**Research Experience for Community College Students: Design and Optimization of Non-Volatile Latch using Resistive Memory Technology (Anti-fuse & Magnetic Tunnel Junction)**

**Alex Hercules1, Anthony Lal1, Michael Gee1,**

**Amelito G. Enriquez1, Tyler Sheaves2, Hamid Mahmoodi2**

**Canada College, Redwood City, CA**

**San Francisco State University, San Francisco, CA**

**Abstract**

Hardware security and intellectual property (IP) protection are some of the top priorities for integrated circuit (IC) chip designers. Security becomes most compromised when secured data travels from secondary memory (i.e. flash or hard disk) to faster random-access memory (RAM), and, finally, to a processing module cache. Once in the cache, data is relatively secure and least susceptible to sophisticated hacking techniques, however, unlike secondary memory, upon power loss data stored in the cache and RAM is erased, therefore the cache and RAM are considered volatile storage and the secondary storage is considered non-volatile (NV). The process of transferring data from slow performing non-volatile to faster volatile memories exposes unencrypted and encrypted data, leaves designs susceptible to advanced hacking techniques, and increases boot times and power draw. To address these issues on-chip, non-volatile (NV), memories may be utilized to store highly sensitive data which should never be exposed and data which is accessed by the processor at every boot-up. In this methodology, design obfuscating bits, encryption keys, or identification tags can be kept on-chip (i.e. in the processing module) and, thus, would never be exposed during secondary memory accesses. To implement an on-chip non-volatile solution, a unit-cell, capable of storing a single bit of information must be provided. In this paper we survey existing non-volatile technologies and provide the necessary transistor-level designs to implement them, we then compare these proposed solutions’ power, delay and area over-head metrics.

**1. Introduction**

**1.1** Computer Architecture Issues:

High speed (non-secondary) memory systems that are used by the high-performance components of modern computing systems rely on capacitively stored electric charge to temporarily store binary information. Over time, this charge can deteriorate and under sudden power failure may be lost potentially resulting in the permanent loss of information. Common examples of this include power outages, system crashes, and power delivery becoming unreliable. This conventional method of storing temporarily cached data is, therefore, considered volatile as it can be lost under many circumstances. Arrays of moderate speed volatile memory are typically manufactured in high density off of the processor (typically 10’s of Gigabytes), which is known as random access memory (RAM). Much smaller (typically several Megabytes) arrays of expensive high-speed volatile memory is manufactured directly on the processor (on-chip) allowing faster access for cached repeated program data. The fastest of all computer memory are the processor registers which store data that is immediately in use during program execution. These registers usually store only several hundred bytes of data. Therefore, an inversely proportional trend exists between memory size, and processor access delay. The primary vulnerability to volatile memory is still its susceptibility to loss during power failure.

Due to the risk of data loss in volatile memories larger, slower and more power-hungry secondary memory is needed to store data independent of system power. These long-term memory solutions are known as non-volatile (NV) memory and are usually implemented in high density (many hundreds or several thousands of Gigabytes) on separate components from the processor due to differing manufacturing technology requirements. Typical NV memories are achieved either through long-term trapping of charge (i.e. flash memory arrays) or through programmed ferro-magnetic regions on a disk (i.e. magnetic disk drives). These solutions offer large quantities of storage, which is relatively safe even in the event of sudden power loss, at a low manufacturing cost. However, these solutions operate at much slower speeds with respect to volatile solutions.

When a Non-Volatile memory (such as a stored program) needs to be accessed it is placed on a bus and transferred from the NV component to a large array of faster volatile memory (also usually manufactured in high density on its own component) which is then provided to a processor’s cache as either instructions or operands, which are eventually loaded into processor registers for evaluation. During this movement of data two primary issues arise: power is consumed at each stage as data must be propagated across relatively large distances (with respect to the distances data travels within a processor) and, perhaps most important to the security of the system, potentially sensitive data may be exposed as the data bus is easily accessed by potential hackers.

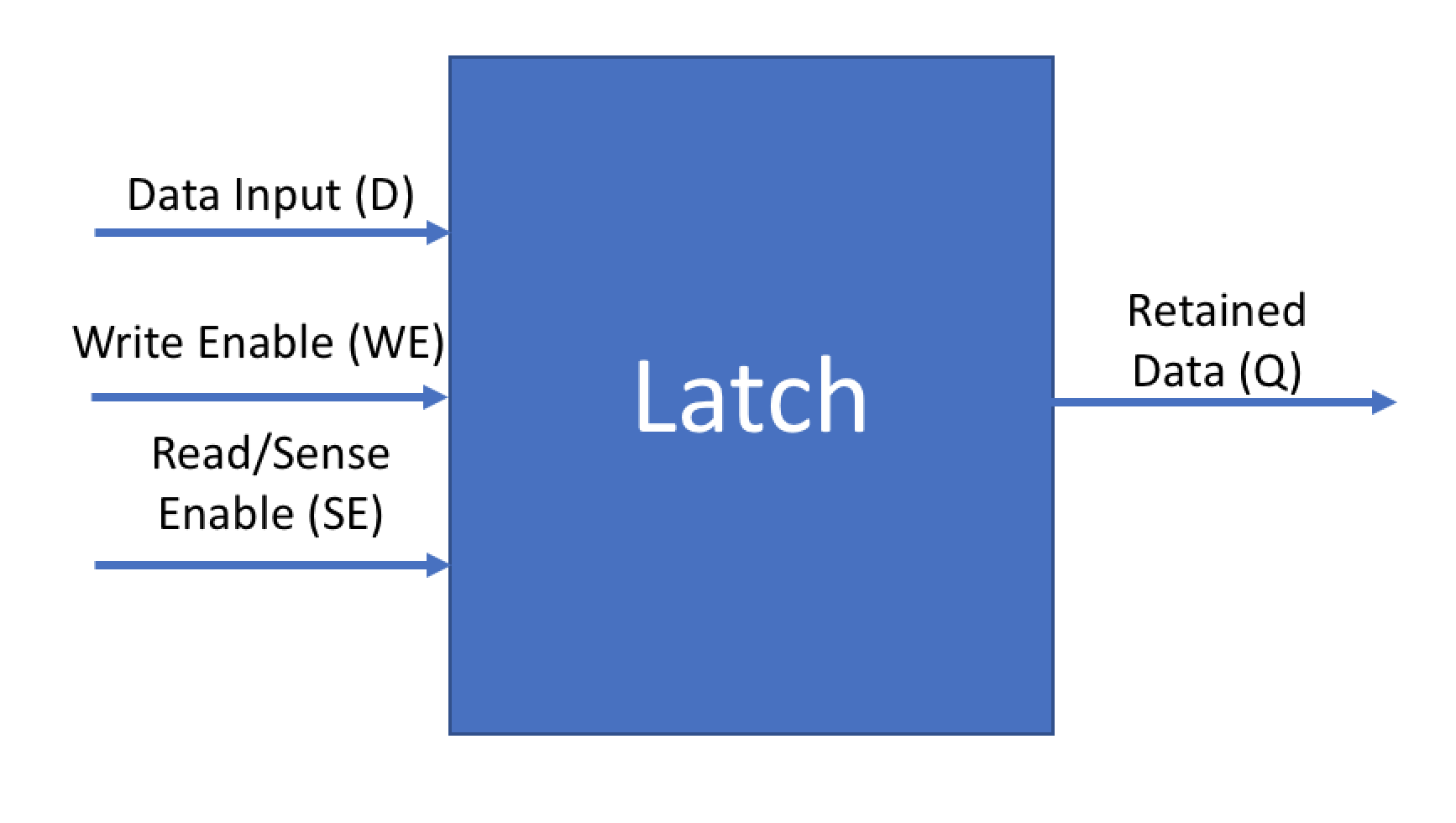
To prevent these issues new technologies are needed which can provide solutions to security vulnerabilities introduced from sensitive data transfers and from the unnecessary power loss of repetitively accessed data such as encryption keys, stored device identification tags, and potentially device configuration information.

**1.2** Proposed Resolution:

The most promising of these new technologies utilize on-chip non-volatile bit-cells, which can permanently retain data without the need for it to be stored on a non-processing component. Therefore, a designer could implement these cells to keep sensitive data, or data that is needed frequently on-chip, and save data propagation delay, access power, and improve upon system security. In the following introductory sub-sections, we will outline the basic latch cell and the devices being developed for on-chip NV applications.

**1.3** The Latch:

The latch is the most fundamental unit of volatile binary data storage. A typical latch must have the ability to be programmed to a binary state and then must retain this binary state which can then be queried by other components if system power is present. As such, a basic latch can have control inputs for reading or sensing input data (‘SE’), a data input (‘D’), and outputs which present its current binary state (referred to as ‘Q’). It may also provide the inverted version of this state (referred to as ‘Qb’). For our purposes we needed to preserve the same input and output signals, yet provide non-volatility, therefore we need an additional programming input to change the state of our non-volatile device. This input we will refer to as write enable (‘WE’).



***Figure 1. Block-Level NV Latch I/O***

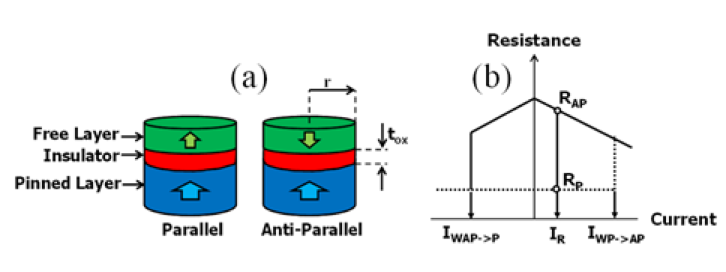
**1.4** Non-Volatile Devices:

To store a single bit of data for use in digital systems we must utilize a latch, but to make the latch store data independent of system power, another device is necessary. These devices must have electrical resistance values which can be altered and then evaluated to a binary electrical voltage state. In this section we will examine the most promising non-volatile device solutions under development: the magnetic tunnel junction and the Anti-fuse.

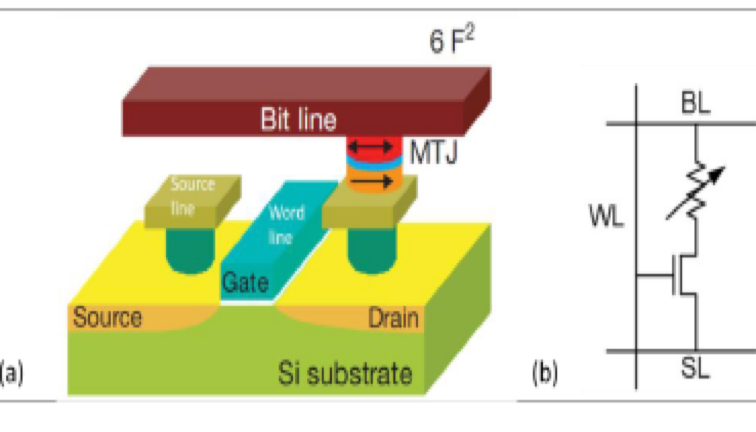
**1.4.1** The Magnetic Tunnel Junction:

The magnetic tunnel junction (MTJ) is a device consisting of several ferromagnetic layers separated by thin insulators. When an electric current is placed across this device electrons travelling through any magnetic layer become spin-polarized with an axis parallel to the magnetic field they are traversing. Electrons entering new ferromagnetic regions may exert a magnetic torque on the region’s magnetic field if it’s spin axis is aligned in an opposing direction. With enough torque a region can have its magnetic field direction shifted by 180 degrees. This shift can be sensed by the proper circuitry as a resistance change. The resistance change in these devices is binary as current can only flow in two directions along a wire, thus electrons are spin polarized along only two axes. Spin polarized electrons can readily bias the resistance of an MTJ if one ferromagnetic layer has strong magnetic field (pinned layer) and another has a weak magnetic field (free layer). The MTJ’s binary resistive state may be toggled by toggling the direction of current flow across the device (see Figure 1). This means the MTJ may be modelled as a binary resistor dependant on current direction and quantity. The resistive ratio (referred to as tunnel magnetoresistance or TMR), calculated as:

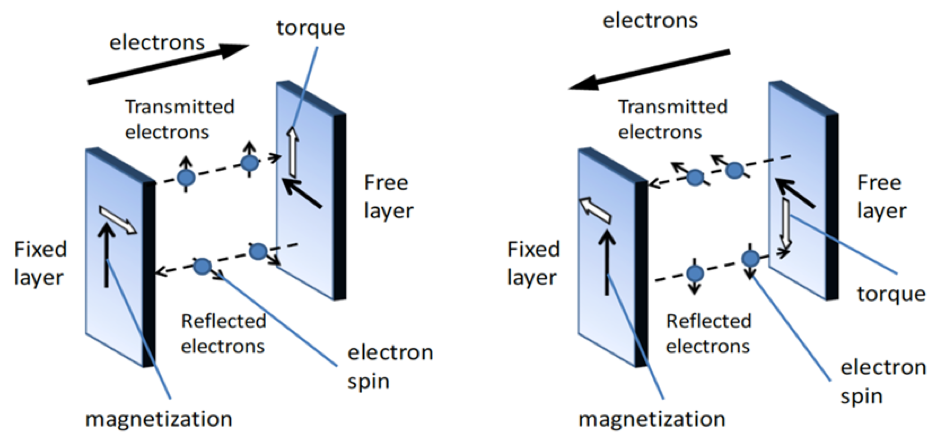
is not very large with reported values up to 2. These devices also require bidirectional current. These considerations make MTJ-based NV Latch cells more difficult to engineer and make the cell more costly in power and area.



***Figure 2. Perpendicular MTJ: (a) Parallel and Antiparallel states, (b) R-I characteristics***



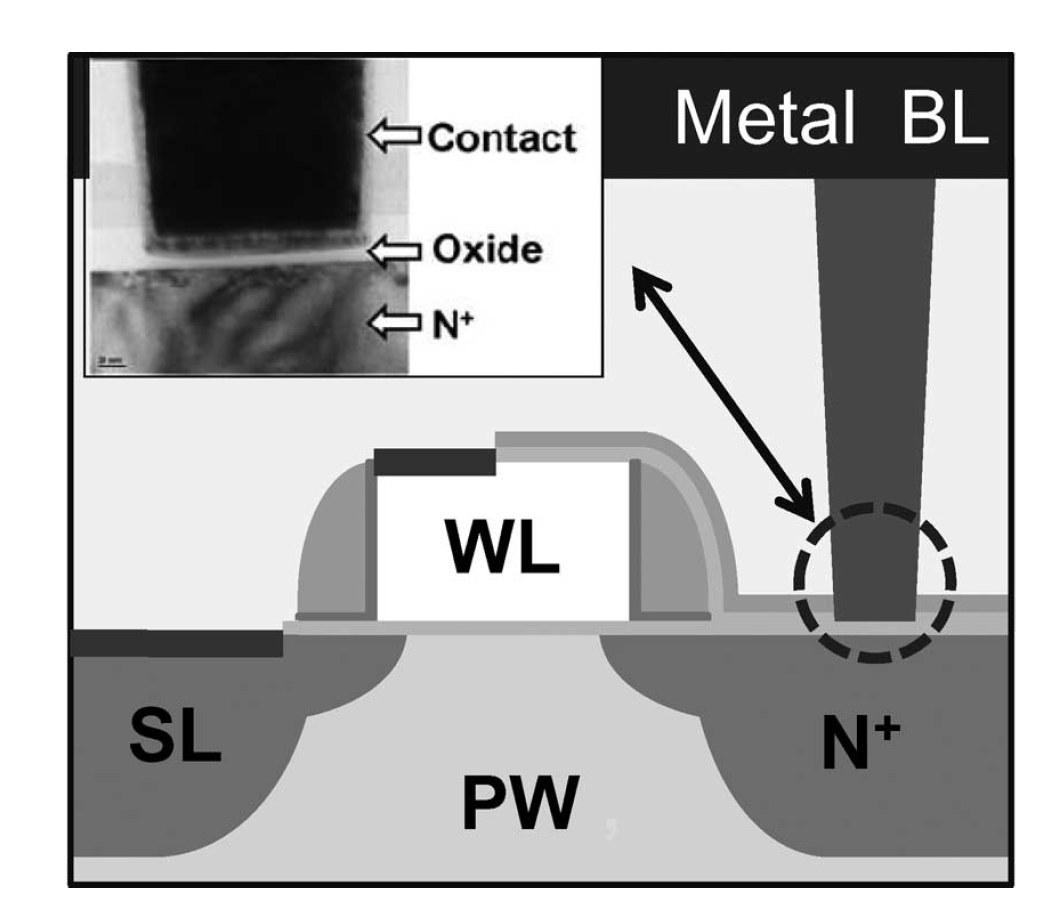
***Figure 3 . A basic MTJ Latch Topology* [Loh, 2009]**



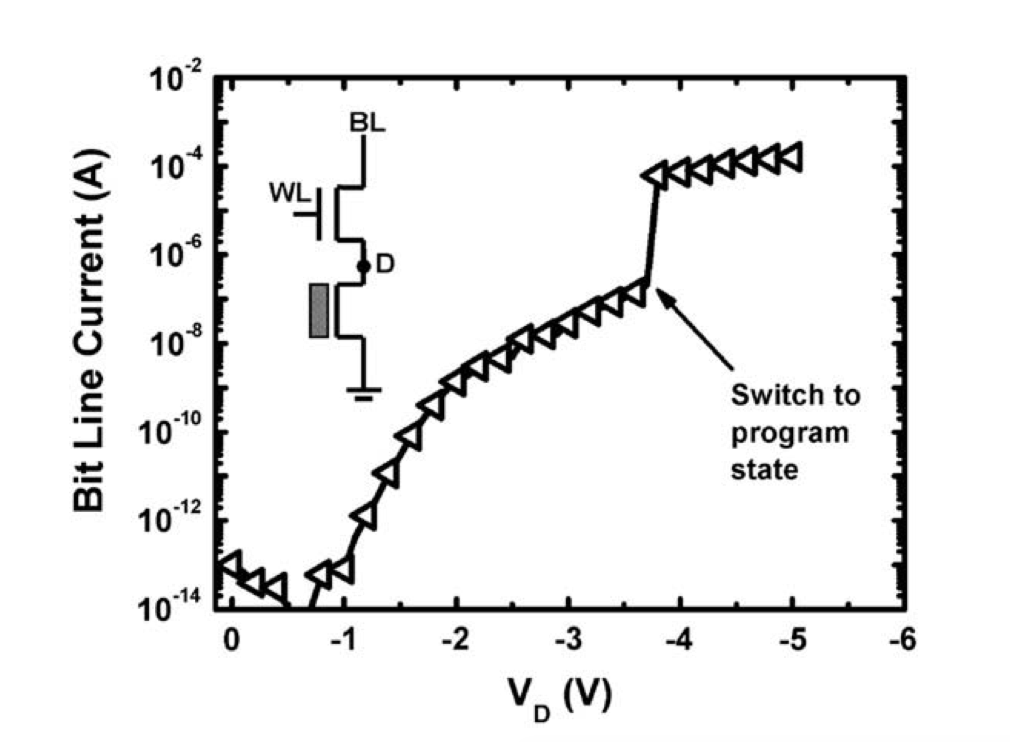
***Figure 4. MTJ Binary Magnetic Orientations Under Differing Current Directions [Loh, 2009*]**

**1.4.2** The One-Time Programmable Anti-Fuse:

The anti-fuse device is a cell whose insulating element breaks down under a high electric field, which is made of transistor-dielectric or metallic materials. Anti-fuses are manufactured as high resistance devices whose resistance can be permanently altered to a low resistance value by placing a sufficient electric field across it. Once programmed the resistance value cannot be reversed making anti-fuse devices less flexible than MTJ devices. However, unlike MTJ devices, anti-fuse devices present massive resistive ratios on the order of 10^5, with low current requirements when compared to MTJs (20uA vs several hundred uA). These devices may also be either fabricated at a metal layer above the transistors of our design at no area cost or may occupy a small area (for our development process of 32 nanometers reported to be 1.04um^2) if implemented using transistor dielectrics (see Figure 3). For metal-to-metal based Anti-fuses depicted in Figure the high electric field results in permanent conductive paths being formed between initially isolated regions of metal layers. Other types of programmable anti-fuses exist, but for the purposes of this paper we will assume a metal-to-metal type cell which can be fabricated above our transistor array, and, therefore, shows lower bit-cell area.



***Figure 5. Programmable-Contact Based Anti-Fuse* [Chen, 2009]**

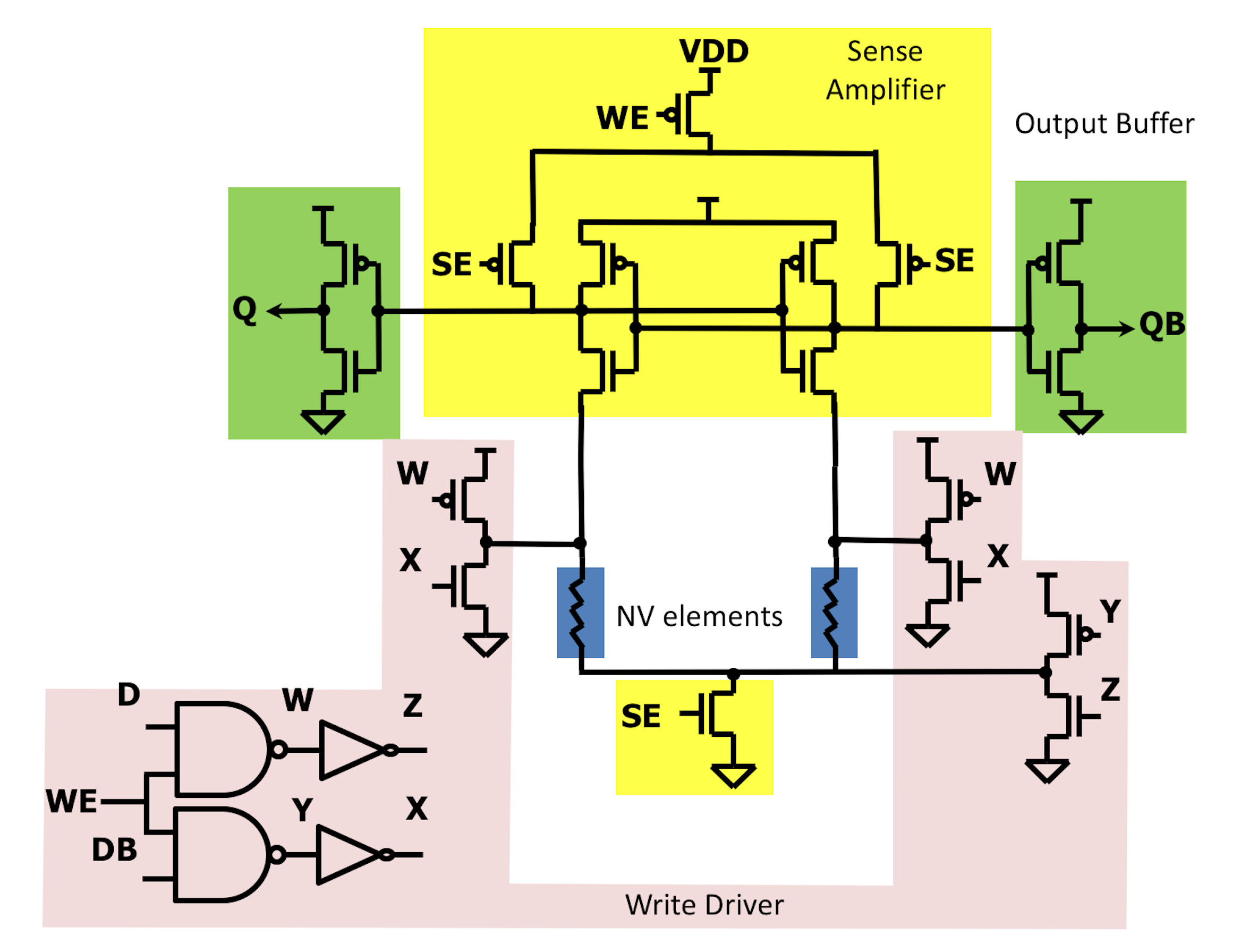
****

***Figure 6. Example of Programming of 45nm Anti-fuse* [Tsai, 2009]**

**2. Proposed Topologies**

In this section we will introduce the two proposed transistor-level topologies to convert the MTJ and anti-fuse NV devices into NV latches with read and write capabilities. The primary difference between these two topologies is stems from the resistive device characteristics. The MTJ is multi-time programmable, and therefore needs bi-directional current, and needs a differential pair due to its low resistive ratio. The Anti-fuse needs only a large electric field, cannot be reprogrammed, and presents a massive resistive ratio, so, while it is less flexible it’s bit-cell is much simpler and cost-efficient.

**2.1 MTJ Latch**

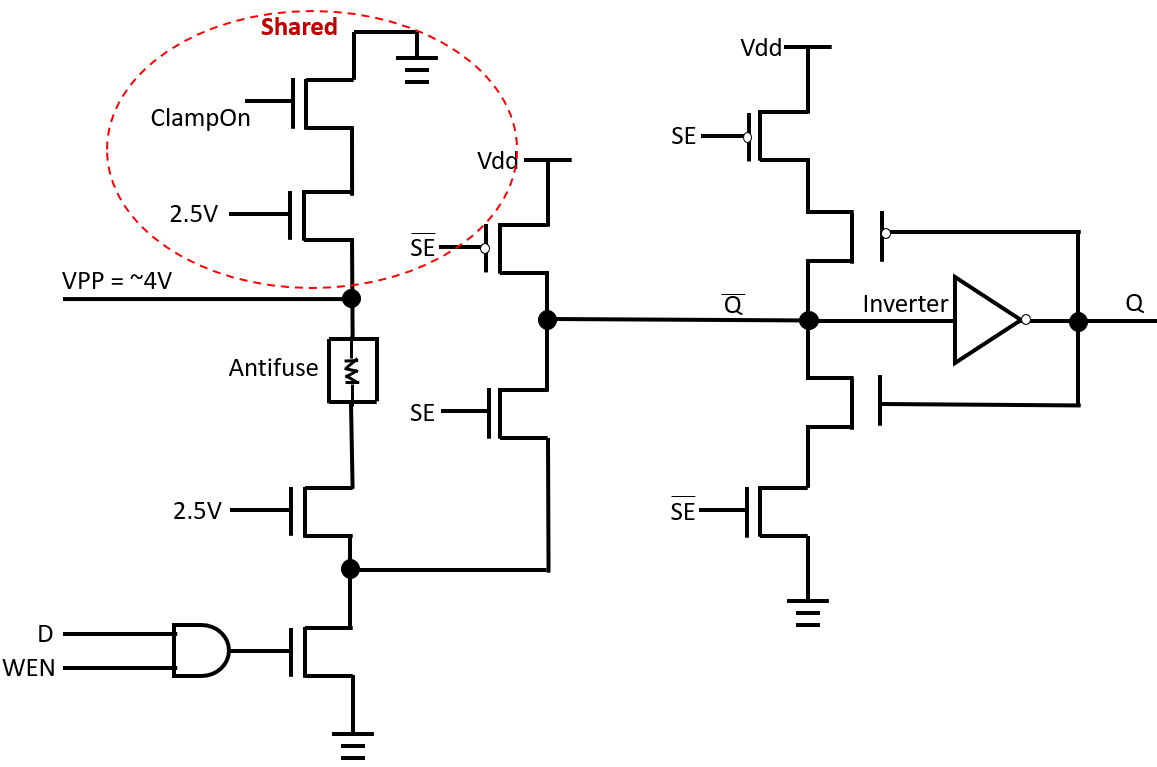


***Figure 7. MTJ Latch with Read/Write Capabilities and Control Logic***

The MTJ latch pictured in Figure has a differential topology, utilizing two separate MTJs whose resistance is amplified in the write mode. In order to write, write enable ‘WE’ must be set to a high value and the proper data must be presented to ‘D’. Sense enable must also be set low during the write operation. The write drivers must be sized adequately to provide sufficient current to flip the MTJ’s weak ferromagnetic layer. The control logic ensures the MTJ devices are written in opposing magnetic directions and thus have opposite resistive states, which can be sensed with a high accuracy by the sensing circuitry (see results section). This makes the write driver contribute large area overhead resulting in a large cell size (see results section).

During read a read signal must have a low to high switch in order to pre-charge the region above the MTJ and then update the ‘Q’ output with respect to the differentially programmed MTJ devices.

**2.2 Anti-fuse Latch**

****

***Figure 8. Anti-Fuse Based NV Latch***

**Figure 2** provides a transistor-level view of the anti-fuse-based latch, which was developed using a 32-nanometer generic process development kit (PDK). VPP is our programming voltage which will provide the necessary electric field, this signal will be set to a reported DC value of 4-5 Volts during write operation. All 2.5V gate-input transistors are always on and serve as protection for core transistors whose maximum rated voltage is 1.05V at the 32-nanometer technology node. The ‘Clamp-On’ signal serves as the circuit’s connection to ground during read operations. ‘Clamp-On’ is off during write to ensure the only DC path to ground is through the anti-fuse device depending on the data input (‘D’) and is on during read operation. ‘D’ is the circuit’s data input and also acts as the user’s input to the circuit allowing the user to either apply an electric field potential of ‘VPP’ or 0 across the anti-fuse. The signal ‘WE’ is the write enable signal which will allow the data input to either alter the potential across the anti-fuse or stop any potential from being applied. The ‘SE’ signal corresponds to the device’s sense enable (read) function. When this signal is high, and if write enable is low, the latch will convert the resistive state of the anti-fuse to a voltage value at output ‘Q’. It is important to note that everything above the anti-fuse element is shared across all AF cells on-chip, which means that the total unit-cell consists of just 16 transistors (only 3 of which are non-minimum size) and an anti-fuse element which may be manufactured on-top of the latch (meaning no additional area cost) or at a small transistor-level cost of 1.04 um^2. Therefore, the anti-fuse NV latch shows extremely low area overhead (see results section).

**3. NV Latch Optimization**

Both designs needed to be optimized to function with a low expected rate of failure. Both devices followed a similar optimization algorithm:

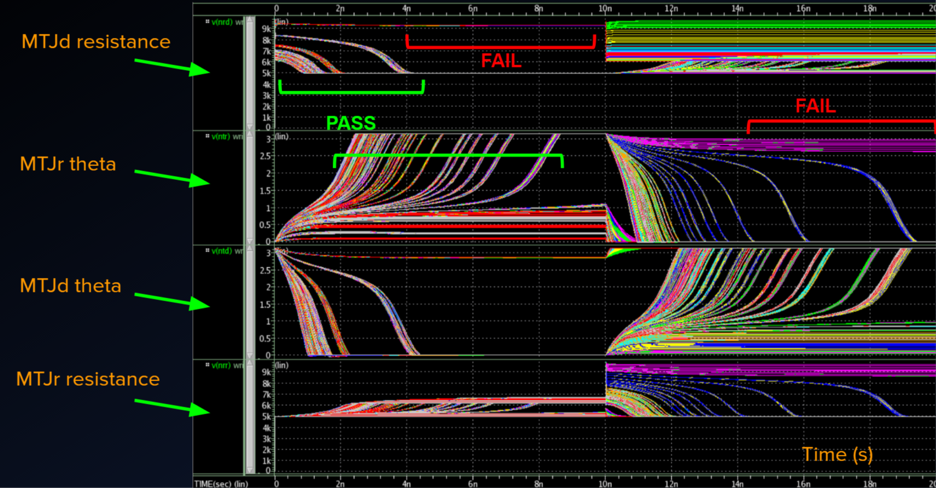
1. Identify transistors which might impact read/write reliability perform parametric sweeps with delay-based Monte Carlo simulations (proposed by Mahmoodi, 2009) at every parametric test to identify trends in reliability.
2. Select sizing of transistors such that pre-layout failure rates are no more than 0+-.001%.
3. Perform Layout.
4. Re-simulate Monte Carlo analysis and re-size transistors if necessary.
5. Characterize bit-cell for comparison to D-FF standard cell (see results section).

All simulations were carried out using a collection of Synopsys tools CDESIGNER, Hercules, IC Validator, and HSPICE provided under the Charles Babbage grant to San Francisco State University’s Nano-electronics and Computing Research Lab (NeCRL).

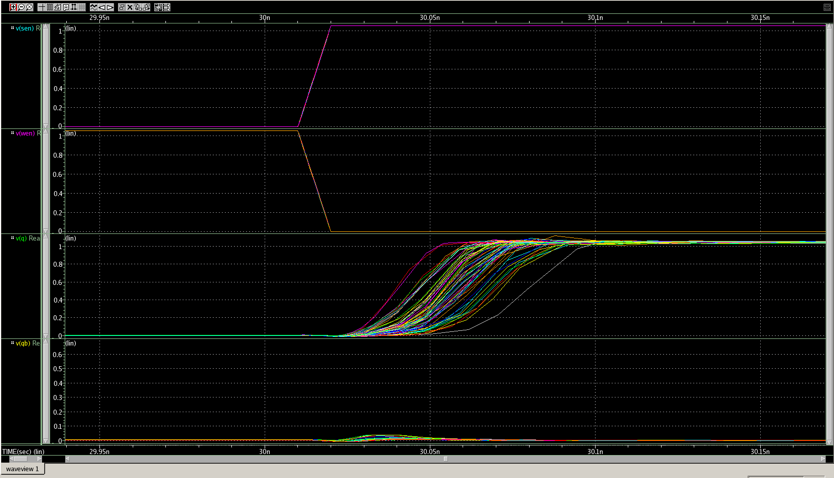
**3.1 MTJ-Based NV Latch Optimization**

Our MTJ-based NV latch was most likely to fail in the write direction if there was not sufficient current to change the magnetic field direction of the free-layer in both devices during the write cycle. Therefore, parametric analysis was performed on all transistors which made up the write driver. It was also possible that read path transistors random process variations would make mis-reads possible even with properly programmed cells, therefore at-risk transistors in the read path were selected as well for parametric analysis. At every parametric value set a Monte Carlo analysis of 10,000 iterations was performed with the following failure criteria:

1. **Inability to program MTJ devices during write mode**. This type of failure is illustrated in Figure. If an MTJ’s free layer does not have a magnetic phase shift, it’s angle will return to 0 radians. If it is successfully written, its magnetic angle will shift by pi radians (See Figure).
2. **Inability to sense written values during sensing mode**. Even with minimum size read path transistors, we were able to properly sense the MTJ resistive state. While sensing delay might increase under process variations, there was no instances of read failure detected given sufficient read time (see Figure).



***Figure 9. Monte Carlo Simulation of Parametric Set with Failure and Pass Instances***

******

***Figure 10. Monte Carlo Results with Respect to Sensing Delay***

We were provided a fully characterized HSPICE/Verilog-A model available as an open-source project from Arizona State University’s Nano-Scale Integration and Modelling Group (NIMO), so we were able to closely monitor MTJ device characteristics such as magnetic angle and resistance to characterize write failure.

**3.2 Anti-Fuse-Based NV Latch Optimization**

Unlike the MTJ, the Anti-fuse was most susceptible to read failure due to its single ended design. A voltage divider was formed during the sensing of a low resistive state. If the PMOS with input SEb was not sized properly sensing failure would manifest itself. Also, since a high voltage (VPP) was present on-chip, specific high voltage transistors had to be monitored to ensure no transistors gate-to-source, gate-to-drain, or source-to-drain voltage exceeded its rated voltage, which we assigned to be 3V. Finally, a sufficient electric field or current might not be present across the anti-fuse device during write which we assigned to be 4 Volts. Therefore, our Monte Carlo failure conditions were as follows:

1. **Sensing failure during Anti-fuse read cycles resulting in mis-read ‘Q’**. The Anti-fuse may have too large of a low-resistance to evaluate to a proper high output. To mitigate this, we create a larger resistance on the PMOS-type transistor with an input of ‘SEb’
2. **Voltage levels across at-risk transistors (see Figure) exceeding 3V.** The transistors used in the write path are rated for over double the core voltage, however, if voltage values across these transistor’s inputs exceeds their ratings during write operations, they may have charge migration or dielectric breakdown occur, which may impair their performance.
3. **Voltage/Current across the Anti-fuse insufficient.** The Anti-fuse itself has specific write current and device voltage needs to achieve a change in resistive state. If these are not met the device cannot be written to properly.

Due to a lack of available robust models, the Anti-fuse was modeled at its measured blown and unblown resistance and capacitive values reported by. With a blown resistance reported to be 15 kOhms, an unblown resistance of 15 MOhms (a ratio of 10^5) and a parasitic unblown capacitance of 5 fF. The Sensing and Write-Mode operational waveforms for our bit-cell are shown in figure. Note that SE and WE cannot be on concurrently and in retention mode must both be off. A failed Monte Carlo iteration can be detected by a gate-drain, gate-source or source-drain voltage exceeding 10% of our 2.5V rated shielding transistors, by a false ‘Q’ or ‘Qb’ reported after sensing (see Figure for example), or by insufficient write current (less than 20uA).



***Figure 11. Anti-Fuse Monte Carlo Results Depicting a Sensing Error***

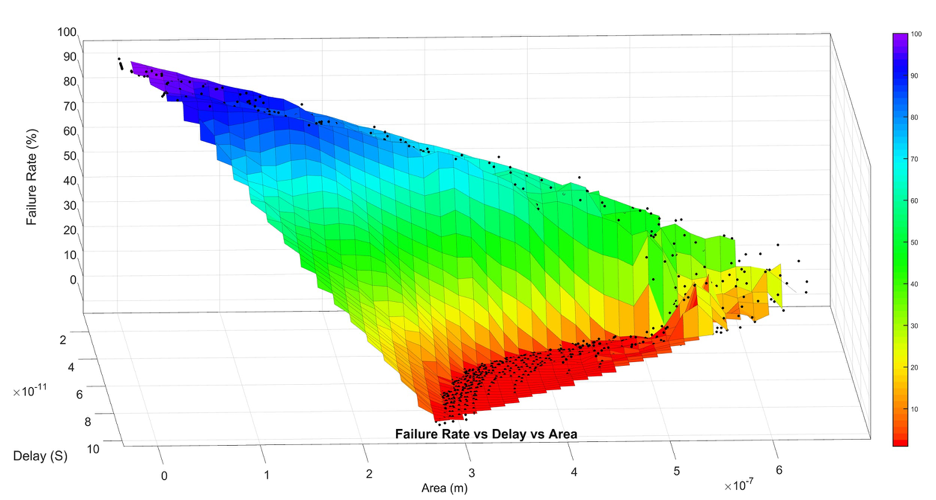
**4. Results and Comparison**

During Write the Anti-fuse had far lower current delivery needs (20uA), and, due to its massive unblown resistance, voltage across the Anti-fuse was consistent across Monte Carlo analysis. The MTJ based latch, however, was extremely susceptible to write errors as depicted in the signal variances in Figure resulting in the need for upsizing of its 6-transistor write driver transistor widths. Once optimized their values ranged from 1um to 1.5um. The 3-transistor 2.5V type-transistor write driver for the Anti-fuse was sufficient to achieve its target reliability with a much smaller maximum transistor width of .36um. This accounts for the large difference in write area between the two topologies.

During Read the MTJ was least susceptible to errors, so its transistors were only optimized for sensing delay. This higher tolerance was due to the MTJ’s differential design. The Anti-fuse required upsizing to achieve higher read reliability, however, its upsizing was only needed on a single transistor, resulting in a low total area cost. The Anti-fuse was able to achieve its reliability goal with a single transistor length upsizing of .7um. During layout this transistor was sized to .8um due to space availability. As all other transistors were able to be kept at minimum size, even the more sensitive Anti-fuse latch read path came in at a lower area when compared to the MTJ’s read path area.

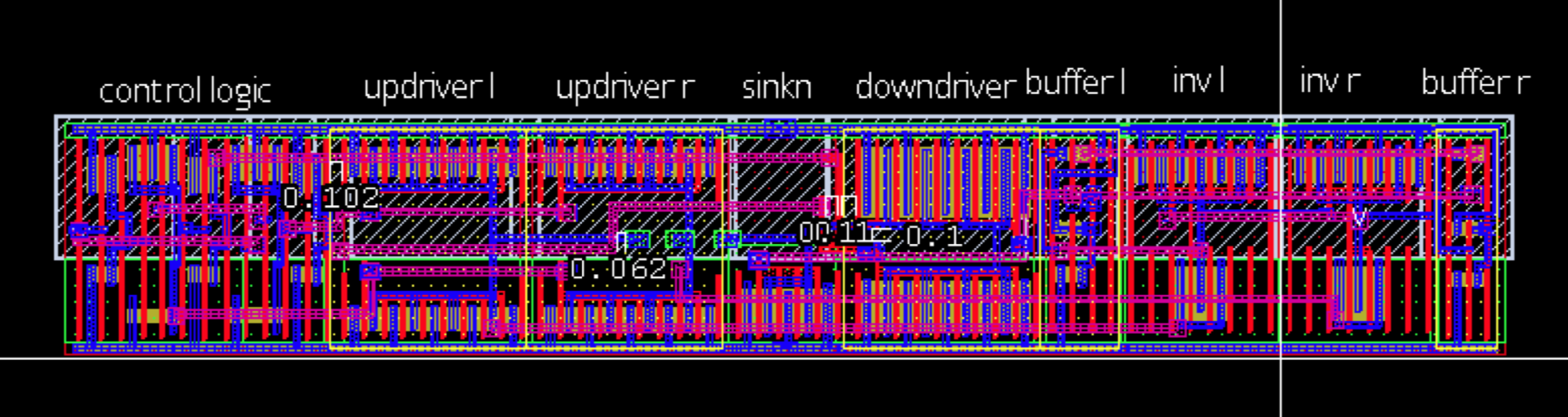
**4.1 Final Design Metrics and Layouts**

**4.1.1 MTJ Results**

****

***Figure 12. MTJ Write Failure Rate with Respect to Write Delay vs. Area***

The MTJ latch would only fail if it’s write drivers were not sufficiently sized. Figure shows the correlation between larger area and device delay. The plot visualizes how a larger write driver area would allow for lower write delays and higher reliability. Due to a robust model’s availability and the ability to use exclusively core-voltage transistors, we were able to simplify the failure criteria for the MTJ bit-cell to simply be a failed write operation or a failed read operation in any Monte Carlo iteration. The optimized bit-cell’s characteristics are summarized in table. Finally, the layout for our optimized bit-cell is given in Figure 13 .



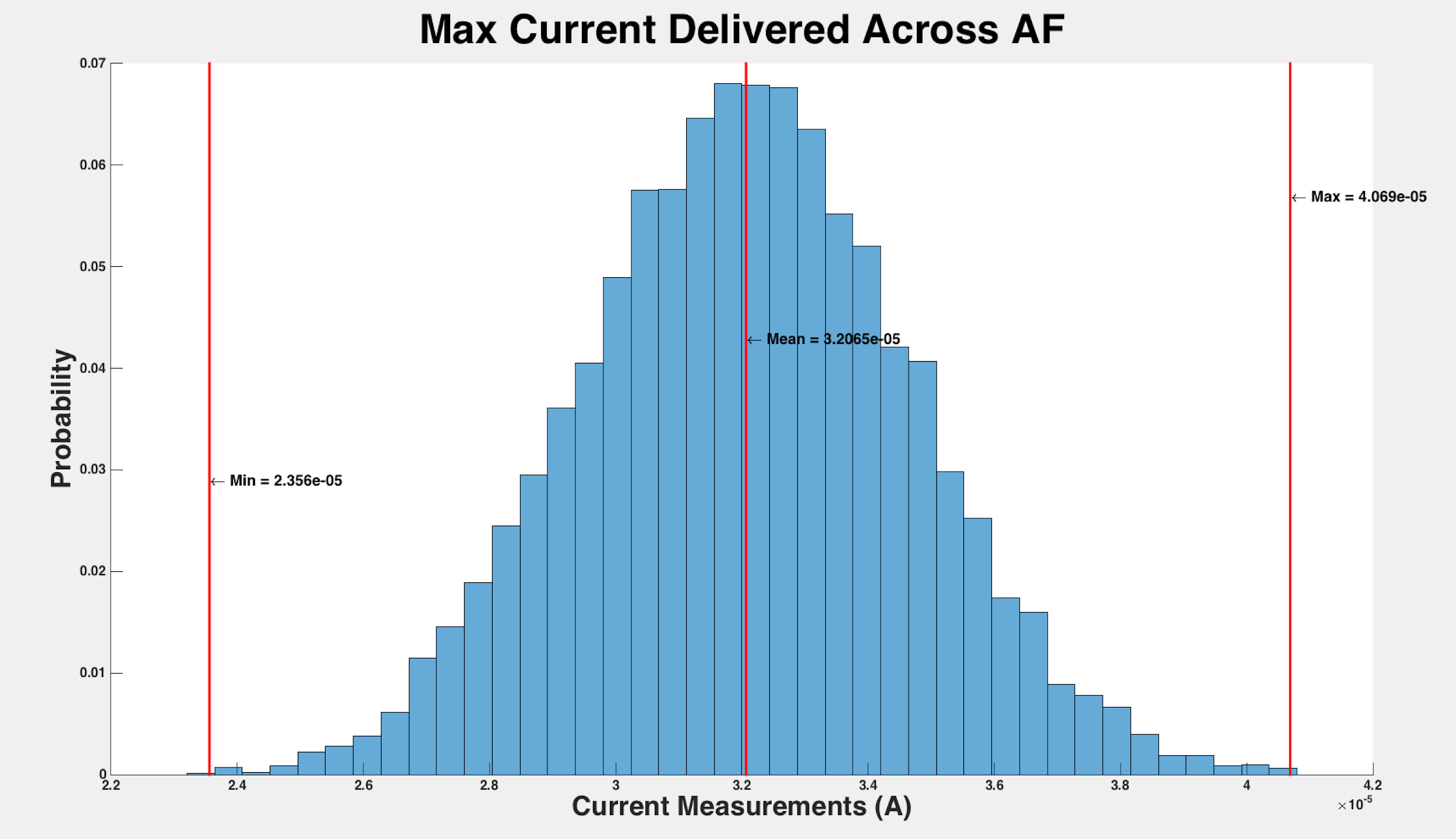
***Figure 13. MTJ-Based NV-Latch Layout***

**4.1.2 Anti-Fuse Results**

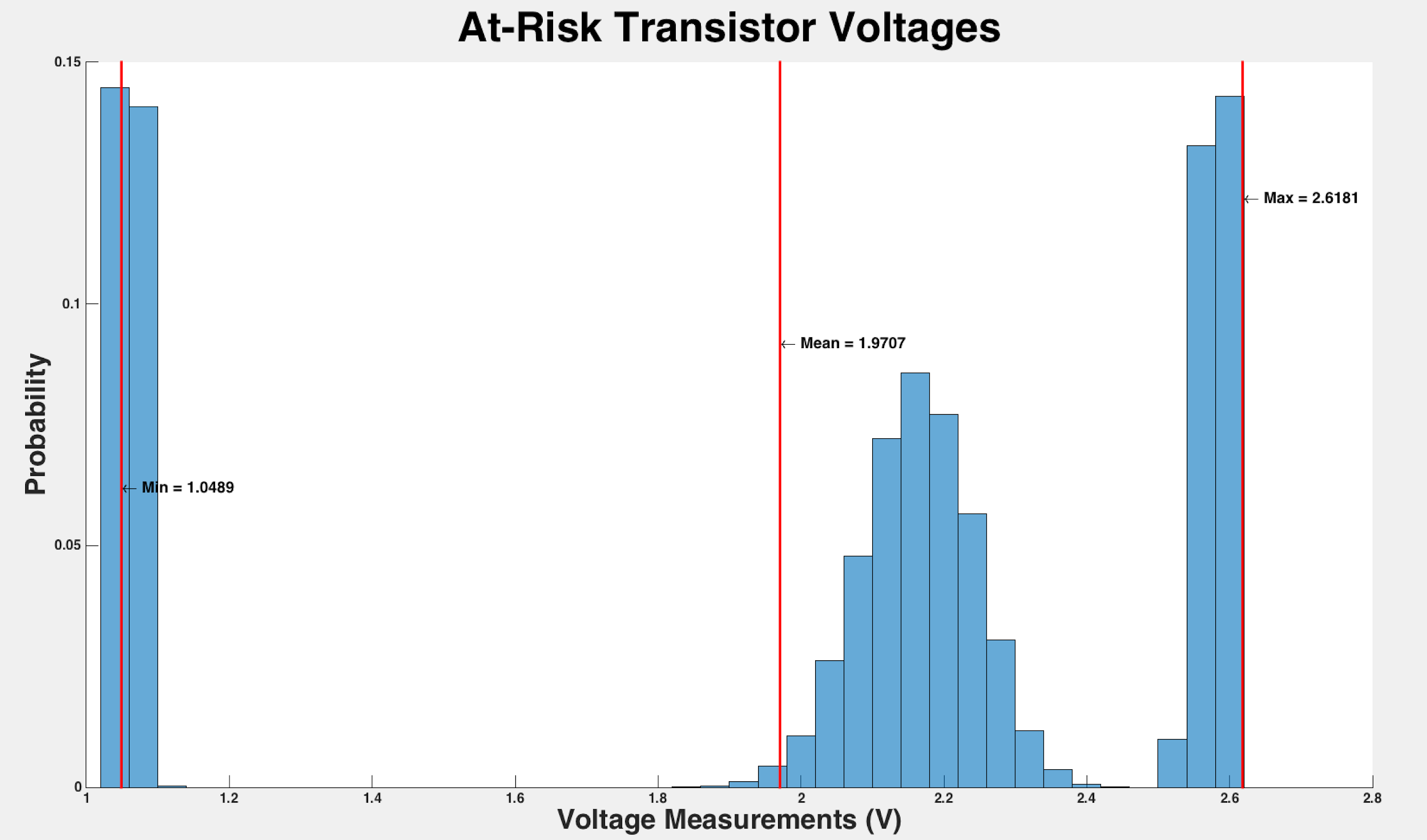
The Anti-fuse showed impressive area, current, and power performance at the cost of sensing delay and reprogrammability. The device was able to operate at safe voltage levels for it’s all of its at-risk transistors and evaluated to proper ‘Q’ and ‘Qb’ values under Monte-Carlo analysis (See Figures - ). Due to its simple write topology the design showed far lower area than the MTJ-based Latch, thus for one-time programmable applications the Anti-fuse is a far more efficient device. Due to a lack of a robust model many failure criteria were necessary which are listed as additional metrics in table. Histograms of optimized post-layout measurements shown in figures through show that all criteria were met at the target reliability. Finally, figure shows the layout of the Anti-fuse bit-cell, which showed large area and power savings when compared to the MTJ bit-cell.



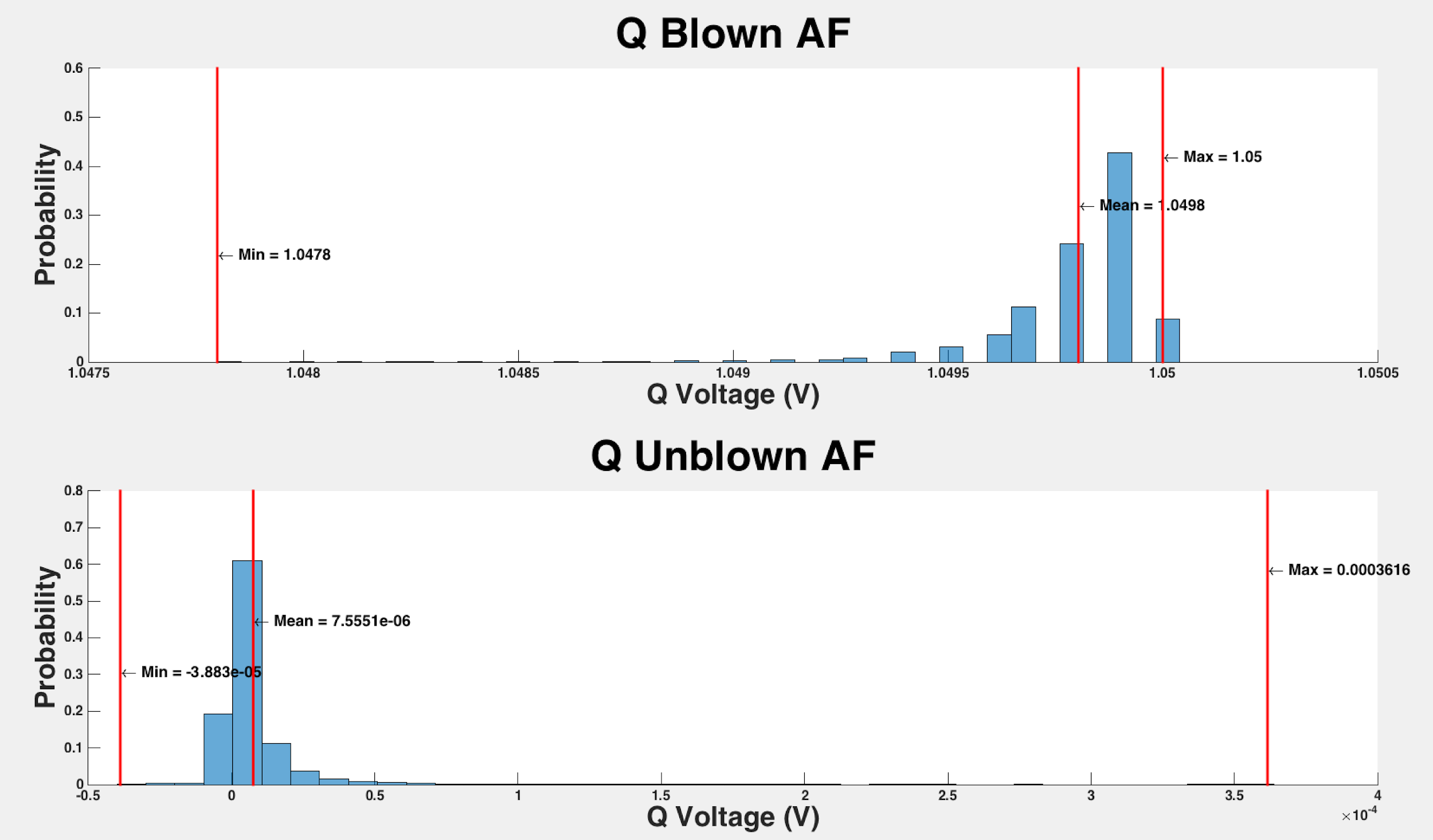
***Table 1. Optimized Anti-fuse Failure Criteria, Final Design Metrics, And DFF Reference Comparison***



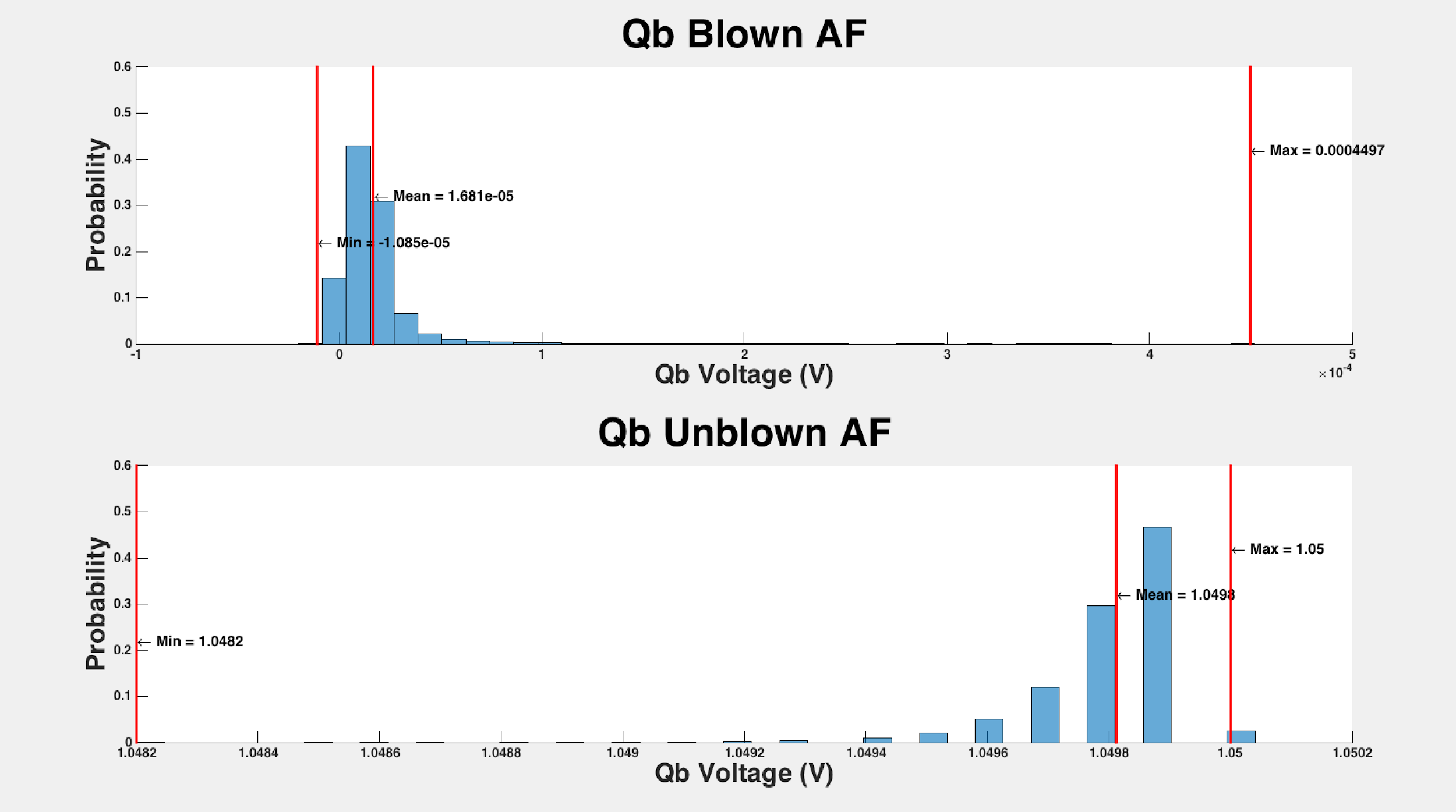
***Figure 14. Max current delivered across Anti-fuse***



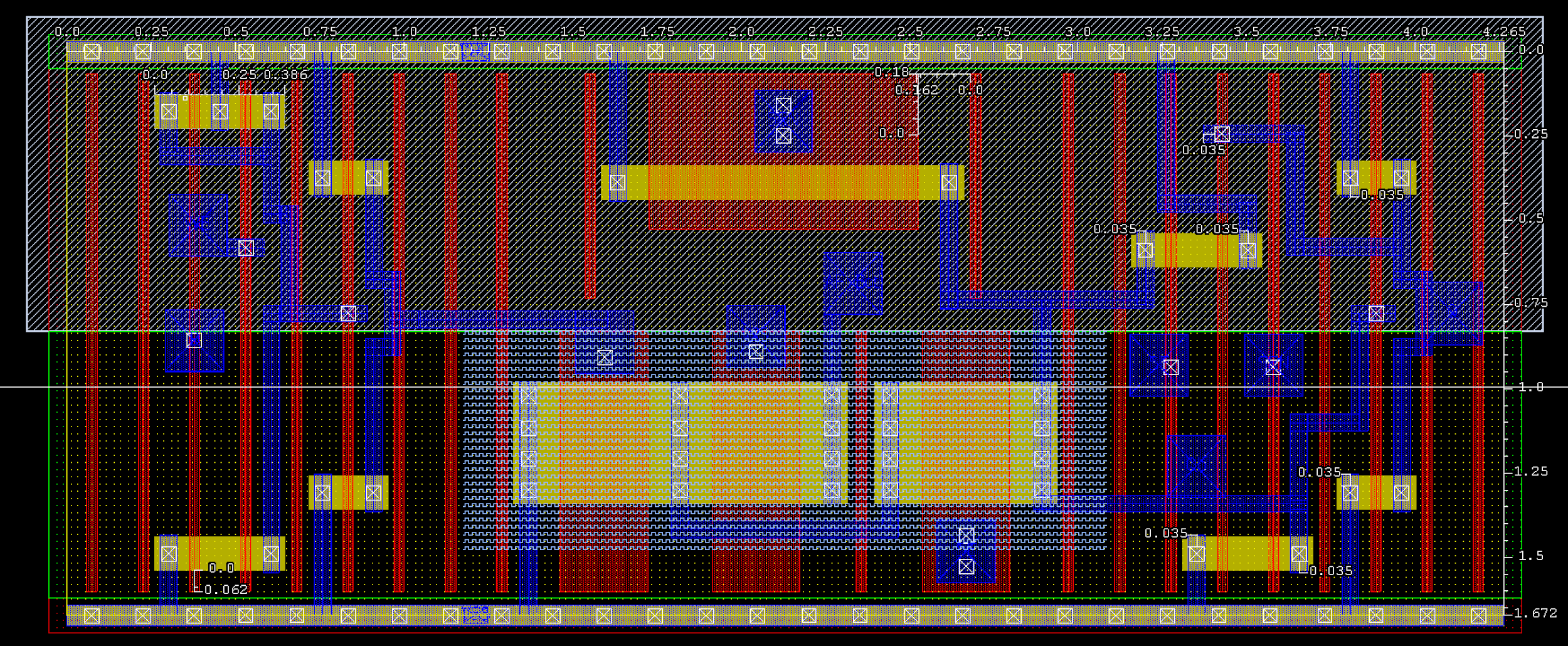
***Figure 15. Transistors at risk of exceeding 3 V for Anti-fuse***



***Figure 16. Anti-fuse Blown and Unblown voltages for Q***



***Figure 17. Anti-fuse Blown and Unblown voltages for Qb***

******

***Figure 18. Anti-fuse Based NV Latch Layout***

**Conclusion**:

In this paper we surveyed existing non-volatile technologies and provided the necessary transistor-level designs to implement them as a solution to on-chip non-volatile memory. We then compared proposed solutions’ power, delay and area overhead metrics, which showed a tradeoff between re-programmability and cost. The Anti-fuse bit-cell is favorable if a smaller area overhead is desired and one-time programmability is sufficient. The anti-fuse also provides smaller power consumption and relatively smaller switching (programming) delay. The MTJ-based bit-cell provides re-programmability at a much higher cost in area and power (both static and switching) but offers lower sensing (read) delay and does not require many different global voltages like the Anti-fuse. The MTJ would also require new on-chip magnetic layer availability to function, while the Anti-fuse is a cell that can be implemented in a similar fashion to typical e-fuse cells which are available as standard cells in most technology’s process development kit standard cell library. D-Flip-Flops (DFFs) were the base comparison of our bit-cells as they are the current volatile solution to on-chip storage. When compared to DFFs the Anti-fuse latch was able to have minimal area overhead (117%). The MTJ, however had far larger area overhead due to its large write drivers. In the future other technologies may need to be examined such as ReRAM, which are reported to offer similar area cost as Anti-fuse and re-programmability without the current draw-backs of MTJ solutions.

**Bibliography**

[1] H. Mahmoodi, A. Attaran, T. Sheaves, “Design of a Non-Volatile Latch using Resistive Memory Technology”   
[2] H. Mahmoodi, S. Srinivasan Lakshmipuram, M. Aora, Y. Asgarieh, H. Homayoun, B. Lin and D. M. Tullsen “Resisitive Computation: A Critique.” IEEE COMPUTER ARCHITECTURE LETTERS, VOL. 13 NO.2, JULY-DECEMBER 2014  
[3] W. Zhao, E. Belhaire and C. Chappert “Spin-MTJ based Non-Volatile Flip-Flop.”  
Proceedings of the 7th IEEE International Conference on Nanotechnology August 2-5   
2007, Hong Kong  
[4]Wicht, Bernhard, Thomas Nirschl, and Doris Schmitt-Landsiedel. "Yield and Speed Optimization of a Latch-Type Voltage Sense Amplifier." IEEE JOURNAL OF SOLID-STATE CIRCUITS , VOL.39 NO.07, JULY 2004

[5]Welser, Jeffrey, S.A. Wolf, and Phaedon Avouris. " 282 CHAPTER 8 APPLICATIONS: NANOELECTRONICS AND NANOMAGNETICS ." Nanotechnology Research Directions for Societal Needs in 2020. London: Springer Dordrecht, 2011.

[6] E. R. Hsieh et al., "The demonstration of low-cost and logic process fully-compatible OTP memory on advanced HKMG CMOS with a newly found dielectric fuse breakdown," 2015 IEEE International Electron Devices Meeting (IEDM), Washington, DC, 2015, pp. 3.4.1-3.4.4.

[7] M. Deloge, B. Allard, P. Candelier, J. Damiens, E. Le-Roux and M. Rafik, "Application of a TDDB Model to the Optimization of the Programming Voltage and Dimensions of Antifuse Bitcells," in IEEE Electron Device Letters, vol. 32, no. 8, pp. 1041-1043, Aug. 2011.