Multi-Domain SystemC Model of 128-Channel Time-Multiplexed Neural Interface Front-End

Kiki Wirianto[1], Amir Zjajo[1], Carlo Galuzzi[2], Rene van Leuken[1]
[1] Circuits and Systems Group, Delft University of Technology, Delft, The Netherlands
[2]BMI Research Group, Maastricht University, Maastricht, The Netherlands
Email: kikiwirianto@student.tudelft.nl, a.zjajo@tudelft.nl, c.galuzzi@maastrichtuniversity.nl, t.g.r.m.vanleuken@tudelft.nl

***Abstract*—A SystemC Multi-Domain Virtual Prototype (MDVP) framework provides multi-domain system integration of multi-electrode neural interface and a fast validation time, offering more efficient design process. In this paper, we propose efficient neuron cell, neuron tissue, and multi-channel neural interface front-end models implemented in a SystemC MDVP framework, which supports chemical, biological, and electrical domain modeling and simulation. The proposed model offers advantages for designers not only in reducing the design time, but also regarding the system integration. We make use of 8×16-channel front-end recording circuits to implement a 128-channel time-multiplexed neural interface front-end model. These models are verified with realistic simulation settings and appropriate input signals.**

Keywords—SystemC MDVP, modeling, multi-channel, front-end neural interface

I. INTRODUCTION

Neural networks have been investigated by researchers for several decades. The experiments performed by Hodgkin and Huxley in 1952 gave way to a clearer understanding and precise modeling of the action potentials generation by neurons. Since then, this field has developed rapidly and the interest began to grow from the observation of single intracellular neuron to the simultaneous recording of several individual neurons activity in the cortex [1]. For the past few decades, microelectrodes have been used, together with the neural interface, to examine the neuronal networks activity, which takes form in the Local Field Potential (LFP) and the Action Potential (AP). Each of these two signals has different nature compared to the other in terms of signal bandwidth and amplitude, so versatility is an important attribute that a neural interface system must have. High temporal and spatial recording resolutions are needed to monitor the activity of a large number of neurons simultaneously, leading to multi-channel implementations [1], [2].

For implantable neural interface, area and power are the two main constraints. A minimum measurement resolution of the front-end electronics (including signal-to-noise ratio, effective number of bits, and dynamic range) should be maintained while reducing the area and power to the minimum. This, in addition to the demand for versatility and multi-channel implementation, requires the designers to possess a clear insight about the trade-off between design parameters, such as area, power consumption, and number of channels. A powerful model which could provide critical information about design constraints trade-off can help the system designers in selecting the most effective design considerations based on a certain metric.

Several domains are present in the modeling and simulation of neural networks. Neuron model possesses chemical properties, such as sodium and potassium activities in the Hodgkin-Huxley (HH) model [3]. Biological characteristics exist in the interface between the electrodes and the tissue which affect the noise generated in the neuron signals reading. These signals are then processed in analog domain by the front-end interface before being converted into digital signals and being processed further by the digital back-end spike processing part. Simulating multi-domain blocks proves to be an obstacle in the development of this system. The design and verification of these multi-domain systems is an iterative process, where parts for each physical domain are developed independently and combined in the last stage to realize the final product. With this approach, design errors are often recognized too late in the design cycle, which leads to additional design spins and delayed schedules. Prior research typically model each domain separately, usually written in MATLAB, Verilog-A/AMS, Spice, Simulink, or Simscape, such as electrochemical impedance spectroscopy (EIS) used to characterize electrode-electrolyte interface [4], neural recording noise prediction model created in electrical domain [5], or a SAR ADC noise modeling and analysis [6]. However, since the modeling languages used did not support different time constants in the simulation, multi domains have not been integrated into one simulation environment.

In this paper we introduce key innovations to implement a unified design environment for virtual prototyping by means of:

- A methodology for system design, architecture exploration, and verification of multi-domain systems,
- Very fast system simulation framework for multi-domain systems,
- A ‘correct by construction’ approach for the integration of multi-domain systems.

This research was supported in part by CATRENE project CA 701, Heterogenous INCEPTION (H-INCEPTION).

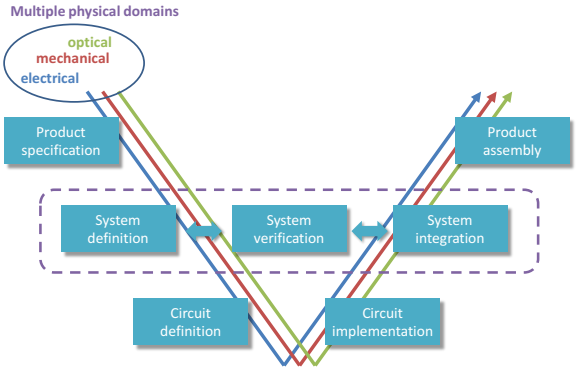

Fig. 1. Scope of H-INCEPTION: system and architecture design and verification phases for multi-domain microelectronics assisted systems.

Fig. 1 depicts the classical V-model describing a typical design cycle, from product definition to product assembly. This approach has several drawbacks: no common entry point for multi-domain design capture, the various tools used are poorly connected, and the whole system simulation lacks a stimuli to represent the parameters associated with the external world/environment in a proper manner.

We focus on the high level modeling of multi-domain electronics-assisted systems to address system definition, architecture exploration, and system-level verification across the relevant physical domains [7-9].

The main contributions of this paper are:

- A full chain of efficient neural interface behavioral models with configurable parameters,
- Models implemented in a SystemC MDVP framework [10-12] which supports multi-domain (chemical, biological, electrical) systems integration and simulation, and
- The neural interface that has a high modularity and facilitates design space exploration and performance optimization.

The remainder of this paper is organized as follows. Section II describes the system architecture and the implemented neural interface front-end models. Section III provides an overview of SystemC modeling. Simulation results are given in Section IV and conclusions are drawn in Section V.

II. System Architecture and Implemented Front-End Models

A. Architecture Overview

The block diagram of the proposed multi-channel neural interface front-end model architecture is illustrated in Fig 2. The system consists of 128-channel (8×16 arrangement) front-end recording circuits (Fig. 2a). Each channel consists of an electrode, a low noise pre-amplifier, a band-pass filter, and a programmable gain post-amplifier (Fig. 2b). An ADC is shared by 16 post-amplifiers through time-multiplexing. With a sampling rate of 32 kS/s for each channel and 8-bit ADC resolution, the raw bit rate is 32.768 Mb/s. The current development of neural transmitter can achieve up to 50 Mb/s raw data rate transmission while consuming 5.6 mW [13]. With front-end circuits and signal processing parts typically consuming up to 20 μW/channel [14], the proposed model is feasible within the power consumption limit of 40 nW/cm^2 to ensure the tissue temperature does not increase more than 2 ^{o}C.

The front-end circuits precondition the neural signals read by the electrodes. In this way, they are ready to be processed by the back-end spike processing part of the neural interface system [15]. Multi-electrode arrays currently available consist of 10s to 100s electrodes and these numbers are projected to double every seven years [16]. The electrode is characterized by its charge density and impedance characteristics [17]. The impedance characteristics play a vital role in neural activity recording since they determine the amount of noise added to the signal. The value for the electrode impedance is between 10-100 kΩ [18]. This impedance determines the amount of thermal noise of the electrode and together with other noise sources, they generate a typical amount of total electrode noise between 5-10 μV_{rms} for a 10 kHz recording bandwidth [19]. Two amplifiers in the neural interface front-end offer high gain without degrading the signal linearity. The first stage is a Low Noise Amplifier (LNA). DC electrode offset is in the order of up to 100s of millivolts and appears at the electrode-tissue interface. As a result, LNAs are usually designed to be AC-coupled to remove this offset. The specifications for this amplifier are: gains of 50-200×, bandwidths of 3-10 kHz, and Input-Referred Noise (IRN) of 2-10 μV_{rms} [20-21]. The gain rolls off at high frequencies due to the device parasitic capacitances, thus a low pass response can be observed. The second stage is a programmable gain amplifier (PGA) with a gain in the range of 10-20× [19]. The band-pass filter accomplishes two tasks: to reject out-of-bands LFPs with the high pass response and to utilize the low pass response to

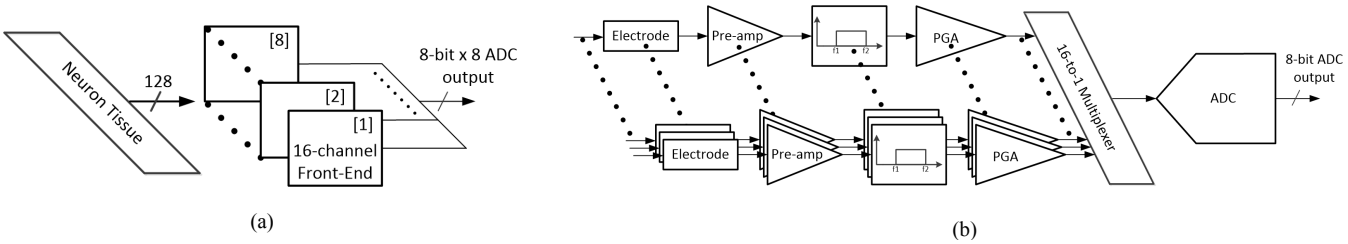

Fig. 2. Block diagram of the implemented 128-channel neural interface front-end model. (a) 8×16-channel front-end recording circuits are used to process 128 neural signals. (b) 16 front-end recording channels share an ADC through time-multiplexing.

prevent aliasing. The cut-off frequency is 100-300 Hz for high pass and 3-10 kHz for low pass [15]. The effect of phase distortion must be minimized since it has a significant impact on the subsequent signal processing. The typical specifications for the ADCs are 8-12 bit resolution and 16-32 kS/s sampling rate and these numbers set the numerical accuracy of the subsequent spike processing part [15]. The ADC is limited by the Signal-to-Noise Ratio (SNR), dynamic range, and the bandwidth of the signal. The non-idealities normally found in ADC implementations are offset and gain error, integral and differential non-linearities, aliasing and quantization effects.

B. Neuron Cell Model

The neuron cell is modelled as a Hodgkin-Huxley (HH) model, which is a mathematical model derived by Hodgkin and Huxley in 1952 to describe the generation of the neuron action potential [3]. Current can be carried through a membrane either by charging the membrane capacity or by moving the ions through the resistance parallel to the capacity,

$$I = C_M \frac{dV}{dt} + I_i \quad (1)$$

where I is the total membrane current density, C_M is the membrane capacity per unit area, V is the displacement of the membrane potential from its resting value, t is time, and I_i is the ionic current density. The ionic current is divided into components carried by sodium and potassium ions (I_{Na} and I_K) and leakage current (I_l):

$$I_i = I_{Na} + I_K + I_l \quad (2)$$

I_{Na} is equal to the sodium conductance (g_{Na}) multiplied by the difference between the membrane potential (V_M) and the equilibrium potential for the sodium ion (E_{Na}). The same equation applies for I_K and I_l:

$$I_{Na} = g_{Na}(V_M - E_{Na}) \quad (3)$$

$$I_K = g_K(V_M - E_K) \quad (4)$$

$$I_l = g_l(V_M - E_l) \quad (5)$$

where g_K is the potassium conductance, E_K is the equilibrium potential for the potassium ion, g_l is the leakage conductance, and E_l is the equilibrium potential for the leakage current. g_{Na} and g_K are functions of time, where E_{Na}, E_K, E_l, C_M, and g_l may be considered as constants.

C. Neuron Tissue Model

The proposed neuron tissue model is illustrated in Fig. 3. The × sign shows the location of the electrode tip and the ○ signs indicate the locations of the neuron cell models. Each electrode reads the signals from 16 neuron cells. As this model represents the extracellular recordings of the neurons, we perform some adjustments to the signals produced by the neuron cell models. Several differences exist between the recording of intracellular and extracellular signals [22-23]. One notable difference is the signal's amplitude, where the amplitude of the measured extracellular signals could be two or three orders of magnitude smaller than their intracellular counterparts. Before the extracellular potential can be determined, first the transmembrane current must be calculated [22]. The electrical potential in the extracellular space is governed by the Laplace's equation and at the boundaries

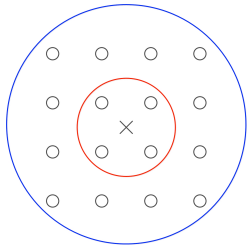

Fig. 3. Neuron tissue model illustration. × indicates the electrode tip location and ○ indicates the neuron cell model locations.

$$(1/\rho) = J_m \quad (6)$$

applies, where J_m is the transmembrane current density and ρ is the extracellular resistivity. Multiple current sources are combined linearly by the superposition principle. The membrane current in real neurons is distributed over elongated cylindrical processes. By using the line source approximation, we simplify the locating of the transmembrane net current for each neuron cell. The equation presenting the extracellular potential is

$$V_{extracellular}(r,h) = (\rho/4\pi)\int_{-\Delta s}^{0} lds/\Delta s\sqrt{r^2 + (h-s)^2} \quad (7)$$

where $V_{extracellular}$ is the extracellular potential, r is the distance from the source to the measurement, h is the longitudinal distance from the end of the source line, and $l = \Delta s + h$ is the distance from the start of the source line. This biological property of the extracellular potential measurement is included in the proposed SystemC MDVP model, where the signal amplitude of 4 closest neurons to the electrode tip is higher than the signal amplitude of the other 12 neurons. Each neuron fires a spike once in 100 millisecond with different random delay, which represents the different activity of neurons in a neural network with 10 Hz firing rate. When the number of channels (electrodes) in the interfacing system increases, the number of neurons that is read also increases. The relation between these two numbers is

$$n_{neuron} = \begin{cases} \left(\sqrt{n_{channel}} + 3\right)^2, & n_{channel} = 1, 4, 16, 64, \ldots \\ \left(\sqrt{n_{channel}/2} + 3\right) \times \left(\sqrt{n_{channel} \times 2} + 3\right), & n_{channel} = 2, 8, 32, \ldots \end{cases} \quad (8)$$

where n_{neuron} is the number of neurons and $n_{channel}$ is the number of channels. Fig. 4 shows the neuron tissue model with $n_{channel}$ = 4, which leads to n_{neuron} = 25. The 4 colored circles indicate the 4 nearest neurons for each electrode. The circles for 12 further neuron are not displayed for the ease of viewing.

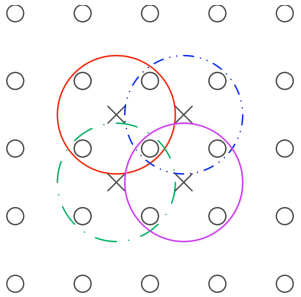

Fig. 4. Neuron tissue model illustration with 4 channels and 25 neurons. The 4 colored circles indicate the 4 nearest neurons for each electrode.

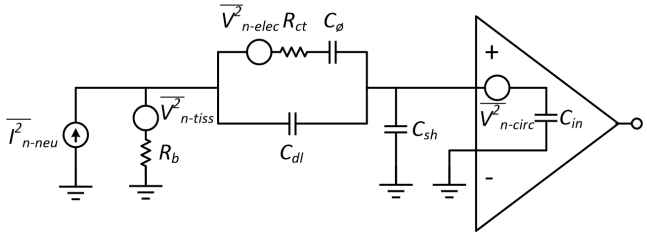

Fig. 5. Noise model for neural recording with microelectrodes [5].

D. Electrode Model

The noise of the electrode model consists of: tissue/bulk thermal noise, electrode-electrolyte interface noise, electronic noise, and biological noise [5], which are illustrated in Fig. 5. The tissue/bulk thermal noise can be expressed as

$$\bar{V}^2_{n-tissue} = 4kTR_b\Delta f \approx kT\frac{\rho_{tissue}}{r_s}\Delta f \tag{9}$$

where k is the Boltzmann constant, T is the temperature, R_b is the bulk resistance, ρ_{tissue} is the tissue/electrolyte resistivity, r_s is the radius of the electrode, and $\Delta f = f_2$-f_1 is the bandwidth of the measurement, with f_2 being the upper corner and f_1 the lower corner of the filter. The electrode noise can be expressed as

$$\bar{V}^2_{n-elec} = \frac{2kT\alpha}{\pi C_{dl}}\tan^{-1}2\pi R_{ct}C_{dl}\alpha f\Big|^{f=f2}_{f=f1} \tag{10}$$

where R_{ct} is the charge transfer resistor, C_{dl} is the double-layer capacitor, $C_\varnothing$ is the pseudocapacitor, and $\alpha = C_\varnothing/(C_\varnothing + C_{dl})$.

The biological noise is generated by the spiking activity of distant neurons and the values are extracted from *in vivo* measurements [5]. We use a fixed value for ρ_{tissue} in (9), which is 300 Ωcm. The configurable parameters for this model are electrode diameter (d_s), upper corner frequency (f_2), and lower corner frequency (f_1) of the filter.

E. Amplifier Model

Fig. 6 shows the circuit implementation for the first amplification stage (LNA), which is composed of an Operational Transconductance Amplifier (OTA) and a feedback network. The gain of the amplifier is given by the ratio between the two capacitances in the feedback network $G_{amp} = C_1/C_2$. The IRN is given by

$$\bar{V}^2_{irn-amp} = \left(1+\frac{1}{G_{amp}}\left(1+\frac{C_{in}}{C_2}\right)\right)^2\cdot\frac{16kT}{3g_m}\cdot f_{Neuron} \tag{11}$$

where C_{in} is the input capacitance of the OTA, g_m is the transconductance of the OTA, and f_{Neuron} is the action potential signal bandwidth [19]. The cut-off frequency of the low pass response can be adjusted by varying the load capacitance C_L.

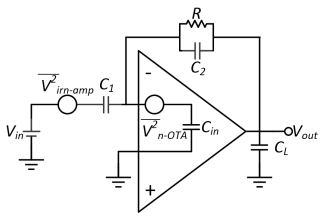

Fig. 6. Circuit implementation of the LNA [19].

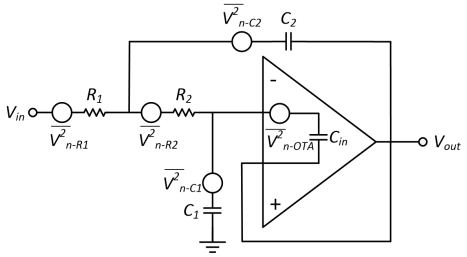

Fig. 7. Second-order low-pass filter circuit implementation [24-25].

The second amplification stage is implemented with non-inverting resistive negative feedback [19]. The configurable parameters for this amplifier model are gain and bandwidth. The low pass response is implemented with first order low pass Butterworth response.

F. Bandpass Filter Model

The band-pass filter is implemented as an active filter, constructed by cascading a high-pass and a low-pass filter. This method is chosen since this band-pass filter is used as a wideband filter, with a ratio of the upper and lower cut-off frequency over 2. The circuit implementation for the second order low-pass filter is illustrated in Fig. 7. It consists of resistors, capacitors, and an OTA [24-25]. Each of these components generate noise with different transfer function. The total noise at the output of this filter is derived from all the noise sources and their corresponding transfer function [24]:

$$\bar{V}^2_{n-lpf} = \sum_{l=1}^{m}|T_l|\,(V_n)^2_l \tag{12}$$

where T_l is the transfer function of the l-th noise source and $(V_n)_l^2$ is the l-th noise source. High-order filters are implemented by cascading several first and second order filters [25]. Filters with an even order number consist of second-order stages only, while filters with an odd order number include an additional first-order stage at the beginning. A high-pass filter is implemented by replacing the resistors of a low-pass filter with capacitors and its capacitors with resistors. The parameters that can be selected for the band-pass filter model are: filter type, filter order, and filter low and high cut-off frequency. The types of filter that can be chosen are Butterworth and Bessel. The user can choose between 2, 4, 6, or 8 as the filter order. Both the high-pass filter and the low-pass filter have the same filter order. The numerator and denominator constants for each type of filters are defined based on the order and the cut-off frequency chosen. These constants are then used by the Laplace transfer function solver in the SystemC MDVP framework to construct the desired band-pass filter [10-12].

G. ADC Model

The ADC model is implemented based on the Successive Approximation Register (SAR) architecture due to its small area and low power operation [19]. This architecture requires minimal amount of analog circuitry and offers a flexibility in varying the resolution for specific application. SAR ADC typically consists of a sample and hold circuit, a capacitive digital-to-analog converter (DAC) network, a comparator, and a SAR logic circuit (Fig. 8). A modeling and analysis of SAR

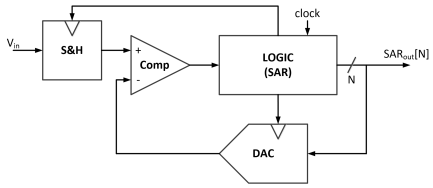

Fig. 8. SAR ADC ciruit diagram.

ADCs noise is illustrated in Fig. 9 [6]. The scheme in Fig. 9a is used to compute the ideal quantization noise $V_{n,qi}$. In this figure, an ideal ADC is used and V_{in} is the analog input, B is the SAR ADC resolution, and V_{DAC} is the reconstructed analog signal. In Fig. 9b, a non-ideal ADC is used and $V_{n,tot}$ is the measure of combined effect of quantization and physical noise in this noisy system. When $V_{n,tot}$ is subtracted by $V_{n,qi}$ obtained from Fig. 9a, the physical measure of the thermal noise $V_{n,th}$ is obtained. A statistical analysis in [6] derives the total ADC thermal noise as

$$\bar{V}^2_{th,ADC} = \frac{kT}{C_{tot}} + F^2_{nv,BC}\left(\frac{\sqrt{\frac{1.6\,kT}{C_{tot}} + V^2_{comp}}}{LSB}\right) \times \left(\frac{1.6\,kT}{C_{tot}} + V^2_{comp}\right) \quad (13)$$

where C_{tot} is the total capacitance of the capacitive DAC network, $F^2_{nv,BC}$ is the noise voltage gain factor in the Bit-Cycling (BC) phase, V^2_{comp} is the intrinsic noise of the comparator, and LSB is the ADC resolution in voltage. The SAR ADC model is built with configurable resolution (number of bits), input range, and sampling rate.

III. SystemC/SystemC-AMS Modeling

The complexity of Systems-on-Chips (SoCs) keeps increasing and has reached a state where a gate-level and Register Transfer Level (RTL) simulation are not feasible to characterize and explore the use-case scenarios. Building models at these abstraction levels are time consuming and the complete models are only available at a late phase of the system design, leading to slow simulation for complex systems. SystemC provides a possibility of higher abstraction level modeling, which includes fewer architecture details and higher simulation speed [11]. SystemC is a C++ class library that allows functional modeling of complex systems. SystemC-AMS is a SystemC analog and mixed-signals (AMS) extension, which is used to describe analog behavior of

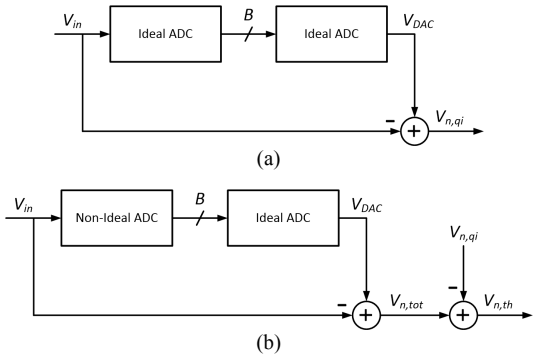

Fig. 9. (a) Model for ideal SAR ADC quantization noise. (b) Model for SAR ADC thermal noise [6].

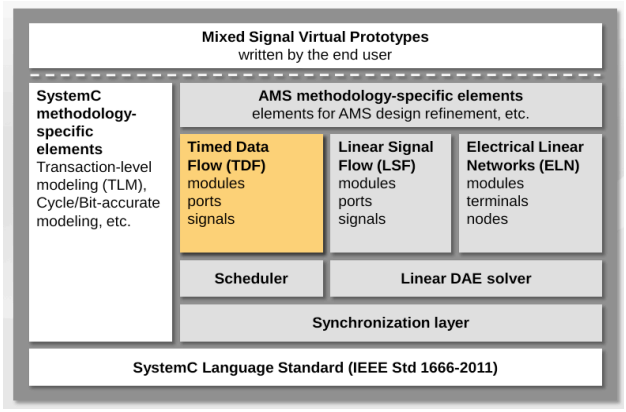

Fig. 10. SystemC AMS language elements [12].

electronic components [12]. SystemC-AMS language elements are illustrated in Fig. 10. SystemC-AMS supports 3 modeling formalisms, which are Timed Data Flow (TDF), Linear Signal Flow (LSF), and Electrical Linear Networks (ELN). TDF modeling is based on synchronous data flow modeling formalism and considers data as signal sampled in time. LSF modeling treats behavior as relations between variables of a linear algebraic equations set and the block diagram consists of predefined LSF modules, such as adders, multipliers, and integrators. An ELN network is composed of primitive modules (resistors, inductors, capacitors, voltage sources, etc.) which are interconnected by ELN nodes. The synchronization layer provides the possibility for multi-domain integration and design space exploration. We develop a SystemC MDVP platform for multi-domain parameter studies, such that an experimental analysis can be performed with biological parameters without the need for in vivo experiments. Scientists will be able to experiment with a large number of chemical compound concentrations and with a large number of neuron cells (10000) enabling them to verify measured parameters with simulation generated parameters. SystemC MDVP designs can be synthesized into gate-level HDLs using various synthesis tools, such as Xilinx Vivado_HLS [26] and Mentor Catapult C++ [27]. TDF modeling formalism is used to implement the proposed models. Fig. 11 and Fig. 12 shows part of the SystemC MDVP code for the proposed electrode model and band-pass filter model, respectively. The code in Fig. 11 shows the calculation of the electrode noises, while in Fig. 12 the band-pass filter numerator and denominator constants are calculated and used by the SystemC MDVP Laplace transfer function solver to construct the desired band-pass filter.

IV. Simulation Results

The functionality of the LNA model is verified by setting the gain to 100× and the bandwidth to 10 kHz. The input signal is the summation of three sine waves, each with 100 µV (-80 dB) amplitude and 1 Hz, 1 kHz, and 100 kHz frequency, respectively. Fig. 13 shows the output of the LNA model in frequency domain. The signal within the bandwidth is amplified by 100 (40 dB) from 100 µV (-80 dB) to 10 mV (-40 dB), while the signal outside the bandwidth stays at -80 dB. In verifying the band-pass filter model, we specified the filter

```
electrode(sc_core::sc_module_name nm, double diameter, double f1, double f2):
  in("in"), out ("out"), noise("noise"),
  diameter_(diameter), f1_(f1), f2_(f2)
  {
    Vn_tissue = sqrt(1.242*1.0e-18*(f2_-f1_)/(diameter_*1.0e-4/2));
    Vn_elec = 50.0*1.0e-6/diameter_;
    Vn_circ = 2.0e-6;
    Vn_neu = 1.0e-6;
    Vn_total = sqrt(pow(Vn_tissue,2) + pow(Vn_elec,2) + pow(Vn_circ,2) + pow(Vn_neu,2));

    noise_gauss1 =  new noise_gauss("noise_gauss1",pow(Vn_total,2),0);
    noise_gauss1->out(noise);

    adder2_1 =  new adder2("adder2_1");

    adder2_1->a(in);
    adder2_1->b(noise);
    adder2_1->c(out);
  }
```

Fig. 11. Part of the SystemC MDVP code for the proposed electrode model.

```
if (type == 1){     //Butterworth type filter
    if (order == 1){
        hpf_num (0) = 0;    //high pass filter Laplace transfer function
        hpf_num (1) = 1.0/(2.0*M_PI*low_cutoff_freq);
        hpf_denom (0) = 1;
        hpf_denom (1) = 1.0/(2.0*M_PI*low_cutoff_freq);

        lpf_num (0) = 1.0;  //low pass filter Laplace transfer function
        lpf_num (1) = 0;
        lpf_denom (0) = 1.0;
        lpf_denom (1) = 1.0/(2.0*M_PI*high_cutoff_freq);
    }
```

```
void processing(){
  double temp1, temp2;
  temp1 = ltf_hpf(hpf_num, hpf_denom, in);
  temp2 = ltf_lpf(lpf_num, lpf_denom, temp1);
  out.write(temp2);
}
```

Fig. 12. Part of the SystemC MDVP code for the proposed band-pass filter model.

type as Butterworth and the band-pass filter order as 4, thus the filter is constructed by a second order low-pass filter and a second order high-pass filter. The low cut-off frequency is set to 10 Hz and the high cut-off frequency to 10 kHz. The input signal for this simulation is the output of the LNA model simulation. Fig. 14 shows the output of the band-pass filter model simulation in frequency domain. The signal with 1 Hz and 100 kHz frequency are attenuated by -40 dB, which fits the roll-off of a second order Butterworth filter. The SAR ADC model is tested by setting the number of bits to 8, the input range between -0.5 V and 0.5 V, and the sampling rate to 512 kS/s. A 217.022 kHz sine wave with 0.5 V amplitude is used as the input signal for this simulation. The output of this simulation in frequency domain can be seen in Fig. 15. To simulate the multi-channel model, we make use of the neuron tissue model to produce 128 neural signals to be read by the electrodes. The multiplexing rate of the multiplexer (f_{s_mux}) and the sampling rate (f_{s_ADC}) of the SAR ADC are 512 kS/s (16 × 32 kS/s) since the ADC is used to convert the analog output of 16 PGAs. The simulation parameters for each block of the neural interface front-end are listed in Table I.

The electrode output in one of the channels can be seen in Fig. 16. The thermal noise and the biological noise are present in the output of this electrode. Fig. 17 shows the output of the PGA in this channel. In Fig. 18 we can observe the output of the multiplexer, which are used by this channel. More spikes, which come from the other 15 PGAs, are present.

TABLE I. NEURAL INTERFACE FRONT-END COMPONENTS SIMULATION PARAMETERS

Model	Parameter	Value	Unit
Neuron Cell	leakage conductance	0.3	mS/cm^2
	potassium equilibrium potential	-12	mV
	sodium equilibrium potential	115	mV
	leakage potential	10.613	mV
	membrane capacitance	1.0	µF/cm^2
Neuron Tissue	4 nearest cell AP attenuation	0.001	-
	12 further cell AP attenuation	0.0009	-
Electrode	diameter	50	µm
	low corner frequency	10	Hz
	high corner frequency	10k	Hz
LNA	gain	100	-
	bandwidth	10	kHz
Bandpass Filter	type	Butterworth	-
	order	4	-
	low cut-off frequency	10	Hz
	high cut-off frequency	10k	Hz
PGA	gain	15	-
Multiplexer	multiplexing rate	512	kS/s
SAR ADC	number of bits	8	bit
	input range	-0.5 to 0.5	V
	sampling rate	512	kS/s

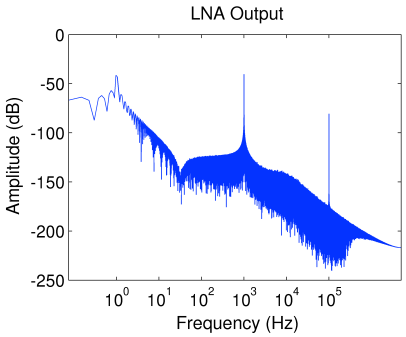

Fig. 13. LNA model output in frequency domain with 100× gain and 10 kHz bandwidth.

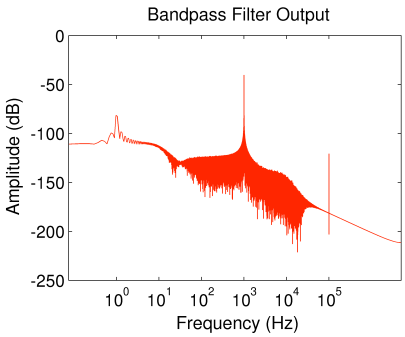

Fig. 14. Bandpass filter model output in frequency domain with 10 Hz low cut-off frequency and 10 kHz cut-off frequency.

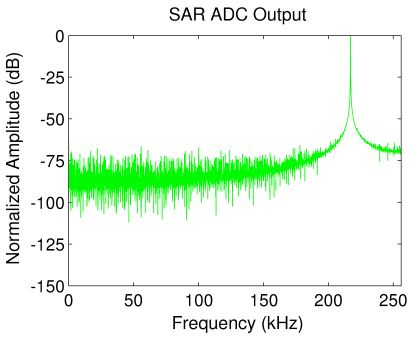

Fig. 15. SAR ADC model simulation output in frequency domain with 8 bit resolution and 512 kS/s sampling rate.

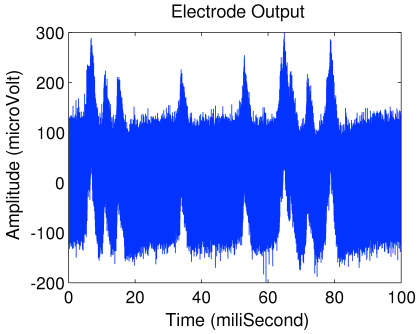

Fig. 16. Output of one electrode in the 128-channel neural interface front-end simulation.

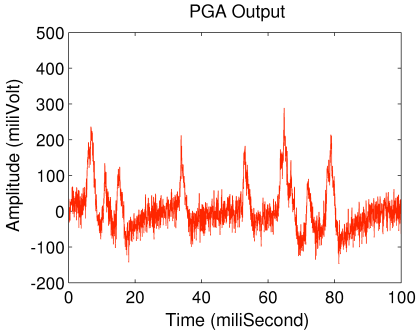

Fig. 17. Output of the PGA with the signal in Fig. 16 as the input.

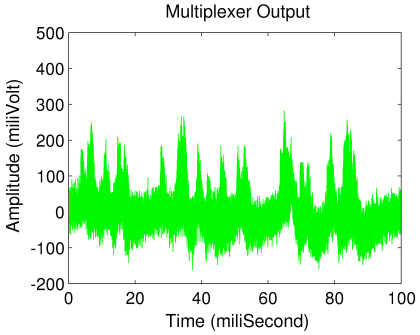

Fig. 18. Output of one of the multiplexers in the 128-channel neural interface front-end simulation.

V. Conclusion

We present a high-level behavioral model of a 128-channel time-multiplexed neural interface front-end implemented in a SystemC MDVP framework. The SystemC MDVP framework is the result of H-INCEPTION project and provides a quick design validation and supports multi-domain design integration. Chemical properties of the neuron cell model, biological characteristics of the tissue-electrode interface, and electrical components, both analog and digital, from the neural interface system can be integrated and simulated in a single environment.

A 128-channel neural interface front-end model is constructed by using 8×16-channel front-end recording circuits. Each channel consists of an electrode, a low noise pre-amplifier, a band-pass filter, and a programmable gain post-amplifier. 16 channels shares a SAR ADC through time-multiplexing. Neuron cell and neuron tissue model are built based on the HH model and the output of these models are used as the input signals for simulating the proposed multi-domain model. The models are verified with realistic settings and appropriate input signals.

References

[1] B. Gosselin, “Recent advances in neural recording microsystems,” *Sensors*, vol. 11, no. 12, pp. 4572-4597, 2011.

[2] C.M. Lopez, D. Prodanov, D. Braeken, I. Gligorijevic, W. Eberle, C. Bartic, R. Puers, and G. Gielen, "A multichannel integrated circuit for electrical recording of neural activity, with independent channel programmability," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 6, no. 2, pp. 101-110, 2012.

[3] A.L. Hodgkin and A.F. Huxley, “A quantitative description of membrane current and its application to conduction and excitation in nerve,” *The Journal of Physiology*, vol. 117, no. 4, pp. 500–544, 1952.

[4] W. Franks, I. Schenker, P. Schmutz and A. Hierlemann, "Impedance characterization and modeling of electrodes for biomedical applications", *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 7, pp. 1295 -1302, 2005.

[5] C.M. Lopez, M. Welkenhuysen, S. Musa, W. Eberle, C. Bartic, R. Puers, and G. Gielen, "Towards a noise prediction model for in vivo neural recording," *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 759-762, 2012.

[6] W.P. Zhang, X. Tong, "Noise modeling and analysis of SAR ADCs," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, in press.

[7] A. Zjajo, C. Galuzzi, R. van Leuken, “Stochastic noise analysis of neural interface front end,” *Proceedings of IEEE International Symposium on Circuits and Systems*, pp. 169-172, 2015.

[8] A. Zjajo, C. Galuzzi, R. van Leuken, “Sequential power per area optimization of multichannel neural recording interface based on dual quadratic programming,” *Proceedings of IEEE International Conference on Neural Engineering*, pp. 9-12, 2015.

[9] A. Zjajo, C. Galuzzi, R. van Leuken, ”Noise analysis of programmable gain analog to digital converter for integrated neural implant front end,” *Proceedings of IEEE International Conference on Biomedical Electronics and Devices*, pp. 5-12, 2015.

[10] H-INCEPTION Project website: https://www-soc.lip6.fr/trac/hinception.

[11] IEEE Computer Society, “1666-2011 IEEE standard SystemC language reference manual,” IEEE, 2012.

[12] Accellera SystemC AMS Working Group, “Standard SystemC AMS extensions 2.0 language reference manual,” Accellera Systems Initiative (ASI), 2013.

[13] S. Diao, Y. Zheng, Y. Gao, S.J. Cheng, X. Yuan, M. Je, and C.H. Heng, "A 50-Mb/s CMOS QPSK/O-QPSK transmitter employing injection

locking for direct modulation," *IEEE Transactions on Microwave Theory and Techniques*, vol. 60, no. 1, pp. 120-130, 2012.

[14] Y.C. Kuan, Y.K. Lo, Y. Kim, , M.C.F. Chang, W. Liu, "Wireless gigabit data telemetry for large-scale neural recording," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 3, pp. 949-957, 2015.

[15] D.Y. Barsakcioglu, Y. Liu, P. Bhunjun, J. Navajas, A. Eftekhar, A. Jackson, R.Q. Quiroga, and T.G. Constandinou, "An analogue front-end model for developing neural spike sorting systems," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 8, no. 2, pp. 216-227, 2014.

[16] I.H. Stevenson and K.P. Kording, "How advances in neural recording affect data analysis", *Nature Neuroscience*, vol. 14, no. 2, pp. 139-142, 2011.

[17] J. Guo, J. Yuan, and M. Chan, "Modeling of the cell-electrode interface noise for microelectrode arrays", *IEEE Transactions on Biomedical Circuits and Systems*, vol. 6, no. 6, pp. 605-613, 2012.

[18] C. Carboni, D. Loi, and G. Angius, "A microcontrolled neural interface with electrode impedance measurement," *2010 Conference on Ph.D. Research in Microelectronics and Electronics (PRIME)*, pp. 1-4, 2010.

[19] M.S.Chae, W. Liu, and M. Sivaprakasam, "Design optimization for integrated neural recording systems," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 9, pp. 1931-1939, 2008.

[20] K. Guillory and R. Normann, "A 100-channel system for real time detection and storage of extracellular spike waveforms", *Journal of Neuroscience Methods*, vol. 91, no. 1, pp. 21 -29, 1999.

[21] W. Wattanapanitch and R. Sarpeshkar, "A low-power 32-channel digitally programmable neural recording integrated circuit", *IEEE Transactions on Biomedical Circuits and Systems*, vol. 5, no. 6, pp. 592-602, 2011.

[22] C. Gold, D.A. Henze, C. Koch, and G. Buzsaki, "On the origin of the extracellular action potential waveform: A modeling study," *Journal of Neurophysiology*, vol. 95, pp. 3113-3128, 2006.

[23] Z. Nadasdy, J. Csicsvari, M. Penttonen, J. Hetke, K. Wise, and G. Buzsaki, "*Extracellular recording and analysis of neuronal activity: from single cells to ensambles. In: H. Eichenbaum and J. L. Davis (editors), Neuronal Ensambles*," New York: Wiley, 1998.

[24] D. Jurisic, G.S. Moschytz, "Low-noise active-RC low-, high- and band-pass allpole filters using impedance tapering," *10th Mediterranean Electrotechnical Conference, MELECON 2000*, vol. 2, pp. 591-594, 2000.

[25] T. Kugelstadt, "*Active filter design techniques*," Texas Instruments SLOD006A (Op Amps for Everyone, Chapter 16).

[26] Xilinx Vivado High-Level Synthesis website: http://www.xilinx.com/products/design-tools/vivado/integration/esl-design.html.

[27] Calypto Catapult website: http://calypto.com/en/products/catapult/overview/.