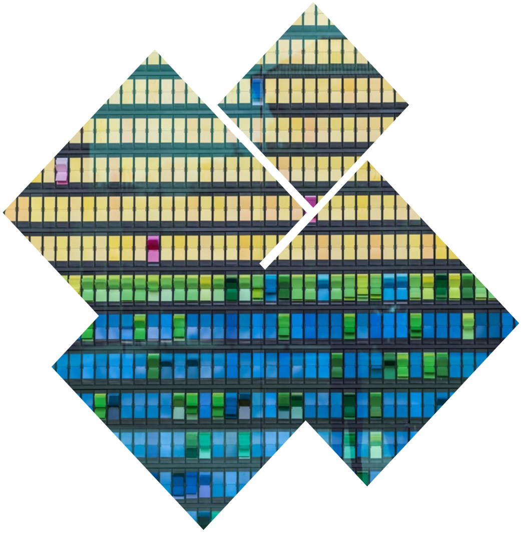
White Paper

Efficient Restricted Boltzmann Machine Training for Deep Learning

Reimagine the Impossible with MemComputing

MemComputing Inc.



Executive Summary

Today, deep learning has permeated many facets of daily life extending from facial recognition for smart photo albums to spam filtering and content recommendations on social networks. With the exponential growth in data, deep learning is poised to transform the landscape of a wide variety of data-driven industries such as medical diagnostics, autonomous vehicles, financial technology, language processing, aviation, and security, among many others.

A key aspect of unlocking the potentials of deep learning is improving the unsupervised training necessary for Deep Neural Networks (DNNs). The current standard training method for DNNs is Contrastive Divergence (CD) (or Gibbs sampling) which provides improvements from prior training methods in terms of speed. However, CD can still be time-consuming given the slow mixing of Gibbs sampling which contains inherent noise. Additionally, it risks being stuck in local minima.

In order to boost the performance of DNNs and leverage its full potential to solve real-world problems, **researchers have proposed numerous approaches to resolve current issues with generative training**. Among these approaches, quantum annealing has generated research interest. However, quantum annealing faces numerous implementation challenges due to the material constraints of quantum computing hardware which prevents it from being available for adoption as a practical solution.

Meanwhile, a new physics-based approach, memcomputing, that is readily scalable has demonstrated notable improvements in speed for generative training and accuracy of predictions in Deep Neural Networks (DNNs).

1 Introduction

Deep learning is a machine learning technique distinguished by its stacked neural network architecture. Unlike the single-hidden-layer structure of traditional machine learning, deep learning consists of multiple hidden layers between the input and output layers. Thus, data goes through a multi-step process which uses the output from the prior layer as input.

One type of Deep Neural Networks (DNNs) are Restricted Boltzmann Machines (RBMs).

- RBMs have visible nodes and hidden nodes.
- Weights and biases relate every visible node to every hidden node.
- However, RBMs are unique in that intra-layer connections are restricted.
- That is, no visible node is connected to another visible node and no hidden node is connected to another hidden node.

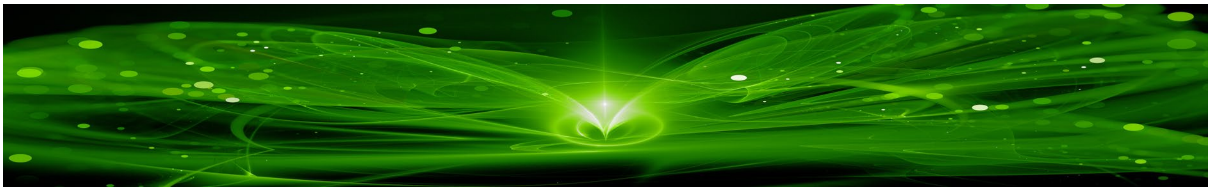
Training for DNNs or RBMs typically involves two phases.

- The first phase consists of the unsupervised, generative training of each individual RBM. The standard training method for this stage is Contrastive Divergence (CD) or Gibbs sampling. Subsequently, CD is followed by the second phase which involves supervised discriminative training.
- Discriminative training fine-tunes the weights and biases in the network by using backpropagation to find the gradient of each weight in relation to the outputs.
- Another term to describe the first stage is pre-training. Effective pre-training is critical to efficient backpropagation which is necessary to optimize the accuracy of a model's predictions.

In RBMs, each layer is trained on a unique set of features given by the previous layer's output.

- Thus, RBMs increase in complexity as one advances deeper into the layers of the neural net.
- This characteristic of RBMs gives them the ability to work with high-dimensional datasets.
- This makes them well-suited for features learning in image processing and classification required for machine vision, real-time threat detection, and photo search.

Furthermore, its ability to train with large amounts of unlabeled data gives it a distinct advantage since the accuracy of neural nets is dependent on the size of its training data. Not to mention, the ability for RBMs to train without supervision makes it beneficial both for organizations working with enterprise-scale datasets as well as organizations with small data science teams looking to scale with limited number of experts.



Gibbs Sampling

As mentioned, pre-training of an RBM can be achieved by using **Gibbs sampling to run a Markov chain to convergence (Contrastive Divergence).** However, the time it takes to reach convergence at each gradient step is slow. CD learning updates weights and biases using

$$\Delta w_{ij} = \epsilon[\, \langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{recon}] \quad (1)$$

which is generally completed with a single-step reconstruction. However, due to the slow mixing of Gibbs sampling and inherent noise, generative training time can still be long.

Quantum Annealing

- **One proposed solution is to use a quantum computer such as the D-Wave quantum annealing machine** to replace classical Gibbs sampling.
-
- **Quantum annealing utilizes quantum tunnelling to find the minimum energy state of the energy landscape to solve problems heuristically**. Given a problem that can be articulated as an energy landscape, quantum tunnelling leverages the abilities of electrons to pass through energy barriers to go directly from one local minimum to another without having to overcome tall, narrow energy peaks.

- At the core of quantum annealing are qubits that exist in superposition of the 0 and 1 states simultaneously. When n qubits are entangled they behave as a single object with 2n potential states. Biases and programmed couplings determine the relative energies of each state. The number of states thus increases exponentially with each additional qubit. **At the end of the quantum anneal, each qubit goes into a 0 or 1 state and represents the minimum energy state.**

- However, the **undesirable interaction between qubits and their environment decreases their utility as an optimization solution**. Nonetheless, it is precisely its non-ideality that inspired researchers to use it instead as a sampling machine in order to accelerate CD learning. Instead of attempting to find the minimum energy state of an energy landscape, like in the case of optimization, sampling solves for a number of low-energy states.

- The D-Wave quantum annealer begins with problems represented by an Ising model $E(\boldsymbol{s}) = \boldsymbol{s}^T \boldsymbol{J} \boldsymbol{s} - \mu \boldsymbol{h}^T \boldsymbol{s}$ (2) where given the variable, $S_i \in \{-1,1\}$, hi is the weight of site i and Jij is the coupling strength between site i and j. Then, RBMs are embedded onto the quantum annealing hardware graph. After a set number of annealing runs, model expectations were calculated using the samples generated from the anneal which were compared to the correct expectation.

- In the case of RBM training, the weights and biases are updated by computing $\Delta w_{ij} \propto \langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model}$ (3) where the second term is estimated using quantum annealing.

- Experimental results suggest that quantum sampling requires fewer iterations in pre-training and discriminative training to reach a preset, post-training accuracy level. However, the results do not dramatically exceed the capabilities of classical algorithms. [1]

- Additionally, the D-Wave quantum annealing machine has significant challenges with hardware implementation. A number of qubits are rendered unusable due to production and calibration difficulties. This means that a bipartite RMB graph could not be fully completed. The D-Wave machine itself is also extremely expensive. Lastly, its limited chip size prohibits the scaling of RBM layers.

2 MemComputing Training Approach

- **MemComputing technology is based on the theoretical concept of universal memcomputing machines (UMMs).** UMMs are a class of memcomputing machines built with interconnected memory units (memprocessors) capable of performing computation in and with memory.

- The scalable version of UMMs is practically realized in the form of digital memcomputing machines (DMMs). DMMs harness the power of self-organizing logic circuits (SOLCs) which are differentiated from traditional circuits through the unique properties of the self-organizing logic gates (SOLGs) used in their construction. These SOLGs, in turn, can be realized in hardware with available (non-quantum) technology or simulated efficiently in software.

Nonlocal Collective State Computation

- **The most significant feature of SOLCs is its manifested long-range order**. Long-range order describes physical systems which demonstrate correlated behavior across remote particles. In other words, systems with long-range order contain components that correspond to the states of other components regardless of distance. This simultaneous collective responsiveness of individual parts describes the temporal and spatial non-locality of the system.

- **The capability of SOLCs to realize long-range order is due to the existence of instantons**. Instantons connect topologically inequivalent critical points in the phase space. They are the classical analogue of quantum tunnelling. Instantons create non-locality in the system which generate the collective, dynamic behavior of SOLGs to correlate at an arbitrary distance. In effect, this collective behavior allows SOLGs to efficiently adapt their truth value to satisfy the logical proposition of another gate without violating their own internal logical proposition. The nonlocality of SOLCs thus allows for simultaneous variable flips which is a necessary task that combinatorial approaches cannot accomplish once they reach a certain number of satisfied constraints.

- **It is precisely the long-range order of SOLCs that produces computation acceleration by orders of magnitude.** As discussed in greater detail in the next section on the demonstrated performance of Falcon©, the system converges quickly to the equilibrium points which represent current closest approximations to the global optimum for complex optimization problems.

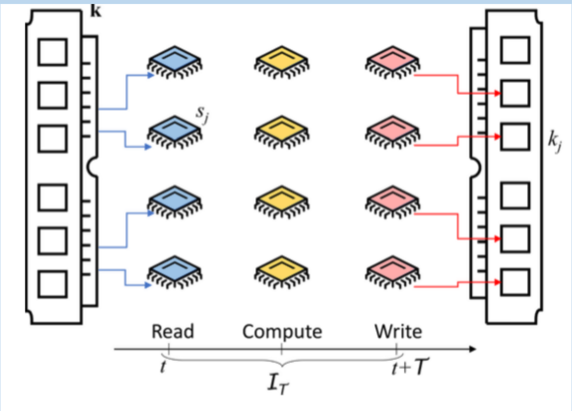
Software Scalability in Poly-Time

A key feature of the MemComputing solution is the ability of DMMs to scale in poly-time, quite often in linear or sub-quadratic time. It is important to note that the polynomial scaling is independent of the input size because the number of logic gates grows linearly with each step requiring only a linear number of floating-point operations and linearly growing memory. In other words, the number of variables can be increased without an exponential growth in computation time which resolves the primary issue with conventional computation solutions.

The configuration of DMMs outlined here is able to support infinite-range correlations in the infinite size limit. This allows for an ideal scale-free behavior of the SOLC in which the correlations do not decay. This was derived analytically using topological field theory.

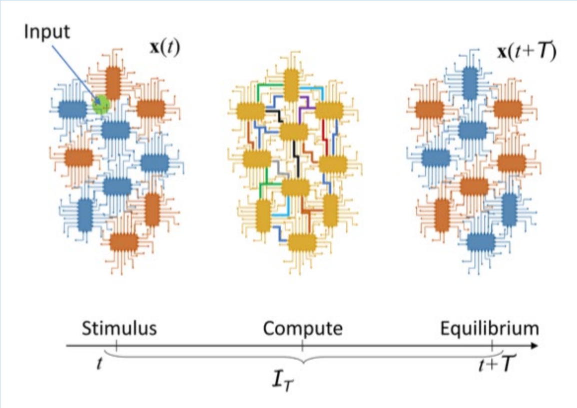
A new logic Framework: Self-Organizing Logic Gates (SOLGs)

Standard Parallelism



Multiple central processing units (CPUs) and parallel machines are becoming the norm in term of computers nowadays. In such parallel machines, all CPUs are synchronized: each of them performs a task in an interval of time T. **At the end of the clock cycle**, and only at the end of the clock cycle, all CPUs share their results, and follow up with the subsequent task.

Intrinsic Parallelism



At any given time, any element of the machine is "knows" what the other elements are doing. Indeed, the the physical interaction among the different constituents of the machine provides collective dynamics to the whole system.

3 Comparative MNIST Results

- In 2015, Lockheed Martin conducted an experiment with the D-Wave quantum annealing machine to accelerate the training needed to improve the inference accuracy of RBMs used to detect handwriting digits from the MNIST database. [1]
-
- Instead of traditional Gibbs sampling, the RBM was trained with a quantum sampling-based training approach in place of CD-based training. While the results showed comparable or improved accuracy with a decrease in the number of iterations necessary for generative training, the hardware challenges of the D-Wave quantum machine severely limit its applications and performance.
-
- MemComputing replicated the experiment using a software solution with comparable pre-training iterations but smaller variations in accuracy as compared to the results of training based on quantum generated sampling.
-
- Additionally, unlike the D-Wave quantum annealing machine, MemComputing's software solution is realizable in available non-quantum electronic components and thus is readily scalable.

Experimental Setup

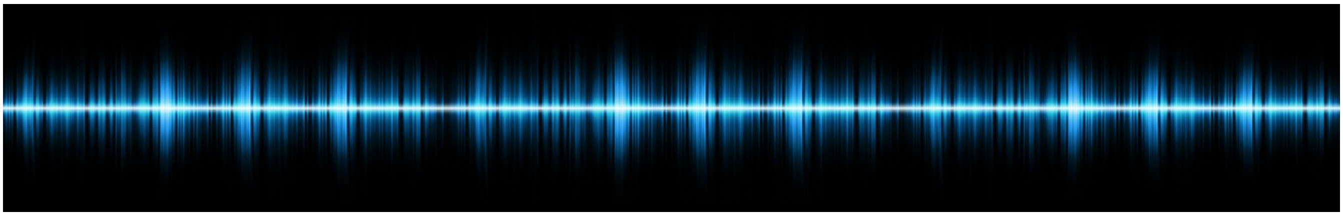
- The MNIST handwritten digits is a standard benchmark for evaluating machine learning algorithms. Each image has 784 greyscale pixels (28x28) which represent handwritten numbers from 0-9. The dataset as a whole has 60,000 training and 10,000 test set images with truth labels.

- In order to provide a direct comparison with the D-Wave quantum annealer results, each image size was reduced to 32 pixels and the RBM size was chosen to have 32 visible and 32 hidden nodes.

The Procedure

- Falcon© was used to sample the lowest-energy configuration of the RBM cost function. The latter can be written as a QUBO problem, which, in turn, can be solved as a weighted Max-SAT problem.

- After pre-training, backpropagation iterations were run using mini-batches to fine-tune the weights and biases.



Quantum annealer vs MemComputing
Performance Comparison after backpropagation iterations

- The **blue** plot on the left represents the results of training using quantum annealing.
- The red plot on the right represents the results of training using MemComputing.
- The **horizontal axis** shows the number of pre-training iterations.
- The **vertical axis** indicates accuracy.
- The **dotted line** is the accuracy of the training set which was averaged over 10 trials.
- The **solid line** is the accuracy of the test set, also averaged over 10 trials.
- **Error bars** present ±1 standard deviation for each trial.

Figure 1
After **100** backpropagation iterations

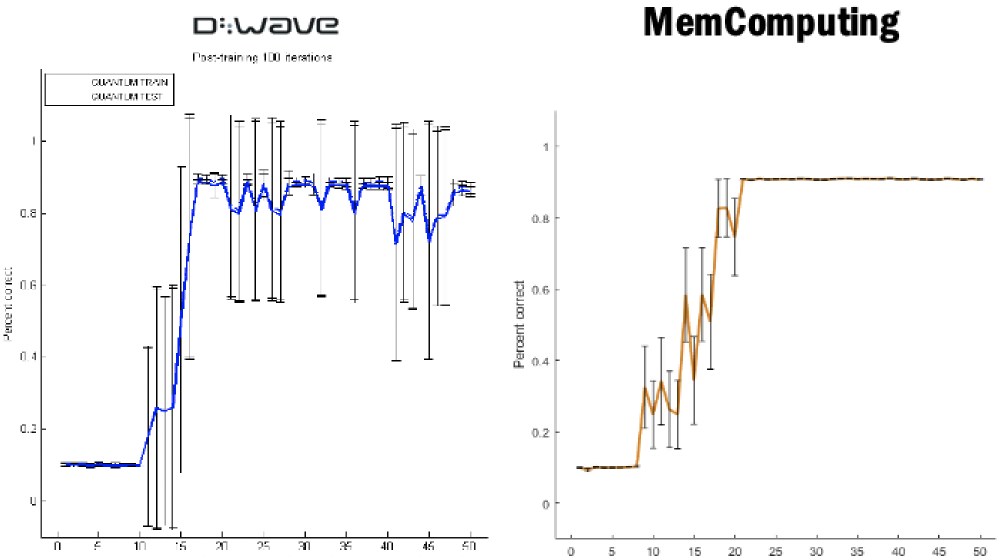
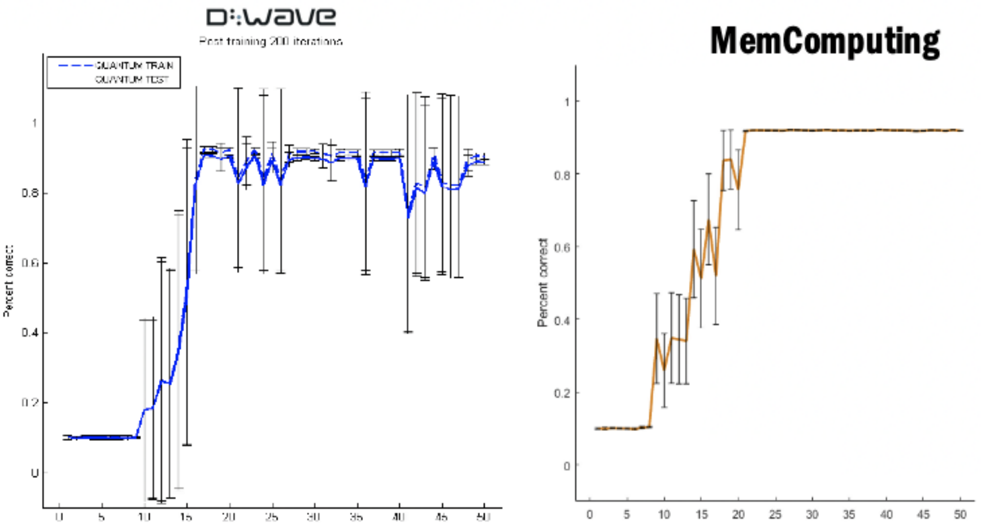


Figure 2
After **200** backpropagation iterations



Quantum annealer vs MemComputing
Performance Comparison after backpropagation iterations

Figure 3
After **400** backpropagation iterations

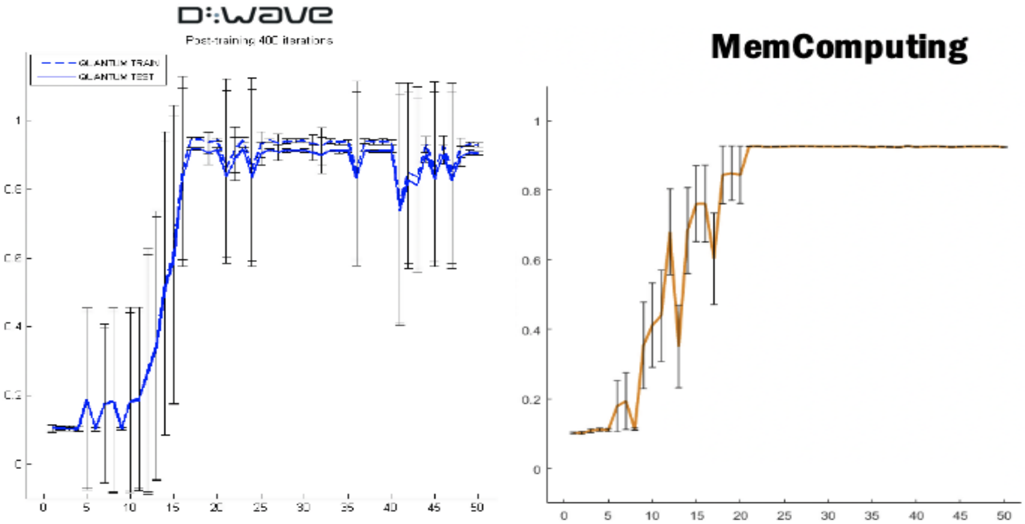
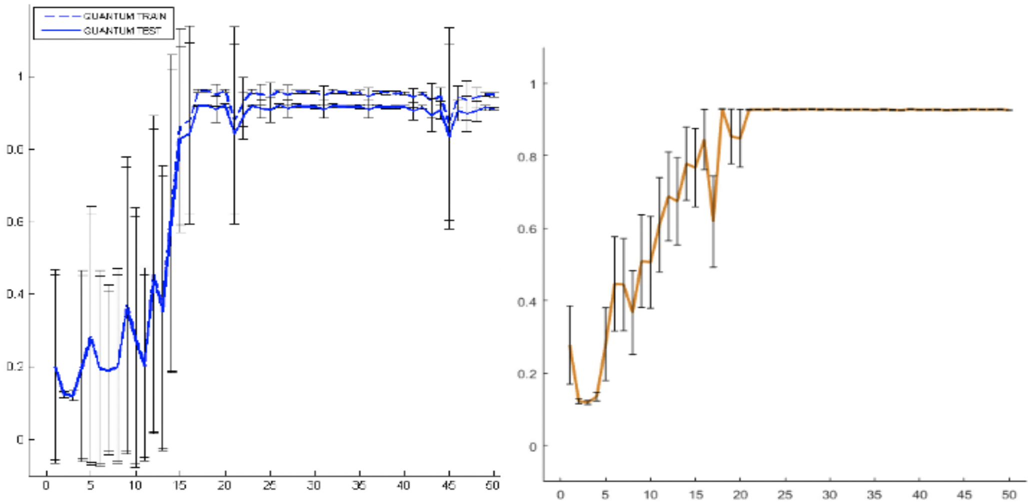


Figure 4
After **800** backpropagation iterations



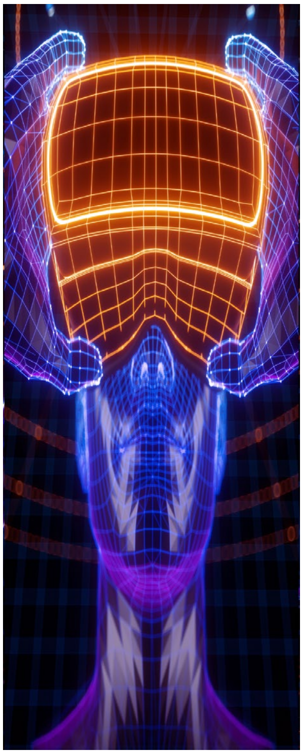
The Results

☑ The results from this experiment demonstrated that **generative training, and by extension discriminative training, can be accelerated without relying on quantum sampling-based training**.

☑ **The software MemComputing solution demonstrated a comparable accuracy as the hardware quantum annealer.**

☑ However, **the variations in accuracy of the MemComputing solution were smaller than the quantum annealer trained for the same number of pre-training and backpropagation iterations**. This difference is especially dramatic in Figure 4 where there were 800 backpropagation iterations.

4 CONCLUSION



- ✓ In conclusion, we have shown empirical evidence that **utilizing DMMs to accelerate generative training for DNN outperforms quantum sampling-based training** in both training speed and prediction accuracy.

- ✓ **MemComputing leverages nonlocal, collective behavior but does not suffer from the hardware constraints of quantum computing** which means that it is readily scalable for DNNs training with high dimensional, multi-modal datasets.

- ✓ In relation to traditional Gibbs samples, **MemComputing accelerated pre-training iterations considerably while increasing the quality of the system predictions**. With the accessibility of non-quantum electronic components, MemComputing is able to be immediately implemented to solve problems in healthcare diagnostics, autonomous vehicle, financial technology, and many other industries seeking to harness the power of Deep Neural Network training.

Reference

[1] S. H. Adachi and M. P. Henderson, "Application of quantum annealing to training of deep neural networks," arXiv:1510.06356, 2015.
[2] H. Manukian, F.L. Traversa, and M. Di Ventra, "Accelerating deep learning with memcomputing", Neural Networks, 110, 1 (2019).

MemComputing, Inc.'s disruptive coprocessor technology is accelerating the time to find feasible solutions to the most challenging operations research problems in all industries. Using physics principles, this novel software architecture is based on the logic and reasoning functions of the human brain.

MemComputing enables companies to analyze huge amounts of data and make informed decisions quickly, bringing efficiencies to areas of operations research such as Big Data analytics, scheduling of resources, routing of vehicles, network and cellular traffic, genetic assembly and sequencing, portfolio optimization, drug discovery and oil and gas exploration.

The company was formed by the inventors of MemComputing, PhD Physicists Massimiliano Di Ventra & Fabio Traversa and successful serial entrepreneur, John A. Beane.

MemComputing Inc.
4250 Executive Square,
Suite 200,
La Jolla, CA 92037

http://www.memcpu.com
info@memcomputing.com

THE INFORMATION IN THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.