
AFRL-RY-WP-TR-2024-0134

# SPINTRONIC STOCHASTIC DATAFLOW COMPUTING

**Haris Suhail, Jiyue Yang, Haoran He, Vinod Kurian Jacob, Alexander Graening, Puneet Gupta, Kang L. Wang, and Sudhakar Pamarti**

**University of California Los Angeles**

**JUNE 2024**
**Final Report**



**AIR FORCE RESEARCH LABORATORY**
**SENSORS DIRECTORATE**
**WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7320**
**AIR FORCE MATERIEL COMMAND**
**UNITED STATES AIR FORCE**

# NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with The Under Secretary of Defense memorandum dated 24 May 2010 and AFRL/DSO policy clarification email dated 13 January 2020. This report is available to the general public, including foreign nationals.

Copies may be obtained from the Defense Technical Information Center (DTIC) (http://www.dtic.mil).

AFRL-RY-WP-TR-2024-0134 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

//Signature//

_____

CHRISTOPHER A. BOZADA
Program Manager
Aerospace Components and Subsystems Division

//Signature//

_____

JOHN S. CETNAR (Acting)
Deputy Chief, Aerospace Components &
Subsystems Technology Division
Sensors Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

*Disseminated copies will show "//Signature//" stamped or typed above the signature blocks.

# REPORT DOCUMENTATION PAGE

**PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION.**

| 1. REPORT DATE | 2. REPORT TYPE | 3. DATES COVERED | |
|---|---|---|---|
| | | START DATE | END DATE |
| June 2024 | Final | 23 August 2018 | 31 August 2023 |

**4. TITLE AND SUBTITLE**
SPINTRONIC STOCHASTIC DATAFLOW COMPUTING

| 5a. CONTRACT NUMBER | 5b. GRANT NUMBER | 5c. PROGRAM ELEMENT NUMBER |
|---|---|---|
| FA8650-18-2-7867 | N/A | 61101E |
| **5d. PROJECT NUMBER** | **5e. TASK NUMBER** | **5f. WORK UNIT NUMBER** |
| N/A | N/A | Y1UC |

**6. AUTHOR(S)**
Haris Suhail, Jiyue Yang, Haoran He, Vinod Kurian Jacob, Alexander Graening, Puneet Gupta, Kang L. Wang, and Sudhakar Pamarti

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**
University of California Los Angeles
405 Hilgard Ave,
Los Angeles, CA 90095

**8. PERFORMING ORGANIZATION REPORT NUMBER**

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**

Air Force Research Laboratory
Sensors Directorate
Wright-Patterson Air Force Base, OH 45433-7320
Air Force Materiel Command
United States Air Forces

Defense Advanced Research Projects Agency
DARPA/MTO
675 North Randolph Street
Arlington, VA 22203

**10. SPONSOR/MONITOR'S ACRONYM(S)**
AFRL/RYD

**11. SPONSOR/MONITOR'SREPORT NUMBER(S)**
AFRL-RY-WP-TR-2024-0134

**12. DISTRIBUTION/AVAILABILITY STATEMENT**
DISTRIBUTION STATEMENT A. Approved for public release; distribution is unlimited.

**13. SUPPLEMENTARY NOTES**
This work was funded in whole or in part by Department of the Air Force contract FA8650-18-2-7867. This material is based on research sponsored by the Air Force Research Laboratory (AFRL) and the Defense Advanced Research Projects Agency (DARPA) under agreement number FA8650-18-2-7867. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Research Laboratory (AFRL), the Defense Advanced Research Projects Agency (DARPA), or the U.S. Government. Report contains color.

**14. ABSTRACT**
This program addressed the memory bottleneck problem in traditional Von Neumann computing architecture. This particularly challenging problem limits advances in artificial intelligence applications because they have an insatiable need for memory. This effort focused on two novel approaches to overcome the bottleneck: new magnetic memory technology and a stochastic computing framework.

**15. SUBJECT TERMS**
Von Neumann architectures, memory bottleneck, magnetic memory, stochastic computing framework, magnoelectric random access memory (MeRAM), magnetic tunnel junction devices

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES |
|---|---|---|---|---|
| a. REPORT | b. ABSTRACT | C. THIS PAGE | | |
| Unclassified | Unclassified | Unclassified | SAR | 73 |

| 19a. NAME OF RESPONSIBLE PERSON | 19b. PHONE NUMBER *(Include area code)* |
|---|---|
| Christopher Bozada | N/A |

# Table of Contents

**Section** **Page**

**Section** **Page**

# List of Figures

**Figure** **Page**

**Figure** **Page**

**Figure** **Page**

# 1 SUMMARY

The DARPA FRANC program challenged researchers to address the so-called memory bottleneck problem in computing systems based on the traditional 'von Neumann' architecture. This is a particularly challenging problem that concerns the modern way of life given the increasing dependence on artificial intelligence applications and their insatiable need for memory and computing power. In this work, we tackle these challenges through two novel approaches.

New magnetic memory technology, MeRAM (Magneto-electric RAM): This method introduced a new magnetic memory technology named MeRAM, a 1T-1MTJ non-volatile memory based on a Voltage-controlled Magnetic Tunnel Junction (MTJ) device. The MeRAM is CMOS compatible, offers very high density, and unlike other MTJ based memories such as STT-MRAM that offer similar benefits, it offers >20x lower write energy and read energy and delay performance comparable to SRAM. Our main contributions in this project are multi-fold.

1. We developed a material stack that achieved the highest Voltage Controlled Magnetic Anisotropy (VCMA) coefficient ever reported under CMOS compatible conditions.
2. With the help of the Industrial Technology Research Institute (ITRI), Taiwan, we successfully achieved the 1st ever CMOS-integrated MeRAM – up to 64kb arrays of 100nm diameter MTJ devices. We demonstrated proper memory operation with ~65% yield, and 0.7ns write latency which is far better than state of the art in other magnetic memories.
3. Since cost and technology availability constraints forced us to integrate in an old (0.18um) CMOS node, we additionally developed appropriate low energy, low latency memory access circuitry and verified it in a layout parasitic aware simulation using a MTJ compact model derived from measured device properties.
4. We demonstrated the 1st ever voltage controlled MTJ based true random number generator (TRNG) under two scenarios: (a) MTJs on a separate die, wire bonded to a 65nm die with TRNG access circuitry, and (b) MTJs integrated on 0.18um CMOS. Even in 0.18um CMOS, we demonstrated better throughput than state-of-the-art STT-MRAM based TRNGs in 28nm CMOS. Furthermore, in case (a), we have showed that the TRNG passes all NIST randomness tests using a simple bias correction circuit.

In terms of program metrics, we achieved a single MTJ read Energy-Delay Product (EDP) of 2.1 fJ x 0.4 nsec (better than the metric #1 target of 1 fJ x 1 nsec), and a random number generator performance metric of 17 fJ/b which is very close to the metric #2 target of 10 fJ/b.

Stochastic Computing (SC) framework: This approach developed a novel computing paradigm where numbers are represented as fractions of 1s in a random (or pseudorandom) binary stream. The peculiar number representation significantly reduces the hardware footprint of complex operations like multiply-and-accumulate enabling massive parallelization of on-chip computation in direct contrast to the traditional von Neumann architecture. This reduces data movement by an order of magnitude with substantial improvements in both energy and latency. Our main contributions in this project are multi-fold:

1. We developed number representation and SC-aware training techniques to overcome the inference accuracy gap between SC and conventional fixed-point computation. Where previous researchers believed that SC needed extremely long bit streams to achieve parity

with fixed-point, we demonstrated results to the contrary and showed EDP benefits at the same accuracy for the CIFAR-10 dataset and small and medium sized CNNs.

2. We developed compute-in-memory (CIM) techniques with SC and showed that energy and area computational efficiency metrics that are better than CIM can be achieved without the need for the analog to digital converters that plague CIM art. We have demonstrated the benefits of SC-in-Memory (SCIM) architectures over state-of-the-art in using machine learning (ML) accelerator IC prototypes in 65nm and 12nm CMOS.

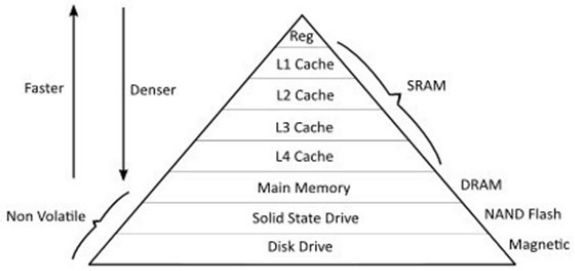
In terms of program metrics, for a 4-layer CNN, we achieved EDPs of 4uJ x 5.2us and 1.6uJ x 7.6us respectively for our SC and SCIM accelerator designs, both of which are better than the target for metric #3. Similarly, for VGGNET-18, we achieved EDPs of 2.8mJ x 3.4ms and 0.52mJ x 5.2ms for our SC and SCIM accelerator designs respectively, both of which are better than the target for metric #4. Given the complexity and the large size of these networks, please note that metrics #3 and #4 are demonstrated partly using measured hardware and partly using careful simulation as explained in the rest of this report.

Impact, Transitions, and Future: The MeRAM is a candidate memory for the future: even when integrated in an old 0.18um CMOS node, it has achieved performance comparable or better than STT-MRAM integrated in 28nm CMOS. The technology has been successfully reproduced in a semi-commercial foundry (ITRI) and we hope that other foundries will follow. An opportunity to integrate in a more advanced node may greatly help in the adoption. In addition, the high throughput, low energy TRNG capability developed should find use in many applications of commercial and military interest. The SCIM technology has been successfully applied, in collaboration with Northrop Grumman (as part of FRANC ERI-DA), to a problem of their interest – high speed object tracking. We except adoption in other applications that can benefit from the low latency and high area computational efficiency offered by this technology. The various techniques developed and accomplishments of this project have also been shared with researchers in the respective IEEE communities in the form of several publications.
The rest of this report elaborates on the technologies developed, the various experiments performed, and results obtained.

# 2 INTRODUCTION

## 2.1 Integrated VC-MTJ Memory Array

Thermal and power considerations have long prevented increasing the raw microprocessor frequencies. This necessitates parallelism at the micro-architectural level to keep up with computing demands. Such a multi-core approach is accompanied by a huge demand for high performance memory for cache, RAM and instruction storage. Furthermore, the rise of data intensive tasks such as Machine Learning and Artificial Intelligence is further increasing data storage demand. The conventional solution to achieve a memory that appears both large and fast is to use a memory hierarchy and caching, such as illustrated in Figure 1. Each level of hierarchy uses a type of memory that has characteristics that are most suitable to that level of hierarchy, which may include read access speed, write access speed, density, dynamic energy, and leakage energy. For example, SRAM is used for cache due to its quick access time, and DRAM is used for main memory as a good compromise between speed, energy, and density. However, conventional memory solutions face significant challenges in today's scaled technology nodes. SRAM, for example, has fast access speed but has large bit-cell size due to the 6T design and suffers from significant leakage and standby power as the technology node scales. eDRAM can allow dense memory design but requires a special process to have access to the high cell capacitance and low leakage access device that is necessary for a DRAM. At the same time, the requirement for periodic refresh can significantly increase energy consumption. Non-volatile memories (NVMs) can lower standby power by allowing power cycling. Some applications require non-volatility as a matter of necessity, such as instruction memory or storage class memory. One possible non-volatile embedded memory is eFlash. However, eFlash does not scale well beyond 28nm and requires several (10-15) extra masks which increases cost significantly.




| Technology | SRAM | DRAM | Flash |
|---|---|---|---|
| High Density | | ✓ | ✓✓ |
| Scalability | ✓ | | |
| Retention | | | ✓ |
| Performance | ✓✓ | | |
| Endurance | ✓✓ | ✓✓ | |
| Integration | ✓ | | |

**Figure 1 - Conventional Memory Hierarchy**

The increased demand for on-chip memory and the scaling challenges of existing memory solutions has prompted development of several emerging memory technologies, such as RRAM, PRAM, and MRAM. Some of these technologies and their properties are summarized in Figure 2.

Among these, the Spin-transfer-torque (STT)-MRAM has shown promise as it can allow for a compact non-volatile memory that is CMOS compatible and scalable. However, despite showing great density and access speeds when compared to RRAM and PRAM, the STT-MRAM relies on the Spin-Transfer-Torque effect for writing, which still requires a large current (~100-200uA) for a long time (~10-30ns) to be applied to the device. This results in a long and energy intensive write operation. Moreover, the small resistance of the STT-MTJ can be comparable to the access transistor's ON resistance, increasing the intrinsic read error rate of the STT-MRAM if a minimum sized access transistor is used.

| | Resistive | | Magnetic | |
|---|---|---|---|---|
| **Technology** | **PCRAM** | **ReRAM** | **STT-MRAM** | **MeRAM** |
| **Cell Size ($F^2$)** | 20-30 | 10-20 | 10-20 | 10 |
| **Write Time (ns)** | 50 - 100 | 10-100 | >5 | <1 |
| **Write Current (μA)** | 80-300 | 10-100 | >50 | 1-10 |
| **Write Voltage (V)** | 1.5-3 | 1-2.5 | ~1 | 0.8-1.8 |
| **Write Energy (pJ/bit)** | 20 | 0.1-2.5 | 0.3 | 0.01 |
| **Endurance** | $10^{5-9}$ | $10^{5-8}$ | $10^{15}$ | $10^{15}$ |
| **Maturity Level** | Commercial | Commercial | Commercial | Device Level, **No CMOS Prototype** |

**Technology in this work**

**Figure 2 - Emerging Memory Technologies and Their Properties**

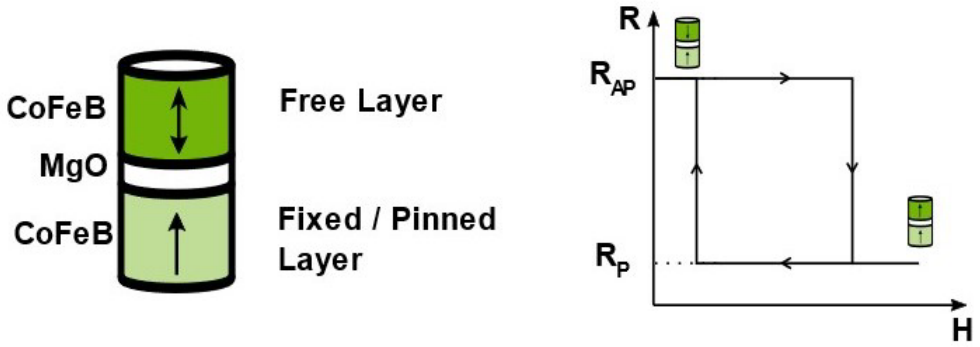



**Figure 3 - MTJ Structure and Resistance vs External Magnetic Field Curve**

In this work, we develop a promising new memory called the Voltage-Controlled (VC)-MRAM, also referred to as Magneto-Electric RAM (MeRAM). It is a novel memory candidate that can drastically improve the write performance and array density when compared to the STT-MRAM.

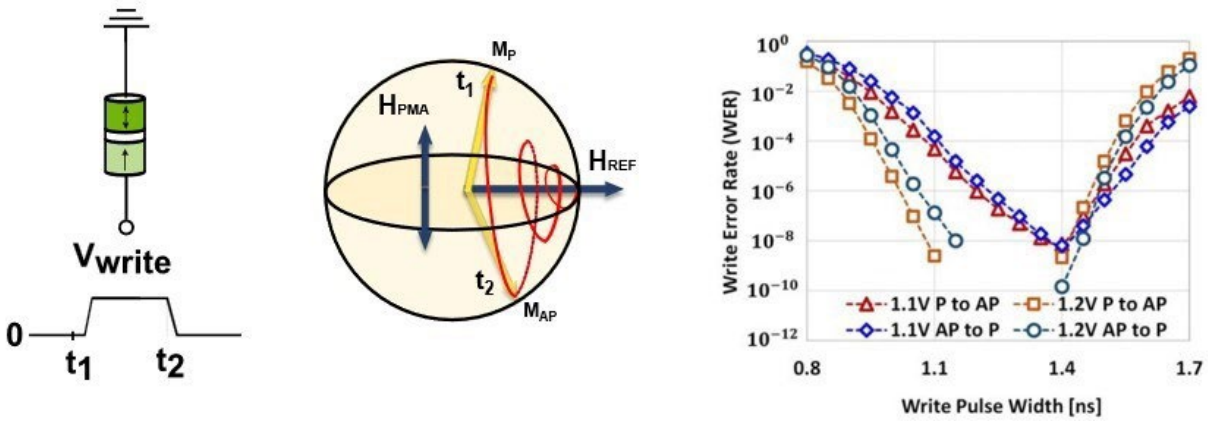



**Figure 4 - Illustration of the VCMA Write Mechanism**

This new memory uses the Voltage-Controlled (VC) Magnetic-Tunnel-Junction (MTJ) as the memory storage device. The VC-MTJ is composed of a free and fixed ferromagnetic layer sandwiching a thin MgO barrier, as shown in Figure 3. When the magnetizations of the two layers are parallel to each other, the device exhibits a low resistance ($R_P$) and when they are anti-parallel, the device exhibits a high resistance ($R_{AP}$). The Tunnel-Magneto-Resistance (TMR) is a measure of how different the two resistances are to each other (TMR = $R_{AP}$ - $R_P$ / $R_P$). This difference in resistance is what can be used to read the device.

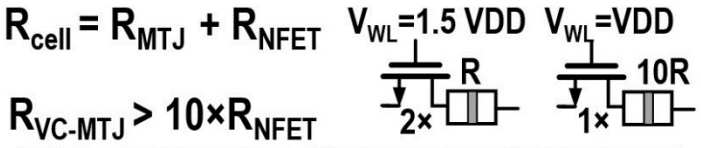
What makes the VC-MTJ unique, however, is its write mechanism. The Voltage-Controlled Magnetic anisotropy (VCMA) effect at the interface of free and barrier layer allows the voltage to modulate the perpendicular field (Figure 4). The applied write voltage can eliminate the perpendicular magnetic anisotropy (PMA) and results in magnetic moment precession along the external field direction. After a precession of half a period, voltage is turned off so that PMA recovers, and the magnetic field will now become stable in the opposite direction. The speed of such switching is sub-ns limited by the ferromagnetic resonance frequency.

A summary of the major differences between the VC-MTJ and the STT-MTJ are illustrated in Figure 5. The voltage-based writing mechanism and high resistance of VC-MTJ (larger than 10x of STT-MTJ) significantly reduces the current through the device during write and allows the access transistor to be minimal size. Furthermore, the unipolar write mechanism of the VC-MTJ makes it much more resilient to read disturbance. The VC-MTJ's higher resistance results in improved cell TMR for the VC-MTJ bit-cell compared to the STT-MTJ bit-cell. This is because for the STT-MTJ bit-cell, the access transistor resistance is comparable to that of the MTJ device, causing a degradation of TMR. Finally, both the writing time and energy for the VC-MTJ is much smaller than that of the STT-MTJ.

Thus, the use of VCMA is a promising avenue for the next generation of MRAM technology. Unlike conventional current-controlled switching mechanisms such as spin-transfer torque (STT), voltage control can significantly reduce switching voltage and current, and thus power

consumption. The power consumption can be scaled down to sub fJ, depending on the VCMA coefficient and the size of device, which is more than one order of magnitude better than currentbased STT switching technology.

In this work, we demonstrate the potential of the VC-MRAM. First, we develop a low power and fast read circuit optimized for single device read that can achieve a 2.4ns * 0.4fJ read EDP, which is better than the Metric 1 target. To have confidence in the results, we perform this simulation in 28nm with layout parasitic included using a compact Verilog-A model of the VC-MTJ device with parameters verified by real device measurements. Secondly, we demonstrate the feasibility of VCMRAM as a high-performance embedded memory by developing the first CMOS integrated VC- MRAM. In doing so, we make the following contributions (1) Integrate VCMRAM with CMOS technology through a foundry-and-lab collaboration; (2) Demonstrate an ultra-fast switching time of 0.7ns with 1.8V write voltage in a 8x8 and 80x80 memory array; (3) Study the variation of VCMTJ's probabilistic switching behavior in an integrated array and demonstrate 92% switching probability under an un-calibrated and uniform pulse duration; (4) Demonstrate VC-MRAM reliability of $>10^{11}$ cycles. The integration requires processing over the full wafer and the 180nm process is chosen for demonstration purposes due to its low cost and ready availability. 2.2




| | STT MTJ | VC MTJ |
|---|---|---|
| Density | 1× | 2× |
| $R_{MTJ}$ | 1× | 10× |
| Cell TMR Ratio | 1× | 1.6× |
| Write Time | 1× | 0.1× |
| Write Energy | 1× | 0.1× |

**Figure 5 - Differences between the STT-MTJ and the VC-MTJ**

## 2.2 Random Bit-Stream Generation with VC-MTJ

True random number generators (TRNG) are key components in cryptography and emerging computing applications such as stochastic computing. With the advent of quantum computing, many of the traditional cryptography algorithms may be impaired or broken in a reasonable number of tries. To prevent security problems in post-quantum cryptography, much longer keys are required. This requires the TRNG hardware to have higher throughput and lower energy costs. Previous works have demonstrated hardware random number generators in CMOS technology using inverter's metastability [7], jitter in a ring oscillator [8] and transistor's oxide breakdown [9].

Most of them require dedicated circuits with large area and power-consuming post processing circuits to remove bias.

One approach to save TRNG area is to generate random numbers in memory arrays that are already a part of the system. These so-called memory based TRNGs reduce the cost of energy and area by reusing the memory array to generate random numbers. Previous works have explored TRNG based on STT-MRAM [10], [11], which exploits metastability in a current controlled spin-transfertorque (STT) MRAM. However, the switching probability of the STT-TRNG is highly sensitive to the amplitude and duration of the critical current. Given inevitable device variability, extensive calibration may be required to find qualified devices. Besides, the STT MRAM suffers from large energy consumption and limited endurance due to large write current.

To overcome these issues, in this work we propose an in-memory TRNG using Voltage-Controlled MRAM that does not require calibration of the write pulse. It improves energy consumption and endurance by having 10×–50× larger resistance area (RA) product than STT-MRAM. Furthermore, a new bias correction digital circuit is proposed to ensure high speed and robust randomness under potential magnetic field interference. The VC-MTJ has the unique property of converging to metastability asymptotically without the requirement of any calibration, making it a perfect solution of generating true random numbers inside the memory. A comparison of the random number generation mechanism between STT-MRAM and VC-MRAM is shown in Figure 6. STTMRAM relies on a critical current that has a pre-determined amplitude and pulse width to achieve metastability. High resolution timing control of the write pulse for each device during the operation or calibration ahead of the operation is needed to achieve high entropy. In contrast, VC-MRAM does not require calibration before or during the operation. When a voltage is applied, the free layer's magnetization precesses along the in-plane axis. It oscillates between P and AP state and asymptotically converges to the in-plane axis due to damping effect. The longer the voltage pulse is, the closer it is aligned to the in-plane direction, which corresponds to 50% probability.

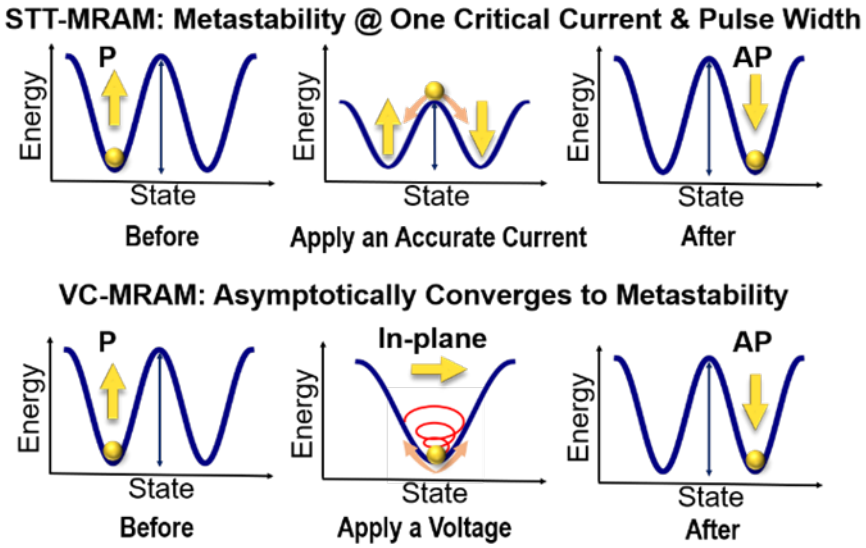



**Figure 6 - Comparison of the Random Number Generation Mechanism between STT and VC MRAM**

VC-MRAM makes the in-memory true random number generation of multiple rows possible when high throughput is required. The array arrangement of the memory makes only one row of the bitcells available for read or write at a time. For the STT-MRAM, since it requires calibration for each individual device inside the memory, the multi-row in-memory operation is not possible to achieve. However, since VC-MTJ does not need calibration, devices in multiple rows can generate random numbers at the same time. In a multi-row operation, several wordlines turn on together. The bitcells on the same bitline share the same write pulse. A longer pulse can make sure that most of the bitcells generate high-entropy random bits under device variations. A post processing circuit can remove any potential bias during read out. The multi-row in-memory RNG can significantly increase the throughput with no extra hardware cost.

In this work, we have developed two TRNG chips that employ VC-MTJ random number source. The first chip utilizes an external VC-MTJ array that is wire-bonded to the CMOS chip. The second chip employs the first CMOS-integrated VC-MRAM array that is discussed in Section 2.1.
Through some additions in the RTL only, the VC-MRAM can operate as a TRNG.

### 2.3 SC/SCIM ML Accelerators

Stochastic Computing (SC) is an emerging computing paradigm that represents number as the probability of ones in a random bit stream and uses tiny logic gates to achieve massive amount of parallelism. The computation is done in probability domain. Logic AND gate and OR gates are used as multiplier and approximate adder unit in SC. They are equivalent to intersection and union operation in probability domain. Due to the tiny footprint of SC, huge amount of SC logic can be packed on chip to accelerate computation-heavy applications such as deep learning. However, several challenges exist. 1) Binary to stochastic conversion consumes a significant amount of energy and large reuse factor can help amortize the average cost per operation. 2) OR-based accumulation is a nonlinear addition function and requires accurate modeling during neural network training to achieve comparable accuracy as fixed point. 3) SC stream length increases exponentially with number precision, which leads to increasing cost of energy and latency compared to fixed-point computation. We come up with several solutions to overcome those challenges and demonstrated several SC based accelerator hardware that met the energy efficiency and latency target set by this program.

First, we realized that reducing SC conversion costs is paramount - by reusing stochastic streams for both activations and weights, the cost of generating those streams is amortized. Similarly, by performing wide reduction operators in parallel, intermediate sums are removed which amortizes SCto-binary conversion. Both those facts severely limit available dataflow choices compared to conventional binary architectures. Figure 7 shows how per-MAC energy and area is reduced when both input and output reuse increases. It also shows two other important conclusions. First, “vanilla” SC computation can be at best as energy efficient as conventional binary. While that energy efficiency can be improved through e.g. computation skipping, SC energy savings generally don’t come directly from computation. Second, when taking advantage of data reuse, SC offers unrivaled computational density.

SC energy savings do not come directly from computation. However, system energy is often dominated by memory. Reducing energy of individual computation very quickly results in diminishing returns. Instead, the main goal should be to reduce the size of memory, both off- and on-chip. This can be also achieved by exploiting data reuse - the more a single value, weight or activation, can be reused

at the same time, the fewer times it has to be fetched from memory. Figure 8 shows how our proposed architecture reduces the number of required memory accesses, both to off-chip DRAM and internal, global SRAM. Both above conclusions show that exploiting data reuse is crucial for SC-based architectures. This is the main reason why we chose Neural Networks as our target application is that they contain ample reuse patterns for both activations and weights, and very wide reductions.

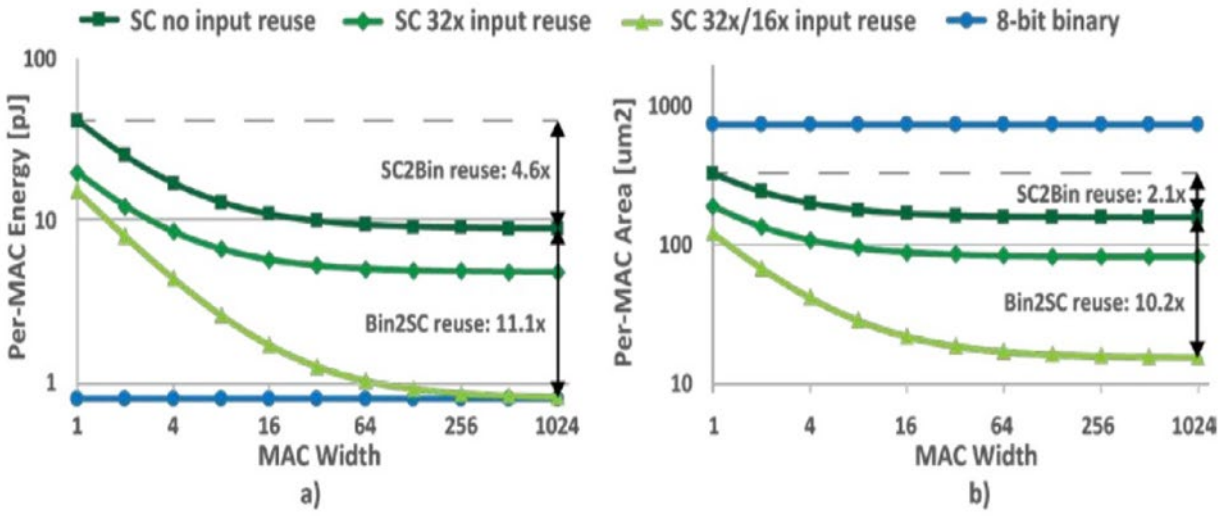



**Figure 7 - Per MAC Energy (a) and Area (b) for 8 bit Binary and SC with Different Amount of Input/Weight Reuse and MAC Width (output reuse)**

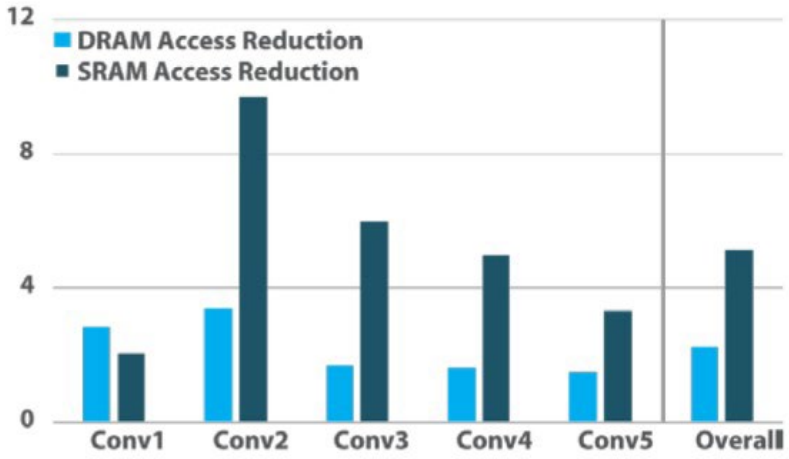



**Figure 8 - DRAM and SRAM Access Reduction of the Proposed SC Architecture vs Eyeriss, on AlexNet Convolutional Layers**

# 3 METHOD, ASSUMPTIONS, AND PROCEDURES

## 3.1 VC-MTJ Device Development and Integration

Improving the performance of the VC-MTJ material stack is crucial. The most significant parameters are PMA, VCMA coefficient, and the resistance-area product (RA). A heightened PMA translates to better data retention. As we scale down the MTJ size, an enhanced PMA is necessary to maintain the same retention time. However, this simultaneously makes switching harder. To counteract the increase in switching voltage, the VCMA coefficient needs enhancement. Note that the switching voltage is directly proportional to PMA/VCMA. Given an MTJ size, the RA determines its resistance, playing a pivotal role in both writing and reading processes. Note that the RA of VC-MTJ is 10x larger than that of STT-MTJ. While substantial research has been conducted on the RA and PMA in STT scenarios, these parameters require re-optimization for VCMTJ. Some studies have reported achieving high VCMA values exceeding 100 fJ/Vm; however, these are not CMOS-compatible, as illustrated in Figure 9 (a). Therefore, it's important to engineer a material stack that aligns with CMOS-backend compatibility.

Beyond the challenges in the material stack, there is a notable absence of reports on the CMOSintegration of VC-MTJ. This calls for a reevaluation of the fabrication methods and a search for ways to enhance the yield. Considering this, we partnered with ITRI to harness high VCMA from the available materials and ultimately achieved a commendably high yield.

### 3.1.1 VCMA Stack Development

In the FRANC phase I and II, we developed new film stacks with a high VCMA coefficient. We tried different materials and their combinations. For example, we used Ti, Ta, W, Ir, and Pt as an underlayer. To maintain PMA after 400℃ annealing, we used Mo as an insertion layer between the heavy metal and the ferromagnetic layer. Additionally, we also tried different ferromagnetic materials, like Co20Fe60B20, Co40Fe40B20, and Co50Fe50. We also investigated different insertion layers (Ir and Pt) between MgO and the ferromagnetic layer. We obtained a large VCMA coefficient, 370 fJ/Vm, with a 0.1-nm Ir insertion. However, due to the diffusion of Ir, after annealing (>200℃), the VCMA decreased rapidly to 50 fJ/Vm.

With our best efforts, we demonstrated a high VCMA coefficient under the CMOS integration condition. The full structure of this sample, shown in Figure 9(b), is as follows: Ir(5) / Mo(0.7) / CoFeB(0.9) / MgO, where the numbers in parentheses indicate the layer thicknesses in nm. We should also emphasize that the VCMA of 113 fJ/Vm is the highest VCMA coefficient in the world under CMOS-compatible conditions, as shown in Figure 9(a).

### 3.1.2 VCMA Stack Integration

In phase III, we transfer the VCMA stack developed in our lab to the foundry ITRI, which deposited the MTJ stack on the backend of CMOS wafer. However, ITRI does not have Ir target. Thus, we collaborated with ITRI to develop high VCMA within available material. Note that it is the first-ever CMOS-integrated VC-MTJ.

The VC-MRAM film layer stack, Ta / Mo / CoFeB (free layer) / MgO / CoFeB / W / Co / Ru / [Co/Pt]n / Capping, was deposited by sputtering on 8 inch 0.18um CMOS wafer at room temperature. The 1 nm Mo insertion improves thermal stress stability and makes MTJ compatible with BEOL processing. The deposited film was annealed at 360°C for 20 minutes. The critical MTJ pattern with diameter of 100 nm was defined by E-beam lithography. A closeup TEM image of the fabricated MTJ with clean sidewall is shown in Figure 10(a), and Figure 10(b) has a wider view showing the integrated MTJ together with the CMOS access transistor.

To characterize the VCMA coefficient, the stack with the same free layer but an in-plane reference layer was grown on the CMOS wafer. Effective PMA as a function of electric field [1] is shown in Figure 10 (d). VCMA coefficient is extracted by the slope of the plot as ~ 48 fJ/Vm, which is comparable with previous study [2] [3]. Note that the same stack grown on Si/SiO2 wafer shows higher VCMA coefficient ~ 65 fJ/Vm. The difference suggests high sensitivity of VCMA effect to the material interface, which is the major challenge of VCMA optimization. Recent studies achieve

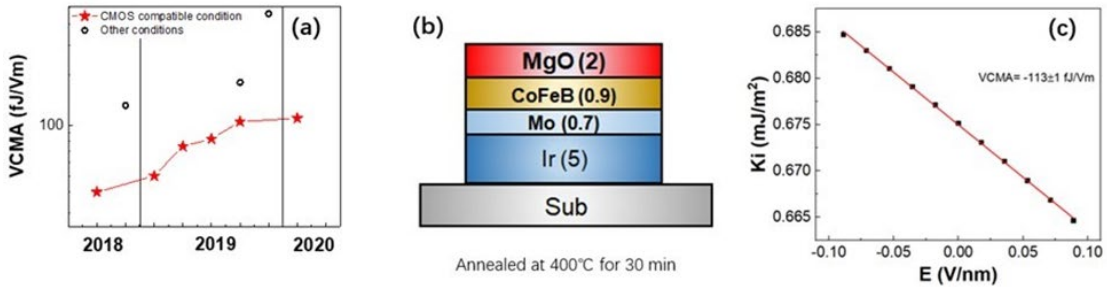



**Figure 9 - (a) VMCA Film Stack Development Evolution. (b) Schematic of the Highest VCMA Coefficient Film Stack. (c) VCMA Coefficient of the Stack shown in (b)**

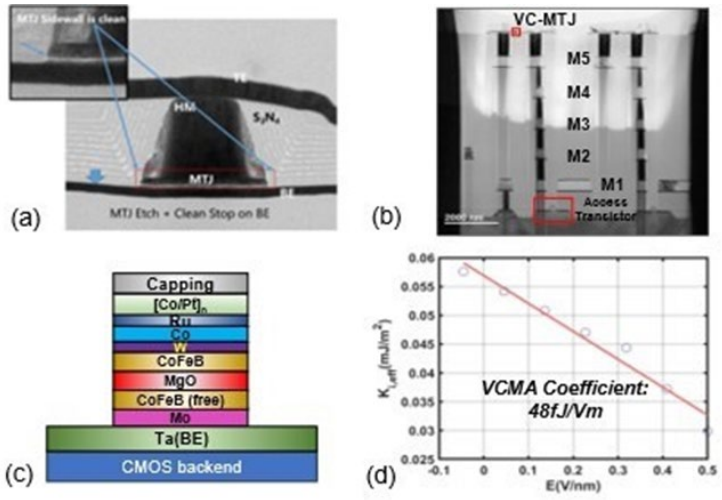



**Figure 10 - (a)TEM Image of the VCMTJ and Clean Side Walls; (b) CMOS Backend TEM Image of 1T-MTJ cell; c) Material Stack of the VC-MTJ; (d) Interfacial PMA vs Electric Field Plot of the VC-MTJ and the VCMA Coefficient is Extracted by the Slope of the Curve as 48 fJ/Vm**

VCMA >100 fJ/Vm by interfacial engineering on Si/SiO2 wafer with CMOS compatible condition [4], however, dedicated process optimization on CMOS wafer might still be essential. Furthermore, we fine-tuned the thickness of CoFeB and MgO. As a result, we reach the target RA (800 Ω*μm$^2$) and device resistance 100kΩ. Also, TMR 120% is obtained from film-level tests, which is comparable to the state-of-art MTJ.

### 3.2 180nm Memory Array Design

The VC-MRAM is being considered as an emerging memory due to its exceptional properties. However, the claims till now have only been based on theoretical and simulation arguments. Functionality of the VC-MRAM has never been proven on silicon which makes it unclear how the VC-MRAM will perform in actual hardware. To that end, in this work we have validated and demonstrated the world's first CMOS-integrated VC-MRAM.

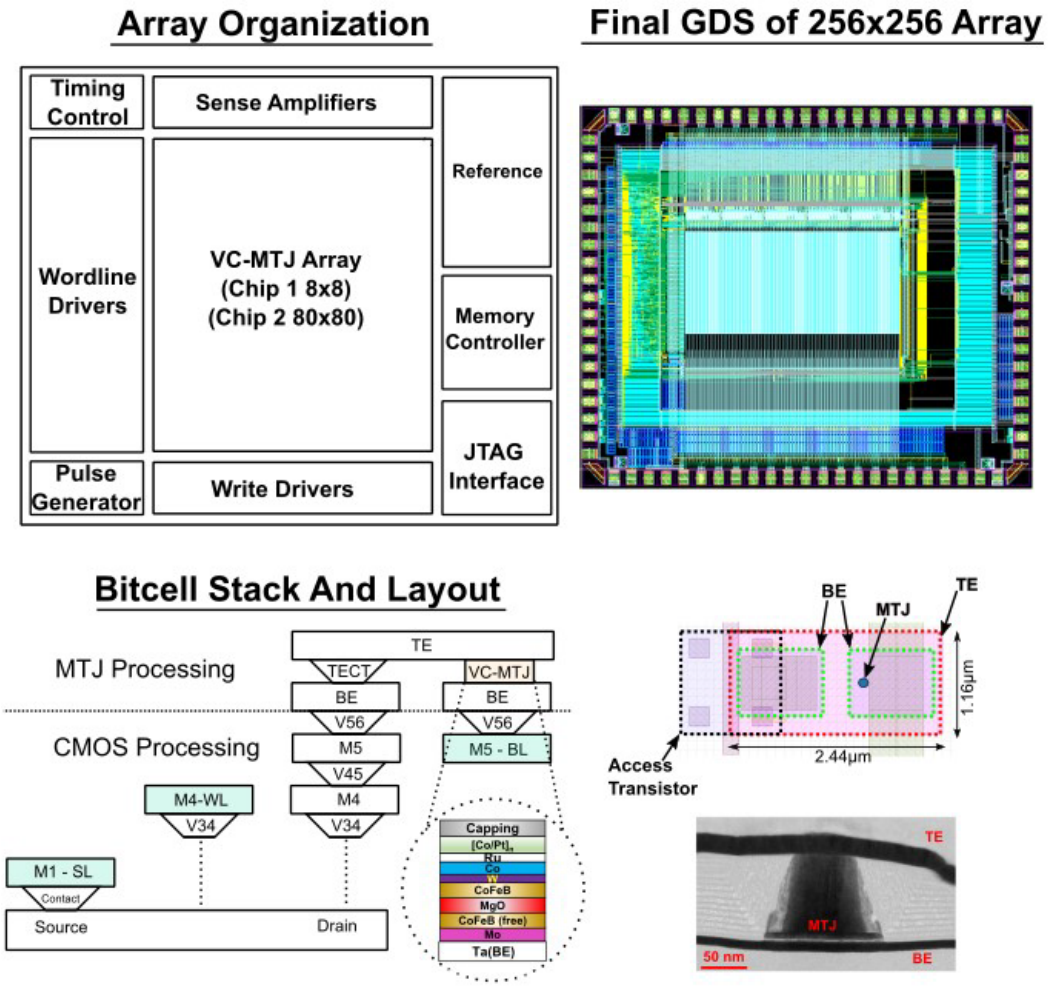



**Figure 11 - Details of Taped-out Chips - Array Organization, GDS, Bitcell Design, Device TEM Image**

The development of this very first VC-MRAM chip has been fraught with many challenges, both technical and logistical. The biggest logistical challenge stems from the fact that this is the very first CMOS-integrated VC-MRAM. As such, there is no commercial foundry that offers a VCMTJ cell. This challenge was resolved with a close collaboration between UCLA, ITRI, and TSMC. The design of the memory access circuitry was completed at UCLA and taped-out through TSMC. The material stack for the VC-MTJ developed by UCLA was provided to ITRI to deposit on top of the TSMC chips. Note that entire processed wafers are required for the subsequent MTJ fabrication. Given the very high cost of full wafers in advanced technology nodes, and given that ITRI's substantial experience with post processing of 180nm node rather than an advanced CMOS node, integration with a 180nm CMOS node was chosen to limit the risk of the entire exercise. This choice introduced some technical challenges due to the high capacitances and slow transistors associated with this old technology node. Another challenge was that ITRI did not have access to Iridium, a critical component of UCLA's VC-MTJ stack. This resulted in a lower VCMA coefficient. Finally, time and risk considerations did not allow integration of a nano-magnet with the VC-MTJ device. This significantly complicated the testing, which now had to be done in a magnetic field generator and the magnetic field's strength and angle became another variable that had to be carefully considered.

As for the technical challenges, the first concern again came from this being the very first CMOS integration attempt – the array level device properties were unknown. As a result, the array was designed keeping in mind a large amount of deviation from the targeted device properties. In doing so, we developed the best array we could while still ensuring that the array can handle a wide range of device variations. Figure 12 shows the targeted device properties, and the maximum and minimum variation that was considered for the designed.

| | Target | Lower Bound | Upper Bound |
|---|---|---|---|
| Rp (kΩ) | 100 | 25 | 400 |
| TMR (%) | 100 | 65 | 200 |
| Write Voltage (V) | 1 | 1 | 2.7 |
| Write Time (ns) | 1 | 0.7 | 2 |
| Rise Time (ps) | - | - | 100 |

**Figure 12 - Target Device and Upper/Lower Bound of Expected Device Variation**

Despite the numerous challenges, as described in Section 4, we have demonstrated a working memory array that uses the VCMA effect for writing and can write in less than 1ns. The results of this work not only guide the development of future VC-MRAMs, but it is also the first time that the array level device properties of a CMOS-integrated VC-MTJ array has been directly measured and reported. There were several different array sizes that were taped out to mitigate risk and allow for incremental testing. The smallest array is an 8x8 array with 3 variants that allow application of different write voltages, giving us the capability to test the circuit with a write voltage from 1.0V to 2.7V without stressing the CMOS devices. The larger 80x80 and 256x256 array are designed for

targeted device properties ($V_{write}$ = 1.0V-1.4V) but can be extended to higher write voltages by overdriving the transistors. Figure 11 shows the array organization, final GDS of the largest array, and the bit-cell cross section and layout of the VC-MTJ cell.

### 3.2.1 Read Circuit

The higher $R_P$ (and $R_{AP}$) of the VC-MTJ reduces read current (and power) but also leads to a long BL charging/discharging time constant. To ensure reasonable read time, a current-sensing topology is employed as shown in Figure 13. The clamping transistor, M2, provides isolation between the slow BL node and the much faster sense node (Q). This reduces the read operation's dependence on the slower BL time constant, which improves the read speed. Note that read disturbance is not a concern for the VC-MRAM since the read voltage is applied to the free layer and is of opposite polarity to write voltage, unlike STT-MRAM. This allows higher read voltages and potentially higher read margins. The three phases of the read circuit operation are shown in Figure 13: the BL is charged to $V_{read}$ in the precharge phase; a sense voltage develops at sense node, Q, in the sense phase; and the latched comparator compares Q and Qb to detect the MTJ state in the compare phase. The reference current can be generated by two parallel VC-MTJ devices in opposite states, or from an on-chip 12-bit resistor DAC for additional programmability and margin. A single reference column is shared among all the memory columns. The sense amplifier devices are sized to support a large current, in case the device resistance is lower than expected. This also helps to reduce offsets, and offset cancellation is not employed.

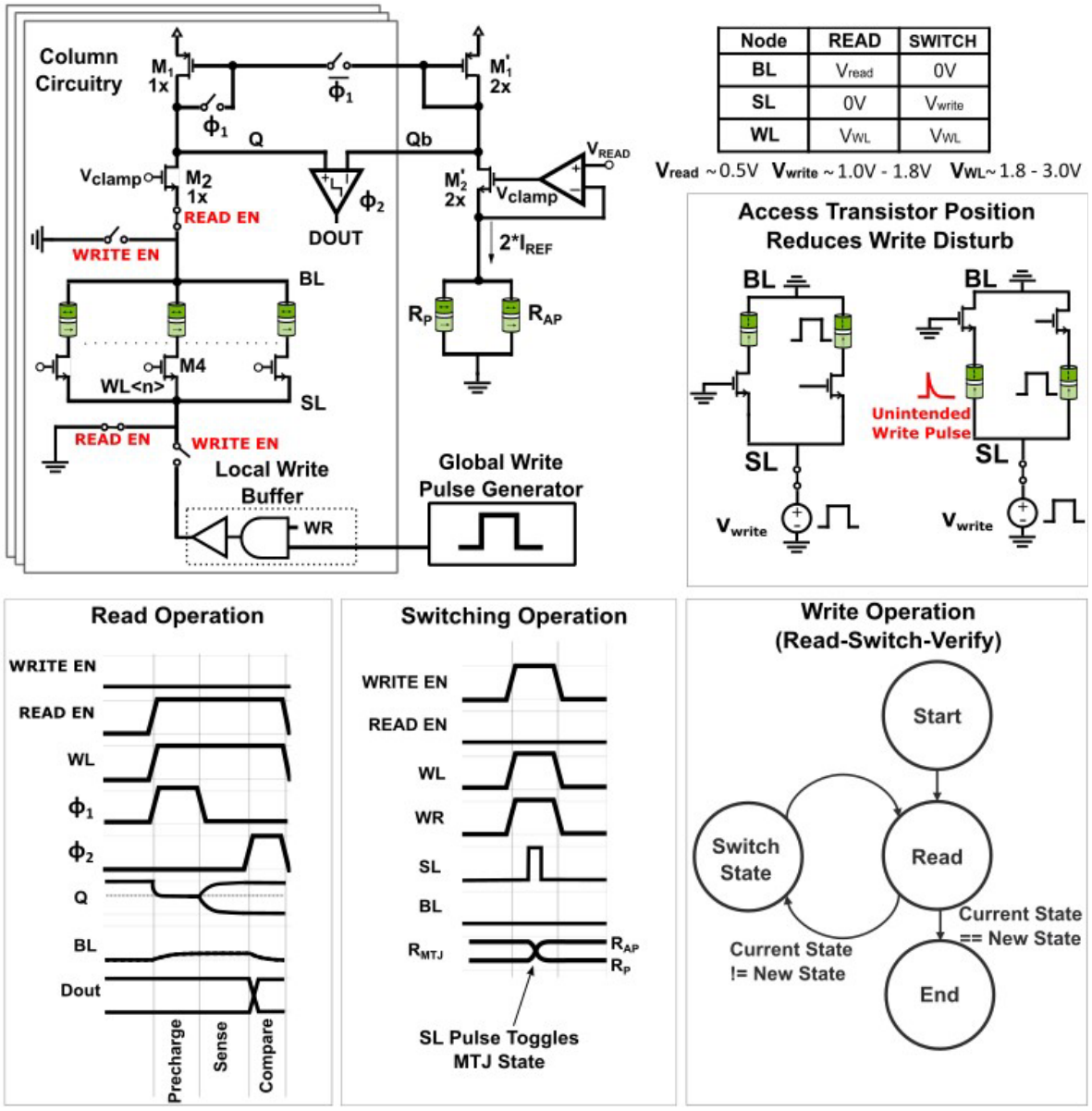
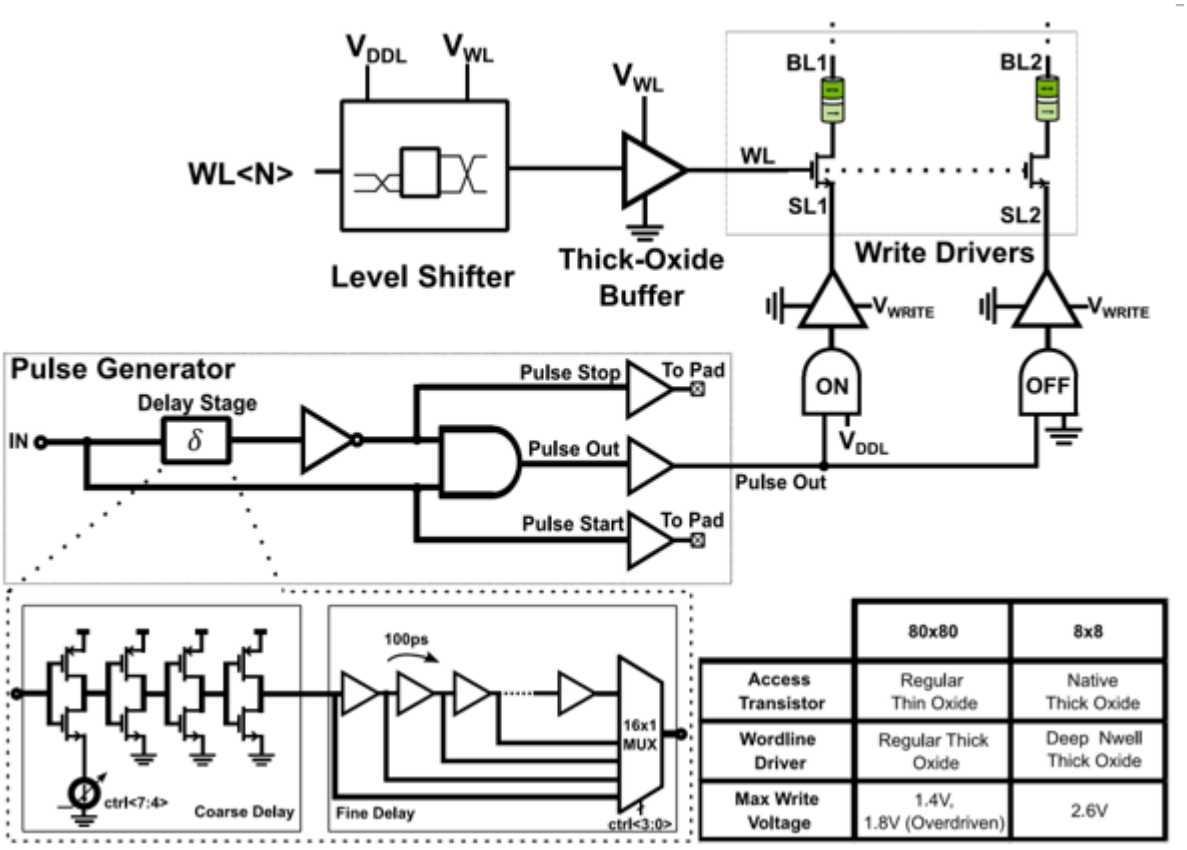



**Figure 13 - Simplified Schematic of Memory Column Waveforms of Read and Write Operation**

### 3.2.2 3.2.2 Write Circuit

The VC-MRAM's write operation is markedly different from conventional MRAMs as it requires a unipolar voltage pulse with a rise/fall time of <100ps (to initiate a strong precession of the magnetic field) and pulse width of <1ns. A 1T-1MTJ bit cell structure with an nMOS access transistor on the source line (SL) side was chosen to apply a positive write pulse on the SL. Alternatively, a pMOS access transistor or a BL side nMOS access transistor could have been used. However, the low pMOS mobility in 0.18um degrades write pulse rise/fall time; also, a negative Word Line (WL) voltage would be needed during read. The BL side nMOS access transistor could result in unintended write pulses on unselected bit-cells as show in Figure 13. These pulses are expected to be small and short but the consequences on the device are not well

understood. The chosen SL side nMOS access transistor limits the write voltage to ($V_{WL}$–$V_{TH}$) which is ~1.4V. Overdriving the WL enables up to ~1.8V write voltage. While this stresses devices on columns that are not being written, the stress is momentary (<1ns duration) and may be acceptable. For additional testability without any stress, the 8x8 chip employs native, thick-oxide access transistors. A global pulse generator produces the write pulse that is distributed to all the columns. Each column has local gating to control the write pulse and write buffers are used to sharpen the pulse to get a 100ps rise/fall time. The memory controller executes a read-then-switch operation while writing the VC-MRAM, which can be repeated up to 15 times to handle write failures. To measure the pulse width, delay matched 'pulse start' and 'pulse stop' signals are brought off the chip. Additional Details of the write circuit are shown in Figure 14.




| | 80x80 | 8x8 |
|---|---|---|
| Access Transistor | Regular Thin Oxide | Native Thick Oxide |
| Wordline Driver | Regular Thick Oxide | Deep Nwell Thick Oxide |
| Max Write Voltage | 1.4V, 1.8V (Overdriven) | 2.6V |

**Figure 14 – MRAM Write Circuit Deatils**

### 3.2.3 Device Characterization Circuit

Since this is the first integration of the VC-MTJ device in CMOS, it is essential to be able to characterize the device before executing memory operations to ensure optimal settings for the memory array. A high degree of additional programmability and individual control was added to each column and row to be able to mirror out current from each device for characterization. This allowed us to measure the RP, RAP, TMR, Hysteresis curves, and the variations in all these properties. The difference between these two modes is highlighted in simplified schematics of Figure 15.

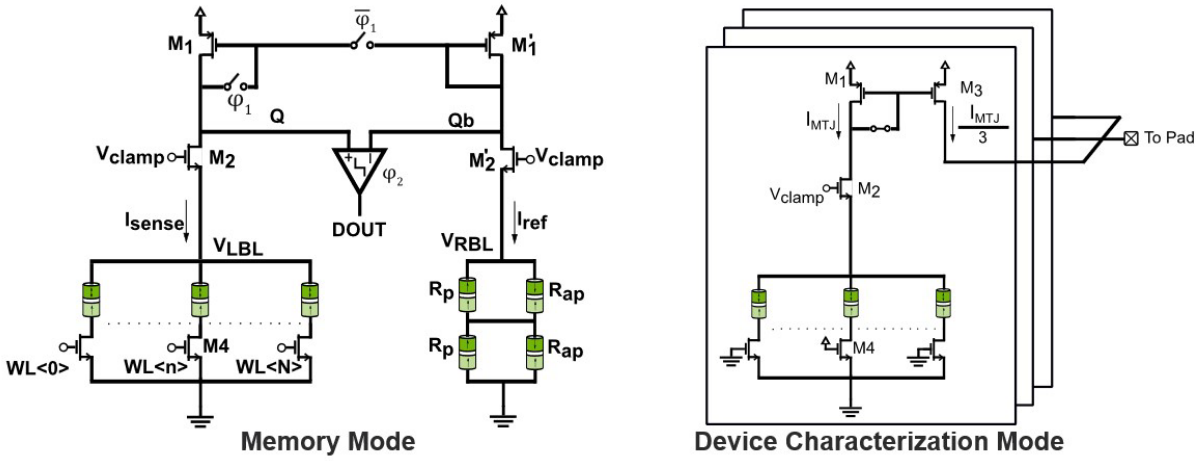



**Figure 15. Read Circuit in Memory Mode and Device Characterization Mode**

### 3.2.4 3.2.4 PCB Design

The design of the PCB was particularly challenging in this work. Due to the necessity of testing in a magnetic field generator, the size of the PCB was restricted to 6.5cm in one dimension. This resulted in a board with a long narrow shape, Figure 16. Secondly, three different array sizes were taped out. The PCB was therefore designed with three different IC footprints and was designed to be compatible with any array size. Finally, there are many different voltage domains on the chip to allow thorough testing at different read and write voltages. These voltages were regulated outside the chip. Due to the magnetic field generator, it was not feasible to bring the voltages from an external source, so the PCB had several LDOs on the board for regulation.

**Figure 16 - 3D Render of the Final MRAM PCB Design**

## 3.3 Low Energy, Fast Read Circuit

The high cell-resistance of the VC-MTJ device (>50 KΩ) can in principle permit a low-energy read operation. However, the larger time constants due to higher resistance may potentially lead to current conduction for a longer duration and would thereby nullify the energy benefit. This may mean longer read times and hence higher energy consumption. This introduces strict trade-offs between read array size, read speed and energy, latency, and column size. Any solution should introduce as small excess capacitance as possible on the critical nodes. Here we introduce a highly compact and low power 8T-1C sense-amplifier called the Magnetic-to-Digital Converter (MDC). Typically, sense-amplifiers seek to minimize the additional capacitance introduced on the shared bit-line. Such a read circuit needs to be very compact and consume low power. However, in finetechnology nodes (e.g., 28nm), large devices to keep Vth mismatches and other offsets to a

minimum. Figure 17 shows the 8T-1C implementation of the MDC based on a local offset cancellation scheme, thereby eliminating large devices, and hence significantly reducing additional capacitances at critical nodes. In the sensing phase ($\phi_{sense}$), $V_{ref}$ sets the read voltage at LBL to about 200 mV nominally and this generates the cell current $I_{cell} = I_P$ or $I_{AP}$ depending on whether the MTJ is in P or AP state. The wordline is asserted and $M_{p1}$ and $M_{p2}$ form a cascoded current source that pushes a reference current $I_{ref} = (I_P + I_{AP})/2$ into the sense node $V_{sense}$. Note that VC-MRAM, by virtue of its higher RA, exhibits comparable (if not larger) cell resistance to the $r_o$ of minimum sized FETs in 28nm in saturation, leading to a large cascode impedance of several 100 KΩ at $V_{sense}$. This translates to a large trans-resistance gain at $V_{sense}$. $I_{ref}$ and $I_{cell}$ are compared at $V_{sense}$ and a large voltage swing of ~400 mV is obtained. The swing at $V_{sense}$ far exceeds the variation range of the decision threshold of a subsequent gain stage, eliminating the need for a precise second stage. A simple minimum sized inverter suffices as the second stage to drive the decision to full rail logic levels. Furthermore, the lower read currents in VC-MRAM ensure that all devices remain in saturation during $\phi_{sense}$, ensuring reliable operation in a low VDD of 0.8 V.

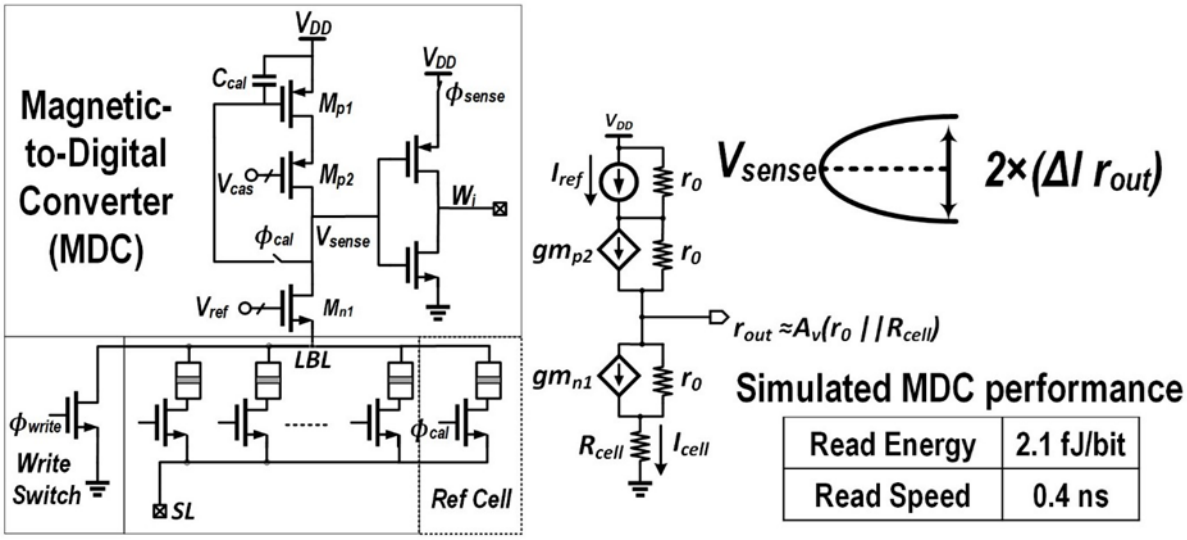



**Figure 17 - MDC Operation in Sensing Phase (sense)**

The accuracy of the first stage is key to achieving a low read-error-rate (RER). Previous works [5],[6] generate a precise reference current and mirror it to be compared with the cell current at $V_{sense}$. The $V_{th}$ mismatch in the mirror transistors as well as the clamping transistors in the two distinct current paths lead to errors in the sense current $\Delta I$. Mismatch is typically controlled by upsizing the current conducting devices. In contrast, the MDC generates $I_{ref}$ locally (as described below) by sampling a corresponding $V_{GS}$ on $C_{cal}$ during $\phi_{cal}$. The stored $V_{GS}$ is reused in the sense phase ($\phi_{sense}$) to compare with the cell current $I_{cell}$. Since $I_{ref}$ and $I_{cell}$ see the same current paths, the $V_{th}$ variations of the FETs in the sense path do not generate an error current, leading to accurate read. The once sampled reference can be reused for several reads before recalibrating. Furthermore, cascode FET $M_{p2}$ prevents coupling of swing on $V_{sense}$ to $C_{cal}$, allowing better calibration reuse. Post-layout SPICE simulations using 28nm CMOS and the VC-MTJ compact model indicate that a small $C_{cal}$ of 0.5 fF is enough to allow re-use for 8 reads. This calibration scheme cancels the circuit offsets and allows to use minimum sized FETs for all devices in the compact 8T-1C read circuit.

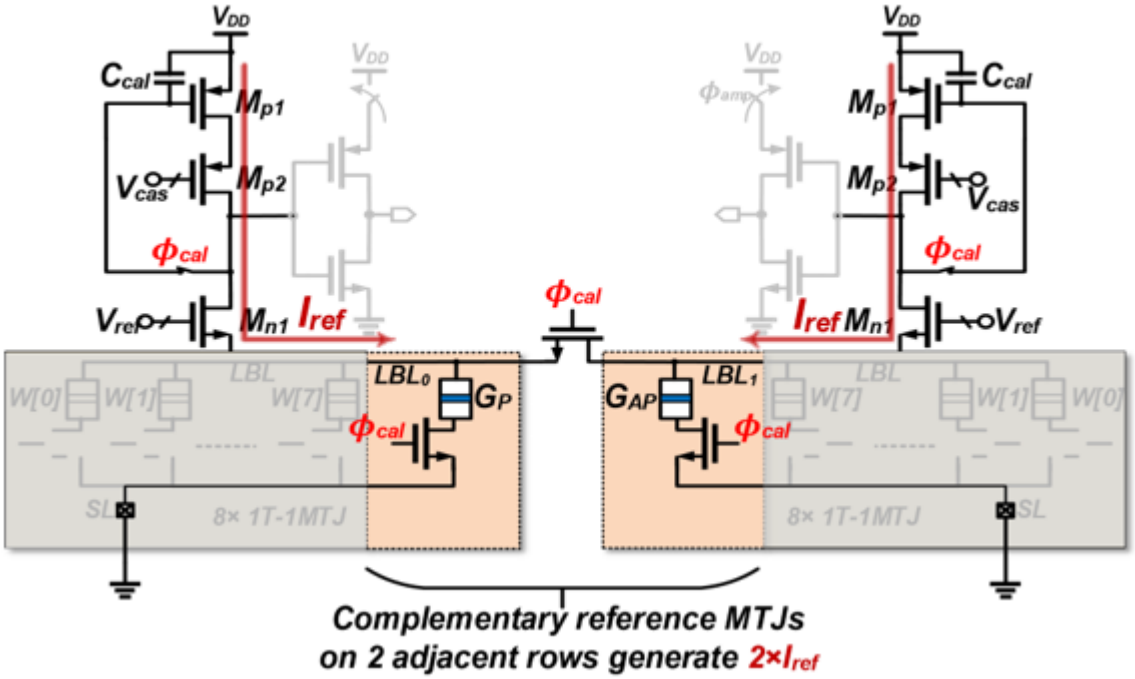



**Figure 18 – Read Circuit during Calibration Phase**

To generate an accurate $I_{ref}$ over PVT corners, one extra VC-MTJ is added in each row. Ideally, the reference VC-MTJ should present a conductance of $(G_P + G_{AP})/2$ during the calibration phase ($\phi_{cal}$) to maximize the sense-margin. This is practically implemented as shown in Figure 18 combining two MTJs: one in P state and the other in AP state. The reference MTJs of adjacent columns store these complementary states. The Bit Lines (LBL) of the adjacent columns are connected by a switch controlled by $\phi_{cal}$. During $\phi_{cal}$, adjacent LBLs are shorted and the two complementary reference VC-MTJs are connected in parallel and present an equivalent conductance of $(G_P + G_{AP})$. $V_{ref}$ sets the voltage across the VC-MTJs, and the current is provided by diode connected $M_{p1}$ within the two identical MDCs. The two MDCs share the generated current and each effectively see $I_{ref} = (I_P + I_{AP})/2$. Since the reference VC-MTJ is inside the local array, it closely tracks the on-chip variation and provides a reference current that maximizes the sensing margin. As will be shown in Section 4.2, the technique described here can achieve a EDP result better than the 1fJ×1ns target of Metric 1.

### 3.4 Random Bit-Stream Generation

VC-MTJ's special switching behavior makes it a good candidate for the applications of True Random Number Generator. We have two designs to validate the VC-MTJ based TRNG: 1) A 65nm integrated circuit with VC-MTJ access circuit and bias correction circuit, but VC-MTJ devices are located on another die and wire bonded with the chip. 2) A 180nm TRNG chip with integrated VC-MTJ devices. When a voltage is applied to the VC-MTJ, the free layer's magnetization precesses along the in-plane axis between the P and AP state. For memory operation, the pulse duration is set to half the precession period resulting in close to 100% switching probability. For RNG applications, a longer voltage pulse is applied causing the free layer magnetization to asymptotically converge to the in-plane axis due to damping effect. The longer the

voltage pulse is, the closer it is aligned to the in-plane direction, which corresponds to 50% probability. After the voltage pulse is removed, the free layer's magnetization is randomly switched to P or AP state under the influence of the thermal noise. The measured switching probability approaches 50% after the voltage pulse is applied in 3ns, as shown in Figure 20(a).

The 65nm test chip is wire bonded with the VC-MTJ's device die, therefore the bit line has a large parasitic capacitance and a high-speed current sense amplifier is required. Figure 21 shows the implementation details of the current sense amplifier (CSA) and write circuits. The conventional voltage-based sense amplifier compares the voltages between the MTJ's bit-line and a reference bit-line. However, the speed of sensing is limited by the large RC constant of the bittine parasitic capacitance and the MTJ resistance. The CSA can achieve faster speed because the small input impedance connected to the bit-line significantly lower the parallel resistance between MTJ and sense amplifier's input impedance. A feedback amplifier regulates the bit-line voltage and further reduces the SA's input impedance by the loop gain. The difference between the MTJ and reference's current is amplified at the read node V0 that has much smaller capacitance. A read access time of 12ns is achieved by the CSA.

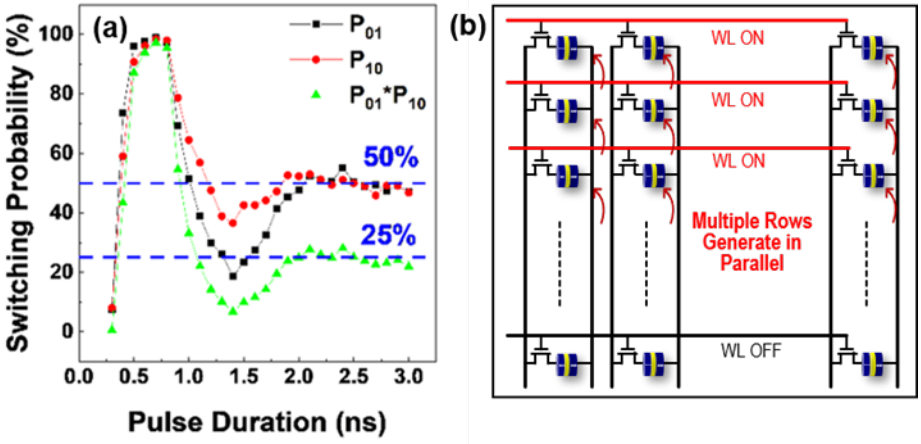



**Figure 19 - (a) VC-MTJ Switching Probability vs Pulse Duration (b) Multi-row Random Number Generation inside the Memory**

The 180nm TRNG chip has VC-MTJs integrated on the back end, therefore the access time is not significantly limited by the large parasitic capacitance of wire bonds as the 65nm design. The sense amplifier shares the same design as the 180nm integrated VC-MRAM chip discussed in the section 3.2.1. A comparison between the sense amplifiers for the 65nm and 180nm design is shown in Figure 20. The SA for the 65nm split design uses a feedback amplifier to reduce the input impedance, while the 180nm integrated design simplifies the SA by removing the feedback amplifier. Since the bit line capacitance in the 180nm design is much smaller, it has negligible impact on the sensing even though the feedback amplifier is not used. The simple design achieves read access time of 8.5nsec. We have also designed a sense amplifier to achieve ultra-fast read time for small VC-MTJ array, as discussed in section 3.3 and shown in Figure 20(right). The SA uses minimal transistors in 28nm and achieves only 0.4nsec read time for an array of 8 MTJs.

Although the ideal switching probability after the metastable state is 50%, multiple sources can cause a probability bias. For example, the stray field from the fixed layer may not be completely

compensated by the external field. Malicious attackers may also apply an interference magnetic field externally to disturb the TRNG. The effective residue bias field can cause the free layer to prefer a state, and thus causing a probability bias in the random number output. The bias correction circuit is shown in Figure 21. The core of the bias correction circuit is a Binary Weighted Stochastic Number Generator (BW-SNG). At every cycle, a raw random number is shifted into the buffer and produce one correction output from BW-SNG, therefore maintaining the input data rate. Every 8-bit raw random numbers are used as the select signals for a chain of 8 mutiplexers. The 0-selected input signal is the output from the previous multiplexer and the 1-selected input signal is an 8-bit binary number. The output of the BW-SNG is a stochastic bit stream with probability correlated with the 8-bit binary number, b. Assuming a stream of Independent and Identically Distributed (I.I.D.) the output probability of the BW-SNG's output bit stream is represented in Equation (1) in Figure 21. The probability tracking circuit counts the number of 1s in 256 output bits in real-time and compares the probability with 50%. Binary number b is added or reduced by 1 based on the comparison result until the output probability is close to 50%. The SGNO probability vs b under different RNN probabilities is shown in Figure 21(a). Starting from 0.5, the bias corrector will automatically find the closest solution of b that results in an unbiased stream. The bias after the correction is under 0.3% across 10%~50% input RRN bias range. Better bias-removing result of < 1e-5 can be achieved by using larger BW-SNG, as shown in yellow and blue curve in Figure 21(b).

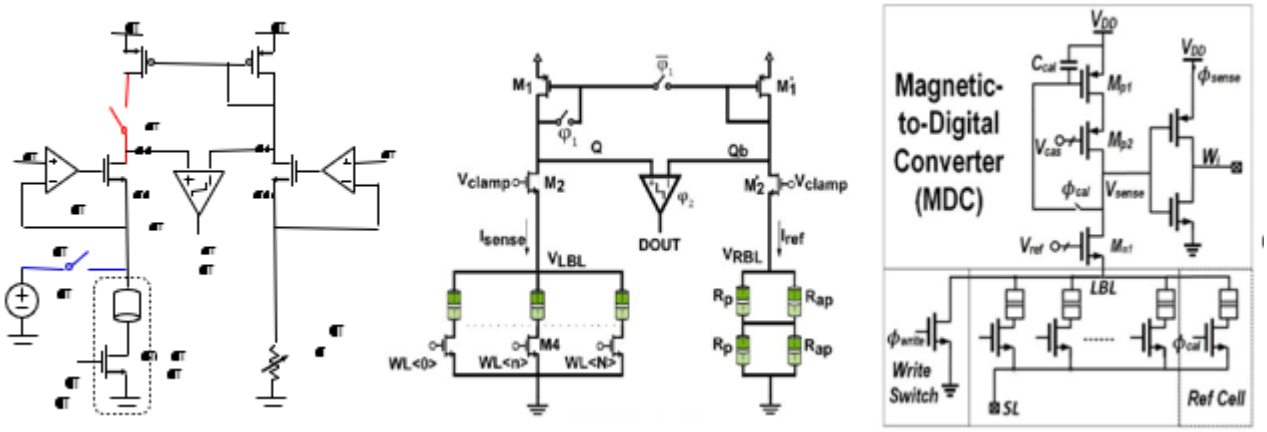
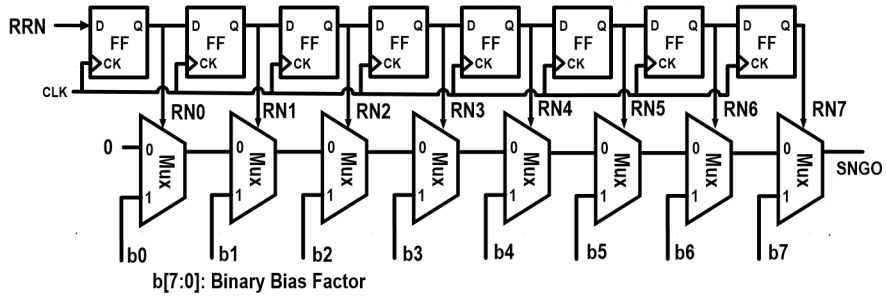
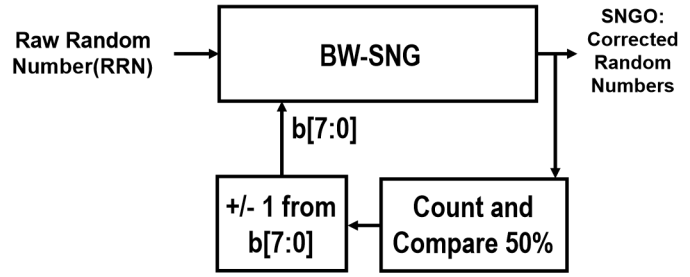
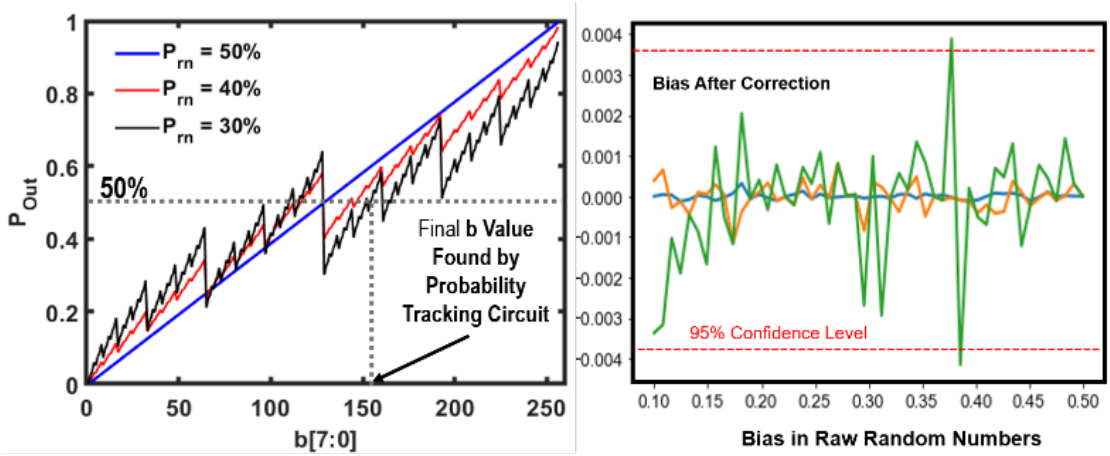



**Figure 20 - VC-MTJ Sense Amplifier for Three Different Designs**




**All-Digital Bias Correction Circuits**




$$P_{SNGO} = \sum_{n=1}^{8} b_{[8-n]} * P_{rrn}(1 - P_{rrn})^{n-1} \qquad (1)$$

**Figure 21 (a)Bias Correction Circuit (top) (b)PSNGO vs b Plot, Right (c) Bias after Correction using 8-bit(Green), 10-bit(Yellow) and 12-bit(Blue) BW-SNG, Equation.1. BW-SNG's Output Prob**

## 3.5 Stochastic Computing ML Accelerator

Stochastic Computing (SC) achieves massive parallelism by using tiny logic gates for computation. In this program, we have explored solutions to improve SC's computation accuracy, reducing the stream length and efficient hardware architecture to use SC to accelerate machine learning applications.

### 3.5.1 Stochastic Computing Split-Unipolar Representation

Stochastic computing offers two alternative number representation formats: unipolar and bipolar (Alternate representations have been proposed but have not been popular due to larger error [12] or larger area [13]). For the former, each bit in the stream has possibility v of being one, and for the latter the possibility is (v+1)/2, where v is the value being represented. In neural networks, maintaining high accuracy mandates using weights with both positive and negative values, which makes bipolar representation the most common choice when implementing SC-based accelerators [14, 15, 16]. However, unipolar requires at least 2X shorter streams than bipolar for same representational error. RMS error of unipolar and bipolar SC representations can be calculated as $\frac{\sqrt{nv(1-v)}}{n} = \sqrt{\frac{v(1-v)}{n}}$ and $\sqrt{\frac{1-v^2}{nb}}$ respectively, where n is the length of the bit stream.

We develop a split-unipolar representation which uses two streams to represent each weight, one for the positive and one for the negative component. For a positive weight value, its corresponding negative stream is 0, and vice-versa. Because activations (inputs) of a neural network layer are typically non-negative due to ReLU activation function in the previous layer, they can be represented using only a single positive stream. The activation streams are multiplied and accumulated separately with positive and negative weight components using up counters (in ACOUSTIC architecture, as described later, layer activations are converted to binary and stored in memory), whose values are then subtracted from each other to obtain final result. Since the counter output is in fixed-point binary domain, ReLU activation is easily implemented as a bitwise AND of the inverted sign with every other bit (Other activation functions require FSM implementations [12, 15] and we do not explore them here.).

Split-unipolar computation can be realized in hardware using temporal unrolling with just a single MAC array, where computation is done in two phases. In the first phase, all negative weights, and by extension their respective multipliers, are gated using their sign. This means only results corresponding to positive weights are being accumulated and the output counters are counting up. In the second phase, the mask is inverted, only negative weights contribute to the outputs, and counters count down. A simple example of a 2-wide MAC with one positive and one negative weight and stream length of 8 is shown on Figure 1. Spatial unrolling (similar idea has been recently independently proposed by [16]) simply doubles the compute arrays (i.e., 50% utilization) but we do not further explore as our target is resource-constrained devices. Split unipolar SC results in overall smaller energy/latency due to > 2x shorter bitstreams 3 and allows for more accurate OR-based accumulation described next.

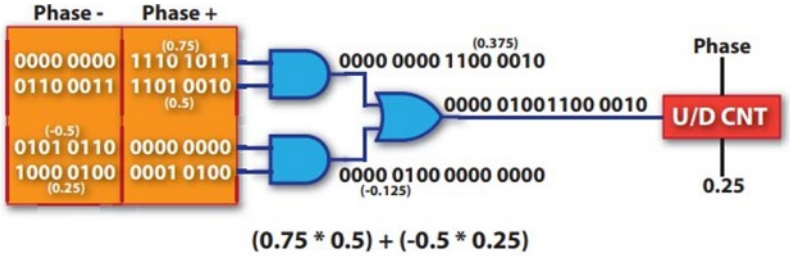



**Figure 22 - Circuit Level Support for Split Unipolar Representation**

### 3.5.2 Mux vs OR Accumulation

One of the main selling points of SC is that computation can be performed using bit-wise operations between two input bit streams. An AND gate performs multiplication: AND(v_1,v_2 )=v_1×v_2, where v_1, v_2 are the input possibilities for two unipolar streams. Similarly, a 2:1 multiplexer (MUX) can be used to compute MUX(v_1,v_2,s)=s×v_1+(1-s)×v_2, i.e., scaled addition between two input streams, where s is the selected input. Thus, MUX can act as a stochastic adder by using a 50% random stream at the select input. However, MUX-based addition degrades accuracy of computation (primarily due to the scaling factor), especially when wide accumulation is performed. Since neural networks generally perform very large matrix multiplications, prior work in SC-based neural network acceleration was often forced to perform accumulation in the binary domain, by either using costly parallel counter [12] or converting the results back to the binary domain after every multiplication [17].

We used OR-based stochastic accumulation [18]. It is scaling-free (important for very wide accumulations in deep CNNS), has reasonable accuracy for unipolar streams (a monte-carlo analysis of 3×3×256=2304 wide accumulation reveals OR having 8x less absolute error than MUX-based accumulation) and is also much more compact (4.2x than [12] and 23.8x than [18] for a 128-wide accumulate) than alternative accumulation methods. However, for a two-input OR, the result is equal to v_1+v_2-v_1 v_2 instead of v_1+v_2. We show later how we address this imperfect accumulation in training of the networks.

### 3.5.3 Pseudo Random Number Generator

RNG Sharing has been shown to be detrimental to stochastic computing accuracy [19], [20], and typically requires complicated methods to decorrelate streams from the same source to avoid incurring large stream generation penalties. However, we hypothesize that a partially-shared generation leads to higher accuracy, especially when coupled with deterministic stream generation and stream-based training.

Deterministic and repeatable (using a pseudorandom RNG) stream generators guarantees obtaining the same outputs from the same inputs, enabling the model to train for a fixed, instead of random error. We achieve determinisim using maximal-length linear feedback shift registers (LFSR) as RNG. When generating streams of length 2^n, an n-bit maximal-length LFSR is used with a cycle of 2^n-1. Apart from guaranteeing an almost accurate generation, LFSR generates the same output with the same input and seed and allows multiple uncorrelated stream generation (by varying the seed of the characteristic polynomial) suitable for large multiply-accumulate operations. Sharing stream generation simplifies the error profile caused by SC. Assuming that all kernels in a layer share the same set of seeds, training only needs to deal with an error associated with one set of seeds.

To test this hypothesis, we implement three levels of sharing for a 4-layer CNN [21] on the SVHN dataset. Streams are represented using the split-unipolar format, and OR is used for accumulation. In the “no sharing” case, each SNG gets a different seed for its LFSR. The “moderate sharing” case shares the same set of seeds across all kernels in a given layer. Finally, in the “extreme sharing” case, all rows of all kernels in a layer use the same set of seeds. The same is done when a true random number generator (TRNG) is used ans an RNG (Due to the lack of

hardware TRNG, we approximate it using the rand function in PyTorch). The results are shown in Figure 24. At moderate sharing levels, LFSR-based SNGs show a significant uplift in the accuracy (up to 6.1% points compared to unshared TRNGs) at both stream lengths, adhering to the hypothesis. TRNG does not see the accuracy improvement with sharing due to the lack of determinism. However, both TRNG and LFSR suffer from a significant drop in accuracy when using extreme sharing. In this case, stream correlation becomes an issue hard to overcome just by training.

These results also mean that low discrepancy (LD) sequences are not suitable for OR accumulation due to the difficulty of generating multiple uncorrelated streams, even though LD sequences can improve accuracy for single operations [23]. We also compared the validation accuracy when using LFSR without modeling it during training. The models are trained using TRNG, but validated using LFSR. No accuracy can be gained from moderate sharing when the model is not trained for it, and extreme sharing reduces accuracy to about 20%. We use the moderate sharing scheme in GEO (up to the limit of availability of unique RNG seeds).

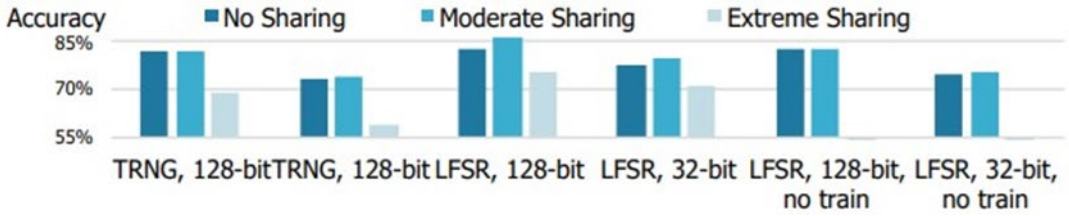



**Figure 23 - Accuracy vs. sharing for TRNG and LFSR based RNG**

### 3.5.4 Stochastic Compute Training

Approximate computing methods such as stochastic computing (SC) can benefit from specialized training. SC is random by nature and SC accumulation with bit-wise adders does not perform accurate accumulation [22]. Based on our tests, we found that comparing the performance of a model trained for fixed point to using a model trained specifically for SC using an accurate model showed up to a 57% improvement in model accuracy.

Since we used linear feedback shift registers for stream generation, AND for multiplication, and OR for addition all of this needed to be modelled in training. We also used 32-bit split unipolar streams for a total of 64 bits processed. For SC, the OR adder performs an $a + b - ab$ addition on average. Accurate addition is much simpler in the backward pass and the nonlinearity of the approximate addition can hinder convergence if it is not taken into consideration. We used an activation proxy added during training in the backward pass, but not added in inference. Training does not converge without the activation function.

Using an activation function requires the computation to be mostly associative. Computation is broken up into positive and negative parts since both work on unipolar inputs. Accumulation within each part is associative, but subtracting the two parts isn't. The activation used was $SC_{act(x)} = (1 - e^{-x_{pos}}) - (1 - e^{-x_{neg}})$. Additionally, there was benefit to adding error injection and fine-tuning.

| Setup | TinyConv | Resnet-tiny |
|---|---|---|
| Stochastic Computing | | |
| No Activation | 41.64% | 10.00% |
| With Activation | 72.18% | 79.76% |

| | TinyConv | | | | Resnet-tiny | | | |
|---|---|---|---|---|---|---|---|---|
| Method | Inference Only | With Model | Error Injection | Fine-tuning | Inference Only | With Model | Error Injection | Fine-tuning |
| Stochastic Computing | 14.87% | 72.18% | 64.01% | 73.09% | 48.57% | 79.76% | 76.71% | 81.29% |

**Figure 24. Accuracy Benefits of using Activation Function (top); Accuracy Impact of Error Injection (bottom)**

### 3.5.5 Digital Stochastic Computing Acceleration

The block diagram of our SC accelerator is shown in Figure 25. The accelerator consists of control logic, an activation scratchpad coupled with buffers and stochastic number generators (SNGs), and six multiply–accumulate (MAC) block rows, each with its weight memory (total 131 kB), and output counters, pooling, and activation logic. Control consists of instruction memory (4 kB), a dispatcher, and distributed control units. The activation scratchpad is organized as a pingpong buffer (two banks of 16 kB). SNG buffers are organized as shift registers that emulate a vertically sliding convolutional window for improved activation and weight temporal reuse. After conversion into stochastic streams, activation broadcast is used across all six MAC rows, enabling a high level of spatial reuse. All MAC rows use the same activations, but each can compute a single output channel in a convolutional layer, or three rows can be coupled to compute one output channel in a fully connected layer. Values are stored in memory using a fixed-point format and converted into stochastic streams only for computation, meaning that variable-precision support is transparent to memory and control logic. Our architecture can be easily scaled up to a larger area to increase performance. For example, the number of MAC rows, or the size and number of MAC columns can be increased to handle more concurrent filters or larger inputs. Likewise, the on-chip memory capacity can be increased to enable larger models. Because of the extensive memory data reuse of our accelerator, we expect the efficiency to be maintained when the design is scaled up, within the constraints of edge-class devices. Server-class inference and training architectures are beyond the scope of this work.

Convolution mapping is shown in Figure 26. We omit the fully connected layer explanation due to a lack of space. Each MAC block row is organized into five columns, each corresponding to one input and filter row across four or eight input channels. Fixed-point input and weight rows are loaded into the SNG buffers of its respective column. This way, a complete 5×5 filter can be computed in parallel, generating one complete row of outputs per MAC block row. Both sets of

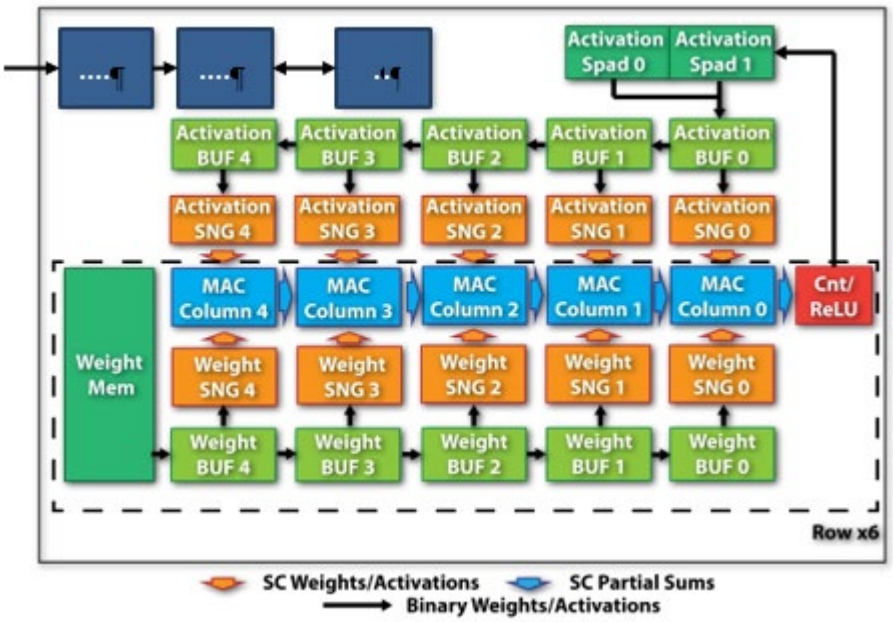



**Figure 25 - SC Deep Learning Accelerator Architecture**

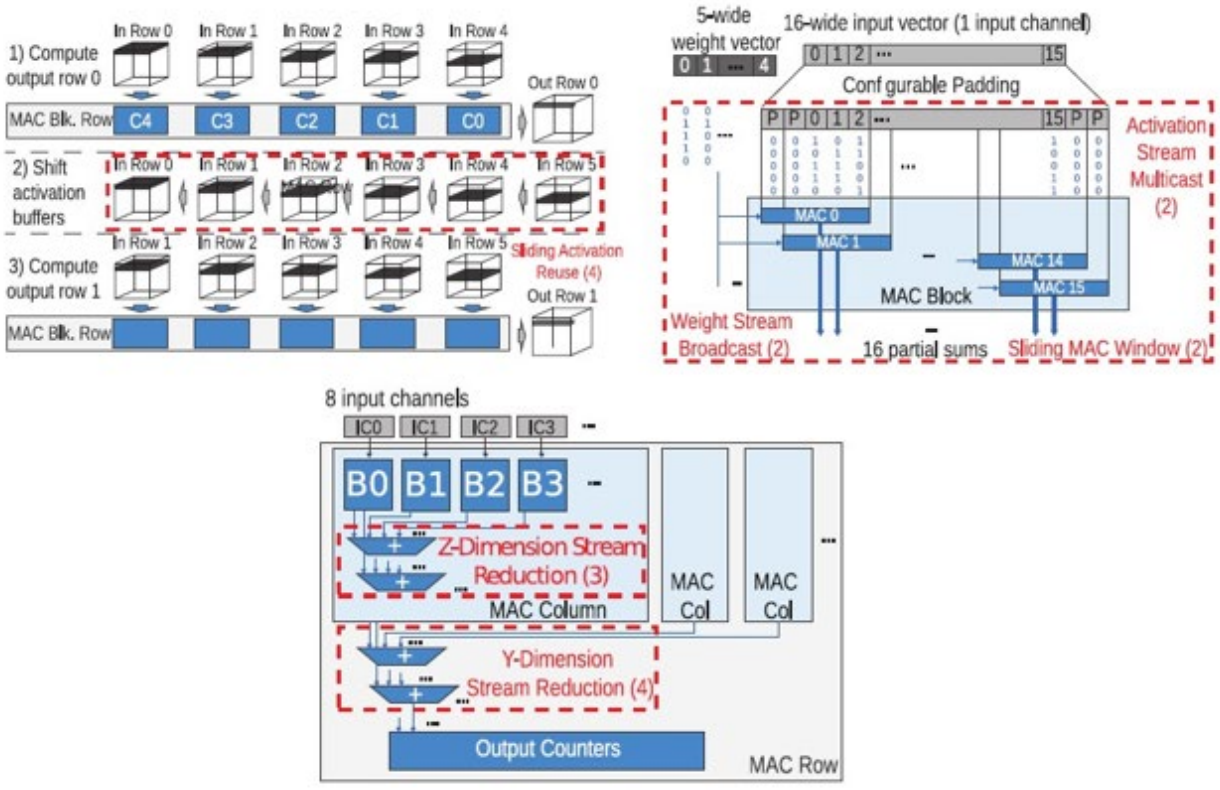



**Figure 26 - Mapping of Convolutional Layers in MAC rows, and Memory/stream Generation Amortization through Data Reuse**

*Shift register organization emulating sliding window left), MAC block with input/weight reuse and padding support (right), and block organization.*

SNG buffers are organized as shift registers, meaning that after one iteration, a new input row is shifted in, and the subsequent output row can be computed. This sliding activation reuse behavior emulates a vertically-sliding convolutional window, reducing the number of required memory accesses.

Each column is organized into eight SC MAC blocks, each corresponding to one input channel, operating in a fully streaming manner. A block takes 16 inputs and 5 weights and implements a sliding MAC window, generating up to 16 partial output streams depending on padding. Activation stream multicast with overlap is used between successive dot products, and weight stream broadcast is used for all dot products in a block, amortizing stream generation cost. Every two neighboring blocks can be coupled to support wider input rows. Corresponding output streams from all eight blocks are then reduced (Z-dimension stream reduction), after which corresponding outputs from all five columns are reduced (Y-dimension stream reduction). This hierarchical, SC reduction enables wide dot products (up to 200-wide) to be unrolled spatially, amortizing the cost of converting stochastic streams to fixed-point outputs. Our reuse-oriented design choices make it possible to perform over 130 MACs per memory access, and over 40 MACs per stream generation, as shown in Figure 26, lowering the relative energy cost of those operations. This high level of data reuse was only possible by using very dense SC computation—fixed-point designs have an order of magnitude lower memory access reuse.

Since computation operates purely on stochastic streams, it can support arbitrary precision by adjusting SC stream length, only limited by the width of the output counters. Precision selection is possible on sublayer granularity, e.g., groups of filters. Our custom instruction set enables selective gating of MAC block rows, columns, and blocks to support smaller activation or filter sizes. Filters or inputs that cannot be fully unrolled using existing resources are computed sequentially. A set of hardware loops with multi-stride memory accesses enables a variety of layer shapes and dataflow choices. Our accelerator can therefore support end-to-end inference with different types and shapes of layers.

### 3.5.6 Extending Range by Multi-Bit OR Accumulation

SC is an approximate computing method that can introduce random errors during both conversion and computation. Therefore, one of the chief concerns when designing SC systems is their accuracy. While extensive efforts have been directed at reducing the error of both stream generation and multiplication they have done little to explore and improve the accuracy of addition operators. The use of multiplexers (MUX) has been largely abandoned for addition because of their scaling factors equal the sizes of accumulation. OR-based adders, while not troubled by scaling factors, come with their own set of issues.

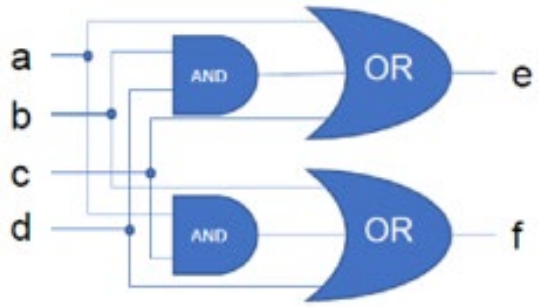



**Figure 27 – Efficient Implementation of OR-2 Circuits**

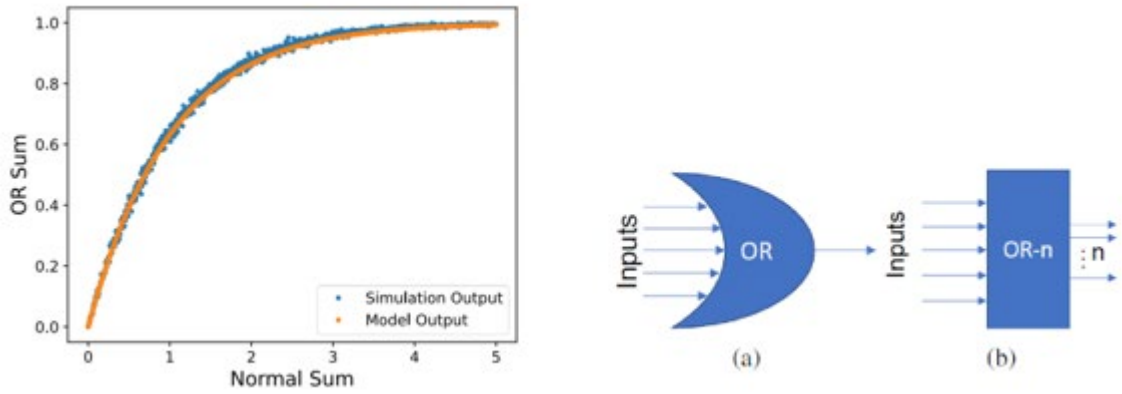



**Figure 28 – Concept of OR and OR-n; OR-1 Sum vs Accurate Sum**

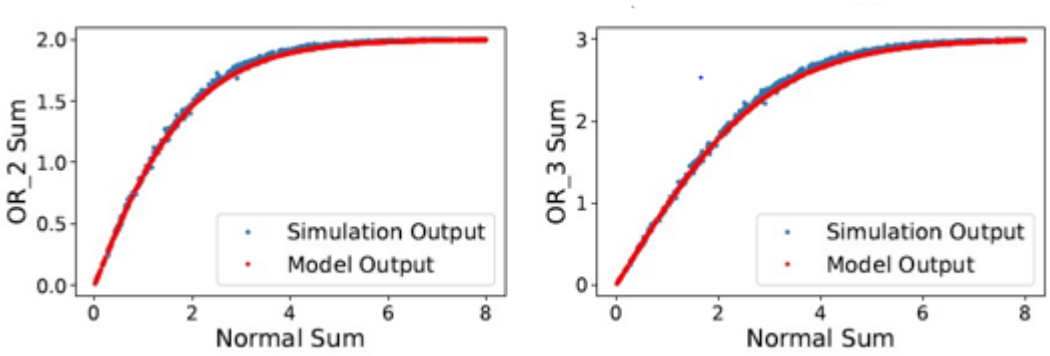



**Figure 29 – OR-N Accumulation Output vs. Accurate Sum**

First, as mentioned before, they do not implement exact addition. For two inputs a and b, an OR gate performs a + b − ab compared to a + b in an exact addition, which leads to output saturation for high-magnitude inputs. The transfer function or the OR-based accumulation is shown in Figure 28 . It is a linear function only for small values, while becoming nonlinear for large values. While the saturation of OR-based adders can be alleviated with small input values, small input values increase relative error. Second, the outputs of OR accumulation have the same precision as the inputs due to the bit-wise computation, which reduces accuracy compared to accumulation without truncation. Recall that adding two N-bit fixed-point numbers requires N+1 bits to avoid overflow.

Prior works have shown that algorithms can be trained for approximate addition and saturation for the first and second issue, for example, by using custom neural network training. Unfortunately, they cannot correct for the loss in intermediate output precision. Other previous works have tried to combine the OR accumulation and fixed-point accumulation to achieve a better trade-off between the two. While it does improve accuracy, combining the two dramatically slows down the training process. In contrast, using various forms of fixed-point accumulation achieves accurate or near-accurate summation between inputs, making them the most common choice of addition implementation in recent SC works.

To improve the accumulation accuracy and potentially reduce the stream length, we have come up with multi-bit OR accumulation. Normal OR operation performs a union operation and generates one bit output. The multi-bit OR operation, denoted as OR-N, generated N bits in order to extend the output range. The transfer function of the OR-N logic is shown in Equation (4). Two inputs $A_1$ and $A_2$ are N-bit stochastic stream.

The output $A_o$ is the accurate sum of the $A_1 + A_2$ if the accurate sum is less than N. If the accurate sum is larger than N, $A_o$ is N. The OR-N output vs accurate sum is plotted for OR-2 and OR-3 accumulation in Figure 29. As the number of output bits increases, the OR accumulation's transfer function becomes more linear and has a larger range. The efficient circuit implementation of the OR-2 circuit is shown in Figure 28. We have designed and evaluated the performance of accelerators based on OR-2 and OR-3, which will be shown in the result sections.

### 3.5.7 Stochastic Compute-In-Memory ML Accelerator

We have also explored combining the benefits of Compute-in-Memory (CIM) architecture with SC to further reduce the data movement cost. Figure 30 shows an overview of the CNN processor based on the concept of Stochastic-Compute-In-Memory (SCIM). It has 32 SCIM macros that perform convolution between activations and weights in Stochastic Computing (SC) domain. The convolution results are converted to fixed-point domain by the parallel counters. ReLU, max pooling and batch normalization are performed in fixed-point. The layer outputs are stored in the output SRAM until the next layer starts processing.

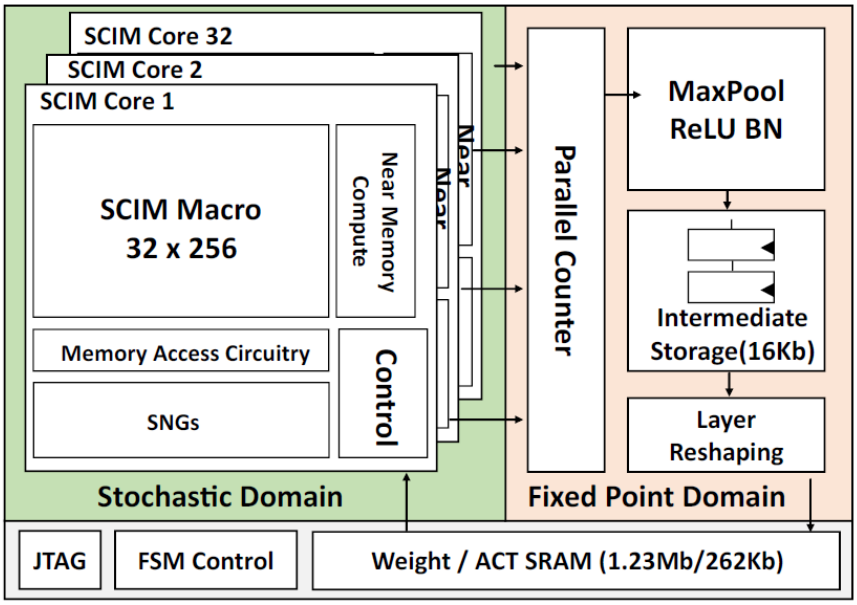



**Figure 30 – SCIM Deep Learning Accelerator**

### 3.5.8 A. Bit Parallel and In-Memory Compute Data Flow

Stochastic Computing (SC) represents number by the probabilities of the stochastic bit streams. Unipolar representation directly maps probability of ones to range between 0 and 1, therefore only representing positive numbers. Split-Unipolar SC representation is a simple way to use the difference between two unipolar bit streams, positive and negative stream, to represent signed number between -1 and 1. When a number is converted to stochastic bit streams, the sign decides whether positive or negative stream will encode the magnitude of the number. The other stream is zero. Figure 31 shows an example.

For a negative number: W=-2/8, the magnitude 2/8 is encoded in the negative stream and the positive stream is zero. The probability difference between positive and negative stream is -2/8. It is not required to always to have one stream as zero, as both streams can become nonzero during computation.

The Multiplication and Accumulate (MAC) arithmetic of the split-unipolar stochastic bit streams needs to account for the cross product between the operands' positive and negative streams. For DNN's application in computer vision, inputs at each layer are always positive numbers. The first layer's inputs are images with positive pixel intensities. Inputs to the hidden

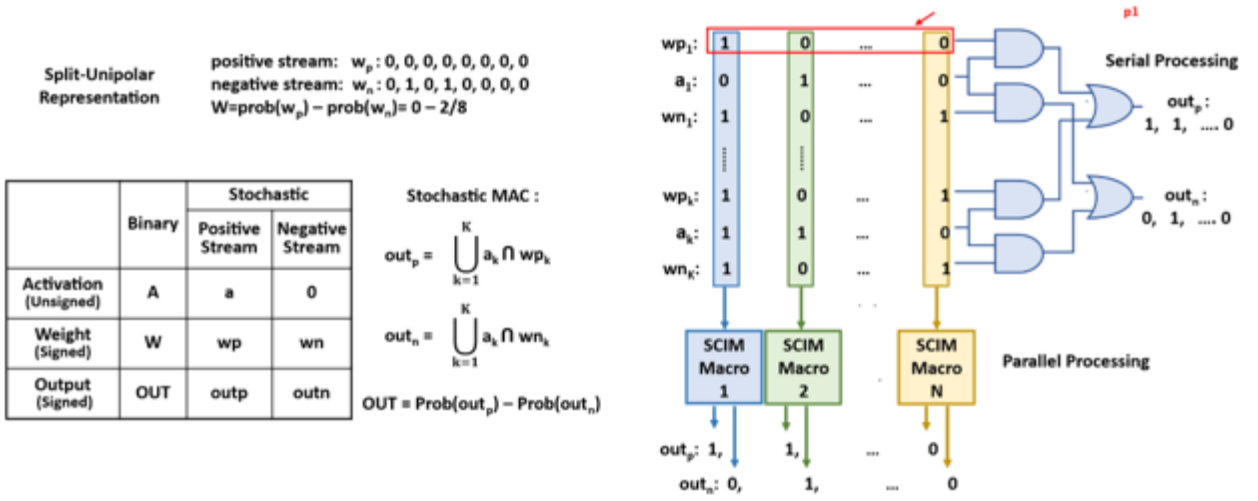



**Figure 31 – Split-unipolar Representation for Signed Number and Bit-parallel Procesing for SCIM**

layers are the values passed through nonlinear activation function such as ReLU and the negative values are clipped to zero. Only one positive stochastic stream can represent the inputs. Weight parameters are signed numbers and require both positive and negative streams. The cross product between input's positive stream: a, and weight's split-unipolar streams: wp/wn is illustrated in Figure 31. Assume both input and weight are 1-D vector with K elements. Each element of input a is multiplied by intersection logic and accumulated by union logic separately with wp and wn. The results of two separate paths are positive and negative stream of the output: outp and outn.

The conventional Stochastic Computing (SC) uses a bit-serial data flow where computation hardware is bit-wise logic gates, and the stochastic bit streams are serially passed through them. Figure 30 (right) shows the bit-serial data flow of the SC MAC. The intersection is done using AND gate and union is done using OR gate. The input and weight stream serially pass the AND gates. The multiplication results of the AND gates are accumulated by two separate multi-input OR gates and lead to positive and negative output stream. In order to reuse the stochastic bit streams, this work proposes a bit-parallel flow that stores all the stochastic bits in CIM macros and computes in parallel. The mapping between conventional bit-serial SC to bit-parallel CIM's parallel processing is shown in Figure 30. Each SCIM macro stores one stochastic bit representation of the activations and computes with one stochastic bit of the weights. N SCIM macros compute in parallel can produce N stochastic bits of the outp and outn in one cycle. Since the input activation is positive and requires 2x shorter bit stream than weights, input stochastic stream is chose to be stored in CIM macros.

### 3.5.9 B. Stochastic Number Generator

A bottleneck with conventional SC is the large energy cost of the conversion from binary to SNs. Stochastic number generators (SNG) typically uses pseudo-random number generators (PRNG) such as linear feedback shift register (LFSR) as the randomness source. The LFSR-based PRNG consumes 25x more energy than an SC MAC unit, which only includes an AND gate and an OR gate. A large reuse factor of the SNG output is required to reduce the energy cost. Figure 32 shows the diagram of the SNG circuit. The LFSR outputs control a chain of multiplexers, which select

between either a binary input bit and the output of the previous multiplexer. N multiplexers are cascaded to convert an N-bit binary number to stochastic bits serially.

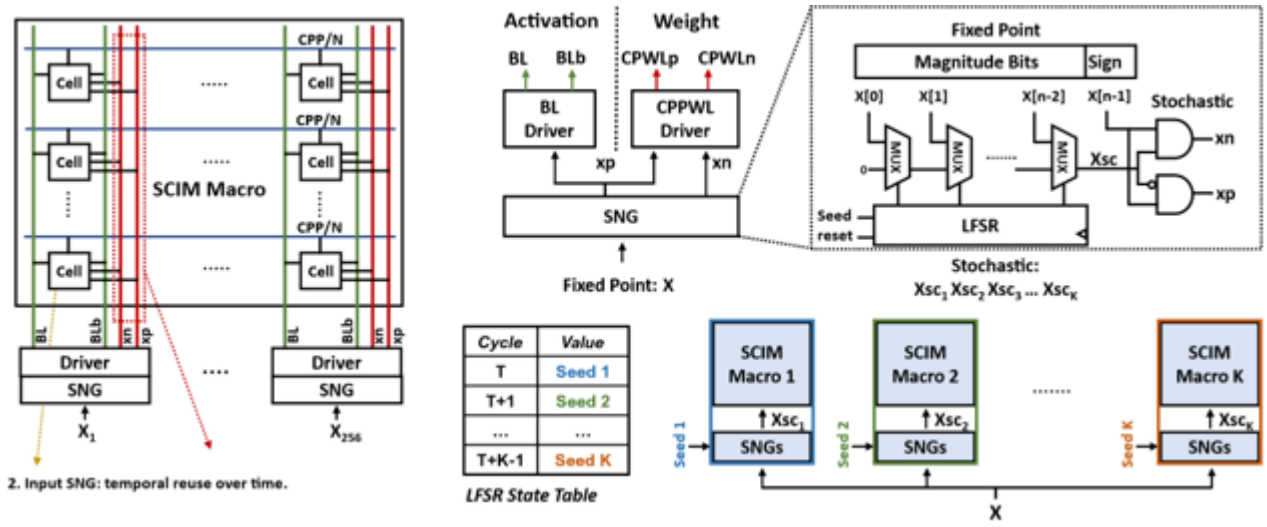



**Figure 32 – Different Data Path of Stochastic Number Generator for Input and Weights; SNGs seen Schedule Supporting Bit-parallel Compute**

It can be shown that the SNG can accurately convert binary numbers if the LFSR size matches the bit width of the binary number. An N-bit maximal-length LFSR will iterate over all the possible combinations of N-bit numbers except zero before repeating the same sequence again. Each N-bit combination is iterated exactly once, and therefore the period of the LFSR state is $2^N$-1. The most significant bit of the binary number is multiplexed to the SNG output for exactly $2^N1$ times within a cycle. Other bits are multiplexed with binary weighted frequencies: $2^{N-2}$, $2^{N-3}$, … , $2^0$, and therefore the frequency of one in the stochastic bit stream accurately represents binary input. An N-bit fixed point number has a sign bit and (N-1) magnitude bits. The LFSR size is chosen to be (N-1) bit. The demultiplexer at the output of the SNG selects between xp and xn based on the sign of the binary input.

The SCIM macro stores the pre-generated stochastic numbers of the activations to increase the reuse factor of the activation SNG. Figure 32 shows the connection between SNG and SCIM macro. Each column has an SNG and driver for BL/BLb and CPWLP/N. The SNG converts the binary number X to stochastic representation: xp/xn. Since the activations in the neural networks are positive by the ReLU function, only xp needs to be stored in the memory. The xp stream is connected to BL driver, which drives BL/BLb and writes the xp to a bit cell. Each macro only stores one stochastic bit of the activation. For example, a binary input X is represented by K stochastic bits: Xsc1, Xsc2, … XscK. The 1st macro stores Xsc1, the 2nd macro stores Xsc2, and the Kth macro stores XscK. A parallel stochastic number generation scheme is shown in Figure 32. The LFSR state table characterizes the changes of shift registers at different cycles. To parallelize the SNG, each LFSR state is used to initialize the LFSR in different SCIM macros: seed 1 for the 1st macro and seed K for the Kth macro. The results of K SNGs in one conversion cycle are the same as a single SNG generating for K cycles.

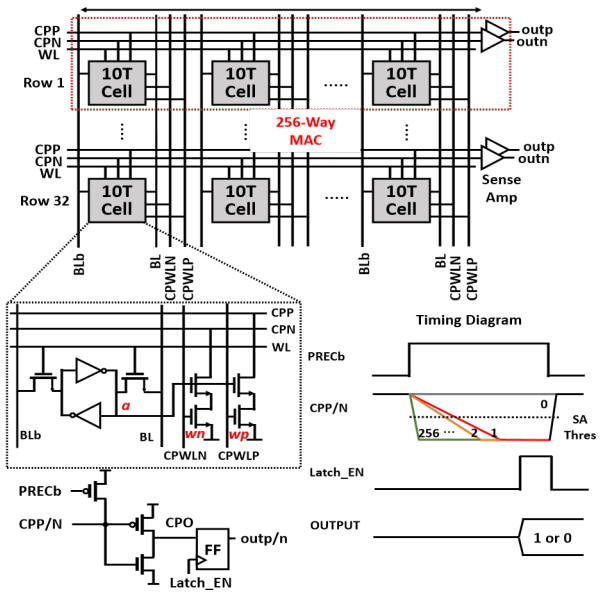



**Figure 33 Stochastic Compute-in-Memory Array**

### 3.5.10 C. Stochastic Compute-In-Memory Array

The Stochastic Compute-In-Memory (SCIM) macro embeds simple digital SC computation logic inside the memory to achieve high energy efficiency for matrix multiplication operations: AND gates for multiplication and OR gates for approximate accumulation. Each 1-bit stochastic representation of the activation is stored in the 10-T bitcell before the computation. The storage cell uses the standard 6-T SRAM cell design. Two extra nMOS transistors perform an AND operation between the stored activation bit and a weight stochastic bit applied at the computewordline (CPWLP/CPWLN). Each 10-T bitcell contains two multiplier circuits which multiply between activation a and wp/ wn. The shared compute port (CPP/CPN) across a row realizes a wired-NOR operation between 256 bitcells. The computation follows the operation of the precharged pseudo-nMOS logic family. The compute port is precharged to VDD before computation and discharged to GND if at least one of the multiplication results is 1. The timing waveform of the MAC operation is shown in Figure 33. Once the precharge phase is over, the compute port starts to discharge. If all the multiplication results are zero, the compute port will remain close to VDD. The leakage current will only cause a small voltage drop. If there is at least one nonzero multiplication result, the compute port will be discharged by the transistor's active current. The evaluation period is designed such that the compute port will be fully discharged to ground for non-zero accumulation results. The sense amplifier is a simple inverter that amplifies the comput port voltage to VDD/GND. Once the compute port evaluation is done, the result is sampled in the flip flop. The SCIM macro MAC operation is DAC/ADC-free, which provides robust digital computation across different supply voltage and process variations. The macro is 32 rows tall and 256 columns wide, which corresponds to 32 of 256-way MAC units.

# 4 RESULTS AND DISCUSSION

## 4.1 Integrated VC-MRAM Chip Testing Results

In Section 3.2, we detailed the development of the MeRAM, a revolutionary memory technology that promises to reshape the landscape of computing. Developed through a collaboration between ITRI and UCLA, the first CMOS integrated MeRAM represents a big step in the field of memory technology. This innovative MeRAM technology holds immense promise, demonstrating the potential to bring about substantial improvements in write power, write time, and memory density when compared to the well-established STT-MRAM (Spin-Transfer Torque Magnetic RandomAccess Memory). Since this was the first CMOS integration of the MeRAM, we carefully designed the circuit to get good performance while ensuring resilience to device variation. In this section, we delve into the testing results of the very first MeRAM chips, providing insights into the performance and characteristics of this new memory technology. These findings mark a significant milestone in our attempt to utilize the full potential of MeRAM and underscore its transformative impact on the realm of memory technology.

### 4.1.1 CMOS Integrated Device Testing Results

On the same CMOS wafer, we designed several test MTJ devices which are not connected to transistor. Thus, we can access each device by probe measurement.

To obtain the target performance of VC-MTJ and improve yield, we worked with ITRI to provide three lots of VC-MTJs. The first lot, lot 1, was a preliminary trial where the thickness of MgO and CoFeB was not yet fine-tuned. This gave a 300 kΩ resistance of parallel state (Rp), which is 3x larger than the designed 100 kΩ, and a high PMA which resulted in higher switching voltage. Resistance-area product (RA) as a function of MgO growth time is shown in Figure 34(a). RA is sensitive to MgO thickness, and 5 s growth time can double the resistance, while TMR is relatively insensitive. Note that TMR is slightly higher with increase of MgO thickness and RA, due to higher quality when MgO layer is thickness. Thus, VC-MTJ after optimization is expected to have better TMR compared with STT-MTJ due to larger MgO thickness. Here we reach film level TMR>120%, which is a reasonable value and can be further optimized. Figure 34(b) shows PMA as a function of free layer CoFeB thickness. A large PMA value provides better memory retention but makes writing difficult. So, a too-strong PMA is unfavorable.

In lot 2, we extract information from lot 1 and fine tune the MgO and CoFeB thickness accordingly, which gives a much better device performance. As shown in Figure 35 critical operation voltage for lot 2 is 1.6 V/1.2 V for MRAM and RNG, respectively. Besides, the yield is significantly improved in lot 2 due to improved fabrication condition, as shown in Figure 36(b) and (c). We planned a lot3 to obtain a better performance and yield, however, lot 3 was delayed and could not be measured in time. But still, the results in lot2 are still good to meet the relevant metrics.

The comparison between our CMOS-integrated VC-MTJ device and previous report without integration as well as STT-MTJ is shown in Figure 38. Our integrated 1st VC-MRAM shows comparable performance compared with previous work without integration and shows major advantages in writing energy and time compared with STT-MRAM. (13x lower write energy, 4x lower write latency than STT-MRAM).

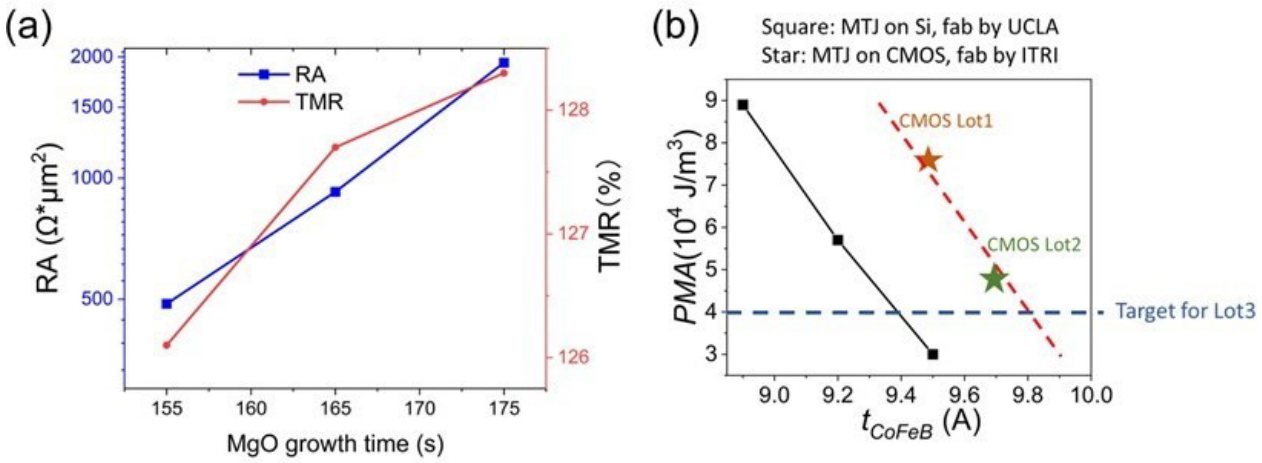



**Figure 34 . (a) RA and TMR as a Function of MgO Growth Time; (b) PMA as a Function of CoFeB Thickness**

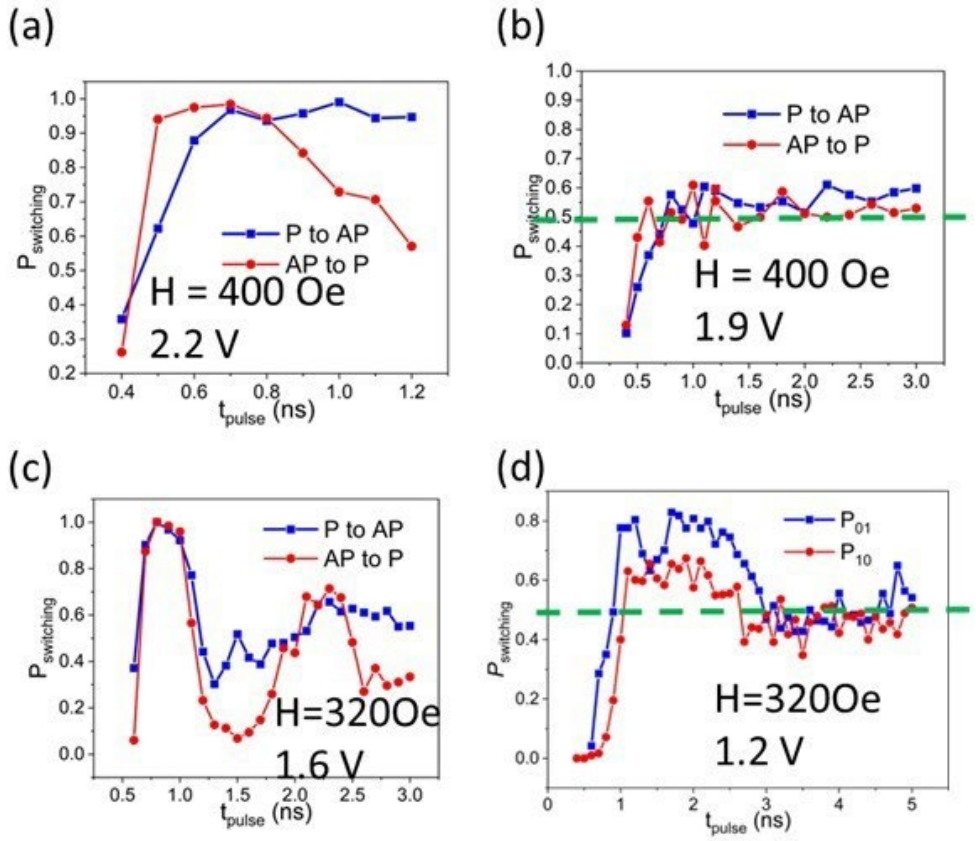



**Figure 35 (a, b) Lot1 Device, Switching Probability as a Function of Time**
*MRAM/RNG operation voltage are 2.2 V/1.9 V respectively. (c, d) Lot2 device, switching probability as a function of time. MRAM/RNG operation voltage are 1.6V/1.2 V respectively.*

(a)

| $t_{CoFeB}$ (Å) | VCMA (fJ/Vm) | PMA ($10^4$ J/m$^3$) | $V_{RNG}$ | $V_{MRAM}$ | Source |
|---|---|---|---|---|---|
| 9.5 (lot1) | 45 | 7.5 | 1.9 V | 2.2 V | Lot1 device |
| 9.7 (lot2) | 48 | 5 | 1.2 V | 1.6 V | Lot2 device |

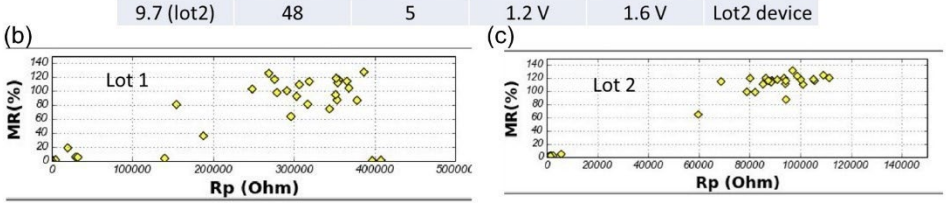



**Figure 36 (a) Device Properties in lot 1 and lot 2, (b) and (c) shows a Significant Yield Improvement in lot 2 Processing**

| | Measured parameter for Lot1 | Measured parameter for Lot2 |
|---|---|---|
| TMR | 100-120% | 100-120% |
| Rp | 300kΩ | 100kΩ |
| VCMA | 45 fJ/Vm | 48 fJ/Vm |
| PMA | ~7.5*$10^4$ J/m$^3$ | ~5*$10^4$ J/m$^3$ |
| $V_{MRAM}$ | 2.2 V (median) | 1.6 V (median), 0.8 ns |
| $V_{RNG}$ | 1.9 V (median) | 1.2 V (median), 3 ns |

**Figure 37 Summarized Device Properties of lot 1 and lot 2**

| Device Type | VCMA | | | STT |
|---|---|---|---|---|
| | **This Work** | VLSI 2020 [3] | IEDM 2021 [4] | IEDM 2021 [23] |
| **Integration Level** | **CMOS Integrated** | No Integration | No Integration | CMOS Integrated |
| **MTJ Diameter (nm)** | **100** | <75 | 100 | 38 |
| **RA(Ω*μm$^2$)** | **800** | Unknown | Unknown | ~10 |
| **TMR Ratio** | **120%** | 246% | 180% | 120% |
| **VCMA (fJ/Vm)** | **48** | 35 | > 100 | NA |
| **Thermal Stability Factor, Δ** | **39** | 54 | > 40 | Unknown |
| **Write Voltage(V)** | **1.8** | Unknown | Unknown | 0.7 |
| **Write Electric field (V/nm)** | **~1.1** | 1.4 | Unknown | NA |
| **Write time (ns)** | **0.7** | 0.9 | 0.6 | 3 |
| **Write energy (fJ/bit)** | **15** | 20 | Unknown | ~200 |
| **Endurance** | **>$10^{11}$** | >$10^{10}$ | >$10^8$ | Unknown |

**Figure 38 Summary of Device Performance Achieved in this Work**

*This work is the first demonstration of CMOS integrated VCMA showing memory operation. Compared to STT, this work demonstrates 4.3x better write speed and 13x better device write energy.*

### 4.1.2 8x8 Array Results

The integrated VC-MRAM die is shown before and after MTJ processing in the micrographs of Figure 39. The die was wire-bonded directly to a test PCB and was tested with a variable magnetic

field generator. We tested and characterized both the array level performance and direct device measurements.

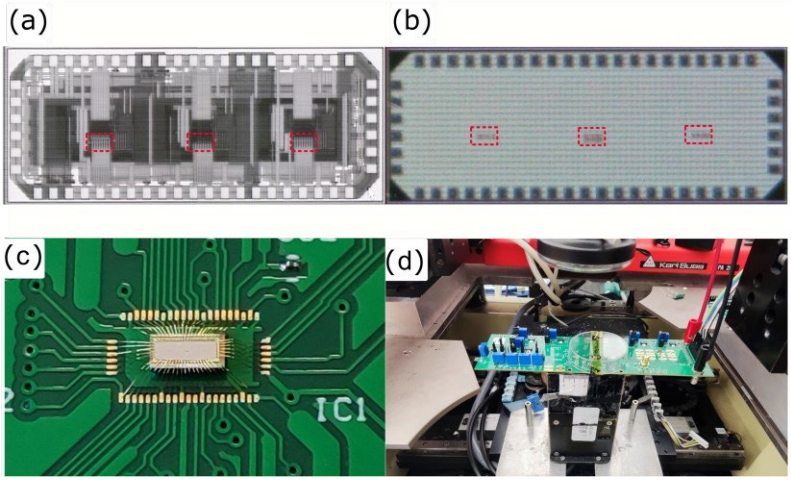



**Figure 39 - 8x8 Chip Micrograph and Testing Setup**

A. *Direct Device Measurements:* The histogram of RP, RAP and TMR of 105 devices from two different dies is plotted in Figure 40. A median cell TMR of 106% is achieved, which is close to the device TMR without access transistor and shows that the high MTJ resistance leads to lower TMR degradation.

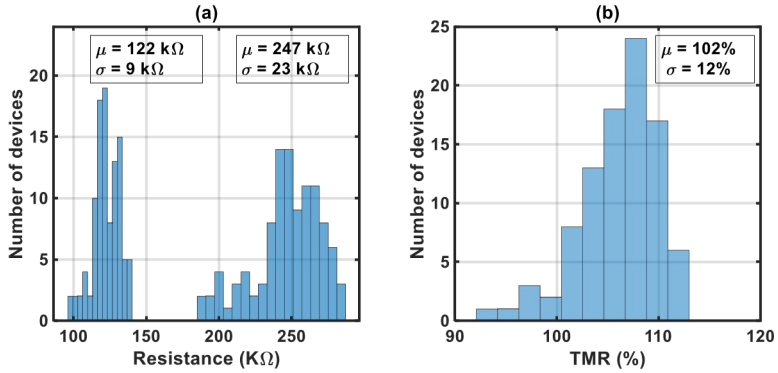



**Figure 40 – Top (a) Measured Distribution of RP and RAP, Top (b) Measured TMR Distribution at a bias Voltage of 100mV for Two Different Dies**
*A median cell TMR of 106% is achieved.*

B. *Array Performance Results:* Reliable switching is essential for memory writes. To test the switching, write pulses of different durations and voltages were applied to all devices and the MTJ state was read back through the memory access circuitry. The average switching probability was calculated for two different dies with 256 write attempts per device. Figure 41 shows the average switching probability of devices from the two dies after removing outliers with switching probability of less than 80% at a pulse-width of 700ps (leaving 70 devices). Because PMA .

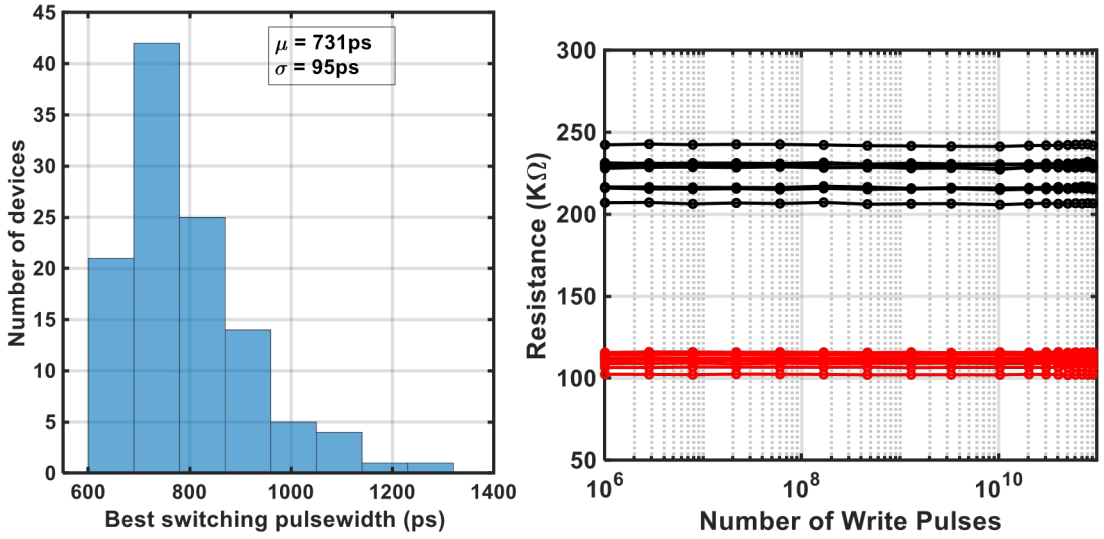



**Figure 41 – (Left) Measured Distriubtion of Optimal Pulse width for 113 Devices from Two Different Dies, (Right) Measured RP and RAP from 8 Devices during Application of 1.8V Write Pulses with a Pulse Width of 700 ps**

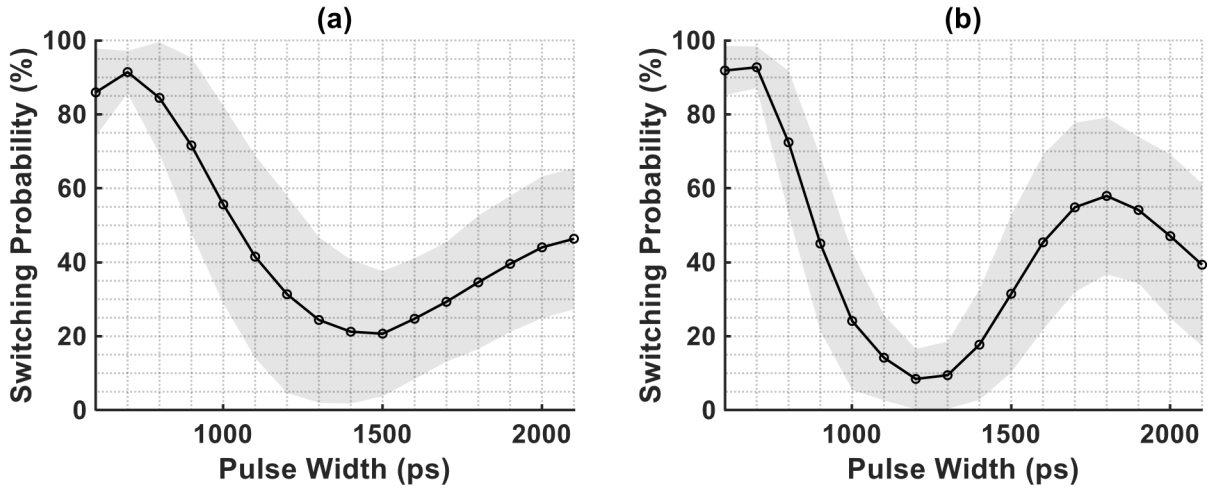



**Figure 42 - Measured Average Switching Probability (with 1-sigma variation shaded) vs Pulse Width for Devices from Two Different Dies at a Write Voltage of (a) 1.8V and (b) 2.2V at an Externally Applied Magnetic Field of 37.5mT at θ~60°**

For memories, it is not practical to calibrate the pulse width for each device, so it is important to achieve good device uniformity and select the best pulse width for the entire array. A distribution of the best pulse width (where switching probability is maximum for each device) for over 100 devices from two different dies is shown in Figure 42(left). Most devices achieve their best switching probability at around 700ps. To test the array's endurance, 1.8V, 700ps write pulses at a rate of 1MHz are applied to one row of 8 devices, and the resistance of each device is measured intermittently. As shown in Figure 42(right), the resistance does not show any sign of drift over $10^{11}$ write cycles, indicating excellent endurance. Different write patterns were written to and

read back from the MTJ array, as shown in Figure 44. Non-volatility was confirmed by power cycling the chip in between writing and reading.

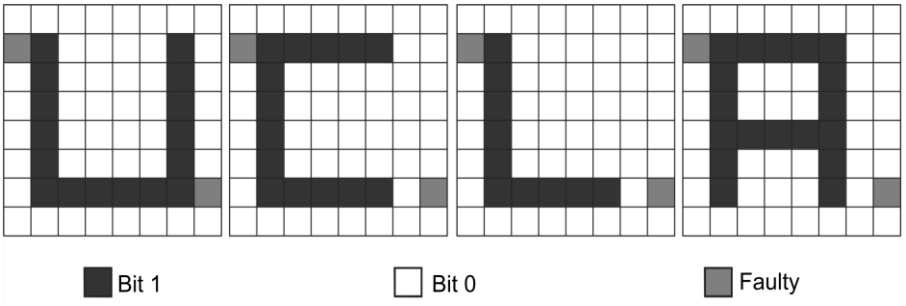



**Figure 43 - Bitmap of the 8x8 MTJ Array Read Back from the Chip after 4 Different Writes, Consisting of the Characters "UCLA"**

*The chip is allowed multiple write attempts to write outlier devices which may have low switching probability.*

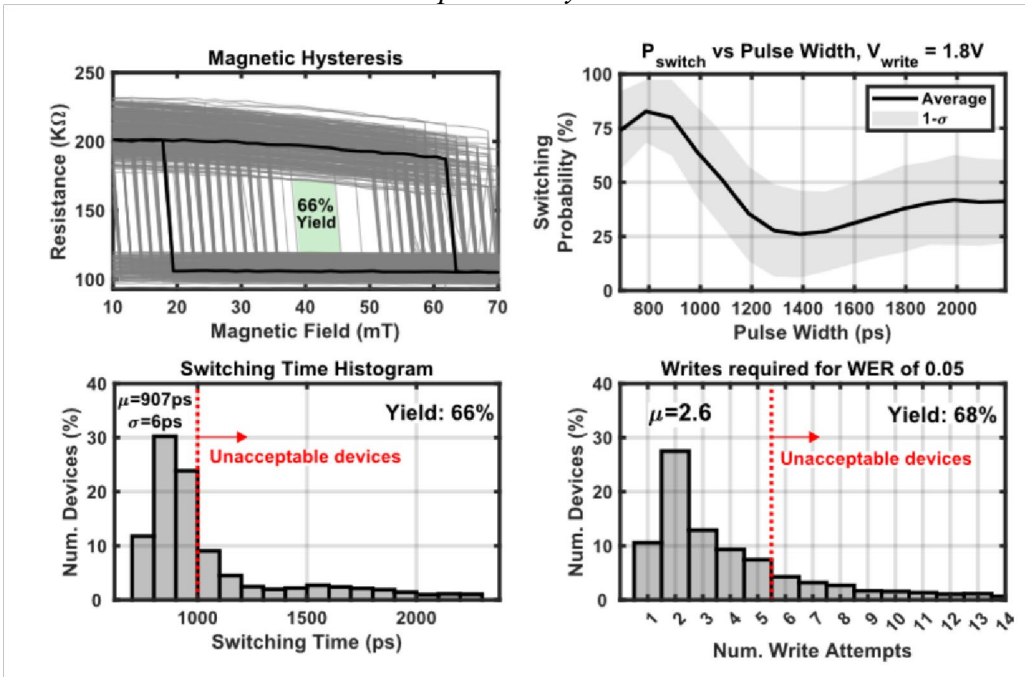



**Figure 44 - Device Characterization Results from the 80x80 Array**

### 4.1.3 80x80 Array Results

Array-level statistics of important device characteristics for the 80x80 chip are shown in Figure 43. On the top left are the magnetic hysteresis curves, with a typical curve highlighted, of 400 evenly sampled devices measured under the influence of an external magnetic field. An open hysteresis window, marked in green, is obtained with a yield of 66%. To measure the devices' switching characteristics, write pulses are applied consecutively to each row 256 times and the MTJ state is read back through the access circuitry after each pulse to determine the switching probability. The average switching probability vs pulse width is determined after removal of devices with poor switching probability (<50%), leaving 66% of the devices. An average switching probability of 83% is achieved, which is expected to improve as the device fabrication processes

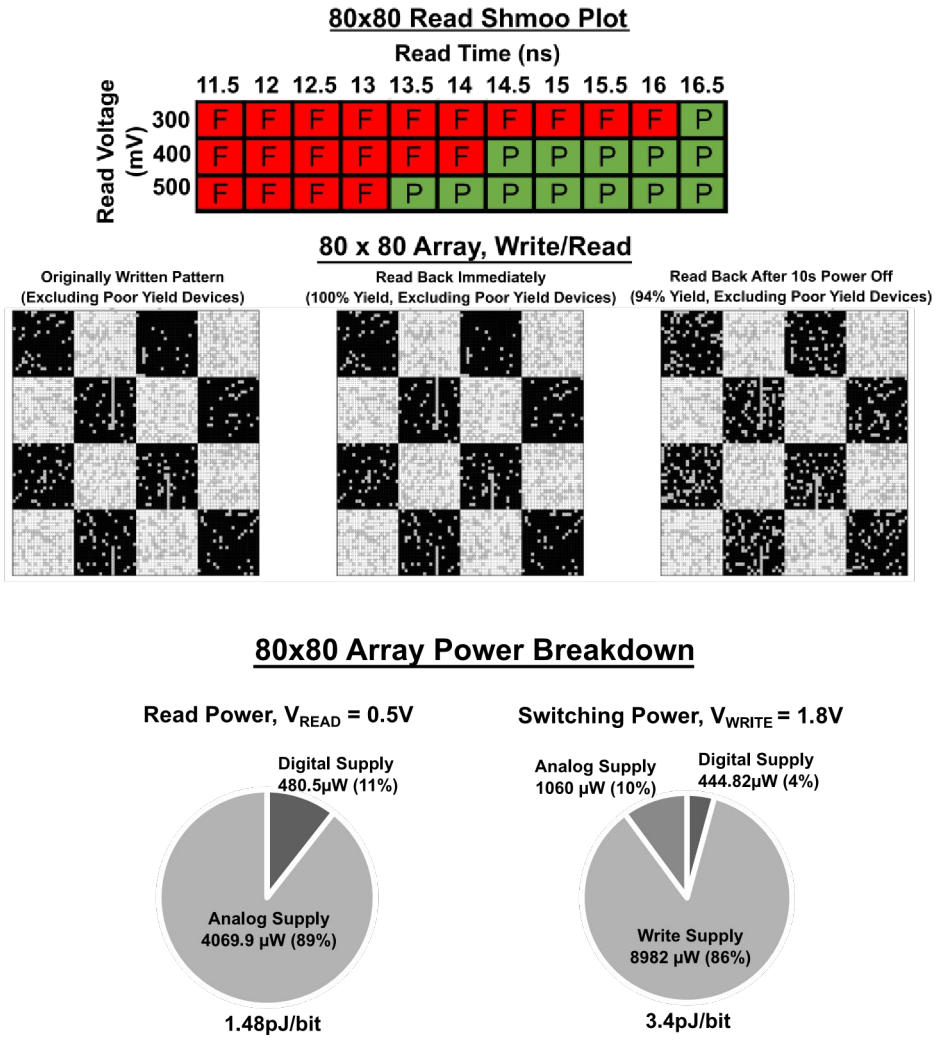
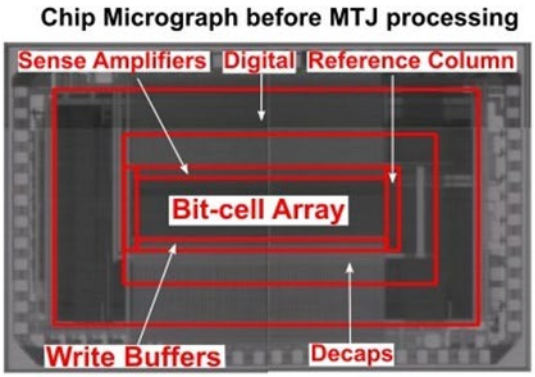
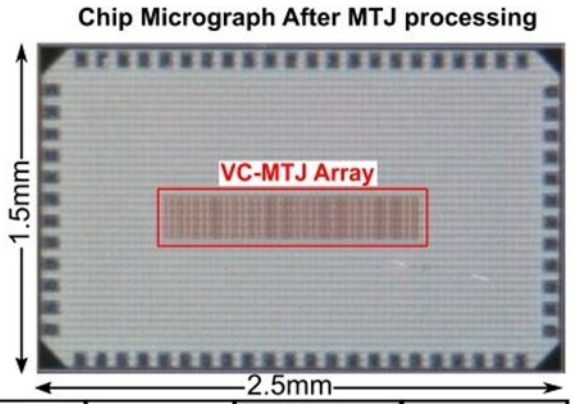



**Figure 45 - Read Shmoo, Write Pattern Test, and Measure Energy for the 80x80 Array Mature**

*The histogram of switching time, which is the pulse width for which the device has highest probability of switching, shows that most devices switch in ~0.9ns. Endurance of $>10^{11}$ write cycles is measured.*

For the devices that did yield well, we were able to show non-volatile memory operation. As shown in Figure 45, a box pattern, and its inverse (not shown) are written to the 80x80 array with 15 attempts allowed per write to account for the variation in write probability. After power cycling the chip, 94% of the devices that can be initially written retain their value. This clearly demonstrates the non-volatile nature of the MeRAM, and the 6% loss is due to low thermal stability and will improve with device maturity.




| | ISSCC'18 | ISSCC'20 | ISSCC'10 | ISSCC'07 | This Work 80x80 | This Work 8x8 |
|---|---|---|---|---|---|---|
| Technology | STT-MRAM | STT-MRAM | STT-MRAM | STT-MRAM | MeRAM | MeRAM |
| Process | 28 nm | 22 nm | 0.13 um | 0.2 um | 0.18 um | 0.18 um |
| Array Size | 1Mb | 32Mb | 16Kb | 2Mb | 6.4 Kb | 64 b |
| Read Time (ns) | 2.8 | 10 | 8 | 40 | 13.5 | 8.5 |
| Read Energy (pJ/bit) | 0.7 | 0.64* | - | - | 1.48 | - |
| Switching Energy (pJ/bit) | - | - | - | - | 3.4 | - |
| Device Switching Time (ns) | - | - | - | - | 0.8 | 0.7 |
| Total Write Time (ns)** | 20 | - | 10 | 100 | 14.3 (Min)<br>37.2 (Average)<br>71.5 (Max) | 9.2 (Min)<br>23.92 (Average)<br>46 (Max) |
| Total Write Energy (pJ/bit)** | 4.5 | 5.06* | 4.32* | 36* | 4.9 (Min)<br>12.7 (Average)<br>24.4 (Max) | - |

***Calculated From Given Power Numbers**

****For MeRAM, calculated based on minimum/average/maximum number of write attempts required for a WER of 0.05, considering only acceptable devices that pass yield criteria. Each write attempt involves a read + switch operation.**

**Figure 46 – Micrographs and Comparison Tables**

To decouple the device write variations from the read performance, the read tests were done by forcing each device into a known state (P or AP) using a strong external magnetic field. As shown in the shmoo plot, a read time of ~13.5 ns is achieved to read devices with a read success rate of 96.6%, which corresponds to devices within +/- 20% TMR variation.

Figure 46 shows a comparison with existing STT-MRAM. The 0.18um MeRAM's total read time, write time, and write energy are comparable or better than 0.13um and 0.2um STT-MRAM prior art. The minimum MeRAM write energy is comparable to even 22nm STT-MRAM. Note that the minimum case corresponds to a single read-switch cycle. Several devices required multiple readswitch cycles before correct writing because of low yield and device characteristic variability. This work required processing over the entire CMOS wafer, and so the 180nm process was selected for its low cost. Significant read and write performance benefits are expected for MeRAM integrated in finer technology nodes as most of the read and write latency and energy are dominated

by 0.18um circuits whereas the major contributor to the STT-MRAM's performance is the device itself and does not scale with CMOS.

Due to the nascent integration process, device variation and the low (~66%) yield were a concern. These issues are expected to be resolved with maturity in the fabrication process. In conclusion, the chips presented in this paper demonstrate the first operational, CMOS-integrated MeRAM, showing a sub-1ns switching time and 3.4pJ/b switching energy.

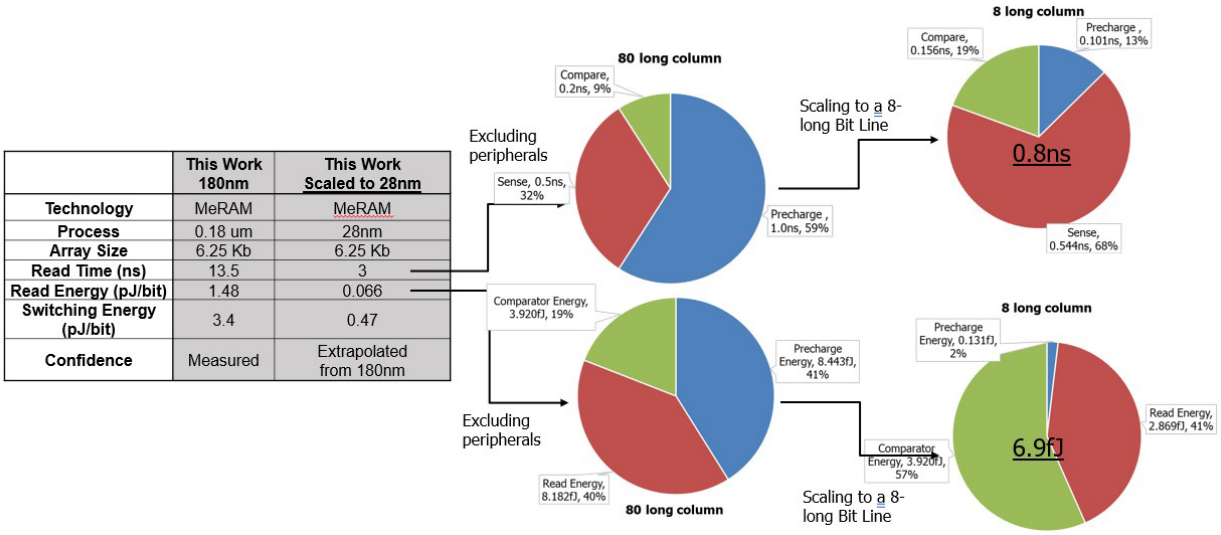



**Figure 47 - 180nm Energy and Latency Breakdown and Scaling to 28nm Short Column for Single Device Read Metric**

### 4.1.4 256x256 Testing Results

The 256x256 (64kb) is the densest array that was developed for this project. It consists of 4 banks of columns, each composed of 256 bit-cells. The banks are selected using the MSB bits of the address. For sparing, the total number of columns is 320. We confirmed the functionality of the 64kb array by two different data patterns. As expected, there are some yield issues resulting in faulty bits, but for the functioning bits, we can show memory operation. Shown in Figure 48 and Figure 49 is a measured bitmap of the 256 x 256 array after specific patterns (box pattern and 'UCLA' pattern) were written and read back from the MeRAM chip.

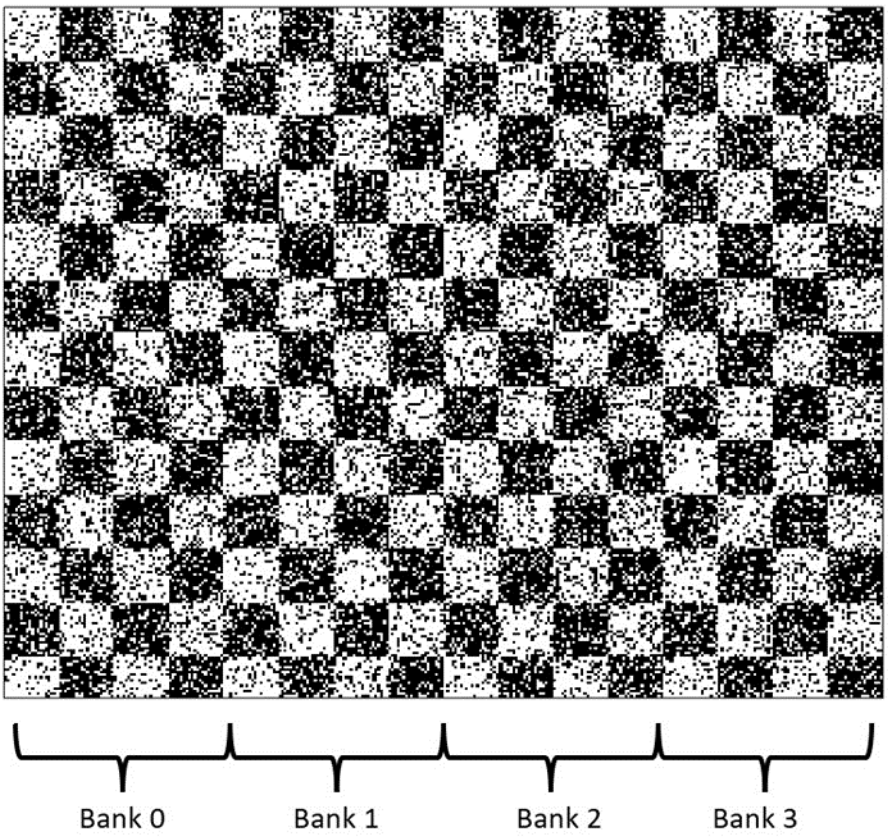



**Figure 48 – Bitmap of a Box Pattern Written to and then Read Back from the 256x256 Array**

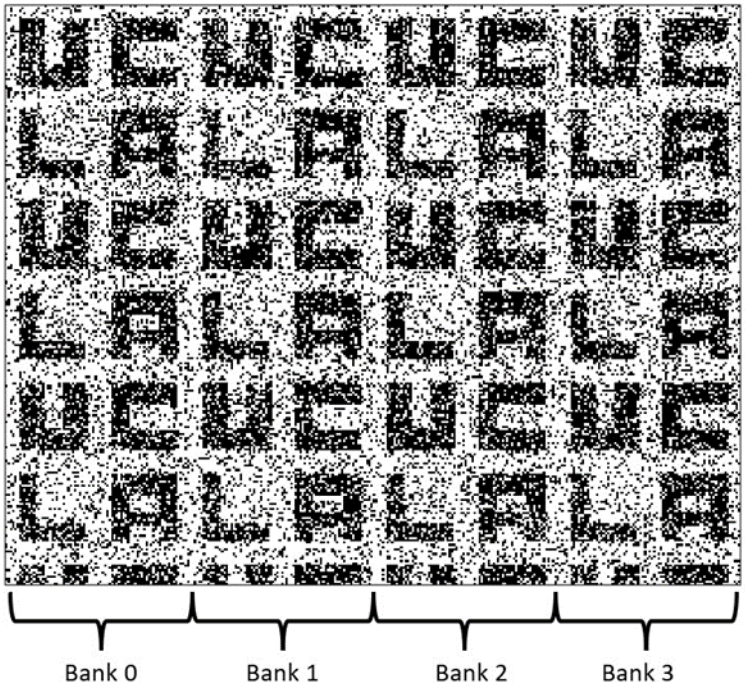



**Figure 49 – Bitmap of a 'UCLA' Pattern Written to and then Read Back from the 256x256 Array**

## 4.2 Evaluation of Metric 1: Single Memory Device Read EDP

In the 180nm test chip, the array sense-amplifier circuit has been optimized to handle longer columns and significantly larger cell capacitances. Access to a finer technology node (e.g., 28nm) enables gains in both read energy and latency. A direct scaling of the 180nm sense-amplifier used in the 6.25 Kb array to 28nm brings down the cell access energy from 1.48 pJ/bit to 66 fJ/bit. Cutting down the array size from 80 long to 8 long brings it further down to 6.9 fJ/bit . This includes 3.9fJ/bit of the additional stage in the comparator to cater to tighter offset requirements and overhead to cater to uncertainty in the device TMR and RA. A wide tolerance range of 65% to 200% for TMR and 50 KΩ to 400 KΩ for $R_P$ was considered at design time for the fabricated prototype. However, this may not be the optimum circuit to use at 28nm. The MDC discussed in Section 3.3 is optimized in 28nm to cater to very short columns and assumes better controlled VCMTJ device properties. Figure 50 shows the simulated VC-MRAM read energy and latency v/s column size in a 28nm CMOS array, using the circuit described in Section 3.2. The circuit achieves a competitive energy delay product of 2.1 fJ/bit×0.4ns for an 8-long column which is ~20% better than the Phase III program target of 1 fJ/bit × 1ns. This additionally positions VC-MRAM as a promising technology candidate in high performance memory systems.

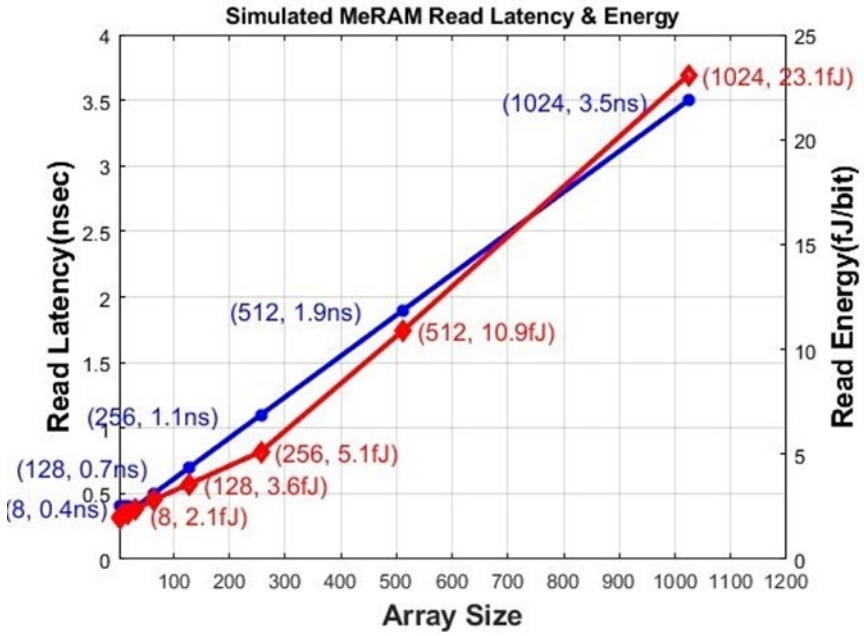


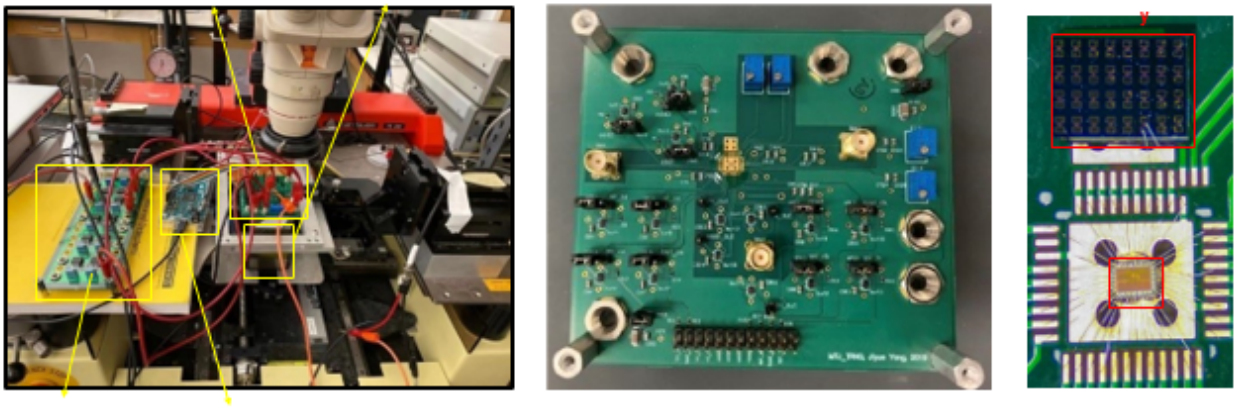

**Figure 50 Read Latency and Read energy v/s VC-MRAM Array Size**

### 4.2.1 Testing of the TRNG Chips

The VC-MRAM based TRNG is demonstrated by two designs: 1) A 65nm chip with the array circuit and the VC-MTJs split in two different die. The connection between the chip and devices are direct by wire bonds. 2) A 180nm chip with a 16x8 VC-MTJ array integrated on the CMOS' backend. The testing setup of the 65nm chip is shown in Figure 51. The core area of the TRNG chip is 200um x 300um. Both CMOS chip and VC-MTJ die are wire bonded to the PCB board and connect through the metal traces. The PCB board is placed on top of a magnetic field generator. The fabricated VC-MTJs do not have a nano-magnet and require an external reference field. The

magnetic field generator provides a reference field and also helps remove part of the stray field that might cause a bias in the random numbers.

The PCB board is controlled by an Arduino micro-controller and the high-speed random number outputs are acquired by Texas-Instrument Data Acquisition System. The raw random numbers are generated by the VC-MTJs first, then accessed by the on-chip sense amplifier. The raw random numbers are processed by the on-chip bias correction circuit and then transmitted off the chip. The write pulse width is 30n sec and does not require calibration process. This is a big advantage over the STT-MRAM based TRNG. Previous work of STT-MRAM TRNG [26] [27] require either onchip or off-chip calibration loop using complicated digital processing or high-precision measurement equipment to find the acceptable pulse width. The random numbers generated from the 65nm chip pass the NIST 800-22 test, shown in Figure 49 (left). Multiple VC-MTJs are tested and all of them pass the randomness test. The supply voltage of the chip is 1.2 V. The throughput is 400Kb/s and the TRNG consumes 135pJ/b.



Summary Table

| | |
|---|---|
| Entropy Source | VC-MRAM |
| Technology | 65nm with Wire Bonded Device |
| Supply Voltage | 1.2V |
| Array Calibration | No |
| Digital Post Processing | Yes |
| Pass NIST 800-22 | All |
| Bit Rate | 400kb/s |
| Energy Efficiency | 135pJ/b |

| | NIST 800-22 Test | DEVICE 1 | | DEVICE 2 | |
|---|---|---|---|---|---|
| | | Pass Rate | $\chi^2$ of P-Value | Pass Rate | $\chi^2$ of P-Value |
| 1 | Frequency | 100% | 0.018 | 100% | 0.740 |
| 2 | Block Frequency | 100% | 0.006 | 100% | 0.013 |
| 3 | Cumulative Sum | 100% | 0.035 | 100% | 0.210 |
| 4 | Runs | 100% | 0.534 | 100% | 0.99 |
| 5 | Longest Run | 100% | 0.637 | 100% | 0.350 |
| 6 | Rank | 95% | 0.909 | 100% | 0.534 |
| 7 | FFT | 100% | 0.740 | 100% | 0.911 |
| 8 | Nonoverlap Template | PASS | PASS | PASS | PASS |
| 9 | Overlap Template | 100% | 0.091 | 100% | 0.637 |
| 10 | Universal | 95% | 0.066 | 100% | 0.350 |
| 11 | Approximate Entropy | 100% | 0.740 | 95% | 0.911 |
| 12 | Random Excursion | PASS | PASS | PASS | PASS |
| 13 | Random Excursion Variant | PASS | PASS | PASS | PASS |
| 14 | Serial | 95% | 0.350 | 95% | 0.091 |
| 15 | Linear Complexity | 100% | 0.440 | 100% | 0.534 |

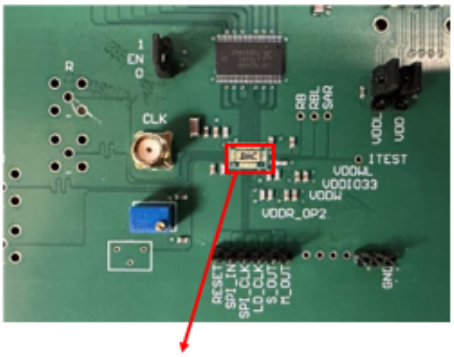
1) Pass rate >90% and P-Value > 0.0001 is considered random.
2) Pass means that the tests of all subcategories are passed

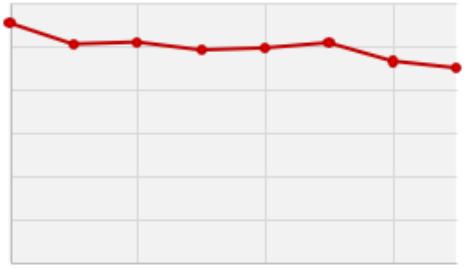
**Figure 51 – (Top) Photo of the Testing Setup (Bottom) Summary of the Chip and Test Results**

The 180nm chip has VC-MTJs integrated on the CMOS' backend and much better performance is achieved. The testing PCB board is shown in Figure 52. The TRNG is directly wire-bonded to PCB. The RNG output is expected to have a very high throughput and a digital buffer is fabricated on board to support high speed I/Os. The probability of ones in the random number generated from the TRNG chip is close to 50% for a large range for external field, shown in Figure 52. The random number outputs pass 6/15 NIST tests and we are still experimenting with a few options to improve the results. The preliminary performance measurement is summarized in Figure 52. The unbiased random numbers have 40Mb/s throughput and 5.3pJ/b energy efficiency.

### 4.2.2 4.3.2 Evaluation of Metric 2

Metric 2 requires energy consumption of 10fJ/b for random number generators. We evaluated the performance of the VC-MTJ based TRNG that can generate unbiased random numbers in a more advanced 28nm technology node. Based on our measurement results of lot 2's fabricated devices, the VC-MTJ can achieve 50% switching probability using a pulse width of 3n sec and voltage of 1.2 V. The switching probability of VC-MTJ vs the pulse width is shown in Figure 53 (top). It is measured directly on the device. We expect the write voltage to reduce to 1V in the lot 3 due to slightly reduced Perpendicular Magnetic Anisotropy (PMA) coefficient. The expected RNG reset energy is 15fJ/b, which corresponding to the power consumption of the write buffer operating for 2.5~3ns.

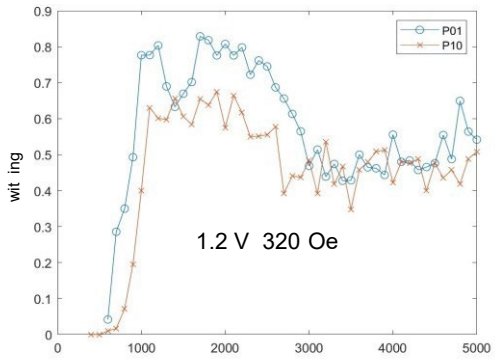


| Technology | 180nm Integrated VC-MTJs |
|---|---|
| Supply Voltage | 1.8V |
| Array Calibration | No |
| Digital Post Processing | Yes |
| Pass NIST 800-22 | 6/15 |
| Bit Rate | 40Mb/s |

**Figure 52(Top) PCB Board of the 180nm Chip Testing; Probability of the Random Bit Stream at Different External Field (Bottom) Summary of the Chip**

The VC-MTJs can form a small array to reduce the parasitic capacitance on the bitline. The ultra low-power sense amplifier described in section 3.4 can be used to minimize the access energy, which consumes 2.1fJ/b and takes 0.4ns. The RNG will consume energy of 17.1fJ/b combining the energy in reset and read cycle. If the write voltage can be reduce further to 0.8V in the future by having a higher VCMA coefficient, the reset energy can be reduce to 9.6 fJ and the total energy can be reduce to 11.7fJ.




| | | | |
|---|---|---|---|
| VCMA oe i ient | 50 70 V | **48 /** | |
| A ied agneti ie d | 500 Oe | **32** | |
| e et e o tage d ration | 0.6 V 4 3 n | **.2 , 3** | E e t 1Vwit ina ot |
| e i tan e M | 100 kO 64% | **k , > %** | |
| N e et Energ | 10.8 8.1 | **2 .5 /** | E e t 15 |
| ead Energ Low M De ign | 2.1 | **2. /** | Cir it i ation in 28n |
| N energ raw trea | 12.9 10.2 | **23. /** | E e t 17 |
| N energ wit 50% t ro g t ia orre tion 4 | 29.8 24.4 | **5 .2 /** | E e t 38 Corre tion a gorit te ted wit data ro re io N |

**Figure 53 Measured Properties of the VC-MTJ**

### 4.3 Network-Level Performance of SC/SCIM ML Accelerators and Evaluation of Metrics 3 and 4

Stochastic Computing (SC) uses simple bitwise logic gates such as AND/OR gates to achieve massive parallelism on chip to reduce the off-chip memory access cost. We have developed an SC accelerator prototype in 14nm using fully synthesizable digital circuit to achieve high energy efficiency and programmability. SC's OR-based accumulation can also be extended to multi-bit operation and further improves the energy efficiency and throughput of digital SC accelerator. Compute-In-Memory is an emerging concept to embed computation logic directly inside the memory to avoid data movement. We have built a Stochastic Compute-In-Memory accelerator in 65nm that achieved very high energy efficiency and macro density. SC is a digital computing paradigm. Embedding SC MAC unit inside the memory does not require large and power-hungry analog data converters. It is also very robust in low supply voltages. We have also built a 14nm chip of SCIM macro to characterize the performance of SCIM in advanced node. Our measurement results validated our design and achieved the metric 3 and 4.

#### 4.3.1 Digital SC Accelerator

The digital SC accelerator chip, shown in Figure 54, was fabricated in 14-nm LPP technology. The core of our accelerator occupies a 0.5 mm2 area and operates at 0.6–0.9 V and a maximum frequency of 500 MHz. Figure 54 shows the accuracy, latency, and energy measurements on different networks and datasets. We highlight that SC exposes a precision-performance tuning knob, by adjusting the stream length. For example, on the MNIST dataset, changing the stream length from 32 to 16 reduces the accuracy by only up to 0.3 p.p., while reducing energy and latency up to 31%. On SVHN, accuracy can be improved by up to 4.7 p.p. with a 50% increase in latency and energy. The nonideal performance scaling is caused by control logic overhead and will be improved in the future. Thanks to highly compact SC compute we achieve higher data reuse than other accelerators, resulting in lower relative memory power contribution. Peak energy efficiency ranges between 9.4 and 75 TOPS/W at 250 MHz/0.6 V. Overall, SC offers unparalleled computational density in terms of MAC units per mm2.

Our chip outperforms fixed point accelerators in energy efficiency while enabling configurable precision. While bit-serial architectures can have high peak energy efficiency, it is only achievable at single-bit precision. Mixed-signal achieves extremely high energy efficiency, but it comes at a cost of little configurability, supporting only one convolutional filter size. It requires using much larger, slower models to improve accuracy. We also outperform or approach the efficiency of neuromorphic and analog designs, without suffering from their scalability and programmability issues, on account of being purely digital. Note that this accelerator was used to meet Phase I and II requirements for metrics 3 and 4. The following sub-section describes the results from a modified version with the OR-N accumulation method described in Section 3 to further improve the metrics.

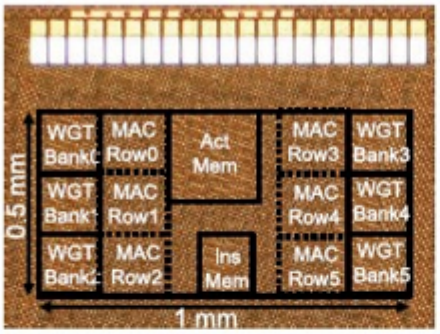
#### 4.3.2 Digital SC Accelerator with OR-N Accumulation

OR-N accumulation can improve neural network accuracy and significantly improve energy efficiency. Accuracy comparisons between OR-N and other alternatives are shown in Figure 55 for TinyConv on CIFAR10. OR 1 uses OR gates for accumulation similar to our 14nm digital SC accelerator chip. SC-Bin uses parallel counters for accumulation. PB uses partial binary accumulation setup Accuracy of OR 2 is comparable to PB at both stream lengths, while OR 3

outperforms both. SCBin has a 2-4% point accuracy advantage over OR 2 and 1- 3% point advantage over OR 3. Compared to FXP6 baseline using 6-bit fixed-point multiply-accumulate, OR 2 has a 4-6% point accuracy deficit, while OR 3 narrows the gap to 3-5% points.

Figure 55 also shows the results of the same concept applied to AND multiplication, denoted as ”AND 2” and ”AND 4”. AND n takes groups of n bits from each of the two multiplicands and multiplies all bits from the first group with all bits from the second group. Traditional AND multiplication can thus be seen as AND 1. AND 2 and AND 4 increase multiplication stream length by 2X and 4X respectively and increase overall area and energy by the roughly same amount. While AND n can also improve accuracy, it is not energy- or area-efficient. AND 4 roughly matches OR 1 when using half the stream length, but consumes $\approx$ 2X the area and energy.

Accuracy results of VGG-16 on CIFAR-10 and Resnet- 18/34 on ImageNet are shown in Figure 53. Both OR 2 and OR 3 achieve similar accuracy to 6-bit fixed-point on VGG-16. While OR 2 and OR 3 improves Top-5 accuracy by 3-5% points compared to OR 1 on ImageNet, they are 25% points lower than FXP6. Accuracy of OR N is likely limited by the training hyperparameters.




| | Specifications |
|---|---|
| Technology | 14nm LPP |
| Die Area | 5 mm² |
| Core Area | 0.5 mm² |
| Supply Voltage | 0.6-0.9V |
| Frequency | 250-500 MHz |
| Precision | 8-64 bitstreams |
| Memory | 167 KB |
| Power | 16 mW @ 250 MHz, 0.6V |
| | 68 mW @ 500 MHz, 0.9V |

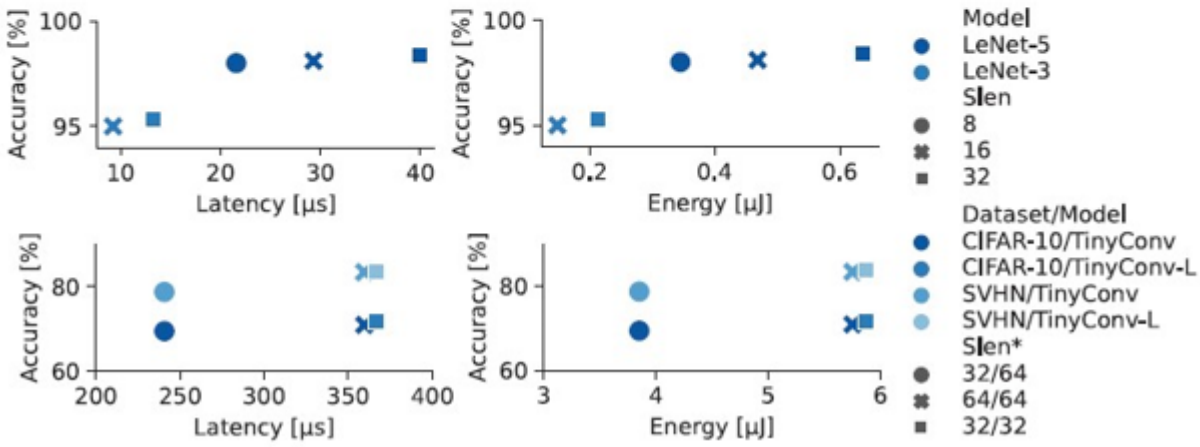



**Figure 54 – Chip photo and Performance Summary; CNN Performance Over Several Datasets**

For instance, doubling the number of epochs from 35 to 70 improves accuracies by 1.5-2% points for OR n, and longer training times should improve accuracies even further. Compared SC-Bin that has unlimited accumulation precision, OR 2 and 3 reduces the accuracy gap while

requiring at most 2 bits of accumulation precision. Compared to fixed-point computation and SC-Bin, OR N enables scaling to higher performance levels. While SC-Bin using 16-bit streams has accuracy between 32-bit and 64-bit OR 3 on TinyConv, it is not a useful setup for the other models. Both fixed-point and SC-Bin have convergence problems in VGG and Resnets when dropping precision further, as weight values tend to underflow the smallest representable value with 5-bit fixed-range quantization (16-bit stream for SC).

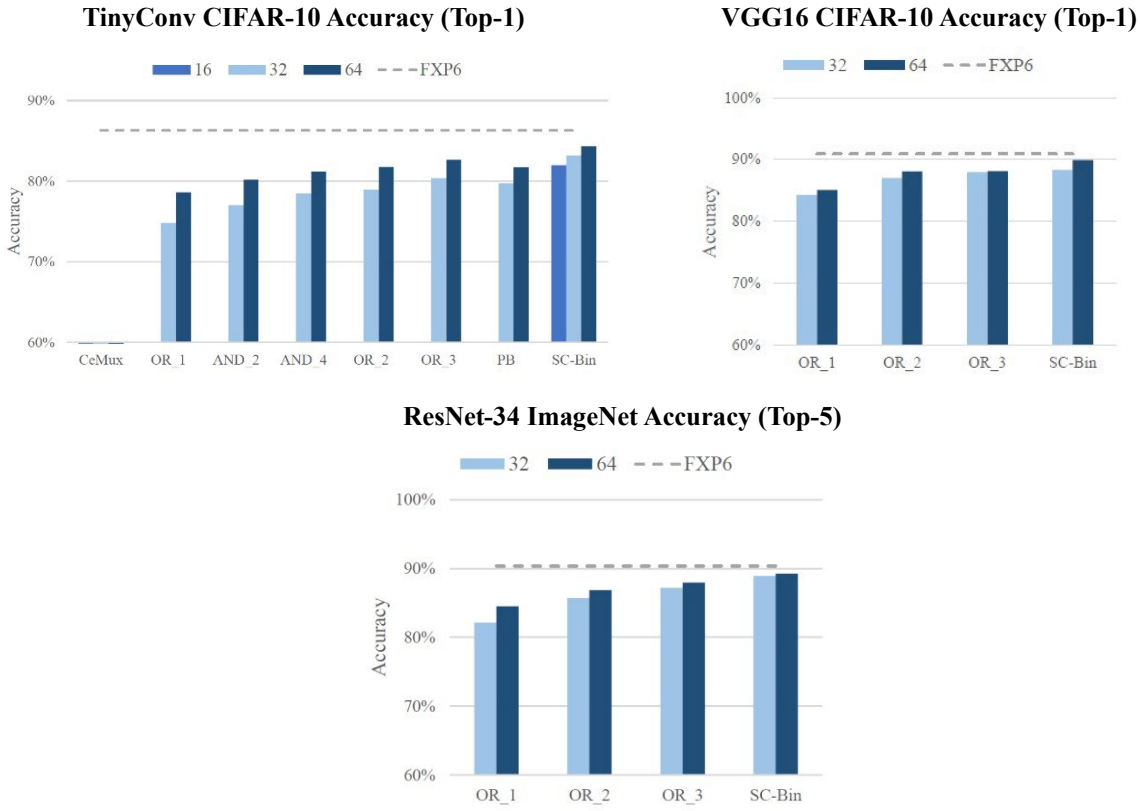



**Figure 55 – CNN Accuracy for Different Networks**

To evaluate the performance benefits of using OR n implementations, we use the previously synthesized adder and dot product results together with buffers and SNG results using a commercial 28nm technology and Cadence Genus synthesis tool. SNGs are based on maximumlength LFSRs, shared across different processing elements, and comparators. We use 6-bit fixedpoint (FXP6) as a baseline, and compare its performance with SC implementations. Besides, SC with binary accumulation (SC Bin), OR 1, OR 2, and OR 3, we include uSADD and uNSADD (scaled and non-scaled adders) from [24]. Both uSADD and uNSADD use uMUL multipliers, which offer high accuracy without retraining, but at a significant hardware cost. To evaluate system-level performance, we assume dot-product processing elements are organized as a N-byN GEMM array, similar to recently proposed SC accelerators [24], [25], [28]. We assume the individual PE to have a width of 256. We use a N = 64 array using OR-based SC dot products as a baseline, which occupies an area of 3.2mm2, and we size up all other configurations to be as close to this area budget as possible.

| Architecture | Stream Length | ImageNet ResNet-34 Latency [us] | Energy [uJ] | E. Impr vs FXP6 |
|---|---|---|---|---|
| FXP6 | | 817.5 | 1413.2 | 1.0 |
| SC Bin | 32 | 449.8 | 1666.0 | 0.8 |
| | 64 | 899.6 | 3142.4 | 0.4 |
| SC uSADD | 32 | 3718.2 | 4243.8 | 0.3 |
| | 64 | 7436.5 | 7850.5 | 0.2 |
| SC uNSADD | 32 | 3936.0 | 4675.4 | 0.3 |
| | 64 | 7872.0 | 8713.7 | 0.2 |
| SC OR_1 | 32 | 285.5 | 267.2 | 5.3 |
| | 64 | 571.0 | 406.4 | 3.5 |
| SC OR_2 | 32 | 417.8 | 404.7 | 3.5 |
| | 64 | 835.5 | 653.6 | 2.2 |
| SC OR_3 | 32 | 424.1 | 437.0 | 3.2 |
| | 64 | 848.1 | 712.8 | 2.0 |

**Figure 56 – Performance Summary on ResNet for ImageNet Classification**

We then try to normalize the throughput w.r.t. FXP6 for all configurations, assuming 64-long streams, by increasing or lowering clock frequency, and corresponding power to take advantage of voltage-frequency scaling. SC with binary accumulation and OR 1 accumulation cannot be fully throughput normalized to FXP6 due to impossibly high and low resulting voltage requirements, respectively. As a result of throughput normalization, OR n configurations can use lower clock frequency, and consume lower power than other SC configurations with similar area. We use an analytical model that takes the array size and layer parameters, such as input and filter size, number of filters etc., and calculates both the number of compute iterations as well as memory accesses required to process a given layer, assuming output stationary dataflow. We focus on convolutional layers, as they dominate runtime and energy for all explored models. Our model assumes the memory bandwidth is provisioned such that computation is never stalled and that computation has maximum valid utilization. We use the clock period, stream length, and iteration count estimate the latency of each model's inference. We estimate inference energy using the latency, synthesized design power, memory access count, and energy obtained using the CACTI 6.5 tool [28].

| Reference | Romaszkan, SSCL 21 | Li, TCAD 23 |
|---|---|---|
| Architecture | SC, 8-bit | SC, OR-3, 8-bit |
| Tech Node | 14nm 0.6V | 28nm 0.6V |
| Clock Frequency | 500MHz | 450MHz |
| Peak Energy Efficiency | 9.4 TOP/S/W | 25 TOP/S/W |
| Peak Throughput | 150 GOPS | 3.1 TOPS |
| Metric 3:TinyConv ExT (Target: 9uJ x 24us) | 5.8uJ x 360us | 4uJ x 5.2us |
| Metric 4:VGG16 ExT (Target: 1.5mJ x 10ms) | Doesn't Support | 2.8mJ x 3.4ms |

**Figure 57 - Performance Summary of Digital SC Accelerator using OR-1 Accumulation and OR-3 Accumulation**

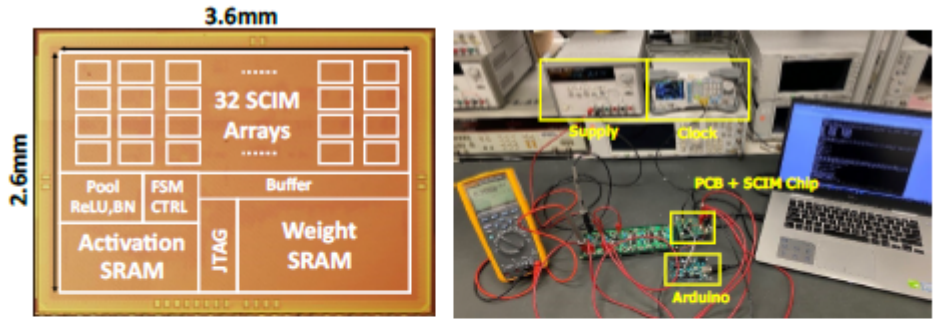
### 4.3.3 SCIM Deep Learning Accelerator

Stochastic Computing can achieve massive parallelism on chips to reduce the off-chip memory access. Stochastic Compute-In-Memory (SCIM) further improve the energy efficiency by combining the benefits of SC and Compute-In-Memory (CIM) architecture. The prototype chip is designed and fabricated in 65nm CMOS technology. The core area of the chip is 9.36 $mm^2$. A die photo is shown in Figure.

The activation and weight SRAM are located at the bottom of the floorplan. 32 SCIM cores contain 260kb in-memory storage and 520Kb of MAC units in total, which achieves 130Kb/mm2 MAC density. The nominal voltage of the 65nm process is 1V, but the chip is working under a wide range of voltages and down to 0.7V with no errors due to the digital compute property of SC. The clock frequency is 5MHz. It is limited by the on-chip power delivery structure and could achieve a much higher frequency and throughput. The energy efficiency of the basic MVM task is measured over different supply voltages and frequencies. The highest energy efficiency of 7.96TOPS/W is achieved at 0.7V and 3.2MHz.

CNNs for MNIST and CIFAR-10 datasets are demonstrated on the chip and the performance is summarized in Figure . We test a small 4-layer CNN because we want to keep all parameters on the

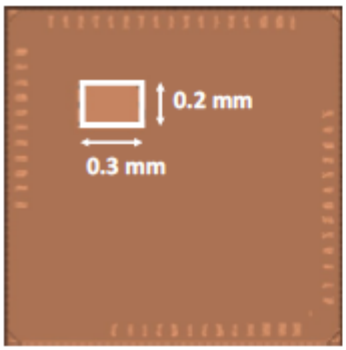
chip during the operation so that measurement of energy consumption can include all parts. The neural network parameters are trained to account for the errors in SC. Before training, each layer is mapped to the accelerator hardware and the LFSR seed of the SNG for each input and weight are recorded. The training algorithm uses the LFSR seed to match the exact bit stream generated on the chip. Activation and weight both have 8-bit precision. The input image and weight parameters of all layers are loaded to the on-chip SRAM. Configuration bits for each layer's dimension and different options are loaded to configuration registers. Once the computation starts, all the intermediate results remain on the chip until the final classification results are stored in the output SRAM. The CNNs for MNIST dataset achieves state-of-the-art accuracy of 99.1%, which is close to the training accuracy. Due to small size of each layer, the peak macro utilization is only 5.1 % and the peak system energy efficiency is 0.45 TOPS/W. CNNs for CIFAR-10 dataset achieves accuracy of 73.5%, compared to the training accuracy of 75%. Since the neural network used for CIFAR-10 has a larger size, the macro utilization is improved to 31.3%. Because the macro is severely underutilized, the energy efficiency and latency are degraded and does not reflect the peak performance of the accelerator.




Measurement Result Summary (MNIST, CIFAR10, MVM)

| Dataset | MNIST | CIFAR-10 | MVM |
|---|---|---|---|
| Network Topology | Con (5x5)x6<br>Avg Pool 2x2<br>Conv (5x5)x16<br>Avg Pool 2x2<br>FC 25 x 120<br>FC 120 x 64<br>FC 64 x 10 | Conv (5x5)x32,<br>Avg Pool 2x2<br>Conv (5x5)x32,<br>Avg Pool 2x2<br>Conv (5x5)x64,<br>Avg Pool 2x2<br>FC 256x10 | Matrix:<br>32x256<br><br>Vector:<br>1x256 |
| Bit Precision | 8bit Weight, 8bit Act, | | |
| Training Accuracy | 99.2% | 75% | N/A |
| Measured Accuracy | 99.1% | 73.5% (Top-1), 98.4%(Top-5) | N/A |
| Peak Hardware Utilization | 5.1% | 31.3% | 100% |
| VDD/CLK | 0.8V/4MHz | | |
| Energy/Classification | 18.3μJ | 35.6μJ | N/A |
| Peak System Energy Efficiency* | 0.45TOPS/W | 1.84TOPS/W | 5.75TOPS/W |
| Latency | 5.85msec/frame | 11.5msec/frame | 1usec |

**Figure 58 - CNN Demonstration and Measurement Result of the SCIM Accelerator Chip**




14nm SCIM Macro Chip

| | 14nm Macro Test Chip | 65nm Accelerator Chip |
|---|---|---|
| SCIM Macro Size | 32 x 32 | 32 x 256 |
| Number of Macro | 16 | 32 |
| Number of 1b SC MAC | 32.7K | 542.3K |
| MAC Density | 850 Kb/mm$^2$ | 130 Kb/mm$^2$ |
| Supply Voltage | 0.65 V | 0.7 V |
| Maximal Frequency | 130MHz | 5MHz |
| SC Stream Length | 32 | 64 or 256 (with avg pool) |
| Equivalent Precision | 5b | 6b or 8b (with avg pool) |
| System Energy Efficiency | 60 TOPS/W | 7.96 TOPS/W |
| Macro Energy Efficiency | 258 TOPS/W | 20 TOPS/W |

**Figure 59 - 14nm SCIM Macro Chip and Comparison between 14nm and 65nm Chips**

Another test chip is fabricated in 14nm to characterize the performance of the SCIM macros. The summary of the chip in comparison with 65nm accelerator chip is shown in Figure . It has 16 SCIM macros of 32x32 array. The bit cell and sense amplifier design are the same as 65nm chip. Each row performs a 32-long dot product and generates 2 stochastic output bits (OUTP/OUTN). The 16 SCIM macros generate 32 stochastic bits for a given MAC operation, which is equivalent to 5b computation precision. The macro's outputs are directly accumulated by the parallel counter without row mux and accumulator. The measured energy efficiency is 258 TOP/S/W at 0.65V supply.

We have evaluated the performance of the SCIM accelerator if scaled to 14nm and run larger deep learning models based on the following assumptions: 1) The SCIM macro's array size is scaled from 32x256 to 32x81 and the loss of throughput is compensated by have 3x more number of SCIM macros, 2) The dataflow and on-chip digital controller is the same as the 65nm design, 3) The on-chip SRAM is increased to support storing the parameters of VGG-Net. The performance of running both a small 4-layer and a large 16-layer neural network are shown below. The table shows the macro utilization, energy consumption and latency of each layer. The accelerator's energy-latency product is 0.78 uJ x 7.76 us for a 4-layer CNN and 515uJ x 5.16ms for a 16-layer VGGNet.

TinyConv CIFAR10 Performance

| Layer | Number of Op (G) | Utilization | Energy (uJ) | Latency(us) |
|---|---|---|---|---|
| 1 | 0.0049152 | 0.185 | 0.88 | 4.42 |
| 2 | 0.0131072 | 1 | 0.43 | 2.18 |
| 3 | 0.0065536 | 1 | 0.21 | 1.09 |
| 4 | 0.00032768 | 1 | 0.01 | 0.05 |
| Total | 0.024 | 0.796 | 1.55 | 7.75 |

VGG-16 ImageNet Performance

| Layer | Number of Op (G) | Utilization | Energy (uJ) | Latency(ms) |
|---|---|---|---|---|
| 1 | 0.173408256 | 0.11 | 2.89 | 0.028 |
| 2 | 3.699376128 | 1 | 61.65 | 0.61 |
| 3 | 1.849688064 | 1 | 30.82 | 0.30 |
| 4 | 3.699376128 | 1 | 61.65 | 0.61 |
| 5 | 1.849688064 | 1 | 30.82 | 0.30 |
| 6 | 3.699376128 | 1 | 61.65 | 0.61 |
| 7 | 3.699376128 | 1 | 61.65 | 0.61 |
| 8 | 1.849688064 | 1 | 30.82 | 0.30 |
| 9 | 3.699376128 | 1 | 61.65 | 0.61 |
| 10 | 3.699376128 | 1 | 61.65 | 0.61 |
| 11 | 0.924844032 | 1 | 15.41 | 0.15 |
| 12 | 0.924844032 | 1 | 15.41 | 0.15 |
| 13 | 0.924844032 | 1 | 15.41 | 0.15 |
| 14 | 0.205520896 | 1 | 3.42 | 0.03 |
| 15 | 0.033554432 | 1 | 0.55 | 0.005 |
| 16 | 0.008192 | 1 | 0.13 | 0.0013 |
| Total | 30.94052864 | 0.944375 | 515.67 | 5.15 |

| DARPA FRANC | | | State of the Art |
|---|---|---|---|
| Reference | Yang, ASSCC 22 | | Jia, ISSCC 21 |
| | Measured | Based on Macro Measurement | |
| Architecture | SCIM, 8-bit | SCIM, 8-bit | CIM 8-bit |
| Tech Node | 65nm, 0.7V | 14nm 0.6V | 14nm, 0.8V |
| Clock Frequency | 5MHz | 600MHz | 200 MHz |
| Peak Energy Efficiency | 7.96 TOP/S/W | 60 TOP/S/W | 30 TOP/S/W |
| Peak Throughput | 37.6GOPS | 4.7 TOPS | 3 TOPS |
| Metric 3:TinyConv ExT (Target: 9uJ x 24us) | 35.6uJ x 11.5ms | 1.6uJ x 7.6us | Not Measure |
| Metric 4:VGG16 ExT (Target: 1.5mJ x 10ms) | Doesn't Support | 0.52mJ x 5.2ms | 1.3mJ x 6.8ms |

**Figure 60 Performance of SCIM Accelerator on TinyConv and VGG-16 with Breakdown in each Layer**

*Summary of the performance and comparison with other state of the art.*

## 4.4 Meeting Program Metrics: A Summary

For Metric 1, we demonstrated a read Energy-Delay Product (EDP) of 2.1 fJ × 0.4 nsec. The obtained performance is better than the EDP targeted for Metric 1 (1 fJ × 1 nsec). This was achieved by designing a Magnetic-to-Digital Converter (MDC) in 28nm that optimizes the read EDP. Accurate simulations were performed that considered layout parasitic and used a validated VerilogA compact model of the Voltage-Controlled Magnetic-Tunnel-Junction (VC-MTJ). Furthermore, we demonstrated the first ever CMOS integrated VC-MTJ based magnetic RAM circuit (VCMRAM) in silicon. This was achieved by taping out memory chips of several sizes in TSMC 180nm. The 180nm process was selected due to its low cost. The VC-MTJ devices were built on top of the TSMC chips through our collaboration with ITRI, with several iterations to optimize the device properties. We were able to verify functionality with a 66% yield, which is a respectable yield for a first integration.
For metric 2, we have demonstrated a true random number generator using VC-MTJ in a 65nm chip with MTJ write/read circuit and bias correction algorithm. The random numbers are able to pass the NIST tests. The energy consumption from the VC-MTJ is expected to be 17fJ/b, which is close to the 10fJ/b target.

For metric 3 and 4, we have built deep learning accelerators based on digital stochastic computing logic and custom stochastic compute-in-memory macros. The performance of digital

SC with OR3 accumulation is 4uJx5.2us for metric 3 and 2.8mJ x 3.4ms for metric 4. The SCIM accelerator scaled to 14nm is expected to achieve 1.6uJ x 7.6us for metric 3 and 0.52mJ x 5.2ms for metric 4.

Both meet the performance target.

### 4.5 Technology Transfer, Publications, and Future

The VC-MTJ material stack and fabrication procedure technology that was developed at UCLA's laboratories has been successfully reproduced in a semi-commercial foundry namely, Industrial Technology Research Institute (ITRI), Taiwan. Furthermore, as discussed in the previous sections, ITRI has successfully integrated the VC-MTJ based MeRAM on a processed TSMC wafer. Potential collaborations with other, more commercial, and US-based foundries to transfer this technology have been explored. However, informal feedback indicated a reluctance driven primarily by the high investment cost in adopting a new memory fabrication technology.

Potential adoption of the MeRAM technology by other foundries will continue to be explored by the performers. Given that excellent MeRAM performance (e.g., much lower write energy than STT-MRAM) was demonstrated even in an old technology node, the performers are hopeful. An opportunity to integrate the VC-MTJ in a more advanced technology node such as TSMC 55nm or TSMC 28nm (ITRI has indicated such capability), may further advance this cause. High cost fueled by full processed wafer procurement and VC-MTJ fabrication expense remains the primary obstacle.

The stochastic computing technology has seen better transition success. Specifically, in collaboration with Northrop Grumman and as part of the DARPA FRANC/ERI-DA program, the performers have successfully demonstrated an ML accelerator that enables very low latency object tracking application. The low latency object tracking application is expected to be of particular interest to commercial (e.g., for tracking objects in industrial assembly lines) and military use (e.g., in obstacle avoidance in drones). The performing organization has filed a preliminary patent application based on some aspects of stochastic computing.

In addition to these technology transition efforts, the methods and accomplishments of this project have been disseminated to the research community through several conference and journal publications. Published and accepted works are listed below excluding those presented at GoMACTech conferences.


1. A. Lee and K. -L. Wang, "Full Memory Encryption with Magnetoelectric In-Memory Computing," 2019 International Symposium on VLSI Technology, Systems and Application (VLSI-TSA), Hsinchu, Taiwan, 2019, pp. 1-2, doi: 10.1109/VLSI-TSA.2019.8804703.
2. A. Lee, D. Wu and K. L. Wang, "Torque Optimization for Voltage-Controlled Magnetic Tunnel Junctions as Memory and Stochastic Signal Generators," in IEEE Magnetics Letters, vol. 10, pp. 1-4, 2019, Art no. 4107604, doi: 10.1109/LMAG.2019.2944805.
3. W. Romaszkan, T. Li, T. Melton, S. Pamarti and P. Gupta, "ACOUSTIC: Accelerating Convolutional Neural Networks through Or-Unipolar Skipped Stochastic Computing," 2020 Design, Automation & Test in Europe Conference & Exhibition (DATE), Grenoble, France, 2020, pp. 768-773, doi: 10.23919/DATE48585.2020.9116289.

4. W. Romaszkan, T.Li, J.Yang, A.Lee, D.Wu, A.Razavi, T.Melton, K.Wang, S.Pamarti , P.Gupta,, “Machine Learning at the Edge Using Spintronic Stochastic Computing” in Government Microcircuit Applications and Critical Technology Conference (GOMACTech), 2020
5. A. Lee, B. Dai, D. Wu, H. Wu, R. N. Schwartz, and K. L. Wang, “A thermodynamic core using voltage-controlled spin–orbit-torque magnetic tunnel junctions,” Nanotechnology, vol. 32, no. 50, p. 505405, Oct. 2021, doi: 10.1088/1361-6528/abeb9b.
6. T. Li, W. Romaszkan, S. Pamarti and P. Gupta, "GEO: Generation and Execution Optimized Stochastic Computing Accelerator for Neural Networks," 2021 Design, Automation & Test in Europe Conference & Exhibition (DATE), Grenoble, France, 2021, pp. 689-694, doi: 10.23919/DATE51398.2021.9473911.
7. J. Yang et al., "A Calibration-Free In-Memory True Random Number Generator Using VoltageControlled MRAM," ESSDERC 2021 - IEEE 51st European Solid-State Device Research Conference (ESSDERC), Grenoble, France, 2021, pp. 115-118, doi: 10.1109/ESSDERC53440.2021.9631784.
8. J.Yang, D.Wu, A.Razavi, P.Gupta, K.Wang, S.Pamarti, “A Robust and Calibration-Free True Random Number Generator Using Voltage-Controlled Magnetic Tunnel Junction” in Government Microcircuit Applications and Critical Technology Conference (GOMACTech), 2021
9. A. Lee, "Next-generation AI: From Algorithm to Device Perspectives", Ph.D dissertation, University of California, Los Angeles, 2021 [Online] (Chapter 4)
10. W. Romaszkan, T. Li, R. Garg, J. Yang, S. Pamarti and P. Gupta, "A 4.4–75-TOPS/W 14-nm Programmable, Performance- and Precision-Tunable All-Digital Stochastic Computing Neural Network Inference Accelerator," in IEEE Solid-State Circuits Letters, vol. 5, pp. 206-209, 2022, doi: 10.1109/LSSC.2022.3200064.
11. J. Yang, T. Li, W. Romaszkan, P. Gupta and S. Pamarti, "A 65nm 8-bit All-Digital StochasticCompute-In-Memory Deep Learning Processor," 2022 IEEE Asian Solid-State Circuits Conference (A-SSCC), Taipei, Taiwan, 2022, pp. 10-11, doi: 10.1109/ASSCC56115.2022.9980613.
12. J.Yang, T.Li, W. Romaszkan, P.Gupta, S.Pamarti, “A 1.84-7.96TOPS/W 65nm DAC/ADCFree Stochastic Compute-In-Memory CNN Accelerator with 8-bit Precision and Robust Operation Under 0.7V-1.05V Supply” in Government Microcircuit Applications and Critical Technology Conference (GOMACTech), 2022.
13. W. Romaszkan, T. Li and P. Gupta, "SASCHA—Sparsity-Aware Stochastic Computing Hardware Architecture for Neural Network Acceleration," in IEEE Transactions on ComputerAided Design of Integrated Circuits and Systems, vol. 41, no. 11, pp. 4169-4180, Nov. 2022, doi: 10.1109/TCAD.2022.3197503.
14. A. Lee et al., "Low-Energy Shared-Current Write Schemes for Voltage-Controlled Spin-OrbitTorque Memory," in IEEE Transactions on Electron Devices, vol. 70, no. 2, pp. 478-484, Feb. 2023, doi: 10.1109/TED.2022.3228831.
15. T. Li , S. Li and P.Gupta, "Training Neural Networks for Execution on Approximate Hardware" arXiv:1612.06830 [cs], Dec. 2016. arXiv:2304.04125 [cs.LG]
16. W. Romaszkan "Efficient Machine Learning Acceleration at the Edge", Ph.D dissertation, University of California, Los Angeles, 2023

17. V. K. Jacob, J. Yang, H. He, P. Gupta, K. L. Wang and S. Pamarti, "A Nonvolatile Computein-Memory Macro Using Voltage-Controlled MRAM and In Situ Magnetic-to-Digital Converter," in IEEE Journal on Exploratory Solid-State Computational Devices and Circuits, vol. 9, no. 1, pp. 56-64, June 2023, doi: 10.1109/JXCDC.2023.3258431.
18. T. Li, W. Romaszkan, S. Pamarti and P. Gupta, "REX-SC: Range-Extended Stochastic Computing Accumulation for Neural Network Acceleration," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, doi: 10.1109/TCAD.2023.3284289.
19. H. Suhail, J. Yang, H. He, K. L. Wang, S. Pamarti, "Analytical Array-Level Comparison of Read/Write Performance Between Voltage Controlled-MRAM and STT-MRAM", 2023 Midwest Symposium on Circuits And Systems
20. T. Li, "Learned Approximate Computing for Machine Learning", Ph.D dissertation, University of California, Los Angeles, 2023 [Online]
21. H. Suhail, H.He, et al, "The First CMOS-Integrated Voltage-Controlled MRAM with 0.7ns Switching Time", 2023 International Electron Device Meeting
22. J.Yang, "Stochastic Compute-In-Memory Hardware Accelerator for Intelligent Edge Devices," Ph.D dissertation, University of California, Los Angeles, 2023. [Online]. Available in 2024. (Chapter2, Chapter 3)
23. Haris Suhail, Haoran He, Jiyue Yang, Vinod K. Jacob, Qingyuan Shu, Puneet Gupta, Kang L. Wang, Sudhakar Pamarti "The First CMOS-Integrated Voltage-Controlled MRAM with sub1ns Switching Time" [Accepted, to be presented] Government Microcircuit Applications and
Critical Technology Conference (GOMACTech), 2024.

# 5 CONCLUSION

We adopted two novel approaches to address the memory bottleneck problem in traditional computing systems.

The first approach involved a new voltage controlled magnetic tunneling junction (VC-MTJ). We developed a material stack that achieved the highest Voltage Controlled Magnetic Anisotropy (VCMA) coefficient ever reported under CMOS compatible conditions. With the help of the Industrial Technology Research Institute (ITRI), Taiwan, we successfully achieved the 1st ever CMOS-integrated MeRAM – up to 64kb arrays of 100nm diameter MTJ devices. We demonstrated proper memory operation with ~65% yield, and 0.7ns write latency which is far better than state of the art in other magnetic memories. Since cost and technology availability constraints forced us to integrate in an old (0.18um) CMOS node, we additionally developed appropriate low energy, low latency memory access circuitry and verified it in a layout parasitic aware simulation using a MTJ compact model derived from measured device properties. We demonstrated the 1st ever voltage controlled MTJ based true random number generator (TRNG) under two scenarios: (a) MTJs on a separate die, wire bonded to a 65nm die with TRNG access circuitry, and (b) MTJs integrated on 0.18um CMOS. Even in 0.18um CMOS, we demonstrated better throughput than state-of-the-art STT-MRAM based TRNGs in 28nm CMOS. Furthermore, in case (a), we have showed that the TRNG passes all NIST randomness tests using a simple bias correction circuit.

The second approach developed a stochastic computing (SC) framework where numbers are represented as fractions of 1s in a random binary stream. The SC framework employs ultracompact MAC hardware which enables massive parallelization of on-chip computation in direct contrast to the traditional von Neumann architecture. This reduces data movement by an order of magnitude with substantial improvements in both energy and latency. We developed number representation and SC-aware training techniques to overcome the inference accuracy gap between SC and conventional fixed-point computation. Where previous researchers believed that SC needed extremely long bit streams to achieve parity with fixed-point, we demonstrated results to the contrary and showed EDP benefits at the same accuracy for the CIFAR-10 dataset and small and medium sized CNNs. We developed compute-in-memory (CIM) techniques with SC and showed that energy and area computational efficiency metrics that are better than CIM can be achieved without the need for the analog to digital converters that plague CIM art. We have demonstrated the benefits of SC-in-Memory (SCIM) architectures over state-of-the-art in using machine learning (ML) accelerator IC prototypes in 65nm and 12nm CMOS.

In terms of program metrics, for a 4-layer CNN, we achieved EDPs of 4uJ x 5.2us and 1.6uJ x 7.6us respectively for our SC and SCIM accelerator designs, both of which are better than the target for metric #3. Similarly, for VGGNET-18, we achieved EDPs of 2.8mJ x 3.4ms and 0.52mJ x 5.2ms for our SC and SCIM accelerator designs respectively, both of which are better than the target for metric #4.

Impact, Transitions, and Future: The MeRAM is a candidate memory for the future. The technology has been successfully reproduced in a semi-commercial foundry (ITRI) and we hope that other foundries will follow. In addition, the high throughput, low energy TRNG capability developed should find use in many applications of commercial and military interest. The SCIM

technology has been successfully applied, in collaboration with Northrop Grumman (as part of FRANC ERI-DA), to a problem of their interest – high speed object tracking. We except adoption in other applications that can benefit from the low latency and high area computational efficiency offered by this technology. The various techniques developed and accomplishments of this project have also been shared with researchers in the respective IEEE communities in the form of several publications.

# 6 REFERENCE


[1] T. Maruyama *et al.*, “Large voltage-induced magnetic anisotropy change in a few atomic layers of iron,” *Nature Nanotech*, vol. 4, no. 3, Art. no. 3, Mar. 2009, doi: 10.1038/nnano.2008.406.

[2] C. Grezes *et al.*, “Ultra-low switching energy and scaling in electric-field-controlled nanoscale magnetic tunnel junctions with high resistance-area product,” *Applied Physics Letters*, vol. 108, no. 1, Jan. 2016, doi: 10.1063/1.4939446.

[3] Y. C. Wu *et al.*, “Deterministic and Field-Free Voltage-Controlled MRAM for High Performance and Low Power Applications,” in *2020 IEEE Symposium on VLSI Technology*, Jun. 2020, pp. 1–2. doi: 10.1109/VLSITechnology18217.2020.9265057.

[4] R. Carpenter *et al.*, “Demonstration of a Free-layer Developed With Atomistic Simulations Enabling BEOL Compatible VCMA-MRAM with a Coefficient ≥100fJ/Vm,” in *2021 IEEE International Electron Devices Meeting (IEDM)*, Dec. 2021, p. 17.6.1-17.6.4. doi: 10.1109/IEDM19574.2021.9720579.

[5] L. Wei *et al.*, “13.3 A 7Mb STT-MRAM in 22FFL FinFET Technology with 4ns Read Sensing Time at 0.9V Using Write-Verify-Write Scheme and Offset-Cancellation Sensing Technique,” in *2019 IEEE International Solid-State Circuits Conference - (ISSCC)*, Feb. 2019, pp. 214–216. doi: 10.1109/ISSCC.2019.8662444.

[6] Q. Dong *et al.*, “A 1-Mb 28-nm 1T1MTJ STT-MRAM With Single-Cap Offset-Cancelled Sense Amplifier and In Situ Self-Write-Termination,” *IEEE Journal of Solid-State Circuits*, vol. 54, no. 1, pp. 231–239, Jan. 2019, doi: 10.1109/JSSC.2018.2872584.

[7] S. K. Mathew *et al.*, “2.4 Gbps, 7 mW All-Digital PVT-Variation Tolerant True Random Number Generator for 45 nm CMOS High-Performance Microprocessors,” *IEEE Journal of SolidState Circuits*, vol. 47, no. 11, pp. 2807–2821, Nov. 2012, doi: 10.1109/JSSC.2012.2217631.

[8] “An All-Digital Edge Racing True Random Number Generator Robust Against PVT Variations | IEEE Journals & Magazine | IEEE Xplore.” Accessed: Oct. 30, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/7422712

[9] “A true random number generator using time-dependent dielectric breakdown | IEEE Conference Publication | IEEE Xplore.” Accessed: Oct. 30, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/5986112

[10] “A Magnetic Tunnel Junction based True Random Number Generator with conditional perturb and real-time output probability tracking | IEEE Conference Publication | IEEE Xplore.” Accessed: Oct. 30, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/7047039

[11] K. Yang *et al.*, “A 28NM Integrated True Random Number Generator Harvesting Entropy from MRAM,” in *2018 IEEE Symposium on VLSI Circuits*, Jun. 2018, pp. 171–172. doi: 10.1109/VLSIC.2018.8502431.

[12] Y. Liu et al. “A stochastic computational multi-layer perceptron with backward propagation”. In: IEEE TC (2018).

[13] A. Ardakani et al. “VLSI implementation of deep neural networks using integral stochastic computing”. In: 2016 ISTC. 2016.

[14] H. Sim et al. “Scalable Stochastic-Computing Accelerator for Convolutional Neural Networks”. In: ASP-DAC 2017. 2017.

[15] A. Ren et al. “SC-DCNN : Highly-Scalable Deep Convolutional Neural Network using Stochastic Computing”. In: ASPLOS. 2017.

[16] S. R. Faraji et al. “Energy-Efficient Convolutional Neural Networks with Deterministic BitStream Processing”. In: DATE. 2019.
[17] H. Sim and J. Lee. “A New Stochastic Computing Multiplier with Application to Deep Convolutional Neural Networks”. In: (2017).
[18] J. A. Dickson et al. “Stochastic arithmetic implementations of neural networks with in situ learning”. In: IEEE ICNN 1993. 1993.
[19] H. Ichihara, S. Ishii, D. Sunamori, T. Iwagaki, and T. Inoue, “Compact and accurate stochastic circuits with shared random number sources,” ICCD, pp. 361–366, 2014.
[20] F. Neugebauer, I. Polian, and J. P. Hayes, “Building a better random number generator for stochastic computing,” in DSD, 2017, pp. 1–8.
[21] L. Lai, N. Suda, and V. Chandra, “CMSIS-NN: efficient neural network kernels for arm cortex-m cpus,” CoRR, vol. abs/1801.06601, 2018.
[22] Wojciech Romaszkan, Tianmu Li, Tristan Melton, Sudhakar Pamarti, and Puneet Gupta. 2020. ACOUSTIC: Accelerating Convolutional Neural Networks through Or-Unipolar Skipped Stochastic Computing. In 2020 Design, Automation Test in Europe Conference Exhibition (DATE). 768–773. https://doi.org/10.23919/DATE48585.2020.9116289
[23] G. Hu *et al.*, “Double spin-torque magnetic tunnel junction devices for last-level cache applications,” in *2022 International Electron Devices Meeting (IEDM)*, Dec. 2022, p. 10.2.1-10.2.4. doi: 10.1109/IEDM45625.2022.10019402.
[24] D. Wu, J. Li, R. Yin, H. Hsiao, Y. Kim, and J. S. Miguel, “uGEMM : Unary Computing Architecture for GEMM Applications,” 2020 ACM/IEEE 47th Annual International Symposium on Computer Architecture (ISCA), pp. 377–390, 2020, iSBN: 9781728146614.
[25] V. K. Chippa, S. Venkataramani, K. Roy, and A. Raghunathan, “StoRM: A Stochastic Recognition and Mining processor,” in Proceedings of the 2014 International Symposium on Low Power Electronics and Design, 2014, pp. 39–44, iSSN: 15334678.
[26] Won Ho Choi et al., "A Magnetic Tunnel Junction based True Random Number Generator with conditional perturb and real-time output probability tracking," 2014 IEEE International Electron Devices Meeting, San Francisco, CA, USA, 2014, pp. 12.5.1-12.5.4.
[27] K. Yang et al., "A 28NM Integrated True Random Number Generator Harvesting Entropy from MRAM," 2018 IEEE Symposium on VLSI Circuits, Honolulu, HI, USA, 2018, pp. 171-172. [28] W. Romaszkan, T. Li, and P. Gupta, “SASCHA—Sparsity-Aware Stochastic Computing Hardware Architecture for Neural Network Acceleration,” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 41, no. 11, pp. 4169–4180, Nov. 2022. [Online].
Available: https://ieeexplore.ieee.org/document/ 9852757/
[29] N. Muralimanohar, R. Balasubramonian, and N. P. Jouppi, “CACTI 6.0 : A Tool to Model Large Caches,” HP laboratories, vol. 27, no. HPL- 2009-85, p. 28, 2009.
[30] J. Yang, T. Li, W. Romaszkan, P. Gupta and S. Pamarti, "A 65nm 8-bit All-Digital StochasticCompute-In-Memory Deep Learning Processor," 2022 IEEE Asian Solid-State Circuits Conference (A-SSCC), Taipei, Taiwan, 2022, pp. 10-11, doi: 10.1109/ASSCC56115.2022.9980613.
[31] T. Li, W. Romaszkan, S. Pamarti and P. Gupta, "REX-SC: Range-Extended Stochastic Computing Accumulation for Neural Network Acceleration," in IEEE Transactions on ComputerAided Design of Integrated Circuits and Systems, doi: 10.1109/TCAD.2023.3284289.
[32] H. Jia et al., "15.1 A Programmable Neural-Network Inference Accelerator Based on Scalable In-Memory Computing," 2021 IEEE International Solid-State Circuits Conference

(ISSCC), San Francisco, CA, USA, 2021, pp. 236-238, doi: 10.1109/ISSCC42613.2021.9365788.

# 7 LIST OF SYMBOLS, ABBREVIATIONS, AND ACRONYMS

| | | |
|---|---|---|
| **BL** | – | Bit Line |
| **ITRI** | – | Industrial Technology Research Institute |
| **MTJ** | – | Magnetic Tunnel Junction |
| **RNG** | – | Random Number Generator |
| **SL** | – | Source Line |
| **SNG** | – | Stochastic Number Generator |
| **STT** | – | Spin Transfer Torque |
| **TRNG** | – | True Random Number Generator |
| **VC-MRAM** | – | Voltage Controlled Magnetic Random Access Memory |
| **VC-MTJ** | – | Voltage Controlled Magnetic Tunnel Junction |
| **VCMA** | – | Voltage Controlled Magnetic Anisotropy |
| **WL** | – | Word Line |
| **SC** | – | Stochastic Computing |
| **CIM** | – | Compute In Memory |
| **M**L | – | Machine Learning |
| **CNN** | – | Convolutional Neural Network |