SOT-MRAM based Analog in-Memory Computing for DNN inference

J. Doevenspeck[1,2], K. Garello[1], B. Verhoef[1], R. Degraeve[1], S. Van Beek[1], D. Crotti[1], F. Yasin[1], S. Couet[1], G. Jayakumar[1]
I. A Papistas[1], P. Debacker[1], R. Lauwereins[1,2], W. Dehaene[1,2], G.S. Kar[1], S. Cosemans[1], A. Mallik[1] and D. Verkest[1]
[1]imec, Leuven, Belgium, [2]KU Leuven ESAT, Leuven, Belgium, email: jonas.doevenspeck@imec.be

Abstract: Deep neural network (DNN) inference requires a massive amount of matrix-vector multiplications which can be computed efficiently on memory arrays in an analog fashion. This approach requires highly resistive memory devices (>MΩ) with low resistance variability to implement DNN weight memories. We propose an optimized Spin-Orbit Torque MRAM (SOT-MRAM) as weight memory in Analog in-Memory Computing (AiMC) systems for DNN inference. In SOT-MRAM the write and read path are decoupled. This allows changing the MTJ resistance to the high levels required for AiMC by tuning the tunnel barrier thickness without affecting the writing.

The target resistance level and variation are derived from an algorithm driven design-technology-co-optimization (DTCO) study. Resistance levels are obtained from IR-drop simulations of a convolutional neural network (CNN). Variation limits are obtained by testing two noise-resilient CNNs with conductance variability. Finally, we demonstrate experimentally that the requirements for analog DNN inference are met by SOT-MRAM stack optimization.

Introduction: DNNs are built using mainly two types of layers: fully connected (FC) and convolutional (CONV) layers. The core operations in both layers are matrix-vector multiplications (MVMs) which can be mapped (Fig. 1) one-to-one to a crossbar array with programmable resistors (referred to as weight memory devices). The vector-matrix product can then be calculated in the analog domain using Ohm's and Kirchhoff's law. This AiMC [1] approach avoids costly memory fetches from an external memory and is more energy-efficient than digital multiply-accumulate (MAC) circuits [2].

Although DNN training requires many weight levels (more than 256 [3]), quantized DNNs for inference can achieve top accuracy using ternary weight levels [4]. A correct AiMC implementation depends on memory devices having stable >MΩ resistance levels and low resistance variability. MRAM is one of the most promising candidates fulfilling these requirements.

The standard writing mechanism in MRAM technology, spin transfer torque (STT), cannot be used to switch high resistance MTJs, contrary to the SOT writing mechanism (Fig. 2). In this work, we demonstrate experimentally that SOT-MRAM is an excellent candidate to implement ternary weights for inference accelerators running quantized DNNs. Also, we develop a DTCO analysis to guide the process optimisation to make SOT-MRAM suitable for this application

Experimental demonstration: Fig. 3 shows a TEM of a SOT-MTJ cell integrated on 300 mm wafers using CMOS-compatible processes. As sketched in Fig.2, the in-plane write current flows from bottom electrode 1 (BE1) to BE2 while the reading current is probed from the top electrode (TE) to BE2. The device resistance of the SOT-MTJ cell can be tuned through the MgO thickness as shown in Fig. 4. The SOT device is based on [5,6]. However, directly adapting this SOT stack with an increased MgO thickness results in severe coercivity (B_c) loss, B_c being a direct evaluation of device retention. Therefore, the stack has been reworked and optimized in order to recover sufficient coercivity at large RA products (Fig. 5). This is exemplified in Fig. 6 by R-H loops for devices with nominal CD (nCD) = 80nm and RA = 5, 20, 50 kΩ.μm^2.

Next, we applied 0.5ns voltage pulses across the SOT-track to switch the same 80nm devices, and we confirm as expected that highly resistive SOT-MTJ cells do not compromise SOT-switching performance (Fig. 7). Finally, we evaluated the resistance statistics for different MTJ RAs and CDs (Fig. 8a). Note that these numbers are obtained from a device test vehicle and do not represent the fundamental lower limit of resistance variability. Importantly, we observe that the coefficient of variation (σ/μ) does not increase for increasing RA products (Fig. 8b). The MTJ resistance spread is indeed mostly determined by process-induced area variations (Fig. 8c), which can be fully optimized at production level. In the next section, we investigate the impact of resistance variation on DNNs over a wide range, with the measured variation as reference point.

Targets for resistance levels and variability:

Assuming a differential conductance pair weight encoding scheme (Fig. 9), the relation between device conductance variability and DNN weight variability can be derived (Fig. 10). Since a weight 0 is mapped on two devices in the AP state, this puts a constraint on the AP state distributions. The weights -1 and 1 are mapped on one P and one AP state. So, they put a constraint on both the P and AP state distributions. The tolerated σ/μ of the conductance state distributions for both the 0 and -1,1 weights is given in Fig. 11. Two networks are used to derive the specifications for the weight memory device. Both networks are used to explore the tolerance against weight variation induced by conductance variations.

Analog inference accelerators add two specific challenges: variation on the network weights and quantization to a very limited amount of activation and weight levels. To avoid accuracy loss due to this mapping, the training procedure needs to be adapted (Fig. 12). Fig 13. shows the test error for CNN 1 on MNIST after training with and without variation added to the weights. Here, the same σ_{weight} (σ_W) is used during testing ($\sigma_{W,test}$) as during training ($\sigma_{W,train}$). Training with weight variation clearly decreases the test error for the whole range of weight variation. The same experiment was repeated for CNN 2 on CIFAR100 (Fig 14). However, to explore the tolerance of the network, each trained network was tested with a range of different $\sigma_{W,test}$ values. Interestingly, training with $\sigma_{W,train} > \sigma_{W,test}$ yields a better accuracy than $\sigma_{W,train} = \sigma_{W,test}$. This means that an aggressively trained network can be deployed on a variety of analog inference accelerators, each with their own specific $\sigma_{W,test}$. At $\sigma_{W,test} = 15\%$, the error increase is limited to 1.5%. Given that CIFAR100 is a representative benchmark for real-world applications and assuming 1.5% error increase is acceptable, $\sigma_{W,test} < 15\%$ is used as target for weight variation.

Next, we evaluate the impact of the wire resistance R_{WIRE} on the network accuracy on the entire test set (10.000 examples). A fully parallel GPU-compatible IR-drop solver performs a DC simulation for each analog MVM and assumes all other operations such as pooling, ReLU, BatchNorm, etc. are performed errorless in the digital domain (Fig. 15a). An example of an IR-drop simulation is given in Fig. 15b. Weights further away from the activation/summation line voltage source see a lower voltage. Hence, they draw less current and contribute too little to the matrix-vector multiplication product, giving rise to a reduced effective weight. Fig. 16 shows the DC simulations results for CNN inference using the line resistance extracted from the layout in Fig. 9 and two other reference points. Below an on resistance (R_{ON}) of 5MΩ, the test error increases due to IR-drop.

In Table I, device metrics relevant for analog DNN inference systems are compared between SOT-MRAM and other popular devices. Our optimized SOT-MRAM devices at nCD=80nm and RA=20kΩ.μm^2 meet the R_{ON}>5MΩ and σ_W<15% requirement while maintaining a low write voltage (V_{WR}).

Conclusions: Using the SOT writing mechanism, we demonstrated for the first time that highly resistive MRAM devices can be used as weight memory for AiMC DNN inference. We conducted a DTCO study where CNNs (for MNIST and CIFAR100) are trained and tested including resistance variation and the effect of wire IR-drop. It shows that SOT-MRAM meets the derived specifications.

Acknowledgments: This research is conducted within the imec IIAP entitled "Machine Learning". This project has received funding from the Electronic Components and Systems for European Leadership Joint Undertaking under grant agreement No 826655.

References: [1] S. Cosemans et al., IEDM 2019 [2] D. Bankmann et al., JSSCC, vol. 54 no. 1 p. 158 (2018) [3] N. Wang et al., NIPS, p. 7686 (2018) [4] J. Choi et al., SysML, (2019) [5] K. Garello et al., VLSI, p. 81 (2018) [6] K. Garello et al., VLSI, JFS4-5 (2019) [7] LeCun et al., Proc. of the IEEE, vol. 86 no. 11 p. 2278 (1998) [8] K. He et al., CVPR, p. 770 (2015)

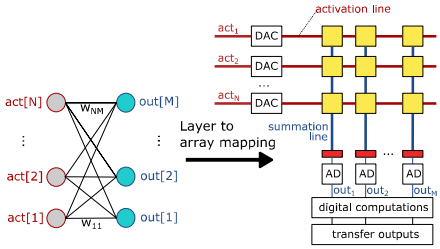


Fig.1: ACIM concept: DNN layer to analog matrix-vector-multiply (MVM) mapping. The matrix-vector product can be calculated in the analog domain using Ohm's and Kirchhoff's law.

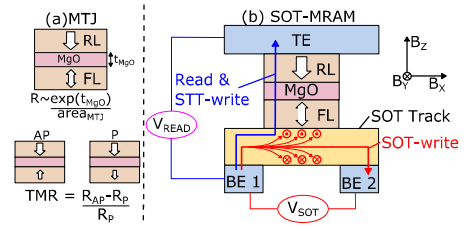


Fig. 2. (a) Magnetic tunnel junction (MTJ) (b) Three-terminal SOT-MRAM device with decoupled read and write path. Therefore, the RA product is not limited by the SOT-write. The SOT-track converts an in-plane current into a spin-current perpendicular to the free layer (FL).

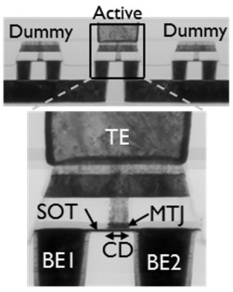


Fig. 3. TEM of one test structure consisting of an MTJ (CoFeB /MgO/CoFeB) pillar on top of a Tungsten (W) SOT-track contacting two W bottom electrodes (BE). More details on integration process and stack composition can be found in [5, 6].

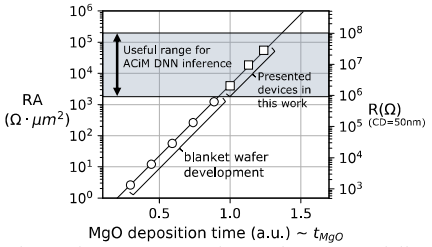


Fig. 4 The MTJ RA product scales exponentially with MgO thickness which allows to easily fabricate high resistance MTJs needed for ACiM DNN inference.

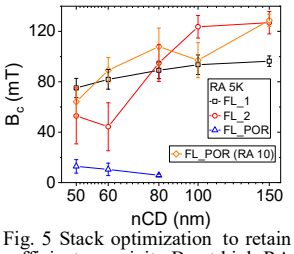


Fig. 5 Stack optimization to retain sufficient coercivity Bc at high RA values: The Bc for a standard FL (POR) severely decreases at large RA. Stack engineering the FL recovers the Bc (FL_1 and FL_2).

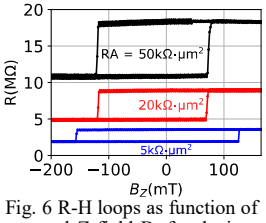


Fig. 6 R-H loops as function of external Z field B_Z for devices with nominal CD (nCD) =80nm and various RAs. The coercivity Bc is retained with increasing RA products.

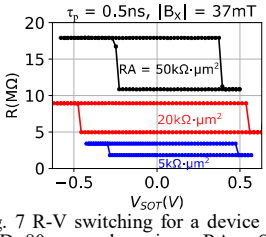


Fig. 7 R-V switching for a device with nCD=80nm and various RAs. SOT-switching enables subnanosecond write speeds (0.5ns). A manufacturable field-free solution was demonstrated in [6] and is fully compatible with our AIMC approach.

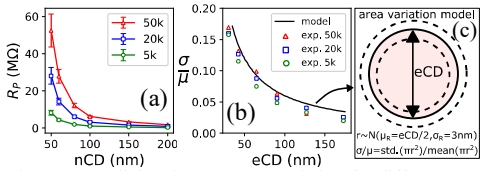


Fig. 8 a) Parallel resistance (R_P) statistics for different MTJ CDs and RA products. b) σ/μ for different electrical CDs (eCD) and RA. (c) Resistance variation can be attributed to MTJ area variation. A gaussian distribution on the MTJ radius with σ=3nm suffices to capture nearly all variation.

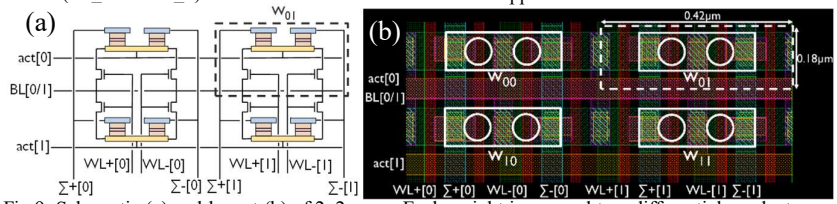


Fig.9: Schematic (a) and layout (b) of 2x2 array. Each weight is mapped to a differential conductance pair. The layout (made with a commercial 22nm PDK) is dimensioned for SOT-MRAM devices with pillar diameter ≤ 80nm and uses one two finger transistor for each SOT-MRAM device. By sharing one SOT-track with two pillars, contact area is reduced. The area of one ternary compute cell is 0.076μm².

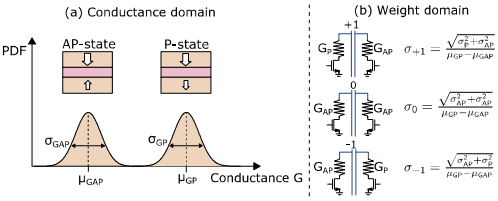


Fig. 10 Methodology to obtain network weight variation from measured conductance distributions. (a) AP and P conductance distributions. (b) Standard deviation of the weight levels $\sigma_{-1,\,0,\,+1}$ depends on σ_{GAP}, σ_{GP} and μ_{GP}-μ_{GAP}.

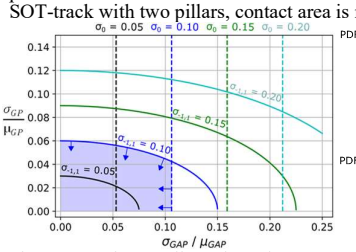


Fig. 11: Maximum allowed conductance distribution variation for given weight variation σ_W for both -1, 0, +1 weights. TMR is assumed to be 150%.

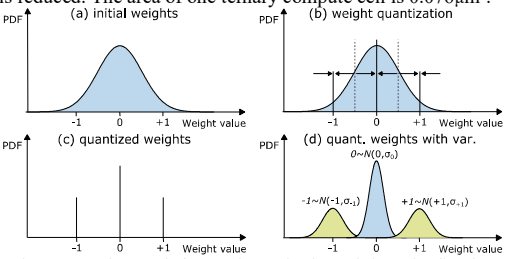


Fig. 12: During each forward pass in the training, the floating point weights are quantized and weight variation $\sigma_{W,train}$ is added.

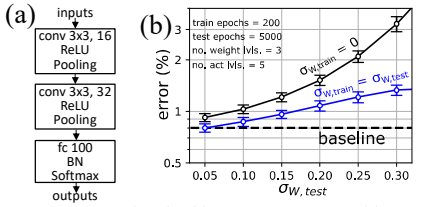


Fig. 13. (a) CNN 1 inspired by LeNet-5 [7] to achieve 99.2% accuracy on MNIST. The network is used to train and test with weight variation and in IR-drop simulations to set resistance level targets. (b) Test error on MNIST for different $\sigma_{W,train}$ and $\sigma_{W,test}$.

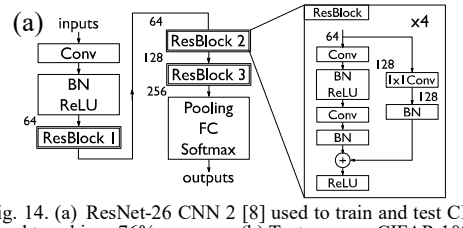
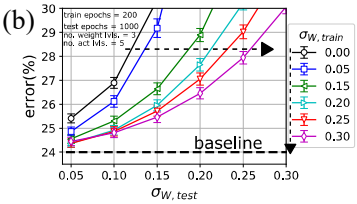


Fig. 14. (a) ResNet-26 CNN 2 [8] used to train and test CIFAR 100 with weight variation. The network is sized to achieve 76% accuracy. (b) Test error on CIFAR 100 after training with different $\sigma_{W,train}$. Up to $\sigma_{W,test}$ = 0.15, the error increase is < 1.5% by using $\sigma_{W,train}$ = 0.30. In general, using a $\sigma_{W,train} > \sigma_{W,test}$ results in smaller error increase compared to using $\sigma_{W,train} = \sigma_{W,test}$. Increasing $\sigma_{W,train}$ beyond 0.30 increases the error at low $\sigma_{W,test}$ so is not desirable (not shown on this figure).



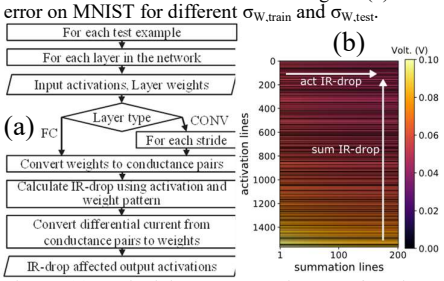


Fig. 15 (a) Methodology to assess impact of IR-drop on network accuracy. (b) Example of IR-drop in the FC layer of CNN 1. IR-drop happens both along the activation and summation line.

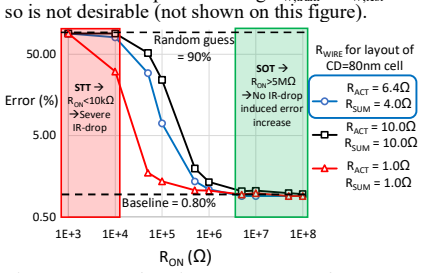


Fig. 16 Impact of IR-drop on test error of CNN 1 on MNIST for different R_{ON} and R_{WIRE}. R_{OFF} is assumed to be 2.5xR_{ON}. Baseline accuracy is recovered at $R_{ON} > 5M\Omega$.

	Flash	RRAM	PCM	STT-MRAM	SOT-MRAM RA=20kΩ.μm² CD =80nm (This work)	Inference specs.
σ_W	<15%	σ_W >> 15% at R_{ON} > 1MΩ	<15%	<15%	σ_W = 15% at R_{ON} = 6MΩ [*]	<15%
R_{ON}	Programm -able		<10MΩ	<10k		>5MΩ [**]
V_{WR}	>7V	V_{FORM} > 3V	>2V	<1V	<1V	<1V

Table I: Simplified comparison of different weight memories for analog DNN inference. Stated numbers represent typical operating conditions and are not fundamental lower limits. [*] Assuming TMR = 150%. [**] This specification depends on array size and wire resistance but is representative for relevant AiMC array sizes (>1000 rows) and sub 22nm nodes.