Wafer-Scale Integration of Analog Neural Networks

Johannes Schemmel, Johannes Fieres and Karlheinz Meier

***Abstract*— This paper introduces a novel design of an artificial neural network tailored for wafer-scale integration. The presented VLSI implementation includes continuous-time analog neurons with up to 16k inputs. A novel interconnection and routing scheme allows the mapping of a multitude of network models derived from biology on the VLSI neural network while maintaining a high resource usage. A single 20 cm wafer contains about 60 million synapses. The implemented neurons are highly accelerated compared to biological real time. The power consumption of the dense interconnection network providing the necessary communication bandwidth is a critical aspect of the system integration. A novel asynchronous low-voltage signaling scheme is presented that makes the wafer-scale approach feasible by limiting the total power consumption while simultaneously providing a flexible, programmable network topology.**

I. Introduction

A key aspect of neuroscience is the modeling of neural systems. Modeling is used to test all kinds of hypotheses derived from the observation of biological nervous tissue in-vivo and in-vitro. One modeling possibility is the numerical integration of a mathematical description consisting of a set of differential equations [1], e.g. performing a computer simulation. This method is mainly limited by the available computing power. As a consequence, most simulations are not performed in biological real-time but up to several orders of magnitude slower. If one is going to model processes like learning or development, which already take from minutes to days in the laboratory, this may seriously affect the modeling possibilities.

In addition, the statistical nature of the neural code often requires several repetitions of a simulation, as do parameter searches. In this paper, a novel concept for the modeling of large networks is presented. Based on a continuous-time analog model realized in VLSI technology it allows the modeling of neural systems with up to several billions of synapses while maintaining an average acceleration factor of 10^4. This speed-up allows to do extensive parameter searches. Most experiments will last only a few milliseconds when done on the presented hardware system. In contrast to most other VLSI implementations (see [2] for a recent review) the presented neural network is targeted at large-scale network models which are currently limited by the available computing resources [3][4].

Compared to previous implementations of highly-accelerated analog neural networks [5][6][7] the presented system increases the maximum network size by three orders of magnitude. The maximum number of pre-synaptic signals a neuron can receive changed from 256 to 16k. Just increasing the numbers of synapses and neurons would not be sufficient for modeling real neural systems. It is also necessary to support the diverse neuron properties found in biology [8] as well as the complex interconnection topologies formed by their axons and dendrites.

The *FACETS* system described in this paper is designed to meet these requirements. It is part of the European research project *FACETS*[1][9]. To cover the technical and theoretical aspects of the wafer-scale neural network development done within this project two papers were submitted: the present paper describes the hardware requirements and implementation while [10] gives an introduction to the algorithms for routing and resource allocation developed for the *FACETS* wafer-scale system and presents experimental results confirming the viability of this approach.

The remainder of this paper is organized as follows: Section II gives an introduction of the *FACETS* system, while section III describes the VLSI implementation. The neuron-to-neuron communication is the topic of section IV.

II. Overview of the FACETS Hardware

Basically, the *FACETS* hardware model consists of a large number of ASICs containing the analog neuron and synapse circuits. Due to the high acceleration factor the necessary communication bandwidth in-between these *Analog Network Chips* (ANC) can exceed 10^{11} neural events per second[2]. The textbook solution would have been the utilization of a very high I/O count which is only feasible in Flip-Chip technology [11] and leads to complicated and expensive printed-circuit boards and high packaging costs. Therefore, a different approach is used in *FACETS*: wafer-scale integration. In this technology the silicon wafers containing the individual chips are not cut into dies, instead, the chips are interconnected directly on the wafer. This method provides the necessary connection density. Two prerequisites need to be fulfilled to make wafer-scale integration feasible: fault tolerance and low power consumption.

A biological neural network is inherently fault tolerant against random neuron loss [12]. There are two main reasons for this: first, all tasks are performed by large populations of neurons and second, the high plasticity allows healthy neurons to take over the functionality of deceased ones. Both mechanisms are also present in the presented analog

Johannes Schemmel, Johannes Fieres and Karlheinz Meier are with the Kirchhoff Institute for Physics, University of Heidelberg, Im Neuenheimer Feld 227, 69120 Heidelberg, Germany (email: schemmel@kip.uni-heidelberg.de)

This work is supported in part by the European Union under the grant no. IST-2005-15879 (FACETS).

[1] The acronym stands for: *"Fast Analog Computing with Emergent Transient States"*.

[2] A neural event encodes the transmission of an action potential from one source neuron to a set of target neurons.

978-1-4244-1821-3/08/$25.00©2008 IEEE

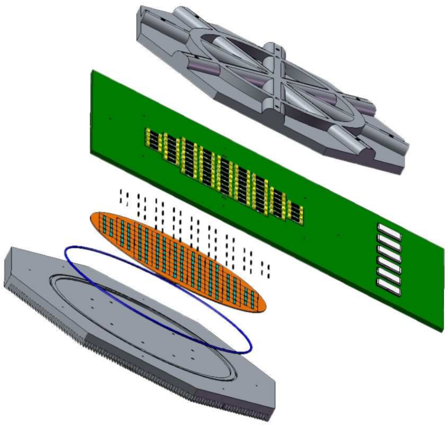

Fig. 1. A *FACETS*-wafer with motherboard and mounting bracket. From top to bottom: top mounting bracket, motherboard containing digital network chips, FPGAs and passive components, elastomeric connectors, silicon wafer with analog network chips, seal ring, bottom mounting bracket.

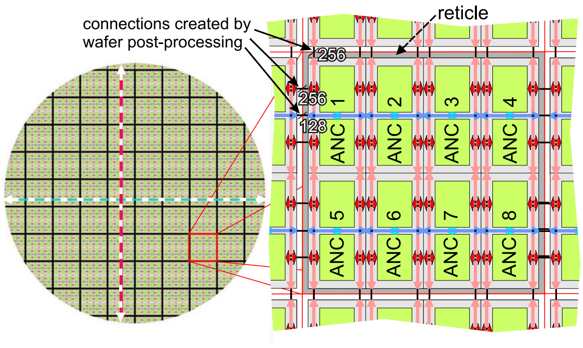

Fig. 2. A *FACETS*-Wafer containing 56 complete reticles. The dashed arrows depict one bundle of horizontal respectively vertical inter-neuron connections. A single reticle is enlarged showing the arrangement of the analog neural network chips (ANC). The number of wires is given for each type of connection created by wafer post-processing.

VLSI neural network chips. Several levels of programmable topology allow replacing defect neurons as well as whole defect ANCs by fully functional ones. This will always provide a completely operational network to the experimenter, similar to the disabling of defect memory blocks in a CPU's cache memory [13]. In addition, the disabling of single defect synapses and neurons will not be necessary for the majority of experiments, because they will also use population coding and therefore won't rely on every single synapse or neuron.

The power consumption is the major issue in realizing wafer-scale integration for the presented analog neural network. The number of events per second is proportional to the acceleration[3] factor of up to 10^5 and the event transmission involves the charging and discharging of the wire capacitances. To limit the power consumption of the event transmission a novel asynchronous differential low-voltage signaling scheme was developed. Also the static power consumption of all circuits is minimized. Especially the synapse, which is by far the most frequent circuit in a neural network, uses no static power at all. Combining these methods the average power consumption is expected to stay below 1 kW for a single wafer. An electrical power dissipation of this magnitude uniformly distributed across the surface area of a silicon wafer with 20 cm diameter equals a power-density of $1.6W/cm^2$. This is well below the limit of standard air cooling methods and allows to densely mount the wafer systems in industry standard racks.

Fig. 1 shows an exploded view of a wafer and its accompanying motherboard. Several of these single wafer units fit vertically in a industry standard 9U crate[4]. A custom backplane connects the wafer units to each other. The motherboard contains *Digital Network Asics* (DNC) that interface the ANC chips on the wafer with several FPGA[5] on the backplane interconnecting the wafer boards in the crate. These FPGA chips implement the necessary communication protocols to exchange neural events between the different network wafers and the host computer[6].

In Fig. 2 the partitioning of the wafer in 56 reticles[7], containing eight ANCs each, is visible. To achieve the necessary inter-neuron connection density a dense layer of horizontal and vertical wires is used. These connections can not be routed from one reticle to another due to the limitations of the manufacturing process. Therefore, a post-processing step is used that deposits and structures an additional layer of metal atop the original wafer[8]. The post-processing connects the inter-neuron connections across the gaps between the reticles. It also provides a solution for the connection between motherboard and wafer. In the center of the reticle multiple pad rows are placed. They make contact to their counterparts on the motherboards by industry standard elastomeric connectors [14] (see Fig. 1). The post-processing is used to form these pads as well as their connections to the original pad-windows of the analog neural network chips on the reticles. These pad-windows can be of minimum size and placed arbitrarily which simplifies the chip's layout.

III. The HICANN Chip

The initial version of the ANC is called HICANN (High Input Count Analog Neural Network). It is the primary build-

[3] The acceleration factor is selectable between 10^3 to 10^5. Due to the bandwidth limitations of the wafer-to-wafer communication, an acceleration factor above 10^4 is restricted to experiments using predominantly intra-wafer communication.

[4] 'U' stands for 'Rack Unit' and is used to measure the height of a device mounted in a 19-inch rack. One rack unit is 1.75-inch (44.45 mm).

[5] Field Programmable Logic Arrays

[6] The DNC chips and the FPGA code are developed by the chair of Prof. Schüffny at the Technical University of Dresden, Germany.

[7] Due to the high resolution required for the mask it is not possible to expose a whole wafer at once. Therefore a certain area, called the reticle, is repeatedly exposed onto the wafer. The reticle has fixed maximum size of usually 20 to 25 mm which corresponds to the maximum feasible chip size.

[8] Since this is done with a much lower feature size (about 5μm) than the VLSI manufacturing (180 nm) a wafer-scale mask can be used.

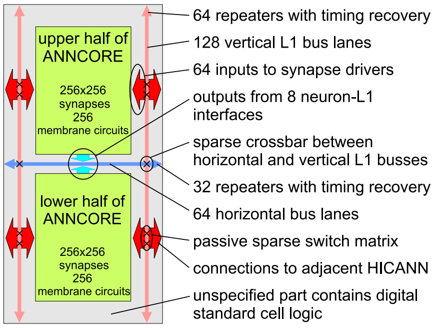
upper half of ANNCORE
256x256 synapses
256 membrane circuits
lower half of ANNCORE
256x256 synapses
256 membrane circuits
64 repeaters with timing recovery
128 vertical L1 bus lanes
64 inputs to synapse drivers
outputs from 8 neuron-L1 interfaces
sparse crossbar between horizontal and vertical L1 busses
32 repeaters with timing recovery
64 horizontal bus lanes
passive sparse switch matrix
connections to adjacent HICANN
unspecified part contains digital standard cell logic

Fig. 3. Block diagram of a HICANN chip.

ing block for the FACETS wafer-scale system. It contains the mixed-signal neuron and synapse circuits as well as the necessary support circuits and the host interface logic. Fig. 2 shows the arrangement of the HICANN chips on the wafer. Eight HICANN chips will be integrated on a single reticle. The size of the HICANN chip is chosen to be 5x10 mm^2. This allows to fully qualify the HICANN in silicon using MPW[9] prototyping only, thus limiting cost. The necessary pitch of the post-processing metal layer that connects the individual reticles directly on the wafer is about 10 μm, allowing for a connection density of 100 wires/mm between adjacent edges of neighboring reticles. This is sufficient for the maximum connection density which occurs at the short edges of the HICANN die, where 512 wires are used to interconnect 256 differential bus lanes (see section IV).

In Fig. 3 the main functional blocks can be identified. The largest one is the *Analog Neural Network Core* (ANNCORE) containing 128k synapses and 512 membrane circuits which can be grouped together to form neurons with up to 16k synapses. The interconnections between the HICANN chips run vertically and horizontally through the chip, with crossbar switches at their intersections. Additional switch blocks give the synapses inside of the ANNCORE access to these signals.

Eight HICANN dies are combined to form the reticle of the wafer-scale system. Fig. 2 shows the connections between adjacent reticles which have to be created by post-processing the wafer. The reticle is larger than the area occupied by eight HICANN dies (grey border) to accommodate the contact pad windows for the post-processing. Inside the reticle the inter-neuron signals of the HICANN dies are edge connected by the topmost metal layer. To achieve the fault tolerance necessary for wafer scale integration each HICANN has individual power supplies as well as an individual connection to the *Digital Network Chip* on the printed circuit board. These connections are also realized by post-processing which rearranges the pads of the eight HICANN dies into regular spaced contact rows inside the reticle suitable for the elastomeric connectors.

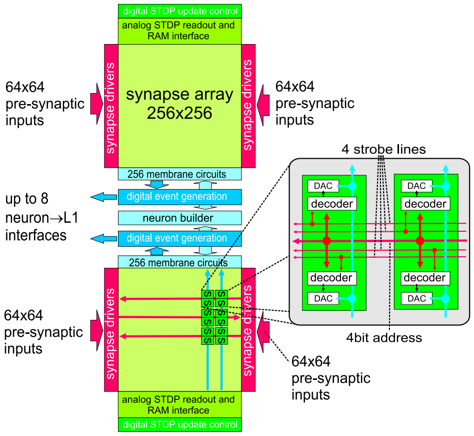
digital STDP update control
analog STDP readout and RAM interface
64x64 pre-synaptic inputs
synapse drivers
synapse array 256x256
synapse drivers
64x64 pre-synaptic inputs
256 membrane circuits
up to 8 neuron→L1 interfaces
digital event generation
neuron builder
digital event generation
256 membrane circuits
64x64 pre-synaptic inputs
synapse drivers
S
synapse drivers
64x64 pre-synaptic inputs
analog STDP readout and RAM interface
digital STDP update control
4 strobe lines
DAC
decoder
DAC
decoder
decoder
DAC
decoder
DAC
4bit address

Fig. 4. Block diagram of the analog network core. The enlarged portion shows the connection pattern between synapses and strobe resp. address lines.

A. ANNCORE circuits

Fig. 4 shows the main elements of the ANNCORE. Its geometry is designed for a maximum input count of 16k pre-synaptic inputs per neuron. In this case an ANNCORE block will implement eight neurons. In contrast, using the maximum neuron number of 512 limits the number of inputs per neuron to 256. The high number of different input signals required for a neuron with 16k synapses leads to an excessive bandwidth demand: considering the case of a mean firing rate of 10 Hz, an acceleration factor of 10^5 and 16k inputs this equals to an average event rate of 164 Giga-events/s, easily crossing the Tera-event/s barrier in periods of bursty neural activity. Using traditional digital coding techniques an event packet would use about 16 to 32 bit, containing target address and delivery time. To make this communication demand feasible the ANNCORE uses a combination of space and time multiplexing. Due to the high density of the connections between the reticles and the on-die wiring between the HICANN chips inside the reticle a large number of signals can be multiplexed spatially. The presented implementation uses 256 bus lanes, consisting of two wires each, running alongside the synapse drivers (see Fig. 3).

Temporal multiplexing is used as the final step to reach the necessary numbers. Each wire pair carries the events from 64 pre-synaptic neurons[10] by serially transmitting 6 bit neuron numbers. For historic reasons, this protocol is called the *Layer 1* or short L1 routing[11]. A widely used protocol for temporal multiplexing in VLSI neural networks is the *Address Event Representation* (AER) [15][16]. It is based on a request-grant scheme to allocate the shared resource. In the presented system this is not possible for spatially separated neurons since the additional power consumption of

[9] Multi Project Wafer

[10] This could be extended to 256 neurons in future implementations.

[11] This is opposed to the non-multiplexed local connections used in previous systems in our lab which are called *Layer 0*. *Layer 2* is the discrete-time event based inter-chip communication layer used between ANC and DNC as well as between DNC and FPGA.

the request and grant signals would be prohibitive. Therefore, temporal multiplexing always combines the signals from a local group of neurons located in a direct neighborhood within a single ANNCORE. A maximum of 64 adjacent neurons share a priority encoder which is used to allocate the time slots on the serial L1 bus.

The complexity at the sender as well as the receiver side of the serial L1 link is further reduced by transmitting the event in continuous time, i.e. the time of a pre-synaptic event is determined by the moment of its arrival at the synapse driver. The drawback here is the potential timing error introduced in the case of heavy simultaneous firing[12]. The average probability of such a collision happening is determined by the duration of the transmission of an event, the acceleration factor, the number of neurons sharing a wire and the joint firing probability of these neurons. The user can always adjust the first three parameters in a way to accommodate his requirements.

1) Synapse drivers: The synapse drivers are the interface between the serialized event data and the synapse array (see Fig. 4). Every two synapse rows share a L1 receiver and synapse driver. Since they are alternately mounted left and right from the synapse array, there is one L1 input every four rows, totaling in 64 inputs per side and block. They contain the deserializer and data capture circuits which will be described in detail in section IV-B.4. The received 6 bit pre-synaptic neuron address is split in two parts. The upper two bits are compared to stored address patterns for a set of strobe lines for the synapse address decoders. The length of these strobe pulses, τ_{STDF}, can encode for the momentary value of the synaptic transconductance if it is modulated by short term plasticity mechanisms (short term depression or facilitation: STDF) [7]. The lower four bit of the sampled 6 bit neuron address are subsequently transmitted into the synapse array. Each synapse driver can address all 256 membrane circuits in its ANNCORE half. As illustrated in Fig. 4 there are always two synapses connected to the same membrane circuit sharing a synapse driver, i.e. a group of 64 possible pre-synaptic neurons.

2) Synapses: The synapse circuits are an enhanced version of the ones reported in [6]: the synaptic weight is stored in a 4 bit static RAM cell. A 4 bit digital-to-analog converter (DAC) translates the stored weight into an output current. The major change is the inclusion of a four bit address decoder replacing the single pre-synaptic signal used previously. Each synapse has a fixed connection to one of the strobe signals from the synapse driver and a programmable four bit address. This allows for a much higher mapping efficiency in the case of sparsely connected random networks [10]. The fixed maximum conductance g_{max} which determines the scale for the synapse DAC can still be set row-wise by a programmable analog parameter. The output signal of a synapse in the case of an input event matching its address is a square current pulse with the amplitude $weight \times g_{max}$ and the length τ_{STDF} controlled by the synapse driver.

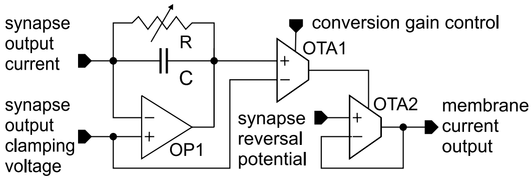

Fig. 5. Operating principle of the circuit modeling the synaptic ion channels.

The correlation measurement circuits in each synapse used to implement spike-time dependent plasticity (STDP) are similar to [6]. Since plasticity is not a topic of this paper they will not be discussed further.

3) Membrane Circuits and Neuron Builder: A neuron is formed by connecting together an arbitrary number of dendrite membrane circuits, called *denmem*. Each *denmem* contains a set of ion-channel emulation circuits connected to the membrane capacitance. These ion-channel circuits represent the following membrane currents: excitatory synapses, inhibitory synapses, leakage, adaptation and spike generation, the latter including the reset current. These circuits allow the implementation of the *adaptive exponential integrate and fire* model [17] as well as a simple conductance-based integrate and fire model implemented in [6], thereby maintaining compatibility with experiments designed for that system. All analog parameters are stored in non-volatile single-poly floating-gate analog storage cells developed for the *FACETS* project [18].

The synapses are connected to the *denmem* circuits by two lines running orthogonal to the pre-synaptic inputs. Each synapse has an output multiplexer selecting which line to use. The current pulses from all synapses connected to a *denmem* circuit add up to two time-varying total input currents. The *denmem* circuit converts these currents into time-varying conductances emulating two different groups of synaptic ion channels. One can be programmed as excitatory and the other as inhibitory, for example. In larger neurons incorporating multiple *denmem* circuits the types of ion channels are not limited to two. It is possible to use two different types of ion channels in each *denmem* circuit.

Fig. 5 shows the operating principle of the synapse ion channel circuit. Operational amplifier OP1, together with capacitor C and tunable resistor R, forms a leaky integrator for the synapse output current. The non-inverting input of OP1 is held at a fixed reference potential, which is set to the optimal output voltage for the synapse current sources. The amplifier clamps the synapse current output to this voltage through the feedback capacitor C. This clamping action of the integrator has two benefits: first, it enhances the precision of the synapse current sources since they always see a constant voltage. Second, it speeds up the network operation by reducing the input voltage swing at OP1 to a minimum.

Operational transconductance amplifier OTA1 converts the

[12] In a packet-based solution there would be FIFO-memories available to equalize the event rate.

output voltage of OP1 which corresponds to the integrated synapse current to a proportional output current. The conversion gain can be set by its bias current. The output current from OTA1 is directly used as the bias current for OTA2, which implements the ion channel conductance. OTA2 translates the voltage difference between the membrane voltage and the reversal potential to a proportional current, thereby emulating a conductance. The resistor R parallel to the integrating capacitor C implements the exponential decay of the synaptic conductance by continuously discharging C. It can be tuned to adjust the time constant of this decay to the type of ion channel emulated.

The neuron builder (see Fig. 4) is a switch matrix connecting groups of *denmem* circuits together. The spike generation circuit is disabled in all connected *denmem* circuits but one. The connections made in-between the *denmem* circuits by the neuron builder are two-fold: first, the membrane potential is shorted and second, the back-propagating action potential generated by the single active spike generation circuit is distributed from the *denmem* propagating the spike to all synapses belonging to the neuron. This is necessary for the STDP correlation measurements performed by the synapses [6].

Eight asynchronous priority encoders with 64 inputs each determine which action potential is transmitted back into the network. There are two kinds of signals generated from these neural events: an asynchronous signal compatible to the L1 inter-neuron protocol and a synchronous version which is transferred to the digital control of the HICANN and distributed further by the packet-based network implemented by the DNC and FPGA on the motherboard.

IV. Neuron-to-Neuron Communication

A. Continuous-Time Event Transmission Protocol

The length of a wire traversing a HICANN die is about 10 mm. Depending on separating distances this wire will see a total capacitance to its surrounding of about 2 pF[13]. Considering a simple square pulse as code for an event the power consumption p_{wire} can be calculated as follows:

$$p_{\text{wire}} = C \cdot V^2 \cdot Events/s \; [\text{W}] \qquad (1)$$

If this wire is driven with the full CMOS swing of 1.8 Volts using an acceleration factor of 10^5 and a mean firing rate of 10 Hz, p_{wire} equals to 6.5 μW. If one scales this up to a whole wafer containing 230k Neurons on about 450 HICANN chips the total power is 1.7 kW for the transmission of the neural event signals alone[14].

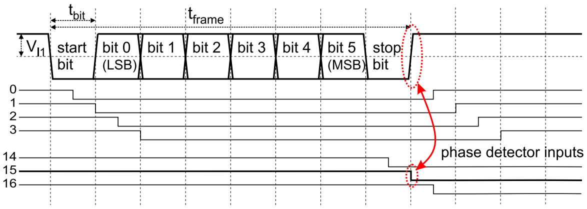

Fig. 6. Timing of a serial L1 data frame. Shown is the differential signal $V_{\text{l1\_pos}} - V_{\text{l1\_neg}}$. The output signals of the delay-locked loop (DLL) used for timing recovery are shown beneath. If the falling-edge of output 15 is aligned with the rising edge of the L1 signal, the DLL is locked.

A serial event coding using a single low-voltage differential signal to transmit the events from up to 64 pre-synaptic neurons was introduced to limit the power consumption. An event is encoded using two frame bits and 6 data bits. Fig. 6 shows a single serial L1 data frame. The timing parameters for the typical process corner are: t_{frame}=4 ns, t_{bit}=500 ps and the differential DC amplitude V_{l1}=150 mV. The average number of transitions per event is 5.5, a rounded number of 6 will be used in further discussions. This reduces the total power consumption to 5.6 watts (in the case of a differential voltage swing of 100 mV). This is a 300-fold reduction compared to the parallel CMOS case.

The resistance of said exemplary wire is $36m\Omega/\Box \times 20k\Box = 720\Omega$ and the time constant therefore $\tau = RC/2 = 0.7ns$ [15]. To reach a bit rate of 2 GBit/s a certain amount of overdrive is needed. The overall geometry of the L1 busses in the HICANN chip shows that the effective length is more than 10 mm. If repeaters are placed along the edges of the chip the worst case for an unbuffered L1 line is 5 mm vertical up to the central crossbar, 5 mm horizontal and 5 additional mm vertical after the crossbar plus two times about 3 to 4 mm input lines to the ANNCORE (see Fig. 3 for reference), branching off the vertical segments. To reduce the total RC-time constant of such a network to a value that can sustain 2 GBit/s the metal width must be increased from the previous example. Simulations have shown that a metal width and spacing of 1.2 μm using the 2.2 μm thick top metal layer gives satisfactory results for all process corners and worst case routing scenarios. Only one additional provision has to be made: the total parasitic capacitance of de-selected switches in the central crossbars as well as the synapse driver switch matrices must be limited. Therefore, these structures are only sparsely populated with switch transistors. See the accompanying paper [10] about the routing algorithms for further details of the switch arrangement.

B. Serial L1 Components

1) Neuron→L1 Interface: Fig. 7 shows an exemplary connection from the neuron labeled N1 to the neuron N2 located on a different HICANN die. The signal starts at the output of the neuron builder (see section III-A.3) as a standard digital CMOS signal, encoding the time of the neuron's firing by its rising edge. An asynchronous priority encoder with 64 inputs determines which neuron's action potential will be transmitted back into the network. The

[13] The considered metal lines had the following parameters: 500 nm width, 800 nm spacing, metal 6, metal 5 orthogonal and only sparsely used, full coupling to metal 4. The dominating capacitance is the coupling capacitance within the layer which accounts for more than 90% of the total capacitance.

[14] In this calculation an event bus uses 6 address bits and 1 strobe bit, the address bits toggle with half the frequency of the strobe signal.

[15] Using a simple model which distributes the total wire capacitance equally at both ends of the wire.

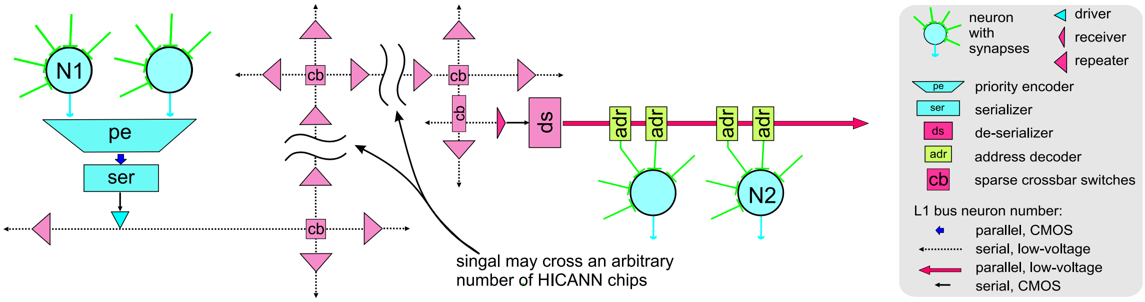

Fig. 7. Schematic diagram of a neuron-to-neuron connection crossing several HICANN chips.

neuron with the highest priority of all simultaneously firing neurons connected to the same priority encoder is selected for transmission. Each neuron has a programmable 6 bit identification number assigned to it which will be sent to the serializer in this case. The serializer generates the bit stream from the parallel neuron number.

The serial CMOS-level data stream is sent through a driver which converts it to the L1 voltage levels. It uses strong pre-emphasis to overcome the large RC time constant. To conserve energy both differential lines are shorted to equalize their potential before the new differential voltage is applied. If the data stream is constant for more than one bit period is is connected to a differential voltage of 100 to 150 mV and a common mode voltage of about 750 mV. The common mode voltage can be adjusted by the external L1 power supply to compensate any PMOS/NMOS imbalance introduced by process variations. This assures that the effective common mode applied by the pre-emphasis driver is the same than the common mode in the DC case.

Each neuron→L1 interface is connected to a fixed horizontal L1 lane using the following scheme: i(nterface) 1→l(ane) 1, i2→l33, i3→l17, i4→l49, i5→l9, i6→l25, i7→l41 and i8→l57. At each HICANN boundary all vertical and horizontal L1 busses are rotated by one bus lane, i.e. for the horizontal case this results in the following mapping: 1→2,2→3,. . . ,64→1. This scheme has the advantage that all bus lanes are equally used without switching the L1 driver outputs between different L1 bus lanes. Due to the low output impedance necessary for the pre-emphasis signal inserting switch transistors in the differential output path would lead to a large increase in the power consumption.

2) Repeaters: The signal now travels on the horizontal and vertical L1 busses from chip to chip, using special repeaters for signal and timing restoration at the chip boundaries. Their components are shown in Fig. 8: receiver, timing restoration circuit and driver. A repeater is bi-directional. Its data flow direction can be set according to the routing requirements. It can also be switched off to isolate the bus lane it is connected to.

The receiver consists of a differential amplifier restoring CMOS levels from the serial L1 signal. Since this receiver is the only circuit consuming a significant amount of static bias current without any L1 activity it is optimized for a minimum power consumption with a positive input ($V_{\mathrm{l1\_pos}} > V_{\mathrm{l1\_neg}}$), which is the inactive line level of the L1 bus. Simulations show that in this case its current consumption stays below 100 μA at a speed still sufficient for an operation with 2 Gbit/s. This is a crucial detail of the L1 implementation since the number of receivers on a wafer is about 260k.

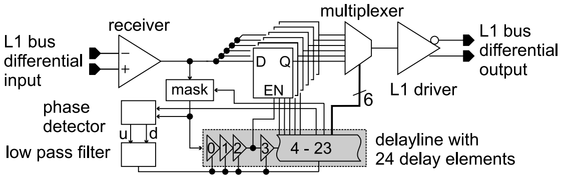

Fig. 8. Schematic diagram of a repeater. A 24-tap DLL locks on the input frame and acts as timing reference for the multiplexer which generates the serial output stream.

The timing restoration consists of a de-serializer and a serializer, both controlled by a single delay-locked loop with 24 delay elements (DLL). The single-ended CMOS L1 signal is used as an input to six dynamic data capture latches and the DLL. The DLL captures the frame timing by aligning the delayed falling edge of the start bit with the original rising edge of the stop bit, thereby dividing the frame in 16 time bins (see Fig. 5).

The training phase of the receiver DLL is divided in two phases. In the startup-phase only L1 frames with neuron address zero are transmitted, allowing all DLLs an initial lock. During normal network operation, a special input circuit masks all transitions of the reference signal that lie outside of the expectation window around the rising edge of the stop bit. The timing information for this mask signal is derived from the DLL itself. Therefore, in the locked case, the DLL can compensate small timing variations caused by temperature drift or leakage from the control voltage storage capacitor without being disturbed by the additional transitions in the signal caused by the random data payload of the frame.

For each data bit exists a time bin which lies exactly in the middle of its data eye and which is used to trigger the capture latch. The serialization is done by a multiplexer. The DLL outputs controlling this multiplexer are selected in a way that each bit is sent in the time-bin directly following its capture.

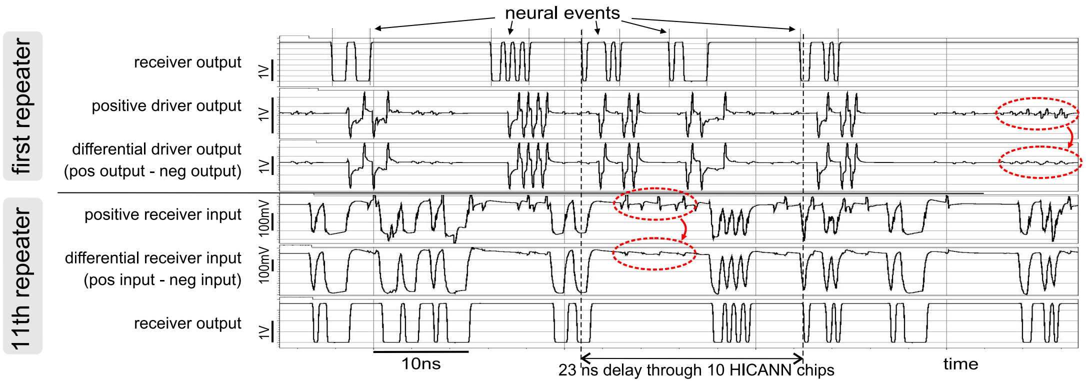

Fig. 9. Simulation results for an L1 bus lane running through a chain of 10 HICANN chips with repeaters at the chip boundaries. See Fig. 10 for an illustration of the setup used. The dashed circles indicate some examples of crosstalk. Note the strong crosstalk supression in the differential mode.

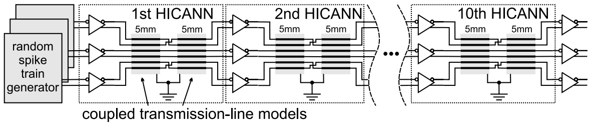
1st HICANN
2nd HICANN
10th HICANN
5mm
5mm
random spike train generator
coupled transmission-line models

Fig. 10. Configuration used for the simulation shown in Fig. 9.

This keeps the delay through the repeater as short as possible. Simulations have shown that the total delay is 2.3 ns for the typical process corner. The differential output driver for the L1 bus is identical to the driver of the sender described previously. Fig. 9 shows some results from an exemplary simulation. Three L1 bus lanes are modeled in 10 connected HICANN chips. The setup is depicted in Fig. 10. The center lane is the one to be examined, the outer lanes model the crosstalk between neighboring L1 busses. At the beginning of each lane a repeater is placed. To examine the signal at the end of the chain an 11th repeater is added there. Fig. 9 shows, from top to bottom, the following signals: the serial CMOS level signal at the output of the differential receiver of the first HICANN chip in the chain, the positive as well as the differential signal at the output of the first L1 driver, the positive as well as the differential input of the 11th repeater and the CMOS level output of its receiver.

At the input of the first HICANN chip in the test chain three random spike trains are fed into the three repeaters. The occurrence of the neural events in each spike train is Poisson distributed with randomly assigned neuron numbers. The mean inter-spike interval is 15 ns. This is a worst case scenario resembling a situation where the total 64 neurons sharing an L1 bus are firing at a mean rate of 100 Hz each (using an acceleration factor of 10^4).

The L1 bus lanes are modeled by a Gaussian quadrature model including the capacitive and inductive coupling between all six individual wires and a ground plane below. The wires have a width of 1μ and a thickness of 2.2μm. The spacing between adjacent wires is 1.2μm and the ground plane distance is 2.2μm. The capacitive coupling between neighboring wires leads to strong crosstalk. Since this coupling is strongest between direct neighbors it introduces a voltage difference between the lanes of a differential pair. This will lead to false transitions seen by the receiver. A reduction of this crosstalk signals is achieved by twisting every second L1 bus in the middle of the HICANN chip, i.e. the positive and negative wires are swapped. This is illustrated in Fig. 10. By this method the positive wire of one bus lane runs in equal lengths in parallel to the positive and the negative wire of the neighbor. This cancels most of the crosstalk.

The results show that despite the signal distortions caused by crosstalk each repeater can restore the original signal. After traversing a distance of 10 cm and 10 HICANN chips the signal has accumulated 23 ns delay but is otherwise unchanged. Due to the high capacitive loading of the 10 mm long bus wires the strong pre-emphasis at the driver is necessary to ensure the build-up of a sufficient input level at the receiver within the bit-period of 500 ps.

3) Crossbar Switches and Synapse Driver Switch Matrices: At each intersection of a horizontal and a vertical L1 segment a crossbar switch (see Fig 3) is located which allows connections between the horizontal and vertical L1 bus lanes. The vertical lanes run in parallel to the synapse driver columns located at both sides of the ANNCORE. A sparse switch matrix allows the coupling of any L1 lane to a synapse driver. To control the capacitive loading of the vertical L1 busses only a certain number of switches and activated connections to the synapse drivers is allowed. To share these signals between adjacent rows, neighboring synapse drivers have a bypass switch between their inputs. This allows forming vertical chains of drivers sharing one common connection to the vertical L1 busses.

The routing algorithm allocates certain neuron groups to vertical L1 buses and needs to selectively connect synapse drivers with L1 lines as well as horizontal and vertical L1 buses. The best solution from a routing point of view would be a fully connected crossbar at these locations. The signal levels of the L1 busses allow the use of NMOS-only switches at the cross-over points. Space considerations permit a fully connected crossbar but the capacitance of the de-selected transistors increases the RC-time constant too much. Therefore a sparse matrix is used for all cross-over points. The exact number of switches is a result of routing simulations and documented in [10]. Electrically about 32 out of a maximum of 256 switches per horizontal line are feasible for the main crossbar. Similar ratios apply for the synapse drivers.

4) Synapse Driver: The synapse driver uses the same receiver and DLL as the repeater. After the hold time of bit 6 has passed the data capture latches contain the parallel data word. The receiver DLL provides the necessary timing information to reliably control the strobe pulse length τ_{STDF} which controls the synapse output current (see section III-A.1).

To limit the power consumption and crosstalk of the parallel L1 data a reduced voltage swing of 1/2 Vdd (0.9 V) is used inside the synapse array. The lower four address bits are decoded in the synapses. They are therefore transmitted pseudo-differentially which also reduces crosstalk significantly.

5) External Event Inputs: There are two possible sources for an L1 bus: a neuron from ANNCORE or an external event arriving at a DNC→L1 converter. The DNC→L1 converter translates the synchronous event packet into an L1 frame. The digital controller of the HICANN uses a clock frequency of $1/t_{\mathrm{frame}}$ generated by an internal PLL from the external reference clock transmitted via the DNC→HICANN link. This link uses a synchronous packet-based protocol to transmit neuron number and event time in a single packet. An internal memory is used in HICANN to store the received external events until the event time matches the local time of the digital controller. The serializer and driver of an DNC→L1 converter are the same as in the neuron→L1 interface.

6) L1 Merging Repeaters: A problem arises with the geometry of the HICANN chips described so far. When using the maximum number of inputs by connecting large numbers of *denmem* circuits through the neuron builder, the total neuron number becomes quite low. In the most extreme configuration a single HICANN implements eight neurons with 16k synapses each. With eight neurons a 6 bit L1 bus can not be fully utilized. To overcome this problem, each HICANN contains a special L1 repeater that is able to merge the output of a local neuron into a partially filled horizontal L1 bus lane. A hidden horizontal L1 lane is implemented for that purpose to avoid complications in the routing algorithm. In the middle of the HICANN chips it is interrupted and the *L1 Merging Repeater* is inserted. The driving direction is from left to right. At the rightmost HICANN, which is the HICANN that completes the L1 bus, the output of the *L1 Merging Repeater* is switched from the hidden horizontal lane to horizontal lane 1. Therefore it appears like the whole L1 bus with all its 64 neurons originates in the rightmost HICANN of such a chain.

V. Summary

This paper gives an overview of the circuit techniques necessary to implement a wafer-scale analog neural network with a programmable topology. The presented architecture solves the three main issues of wafer-scale integration: power consumption, fault tolerance and the transferability of biological networks. At the time of this writing the circuit design for the HICANN chip is completed and all the circuits presented in this paper have been proven in simulation. The tape-out of the first prototype is planned for the first half of 2008.

References

[1] W. Gerstner and W. Kistler, *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press, 2002.

[2] S. Renaud, J. Tomas, Y. Bornat, A. Daouzli, and S. Saïghi, "Neuromimetic ICs with analog cores: an alternative for simulating spiking neural networks," in *ISCAS*, 2007, pp. 3355–3358.

[3] M. Lundqvist, M. Rehn, M. Djurfeldt, and A. Lansner, "Attractor dynamics in a modular network model of neocortex." *Network*, vol. 17, no. 3, pp. 253–76, 2006.

[4] A. Morrison, A. Aertsen, and M. Diesmann, "Spike-timing-dependent plasticity in balanced random networks." *Neural Computation*, vol. 19, no. 6, pp. 1437–1467, June 2007. [Online]. Available: http://dx.doi.org/10.1162/neco.2007.19.6.1437

[5] J. Schemmel, K. Meier, and E. Mueller, "A new VLSI model of neural microcircuits including spike time dependent plasticity," in *Proceedings of the 2004 International Joint Conference on Neural Networks (IJCNN'04)*. IEEE Press, 2004, pp. 1711–1716.

[6] J. Schemmel, A. Gruebl, K. Meier, and E. Mueller, "Implementing synaptic plasticity in a VLSI spiking neural network model," in *Proceedings of the 2006 International Joint Conference on Neural Networks (IJCNN'06)*. IEEE Press, 2006.

[7] J. Schemmel, D. Bruederle, K. Meier, and B. Ostendorf, "Modeling synaptic plasticity within networks of highly accelerated I&F neurons," in *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*. IEEE Press, 2007, pp. 3367–3370.

[8] H. Markram, M. Toledo-Rodriguez, Y. Wang, A. Gupta, G. Silberberg, and C. Wu, "Interneurons of the neocortical inhibitory system." *Nat Rev Neurosci*, vol. 5, no. 10, pp. 793–807, October 2004. [Online]. Available: http://dx.doi.org/10.1038/nrn1519

[9] M. Ehrlich, C. Mayr, H. Eisenreich, S. Henker, A. Srowig, A. Grübl, J. Schemmel, and R. Schüffny, "Wafer-scale VLSI implementations of pulse coupled neural networks," in *Proceedings of Fourth International Multi-Conference on Systems, Signals & Devices (IEEE SSD07)*. Hammamet, Tunisia: IEEE Press, March 2007.

[10] J. Fieres, J. Schemmel, and K. Meier, "Algorithms for implementing biological spiking network models on a configurable wafer-scale hardware system," in *Accepted for publication in the 'Proceedings of the 2008 International Joint Conference on Neural Networks (IJCNN'08)'*. IEEE Press, 2008.

[11] K. Gilleo, *Area Array Packaging Processes: For BGA, Flip Chip and CSP*. McGraw-Hill Professional, 2003.

[12] A. Kalampokis, C. Kotsavasiloglou, P. Argyrakis, and S. Baloyannis, "Robustness in biological neural networks," *Physica A*, vol. 317, no. 3, pp. 518–590, 2003.

[13] C. Hampson, "Redundancy and high-volume manufacturing methods," *Intel Technology Journal Q4 1997*, 1997.

[14] "Zebra elastomeric connectors." [Online]. Available: www.fujipoly.com

[15] M. Mahowald, *An Analog VLSI System for Stereoscopic Vision*. Kluwer, 1994.

[16] A. Mortara and E. A. Vittoz, "A communication architecture tailored for analog VLSI artificial neural networks: intrinsic performance and limitations," *IEEE Trans. on Neural Networks*, vol. 5, pp. 459–466, 1994.

[17] R. Brette and W. Gerstner, "Adaptive exponential integrate-and-fire model as an effective description of neuronal activity," *J. Neurophysiology*, pp. 3637–3642, 2005.

[18] A. Srowig, "Analog floating gate memory in a 0.18 μm single-poly CMOS process," *Internal FACETS documentation.*, 2007.