Long-term neural activity recorders using energy-based sensing, compressive computation and data logging

Darshit Mehta
Washington University in St. Louis

Follow this and additional works at: https://openscholarship.wustl.edu/eng_etds

Part of the Biomedical Engineering and Bioengineering Commons, Electrical and Electronics Commons, and the Neuroscience and Neurobiology Commons

Recommended Citation

This Dissertation is brought to you for free and open access by the McKelvey School of Engineering at Washington University Open Scholarship. It has been accepted for inclusion in McKelvey School of Engineering Theses & Dissertations by an authorized administrator of Washington University Open Scholarship. For more information, please contact digital@wumail.wustl.edu.
Long-term Neural Activity Recorders Using Energy-based Sensing, Compressive Computation and Data Logging

by

Darshit Mehta

A dissertation presented to the Graduate School of Washington University in partial fulfillment of the requirements for the degree of Doctor of Philosophy

August 2021
Saint Louis, Missouri
copyright by

Darshit Mehta

2021
Contents

List of Tables ................................................................. v
List of Figures ................................................................. vi
Acknowledgments ............................................................ viii
Abstract ................................................................. xi

1 Introduction ................................................................. 1

2 Energy-based sensing for explosive detection .......................... 8
  2.1 Surgical procedure ..................................................... 10
  2.2 Estimation of energy in the neural signal ................................ 10
  2.3 Classification analysis ................................................ 12
  2.4 Wisdom of the swarm ............................................... 13
  2.5 Rapid recognition of the target chemicals ............................ 16
  2.6 Real time classifier .................................................. 17

3 Compressive sensing ......................................................... 20
  3.1 Introduction ......................................................... 20
  3.2 Background and Principle of Operation ............................... 24
  3.3 SoC Implementation .................................................. 26
    3.3.1 Linear Injector circuit ...................................... 28
    3.3.2 Linear Feedback Shift Register ............................... 29
  3.4 Experimental results ................................................ 31
  3.5 Conclusions and Discussions ..................................... 33
  3.6 Application of CS for neural recording ............................. 34

4 Fowler-Nordheim Tunneling based Dynamical Memory (FN-MEM) .... 39
  4.1 Introduction .......................................................... 41
  4.2 Results ............................................................... 43
    4.2.1 Long-term reliable synchronized dynamical systems .............. 43
    4.2.2 A simple behavioral model explains the data-logging principle .. 46
    4.2.3 Energy budget, sensing and retention limits ...................... 49
    4.2.4 FN memory write energy ........................................ 53
# Table of Contents

4.2.5 Parametric analysis ...................................................... 54
4.2.6 Impedance analysis and power estimation .......................... 54
4.3 Methods ................................................................. 57
   4.3.1 Programming and synchronization ............................... 57
   4.3.2 One-time programming ........................................... 58
   4.3.3 Calibration .......................................................... 59
   4.3.4 Initialization ...................................................... 59
   4.3.5 Model derivation .................................................. 60

5 FN Dynamic Adaptive Memory ............................................. 64
   5.1 Introduction .......................................................... 65
   5.2 Results ................................................................... 71
      5.2.1 Dynamic analog memory with an asymptotic nonvolatile storage ... 71
      5.2.2 Characterization of FN-DAM ................................... 73
      5.2.3 FN-DAM based Co-design of Classifiers and Neural Networks ....... 76
   5.3 Discussions ............................................................. 78
   5.4 Methods ................................................................. 82
      5.4.1 FN-DAM based CNN Implementation ........................... 87

6 Self-powered Analog Sensor Logger ........................................ 88
   6.1 Introduction ............................................................ 89
   6.2 Results ................................................................... 91
      6.2.1 A simple behavioral model explains the data-logging principle .... 91
      6.2.2 Model validation .................................................. 93
      6.2.3 Self-powered operation of the proposed device .................... 96
      6.2.4 Self-powered sensing of action due to ambient acceleration ....... 96
      6.2.5 Parametric analysis .............................................. 98
      6.2.6 Temporal dependence ............................................ 100
      6.2.7 Simulation of intracellular neural recording ...................... 101
   6.3 Discussion ............................................................... 102

7 System Integration ............................................................... 108
   7.1 Analysis Framework .................................................... 110
      7.1.1 Energy sources .................................................... 111
      7.1.2 Amplifier .......................................................... 111
      7.1.3 Energy extraction circuit ....................................... 112
      7.1.4 Analog to Digital Converter .................................... 113
      7.1.5 Classifier .......................................................... 113
      7.1.6 Compressive sensing module ................................... 114
      7.1.7 FN-MEM .......................................................... 115
      7.1.8 Wireless transmission ............................................ 117
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Specifications</td>
<td>33</td>
</tr>
<tr>
<td>5.1</td>
<td>Specifications</td>
<td>78</td>
</tr>
<tr>
<td>7.1</td>
<td>Parameters</td>
<td>120</td>
</tr>
<tr>
<td>7.2</td>
<td>Comparison</td>
<td>128</td>
</tr>
<tr>
<td>A.1</td>
<td>Odor Panel</td>
<td>132</td>
</tr>
<tr>
<td>A.2</td>
<td>Device Parameters</td>
<td>134</td>
</tr>
<tr>
<td>B.1</td>
<td>Design Specifications</td>
<td>148</td>
</tr>
</tbody>
</table>
List of Figures

1.1 Neural recording paradigms ................................................. 5
2.1 Signal energy calculation .................................................. 11
2.2 Odor classification .......................................................... 14
2.3 Rapid crowd-sourced classification ....................................... 15
2.4 Rapid crowd-sourced classification ....................................... 16
2.5 Real time linear classifier .................................................. 18
3.1 Signal flow diagram showing the use of the proposed compressive sensing-storage ................................................. 22
3.2 Architecture of the compressive sensing-storage ........................ 23
3.3 Chip micrograph .............................................................. 26
3.4 Linearity of linear injection circuits ....................................... 27
3.5 Random matrix characterization .......................................... 30
3.6 Experimental results ........................................................ 32
3.7 Signal reconstruction after 4x compression ............................. 35
3.8 Classification performance after compressive sensing ................. 36
3.9 Compressive sensing implemented for a real time classifier .......... 37
4.1 Operating principle and architecture of the proposed memory device. .................................................. 40
4.2 Dynamic memory implemented via leakage mechanisms .............. 42
4.3 Differential FN dynamical device and its measured response. .... 44
4.4 Rectifying response of the sensor-data-logger device. ............... 47
4.5 Data retention capacity ...................................................... 52
4.6 FN Memory write energy .................................................... 53
4.7 Parametric analysis .......................................................... 55
4.8 Impedance analysis and power estimation ................................ 56
4.9 Programming and synchronization ....................................... 58
5.1 Operating principle and architecture of the proposed dynamically adaptive memory. .................................................. 68
5.2 FN-DAM adaptive response ................................................ 70
5.3 FN Memory retention time .................................................. 72
5.4 FN-DAM controlled programming ......................................... 74
5.5 FN-DAM Device characterization ......................................... 75
5.6 Neural network training with FN-DAM .................................. 77
6.1 ODE solver vs action model for small signals ......................... 94
6.2 ODE solver vs action model for large signals ......................... 95
6.3 Action tracks energy of the signal ...................................... 95
6.4 Verification of the proposed device for self-powered sensing and data-logging. 97
6.5 Self-powered sensing and data-logging of mechanical acceleration. .... 99
6.6 Temporal dependence of recorder ...................................... 101
6.7 Self-powered neural recording ........................................... 102
6.8 Ensemble response ....................................................... 103

7.1 Neural recording architectures ........................................... 109
7.2 Dynamic memory cell ..................................................... 117
7.3 Integrated sensor logger .................................................. 119
7.4 Power breakdown ......................................................... 121
7.5 Battery and memory constraints ........................................ 122
7.6 Operational lifetime due to compressive sensing ....................... 124
7.7 Logging vs Wireless ..................................................... 125
7.8 Operational lifetime vs information throughput ....................... 127

A.1 PID data ........................................................................ 133
A.2 Drift correction ............................................................. 135
A.3 Data outlier ........................................................................ 136
A.4 FN Memory read energy ................................................... 137

B.1 Locust with a backpack .................................................... 140
B.2 Locust backpack design .................................................... 150
B.3 Backpack power consumption ........................................... 151
B.4 Neural data recording ...................................................... 152
B.5 Filtered RMS response ..................................................... 153
Acknowledgments

I would like to thank Washington University in St. Louis for creating a world-class research environment and allowing us to carry our research in the most friction-less manner possible. I have only had positive experiences with everyone who works here; from BME/ESE staff, to OISS, to the Liberman Graduate Center and the entire WashU faculty.

I would like to thank the Office of Naval Research for funding a truly unique project and National Institutes of Health for seeing the value in pushing the limits of neural technologies. I would like to thank my advisors Prof. Shantanu Chakrabortty and Prof. Barani Raman for giving me guidance and support.

I would like to thank members of the AIM Lab including Kenji Aono, Liang Zhou, Oindrila Chatterjee, Ahana Gangopadhyay, Sri Harsha Kondapalli, Yarub Alazzawi, Owen Pochettino, Mingquan Yuan, Brittany Scheid and Mustafizur Rahman, as well as all of my collaborators from the Raman Lab for being great colleagues and friends.

I would like to thank my parents for their unconditional love and support as I pursued my studies on the other side of the world. They are not only my parents from whom I have learnt so much; but without doubt, my best friends as well. A special shoutout to Maggi, the tiniest member of our family, who kept me grounded through her sheer innocence. To all my friends in St. Louis, thank you! From float trips to cha-cha-over-chai, you even managed to make the lockdown one of the best times of my life! And finally, I would like to thank
Apoorva, the love of my life, for making everything so much fun and wonderful. We loved calling St. Louis our home and absolutely cherish the memories we have made here. She was my support and my companion in this long and amazing journey. Thank you Apoorva and I can’t wait to begin our next adventure!

Darshit Mehta

Washington University in Saint Louis
August 2021
Dedicated to my family.
Insects are ideal candidates for developing bio-robotic systems owing to their ability to thrive in almost any environment. For example, neurons in their exquisite olfactory sensory systems can be tapped to create a sensing platform for standoff chemical monitoring. However, for enabling such cyborg systems, it is vital that the neural activity of a freely behaving organism can be measured for long periods of time. The current state-of-the-art neural recording techniques are power-intensive and they either need batteries, which make them too bulky for insects, or they have to maintain a continuous telemetry link to an external power source which restricts the mobility of the organism. In this dissertation, I explore algorithmic and device based approaches for overcoming the limitation of current neural activity recording-technologies in terms of energy efficiency and longevity.

In the first part of my thesis I show that changes in energy content of the extracellular neural recordings contain sufficient information for specific stimulus identification tasks. Measuring the energy content of the signal can be achieved using sampling rates that are significantly
lower than what is required for spike-sorting and hence could lead to significant improvements in energy-efficiencies. This hypothesis was investigated and verified using the locust olfactory system for odor identification. We could reliably classify odors based on energy information collected from the extracellular medium in the locust antennal lobe.

To eliminate the high power consumption of long range wireless communication, I investigated energy-efficient approaches for storing data in non-volatile memory. I implemented ultra-low-power compressive sensing-storage circuits that can efficiently store a compressed signal in floating-gate memory cells. The compressed information could be retrieved later and the events of interest reconstructed offline. I further reduced the power consumption of the memory cells themselves. By modulating the trajectory of synchronized dynamical systems, I show that information can be stored by expending less than 1pj of energy per bit, a four orders of magnitude improvement. These dynamical systems are based on the physics of quantum-mechanical tunneling of electrons through thin-oxide layers. Because these devices are thermodynamically driven, the recorder based on this principle can operate at femto-watt power levels.

Finally, I investigate the feasibility of a completely self-powered neural recording paradigm in which the neuronal action-potentials modulate the dynamical memory cells directly. Through modeling studies, I show that modulation of the quantum-tunneling barrier by a neural activity can be registered as a change in the response of the self-powered dynamical system. However, the signals are smaller than the detection limits of current readout circuits. I propose techniques for recovering the information and reconstructing the time history of the input signal.
Chapter 1

Introduction

Millions of years of evolutionary forces have enabled biological organisms to develop exquisite mechanisms to survive and thrive in their environments. Among multicellular terrestrial animals, insects have distinguished themselves as being the most successful group on the planet with more than 5 million extant species [51]. Insects can sense chemicals [59], sound [155], and light [15] among other things [94,95]. They are able to move with energy efficiencies surpassing similar-sized robotic systems [115]. For these reasons, insects are an ideal platform to build mobile sensor networks for remote sensing in any sort of environment [73].

However, the question remains as to what signals are to be measured from the insect and how are they to be retrieved. In literature several hybrid approaches have been reported that tap into the insects’ ability to sense and behaviorally respond to specific environmental conditions [7,37,65,100]. Unfortunately, extracting relevant information from such an organism requires a means to observe this behavior without constraints on movement, which can be difficult or even impossible in field deployments. Also, monitoring just the changes in behavior of the organism in response to changes in the sensing environment might lead to false alarms since many factors or targets could produce similar changes. Alternatively, some
approaches isolate and harvest the functional element (antenna, auditory organ or the sensing neuron itself) from the insect [43, 45, 83, 97]. Though these approaches provide specificity, the viability and ambulatory capability of an intact organism is sacrificed. Yet another class of bio-hybrid insect systems utilize the insect primarily as an efficient vehicle for carrying payloads [130, 132] and utilize artificial sensors for the sensing tasks [73]. These studies have developed flight control strategies and localization algorithms. This approach is suitable for applications where the artificial sensors show superior performance than the biological ones. However, in a task like chemical detection, current artificial sensors are outperformed by biological sensors in terms of sensitivity and range of chemicals detected [69].

Monitoring neural responses from an intact organism, with minimal disruptions, could provide advantages of all the different approaches. Neurons are highly specific and the feasibility of measuring specific groups of neurons from an insect brain and correlating the response to a relevant stimulus have been reported for a range of insects [16, 127]. The intact organism would be ambulatory and potentially controllable [131]. In addition, insects possess useful processing circuits which can be co-opted. For example in a locust, the antennal lobe (with around 800 projection neurons and 300 local neurons) is the primary olfactory processing center, which receives information from 50,000 upstream olfactory receptor neurons, integrates and compresses these high-dimensional sensory signals and sends them to downstream centers like the mushroom body and lateral horn [86]. In [129] (some data shown in Chapter 2), it was shown that explosive chemicals can be accurately classified by tapping into the antennal lobe of the locust.

There has been significant progress in many areas of electrophysiological neural recording, mainly in electrode design, bio-compatibility, high density channel count, data processing and short range wireless power transfer and communication [1, 30, 53, 58, 124]. However
designing a neural recording system that can mounted on a flying insect for long distance remote sensing applications is still extremely challenging. The primary challenge stems from the delivery of sufficient power for amplification of microvolt neural signals and for high data rate transmission, while being severely constrained in terms of weight and form factor. Apart from batteries, there are two general methods of powering neural implants. First method is transferring power wirelessly from an external source. Examples include the inductively systems powered by [4, 32, 42, 60], near-infrared light [62] or ultrasonic signals [137]. The main disadvantage of remotely powered neural activity monitors is that the power-source and the animal must be in close proximity to each other. Therefore, these methods limit the movement of the animal within a small area (for e.g. an electromagnetic cage [80]). Another limitation is that the size of the energy harvester could be prohibitively large for use in small organisms like insects. An emerging and promising alternative method of powering mechanism is by harvesting energy from within the organism or from the ambient environment. Internal harvesters have the advantage of being always-on. These include sensors powered by vibrations inside the body [85] or from blood glucose or ATP [122]. Unfortunately, the power densities of these methods are low and hence these methods require a large harvester volume to be useful for continuous monitoring of neural activity in insects. Some external energy sources do have sufficient power densities for neural recording but they may not be reliably available. A combination of such harvesters and batteries might be a viable approach to increase the operational lifetime of neural implants in the field.

In literature, very few wireless systems that are compact enough for a flying insect have been reported [46, 144]. In [46], a 170 mg battery operated system was reported with a range of 2 m and an operational lifetime of 5 hours. In [144], a 38 mg battery-free system with wireless power transfer was reported that could last indefinitely but its range was limited to 1.5 m. Thus significant improvements in energy efficiencies need to occur in order to
increase the range and the operational lifetime of neural recording systems. In Appendix B, a neural recording and long range wireless communication system that our lab designed using Commercial-Off-The-shelf components is described. The system weighed 1.2 g (with a 0.4 g battery), had a transmission range of 75 m and lasted for two and a half hours. We observed that, with the backpack, the locust was able to locomote freely on the ground. However, there was a sharp decline in its flying ability, possibly due to obstructions to its wings or due to the added aerodynamic load.

In this dissertation, I explore algorithmic and device based approaches for overcoming the limitation of current neural activity recording-technologies in terms of energy efficiency and longevity. A locust insect model (Schistocerca Americana) was chosen with the express task of classifying the olfactory signals that the locust encounters in its environment using neural signatures from its olfactory centers. Specifically, I investigate four neural recording paradigms for sensing and logging (and/or transmitting) neural signals from the brains of flying insects. These are summarized in Fig. 1.1. A Type-I system describes a traditional neural recording setup (for example, the system described in B), against which I compare my approaches.

In Chapter 2, I investigate whether changes in energy content of the extracellular neural recordings contain sufficient information for specific stimulus identification tasks. Measuring the energy content of the signal can be achieved using sampling rates that are significantly lower than what is required for transmitting raw full bandwidth data and hence could lead to significant improvements in energy-efficiencies. This hypothesis was investigated and verified using the locust olfactory system for odor identification. We could reliably classify odors based on energy information collected from the extracellular medium in the locust
Figure 1.1: Neural recording paradigms
an antennal lobe. Based on the energy signal, I implemented a real-time classifier that can classify odor information on-chip.

To eliminate the high power consumption of long range wireless communication, I investigated energy-efficient approaches for storing data in non-volatile memory. In Chapter 3, I demonstrate ultra-low-power compressive sensing-storage circuits that can efficiently store a compressed signal in floating-gate memory cells. The compressed information could be retrieved later and the events of interest reconstructed offline.

I then investigated novel memory devices to reduce the power consumption of the memory cells themselves. In Chapter 4, I show that by modulating the trajectory of synchronized dynamical systems, information can be stored by expending less than 1pJ of energy per bit, a four orders of magnitude improvement. These dynamical systems are based on the physics of quantum-mechanical tunneling of electrons through thin-oxide layers. Because these devices are thermodynamically driven, the memory device (FN Memory) based on this principle can operate at femto-watt power levels.

In Chapter 5, I exploit the dynamical nature of the FN Memory to create a synaptic device which addresses the energy-efficiency imbalance between the training and the inference phases observed in neuromorphic learning systems. Our proposed synaptic element is adaptive with respect to its data retention capacity which can then be traded-off with respect to the energy-dissipation per update. This system could potentially be used for training a low-power classifier in the field. Compressively recording the energy signal on FN memory forms a Type II system while compressively recording the classified signal forms a Type III system.
In Chapter 6, I push the limits of the FN Memory cell. I investigate the feasibility of a completely self-powered neural recording paradigm in which the neuronal action-potentials modulate the dynamical memory cells directly. Through modeling studies, I show that modulation of the quantum-tunneling barrier by a neural activity can be registered as a change in the response of the self-powered dynamical system. However, the signals are smaller than the detection limits of current readout circuits. I propose techniques for recovering the information and reconstructing the time history of the input signal. This recorder forms a Type IV system.

Finally, I conclude my dissertation in Chapter 7. I systematically investigate the four scenarios outlined in Fig. 1.1, implementing the techniques described in this dissertation. I estimate the power consumption, operational lifetime and information throughput for each scenario.
Chapter 2

Energy-based sensing for explosive detection


**Abstract:** Stand-off chemical sensing is an important capability with applications in several domains including homeland security. Engineered devices for this task, popularly referred to as electronic noses, have limited capacity compared to the broad-spectrum abilities of the biological olfactory system. Therefore, we propose a hybrid bio-electronic solution that directly takes advantage of the rich repertoire of olfactory sensors and sophisticated neural computational framework available in an insect olfactory system. We show that select subsets of neurons in the locust (*Schistocerca americana*) brain were activated upon exposure to various explosive chemical species (such as DNT and TNT). Responses from an ensemble of neurons provided a unique, multivariate fingerprint that allowed discrimination of explosive...
vapors from non-explosive chemical species and from each other. Notably, target chemical recognition could be achieved within a few hundred milliseconds of exposure. In sum, our study provides the first demonstration of how biological olfactory systems (sensors and computations) can be hijacked to develop a cyborg chemical sensing approach.

**Author contributions:** BR conceived the study and designed the experiments/analyses. EA demonstrated initial feasibility using the invasive preparation. DS and RC independently developed minimally invasive recording approaches. DS showed feasibility of the overall approach and initial mobile robotic sensing experiments. DM setup the acquisition system. RC performed stability analysis, repeated experiments and validated the results. DM performed all the analysis. DS, DM and RC generated all the figures and co-wrote the methods section. MT developed the odor box and RL programmed the line-following robot. PG calculated the concentrations of the explosive vapors delivered in our experiments. SS provided the explosive chemical samples. SC advised on the instrumentation aspects of the work. BR wrote the paper taking inputs from all the authors and supervised all aspects of the work.

In this chapter, I demonstrate that changes in energy content of the extracellular neural recordings contain sufficient information for specific stimulus identification tasks. A key thing to note here is that instead of using raw neural data collected at 15 kHz, we were able to carry out classification tasks using the energy of the signal present at the electrode which was calculated in 50 ms time-bins (20 Hz rate). These results show that, if the signal energy was calculated on the backpack, the rate of data transmission (or of data logging) can be reduced by a factor of 750 thereby leading to significant energy savings. In addition, a real-time classifier is described which can further reduce data rates by only logging data when the target chemical has been identified.
2.1 Surgical procedure

For detailed description of the methods employed in the collection of neural signals, please refer to [129]. In brief, a minimally invasive surgical protocol was developed, following which twisted wire tetrodes were inserted into the antennal lobe of an adult locust. 2 sets of odor panels, each consisting of explosive chemicals, non-explosive volatile odorants and controls. The data for odor panel consisting of TNT, DNT, Hexanol, Benzaldehyde and Hot Air are analyzed in this chapter. Each locust was subjected to 10 trials for each of the chemical. Each trial lasted for 40 seconds and the stimulus corresponding to that trial was presented at the 10 second mark by an automated pump. Data were collected by Intan RHD2132 amplifier board.

2.2 Estimation of energy in the neural signal

We observed that spike sorting led to heavy loss of information and used an alternative data processing approach (signal energy) for pre-processing signals collected using the minimally-invasive procedure. The collected raw data were filtered using a bandpass filter between 300-5000 Hz and passed through a continuous moving RMS filter with a 20 ms window size (using standard MATLAB DSP toolbox). The data were then down-sampled by a factor of 150, smoothed by a ten-point moving average filter and further down-sampled by a factor of 5. The final temporal resolution that was used for rest of the analysis was 50 ms (same temporal resolution as the invasive preparation). We defined the baseline RMS signal level for each trial by taking a mean of the RMS signals observed in a two-second pre-stimulus
window. To obtain the odor-evoked responses, baseline RMS voltage was subtracted to obtain the $\Delta$RMS values (Fig. 2.1B).

**Figure 2.1:** A) A picture of locusts with electrodes implanted into its antennal lobe using a minimally invasive surgery procedure is shown. The expanded panel shows the relative sizes of the implanted locust and amplifier (Intan Technologies). B) Calculation of signal energy from raw recorded neural activity. Colored rectangle indicates the time window when odor stimulus was presented. Briefly, it is obtained by calculating the RMS signal in a 50 ms moving window. The signals were smoothed and baseline subtracted. For details see Methods. C) Example traces of processed RMS signals recorded from two different electrodes are shown. Colored rectangles indicate time when odor was presented (4 s). RMS signal observed in different electrodes are distinct for different chemicals.

This operation is both computationally cheap and ensures that all signals recorded are used towards discriminating the odorants. Using this approach, we were able to reliably extract odor evoked signals both during stimulus presentation (i.e. an ‘ON’ response) and after
cessation of odorant (i.e. an ‘OFF’ response). Note that both ON and OFF responses can provide stimulus-specific information for classification (Saha et al. 2017).

Further, total signal energy was monitored for each electrode in a 50 ms moving window, and averaged across trials. Measurements from different electrodes were concatenated to create a multivariate neural response vector that was used for evaluating the response specificity. Fig. 2.1C shows the unique RMS signatures evoked by different odorants in different recording electrodes.

We probed the responses of neurons in the antennal lobe while the locusts were exposed to an odor panel consisting of TNT, DNT, hexanol, benzaldehyde and hot air (Fig. 2.2A). For data visualization (Fig. 2.2), we performed two kinds of dimensionality reduction analyses – Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA). For further details regarding these techniques, please refer to [127]. Fig. 2.2B shows dimensionality reduced neural response measurements made from locusts while they were exposed to various analytes in the odor panels. Note that different chemical species evoked neural responses that were unique and different from the others indicating that every odorant used in the odor panel, including the different explosive chemicals, could be detected and precisely recognized. Also, explosive and non-explosive chemicals formed distinct clusters when discriminability is considered for the broad categorical case (explosive vs non-explosive vs control; Fig. 2.2B).

### 2.3 Classification analysis

Observations were labelled as described for the LDA analyses (see above). For obtaining an unbiased estimate of classification accuracy, we performed leave one trial out cross validation
analysis (Fig. 2.2D). During each iteration, data from one trial for all odors were removed from the training set and used as test data. A quadratic discriminant was fit on the remaining data where the predictors were the neural responses in one time-bin and the expected classifier output was the class label for that time-bin. Thus, for a test trial for one odor, we obtained 80 predicted responses. The class label for a trial was taken to be the mode of those 80 responses. A confusion matrix (Fig. 2.2D) was created by comparing the predicted responses to the known responses. Briefly, $C_{ij}$ is the number of trials of Odor $i$, predicted to be Odor $j$. A fully diagonal matrix indicates 100% classification accuracy. The confusion matrix was largely diagonal indicating low misclassification rates.

These results show that, if the signal energy was calculated on the backpack, the rate of data transmission (or of data logging) can be reduced by a factor of 750 thereby leading to significant energy savings while maintaining a high level of accuracy.

### 2.4 Wisdom of the swarm

A major challenge in stand-off chemical sensing and localization is detecting target odorants which are heavily dispersed and subject to complex wind dynamics. Biological systems have evolved to mitigate these challenges. However, when tapping into these biological systems, there is a loss of information at each stage of the recording process. For example, we can record from only a subset of the neurons in the antennal lobe and increasing the number of recording sites could compromise the biological system. Data processing and classification lead to further loss in information. Using multiple locusts increases the likelihood of an ORN on the antenna getting activated by the target odorant and our recording system picking up the downstream activity in the antennal lobe. Therefore, we investigated whether increasing
Figure 2.2: A) Odor panel A included explosives (TNT, DNT), volatile chemicals (hexanol, benzaldehyde) and control (hot air). TNT and DNT were in crystal form and were heated to 50 oC to generate enough vapors. Therefore, puffs of air heated to the same temperature was used as an additional control. B) Visualization of high dimensional data through dimensionality reduction using a 3-class linear discriminant analysis (LDA) is shown (Variance in parentheses). Chemicals were labelled as explosives (red), non-explosives (blue) or controls (gray). LDA shows clear separation between explosives and non-explosive chemicals (n = 15 recordings). C) Visualization of high dimensional data through a multi class LDA where each chemical was treated as its own class is shown (Variance in parentheses). These plots show that individual explosives show distinct responses (n = 15 recordings for panel A). D) Confusion matrices summarizing the results from classification analyses are shown. Note that the matrices are mostly diagonal indicating that both explosive and non-explosive chemicals can be correctly identified based on the neural responses they evoke.

the number of locusts increases the signal-to-noise ratio and thus the classification accuracy of explosive chemicals. First, we performed classification analysis on single locusts, using signal from just one electrode (that had the highest variance for the duration of the experiment). We found that responses from any single locust had enough information to outperform a
Figure 2.3: A) Classification performance using data recorded from the best channel/electrode in each individual locust is shown. Locusts are sorted based on the classification accuracy (low to high). Dashed line indicates performance of a naïve classifier for a 5-class problem (20% chance). B) Monte-Carlo simulations showing improvement in classification performance as data collected from multiple locusts are combined. Accuracy increases with number of locusts and reaches 80% using data from only seven locusts.

naïve classifier (Fig. 2.3A). However, the gains in classification accuracy came from the locust’s ability to robustly classify relevant naturally occurring odors for which they have highly tuned responses. Only one locust was able to classify with an accuracy greater than 60%. To quantify the performance of a population of locusts, we performed Monte Carlo simulations by selecting data from a random subset of locusts.

For determining how performance varied as a function of number of locusts used in the analyses (Fig. 2.3B), a random subset of locusts were chosen (from n locusts choose k random locusts; k was varied) and signals from one electrode with the maximum variance was selected for each locust. The signals from different locusts were combined and the classification analyses was repeated. Accuracy was calculated as ratio of correctly classified trials to total trials. This was repeated 20 times, and the mean and standard deviations of accuracy for each group size were calculated and plotted in Fig. 2.3B. Combining data from multiple organisms led to significant improvements in performance, with average accuracy reaching 80% with just 7 locusts (Fig. 2.3B). Thus, as can be expected, our results indicate that
sensing with multiple organisms would lead to more efficient detection of the target chemical species.

## 2.5 Rapid recognition of the target chemicals

![Figure 2.4: Rapid identification of chemicals using neural responses: Top panels: high dimensional responses and how they evolve over time are visualized in two dimensions following principal component analysis. Numbers within parentheses indicate the variance captured along that axis. The three panels show evolution of neural responses for the first 250 ms, 500 ms and 750 ms after stimulus presentation. Note that neural responses are similar at 250 ms, but become distinct as they continue to evolve. Maximum separation is reached around 500 ms and and the responses start returning towards baseline within 750 ms. Thus, the transient neural responses can be utilized for rapid chemical identification. Bottom panel: pairwise distances between neural responses are plotted as a function of time. Black line indicates the mean pairwise distance across all chemicals. Maximum distance indicating maximum separation occurs at 500 ms after the onset of chemical stimuli.

In many applications, rapid recognition of the target chemicals is highly desirable. Therefore, we sought to examine how quickly we could resolve the identity of the encountered chemical...
based on the neural signatures obtained. To understand how response patterns evolve over time, we performed a response trajectory analysis [127,141]. For this analysis, the multivariate signal energy across electrodes were projected onto the top three eigenvectors of the covariance matrix (i.e. PCA dimensionality reduction; Fig. 2.4). PCA analysis showed that both explosive and non-explosive odorants generated neural responses that evolved over time. The responses started from overlapping pre-stimulus baseline activity and quickly became odor-specific (Fig. 2.4). We found that odorants became discriminable as the odor-evoked neural activity distributed across the ensemble of spiking neurons became odor-specific within 500 ms of their onset. To verify this result, we computed pairwise distance between odorants and plotted them as a function of time (Fig. 2.4). Consistent with the results from the PCA analysis, we found that the peak distance between pairs of odorants happened within 500 ms for all odor pairs (around 500 ms after stimulus onset). These results indicate that the neural response within a few hundred milliseconds of the odor onset is highly unique and can be used for rapid recognition of the chemical identity.

### 2.6 Real time classifier

A real-time linear classifier is implemented that calculates the probability of an observation $x$ belonging to a class $C$ based on its Mahalanobis distance to the cluster mean, $d(x, C)$

$$d(x, C) = (x - \mu_C)\Sigma^{-1}(x - \mu_C)^T$$  \hspace{1cm} (2.1)

Here $x$ is the instantaneous neural energy signal for each electrode and $C$ is the target chemical stimulus. The cluster means $\mu_k$ and the covariance matrix $\Sigma^{-1}$ are pre-calculated during the training phase and stored on chip. Based upon the relative distances to the labeled
Figure 2.5: Real time linear classifier a) Input signal from four electrodes. b) Probability of current activity being generated by a target stimulus. c) Average probability over a moving window. Shaded regions indicate application of stimulus.
clusters, a probability is calculated for that observation to belong to each class \( P(x|C) \). The probabilities are summed over a 5 second window \( (T_{class}) \) and if the sum is above a threshold, a classification event is asserted. A window is used because stimuli encountered by an insect are expected to last a few seconds while sporadic neural activity that might lead to a false positive can be ignored.
Chapter 3

Compressive sensing

This chapter is based in part on a manuscript under review [104]: Mehta, Darshit and Shantanu Chakrabarty "A Nano-watt Compressive Sensing-Storage System-on-Chip using Linear Floating-gate Injectors" (In Review)

In this chapter, I investigate energy-efficient approaches for storing data in non-volatile memory. I demonstrate ultra-low-power compressive sensing-storage circuits that can efficiently store a compressed signal in floating-gate memory cells. The compressed information could be retrieved later and the events of interest reconstructed offline. The architecture is generic enough to compressively store any signal of interest.

3.1 Introduction

Remote wireless sensors like unattended ground sensors [75], wild-life tracking sensors [77] or structural health monitoring sensors [8] need to be active at all times to ensure that it can record events of interest, like seismic activity or illegal intrusion. Often, the events of
interest are infrequent or rare, as such a common practice for reducing the sensor's energy-budget is to use wake-up circuits (energy-detection circuits) or an embedded pattern classifier to gate the energy-intensive wireless transmission [12]. The wireless transmission on these wake-up systems is asynchronously triggered [8] before the sensor data can be transmitted, therefore, the information encoding the sparse events needs be stored on a non-volatile memory. Previously, we had reported several variants of self-powered and quasi-self-powered piezo-floating-gate (PFG) sensors [8, 67, 159] based on this asynchronous retrieval paradigm. However, the previous architectures only allowed for sensing key statistics of the events (for example level crossing or time-of-occurrence), as a result, the temporal evolution of the events could not be inferred. In this work we exploit the time-sparsity of the rare-events in conjunction with a compressive sensing-storage approach to design ultra-energy-efficient sensor systems that can allow for near complete temporal reconstruction of the rare events.

In literature compressive sensing (CS) techniques have been used for sensing sparse, naturally occurring signals, and CS systems have resulted in significant savings of critical resources like channel bandwidth, number of expensive sensors or time of acquisition [11, 22]. CS systems have been deployed for diverse application including healthcare, imaging, communication, defense etc. [33, 50, 92, 121]. In this work, we have designed a compressive sensing-storage (CSS) system that directly uses analog non-volatile storage to compressively store time-history of rare events. A generic CSS system is illustrated in Fig. 3.1 where an analog front-end converts the salient features of time-series input $S$ into a time-encoded signal $x$. This analog front-end could be a simple wake-up circuit based on energy-detection or an ultra-low power classifier [26]. As shown in Fig. 3.1, the event is time-encoded as pulse-duration or using pulse-intensity. The time-encoded signal $x$ is measured and compressed directly on an analog non-volatile memory. In this paper, a combination of linear floating-gate injector array and a LFSR pseudo-random generator has been used for implementing
Figure 3.1: Signal flow diagram showing the use of the proposed compressive sensing-storage. The energy of the input signal (S) is converted into a time-encoded sparse signal x which is compressively sensed and stored as signal y distributed spatially across multiple storage cells. When sufficient external power is available y is retrieved to reconstruct the sparse events $\hat{x}$.
Figure 3.2: Architecture of the compressive sensing-storage SoC comprising of 256 daisy-chained core cells. Each cell consists of a floating-gate (FG) injector as analog non-volatile memory where hot-electron injection is used to program the floating-gates. A digital controller is used to select the cell and determine the state of the cell. A Linear Feedback Shift Register (LFSR) is used for generating a maximum length pseudo-random sequence of 256 bits that is used to create the sensing matrix $A$. 
the CSS module. The analog output of the module $y$ is stored across multiple floating-gate memory cells and due to its non-volatile nature is robust to brown-outs or due to battery failures (due to thermal cycling or leakage). This non-volatile data can be retrieved at a later timer $T_{\text{meas}}$ using a wireless interrogation module that is externally powered or triggered. The retrieved $\hat{y}$ is used to reconstruct the time-encoded signal $\hat{x}$. Even though in this paper we demonstrate the use of this system for ultra-low-power recording and time-stamping of rare events, it is flexible enough to be used for other applications.

### 3.2 Background and Principle of Operation

Compressive sensing was formulated as a way of recovering signals perfectly from sub-Nyquist sampling by making assumptions about sparsity of signals on some specific basis [22, 41]. Detailed mathematical analysis, reviews and application-specific implementations of compressive sensing can be found in these articles [11, 41, 121]. Here, we briefly describe the background information required for our particular application.

If an input signal $x \in \mathbb{R}^N$ is known to be $k$-sparse, that is, it has at most $k$ non-zero coefficients in some orthonormal basis, the sparsest solution to the problem

$$ y = Ax \quad (3.1) $$

where $y \in \mathbb{R}^M$ and $A \in \mathbb{R}^{M \times N}$, can be found by

$$ \min_x ||x||_0 \ s.t. \ Ax = y \quad (3.2) $$

24
This problem can be exactly solved with just $M = k + 1$ measurements. However, it is a combinatorial problem and, in general, computationally intractable. Candès et al. showed that under certain conditions [21], minimizing $||x||_1$ would yield the same result, but is much easier to solve computationally. The resulting optimization problem is described as

$$\min_x ||x||_1 \text{ s.t. } Ax = y$$  \hspace{1cm} (3.3)$$

The conditions involve taking measurements of projections of $x$ on a basis that is incoherent with the basis in which $x$ is sparse. An i.i.d Gaussian distribution or random Bernoulli matrix is a suitable candidate as the sensing matrix $A$ for a wide range of problems. In case of noise present in the measurements (often the case for sensory systems), equality constraint in (3.3) can be recast as an inequality constraint that allows for sparsest signal to explain the observations with some level of inaccuracy

$$\hat{x} = \arg\min_x ||x||_1 \text{ s.t. } ||Ax - y||_2 < \epsilon$$  \hspace{1cm} (3.4)$$

where $\epsilon$ can be chosen based on the noise characteristics of the problem to be solved. Solution to the above equation can be obtained by Second Order Conical Programming (SOCP). In this paper, we report design and experimental validation of a low power recording system interfaced with an on-chip generated random matrix $A$. Error in the reconstruction is estimated relative to the norm of the input signal

$$E = \frac{||\hat{x} - x||_2}{||x||_2}$$  \hspace{1cm} (3.5)$$
Figure 3.3: Micrograph of the chip fabricated on 0.5µm CMOS process. Inset shows layout of the linear injector cell.

3.3 SoC Implementation

The compressive sensing-storage SoC comprises of an array of 256 cells, with each cell consisting of a floating-gate transistor, a linearized hot-electron injection programming system and control logic (Fig. 3.2). Depending on the state of the internal DFF, a cell could be either active or inactive. During recording, when an input signal is present, the linear hot-electron injection system of the active cell turns on and injects electrons into the respective floating gates for the duration of the signal. A compressive sensing system is implemented by selecting a group of cells (in a pseudorandom manner) that will be active in a given time-bin. The pseudorandom sequence is implemented by a Linear Feedback Shift Register described below. A micrograph of the circuit fabricated on 0.5µm CMOS process is shown in Fig. 3.3.
Figure 3.4: a) Demonstration of linearity of injection system. Cells which were not linear (eg. Cell 2) were discarded. b) Control of rate of injection through modulation of Vref. c) Variation of injection rates across all cells, used for normalization.
3.3.1 Linear Injector circuit

Rate of injection onto the floating-gate depends on the potential difference across the channel of the transistor and on its gate voltage, which in-turn depends on the charge stored on the floating-gate. By using a negative feedback approach described in detail in [67], rate of injection can be made independent of these parameters. During injection, M1 (in Fig. 3.2) as well as the feedback amplifier (A1) are turned on. Due to the gain of the amplifier, at equilibrium, source of the floating gate PMOS M2, is set to $V_{ref}$. It achieves this by setting the FG node to an appropriate potential since M2 is configured as a source follower. The constant rate of injection leads to a linear relationship between amount of injection and duration of injection (Fig. 3.4a). The slope of this curve is used to estimate the rate of injection for that cell. If some cell did not exhibit a linear injection response (Cell 2 in Fig. 3.4a), that cell was discarded during reconstruction. Eventually, we used 236 out of the 256 cells for reconstruction. Another important feature for the linear injector based design is that rate of injection can be easily modulated by choosing an appropriate value of $V_{ref}$ (Fig. 3.4b). Rate of injection follows an exponential relation with $V_{ref}$. If short events are expected, than a higher $V_{ref}$ can be used. Or $V_{ref}$ could be modulated with time to introduce more randomness in the sensing matrix. Due to mismatch, rate of injection as a function of $V_{ref}$ needs to be characterized for each cell and used for calibration (Fig. 3.4c). The final measured changes in floating gate voltages is then normalized by their rate of injection to enable accurate reconstruction of the input signal. The change in floating-gate voltage $\Delta V_{i,n}$ of $i$-th cell during the time bin $t$ is given by

$$
\Delta V_{i,n} = k_i A_{i,t} x_n
$$

(3.6)
where \( k_i \) is the rate of injection for that cell. \( A_{i,n} \) is the state of the DFF of that cell for time bin \( n \), and \( x_n \) is duration of event in that time bin. For compressive sensing, \( x \) is considered to be sparse. Due to the non-volatile nature of floating gate, change in floating gate voltage after \( N \) time-bins is the cumulative addition of changes in each of the time-bins.

\[
\Delta V_i = k_i \sum_{n=1}^{N} A_{i,t} x_n
\]

(3.7)

After normalization \( \hat{y}_i = \Delta V_i / k_i \), system equation can be rewritten in the matrix form (equivalent to (3.1))

\[
\hat{y} = A x
\]

(3.8)

The reconstruction problem is formulated as finding \( \hat{x} \) that satisfies (3.8) within certain level of measurement error.

\[
\hat{x} = \arg\min_{x} \|x\|_1 \quad s.t. \quad \|Ax - \hat{y}\|_2 < \epsilon
\]

(3.9)

In our previous work, we have shown that floating gates can be programmed via linear injection with an accuracy greater than 13 bits. Hence for an operating programming range of 1V, we chose \( \epsilon = \pm 100\mu V \).

### 3.3.2 Linear Feedback Shift Register

We generated pseudo-random Bernoulli sensing matrix \( A \), using a Fibonacci Linear Feedback Shift Register (LFSR). A Fibonacci LFSR is guaranteed to generate Maximum Length Sequence with good statistical properties for randomness. At the start of the sensing-storage operation, a Reset command while in recording mode generates an alternate sequence of 32
Figure 3.5: a) Sensing matrix implemented by the LFSR. b) Accuracy of the LFSR based sensing matrix for a range of input vector lengths with varying levels of sparseness.
ones and 32 zeros. During each clock cycle, the register is shifted to the right by one and 1st bit is set via an XOR operation of the state at the taps 246, 251, 254 and 256. The taps for the 256-bit LFSR were obtained from [148]. The generated sensing matrix is shown in Fig. 3.5a. For the generated matrix, Fig. 3.5b shows the performance trade-offs between the number of non-sparse events that can be accurately reconstructed and the length of the input signals. By providing an override to serially feed in a custom data stream into the first bit, say via a micro-controller, a different sensing matrix can be implemented. During the readout phase, since each outputs from all cells are read out via a multiplexer, only one cell can be active. Here, Reset leads to just the first cell being selected, the LFSR feedback is turned off and Bit 256 is directly fed into the first cell, which allows sequential cycling through all cells.

3.4 Experimental results

Shown in Fig. 3.6, are experimental results measured using the fabricated chips. The input signal lasted for 15,000 seconds. It was divided into 30 second time bins with 500 time bins. At the conclusion of each time interval, a clock signal performs a bit shift operation in the shift register. The input to channel 1 is the XOR combination of the LFSR taps. In 20 randomly selected time bins we simulated an encounter with the signal of interest. The encounter lasted for a random duration between 5 and 15 seconds. Whenever the signal was present, a pre-decided subset (for that time bin) of injector cells would be activated and inject electrons into their respective floating-gates for the duration of the event. After 15,000 seconds, we measured floating gate voltages for each cell and calculated the change relative to the pre-experiment state. Since injection rates show variation across cells, we normalized
Figure 3.6: Experimental results: a) Change in floating gate voltages after the conclusion of the experiment lasting 15000 seconds. The actual measurements matched the expected values with an RMSE of 9.77 mV. b) Results from CS reconstruction.
the total injection with that cell’s rate of injection. The normalized change in floating gate voltages that we obtained is shown in the upper panel of figure below. It matches very closely to the expected changes with an RMSE of 9.77mV. We then applied reconstruction algorithm formulated by Candes [20] as a MATLAB toolbox and our results are shown in bottom panel. Of the 20 events, we were able to accurately recover 18 of those events. There were 2 false negatives and 10 false positives. Considering the actual event intensity values, we obtained a normalized error (using equation 3.5) of 0.55. We estimated the power consumed by our system to be 532 nW (Table 3.1). The most significant source of power dissipation in the SoC is the hot electron injection, which is activated for around half of the cells whenever the sparse signal of interest is encountered. In addition, continuous power is needed to keep the reference circuits and DFFs active and for operating the LFSR circuit.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology</td>
<td>0.5μm</td>
</tr>
<tr>
<td>Cell Area</td>
<td>255μm × 70μm</td>
</tr>
<tr>
<td>Quiescent power</td>
<td>95nW</td>
</tr>
<tr>
<td>LFSR switching power</td>
<td>27nW</td>
</tr>
<tr>
<td>Injection power (instantaneous)</td>
<td>30.7μW</td>
</tr>
<tr>
<td>Injection power (average)</td>
<td>410nW</td>
</tr>
<tr>
<td><strong>Average total power</strong></td>
<td><strong>532 nW</strong></td>
</tr>
</tbody>
</table>

### 3.5 Conclusions and Discussions

In this paper we have reported a low-power SoC implementation of a CSS system. Using a fabricated prototype we were able to record a 15,000 seconds long sparse signal, with a resolution of 30 second time intervals at an average power consumption of around 530 nW
as shown in Table 3.1. It can be envisioned that an SoC with more linear floating-gate injectors would be able to reconstruct more events, however, its important to note that the static power dissipation of the system will also increase. One mechanism to reduce the power dissipation is to reduce the magnitude of the injection current which will reduce the change in voltage recorded on the floating-gate per event. The fundamental limit on the measurable change is determined by the thermal-noise in the hot-electron injection process. In [67], we presented a theoretical analysis which showed feasibility of recording $100\mu V$ changes which corresponds to a precision of 13 bits. The operational life of the system is limited by the number of false-alarms that is generated by the wake-up system or the integrated classifier. Therefore, future research will focus on improving the robustness of the classifier and in investigating sparser sensing matrices that would lead to fewer injector cells being active and lower power consumption.

### 3.6 Application of CS for neural recording

**Energy waveform reconstruction**

Fig. 3.7 shows reconstruction of an energy waveform after being compressed by a factor of 4. It shows that the salient transient features of the waveform at the onset of the stimulus and after the removal of the stimulus were successfully reconstructed. It has been shown that these transient signals contain maximum information about the stimulus. However, high frequency signals were observed. This is a common problem in reconstruction of under-determined systems. One way to avoid this would be to use a regularizing term that penalizes large variations in the reconstructed signals. Another reason for poor reconstruction performance is that, in this proof of concept example, we did not attempt different basis functions.
Figure 3.7: Signal reconstruction after 4x compression

for $\Psi$. Instead, the signal is assumed to be sparse in the time domain, which it is evidently not. If a proper basis space $\Psi$ is identified, in which the energy waveform is sparse, then a superior performance of the reconstruction algorithm is essentially guaranteed.

In spite of subpar performance for individual waveforms, an ensemble of waveforms acquired from multiple electrodes for different stimuli shows high robustness. When projected onto a lower dimension using Principal Component Analysis, we see clear stimuli-specific differentiation between temporal trajectories for the ensemble Fig. 3.8. The differentiation allows for accurate classification between the stimuli using their neural signatures.
Figure 3.8: a) Principal Component Analysis of original data. b) Classification accuracy of original data. c) Principal Component Analysis of reconstructed data. b) Classification accuracy of reconstructed data.
Figure 3.9: 1) Compressed storage on 128 FN-DAM cells. b) Reconstructed signal from the compressed data. c) Error in reconstruction
Classification reconstruction

Reconstruction of classification events from the output of the classifier module is shown in Fig. 3.9. A compression ratio of 8 was used in this example. As the waveform is expected to be sparse in the time domain, there is no need to find the domain $\Psi$ in which the signal is sparse. All the time instances when the classifier output exceeded the threshold were correctly identified. Even the probabilities of these events were estimated accurately with maximum error in estimation being 0.018.
Chapter 4

Fowler-Nordheim Tunneling based
Dynamical Memory (FN-MEM)


This chapter describes the design of an information storage cell implemented by a differential pair of synchronized systems. To store information, one of the devices is modulated and thus gets desynchronized with respect to the other. The magnitude of the desynchronization is a function of intensity of the input signal. This ultra-low-power memory device can be used in two ways: Either as a standard memory where it is modulated by pulses from a digital circuit (Chapter 5) or as an integrated sensor-logger where the signal being sensed directly modulates the device (Chapter 6).
Figure 4.1: Operating principle and architecture of the proposed memory device.

a) Principle of sensing and data-logging where the input signal leaves its trace on a pair of synchronized dynamical system through a desynchronization process. b) Equivalent circuit model of a self-powered dynamical system where the charge on a capacitor $C$ stores the dynamical state of the system and the dynamics is governed by a leakage current $I(V_t)$ and ambient stimuli $x_t$. c) Band-diagram corresponding to the tunneling junction where the electrons tunnel across the triangular energy barrier and the input signal $x_t$ modulates the barrier shape. d) Cross-section of the sensor-data-logging device showing the FN tunneling junction, the floating-gate which is coupled to a read-out transistor $P$ and a buffer $B$. e) Micrograph of the fabricated devices, with inset showing a pair of dynamical systems configured in a differential architecture.
4.1 Introduction

We propose a self-powered sensing system, where instead of harvesting the energy to switch between static memory states, the sensing signal is used for modulating a synchronized dynamic state. In this regard, dynamical systems, both natural and artificial, have been shown to store information in their dynamic states [9,36,49]. In this work, we show the feasibility of this approach for self-powered sensing and data-logging, but at chip-scale. This is illustrated in Fig. 4.1a which shows two synchronized globally asymptotically stable (G.A.S) dynamical systems; a sensing system and a reference system. A time-varying input signal modulates the state trajectories of the sensing dynamical system leading to its desynchronization with respect to the reference dynamical system. The relative degree of desynchronization between the two systems serves as a medium for sensing and storing the cumulative effect of the input modulation. While the principle is relatively straightforward, there exists two key challenges in implementing the proposed concept at a chip-scale. First, due to self-powering requirements, the synchronized G.A.S. dynamical system can only be implemented using leakage processes driven by intrinsic thermal or quantum transport of electrons. The simplest of such a system can be modeled by an equivalent circuit shown in Fig. 4.1b. The capacitor $C$ in the circuit models the dynamical state (denoted by the time-varying voltage $V_t$) and the time-dependent system trajectory is determined by a leakage current $I(V_t)$. The capacitor $C_{in}$ couples the input signal $x_t$ into the dynamical system. The challenge is that an ultra-low leakage current $I(V_t)$ is required to ensure that the dynamical system is operational for the duration of sensing and data-logging. For instance, a 1 V change across a 1 pF on-chip capacitor over a duration of 1 day would require a leakage current of 10 attoamperes. Even if it were possible to implement such low-leakage currents, it is difficult to ensure that the magnitude of the currents match across different devices to ensure state synchronization.
The second challenge with regard to data-logging is that there exists a trade-off between the non-linearity in the dynamical systems response and the duration over which the information can be retained. As shown in Fig. 4.2a–b, if a constant leakage element (for example reverse leakage current) is used, not only do the system trajectories rapidly converge to the final steady state, but the modulation signal does not cause a change in the sensing system trajectory with respect to the reference system trajectory. On the other hand, a resistive or a direct-tunneling leakage element will be sensitive to the changes in modulation signal but will be unable to keep the two trajectories separated for long periods of time, leading to low retention-time. In this report, we show that a differential G.A.S. dynamical system [105] implemented using a Fowler-Nordheim (FN) quantum tunneling device [160] can address all these challenges.

**Figure 4.2:** a) Different leakage elements (Fowler-Nordheim tunneling, reverse biased diode and resistor) elicit different resynchronization responses. b) Note that the desynchronization response for the reverse diode case is zero. a) Three different types of dynamical systems are simulated based on different leakage element ($I(V)$) in Fig.4.1. All systems would respond to an external signal (a square pulse), deviate from baseline and then resynchronize back to their baseline response (reference profile). b) Desynchronization between sensor and reference. Note: When the leakage elements is a resistor, the dynamics follow an exponential characteristic. However, an extremely large resistance would be required to sustain the effects of the input pulse (or transient response). As an example, for $C = 1 \text{pF}$, $R = 1 \text{T}\Omega$, $V_0 = 3\text{V}$, a 1-second-long, 100 mV input signal will elicit a response that can be observed for 5.5 seconds.
4.2 Results

4.2.1 Long-term reliable synchronized dynamical systems

The operating physics of an FN quantum-tunneling based dynamical system is illustrated using an energy-band diagram in Fig. 4.1c [90]. From a practical point-of-view, this energy-band configuration can be achieved across a thermally grown gate-oxide (silicon-di-oxide), which acts as an FN tunneling barrier that separates a lightly-doped n-type semiconductor substrate from an electrically insulated but conductive polysilicon island (labeled as a floating-gate). A two-dimensional electron gas and a triangular FN tunneling barrier (as shown in Fig. 4.1c) is created by initiating a large potential difference across the semiconductor-floating-gate interface. Thermally-excited electrons then tunnel through the triangular FN tunneling barrier onto the floating-gate \((FG)\) and cannot escape due to the surrounding electrical insulation. Each electron that tunnels through the barrier and is retained, changes the potential of the floating-gate which in turn decreases the slope of the FN tunneling barrier (shown in Fig. 4.1c). Fig. 4.1d shows the cross-section of such an FN tunneling device, whereby the floating-gate is coupled to a programming transistor \(P\) and a source follower buffer \(B\). The read-out procedure and the procedure to initialize the charge on the floating-gate is described in the Methods section. In [160], we showed that the continuous-time dynamics of this device can be modeled using a first-order differential equation which results in the change in floating-gate voltage \(V_t\) at time-instant \(t\) as

\[
V_t = \frac{k_2}{\log(k_1 t + k_0)} + k_3
\]
Figure 4.3: Differential FN sensor-data-logging device and its measured response. a) Equivalent circuit of the differential FN device coupled to the read-out circuitry. b) Sensor and reference output voltages measured across nine trials after the device is initialized. c) Change in sensor and reference values compared to the initial value $V_0$ as $\Delta V_t = V_t - V_0$. Shaded region in inset shows ±1 standard deviation. d) Measured desynchronization between the sensing and reference devices, with bold line showing mean across trials. e) Synchronization measured across a range of operating temperatures ($5^\circ$C to $40^\circ$C). The gradient (dark red to yellow) denotes an increase in operating temperature. f) Standard deviation measured for the sensor, reference and the difference over 36 trials and across range of operating temperatures.
where $k_0$–$k_3$ are model parameters. The parameters $k_1$ and $k_2$ depend on the area of tunneling junction, capacitance, temperature and material properties and the device structure, the parameter $k_3$ depends on the read-out mechanisms and the parameter $k_0$ depends on the initial conditions. For the proposed sensor-data-logger we employ a differential configuration as shown in Fig. 4.3a. The initial voltage (equivalently, charge) on each floating-gate is precisely programmed through a combination of tunneling and hot-electron injection (see Calibration and Initialization in Methods) [60]. One of the FN device’s (labeled as the sensor) dynamics is modulated by an input signal $x_t$, and its desynchronization is measured with respect to a reference FN device as:

$$\hat{Y}_t = V^R_t - V^S_t$$

(4.2)

Here, $V^S_t$ and $V^R_t$ refer to the sensor and reference floating-gate voltages respectively. A capacitive divider (formed by $C_c$ and $C_{FG2}$) followed by a source-follower is used to read-out the floating-gate potential through the output node as shown in Fig. 4.3a. The floating-node formed at the capacitive divider is independently programmed to a lower value ($\approx 3$ V) to ensure low probability of unwanted tunneling or injection through the transistor $FG_2$.

The outputs of the sensor and reference nodes, $V^\text{sensor}_t$ and $V^\text{ref}_t$ respectively, are measured using an external data acquisition system (Keithley DAQ6510) and shown in Fig. 4.3b. The differential output $Y_t$ in Fig. 4.3a is measured with respect to the initial value as

$$Y_t = (V^\text{ref}_t - V^\text{sensor}_t) - (V^\text{ref}_0 - V^\text{sensor}_0) = \Delta V^\text{ref}_t - \Delta V^\text{sensor}_t \propto \hat{Y}_t$$

(4.3)

For calculating $Y_t$, we use the change from their initial voltages at time-instant $t = 0$ seconds ($\Delta V_t$ in Fig. 4.3c) to eliminate the offset in the read-out stage. For each device, less than 1 % deviation was observed across trials, demonstrating the reliability of the tunneling dynamics.
and the reliability of the measurement setup. With respect to the differential measurements, $Y_t$ should be $0\,\text{V}$ in a perfectly synchronized system. However, due to device mismatch and due to differences in the initialization procedure, we observe a baseline drift across all trials. This manifests as variations in device parameters $k_1-k_3$, which were estimated by regressing equation 4.1 to the empirical data (Appendix Table A.2). The estimated parameters were then used to compensate for drift and to determine the sensor output (Appendix Figure A.2). Post-drift corrections are shown in Fig. 4.3d, which shows the maximum difference between a pair of trials to be less than $300\,\mu\text{V}$. We measured the desynchronization of the differential FN device across temperatures ranging from $5\,^\circ\text{C}$ to $40\,^\circ\text{C}$. Higher temperatures led to faster tunneling, which led to a larger variation in $\Delta V_t$ within the range of $200$ to $260\,\text{mV}$ as a function of temperature (Fig. 4.3e). Despite this variation, the measured desynchronization $Y_t$ had a significantly lower variance with standard deviation below $1\,\text{mV}$. These results show that the differential architecture is capable of compensating for variations in temperature. Note that an incorrect initialization of the reference device with respect to the sensor device will make the temperature compensation less robust, as shown by an outlier in Appendix Figure A.3.

### 4.2.2 A simple behavioral model explains the data-logging principle

In the Methods section, we have derived a tractable mathematical model for the data sensed and stored by the sensor-data-logger in response to an arbitrary time-varying input signal $x_t$. We found that the output of the data-logger $Y_T$ measured at time-instant $T$ can be expressed as

$$Y_T = R(T)A_x(T)$$

(4.4)
Figure 4.4: Rectifying response of the sensor-data-logger device. a) Output measured from the device when subjected to an input pulse. During the positive half of the input pulse, the tunneling-rate increases and desynchronizes the sensor device with respect to the reference device. b) Responses measured from three loggers across three trials. The loggers were initialized to different conditions, hence the difference in their measured responses. c) Sensor responses for input signals over a range of amplitudes. Responses follow an exponential model, which can be accurately modeled by the action model and an ODE solver.
where $A_x(T)$ represents the total “action” due to the input signal $x_t$ accumulated up to the time instant $T$ and $R(T)$ is a “forgetting” factor that is independent of the input signal $x_t$. $R(T)$ models the data retention capability and arises due to resynchronization of the sensor and reference FN devices, after the sensor device is perturbed by $x_t$. In the Methods section, we show that the action $A_x(T)$ can be expressed in terms of device parameters as

$$A_x(T) = \frac{k_1}{k_2} V_0^2 \exp \left( \frac{-k_2}{V_0} \right) \int_0^T \left[ \left( 1 + \frac{C_R x_t}{V_t} \right)^2 \exp \left( \frac{k_2 C_R x_t}{V_t(V_t + C_R x_t)} \right) - 1 \right] dt \quad (4.5)$$

and the resynchronization term $R(T)$ can be expressed as

$$R(T) = \frac{V^2}{V_0^2} \exp \left( \frac{k_2}{V_0} - \frac{k_2}{V_T} \right). \quad (4.6)$$

Here $V_t$ is given by equation 4.1 with $V_0$ and $V_T$ representing the device voltage at time-instant $t = 0$ and $t = T$ seconds. The parameter $C_R$ in equation 4.5 models a capacitive divider that is formed due to the coupling of the input capacitance onto the floating-gate. (Figs. 6.1 and 6.2) show several examples of signals $x_t$ for which the first-order action model given by equation 4.4 accurately tracks a more computationally intensive ordinary differential equation (ODE) based device model. In Figure 6.3, we show the “action” $A_x(T)$ corresponding to different signal types with different magnitude and energy. The results show that $A_x(T)$ is monotonic with respect to energy and hence can be used as a measure of cumulative energy.

In our controlled experiments we subjected the FN data-logging device to a square pulse of varying magnitude but with a fixed duration of 120 seconds. This duration was chosen because it is sufficiently long enough to elicit a measurable response and for the purpose of device characterization. Also, the pulse was applied at a fixed time (1,800 seconds), after which the desynchronization $Y_T$ was measured at different values of measurement time.
Experiments were conducted over a duration of 10,800 seconds (3 hours), with the data-logger responses measured every 30 seconds. Each data-logger was calibrated to similar initial conditions for all experiments wherein the sensor and the reference nodes were initialized to equal tunneling rates. A typical experiment demonstrating the recorder in operation is shown in Fig. 4.4a, which matches the model described in the Methods section. The RMSE between the model and measured data is 61µV with an $R^2$ of 0.9999.

Measurement results across three repeated trials for input signals of magnitude 100 mV and $-100$ mV are shown in Fig. 4.4b. The 100 mV signal resulted in a sensor response of $0.8$–$1.5$ mV for the three data-logging devices. At the end of three hours, due to resynchronization, the sensor response decreases down to $0.5$–$0.6$ mV. For the $-100$ mV input, responses after the modulation were in the range of $-0.5$ to $-0.9$ mV, which dropped to $-0.2$ mV at the end of three hours. Though the three recorders had different responses, they were consistent across trials for the same recorder. The device responses at the end of three hours for input signals of different magnitudes are shown in Fig. 4.4c. From the figure, it is evident that the data-logging device response is similar to a rectifier as summarized by the action model in equation 4.5. The action model fits the data for this wide range of input conditions with an $R^2$ of 0.9855.

4.2.3 Energy budget, sensing and retention limits

The rectification property of the FN data-logging device can be useful for measuring and logging the intensity of a time-varying signal like bio-potentials or accelerometer output. The device is sensitive to input signals of any intensity since there is no threshold requirement on the input signal to activate the sensor. The caveat being, the data retention times for
small magnitude signals will be shorter due to the resynchronization (modeled by $R(T)$ in equation 4.6) and operational noise in the recorder. In a perfectly matched differential system, and in the absence of any input, the device response should be exactly 0 V, because of synchronization. However, environmental factors, mismatch between the sensing and reference nodes, or stochasticity in tunneling mechanism, cause desynchronization and the recorder response deviates from the baseline. In general, the variance in the output increases with time (See Fig. 4.3d for example). This increase in variance over time is a form of operational noise in the system ($\sigma_t$). A model for $\sigma_t$ could be estimated by letting the recorder operate with respective inputs connected to the ground (similar to input referred noise experiments) and measuring the deviation of output from the baseline. Another source of noise is the readout noise ($N_0$) which limits the resolution to which charge on the floating gate can be measured. Total noise ($N_t$) is the sum of these two noise sources. While noise increases with time, recorder response decreases due to resynchronization. For a signal of given action, there will be a time instance $T_{ret}$, beyond which the signal-to-noise ratio (SNR) goes below a chosen threshold and input signal cannot be reconstructed with a desirable degree of certainty. We chose unity as our threshold for SNR, and we defined data retention as the time at which the signal falls below system noise. Retention time for a given input signal was found by a fixed point method. First, a noise model was generated using experiments without any input modulation. Standard deviation ($\sigma_t$) was calculated across all runs as a function of time. Ideally, if the dynamics were perfectly synchronized, then the $\sigma_t$ obtained would be 0. However, we find that $\sigma_t$ increases with time due to integration of noise. We fit a rational equation on this noise.

$$\sigma_t = \frac{at}{t + b}$$
We chose this equation so that it stays bounded as time approaches \( \infty \). Total noise in the system is given by

\[
N_t = \sigma_t + N_0
\]

where \( N_0 \) is the noise associated with readout circuits and data acquisition system.

The time of retention \( T_{\text{ret}} \) was defined as the time instance at which the expected recorder response \( (Y) \) becomes lower than the predicted noise in the system \( N_t \), i.e. the signal-to-noise ratio goes below unity. Thus, at time \( t = T_{\text{ret}} \)

\[
Y_{T_{\text{ret}}} = N_{T_{\text{ret}}}
\]

Figure 4.5a shows via an illustration how data retention capacity for a given noise model can be estimated. Figure 4.5b shows that data retention capacity increases exponentially with the signal action. For a 10 mV action, we could expect to measure significant deviation from baseline for over 300,000 seconds (\( \approx 4 \) days). The action model can be used to estimate the energy-budget requirement on the sensing signal. Since the average FN tunneling current is \( 10^{-17} \) A, the energy budget is less than an attojoule. Note that this is the energy to trigger desynchronization. However, isolating the energy dissipated due to FN tunneling from other energy dissipation factors is challenging because the FN tunneling current is on the order of attoamperes, which is orders of magnitude smaller than the reactive current generated by the transducer and the leakage current flowing through ESD protection diodes. In the Supplementary Note 9, we estimate the energy budget when the proposed sensor-data-logger is driven by an arbitrary sensor signal.

Noise in the system can also be described by the effective number of bits (ENOB) (Figure 4.5c). For an assumed action range of 10 mV, 10 bits precision can be initially expected
Figure 4.5: a) Colored traces indicate recorder response for input signals with 100 and 200 µV actions. The gray trace indicates constant noise ($N_0$), due to readout and measurement circuits. Black curve indicates noise ($N_t$) due to unintended desynchronization occurring in absence of an input signal. It is obtained by adding the readout noise to the input referred noise ($\sigma_t$, assumed to be the standard deviation across trials for a recorder with no input signal - see variance in Fig. 2d). Intersection of the response curve with the noise curve (eg. Points A and B) is an estimation of system retention time, $T_{ret}$, at which point, the SNR of the system goes below 1. b) Retention time, $T_{ret}$, plotted as function of action for different noise profiles. Points A and B correspond to the intersection points in panel (a). $T_{ret}$ varies exponentially with action. c) Amount of information, measured as the effective number of bits (ENOB), stored in the system which decreases with time.
Figure 4.6: a) Target voltage, floating gate voltage and training voltage as a function of time. b) Energy required to charge unit capacitance as a function of time.

in a system with 10\,\mu V readout noise. In a perfectly matched system, ENOB would drop to 0 at $\approx 2 \times 10^6$ seconds (total recorder lifetime), but with the added operational noise it takes $\approx 3 \times 10^5$ seconds to reach 0. Readings from multiple recorders can be combined to increase the effective number of bits of the system.

4.2.4 FN memory write energy

The magnitude of input pulse required, $V_{in}(t)$ (Fig. 2a) so that the floating gate node at current potential $V_{FG}(t)$ shifts to a target voltage $V_T$ is given by:

$$V_{in}(t) = (V_T - V_{FG}(t))/C_R$$

(4.7)

Where $C_R$ is the input capacitive coupling ratio $C_R = C_C/(C_C + C_{FG})$. The floating gate voltage $V_{FG}(t)$ is estimated by Equation 4.1.

The energy required to charge the input capacitor is given as

$$E(t) = \frac{1}{2} C_{in} V_{in}^2$$

(4.8)
Figure 4.6 shows instantaneous energy required to charge unit capacitance when $V_T = 7.6V$ and $V_{FG}(0) = 7.5V$. The input capacitance of our device was 1 pF, and the instantaneous write energy per update increased from 5 fJ to 2.5pJ over 12 days.

### 4.2.5 Parametric analysis

Using modeling and simulations, we conducted parametric analysis for our system. Parameters $T$ (Fig. 4.7a) and $V_0$ ((Fig. 4.7b) are operational parameters that can be set at run time according to application requirements. $k_1$ (Fig. 4.7c) depends on the area of the tunneling junction and on the capacitance associated with the floating gate node. $k_1$ and $k_1$ (Fig. 4.7d) are also influenced by the thickness of the insulating material and other material properties like the barrier height at the interface between the conductor and the insulator.

### 4.2.6 Impedance analysis and power estimation

We performed simulation studies to characterize the input impedance of our sensor logger which can be modeled by the equivalent circuit shown in Fig. 4.8a. The DC input impedance of our system is on the order of $10^{18}$ Ω, since the input is connected to the gate of a MOSFET and the FN tunneling current is in the order of attoamperes. In this case, the impedance of the ESD protection diodes dominate the input impedance at DC frequency. We ignore this leakage path for our analysis and the high pass cut-off frequency is found to be at $10^{-5}$ Hz (Fig. 4.8b). At higher frequencies, the input capacitance and gate-to-substrate capacitance, along with input parasitic resistances create a low impedance path. However, this power is predominantly reactive in nature (Fig. 4.8c) and can be minimized with suitable source
Figure 4.7: Simulation results for sensitivity and parametric analysis. Default parameters are $k_1 = \exp(38.5), k_2 = 346, V_0 = 7.5V$. $Y'_T$ is the baseline response at time $T$ for a single square pulse of magnitude 100 mV and duration 1 s. a) Time of sampling determines the amount of resynchronization b) Initial programming voltage affects the sensitivity of the recorder, but its effect becomes attenuated as time of sampling increases (due to resynchronization) c,d) Device parameters $k_1$ and $k_2$ can be tuned via system design and material selection to optimize the recorder response. $\alpha$ and $\beta$ are material parameters [90].
Figure 4.8: a) Equivalent circuit model of a sensor interfaced with sensor-data-logger. b) Relation of $V_{\text{gate}}$ to system input. c) Equivalent input impedance. d) Power spectral density for a matched system.
impedance matching. The power dissipated by the sensor-data-logger can be estimated as

\[ P(\omega) = \Re \left( \frac{V^2}{2Z_{in}(\omega)} \right) \]

Assuming that the natural dynamics of the FN device lie less than the frequency range of 1mHz, the average power dissipated can be estimated over the signal bandwidth of 1mHz to 1kHz to be 0.05 aW. So for an event lasting 100 seconds, energy dissipated is on the order of 5 aJ.

4.3 Methods

4.3.1 Programming and synchronization

The differential sensor-data-logging system consists of two nodes: sensor and reference node. Each node contains two floating gates decoupled via a capacitor. The charge on the four gates of the system can be individually programmed using a combination of tunneling (increases charge, course) and hot electron injection (decreases charge, fine). The programming block for each gate is selected via a switch. Injection is initiated by setting \( V_{DD} = 7 \) V, and setting the input pin to a value (via a DAC), such that \( V_{DS} \) is above 4.2 V. \( V_{DS} \) can be modulated via the gate voltage because the PMOS is in a source follower configuration. Tunneling is realized by bring \( V_{\text{tun}} \) to a high potential. For programming the tunneling node to be in the FN tunneling regime, we used \( V_{\text{tun}} = 21 \) V. Except for the self-powered experiments using piezo crystals, we did not have to program the tunneling node in FN tunneling regime using \( V_{\text{prog}} \) pin. Instead, we could set the input pin to a stable voltage (analog ground) which would push the node into FN regime. The DAC voltage was calculated for each run such that a
tunneling node’s potential at the start of each experiment was the same (as measured by the readout node). When the needed DAC voltage exceeded 5V, we would initiate tunneling. This process allowed us to limit the number of high voltage tunneling cycles and increase the experimental life of the recorder. This process cannot be done in actual deployment because there would not be an external DC source. Hence, for self-powered piezo experiments, we carried out tunneling for each trial.

**Figure 4.9:** Programming and synchronization: a) Each floating gate node can be individually programmed through the Floating Gate Programming (FGP) block. b) Electrons can be tunnelled out of the floating gate by setting the $V_{\text{prog}}$ to a high potential. Electrons can be injected into the floating gate via hot electron injection. The switch $S_j$, set via a shift register, allows for individual control of the injection channel. $V_{\text{inj}}$ node is monitored during injection.

### 4.3.2 One-time programming

For each node of each recorder, the readout voltage was programmed to around 3V while the tunneling node was operating in the tunneling regime. This was achieved through a combination of tunneling and injection. Specifically, VDD was set to 7V, input to 5V and the program tunneling pin was gradually increased to 23V. Around 12 to 13V, the tunneling
node’s potential would start increasing. The coupled readout node’s potential would also increase. When the readout potential went over 4.5 V, electrons would start injecting into the readout floating gate, thus ensuring its potential was clamped below 5 V. After this initial programming, VDD was set to 5 V for the rest of the experiments.

4.3.3 Calibration

After one-time programming, input was set to 0 V, Vprog to 21.5 V for 1 minute and then the floating gate was allowed to discharge naturally. Readout voltages for the sensor and reference nodes were measured every 30 seconds, for 3 hours. The rate of discharge for each node was calculated; and a state where the tunneling rates would be equal was chosen as the initial synchronization point for the remainder of the experiments.

4.3.4 Initialization

Before the start of each experiment, floating gates were initialized to the initial synchronization point, estimated in the previous section. This was done by either setting the input to stable DC point through a digital to analog converter (DAC) or if the DAC value needed was beyond its output limit, then the potential would be increased by setting Vprog pin to 21 V.
4.3.5 Model derivation

FN tunneling current density $J_{\text{FN}}$ across a triangular barrier can be expressed as a function of the electric field $E$ across the barrier [90]:

$$J_{\text{FN}}(E) = \alpha E^2 \exp(-\beta/E) \quad (4.9)$$

where $\alpha$ and $\beta$ are process and device specific parameters [90].

Thus, for a tunneling junction with cross-sectional area $A$ and thickness $t_{\text{ox}}$, the tunneling current $I_{\text{FN}}$ for a time-varying voltage $V_t$ is given by

$$I_{\text{FN}}(V_t) = A\alpha(V_t/t_{\text{ox}})^2 \exp(-\beta t_{\text{ox}}/V_t). \quad (4.10)$$

Referring to the equivalent circuit in Fig. 4.3a, the dynamical system model when the sensing signal $x_t$ is absent is given by

$$I_{\text{FN}}(V_t) = -C_{\text{total}} \frac{dV_t}{dt} \quad (4.11)$$

where $C_{\text{total}} = C + C_{\text{in}}$ is the total capacitance at the floating-gate node. The solution of the equation can be expressed as:

$$V_t = \frac{k_2}{\log(k_1 t + k_0)} \quad (4.12)$$

where

$$k_1 = \frac{A\alpha \beta}{Ct_{\text{ox}}} \quad k_2 = \beta t_{\text{ox}} \quad (4.13)$$

depend on material properties and device structure, while
\[ k_0 = \exp \left( \frac{k_2}{V_0} \right) \]

depends on the initial conditions.

Now, let

\[ f(V_t) = -\frac{I(V_t)}{C_{\text{total}}} = -\frac{k_1}{k_2} V_t^2 \exp \left( -\frac{k_2}{V_t} \right) \]  \hspace{1cm} (4.14)

Desynchronization between the sensor and reference nodes shown in Fig. 4.3a occurs because of differences in rates of tunneling, which are caused by differences in electric potentials across the respective floating-gates

\[ \frac{dY_t}{dt} = \frac{I_{\text{FN}}(V_t^S)}{C_{\text{total}}} - \frac{I_{\text{FN}}(V_t^R)}{C_{\text{total}}} \]
\[ = f(V_t^R) - f(V_t^S) \]  \hspace{1cm} (4.15)

The reference node \( V_t^R \) follows the dynamics of equation 4.12 as it is not under the action of an external field. Thus, \( V_t^R = V_t \). The potential across the sensing node is given by how much it has desynchronized from the reference node \( (V_t^R - Y_t) \) and the effect of the external field, \( x_t \), through the input capacitor \( C_{\text{in}} \).

\[ V_t^S = V_t + C_R x_t - Y_t \]  \hspace{1cm} (4.16)

where \( C_R \) is the coupling ratio due to capacitive divider formed by \( C_{\text{in}} \) and \( C_{\text{fg}} \).

\[ C_R = \frac{C_{\text{in}}}{C_{\text{total}}}; C_{\text{total}} = C_{\text{in}} + C_{\text{FG1}} + C_C||C_{\text{FG2}} \]  \hspace{1cm} (4.17)
Substituting $V_t^R$ and $V_{rt}^R$ in equation 4.15

\[
\frac{dY_t}{dt} = f(V_t) - f(V_t + C_R x_t - Y_t)
\]  

(4.18)

The above equation is the constitutive differential equation and can be solved using numerical methods for any input signal. To obtain an explicit expression for estimating the response $Y_t$, we assume that $Y_t \ll V_t$ and $E(x_t) = 0$ for all $t$, and use Taylor series expansion with first order approximation.

\[
\frac{dY_t}{dt} = f(V_t) - f(V_t + C_R x_t) + \frac{d(f(V_t))}{dV_t} Y_t
\]

(4.19)

Multiplying both sides of equation 4.19 by $1/f(V_t)$, substituting $dV_t = f(V_t) dt$ (from equations 4.11 and 4.14) and simplifying:

\[
\frac{dY_t}{f(V_t) dt} - \frac{d(f(V_t))}{f(V_t)^2 dt} Y_t = \frac{1}{f(V_t)} (f(V_t) - f(V_t + C_R x_t))
\]

(4.20)

Integrating both sides with respect to $dt$ between the limits 0 and $T$:

\[
\frac{Y_T}{f(V_T)} - \frac{Y_0}{f(V_0)} = \int_0^T \left( 1 - \frac{f(V_t + C_R x_t)}{f(V_t)} \right) dt
\]

\[
\frac{Y_T}{f(V_T)} = \int_0^T \left( 1 - \frac{f(V_t + C_R x_t)}{f(V_t)} \right) dt
\]

(4.21)

\[
Y_T = f(V_T) \int_0^T \left( 1 - \frac{f(V_t + C_R x_t)}{f(V_t)} \right) dt
\]
Substituting $f(V_t)$ from equation 4.14 into equation 4.21

$$Y_T = \frac{k_1}{k_2} V_T^2 \exp\left(\frac{-k_2}{V_T} \right) \int_0^T \left[ \left(1 + \frac{C_R x_t}{V_t}\right)^2 \exp\left(\frac{k_2 C_R x_t}{V_t(V_t + C_R x_t)}\right) - 1 \right] dt \quad (4.22)$$
Chapter 5

FN Dynamic Adaptive Memory


In this chapter, I demonstrate that the dynamical nature of the FN Memory can be exploited to create a synaptic device which addresses the energy-efficiency imbalance between the training and the inference phases observed in neuromorphic learning systems. Our proposed synaptic element is adaptive with respect to its data retention capacity which can then be traded-off with respect to the energy-dissipation per update. This system could potentially be used for training a low-power classifier in the field. Another key thing to note is that the memory can be updated in either directions by the application of digital unipolar pulses as shown in Fig. 5.4 and Fig. 5.5.
5.1 Introduction

Implementation of reliable and scalable synaptic weights or memory remains an unresolved challenge in the design of energy-efficient machine learning (ML) and neuromorphic processors [19]. Ideally, the synaptic weights should be analog and should be implemented on a non-volatile, easily modifiable storage device [153]. Furthermore, if these memory elements are integrated in proximity with the computing circuits or processing elements, then the resulting compute-in-memory (CIM) architecture [134, 152] has the potential to mitigate the memory wall [66, 70, 113] which refers to the energy-efficiency bottleneck in ML processors that arises due to repeated memory access. In most practical and scalable implementations, the processing elements are implemented using CMOS circuits; as a result, it is desirable that the analog synaptic weights be implemented using a CMOS-compatible technology. In literature, several multi-level non-volatile memory devices have been proposed for implementing analog synapses. These include the cross-bar memristor based resistive random-access memories (RRAM) [3], magnetic random-access memories (MRAM) [140], Phase Change Memory (PCM) [18], Spin Torque Transfer RAM (STTRAM) [79], Conductive Bridge RAM [74] or the three terminal devices like the floating-gate transistors [107], ferroelectric field-effect transistor-based RAM (FeRAM) [44], Charge Trap Memory [55] and Electrochemical RAMs (ECRAM) [142]. In all of these devices the analog memory states are static in nature, where each of the states needs to be separated from others by an energy barrier $\Delta E$. For example, in memristive devices the state of the conductive filament between two electrodes determines the stored analog value, whereas in charge-based devices like floating-gates or FeFET, the state of polarization determines the analog value. To ensure non-volatile storage, it is critical that the energy-barrier $\Delta E$ is chosen to be large enough to prevent memory leakage due to thermal-fluctuations or other environmental disturbances. However, the height of the energy
barrier $\Delta E$ also sets the fundamental limit on the energy dissipated to switch between different analog storage states. For example, switching the RRAM memory state requires 100 fJ per bit [154], whereas STT-MRAM requires about 4.5pJ per bit [40]. A learning/training algorithm that adapts the stored weights in quantized steps $(\ldots, W_{n-1}, W_n, W_{n+1}, \ldots)$ so as to minimize a loss-function $L(W)$, as shown in Fig. 1(a), has to dissipate a minimum energy of $(\ldots, \Delta E_{n-1}, \Delta E_n, \Delta E_{n+1}, \ldots)$ for memory updates. Separating the static states by an energy-barrier also allows the learning algorithm to precisely control the parameter retention time (parameter leakage) between subsequent parameter updates, however, this mode of updates do not exploit the physics of learning to optimize for energy-efficiency. In many energy-efficient ML training formulations, and in particular analog ML systems, the loss-function $L(W)$ is represented by an equivalent energy-functional of a physical ML system and learning/training involves a natural evolution of the system dynamics towards the minimum energy (optimal) state based on input stimuli (or equivalently training data). Thus, the physics of the system evolution process selects the minimum energy path towards the desired optimum. A synaptic element that is matched to this system dynamics needs to adaptive with respect to its memory retention time which can then be traded-off with respect to the energy-dissipation per update. In this paper we present such a synaptic element that uses dynamical states (instead of static states) to implement analog memory and is matched to the dynamics of ML training. The core of the proposed device is itself a micro-dynamical system and the system-level learning/training process modulates the dynamical state (or state trajectory) of the memory ensembles. The concept is illustrated in Fig. 1(b), which shows a reference ensemble trajectory that continuously decays towards a zero vector without the presence of any external modulation. However, during the process of learning, the trajectory of the memory ensemble is pushed towards an optimal solution $W^*$. The main premise of this paper is that the extrinsic energy $(\ldots, \Delta E_{n-1}, \Delta E_n, \Delta$
$\mathbf{E}_{n+1, \ldots}$ required for modulation, if matched to the dynamics of learning, could reduce the energy-budget for ML training. This is illustrated in Fig. 1(c) which shows a convergence plot corresponding to a typical ML system as it transitions from a training phase to an inference phase. During the training phase, the synaptic weights are adapted based on some learning criterion whereas in the inference phase the synaptic weights remain fixed or are adapted intermittently to account for changes in the operating conditions. Generally, during the training phase the amount of weight updates is significantly higher than in the inference phase, as a result, memory update operations require a significant amount of energy. Take for example support-vector machine (SVM) training, the number of weight updates scale quadratically with the number of support vectors and the size of the training data, whereas adapting the SVM during inference only scales linearly with the number of support-vectors [52]. Thus, for a constant energy dissipation per update, the total energy-dissipated due to weight updates is significantly higher in training than during inference. However, if the energy-budget per weight updates could follow a temporal profile as shown in Fig.1c, wherein the energy dissipation is no longer constant, but inversely proportional to the expected weight update rate, then the total energy dissipated during training could be significantly reduced. One way to reduce the weight update or memory write energy budget is to trade-off the weight’s retention rate according to the profile shown in Fig. 1c. During the training phase, the synaptic element can tolerate lower retention rates or parameter leakage because this physical process could be matched to the process of weight decay or regularization, techniques commonly used in ML algorithms to achieve better generalization performance [93]. As shown in Fig. 1c, the memory’s retention rate should increase as the training progresses such that at convergence or in the inference phase the weights are stored on a non-volatile memory.
Figure 5.1: Operating principle and architecture of the proposed dynamically adaptive memory. Motivation and principle of operation for the proposed synaptic memory device: (a) conventional non-volatile analog memory where transition between analog static states dissipates energy; (b) Dynamic analog memory where an external energy is used to modulate the trajectory of the memory states towards the optimal solution; (c) desired analog synapse characteristic where the memory retention rate is traded-off with the write energy; reducing the energy dissipation per weight update in training phase by matching the dynamics of the dynamic analog memory to the weight decay; (d) micrograph of a fabricated DAM array along with (e) its equivalent circuit where the leakage current $I_{FN}$ is implemented by (f) the electron transport across a Fowler-Nordheim (FN) tunneling barrier; (g) Implementation of the FN tunneling based DAM where dynamic states g1-g3 determines the energy dissipated per memory update and memory retention rate.
In this paper we describe a dynamic analog memory (DAM) that can exhibit a temporal profile similar to that of Fig. 1c. Furthermore, the memory is implemented on a standard CMOS process without the need for any additional processing layers. Fig. 1e shows a micrograph of a DAM array and in chapter 4 we described the circuit implementation details. The proposed DAM requires a Fowler-Nordheim (FN) quantum-tunneling barrier which can be created by injecting sufficient electrons onto a polysilicon island (floating-gate) that is electrically isolated by thin silicon-di-oxide barriers [90]. As the electron tunnels through the triangular barrier, as shown in Fig. 1f, the barrier profile changes which further inhibits the tunneling of electrons. We have previously shown that the dynamics of this simple system is robust enough to implement time-keeping devices [160] and self-powered sensors [101].

In this paper, we use a pair of synchronized FN-dynamical systems to implement a DAM suitable for implementing ML training/inference engines. Figure 1(f) shows the dynamics of two FN-dynamical systems, labeled as SET and RESET, whose analog states continuously and synchronously decay with respect to time. In our previous work [101], we have shown the dynamics across different FN-dynamical systems can be synchronized with respect to each other with an accuracy greater than 99.9% . However, when an external voltage pulse modulates the SET system, as shown in Fig. 1f, the dynamics of the SET system becomes desynchronized with respect to the RESET system. The degree of desynchronization is a function of the state of the memory at different time instances (Fig. 1g, insets g1-g3) which determines the memory’s retention rate. For instance, at time-instant $t_1$, a small magnitude pulse would produce the same degree of desynchronization as a large magnitude pulse at the time-instant $t_3$. However, at $t_1$ the pair of desynchronized systems (SET and RESET) would resynchronize more rapidly as compared to desynchronized systems at time-instants $t_2$ or $t_3$. This resynchronization effect results in shorter data retention; however, this feature could be leveraged to implement weight-decay in ML training. At time-instant $t_3$, the
Figure 5.2: a) $W_S$ (solid line) and $W_R$ (dashed line) response at 3 different operating conditions (zoomed insets: a1, a2, a3). b-d) FN-DAM response ($w$) calculated as difference between $W_S$ and $W_R$ voltage values.

resynchronization effect is weak enough that the FN-dynamical system acts as a persistent non-volatile memory with high data-retention time. In addition, we describe how the FN-dynamical system mathematical model can be matched to ML training formulation. The model shows that the voltage or energy required for updating the memory can be annealed according to the profile shown in Fig. 1c.
5.2 Results

5.2.1 Dynamic analog memory with an asymptotic nonvolatile storage

The dynamics of the FN-tunneling based DAM (or FN-DAM) were verified using prototypes fabricated in a standard CMOS process (micrograph shown in Fig. 1e). The FN-DAM devices were programmed and initialized through a combination of FN tunneling and hot electron injection. Detailed description of the general programming process can be found in Chapter 4. The tunneling nodes (WS and WR in Fig. 1e) were initialized to around 8 V and decoupled from the readout node by a decoupling capacitor to the sense buffers (shown in Fig. 4.9). The readout nodes were biased at a lower voltage (∼3 V) to prevent hot electron injection [24] onto the floating gate during readout operation. Fig. 2 shows the measured dynamics of the FN-DAM device in different initialization regimes used in ML training, as described in Fig. 1c. The different regimes were obtained by initializing the tunneling nodes (WS and WR) to different voltages (see Methods section), whilst ensuring that the tunneling rates on the WS and WR nodes were equal. Initially (during the training phase), tunneling-node voltages were biased high (readout node voltage of 3.1 V), leading to faster FN tunneling (Fig. 2, inset a). A square input pulse of 100 mV magnitude and 500 ms duration (5 fJ of energy) was found to be sufficient to desynchronize the SET node by 1 mV. This desynchronization, \( w = (W_S - W_R) \), stores the state of the dynamical analog memory. However, as shown in Fig. 2(b), the rate of resynchronization in this regime is high leading to a decay in the stored weight down to 30% in 40 s. At \( t = 90 \) s, the voltage at node WS has reduced (readout node voltage of 2.9 V), and a larger voltage amplitude (500 mV) is required to achieve the same desynchronization magnitude of 1 mV, corresponding to an
energy expenditure of 125 fJ. However, as shown in Fig. 2(c), the rate of resynchronization is low in this regime, leading to a decay in the stored weight down to 70% its value in 40 s. Similarly, at a later time instant \( t = 540 \) s, a 1 V signal desynchronizes the recorder by 1 mV, and as shown in Fig. 2(d), in this regime 95% of the stored weight value is retained after 40 s. This mode of operation is suitable during the inference phase of machine learning when the weights have already been trained, but the models need to be sporadically adapted to account for statistical drifts. Modeling studies described in Appendix Fig. 4.6 shows that the write energy per update starts from as low as 5 fJ and increases to 2.5 pJ over a period of a period of 12 days, while Appendix Fig. 5.3 indicates that at greater instants of time the memory has retention times similar to other FLASH based memory.

Each DAM in the FN-DAM device was programmed by independently modulating the SET and RESET junctions shown in Fig. 1(e). The corresponding \( W_S \) and \( W_R \) nodes were initially synchronized with respect to each other. After a programming pulse was applied to the SET or RESET control gate, the difference between the voltages at the \( W_S \) and \( W_R \) nodes
were measured using an array of sense buffers. In results shown in Fig. 3a-d, a sequence of 100 ms SET and RESET pulses were applied. The measured difference between the voltages at the $W_S$ and $W_R$ nodes indicates the state of the memory. Each SET pulse increases the state while a RESET pulse decreases the state. In this way, the FN-device can implement a DAM that is bidirectionally programmable with unipolar pulses. Fig. 3d also shows the cumulative nature of the FN-DAM updates which implies that the device can work as an incremental/decremental counter. Fig. 3e-f show measurement results which demonstrate the resolution at which a FN-DAM can be programmed as an analog memory. The analog state can be updated by applying digital pulses of varying frequency and variable number of pulses. In Fig. 3e, four cases of applying a 3 V SET signal for a total of 100 ms are shown: a single 100 ms pulse; two 50 ms pulses; four 25 ms pulses; and eight 12.5 ms pulses. The results show the net change in the stored weight was consistent across the 4 cases. A higher frequency leads to a finer control of the analog memory updates. Note that any variations across the devices can be calibrated or mitigated by using an appropriate learning algorithm [25]. The variations could also be reduced by using careful layout techniques and precise timing of the control signals.

5.2.2 Characterization of FN-DAM

The FN-DAM device can be programmed by changing the magnitude of the SET/RESET pulse or its duration (equivalently number of pulses of fixed duration). Fig. 4a shows response when the magnitude of the SET and RESET input signals varies from 4.1 V to 4.5 V. The measured response shown in Fig. 4a shows an exponential relationship with the amplitude of the signal. When short-duration pulses are used for programming, the stored value varies linearly with the number of pulses, as shown in Fig. 4b. However, repeated application of
Figure 5.4: (a-b) SET and RESET input sequence. c) Change in $W_S$ and $W_R$ potentials due to SET and RESET pulses. d) DAM response calculated as difference between $W_S$ and $W_R$ voltages. e-f) FN-DAM response to SET pulses of varying frequency. Error bars indicate standard deviation estimated across 12 devices.

Pulses with constant magnitude produce successively smaller change in programmed value due to the dynamics of the DAM device (Fig. 4a). One way to achieve a constant response is to pre-compensate the SET/RESET control voltages such that a target voltage difference $w = (W_S - W_R)$ can be realized. The differential architecture increases the device state robustness against disruptions from thermal fluctuations (Fig. 4d). The stored value on DAM devices will leak due to thermal-induced processes or due to trap-assisted tunneling. However, in DAM, the weight is stored as difference in the voltages corresponding to $W_S$ and $W_R$ tunneling junctions which are similarly affected by temperature fluctuations. To verify this, we exposed the FN-DAM device to temperature ranging from 5 – 40 °C. Fig. 4d shows that the DAM response is robust to temperature variation and the amount of desynchronization for a single recorder never exceeds 20 mV. When responses from multiple FN-DAM devices are pooled together, the variation due to temperature further reduces.
**Figure 5.5:** Device characterization: (a) Change in DAM response with each pulse of same magnitude and duration. b) DAM response to varying number of pulses. c) DAM response to pulses of different magnitude. d) device state drift due to temperature variations after 1, 2 and 3 hours.
5.2.3 FN-DAM based Co-design of Classifiers and Neural Networks

In this section we experimentally demonstrate the benefits of exploiting the dynamics of FN-DAM weights when training a simple linear classifier. For this results, two FN-DAM devices were independently programmed according to the perceptron training rule [26]. We trained the weights of a perceptron model to classify a linearly separable dataset comprises 50 instances of two-dimensional vectors, shown in Fig. 5a. During each epoch, the network loss function and gradients were evaluated for every training point in a randomized order, with time interval between successive training points being two seconds. Fig. 5b shows that after training for 5 epochs, the learned boundary can correctly classify the given data. Fig. 5c shows the evolution of weights as a function of time. As can be noted in the figure, initially the magnitude of weight updates (negative of the cost function gradient) was high for the first 50 seconds, after which the weights stabilized and required smaller updates. The energy consumption of the training algorithm can be estimated based on the magnitude and number of the SET/RESET pulses required to carry out the required update for each misclassified point. As the SET/RESET nodes evolve in time, they require larger voltages for carrying out updates, shown in Fig. 5d. The gradient magnitude was mapped onto an equivalent number of 1 kHz pulses, rounding to the nearest integer. Fig. 5e shows the energy (per unit capacitance) required to carry out the weight update whenever a point was misclassified. Though the total magnitude of weight update decreased with each epoch, the energy required to carry out the updates had lower variation (Fig. 5f). The relatively larger energy required for smaller weight updates at later epochs led to longer retention times of the weights. The FN-DAM dynamical model summarized in the Methods section can be used to evaluate the energy-efficiency gains that can be obtained by co-designing a large-scale neural network training engine using FN-DAMs. The result of the co-design is shown in Fig. 5(g-h)
Figure 5.6: Synaptic memory for neuromorphic applications a) Test data set with randomly initialized decision boundary b) Decision boundary after training. c) Evolution of weights after 5 epochs. d) Input voltage required for initiating a unit change in weight. e) Energy expended in updating the weights. f) Average magnitude of weight update and average energy required for each epoch. g) Energy spent in updating the network weights for 3 types of network models. Inset shows same data with X axis in log scale. h) Network loss for 3 types of network models. Inset shows same data with X axis in log scale.
where we show that an FN-DAM based deep neural network (DNN) can achieve similar classification accuracy as a conventional DNN (Table 5.1) while dissipating significantly less energy during training. Note that for this demonstration, only the fully connected layers were trained while the feature layers were kept static. This mode of training is common for many practical DNN implementations on edge computing platforms where the goal is not only to improve the energy-efficiency of inference but also for training [135].

<table>
<thead>
<tr>
<th>Model/Accuracy (%)</th>
<th>After 9 epochs</th>
<th>After 10 epochs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard CNN</td>
<td>98.9</td>
<td>98.9</td>
</tr>
<tr>
<td>FN-DAM CNN</td>
<td>98.9</td>
<td>99.2</td>
</tr>
<tr>
<td>FN-DAM CNN with mismatch</td>
<td>97.9</td>
<td>97.3</td>
</tr>
</tbody>
</table>

5.3 Discussions

In this paper we reported a Fowler-Nordheim quantum tunneling based dynamic analog memory (FN-DAM) whose physical dynamics can be matched to the dynamics of weight updates used in machine learning (ML) or neural network training. During the training phase, the weights stored on FN-DAM are plastic in nature and decay according to a learning-rate evolution that is necessary for the convergence of gradient-descent training [112]. As the training phase transitions to an inference phase, the FN-DAM acts as a non-volatile memory. As a result, the trained weights are persistently stored without requiring any additional refresh steps (used in volatile embedded DRAM architectures [147]). The plasticity of FN-DAM during the training phase can be traded off with the energy-required to update the weights. This is important because the number of weight updates during training scale
quadratically with the number of parameters, hence the energy-budget during training is significantly higher than the energy-budget for inference. The dynamics of FN-DAM bears similarity to the process of annealing used in neural network training and other stochastic optimization engines to overcome local minima artifacts [29]. Thus, it is possible that FN-DAM implementations or ML processors can naturally implement annealing without dissipating any additional energy. If such dynamics were to be emulated on other analog memories, it would require additional hardware and control circuitry.

Several challenges exist in scaling the FN-DAM to large neural-networks. Training a large-scale neural network could take days to months [139] depending on the complexity of the problem, complexity of the network, and the size of the training data. This implies that the FN-DAM dynamics need to match the long training durations as well. Fortunately, the $1/\log$ characteristics of FN devices ensures that the dynamics could last for durations greater than a year [161]. The other challenge that might limit the scaling of FN-DAM to large neural network is the measurement precision. The resolution of the measurement and the read-out circuits limit the energy-dissipated during memory access and how fast the gradients can be computed (Appendix Fig. A.4). For instance, a 1 pF floating-gate capacitance can be initialized to store $10^7$ electrons. Even if one were able to measure the change in synaptic weights for every electron tunneling event, the read-out circuits would need to discriminate 100 nV changes. A more realistic scenario would be measuring the change in voltage after 1000 electron tunneling events which would imply measuring 100 $\mu$V changes. However, this will reduce the resolution of the stored weights/updates to 14 bits. This resolution might be sufficient for training a medium sized neural network; however, it is still an open question if this resolution would be sufficient for training large-scale networks [31, 57]. A mechanism to improve the dynamic range and the measurement resolution is to use a current-mode readout integrated with current-mode neural network architecture. If the read-out transistor is biased
in weak-inversion, 120 dB of dynamic range could be potentially achieved. However, note that even in this operating mode, the resolution of the weight would still be limited by the number of electrons and the quantization due to electron transport. Addressing this limitation would be a part of future research.

Another limitation that arises due to finite number of electrons stored on the floating-gate and transported across the tunneling barrier during SET and RESET, is the speed of programming. Shorter duration programming pulses would reduce the change in stored voltage (weight) which could be beneficial if precision in updates is desired. In contrast, by increasing the magnitude of the programming pulses, as shown in Fig. 4(a), the change in stored voltage can be coarsely adjusted. However, this would limit the number of updates before the weights saturate. Note that due to device mismatch the programmed values would be different on different FN-DAM devices.

In terms of endurance, after a single initialization the FN-DAM can support $10^3$–$10^4$ update cycles before the weight saturates. However, at the core FN-DAM is a FLASH technology and could potentially be reinitialized again. Given that the endurance of FLASH memory is $10^3$ [54], it is anticipated that FN-DAM to have an endurance of $10^6$–$10^7$ cycles. Also, FN-DAM when biased as a non-volatile memory requires on-chip charge-pumps only to generate high-voltage programming pulses for infrequent global erase; thus, compared to FLASH memory, FN-DAM should have fewer failure modes [24].

The main advantage of FN-DAM compared to other emerging memory technologies is its scalability and compatibility with CMOS. At its core, FN-DAM is based on floating-gate memories which have been extensively studied in context of machine learning architectures [13]. Furthermore, from an equivalent circuit point of view, FN-DAM could be viewed as a
capacitor whose charge can be precisely programmed using CMOS processing elements. FN-DAM also provides a balance between weight-updates that are not too small so that learning never occurs versus weight-updates being too large such that the learning becomes unstable. The physics of FN-DAM ensures that weight decay (in the absence of any updates) towards a zero vector (due to resynchronization) which is important for neural network generalization [151].

Like other analog non-volatile memories, FN-DAM could be used in any previously proposed compute-in-memory (CIM) architectures. However, in conventional CIM implementations the weights are trained offline and then downloaded on chip without retraining the processor [26]. This makes the architecture prone to analog artifacts like offsets, mismatch and non-linearities. On-chip learning and training mitigates this problem whereby the weights self-calibrate for the artifacts to produce the desired output [6]. However, to support on-chip training/learning, weights need to be updated at a precision greater than 12 bits [57]. In this regard FN-DAM exhibit a significant advantage compared to other analog memories. Even though in this proof-of-concept work, we have a used a hybrid chip-in-the-loop training paradigm, it is anticipated that in the future the training circuits and FN-DAM modules could be integrated together on-chip.

From a neuromorphic point of view, FN-DAMs could be used to mimic network level synaptic adaptation or pruning which plays a pivotal role in determining the optimal network configuration during the process of learning. For instance, it has been reported that a child’s brain has significantly denser connectivity than an adult brain [84] and consumes 50% of the body’s resting energy metabolism (BMR). Years of learning and synaptic pruning produces a network that tends towards optimality in terms of both energy and performance in
adulthood, when the brain accounts for only 20% of the BMR [84]. The adaptability of the proposed FN-DAM could be used to mimic this effect in artificial machine learning systems.

If the FN-DAM updates are driven by a constant voltage pulses (or fixed energy pulses like spikes) then the memory could be used to emulate the ageing effects in synaptic plasticity that is observed in neurobiological systems [17]. Like biological synapses, the relative change in value stored on FN-DAM or synaptic efficacy reduces with time for the same magnitude of applied input voltage pulses (or stimuli). Exploiting this feature of the FN-DAM to mimic neurobiologically relevant synaptic dynamics in artificial neural networks would also be a topic of future research.

5.4 Methods

Weight decay model and FN-DAM dynamics

Many neural network training algorithms are based on solving an optimization problem of the form [26]:

\[
\min_{\overline{w}} H (\overline{w}) = \frac{\alpha}{2} \| \overline{w} \| + L (\overline{w})
\]  

(5.1)

where \( \overline{w} \) denotes the network synaptic weights, \( L (\cdot) \) is a loss-function based on the training set and \( \alpha \) is a hyper-parameter that controls the effect of the \( L_2 \) regularization. Applying gradient descent updates on each element \( w_i \) of the weight vector \( \overline{w} \) as:
\[ w_{i,n+1} - w_{i,n} = -\alpha \eta_n w_{i,n} - \eta_n \frac{\delta L(w)}{\delta w_{i,n}} \]  

(5.2)

Where the learning rate \( \eta_n \) is chosen to vary according to \( \eta_n \sim O(1/n) \) to ensure convergence to a local minimum [28]: The naturally implemented weight decay dynamics in FN-DAM devices can be modeled by applying Kirchhoff’s Current Law at the SET and RESET floating gate nodes (see Fig. 1e).

\[ C_T \frac{d}{dt} (W_S) + I_{FN} (W_S) = C_C \frac{d}{dt} (V_{SET}) \]  

(5.3)

\[ C_T \frac{d}{dt} (W_R) + I_{FN} (W_R) = C_C \frac{d}{dt} (V_{RESET}) \]  

(5.4)

Where \( C_{FG} + C_C = C_T \) is the total capacitance at the floating gate. Taking the difference between the above two equations, we get:

\[ C_T \frac{d}{dt} (W_S - W_R) + I_{FN} (W_S) - I_{FN} (W_R) = C_C \frac{d}{dt} (V_{SET} - V_{RESET}) \]  

(5.5)

For the differential architecture, \( w = W_S - W_R \). Let \( V_{train} = V_{SET} - V_{RESET} \), the training voltage calculated by the training algorithm. In addition, \( I_{FN} \) is substituted from Eqn. 2. Let \( C_C/C_T = C_R \), the input coupling ratio:

\[ \frac{dw}{dt} = -\frac{(I_{FN} (W_S) - I_{FN} (W_R))}{C_T} + C_R \frac{d}{dt} (V_{train}) \]  

(5.6)
\[
\frac{dw}{dt} = -\left(\frac{k_1}{k_2}\right) W_R^2 \exp\left(-\frac{k_2}{W_R}\right) + \left(\frac{k_1}{k_2}\right) W_S^2 \exp\left(-\frac{k_2}{W_S}\right) \frac{W_R - W_S}{w} + C_R \frac{d}{dt} (V_{\text{train}}) \quad (5.7)
\]

Discretizing the update for a small time-interval \(\Delta t\)

\[
w_{n+1} = w_n + \left(-\frac{k_1}{k_2}\right) W_R^2 \exp\left(-\frac{k_2}{W_R}\right) + \left(\frac{k_1}{k_2}\right) W_S^2 \exp\left(-\frac{k_2}{W_S}\right) \frac{W_R - W_S}{w} \Delta t + C_R \Delta V_{\text{train},n} \quad (5.8)
\]

Let \(\mu = \frac{W_R}{W_S}\)

\[
w_{n+1} = w_n - \left(\frac{k_1}{k_2}\right) W_S \exp\left(-\frac{k_2}{W_S}\right) \mu^2 \exp\left(-\frac{k_2}{W_S} \left(1 - \frac{1}{\mu}\right)\right) - 1 \frac{w_n \Delta t + C_R \Delta V_{\text{train},n}}{\mu - 1} \quad (5.9)
\]

Assuming that the stored weight (measured in mV) is much smaller than node potential (> 6V) i.e., \(w \ll W_R\) (and \(W_R \approx W_S\)) and taking the limit \((\mu \to 1)\) using L'Hôpital's rule:

\[
w_{n+1} = \left(1 - \left(\frac{k_1}{k_2}\right) (2W_S + k_2) \exp\left(-\frac{k_2}{W_S}\right) \Delta t\right) w_n + C_R \Delta V_{\text{train},n} \quad (5.10)
\]

\(W_S\) follows the temporal dynamics given in Eqn. 1,

\[
w_{n+1} = -k_1 \left(\frac{2}{\log (k_1 n \Delta t + k_0)} + 1\right) \left(\frac{1}{k_1 n \Delta t + k_0}\right) w_n \Delta t + C_R \Delta V_{\text{train},n} \quad (5.11)
\]
Comparing above equation to Eqn. 4, the weight decay factor for FN-DAM system is given as:

$$\alpha \eta_n = k_1 \left( \frac{2}{\log (k_1 n \Delta t + k_0)} + 1 \right) \left( \frac{1}{k_1 n \Delta t + k_0} \right) \to O \left( \frac{1}{n} \right) \quad (5.12)$$

**Chip-in-the-loop linear classifier training**

A hybrid hardware-software system was implemented to carry out an online machine learning task. The physical weights \((\overline{w} = [w_1, w_2])\) stored in two FN-DAM devices were measured and used to classify points from a labelled test data set in software. We sought to train a linear decision boundary of the form:

$$f (\overline{x}, \overline{w}) = x_2 + w_1 x_1 + w_0 \quad (5.13)$$

\(\overline{x} = [x_1, x_2]\) are the features of the training set. For each point that was misclassified, the error in the classification was calculated and a gradient of the loss function with respect to the weights was calculated. Based on the gradient information, the weights were updated in hardware by application of SET and RESET pulses via a function generator. The states of the SET and RESET nodes were measured every 2 seconds and the weight of each memory cell, \(i\), was calculated as:

$$w_i = 1000 \times (W_{R,i} - W_{S,i}) \quad (5.14)$$
The factor of 1000 indicates that the weight is stored as the potential difference between the SET and RESET nodes as measured in mV. We followed a stochastic gradient descent method. We defined loss function as:

\[ L_n(w) = ReLU(1 - y_n f(x_n, w)) \]  \hspace{1cm} (5.15)

The gradient of the loss function was calculated as:

\[ G_n(w) = \frac{\partial L_n(w)}{\partial w} \]  \hspace{1cm} (5.16)

The weights needed to be updated as

\[ w_{n+1} = w_n - \lambda_n G_n(w) \]  \hspace{1cm} (5.17)

Here \( \lambda_n \) is the learning rate as set by the learning algorithm. The gradient information is used to update FN-DAM by applying control pulses to SET/RESET nodes via a suitable mapping function \( T \):

\[ V_{\text{train},n} = T(\lambda_n G_n(w)) \]  \hspace{1cm} (5.18)

Positive weight updates were carried out by application of SET pulses and negative updates via RESET pulses. The magnitude of the update was implemented by modulating the number of input pulses.
5.4.1 FN-DAM based CNN Implementation

The performance of FG-DAM model was compared to that of a standard network model. A 15-layer convolutional neural network was trained on the MNIST dataset using the MATLAB Deep Learning Toolbox. For each learnable parameter in the CNN, a software FN-DAM instance corresponding to that parameter was created. In each iteration, the loss of the network function and gradients were calculated. The gradients were used to update the weights via Stochastic Gradient Descent with Momentum (SGDM) algorithm. The updated weights were mapped onto the FN-DAM array. The weights in the FN-DAM array were decayed according to Eqn. 14. These weights were then mapped back into the CNN. This learning process was carried on for 9 epochs. In the 10th epoch, no gradient updates were performed. However, the weights were allowed to decay for the last epoch (note that in the standard CNN case, the memory was static). A special case with a 0.1% randomly assigned mismatch in the floating gate parameters (\( k_1 \) and \( k_2 \)) was also implemented.
Chapter 6

Self-powered Analog Sensor Logger


This chapter describes a specialized application of the FN Memory described in 4. The action model derived in that chapter suggests that any arbitrary signal can be coupled to the dynamical system and it will act on the system according to Equation 4.5. Thus it will leave a trace of its entire history. According to Section 4.2.6, less than femtowatts of power is required to modulate the system, which is lower than the power levels of most signals of interest. In addition the non-linear nature of FN Memory (Fig. 4.4) means that even AC signals can be recorded. In this chapter, we show the FN memory response to arbitrary signals and experimentally show that such signals can be sensed and recorded without any external batteries.
6.1 Introduction

For sensing systems like IoT devices or biomedical implants that operate in resource-constrained settings, utilizing a battery may be impractical due to biocompatibility concerns, size constraints or due to technical challenges involved in replacing the battery. Self-powered sensors (SPS) can obviate the need for batteries by harvesting their operational energy directly from ambient sources, such as light [98] or mechanical vibrations [98]. SPS achieve this by first buffering ambient energy using standard power-conditioning techniques before activating the basic computation/sensing and sometimes telemetry functions [99, 145]. However, when the objective is to sense and compute a simple function, like the total signal energy or a cumulative “action”, an application specific but ultra-energy-efficient variant of SPS could be designed by combining the operational physics of signal transduction, rectification and non-volatile data storage.

One such SPS was reported in [25, 67] where a cumulative measure of mechanical activity was sensed, computed and directly stored on floating-gate memories [158]. Similar techniques could also be applied to other non-volatile technologies for sensing the event of interest as an equivalent change in magnetoresistance in MRAM [2], change polarization in FeRAM [10], or change in electrical conductance in memristor-type systems [34]. However, these approaches require power conditioning such as rectification or voltage-boosting to meet the activation thresholds and to initiate the non-volatile state-change. Operational limits arise due to rectification efficiency, and due to material properties that influence diode thresholds or leakage currents. Note that some energy harvesting systems report low voltage continuous operation (i.e. $<50 \text{ mV}$), however they require higher activation thresholds for initial start up conditions (e.g. $>600 \text{ mV}$) [71, 106, 118].
We propose a self-powered sensing system, where instead of harvesting the energy to switch between static memory states, the sensing signal is used for modulating a synchronized dynamic state. In this regard, dynamical systems, both natural and artificial, have been shown to store information in their dynamic states \([9,36,49]\). In this work, we show the feasibility of this approach for self-powered sensing and data-logging, but at chip-scale. This is illustrated in Fig. 4.1a which shows two synchronized globally asymptotically stable (G.A.S) dynamical systems; a sensing system and a reference system. A time-varying input signal modulates the state trajectories of the sensing dynamical system leading to its desynchronization with respect to the reference dynamical system. The relative degree of desynchronization between the two systems serves as a medium for sensing and storing the cumulative effect of the input modulation. While the principle is relatively straightforward, there exists two key challenges in implementing the proposed concept at a chip-scale. First, due to self-powering requirements, the synchronized G.A.S. dynamical system can only be implemented using leakage processes driven by intrinsic thermal or quantum transport of electrons. The simplest of such a system can be modeled by an equivalent circuit shown in Fig. 4.1b. The capacitor \(C\) in the circuit models the dynamical state (denoted by the time-varying voltage \(V_t\)) and the time-dependent system trajectory is determined by a leakage current \(I(V_t)\). The capacitor \(C_{in}\) couples the input signal \(x_t\) into the dynamical system. The challenge is that an ultra-low leakage current \(I(V_t)\) is required to ensure that the dynamical system is operational for the duration of sensing and data-logging. For instance, a 1V change across a 1 pF on-chip capacitor over a duration of 1 day would require a leakage current of 10 attoamperes. Even if it were possible to implement such low-leakage currents, it is difficult to ensure that the magnitude of the currents match across different devices to ensure state synchronization. The second challenge with regard to data-logging is that there exists a trade-off between the non-linearity in the dynamical systems response and the duration over which the information
can be retained. As shown in Appendix Fig. 4.2a–b, if a constant leakage element (for example reverse leakage current) is used, not only do the system trajectories rapidly converge to the final steady state, but the modulation signal does not cause a change in the sensing system trajectory with respect to the reference system trajectory. On the other hand, a resistive or a direct-tunneling leakage element will be sensitive to the changes in modulation signal but will be unable to keep the two trajectories separated for long periods of time, leading to low retention-time. In this report, we show that a differential G.A.S. dynamical system [105] implemented using a Fowler-Nordheim (FN) quantum tunneling device [160] can address all these challenges.

6.2 Results

6.2.1 A simple behavioral model explains the data-logging principle

In Chapter 4, I derived a tractable mathematical model for the data sensed and stored by the FN memory to an input signal $x_t$. In Chapter 5, the memory was updated using digital pulses. Here the memory is converted into an integrated sensor-data-logger which can be modulated by any arbitrary time-varying input signal $x_t$. As derived in 4, we the output of the data-logger $Y_T$ measured at time-instant $T$ can be expressed as

$$Y_T = R(T)A_x(T)$$  \hspace{1cm} (6.1)$$

where $A_x(T)$ represents the total “action” due to the input signal $x_t$ accumulated up to the time instant $T$ and $R(T)$ is a “forgetting” factor that is independent of the input signal $x_t$. $R(T)$ models the data retention capability and arises due to resynchronization of the sensor
and reference FN devices, after the sensor device is perturbed by $x_t$. In the Methods section, we show that the action $A_x(T)$ can be expressed in terms of device parameters as

$$A_x(T) = \frac{k_1}{k_2} V_0^2 \exp \left( -\frac{k_2}{V_0} \right) \int_0^T \left[ \left( 1 + \frac{C_R x_t}{V_t} \right)^2 \exp \left( \frac{k_2 C_R x_t}{V_t} \right) \right] dt$$

and the resynchronization term $R(T)$ can be expressed as

$$R(T) = \frac{V_T^2}{V_0^2} \exp \left( \frac{k_2}{V_0} - \frac{k_2}{V_T} \right).$$

Here $V_t$ is given by equation 4.1 with $V_0$ and $V_T$ representing the device voltage at time-instant $t = 0$ and $t = T$ seconds. The parameter $C_R$ in equation 6.2 models a capacitive divider that is formed due to the coupling of the input capacitance onto the floating-gate. (Figs. 6.1 and 6.2) show several examples of signals $x_t$ for which the first-order action model given by equation 6.1 accurately tracks a more computationally intensive ordinary differential equation (ODE) based device model. In Figure 6.3, we show the “action” $A_x(T)$ corresponding to different signal types with different magnitude and energy. The results show that $A_x(T)$ is monotonic with respect to energy and hence can be used as a measure of cumulative energy.

In our controlled experiments we subjected the FN data-logging device to a square pulse of varying magnitude but with a fixed duration of 120 seconds. This duration was chosen because it is sufficiently long enough to elicit a measurable response and for the purpose of device characterization. Also, the pulse was applied at a fixed time (1,800 seconds), after which the desynchronization $Y_T$ was measured at different values of measurement time $T$. Experiments were conducted over a duration of 10,800 seconds (3 hours), with the data-logger responses measured every 30 seconds. Each data-logger was calibrated to similar initial conditions for all experiments wherein the sensor and the reference nodes were initialized.
to equal tunneling rates. A typical experiment demonstrating the recorder in operation is shown in Fig. 4.4a, which matches the model described in the Methods section. The RMSE between the model and measured data is 61\(\mu\)V with an \(R^2\) of 0.9999.

Measurement results across three repeated trials for input signals of magnitude 100 mV and \(-100\) mV are shown in Fig. 4.4b. The 100 mV signal resulted in a sensor response of 0.8–1.5 mV for the three data-logging devices. At the end of three hours, due to resynchronization, the sensor response decreases down to 0.5–0.6 mV. For the \(-100\) mV input, responses after the modulation were in the range of \(-0.5\) to \(-0.9\) mV, which dropped to \(-0.2\) mV at the end of three hours. Though the three recorders had different responses, they were consistent across trials for the same recorder. The device responses at the end of three hours for input signals of different magnitudes are shown in Fig. 4.4c. From the figure, it is evident that the data-logging device response is similar to a rectifier as summarized by the action model in equation 6.2. The action model fits the data for this wide range of input conditions with an \(R^2\) of 0.9855.

6.2.2 Model validation

The assumptions made in the derivation have been validated against a general ODE solver (Figs. 6.1 and 6.2). As shown in the figure, the error was less than 10 \(\mu\)V for a response of 1.5 mV. Same analysis was run 100 times and the relative error was always less than 1 \%. The action model was computationally faster to solve than the ODE solver by a factor of 10\(^5\). The explicit action model led to large errors when the input signal was large (Fig. 6.2). The error arises due to assumption \(Y \ll V\), made during linearizing the equation 16, to estimate the resynchronization of the response. For large \(Y\), higher order terms can no longer be
ignored and the resynchronization will be faster. As the 1st order model ignores these terms, it always overestimates the expected action at time $T$. The error in the model can be empirically reduced by fitting a model between the expected response (as generated by the ODE solver) and response calculated by the action model (Fig. 6.2c).

**Figure 6.1:** Comparison between ODE solver and the action model for a) Short pulses and b) Small magnitude random signal. In either case, the action model closely tracks the ODE solver. The maximum error was less than $10\mu V$ for both cases.
Figure 6.2: Comparison between ODE solver and the action model. a) Harmonic input signal leads to continuous increase in action, the error increases as action increases. b) Large continuous random signals lead to large errors in the action model compared to the ODE solver. However there is a relationship between the action and the error. An empirical model was fit that modeled error as a function of action. c) Compensation with this model leads to higher accuracy.

Figure 6.3: Simulation results for recorder response to AC signals. a) Action induced by signals of different shapes as a function of amplitude of the signal. System is sensitive to biphasic signals because of the rectification induced by FN tunneling. b) Action induced by signals of different shapes as a function of energy of the signal.
6.2.3 Self-powered operation of the proposed device

The self-powered dynamical system created by FN tunneling leakage implies that the device can operate without any external power source, once initialized. We have verified this mode of operation by first disconnecting the sensor-data-logger from any power-supplies and then applying an external signal as an input. The experimental protocol and representative results are shown in Fig. 6.4a. Immediately after powering on the system, the output of the reference node was measured to be lower than the value predicted by the model given by equation 4.1. However, the measurement stabilized within 200 seconds and the output closely matched the model for the rest of the experiment, indicating that FN tunneling dynamics were conserved in the self-powering mode. Additionally, desynchronization between sensor and reference nodes was observed immediately after power was turned on, indicating that the external signal “acted” on the sensing node. Errors introduced during the stabilization period were consistent between the sensor and reference nodes — the differential architecture attenuated these errors. The magnitude of the response is an exponential function of the input signal magnitude (Fig. 6.4b) as predicted by the action model of equation 4.5. Similarly, the recorder was able to record the number (and thus the energy) of discrete pulses applied (Fig. 6.4c) in the power-off state. The mean absolute error between model and observed data was 0.7 mV, higher than the errors obtained for continuously-powered case (Fig. 6.4d).

6.2.4 Self-powered sensing of action due to ambient acceleration

In this section, we demonstrate the use of the proposed sensor-data-logger for battery-free sensing of ambient acceleration. We chose a piezoelectric transducer for sensing mechanical acceleration and for directly powering the sensor-data-logger device. Note that in this
Figure 6.4: Verification of the proposed device for self-powered sensing and datalogging. a) Power to the system is switched off at the 1 hour mark, and then turned back on at 2 hour mark. The input pulse is applied at the 1.5 hour mark for a duration of 120 s seconds. b-c) Output measured from the recorder when the power is ON and the comparison with the predicted model showing the process of desynchronization. d) Recorder responses for input signals over a range of amplitudes. Responses follow an exponential model, which can be modeled by the action model. e) Recorder responses for varying number of pulses (400 mV magnitude, 50 s duration each). f) Distribution of absolute errors between measured data and model predictions, for externally powered and self-powered cases estimated across all experiments.
regard, other transducers for e.g. photodiodes, RF antennas, thermocouples could also be directly interfaced to the FN data-logging device to create other self-powered sensing modalities. A schematic of the experimental setup is shown in Fig. 6.5a. A PVDF (polyvinylidene difluoride) cantilever [TE Connectivity’s Measurement Specialties MiniSense 100 Vibration Sensor with nominal resonant frequency – 75 Hz] was mounted on a benchtop vibration table [3B Scientific Vibration Generator - U56001] that is externally actuated by a function generator. The table was actuated at an off-resonant frequency of 72 Hz for a range of actuating amplitudes. We simultaneously measured acceleration using a 3-axis accelerometer [Adafruit LIS3DH accelerometer] to use as the ground truth. Results are shown in Fig. 6.5b–c. We observed significant responses for vibration signals down to an acceleration of 0.0052 g (0.05 m/sec²). For context, a refrigerator vibrates with an acceleration of around 0.1 m/sec² [126]. The expected maximum output power of the piezoelectric sensor is on the order of tens of nanowatts of which only a fraction is used by the recorder to store the information. In the final experiment, we electrically disconnected all power to the recording system at the 1 hour (3,600s) mark, actuated the vibration table at 1.5 hours (5,400 s) and reconnected the system at 2 hours (7,200 s) to readout the output of the data-logger. We observed vibration-induced desynchronization in this set of experiments as well, with the deviation as expected based on the earlier characterization tests.

6.2.5 Parametric analysis

Using modeling and simulations, we conducted parametric analysis for our system. Parameters $T$ (Fig. 4.7a) and $V_0$ ((Fig. 4.7b) are operational parameters that can be set at run time according to application requirements. $k_1$ (Fig. 4.7c) depends on the area of the tunneling junction and on the capacitance associated with the floating gate node. $k_1$ and $k_1$ (Fig. 4.7d)
Figure 6.5: Self-powered sensing and data-logging of mechanical acceleration. a) Experimental setup showing a piezoelectric (PVDF) transducer connected to the FN sensor-data-logger chipset. b) Logger response when 58.6 mg (0.57 m/sec^2) acceleration was applied to the piezo cantilever (gain of 6 V/g at 75 Hz resonant frequency) at 72 Hz for 100 sec. c) Recorder responses at different readout times for a range of input frequencies. All modulated responses were statistically different from the unmodulated case at all readout times. d) Recorder was powered off in the shaded region. During that time period, vibration table was actuated, which was recorded as evidenced by the recorder value when power supply was turned on.
are also influenced by the thickness of the insulating material and other material properties like the barrier height at the interface between the conductor and the insulator.

We performed simulation studies to characterize the input impedance of our sensor logger which can be modeled by the equivalent circuit shown in Fig. 4.8a. The DC input impedance of our system is on the order of $10^{18} \Omega$, since the input is connected to the gate of a MOSFET and the FN tunneling current is in the order of attoamperes. In this case, the impedance of the ESD protection diodes dominate the input impedance at DC frequency. We ignore this leakage path for our analysis and the high pass cut-off frequency is found to be at $10^{-5}$ Hz (Fig. 4.8b). At higher frequencies, the input capacitance and gate-to-substrate capacitance, along with input parasitic resistances create a low impedance path. However, this power is predominantly reactive in nature (Fig. 4.8c) and can be minimized with suitable source impedance matching. The power dissipated by the sensor-data-logger can be estimated as

$$P(\omega) = \Re \left( \frac{V^2}{2Z_{in}(\omega)} \right)$$

Assuming that the natural dynamics of the FN device lie less than the frequency range of 1mHz, the average power dissipated can be estimated over the signal bandwidth of 1 mHz to 1 kHz to be 0.05 aW. So for an event lasting 100 seconds, energy dissipated is on the order of 5 aJ.

### 6.2.6 Temporal dependence

Fig. 6.6 shows the weak dependence of the recorder output to the time of occurrence of events. Earlier events lead to larger desynchronization, but have more time to recover. Modeling studies show that the net result is that later events lead to a larger output at
Figure 6.6: Temporal dependence of recorder output to time of occurrence of an event. a) Recorder dynamics for events occurring at different times. b) Final output of three recorders, averaged over three trials, for events occurring at different times.

readout. However, the change in expected output was less than $10\mu V$, smaller than the errors arising due to measurement and operational desynchronization.

6.2.7 Simulation of intracellular neural recording

Fig. 6.7 shows the results from the integrated sensor-logger system when modulated by intracellular signals. At the onset of the stimulus, large firing activity in the neuron is sufficient to elicit a response in the self-powered system. The magnitude of response is proportional to the number of spikes for that stimulus. Thus the response is dependent on the tuning curve of the neuron for a specific stimulus. If multiple such systems were integrated with multiple cells, each with different tuning curves, then the ensemble response may give an indication of the identity of the stimulus. It should be noted that the hyperpolarization of the membrane potential causes a decrease in the sensor-logger response. Fig. 6.7c shows the ‘forgetting’ effect of the sensor-logger system due to the resynchronization between the dynamical systems (Eqn. 7.12). The system data would have to be retrieved before the response goes below the noise limit. Fig. 6.8 shows retrieved responses from 2 PNs for the
Figure 6.7: a) Intracellular recordings from two projection neurons in the locust antennal lobe for three different odors. Shaded area indicates delivery of odor stimulus b) Self-powered sensor output on being modulated by the input signal. c) Decay of memory as time elapses.

three odors across 5 trials measured after 1 hour. The responses for the different trials are clearly clustered according to the stimulus owing to the differentiated but reliable response of each PN to different odors.

6.3 Discussion

In this report, we proposed a novel method for designing an ultra-energy-efficient sensor-data-logging device, where the energy of the sensing signal is used to modulate the state trajectories of a synchronized dynamical system. We showed that a Fowler-Nordheim (FN)
quantum tunneling device [160] can be used to implement the proposed sensor-data-logger on a standard silicon process.

Our modeling study summarized in Figure 4.7 shows that there are multiple parameters, both operational and design parameters, that affect the retention time (or resynchronization) of the FN device. Change in any parameter that increases (decreases) the “action” of the signal, would also lead to faster (slower) resynchronization. Thus, its net effect on the system depends on the total duration for which the input signal was applied. The initial charge on the floating-gate and time to sample are operational parameters as they can be set at run-time, as required by the specific application. Larger time intervals allow the input signal to be integrated over a longer period of time, but it does not change the sensitivity to the signal. For a signal of given action, the measured value decreases as $T$ increases due to resynchronization (Fig. 4.7a). Initializing the device to a higher voltage leads to higher sensitivity but only up to a certain limit (Fig. 4.7b). The reason is that higher sensitivity also leads to faster resynchronization determined by design parameters $k_1$ and $k_2$. $k_1$ can be tuned by varying the area of tunneling junction and capacitance sizing. We found that
increasing the area or lowering the capacitance would increase the sensitivity of the system but only to a certain point (Fig. 4.7c). Beyond this point, the gains are only marginal, at the expense of a larger footprint. Moreover, the capacitance is a function of the tunneling junction area and thus the ratio of area to capacitance is bounded and depends on the permittivity of the insulator. Smaller oxide thickness would decrease $k_2$ and sharply increase the sensitivity (Fig. 4.7d). However, at these scales the effect of other processes like direct tunneling cannot be ignored. Exploring different materials could have significant impact on both $k_1$ and $k_2$, as they affect the parameters $\alpha$ and $\beta$ (equations 4.9 and 4.13). When the input signal is a single pulse, the time-of-occurrence of the pulse also plays a role in the measured response, as shown in Figure 6.6. However, this effect is weaker than that of other factors.

The desynchronization based approach reduces the energy budget required for data-logging, we estimate that the proposed device can operate at an energy budget lower than an attojoule while retaining the information for at least 3 hours. In standard analog sensor circuits, quiescent current is sourced from a power source for continuous operation. In the proposed device, the quiescent current is the FN quantum tunneling current, which is sourced from the pre-charged capacitor and ambient thermodynamic fluctuations. Hence, no external power source is required for operation. For modulating the sensor, energy is extracted from the signal being sensed. If the energy dissipated at the input signal source (due to finite source impedance) is ignored, the energy budget required to modulate the state of the FN device is less than $100 \text{ aJ}$. In practice, the energy from the source is spent on charging the capacitor, and for maintaining DC voltage at the source. For example, when the magnitude of the input signal is $100 \text{ mV}$, $15 \text{ fJ}$ is used for charging a $300 \text{ fF}$ input capacitor. The DC input impedance of the proposed device was measured to be greater than $10^{17} \Omega$; thus, the energy required to maintain a voltage potential of $100 \text{ mV}$ for 120 sec is less than $100 \text{ aJ}$. Many
signals of interest have power levels greater than this, and can provide sufficient energy for modulating the sensor, provided the system impedance is matched to the source.

However, for time-varying sensor signals, a more efficient power transfer or sensing is achieved as some energy stored in reactive elements like capacitors can be recovered. In Figure 4.8, an equivalent circuit model corresponding to the FN tunneling device is shown along with a simplified sensor-transducer equivalent model. We note that the input impedance of the recorder is predominantly capacitive and the only dissipative factor arising during sensing/data-logging is due to the FN tunneling current. The equivalent circuit model allows estimation of the power dissipated by a device that is excited by an arbitrary sensor signal. In 4.8, we showed that a broadband AC input signal with upper cutoff frequency of 1 kHz and amplitude of 100 mV has an estimated energy dissipation by the system of 5 aJ for an event lasting 100 seconds.

Using FN quantum tunneling to implement the dynamical system has some key advantages. Its stability allowed us to create a pair of synchronized devices which is compensated for environmental variations. Its predictability was used for modeling, and we were able to derive a recorder response model that matched experimental data with 98.8% accuracy. Its dynamics follow a $1/\log(t)$ characteristic, which yields a long operational life. The non-linear response leads to rectification of input signals and offers an opportunity for time stamping and reconstruction. A more rigorous and theoretic investigation into the use of dynamical systems for information reconstruction will be the topic of future research.

At its core, the proposed device consists of four capacitors and two transistors (4C-2T), and can be implemented on any standard CMOS process. The current design is a proof-of-concept and is not optimized for sensitivity or form factor. Modeling analysis in Figure 4.7 shows that both of these parameters can be improved by minimizing the capacitance, while maintaining
the capacitance ratio \((C_R)\). To achieve this, an optimum balance between the input capacitor, decoupling capacitor and parasitic capacitance at the poly-substrate tunneling junction needs to be obtained. Better matching of the sensor and reference nodes (tunneling junctions, capacitors and readout circuits) using advanced analog layout techniques should be able to reduce the operational noise in the recorder and thereby increase the data retention capacity. Readout and common-mode noise can be further reduced by implementing a low-noise on-chip instrumentation amplifier. Multiple units of independent recorders could be used to increase the SNR of the recordings.

Any passive sensor that is capable of transducing a physical signal into an electrical signal (voltage or current) can be interfaced with our system. These include piezoelectric transducers, photodiodes, radio-frequency antennas, thermocouples, triboelectric generators etc. Passive sensors like strain gauges, that do not directly produce electrical output but instead effect a change in resistance are not compatible with our system. Similarly, many chemical transducers like dopamine sensors that require an activation voltage (external biasing or power) are also not applicable for self-powered data-logging. However, chemical sensors like amperometric glucose sensor that have the ability to generate electrical charge during the process of sensing should work with our system. In addition, there can be practical issues in measuring certain types of signals. For example, the limited action generated by signals like neural action-potentials may not be measurable due to resynchronization and system-noise. Finally, the proposed recorder could be directly integrated with FET (field-effect transistor) based sensors \([13, 108, 146]\), which have been developed for a wide range of applications. As there are no extrinsic powering requirements, there is the potential of integrating these devices on “smart dust” platforms as well \([137, 149]\).
In conclusion, we have described a self-powered sensor-data-logger device that records a cumulative measure of the sensor signal intensity over its entire duration. To achieve this, we designed a pair of synchronized dynamical systems whose trajectories are modulated by an external signal. The modulation leaves its trace by desynchronizing one of the synchronized pairs. The total cumulative measure or action is stored as a dynamical state which is then measured at a later instant of time. The self-powered dynamical system was designed by exploiting the physics of Fowler-Nordheim quantum tunneling in floating gate transistors. We modeled the response of our system to an arbitrary signal and verified the model experimentally. We also demonstrated the self-powered sensing capabilities of our device by logging mechanical vibration signals produced by a small piezoelectric transducer, while being disconnected from any external power source.

Data availability

All the software and experimental data used for generating the figures have been deposited in a public repository (https://doi.org/10.6084/m9.figshare.12814592.v1) [103].
Chapter 7

System Integration

In this chapter, I investigate the feasibility of designing ultra-small and light-weight head-stages for recording neural activity in freely behaving insects which are used for autonomous cyborg sensing platform. We consider different types of neural activity recording platform for insects as shown in Fig. 7.1 and theoretically analyze their respective architectural trade-offs that affects the overall system energy-efficiency and hence their form-factors. For our analysis we consider the factors that determines the form-factor and energy-efficiency of the modules used for designing the head-stage. Specifically, we analyze the implication of integrating ultra-low-power non-volatile memory (NVM) on the recording platform. Instead of continuously transmitting data, the neural data is buffered onto the non-volatile memory and the data is transmitted only when the remote receiver is in-range. Our trade-off analysis is based on a Fowler-Nordheim quantum tunneling based non-volatile memory (NVM) technology that can be updated at energy-budgets less than 1pJ per event. We consider three neural recording architectures: (a) When the NVM is used to buffer compressed neural data before using a burst-mode wireless transmission mode; (b) When an embedded classifier is used to detect events and sparsified data is stored on NVM before wireless transmission; and (c) When the non-volatile memory is operated in a self-powered mode, where its write
Figure 7.1: a) Raw signal is logged b) Energy of the raw signal is logged. c) Classifier output is logged d) Statistics of the signal are logged without any batteries.

energy is harvested directly from the neural signals. Using the neural data collected from a locust insect model (Schistocerca americana), we highlight these trade-offs while describing the advantages and limitations of each of the approaches.
### 7.1 Analysis Framework

Studying the atlas of the locust brain [82] reveals that a system designed under $1\text{mm}^3$ should be implantable within the locust head. Such a system follows the trend of creating mm-scaled sensory ‘dust’ like motes [89, 116, 137, 157]. However, other ‘dust’ platforms relied on short range wireless power transfer and communication, which is infeasible for our application where we plan to send the locust out to a remote field. In this report, we investigate electronic circuits, systems and devices that would lead to lowest power operation while being suitable for our application. From the literature we identify state of the art sensing and computation techniques and circuitry to decrease the overall power consumption of our system. We identified that long range communication would be the most significant consumer of power in the system. Therefore, instead of continuously transmitting data, the neural data is buffered onto the non-volatile memory and the data is transmitted only when the remote receiver is in-range. Specifically, our trade-off analysis is based on a non-volatile memory technology using Fowler-Nordheim dynamic analog memory (FN-MEM) which can be updated at energy-budgets less than 1pJ per bit. We consider three neural recording architectures: (a) When the FN-MEM is used to buffer compressed neural data before using a burst-mode wireless transmission mode; (b) When an embedded classifier is used to detect events and sparsified data is stored on FN-MEM before wireless transmission; and (c) When the non-volatile memory is operated in a self-powered mode, where its write energy is harvested directly from the neural signals. Using the neural data collected from a locust insect model (Schistocerca americana), we highlight these trade-offs while describing the advantages and limitations of each of the approaches.
7.1.1 Energy sources

Battery energy density drastically decreases as size of the battery decreases [162]. Currently available microbatteries reach areal energy capacity of around $2mA \cdot h/cm^2$ with a discharge current of around $0.07mA/cm^2$ [68]. Reversibly stretchable batteries of similar capacities have also been created [111]. The discharge current of microbatteries is very low. For systems that require bursts of currents for tasks like RF transmission, they could incorporate a microsupercapacitor which has a lower energy density but a faster discharge current ($>3mA/cm^2$) [91]. In this study, we consider a microbattery with a constant areal capacity ($\rho_E$) of $2mA \cdot h/cm^2$. Limits on maximum current discharge are not taken into account.

7.1.2 Amplifier

Neural action potentials in the extracellular medium are on the order of tens to hundreds of microvolts, while noise in the medium is on the order of few microvolts. A low noise amplifier is essential to condition the signal for downstream electronics. Vast amount of research has been carried out in the field of low power amplifier design, as it is often the highest consumer of power in neural recording systems. The noise efficiency factor, $NEF$, is a figure of merit that describes the relationship between the amplifier current, the input referred noise ($v_n$) and the signal bandwidth.

$$NEF = v_{n,\text{rms}} \sqrt{\frac{2I}{\pi V_T 4kTBW_{\text{amp}}}}$$ (7.1)

Minimum theoretical limit of $NEF$ is 1, which denotes the thermal noise characteristics of a single CMOS transistor. Practical amplifiers contain a multitude of transistors and thus
their $NEF$ is always greater than 1. In literature, multiple designs have been reported with an $NEF$ between 2 and 3 $[63,123,150,156,163]$. As an example, in $[150]$, the reported amplifier had an input-referred noise of 3.06 µV rms while consuming 7.56µW of power @ 2.8$V_{DD}$, corresponding to an $NEF$ of 2.67. They also carried out a theoretical analysis to show that with the theoretical limit for their design was 2.02.

In $[63]$ a low power pre-amplifier was designed for neural recording applications by operating in an open-loop configuration with current reuse. The amplifier was AC-coupled, with a pass-band from 0.3 Hz to 4.7kHz and consumed 805nA from a 1.0V supply, corresponding to an $NEF$ of 1.8. In $[123]$, The total power consumption per channel was 9 µW, for an input referred noise of 3.2µ$V_{rms}$, achieving a measured $NEF$ of 1.94.

In this analysis, we have assumed $NEF = 2$, $A_{BW} = 2kHz$ and $v_n = 10µV$. From these parameters we get an $I_{dd} = 30nA$. Assuming $V_{dd} = 1.8µV$, power dissipated by the amplifier is given as:

$$P_{AMP} = 54nW$$

(7.2)

### 7.1.3 Energy extraction circuit

In systems neuroscience studies and brain computer interface applications, a spike sorting is performed after acquiring raw data signals. In spike sorting, different spikes are assigned to individual neurons. Spike sorting helps in studying dynamics like Hebbian at a neuronal resolution. However, it is a computationally expensive process and state of the art online sorting algorithms consume µW of power $[38]$. Instead of spike sorting, many other processing algorithms have been studied to reduce the data rate of raw neural data. In this study, we consider the energy of the signal at the recording electrode as a lower data rate for
neural signature. In [129], we had shown that energy signatures measured at the electrode sites carry sufficient information to discriminate between different stimuli. In [61], an energy extraction circuit for local field potential signals was presented. The circuit operated at 5nW at a bandwidth of 90Hz. To estimate the energy consumption for our circuit:

\[ P_{EE} = 5 \cdot A_{BW}/90 \]  \hspace{1cm} (7.3)

### 7.1.4 Analog to Digital Converter

We considered the ADC performance metrics from an exhaustive survey of state-of-the-art ADCs maintained by [110]. We selected the ADC described in [76] as it was used for neural recording and had an energy efficiency, \( E_{ADC} \), of 32pJ/conversion.

\[ P_{ADC} = E_{ADC} f_{ADC} \]  \hspace{1cm} (7.4)

The ADC sampling frequency \( f_{ADC} \) varies for different application scenarios. For whole wave recording it is equal to the amplifier bandwidth \( A_{BW} \), while for the energy based recording and classification scenarios, it is equal to the reduced sampling frequency \( f_{Energy} \).

### 7.1.5 Classifier

A real-time linear classifier is implemented that calculates the probability of an observation \( x \) belonging to a class \( C \) based on its Mahalanobis distance to the cluster mean, \( d(x, C) \)

\[ d(x, C) = (x - \mu_C)\Sigma^{-1}(x - \mu_C)^T \]  \hspace{1cm} (7.5)
The cluster means $\mu_k$ and the covariance matrix $\Sigma^{-1}$ are pre-calculated during the training phase and stored on chip. Based upon the relative distances to the labeled clusters, a probability is calculated for that observation to belong to each class $P(x|C)$. The probabilities are summed over a 5 second window ($T_{class}$) and if the sum is above a threshold, a classification event is asserted. A window is used because stimuli encountered by an insect are expected to last a few seconds while sporadic neural activity that might lead to a false positive can be ignored. The number of operations performed for $m$ recording channels and $k$ classification classes is given by:

$$P_{CLASS} = m^4kE_{MAC}f_{Energy}$$

where $E_{MAC}$ is the energy required to perform a multiply and accumulate operation.

### 7.1.6 Compressive sensing module

Compressive sensing (CS) is a technique that can perfectly recover signals sampled at a sub-Nyquist rate, provided that the signal is sparse on some basis $\Psi$. This is accomplished by taking projections of the signal on a basis incoherent to $\Psi$. It has been shown that a matrix generated randomly with independent identically distributed (i.i.d.) entries (for example, generated from Gaussian or Bernoulli distributions) will show a high level of incoherence with the fixed basis $\Psi$. In this regard, CS is a universal encoding scheme where the complexity of reconstruction is passed on to the decoder [23]. Detailed discussions on the theory and applications of CS can be found in [11, 23]. Neuroscience applications of CS have been explored in [35, 92, 136]. A primary motivation for these systems was to lower the power consumption of the system. For example, [48] reported a CS analog front end for EEG recording. It was a low bandwidth system which consumed only $1.8\mu W$ for 64 channels. In
a CS system for ECG reconstruction was reported which had lower energy consumption than the discrete wavelet transform and thus led to an increase in device lifetime by 40%. A detailed analysis of the energetics of different CS hardware systems revealed that it was more efficient to perform CS calculations in the digital domain [28]. Based upon their analysis, they created an EEG recording system that consumes only 1.9µW at 0.6 V for sub-20 kS/s sampling rates.

For feasibility analysis, we consider a compressive sensing system as implemented in [104], but with FN-MEM replacing the linear injectors. Similar to a floating gate linear-injector, FN-MEM can be programmed linearly as an analog accumulator. But power consumption of 1pJ/bit is significantly lower than that of an injector (10nJ/bit). A block based CS module is implemented here. For a block size of \( D \), a signal of length \( N = CR \times D \) is compressed by the factor \( CR \). During each clock cycle, on an average half of the blocks will be active. The power dissipated during this process is given by:

\[
P_{CS} = 0.5DE_{DFF}f_{CS}\tag{7.7}
\]

The compressive sensing clock rate \( f_{CS} \) varies for different application scenarios. For whole wave and energy based recording, it is equal to the ADC sampling rate \( (f_{ADC}) \), while for the classification scenario, it is equal to \( 1/T_{class} \).

### 7.1.7 FN-MEM

Fowler-Nordhein Dynamic Adaptive Memory (FN-MEM) is an analog memory device that stores information by desynchronizing a pair of synchronized dynamical systems [102]. By utilizing a differential architecture, both Program and Erase can be carried out with unipolar
pulses of small magnitudes, thus eliminating the need for energy inefficient charge pumps. We reported adaptive energy expenditure ranging from 5 fJ/bit to 2.5 pJ/bit. The prototype FN-MEM devices occupied substantial area in [102]. However, their area can be expected to be around four times that of a NOR Flash cell while being able to store an effective 10 bits of analog information. The comparison to NOR Flash is valid because, like NOR Flash, all FN-MEM share a global ERASE via FN tunneling. However, instead of global ERASE, a lower power RESET is implemented by the differential architecture. In addition to the two floating gate transistors per cell, each FGT also requires a second transistor to serve as an access control to the control gate. As a source of comparison, in [39], a low power embedded NOR Flash was reported with a program energy of 49 pJ/bit and an erase energy of 9.4 pJ/bit. The areal density of the memory was 1.5 Mb/mm2. For the present feasibility study, we assume a memory density ($\rho_M$) of 10Mb/mm$^2$ (1M devices with 10b memory each) and program/erase energy ($E_{FNDAM}$) of 1 pJ/bit. If the system could be implemented in a 3D NAND architecture, a much higher density (> 1Gb/mm$^{-2}$) could be potentially achieved [88,114].

$$P_{FNDAM} = R \cdot E_{FNDAM}$$  \hspace{1cm} (7.8)

Where $D_R$ is the data rate given by:

$$D_R = B f_{CS}/CR$$  \hspace{1cm} (7.9)

where $B$ is the precision in bits of the ADC/Classifier output.
Figure 7.2: a) Floating gate structure for implementing dynamic memory. b) Equivalent circuit of FN-MEM. c) Unipolar SET and RESET creates an incremental/decremental counter.

7.1.8 Wireless transmission

To compare the performance of storage-than-retrieval system, we analyzed the power consumption for two wireless transmission scenarios. Short range transmission (SR Tx) over 50 m implemented by a sub-1Ghz radio [143] and a long range communication (LR Tx) over LoRa radio [14].

7.1.9 Integrated self-powered sensor and logger

The self-powered sensor-logger-system is an extended concept of the FN-MEM memory 7.3. Instead of modulating the two dynamical systems via SET/RESET pulses from a digital system, it is allowed to be modulated directly by the signal being sensed. In this case, the SET node is directly coupled to the external signal via a suitable transducer and the RESET node is kept at a stable reference voltage.
In [101], we had derived a tractable mathematical model for the data sensed and stored by the sensor-data-logger in response to an arbitrary time-varying input signal \( x_t \). We found that the output of the data-logger \( Y_T \) measured at time-instant \( T \) can be expressed as

\[
Y_T = R(T)A_x(T)
\]  

(7.10)

where \( A_x(T) \) represents the total “action” due to the input signal \( x_t \) accumulated up to the time instant \( T \) and \( R(T) \) is a “forgetting” factor that is independent of the input signal \( x_t \). \( R(T) \) models the data retention capability and arises due to resynchronization of the sensor and reference FN devices, after the sensor device is perturbed by \( x_t \). In the Methods section, we show that the action \( A_x(T) \) can be expressed in terms of device parameters as

\[
A_x(T) = \frac{k_1}{k_2} V_0^2 \exp\left(-\frac{k_2}{V_0}\right) \int_0^T \left[ \left(1 + \frac{C_R x_t}{V_t}\right)^2 \exp\left(\frac{k_2 C_R x_t}{V_t(V_t + C_R x_t)}\right) - 1\right] dt
\]  

(7.11)

and the resynchronization term \( R(T) \) can be expressed as

\[
R(T) = \frac{V_t^2}{V_0^2} \exp\left(\frac{k_2}{V_0} - \frac{k_2}{V_T}\right).
\]  

(7.12)

Here \( V_t \) is given by equation 7.13:

\[
V_t = \frac{k_2}{\log(k_1 t + k_0)} + k_3
\]  

(7.13)

with \( V_0 \) and \( V_T \) representing the device voltage at time-instant \( t = 0 \) and \( t = T \) seconds. The parameter \( C_R \) in equation 7.11 models a capacitive divider that is formed due to the coupling of the input capacitance onto the floating-gate. It was shown in [101] that the
first-order action model given by equation 7.10 accurately tracks a more computationally intensive ordinary differential equation (ODE) based device model. The action $A_x(T)$ is monotonic with respect to energy and hence can be used as a measure of cumulative energy.

7.2 Results

7.2.1 Power consumption

Fig. 7.4 shows the breakdown of power consumed by different modules. Power consumed by the amplifier is the same in all cases because it is determined by the SNR and bandwidth of the input signal. ADC is consumption is highest when the raw signal is being digitized. In the other two scenarios, ADC power consumption is reduced but there is an additional power consumer added in the form of the energy extraction module. Though there is a marginal addition to the power consumption due to this module, its effects downstream are significant due to the reduction in data rate for the subsequent modules. The total power consumed by
# Table 7.1: Parameters

<table>
<thead>
<tr>
<th>Module</th>
<th>Parameter</th>
<th>Value</th>
<th>Unit</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>MicroBattery</td>
<td>$\rho_E$</td>
<td>2</td>
<td>$2 mA \cdot h/cm^2$</td>
<td>[68]</td>
</tr>
<tr>
<td>Amplifier</td>
<td>NEF</td>
<td>3</td>
<td>$\mu V$</td>
<td>[150]</td>
</tr>
<tr>
<td></td>
<td>$v_n$</td>
<td>10</td>
<td>$kHz$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>BW</td>
<td>2</td>
<td>$V$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$V_{DD}$</td>
<td>1.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADC</td>
<td>$E_{ADC}$</td>
<td>32</td>
<td>pJ</td>
<td>[76, 110]</td>
</tr>
<tr>
<td></td>
<td>$B$</td>
<td>16</td>
<td>bits</td>
<td></td>
</tr>
<tr>
<td>Classifier</td>
<td>$m$</td>
<td>4</td>
<td></td>
<td>[5]</td>
</tr>
<tr>
<td></td>
<td>$K$</td>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$E_{MAC}$</td>
<td>1</td>
<td>pJ</td>
<td></td>
</tr>
<tr>
<td>Compressive sensing</td>
<td>$D$</td>
<td>1024</td>
<td></td>
<td>[28]</td>
</tr>
<tr>
<td></td>
<td>$E_{DFF}$</td>
<td>30</td>
<td>fJ</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$CR$</td>
<td>[1,2,4,8]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FN-MEM logger</td>
<td>$E_{FNDAM}$</td>
<td>1</td>
<td>pJ</td>
<td>[102]</td>
</tr>
<tr>
<td></td>
<td>$\rho_M$</td>
<td>10M</td>
<td>bits/mm$^2$</td>
<td></td>
</tr>
<tr>
<td>LR Tx</td>
<td>Modality</td>
<td>LoRa</td>
<td>km</td>
<td>[14]</td>
</tr>
<tr>
<td></td>
<td>Distance</td>
<td>2</td>
<td>mW</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tx Power</td>
<td>25</td>
<td>bits/s</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Data rate</td>
<td>1k</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SR Tx</td>
<td>Modality</td>
<td>Sub-1Ghz</td>
<td>m</td>
<td>[117]</td>
</tr>
<tr>
<td></td>
<td>Distance</td>
<td>75</td>
<td>mW</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tx Power</td>
<td>15</td>
<td>bits/s</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Data rate</td>
<td>80k</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure 7.4: Power breakdown of different modules for the three scenarios.
Figure 7.5: Operational lifetime of the system based on battery constraint and memory constraint. The dashed line indicates an optimum scenario when both are used up simultaneously.

the whole system is given by:

\[ P_T = P_{AMP} + P_{ADC} + P_{Class} + P_{EE} + P_{CS} + P_{FNDAM} \] (7.14)
7.2.2 Operational Lifetime

The operational lifetime of the sensor $T$ depends on whether the battery gets used up or if the memory space fills out, whichever happens earlier 7.5.

$$T = \min \left( \frac{\rho E A_b}{P_T}, \frac{\rho M A_M}{D_R} \right) \quad (7.15)$$

Areas of battery $A_b$ and memory $A_M$ are assumed to be equal (at $1mm^2$ each) and the time durations after each would be used up is shown in Fig. 7.5. As a first order optimization protocol, the battery and the memory can be sized such that they get used up at the same time. This method does not take into account non-linear effects such as decline of battery density with decreasing battery size.

$$A_b = A_T \frac{1}{1 + \frac{\rho_E R}{\rho_M P_T}} \quad (7.16)$$

And the optimal time is given by:

$$\hat{T} = \frac{A_b \rho_E}{P_T} \quad (7.17)$$

7.2.3 Compressive sensing results

7.2.4 FN-MEM vs WirelessTx

For long distance communication, such as the specific case of 2 km considered here, the power dissipation is on the order of mW for a bitrate of less than 1000 bps. In such cases, storage of data on non-volatile memory would be ideal. There is a 7 order of magnitude difference in power consumption between FN-MEM and RF. However, at high data rates
Figure 7.6: Effect of compressive sensing on operational lifetime for the three scenarios
Figure 7.7: Operational lifetime for different Compression Ratios (CR), for FN-MEM and Short Range and Long Range Wireless Transmission conditions.
the 10 Mb FN-MEM quickly fills up. For example, raw data from 1 channel at the rate of 2kSps with 16 bits/sample will fill $1mm^2$ of FN-MEM in 5 minutes. Another thing to note is that the FN-MEM occupies real estate that could have been occupied by a battery. 1 bit of FN-MEM occupies an area of $0.1\mu m^2$. In the same area, a battery with energy capacity of 20 nJ could be implemented. Thus, if the cost to transmit the data was less than 20 nJ, it would be more beneficial to have the battery. However, such limits can only be reached in short range communications.

7.2.5 Self-powered intracellular recording

In Chapter 6 I reported the feasibility of a zero-power timing device that can continuously operate without any external powering and instead be powered directly by the signal being sensed. These devices are thermodynamically and quantum-mechanically activated and can operate at power levels below $10^{-18} - 10^{-16}W$, less than the energy that can be scavenged from a single neuron action potential. The underlying hypothesis for this research will be that neural action-potentials can modulate the responses of zero-power timing devices. Thus, the state of the timing device at any point in time will capture the history of the ensemble activity. The state can be wirelessly interrogated (without any real-time telemetry requirements) using a retrieval modality (radiofrequency or ultrasound that can achieve the smallest sensor form factor.)
Figure 7.8: a) Operational lifetime for different recovered information rates for different Compression Ratios (CR), for both FN-MEM and WirelessTx conditions. b) Optimal battery and memory sizing for different scenarios to maximize operational lifetime given the area constraints.
7.3 Discussions and Conclusions

Comparisons of different modalities are summarized in Table 7.2. Data logging significantly increases the operational lifetime if remote sensing is desired. Processing and compressing the signal would increase the lifetime of the wireless system as well, but it will still be lower than an equivalent data logger. Energy extraction, classification and compressive sensing each decrease the number of bits that need to be logged and thus increase the operational lifetime. The net effect is cumulative. The amount of compression depends on the sparsity and will need to optimized by a domain expert. Analog-sensor-logger (Type IV system) provides an estimate of the cumulative action of the signal. Integrating multiple sensor-loggers increases the sensitivity of the system. Having multiple differentiated sensor-loggers does lead to more information, but recovering the original signal is an ill-posed problem and needs to be researched further.

Table 7.2: Comparison

<table>
<thead>
<tr>
<th>Paradigm</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power consumption</td>
<td>LR 250mW</td>
<td>SR 15mW</td>
<td>150nW</td>
<td>155nW</td>
</tr>
<tr>
<td>Operational lifetime</td>
<td>&lt; 1 min</td>
<td>&lt; 40 mins</td>
<td>&lt; 1 day</td>
<td>&lt; 2 months</td>
</tr>
<tr>
<td>Range</td>
<td>2 km</td>
<td>100 m</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Information throughput</td>
<td>100 kbps</td>
<td>100 kbps</td>
<td>600 bps</td>
<td>0.1 bps</td>
</tr>
</tbody>
</table>
Chapter 8

Conclusion

8.1 Concluding remarks

In this dissertation, I have explored algorithmic and device based approaches for overcoming the limitation of current neural activity recording-technologies in terms of energy efficiency and longevity. I demonstrated that measuring the neural energy of projection neurons in a locust antennal lobe is sufficient to discriminate between different stimuli that the locust is exposed to. Signal energy can be estimated with little less power consumption and there are significant savings in downstream processing. To eliminate power loss in long distance communication, I investigated novel data-logging techniques. This involved implementing compressive sensing technique which is a very active area of research in the signal processing community. The system I implemented is generic enough for compressively logging any type of data. One of my key scientific contributions is the investigation of a new approach to storing information in synchronized dynamical systems by desynchronizing them. With this approach, one can reach energy efficiencies unmatched by any current system. I demonstrated the feasibility of combining these tools and being able to record classified neural data for over a month in a system small enough to be implantable within the locust.
In further exploration of the memory device, I hypothesized that the energy requirement is so small that it can potentially be powered by the signal being sensed directly. I experimentally demonstrated this concept by recording tiny vibrations from a piezo material while being completely disconnected from any power source. Though modeling studies suggest that the system can log intracellular neural data, more research needs to be done before it can be used for practical neural recording. I also showed that the dynamic nature of the memory cell can be utilized to create artificial synapses that can naturally adapt to differences in memory requirements during the training and inference phases of machine learning.

8.2 Future directions

Based on the concepts described in this dissertation, here are some research question that I think are worth exploring:

- The true power of compressive sensing lies in exploiting the sparsity of signal, be it on any domain. But knowledge of that domain is crucial for accurate reconstruction. In what domain is the neural signal sparsest?

- How can the sensitivity of the self-powered sensor logger be increased so that it can log extracellular neural signals?

- Storing information in differentiated dynamical systems may allow for reconstruction of the input signal. But it is an ill-posed problem. How can one go about regularizing the problem so that some meaningful information about the input signal can be obtained?

- Can the self-powered dynamical systems serve as building blocks to create more exotic self-powered circuits that can do useful computations?
For what real-world problems is knowing the cumulative signal history good enough?

For such problems, the self-powered approach is readily implementable.
Appendix A

Supplementary Information

A.1 Odor Panel

<table>
<thead>
<tr>
<th>Chemical</th>
<th>CAS Number</th>
<th>Delivery Method</th>
<th>Concentration</th>
</tr>
</thead>
<tbody>
<tr>
<td>TNT</td>
<td>118-96-7</td>
<td>Solid, heated to 50°C</td>
<td>30.6 ppb</td>
</tr>
<tr>
<td>DNT</td>
<td>121-14-2</td>
<td>Solid, heated to 50°C</td>
<td>779 ppb</td>
</tr>
<tr>
<td>Hexanol</td>
<td>111-27-3</td>
<td>1% v/v dissolved in mineral oil</td>
<td>4.44 ppm</td>
</tr>
<tr>
<td>Benzaldehyde</td>
<td>100-52-7</td>
<td>1% v/v dissolved in mineral oil</td>
<td>7.46 ppm</td>
</tr>
<tr>
<td>Air</td>
<td></td>
<td>Heated to 50°C</td>
<td>1</td>
</tr>
</tbody>
</table>

*Table A.1: Odor Panel*
A.2 PID data

**Figure A.1:** Response of photoionization detector showing the temporally precise delivery and removal of the olfactory stimuli. X-axis is time in seconds, and Y-axis indicates the PID signal in volts. The color bars indicate the time when the stimulus was delivered. Solid traces show the mean PID signal and individual traces (ten trials) are shown in gray to illustrate reproducibility.
A.3 Device parameters and drift correction

Device parameters from equation 1 (Main text) can be experimentally obtained by allowing the floating gate to discharge via Fowler-Nordheim tunneling and fitting the model on observed data.

We obtained the following parameters:

<table>
<thead>
<tr>
<th>Device No.</th>
<th>Node</th>
<th>log(k_1)</th>
<th>(k_2)</th>
<th>(k_3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Sensor</td>
<td>38.59</td>
<td>347.20</td>
<td>-4.24</td>
<td></td>
</tr>
<tr>
<td>Ref</td>
<td>39.47</td>
<td>359.04</td>
<td>-4.43</td>
<td></td>
</tr>
<tr>
<td>2 Sensor</td>
<td>44.53</td>
<td>425.01</td>
<td>-4.86</td>
<td></td>
</tr>
<tr>
<td>Ref</td>
<td>42.06</td>
<td>389.18</td>
<td>-4.57</td>
<td></td>
</tr>
<tr>
<td>3 Sensor</td>
<td>42.14</td>
<td>381.26</td>
<td>-4.35</td>
<td></td>
</tr>
<tr>
<td>Ref</td>
<td>41.16</td>
<td>370.20</td>
<td>-4.30</td>
<td></td>
</tr>
</tbody>
</table>

**Table A.2:** Device Parameters

\(k_0\) depends on the initial conditions. The starting voltages for each node can be chosen such that the sensor and reference have the same rates and are thus synchronized. However, the mismatch in other parameters causes the two nodes to drift. The drift is predictable and can be corrected as shown in Fig. A.2. \(k_3\) was subsumed into \(V_t\) by setting \(V_t \rightarrow V_t - k_3\). The modified \(V_t\) is used for derivation of the explicit model in equation 4.
Figure A.2: Synchronization after correcting for drift: a) Experimentally measured values of sensor and reference output voltages. b) Change in sensor and reference values from the baseline $\Delta V_t = V_t - V_0$. c) Desynchronization between the sensing and reference nodes. A consistent drift is observed across trials which can be compensated. d) Desynchronization compensated for drift, which is the final recorder response.
A.4 Temperature compensation

To ensure accurate temperature compensation, it is important to accurately initialize the reference and the sensor devices. To illustrate this we show raw measured data from our experiments (Fig.2) for the sensor and the reference devices. The dataset shows an outlier due to incorrect initialization of the reference device (as highlighted in Fig. A.3, where the reference node was incorrectly initialized lower than the target 50 mV difference from the sensor node). This difference results in improper compensation of the temperature variations. To ensure consistency, we have removed the outlier from Fig.2e.

Figure A.3: a-b) Raw sensor and reference data for temperature compensation experiments highlighting the outlier which results in incorrect temperature compensation, as shown in c)
A.5 FN Memory Read Energy

The readout power is dependent on the readout accuracy required and the speed at which it operates. For a PMOS in a source follower configuration, the readout noise is given by:

\[
v_{n,\text{rms}}^2 = \frac{4kT}{g_m} \Delta f = \frac{4U_T q \Delta f}{g_m}
\]  

(A.1)

For subthreshold operation

\[
g_m = \frac{\kappa I_d}{U_T}
\]  

(A.2)

Therefore

\[
v_{n,\text{rms}}^2 = \frac{4U_T^2 q \Delta f}{\kappa I_d} = \frac{4U_T^2 q V_{DD} \Delta f}{\kappa P_{\text{read}}}
\]  

(A.3)

Above equation is plotted in Fig A.4 for different noise floors and readout frequency for \(V_{DD}=5\text{V}\), \(U_T=26\text{mV}\) and \(\kappa=0.7\).

\[137\]
Appendix B

Backpacks for insects

This chapter is based on work submitted for conference proceedings [117]: Pochettino, Owen, Darshit Mehta, Deb Saha, Barani Raman, Kenji Aono, and Shantanu Chakrabarty. "A Backpack Recording Platform for Neural Measurements in Ambulatory Insects." (In Review)

Author contributions: SC conceived the study and designed the experiments/analyses. OP designed and fabricated the board. OP and DS collected data from locusts. OP, DM and KA analyzed the data. All authors participated in the writing of the manuscript, with OP being the primary author.

B.1 Introduction

While cell-based and chemical olfactory sensors promise sensitivity and selectivity in the lab, performance of these sensors [87, 125, 138] is susceptible to changes in environmental conditions, a common occurrence in the field, while also suffering from limited operational lifetime [64, 78]. Insects, such as the locust or moth, provide a low-cost biosensor that responds to odors with predictable behavioral changes, especially if they have undergone
Pavlovian training [100]. In literature, several hybrid approaches are reported that tap into an insects’ ability to sense and behaviorally respond to specific environmental conditions [7, 37, 65, 100, 120, 129]. Unfortunately, extracting relevant information from such an organism requires a means to observe this behavior without constraints on movement, which can be difficult or even impossible in large-scale field deployments.

In this work, instead of observing the behavioral response of an insect in response to an extrinsic stimuli, we aim to monitor the neural response of the insect remotely. The feasibility of measuring specific groups of neurons inside of a locust brain and correlating that response to an odor stimuli has been reported by [127, 128] using recordings from immobile preparations and mobile, robotic settings [129]. Since our objective is to enable odor monitoring in the field, we have designed a neural recording backpack for a locust, that does not significantly inhibit its ability to remain ambulatory. Wireless neural recording devices have been reported in literature [27, 81, 109], but few of those [46, 130, 144] are compact enough for an ambulatory insect. We believe that the proposed neural recording system herein has the smallest form factor reported to date, whilst maintaining the ability to achieve wireless transmission distances up to 75 m.

B.2 System Architecture

For this manuscript, we focus on two aspects of the cyborg insect platform: 1) amplification and measurement of locusts’ neural activity, and 2) the wireless transmission of said activity to a remote user. As illustrated in Figure B.1, a locust handles the sensing of odors in the biological domain and our silicon system captures the evoked neural responses and transmits corresponding amplified, filtered signal.
Figure B.1: An example of a locust outfitted with a light enough neural recording backpack that it can move about freely. Odor puffs can be applied to the antenna of the locust, which will cause a neural response to propagate in the antennal lobe, which can be captured using a silicon recording platform.
B.2.1 Biological Frontend and Design Requirements

The antenna of the locust (Schistocerca americana) comprises of thousands of olfactory receptor neurons (ORNs) which act as a biological transducer converting the chemical signatures of the odor into spikes in the electrical domain. The antenna lobe (with around 800 projection neurons and 300 local neurons) is the primary olfactory processing center in insects which integrates and compresses the high-dimensional sensory signals for downstream centers like the mushroom body and lateral horn [127–129]. To leverage the locusts’ inherent chemical-electrical transduction properties, we recorded extracellular neural signals from the antennal lobe that were evoked by odorants. To physically interface with the neural tissue, we used twisted tetrode wire for recording as they have high flexibility and can readily conform to desired shapes without exerting stress when affixed to a recording site. From in-lab measurements using standard benchtop equipment with fixed gain of 192 [72, 119], the action potentials associated with odors of interest were measured as $\approx 100 \mu V$ with spike frequencies near 1 kHz.

Our design requirements were primarily constrained by the difficulty of obtaining microvolt signals from a mobile system and the limitations arising from the physical capabilities of our organism of consideration. A typical locust body measures $\approx 0.5$ cm by 2.5 cm, with the capacity to carry up to 2 g while retaining the ability to walk. We targeted our system weight to be under 1 g (including battery) and wanted to ensure that its physical footprint was in accord with the locust body shape so as not to interfere with its locomotion. Due to the inherent difficulty of precisely placing a single electrode, a multi channel acquisition system would that could switch between electrodes during runtime would allow the selection of the electrode channel with highest SNR. To enable studies of insects in ambulatory settings, we
designated that the wireless telemetry link must support distances in excess of 50 m and have an operational lifespan greater than \( \approx 1 \) hour.

B.2.2 Hardware

To satisfy the signal conditioning needs of neural recordings, we use the Intan RHD2132, which is a commercial-off-the-shelf System-on-Chip (SoC) that is used extensively by neuroscientists [47, 56, 133]. This SoC can handle the large dc offsets between the reference electrode and the recording electrode while also supporting the filtering, amplifying, and digitizing of up to 32 electrode channels. Each channel of amplification consumes \( 7.6 \mu A/\text{kHz} \) (we use a lowpass filter with 2.5 kHz cutoff, so each amplification channel uses 19 \( \mu A \)) and the ADC used for digitizing has a quiescent current of 510 \( \mu A \) [72] with active consumption of 2.14 \( \mu A/\text{kHz} \), or 10.7 \( \mu A \) at a 5 kHz sampling rate. Individual channels have power-gates, thereby allowing the SoC power consumption to scale with the number of recording channels being used. The digitized signals can be offloaded using a standard SPI interface, which also allows users to send configuration settings to the SoC. Additional fixed supply draws from the RHD2132 with a 250 Hz highpass filter are on the order of 240 \( \mu A \), giving an estimated average supply current of 780 \( \mu A \) for the signal conditioning stage.

For enabling long range wireless telemetry, a Texas Instruments (TI) CC130 Wireless Microcontroller (MCU) was chosen due to the ease of integration with the rest of the system in addition to its operation in the 868 to 920 MHz ISM band [143]. The low power sub-GHz telemetry link can allow communication distances beyond 1 km when paired with efficient antenna configurations and ample power. The CC1310 also contains a secondary 16-bit ARM core independent from the main controller, which allows the MCU to stay in low power
modes while the secondary core asynchronously collects data from the RHD2132. The secondary core consumes 600 µA when fully utilized, and the main core idles at around 570 µA with standby currents down to 700 nA. Additional details on the use of the CC1310 will be presented in the Firmware subsection.

The RHD2132 frontend and CC1310 MCU were mounted on a flexible PCB along with the required supporting components (e.g. 915 MHz ceramic antenna, JTAG programming interface, crystal oscillators, bypass capacitors, etc.) to create a compact backpack system. The integrated system is shown in Figure B.2a and a block diagram illustrating the signal pathway is seen in Figure B.2b. The use of a flexible polyimide substrate allows the PCB weight to be reduced to 258 mg as compared to a traditional FR4 board which would have weighed 1.265 g. Excluding the JTAG programming header, which is removed prior to deployment, the PCB area is 2.897 cm². Not shown in the figure is a 3 V coin cell battery with a rated capacity of 11 mAh, diameter of 9.5 mm, and weight of 504 mg.

**B.2.3 Firmware**

Typical neural recording experiments conducted in-lab, using locusts, can sample as high as 15 kHz, but for many cases of odor detection, the signal of interest lies within \( \approx 2 \text{kHz} \) [119, 128]. Therefore, a sampling rate of 5 kHz was targeted to satisfy the Nyquist sampling rate for these signals of interest. However, the actual sampling was slightly higher due to the closest divided rate of the 16 bit low power secondary core being \( 2^{16}/13 = 5.041 \text{kHz} \).

When the secondary core’s local memory is filled with 250 samples from the RHD2132, the main core of the CPU wakes up to transfer those samples into the primary RAM, this occurs every 50 ms. On the eighth transfer (i.e. when 2,000 samples are available), the MCU
assembles the samples into a radio frequency (RF) packet and uses its integrated radio to perform a wireless transmission (i.e. every 400 ms). The main core then reenters a low power mode. All the while, the secondary core will continue to collect data.

A companion CC1310 was programmed to serve as a base-station for receiving all transmitted readings from any deployed backpacks and relaying them to a standard computer over a wired interface. This companion CC1310 was also used to send commands from the computer to the backpack, wirelessly, that enable configuration changes to the RHD2132, such as: which electrode to sample from, the prefilter parameters, and gain of the amplifiers. In low power mode, the CC1310 firmware will only sample from one channel at a time, keeping the remaining channels of the RHD2132 power-gated off.

B.3 Validation

B.3.1 Laboratory Testing

Several stages of validation were performed to examine the backpack’s performance as a wireless neural recorder and to qualitatively determine the affect that it had on the locusts mobility. Preliminary testing was done in a laboratory setting, where the device was validated against a previously recorded locust response using the set up as described within [127, 128]. The performance of the backpack in this setting was deemed acceptable as every off-the-shelf competent was performing per their respective manufacturer specification [72, 143]. However, there were some periodic in-band noise spikes that were introduced by our system that upon closer examination, were found to be directly related to the MCU operation design. As described in Section B.2.3, and can be observed from the supply current measurements shown
in Figure B.3(a), the main CPU on the MCU wakes up after every 250 samples to transfer data from the peripheral RAM into the CPU’s RAM and after 2,000 samples the MCU wirelessly transmits the buffered data. These periods were synchronous with the periodic noise spike that appeared in the recordings, with spikes of larger magnitude occurring during wireless transmission wakeups and the smaller, sometimes nonexistent, spikes occurring each time the main core woke up to transfer data from the secondary core. The additional supply current draw of the main core during data transfers would cause supply voltage droops, and part of the RF energy from the wireless transmission would also couple onto the sensitive analog frontend.

To reduce this noise, a two pronged solution was implemented. Firstly, we increased the size of the system ground plane by increasing the amount of metal in the PCB ground fill. Although this increased the system weight, it did help in providing stability to the supply rails as well as giving a path for energy to flow without coupling onto the analog signals. To highlight the impact of the periodic noise, the input channel was shorted to the reference signal and data collected from the wireless backpack, as shown in Figure. B.3(b). The noise RMS (Root Mean Square) was calculated as $771\mu V$ after the dc offset of the amplification stage was removed and only the ac signal or quantization noise was considered. Since the periodicity of the spikes are correlated to known events (main cpu wakeup/transmission), we could mark those particular events for correction in post processing without difficulty.

### B.3.2 Locust Neural Testing

In order to conduct *in-vivo* testing on the live locust (*Schistocerca Americana*), a surgical procedure was performed, following the methods in [119, 128, 129], to expose the antennal
lobe. Once the antennal lobe had been exposed, an electrode was placed into the antennal lobe; additionally, a Ag/AgCl electrode was submerged in a reference saline solution [127]. Once prepped, the locust and the wireless recording backpack were placed in an experimental setup described in [127, 128], where the locust would be repeatedly exposed to two target odors it might commonly encounter in its operating environment, hexanol and benzaldehyde.

Ten trials were conducted for each of these odors. Figure B.4(a) shows the raw signal for a single trial of hexanol exposure. The period where an odor is being introduced to the environment is indicated by colored shading. Figure B.4(b) shows the corresponding signal after filtering out the dc and main CPU artifacts. Upon close examination of Figure B.4(a) and (b), there is a dense collection of spiking activity at ≈10.5 seconds, right after the odor introduction. The significance of this activity is more clearly visible when examining the change in RMS which is calculated on a moving window of 100 samples, as shown in Figure B.4(c).

Figure B.5 shows the average filtered RMS for 10 trials of two odors (grey shaded area indicates standard error of mean across trials). As in prior work, we are able to capture distinct responses of odor evoked responses, which can include increased neural activity at the onset of odor exposure, or even after the removal of the odor.

B.3.3 Mobility Testing

With the prototype validated as a neural recorder, its feasibility as a backpack can be demonstrated. The prototype recording platform (backpack) was affixed to the back of adult locusts using wax as an adhesive, see Figure B.1 for an example, sans electrode wiring between the backpack and antennal lobe. Preliminary, qualitative experiments were conducted in
order to assess locusts’ mobility. The locusts' ability to walk was first examined, and the insect was able to hop around with no visible impairment despite the added weight of the backpack. The penultimate demonstration would be to have the locust fly around while wearing the system, so further experimentation was conducted with locusts outfitted with the backpack — these subjects were lifted up by a person and lightly dropped from a height. Although the locust performed a controlled gliding descent, it did not appear to fly. The same locust was tested without a backpack in a similar lift and drop test and its ability to fly was validated. It is hypothesized that the form factor of the backpack, which was limited to a single-sided board with hand-solderable parts was a large enough presence to prevent regular flight; additional miniaturization will need to take place for supporting in-flight recordings.

B.4 Performance

The full backpack recording platform was able to perform its duties of recording and transmitting neural data without preventing basic ambulatory observations in locust, and the measured specifications are presented here. The first of which was the weight; the fully populated PCB weighed approximately 703 mg, placing the full system weight, including the battery and electrode preparation, at 1.198 g, which is well under the provided target specification of 2 g. The second major specification was battery life and with a measured average supply current from the 3 V battery around 4.235 mA, the selected coin cell battery is able to provide operational power for 2.57 hours — more than double the time required for useful deployments. Finally, the transmission distance of the device was verified at a
transmission power of $-10\,\text{dBm}$ and the device was able to successfully connect over a distance of approximately 75 m for line-of-sight transmission. A full table of the prototypes specifications can be see below in Table B.1.

Table B.1: Design Specifications

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Target</th>
<th>Measured</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCB Area</td>
<td>1.25 cm$^2$</td>
<td>2.897 cm$^2$</td>
</tr>
<tr>
<td>Radio Band</td>
<td>N/A</td>
<td>915 MHz ISM</td>
</tr>
<tr>
<td>Battery Capacity</td>
<td>N/A</td>
<td>11 mAh</td>
</tr>
<tr>
<td>Avg. Current Consumption</td>
<td>5.5 mA</td>
<td>4.235 mA</td>
</tr>
<tr>
<td>Lifetime</td>
<td>&gt;1 hours</td>
<td>2.57 hours</td>
</tr>
<tr>
<td>Weight</td>
<td>&lt; 2 g</td>
<td>1.198 g</td>
</tr>
<tr>
<td>TX Distance</td>
<td>50 m</td>
<td>≈75 m</td>
</tr>
<tr>
<td>TX Power</td>
<td>N/A</td>
<td>$-10,\text{dBm}$</td>
</tr>
</tbody>
</table>

B.5 Conclusion

In this manuscript, we have explored the design and validation of a multiplexed, single-channel wireless insect neural recording device that allows tapping into a biological sensing and processing frontend to create a machine olfaction system. This initial prototype has been tested in both laboratory and simulated deployment conditions, where the system’s performance as a neural recorder was evaluated in conjunction with the its feasibility as a portable backpack for locust.
Acknowledgement

This material is based upon work supported by the Office of Naval Research, Grant No. N00141612426.
Figure B.2: (a) Block diagram of the proposed prototype showing the key components. Note that details of the Intan RHD2132 are omitted, further details can be found in [72]. (b) Implemented prototype on flexible PCB comprising: (i) Electrode Interface, (ii) Intan RHD2132, (iii) Power Connection & Crystal, (iv) TI CC1310, (v) Removable JTAG Header, (vi) RF Matching & Antenna.
Figure B.3: (a) Trace of the power consumption for the locust backpack over time, each sample taken with NPLC 0.2 integration on 60 Hz power (approximately every 3.3 ms). (b) A plot of the locust backpack signal recording with shorted inputs.
Figure B.4: Plots showing the (a) raw neural recording from the backpack, (b) time-domain filtered to remove the dc offset and coupled transmission noise, and (c) the processed RMS response using methods as described in [128]. The highlighted region indicates an active odor puff.
Figure B.5: Filtered RMS responses of two experiments of ten trials each using the backpack, where odors (a) hexanol and (b) benzaldehyde were used as stimuli. Shaded color regions indicate time periods when odors were released.
Bibliography


159


[80] Mehdi Kiani, Ki Yong Kwon, Fei Zhang, Karim Oweiss, and Maysam Ghovanloo. Evaluation of a closed loop inductive power transmission system on an awake behaving


