A 1GMACS/mW MIXED-SIGNAL DIFFERENTIAL-CHARGE CID/DRAM PROCESSOR

Roman Genov, *Member, IEEE*
Department of Electrical and Computer Engineering, University of Toronto,
10 King's College Road, Toronto, ON M5S 3G4, Canada
roman@eecg.toronto.edu

Abstract

The mixed-signal processor performs digital vector matrix multiplication using internally analog fine-grain parallel computing, for real-time linear transforms. The three-transistor CID/DRAM unit cell combines single-bit dynamic storage, binary multiplication, and zero-latency analog accumulation. Two CID/DRAM cells with complimentary inputs and stored coefficients comprise a signed multiply-and-accumulate cell. Differential charge sensing between the complementary cells doubles the integration density of the array compared to the sensing mechanism utilizing dummy cells. Delta-sigma analog-to-digital conversion of the analog array outputs is combined with oversampled unary coding of the digital inputs. The 256 × 128 CID/DRAM processor with integrated 128 delta-sigma ADCs measures 3 mm × 3 mm in 0.5 μm CMOS and delivers 6.5 GMACS dissipating 5.9 mW of power.

1. Introduction

Real-time computing of linear transforms on a battery-powered mobile platform imposes great demands on computational throughput and power consumption. The computational core of linear transforms in signal processing and communications, such as discrete cosine transform (DCT), discrete wavelet transform (DWT) and vector quantization (VQ), is that of vector-matrix multiplication (VMM) in high dimensions:

$$Y_m = \sum_{n=0}^{N-1} W_{mn} X_n \tag{1}$$

with N-dimensional input vector X_n, M-dimensional output vector Y_m, and $M \times N$ matrix elements W_{mn} (templates).

The presented mixed-signal VMM processor contains a fine-grain parallel computational array, achieving a computational throughput of 1.1 GMACS for every mW of power. In what follows we concentrate on massively parallel VMM computation in its oversampled mixed-signal differential-charge architecture.

2. Mixed-Signal Computation

2.1. Internally Analog, Externally Digital Computation

The approach combines the computational efficiency of analog array processing with the precision of digital processing and the convenience of a programmable and reconfigurable digital interface.

The digital representation is embedded in the analog array architecture, with matrix elements stored locally in bit-parallel form

$$W_{mn} = \sum_{i=0}^{I-1} 2^{-i-1} w_{mn}^{(i)} \tag{2}$$

and inputs presented in bit-serial fashion

$$X_n = \sum_{j=0}^{J-1} \gamma_j x_n^{(j)} \tag{3}$$

where the coefficients γ_j are assumed in radix two, depending on the form of input encoding used. The VMM task (1) then decomposes into

$$Y_m = \sum_{n=0}^{N-1} W_{mn} X_n = \sum_{i=0}^{I-1} 2^{-i-1} Y_m^{(i)} \tag{4}$$

with VMM partials

$$Y_m{}^{(i)} = \sum_{j=0}^{J-1} \gamma_j Y_m^{(i,j)}, \tag{5}$$

$$Y_m{}^{(i,j)} = \sum_{n=0}^{N-1} w_{mn}^{(i)} x_n^{(j)} \ . \tag{6}$$

The binary-binary partial products (6) are conveniently computed and accumulated, with zero latency, using an analog VMM array [1]-[2]. In principle, the VMM partials (6) can be quantized by a bank of flash analog-to-digital converters (ADCs), and the results accumulated in the digital domain according to (5) and (4) to yield a digital output resolution exceeding the analog precision of the array and the quantizers [3]. In the present work, an oversampling ADC accumulates the sum (5) in the analog domain, with inputs encoded in unary format ($\gamma_i = 1$). This

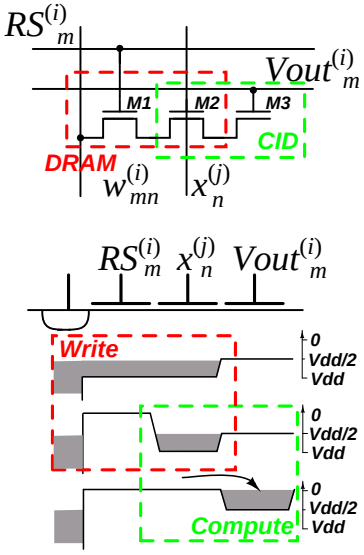

Figure 1. CID (charge injection device) multiply-and-accumulate cell with integrated DRAM storage (*top*). Charge transfer diagram for active write and compute operations (*bottom*).

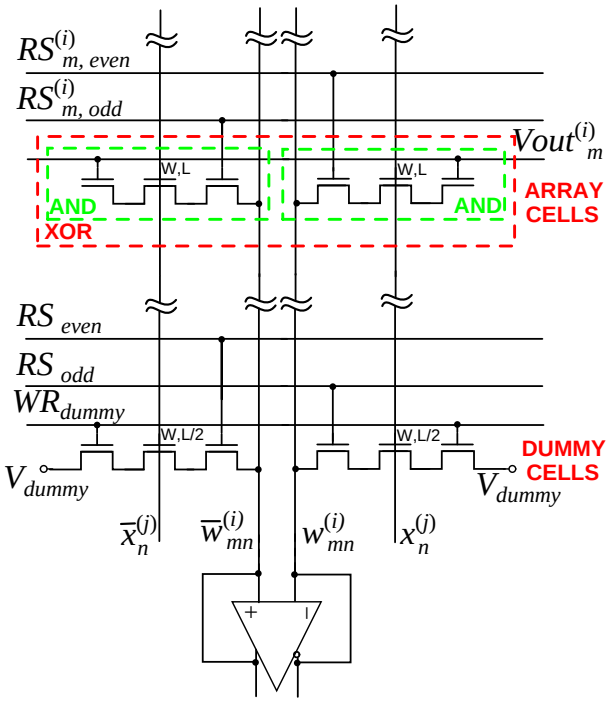

Figure 2. The CID/DRAM array architecture with folded bit lines. Two CID/DRAM array cells and two dummy cells, one dummy cell per bit line, are shown. Even and odd cells are cross-coupled with odd and even dummy cells respectively, for time-multiplexed charge sensing and readout.

avoids the need for high-resolution flash ADCs, which are replaced with single-bit quantizers in the delta-sigma loop.

2.2. CID/DRAM Cell and Array

The unit cell in the analog array combines a CID (charge injection device [4]) computational element [2] with a DRAM storage element. The cell stores one bit of a matrix element ${w_{mn}}^{(i)}$, performs a one-quadrant binary-unary (or binary-binary) multiplication of ${w_{mn}}^{(i)}$ and ${x_n}^{(j)}$ in (6), and accumulates the result across cells with common m and i indices. The circuit diagram and operation of the cell are given in Fig. 1. It performs a non-destructive computation since the transferred charge, Q, is sensed capacitively at the output. An array of cells thus performs (unsigned) binary-unary multiplication (6) of matrix ${w_{mn}}^{(i)}$ and vector ${x_n}^{(j)}$ yielding ${Y_m}^{(i,j)}$, for values of i in parallel across the array, and values of j in sequence over time.

2.3. Differential-Charge CID/DRAM Architecture

To improve linearity and to reduce sensitivity to clock feedthrough, we use differential encoding of input and stored bits in the CID/DRAM architecture using twice the number of columns and unit cells as shown in Fig. 2. This amounts to exclusive-OR (XOR), rather than AND, multiplication on the analog array, using signed, rather than unsigned, binary values for inputs and weights, ${x_n}^{(j)} = \pm 1$ and ${w_{mn}}^{(i)} = \pm 1$. Such a configuration finds applications in a wider class of problems where signed multiply-and-accumulate computations are performed, such as in the stochastic mixed-signal VMM architecture [5].

Besides computing, charge sensing and readout are also differential. The bit lines of the CID/DRAM array in Fig. 2 are folded, splitting in two matched halves. Each half is connected with a column of CID/DRAM cells and one dummy cell. Even and odd cells of a row selected to be refreshed are cross-coupled with odd and even dummy cells respectively. The dummy cells are always written with half of the computing cell charge, $Q/2$. A conventional time-multiplexed charge sensing and readout is implemented. First, the odd cells in a selected row of the array are connected to the odd bit lines while the even dummy cells are connected to the even bit lines. Depending on the stored values, the sense amplifier discriminates between two charges: Q and $Q/2$, or 0 and $Q/2$. The minimum charge to be discriminated is therefore equal to $Q/2$. Next, the even cells are refreshed or read out in a similar manner. The ratio of the bit-line capacitances to the cell capacitances is limited by the sensitivity of the sense amplifier.

The inherent drawback of the signed multiply-and-accumulate architecture shown in Fig. 2 is the doubled area compared to the unsigned computation architecture with the same number of input dimensions, for the same sense amplifier sensitivity. A differential-charge signed multiply-and-accumulate architecture depicted in Fig. 3 avoids this area penalty. Instead of dummy cells, sens-

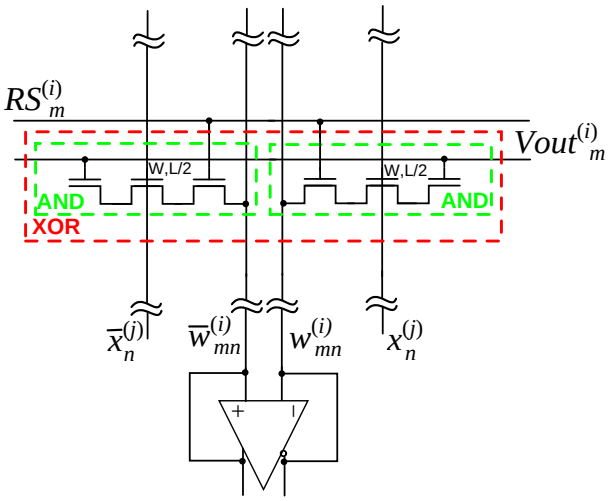

Figure 3. The complementary-charge CID/DRAM array architecture. Two charge-mode AND cells are configured as an exclusive-OR (XOR) signed multiply-and-accumulate gate. Differential sensing and computing allows to halve the area of the storage cell doubling the density of the array integration in the case of signed vector-matrix multiplication.

ing discrimination is performed directly on every two neighboring array cells, with complementary bits stored in them. Such differential sensing on complementary cells allows to halve the area of the cells, with the maximum stored charge equal to $Q/2$. The minimum charge to be discriminated is maintained equal to $Q/2$. Assuming no limitation in the wiring area, this doubles the integration density of the array, for the same sensitivity of sense amplifier.

3. Oversampling Mixed-Signal Array Processing

The conventional delta-sigma ($\Delta\Sigma$) ADC design paradigm allows to reduce requirements on precision of analog circuits to attain high resolution of conversion, at the expense of bandwidth. In the presented architecture a high conversion rate is maintained by combining delta-sigma analog-to-digital conversion with oversampled encoding of the digital inputs, where the delta-sigma modulator integrates the partial multiply-and-accumulate outputs (6) from the analog array according to (5).

Fig. 4 depicts one row of matrix elements W_{mn} in the $\Delta\Sigma$ oversampling architecture, encoded in $I = 4$ bit-parallel rows of CID/DRAM cells. One bit of a unary-coded input vector is presented each clock cycle, taking J clock cycles to complete a full computational cycle (1). The data flow is illustrated for a digital input series $x_n{}^{(j)}$ of $J = 16$ unary bits.

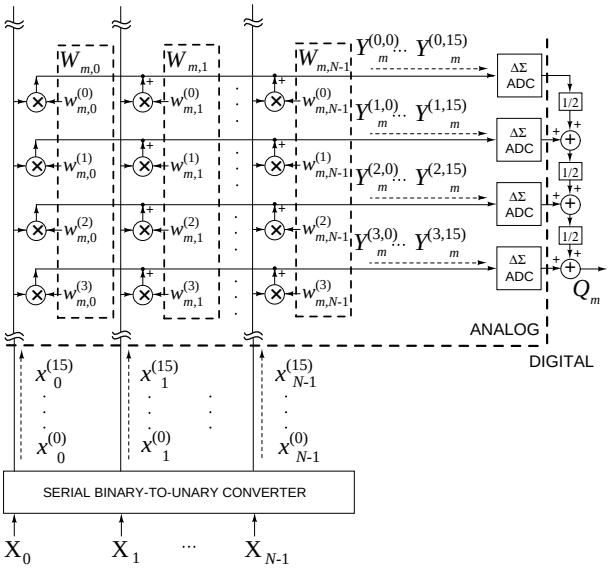

Figure 4. Block diagram of one row in the matrix with binary encoded elements $w^{(i)}{}_{mn}$, for a single m and unary encoded inputs. Data flow of bit-serial inputs $x^{(j)}{}_n$ and corresponding partial product outputs $Y^{(i,j)}{}_m$, with $J = 16$ bits. The full product for a single row $Y^{(i)}{}_m$ is accumulated and quantized by a delta-sigma ADC. The final product is constructed in the digital domain according to (4).

Over J clock cycles, the oversampling ADC integrates the partial products (6), producing a decimated output

$$Q_m^{(i)} \approx \sum_{j=0}^{J-1} \gamma_j Y_m^{(i,j)} \ , \qquad (7)$$

where $\gamma_j = 1$ for unary coding of inputs. Decimation for a first-order delta-sigma modulator is achieved using a binary counter.

Higher precision can be obtained in the same number of cycles J by using a higher-order delta-sigma modulator topology. However this drastically increases the implementation complexity. Instead, we use a modified topology that resamples the residue of the integrator after initial conversion [6]. A sample-and-hold resamples the residue voltage of the integrator and presents it to the modulator input for continued conversion at a finer scale. With a single resampling of the residue, the $\Delta\Sigma$ modulator obtains 8-bit effective resolution in 32 cycles.

4. Experimental Results

A mixed-signal VMM processor prototype integrated on a 3×3 mm^2 die was fabricated in 0.5 μm CMOS technology. The chip contains an array of 256×128 CID/DRAM cells, and a row-parallel bank of 128 $\Delta\Sigma$ algorithmic ADCs. Fig. 5 depicts the micrograph and system floorplan of the chip. The layout size of the CID/DRAM cell is $18\lambda \times 45\lambda$ with $\lambda = 0.3\mu m$.

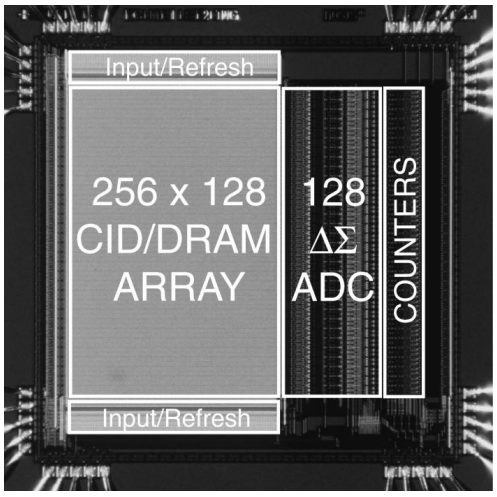

Figure 5. Micrograph of the mixed-signal pattern recognition processor prototype, containing an array of 256×128 CID/DRAM cells, and a row-parallel bank of 128 $\Delta\Sigma$ algorithmic ADCs. Die size is 3 mm $\times$ 3 mm in 0.5 μm CMOS technology.

Table 1. Summary of measured performance

Technology	0.5 μm CMOS
Area	3mm $\times$ 3mm
Power	5.9 mW
Supply Voltage	5 V
Dimensions	256 inputs $\times$ 128 templates
Throughput	6.5 GMACS
Output Resolution	8-bit

The processor interfaces externally in digital format. Two separate shift registers load the templates along odd and even columns of the DRAM array. Integrated refresh circuitry periodically updates the charge stored in the array to compensate for leakage. Vertical bit lines extend across the array, with two rows of sense amplifiers at the top and bottom of the array. The refresh alternates between even and odd columns, with separate select lines.

Fig. 6 shows the measured linearity of the computational array, configured differentially for signed (XOR) multiplication. For every shift in the input register, a computation is performed and the result is observed on the output sense line.

The chip contains 128 row-parallel $\Delta\Sigma$ algorithmic ADCs, *i.e.*, one dedicated ADC for each m and i. In the present implementation, Y_m is obtained off-chip by combining the ADC quantized outputs ${Y^{(i)}}_m$ over i (rows) according to (4). The $\Delta\Sigma$ ADC yields 8-bit resolution over two subranging cycles of 4-bit each, for a total of 32 clock cycles.

Table 1 summarizes the measured performance. The CID/DRAM array dissipates 3.3 mW for a 10 μs computational cycle, and the bank of $\Delta\Sigma$ ADCs dissipates 2.6 mW yielding a combined conversion rate of 12.8 Msamples/s at 8-bit resolution.

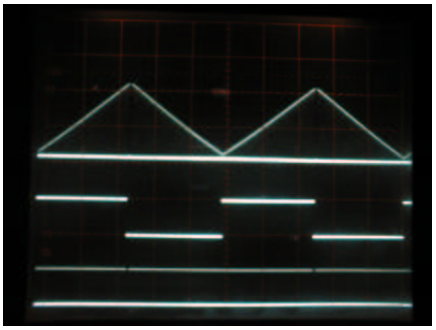

Figure 6. Measured linearity of the computational array configured for signed multiplication on each cell (XOR configuration). Two cases are shown: binary weight storage elements are all actively charged, and all discharged. Waveforms shown are, *top to bottom*: the analog voltage output on the sense line; input data (in common for both input and weight shift register); and input shift register clock.

5. Conclusions

An oversampling charge-mode VLSI architecture for real-time linear transforms has been presented. An internally analog, externally digital architecture offers the best of both worlds: the density and energetic efficiency of an analog VLSI array, and the convenience and versatility of a digital interface. A $\Delta\Sigma$ oversampled algorithmic ADC architecture relaxes precision requirements in the quantization.

A 256×128 cell prototype was fabricated in 0.5 μm CMOS. The combination of analog array processing, oversampled input encoding, and $\Delta\Sigma$ algorithmic analog-to-digital conversion delivers a computational throughput of over 1 GMACS per mW of power, while maintaining 8-bit effective digital resolution.

[1] A. Kramer, "Array-based analog computation," *IEEE Micro*, vol. **16** (5), pp. 40-49, 1996.

[2] V. Pedroni, A. Agranat, C. Neugebauer, A. Yariv, "Pattern matching and parallel processing with CCD technology," Proc. IEEE Int. Joint Conference on Neural Networks (IJCNN'92), vol. **3**, pp 620-623, 1992.

[3] R. Genov, G. Cauwenberghs "Charge-Mode Parallel Architecture for Matrix-Vector Multiplication," *IEEE T. Circuits and Systems II*, vol. **48** (10), 2001.

[4] M. Howes, D. Morgan, Eds., *Charge-Coupled Devices and Systems*, John Wiley & Sons, 1979.

[5] R. Genov, G. Cauwenberghs, "Stochastic Mixed-Signal VLSI Architecture for High-Dimensional Kernel Machines," *Advances in Neural Information Processing Systems*, Cambridge, MA: MIT Press, vol. 14, 2002.

[6] G. Mulliken, F. Adil, G. Cauwenberghs, R. Genov, "Delta-Sigma Algorithmic Analog-to-Digital Conversion," IEEE Int. Symp. on Circuits and Systems (ISCAS'02), Scottsdale, AZ, May 26-29, 2002. ISCAS'2002.