Real-Time Simulation of Passage-of-Time Encoding in Cerebellum Using a Scalable FPGA-Based System

Junwen Luo, Graeme Coapes, Terrence Mak, Tadashi Yamazaki, Chung Tin, and Patrick Degenaar

Abstract—The cerebellum plays a critical role for sensorimotor control and learning. However, dysmetria or delays in movements' onsets consequent to damages in cerebellum cannot be cured completely at the moment. Neuroprosthesis is an emerging technology that can potentially substitute such motor control module in the brain. A pre-requisite for this to become practical is the capability to simulate the cerebellum model in real-time, with low timing distortion for proper interfacing with the biological system. In this paper, we present a frame-based network-on-chip (NoC) hardware architecture for implementing a bio-realistic cerebellum model with $\sim$100 000 neurons, which has been used for studying timing control or passage-of-time (POT) encoding mediated by the cerebellum. The simulation results verify that our implementation reproduces the POT representation by the cerebellum properly. Furthermore, our field-programmable gate array (FPGA)-based system demonstrates excellent computational speed that it can complete 1sec real world activities within 25.6 ms. It is also highly scalable such that it can maintain approximately the same computational speed even if the neuron number increases by one order of magnitude. Our design is shown to outperform three alternative approaches previously used for implementing spiking neural network model. Finally, we show a hardware electronic setup and illustrate how the silicon cerebellum can be adapted as a potential neuroprosthetic platform for future biological or clinical application.

Index Terms—Cerebellum, field-programmable gate array (FPGA), network on chip (NoC), neural-rehabilitation, passage-of-time (POT).

J. Luo was with the Department of Mechanical and Biomedical Engineering, City University of Hong Kong, Hong Kong. He is now with the School of Electrical and Electronic Engineering, Newcastle University, Newcastle upon Tyne NE1 7RU, U.K. (e-mail: j.w.luo@newcastle.ac.uk).

G. Coapes and P. Degenaar are with the School of Electrical and Electronic Engineering, Newcastle University, Newcastle upon Tyne NE1 7RU, U.K. (e-mail:graeme.coapes@newcastle.ac.uk; patrick.degenaar@newcastle.ac.uk).

T. Mak is with the Department of Electronics and Computer Science, University of Southampton, Southampton SO9 5NH, U.K. (e-mail: terrencemak@gmail.com).

T. Yamazaki is with The University of Electro-Communications, Chofu, 182-8585 Tokyo. Japan (e-mail: ieee14@neuralgorithm.org).

C. Tin is with City University of Hong Kong, Kowloon, Hong Kong (e-mail: chungtin@cityu.edu.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

I. INTRODUCTION

SMOOTH and robust motor control requires precisely timed muscle activations at specific strengths. This is critically mediated by the cerebellum which functions to represent the passage-of-time (POT) over a range of tens to hundreds of milliseconds, and is essential for organizing movements of different body parts into a coordinated action [1]. Errors in POT encoding consequent to cerebellar damages can lead to dysmetria or delays in movements onsets in these patients [2]. Such condition, usually described as ataxia, cannot be cured completely at the moment, and is impacting millions of patients worldwide. To foster a potential cure based on neuroprosthetic technology, an efficient computational platform that can favorably mimic the complex function of the cerebellar neural network will be important. Fig. 1 shows a conceptual closed-loop system for a cerebellar prosthesis.

POT representation in the cerebellum is clearly evident in the classical Pavlovian delayed eyeblink conditioning [3], [4] where animals learn the inter stimulus interval (ISI), or POT, between the conditioned (CS) and the unconditioned stimulus (US) onsets upon repetitive training. It has been suggested that this information of POT is encoded in the extensive cerebellar granular layer. When excited by CS through mossy fibers (MFs), population of granule cells exhibit different bursting dynamics such that the sequence of active cells does not recur for a sufficiently long time. This forms a one-to-one correspondence between the active cell population and a time interval. Various computational models have been proposed to investigate the possible mechanism in the granular layer for POT representation. Four classes of such models have been reviewed in [5], which includes the delay line model [6], [7], spectral timing model [8], oscillator model [9] and random projection model [10], [11]. Among these computational models, the random projection model is suggested to be both a robust and biologically plausible framework in the representation of POT, which can also be used to reproduce the classical Pavlovian delay eyeblink conditioning. This spiking network model makes use of two critical properties of the cerebellar granular-Golgi layers: 1) extensive random recurrent connections between granule and Golgi cells; and 2) long temporal integration of input signals by the NMDA receptors, which are both evident in the biological systems.

Thus far, this large-scale ($\sim$10^6 cells) spiking network model of cerebellum has been investigated by software simulation using PC and GPU implementation [11], [12]. However, in order to use the model in real-time biological experiments, particularly *in vivo*, some form of compact digital real-time implementation with versatile I/O's would prove valuable with

noise free, scalable communication bandwidth and precise timing management features. A scalable hardware platform that can be tailored designed and takes advantage of highly parallel computing capability would be greatly preferred. Such a system would be a powerful tool to help explore the POT mechanism and related disease mechanisms in the cerebellum. Future neuroprosthetic developments could also benefit from an efficient hardware platform for implementing a large-scale spiking network model for real-time computation.

In general, CPU based process platforms are limited by their sequential computing architecture. The large latency makes them difficult to be used in real-time brain-machine interfaces (BMI). GPUs, on the other hand, are capable of parallel computing but are constrained by memory and communication bandwidth issues [13]. Models can be implemented directly onto CMOS, [14], [15] but a single implementation can be time consuming. Field-programmable gate arrays (FPGAs) are a versatile reconfigurable digital computational platform which can be used for both direct computational implementation and as a stepping stone to compact low power CMOS chip implementation. It contains massive flexible programmable logic with concurrent operation allowing direct use in bench-top *in vitro* and constrained *in vivo* systems. If designs are then translated to CMOS, the subsequent chips can be applied to implantable neuro-prosthetic devices. In recent years, FPGAs are extensively used in neural system modeling and simulation of large scale biologically realistic neural systems [16]–[19].

Hardware implementations of cerebellar neural networks for neuroprosthesis have already attracted the interest of neuroscientists and engineers. Bamford *et al.* [15] has designed a VLSI field-programmable mixed-signal array to produce the eyeblink conditioning performances by modeling the cerebellum system. This has been fabricated as a core on a chip prototype intended for use in an implantable closed-loop prosthetic system aimed at rehabilitation of associated behavior. While they have demonstrated a proof-of-concept of success in their implementation, a highly simplified neural model with abstract modeling of cerebellar information processing is used in the work. Such simplification is convenient for hardware implementation, but lacks direct physiological correspondence for quantitative comparison with the biological system. In contrast, Yamazaki and Tanaka's model [11] is more biologically realistic and pays specific attention to the role of the granular-Golgi layer in timing and gain control by the cerebellar cortex to reproduce experimental results. However, this comes with the cost of a significant increase in the size and complexity of the computational model in order to produce a robust system behavior. As such, an efficient implementation is required to overcome these computational challenges, especially when real-time application is required.

Previously we presented the concept of an FPGA-based network-on-chip (NoC) hardware architecture for implementing the granular layer of random projection cerebellum model [20], [21]. It produced a network behavior of POT representation consistent with the simulation results presented in the original paper by Yamazaki and Tanaka [11]. In this work we have made a more in-depth investigation on the details of implementation and analysis of system performance. The system contains ~100 000 granule cells and ~1000 Golgi cells, using
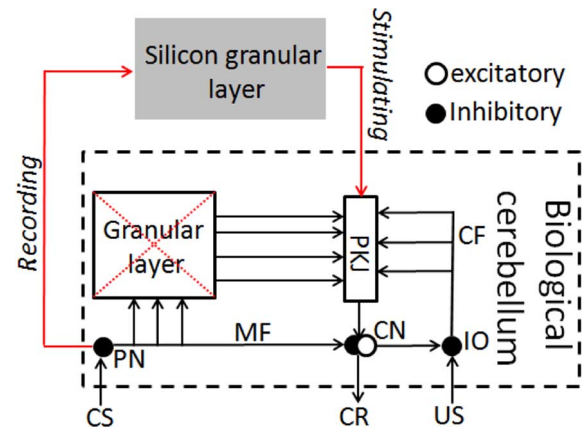

Fig. 1. The conceptual closed-loop system for cerebellum passage-of-time (POT) prosthetic. Damaged biological granular layer is replaced by FPGA-based granular layer system. CS is a conditioned stimulus while the US is an unconditioned stimulus. MF is the mossy fiber and CF is the climbing fiber, PKJ is the Purkinje cell. PN is the precerebellar nucleus, CN is the cerebellar nucleus, and CR is the conditioned response.

a conductance-based, leaky integrate-and-fire neuron model. The parameter values all have experimental basis, such that the network model produce realistic firing behavior. In particular, three accomplishments are highlighted in this paper: 1) we have reproduced the granular layer firing patterns for representation of POT in real-time under normal as well as pharmacologically perturbed conditions; 2) our architecture allows for efficient scalability to 100 000 neurons and beyond and can be used for more complex biological neural network applications; and 3) we have eliminated multiplexing timing errors and allows for network profiling at key time points.

II. THE PASSAGE-OF-TIME COMPUTATIONAL MODEL

The cerebellar granular layer consists of two main cell types, namely the granule cells and Golgi cells. Input signal from the pre-cerebellar nucleus to the granule cells is conveyed by MFs (Fig. 1). The spiking network of cerebellar granular layer proposed in [11] is modelled as a 1 mm^2 virtual sheet composed of a square lattice arrangement of $32*32$ Golgi cells and glomeruli, and $320*320$ granule cells. The same network with minor changes is used in this paper. Fig. 2 describes the topology between Golgi and granule cells.

Fig. 2(a) illustrates the topology of our granular layer model which contains 1024 granule-cell clusters and Golgi cell, the different colors represents communities of closely connected cells within the network. Each granule-cell cluster contains 100 granule cells. The size of the circles is proportional to the number of other clusters that it is connected to. Each dot represents one granule-cell cluster and one Golgi cell, as is shown in Fig. 2(b). Every Golgi cell receives excitatory input from its nearest granule-cell cluster, while Golgi cells project randomly to the nearby granule-cell clusters such that each granule-cell cluster receives inhibitory inputs from ~8 Golgi cells on average. The probability distribution of number of synaptic connection from Golgi cell to granule-cell cluster is shown in Fig. 2(c).

The equations for modeling the neurons and analysis have been detailed in [11] and we briefly repeat the key ones here. The
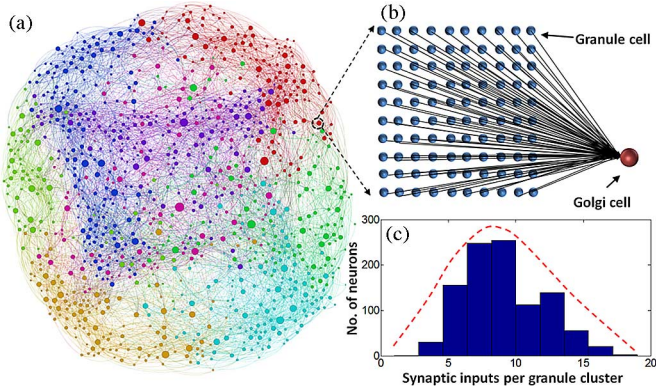
Fig. 2. The topology of granular layer model. (a) Contains 1024 granule-cell clusters and Golgi cell, the different colours represents communities of closely connected cells within the network. The size of the circles is proportional to the number of other clusters that it is connected to. (b) Each dot represents one granule-cell cluster and one Golgi cell. (c) The synaptic input number distribution.

granule and Golgi cells were modelled as conductance-based, leaky integrate-and-fire units, as described in (1)

$$C\frac{dV(t)}{dt} = g_{\text{leak}}(E_{\text{leak}} - V(t)) + g_{\text{ex:AMPA}}(t)(E_{\text{ex}} - V(t))$$
$$+ g_{\text{ex:NMDA}}(t)(E_{\text{ex}} - V(t)) + g_{\text{inh}}(t)(E_{\text{inh}} - V(t))$$
$$+ g_{\text{ahp}}(t - \hat{t})(E_{\text{ahp}} - V(t)) \tag{1}$$

where $V(t)$ and C are the membrane potential at time t and the capacitance, respectively, E's are the reversal potential and $\hat{t}$ denotes the last firing time of the neuron. The membrane potential depends on five types of currents: α-amino-3-hydroxy-5-methyl-4-isoxazolepropionic (AMPA) receptor-mediated, N-methyl-D-aspartate (NMDA) receptor-mediated, leak current, inhibition current and the after-hyperpolarization current. The conductance, $g(t)'s$, are calculated by convolving the alpha function $\alpha(t)$ with the spike event $\delta_j(t)$ of presynaptic neuron j at time t as follows:

$$g_c(t) = \bar{g}_c \sum_j w_j \int_{-\infty}^{t} \alpha(t - s)\delta_j(s)ds \tag{2}$$

where $\bar{g}_c$ is the maximum conductance and w_j is the synaptic weight from presynaptic neuron j. A neuron fires a spike at time t ($\delta_j(t) = 1$) when its membrane potential exceeded a threshold θ, and the after-hyperpolarization would follow. The conductance for the after-hyperpolarization was given by

$$g_{ahp}(t - \hat{t}) = \exp(-(t - \hat{t})/\tau_{\text{ahp}}. \tag{3}$$

We followed the same analysis procedures as in [11] for evaluating the POT behavior produced by the simulation model. We first computed $z_i(t)$ which represents the average activity of a granule-cell cluster i.

$$z_i(t) = \frac{1}{\tau}\sum_{s=0}^{t} \exp\left(-\frac{t-s}{\tau}\right)\left(\frac{1}{N_{gr}}\sum_j \delta_j(s)\right) \tag{4}$$

where $\delta_j(s)$ is the spike event in the granule cell j in the cluster at time s, N_{gr} is the number of granule cells in a cluster (100 in this case) and τ is the decay time constant, which was set at 8.3 ms.

How the activity patterns of granule cell clusters evolved over time is evaluated based on the similarity index, $S(\Delta t)$. We first computed the autocorrelation of the activity pattern between time t and $t + \Delta t$ as follows:

$$C(t, t + \Delta t) = \frac{\sum_i z_i(t)z_i(t + \Delta t)}{\sqrt{\sum_i z_i^2(t)}\sqrt{\sum_i z_i^2(t + \Delta t)}}. \tag{5}$$

$C(t, t + \Delta t)$ takes the value between 0 and 1 since $z_i(t)$ is always non-negative. It would be 1 if the activity pattern vectors $z_i(t)$ and $z_i(t + \Delta t)$ are identical, and it would be 0 when they are orthogonal, indicating that the activity patterns have no overlap. Then the similarity index is computed as the timed average of (5) over the CS duration, T, shown as follows:

$$S(\Delta t) = \frac{1}{T}\sum_{t=0}^{T} C(t, t + \Delta t). \tag{6}$$

$S(\Delta t)$ represents how two activity patterns separated by Δt are correlated, on average. If the similarity index decreased as Δt increased, it indicates that an activity patterns evolved with time into uncorrelated patterns.

We further computed the reproducibility index $R(t)$ as follows:

$$R(t) = \frac{\sum_i z_i^{(1)}(t)z_i^{(2)}(t)}{\sqrt{\sum_i z_i^{(1)2}(t)}\sqrt{\sum_i z_i^{(2)2}(t)}} \tag{7}$$

where $z_i^{(1)}(t)$ and $z_i^{(2)}(t)$ are the activity patterns of granule-cell cluster i at time t for two different input signals. The reproducibility index quantifies how activity patterns elicited by two different input signals differ from each other over time and serves as a measure for the robustness of the POT representation by the network model.

III. HARDWARE ARCHITECTURE DESIGN

To implement the POT model, we propose a frame-based network on chip (NoC) hardware architecture on FPGA. The conceptual structure is shown in Fig. 3.

In Fig. 3, the left side shows the n by m frame based NoC system, where the size can be adjusted as needed. The architecture consists of three main components: the neural processor, the router, and the global controller. In this work, we implemented a NoC system containing 48 processors, which calculates the neural activates. Each processor implements 2000 granule cells and 20 Golgi cells with connection ratio of 100:1. The router is used for implementing the inhibitory connections from Golgi cells to granule-cell clusters. The interface modules packetize spike events received from the processor ready for transmission through the network. When the interface modules receive packets the message is decoded and transmitted to the required cells within the neural processor. Finally, a frame master is implemented to coordinate neural and communication processing periods.
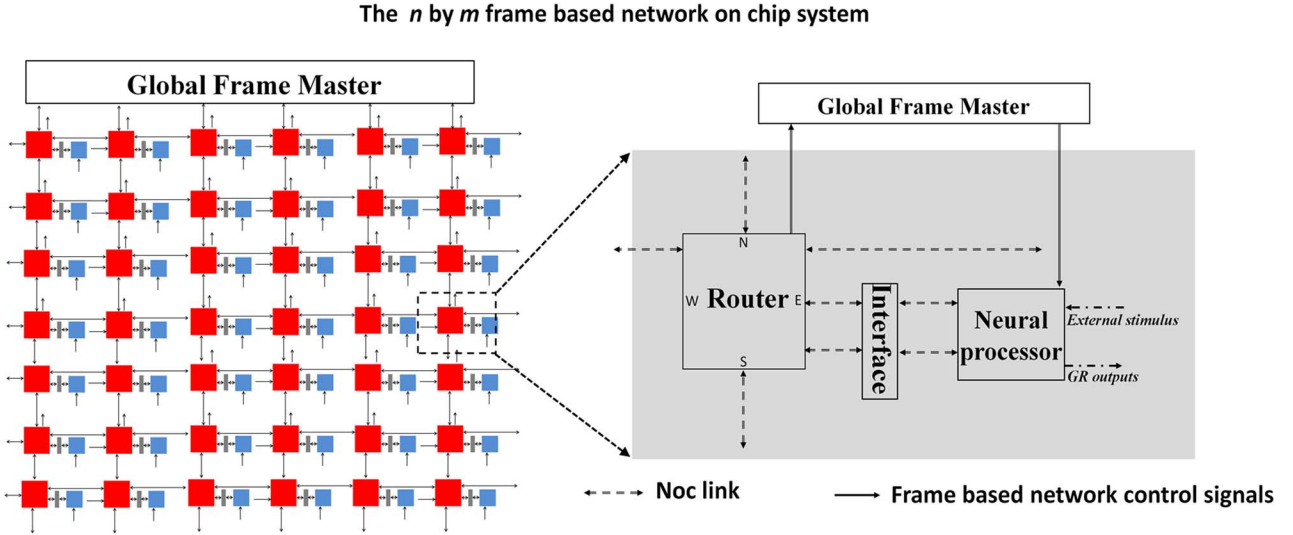
The *n* by *m* frame based network on chip system

Fig. 3. A conceptual FPGA-based network on chip hardware architecture. The figure on the left is the scalable *n* by *m* structure of frame based network on chip system. It contains n ∗ m neural processors, n ∗ m routers and one global controller. This architecture can be scaled up depending upon on the required model. In this paper, we implemented a network on chip system which contains 48 processors. On the right, there is a detailed structure of a module. The neural processor calculates the neural activity, with each processor implementing 2000 granule cells and 20 Golgi cells with connection ratio 100:1. The router is for implementing the connections from Golgi to granule-cell clusters. The interface modules packetize spike events received from the processor ready for transmission through the network. When the interface modules receive packets the message is decoded and transmitted to the required cells within the neural processor. Finally, a frame master is proposed to coordinate neural and communication processing periods.

A. Neural Computing

The neural processor data path is shown in Fig. 4. Two types of neurons are implemented in the processor, the granule cell (GR) and the Golgi cell (GO). Both models use the same hardware architecture but with different parameters. Each granule-cell cluster, containing 100 granule cells, connects to one Golgi cell. The activities (1 or 0) of all the 100 granule cells will be first calculated; whilst an accumulator will add all of them together and at the 100th clock cycle send the summated value to the Golgi cell model as an excitatory input [Fig. 4(a)].

Fig. 4(b) details the data path inside the neural model, which takes two computing stages: ion channel computing and integration. Each stage takes 4 clock cycles. Because computation is performed in parallel, the latency in each individual path has to be consistent; therefore appropriate delay blocks (the rectangular blocks) are added as necessary.

Fig. 4(c) and (d) show the sub-component circuits, including the inhibition and excitation circuits and FIFO-based delay circuits. Since each neural processor implements 2000 granule cells and 20 Golgi cells, a pipelining technique is applied for reducing hardware resources. A long pipelining stage is required for storing granule cells calculation intermediate values. A First-In First-Out (FIFO) based delay circuit is designed for achieving long computational stages.

B. Network-on-Chip

To manage the transmission of action potentials from Golgi cells to granule-cell clusters we have developed a NoC infrastructure. This system allows for arbitrary connectivity between Golgi cells and granule-cell clusters. Each processing element is connected to a router through which the action potentials are communicated. The routers are connected together in a mesh topology [22] as shown in Fig. 5(b).
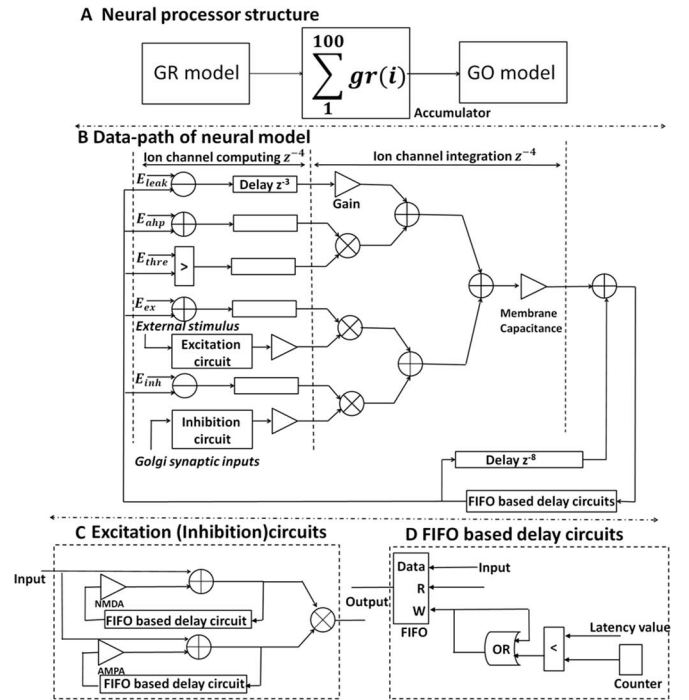

Fig. 4. The neural processor structure and the data path of neural model. (a) shows the conceptual structure of the processor. (b) shows the data path of the neural model. Both GR and GO models use the same hardware architecture but with different parameters. The rectangular block is the delay function and triangle block (gain) is the different ion channel conductance's which refer to (2). (c) and (d) show the sub-component circuits: excitation (inhibition) circuits and FIFO based delay circuits. The triangle blocks denote the NMDA and AMPA receptor conductance.

When a Golgi cell produces an action potential the interface fetches a list of destination granule-cell clusters from memory, an individual packet is generated to be sent to each of these
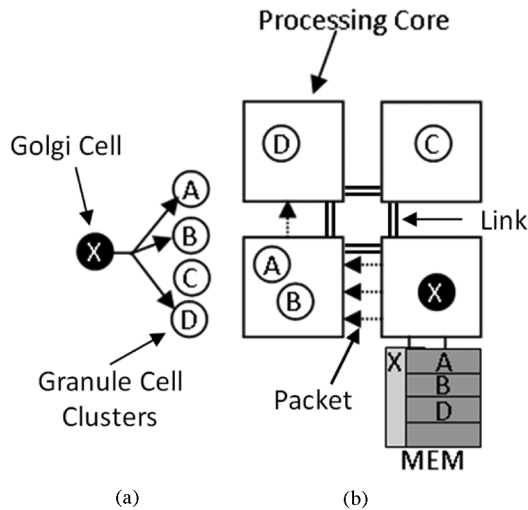
Fig. 5. Example of mapping of neural network to a network-on-chip. (a) Neural Network. A sample Golgi neural network with a single Golgi cell connected to three out of four granule-cell clusters. (b) Network-on-Chip. Four processing cores are shown. Each core may model multiple Golgi cells. When the Golgi cell X produces an action potential, individual packets are transmitted to each connected granule-cell cluster. The targeted granule-cell clusters are distributed throughout the mesh NoC.

Fig. 6. The frame master performances. In the frame 1, the router processing time is longer than the processor's, so the frame master temporally disabled the neural processor at $t_3 - t_4$ periods until the router finished its current traffic loads. While in the frame 2 and 3, because of the routing time is shorter than the processor time, the processor clock was continuously running.

TABLE I
STANDARD SPIKE PACKAGE FORMAT

Name	Number of bits	Value
Golgi Spike Packet	2-bit	x0
Configuration Packet	2-bit	x1
Core ID	6-bit	\
Cluster ID	5-bit	\

destinations within the network accordingly. The connectivity of the neural network can be updated by adjusting the contents of the memory. A user may alter the contents of the memory to adjust the connectivity by injecting configuration packets into the network. This can be done at start-up or part way through simulation if required by halting the system using the global frame master.

The packet format is shown in Table I. Packets are classified by the setting of a 2-bit type identifier. The generated spike packet contains the address of the targeted granular cell (Core and Cluster ID), allowing for the routers to direct the packet to the correct processing elements. Each granule-cell cluster summates the packets received. This summed value is used as an inhibitory input into the granule-cell clusters. Packets are transmitted between routers using a 4-phase asynchronous protocol [23] and a parallel data bus. The routers are output buffered using a 2-deep FIFO memory element. To inspect the state of the model the network-on-chip is also responsible for transmitting information externally. When a Golgi cell produces an action potential, a 'Golgi Message' packet is also transmitted to a specialist processing element. This processing element buffers all received packets and transmits these packets to a PC. This allows for a user to review the state of each Golgi cell at any time.
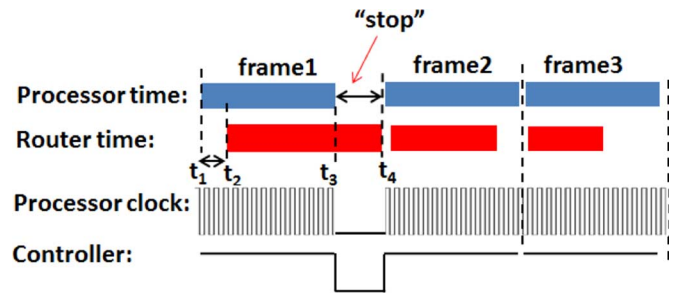
C. Frame Master

In order to maintain synchronicity within the system a frame master is used. The master is responsible for ensuring that all packets are transmitted to their destination before the processing elements start to process the next time step. This ensures that the granule-cell clusters receive all their updates within the correct time period.

For example, as shown in Fig. 6, the time required for network communication depends on the load of the network, which is determined by the frequency of Golgi cells spiking and the network topologies. This varies for each frame. In each frame, once the first Golgi cell spike event is released (at time t_2), the router starts to process the corresponding synaptic packages. After all 20 Golgi cell spike events are computed (at time t_3), the processor's duty in frame 1 is finished. Then the neural processor needs to start computing the next 20 Golgi cell activities for frame 2. However, at the end of frame 1 (time t_3), the network has not completed its communication for the current 20 Golgi cells. Therefore extra time is allocated for the network to finish this task before frame 2 begins. As results of this, the frame master generates a low level signal that disables the processor clock for time $t_3 - t_4$ until the network has completed its routing task. The frame master then enables the processor to allow it to start computing again (time t_4).

IV. RESULTS

A. Hardware Simulation Results for Passage-of-Time (POT)

Fig. 7 shows a comparison of the membrane potential of a single granule (1) neuron model simulated by the FPGA neural processor and by software (implemented in C with floating-point data type). A fixed point system with 40-bit and 22-fractional bit is employed in this FPGA system, and this length of bits has been selected to guarantee each operation to have sufficient precision to avoid data overflows and mismatch. The same inputs (30 Hz Poisson spike train) were given to both simulations.

The two simulations produce essentially identical results with very minor differences due to hardware truncation errors. This validated the hardware implementation of the neural model. Increasing the length of bits can eliminate truncation errors but introduce resources utilizations waste.
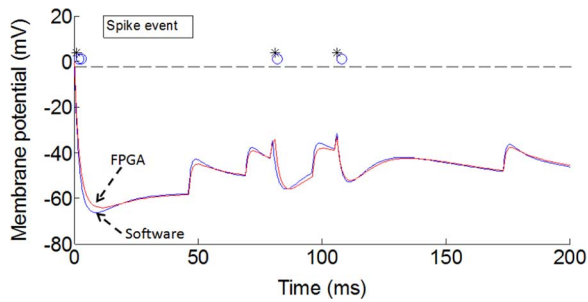
Fig. 7. The comparison results of a fundamental granule (1) neuron model simulated by the FPGA neural processor and CPU. The CPU implementation is the original software described in [11], running with an Intel Quad Core™ i7 CPU with 8 GB of RAM under the Ubuntu operating system.

Then we investigated the simulation results of the complete network model. The hardware POT simulation results are summarized in Fig. 8. Poisson spikes were fed into the simulated network to represent CS inputs through MFs. The simulated network was first fed at each MF with 5-Hz Poisson spikes for 300 ms to set the network to steady-state then 30-Hz Poisson spikes, preceded by 5 ms 200-Hz spikes, are given to excite the network.

Fig. 8(a) shows the spike patterns of 40 granule cells randomly chosen from different granule-cell clusters. These granule cells show different temporal activity patterns. Specifically, they show a random repetition of transitions between bursting and silent states. These bursts are sustained for tens to hundreds of milliseconds. In contrast, the Golgi cells fire rather regularly as shown in Fig. 8(a) bottom panel. Fig. 8(b) shows the similarity index of the activity pattern against the time shift Δt (6). The gradual decrease of the similarity index with $|\Delta t|$ demonstrates a smooth encoding of POT from the onset of CS, indicating that the populations of active granule cells change gradually over time such that no active granule-cell clusters appear more than once throughout the stimulation. Both of our software and hardware simulation results are consistent with results shown in [11], which confirms a proper POT behavior in our simulation, in that the sequence of active granule cell population maintains a one-to-one correspondence with the POT from the CS onset. The hardware simulation result is well comparable with software simulation with mean error being less than 5% [Fig. 8(b)]. The error is mainly caused by hardware truncation errors. Fig. 8(c) shows the reproducibility index (7) from the hardware simulation, which compares the activity pattern generated by two different Poisson spike inputs. The reproducibility index remains high (>0.7), indicating that the POT encoding will remain robust despite of variability of signals in the two stimulating inputs through MFs. This shows that the neuron population can maintain consistent POT representation across trials when, for instance, learning of delayed eyeblink conditioning over multiple training sessions is to be incorporated in the model [11].

B. Effects of Blocking NMDA Channel on POT Representation

To further verify our hardware simulation results, we also investigated the effect of blocking NMDA channels, which play a

critical role in delayed eyeblink conditioning [24], in our simulations. The hardware and software simulation results are summarized in Fig. 8(d)–(f). When NMDA channels are blocked in either granule cells or Golgi cells, granule cells lose the temporal structure in their firing, instead, they fire spikes at a rather regular manner [Fig. 8(d)]. The similarity index becomes flat except for $|\Delta t|$ smaller than $\sim$30 ms. Within the time scale of 30 ms, there are very limited number of spikes to encode robust temporal structure for POT. On the other hand, 30 ms is too short for physiologically relevant POT in a classic delayed eyeblink conditioning experiment. Hence, the GR firing pattern after NMDA-R blockade cannot capture a temporal structure at a time scale of physiologically relevance. The disruption of POT encoding consequent to NMDA channels blockade is reflected by both software [Fig. 8(e)] and hardware simulation [Fig. 8(f)] The results (both software and hardware) are consistent with those presented in [11].

C. Network on Chip Performance

To investigate the performance of the NoC infrastructure, we replaced the processing elements with configurable random packet generators. Packets were then injected into the network at a defined rate and the latency and throughput of these packets analyzed. The results for a 48-core system involving 960 Golgi cells, identical to the networks described in the sections above, are highlighted in Fig. 10. As the mean firing rate of the Golgi cells increases the median and range of latencies increases slightly. However, all packets are transmitted in under 100 processor clock cycles, which is the time it takes to update the state of a single Golgi cell. No packets are lost at any of the measured input frequencies, indicating that the frame master should not be required apart from when the Golgi firing rate exceeds expectations or when the user intervenes. Within typical cerebellar systems, the Golgi cells fire at a rate of between 40 and 60 Hz, which is within the defined network performance characteristics.

D. FPGA-Based Granular Layer for Neural-Rehabilitation

We illustrated a hypothetical *in vivo* experimental setup for closed-loop prosthetic application using our FPGA granular layer system in Fig. 9(a). Biological neuronal spike signals would be recorded by using a multichannel neural recording system from the pontine nucleus or from the mossy fibers, which would then be used as inputs to the silicon granule layer model. These neuronal spikes will be processed by the silicon-granular layer, which then generate the appropriately timed output discrete spikes to trigger the stimulation to be injected into the animal. Fig. 9(b) shows an electronic system setup to demonstrate such experiment. A Virtex-5 board is employed to simulate the neural spikes inputs conveyed by MFs, which are delivered to the FPGA cerebellum model via four bit wires. The input discrete spikes are modeled as two 5 Hz and two 30 Hz Poisson spike trains in 4-bits signals. The proposed silicon granular layer is implemented on the Virtex-7 board with the I/O interface for displaying the system output on the oscilloscope in real-time [Fig. 9(c)]. The displayed GR spikes were taken from three neural processors. The frame-based signal is also shown which is used to monitor and verify system
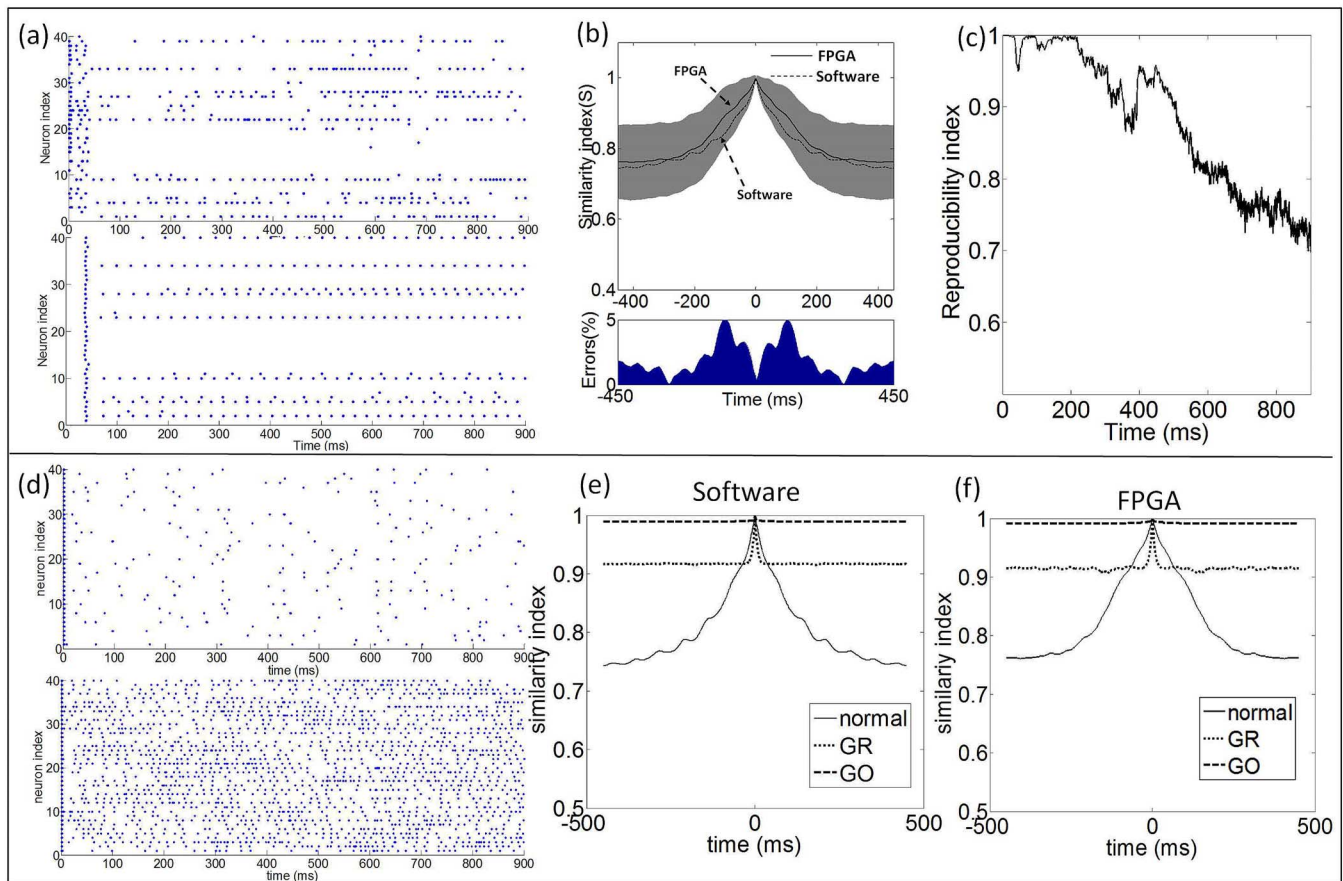
Fig. 8. (a) Spike patterns of 40 granule cells and Golgi cells chosen randomly in an implemented granular layer. (b) Comparison of similarity index between software and FPGA simulations. The grey areas are the standard deviations of the hardware results. The errors between the two results are shown at the bottom. The maximum error is less than 5%. (c) The reproducibility index is calculated by (5). It maintains a high value which suggested a robust POT representation despite the input variability. (d) Spike patterns of 40 granule cells when NMDA channels of granule cells (upper panel) and Golgi cells (lower panel) were blocked. Each neuron was chosen randomly from 40 different granule-cell clusters. The firing of the cells become rather regular and hence lost the capability of encoding temporal information of POT. (e), (f) Comparison of similarity index between software and FPGA simulations when NMDA channels of granule cells (dotted line) or those of Golgi cells (dashed line) were blocked. The similarity indices become flat indicating a loss of temporal structure in the granule cells' activity pattern.

processing behaviors. When the task of each frame is finished, the frame-based signal is changed to a high level value, and each frame uses 25.6 us (the time between X1 and X2) to mimic 1 ms real-world activities. Hence, this setup can complete 1 sec real-world activities in 25.6 ms at full speed as shown at Fig. 11. The system specifications are summarized in Table II.

V. DISCUSSION

A. Scalability of Different Platforms

In Fig. 12 we compare the performance of our design with three alternative approaches previously proposed for implementing spiking neural network (CPU, GPU and multi-core bus). In addition to its higher computational speed, our FPGA-based NoC approach clearly demonstrates scalability compared with the other approaches. The computation time remains almost constant even if the network size increases by an order of magnitude.

An alternative is to use GPU processors which can supplement or even replace CPU's for parallelizable code. The rise

of GPU languages such as CUDA and Open CL have simplified their use enormously. Modern GPU's exceed 5000 cores and can increase processing speed by orders of magnitude for parallelizable tasks [25], [26], [12]. Additionally, GPU's offer extremely high raw memory bandwidth, though this is difficult to achieve in practice and requires adhering to strict memory access patterns [25]. Nevertheless, with sufficient power, it is possible to implement spiking neural networks for high speed computation on a massively multi-core GPU. However desktop systems require relatively large power consumption and are not scalable to prosthetic devices. Mobile GPU systems found on typical mobile phones are significantly more power efficient, but have fewer cores, and the shifting incoming/outgoing data via the CPU would significantly reduce their effectiveness. We therefore chose an FPGA platform with large numbers of I/O's for potential *in vitro* and *in vivo* operation.

One key difference between our FPGA platform and processor based implementations is that we utilize distributed, localized memory banks that avoid sharing of global memory resources. This avoids delays associated with accessing global memory and reduces power consumption by minimizing the
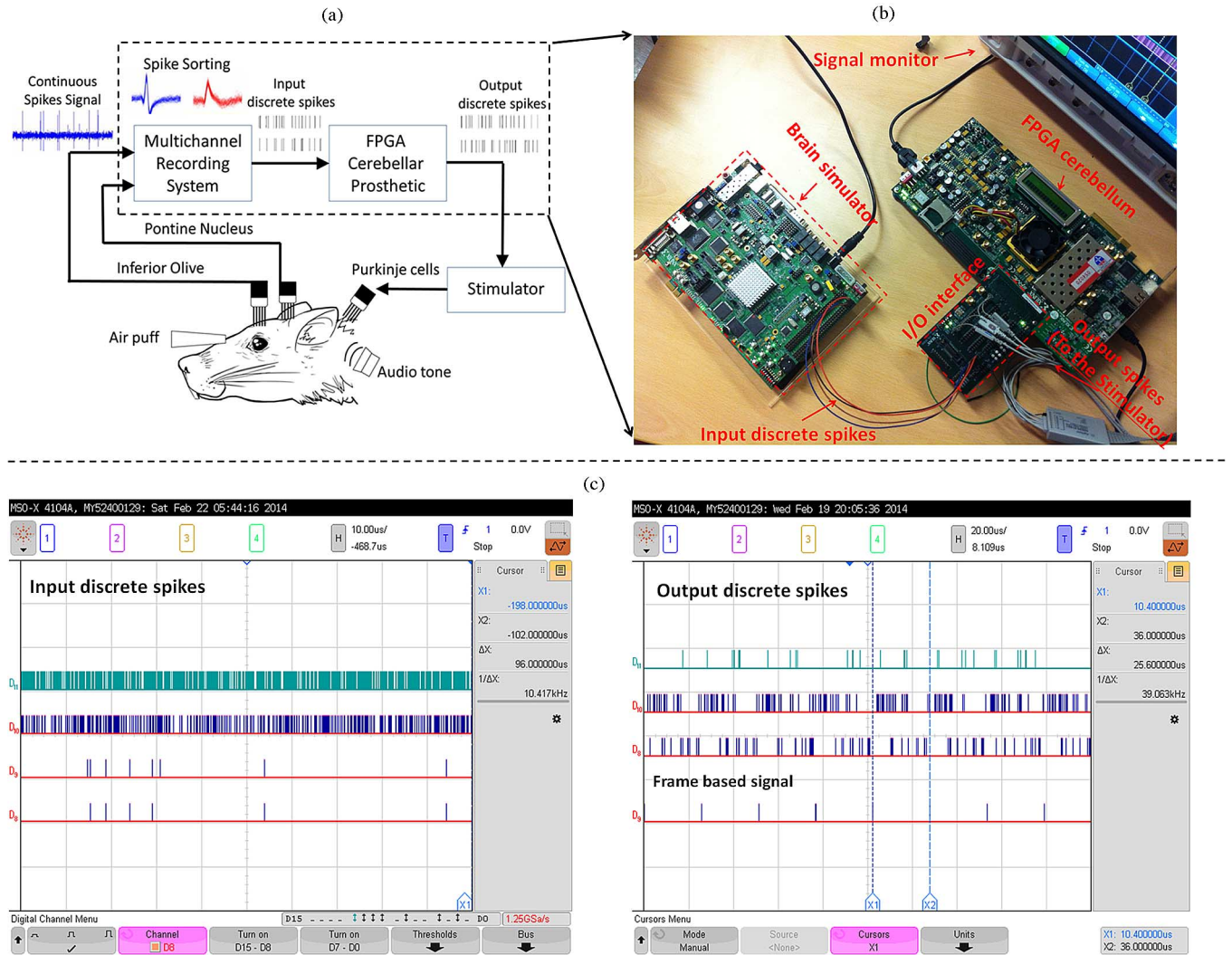
Fig. 9. The overall system experimental setup. (a) The closed-loop prosthetic system. The hypothetical *in vivo* closed-loop experimental setup for cerebellum rehabilitation. (b) The electronic experimental setup. An electronic setup to demonstrate the feasibility of the *in vivo* experiment. A Virtex-5 board (brain simulator) is employed to simulate the biological spikes conveyed by MFs, which are delivered to the FPGA cerebellum model via four bit wires. The input discrete spikes are modeled as two 5 Hz and two 30 Hz Poisson spike trains in 4-bits signals. The proposed silicon granular layer is implemented on the Virtex-7 board with the I/O interface for displaying the system output on the oscilloscope in real-time. (c) The real-time Input/Output discrete spikes. Shows the real-time input/output discrete spikes and the frame-based signal. When interfaced with an animal in a behavioral experiment, the output of the FPGA could be linked to the stimulators for delivering timed electrical pulse stimulation to the brain. The outputs can alternatively be linked to model Purkinje cells which then be linked to the stimulators.

size and operating frequency of channels between processors and memory.

The memory usage grows linearly with an increasing number of cells. The major memory consumption is within the storage of connectivity information between Golgi and granule cells. On average, each Golgi cell is connected to 8 granule cells, so the memory requires only 8 words per Golgi cell to store this connectivity information. For each additional Golgi cell another 8 words is required.

A further variance on previous work is the use of frame based encoding. One issue with real-time NoC systems is that spiking information encoded in latency or frequency can be prone to distortion due to congestion [27]–[29]. In contrast, we utilize a stop-start approach whereby all the neural spikes processed and then stopped to allow full transmission around the network whenever necessary. This is actually akin to biology, whereby synaptic transmission, dendritic signal integration and action

potential initialization can take time, but transmission speed is actually very fast [30]. In addition to low distortion, this approach also allows us to easily compare among computational models. We can simply extract a specific frame N of the simulations in all cases for detail comparison.

An alternative digital implementation to a NoC is perhaps a bus between processing cores. However, increases in firing frequency will lead to distortion of the information, which will limit the system performance. Alternatively some of these effects can be alleviated using traffic management via hierarchical AER architectures [31].

Using a NoC infrastructure as opposed to a bus also reduces power consumption within the design as it allows for much reduced clock frequency. Using Xilinx XPower Analyzer we estimate that when implemented upon a Virtex-7 VC707 XC7VX485T-2FFG1761C Evaluation Kit each module, containing a processor, router and interface, consumes 60 mW
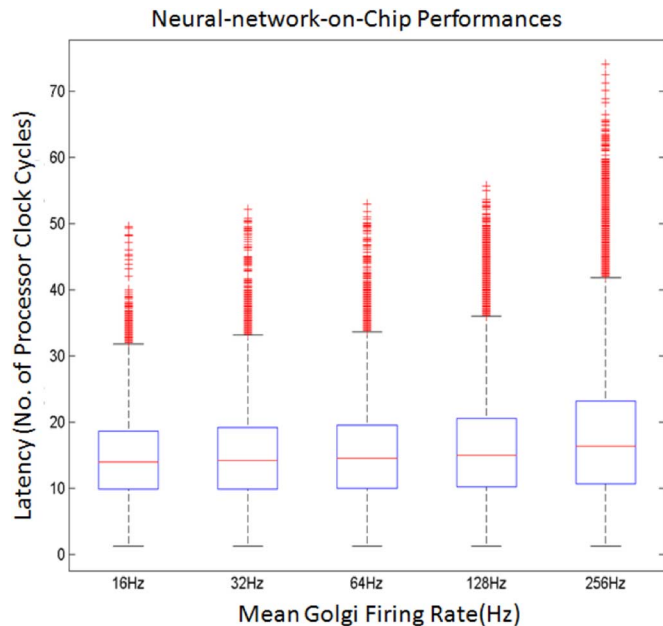
Fig. 10. The network on chip performance. It is the latency of packet transmission against the rate of packet injection into the network. Packet injection rate has only a minor impact upon latency within the expected range of operation (40–60 Hz). The data plotted in red shows box plots of the latency of packets through the network for varying degrees of background traffic.
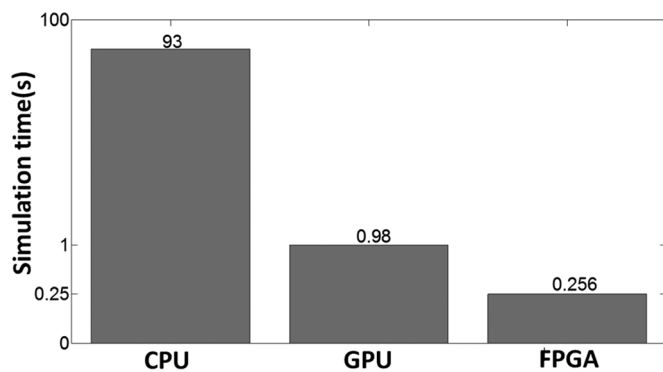

Fig. 11. The real-time computational condition among CPU, GPU and FPGA for simulating 1 s activities. The CPU and GPU results are cited from previous work [12].

TABLE II
FPGA-BASED GRANULAR LAYER SPECIFICATIONS

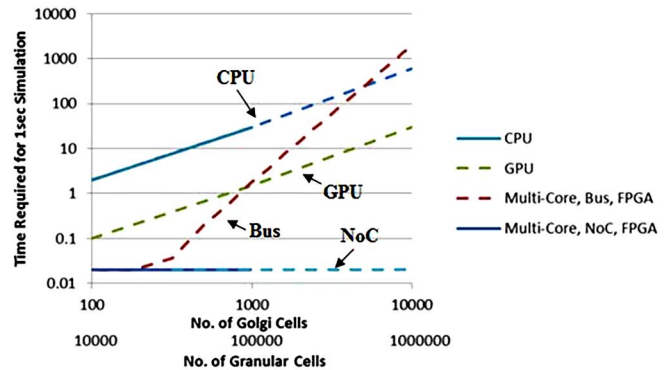| Timing issues				
Maximum clock frequency				121.945MHz
Minimum period				8.2ns
Hardware resources utilization				
	Processor	Router	Module	Total
Slice register	2884	792	3676	176424 (29%)
Slice LUTs	4379	1213	5592	268455 (88%)
Block RAM/FIFO	20	0	20	960 (93%)
DSP48E1s	48	0	48	2304 (82%)
Power Consumption				
Dynamic Power	-	-	60mW	2.88W

Fig. 12. Scalability of four different approaches. The dotted lines represent our estimation of system performances, whereas solid lines represent our measurements. The FPGA-based NoC computation time remains constant due to its parallel nature and the efficient communication system.

of dynamic power, equating to a total dynamic power consumption of 2.88 W when running at full-speed, or 60 mW per processing module.

B. Compare to Other Hardware Implementation Techniques

There are several possible alternative techniques to our frame based network-on-chip architecture. To date, SpiNNaker [32], Neurogrid [33], IBM SyNAPSE [34] are projects that build custom chips or systems for efficient large-scale simulation of general neural network models. These systems are powerful and innovative; however, they may not be optimal for the system that we are implementing in this paper. Based on the NoC analysis, it was found that with the unique network connectivity of the granular layer model that unicasting was more efficient in terms of memory resources by a factor of approximately $2\times$. The bandwidth for both approaches remained approximately equal. The results of this investigation alongside the reduced complexity of configuration within unicasting made it the preferred choice. Neurogrid employs a smart approach to combine analogue circuits for mimicking neural process and digital circuits for implementing routing components. It can potentially save significant amount of energy consumptions. But, analogue circuits based dimensionless models are not ideal to map conductance-based leaky integration-and-fire neuron in POT model. IFAT [35] is also a well-established platform for brain network real-time operation, but the analogue based integrated and fire array may not provide good scalability.

We are seeking to further optimize our system and to use it for other application. Cassidy *et al.* [36], [37] proposed a neuro-array architecture for general large-scale neuromorphic system with corresponding analysis. Their design principles, include external SRAM technique, can provide new insight for optimizing our system. Also, applying our proposed silicon granular layer to perform pattern recognition would be another application which is similar to new IBM chip TrueNorth [34].

In fact, our proposed frame based network on chip architecture is general for spiking neural networks, although in order to implement other models, we need to modify the components appropriately for the target model. For instance, in this work the routing components (transmitter, router and receiver) are customer designed for implementing POT recurrent random

network connections; and neural processor architecture is also specifically designed for mapping the connections from granule cells to Golgi cells. Further system tweaking will be required to optimize the performance for a different target model.

C. Neuro-Prosthesis Applications

For translation into neuroprosthesis, our architecture lends itself easily to electrical [38] or optical stimulation methodologies [39], [40]. The FPGA-based granular model can robustly predict responses of POT behavior and thus be used to interface with *in vivo* and *in vitro* experiment. Furthermore it is straight-forward to translate generated spikes directly to tissue as each will be encoded with a destination address.

For long-term neuro-prosthesis experiments this design can be translated directly to an ASIC platform in order to increase portability and to reduce power consumption. We estimate that by translating our design into CMOS, each module will consume 1.3 mW in high-speed operation and only 0.6 mW in real-time operation, giving a total power consumption of 28.8 mW to implement a neural network containing 100 000 elements. This compares favorably to power requirements in the brain whereby exceeding 100 mW can cause thermal damage [41].

VI. CONCLUSION

The goal of the work has been to implement a real-time cerebellar granular layer model onto a FPGA hardware platform utilizing a NoC hardware architecture. Our design can achieve (more than) real time operation for a system of 1000 Golgi cells and 100 000 granule cells on a single FPGA board. This is achieved via an efficient implementation of the mathematical models of the neuron cells; and the use of a frame based architecture which eliminates congestion distortion of spike timing in multiplexed networks. Our design is also highly scalable that computation time remains almost unchanged for a much larger network model.

The major contributions of this paper are summarized as follows: 1) an efficient FPGA-based NoC hardware architecture is proposed for implementing a large-scale cerebellar granular-Golgi layer model for POT encoding; 2) our implementation is computationally efficient that it can complete 1 sec simulation in 25.6 ms and that FPGA provides precise timing control. Together they allow our design to be readily adapted for real-time closed-loop *in vitro* or *in vivo* experiment; 3) our NoC architecture is highly scalable and hence it is now possible to simulate the full-scale granular layer with cell density of 1 million cells/mm^3 as in the real brain, which is 10 times the size of the current model. Such simulation power can open up new possibility for understanding the dynamics of the cerebellar network; and 4) our design can be potential neuro-prosthetics tool for future experimental and clinical applications owing to its high computational power, flexibility, high scalability and power efficiency.

REFERENCES

[1] R. B. Ivry and R. M. C. Spencer, "The neural representation of time," *Curr. Opin. Neurobiol.*, vol. 14, no. 2, pp. 225–232, Apr. 2004.
[2] J. D. Schmahmann, "Disorders of the cerebellum: Ataxia, dysmetria of thought, and the cerebellar cognitive affective syndrome," *J. Neuropsych. Clin. Neurosci.*, vol. 16, no. 3, pp. 367–378, Jan. 2004.
[3] K. M. Christian and R. F. Thompson, "Neural substrates of eyeblink conditioning: Acquisition and retention," *Learn. Mem.*, vol. 10, no. 6, pp. 427–455.
[4] M. D. Mauk and N. H. Donegan, "A model of Pavlovian eyelid conditioning based on the synaptic organization of the cerebellum," *Learn. Mem.*, vol. 4, no. 1, pp. 130–158.
[5] T. Yamazaki and S. Tanaka, "Computational models of timing mechanisms in the cerebellar granular layer," *Cerebellum*, vol. 8, no. 4, pp. 423–432, Dec. 2009.
[6] J. W. Moore, J. E. Desmond, and N. E. Berthier, "Adaptively timed conditioned responses and the cerebellum: A neural network approach," *Biol. Cybern.*, vol. 62, no. 1, pp. 17–28, 1989.
[7] J. E. Desmond and J. W. Moore, "Adaptive timing in neural networks: The conditioned response," *Biol. Cybern.*, vol. 415, pp. 405–415, 1988.
[8] D. Bullock, J. C. Fiala, and S. Grossberg, "A neural model of timed response learning in the cerebellum," *Neural Netw.*, vol. 7, no. 6–7, pp. 1101–1114, Jan. 1994.
[9] M. Fujita, "Adaptive filter model of the cerebellum," *Biol. Cybern.*, vol. 45, no. 3, pp. 195–206, 1982.
[10] T. Yamazaki and S. Tanaka, "Neural modeling of an internal clock," *Neural Comput.*, vol. 17, no. 5, pp. 1032–1058, May 2005.
[11] T. Yamazaki and S. Tanaka, "A spiking network model for passage-of-time representation in the cerebellum," *Eur. J. Neurosci.*, vol. 26, no. 8, pp. 2279–2292, Oct. 2007.
[12] T. Yamazaki and J. Igarashi, "Realtime cerebellum: A large-scale spiking network model of the cerebellum that runs in realtime using a graphics processing unit," *Neural Netw.*, vol. 47, pp. 103–111, Nov. 2013.
[13] M. Wang, B. Yan, J. Hu, and P. Li, "Simulation of large neuronal networks with biophysically accurate models on graphics processors," in *Proc. Int. Joint Conf. Neural Networks*, 2011, pp. 3184–3193.
[14] C. Hofstoetter, M. Gil, K. Eng, G. Indiveri, M. Mintz, J. Kramer, and P. F. Verschure, "The cerebellum chip: An analog VLSI implementation of a cerebellar model of classical conditioning," in *Proc. Advances in Neural Information Processing Systems*, 2004, pp. 577–584.
[15] S. A. Bamford, R. Hogri, A. Giovannucci, A. H. Taub, I. Herreros, P. F. M. J. Verschure, M. Mintz, and P. Del Giudice, "A VLSI field-programmable mixed-signal array to perform neural signal processing and neural modeling in a prosthetic system," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 20, no. 4, pp. 455–467, Jul. 2012.
[16] G. Coapes, T. Mak, J. W. Luo, A. Yakovlev, and C.-S. Poon, "A scalable FPGA-based design for field programmable large-scale ion channel simulations," in *Proc. 22nd Int. Conf. Field Programmable Logic and Applications*, 2012, pp. 112–119.
[17] T. S. T. Mak, G. Rachmuth, K.-P. Lam, and C.-S. Poon, "A component-based FPGA design framework for neuronal ion channel dynamics simulations," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 14, no. 4, pp. 410–418, Dec. 2006.
[18] E. L. Graas, E. A. Brown, and R. H. Lee, "An FPGA-based approach to high-speed simulation of conductance-based neuron models," *Neuroinformatics*, vol. 2, no. 4, pp. 417–436, Jan. 2004.
[19] R. K. Weinstein, M. S. Reid, and R. H. Lee, "Methodology and design flow for assisted neural-model implementations in FPGAs," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 15, no. 1, pp. 83–93, Mar. 2007.
[20] J. Luo, G. Coapes, T. Mak, T. Yamazaki, C. Tin, and P. Degenaar, "A scalable FPGA-based cerebellum for passage-of-time representation," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Aug. 2014, vol. 2014, pp. 3102–3105.
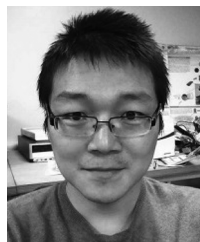[21] J. Luo, G. Coapes, P. Degenaar, T. Yamazaki, T. Mak, and C. Tin, "A real-time silicon cerebellum spiking neural model based on FPGA," in *Proc. Int. Symp. Integrated Circuits*, 2014, pp. 276–279.
[22] T. Bjerregaard and S. Mahadevan, "A survey of research and practices of network-on-chip," *ACM Comput. Surv.*, vol. 38, no. 1, p. 1-es, Jun. 2006.
[23] M. B. Josephs, S. M. Nowick, and C. H. Van Berkel, "Modeling and design of asynchronous circuits," *Proc. IEEE*, vol. 87, no. 2, pp. 234–242, 1999.
[24] Y. Kishimoto, S. Kawahara, Y. Kirino, H. Kadotani, Y. Nakamura, M. Ikeda, and T. Yoshioka, "Conditioned eyeblink response is impaired in mutant mice lacking NMDA receptor subunit NR2A," *Neuroreport*, vol. 8, no. 17, pp. 3717–3721, Dec. 1997.

[25] A. K. Fidjeland and M. P. Shanahan, "Accelerated simulation of spiking neural networks using GPUs," in *Proc. Int. Joint Conf. Neural Networks*, 2010, pp. 1–8.

[26] J. M. Nageswaran, N. Dutt, J. L. Krichmar, A. Nicolau, and A. V. Veidenbaum, "A configurable simulation environment for the efficient simulation of large-scale spiking neural networks on graphics processors," *Neural Netw.*, vol. 22, no. 5–6, pp. 791–800.

[27] R. Emery, A. Yakovlev, and G. Chester, "Connection-centric network for spiking neural networks," in *Proc. 3rd ACM/IEEE Int. Symp. Networks-on-Chip*, 2009, pp. 144–152.

[28] D. Vainbrand and R. Ginosar, "Scalable network-on-chip architecture for configurable neural networks," *Microprocess. Microsyst.*, vol. 35, no. 2, pp. 152–166, Mar. 2011.

[29] T. T. Mak, P. Sedcole, P. K. Cheung, and W. Luk, "On-FPGA communication architectures and design factors," in *Proc. Int. Conf. Field Programmable Logic and Applications*, 2006, pp. 1–8.

[30] B. L. Sabatini and W. G. Regehr, "Timing of synaptic transmission," *Annu. Rev. Physiol.*, vol. 61, pp. 521–542, Jan. 1999.

[31] S. Carrillo, J. Harkin, L. J. McDaid, S. Pande, S. Cawley, B. McGinley, and F. Morgan, "Hierarchical network-on-chip and traffic compression for spiking neural network implementations," in *Proc. IEEE/ACM 6th Int. Symp. Networks-on-Chip*, 2012, pp. 83–90.

[32] S. B. Furber, D. R. Lester, L. A. Plana, J. D. Garside, E. Painkras, S. Temple, and A. D. Brown, "Overview of the SpiNNaker system architecture," *IEEE Trans. Comput.*, vol. 62, no. 12, pp. 2454–2467, Dec. 2013.

[33] B. V. Benjamin, P. Gao, E. McQuinn, S. Choudhary, A. Chandrasekaran, J.-M. Bussat, R. Alvarez-Icaza, J. V. Arthur, P. Merolla, and K. Boahen, "Neurogrid: A mixed-analog-digital multichip system for large-scale neural simulations," *Proc. IEEE*, vol. 102, no. 5, pp. 699–716, May 2014.

[34] P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, A. S. Cassidy, J. Sawada, F. Akopyan, B. L. Jackson, N. Imam, C. Guo, Y. Nakamura, B. Brezzo, I. Vo, S. K. Esser, R. Appuswamy, B. Taba, A. Amir, M. D. Flickner, W. P. Risk, R. Manohar, and D. S. Modha, "A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science*, vol. 345, no. 6197, pp. 668–673, Aug. 2014.

[35] R. J. Vogelstein, U. Mallik, E. Culurciello, G. Cauwenberghs, and R. Etienne-Cummings, "A multichip neuromorphic system for spike-based visual information processing," *Neural Comput.*, vol. 19, no. 9, pp. 2281–300, Sep. 2007.

[36] A. Cassidy, A. G. Andreou, and J. Georgiou, "Design of a one million neuron single FPGA neuromorphic system for real-time multimodal scene analysis," in *Proc. 45th Annu. Conf. Information Sciences and Systems*, 2011, pp. 1–6.

[37] A. S. Cassidy, J. Georgiou, and A. G. Andreou, "Design of silicon brains in the nano-CMOS era: Spiking neurons, learning synapses and neural architecture optimization," *Neural Netw.*, vol. 45, pp. 4–26, Sep. 2013.

[38] Y. Ugawa, K. Genba-Shimizu, J. C. Rothwell, M. Iwata, and I. Kanazawa, "Suppression of motor cortical excitability by electrical stimulation over the cerebellum in ataxia," *Ann. Neurol.*, vol. 36, no. 1, pp. 90–96, Jul. 1994.

[39] B. McGovern, R. B. Palmini, N. Grossman, E. Drakakis, V. Poher, M. A. A. Neil, and P. Degenaar, "A new individually addressable micro-LED array for photogenetic neural stimulation," *IEEE Trans. Biomed. Circuits Syst.*, vol. 4, no. 6, pp. 469–476, Dec. 2010.

[40] X. Liu, S. Ramirez, P. T. Pang, C. B. Puryear, A. Govindarajan, K. Deisseroth, and S. Tonegawa, "Optogenetic stimulation of a hippocampal engram activates fear memory recall," *Nature*, vol. 484, no. 7394, pp. 381–385, Apr. 2012.

[41] S. Kim, P. Tathireddy, R. A. Normann, and F. Solzbacher, "Thermal impact of an active 3-D microelectrode array implanted in the brain," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 15, no. 4, pp. 493–501, Dec. 2007.

Graeme Coapes received the M.Eng. degree in electronic and computer engineering from Newcastle University, Newcastle upon Tyne, U.K., in 2011.

Currently, he is working toward the Ph.D. degree in microelectronics at Newcastle University. He is a Research Associate at Heriot-Watt University, Edinburgh, U.K., where he works with Siemens plc to develop novel algorithms, particularly neural networks, for condition monitoring of high voltage transmission and distribution systems.

Terrence Mak is an Associate Professor in Electronics and Computer Science, University of Southampton, Southampton, U.K.

Supported by the Royal Society, he was a Visiting Scientist at the Massachusetts Institute of Technology, Cambridge, MA, USA. Also, he is affiliated with the Chinese Academy of Sciences as a Visiting Professor. He has strong interests in bridging cutting-edge computing systems and emerging applications using novel architectures, algorithms and technologies. His work and contributions focus on VLSI, many core and FPGA architectures and systems design. He has authored work in over 30 journals and more than 60 conference proceedings. He received the prestigious Best Paper Award from three major conferences, DATE'11, VLSI-SoC'14, and PDP'15.

Tadashi Yamazaki received the B.E. degree in computer science from The University of Electro-Communications, Tokyo, Japan, in 1996, and the M.E. and Ph.D. degrees in computer science from the Tokyo Institute of Technology, Tokyo, Japan, in 1998 and 2002, respectively.

From 2002 to 2012, he was a Research Scientist with the RIKEN Brain Science Institute. Since 2012, he has been an Assistant Professor with the Graduate School of Informatics and Engineering, The University of Electro-Communications. His research interests include theoretical/computational modeling of the cerebellum, high-performance computer simulation of neural networks, and adaptive robotics.

Dr. Yamazaki received the Young Scientist Award and Best Paper Award from the Japanese Neural Network Society in 2006 and 2013, respectively. He is a member of The Japan Neuroscience Society, Japanese Neural Network Society, The Robotics Society of Japan, and Society for Neuroscience.

Junwen Luo received the Master's degree in automation and control and the Ph.D. degree in microsystem design from the School of Electrical and Electronic Engineering, Newcastle University, Newcastle upon Tyne, U.K., in 2010 and 2014, respectively.

Currently, he is a Research Associate at Newcastle University. His research areas mainly include FPGA-based neural modelling systems as well as bio-silicon neural interface and bio-inspired techniques.

Chung Tin received the B.Eng. degree in mechanical engineering from the University of Hong Kong, Pokfulam, Hong Kong, in 2002, and the M.Sc. and Ph.D. degrees in mechanical engineering from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2004 and 2011 respectively.

Since 2012, he has been an Assistant Professor in the Department of Mechanical and Biomedical Engineering, City University of Hong Kong, Kowloon Tong, Hong Kong. His research interests include computational neuroscience, sensorimotor learning, and neuroprosthetic system.

Dr. Tin was the recipient of the Croucher Foundation Scholarship (Hong Kong), American Heart Association Predoctoral Fellowship, and Early Career Award (Research Grant Council, Hong Kong).

Patrick Degenaar received the Bachelor's (1st class) and M.Res. degrees in applied physics from Liverpool University, Merseyside, U.K., and the Ph.D. degree in bioelectronics from the Japan Advanced Institute of Science and Technology, Nomi, Japan.

Currently, he is a Reader in biomedical engineering at The School of Electrical and Electronic Engineering, Newcastle University, Newcastle upon Tyne, U.K. Until 2010, he held a senior lectureship at Imperial College London, London, U.K., where he also held a RCUK fellowship. His core interests lie in neuroprosthetics and bringing devices to clinical practice. He previously led the FP7 OptoNeuro consortium and is now the engineering lead on the CANDO project (http://www.cando.ac.uk) to bring a next generation optogenetic/optoelectronic implants for epilepsy to clinical practice.