Built-in Self-Test of Vector Matrix Multipliers on a Reconfigurable Device

Aishwarya Natarajan and Jennifer Hasler
Georgia Institute of Technology, anatarajan35@gatech.edu

***Abstract*—An analog Vector Matrix Multiplier (VMM) along with its interface through Multi-Input Translinear Elements (MITE) is compiled on a reconfigurable Field Programmable Analog Array (FPAA) on a 350nm process. The paper focuses on the tuning algorithm to set the weights on the VMM as well as to produce levels of voltage near the power supply rails, V_{dd}, to the source-driven VMMs, by application of a wide range of input levels. The constraints during the design and implementation process, accounting for the mismatch in devices, are discussed. A significant reduction in the variation in the levels of the desired voltage has been demonstrated in the paper.**

***Index Terms*—VMM, MITE, Floating gates, weights, injection**

I. Built-In Self-Test of VMM blocks

Vector Matrix Multipliers (VMM) are the fundamental building blocks in a variety of applications, including in classifiers and signal processing [1]–[3], spiking [4], deep neural networks and neuromorphic systems [5]–[7] etc. An analog implementation of the VMM has been shown to be highly energy efficient [8] and in other mixed-signal implementations too [9], [10]. The VMM implemented on the routing fabric of the Floating Gate (FG) Field Programmable Analog Array (FPAA) performs computing in memory and performs multiplication of a set of input vectors and weights on the FG devices.

Figure 1 shows the concept behind the tuning algorithm for handling the mismatch for the VMM blocks on the FPAA. A source-driven differential VMM is implemented to perform a 4 quadrant multiplication. Since the VMM operates near V_{dd}, an interface circuitry through a network of MITE elements, drives the sources of the FG devices on the VMM, which enables the use of a wide range of inputs. There are a number of parameters to tune and the weights of the VMM have to be set, taking into account, the mismatch between the threshold voltages of the devices [11] as well as to achieve a multiplication due to the dynamic resulting from the inputs themselves. A tuning algorithm for a built-in self-test [12] is proposed to take care of these effects and to help find a solution and converge on a set of weights and biases for the system, especially through fast injection [13].

Once the system has been tuned by measuring the different intermediate nodes and overall system outputs through scanner elements, the algorithm could be used in a number of applications, to provide fast training of the weights on the VMM. This approach can be extended and applied to other analog platforms as well, especially to counteract for mismatch variations.

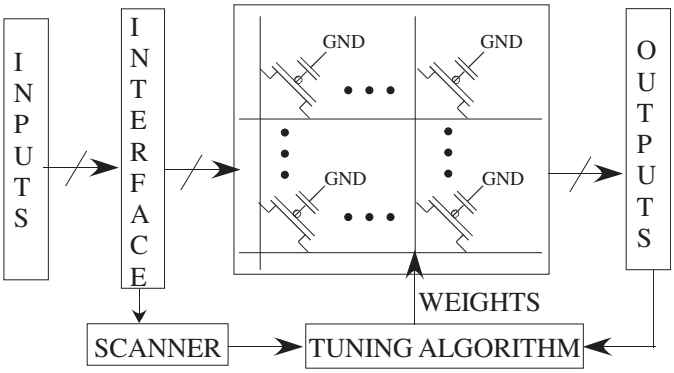

Fig. 1. The routing lines of the Computational Analog Block (CAB) in the FPAA are used to implement the differential VMM for a 4 quadrant multiplication.A built-in self-test is shown here, compensating for the differential amplifier constraints on the inputs as well as a way to figure out the mismatch in the weights due to the indirect programming methodology. The algorithm helps one to find a solution to this problem and converge on a set of weights and biases for the system. This could be used in any application which requires the use and interface of VMMs. A front-end interface to drive the VMMs to produce signals close to the supply voltage is shown as well through the use of MITEs, which could also be used and applied in a number of cases.

This paper proposes a tuning scheme for the implementation of a 6x2 VMM block on the FPAA and the variations are observed through the experimental measurements. Section II gives an overview of the macroblocked circuit on the FPAA, while Section III describes the interface circuitry to the VMM. Section IV addresses the compensation for mismatch and the measurements during the tuning of the parameters while Section V concludes the discussion.

II. Design of a differential 6x2 VMM on a Single CAB of the FPAA

A reconfigurable platform, namely the FG FPAA System on Chip (SoC) [1], fabricated on a 350nm process, with abstractable blocks [14] in the open-source tool infrastructure [15] has been used to ideate, design, implement and test the algorithm. The Computational Analog Blocks (CABs) which consist of Operational Transconductance Amplifiers (OTAs), FGOTAs, capacitor banks, transistors and transmission gates and the Computational logic blocks (CLBs) make up the fabric array. They are interfaced with MSP430, SRAM etc all on-chip to provide control and interface signals for programming.

A 6x2 VMM along with a TransImpedance Amplifier (TIA) core cell, with voltage inputs and voltage outputs, has been macroblocked in a single CAB to make a compact, energy-

978-1-7281-3320-1/20/$31.00 ©2020 IEEE

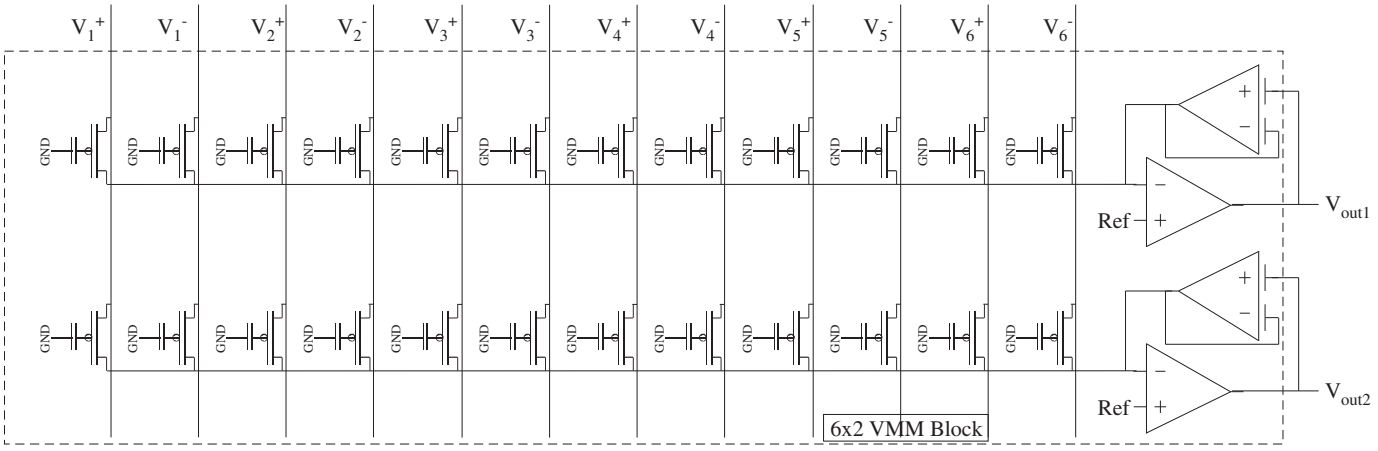

Fig. 2. A differential 6x2 VMM with the TIA to translate the currents to voltages is shown. The FG elements of the VMM perform the 4 quadrant multiplication, with the weights being set by the FG-pFETs on the positive and negative input. The inputs V_i^+ and V_i^- that drive the sources of the VMM have to operate near V_{dd}. Six differential column lines are selected from the CAB, with 2 row output lines that go into the CAB elements that comprise the TIA, while the local routing elements in the fabric consisting of the FG elements make up the VMM structure. The output levels after the TIA are adjusted primarily through the FGOTA element at the feedback of each of the TIA, which controls the gain, offset and the operating range.

efficient design. Six differential lines or 12 columns are selected from the CAB, with 2 row output lines that go into the CAB elements. The routing fabric stores each of the weights of the FG-pFETs of the VMM since the switches in the crossbar can be programmed to a target value depending on the desired charge required on the FG node. A size of 2 rows has been designed since with the number of OTAs and FGOTAs available in one CAB, 2 TIAs can be built which convert the currents summed from each row to a voltage level which can be tuned by adjusting the conductances of the TIAs.

The I-V element from the CAB, TIA, consists of an OTA to which the current is fed into, with a FG-OTA in a source follower configuration that feeds back the output. A FGOTA has been chosen in the feedback for primarily two reasons. Due to the capacitive divider at the two inputs of the FGOTA, it shows a higher linearity for the output of the computation than an OTA. Moreover, the voltage offset and gain at the output can be modulated by setting the charges on the FG-pFETs at the input terminals of the FGOTA. The DC level at the output can be changed by moving the V_{ref} voltage as well, which is an input to the feed-forward OTA.

A higher order VMM cell could be built either by scaling these core cells across the different CABs or by building a macroblock with voltage input and current output by utilizing all the routing lines in one CAB.

III. Interface circuitry to VMM

Figure 2 shows the entire chain of circuitry from the inputs to the interface to the VMM 6x2 macroblock. The inputs to the VMM are differential, with the source terminal as the input. with the multiplication arising since there in an exponential relationship between the current and source voltage. The structure requires inputs close to the supply voltage, V_{dd}, being 100mV below V_{dd}. The two lines to each pair of FG-pFETs in a row represent the signed input, with a single current output.

Another requirement of such an interface is that the circuit should not act like a current source type input since it adds constraints such as a differential pair which would effectively lead to the loss of an input best case. Hence, a set of MITE elements as shown in Fig. 4 was utilized to drive the source of VMMs. The inputs are applied to one of the capacitive nodes, while the other is kept at a reference voltage level. The FG on the MITE allows one to move the inputs in different places between the power supply rails. A wider input signal range is obtained due to the low input coupling ratio on the MITE capacitive gates, thereby allowing the current VMM to have only a few U_T of single-ended voltage swing.

A. Characterization of MITE elements

The MITE element [16] is characterized by measuring the current against the gate, V_g and drain voltages, V_d. A parametric sweep is done where the input voltage to the first capacitance is swept, for different input voltages to the smaller capacitance, as shown in the Fig. 3. Both the FG inputs with the shown input capacitances are tied to a voltage which is swept, as shown in the x-axis. The capacitive coupling can be seen in the slope and a range of currents are obtained as the input voltage is swept. These characterizations help to extract parameters like κ, the fractional change in the surface potential, further giving insights into biasing of the elements for the algorithm. As expected, the currents decrease as the input terminal approaches V_{dd}.

B. Near V_{dd} circuitry measurements with MITE elements

Figure 4 shows the scan chain of elements on test along with the algorithm for obtaining the near V_{dd} values along with the variation in the voltages of the middle nodes across iterations. The characterization of the MITE elements above through the current measurements helps one to figure out the initialization of the parameters, specifically, the currents to which the FG

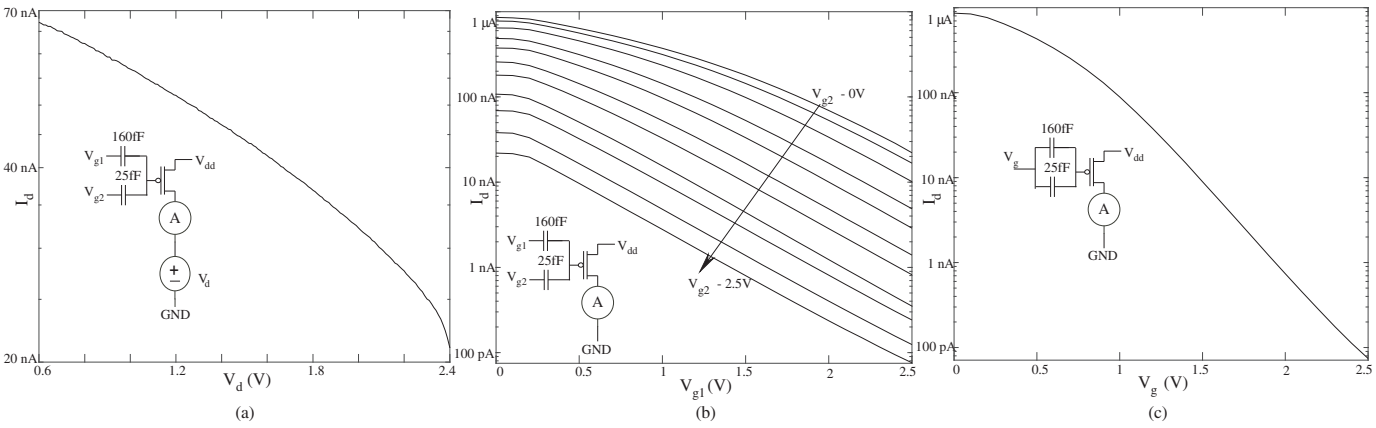

Fig. 3. Characterization of the MITE element and measured current as a function of gate voltage and drain voltage in log scale, with the Source voltage being fixed at a V_{dd} of 2.5V. (a) The drain voltage is swept to extract σ since the slope is $\frac{\kappa}{U_T}\frac{C_{ov}}{C_T} + \sigma_p$. (b) A parametric sweep is done where the input voltage to the first capacitance is swept, for different input voltages of the smaller capacitance.(c) Both the FG inputs with the shown input capacitances are tied to a voltage which is swept, as shown in the x-axis. κ is extracted from a slope of $\frac{\kappa}{U_T}\frac{C}{C_T}$. As expected, the currents decrease as the input terminal approaches V_{dd}, due to the capacitive coupling.

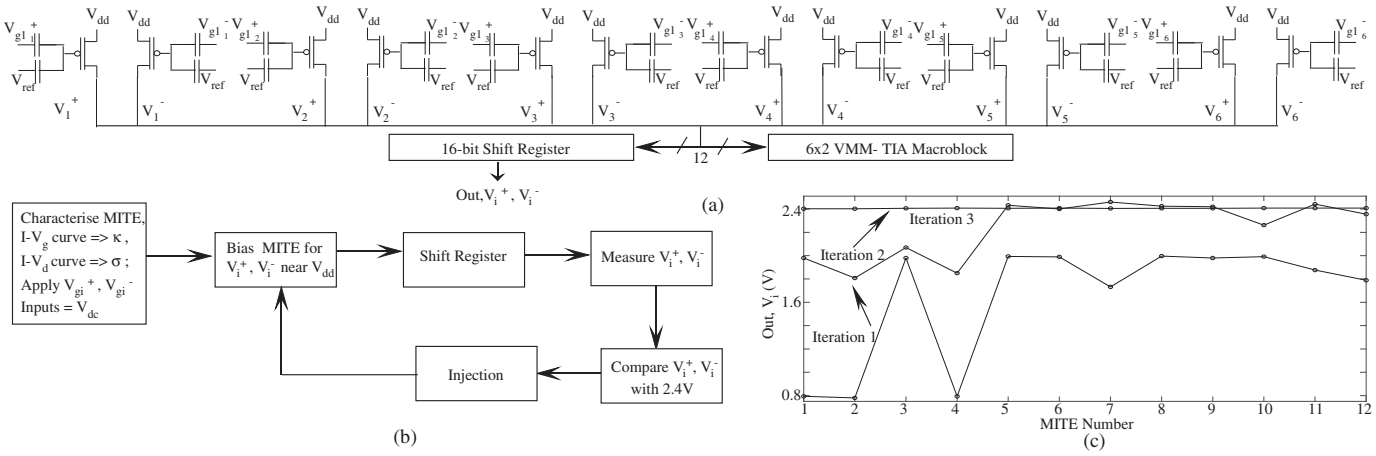

Fig. 4. (a) The MITE elements' outputs are characterized and recorded through a shift register for calibration of the MITE elements to input to the VMM rows. (b) The voltages at the output intermediate nodes are set such that it is as close to V_{dd} as possible and the FG devices are biased to avoid a differential pair effect on the columns. The levels are compared and the MITE elements are injected accordingly. (c) The voltages of the individual MITEs are plotted after three iterations and a significant reduction in the variation is observed from a change of 1.6V to 5mV.

devices on the MITE elements are programmed to. A 16-bit shift register, whose clock and data lines are controlled by the MSP 430 processor on chip, is used to scan out the outputs from the MITE circuitry.

Once calibrated and the built-in self test is done, the scanner is not required for the application. ${V_{gi}}^+$ and ${V_{gi}}^-$ represent the inputs to the MITE elements for the positive and negative weights respectively while ${V_i}^+$ and ${V_i}^-$ represent the intermediate inputs to the VMM weights.

The FGs on the MITEs are initially programmed such that it does not act just as a current source to prevent any onset of differential pair effect on the columns, thereby not affecting the weight values on the VMMs themselves. The measured ${V_i}^+$ and ${V_i}^-$ should be such that it is within $2U_T$ of 2.4V. A fast tuning is done through injection on the corresponding MITE FG device as soon as ${V_i}^+$ or ${V_i}^-$ falls below 2.4V. A series of iterations is shown in Fig. 4c where the variation in the nodes from the desired 2.4V drops significantly from a range of 1.6V to 5mV.

IV. Built-in Self Test for the full chain

The algorithm flow for the entire chain from the interface circuits to the VMM is shown in Fig. 5. The MITE elements are biased, as discussed in section III. The rows of weights on the VMM have to take into account, the inherent mismatch in the threshold voltages between the FG devices on the VMM, due to the indirect programming methodology [17] where the pFETs for computation and programming are different.

The maximum input range, X_k, '1' and '-1' is chosen that corresponds to the positive and negative weights, $W^+$and W^-, such that the actual weight is the difference between W^+ and W^-. Initially, for all inputs individually, the pair of positive

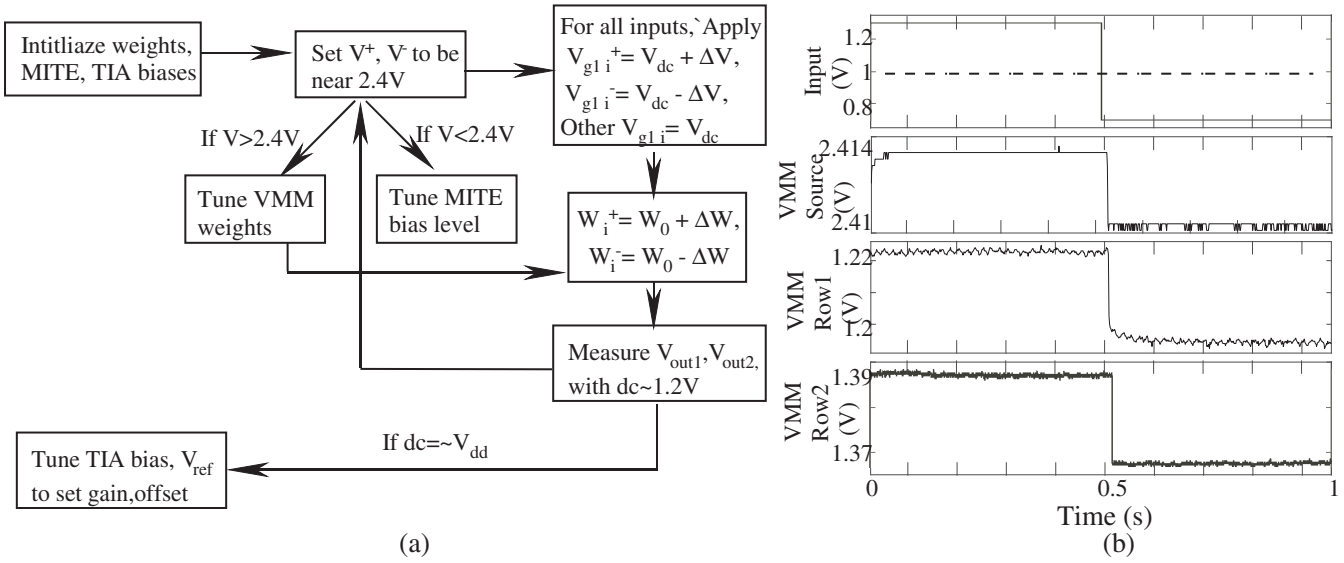

Fig. 5. (a) The algorithm flow to set the weights of the VMM and calculate the variation change with respect to the different inputs applied and measuring the intermediate nodes and repeating the process for all inputs, is shown. After an initialization of the parameters, the source inputs to the VMM and system outputs are measured and compared and the weights or the MITE biases are fine tuned accordingly depending on the variation. (b) For a set of inputs, the voltage swing from the input MITE element to the source of the VMM is observed. The output voltages at each row of the VMM 6x2 block is measured as well, whose gain and operating range can be controlled.

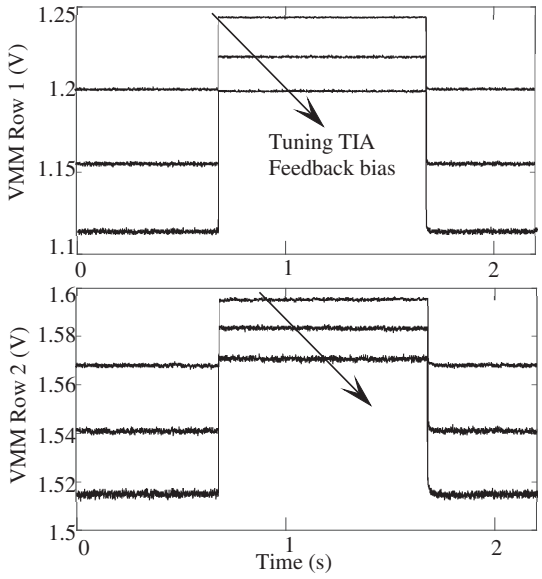

Fig. 6. The level of the outputs of the VMM row 1 and 2 after the TIA is shown, in response to a step input. The gain and offset can be tuned by changing the bias on the FG OTA at the feedback of the TIA. As the feedback bias current is increased, the gain drops on the output signal. A higher gain is further obtained by decreasing the bias on the FGOTA of the TIA.

and negative FG devices on the row are given '+1' and '-1' inputs while the rest of the FG devices are given '0' input. These are normalized with a V_0 level as '0' swinging around a $|\Delta V|$ for the +1' and '-1' inputs. The scanner measures all the intermediate nodes, V_i^+, V_i^- such that the DC level is 2.4V with a swing within U_T. Summing the currents from all the FG-pFETs of the VMM on one row, and under the condition that $|\Delta V| \leq U_T$, the resultant output current, I_{out}, is a product of the input and the difference of the weights and a bias current, with an offset. Hence, the ranges and values of the weights could be initially analytically obtained or from simulating the system [18] as well in our tools.

The variation in the weights are adjusted for and the common change is accounted for in the feedback FGOTA in the TIA, maintaining a DC level around the mid-region of V_{dd}. The middle nodes, V_i^+, V_i^- are scanned again and compared against 2.4V. The VMM pair devices are injected if V_i^+ and V_i^- exceeds 2.4V while the corresponding MITE element is injected for the other case. This is repeated for all the inputs. Fig. 5b shows the measured outputs and voltage swing at different nodes for a set of inputs.

Figure 6 shows the response of the VMM for a set of biases to a step input. The output voltage levels of the VMM are controlled by the TIA. The feedforward OTA is programmed to a high value. As the bias current of the feedback FG OTA is reduced, there is an increase in the output gain of the signals while their offset charge is used to center the output waveform.

V. Discussion

The algorithm proposed in this paper is applied and tested out by the implementation on a reconfigurable mixed-signal platform, the FPAA which could be further generalized to other analog devices and platforms as well, extending along this idea. The VMMs can be scaled up to implement programmable filters [19] where the input data would be from a microphone or similar sensor, and outputs coming from the VMM block.

REFERENCES

[1] S. George, S. Kim, S. Shah, J. Hasler, M. Collins, F. Adil, R. Wunderlich, S. Nease, and S. Ramakrishnan, "A programmable and configurable mixed-mode FPAA soc," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 6, pp. 2253–2261, June 2016.

[2] S. Ramakrishnan, A. Basu, L. K. Chiu, J. Hasler, D. Anderson, and S. Brink, "Speech processing on a reconfigurable analog platform," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 2, pp. 430–433, Feb 2014.

[3] H. Töreyin and P. T. Bhatti, "A low-power asic signal processor for a vestibular prosthesis," *IEEE transactions on biomedical circuits and systems*, vol. 10, no. 3, pp. 768–778, 2016.

[4] C. S. Thakur, J. L. Molin, G. Cauwenberghs, G. Indiveri, K. Kumar, N. Qiao, J. Schemmel, R. Wang, E. Chicca, J. Olson Hasler, J.-s. Seo, S. Yu, Y. Cao, A. van Schaik, and R. Etienne-Cummings, "Large-scale neuromorphic spiking array processors: A quest to mimic the brain," *Frontiers in Neuroscience*, vol. 12, p. 891, 2018. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnins.2018.00891

[5] J. Binas, D. Neil, G. Indiveri, S.-C. Liu, and M. Pfeiffer, "Precise deep neural network computation on imprecise low-power analog hardware," *arXiv preprint arXiv:1606.07786*, 2016.

[6] M. Bavandpour, M. R. Mahmoodi, and D. B. Strukov, "Energy-efficient time-domain vector-by-matrix multiplier for neurocomputing and beyond," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 66, no. 9, pp. 1512–1516, Sep. 2019.

[7] A. Natarajan and J. Hasler, "Implementation of synapses with hodgkin huxley neurons on the fpaa," in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2019, pp. 1–5.

[8] C. R. Schlottmann and P. E. Hasler, "A highly dense, low power, programmable analog vector-matrix multiplier: The fpaa implementation," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 1, no. 3, pp. 403–411, Sep. 2011.

[9] R. Genov and G. Cauwenberghs, "Charge-mode parallel architecture for vector-matrix multiplication," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, no. 10, pp. 930–936, Oct 2001.

[10] M. R. Mahmoodi and D. Strukov, "An ultra-low energy internally analog, externally digital vector-matrix multiplier based on nor flash memory technology," in *2018 55th ACM/ESDA/IEEE Design Automation Conference (DAC)*, June 2018, pp. 1–6.

[11] S. Kim, S. Shah, and J. Hasler, "Calibration of floating-gate SoC FPAA system," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2017.

[12] S. Shah and J. Hasler, "Tuning of multiple parameters with a bist system," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 64, no. 7, pp. 1772–1780, July 2017.

[13] S. Kim, J. Hasler, and S. George, "Integrated floating-gate programming environment for system-level ics," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. PP, no. 99, pp. 1–9, 2016.

[14] J. Hasler, A. Natarajan, and S. Kim, "Enabling energy-efficient physical computing through analog abstraction and ip reuse," *Journal of Low Power Electronics and Applications*, vol. 8, no. 4, p. 47, 2018.

[15] M. Collins, J. Hasler, and S. George, "An open-source tool set enabling analog-digital-software co-design," *Journal of Low Power Electronics and Applications*, vol. 6, no. 1, p. 3, 2016.

[16] B. A. Minch, C. Diorio, P. Hasler, and C. A. Mead, "Translinear circuits using subthreshold floating-gate mos transistors," *Analog Integrated Circuits and Signal Processing*, vol. 9, no. 2, pp. 167–179, 1996.

[17] D. W. Graham, E. Farquhar, B. Degnan, C. Gordon, and P. Hasler, "Indirect programming of floating-gate transistors," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 54, no. 5, pp. 951–963, May 2007.

[18] A. Natarajan and J. Hasler, "Modeling, simulation and implementation of circuit elements in an open-source tool set on the FPAA," *Analog Integrated Circuits and Signal Processing*, vol. 91, no. 1, pp. 119–130, 2017.

[19] M. Kucic, A. Low, P. Hasler, and J. Neff, "A programmable continuous-time floating-gate fourier processor," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, no. 1, pp. 90–99, Jan 2001.