Advanced Spintronic Memory and Logic
For Non-Volatile Processors

Robert Perricone*, Ibrahim Ahmed†, Zhaoxin Liang†, Meghna G. Mankalale†,
X. Sharon Hu*, Chris H. Kim†, Michael Niemier*, Sachin S. Sapatnekar†, and Jian-Ping Wang†

*Department of Computer Science and Engineering, University of Notre Dame
Notre Dame, IN 46556, USA, Email: {rperrico,shu,mniemier}@nd.edu}
†Department of Electrical and Computer Engineering, University of Minnesota
Minneapolis, MN 55455, USA, Email: {ahmed589,zxliang,manka018,chriskim,sachin,jpwang}@umn.edu

Abstract—Many ultra-low power Internet of things (IoT) systems may be powered by energy harvested from ambient sources (e.g., solar radiation, thermal gradients, and WiFi). However, these energy sources can vary significantly in terms of their strengths and on/off patterns. For volatile systems, the intermittent nature of the energy sources necessitates the use of backup/recovery schemes to guarantee computational correctness and forward progress, which incur performance, area and energy overhead. Non-volatile (NV) processors based on spintronic devices, such as Spin-Transfer Torque (STT) memory and All-Spin-Logic (ASL), are more attractive alternatives. These NV devices are capable of achieving forward progress without relying on backup/recovery schemes. This work establishes a general framework for evaluating NV device-based processors for energy harvesting applications. Results demonstrate that NV spintronic processors can achieve significant energy savings (up to $83\times$) versus a hybrid CMOS (computation) and STT-RAM (backup) implementation.

I. INTRODUCTION

With the rapid growth of the internet of things (IoT), demands for battery-less systems are ever increasing. There are numerous examples of systems that might benefit from information processing hardware that is powered by ambient energy sources. While many such systems may offer new opportunities and capabilities for personal entertainment, self-powered computational systems have obvious societal benefits when deployed for medical monitoring, environmental sensing, etc. As such, there are a growing number of research efforts targeting battery-less systems that harvest energy from solar, vibration, WiFi, and radio frequency (RF) sources [1], [2].

That said, researchers face numerous challenges when architecting systems that must rely solely on ambient sources of energy. Most energy sources exhibit low conversion efficiency, i.e., a small fraction of the total harvested power can actually be used. Moreover, many ambient sources are simply not reliable, e.g., ambient RF or WiFi power varies significantly based on the power source, frequency, distance from the transmitter, environmental obstacles, etc. [3].

To combat these challenges, non-volatile (NV) memory and logic technologies are of growing interest. In the simplest form, NV storage elements offer a way to periodically back up (or "checkpoint") processor states. Thus, if an ambient energy source becomes insufficient to power the system that processes information, the most recently saved processor states can be restored from NV storage. Assuming the energy and delay associated with backup/recovery is sufficiently low, it is possible to make incremental progress toward completing the computational task that the processor is responsible for [4].

Processors implemented with NV (e.g., spintronic) logic elements offer another alternative for systems powered by harvested energy since processor states are automatically retained upon power loss. However, studies of existing spintronic logic based processors (e.g., [5], [6], [7]) have mainly focused on performance and energy in the context of high-performance computing applications, where it seems challenging for spintronic logic to compete with CMOS. Using all-spin logic (ASL) as a representative example, at high frequencies, power consumption can be extremely high. Even at lower frequencies (i.e., 25 MHz) the power associated with ASL can be more than an order of magnitude higher than what is projected for CMOS (assuming different processor power states) [5]. Nonetheless, material advances (e.g., Heusler based ASL) and relaxing device retention times open the door for spintronic logic devices to exhibit lower power than CMOS counterparts [5].

In systems powered by harvested energy, spintronic logic devices offer several distinctive advantages. First, spintronic logic devices are inherently NV. This can eliminate the need for backup/recovery to/from NV memory, as well as the energy and delays associated with the backup and recovery operations. Second, processors in such resource constrained environments typically have very low clock rates (e.g., [4] considers a clock rate of just 8 KHz for NV processors powered by WiFi sources). Lower clock rates should help to reduce logic power dissipation when employing spintronic logic. Indeed, our preliminary projections for an in-order ASL processor (design from [4]) based on Heusler alloys indicates that, depending on the reliability of the power supply, the energy/instruction for the ASL processor could be between $3\times$ and $30\times$ better than a CMOS-based NV processor (NVP).

In the rest of this paper, we first review several emerging spintronic technologies that could enable battery-less systems (Sec. II). We then discuss our benchmarking framework for evaluating these spintronic technologies at the architectural level (Sec. III). Finally, we present several case-study results obtained by our framework through analyzing a battery-less non-pipelined processor (originally introduced in [4]) based on the state-of-the-art spintronic devices (Sec. IV).

II. BACKGROUND

In this section, we provide a brief overview of the spintronic devices and the NV processor utilized in the case study to be discussed in Sec. IV.

A. Overview of Select Spintronic Devices

We have selected four spintronic devices – STT-RAM, SHE-RAM, ASL, and CoMET – for our evaluation work as they represent near, mid and long term developments in the research spectrum of spintronic devices. Below, we discuss the basic operating principles of these spintronic devices.

STT-RAM: A Spin Transfer Torque RAM (STT-RAM) bit-cell consists of an access transistor and a Magnetic Tunnel
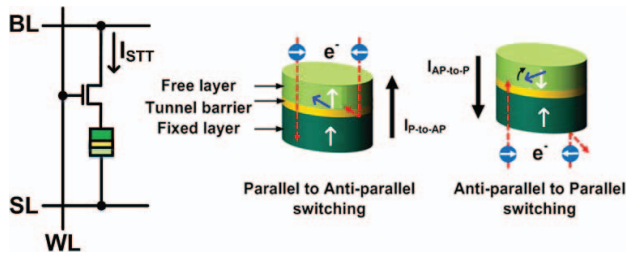
972

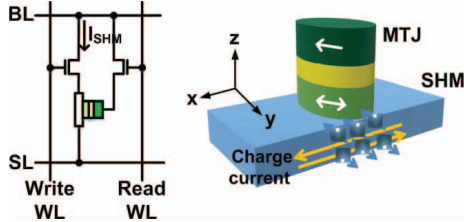Fig. 1: STT-RAM bit cell and write operation [5].

Fig. 2: SHE-RAM bit cell and write operation [5].

Fig. 3: Conceptual diagram of an ASL-based inverter [5].

(a) Pair of cascaded CoMET inverters

(b) CoMET–based three-input majority (MAJ3) gate

Fig. 4: Illustrations of two elementary CoMET circuits: (a) inverter and (b) three-input majority gate.

Junction (MTJ) as shown in Fig. 1. The write operation is accomplished by applying a write current in one of two directions. The direction of the write current through the access transistor changes the state (which indicates a '1' or '0') of the MTJ device. The state of the MTJ is changed as this charge current becomes spin polarized based on the pinned layer magnetization and exerts a torque on the free layer as shown in Fig. 1. The read operation is performed using a small read current through the access transistor to sense the resistance of the MTJ. STT-RAM chips with sufficient capacities have been experimentally demonstrated (e.g., [8], [9]).

SHE-RAM: Spin Hall Effect RAM (SHE-RAM) consists of two access transistors, an MTJ, and a spin Hall metal (SHM) structure as shown in Fig. 2. The write operation is performed with a bidirectional current through the SHM. Here, the charge current becomes spin polarized, traversing along the direction of charge current, as shown in Fig. 2. The charge current is along the x-axis and the spin current is polarized along the y-axis. The spin current along the z-axis exerts a torque on the MTJ free layer sitting on top of the SHM. SHE can switch an in-plane MTJ naturally, but requires an external in-plane field to switch a perpendicular anisotropy MTJ. The read operation is performed using the MTJ access transistor which is identical to the STT-RAM read operation. Spin Hall effect has been predicted to be more efficient at switching the MTJ than the spin transfer torque mechanism [10], [11], [12]. Spin-torque switching by SHE has been experimentally demonstrated (e.g., [13], [14]).

ASL: An elementary All-Spin Logic (ASL) [15] gate consists of input and output magnets that have two stable magnetization states and are connected through a channel as shown in Fig. 3. A non-zero voltage pulse applied to V_{supply} induces a spin current (I_{supply}) that passes through the input magnet, which results in spin-polarized electrons in the channel. The accumulated spins produce the channel spin current (I_{spin}), which transfers angular momentum to the output magnet. If I_{spin} exceeds a certain threshold on the output magnet, the magnetization state of the output magnet is toggled. Depending on the polarity of V_{supply}, a $COPY$ or $INVERT$ operation is accomplished. Majority gates can be readily realized with ASL. A key factor here is that the spin current propagation distance is limited by the length and material of the channel.
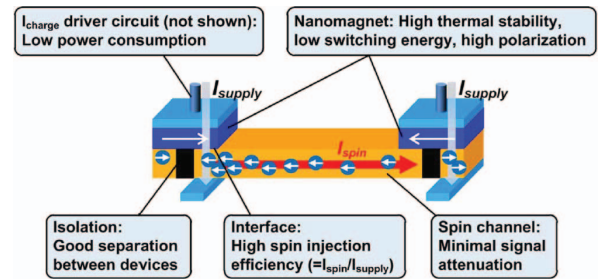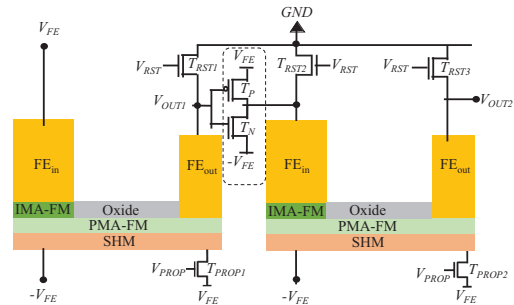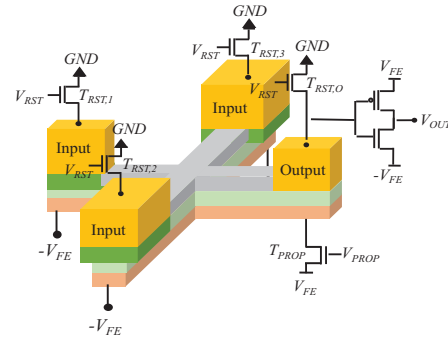
This material property is known as the "spin diffusion length" and it is an important consideration for processor design as it necessitates the placement of interconnect buffers.

CoMET: Composite input MagnetoElectric-based logic Technology (CoMET) [16] is a fast, low-energy spintronics-based logic device concept. Fig. 4a shows two cascaded CoMET inverter stages, and we briefly illustrate the principle of operation of the first stage. A voltage applied on an input ferroelectric (FE) capacitor nucleates a domain wall (DW) through the magnetoelectric (ME) effect [17]. A composite ferromagnet (FM) structure with an in-plane magnetic anisotropy (IMA) layer is placed above the perpendicular magnetic anisotropy (PMA)–FM channel to enable fast, energy-efficient nucleation in the PMA–FM at the input end. The DW is propagated to the output end of the FM channel, using a charge current applied to a layer of spin-Hall material placed under the PMA channel. The inverse–ME (IME) effect induces a voltage at the output, and a dual-rail inverter structure efficiently transmits this voltage to the input of the next gate. The device maps easily to a majority logic structure, as shown in Fig. 4b, where DWs from the three inputs compete to create the majority magnetization under the output. As before,

a voltage is induced at the output node using the IME, and is transmitted to the next stage through the dual-rail inverter. The device concept of CoMET has been proposed very recently and been shown through simulations to be viable when mapped to realistic material parameters. An experimental demonstration of the device will be explored in future work.

B. Related Work on NVPs

The use of spintronic memory devices (such as STT-RAM) in microprocessors has been previously studied for various purposes such as leveraging their zero leakage energy [7] and implementing hybrid CMOS+NV memory checkpointing architectures [4], [18]. NVPs based on ferroelectric flip-flops have also be proposed and evaluated [5], [6]. A recent work by Ma et al. [4] systematically studied a NVP based on STT-RAM cache as well as main memory. Existing works all consider a specific NV device in their respective NVPs. In this work, our goal is to evaluate the different emerging NV devices discussed in Sec. II in NVPs. In this section, we review our target NVP architecture from Ma et al. [4], which implements a 32-bit MIPS non-pipelined processor (NPP) comprised of CMOS and STT-RAM technology. We choose this specific processor as the authors have done a thorough job in analyzing energy savings by the NVP operating in an harvested energy environment.

The NPP presented in [4] runs at 8 kHz since it assumes a weak WiFi power source. The instruction memory and instruction cache are assumed to be ROM and a NV memory, respectively. The data memory and data cache are also implemented by NV memories with the data cache assuming a write-back policy that preserves dirty data on power outages. As the processor is non-pipelined, a single instruction characterizes the entire state of the processor. On power outages, architectural state is preserved by writing the program counter (PC) and register file (RegFile) to a NV memory.

The baseline design evaluated by Ma et al. involves a hybrid CMOS+STT-RAM design. CMOS is used to implement all logic and register memory components of the NPP. The processor is powered through a weak WiFi signal that charges a capacitor. When the energy of the capacitor drops below a certain threshold, the NPP backups the PC and RegFile to a separate STT-RAM memory. Once the energy of the capacitor exceeds a certain threshold, the NPP begins its recovery operation by restoring the CMOS-based PC and RegFile.

As note in [4], writing NV memory is an energetically expensive operation. Thus, to minimize the amount of data that needs to be written, three backup/recovery policies were studied in [4]. For brevity, we select the most energy efficient policy, i.e., the "On Demand Selective Backup" (ODSB) policy. The ODSB policy adds additional data bits to the RegFile to determine which lines of the RegFile have changed since the last power outage. Only the changed lines are written to the backup STT-RAM on a power outage.

To further explore the design space of the NPP, Ma et al. consider the intermittent nature of the power supply. Three instruction backup intervals (BIs) are examined, which represent the expected number of instructions to be completed before a backup event occurs. BIs of 1, 10, and 1000 instructions are examined, and their average energy per instruction (summation of computational and backup/recovery energy) is reported.

III. Design and Evaluation Methodology

In this section, we present our framework for benchmarking NVPs. We categorize NVPs into three types as shown in Fig. 5:
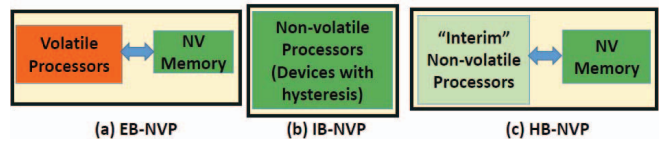

Fig. 5: Three types of NVPs: (a) NVP with explicit backup, (b) NVP with implicit backup, and (c) NVP with hybrid backup.

NVP with explicit backup (EB-NVP), NVP with implicit backup (IB-NVP), and NVP with hybrid backup (HB-NVP).

In EB-NVPs, processor states must be explicitly backed up to and restored from NV memory. Most existing NVPs belong to this category. IB-NVPs use NV devices to realize all the storage elements (i.e., all the flip-flops and latches) while the combinational circuits in the processor may or may not be implemented in NV devices. Thus, for an IB-NVP, processor states are automatically retained in the NV storage elements and no explicit backup/recovery is needed. For HB-NVPs, the retention time of the storage elements in the "interim" NVP cannot be treated as "infinitely" long as for IB-NVPs. Thus, NV memory is still needed. However, if the power outage time is shorter than the retention time of the "interim" NVP, no backup/recovery is needed. For many NV devices, reducing retention time could significantly reduce operating energy. Therefore, HB-NVP can provide an effective way to trade off operating energy with backup/recovery overhead.

In this work, we focus on NVP processors that rely on a simple drop-in replacement of MOSFET devices. Below, we first summarize a drop-in approach (Sec. III-A) from Kim et al. [5] that forms the foundation of our framework ((Sec. III-B). We then discuss the design and evaluation methodology of our benchmarking framework.

A. Drop-in Replacement Design Approach

To implement an NVP based on a von Neumann architecture and to exploit the spintronic devices discussed in Sec. II-A, a straightforward approach is to do a drop-in replacement, i.e., replacing basic CMOS circuit/logic elements with spintronic elements. The STT-RAM and SHE-RAM can be readily used as the memory elements for cache or main memory, while ASL and CoMET devices can readily implement combinational logic and flip-flops. With the drop-in replacement approach, many of the existing high-level design tools as well as architectural level techniques can be leveraged, which should significantly reduce development effort. Below we discuss the design and evaluation of an ASL-based IB-NVP, which can also be tuned to be a HB-NVP.

In our recent work, we have studied an ASL-based Intel Core i7 processor [5]. We leverage a drop-in approach to replace all CMOS-based pipeline logic and flip-flops by their equivalent ASL-based logic gates and storage elements (note that the cache is not considered). We have shown how ASL gates can be cascaded to form pipeline logic, how edge-triggered ASL flip-flops can be formed, and how the clock signal can be manipulated to achieve equivalent functionality to their CMOS counterparts. A key benefit to these designs is that an ASL flip-flop only requires 4 ASL devices whereas the equivalent CMOS flip-flop would require at least 20 transistors.

The ASL devices selected are $5 \text{ nm} \times 5 \text{ nm} \times 4 \text{ nm}$ PMA magnets connected via a copper channel with 10 year retention time [5]. A clock frequency of 25 MHz is used as higher frequencies lead to much higher power consumption. A single

pipeline stage consists of 20 logic gates along the critical path[1]. Thus, the target ASL device switching time can be computed as $1/(25 \text{ MHz} \times 20 \text{ gates}) = 2$ ns. The retention time and clock frequency make the design an IB-NVP. However, a HB-NVP could be achieved by reducing the retention time closer to the clock frequency, which would lower the switching energy of each ASL device. For example, the 25 MHz clock represents a cycle time of 40 ns. One could set the retention time to be closer to 40 ns to ensure non-volatility of the processor is preserved. The retention time of spintronic devices can be tuned through varying the material composition and/or size of the magnets. Once the design parameters are determined, the energy per cycle of the processor can be estimated.

With the drop-in replacement approach, we can uses the number of transistors per core to estimate the number of spintronic devices that would be needed to implement the equivalent core logic. Furthermore, the number of transistors is also used to estimate the number of buffer interconnects (ICs) that would be necessary for a given material (i.e., its spin diffusion length). Once the number of spintronic logic devices and buffer ICs are known, the energy per cycle of the processor can be computed as shown in Eq. 1.

$$\text{Total Energy} = N_{logic}^{spin} \times E_{logic}^{spin} + N_{IC}^{spin} \times E_{IC}^{spin} \quad (1)$$

In Eq. 1, N_{logic}^{spin} and N_{IC}^{spin} represent the number of spintronic logic devices and ICs, respectively. Similarly, E_{logic}^{spin} and E_{IC}^{spin} represent the switching energy of the logic and interconnect buffers, respectively.

To estimate the number of spintronic logic devices, we examine the number of devices needed to implement CMOS-based logic gates versus ASL-based logic gates. Like most spintronic devices, ASL-based logic gates are formed using majority logic. Compared to CMOS, ASL can implement equivalent logic gates using approximately 50% less devices. Therefore, the number of spintronic logic devices (N_{logic}^{spin}) can be estimated from the number of CMOS logic transistors (N_{logic}^{CMOS}) as shown in Eq. 2.

$$N_{logic}^{spin} = N_{logic}^{CMOS}/2 \quad (2)$$

As discussed in Sec. II-A, spintronic devices that use spin current to transfer data require buffer ICs to overcome attenuation of the spin signal. To determine the number of buffer ICs, two parameters are needed: (i) the spin diffusion length of the material and (ii) the number of logic gates in a the processor. The spin diffusion length (λ_{ch}) of a material determines how far a spin signal can propagate. Furthermore, as the number of buffer ICs increases, the switching energy of the processor also increases. Thus, it is preferable to limit the number of buffer ICs. This can be achieved by using materials with a larger λ_{ch}. Typical values of λ_{ch} range from 400 nm for copper to beyond 2 μm for graphene [5].

To estimate the number of logic gates per processor core, we assume that the average number of transistors per logic gate is 4. A typical spintronic device is assumed to have a gate pitch of 10 nm and an average fanout of 4. Collectively, these parameters can be utilized in a probability density function based on Rent's rule [19] to model the statistical distribution of wire lengths in a random logic block (see [20]).

Once the number of spintronic logic devices and buffer ICs have been determined, their respective switching energies need to be computed. This process involves simulating the target spintronic technology through a Landau-Lifshitz-Gilbert (LLG) solver. The LLG solver uses device material, geometric and operating parameters to determine the energy needed to cause the output device to switch. For brevity, we point the reader to the process discussed in [5] for more information.

B. NVP Evaluation Framework

In this subsection, we present our NVP benchmarking framework, referred to as `EvaNVP`. `EvaNVP`, illustrated in Fig. 6, builds on the drop-in replacement approach from Sec. III-A to estimate the total computation and backup/recovery energy of the NVP. The main idea behind `EvaNVP` is to integrate (i) the power supply patterns, (ii) the backup/recovery policies, and (iii) the processor performance/energy models to account for their collective impacts on the overall system energy. `EvaNVP` consists of four separate modules, and the details of the modules are summarized below.

Power Supply Profile Modeling: This module captures and quantifies the power supply behavior. For a given power supply profile, we determine the total number of backups and recoveries required (i.e., the backup interval from Sec. II-B) for the target architecture. If the target architecture is comprised completely of NV devices with retention times that exceed the expected outage time (i.e., for IB-NVPs), this set of data is unnecessary. For HB-NVPs, this information is used to derive the actual number of backups/recoveries needed. As will be seen later, if one is only concerned with backup/recovery patterns within a backup interval (BI) having a certain number of instructions, the data here will be averaged to obtain the corresponding values.

Backup Strategy Modeling: This module captures the data relevant to energy per backup/recovery. As pointed out in [4], how much data to save at each backup depends on the specific backup policy. It also depends on the type of NVPs. For EB-NVPs and HB-NVPs, we first determine the total number of data bits required per backup/recovery operation on average. Next, the energy associated with writing and reading the NV backup memory is input[2].

Processor Architecture Modeling: This module models the target processor architecture. Here, we use 3 high-level parameters: (i) processing element (PE) types, (ii) number of PEs per PE type, and (iii) the number of interconnect buffers required for the spintronic backup technology. We divide PEs into different types in order to handle the cases where different PE types have different switching energy per PE. Such cases arise since different technologies may be used in a NVP or PEs may have different granularity. The number of PEs per PE type represents the number of devices, pipeline components, cores, etc. Lastly, the number of interconnect buffers is determined as summarized in Sec. III-A.

NV Processor Modeling: This module calculates the total processor energy per instruction, E_{total}, based on the inputs from the three modules above[3]. E_{total} is computed as

$$E_{total} = E_{br}(BI)/BI + E_{inst}, \quad (3)$$

---

[1]Each gate in a pipeline stage is assumed to be sequentially pulsed to reduce power consumption.

[2]This value can be obtained from simulation or experimental results, and our framework assumes that this value is known.

[3]We use total processor energy per instruction instead of absolute total energy to avoid the dependence on specific programs.
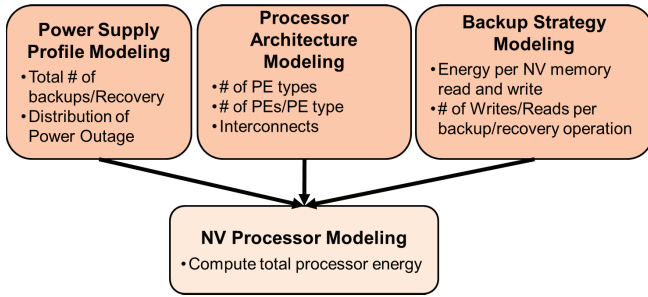
Fig. 6: Our proposed NVP benchmarking framework.

where $E_{br}(BI)$ is the total backup/recovery energy per BI and E_{inst} is the total computational energy per instruction. E_{br} is computed by

$$E_{br} = N_{backup} \times E_{backup} + N_{recover} \times E_{recover}, \text{ where} \quad (4)$$

$$E_{backup} = N_{wr/bac} \times E_{wr,NVM} + N_{rd/bac} \times E_{rd,VM}, \quad (5)$$

$$E_{recover} = N_{rd/rec} \times E_{rd,NVM} + N_{wr/rec} \times E_{wr,VM}, \quad (6)$$

where $N_{wr/bac}$ and $E_{wr,NVM}$ represent the number of writes per backup and the energy per write into the NV memory. The other parameters are defined in the same manner. E_{inst} represents the energy consumed by logic components of the processor and is calculated as

$$E_{inst} = \sum_{PE \text{ types}} N_{PE_i} \times E_{PE_i} + N_{IC(PE_i)} \times E_{IC(PE_i)} \quad (7)$$

where N_{PE_i} and E_{PE_i} are the number of type i PEs (i.e., PE_i) and PE_i's computation energy, respectively, and $N_{IC(PE_i)}$ and $E_{IC(PE_i)}$ are the number of interconnect buffers and energy associated with PE_i. If the logic is comprised of spintronic technology, the drop-in replacement approach from Sec. III-A can be used to determine the values of these parameters.

IV. NVP EVALUATION RESULTS

In this section, we present a case study that illustrates the use of EvaNVP. Our target architecture is the 32 bit MIPS NPP described in Sec. II-B. This processor runs at 8 kHz and is comprised of CMOS (computation) and STT-RAM (backup) technology. In our study, we compare three different NV technologies to the baseline CMOS+STT-RAM architecture: (i) ASL, (ii) CoMET, and (iii) SHE-RAM. For brevity, we discuss the important device parameters as necessary in the text below and point the reader to the individual device references for additional parameters.

To make a fair comparison with the current state-of-the-art technologies, we scale the 45 nm CMOS+STT-RAM results from [4] to the 15 nm technology node. To achieve this scaling, we leverage two sources: the Beyond-CMOS Benchmarking (BCB) methodology [21] and the data from the 2011 ITRS report [22]. The BCB methodology provides a uniform approach to benchmarking CMOS and beyond-CMOS devices at the circuit level. Previous work has shown this methodology to be a better predictor of CMOS scaling than the MASTAR simulator used by the ITRS [23]. Here, we use the BCB methodology to determine how CMOS energy scales from 45 nm to 15 nm for an inverter fanout-of-4 circuit. Next, we use the ITRS 2011 report to determine how STT-RAM energy scales from 45 nm to 15 nm. We use the ITRS report as this forms the foundation for STT-RAM used by NVSim [24], which is the tool used to compute STT-RAM read/write energy

for the baseline NPP in [4]. With the aforementioned approach, we find that CMOS and STT-RAM energies scale by a factor of $0.15\times$ and $0.14\times$, respectively, from 45 nm to 15 nm. These scaling factors are applied to the 45 nm NPP energy from [4] with the scaled results illustrated in Fig. 7 for the columns marked "CMOS+STT-RAM".

We now apply EvaNVP to examine the impact of ASL and CoMET based NV NPP. The first step in EvaNVP is *Power Supply Profile Modeling*. For both the ASL and CoMET technologies, their retention times are selected to be 10 years; therefore, they perform implicit backups (i.e., IB-NVP) for three selected BIs (i.e., 1, 10, and 1000 instructions) associated with the weak WiFi power source. Similarly, for *Backup Strategy Modeling*, we do not specify any inputs as both ASL and CoMET lead to IB-NVPs.

The next step in our framework is *Processor Architecture Modeling*. As both the ASL and CoMET processor implementations are comprised completely of one technology, we have only 1 PE type. Next, we select the logic gate level for our PEs, which is consistent with the drop-in replacement approach used in this study. Based on the logic area of the NPP reported by Ma et al. [4], we estimate a total of 7000 CMOS transistors. According to Eq. 2, we can determine that approximately 3500 ASL/CoMET devices would be necessary to completely implement the logic of the NPP. The final step in *Processor Architecture Modeling* is to determine the number of buffer interconnects. Using the probability distribution function summarized in Sec. III-A, we found that a processor of this size would not require any interconnect buffers. CoMET also does not require interconnect buffers as data are propagated through wires connecting CMOS devices.

For the ASL-based NVP, we examined 3 different retention times as indicated by the 3 leftmost bars in Fig. 7[4]. Our results illustrate that the base ASL (IB-NVP) is more energy efficient than CMOS+STT-RAM for a BI of 1. To further lower the energy, we consider a HB-NVP by reducing the retention time to be closer to the clock cycle time of $125\mu s$. By reducing the retention time to 1 s, the ASL energy was reduced by $2.4\times$, which is $6.25\times$ and $1.5\times$ more energy efficient than CMOS+STT-RAM for BIs of 1 and 10, respectively.

We further explored the design space of ASL under different retention time requirements by targeting a 250 ms retention time. This time is selected based on the clock cycle time of $125\mu s$ and instruction BI of 1000. By reducing the retention time to 250 ms, the computational energy is reduced by $13.6\times$ and $5.7\times$ versus the 10 year and 1 s retention times, respectively (third bar from the left in Fig. 7). This HB-NVP achieves $3.6\times$ more energy efficiency vs. CMOS+STT-RAM.

To benchmark CoMET, we use the parameters given in [25] where the switching energy of a 3-input majority gate comprised of CoMET technology is computed for the 10 nm technology node with 10 year retention time. We use 0.93 V supply in our benchmarking as it provides the best overall energy savings at 10 nm. Our result, the 4th bar in Fig. 7, shows that CoMET based NVP is about $2.3\times$ more energy efficient than the best ASL data point (250 ms retention time) and between $83\times$ and $8.3\times$ more energy efficient than the CMOS+STT-RAM for BIs between 1 and 1000, respectively.

Our final analysis involves STT-RAM vs. SHE-RAM. Here,

---

[4]The ASL parameters are taken from [5] and represent 5 nm technology node with $\lambda_{ch} = 1\mu m$ from [5].
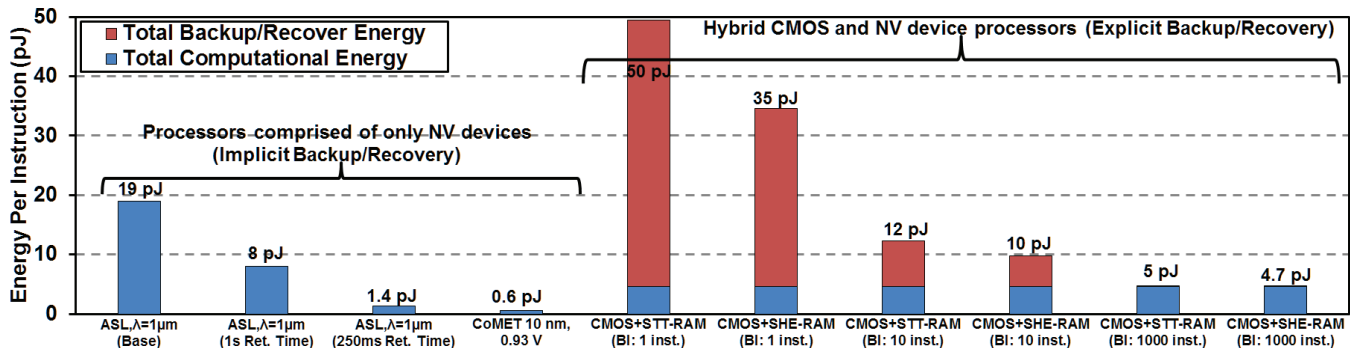
Fig. 7: Benchmarking results comparing implicit (leftmost 4 bars) versus explicit (rightmost 6 bars) backup/recovery strategies.

we have replaced the STT-RAM for backup storage in the NPP by SHE-RAM. The parameters for SHE-RAM are obtained from LLG simulation and are as follows: write current is $320\mu A$, write time is 0.84 ns and the technology node is 14 nm. For a similar technology node, STT-RAM has a write current of $150\mu A$ and a write time of 2.7 ns. Given that the switching energy is the product of the supply voltage, write current, and write time, we can estimate that the SHE-RAM is $1.5\times$ more energy efficient than the STT-RAM (assuming equal supply voltage). The total energy per instruction for the SHE-RAM based NVP is represented by the 1st, 3rd and 5th bar from the right in Fig. 7 for BI of 1000, 10, and 1, respectively.

V. CONCLUSION

As research on fundamental theories of spintronic devices is advancing steadily, it is imperative to understand the potential of spintronic devices used as memory and/or logic elements in the context of different application domains. In this paper, we introduce a high-level framework that aims to estimate overall energy savings for NVPs based on NV devices including spintronic devices. Employing this framework, we conducted preliminary studies of representative advanced spintronic memory (SHE-RAM) and logic (ASL and CoMET) elements as building blocks of both EB-NVPs and HB-NVPs. Our results show that. As future work, experimental demonstration of these advanced spintronic devices is being planned. Further studies are also being performed on investigating energy saving potentials of the NVPs for low-power applications that are not powered by harvested energy.

REFERENCES

[1] A. N. Parks et al., "A wireless sensing platform utilizing ambient rf energy," in IEEE Topical Conference on Biomedical Wireless Technologies, Networks, and Sensing Systems, Jan 2013, pp. 154–156.

[2] K. Gudan et al., "Feasibility of wireless sensors using ambient 2.4ghz rf energy," in IEEE Sensors, Oct 2012, pp. 1–4.

[3] H. J. Visser et al., "Ambient rf energy scavenging: Gsm and wlan power density measurements," in European Microwave Conference, Oct 2008, pp. 721–724.

[4] K. Ma et al., "Architecture exploration for ambient energy harvesting nonvolatile processors," in International Symposium on High Performance Computer Architecture, Feb 2015, pp. 526–537.

[5] J. Kim et al., "Spin-based computing: Device concepts, current status, and a case study on a high-performance microprocessor," Proceedings of the IEEE, vol. 103, no. 1, pp. 106–130, Jan 2015.

[6] Y. Wang et al., "A 3us wake-up time nonvolatile processor based on ferroelectric flip-flops," in ESSCIRC, Sept 2012, pp. 149–152.

[7] X. Guo et al., "Resistive computation: Avoiding the power wall with low-leakage, stt-mram based computing," in International Symposium on Computer Architecture. ACM, 2010, pp. 371–382.

[8] R. Takemura et al., "A 32-mb sram with 2t1r memory cell, localized bi-directional write driver and '1'/'0' dual-array equalized reference scheme," IEEE J. of Solid-State Cir., vol. 45, no. 4, pp. 869–879, 2010.

[9] T. Ohsawa et al., "A 1 mb nonvolatile embedded memory using 4t2mtj cell with 32 b fine-grained power gating scheme," IEEE Journal of Solid-State Circuits, vol. 48, no. 6, pp. 1511–1520, 2013.

[10] S. Manipatruni et al., "Energy-delay performance of giant spin hall effect switching for dense magnetic memory," Applied Physics Express, vol. 7, no. 10, p. 103001, 2014.

[11] L. Liu et al., "Spin-torque switching with the giant spin hall effect of tantalum," Science, vol. 336, no. 6081, pp. 555–558, 2012.

[12] L. Liu et al., "Current-induced switching of perpendicularly magnetized magnetic layers using spin torque from the spin hall effect," Phys. Rev. Lett., vol. 109, p. 096602, Aug 2012.

[13] L. Liu et al., "Spin-torque switching with the giant spin hall effect of tantalum," Science, vol. 336, no. 6081, pp. 555–558, 2012.

[14] Z. Zhao et al., "Spin hall switching of the magnetization in ta/tbfeco structures with bulk perpendicular anisotropy," Applied Physics Letters, vol. 106, no. 13, p. 132404, 2015.

[15] B. Behin-Aein et al., "Proposal for an all-spin logic device with built-in memory," Nature nanotechnology, vol. 5, no. 4, pp. 266–270, 2010.

[16] M. G. Mankalale et al., "CoMET: Composite input magnetoelectric based logic technology," 2016, https://arxiv.org/abs/1611.09714.

[17] S. Fukami et al., "Micromagnetic analysis of current driven domain wall motion in nanostrips with perpendicular magnetic anisotropy," Journal of Applied Physics, vol. 103, no. 7, p. 07E718, 2008.

[18] S. Kannan et al., "Optimizing checkpoints using nvm as virtual memory," in International Symposium on Parallel and Distributed Processing, May 2013, pp. 29–40.

[19] B. S. Landman and R. L. Russo, "On a pin versus block relationship for partitions of logic graphs," IEEE Transactions on Computers, vol. C-20, no. 12, pp. 1469–1479, Dec 1971.

[20] J. A. Davis et al., "A stochastic wire-length distribution for gigascale integration (gsi)–part i: Derivation and validation," IEEE Transactions on Electron Devices, vol. 45, no. 3, pp. 580–589, 1998.

[21] D. Nikonov and I. Young, "Benchmarking of beyond-cmos exploratory devices for logic integrated circuits," IEEE J. on Exploratory Solid-State Computational Devices and Circuits, vol. 1, pp. 3–11, Dec. 2015.

[22] International technology roadmap for semiconductors, 2011 report. [Online]. Available: http://www.itrs.net

[23] R. Perricone et al., "Can beyond-cmos devices illuminate dark silicon?" in Design, Automation Test in Europe Conference Exhibition, March 2016, pp. 13–18.

[24] X. Dong et al., "Nvsim: A circuit-level performance, energy, and area model for emerging nonvolatile memory," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 31, no. 7, pp. 994–1007, July 2012.

[25] M. G. Mankalale et al., "A fast magnetoelectric device based on current–driven domain wall propagation," in IEEE Device Research Conference, June 2016.