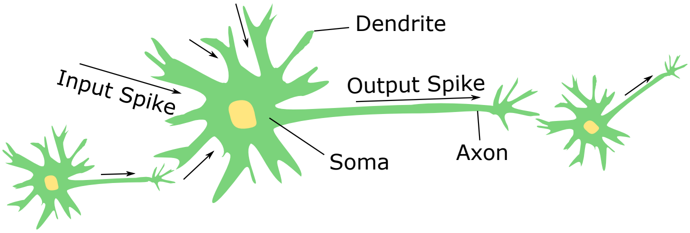


Figure 4.1: Neuron Structure.

4.2. BACKGROUND

In this section we introduce the neuron structure, the nonlinear Leaky Integrate-and-Fire (LIF) model, and the fundamental building block for the proposed graphene-based neuron.

As illustrated in Figure 4.1, a neuron comprises: (i) a soma, which is the neuron's cell body where the main neuronal dynamics occur (e.g., membrane potential evolution, spike generation, and refractory effect), (ii) dendrites, which connect the neuron with other neurons, receive and process input spikes, and generate neuronal input trains to the soma, and (iii) an axon, which is a long nerve cell, that transmits the output spike generated by the soma to neighbouring neurons. Various neuron models are proposed to describe the behavior of biological neurons, among which the Integrate-and-Fire neuron model attracts particular interest, due to its low complexity that makes it easy to analyze neuronal behavior while being able to capture the essential properties of biological neurons. In a standard nonlinear Leaky Integrate-and-Fire (LIF) neuron, the membrane potential evolution is in line with the following equation:

$$du/dt = F(u) + G(u) \cdot I, \tag{4.1}$$

where u is the membrane potential, $F(u)$ denotes a voltage-dependent leak term, and $G(u)$ is the voltage-dependent input resistance, which accounts for the membrane potential accumulation due to the neuron input current I. The neuronal dynamics of a nonlinear LIF neuron can be described via: (i) an integration process, when the membrane potential u increases continuously (starting from the resting potential u_{rest}) due to input spikes contributions, (ii) a firing event that generates a neuron output spike when the membrane potential reaches a certain firing threshold θ and then immediately resets to a new value $u_{\text{reset}} < u_{\text{rest}}$, and (iii) a refractory period, during which the neuron cannot fire, and the membrane potential is reset to the resting potential u_{rest}.

To implement the nonlinear LIF neuron with graphene-based devices, we rely on the basic building block, illustrated in Figure 4.2 (a), which consists of a monolayer Graphene Nanoribbon (GNR) located above an insulating material and a doped substrate that serves as back-gate. The GNR works as a conduction channel when applying a bias voltage V_{d}-V_{s} between the source and drain terminals. The GNR conductance can be modulated by changing the graphene sheet geometry and the contacts topology as well as by means of external voltages via the top/back gates. Figure 4.2 (b) illustrates the equivalent capacitive circuit of the device in Figure 4.2 (a), where C_{ox} is the top gate oxide capacitance, C_{q} the

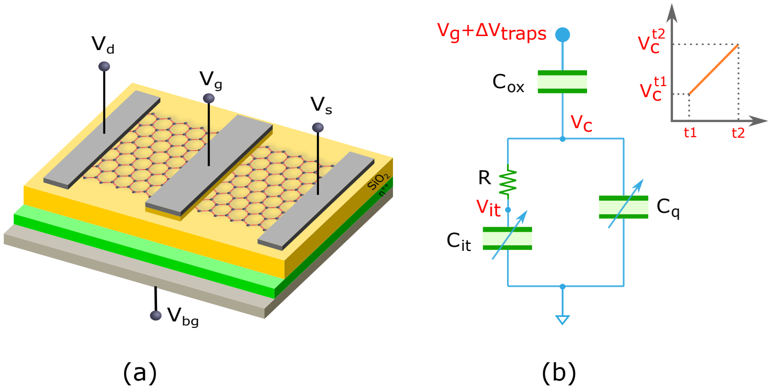


Figure 4.2: Graphene-based Device for Artificial Neurons: (a) Basic GNR-based Device, and (b) Equivalent Traps-aware Capacitive Circuit.

GNR quantum capacitance, and C_{it} the capacitance caused by interface traps. Note that, it was experimentally observed that GNR devices inherently exhibit near-interface traps [81], which will trap/release charges via capacitance C_{it} in an analogue manner with the membrane potential accumulation. When applying a top gate voltage, V_{g}, charge transfer to/from graphene to the interface traps causes an equivalent shift of V_{g}, with a quantity denoted as ΔV_{traps} [128]. Considering a piece-wise linear V_{g}, when the GNR surface potential V_{c} changes from V_{c}^{t1} at time moment $t1$ to V_{c}^{t2} at time moment $t2$, the interface traps charges can be obtained as:

$$Q_{\text{it}}(t) = C_{\text{it}} \cdot [(V_{\text{c}}^{t1} + \alpha \cdot t - \alpha \cdot \tau) + e^{-\frac{t}{\tau}} \cdot (\alpha \cdot \tau - V_{\text{c}}^{t1} + V_{\text{it}}^{t1})], \tag{4.2}$$

where V_{it}^{t1} is the accumulated voltage drop on C_{it} at time moment $t1$, τ is the trapping/detrapping time constant, and α is the V_{c} ramp slope from $t1$ to $t2$. Thus with a single graphene device, the membrane integration features are naturally captured by the interface charge trapping/detrapping phenomena.

4.3. Graphene-based Neuron

In this section we introduce the proposed graphene-based nonlinear LIF neuron circuit and describe its operation.

As illustrated in Figure 4.3 (a), the graphene-based neuron comprises six GNR-based devices, which can be divided into 2 blocks: the integrate-and-fire block, which mimics the membrane potential dynamics and the output block, which generates the output spikes. To aid the explanation, we make use of the basic operation example depicted in Figure 4.4. The neuron kernel is $\text{GNR}_{\text{up}}^{2}$, which captures the membrane potential dynamics via its conductance. Due to the GNR inherent interfacial traps, electrical charges proportional to the GNR applied voltages can be accumulated or released. Starting from the membrane resting level, such behavior can be observed until reaching the membrane firing threshold, at which point, there is a maximum accumulation of charges (which corresponds to

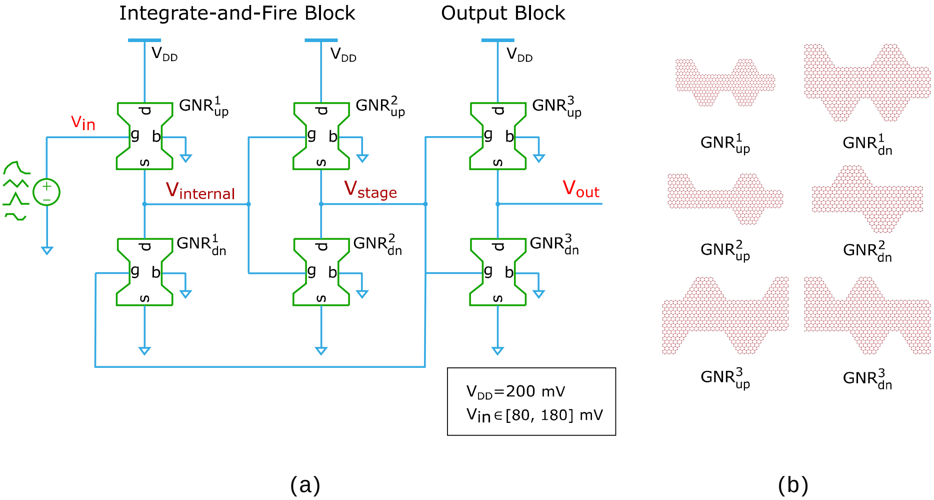
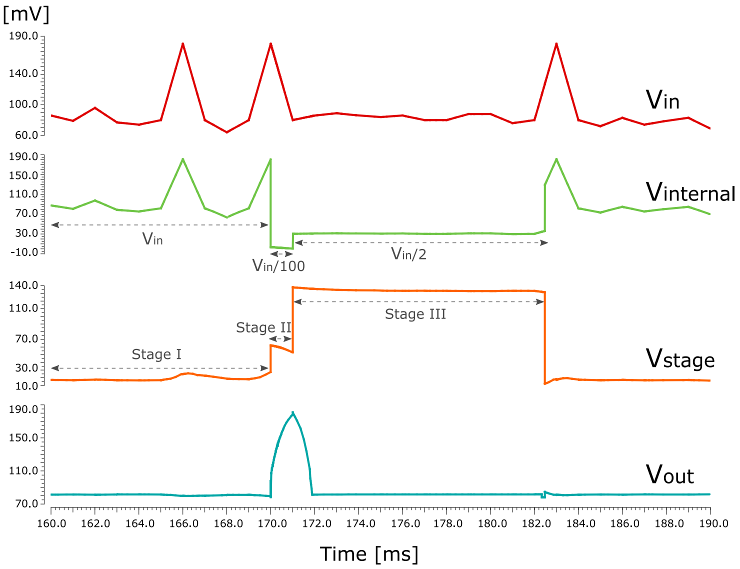


Figure 4.3: GNR-based LIF Neuron: (a) Neuron Structure, and (b) GNR Topologies.

a maximum conductance value). We denote this integrate-and-fire region as Stage I. Further, to emulate the membrane potential reset, most of the trapped charges need to be released, situation which happens only when the GNR$^2_{\text{up}}$ top gate voltage is very small (e.g., ≈ 100× smaller V_{in}) - Stage II. Then, during the refractory period, a gradual accumulation of charges should follow in order to reach the membrane resting level, situation which is achieved when applying a slightly bigger top gate voltage (but smaller than the membrane resting level), e.g., ≈ 2× smaller V_{in} - Stage III. The sub-circuit composed out of GNR$^1_{\text{up}}$ and GNR$^1_{\text{dn}}$, receives the neuronal input spike train V_{in} and controls the top gate voltage of GNR$^2_{\text{up}}$ via V_{internal} (it either directly outputs the neuron input V_{in} during Stage I or a magnitude down-scaled neuron input, i.e., ≈ $V_{\text{in}}/100$ during stage II and ≈ $V_{\text{in}}/2$ during Stage III). The output block containing GNR$^3_{\text{up}}$ and GNR$^3_{\text{dn}}$ devices generates the neuron output spike V_{out}.

As illustrated in Figure 4.4, initially, V_{internal} follows V_{in} and V_{stage} values are afferent to Stage I. When $V_{\text{internal}} + \Delta V_{\text{traps}}$ reaches the firing threshold, V_{stage} switches to Stage II and V_{internal} becomes equal to $V_{\text{in}}/100$. Charges are depleted, the membrane potential resets, and an output spike V_{out} is triggered. When $V_{\text{internal}} + \Delta V_{\text{traps}}$ reaches the voltage value which corresponds to the end point of the neuron input spike V_{in}, V_{stage} transitions to Stage III, and V_{internal} is generated equal to $V_{\text{in}}/2$. When $V_{\text{internal}} + \Delta V_{\text{traps}}$ reaches a fixed out of refractory threshold voltage level, V_{stage} switches back to Stage I, and the neuron activity resumes.

To obtain the desired GNR topologies, we performed a design space exploration, by changing the GNR geometry, and the width and position of the top-gate, such that for every up/down pair of GNRs the in-between voltage follows the aforementioned behavior. The in-between voltage can be calculated by using a voltage divider $V_{\text{DD}} \cdot G_{\text{up}}/(G_{\text{dn}} + G_{\text{up}})$, where G_{up} and G_{dn} represent the conductance of GNR$_{\text{up}}$ and GNR$_{\text{dn}}$, respectively, and $V_{\text{DD}} = 0.2\,\text{V}$ denotes the supply voltage. Figure 4.3 (b) depicts the obtained GNR topolo-



(c)

Figure 4.4: GNR-based LIF Neuron Basic Operation.

gies, with $W \times L$ dimensions $23a \times 30\sqrt{3}a$ and $35a \times 30\sqrt{3}a$ for GNR^1_{up} and GNR^1_{dn}, respectively, $23a \times 30\sqrt{3}a$ and $29a \times 25\sqrt{3}a$ for GNR^2_{up} and GNR^2_{dn}, and $35a \times 30\sqrt{3}a$ for both GNR^3_{up} and GNR^3_{dn}, where a =0.142 nm is the distance between 2 adjacent carbon atoms.

4.4. SIMULATION RESULTS

In order to model the graphene electronic transport properties we make use of the atomistic-level tight binding Hamiltonian matrix to describe the interactions between carbon atoms and external graphene potentials, the Non-Equilibrium Green Function (NEGF) formalism to solve the Schrödinger equation, and the Landauer-Büttiker formula to derive the GNR current and conductance [102]. As interface traps profile, we employed a trapping/detrapping time constant of 1.6 ms and an interface trap density of $2.363 \cdot 10^{13}$ cm^{-2}(eV)$^{-1}$ [113] , [140].

The neuron circuit was functionally validated and evaluated by means of SPICE simulation in Synopsys HSPICE [141]. In order to preserve the GNRs physical simulation accuracy degree, we developed a Verilog-A SPICE compatible generic model, which relies on look-up tables containing GNRs conductance values for varying input profiles, which are obtained with aforementioned atomistic-level formalization. For instance, to calculate the GNR^2_{up} conductance for a certain top gate voltage V_g and drain-to-source potential V_{ds} at the current time moment t_i in the presence of traps, we rely on the previously applied V_g at time moment t_{i-1}, on the time difference between the sampling points $t_i - t_{i-1}$, as well as on the traps-induced accumulation ΔV_{traps} at moment t_{i-1}. All these values are then logged in the GNR^2_{up} corresponding table for a wide range of scenarios.

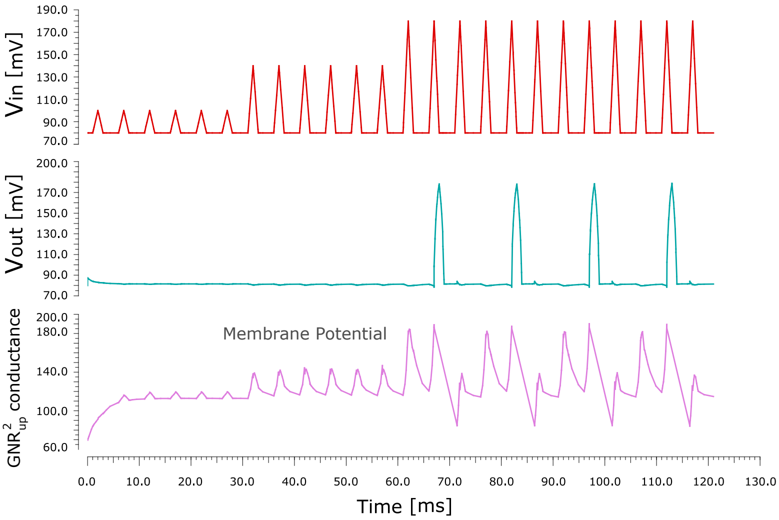


Figure 4.5: Integrate and Fire Dynamics.

To validate the integrate-and-fire behavior of the proposed graphene-based neuron, we applied as indicated in Figure 4.5, a deterministic periodic neuronal input V_{in} with 2 ms spike duration and 5 ms inter-spike intervals and gradually increased the V_{in} peak amplitude from 100 mV to 180 mV. We observe that individual V_{in} spikes contributions are gradually accumulated and proportionally reflected in the GNR^2_{up} conductance G increase. Also, we see that for smaller V_{in} spikes (<180 mV), the conductance increase saturates at a level below the firing threshold, while for 180 mV V_{in} spikes it can reach the firing threshold, and as a result an output spike event is triggered and reflected in the V_{out} value.

As biochemical processes of individual neurons, as well as surrounding neuronal network activities exhibit stochasticity, the neuronal spike trains exhibit inherent variability. To evaluate the proposed neuron behavior in such conditions, we considered a stochastic input spike train (sampled from a Poisson distribution) with a firing frequency of 50 spikes per second (comparable scenario with that of biological neurons), and added a white Gaussian noise floor with signal-to-noise ratio SNR = 17. Figure 4.6 illustrates the neuron corresponding firing response. We note that every firing event is triggered by an input spike and not by the noise, even though the noise does contribute to the membrane potential accumulation. This suggests that the proposed neuron is robust to input noise. To gain better insight and quantify the variability of the output spike train produced by the proposed neuron, we consider a range of input firing rates from 20 to 200 spikes per second and calculate the output mean firing rate and the variation coefficient CV_{ISI}, which is equal to the standard deviation of the inter-spike timing intervals divided by their mean. Simulation results, depicted in Figure 4.7, indicate a steady linear increase of the mean output firing rate, suggesting a regular firing behavior for the proposed graphene-based neuron. The output spike train propensity for regularity is also confirmed by a slightly decreasing and < 1 inter-spike interval coefficient of variation.

In retrospective, the proposed graphene-based neuron exhibits a small footprint (max.

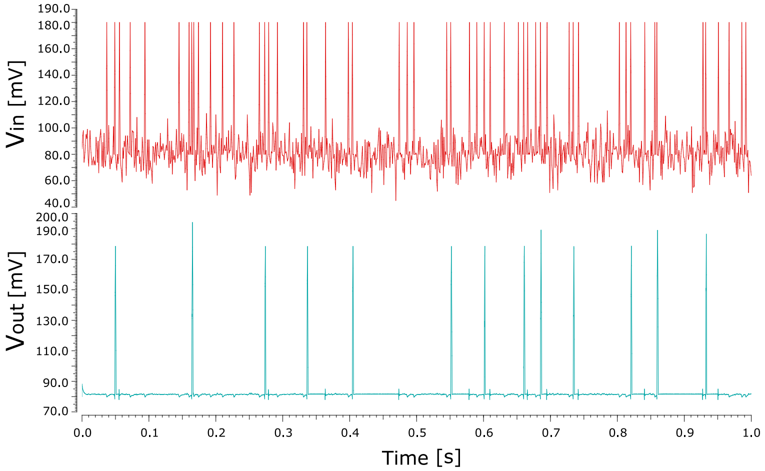


Figure 4.6: Graphene-based Neuron Dynamics Under Random Input.

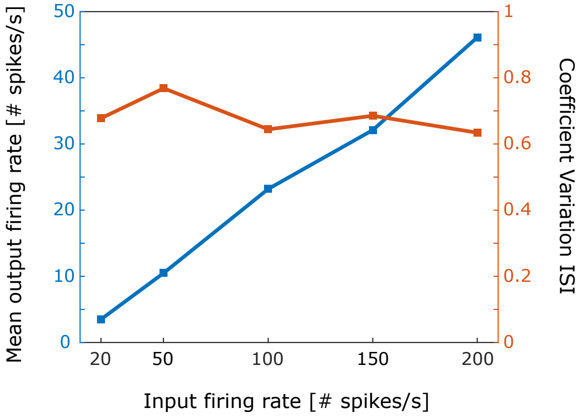


Figure 4.7: Output Spike Statistics For Variable Input Firing Rate.

36 nm^2 per GNR device), and low voltage operation (e.g., 200 mV), which are desired characteristics for artificial neural networks large-scale implementations. Our simulations indicate regularity of firing events under noisy stochastic input spike trains. Furthermore, the considered 2 ms spike duration and 100 mV spike amplitude are comparable with that observed in biological neurons, suggesting the potential to fabricate biologically plausible artificial neurons potentially interface-able with biological tissues.

4.5. CONCLUSIONS

In this chapter, we proposed a compact, all graphene-based nonlinear leaky integrate-and-fire neuron. By means of SPICE simulation, we demonstrated that the proposed neuron can properly emulate the basic spiking neuron dynamics under periodic input spikes. We further investigated the output spikes' behavior under stochastic noisy input spike trains. Our simulation results indicated variability resilience and neuronal output firing regularity for a varying input firing rate (from 20 to 200 spikes per second). The small area, low energy (inherent to the 200 mV supply voltage) are certainly enabling factors for the potential implementation of large-scale artificial neural systems.

5

Graphene-based Spiking Neural Network

In this chapter, we introduce a basic SNN unit, which comprises a graphene-based synapse and a spiking neuron with input-output compatibility, and can be utilized to implement complex SNNs. We first demonstrate the proper operation of the graphene SNN unit by relying on the mixed simulation approach that embeds the high accuracy of atomistic level simulation of graphene structures conductance within the SPICE framework. Subsequently, we analyze the way graphene synaptic plasticity affects the behavior of a 2*-layer SNN example consisting of* 6 *neurons and demonstrate that LTP significantly increases the number of firing events while LTD is diminishing them, as expected. To assess the plausibility of the graphene SNN reaction to input stimuli we simulate its behavior by means of both SPICE and NEST, a well established SNN simulation framework, and demonstrate that the obtained reactions, characterized in terms of total number of firing events and mean Inter-Spike Interval length, are in close agreement, which clearly suggests that the proposed design exhibits a proper behavior. Further, we prove the unsupervised learning capabilities of the proposed design by considering a* 2*-layer SNN consisting of* 30 *neurons meant to recognize the characters "A", "E", "I", "O", and "U", represented with a* 5 *by* 5 *black and white pixel matrix. The SPICE simulation results indicate that the graphene SNN is able to perform unsupervised character recognition associated learning and that its recognition ability is robust to input character variations. Finally, we note that the proposed SNN unit requires a small real-estate footprint and operates at* 200 mV *supply voltage, which suggest its suitability for the design of large-scale energy-efficient computing systems.*

The content of this chapter is based on the following paper:

H. Wang, N. Cucu Laurenciu, Y. Jiang, and S.D. Cotofana, "Compact Graphene-Based Spiking Neural Network With Unsupervised Learning Capabilities", *IEEE Open Journal of Nanotechnology (OJ-NANO)*, vol. 1, pp. 135-144, 2020.

5.1. Introduction

In previous chapters we proposed graphene-based implementations concentrated on individual synapse and neuron designs while disregarding input-output compatibility aspects, which preclude their direct utilization for the implementation of graphene-based Spiking Neural Networks. In this chapter, we propose a graphene-based synapse (comprising 2 graphene devices) and a spiking neuron (comprising 6 graphene devices), which form together a basic Spiking Neural Network (SNN) unit and can be utilized for the implementation of complex graphene-based SNNs. Specifically, the proposed artificial synapse emulates two basic synaptic functionalities, i.e., Spike-Timing-Dependent Plasticity (STDP) and Long-Term Plasticity, while the same synapse can exhibit Long-term Potentiation (LTP) or Long-term Depression (LTD) by properly adjusting the back-gate bias voltage of one of its composing graphene device. The proposed artificial neuron exhibits the essential Leaky Integrate and Fire (LIF) spiking neuron behavior with post firing refractory interval and provides the feedback signal required for the SDTP associated synaptic transmission efficiency modulation.

We first demonstrate the proper operation of the graphene SNN unit by relying on a mixed simulation approach that embeds the high accuracy of atomistic level simulation of graphene structures conductance within the SPICE framework. Subsequently, we analyze the way the synaptic plasticity affects the graphene SNN behavior by making use of a 2-layer SNN example consisting of 6 neurons and the obtained results indicate that LTP significantly increases the number of SNN firing events while LTD is diminishing them, as expected. To get some inside on the 2-layer graphene SNN reaction to input stimuli plausibility we also simulate its behavior by means of NEST [124], a well established SNN simulation framework. Our experiments indicate that the SPICE obtained reaction, characterized in terms of total number of firing events and mean Inter-Spike Interval (ISI) length, is in close agreement with the one reported by means of NEST based simulation, which clearly suggests that the proposed design exhibit a proper behavior. Further, we demonstrate the unsupervised learning capabilities of the proposed design by considering a two layer SNN consisting of 30 neurons meant to recognize the characters (and variations of them) "A”, "E”, "I”, "O”, and "U", represented with a 5 by 5 black and white pixel matrix. The simulation results indicate that the graphene SNN is able to perform unsupervised learning and that the enabled recognition ability is robust to input character variations. Finally, we note that our proposal results in a small real-estate footprint (max. 30 nm^2 are required by one graphene-based device) and operates at 200 mV supply voltage, which suggest its suitability for the design of large-scale energy-efficient computing systems.

The remaining of this chapter is organized as follows: Section 5.2 presents the utilized simulation framework. Section 5.3 introduces the graphene-based SNN design and its basic operation. Section 5.4 presents the simulation results and Section 5.5 concludes the chapter.

5.2. Simulation Framework

In order to properly validate and evaluate the graphene-based SNN circuits, we rely on a mixed simulation approach incorporating atomistic level graphene-based device modelling and SPICE simulation in Cadence [142].

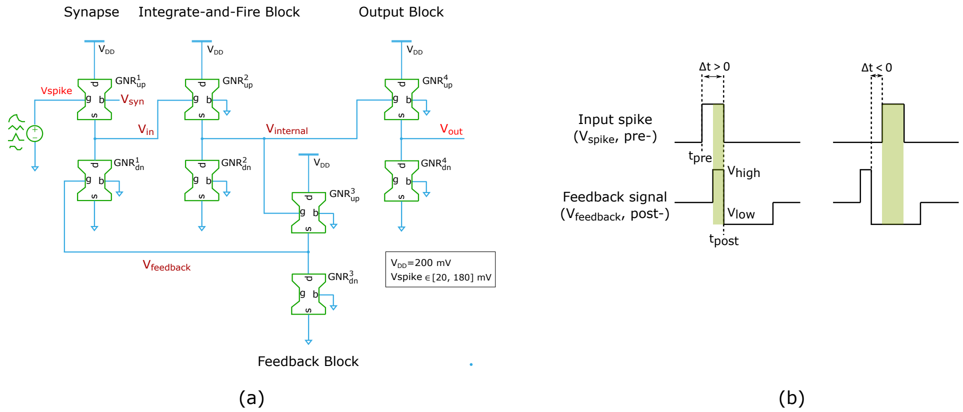


Figure 5.1: Graphene-based Spiking Neural Network: (a) SNN Circuit, (b) Pre- vs Post-Synaptic Spikes Timing.

To enable the desired synapse and neuron functionalities, we rely on instances of the generic graphene-based device depicted in Figure 4.2 (a). For the graphene-based device electronic transport properties calculation, we utilize the atomistic level Tight-Binding Hamiltonian to model the carbon atom interactions and external potentials, the Non-Equilibrium Green Function (NEGF) to solve the Schrödinger equation, and the Landauer-Büttiker formula to calculate the GNR channel current and conductance [102]. The potential distribution on graphene sheet is obtained by solving a 3D Poisson equation self-consistently, and the effect of trapping/detrapping phenomenon on the device operation is accounted for by calculating the equivalent voltage shift caused by interface trapped charges [128].

To enable high accuracy circuit simulation, we make use of a Verilog-A graphene device generic model [103], which in order to enable time effective SPICE simulation of graphene circuit relies on GNR topology specific precomputed look-up tables containing graphene conduction simulation data obtained with the aforementioned atomistic level simulation methodology.

5.3. Graphene-based Spiking Neural Networks

In this section we present the proposed graphene-based Spiking Neural Network design and describe its basic operation principle.

The schematic illustration of the graphene-based SNN unit (consisting of one synapse and one neuron) is depicted in Figure 5.1 (a) and comprises four blocks: (i) synapse, (ii) integrate-and-fire, (iii) feedback, and (iv) output. Each block consists of two GNR-based devices and its output voltage (V_{in}, V_{internal}, V_{feedback}, and V_{out}) is governed by the $V_{\text{DD}} \cdot G^i_{\text{up}}/(G^i_{\text{up}} + G^i_{\text{dn}})$ relation, where V_{DD} is the supply voltage (200 mV), and G^i_{up} and G^i_{dn} denote the conductance of the i^{th} GNR_{up} and GNR_{dn}, respectively.

The synapse receives input spikes V_{spike} from another neuron, potentiates or suppresses them according to its transmission efficiency (weight), and generates V_{in} to be utilized

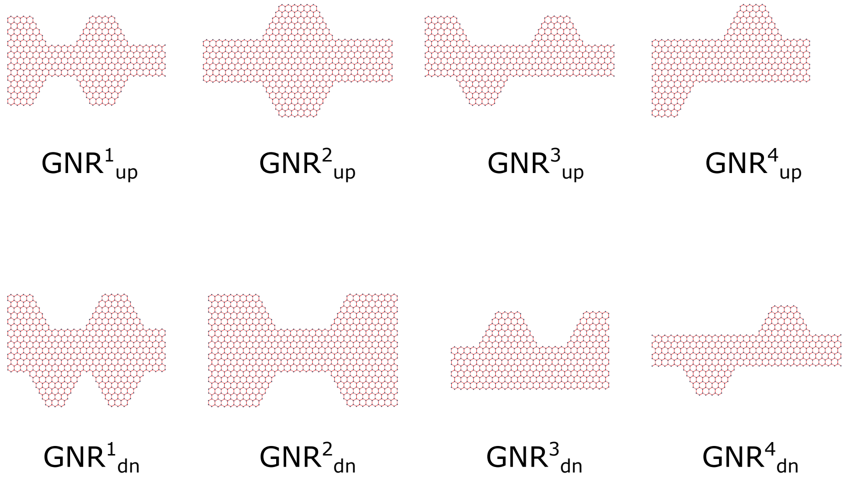


Figure 5.2: Basic SNN Unit GNR Shapes.

as neuron block input. The initial synaptic weight value is determined by the V_{syn} potential applied on the back-gate of GNR^1_{up}. The synapse exhibits two types of plasticity: Spike-Timing-Dependent Plasticity (STDP) and Long-Term Plasticity. STDP modulates the synaptic weight by accounting for the time difference $\Delta t = t_{post} - t_{pre}$ between the input spike V_{spike} occurrence and V_{feedback} transition from V_{high} to V_{low}, as depicted in Figure 5.1 (b). When $\Delta t > 0$, i.e., V_{feedback} is asserted before the end of the input spike, the synaptic transmission efficiency is increased and V_{spike} contribution to V_{in} is strengthened. When $\Delta t < 0$, i.e., the input spike occurrence is not generating a V_{feedback} transition to V_{high}, the synaptic transmission efficiency is decreased and V_{spike} contribution to V_{in} is weakened. The input spike potentiation/depression is controlled by the V_{feedback} signal, which by being connected to GNR^1_{dn} top-gate modulates its conductance. The long-term plasticity emulation relies on the fact that when applying input spikes on GNR^1_{up} top-gate trapped charges are accumulated and as such modulate its conductance persistently, which depending on the V_{syn} value results in Long-Term Potentiation (LTP) or Long-Term Depression (LTD) of the synaptic weight, e.g., 0 mV for LTP and −100 mV for LTD.

The integrate-and-fire block is the kernel of the graphene spiking neuron and emulates the main neuronal functionalities, including membrane potential integration and the generation of the output firing events. The integrate and fire behavior builds upon the interface trapping phenomenon, which results in charge accumulation when V_{in} spikes are applied on GNR^2_{up} top-gate. The trapped charges cause an equivalent shift ΔV_g of the top-gate voltage V_g and when $V_g + \Delta V_g$ reaches a certain level, i.e., the neuron firing threshold, GNR^2_{up} conductance increases abruptly, which triggers a firing event, i.e., generates a spike on the $\text{V}_{\text{internal}}$ signal.

While this is enough to emulate spiking neuron functionality $\text{V}_{\text{internal}}$ requires some extra processing in order to be compatible in terms of voltage levels and duration with the input spike applied on V_{spike}, which assumes values between 20 mV and 180 mV and has

Table 5.1: GNR Topologies

	GNR^1_{up}	GNR^1_{dn}	GNR^2_{up}	GNR^2_{dn}	GNR^3_{up}	GNR^3_{dn}	GNR^4_{up}	GNR^4_{dn}
W [a]	23	29	29	29	23	29	29	29
L [a]	$25\sqrt{3}$	$25\sqrt{3}$	$30\sqrt{3}$	$30\sqrt{3}$	$30\sqrt{3}$	$25\sqrt{3}$	$25\sqrt{3}$	$30\sqrt{3}$
PV_g [a]	$6\sqrt{3}$	$6\sqrt{3}$	$6\sqrt{3}$	$6\sqrt{3}$	$6\sqrt{3}$	$4\sqrt{3}$	$6\sqrt{3}$	$6\sqrt{3}$
WV_g [a]	$8\sqrt{3}$	$6\sqrt{3}$	$8\sqrt{3}$	$6\sqrt{3}$	$6\sqrt{3}$	$8\sqrt{3}$	$6\sqrt{3}$	$4\sqrt{3}$

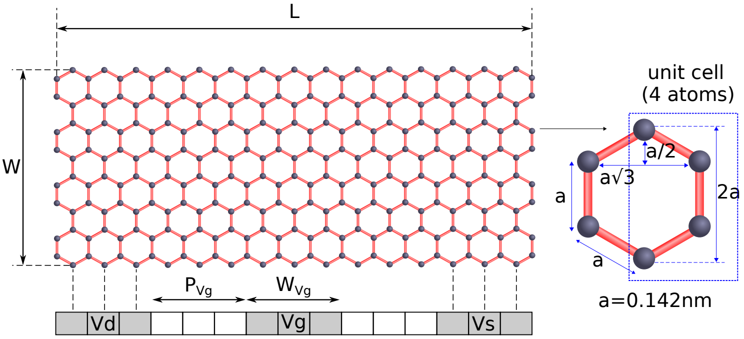


Figure 5.3: GNR Geometry and Contact Topology.

a time duration of 2 ms. As such the output block further processes $V_{internal}$ and produces V_{out} that is level and duration compatible with synapse input spikes, which enables the direct cascading of SNN basic units. Moreover, as the neuron output is playing a crucial role in the STDP process the $V_{internal}$ spike occurrence has to be signalled to the synapse block. Again $V_{internal}$ cannot be directly utilized and the feedback block is responsible for the generation of $V_{feedback}$ that is connected to GNR^1_{dn} top-gate to internally signal the firing event occurrence. Apart of contributing to the synaptic weight adaptation $V_{feedback}$ is also placing the neuron into the refractory state, which has to occur after any output firing event. This is enabled by the $V_{feedback}$ transition from V_{high} to V_{low}, which is increasing GNR^1_{dn} conductance resulting in a significant V_{in} magnitude reduction that inhibits the trap accumulation and as such incoming input spikes cannot trigger a firing event while $V_{feedback} = V_{low}$.

The basic SNN unit behavior is actually dependent on the conductance variation exhibited by each of the GNRs it comprises. Thus to guaranty proper SNN functionality 4 GNR geometry pairs, which conductance maps fit the variation profile required to achieve the desired behavior of the V_{in}, $V_{internal}$, $V_{feedback}$, and V_{out} signals, respectively, should be find.

Figure 5.3 illustrates the parameters related to GNR geometry and contacts topology. Specifically, W and L denote the width and length of the graphene sheet, respectively, P_{V_g} the distance between the top-gate and the drain contact, and W_{V_g} the top-gate width. The distance between two neighbor carbon atoms is denoted as $a = 0.142\,\text{nm}$. Figure 5.2 depicts the GNR topologies we identified for the proposed SNN circuit, by means of an

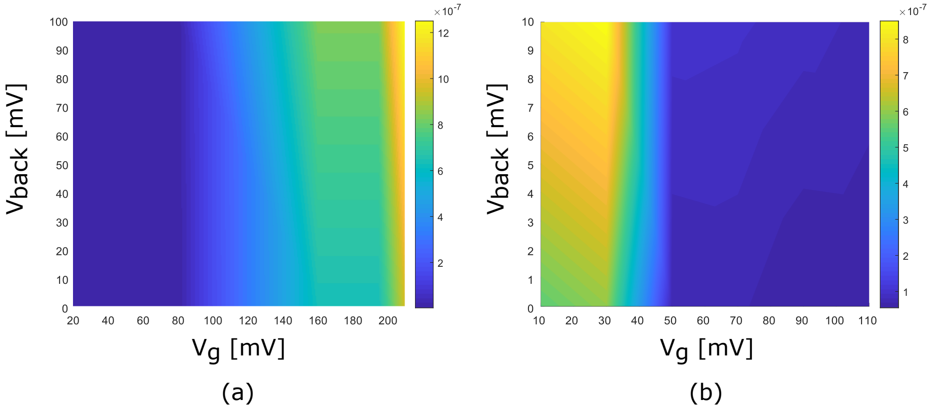


Figure 5.4: GNR Conduction Maps: (a) $\mathrm{GNR}^1_{\mathrm{up}}$, (b) $\mathrm{GNR}^1_{\mathrm{dn}}$.

atomistic model based Design Space Exploration (DSE) process and Table 5.1 summarizes their dimensions expressed in terms of the distance between adjacent carbon atoms in the graphene sheet $a = 0.142\,\mathrm{nm}$. Concerning the interface trap profile for the atomistic-level graphene-based device modelling in DSE, we assume an interface trap density of $2.363 \cdot 10^{13}\ \mathrm{cm}^{-2}(\mathrm{eV})^{-1}$ and a trapping/detrapping time constant of 1.6 ms [108, 113].

To provide inside on the relation between the chosen GNR topologies and SNN circuit behavior we present in Figure 5.4 the conduction maps of $\mathrm{GNR}^1_{\mathrm{up}}$ and $\mathrm{GNR}^1_{\mathrm{dn}}$ that form the synapse block. As one can observe in Figure 5.4 $\mathrm{GNR}^1_{\mathrm{up}}$ conduction is high under large top-gate voltages and varies with back-gate voltage value thus can provide different initial synaptic weights. $\mathrm{GNR}^1_{\mathrm{dn}}$ conductance is high under low top-gate voltages and small under high top-gate voltages, which allows V_{feedback} to induce synaptic transmission potentiation and depression when being V_{high} and V_{low}, respectively. A similar analysis can be carried on for the other GNR pairs in the circuit but we omit it in view of page limit.

SPICE simulation results concerning the SNN unit basic operation (with $V_{\mathrm{syn}} = 0\,\mathrm{mV}$) are illustrated in Figure 5.5. As seen from the point of view of V_{feedback} value the basic operation follows three phases. In Phase I V_{feedback} has an initial after circuit reset value and the neuron input V_{in} follows the synapse input V_{spike}. When $V_g + \Delta V_g$ of $\mathrm{GNR}^2_{\mathrm{up}}$ reaches the firing threshold a spike is induced on V_{internal} that makes V_{feedback} to enter into Phase II when V_{in} magnitude increases to $V_{\mathrm{spike}} \times 1.1$ for a short time period as result of the pre-spike before post-spike $\Delta t > 0$ induced STDP potentiation. Immediately after the firing event V_{internal} returns to its initial value and as the V_{internal} induced trapped charges are still present V_{feedback} becomes V_{low} and the SNN unit enters Phase III. In this period V_{in} magnitude decreases to $V_{spike}/2.2$ as a result of pre-spike after post-spike $\Delta t < 0$ induced STDP depression. As no firing events can be triggered during Phase III, it accounts for the spiking neuron refractory interval. When the feedback block trapped charges decay to the initial level, V_{feedback} returns to its after reset value, Phase III finishes and the circuit switches back to Phase I. Related to the refractory interval influence on the neuron behavior one can observe in Figure 5.5 that the first output spike is triggered by 2 input spikes

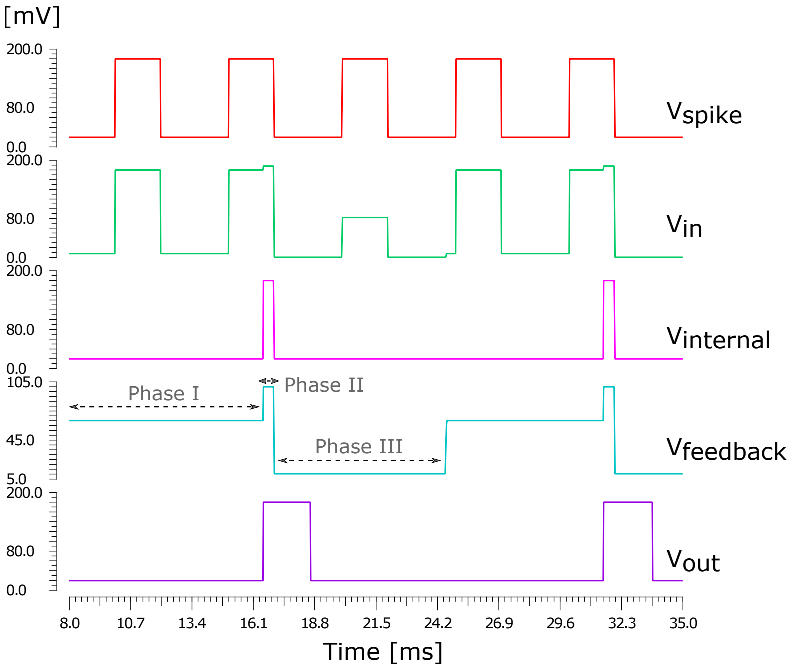


Figure 5.5: Graphene-based SNN Unit Basic Operation.

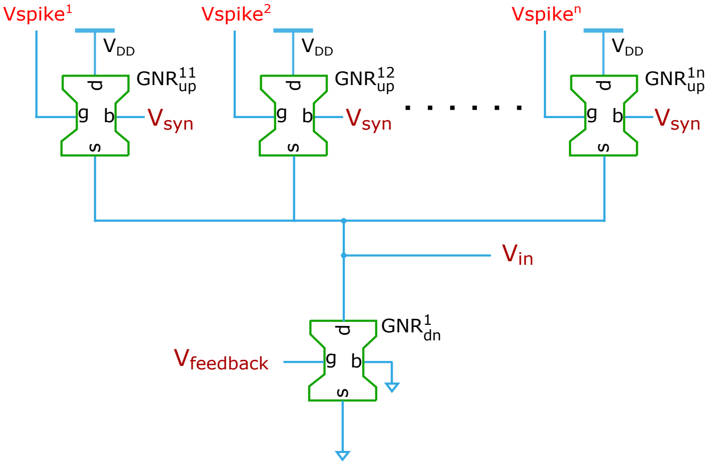


Figure 5.6: Multi-input Synapse Block.

while the second one occurs after 3 input spikes.

The basic SNN unit in Figure 5.1 (a) assumes that the neuron process input spikes com-

Table 5.2: SNN Activity Overview

	Input firing rate: 200 Hz		Input firing rate: 250 Hz	
	Total Number	Mean ISI	Total Number	Mean ISI
Graphene-based SNN	16	30	22	22.6
NEST simulator	17	28.5	22	22.3

ing from one previous neuron only, i.e., has a fan-in of 1, which is certainly not the case in any relevant SNN. To accommodate for a fan-in of n we extend the synapse block by replacing $\mathrm{GNR}^1_{\mathrm{up}}$ with n GNRs as illustrated in Figure 5.6. In this case the in-between voltage V_{in} is calculated as:

$$V_{in} = V_{DD} \cdot \frac{G^{11}_{up} + G^{12}_{up} + \cdots + G^{1n}_{up}}{G^{11}_{up} + G^{12}_{up} + \cdots + G^{1n}_{up} + G^{1}_{dn}}, \tag{5.1}$$

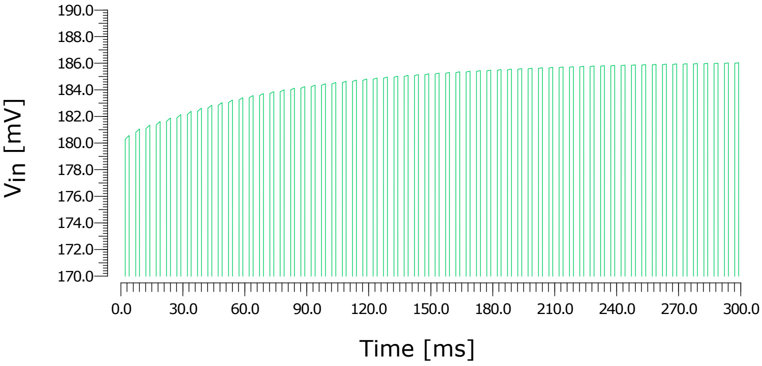
where $\mathrm{G}^{1n}_{\mathrm{up}}$ denotes the conductance of the n^{th} up GNR.

5.4. SIMULATION RESULTS

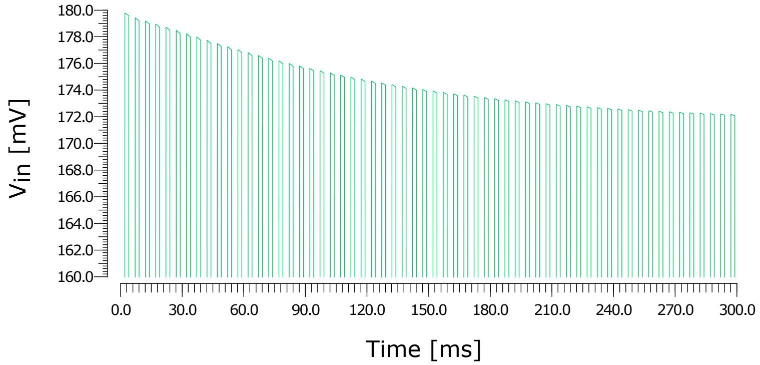
To get inside into the actual capabilities of the proposed SNN unit we consider and evaluate by means of SPICE simulation two graphene-based SNN examples. We first study the effect of the graphene enabled synaptic plasticity on a 2-layer 6-neuron SNN and compare its SPICE derived behavior with the one obtained by means of NEST based simulations [124]. Subsequently, we demonstrate the capability of our proposal to perform unsupervised character recognition. In all simulations, the input spikes are 2 ms long pulses varying between 20 mV and 180 mV, and $V_{\mathrm{DD}} = 200\,\mathrm{mV}$. We note however that our proposal is general and can be adapted to operate on different power supply values and input spike formats.

5.4.1. GRAPHENE-BASED SNN BEHAVIOR EVALUATION

To evaluate how the long-term plasticity exhibited by the graphene devices modulates the neuron input signal V_{in}, we consider a single synapse block comprising $\mathrm{GNR}^1_{\mathrm{up}}$ and $\mathrm{GNR}^1_{\mathrm{dn}}$, as depicted in Figure 5.1 (a), and set the feedback signal V_{feedback} to the Phase I value. In such a setup the synapse output magnitude follows the synapse input and if V_{spike} receives a train of spikes Long-Term Plasticity should be observed. To capture this phenomenon we apply a 200 Hz periodic input spike train with 180 mV peak amplitude on the $\mathrm{GNR}^1_{\mathrm{up}}$ top-gate and simulate the circuit evolution for 300 ms. The obtained dynamics of the synapse output signal V_{in} is depicted in Figure 5.7 (a) and (b) for Long-Term Potentiation (LTP) and Long-Term Depression (LTD), respectively. Note that both LTP and LTD are acquired with the same synapse by properly changing the back-gate voltage of $\mathrm{GNR}^1_{\mathrm{up}}$, i.e., 0 mV for LTP and −100 mV for LTD. As expected, we observe a continuous V_{in} magnitude increase and decrease for LTP and LTD, respectively. After 300 ms the amplitude potentiation and depression are around 3.3% and 4.5%, for LTP and LTD. respectively, and exhibit an obvious saturation trend.



(a) Long-term potentiation.



(b) Long-term depression.

Figure 5.7: Synapse Long-Term Plasticity.

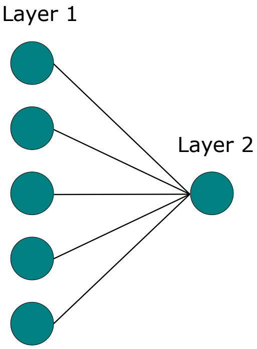


Figure 5.8: Two Layers 6-neuron SNN.

To explore the implication of the obtained long-term plasticity on SNN's firing events profile, we make use of a 2-layer SNN consisting of 6 neurons as illustrated in Figure 5.8. The neuron in layer 2 is fully connected with all the neurons in layer 1 via identical synapses. In the simulations we considered three synapse types: (i) without long-term plasticity (assuming that the trapped charges do not affect the graphene device conductance), (ii) with long-term potentiation, and (iii) with long-term depression.

To evaluate the SNN behavior in the previously mentioned conditions we perform SPICE simulations assuming that all layer 1 neurons receive identical 200 Hz periodic input signals (V_{spike}) on their synapse block for 200 ms, thus all layer 1 neurons generate identical firing events. Figure 5.9 (a) depicts the SNN output reaction with the synapses do not exhibit long-term plasticity. We observe that periodic output spike trains are generated by all neurons while layer 2 neuron firing rate of the neuron is lower than that of the layer 1 neurons. During the simulation there are in total 12 output spikes for every neuron in layer 1 and 5 output spikes for neuron in layer 2. Figure 5.9 (b) depicts the SNN output firing events with Long-Term Potentiation. As expected LTP induces an increase of the number of firing events in both layers, which now raise to 15 and 13 for neurons in layer 1 and layer 2, respectively. Thus LTP induces a 25% firing event increase in layer 1 and 160% in layer 2. On the contrary, in the case of SNN with Long-Term Depression, the simulation result is depicted in Figure 5.9 (c), we observe a significant decrease tendency of the number of firing events in both layers. Specifically, the layer 2 neuron stops generating any fire event after 60 ms, which is related to the fact that due to LTD layer 1 neurons are less active and as such cannot trigger a firing event of the neuron in layer 2. The total number of firing events for neurons in layer 1 and layer 2 are 9 and 1, which is equivalent with a 25% and 80% decrease, respectively.

To get some inside of the plausibility of the LTP and LTD influence on the considered SNN example we implement it in NEST with standard leaky Integrate-and-Fire neurons connected via synapses with Long-Term Potentiation, apply 200 Hz and 250 Hz periodic input spike trains, and record its reaction a time period of 200 ms. The number of layer 2 neuron firing events as well as the mean Inter-Spike Interval (ISI) between output spikes obtained by the SPICE simulation of the graphene-based SNN with LTD and the ones reported by means of NEST simulation are summarized in Table 5.2. In terms of the total number of firing events, the graphene SNN produces an almost identical response with the NEST based simulation one, i.e., 1 spike difference at 200 Hz input and the same number at 250 Hz input. As for the mean ISI, which represents the average time interval between adjacent output spikes, the values are quite close with a maximum difference of 5% (1.5 ms) between the SPICE and NEST predicted results. The obtained results clearly suggest that the proposed graphene SNN exhibits similar behavior with the one predicted by the well established NEST simulation framework.

5.4.2. Unsupervised Character Recognition

To demonstrate the learning abilities of our proposal, we consider a 2-layer SNN consisting of 30 neurons as depicted in Figure 5.10 (a), which is meant to recognize the characters (and variations of them) "A", "E", "I", "O", and "U", represented with a 5 by 5 black and white pixel matrix. Layer 1 comprises 25 neurons, which receive input spikes if the pixel in their position is black and no spikes if the pixel is white, and layer 2 consists of 5 neurons meant to indicate the recognition result. We assume that: (i) LTP synapses with identical initial synaptic weight are utilized for every neuron in layer 1 and (ii) Every neuron in layer 2 is connected with all the layer 1 neurons via LTP synapses with randomly initialized synaptic weights. This is achieved by biasing V_{syn} (the back-gate voltage of $\text{GNR}_{\text{up}}^{1}$) with fixed values between 0 mV and 100 mV, such that layer 2 neurons exhibit different firing profile. For a given input character, we stimulate the layer 1 neurons corresponding to black pixels with identical 200 Hz periodic spike trains as illustrated in Figure 5.10 (a). Each layer 2 neuron is meant to signal the recognition of one character in the vowel set and to indicate that we employ the "time-to-first-spike” scheme [53], i.e., the layer 2 neuron that first fires is the one that recognized the input character.

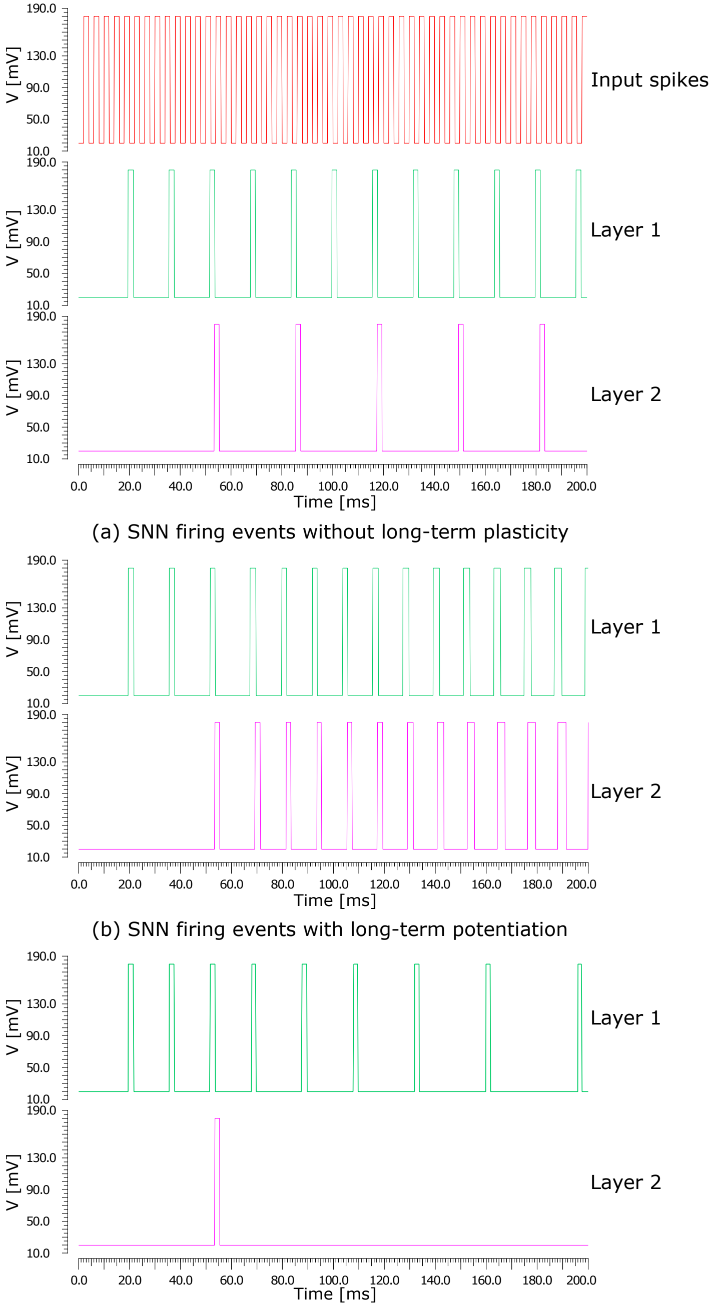
To validate the learning ability of the proposed design, we apply the 5 characters "A”, "E”, "I”, "O”, and "U", to the graphene-based SNN one at a time and the learning process for each of them is depicted in Figure 5.10 (b), (c), (d), (e), and (f), respectively.

In each case, we observe that initially there are no firing events on any layer 2 neurons. However, during the learning process, the connections corresponding to the layer 1 stimulated neurons (the one driven by black pixels) are strengthened because of long-term potentiation. Thus, after some time one neuron in layer 2 fires (indicating the recognition result) and eventually other neurons in layer 2 may fire afterwards. Figure 5.10 (b), (c), (d), (e), and (f), clearly indicate that characters "A”, "E”, "I”, "O”, and "U" are recognized by Neuron_1, Neuron_2, Neuron_4, Neuron_3, and Neuron_5, respectively. The learning time for characters "A”, "E”, "O”, and "U” is around 125 ms while for "I” is around 165 ms as it stimulates less layer 1 neurons than the other characters. As the result of this unsupervised learning process each layer 2 neuron is labeled with the character which presence in the input it recognizes and based on this labelling one can tell if a new unknown character is one of the 5.

To test the recognition ability of the graphene SNN we make use of different variations of the original characters as inputs. As an example, we present the recognition processes for six character variations that gradually degrade from "E” to "O”, as illustrated in Figure 5.11. When applying inputs that maintain the "E” character profile as depicted in Figure 5.11 (a), (b), (c), (d), and (e), one can observe that Neuron_2 first fires, which indicates that the graphene SNN correctly recognizes those inputs as character "E”. The time needed for the SNN to recognize the inputs in each case are 125 ms, 135 ms, 155 ms, 135 ms, and 145 ms, respectively, which is in line with the observation that when an input character stimulates less input neurons in layer 1, the SNN recognition takes more time. When applying an input that fundamentally deviate from "E” as depicted in Figure 5.11 (f), Neuron_3 first fires after around 125 ms, which indicates that the SNN recognizes the input character as an "O” and not as an "E". The fact that the degraded character is closer to an "O" than to an "E" is also obvious by visual inspection and as such the SNN made the correct decision. The aforementioned results demonstrate the applicability of the proposed graphene SNN for provide support for unsupervised character recognition, and that the learning ability is robust.

5.5. CONCLUSIONS

In this chapter we proposed a basic graphene-based Spiking Neural Network (SNN) unit consisting of a synapse and a spiking neuron that can be utilized to implement complex SNNs. The proposed design enables Spike-Timing-Dependent Plasticity (STDP) and Long-Term Plasticity, and both Long-Term Potentiation (LTP) and Long-Term Depression (LTD) can be induced in the same synapse by properly bias adjustments. By means of SPICE simulation, we validated the basic operation of the proposed design and analyzed how the enabled synaptic plasticity affects the SNN behavior. To this end we assumed a 2-layer SNN, derived its reaction to the same input stimuli by means of SPICE and NEST simulations, and demonstrated the close agreement between the obtained results in terms of total number of firing events and mean Inter-Spike Interval (ISI) length. Further, we demonstrated the unsupervised learning capabilities of the proposed design by considering a two layer SNN consisting of 30 neurons meant to recognize the characters (and variations of them) "A”, "E”, "I”, "O”, and "U", represented with a 5 by 5 black and white pixel matrix. The simulation results indicated that the graphene SNN is able to perform unsupervised character recognition and that its recognition ability is robust to input character variations.

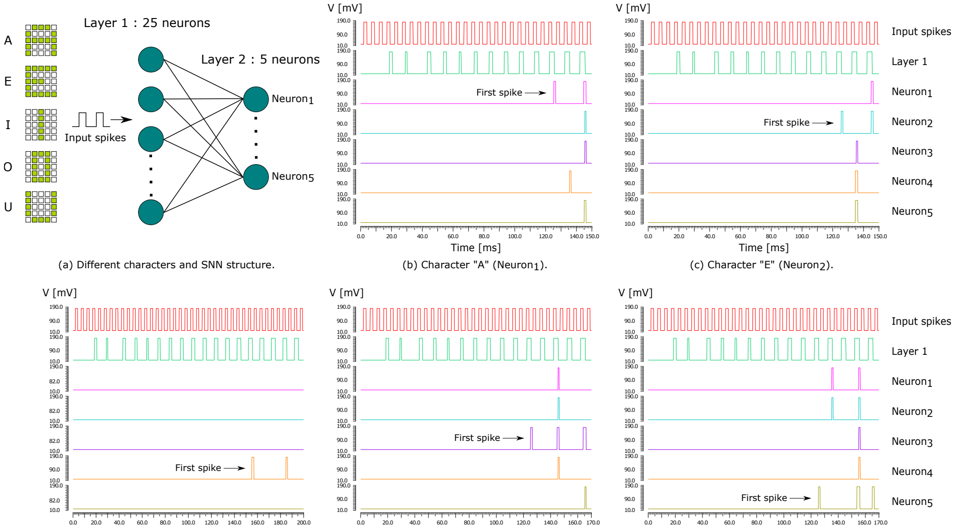


(a) SNN firing events without long-term plasticity

(b) SNN firing events with long-term potentiation

(c) SNN firing events with long-term depression

Figure 5.9: Two Layers 6-neuron SNN Output Firing Events.



(a) Different characters and SNN structure.

(b) Character "A" (Neuron1).

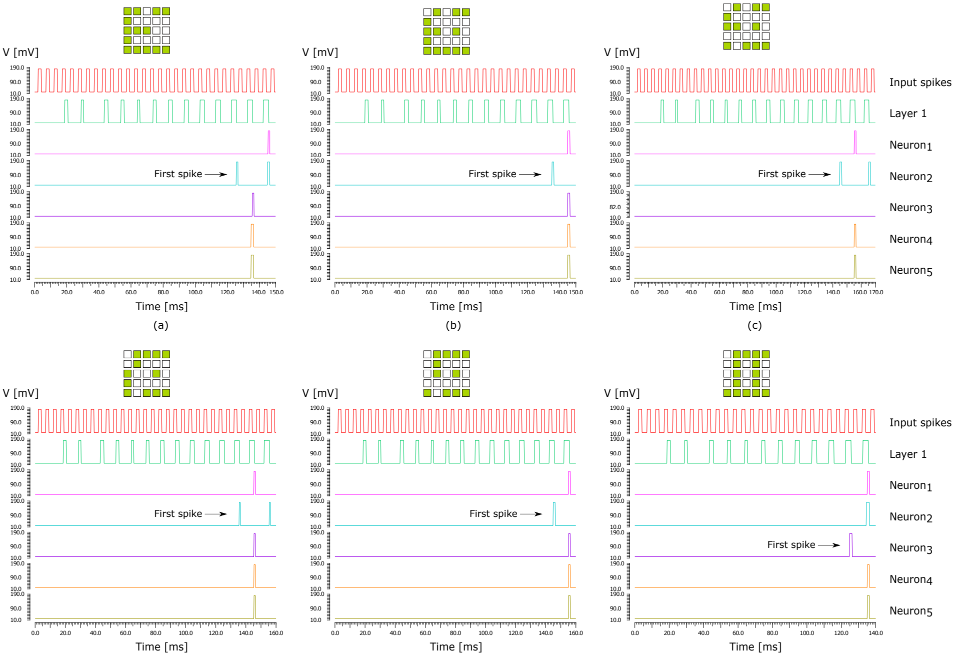
(c) Character "E" (Neuron2).

(d) Character "I" (Neuron4).

(e) Character "O" (Neuron3).

(f) Character "U" (Neuron5).

Figure 5.10: "A”, "E”, "I”, "O”, and "U" Recognition Associated Unsupervised Learning.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 5.11: Character "E" Recognition.

6

Reconfigurable Graphene-based Spiking Neural Network Architecture

To explore and enrich the potential of graphene-based neuromorphic computing, we propose a reconfigurable graphene-based Spiking Neural Network (SNN) architecture and a training methodology for initial synaptic weight values determination. The proposed graphene-based platform is flexible, comprising a programmable synaptic array which can be configured for different initial synaptic weights and plasticity functionalities and a spiking neuronal array, onto which application dependent neural network structures can be mapped. To reconfigure the proposed graphene-based platform for a practical application, an SNN topology tailored for the application and an initial SNN state (initial synaptic weights, plasticity type), which can be determined by proposed training methodology are required. To demonstrate the validity of the synaptic weights training methodology and the suitability of the proposed SNN architecture for practical utilization, we consider 2 *applications, i.e., character recognition and edge detection. In each case, the graphene-based platform is configured according to the application tailored SNN topology and initial state and SPICE simulated to evaluate its reaction to input stimuli. For the first application, a* 2*-layer SNN with* 30 *neurons is used to reconfigure the proposed graphene-based architecture and perform character recognition for* 5 *vowels, i.e., "A", "E", "I", "O", and "U" variations. Our simulation indicates that the graphene-based SNN can achieve up to* 94.5 % *recognition accuracy for the considered test datasets, which is comparable with the one delivered by a functionally equivalent Artificial Neural Network (ANN). Further, we reconfigure the architecture for a* 3*-layer* 13 *neurons SNN to perform edge detection on* 2 *grayscale images, Lena and Cameraman. SPICE simulation results indicate that the edge extraction results are close*

The content of this chapter is based on the following paper:

H. Wang, N. Cucu Laurenciu, and S.D. Cotofana, “A Reconfigurable Graphene-based Spiking Neural Network Architecture”, *IEEE Open Journal of Nanotechnology (OJ-NANO)*, vol. 2, pp. 59-71, 2021.

*agreement with the one produced by classical edge detection operators, i.e., Canny, Roberts, Sobel, and Prewitt, in terms of visual perception, Peak Signal-to-Noise Ratio (PSNR), and Mean Squared Error (MSE). Our results demonstrate that the graphene SNN platform is able to properly perform character recognition and edge detection tasks, which suggests the feasibility and flexibility of the proposed approach for various application purposes. Moreover, the utilized graphene-based synapses and neurons operate at low supply voltage (*200 mV*), consume low energy per spike for both neuron (*43 pJ *and* 5.2×10^{-7} pJ *at* 200 Hz *and* 20 GHz *spike frequency scale, respectively) and synapse (*5.1 pJ *and* 6.0×10^{-8} pJ *at* 200 Hz *and* 20 GHz *spike frequency scale, respectively), and a graphene-based synapse occupies an active area of* ≈45 nm^2 *(*2 *GNR devices) and a neuron an active area of* ≈176 nm^2 *(*6 *GNR devices), which are desired properties for large-scale energy-efficient implementations.*

6.1. INTRODUCTION

Previous work demonstrated graphene's suitability for artificial synapse [100, 143], neuron [144], and SNN unit implementations [145], thus a versatile, generic graphene-based SNN architecture that can be reconfigured for various practical tasks, would facilitate the exploration of grahene-based neuromorphic computing capabilities. In this chapter, we propose a reconfigurable graphene-based SNN architecture and an associated training methodology for initial synaptic weight values determination. Specifically, the reconfigurable SNN architecture comprises a synaptic array (consisting of graphene-based programmable synapses) and a neuronal array (consisting of graphene-based spiking neurons), onto which various network structures can be mapped for different application scenarios. Furthermore, the synapses can be configured for different initial synaptic weights and plasticity, e.g., Long-Term Potentiation (LTP) and Long-Term Depression (LTD). To reconfigure the proposed graphene-based platform for a practical application, two ingredients are required: an SNN topology and an initial SNN state, e.g., initial synaptic weights. The general flow for using the proposed reconfigurable platform for real-life scenarios is as follows: For a given application and SNN topology, the SNN initial synaptic weight values are first determined by means of the specific training method described in Section 6.3.2. Subsequently, the SNN topology is mapped onto the graphene-based SNN architecture by establishing the configuration of the programmable interconnect matrix, and the synaptic array initial state (synaptic weight values and plasticity types).

To investigate the versatility and suitability of the proposed reconfigurable architecture for practical applications, we consider 2 SNN topologies tailored for character recognition and edge detection, map them on the proposed graphene neuro-platform, and evaluate their performance by means of SPICE simulations. For the first application, a 2-layer SNN consisting of 30 neurons is utilized to recognize the vowel characters, i.e., "A", "E", "I", "O", and "U" (and their variations), represented by 5 × 5 black and white pixel matrices. The response of the graphene-based SNN architecture, reconfigured according to the considered SNN topology and different initial synaptic weight values is evaluated for multiple input datasets comprising the original characters and their variations, is evaluated by means of SPICE simulations. The obtained results indicate that a recognition accuracy of up to 94.5 % is achieved, which is in line (maximum 7.8% deviation) with the one obtained by means of Matlab simulation of a functionally equivalent Artificial Neural Network (ANN).

For the second application, we consider a 3-layer SNN consisting of 13 neurons for performing edge detection on 2 images, i.e., Lena and Cameraman. To this end, for each and every image pixel, we should determine whether it belongs to an edge or not and this can be done by sequentially analyzing the pixel configuration of the 3 × 3 grayscale pixel matrix centered around it. To obtain the SNN initial synaptic weights, we make use of a set of directional edge and non-edge 3 × 3 kernels. The graphene-based SNN architecture is configured according to the SNN topology and initial state, and then SPICE simulations of all possible 3 × 3 pixel matrix instances are performed to obtain the edge extracted output image. Simulation results reveal that the graphene-based SNN platform delivers comparable results when compared with to one produced by classical edge detectors, i.e., Canny, Roberts, Sobel, Prewitt [146], which suggests good perceptual edge extracted image quality. If Peak Signal-to-Noise Ratio (PSNR) and Mean Squared Error (MSE) are utilized as evaluation metrics the SNN approach delivers slightly worse PSNR and MSE figures, i.e., 2.3% lower PSNR, and 3% higher MSE, for Lena, while for the Cameraman it marginally outperforms the classical edge detectors by 2.7% for PSNR and 2.9% for MSE.

The simulation results demonstrate that the proposed SNN platform is able to properly perform character recognition and edge detection tasks, which suggests the feasibility and flexibility of the proposed approach for various application purposes. Moreover, the utilized graphene-based synapses and neurons operate at low supply voltage (200 mV), consumes low energy per spike for both neuron (43 pJ and 5.2×10^{-7} pJ at 200 Hz and 20 GHz spike frequency scale, respectively) and synapse (5.1 pJ and 6.0×10^{-8} pJ at 200 Hz and 20 GHz spike frequency scale, respectively), and a graphene-based synapse occupies an active area of ≈45 nm^2 (2 GNR devices) and a neuron an active area of ≈176 nm^2 (6 GNR devices), which are desired properties for large-scale energy-efficient implementations.

The remaining of this chapter is organized as follows: In Section 6.2 we describe the graphene-based SNN unit, and present a general view of the simulation framework. Section 6.3 introduces the proposed reconfigurable graphene-based SNN architecture and the associated training methodology for deriving the initial synaptic weight values. Section 6.4 presents the simulation results for character recognition and edge detection applications, while Section 6.5 concludes the chapter.

6.2. Background

In this section, we present the the generic graphene-based SNN unit and a brief account on the utilized graphene circuit SPICE simulation framework.

6.2.1. Graphene-based SNN Unit

Figure 6.1 schematically illustrates the graphene-based SNN unit [145], which implements a LIF neuron and a synapse with timing dependent plasticity via 4 pairs of GNR-based devices.

The *synapse* core functionality is provided by Block 1, which receives input spikes from both the pre-synaptic (V_{spike}) and post synaptic (V_{feedback}) neurons, and generates the post-synaptic neuron input signal (V_{in}). Initial synaptic weight values can be set through GNR^1_{up} and GNR^1_{dn} back-gate bias voltages, V_{up} and V_{dn}, respectively. As for synaptic plasticity, the GNR^1_{up} cumulated trapped charges emulate long-term plasticity, while pairwise STDP modulates the synapse output signal V_{in} amplitude based on the timing difference between the pre-spike V_{spike} and post-spike V_{feedback} occurrences. Furthermore, by properly adjusting the back-gate bias voltage V_{up}, the same synapse can exhibit both Long-Term Potentiation (LTP) and Long-Term Depression (LTD), e.g., 100 mV for LTP and −100 mV for LTD. The GNR_{dn} back-gate voltage controls the inhibitory synaptic ability, i.e., 0 mV for no inhibition and 180 mV for inhibiting all incoming spikes.

The *LIF neuron* comprises the remaining blocks 2, 3 and 4. Block 2 receives V_{in} as input signal from the synapse and is responsible for capturing the integrate-and-fire membrane potential dynamics. Charges trapped into GNR^2_{up} gate oxide can cause an equivalent voltage shift denoted as ΔV. When $\text{V}_{\text{in}} + \Delta V$ reaches the firing threshold, block 2 signals a firing event occurrence via the $\text{V}_{\text{internal}}$ signal. We note that block 2 suffices to emulate the spiking neuron's dynamics. However, since the neuron spike plays an important role for the STDP process, $\text{V}_{\text{internal}}$ is further processed by block 3 in order to send a post-spike

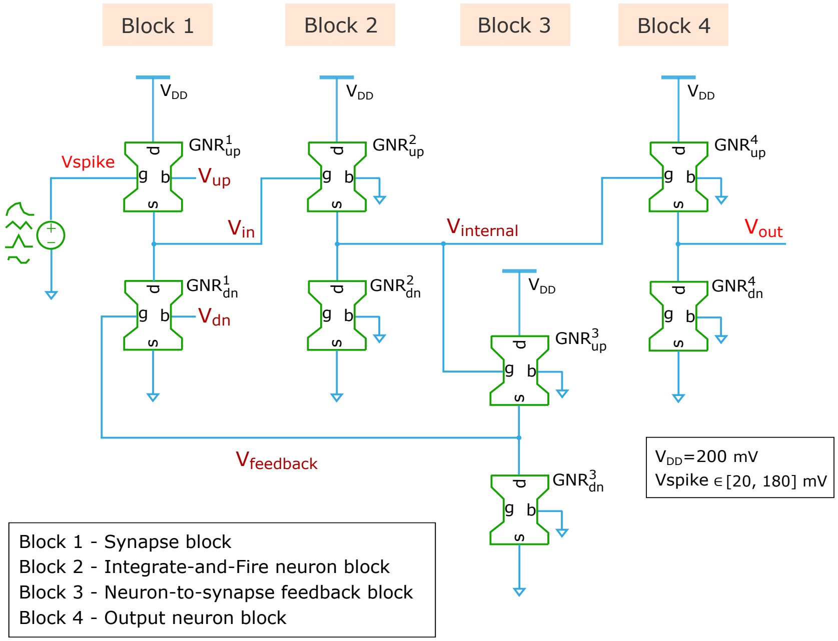


Figure 6.1: Generic SNN Unit Circuit.

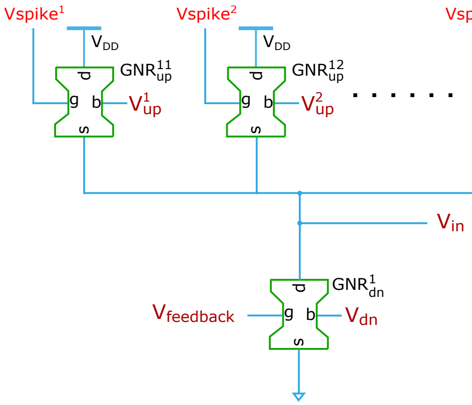


Figure 6.2: Multiple Input Synapse Block.

back to the synapse to activate the pairwise STDP plasticity. Furthermore, to enable direct cascading of SNN units, V_{internal} is also processed by block 4, which generates an output spike V_{out} that is compatible with the input spike V_{spike} in terms of voltage range (20 mV to 180 mV) and duration (2 ms). Note that the GNR topology details for all the SNN circuit devices are presented in [145].

As the basic SNN unit includes one synapse it can get input from one other neuron only while in practically relevant SNNs a neuron can be connected to hundreds of other neurons. To accommodate larger than one fan-in (i.e., n synapses connected to the same neuron) situations, the synapse block can be extended by replacing $\mathrm{GNR}_{\mathrm{up}}^{1}$ with n GNRs in parallel as illustrated in Figure 6.2 [145]. The synapses joint output voltage can then be derived as:

$$V_{in} = V_{DD} \cdot \frac{G_{up}^{11} + G_{up}^{12} + \cdots + G_{up}^{1n}}{G_{up}^{11} + G_{up}^{12} + \cdots + G_{up}^{1n} + G_{dn}^{1}}, \tag{6.1}$$

where $\mathrm{G}_{\mathrm{up}}^{1n}$ denotes the conductance of the n^{th} $\mathrm{GNR}_{\mathrm{up}}$.

6.2.2. Simulation Framework

A hybrid framework combining atomistic-level simulation for graphene-based devices and circuit level SPICE simulation in Cadence [142] is utilized to properly evaluate the proposed reconfigurable graphene-based SNN architecture.

For the GNR device electronic transport properties calculation we make use of: (i) the Tight-Binding Hamiltonian to model the external potentials and the interactions between Carbon atoms, (ii) the Non-Equilibrium Green Function (NEGF) to solve the Schrödinger equation, and (iii) the Landauer-Büttiker formula to compute the graphene channel current and conductance [102]. The GNR potential distribution profile is obtained by solving a 3D Poisson equation self-consistently. Additionally, by calculating the equivalent voltage shift induced by interface trapped charges we account for the trapping/detrapping phenomena influence on the GNR device operation [128].

For the graphene-based SNN circuit evaluation, a Verilog-A GNR device simulation model [103] is employed. To enable high accuracy and time effective SPICE simulation, we make use of precomputed look-up tables containing atomistic level GNR simulation data for the utilized graphene-based devices. Additionally, we developed a Matlab simulation model to allow for the determination of the initial synaptic weight values according to the training method described in Section 6.3.2. The obtained weights are subsequently converted into appropriate bias values that are utilized to initialize the SNN synaptic weights (via synapse $\mathrm{GNR}_{\mathrm{up}}$ back-gate voltage) in the SPICE circuit model.

6.3. Reconfigurable Graphene-based SNN Architecture

In this section we present the proposed reconfigurable graphene-based Spiking Neural Network (SNN) architecture, explain the mapping methodology of a generic SNN structure onto the proposed platform, and introduce a general training method for the determination of the initial synaptic weight values.

6.3.1. Architecture Overview

Figure 6.3 depicts a general overview of proposed reconfigurable graphene-based SNN platform. It mainly comprises a neuronal array, a synaptic array, and a peripheral Input/Output (I/O) block, which allows for SNN's communication with the computation

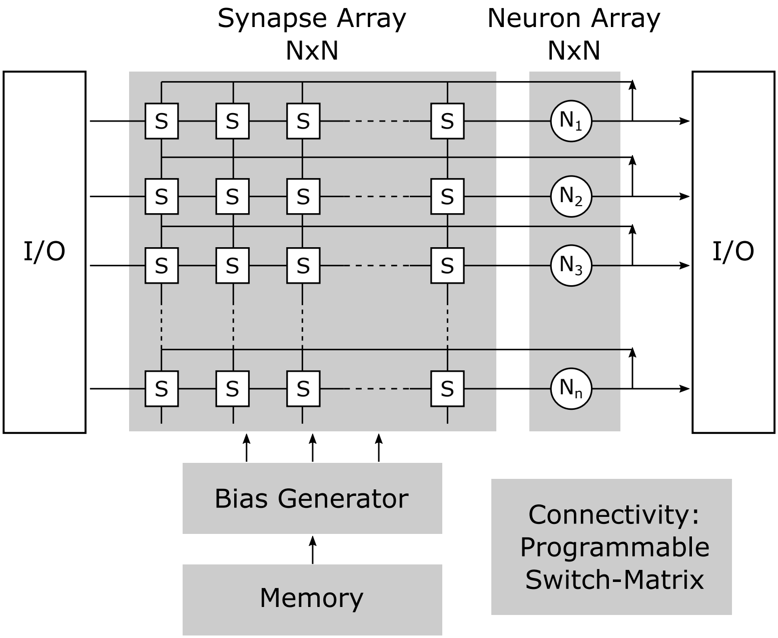


Figure 6.3: Reconfigurable Graphene-based SNN Platform.

platform (application) it embeds it. Specifically, the neuron array consists of *N* graphene-based spiking neurons, that can have their output connected either to the I/O block, when the neuron resides into the output SNN layer, or to the synapse array, when connecting with other layer neurons. The synapse array consists of *N* × *N* programmable graphene-based synapses, that can enable a connection either between neurons in different layers, or between the I/O block and SNN input layer neurons. The platform reconfiguration is enabled by means of: (i) a programmable switch matrix, which allows for SNN topology mapping onto the neuronal and synaptic arrays and (ii) an initialization module that programs the initial SNN network state (e.g., synaptic weights, plasticity types). The programmable switch matrix ensures that the signal routing within the neuronal and synaptic arrays is reflecting the desired SNN topology, by activating the appropriate interconnect crossbar row/column connections. The initialization module comprises a memory to store SNN's initial status data and a bias generator that decodes status information into voltage/current values to map the initial SNN status at electrical level. In particular, the memory module stores the synaptic weight initial values and plasticity type (e.g., LTP, LTD) for each SNN synapse while the bias generator converts these values into voltages to be applied to the GNR_{up} and GNR_{dn} back-gates of the corresponding physical synapse (as detailed in Section 6.2.1).

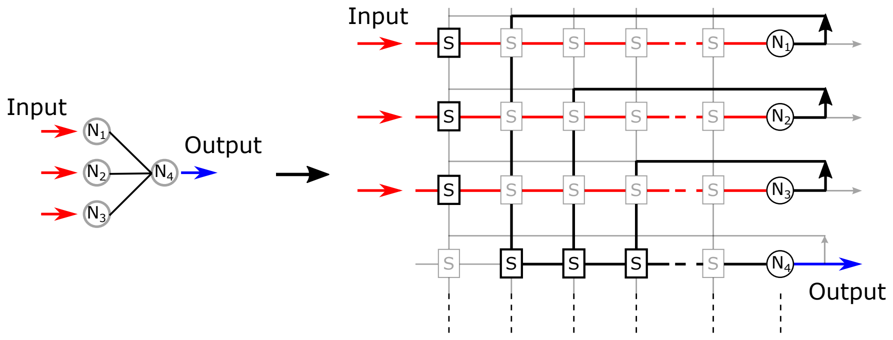


Figure 6.4: SNN Topology Mapping Example.

6.3.2. Architecture Configuration

To deploy a given application on the proposed reconfigurable graphene-based SNN architecture we make use of the following approach. First, we identify by means of state of the art approaches, e.g., [147], [148] an appropriate SNN topology for the considered application (e.g., number of layers, inter-layer connectivity, number of neurons per layer, synaptic plasticity types). Once the specific SNN topology is available, the SNN platform is reconfigured accordingly via the switch matrix. Subsequently, we identify an appropriate initial status of the SNN synaptic components by means of a training method able to determine suitable initial synaptic weight values. Finally, the per synapse weight value and plasticity type are transformed into bias voltages for the SNN synaptic array, and at this point, the SNN architecture is fully configured and ready to be utilized for the given application.

To have a better inside on the mapping process, let us consider a 2-layer SNN consisting of 3 neurons in layer 1 and 1 output neuron in layer-2 as depicted Figure 6.4. To structurally emulate this SNN on the proposed platform the neuronal array is configured such that neurons N_1, N_2, N_3 map the SNN layer 1 neurons and neuron N_4 maps the SNN layer 2 neuron. The left most column of synapses are utilized for receiving SNN input and transmitting it to the layer 1 neurons. The layer 1 to layer 2 connectivity is enabled by the row of synapses corresponding to the N_4 neuron. N_4 is also connected further to the I/O block for SNN output readout.

For the identification of the initial synaptic weight values we propose the general training flow summarized in Algorithm 1, which assuming a given SNN topology and application specific input patterns, aims to identify the best set of initial synaptic weight values that can generate the desired SNN reaction for the applied inputs, e.g., for classification tasks, different inputs patterns should be discriminated by different output neurons. The training process can be divided into two stages: (i) *Stage* 1 (steps (1) to (4) in Algorithm 1), which concerns with the definition of the desired SNN reaction by labeling the output neurons according to the input patterns they react or should react to, and (ii) *Stage* 2 (steps (5) to (9) in Algorithm 1), during which the synaptic weight values are computed via an iterative process that minimizes the difference between the obtained and the desired

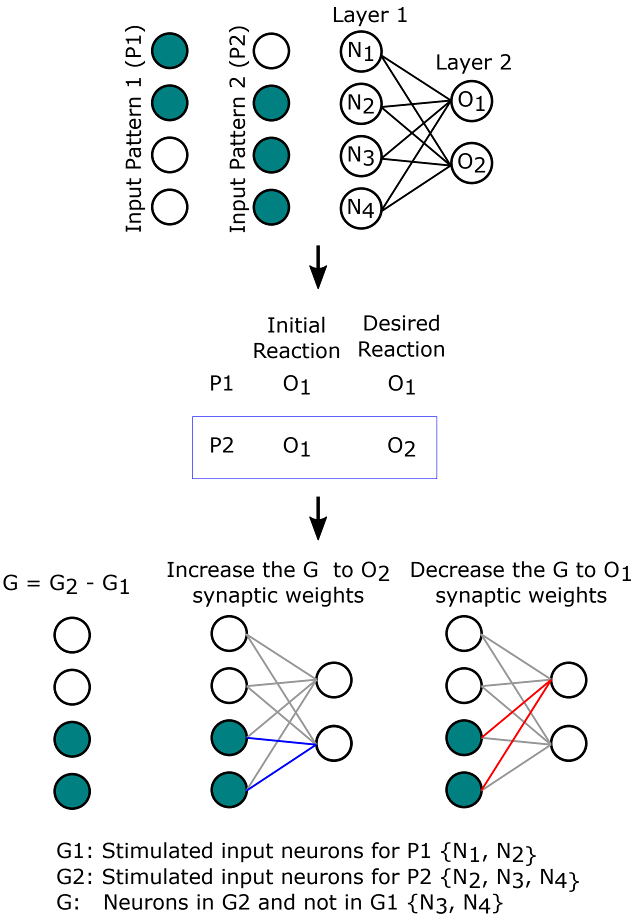

Input Pattern 1 (P1)
Input Pattern 2 (P2)
Layer 1
Layer 2
N_1
N_2
N_3
N_4
O_1
O_2
Initial Reaction
Desired Reaction
P1 O_1 O_1
P2 O_1 O_2
G = G_2 - G_1
Increase the G to O_2 synaptic weights
Decrease the G to O_1 synaptic weights
G1: Stimulated input neurons for P1 {N_1, N_2}
G2: Stimulated input neurons for P2 {N_2, N_3, N_4}
G: Neurons in G2 and not in G1 {N_3, N_4}

Figure 6.5: SNN Synaptic Weights Training Example.

SNN reaction. The training according to Algorithm 1 is carried out by means of Matlab.

In *Stage* 1, we define the desired SNN output response, e.g., which neuron should react to which input pattern such that all input patterns can be properly discriminated. To this end, we first instantiate the synaptic weights with random values (step (1)), apply the input patterns and obtain the SNN output response (step (2)). We then match all output neurons to the different SNN input patterns. Some neurons might already appropriately fire for the assumed input patterns and thus labeling them is straightforward, while others might not, case in which we enforce a label assignation. Once every output neuron has an assigned label (a designation for an SNN input category), the SNN desired reaction has been defined (step (4)).

In *Stage* 2, we update the synaptic weights repetitively until the SNN exhibits the desired output neuronal reaction for all input patterns. Specifically, the synaptic connec-

Algorithm 1 Initial synaptic weight values determination.

```
Input: SNN topology & application specific input patterns
Output: Initial synaptic weight values
1: Randomly instantiate the synaptic weight values for the given SNN;
2: Apply input patterns to the SNN, and obtain the initial SNN output response;
3: Current SNN output neuronal response ← initial SNN output neuronal response from (2);
4: Determine the desired SNN response (output neurons labeling);
5: while current SNN output response $\neq$ SNN desired output response do
6:     Update the synaptic weights based on the current and desired SNN output response (inhibit connections that might trigger undesired SNN reaction and enhance connections that help to produce the desired reaction);
7:     Apply input patterns to the SNN with updated synaptic weights, obtain new SNN output response;
8:     Current SNN output response ← new SNN output response obtained from (7).
9: end while
```

tions that contribute towards the desired SNN reaction are potentiated, while the connections that might trigger an undesired SNN reaction are depressed. For the sake of simplicity let us assume a 2 layer and 2 output SNN that has to classify input data according to two patterns P_1 and P_2. After $Stage$ 1 the output neurons O_1 and O_2 are labeled as O_1 should react to P_1, and O_2 to P_2 while the current SNN reaction is that both O_1 and O_2 react to P_1, which is not the desired behaviour. To determine which synaptic weights should be potentiated and which ones should be depressed, we first determine the reaction of the layer-1 neurons. In particular, we identify the set of input neurons that are stimulated by P_1 and P_2, and denote them by G_1 and G_2, respectively. Since O_1 is already reacting only to P_1 as it should, we are interested in changing only the O_2 reaction via synaptic weights modification. To this end, we determine the difference set $G = G_2 - G_1$, which includes all input neurons that are excited by P_2 and not excited by P_1. Then, we: (i) potentiate the synaptic connections between the neurons belonging to the difference set G and the output neuron O_2 (since we desire O_2 to react for input pattern P_2), and (ii) depress all the synaptic connections between neurons belonging to G and output neuron O_1 (as O_1 shouldn't react for input pattern P_2). Figure 6.5 illustrates an example for the synaptic weights updating process. To ensure the desired SNN reaction, i.e., O_1 reacts for P_1, and O_2 for P_2, (i) the synapses between the input neurons excited solely by P_2 and output neuron O_2 are potentiated (blue connections), and (ii) synapses from the input neurons excited solely by P_2 to output neuron O_1 are depressed (red connections).

We note that for larger SNN and problem size dimensionality, the weights update methodology described above can be applied in a sequential pairwise manner. The synaptic weights update is an iterative optimization process that ends when the difference between the current and the desired SNN output neuronal response is minimal. When completed, Stage 2 provides the set of initial synaptic weights values for the considered SNN topology.

Having generated the set of initial synaptic weights, the platform synapses GNR devices are biased, as described in Section 6.2, with back gate voltages afferent to these weights,

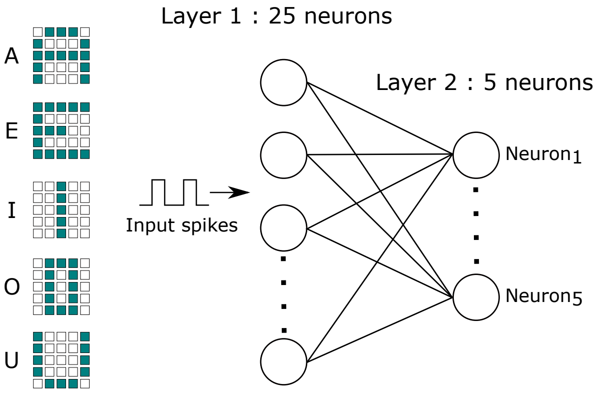


Figure 6.6: SNN for Character Recognition.

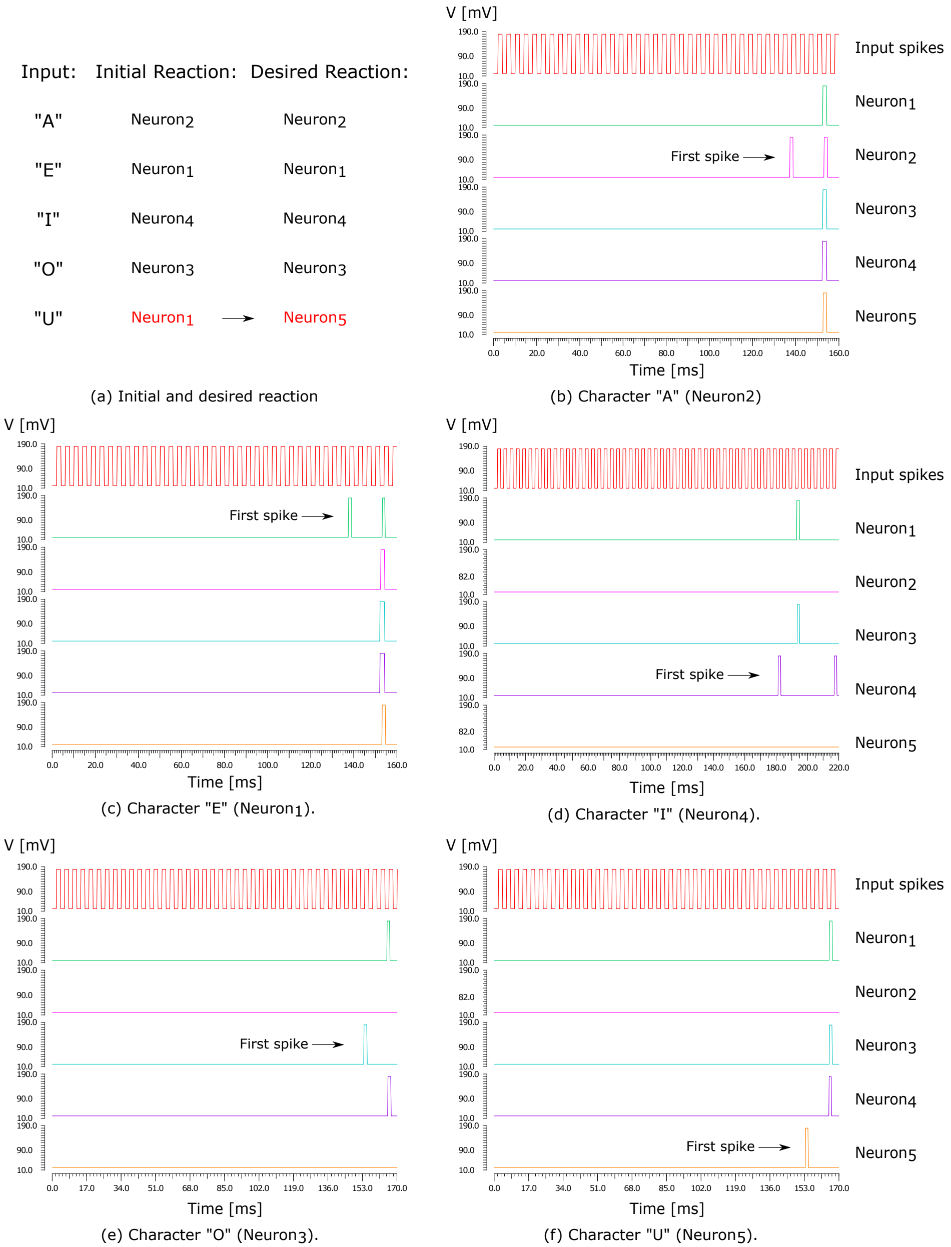
and at this point, the platform is fully configured.

6.4. SIMULATION RESULTS

To demonstrate the suitability of proposed reconfigurable graphene-based SNN architecture for various application scenarios, as well as the plausibility of the synaptic weights training method, we consider two SNN topologies that are particularly designed for character recognition and edge detection, respectively, map them on the proposed reconfigurable graphene-based SNN architecture, and investigate their run-time performance by means of SPICE simulation. In both cases we make use of 2 ms input spike pulses varying from 20 mV to 180 mV and a supply voltage V_{DD} = 200 mV. We note however that our proposal is general and can be adapted to operate on different power supply values and input spike formats.

6.4.1. CHARACTER RECOGNITION

For character recognition we rely on the 2-layer SNN comprising 30 neurons depicted in Figure 6.6, intended to recognize the vowel (and their variations) "A", "E", "I", "O", and "U". Each character is represented by a 5 × 5 black and white pixel matrix. The 25 neurons in layer 1 (L_1) serve as input neurons and each neuron corresponds to a pixel in the character matrix. For a given input character, each L_1 neuron receives input spikes if its corresponding pixel is black and no spikes if the pixel is white. The 5 neurons in layer 2 (L_2) are output neurons, each one being meant to recognize a different character. In the considered SNN, we assume that the input pixels are fed to the L_1 neurons via LTP synapses with identical weights. As concerns the L_1 to L_2 connectivity every L_2 neuron is connected with all the L_1 neurons via LTP synapses with synaptic weight values determined by means of the proposed training method in Section 6.3.2. When applying an input character, we stimulate the input neurons corresponding to the black pixels with identical 200 Hz periodic input spike trains and employ "time-to-first-spike" scheme to



Input:	Initial Reaction:	Desired Reaction:
"A"	Neuron2	Neuron2
"E"	Neuron1	Neuron1
"I"	Neuron4	Neuron4
"O"	Neuron3	Neuron3
"U"	Neuron1 ⟶	Neuron5

(a) Initial and desired reaction

(b) Character "A" (Neuron2)

(c) Character "E" (Neuron1).

(d) Character "I" (Neuron4).

(e) Character "O" (Neuron3).

(f) Character "U" (Neuron5).

Figure 6.7: SNN Recognition Reaction for Original Vowel Characters.

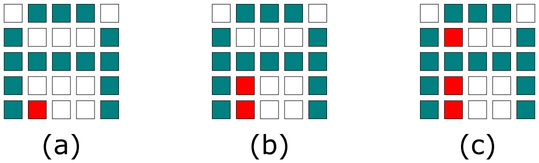


Figure 6.8: "A" Variations with Additional Pixels.

indicate the recognition result, i.e., the output neuron that fires first is the one that recognized the input character.

To determine the weights of the L_1 to L_2 synapses, we first randomly instantiate them for every synapse. Then, we apply the 5 characters to the SNN one at a time and obtain the initial recognition results depicted in Figure 6.7 (a). The Figure indicates that "E" and "U" are both recognized by the same neuron Neuron$_1$, while each of the other 3 characters is recognized by a different unique output neuron. Based on this initial recognition results, we determine the desired SNN output neuronal response, i.e., characters "A", "E", "I", "O", and "U" should be recognized by neurons Neuron$_2$, Neuron$_1$, Neuron$_4$, Neuron$_3$, and Neuron$_5$, respectively. Thus, we now need to adjust synaptic weight values such that Neuron$_5$ recognizes "U" instead of Neuron$_1$ and preserve the SNN output reaction for the other 4 characters. To achieve this we update the synaptic weights of connections between input neurons stimulated by "U" and not by "E", and the output neuron Neuron$_1$ and/or Neuron$_5$ (depression and/or potentiation). After we obtain the desired SNN output neuronal response and so the initial values of the synaptic weights, we configure the graphene-based SNN architecture and set the initial state of the synaptic array. We do so, by adjusting the back-gate bias voltage (V_{up} in each synapse block, as detailed in Section 6.2), which can take values between 0 mV and 200 mV with a 10 mV resolution. The SNN reaction for each character obtained by means of SPICE simulation of the configured graphene-based architecture is depicted in Figure 6.7 (b), (c), (d), (e), and (f), which clearly indicate that the obtained results are in line with desired recognition behaviour. As can be seen in Figure 6.7 (b) - (f), in all cases, initially, there are no firing events for the output neurons. After some time, one L_2 neuron fires (a different one for every character) and other output neurons may or may not fire afterwards. The reaction time for input characters "A", "E", "I", "O", and "U" are 135 ms, 135 ms, 180 ms, 150 ms, and 150 ms, respectively. As expected, the SNN exhibits longer reaction time for character "I" as it stimulates less input neurons in L_1 than the other characters.

The aforementioned simulation experiments utilized the 5 initial characters as input patterns. However, to get a more comprehensive assessment of the character recognition ability of the proposed graphene-based SNN, we extend the original 5 characters input patterns, with additional datasets containing variations of the original characters, obtained by adding 1, 2, or 3 extra pixels to the original characters, as exemplified in Figure 6.8 for "A". Specifically, for each original character we generate datasets corresponding to each types of considered variation and Table 6.1 summarizes the dataset cardinality for each character and variation type. To investigate the effect of the initial synaptic weights values on the SNN recognition performance we derive 4 different initial synaptic config-

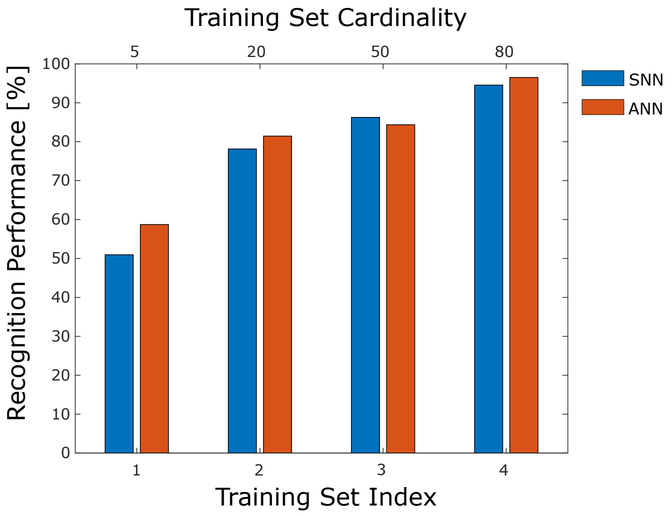


Figure 6.9: SNN vs. ANN Character Recognition Performance.

Table 6.1: Extended Dataset Cardinalities.

Variation Type	"A"	"E"	"I"	"O"	"U"
1 additional pixel	11	10	20	13	14
2 additional pixels	55	45	190	78	91
3 additional pixels	165	120	1140	286	364

urations based on the following training sets: ($S1$) - the 5 original characters, ($S2$) - ($S1$) and 15 variations (1 new pattern per character for each variation type), ($S3$) - ($S1$) and 45 variations (2 new patterns per character for each variation type), and ($S4$) - ($S1$) and 75 variations (3 new patterns per character for each variation type). For each obtained initial synaptic configuration we instantiate the corresponding graphene-based SNN architecture and evaluate its recognition performance on a test dataset comprising all the character variations input patterns not employed in the corresponding training set.

Furthermore, to put our results into proper perspective we compare the obtained classification capabilities against the ones of an Artificial Neural Network (ANN), paradigm that is widely utilized for character recognition, trained (with the gradient descent method) and evaluated on the same datasets. The Matlab modelled ANN is a 3-layer feed forward network, with 25 input neurons, a hidden layer with 5 neurons, and an output layer with 5 neurons to indicate the recognized character. The ANN and graphene-based SNN character recognition performance is presented in Figure 6.9. As can be observed in the Figure the recognition ability of both ANN and SNN improves for larger size training datasets, which is expected, from ≈ 55% for training set $S1$ with cardinality 5, up to ≈ 95% for training set $S4$ that contains 80 input patterns. Moreover, the SNN approach exhibits similar with ANN recognition performance (max. 7.8% variation), even outperforms ANN for the

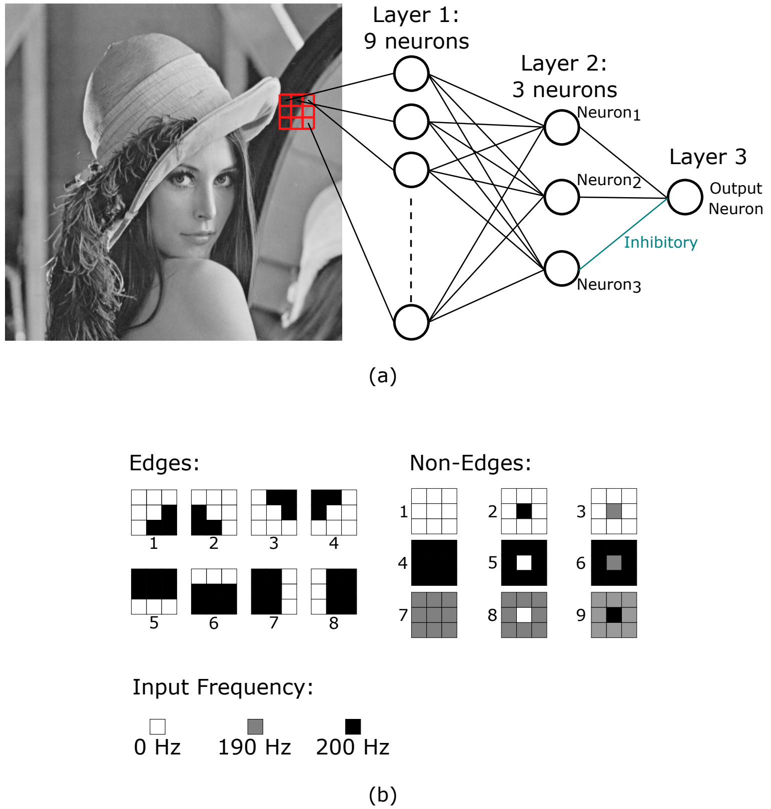
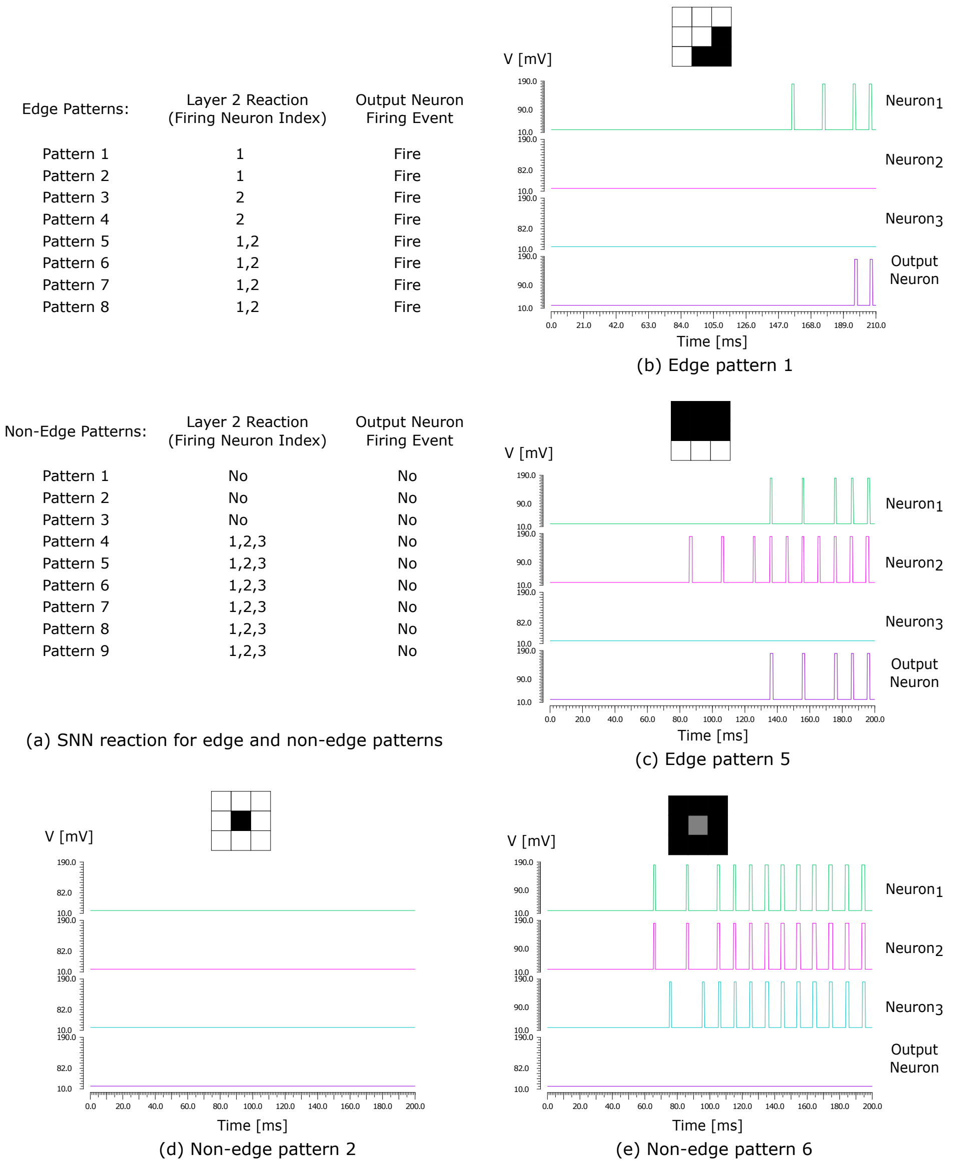


Figure 6.10: Edge Detection SNN Illustration, (a) SNN Structure, (b) Edge Patterns and Non-edge Patterns.

training set $S3$, while benefiting of all spike and graphene induced energy consumption and area advantages.

6.4.2. Edge Detection

To further demonstrate the capabilities of the proposed reconfigurable SNN architecture we consider the 3-layer SNN comprising 13 neurons depicted in Figure 6.10 (a) and employ it to perform edge detection on the celebrated Lena and Cameraman images. To this end, for each and every image pixel, we should determine whether it belongs to an edge or not and this can be done by sequentially analyzing the pixel configuration of the 3×3 grayscale pixel matrix centered around it. Each layer 1 (L_1) neuron receives an input spike train which frequency is determined by the grey levels of the 3×3 matrix pixel it connects with. We assume that: (i) the input patterns are fed to the L_1 neurons via LTP synapses with identical synaptic weights, (ii) L_1 and L_2 neurons are fully connected via LTP synapses with initial weight values determined based on a set of directional filters that detect 3×3 edge and non-edge patterns, and (iii) for L_2 to L_3 connectivity, LTP synapses with identi-



Edge Patterns:	Layer 2 Reaction (Firing Neuron Index)	Output Neuron Firing Event
Pattern 1	1	Fire
Pattern 2	1	Fire
Pattern 3	2	Fire
Pattern 4	2	Fire
Pattern 5	1,2	Fire
Pattern 6	1,2	Fire
Pattern 7	1,2	Fire
Pattern 8	1,2	Fire

Non-Edge Patterns:	Layer 2 Reaction (Firing Neuron Index)	Output Neuron Firing Event
Pattern 1	No	No
Pattern 2	No	No
Pattern 3	No	No
Pattern 4	1,2,3	No
Pattern 5	1,2,3	No
Pattern 6	1,2,3	No
Pattern 7	1,2,3	No
Pattern 8	1,2,3	No
Pattern 9	1,2,3	No

(a) SNN reaction for edge and non-edge patterns

(b) Edge pattern 1

(c) Edge pattern 5

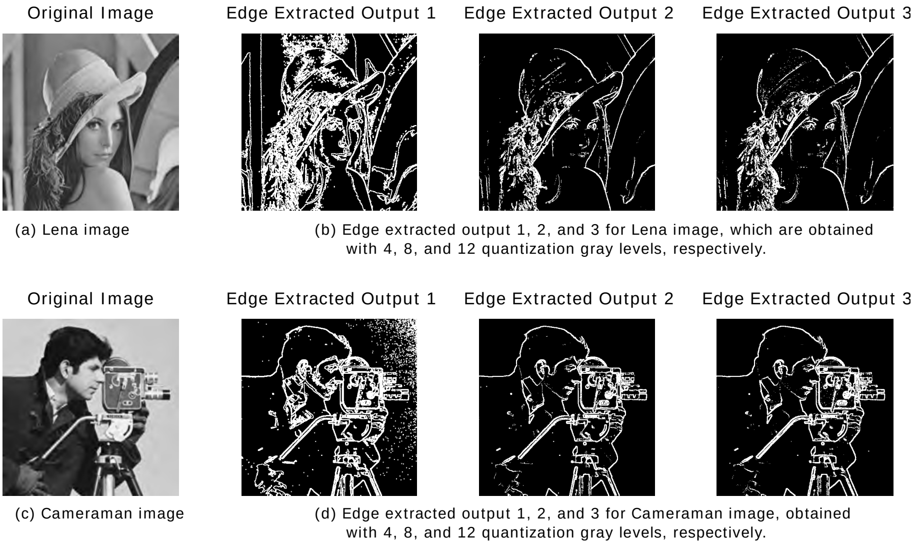
(d) Non-edge pattern 2

(e) Non-edge pattern 6

Figure 6.11: SNN Reaction for Edge and Non-Edge Input Patterns.

cal synaptic weights are utilized to connect Neuron$_1$ and Neuron$_2$ to the output neuron, while an inhibitory connection is in place between Neuron$_3$ and the output neuron.

To determine the initial weight values for the synapses connecting L_1 and L_2, we first assign them random values. We use as SNN input patterns a series of edge and non-edge patterns [148] formalized as 3×3 grayscale pixels matrices, as depicted in Figure 6.10 (b).



(a) Lena image

(b) Edge extracted output 1, 2, and 3 for Lena image, which are obtained with 4, 8, and 12 quantization gray levels, respectively.

(c) Cameraman image

(d) Edge extracted output 1, 2, and 3 for Cameraman image, obtained with 4, 8, and 12 quantization gray levels, respectively.

Figure 6.12: Edge Detection Results.

To represent the 3 grey levels (white, grey, black) in the considered edge and non-edge patterns, we use input spike trains with 0 Hz, 190 Hz, and 200 Hz frequency, respectively. As desired SNN reaction, we would like the SNN output neuron to fire when an edge pattern is applied as input, and to not fire for non-edge input patterns. To induce the desired SNN output reaction, we need to update the L_1 to L_2 synaptic weights. As can be observed in Figure 6.10 (b), the edge patterns and non-edge patterns stimulate different numbers of input neurons. Specifically, the edge patterns stimulate 3 or 6 input neurons while the non-edge patterns stimulate 0, 1, 8 or 9 input neurons. Thus, we expect the non-edge patterns either to induce no firing event in L_2 neurons, or to induce firing events in more L_2 neurons than the edge patterns do. Therefore, we would like to take advantage of the L_2 neurons that are firing only for the non-edge patterns in order to induce the desired SNN output neuron reaction. Since this reaction for non-edge patterns is "do not fire", we can exploit the spiking of the neurons in L_2 that are firing only for non-edge patterns, in order to inhibit the SNN output neuron. In particular, we designate Neuron$_3$ in L_2 to fire only for non-edge patterns. Thus, we depress all incoming synaptic connections to Neuron$_3$ and inhibit its outgoing connection to the SNN output neuron. Since for this particular application, the initial synaptic weights values for desired SNN reaction to edge and non-edge patterns can be derived as previously described, we don't need to make use the methodology introduced in Section 6.3. After obtaining the initial synaptic weights, the graphene-based SNN is configured accordingly, and the SNN reaction to the 18 3×3 edge and non-edge input patterns evaluated by means of SPICE simulation. The SNN reaction is summarized in Figure 6.11(a) and, for exemplification purpose, graphically presented in Figure 6.11 (b) - (e), for edge pattern 1, edge pattern 5, non-edge pattern 2, and non-edge pattern 6, respectively. We note that for edge patterns at least one L_2 neuron is firing,

Table 6.2: Edge Detection Results Quantitative Evaluation.

	Lena Image		Cameraman Image	
	PSNR	MSE	PSNR	MSE
Canny	5.4486	$1.8545E+04$	4.6640	$2.2217E+04$
Roberts	5.5965	$1.7924E+04$	4.8678	$2.1198E+04$
Sobel	5.5718	$1.8026E+04$	4.8609	$2.1232E+04$
Prewitt	5.5708	$1.8030E+04$	4.8609	$2.1232E+04$
SNN 4 levels	5.1763	$1.9744E+04$	4.9972	$2.0576E+04$
SNN 8 levels	5.4381	$1.8589E+04$	4.8736	$2.1170E+04$
SNN 12 levels	5.4690	$1.8458E+04$	4.9798	$2.0658E+04$

which makes the SNN output neuron to fire while for the non-edge patterns, there are no firing events for the SNN output neuron, either because none of the L_2 neurons fire, or because the inhibitory neuron Neuron$_3$ fires and suppresses all the other L_2 to L_3 firing activity.

To evaluate the edge detection ability of the obtained SNN, we consider two grayscale images, i.e., Lena and Cameraman, depicted in Figure 6.12 (a) and (c), and rely on SPICE simulation to obtain the detection results. Prior to image processing they are first quantized in order to reduce the number of gray levels. Each pixel appurtenance or not to an edge is determined by the SNN processing of the 3 × 3 window centered in that pixel and after all pixels are scanned the edge detection result forms a black and white image, where black pixels belong to edges and white pixels don't.

To get inside on the way quantization affects the edge detection performance, we perform edge detection on 4-, 8-, and 12-levels quantized images. We encode each grey level pixel as a spike train with a different frequency ranging from 0 Hz (white) to 200 Hz (black). Figure 6.12 (b) and (d) present the edge extraction results for different quantization levels for Lena and Cameraman, respectively. A visual inspection of Figure 6.12 images reveals that 4 grey level quantization results in blurred edge images, while 8- and 12-level quantization in clear and sharp edges. Note that the image quality improvement becomes only marginal beyond a certain number of quantization levels (e.g., the difference between 8-level and 12-level quantized images generated edges are almost imperceptible).

To assess the quality of the SNN edge detection results we compare them against the Matlab obtained results for 4 classical edge detection operators, i.e., Canny, Roberts, Sobel, and Prewitt [149] applied directly on the original images (without prior quantization), in terms of the Peak Signal-to-Noise Ratio (PSNR) and the Mean Squared Error (MSE), which measure the perceptual distortion between the original images and the edge extracted counterparts [150]. We note that a higher PSNR indicates a higher quality image, while a lower MSE value promises a better image quality. Table 6.2 presents a comparative summary of the PSNR and MSE values computed for the edge extracted images, obtained with the classical edge detection algorithms and with the proposed graphene-based SNN when using different quantization levels. By inspecting the PSNR and MSE results for Lena image in Table 6.2, we note that the best performing edge detector is Roberts and that the SNN counterpart exhibit only slightly worse performance: (i) 7.5%, 2.8%, and 2.3% lower PSNR values and (ii) 10%, 3.7%, and 3% higher MSE values, for the SNN with 4, 8, and 12

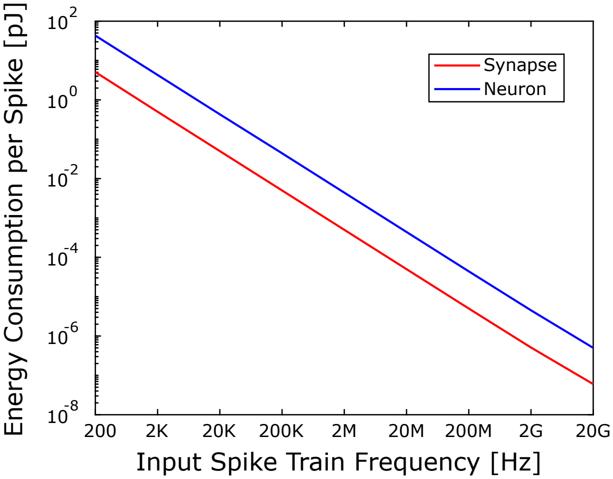


Figure 6.13: SNN Unit Energy Consumption vs. Input Spike Train Frequency.

gray levels, respectively. Looking at the SNN results, we note that in general the finer the gray level quantization the better the edge image quality, which is in agreement with the visual perception one gets when inspecting Figure 6.12 (b). For the Cameraman image, the SNN detector outperforms the classical counterparts, with Roberts fairing the best among the classical detectors. In particular, when comparing to Roberts figures, the SNN provides (i) 2.7%, 0.1%, and 2.3% higher PSNR values and (ii) 2.9%, 0.1%, and 2.6% lower MSE values for the SNN with 4, 8, and 12 gray levels, respectively. To conclude, the edge detection simulation results indicate that the graphene-based SNN platform delivers comparable performance with classical edge detectors for the considered Lena and Cameraman images, while providing all the benefits of SNN base processing paradigm.

6.4.3. AREA AND ENERGY EVALUATION

While an accurate evaluation of the area and energy consumption of our proposal is not possible at this stage of development it is of interest to get some inside on those two aspects. As it concerns the SNN unit area, a graphene-based synapse occupies an active area of ≈45 nm^2 (2 GNR devices) and a neuron requires an active area of ≈176 nm^2 (6 GNR devices) [145]. To evaluate the energy consumption and get sight into energy expenditures at different time scales, we considered an SNN unit and apply rectangular spikes with 40% duty cycle as input, while varying the input spike frequency within the range of 200 Hz to 20 GHz. We then measure the energy required by the SNN unit neuron to generate a spike and by the SNN unit synapse to perform plasticity modulation and spike transmission. The obtained results graphically presented in Figure 6.13 indicate ≈ 8 orders of magnitude decrease in energy consumption per spike for both the neuron (from 43 pJ at 200 Hz to 5.2×10^{-7} pJ at 20 GHz) and the synapse (from 5.1 pJ at 200 Hz to 6.0×10^{-8} pJ at 20 GHz).

Table 6.3: Area and Energy Consumption for Biological and State-of-the-Art Artificial Neurons.

	Neuron Type	Technology Type	Supply Voltage [V]	Area [μm^2]	Energy per Spike [pJ]	Spike Timescale
Biological	-	-	-	-	2.5×10^{5}	ms
2020 [151] biomimetic	Izhikevich	CMOS (65 nm)	0.3	1.5×10^{2}	2.4×10^{-2}	μs
2020 [151] processing	Izhikevich	CMOS (65 nm)	0.3	1.5×10^{2}	2.0×10^{-2}	μs
2017 [152]* biomimetic	ML	CMOS (65 nm)	0.2	2.0×10^{2}	7.8×10^{-2}	μs
2017 [152]* processing	ML	CMOS (65 nm)	0.2	3.5×10^{1}	4.0×10^{-3}	μs
2015 [153]	LIF	CMOS-SOI (20 nm)	0.8	9.0×10^{-2}	1.2×10^{1}	μs
2017 [154] biomimetic	LIF	CMOS-NEMS (28 nm)	0.5	8.0×10^{-2}	2.5×10^{-1}	μs
2018 [155]*	LIF	Memristor	-	-	1.0×10^{-2}	ns
Proposed	LIF	Graphene	0.2	1.8×10^{-4}	$5.2 \times 10^{-7} - 4.3 \times 10^{1}$	ps – ms

* Experimental results from fabricated devices.

Table 6.4: Area and Energy Consumption for Biological and State-of-the-Art Artificial Synapses.

	Synapse Type	Technology	Operating Voltage [V]	Area [μm^2]	Energy/Spike [pJ]	Spike Timescale
Biological	-	-	Spike: $[-0.04, 0.07]$	-	$1.0 - 1.0 \times 10^{1}$	ms
2020 [69]	STDP	CMOS (65 nm)	Supply: 1.2	-	$1.0 \times 10^{-3} - 5.0 \times 10^{-2}$	μs
2015 [134]*	STDP	CMOS (28 nm)	Supply: 0.2	1.3×10^{2}	$2.3 \times 10^{-6} - 3.0 \times 10^{-5}$	ms
2017 [133]*	STDP	Memristor	Spike: $[-5, 1.5]$	1.0×10^{-2}	-	ms
2020 [156]	LTP	PCM	Control: 2.5	-	$5.0 - 3.0 \times 10^{1}$	ns
Proposed	STDP+LTP	Graphene	Supply: 0.2 Spike: $[0.02, 0.18]$	4.5×10^{-5}	$6.0 \times 10^{-8} - 5.1$	ps – ms

* Experimental results from fabricated devices.

Further, to have a better view on the potential of using the proposed graphene-based SNN architecture for large-scale energy efficient implementations, we summarize in Table **??** and **??** the area and energy consumption figures for biological and state-of-the-art artificial neurons and synapses, respectively. We note that, the graphene-based SNN unit (neuron+synapse) can potentially save at least ≈ 2 orders of magnitude area estate when compared with both neurons and synapses state-of-the-art implementations.

From the energy standpoint, the proposed SNN unit can consume up to 1 order of magnitude more than state-of-the-art counterparts if operated with spike pulse width in the order of, e.g., µs due to leakage but can achieve up to 4 orders of magnitude energy savings if operated with short spike pulse width in the order of, e.g., ps. For the presented applications, we considered a biologically plausible time scale, i.e., ms, and, as such, we obtained an average energy consumption for the entire SNN architecture neuronal and synaptic arrays of 1.98×10^4 pJ per character for the vowels recognition application, and of 1.21×10^4 pJ and 1.39×10^4 pJ per edge pixel and non-edge pixel, respectively, for the edge detection application.

Note that the proposed SNN unit is generic thus it is by no means restricted to the considered design constraints. The SNN architecture can be tailored to function under different spike width scales by considering different trapping/de-trapping time constant values for the GNRs of the neuronal and synaptic array devices. Thus, we can target both biological, which require low input frequency and a specific voltage ranges, and fast processing scenarios for which a ns timing scale operation would be more appropriate from the computation speed point of view. To accommodate different application targets, the SNN architecture neuronal and synaptic arrays can be partitioned into different frequency islands, i.e., for each frequency island the neurons and synapses GNRs are designed with trapping mechanisms that match the island time scale. At run-time, depending on the application constraints, one can map the SNN topology either to the higher frequency islands for fast non-cortical processing, or to the lower frequency ones for bio-mimetic processing.

6.5. CONCLUSIONS

In this chapter, we proposed a reconfigurable graphene-based Spiking Neural Network (SNN) architecture and a training methodology for obtaining the initial SNN synaptic weight values. The proposed architecture supports artificial synapses with programmable plasticity and reconfigurable connectivity between the neuronal and synaptic arrays. To investigate the versatility and suitability of the proposed architecture to accommodate and evaluate the behaviour of different SNN topologies we considered 2 SNN applications particularly designed for character recognition and edge detection. We mapped on the generic graphene-based platform a 2-layer SNN tailored for vowel characters recognition and demonstrated by mens of SPICE simulations that it can achieve up to 94.5 % recognition accuracy for the considered training and evaluation datasets, which is very close to the one achieved by a functionally equivalent ANN counterpart. Further, we mapped and evaluated a 3-layer SNN to perform edge detection on Lena and Cameraman images and demonstrated that the edge detection result quality matches and even outperform the one obtained with classical edge detection operators. In summary, the proposed reconfigurable graphene SNN architecture exhibits: (i) area and energy efficiency due to effective graphene-based implementation of neurons and synapses, (ii) flexible support for SNN applications mapping due to FPGA-alike reconfiguration feature, and (iii) training process simplicity due to Spike-Timing-Dependent Plasticity (STDP) and Long-Term Plasticity support.

7

CONCLUSIONS

In this thesis, we first introduced an atomistic-level simulation model to calculate graphene electronic transport properties, which captures the hysteresis effects induced by interface charges trapping/detrapping phenomena. Second, we investigated the graphene potential for artificial synapse and neuron implementations, and proposed generic one- and two-top gates graphene-based synapse structures, as well as a compact entirely graphene-based neuron that mimics nonlinear leaky integrate-and-fire spiking neuron dynamics. Subsequently, we introduced a basic graphene-based Spiking Neural Network (SNN) unit consisting of a synapse and a spiking neuron with input-output compatibility, which can be utilized to implement complex SNNs. An extension approach is also provided to accommodate larger than one fan-in situation, i.e., multiple synapses connected to the same neuron. Finally, we proposed a reconfigurable graphene-based SNN architecture and an associated training methodology for initial synaptic weight values determination. This chapter summarizes the overall achievements of this thesis and highlights some future research directions.

7.1. SUMMARY

Chapter 1 : Introduction
In this chapter, we briefly discussed the neuromorphic computing landscape and introduced the research questions addressed in this thesis. We first presented the motivation for this thesis work. Subsequently, we presented the limitations of state-of-the-art neuromorphic systems implemented with conventional and emerging technologies, and the opportunities and challenges for graphene-based neuromorphic computing. Next, we discussed the research questions and summarized the contributions of this thesis.

Chapter 2 : Graphene Structures Electron Transport Model
In this chapter, we introduced an atomistic-level graphene device simulation model that can calculate the electric transport properties of Graphene NanoRibbon (GNR) devices with rectangular or non-rectangular GNR geometries and capture hysteretic effects caused by interface charges trapping/detrapping phenomena. We applied the model on a rectangular graphene shape and validated the results against experimentally measured data. As opposed to state-of-the-art counterparts, which are only applicable for rectangular topology the proposed approach can capture hysteresis effects on structures with various graphene geometries and we exemplified this for two non-rectangular GNRs. The experiments indicated good consistency between simulated and measured results, indicating that the model is suitable for traps-aware evaluation of graphene-based device and circuit conduction.

Chapter 3 : Graphene-based Synapses with Versatile Plasticity
In this chapter, we proposed generic one- and two- top gates graphene-based synapse structures. We demonstrated that by properly changing GNR dimensions, shape, and contacts topology, graphene devices are capable of mimicking various synaptic plasticity types. We successfully emulated two fundamental synaptic functionalities: Spike-Timing-Dependent Plasticity (STDP) and Long-Term Plasticity, including Long-Term Potentiation (LTP) and Long-Term Depression (LTD). Moreover, the graphene synapses are programable by means of back-gate bias voltage and the same device can exhibit both LTP and LTD. Our simulation results indicated that the one-top-gate synapse can achieve the 100% plasticity change provided by natural synapses. The two-top-gates synapse exhibits STDP with spike duration dependent potentiation/depression time scale without affecting the obtained synaptic weight change amplitude while achieving a maximum of 30% synaptic weight change and potentiation/depression time scale range from $[-1.5\,\text{ms}, 1.1\,\text{ms}]$ to $[-32.2\,\text{ms}, 24.1\,\text{ms}]$. We note that given that we made use a generic approach to identify appropriate GNR topology for a desired synaptic plasticity, our proposal is not limited to the 4 STDP and 2 LTP forms discussed in the chapter. Furthermore, we investigated the impact of two-top-gates synapse at the SNN level by performing NEST based simulations. Our experiments indicated a strong corelation between the synaptic plasticity type, i.e., Hebbian and anti-Hebbian, and the number of firing events occurring within the network and that the number of SNN output firing events monotonously varies with respect to the input spikes frequency. For Hebbian STDP and a spike duration of 20 ms we obtained an SNN behavior similar with the one provided by the same SNN with biological STDP. The proposed graphene-based synapses have small area ($30\,\text{nm}^2$), operate in the

100 mV bias and input range, and can emulate various plasticity types, which is making them very promising candidates for scalable energy-efficient neuromorphic system implementations.

Chapter 4 : Graphene-based Nonlinear Leaky Integrate-and-Fire Spiking Neuron
In this chapter, we proposed an entirely graphene-based ultra-compact and low voltage nonlinear leaky integrate-and-fire spiking neuron. It consists of 6 GNR-based devices controlled via top-gate voltages, one of them emulating the membrane potential dynamics, and the remaining 5 generating the necessary control signals as well as the output spikes. We validated, by means of SPICE simulations, the basic neuron functionality under periodic input spike trains and noisy stochastic inputs. Our simulation results indicated variability resilience and neuronal output firing regularity for a varying input firing rate (from 20 to 200 spikes per second). For all simulation, we utilized spike duration and amplitude of 2 ms and 100 mV, respectively, which are comparable to those observed in biological neurons. Moreover, the small area (GNR-based device area of max. 36 nm^2) and low energy (inherent to the 200 mV supply voltage) are certainly enabling factors for the potential implementation of large-scale artificial neural systems.

Chapter 5 : Graphene-based Spiking Neural Network
In this chapter, we introduced a basic graphene-based SNN unit consisting of a synapse and a spiking neuron with input-output compatibility that can be utilized to implement complex SNNs. An extension approach is provided to accommodate larger than one fan-in, i.e., multiple synapses connected to the same neuron. We validated the basic operation of the proposed design by relying on the mixed atomistic-circuit simulation framework and investigated how the enabled synaptic plasticity affects SNN behaviour. To this end we assumed a 2-layer SNN, evaluated its response to the same input stimuli by means of SPICE and NEST simulations, and demonstrated a close agreement between the obtained results in terms of total number of firing events and mean Inter-Spike Interval (ISI) length. Further, we demonstrated the unsupervised learning capability of our proposal by considering a two layer SNN consisting of 30 neurons meant to recognize the characters "A”, "E”, "I”, "O”, and "U" (and variations of them). Each character is represented with a 5 by 5 black and white pixel matrix and the simulation results indicated that the graphene SNN is able to perform unsupervised character recognition and that its recognition capacity is robust to input character variations. Finally, we noted that the proposed SNN unit requires a small real-estate footprint (max. 30 nm^2 are required by one graphene-based device) and operates at 200 mV supply voltage, which suggested its suitability for the design of scalable energy-efficient computing systems.

Chapter 6 : Reconfigurable Graphene-based Spiking Neural Network Architecture
In this chapter, we proposed a reconfigurable graphene-based SNN architecture and a training methodology for obtaining the initial SNN synaptic weight values. The proposed architecture supports artificial synapses with programmable plasticity and reconfigurable connectivity between its neuronal and synaptic arrays. To investigate the versatility and suitability of the proposed architecture for practical utilization, we considered 2 SNN applications, i.e., character recognition and edge detection. We first mapped on the generic graphene-based platform a 2-layer SNN tailored for vowel characters recognition and demon-

strated by means of SPICE simulations that it can achieve up to 94.5 % recognition accuracy for the considered training and evaluation datasets, which is very close to the one achieved by a functionally equivalent Artificial Neural Network (ANN) counterpart. Further, we mapped a 3-layer edge detection tailored SNN, evaluated it on Lena and Cameraman images and demonstrated that its edge detection abilities matches and even outperforms the one of classical edge detection operators. Our results demonstrated the feasibility and flexibility of the proposed approach for various application purposes. Moreover, the proposed reconfigurable graphene SNN architecture exhibits area and energy efficiency due to effective graphene-based implementation of neurons and synapses, which are desired properties for scalable energy-efficient implementations.

7.2. FUTURE RESEARCH DIRECTIONS

Subsequently, we discuss several future research directions that are suggested to further explore and enrich the potential of graphene for neuromorphic computing implementations.

- **Graphene-based biologically plausible neuromorphic systems that can aid neuroscience research.**
  One of the primary goals of neuromorphic computing is to develop neuromorphic systems that can help the investigation of the complex human brain functionalities by neuroscience researchers. Such neuromorphic systems ought to be capable of precisely emulating biological neuronal and synaptic dynamics. To this end, spiking neuron models with high biological plausibility, e.g., the Hodgkin–Huxley model, and synapses with accurate emulation of biological synaptic behaviors are required. As graphene devices exhibit outstanding capability for complex functionality emulation while preserving low energy operations and small footprints, they are promising candidates for biologically plausible neuromorphic implementations. To this end, a more sophisticated graphene-based circuitry than the one employed to mimic the Integrate-and-Fire spiking neuron presented in Section 4 may be required to emulate the complex biologically plausible neuron functionality. The identification of biologically plausible synapse and neurone circuits can certainly builds upon the work in this thesis and make use of the hybrid atomistic-circuit level simulation framework.

- **Graphene-based neuromorphic systems as neural interfaces.**
  Graphene's outstanding electrical and mechanical properties, e.g., high current density, good electrical conductivity, ultimate thinness, and flexible, as well as its biocompatibility, make it an appealing option for building brain-machine interfaces. The graphene-based neural interfaces may be inserted or worn as part of medical care or tracking items, or they can interact directly with biological tissues. To allow these graphene-based neural interfaces must be capable of interacting with biological systems using the same protocol, e.g., receiving, decoding, and encoding spike-based signals. Despite the fact that the capacity of graphene-based neuromorphic systems to process spike-based signals is validated in the thesis, the real signals in the human brain are noisy and stochastic thus differ from the ones used

in the proposed graphene-based SNN assessments. As such, further investigations on graphene-based implementations for achieving the ability to process real human brain signals are required to enable the direct brain-machine interaction. This research can further explore and enrich the potential of graphene for the implementations of various neuromorphic systems for diverse purposes.

- **Investigation of graphene-based SNNs' utilization for other application beyond the considered image processing tasks.**
  In the thesis, we explored the capability of graphene-based SNNs to perform image processing practical tasks, i.e., character recognition and edge detection. When referring to other practical application types, e.g., speech and video processing, different SNN structures and associated learning abilities need to be investigated as the to be processed data are fundamentally different, e.g., contain time series information. Furthermore, to facilitate the utilization of graphene-based neuromorphic systems for diverse applications, a generic methodology that can assist in determining the appropriate application-specific SNN configurations, e.g., network topology, is required. This research can help to improve the capability of the graphene-based neuromorphic system to solve various practical tasks.

REFERENCES

[1] D. C. Ciresan, U. Meier, L. M. Gambardella, and J. Schmidhuber. “Convolutional neural network committees for handwritten character classification”. In: *2011 International Conference on Document Analysis and Recognition*. IEEE. 2011, pp. 1135–1139.

[2] S. Lai, L. Xu, K. Liu, and J. Zhao. “Recurrent convolutional neural networks for text classification”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 29. 1. 2015.

[3] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu. “Textboxes: A fast text detector with a single deep neural network”. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 31. 1. 2017.

[4] L. Xu, J. S. Ren, C. Liu, and J. Jia. “Deep convolutional neural network for image deconvolution”. In: *Advances in neural information processing systems* 27 (2014), pp. 1790–1798.

[5] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang. “Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising”. In: *IEEE transactions on image processing* 26.7 (2017), pp. 3142–3155.

[6] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang, and W. Xu. “Cnn-rnn: A unified framework for multi-label image classification”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2285–2294.

[7] W. Chan, N. Jaitly, Q. Le, and O. Vinyals. “Listen, attend and spell: A neural network for large vocabulary conversational speech recognition”. In: *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2016, pp. 4960–4964.

[8] H. Sak, A. Senior, K. Rao, and F. Beaufays. “Fast and accurate recurrent neural network acoustic models for speech recognition”. In: *arXiv preprint arXiv:1507.06947* (2015).

[9] Y. Miao, M. Gowayyed, and F. Metze. “EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding”. In: *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE. 2015, pp. 167–174.

[10] M.-O. Gewaltig and M. Diesmann. “Nest (neural simulation tool)”. In: *Scholarpedia* 2.4 (2007), p. 1430.

[11] D. F. Goodman and R. Brette. “Brian: a simulator for spiking neural networks in python”. In: *Frontiers in neuroinformatics* 2 (2008), p. 5.

[12] M. L. Hines and N. T. Carnevale. “NEURON: a tool for neuroscientists”. In: *The neuroscientist* 7.2 (2001), pp. 123–135.

[13] C. Mead. “Neuromorphic electronic systems”. In: *Proceedings of the IEEE* 78.10 (1990), pp. 1629–1636.

[14] C. D. Schuman, T. E. Potok, R. M. Patton, J. D. Birdwell, M. E. Dean, G. S. Rose, and J. S. Plank. “A survey of neuromorphic computing and neural networks in hardware”. In: *arXiv preprint arXiv:1705.06963* (2017).

[15] G. Indiveri, B. Linares-Barranco, T. J. Hamilton, A. Van Schaik, R. Etienne-Cummings, T. Delbruck, S.-C. Liu, P. Dudek, P. Häfliger, S. Renaud, *et al.* “Neuromorphic silicon neuron circuits”. In: *Frontiers in Neuroscience* 5 (2011), p. 73.

[16] Y. van De Burgt, A. Melianas, S. T. Keene, G. Malliaras, and A. Salleo. “Organic electronics for neuromorphic computing”. In: *Nature Electronics* 1.7 (2018), pp. 386–397.

[17] N. Qiao, H. Mostafa, F. Corradi, M. Osswald, F. Stefanini, D. Sumislawska, and G. Indiveri. “A reconfigurable on-line learning spiking neuromorphic processor comprising 256 neurons and 128K synapses”. In: *Frontiers in Neuroscience* 9 (2015), p. 141.

[18] B. V. Benjamin, P. Gao, E. McQuinn, S. Choudhary, A. R. Chandrasekaran, J.-M. Bussat, R. Alvarez-Icaza, J. V. Arthur, P. A. Merolla, and K. Boahen. “Neurogrid: A mixed-analog-digital multichip system for large-scale neural simulations”. In: *Proceedings of the IEEE* 102.5 (2014), pp. 699–716.

[19] F. Akopyan, J. Sawada, A. Cassidy, R. Alvarez-Icaza, J. Arthur, P. Merolla, N. Imam, Y. Nakamura, P. Datta, G.-J. Nam, *et al.* “Truenorth: Design and tool flow of a 65 mw 1 million neuron programmable neurosynaptic chip”. In: *IEEE Transactions on Computer-aided Design of Integrated Circuits and Systems* 34.10 (2015), pp. 1537–1557.

[20] M. Davies, N. Srinivasa, T.-H. Lin, G. Chinya, Y. Cao, S. H. Choday, G. Dimou, P. Joshi, N. Imam, S. Jain, *et al.* “Loihi: A neuromorphic manycore processor with on-chip learning”. In: *IEEE Micro* 38.1 (2018), pp. 82–99.

[21] D. Monroe. *Neuromorphic computing gets ready for the (really) big time*. 2014.

[22] J. Schemmel, D. Briiderle, A. Griibl, M. Hock, K. Meier, and S. Millner. “A wafer-scale neuromorphic hardware system for large-scale neural modeling”. In: *IEEE International Symposium on Circuits and Systems*. 2010, pp. 1947–1950.

[23] J. Shen, D. Ma, Z. Gu, M. Zhang, X. Zhu, X. Xu, Q. Xu, Y. Shen, and G. Pan. “Darwin: a neuromorphic hardware co-processor based on spiking neural networks”. In: *Science China Information Sciences* 59.2 (2016), pp. 1–5.

[24] E. Chicca, F. Stefanini, C. Bartolozzi, and G. Indiveri. “Neuromorphic electronic circuits for building autonomous cognitive systems”. In: *Proceedings of the IEEE* 102.9 (2014), pp. 1367–1388.

[25] A. F. Murray and A. V. Smith. “Asynchronous VLSI neural networks using pulse-stream arithmetic”. In: *IEEE Journal of Solid-State Circuits* 23.3 (1988), pp. 688–697.

[26] F. Blayo and P. Hurat. “A VLSI systolic array dedicated to Hopfield neural network”. In: *VLSI for Artificial Intelligence*. Springer, 1989, pp. 255–264.

[27] F. Salam. “A model of neural circuits for programmable VLSI implementation”. In: *IEEE International Symposium on Circuits and Systems,* IEEE. 1989, pp. 849–851.

[28] J. B. Burr. “Digital neural network implementations”. In: *Neural networks, concepts, applications, and implementations* 3 (1991), pp. 237–285.

[29] K. Madani, P. Garda, E. Belhaire, and F. Devos. “Two analog counters for neural network implementation”. In: *IEEE journal of solid-state circuits* 26.7 (1991), pp. 966–974.

[30] A. F. Murray, D. Del Corso, and L. Tarassenko. “Pulse-stream VLSI neural networks mixing analog and digital techniques”. In: *IEEE Transactions on Neural Networks* 2.2 (1991), pp. 193–204.

[31] L. Tarassenko, M. Brownlow, G. Marshall, J. Tombs, and A. Murray. “Real-time autonomous robot navigation using VLSI neural networks”. In: *Advances in neural information processing systems.* 1991, pp. 422–428.

[32] P. Hasler and L. Akers. “VLSI neural systems and circuits”. In: *Ninth Annual International Phoenix Conference on Computers and Communications. 1990 Conference Proceedings.* IEEE. 1990, pp. 31–37.

[33] J.-C. Lee and B. J. Sheu. “Parallel digital image restoration using adaptive VLSI neural chips”. In: *Proceedings., 1990 IEEE International Conference on Computer Design: VLSI in Computers and Processors.* IEEE. 1990, pp. 126–129.

[34] S. H. Jo, T. Chang, I. Ebong, B. B. Bhadviya, P. Mazumder, and W. Lu. “Nanoscale memristor device as synapse in neuromorphic systems”. In: *Nano Letters* 10.4 (2010), pp. 1297–1301.

[35] Y. Babacan and F. Kaçar. “Memristor emulator with spike-timing-dependent-plasticity”. In: *AEU-International Journal of Electronics and Communications* 73 (2017), pp. 16–22.

[36] T. Tuma, A. Pantazi, M. Le Gallo, A. Sebastian, and E. Eleftheriou. “Stochastic phase-change neurons”. In: *Nature Nanotechnology* 11.8 (2016), p. 693.

[37] F. Boi, T. Moraitis, V. De Feo, F. Diotalevi, C. Bartolozzi, G. Indiveri, and A. Vato. “A bidirectional brain-machine interface featuring a neuromorphic hardware decoder”. In: *Frontiers in Neuroscience* 10 (2016), p. 563.

[38] E. Nurse, B. S. Mashford, A. J. Yepes, I. Kiral-Kornek, S. Harrer, and D. R. Freestone. “Decoding EEG and LFP signals using deep learning: heading TrueNorth”. In: *Proceedings of the ACM International Conference on Computing Frontiers.* 2016, pp. 259–266.

[39] F. Corradi and G. Indiveri. “A neuromorphic event-based neural recording system for smart brain-machine-interfaces”. In: *IEEE Transactions on Biomedical Circuits and Systems* 9.5 (2015), pp. 699–709.

[40] Z. S. Zumsteg, C. Kemere, S. O’Driscoll, G. Santhanam, R. E. Ahmed, K. V. Shenoy, and T. H. Meng. “Power feasibility of implantable digital spike sorting circuits for neural prosthetic systems”. In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 13.3 (2005), pp. 272–279.

[41] C. Yakopcic, R. Hasan, and T. M. Taha. “Memristor based neuromorphic circuit for ex-situ training of multi-layer neural network algorithms”. In: *2015 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2015, pp. 1–7.

[42] H. Kim, M. P. Sah, C. Yang, T. Roska, and L. O. Chua. “Memristor bridge synapses”. In: *Proceedings of the IEEE* 100.6 (2011), pp. 2061–2070.

[43] S.-C. Liu and M. Oster. “Feature competition in a spike-based winner-take-all VLSI network”. In: *2006 IEEE International Symposium on Circuits and Systems*. IEEE. 2006, 4–pp.

[44] P. Knag, J. K. Kim, T. Chen, and Z. Zhang. “A sparse coding neural network ASIC with on-chip learning for feature extraction and encoding”. In: *IEEE Journal of Solid-State Circuits* 50.4 (2015), pp. 1070–1079.

[45] W.-Y. Tsai, D. R. Barch, A. S. Cassidy, M. V. DeBole, A. Andreopoulos, B. L. Jackson, M. D. Flickner, J. V. Arthur, D. S. Modha, J. Sampson, *et al.* “Always-on speech recognition using truenorth, a reconfigurable, neurosynaptic processor”. In: *IEEE Transactions on Computers* 66.6 (2016), pp. 996–1007.

[46] Q. Wang, Y. Li, and P. Li. “Liquid state machine based pattern recognition on FPGA with firing-activity dependent power gating and approximate computing”. In: *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE. 2016, pp. 361–364.

[47] S. N. Truong, S.-J. Ham, and K.-S. Min. “Neuromorphic crossbar circuit with nanoscale filamentary-switching binary memristors for speech recognition”. In: *Nanoscale Research Letters* 9.1 (2014), pp. 1–9.

[48] S. K. Esser, A. Andreopoulos, R. Appuswamy, P. Datta, D. Barch, A. Amir, J. Arthur, A. Cassidy, M. Flickner, P. Merolla, *et al.* “Cognitive computing systems: Algorithms and applications for networks of neurosynaptic cores”. In: *The 2013 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2013, pp. 1–10.

[49] J. Daly, J. Brown, and J. Weng. “Neuromorphic motivated systems”. In: *The 2011 International Joint Conference on Neural Networks*. IEEE. 2011, pp. 2917–2924.

[50] T. Yamazaki and J. Igarashi. “Realtime cerebellum: A large-scale spiking network model of the cerebellum that runs in realtime using a graphics processing unit”. In: *Neural Networks* 47 (2013), pp. 103–111.

[51] P. Arena, L. Fortuna, M. Frasca, L. Patane, and M. Pollino. “An autonomous mini-hexapod robot controlled through a CNN-based CPG VLSI chip”. In: *2006 10th International Workshop on Cellular Neural Networks and Their Applications*. IEEE. 2006, pp. 1–6.

[52] C. Denk, F. Llobet-Blandino, F. Galluppi, L. A. Plana, S. Furber, and J. Conradt. “Real-time interface board for closed-loop robotic tasks on the SpiNNaker neural computing system”. In: *International Conference on Artificial Neural Networks*. Springer. 2013, pp. 467–474.

[53] W. Gerstner, W. M. Kistler, R. Naud, and L. Paninski. *Neuronal dynamics: From single neurons to networks and models of cognition*. Cambridge University Press, 2014.

[54] H. Paugam-Moisy and S. M. Bohte. “Computing with spiking neuron networks.” In: *Handbook of Natural Computing* 1 (2012), pp. 1–47.

[55] S. Ghosh-Dastidar and H. Adeli. “Spiking neural networks”. In: *International Journal of Neural Systems* 19.04 (2009), pp. 295–308.

[56] B. Yegnanarayana. *Artificial neural networks.* PHI Learning Pvt. Ltd., 2009.

[57] W. S. McCulloch and W. Pitts. “A logical calculus of the ideas immanent in nervous activity”. In: *The bulletin of mathematical biophysics* 5.4 (1943), pp. 115–133.

[58] A. N. Burkitt. “A review of the integrate-and-fire neuron model: I. Homogeneous synaptic input”. In: *Biological cybernetics* 95.1 (2006), pp. 1–19.

[59] E. M. Izhikevich. “Simple model of spiking neurons”. In: *IEEE Transactions on Neural Networks* 14.6 (2003), pp. 1569–1572.

[60] S. Binczak, S. Jacquir, J.-M. Bilbault, V. B. Kazantsev, and V. I. Nekorkin. “Experimental study of electrical FitzHugh–Nagumo neurons with modified excitability”. In: *Neural Networks* 19.5 (2006), pp. 684–693.

[61] M. Storace, D. Linaro, and E. de Lange. “The Hindmarsh–Rose neuron model: bifurcation analysis and piecewise-linear approximations”. In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 18.3 (2008), p. 033128.

[62] A. L. Hodgkin and A. F. Huxley. “A quantitative description of membrane current and its application to conduction and excitation in nerve”. In: *The Journal of physiology* 117.4 (1952), pp. 500–544.

[63] H. Lecar. “Morris-lecar model”. In: *Scholarpedia* 2.10 (2007), p. 1333.

[64] L. F. Abbott and S. B. Nelson. “Synaptic plasticity: taming the beast”. In: *Nature Neuroscience* 3.11 (2000), pp. 1178–1183.

[65] S. J. Martin, P. D. Grimwood, and R. G. Morris. “Synaptic plasticity and memory: an evaluation of the hypothesis”. In: *Annual Review of Neuroscience* 23.1 (2000), pp. 649–711.

[66] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams. “The missing memristor found”. In: *Nature* 453.7191 (2008), pp. 80–83.

[67] H.-S. P. Wong, S. Raoux, S. Kim, J. Liang, J. P. Reifenberg, B. Rajendran, M. Asheghi, and K. E. Goodson. “Phase change memory”. In: *Proceedings of the IEEE* 98.12 (2010), pp. 2201–2227.

[68] A. K. Geim and K. S. Novoselov. “The rise of graphene”. In: *Nanoscience and technology: a collection of reviews from nature journals.* World Scientific, 2010, pp. 11–19.

[69] C. Huan, Y. Wang, X. Cui, and X. Zhang. “A reconfigurable mixed signal CMOS design for multiple STDP learning rules”. In: *IEEE 3rd International Conference on Electronics Technology (ICET).* 2020, pp. 639–643.

[70] S. B. Furber, D. R. Lester, L. A. Plana, J. D. Garside, E. Painkras, S. Temple, and A. D. Brown. “Overview of the spinnaker system architecture”. In: *IEEE Transactions on Computers* 62.12 (2012), pp. 2454–2467.

[71] Y. Babacan, F. Kaçar, and K. Gürkan. "A spiking and bursting neuron circuit based on memristor". In: *Neurocomputing* 203 (2016), pp. 86–91.

[72] B. Linares-Barranco and T. Serrano-Gotarredona. "Memristance can explain spike-time-dependent-plasticity in neural synapses". In: *Nature Precedings* (2009), pp. 1–1.

[73] S. Ambrogio, N. Ciocchini, M. Laudato, V. Milo, A. Pirovano, P. Fantini, and D. Ielmini. "Unsupervised learning by spike timing dependent plasticity in phase change memory (PCM) synapses". In: *Frontiers in Neuroscience* 10 (2016), p. 56.

[74] K. D. Cantley, A. Subramaniam, H. J. Stiegler, R. A. Chapman, and E. M. Vogel. "Hebbian learning in spiking neural networks with nanocrystalline silicon TFTs and memristive synapses". In: *IEEE Transactions on Nanotechnology* 10.5 (2011), pp. 1066–1073.

[75] M. Chu, B. Kim, S. Park, H. Hwang, M. Jeon, B. H. Lee, and B.-G. Lee. "Neuromorphic hardware system for visual pattern recognition with memristor array and CMOS neuron". In: *IEEE Transactions on Industrial Electronics* 62.4 (2014), pp. 2410–2419.

[76] Y. Kim, Y. Zhang, and P. Li. "A reconfigurable digital neuromorphic processor with memristive synaptic crossbar for cognitive computing". In: *ACM Journal on Emerging Technologies in Computing Systems (JETC)* 11.4 (2015), pp. 1–25.

[77] D. Chabi, Z. Wang, C. Bennett, J.-O. Klein, and W. Zhao. "Ultrahigh density memristor neural crossbar for on-chip supervised learning". In: *IEEE Transactions on Nanotechnology* 14.6 (2015), pp. 954–962.

[78] N. Zheng and P. Mazumder. "Learning in memristor crossbar-based spiking neural networks through modulation of weight-dependent spike-timing-dependent plasticity". In: *IEEE Transactions on Nanotechnology* 17.3 (2018), pp. 520–532.

[79] P. Avouris and C. Dimitrakopoulos. "Graphene: synthesis and applications". In: *Materials Today* 15.3 (2012), pp. 86–97.

[80] M. J. Allen, V. C. Tung, and R. B. Kaner. "Honeycomb carbon: a review of graphene". In: *Chemical Reviews* 110.1 (2010), pp. 132–145.

[81] A. C. Ferrari, F. Bonaccorso, V. Fal'Ko, K. S. Novoselov, S. Roche, P. Bøggild, S. Borini, F. H. Koppens, V. Palermo, N. Pugno, *et al.* "Science and technology roadmap for graphene, related two-dimensional crystals, and hybrid systems". In: *Nanoscale* 7.11 (2015), pp. 4598–4810.

[82] M. J. Allen, V. C. Tung, and R. B. Kaner. "Honeycomb carbon: a review of graphene". In: *Chemical Reviews* 110.1 (2009), pp. 132–145.

[83] K. Matsumoto. *Frontiers of graphene and carbon nanotubes: devices and applications*. Springer, 2015.

[84] W. Han, R. K. Kawakami, M. Gmitra, and J. Fabian. "Graphene spintronics". In: *Nature nanotechnology* 9.10 (2014), pp. 794–807.

[85] A. Candini, S. Klyatskaya, M. Ruben, W. Wernsdorfer, and M. Affronte. "Graphene spintronic devices with molecular nanomagnets". In: *Nano letters* 11.7 (2011), pp. 2634–2639.

[86] E. W. Hill, A. Vijayaragahvan, and K. Novoselov. “Graphene sensors”. In: *IEEE Sensors Journal* 11.12 (2011), pp. 3161–3170.

[87] Y. Liu, X. Dong, and P. Chen. “Biological and chemical sensors based on graphene materials”. In: *Chemical Society Reviews* 41.6 (2012), pp. 2283–2307.

[88] H. J. Yoon, J. H. Yang, Z. Zhou, S. S. Yang, M. M.-C. Cheng, *et al.* “Carbon dioxide gas sensor using a graphene sheet”. In: *Sensors and Actuators B: Chemical* 157.1 (2011), pp. 310–313.

[89] H. Shen, L. Zhang, M. Liu, and Z. Zhang. “Biomedical applications of graphene”. In: *Theranostics* 2.3 (2012), p. 283.

[90] C. Chung, Y.-K. Kim, D. Shin, S.-R. Ryoo, B. H. Hong, and D.-H. Min. “Biomedical applications of graphene and graphene oxide”. In: *Accounts of chemical research* 46.10 (2013), pp. 2211–2224.

[91] G. Reina, J. M. González-Domí nguez, A. Criado, E. Vázquez, A. Bianco, and M. Prato. “Promises, facts and challenges for graphene in biomedical applications”. In: *Chemical Society Reviews* 46.15 (2017), pp. 4400–4416.

[92] Y.-M. Lin, A. Valdes-Garcia, S.-J. Han, D. B. Farmer, I. Meric, Y. Sun, Y. Wu, C. Dimitrakopoulos, A. Grill, P. Avouris, *et al.* “Wafer-scale graphene integrated circuit”. In: *Science* 332.6035 (2011), pp. 1294–1297.

[93] D. A. Areshkin and C. T. White. “Building blocks for integrated graphene circuits”. In: *Nano Letters* 7.11 (2007), pp. 3253–3259.

[94] S. Lee, K. Lee, C.-H. Liu, G. S. Kulkarni, and Z. Zhong. “Flexible and transparent all-graphene circuits for quaternary digital modulations”. In: *Nature communications* 3.1 (2012), pp. 1–7.

[95] R. Sordan, F. Traversi, and V. Russo. “Logic gates with a single graphene transistor”. In: *Applied Physics Letters* 94.7 (2009), p. 51.

[96] Y. Jiang, N. C. Laurenciu, H. Wang, and S. D. Cotofana. “Graphene nanoribbon based complementary logic gates and circuits”. In: *IEEE Transactions on Nanotechnology* 18 (2019), pp. 287–298.

[97] G. Liu, S. Ahsan, A. G. Khitun, R. K. Lake, and A. A. Balandin. “Graphene-based non-Boolean logic circuits”. In: *Journal of Applied Physics* 114.15 (2013), p. 154310.

[98] S. Moysidis, I. G. Karafyllidis, and P. Dimitrakis. “Graphene logic gates”. In: *IEEE Transactions on Nanotechnology* 17.4 (2018), pp. 852–859.

[99] S. Cotofana, P. Dimitrakis, M. Enachescu, I. Karafyllidis, A. Rubio, and G. C. Sirakoulis. “On Graphene Nanoribbon-based Nanoelectronic Circuits Viability”. In: *The 42nd Workshop on Compound Semiconductor Devices and Integrated Circuits held in Europe.* 2018, pp. 35–36.

[100] H. Tian, W. Mi, X.-F. Wang, H. Zhao, Q.-Y. Xie, C. Li, Y.-X. Li, Y. Yang, and T.-L. Ren. “Graphene dynamic synapse with modulatable plasticity”. In: *Nano Letters* 15.12 (2015), pp. 8013–8019.

[101] M. T. Sharbati, Y. Du, J. Torres, N. D. Ardolino, M. Yun, and F. Xiong. “Low-Power, electrochemically tunable graphene synapses for neuromorphic computing”. In: *Advanced Materials* 30.36 (2018), p. 1802353.

[102] S. Datta. *Quantum transport: atom to transistor*. Cambridge university press, 2005.

[103] Y. Jiang, N. C. Laurenciu, and S. Cotofana. “Non-equilibrium green function-based Verilog-A graphene nanoribbon model”. In: *2018 IEEE 18th International Conference on Nanotechnology (IEEE-NANO)*. IEEE. 2018, pp. 1–4.

[104] H. Wang, Y. Wu, C. Cong, J. Shang, and T. Yu. “Hysteresis of electronic transport in graphene transistors”. In: *ACS Nano* 4.12 (2010), pp. 7221–7228.

[105] M. Aliofkhazraei, N. Ali, W. I. Milne, C. S. Ozkan, S. Mitura, and J. L. Gervasoni. *Graphene science handbook: Size-dependent properties*. CRC Press, 2016.

[106] P. Joshi, H. Romero, A. Neal, V. Toutam, and S. Tadigadapa. “Intrinsic doping and gate hysteresis in graphene field effect devices fabricated on SiO2 substrates”. In: *Journal of Physics: Condensed Matter* 22.33 (2010), p. 334214.

[107] M. Bonmann, A. Vorobiev, J. Stake, and O. Engström. “Effect of oxide traps on channel transport characteristics in graphene field effect transistors”. In: *Journal of Vacuum Science & Technology B, Nanotechnology and Microelectronics: Materials, Processing, Measurement, and Phenomena* 35.1 (2017), 01A115.

[108] Y. G. Lee, C. G. Kang, U. J. Jung, J. J. Kim, H. J. Hwang, H.-J. Chung, S. Seo, R. Choi, and B. H. Lee. “Fast transient charging at the graphene/SiO 2 interface causing hysteretic device characteristics”. In: *Applied Physics Letters* 98.18 (2011), p. 183508.

[109] F. Najam, K. C. Lau, C. S. Lim, Y. S. Yu, and M. L. P. Tan. “Metal oxide-graphene field-effect transistor: interface trap density extraction model”. In: *Beilstein Journal of Nanotechnology* 7.1 (2016), pp. 1368–1376.

[110] M. Winters, E. Sveinbjörnsson, and N. Rorsman. “Hysteresis modeling in graphene field effect transistors”. In: *Journal of Applied Physics* 117.7 (2015), p. 074501.

[111] I. G. Karafyllidis. “Current switching in graphene quantum point contacts”. In: *IEEE Transactions on Nanotechnology* 13.4 (2014), pp. 820–824.

[112] P. I. Hagouel and I. G. Karafyllidis. “Nanoelectronic graphene devices”. In: *2017 IEEE 30th International Conference on Microelectronics (MIEL)*. IEEE. 2017, pp. 63–66.

[113] U. Jung, Y. J. Kim, Y. Kim, Y. G. Lee, and B. H. Lee. “Extraction of the interface state density of top-gate graphene field-effect transistors”. In: *IEEE Electron Device Letters* 36.4 (2015), pp. 408–410.

[114] J. R. Nagel *et al.* “Solving the generalized Poisson equation using the finite-difference method (FDM)”. In: *Lecture Notes, Dept. of Electrical and Computer Engineering, University of Utah* (2011).

[115] E. M. Godfrin. “A method to compute the inverse of an n-block tridiagonal quasi-Hermitian matrix”. In: *Journal of Physics: Condensed Matter* 3.40 (1991), p. 7843.

[116] G. I. Zebrev, E. V. Melnik, and D. K. Batmanova. “Using capacitance methods for interface trap level density extraction in graphene field-effect devices”. In: *2012 28th International Conference on Microelectronics Proceedings*. IEEE. 2012, pp. 335–338.

[117] A. Citri and R. C. Malenka. “Synaptic plasticity: multiple forms, functions, and mechanisms”. In: *Neuropsychopharmacology* 33.1 (2008), pp. 18–41.

[118] W. Gerstner and W. M. Kistler. *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge University Press, 2002.

[119] Y. Dan and M.-m. Poo. “Spike timing-dependent plasticity of neural circuits”. In: *Neuron* 44.1 (2004), pp. 23–30.

[120] M. F. Bear and R. C. Malenka. “Synaptic plasticity: LTP and LTD”. In: *Current Opinion in Neurobiology* 4.3 (1994), pp. 389–399.

[121] S. B. Furber, D. R. Lester, L. A. Plana, J. D. Garside, E. Painkras, S. Temple, and A. D. Brown. “Overview of the spinnaker system architecture”. In: *IEEE Transactions on Computers* 62.12 (2013), pp. 2454–2467.

[122] P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, A. S. Cassidy, J. Sawada, F. Akopyan, B. L. Jackson, N. Imam, C. Guo, Y. Nakamura, *et al.* “A million spiking-neuron integrated circuit with a scalable communication network and interface”. In: *Science* 345.6197 (2014), pp. 668–673.

[123] S. Saïghi, C. G. Mayr, T. Serrano-Gotarredona, H. Schmidt, G. Lecerf, J. Tomas, J. Grollier, S. Boyn, A. F. Vincent, D. Querlioz, *et al.* “Plasticity in memristive devices for spiking neural networks”. In: *Frontiers in Neuroscience* 9 (2015), p. 51.

[124] T. Fardet, S. B. Vennemo, J. Mitchell, H. Mørk, S. Graber, J. Hahne, S. Spreizer, R. Deepu, G. Trensch, P. Weidel, J. Jordan, J. M. Eppler, D. Terhorst, A. Morrison, C. Linssen, A. Antonietti, K. Dai, A. Serenko, B. Cai, P. Kubaj, R. Gutzen, H. Jiang, I. Kitayama, B. Jürgens, and H. E. Plesser. *NEST 2.20.0*. Jan. 2020. DOI: 10.5281/zenodo.3605514.

[125] N. Caporale and Y. Dan. “Spike timing–dependent plasticity: a Hebbian learning rule”. In: *Annual Review of Neuroscience* 31 (2008), pp. 25–46.

[126] G.-q. Bi and M.-m. Poo. “Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type”. In: *Journal of Neuroscience* 18.24 (1998), pp. 10464–10472.

[127] S. Datta. “Nanoscale device modeling: the Green’s function method”. In: *Superlattices and Microstructures* 28.4 (2000), pp. 253–278.

[128] H. Wang, N. C. Laurenciu, Y. Jiang, and S. D. Cotofana. “Atomistic-level hysteresis-aware graphene structures electron transport model”. In: *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE. 2019, pp. 1–5.

[129] D. E. Feldman. “The spike-timing dependence of plasticity”. In: *Neuron* 75.4 (2012), pp. 556–571.

[130] R. C. Froemke, M.-m. Poo, and Y. Dan. “Spike-timing-dependent synaptic plasticity depends on dendritic location”. In: *Nature* 434.7030 (2005), p. 221.

[131] C. Hartmann, D. C. Miner, and J. Triesch. “Precise synaptic efficacy alignment suggests potentiation dominated learning”. In: *Frontiers in Neural Circuits* 9 (2016), p. 90.

[132] W. Huang, Y.-W. Fang, Y. Yin, B. Tian, W. Zhao, C. Hou, C. Ma, Q. Li, E. Y. Tsymbal, C.-G. Duan, *et al.* “Solid-state synapse based on magnetoelectrically coupled memristor”. In: *ACS Applied Materials & Interfaces* 10.6 (2018), pp. 5649–5656.

[133] Z. Wang, S. Joshi, S. E. Savel’ev, H. Jiang, R. Midya, P. Lin, M. Hu, N. Ge, J. P. Strachan, Z. Li, *et al.* “Memristors with diffusive dynamics as synaptic emulators for neuromorphic computing”. In: *Nature Materials* 16.1 (2017), pp. 101–108.

[134] C. Mayr, J. Partzsch, M. Noack, S. Hänzsche, S. Scholze, S. Höppner, G. Ellguth, and R. Schüffny. “A biological-realtime neuromorphic system in 28 nm CMOS using low-leakage switched capacitor circuits”. In: *IEEE Transactions on Biomedical Circuits and Systems* 10.1 (2015), pp. 243–254.

[135] T. Serrano-Gotarredona, T. Masquelier, T. Prodromakis, G. Indiveri, and B. Linares-Barranco. “STDP and STDP variations with memristors for spiking neuromorphic learning systems”. In: *Frontiers in Neuroscience* 7 (2013), p. 2.

[136] J. H. Wijekoon and P. Dudek. “A CMOS circuit implementation of a spiking neuron with bursting and adaptation on a biological timescale”. In: *2009 IEEE Biomedical Circuits and Systems Conference*. IEEE. 2009, pp. 193–196.

[137] X. Wu, V. Saxena, K. Zhu, and S. Balagopal. “A CMOS spiking neuron for brain-inspired neural networks with resistive synapses and in situ learning”. In: *IEEE Transactions on Circuits and Systems II: Express Briefs* 62.11 (2015), pp. 1088–1092.

[138] M. Al-Shedivat, R. Naous, E. Neftci, G. Cauwenberghs, and K. N. Salama. “Inherently stochastic spiking neurons for probabilistic neural computation”. In: *2015 7th International IEEE/EMBS Conference on Neural Engineering (NER)*. IEEE. 2015, pp. 356–359.

[139] C. D. Wright, P. Hosseini, and J. A. V. Diosdado. “Beyond von-Neumann computing with nanoscale phase-change memory devices”. In: *Advanced Functional Materials* 23.18 (2013), pp. 2248–2254.

[140] G. Kalon, Y. Jun Shin, V. Giang Truong, A. Kalitsov, and H. Yang. “The role of charge traps in inducing hysteresis: Capacitance–voltage measurements on top gated bilayer graphene”. In: *Applied Physics Letters* 99.8 (2011), p. 083109.

[141] *Synopsys. [Online]. Available: https://www.synopsys.com/verification/ams-verification /hspice.html/.*

[142] *Cadence. [Online]. Available: https://www.cadence.com/.*

[143] H. Wang, N. C. Laurenciu, Y. Jiang, and S. D. Cotofana. “Graphene nanoribbon-based synapses with versatile plasticity”. In: *2019 IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH)*. IEEE. 2019, pp. 1–6.

[144] H. Wang, N. C. Laurenciu, Y. Jiang, and S. Cotofana. “Ultra-Compact, Entirely Graphene-based Nonlinear Leaky Integrate-and-Fire Spiking Neuron”. In: *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE. 2020, pp. 1–5.

[145] H. Wang, N. Cucu Laurenciu, Y. Jiang, and S. D. Cotofana. “Compact Graphene-Based Spiking Neural Network With Unsupervised Learning Capabilities”. In: *IEEE Open Journal of Nanotechnology* 1 (2020), pp. 135–144.

[146] R. C. Gonzalez and R. E. Woods. *Digital image processing, 4th edition.* Pearson, 2018.

[147] A. Gupta and L. N. Long. “Character recognition using spiking neural networks”. In: *2007 International Joint Conference on Neural Networks.* IEEE. 2007, pp. 53–58.

[148] D. Lu, X.-H. Yu, X. Jin, B. Li, Q. Chen, and J. Zhu. “Neural network based edge detection for automated medical diagnosis”. In: *2011 IEEE International Conference on Information and Automation.* IEEE. 2011, pp. 343–348.

[149] G. Shrivakshan and C. Chandrasekar. “A comparison of various edge detection techniques used in image processing”. In: *International Journal of Computer Science Issues (IJCSI)* 9.5 (2012), p. 269.

[150] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. “Image quality assessment: from error visibility to structural similarity”. In: *IEEE Transactions on Image Processing* 13.4 (2004), pp. 600–612.

[151] S. Szczesny and D. Huderek. “60 pW 20um size CMOS implementation of an actual soma membrane”. In: *Journal of Computational Electronics* 19.1 (2020), pp. 242–252.

[152] I. Sourikopoulos, S. Hedayat, C. Loyez, F. Danneville, V. Hoel, E. Mercier, and A. Cappy. “A 4-fJ/spike artificial neuron in 65 nm CMOS technology”. In: *Frontiers in Neuroscience* 11 (2017), p. 123.

[153] V. Ostwal, R. Meshram, B. Rajendran, and U. Ganguly. “An ultra-compact and low power neuron based on SOI platform”. In: *2015 International Symposium on VLSI Technology, Systems and Applications.* IEEE. 2015, pp. 1–2.

[154] S. Moradi, S. A. Bhave, and R. Manohar. “Energy-efficient hybrid CMOS-NEMS LIF neuron circuit in 28 nm CMOS process”. In: *2017 IEEE Symposium Series on Computational Intelligence (SSCI).* IEEE. 2017, pp. 1–5.

[155] Y. Zhang, W. He, Y. Wu, K. Huang, Y. Shen, J. Su, Y. Wang, Z. Zhang, X. Ji, G. Li, *et al.* “Highly compact artificial memristive neuron with low energy consumption”. In: *Small* 14.51 (2018), p. 1802188.

[156] D. Kaushik, U. Singh, U. Sahu, I. Sreedevi, and D. Bhowmik. “Comparing domain wall synapse with other non volatile memory devices for on-chip learning in analog hardware neural network”. In: *AIP Advances* 10.2 (2020), p. 025111.

LIST OF PUBLICATIONS

Publications related to the thesis

International Journal Papers

1. **H. Wang**, N. Cucu Laurenciu, and S.D. Cotofana, "A Reconfigurable Graphene-based Spiking Neural Network Architecture", *IEEE Open Journal of Nanotechnology (OJ-NANO)*, vol. 2, pp. 59-71, 2021.

2. **H. Wang**, N. Cucu Laurenciu, Y. Jiang, and S.D. Cotofana, "Graphene-based Artificial Synapses with Tunable Plasticity", *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, vol. 17, pp. 1-21, 2021.

3. **H. Wang**, N. Cucu Laurenciu, Y. Jiang, and S.D. Cotofana, "Compact Graphene-Based Spiking Neural Network With Unsupervised Learning Capabilities", *IEEE Open Journal of Nanotechnology (OJ-NANO)*, vol. 1, pp. 135-144, 2020.

International Conference Proceedings

1. **H. Wang**, N. Cucu Laurenciu, Y. Jiang, and S.D. Cotofana, "Ultra-compact, Entirely Graphene-based Non-linear Leaky Integrate-and-Fire Spiking Neuron", *IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1-5, 2020.

2. **H. Wang**, N. Cucu Laurenciu, Y. Jiang, and S.D. Cotofana, "Graphene Nanoribbon-Based Synapses with Versatile Plasticity", *IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH)*, pp. 1-6, 2019.

3. **H. Wang**, N. Cucu Laurenciu, Y. Jiang, and S.D. Cotofana, "Atomistic-level Hysteresis-aware Graphene Structures Electron Transport Model", *IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1-5, 2019.

Other publications

International Journal Papers

1. Y. Jiang, N. Cucu Laurenciu, **H. Wang**, and S.D. Cotofana, "Graphene Nanoribbon Based Complementary Logic Gates and Circuits", *IEEE Transactions on Nanotechnology (TNANO)*, vol. 18, p. 287-298, 2019.

International Conference Proceedings

1. Y. Jiang, N. Cucu Laurenciu, **H. Wang**, and S.D. Cotofana, "A Study of Graphene Nanoribbon-based Gate Performance Robustness under Temperature Variations", *IEEE 20th International Conference on Nanotechnology (IEEE-NANO)*, pp. 62-66, 2020.

CURRICULUM VITÆ

He WANG

H. Wang was born 1992 in Henan, China. He received his B.Sc. degree in software engineering in 2014 from Central South University, Changsha, China. After that he received M.Sc degree in computer science and technology in 2017 from National University of Defence Technology, Changsha, China. In October 2017, he joined the Department of Quantum and Computer Engineering at the Faculty of Electrical Engineering, Mathematics, Computer Science in Delft University of Technology to pursue his Ph.D. His research interests include graphene-based computing and neuromorphic computing.