CMOS compatible Ising and Potts annealing using single-photon avalanche diodes: Supplementary material

William Whitehead, Zachary Nelson, Kerem Camsari, Luke Theogarajan

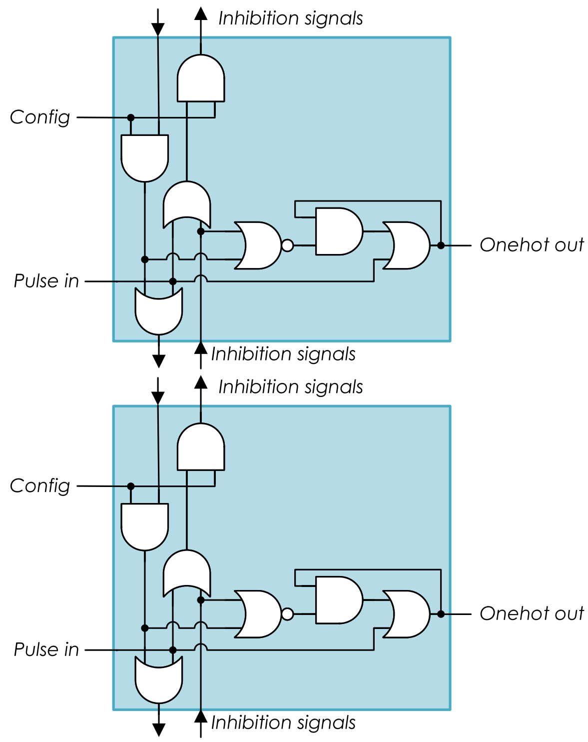

Figure S1: Logic design of a scalable one-hot latch.

1 Scalable one-hot latching circuit

The one-hot latch required for Potts model CTMC neurons can be implemented as in figure S1. This circuit is composed of a chain of repeating elements (blue rectangle; two are shown). Each circuit has a pulse input that when high, sends an 'inhibition' signal cascading through all connected elements which sets the output of all elements to zero. Meanwhile, the output of the latching element with the pulse input is set to high (and sees no inhibition signal); thus for q connected elements, one will always have a high output and the rest will have low outputs. Additionally, each element can come equipped with a configuration input that can block or pass the inhibition inputs. When the inhibition signals can pass, the two adjacent elements (and their corresponding variable-rate SPADs) competitively participate as a single Potts node; when the inhibition signals are blocked, the two adjacent elements (and all elements to each side) do not compete; this can effectively convert a string of q latching elements into two strings of smaller size. In this way a bank of SPADs and latching elements may be configured into any number of CTMC neural configurations, allowing a single Potts annealer to operate with any combination of Potts model node sizes.

As an asynchronous circuit with potentially many stages of gates, two concerns arise regarding timing: the behavior under coincident pulse inputs and effects on the annealing algorithm as the latch is transitioning and the one-hot constraint is momentarily not met. In the first case, near-coincident pulse inputs have the capability to leave the one-hot latch with no high outputs. This would occur if the falling edges of two pulses were coincident, leaving an inhibitory signal in all elements but no excitation signals. Since pulse events are stochastic by design, it is inevitable that this will happen sometimes. Similarly, delay in the inhibitory signal means that there will usually be brief instances during a normal transition during which multiple outputs are hot. Both of these phenomena will be felt at the algorithmic level as intermittent inaccuracy of the energy calculations, which may or may not be problematic depending on the optimization problem. In general however the incorrect energy calculation will be minimal: if a CTMC neuron changes state at roughly 100 MHz, and the delay through each latching stage is only 10 or 20 ps, a $q = 10$ latch would be 100x faster than its input and would spend a similarly small portion of time in an incorrect state.

2 Mapping an Ising model to a Potts model

During testing we matched a single theoretical distribution to the samples from both the Ising and Potts Boltzmann machines. This required mapping from an eight-neuron Ising model to a Potts model with four four-state neurons. Ising states, biases, and weights will be denoted as In, Ih, and Iw respectively; the Potts states, biases, and weights will be Pn, Ph, and Pw. Each state of each Potts neuron is assigned a pair of states from two Ising neurons, shown in table S1. The bias of each Potts state is the energy that state represents intrinsically, without interacting with other neurons in the Potts model; it includes the biases of the two Ising neurons that it is representing, and also the interaction energy between those two Ising neurons since they are subsumed within the single Potts neuron, also shown in table S1.

Pn_i	In_{2i-1}	In_{2i}	$Ph_i(Pn_i)$
1	-1	-1	$-Ih_{2i-1} - Ih_{2i} + Iw_{2i-1,2i}$
2	-1	+1	$-Ih_{2i-1} + Ih_{2i} - Iw_{2i-1,2i}$
3	+1	-1	$+Ih_{2i-1} - Ih_{2i} - Iw_{2i-1,2i}$
4	+1	+1	$+Ih_{2i-1} + Ih_{2i} + Iw_{2i-1,2i}$

Table S1: Defining one Potts neuron based on two Ising neurons.

Since a weight between two Potts neurons represents the interaction between four Ising neurons, there are four Ising weights that each Potts weight must account for (the two additional Ising weights found in a network of four Ising neurons are accounted for in the biases in table S1). Thus when mapping from the Ising model to the Potts model, each Potts weight is based on four terms,

$$Pw_{i,j}(P_i, P_j) = \pm Iw_{2i-1,2j-1} \pm Iw_{2i-1,2j} \pm Iw_{2i,2j-1} \pm Iw_{2i,2j} \tag{1}$$

The sign of each term is determined by the corresponding signs of In_{2i-1}, In_{2i}, In_{2j-1} and In_{2j}. There are 16 combinations of signs, and 16 weight values connecting Potts neurons i and j.

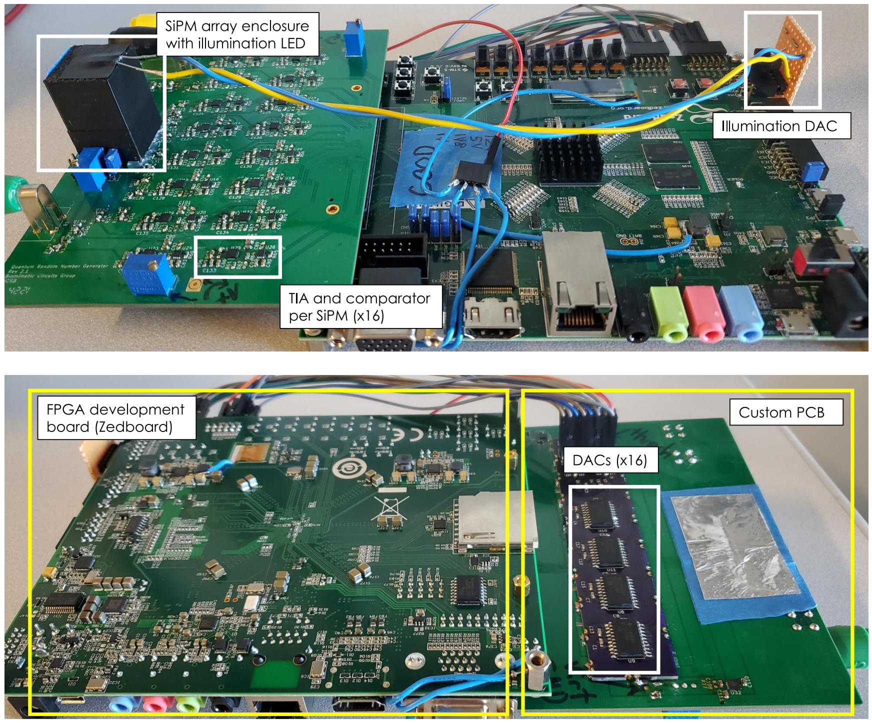

Figure S2: Hardware for demonstrating SPAD CTMC neurons.

3 Hardware Implementation

All experimental data demonstrating SPAD CTMC neurons was collected using the hardware setup shown in figure S2. The setup consisted of 16 variable-rate SPAD circuits implemented on a custom PCB and an FPGA development board (Zedboard). Asynchronous pulses from the variable rate SPADs (ON Semi ARRAYJ-30020-16P-PCB, using 16x LMH6629 for transimpedance sense amplifiers) were captured by the FPGA and used to control either binary RS latches (forming Ising neurons) or four-way latches (forming Potts neurons). The latch states controlled adder trees inside the FPGA which calculated values (energies) to be written to the DAC of each variable-rate SPAD. Due to the evolutionary nature of research, the DACs exist on an add-on PCB attached to the back of the main PCB. Each time one of the neural state latches changed, all 16 DACs were updated; since the DACs could only be written over a serial interface (4x AD7226; 8-bit parallel, 50 MHz, 16 cycles to update all 16 DAC outputs), SPAD pulses were ignored until the DAC update completed in order to maintain correct sampling statistics. In a scaled CMOS implementation, this delay would not be present. The DACs set the thresholds on comparators (16x MAX40026) in each variable-rate circuit, affecting the pulse rate of each SPAD and in turn controlling the distribution of latch states held in the FPGA, forming a Boltzmann machine. The behavior of this system was studied using the accompanying processor on the Zedboard's ZYNQ 7020 SoC, which ran Ubuntu with the PYNQ software layer, allowing fast collection of large quantities of data from a Jupyter Notebooks interface. Illumination of the SiPM array was controlled by driving an LED with a DAC; the LED light was attenuated before reaching the SiPMs, and the scale of the illumination intensity was not measured. Thus the illumination intensity is only reported in arbitrary units (corresponding to the value set on the DAC).

4 Sampling methods

Many measurements of the behavior of CTMC neurons, from the transfer function of single variable-rate SPADs to distributions of complete Ising and Potts annealers, required statistical sampling. In general enough samples were accumulated such that deviations from ideal behavior could be attributed to the system itself rather than the distribution of the sample mean.

4.1 Variable-rate SPAD transfer functions

Two different sampling methods were used to measure the threshold-to-pulse-rate transfer functions of variable rate SPADs; both methods gave the same results. The first method, which was only applied to the experimental variable-rate SPAD circuits, was to directly count the pulse edges received from the comparators by the FPGA. In this case a threshold was physically set using the DACs and the resulting pulse edges were counted over a 100 ms time period by the FPGA; this was done iteratively over the full range of the 8-bit DACs, yielding 256-point transfer functions.

The second method was a computational analysis of the filtered SPAD outputs shown in figures (xx), obtained either from simulation or directly measured from the transimpedence amplifier using an oscilloscope. In this method the comparator component of the variable-rate SPADs was performed computationally: using a single trace, positive crossings of a set threshold were counted, and the threshold was iteratively set to different values to measure a transfer function. Performing this measurement on raw traces yielded higher event rates than (but the same exponential relationship as) the first method; this discrepancy was due to high-frequency sequential threshold crossings that were filtered out by the actual hardware. Performing a low-pass filtering operation before computationally measuring the variable-rate SPAD transfer functions brought the magnitude of the rates in line with those obtained directly from hardware.

The computational approach to measuring transfer functions was required especially for the simulation circuit results. Otherwise a few hundred nearly identical simulations, each with a different threshold, would have been required; this would have been computationally prohibitive. Instead computational time was spent on simulating a longer single trace, so that the tail of the transfer function could be measured with greater accuracy.

Table S2 lists lists measurement conditions and associated sampling uncertainty for the various variable-rate SPAD transfer function measurements. The uncertainty depends on how many pulses were recorded during the sampling period, with higher pulse rates resulting in lower uncertainty. The table lists the lowest measured pulse rates and associated number of detected pulses, from which a worst-case relative measurement error is determined. The measurement quantity (number of pulses over the measurement timeframe) is modeled as a Poisson distribution.

circuit	threshold	figures	sampling time	min. count @ rate	relative error
simulated	computational	5g, 5h	242 μs	242 @ 1 MHz	0.064
experimental	computational	5e, 5f	10 ms	100 @ 10 kHz	0.100
experimental	en vivo	6	100 ms	1K @ 10 kHz	0.031

Table S2: Relative sampling error in variable-rate SPAD transfer function measurements

4.2 Tanh Ising neuron transfer curves

The transfer function of Ising neurons, which were only measured for the experimental demonstration of SPAD CTMC neurons, was directly done en vivo by sweeping the biases h_i of each neuron while leaving the weights w_ij unused. This was done after calibrating the rates of the individual variable-rate SPADs to be equal (by introducing bias offsets to each SPAD circuit). With a set h_i bias 10^5 samples of the neuron states were collected, and the fraction of samples in the $+1$ state were tabulated for each Ising neuron. This process was repeated for each integer bias value in the range $[-75, 75]$, representing the full usable input range of each neuron. Since updates in a SPAD CTMC neuron are not clocked but occur whenever a SPAD circuit produces a pulse, samples were taken at a relatively slow 200 kHz so that successive samples were not too correlated. Thus each point on the measured neuron transfer function represents the mean of 10^5 roughly

I.I.D. samples of a Bernoulli random variable, yielding a worst-case measurement uncertainty of $7x10^{-4}$ at neutral bias.

4.3 Ising and Potts model distributions

The distribution of samples produced by SPAD CTMC Ising and Potts models was measured by setting up a network configuration (weights, biases, and computational temperature) and then recording 5 million samples of the network state. As with the measurement of individual Ising neuron transfer characteristics, consecutive samples are not independent. For full Ising and Potts models the issue is even more acute, since it can take many updates for the collective model state to transition between regions of its state space; this is often referred to as the mixing problem, and its severity further depends on the particular configuration of weights and biases. In addition, samples were recorded on a 50 MHz clock even though DAC update pausing (see Hardware Implementation) prevented the state from updating faster than 2 MHz. In light of these obfuscations we present empirical evidence of measurement uncertainty instead of performing any rigorous bounding. The 5 million samples were collected in 50 sub-batches, and after each batch the Kullback-Leibler (KL) divergence between the measured distribution (cumulative across previously captured samples) and the theoretical distribution was calculated, shown in figure S3. This figure shows that rather than continuing to decrease, the KL divergence plateaus, indicating that the difference between the measured and theoretical distributions is real and not an artifact of measurement uncertainty - otherwise the KL divergence would continue to decrease as more samples were recorded. From this we can conclude that the measurement uncertainty of the Ising and Potts distributions is less than the visible deviations from the ideal distribution, and that, unsurprisingly, the SPAD-based Boltzmann machine does slightly deviate from ideal behavior.

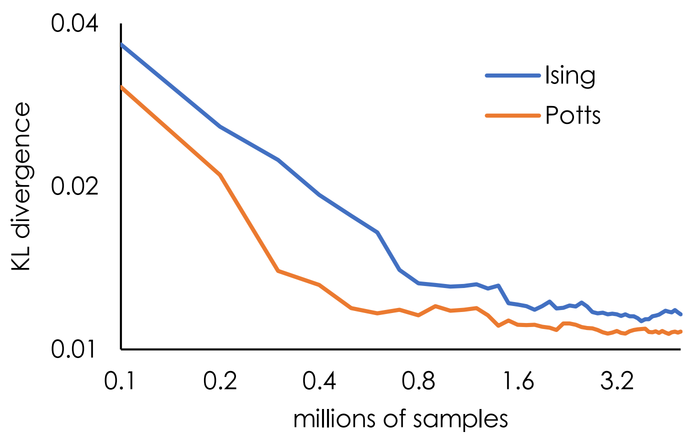

Figure S3: Kullback-Leibler divergence of distributions sampled from Ising and Potts SPAD CTMC neurons, in reference to the theoretical distribution.