The Design of an Analog Associative Memory Circuit for Applications in High-Dimensional Computing

Miles Rusch
Jan M. Rabaey, Ed.

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2018-72
http://www2.eecs.berkeley.edu/Pubs/TechRpts/2018/EECS-2018-72.html

May 18, 2018

Copyright © 2018, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

Acknowledgement

Professor Jan Rabaey
Taiwan Semiconductor Manufacturing Company (TSMC)

The Design of an Analog Associative Memory Circuit for Applications in High-Dimensional Computing

Miles Rusch

Master of Science in Engineering - Electrical Engineering and Computer Sciences University of California, Berkeley

Professor Jan Rabaey

***Abstract*—An Associative Memory is designed for computing in high-dimensional (HD) vector spaces. The AM is a crucial part of the part of the Vector Symbolic Architecture (VSA), in which data is mapped into a HD vector space while preserving the similarity of data samples. VSA has been used to implement supervised classifiers that learn more quickly than artificial neural networks. The Associative Memory (AM) stores high-dimensional vectors and, given an input vector, searches its contents in parallel for the nearest vector. The AM is similar to a content-addressable memory (CAM), which is a memory system dedicated to searching for a perfect match between the input data and its stored data. Nearest neighbor search is an essential part of VSA classification algorithms. Two AM architectures, one digital and one analog, are designed and compared.**

I. Introduction

Vector Symbolic Architecture (VSA), also known as High-Dimensional Computing, is a means of symbolic reasoning that resembles cognition in the brain. Inspired by the high dimensional firing patterns created by populations of neurons in the brain, computing with high-dimensional (HD) vectors is explored. HD computing shows promise in many machine learning tasks, such as EMG and EEG classification. In this paper, a language recognition task will be considered.

In general, HD classification algorithms transform data into a high dimensional vector space in such a way that data from the same class will end up clustered in one region of the vector space. This transformation is achieved by an encoder which specific to each application. The encoder uses vector operations, such as circular convolution and addition, to associate and superimpose orthogonal vectors into a single vector [1].

The information of a symbolic vector is evenly distributed among its elements. If some elements become corrupted due to circuit errors, the vector still contains a good approximation of its symbol. This is unlike the binary encoding of numbers widely used by digital computers, where most of the information is contained in the most-significant bits of a number. Machine learning algorithms employing VSA demonstrate a graceful degradation in accuracy with increasing memory bit error rate, while algorithms that use binary codes experience a sharp increase in circuit failure at lower bit error rates [2]. Thus VSA circuits can operate at a wide range of voltages to suit the needs of a specific application; meanwhile, digital circuits are limited to higher voltages, in particular by SRAM noise margins [3] and the sensitivity to increasing bit error rates.

In order for vectors to be compared for similarity, a distance metric must be defined. In this paper, vectors will have binary elements and the distance metric will be the Hamming Similarity (S), or the number of matching elements between two vectors. In a classification task,

a vector generated by data from the test set is compared with the vectors representing each class from the training set. Since vectors are high dimensional (often the dimension D=10,000 [2] [4]), performing this search on a computer processor would be very expensive due to the amount of data required to be moved out of memory.

Due to this inefficiency, a specialized circuit is explored called an associative memory. The associative memory architecture brings the logic required to compute the hamming similarity, S, as close to the memory as possible. In this paper, two associative memory architectures will be explored and compared. One is a digital circuit computes S and finds the maximum S, S_{max}, comparisons with logic gates. The other is an analog circuit, which computes S and finds S_{max} by generating an measuring analog voltages.

II. Associative Memory

In general, an associative memory (AM) receives noisy data and searches its contents for similar data. In the context of Vector Symbolic Arithmetic, data is in the form of high-dimensionl vectors, and the AM uses the Hamming distance metric in order to determine the similarity between two vectors. This memory is often referred to as a "clean up" memory, since it can return a clean version of a noisy input vector.

An associative memory is similar to a content-addressable memory (CAM). A CAM is a memory circuit that receives a data word as input and searches through its stored data in order to find a word that matches the input word. The specialized hardware of a CAM has low search latency, due to its circuit-level parallelism. These types of search applications are useful in associative caches and network look-up tables [5].

Both AMs and CAMs are designed to search through all of its memory in parallel. Their memory cells, depicted in Figure 1, are similar to 6T SRAM cells but have 4 transistors added (Q2-Q5 in Figure 1) to determine if an input bit matches with the memory bit. The cells of each word in memory share a common wire, called a matchline (ML).

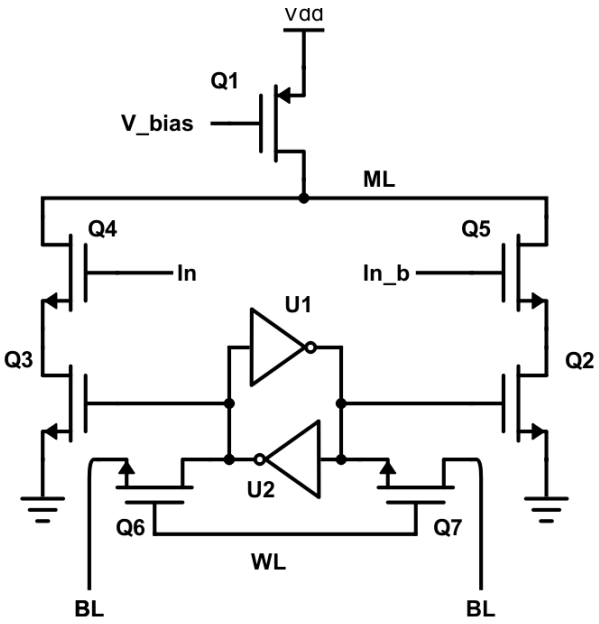

Figure 1: Associative Memory Cell. Four NMOS are added to a standard 6T SRAM cell. If the value of the input data, In, matches the value stored in the SRAM cell, one of the two NMOS stacks (formed by Q2 and Q5, or Q3 and Q4) will conduct current from the ML, ML.

In a CAM, to perform a search, the ML is pre-charged to VDD and the input data is presented so that a mismatch between a bit of the memory word and the input word causes the CAM cell to conduct current and discharge the ML. If all bits match then no CAM cell will discharge the ML and the ML voltage remains at VDD [5].

The AM architecture is similar to the CAM architecture. The AM uses the same memory cells and common ML as a CAM. However the AM stores and compares high-dimensional binary vectors, while the CAM compares standard-length binary words. Another major difference between the two is the precision with which the number of matches must be measured.

In HD applications, two vectors will almost never be equal, due to the assumptions of VSA. Thus the associative memory needs to search for *close* matches while the CAM is searching for *perfect* matches. This requires that the AM have more precision than the CAM in the generation and measurement of the analog current representing the Hamming similarity between vectors, $S(V_1, V_2)$.

The AM cell computes the XNOR function between the input and the memory bit as an analog current using Q2-Q5 in Figure 1. Since each cell is connected to a common node, the ML, the total change in charge of the ML is equal to the sum of the charges drained through each matching AM cell, by Kirckoff's Current Law. A PMOS load, Q1 in Figure 1, is added so that the total amount of current drained by the matching AM cells passes through the PMOS load. The PMOS load converts the current into a voltage proportional to the number of matching AM cells. Finally, each matchline voltage must be compared to find the best match.

III. Comparing Analog and Digital Implementations

Given two HD vectors, V_1 and V_2, to compute $S(V_1, V_2)$ the number of matching elements must be counted. This can be achieved by first computing the XNOR between each element of the two vectors, and then counting the number of 1's contained in output of the XNORs.

No matter the method, computing $S(V_1, V_2)$ requires a reduce operation in which the distributed information held by an HD vector is compressed into a numerical representation. Thus S, being a real number-valued metric, does not exist in the vector space and thus its encoding does not necessarily have the noise-tolerance of high-dimensional vectors. It may be worth introducing error codes to make the metric code more robust to noise [6]. Due to the seeming incompatibility with real numbers in HD computing, encoding the metric with analog voltage or current will be explored in the bulk of this paper.

The challenge of computing this sum in the digital domain is that the binary representation used by standard digital logic is not as robust to errors as high-dimensional vectors of VSA. Since the information contained in an HD vector is distributed evenly among all of its elements, the information is robust to noise. This is not the case for the binary representation of numbers, where errors in the most significant bits can cause catastrophic errors. Additionally, any errors in the control logic used by the digital architecture will lead to incorrect execution of the algorithm. To avoid these catastrophic errors requires that the logic responsible for control and numeric representations be more reliable, and thus at a higher voltage than the local elementwise logic operating on HD vectors.

Digital logic also requires more circuit complexity. Computing $S(V_1, V_2)$ in a D-dimensional vector space using digital logic first requires requires D CMOS XNOR gates. Then the outputs must be accumulated by an adder tree, which scales with vector dimension, D, as $log(D)$ [7], using

the majority of the circuit area. Not only is this circuit large, but, since it uses the binary encoding, it cannot be treated as an error tolerant circuit. Meanwhile, the memory and XNOR circuits are more resilient to errors because they are dealing with data in the HD vector space. In a physical implementation, all the circuits will be made with the same process, however the circuits operating on binary-encoded data will be limited with respect to the minimum supply voltage due to their relatively poor error resilience.

As discussed in the previous section, the analog implementation eliminates the need for an expensive digital accumulator by accumulating charge on a common node. In this way, the analog computation of S requires less complexity. However the drawback is that this current is affected by and device variations, reducing the resolution of the computation. It will be shown in the next section that, in HD applications, in which many similarities need to be compared, the largest similarity, S_{max}, is usually significantly larger than all other distances. Thus, an analog circuit with less resolution than a digital circuit is sufficient to distinguish the best match.

IV. Analysis of Digital Implementations

The digital architecture design brings the logic close to the memory. Each vector is stored in a register is connected to elementwise XNOR gates that compute bitwise matches with the input vector. The bit matches are counted by an adder tree which computes the similarity between the vector stored in memory and the input vector. This logic is copied for every row of registers, so that the architecture computes all similarities in parallel. Finally, each row outputs its similarity to a combinational circuit that finds the maximum.

Compared to a general computer processor design, this specialized digital circuit requires lower latency and energy. A general processor must store the AM vectors in an SRAM, which creates a memory bottleneck. However, providing each vector in memory with its own logic significantly reduces the density of the memory and increases the area of the AM circuit.

The adder tree modules take up the majority of the area of the circuit. Adder trees scale with vector dimension, D, as $log(D)$ [7].

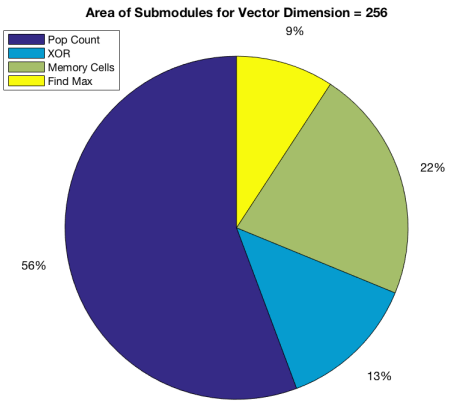

Figure 2: Area breakdown of submodules of the parallel AM architecture.

The plot in Figure 2 suggests that the area scales linearly with vector dimension, D. The logic synthesis and routing tools struggle with AM architectures of higher vector dimension, however the data in Figures 2, 3, and 4 can be extrapolated to higher values of D. For D=8192, the area of

the parallel architecture is $8mm^2$.

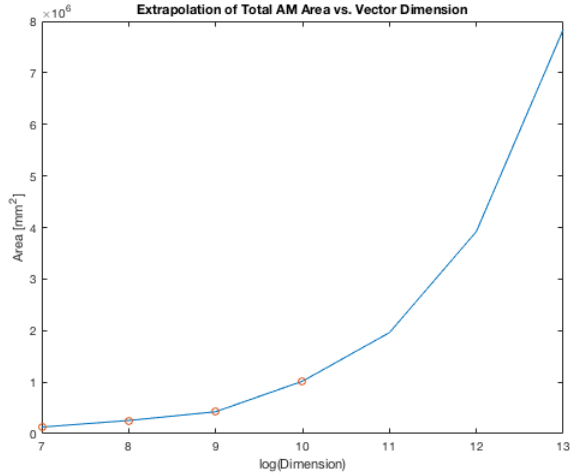

Figure 3: Total AM area vs vector dimension, D. Data points from place and route simulation are shown by the red circles. The linear relationship is extrapolated to dimension D=8192.

However there is a significant advantage with energy and latency, as shown in Figure 4 and Figure 5.

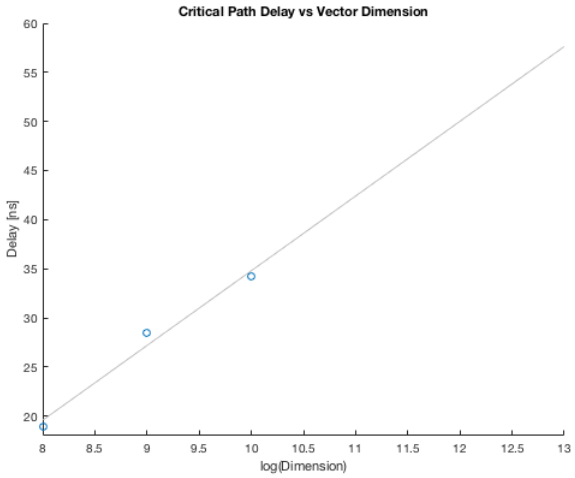

Figure 4: Latency vs Vector Dimension, extrapolated to D=8192.

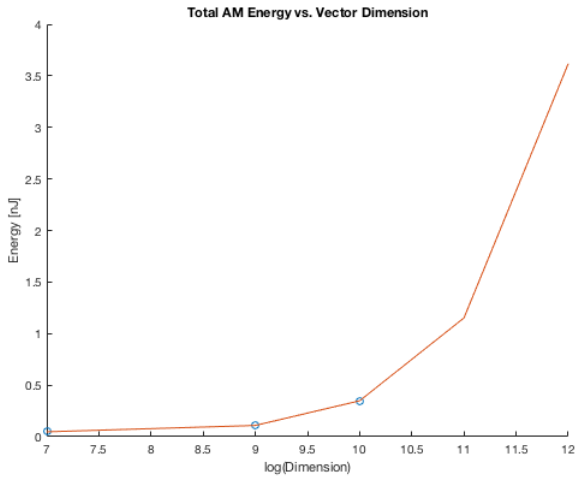

Figure 5: Energy per Classification vs Vector Dimension, extrapolated to D=8192.

These results suggest that the parallel AM is ideal for low-energy applications in which area is not a concern. Migrating the design to a more advanced technology, such as 28nm, will reduce the area to more manageable values. For applications requiring smaller area, an iterative algorithmic approach, such as a processor, should be used, at the cost of greater latency and energy per search.

These results can also be used to compare with the analog implementation described in the next sections.

V. Design of the Matchline Circuit

In the analog associative memory, the ML circuit is responsible for computing $S(V_1, V_2)$, V_1 being stored in SRAM cells, and V_2 being presented to the circuit as input. To implement the memory cells, the standard 6T SRAM cell is augmented with 4 NMOS transistors in two stacks, as shown in Figure 1. If the input bit matches with the memory bit, one of the two stacks will conduct current and pull the ML voltage, V_{ML} low by a small amount. V_{ML} will decrease further as the number of matching AM cells increases.

The ML is a common node shared by all memory cells of the stored vector. When In and In_b are driven by input data, the number of cells with matching data will determine the amount of current conducting through each PMOS load, Q1, in Figure 1. Then, assuming linearity, V_{ML} will then settle to a value that is proportional the number of matching vector elements.

The linearity of the transfer function, $V_{ML}(N_{bits})$, shown in Figure 6, where N_{bits} is the number of matching vector elements, depends on the linearity of the resistive divider formed by the PMOS load and NMOS in the AM cells. Linearity is maximized when both the PMOS and NMOS are in saturation. Therefore, the operating condition of the matchline is that, $V_{th,n} < V_{ML} < V_{dd} - V_{th,p}$. From this, the full scale voltage range is defined to be $V_{FS} = V_{dd} - V_{th,p} - V_{th,n}$. Notice in Figure 6 the nonlinearity appearing at around 0.75V.

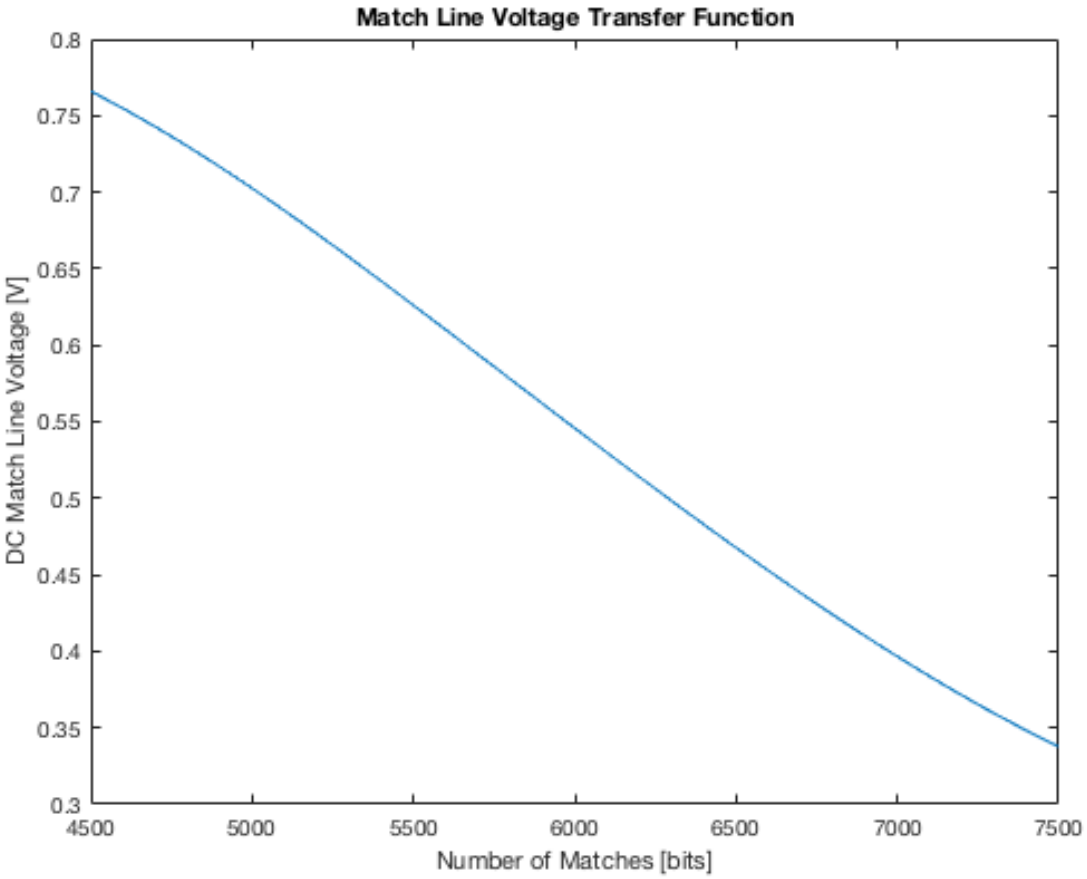

Figure 6: Transfer Function of Matchline Voltage. Note the nonlinearity appearing as Match Line Voltage approaches the transistor threshold voltage.

A. Effects of Variability and Mismatch

The four NMOS transistors Q2-Q5 in Figure 1 are subject to process variations originating from imperfect sizing and doping. This variability affects how strongly each memory cell affects the equilibrium V_{ML}. The variation in cell strength causes a variation in equilibrium V_{ML} with a constant number of memory cells on.

Independent variations are mitigated due to the high number of memory cells connected to the ML. This is due to the Central Limit Theorem from Probability Theory, which states that the mean of samples from an independent random variable approaches the true mean of the distribution as the number of samples increases. The current of each AM cell can be thought of as a random variable with a distribution depending on the device models. Whatever the distribution, the CLT states that the mean of the random currents will have a gaussian distribution with mean equal to the true mean of the distribution of individual currents. Additionally, as the number of samples increases (in our case, as the number of AM cells increases), the variance of the observed mean, σ^2_{sample}, will decrease as $\frac{\sigma^2}{n}$, where σ^2 is the variance of the original distribution of currents [8]. Many HD applications use vectors with a high number of dimensions, D, such as $D = 10{,}000$, which means there will be D AM cells per ML, and thus the distribution of currents will be samples D times. This many samples leads to a mean very close to the true mean of the distribution, and a

very small variation of matchline voltages. A Monte Carlo simulation, shows that over 200 trials modelling independent process variations, the equilibrium V_{ML} has a standard deviation of $\sigma_{V_{ML}} = 2.6\,mV$.

Unfortunately, systematic variations between matchlines do have a significant effect on V_{ML}. If many cells on a ML are stronger than average, the ML will be more likely to indicate a larger similarity to the input vector than other MLs. A Monte Carlo simulation confirms this; over 200 trials in which systematic process variations are modelled, $\sigma_{V_{ML}} = 143\,mV$. This leads to a very low resolution in the computation of similarities and causes a significant amount of mispredictions.

To mitigate the problem of systematic process variations, for each ML a feedback amplifier is used to set the V_{gs} of the PMOS load to equalize the V_{ML} voltages.

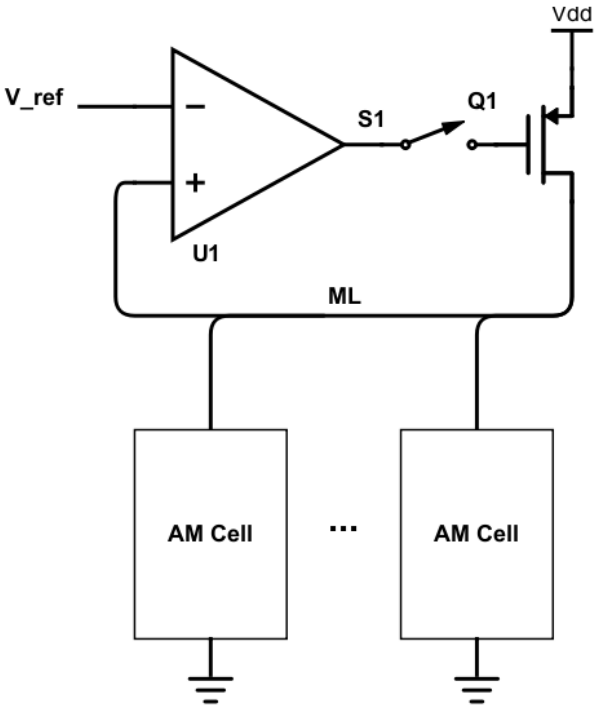

Figure 7: Feedback configuration to calibrate the matchline voltages in the presence of systematic process variations.

This setup requires an initial calibration phase which sets the V_{gs} of Q1, in Figure 7, for each ML in the memory. During the calibration phase, switch S1 of every ML is closed and a random vector is input into the memory. With the switch closed, the negative feedback drives V_{ML} to a desired global reference voltage, V_{ref}. This way, every ML settles to the same voltage, V_{ref}.

The vector input during the calibration phase is randomly generated, so that it is uncorrelated with all the vectors stored in the associative memory. This way, the strength of each PMOS load, Q1, is set so that each V_{ML} is equal while each ML has approximately half of its memory cells on. After the system has settled, switch S1 is opened and the charge stored on the gate of Q1 maintains the calibration. Switch S1 is a CMOS transmission gate made of high threshold-voltage transistors, so the charge leaks away relatively slowly, at $30\frac{mV}{us}$. Extra capacitance can be added to the gate of Q1 to decrease the voltage error accumulated over time.

It is important to note that the calibration is being done using a subset of the ML's AM cells,

since only half of the AM cells are on, approximately. After calibration, when a valid vector is input to the AM, a different subset of AM cells will be on. In statistical terms, different vector inputs draw different samples from the distribution of currents. Again, according to the CLT, the mean current, and thus the equilibrium V_{ML}, will have a small variance if D is large. Thus, using just one uncorrelated input, we can calibrate the transfer function such that $V_{ML}\left(\frac{D}{2}\right) = V_{ref}$, approximately, for all uncorrelated inputs.

After including the feedback amplifier, a final Monte Carlo simulation shows that despite systematic variations, $\sigma_{ML} = 15mV$. This will be shown in the results section. The other analog components in the AM are designed to have similar voltage errors. The total voltage variation contributing to the error in the calculation of similarity, σ_T^2, is found from including the offset voltage of the comparator connected to the ML. σ_T^2 will be referred to as the "noise limit," and is derived from $\sigma_T^2 = \sigma_{ML}^2 + \sigma_{SA}^2$, where σ_{SA} is the offset voltage of the comparators.

B. Matchline Resolution

The matchline is essentially a digital-to-analog converter (DAC). The digital input to the DAC is the input vector to the AM, and the analog output is V_{ML}. While most DACs use the binary digital code as input, the ML uses a unary code (or thermometer code) as input, with one AM cell for each digit of the thermometer code. Each matching AM cell provides a 1 to the thermometer code.

Another key difference between the ML and a DAC is the nature of the input signal. A DAC generally deals with time-varying signals, so it is useful to consider dynamic performance in the time or frequency domain. On the other hand, the input signal to the AM is just a single vector (often the vector encodes a time varying signal [4]). Consecutive vectors are from different data samples, so there is no relationship between them, and thus no notion of time. The lack of time means that frequency domain metrics do not apply. Despite the different nature of input signals, common metrics such as quantization error, resolution, and Signal to Quantization Noise Ratio (SQNR).

The quantization noise can be defined in the usual way, $\varepsilon_{QN} = \frac{\Delta}{\sqrt{12}}$, where Δ is the minimum step size of V_{ML} [9]. Given the full scale input range, R_i, $\Delta = \frac{\mathrm{V_{FS}}}{R_i}$. Figure 8 shows the distribution of hamming similarities generated by the AM for a language recognition task [2]. From Figure 8, R_i need not be greater than 2000 bits, since all similarities lie within 5000 and 7000 bits.

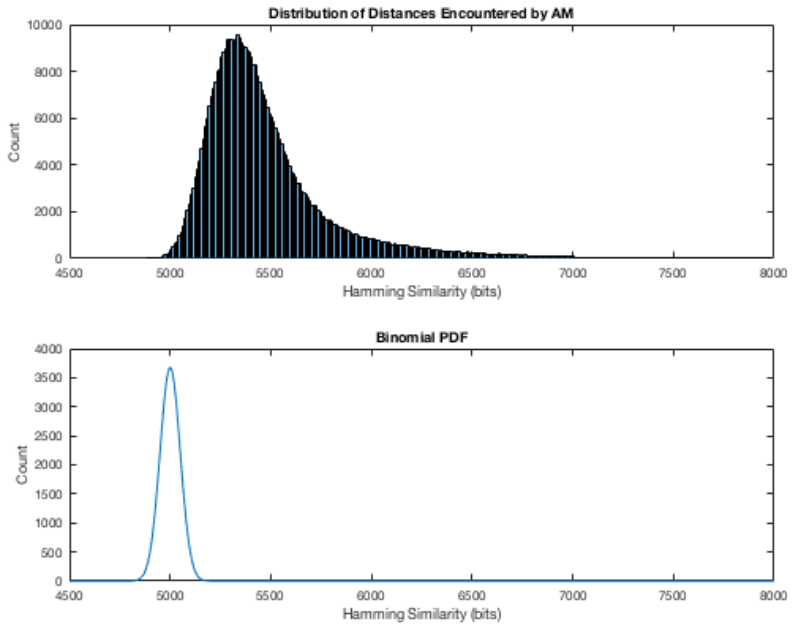

Figure 8: The top plot shows the distribution of similarities of vectors generated by a language recognition task and indicates that the required input range of the ML circuit is approximately 5000 to 7000 bits.. The bottom plot shows the distribution of similarities between randomly generated vectors. It is apparent that the presence of correlated vectors creates a long tail in the distribution.

Given $R_i = 2000$ and $V_{FS} = 500mV$, quantization noise error can now be determined from $\varepsilon_{QN} = \frac{\mathrm{V_{FS}}}{R_i\sqrt{12}} = 72\mu V$. ε_{QN} is significantly smaller than the noise due to process variations, $\sigma_{V_{ML}} = 15mV$, therefore ε_{QN} will be ignored, and the minimum output voltage step size is considered to be

$$\Delta = \sigma_T = \sqrt{\sigma_{ML}^2 + \sigma_{SA}^2}.$$

To find the SQNR, the signal power can be found from the expected value of V_{ML} over many trials. Assuming that V_{ML} has a uniform probability distribution over the range $[V_{th,n}, V_{dd} - V_{th,p}]$, or equivalently $[0, V_{FS}]$, the signal power is found from the expected value of V_{ML}^2,

$$E[V_{ML}^2] = \int_0^{V_{FS}} \frac{1}{V_{FS}} V_{ML}^2 dV_{ML} = \frac{V_{FS}^2}{3}.$$

The noise power is $\varepsilon_N = \frac{\Delta^2}{12} = \frac{\sigma_T^2}{12}$, ignoring quantization noise. The SQNR is then [9],

$$SQNR = \frac{P_{signal}}{P_{noise}} = \frac{E[V_{ML}^2]}{\varepsilon_{\mathrm{N}}^2} = \frac{4V_{FS}^2}{\sigma_T^2} = 36.5\ [dB].$$

The SQNR can be increased slightly by increasing the supply voltage to 1.2V. A more accurate approximation of the SQNR can be found by transforming the distribution of hamming similarities,

S, in Figure 8 to $V_{ML}(S)$ by using the ML transfer function in Figure 6.

The resolution, N, in bits, is defined as $N = \log_2 \frac{V_{FS}}{\Delta} = 5.06$ bits [9]. This means that $2^N = 33$ discrete signal levels can be resolved. However, the input signal range, $R_i = 2000$, many more than 33. This means that, between two vectors, the number of matching elements that can be resolved is

$$dS = \frac{R_i}{2^N} = R_i \frac{\sigma_T}{V_{FS}} = 60\ bits\ (1).$$

In comparison, the resolution without calibration would be $dS = 2000 \frac{0.143}{1} = 286$ bits.

dS can be decreased by increasing V_{dd}, by decreasing σ_T, or by decreasing R_i. R_i, the input range, is set by the distribution of similarities unique to the data being classified. V_{dd} is limited by the technology used. Therefore, to maximize dS, σ_T must be minimized.

It is important to note that the formula $dS = R_i \frac{\sigma_T}{V_{FS}}$ is an optimistic estimate since it assumes no non-linearity in the transfer function. Any non-linearity in the transfer function, $V_{ML}(S)$, where S is the hamming similarity between the two vectors, will cause dS to be larger for some V_{ML}, so the worst case must be assumed.

In the language recognition task previously mentioned, the AM receives an HD vector encoding information on a sample of text and computes S between this sample vector and all the language vectors stored in its memory. There are 21 different possible languages, so the AM computes the 21 similarities between the input and the language vectors. Figure 9 shows the PDF and CDF of the distribution of the difference between the two highest similarities found during the classification of a single text sample.

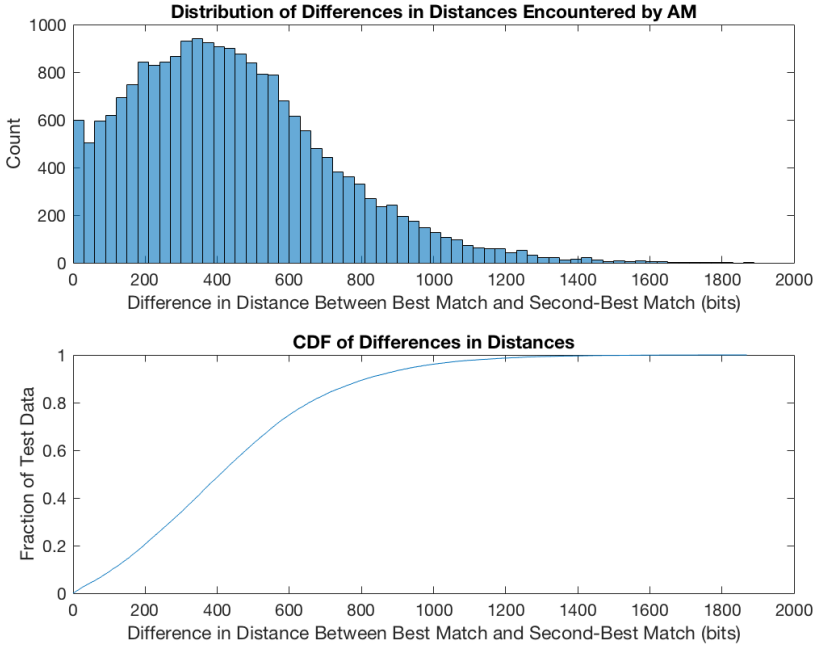

Figure 9: The top plot shows the histogram of differences between the smallest and second smallest distances. The bottom plot shows the CDF of this distribution, which gives a sense of the error rate of the analog computation given a certain resolution.

The CDF in Figure 9 shows that 0.267 of the test data requires a resolution less than dS=286 bits

in order to be classified correctly. This resolution would contribute a significant amount of error into the final classification accuracy. In comparison, after the calibration circuit is added and dS=60bits, from the CDF 0.0419 of the data requires a resolution of less than 60 bits. So an AM with this resolution would not contribute a significantly to the error rate of an algorithm with error rates above 4%.

C. Design of the Feedback Amplifier

In order for the negative feedback to drive $V_{ML} = V_{ref}$, the loop gain of the calibration circuit must be large. However, some gain error can be tolerated, since each matchline will have roughly the same gain error. This is fortunate because the intrinsic gain of the high-theshold-voltage (HVT) transistors used is quite low.

The amplifier design, shown in Figure 10, is chosen to be a PMOS-input differential cascode. To avoid instability, only one stage is used. Since the amplifier is driving a PMOS gate, its lower output voltage limit should be as low as possible. The bias voltages for the cascode transistors Q5-Q8 are generated by a matching network. A current source is generated by a PTAT current source [10].

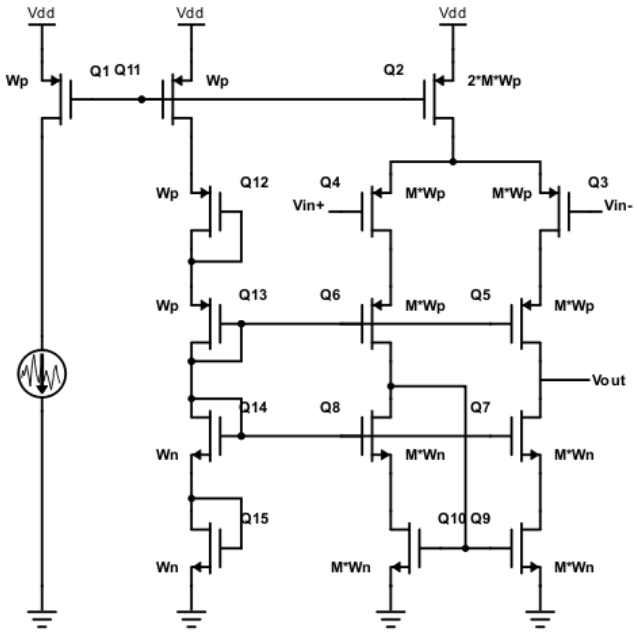

Figure 10: Differential-Input Single-Ended-Output Cascode Amplifier.

The two poles in the feedback loop are at the gate of the PMOS load and at the drain of the PMOS load. The pole at the PMOS gate is dominant because this node has higher capacitance and resistance than the node at the PMOS load. First, the gate capacitance is larger than the drain capacitance for similarly sized transistors. Additionally, the feedback decreases the resistance at the PMOS load to approximately $\frac{1}{A_{OL}g_m}$, where A_{OL} is the open loop gain of the amplifier and g_m is the transconductance of the PMOS load. Meanwhile, the resistance at the output of the amplifier is $g_m r_o^2$, which is much larger than $\frac{1}{A_{OL}g_m}$.

Settling time is not a priority since the calibration phase does not need to happen prior to every

search. Additionally, due to the cascode configuration, headroom is a concern. Thus, rather than bias the transistors to have the optimal gain-bandwidth product, the transistors were sized to have high $\frac{g_m}{I_d}$, and thus low minimum V_{ds}.

Finally the transistors widths, M, can be determined by the desired tradeoff between phase margin and settling time.

VI. Design of the Comparator Circuit

The voltages on each ML must be compared in order to find the smallest distance. There are two ways to do this: by comparing the ML voltages with each other in a tree structure as shown in Figure 11, or by comparing each V_{ML} to a common reference voltage as shown in Figure 12. Both Figure 11 and Figure 12 depict a comparator circuit for 4 MLs. Both structures can be expanded to compare multiple MLs. The circuit in Figure 11 requires N-1 comparators to compare N MLs (assuming N is a power of 2), while the circuit in Figure 12 requires N comparators to compare N MLs, so the complexity of each circuit scales similarly.

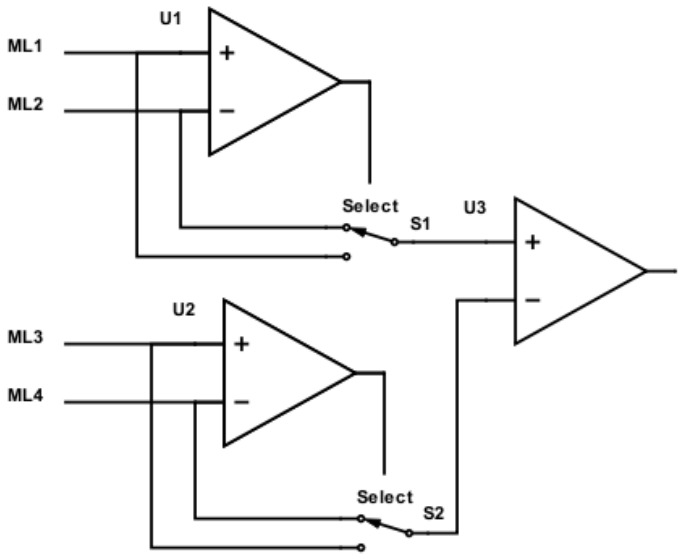

Figure 11: Comparator Tree

In the tree structure, something must drive the comparator inputs at each stage of the tree. If the ML circuit drives the tree structure, the ML must drain current for

$$\tau_{ML} + (\tau_{SA} + \tau_{switch})log(C),$$

where C is the number of vectors being compared, τ_{ML} is the time required for the matchline circuit to drive the input capacitance of the sense amps, τ_{SA} is the delay of one sense amplifier and τ_{switch} is the delay of a switch. With high vector dimension, there are many AM cells draining current, and the total current is very large, on the scale of milli-Amps. Thus, to save energy, the ML circuit must remain active for as little time as possible.

In order to reduce the amount of time the MLs actively drain current, track-and-hold amplifiers can be used to capture the matchline voltage and drive the tree structure. This way, the matchline circuit only must remain active until it has settled to equilibrium and driven the track-and-hold

amplifiers, or $\tau_{ML} + \tau_{T\&H}$, where $\tau_{T\&H}$ is the extra time it takes the matchline to drive the track-and-hold amplifier input capacitance. Using a track-and-hold amplifier would reduce latency and energy consumption since it can be designed so that $\tau_{T\&H} < (\tau_{SA} + \tau_{switch})log(C)$.

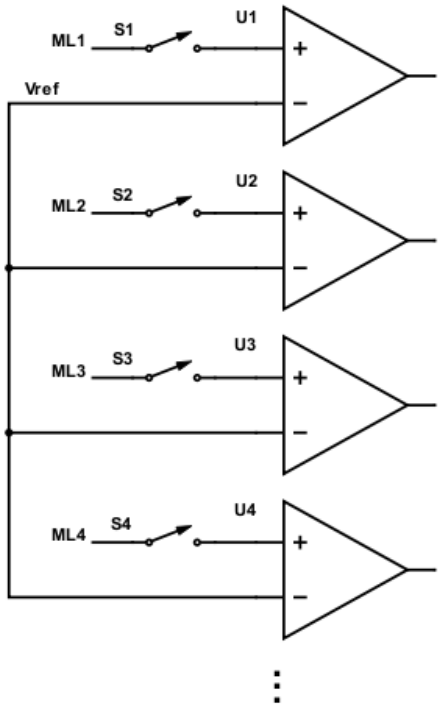

Figure 12: Alternate comparator structure which compares each V_{ML} with a global reference voltage. The switches S1-S4 represent track-and-hold amplifiers.

The global reference structure can be implemented in a similar fashion, by using track-and-hold amplifiers to reduce the amount of time the ML is active. This circuit uses a global reference voltage as a threshold; if $V_{ML} < V_{ref}$, then the sense-amplifier output latches low, indicating similarity. The reference voltage, V_{ref} is a free variable that needs to be controlled so that it successfully distinguishes the best match from the rest, which occurs when the value of the reference voltage lies between the lowest and the second lowest V_{ML}. This search can be performed by a digital controller.

In this paper, the global reference structure is explored and the track and hold amplifier will not be used; instead, the matchline circuit directly drives the comparator inputs. However the results presented in later sections suggest that the tree structure may have lower latency and thus lower energy consumption.

A. Design of the Comparator

A comparator is required for each ML to determine whether the V_{ML} is greater or less than the global reference voltage. In order to minimize the offset voltage below the V_{ML} noise due to process variations, a latched sense amplifier topology is considered: a simple cross-coupled sense amplifier, depicted in Figure 13.

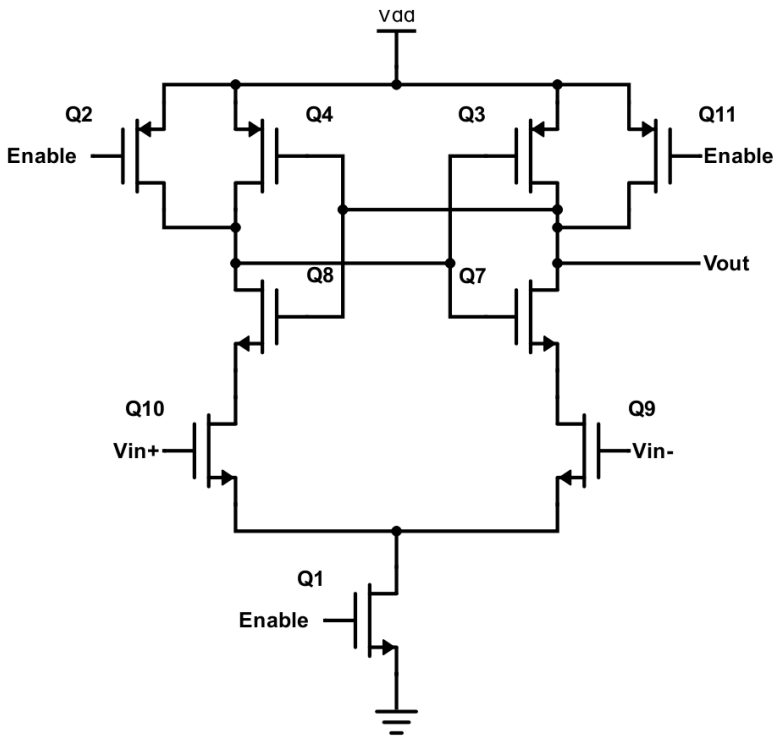

Figure 13: Latched Cross-Couple Sense Amplifier

In this design, when the Enable voltage is low, the cross-coupled inverter nodes are both forced to Vdd, while the NMOS tail is disabled. When the Enable voltage transitions to Vdd, the cross-coupled inverters are allowed to settle to a stable state. If Vin+ is higher than Vin-, the node at the drain of Q8 is driven lower than the node at the drain of Q7, Vout. The inverter feedback then drives Vout high. If Vin- is higher than Vin+, as similar process drives Vout low.

There is a tradeoff between minimizing the offset voltage and minimizing power and area. For the purposes of the AM, the SA transistors are sized large enough so that the SA makes the correct prediction 95% of the time for an input voltage difference of 10mV, which is less than σ_T.

A offset-cancelling sense-amplifier may be used in order to decrease the offset voltage even further . However, the offset-cancelling phase of these topologies creates extra delay [11]. From the previous section, minimizing delay is necessary to minimize the amount of energy consumed by the AM during a search. To minimize both delay and energy, the faster sense-amplifier is used. Using the fast sense-amplifier comes at the cost of area, since it needs to be sized larger in order to achieve the same offset voltage as the offset-cancelling sense-amplifier.

B. Generating a Global Reference Voltage

As mentioned earlier, the global reference, V_G, must be controlled in some way so that only one comparator indicates similarity. The global reference can be generated by a continuous time circuit, or a discrete time circuit. In order to take advantage of the precision of clocked sense amplifiers, a discrete time circuit will be considered. During each clock cycle, a new V_G is generated by a digital-to-analog converter (DAC). The settling time of the DAC contributes to the critical delay path.

Searching in discrete time also allows for V_G to be generated by a digital-to-analog converter

with a non-zero settling time. This system, which converges to a desired V_G based on some search logic, is similar to a successive-approximation-register digital-to-analog converter (SAR ADC) [9]. In general, the DAC in a SAR ADC finds a binary representation of the desired output voltage one bit at a time over multiple cycles, starting with the most-significant bit. Due to its ability to store state over a series of clock cycles, a capacitive DAC is used in the AM design, as shown in Figure 14 [11].

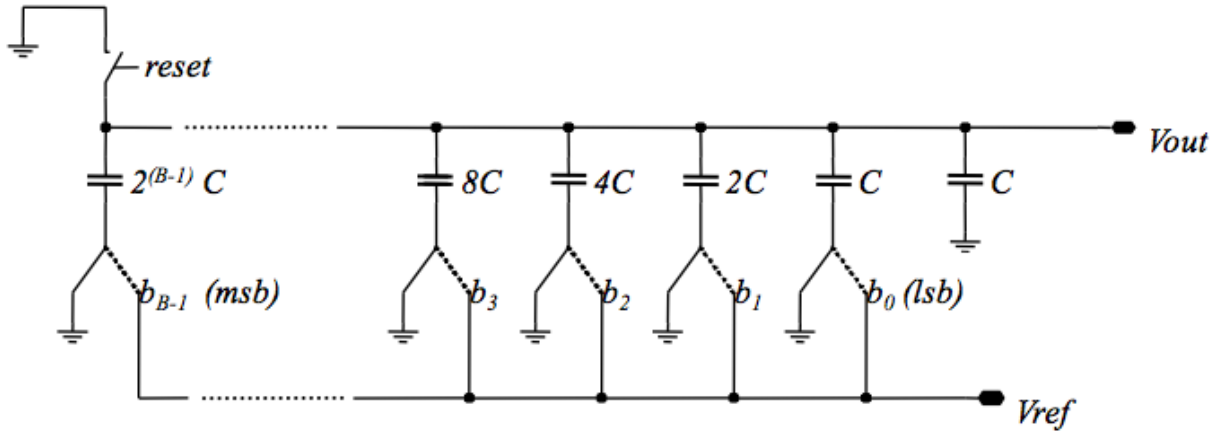

Figure 14: B-bit Capacitive DAC [11].

C. Digital Control of the DAC

In this specific application, the binary representation of V_G is not known beforehand, but must be deduced over multiple clock cycles from the comparator outputs. Remember that, during a search, a lower V_{ML} corresponds to a better match and that if $V_{ML} > V_G$, the comparator output, V_{SA}, is V_{dd}. This means that the circuit seeks a V_G such that only one $V_{SA} = 0V$, or ground. At the beginning of a clock cycle, after the sense amplifiers have been triggered, there are two possible cases to which the DAC must respond. If all n comparator outputs, $\{V_{SA_1}, \dots, V_{SA_n}\}$, equal V_{dd}, then V_G must be increased in an attempt to bring at least one $V_{SA} = 0V$. If more than one comparator output is low, then V_G must be decreased. If exactly one $V_{SA} = 0V$, then V_G is successfully distinguishing the best match from the rest, and V_G is a higher bound for the lowest V_{ML}, $V_{ML_{low}}$, which represents the maximum similarity, $S(V_{input}, V_{i_{max}})$. The index i_{max} is the address to the AM vector most similar to the input vector V_{input}.

It is desirable for V_G to converge to $V_{ML_{low}}$, because then the DAC can also be used to measure $V_{ML_{low}}$. This will be explained shortly. To force V_G to $V_{ML_{low}}$, whenever exactly one $V_{SA} = 0V$, V_G will be increased. This leads to two cases: if all comparator outputs are high, increase V_G, else decrease V_G.

To increase V_G, the input to the DAC, V_{REF}, must equal V_{dd} and to decrease V_G, V_{REF} must equal ground, 0V. Therefore, the control logic can simply be implemented with an AND gate. If the comparator inputs are inverted, so that $V_{SA} = 0V$ if $V_{ML} > V_G$, then the control logic can be implemented with a NOR gate. In either case, a buffer or an inverter is required to help the control logic drive the input to the DAC.

Additionally, flip-flops must be placed between the sense amplifier outputs, $\{V_{SA_1}, \dots, V_{SA_n}\}$ and the AND gate to hold the value of each V_{SA} (which is either V_{dd} or 0V). The sense amplifiers are triggered by a rising clock edge, and the falling clock edge resets V_{SA} to V_{dd}. Thus, V_{SA} is valid

while the clock is high and invalid while the clock is low. The flip flops capture the binary value of V_{SA} so that the AND gate can hold its output to the DAC after the falling clock edge. To avoid the hold-time and settling-time violations of the flip-flops, the clock to the flip-flops is phase shifted 90^o from the clock to the sense amplifiers.

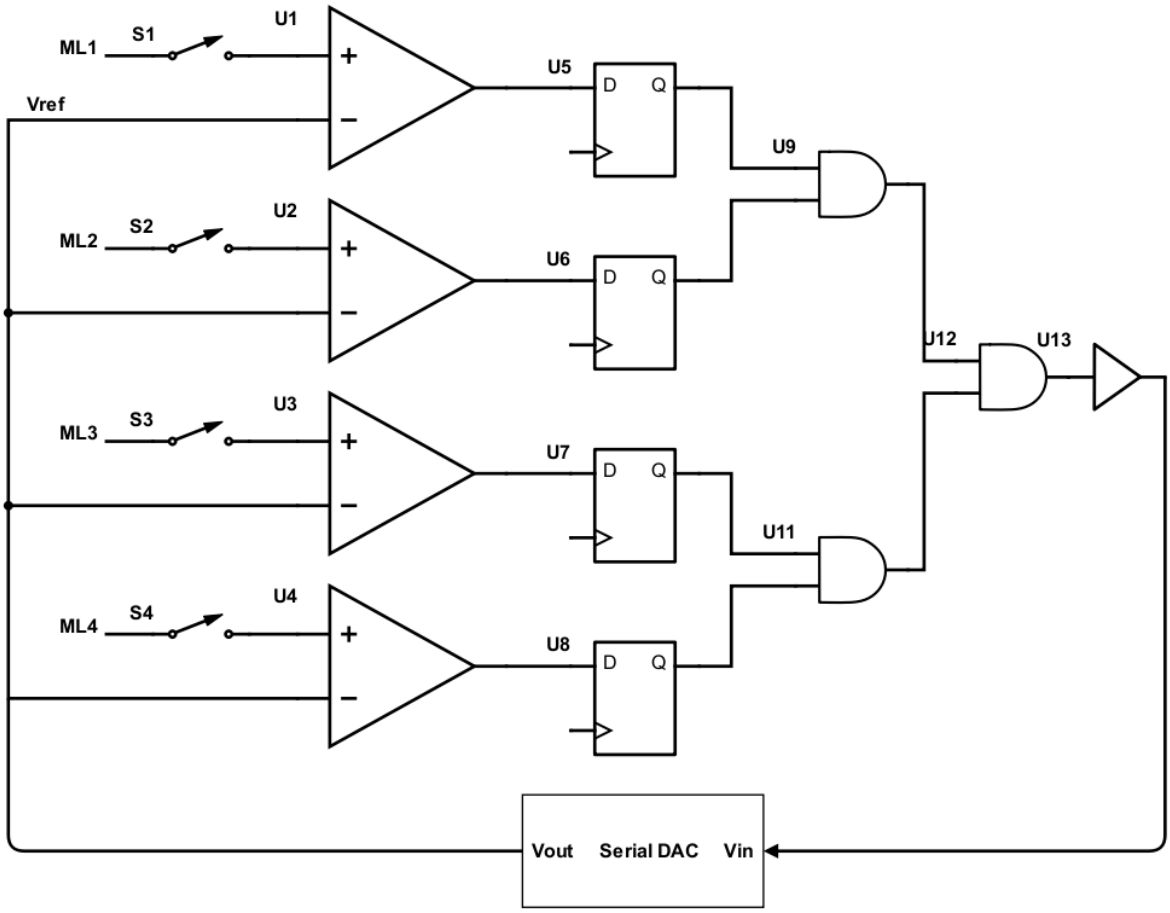

Figure 15: Control Loop consisting of sense amplifiers,, flip-flops, control logic (AND gates), buffer, and DAC. The clock to the comparators is phase shifted 90 degrees from the clock to the flip-flops (clocks not shown). In this circuit, only 4 matchlines are compared. To compare n matchlines, use n comparators, n flip-flops, and an n-input AND gate.

Since V_G converges to $V_{ML_{low}}$, the serial input to the DAC can be interpreted as a binary number proportional to $V_{ML_{low}}$. This binary number must be stored in a register in order to properly control the capacitive DAC. Thus this configuration using a serial DAC finds both the address to the vector that best matches the input vector, i_{max}, as well as the maximum similarity $S(V_{input}, V_{i_{max}})$.

After a certain number of cycles, V_G will have converged to a voltage within the noise limit, σ_T. After the first cycle, $V_G = \frac{V_{dd}}{2} = 0.5V$, and after 7 cycles, V_G will have converged to within $\frac{V_{dd}}{128} = 7.8mV$ of $V_{ML_{low}}$, which is below the noise limit. For this reason, the search will terminate after 7 cycles. This also sets the precision required of the capacitive DAC to $\log_2(128) = 7$ bits. Thus a 7-bit capacitive DAC is used.

D. Digital Encoder

Ideally, the comparators collectively produce a one-hot signal, which can be translated into an

address by an encoder. Note that in this case "one-hot" means one signal equals 0V and the rest equal V_{dd} (one might be tempted to say "one-cold"). It is possible for multiple comparators to signal a match at the end of the 7-cycle search. Since each matchline voltage is compared to a global reference, there is a possibility that during a search, the system fails to generate a V_G which distinguishes just one vector from the rest. If multiple matchline voltages are within σ_T of each other, then multiple comparator signals, $\left\{V_{SA_i}, \dots, V_{SA_j}\right\};\ i,j \in \{1,n\}$, may be low at the end of the search. In this case, any of the vectors corresponding to $\{V_{SA_i}, \dots, V_{SA_j}\}$ are suitable choices to return as the best match. This complicates the design of the digital encoder, since in general an encoder assumes a one-hot encoding at the input, and outputs an invalid address if the one-hot encoding is violated. Now the encoder must choose a valid address when multiple comparator outputs are low. One possible design simply uses a chain of multiplexers in series. Unfortunately, the delay of this circuit scales linearly with the number of matchlines, rather than logarithmically. However, the encoder does not lie in the feedback loop, so its delay does not contribute to the critical path of the closed-loop system.

It is also important to note that if the number of cycles per search is fixed, then it is not guaranteed that at least one $V_{SA} = 0V$. Since the system attempts to converge V_G to $V_{ML_{low}}$, it will often be the case that $V_G < V_{ML_{low}}$ at the end of the 7th cycle, in which case all $V_{SA} = V_{dd}$. In this case, no signals are hot, so the encoder does not have any addresses to choose from.

This possibility is resolved by including a second set of flip-flops to store the comparator outputs on a cycle when there are at least one $V_{SA} = 0V$. To selectively write the flip-flops, they must be enabled only when at least one $V_{SA} = 0$. The encoder logic takes the output of this flip-flop as input.

E. Summary of Complete Closed-Loop System

Consider an AM with capacity to store C vectors of dimension D, $X_1, \dots, X_C$. The AM has C rows of memory, each connected to a ML. To initialize the AM, the calibration phase occurs, in which each ML voltage is set equal $V_{ML_1} = V_{ML_2} = \cdots = V_{ML_C}$, for an input vector, $X_{calibrate}$. $X_{calibrate}$ is uncorrelated with each $X_1, \dots, X_C$, so that each row has approximately the same number of on AM cells. The number of on AM cells is equal to the hamming similarity, $S(X_{calibrate}, X_i)$, for $i \in [1, C]$. When all voltages have settled, each feedback loop is opened with a switch, and the charge stored on the gate of the PMOS load maintains the calibration.

Once the AM is initialized, a vector is input to the AM. Each ML will settle to a voltage determined by the relative strength between the PMOS load and the AM cells. The pulldown strength of the AM cells is proportional to $S(X_{input}, X_i)$. Thus, each V_{ML_i} is proportional to $S(X_{input}, X_i)$.

C comparators compare each V_{ML_i} to a global reference voltage, V_G. V_G is generated by a capacitive DAC. In the fashion of a SAR ADC, the DAC is controlled so that V_G converges to $V_{ML_{low}}$, the voltage proportional to the maximum $S(X_{input}, X_i)$. After the search the AM outputs a binary encoding of $V_{ML_{low}}$ and the address of the best match, i_{max}. After some searches, there will be multiple winners, in which case only one is selected as the best match.

VII. RESULTS

An analog AM circuit is simulated in a 65nm technology with a 1V supply voltage. It can store

up to 32 binary vectors of dimension up to 10,000, which requires 10,000 bits per vector. This means that the AM has 32 matchlines, each with 10,000 AM cells, and 32 sense amplifiers.

A. Matchline Analysis

The transfer function of a single matchline is simulated for a bitline with 10,000 AM cells. Sweeping the AM strength by changing the gate voltage on the input NMOS of the AM cells, shows that the resolution decreases for higher NMOS gate voltage (and hence a stronger pull-down network). However, increasing the strength of the pull-down network also increases the non-linearity of the transfer function.

V_{ref} in Figure 7 can be varied in order to control the input range, since the calibration phase sets D/2 bit matches to V_{ref} in the transfer function, where D is the vector dimension. The histogram in Figure 8 shows that D/2, where D is the vector dimension, is the lower bound of measured distances, and thus should be the lower limit of the input range. To achieve this, V_{ref} should be set to the upper voltage limit within the ideal operating region of the matchline, which is about 750mV. Then, the gate voltages of the NMOS pull down network can be varied in order to change the resolution in Volts/bit, as shown in Figure 16.

Different applications will have different distance distributions, and V_{ref} can be chosen so that the input range of the matchline circuit includes all relevant distances for the specific application.

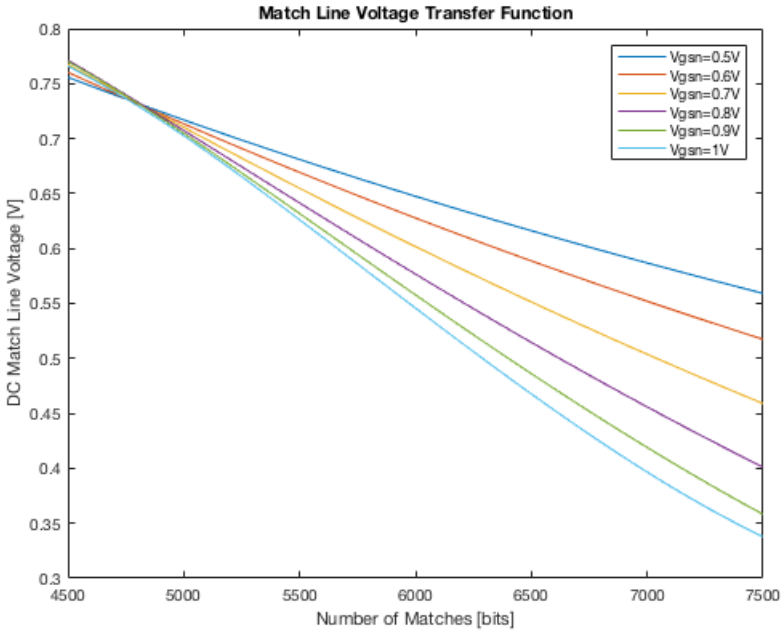

Figure 16: Matchline Voltage Transfer Function. V_{ref} is set to 0.7V.

To measure the linearity of the transfer function, the differential non-linearity, DNL, is used, $DNL = \frac{dV_{actual}}{dV_{ideal}}$. Ideally, the DNL=1 for all bit matches. A plot of the DNL in Figure 17 shows that for NMOS gate voltage Vgsn=0.9, the DNL is within 10% of the ideal for the input range of 5000 to 7000 bits.

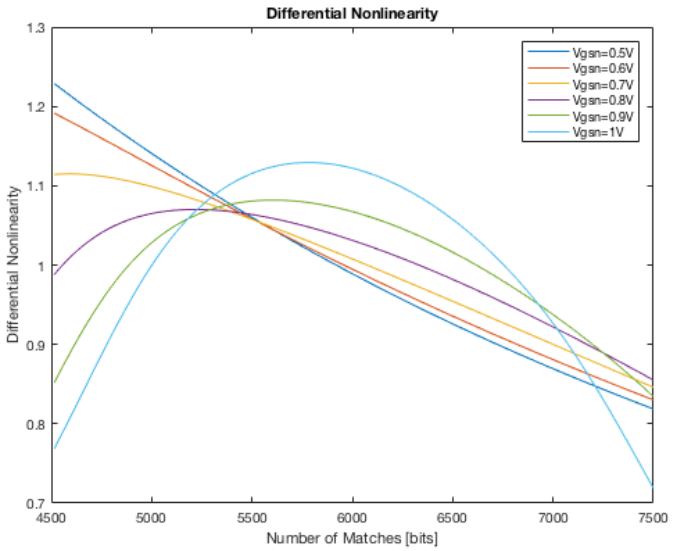

Figure 17: Differential Nonlinearity for different values of Vgsn.

It is important to analyze the effect of process variations. Systematic process variations cause a significant amount of variation of V_{ML}, as shown in Figure 18.

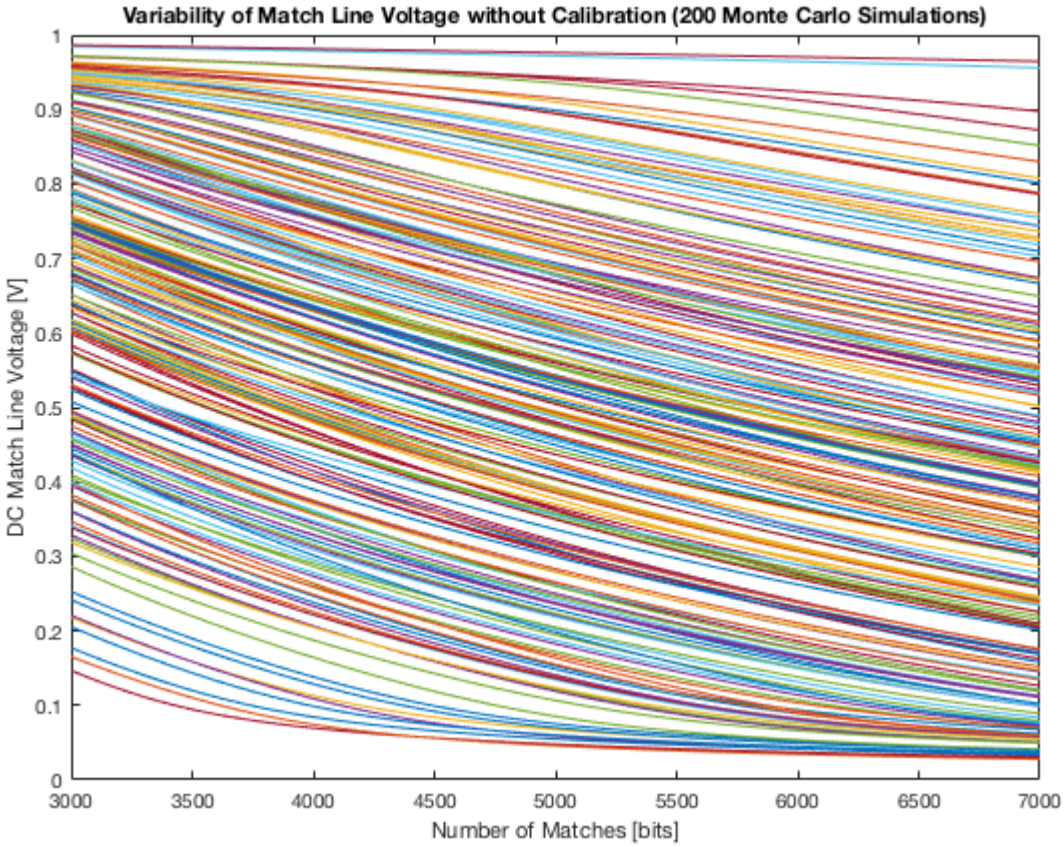

Figure 18: Transfer function $V_{ML}(B)$, in the presence of noise. No calibration is used.

The presence of noise does not greatly affect the transfer function if the calibration feedback is added. As shown in Figure 19, the maximum standard deviation of V_{ML}, across the entire input range, $\sigma_{ML_{max}} = 15mV$. The feedback amplifier has a gain of just 26.

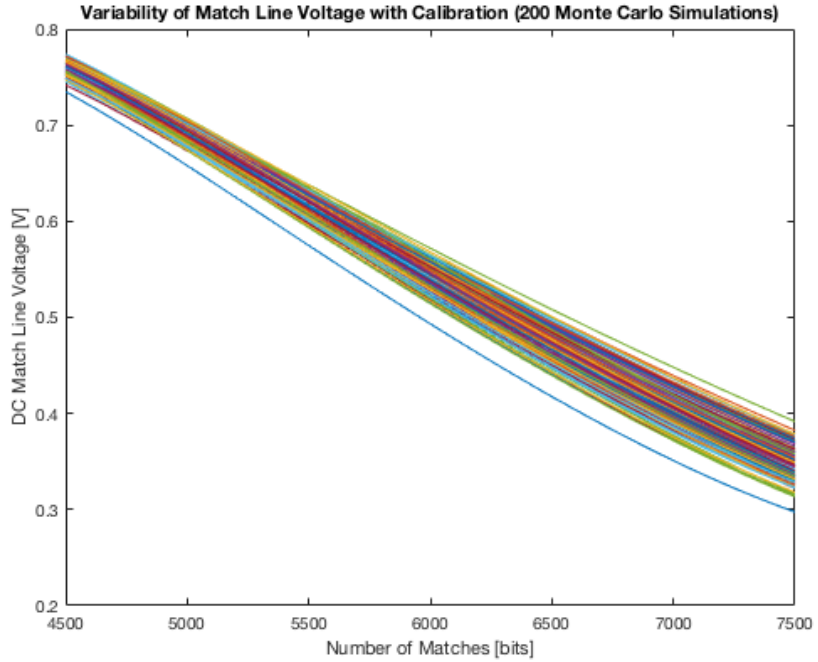

Figure 19: Transfer function $V_{ML}(B)$ in the presence of noise and using calibration.

The DNL is within 15% of the ideal value.

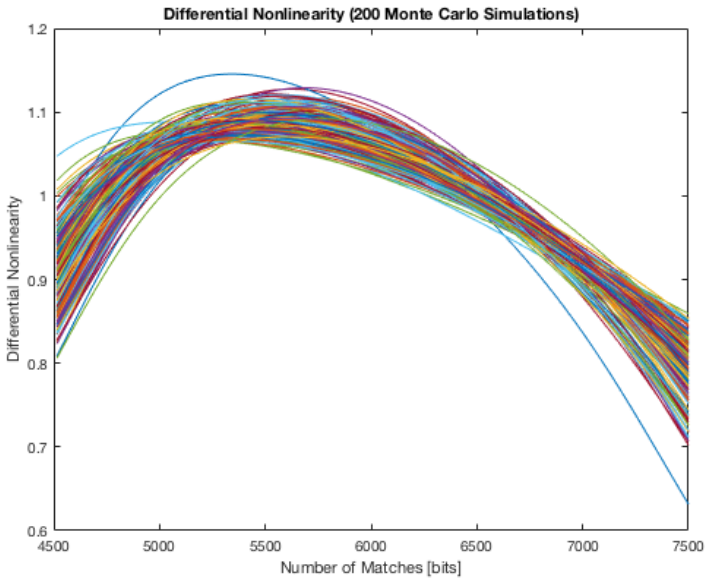

Figure 20: Differential Non-Linearity. Ideally, DNL=1 for all number of matching bits, B.

From the above definition of resolution, $\text{dB} = R_D \frac{\sigma_T}{V_{op}}$, where V_{op} is the voltage range of desirable operating region, $\text{dB} = R_D \frac{\sigma_T}{V_{op}} = \frac{(7000-5000)15mV}{750mV-250mV} = 60\ bits$. However this is a coarse-grained estimate of resolution that assumes no non-linearity. To gain a better understanding of how the resolution changes over the range of matching bits, consider the derivative of the transfer function plotted in Figure 21, in units of Volts/bit.

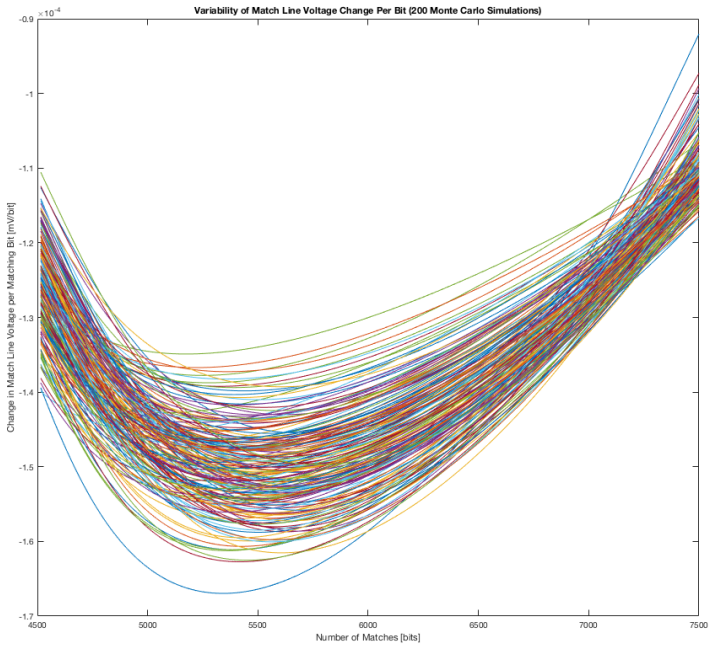

Figure 21: Derivative of $V_{ML}(B)$, where B is the number of matching bits. From this plot, the worst-case resolution can be derived.

Figure 21 shows the derivative is smallest in magnitude at the edge of the input range, $\frac{dV_{ML}}{dS}(S = 7000) = 0.12\frac{mV}{bit}$. Taking into account σ_T, the worst-case resolution can be found from the expression,

$$\sigma_T \frac{db}{dV_{ML}} = \frac{15mV}{\frac{0.12mV}{bit}} = 125\ bits\ (2).$$

(2) is the differential form of (1) derived earlier, and is shown as a function of S, the number of matches, in Figure 22. One would expect that the average value of the differential resolution, (2), would be equal to (1). It is larger, however, because the worst-case $\frac{dV_{ML}}{dS}$ from the Monte Carlo simulation shown in Figure 21 is used.

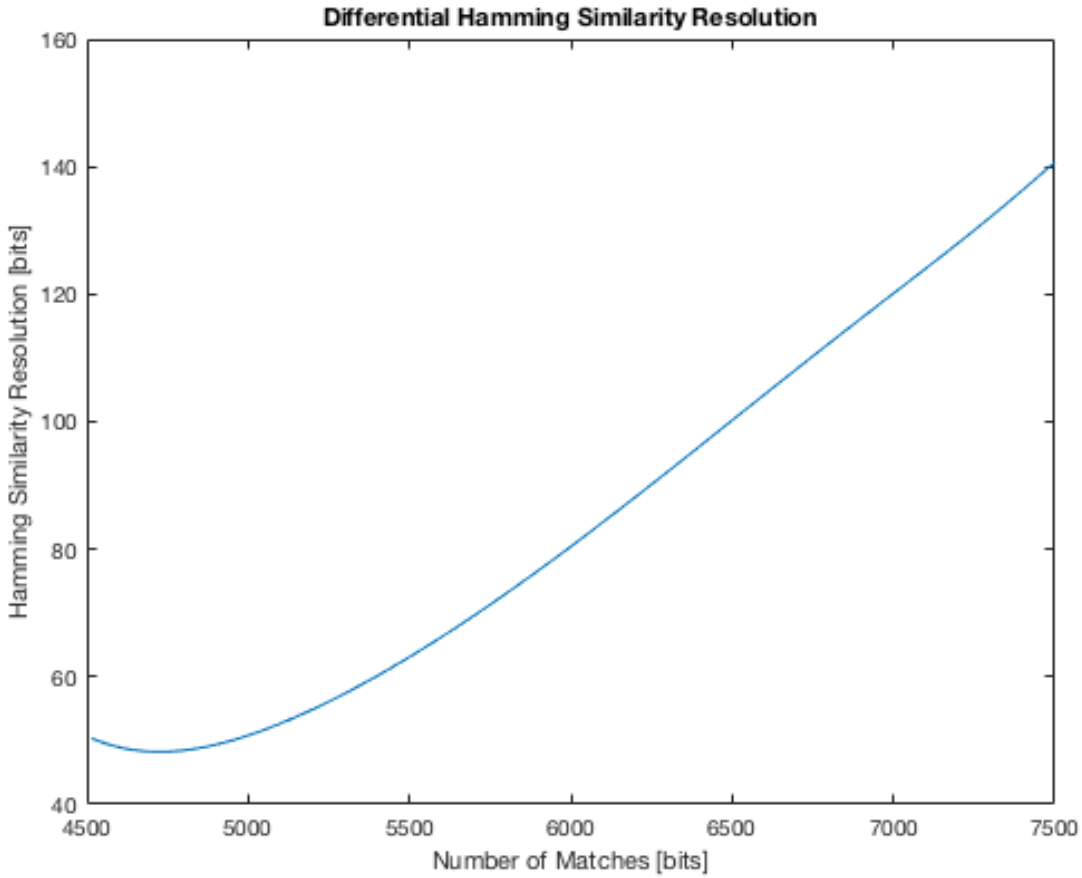

Figure 22:Differential Hamming Similarity Resolution. The y-axis can be interpreted as the minimum difference in hamming similarity that can be resolved at each point within the input range.

Figure 4 shows that in 9.1% of the test cases, the required distance resolution is 125bits or less. If the error rate of the ideal algorithm is greater than 9.1%, then the hardware will not significantly affect the error rate. However, the AM will limit the error rate if the algorithm has a lower error rate.

B. Sense Amplifier Accuracy

The sense amplifier is sized large enough so that given input voltage difference equal to 10mV, the accuracy is at least 95%. Figure 23 is a table showing results for energy, delay, and accuracy of differently sized sense amplifiers. An offset cancelling sense amplifier is also simulated, with results shown in Figure 24. However, since its delay is significantly larger, the simple sense amplifier is a better option [12].

Size	Energy (fJ)	Delay (ns)	Accuracy
$50 * W_{min}$	27	1.3	0.865
$100 * W_{min}$	54	1.3	0.94
$150 * W_{min}$	80	1.3	0.97

Figure 23:Energy, Delay, and Accuracy for a simple cross-coupled sense amplifier. Accuracy is measured by the fraction of correct predictions from an input voltage difference of 10mV.

Size	Energy (fJ)	Delay (ns)	Accuracy
$10 * W_{min}$	604	20	0.92
$20 * W_{min}$	942	20	0.96

Figure 24: Energy, Delay, and Accuracy for an offset-cancelling sense amplifier. Note that the delay is significantly longer, due to a more complicated reset procedure.

C. Transient Simulation

A transient simulation is done to measure the latency and total energy per search operation of the AM. The capacitive DAC has not been implemented yet, but the energy can still be approximated by including its effective capacitance in the loop, and thus its delay, τ_{DAC}. τ_{DAC} is approximated by adding the largest capacitor of the capacitive DAC to the circuit. For a 7-bit DAC, $C_{max} = 128C_{min}$. The RC time constant associated with C_{max} is the worst-case delay of the DAC, and should be used in the critical path of the closed-loop system. The clock period is set to

$$T_{clk} = 15ns > \tau_{CQ} + \tau_{AND} + \tau_{DAC} + \tau_{SA} + \tau_{setup}.$$

Due to the large capacitance on the matchline, $T_{clk} < \tau_{ML} < 30ns$, so the control circuit must wait τ_{ML} before starting the search for $V_{ML_{low}}$. Thus the total delay of the AM is $\tau_{ML} + 7T_{clk} = 135ns$.

Note that T_{clk} is dominated by τ_{DAC}, and thus the total delay is dominated by τ_{DAC}, and to a lesser extent τ_{ML}. The sense amplifiers have a relatively low latency, $\tau_{SA,max} = 1.3ns$. Thus the comparator tree structure discussed earlier would have a latency of $\tau_{ML} + \log_2(C)\,(\tau_{SA} + \tau_{switch})$, where C is the number of vectors stored in the AM. For C=32, $\tau_{ML} + \log_2(C)\,\tau_{SA} = 36.5ns$, approximately.

Given the delay, the energy can be found by integrating the power of each component with respect to time. While the energy consumed by the sense amplifier, the feedback amplifier, the DAC, and the digital control are on the order of pJ, the energy consumed by the matchline is $E_{ML} = 12.5nJ$. This is the energy consumed per matchline. For 32 matchlines, the total energy consumed is 400nJ. To reduce energy consumption, it is imperative to decrease the time the matchline is active by capturing each V_{ML} with a track-and-hold amplifier.

D. *Comparison with Digital Implementation*

The table in Figure 25 compares the two implementations. The digital design uses substantially less energy and has a lower latency. The exploration of an analog design came from the possibility of higher energy efficiency at the cost of resolution. The analog design can likely be improved. Using the comparator tree structure shows some promise, by reducing the latency and thus the amount of time the ML dissipates power. If the latency could be reduced to 36ns, then the total energy would be reduced by a factor of 135/35= 3.86, to 104nJ. This is still much greater than the digital implementation, so further optimizations must be explored.

	Digital Register Implementation	**Analog Implementation**
Energy/Search	16nJ	400nJ
Time/Search	50ns	135ns
Area	6mm^2	2.57mm^2
Resolution	1 bit	40 bits
Input Range	8192 $bits$	4000 $bits$

Figure 25: Comparison of Analog and Digital Implementations

VIII. Conclusion and Future Work

An Associative Memory is implemented as both a digital and an analog circuit for high-dimensional computing applications. The error robustness of HD computing allows for the analog design – the loss in resolution and accuracy can be tolerated by the algorithm. However, the algorithm considered in this paper, language recognition, has a strict resolution requirement if the hardware errors are not to significantly affect the algorithmic errors.

For an analog AM with a capacity of 32 vectors of 10,000 dimensions, a single search takes 135ns and consumes 400nJ. The analog hardware has a worst-case resolution of 125 bits. Meanwhile, the digital AM performs the same search in 50ns using 16nJ. The digital AM has lower latency and energy consumption, which is surprising.

There are opportunities to significantly improve the analog design. A different distance comparison architecture can be used to significantly decrease the delay by 74%, to 35ns. Rather than compare each ML voltage to a global reference voltage, a reduction tree of comparators can be used to compare the ML voltages to each other. Additionally, the matchline circuit consumes the vast majority of the total energy. Therefore, track-and-hold amplifiers should be used to decrease the time the matchline remains active.

Finally, the time required to charge the matchline contributes a significant amount of the delay, 30ns. This delay may be decreased by replacing the PMOS load with a transimpedance amplifier. In this paper, only a PMOS load is explored, due to the ease with which the PMOS load is calibrated to overcome process variations.

References

1. Kanerva, P. 2009. Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors. Cognitive Computation 1:139–159.
2. Abbas Rahimi, Sohum Datta, Denis Kleyko, Edward Paxon Frady, Bruno Olshausen, Pentti Kanerva, Jan M. Rabaey, "High-Dimensional Computing as a Nanoscalable Paradigm", *Circuits and Systems I: Regular Papers IEEE Transactions on*, vol. 64, pp. 2508-2521, 2017, ISSN 1549-8328.
3. M. E. Sinangil *et al*., "A 28 nm 2 Mbit 6 T SRAM With Highly Configurable Low-Voltage Write-Ability Assist Implementation and Capacitor-Based Sense-Amplifier Input Offset Compensation," in *IEEE Journal of Solid-State Circuits*, vol. 51, no. 2, pp. 557-567, Feb. 2016.
4. A. Rahimi, A. Tchouprina, P. Kanerva, J. del R. Millán, J. M. Rabaey, "Hyperdimensional Computing for Blind and One-Shot Classification of EEG Error-Related Potentials," In *ACM/Springer Mobile Networks & Applications (MONET)*, Special Issue on Biologically Inspired Networking, 2017.
5. I. Arsovski *et al*., "12.4 1.4Gsearch/s 2Mb/mm^2 TCAM using two-phase-precharge ML sensing and power-grid preconditioning to reduce Ldi/dt power-supply noise by 50%," *2017 IEEE International Solid-State Circuits Conference (ISSCC)*, San Francisco, CA, 2017, pp. 212-213.
6. Rao, Thammavarapu RN. *Error coding for arithmetic processors*. Elsevier, 1974.
7. ZIMMERMANN, RETO. "Binary Adder Architectures for Cell-Based VLSI and Their Synthesis." SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH, 1997, citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.68.9657&rep=rep1&type=pdf.
8. *Van der Vaart, A. W. (1998). Asymptotic statistics. New York: Cambridge University Press.*
9. Razavi, Behzad. *Principles of data conversion system design*. Vol. 126. New York: IEEE press, 1995.
10. C. Christoffersen, G. Toombs and A. Manzak, "An ultra-low power CMOS PTAT current source," *2010 Argentine School of Micro-Nanoelectronics, Technology and Applications (EAMTA)*, Montevideo, 2010, pp. 35-40.
11. Khorramabadi, Haideh. 2010. *EE247: Analog-Digital Interface Integrated Circuits, Lecture 13*. https://inst.eecs.berkeley.edu/~ee247/fa10/lectures.html
12. B. Giridhar, N. Pinckney, D. Sylvester and D. Blaauw, "13.7 A reconfigurable sense amplifier with auto-zero calibration and pre-amplification in 28nm CMOS," *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, San Francisco, CA, 2014, pp. 242-243.