University of Virginia
Charles L. Brown Department of Electrical and Computer Engineering

ECE 4332: Introduction to VLSI Design
Design Review 2

Spring 2024 4/4/2024

August Bresnaider, Robert Smith, Thomas Johnson

Prof. Calhoun and the Executives at PICo,

I hope this memo finds you well. This document intends to serve as a guide as to what has been completed since our last proposal, as well as a final roadmap to the completion and optimization of our low power SRAM.

I. Challenges Met

First, we were able to generate logical sizings of our pull-down, pull-up, and access transistors in our bitcell. We created a python program that took data from Virtuoso Visual Analyzer and returned values for the static noise margin and read noise margins, using these as a metric to pick the optimal transistor sizes. Below is a picture of the program working, plotting 625 RNM plots. Though this information is not very useful, the largest possible square is output for each curve into a CSV file for further analysis.

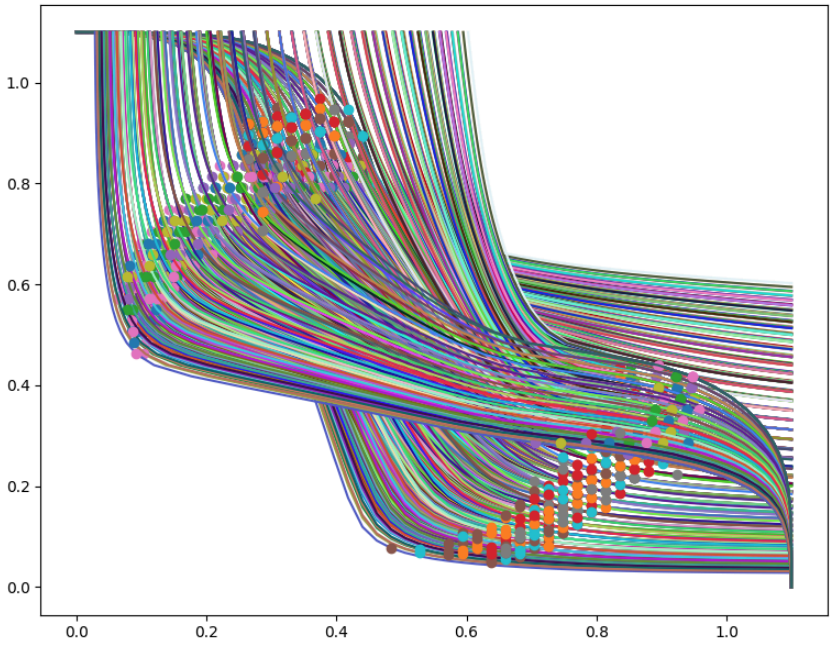


Figure 1: Read Noise Margins with Maximal Corners Highlighted

For the reference of future students and yourself, the python code can be accessed on the Illogical Effort wiki page. From the sizing of our transistors, we were able to create the following bitcell:

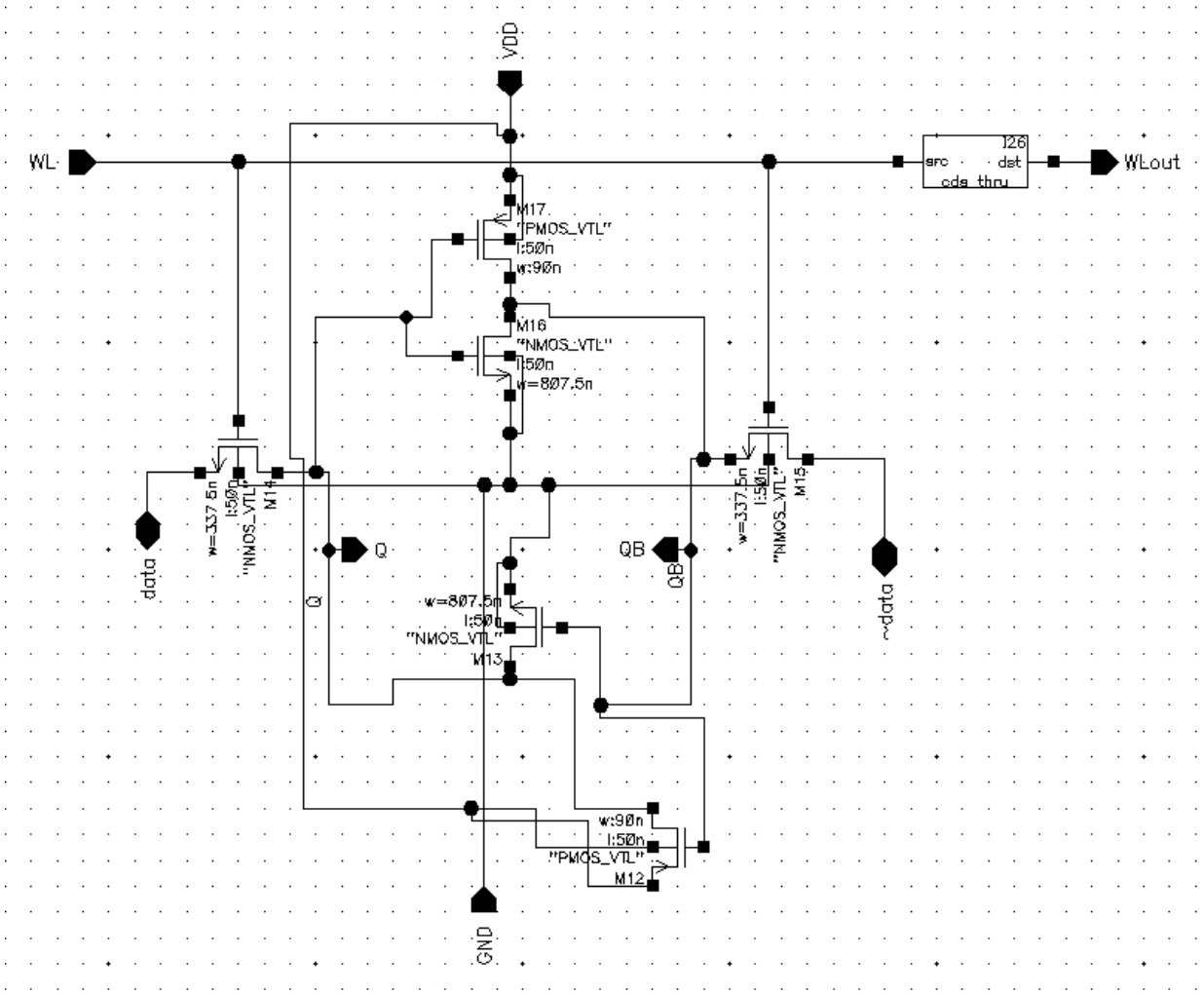


Figure 2: Standard Bitcell

Using this bitcell as a reference, we were able to generate static noise margins and read noise margins for the 4 process corners, as well as the typical-typical case. These plots can be found in the below appendix.

After deciding on our bitcell transistor widths, our next biggest decision was to determine the optimal block size. We ran transient simulations to find the effects of more and more bitcells on the bitlines and wordlines, to see how it affected the output waveform, as well as power draw.

We have the included csv files on our wiki page, but for brevity's sake here, we found that there should be at most 64 bitcells per column. Since the bitlines are in full swing, and Elmore delay is quadratic in nature, any extra doublings of the bitcells per column will severely warp the output waveform, as well as double the active power. As for the rows, because the wordlines are not in full swing, Elmore delay plays a much smaller role, and thus, we found we

are able to include 512 bitcells (or 32 words) per row. A figure of our block schematic can be found below:

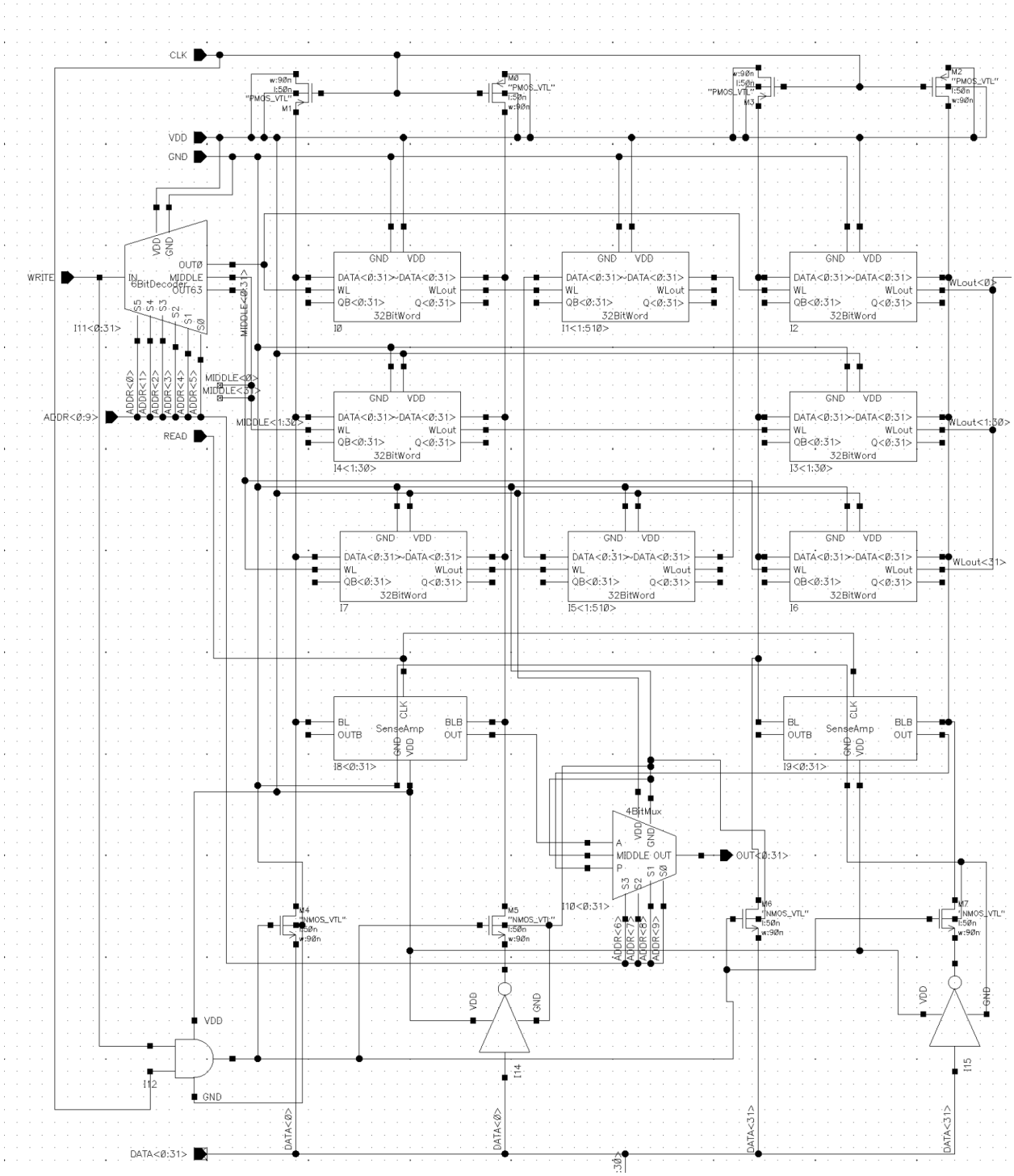


Figure 3: 512x64 Block Schematic

As can be seen in the above figure, we also finished work on a strong arm sense-amp. Below is a schematic of that:

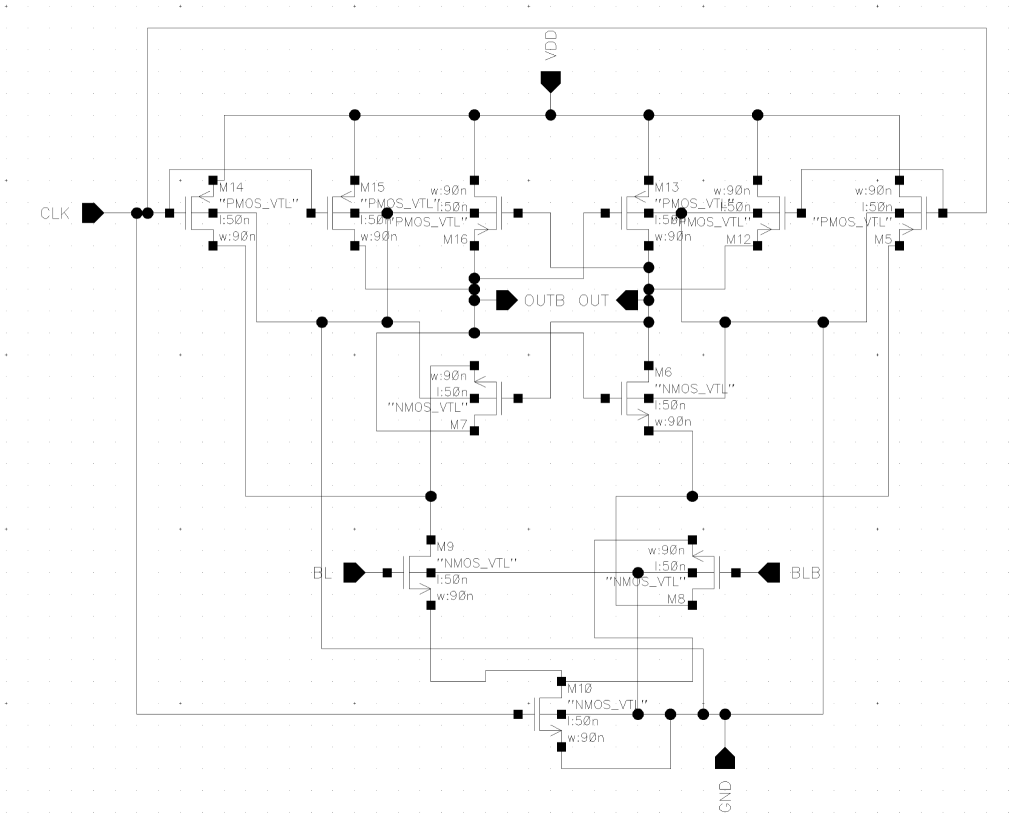


Figure 4: Strong Arm Sense Amp Schematic

We were also able to test this using the following testbench:

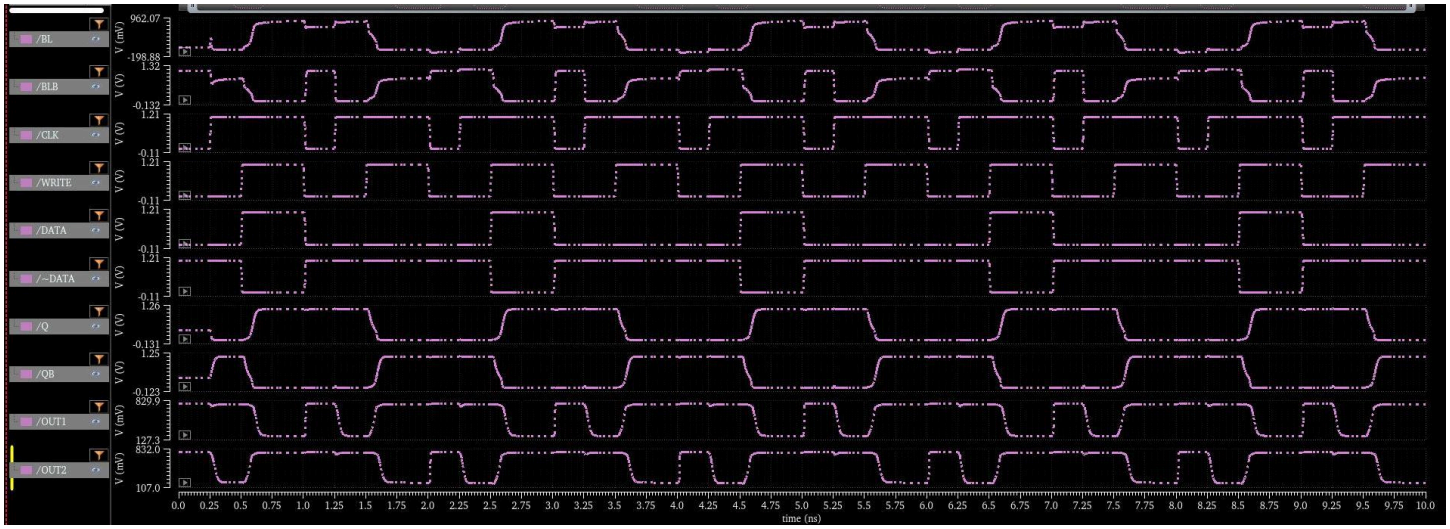


Figure 6: Bitcell / Sense Amp Testbench Waveform

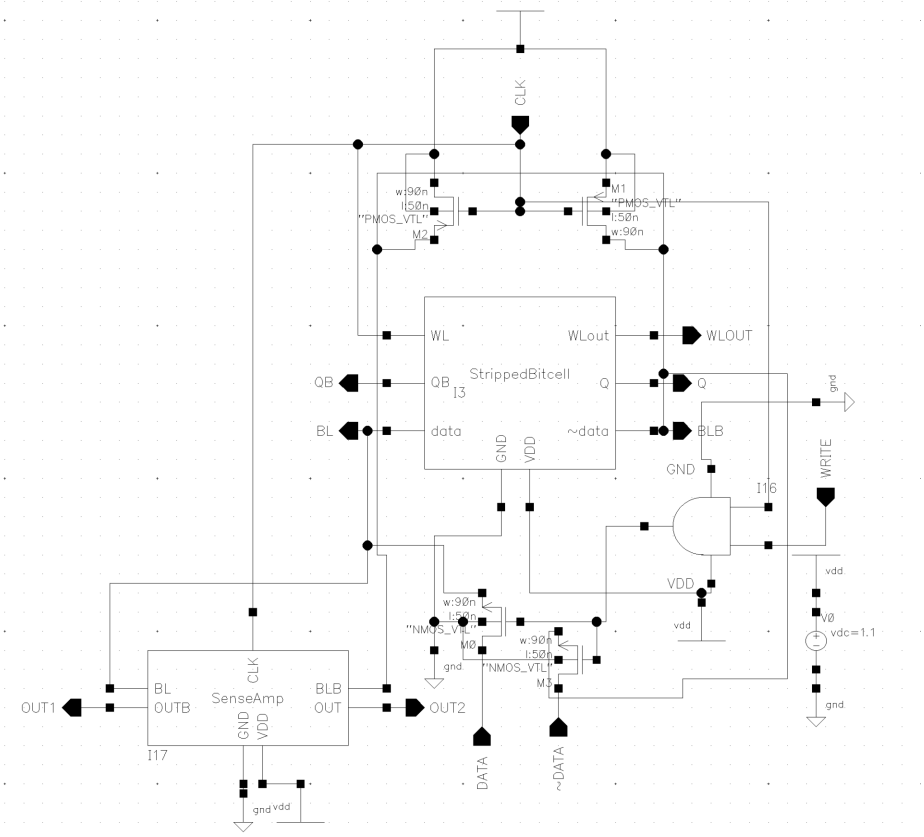


Figure 5: Bitcell/ Sense Amp Testbench Schematic

Finally, relating to the block, we were able to use the same testbench inputs and send them to address (0, 0) and address (511, 63), testing the output at the process corner SF:

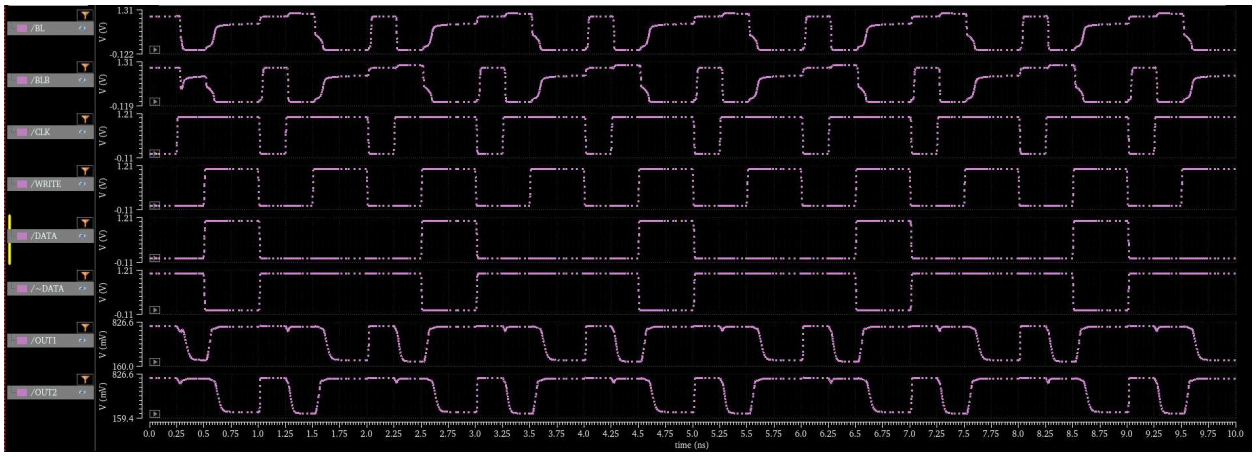


Figure 6: Block Address (0, 0) SF Process Corner Output Waveform

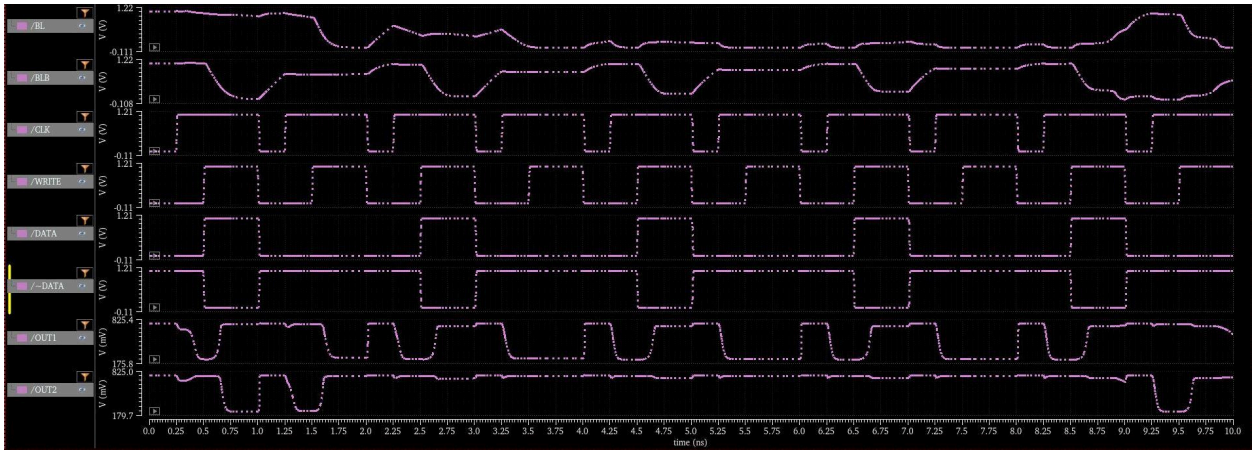


Figure 7: Block Address (511, 63) SF Process Corner Output Waveform

II. Special Features

From our design proposal, we have been working on implementing a body bias on the PMOS' of the circuit, as well as potential implementation of a 7T bitcell. We were able to generate schematics of both features, though we were unable to draw reasonable conclusions from our simulations, and will need to develop new testbenches to check the efficacy of these features.

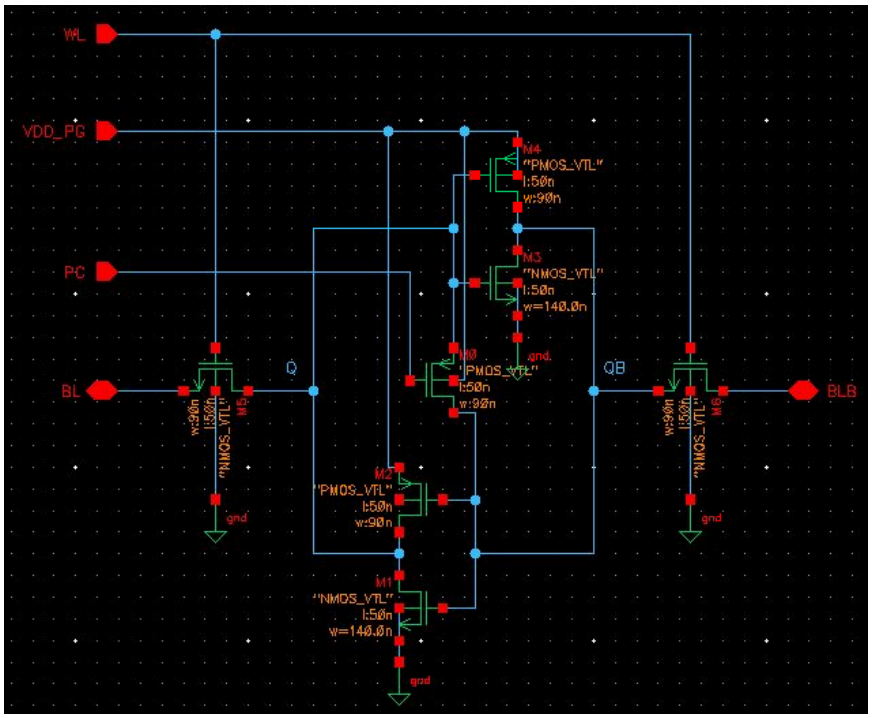


Figure 7: Proposed 7T schematic

III. Further Challenges

Percentage Completion	Bitcell	Block	Sense Amplifier	Mux / Decoders	Multi-Block	Metric Optimization
Design	**85%**	**100%**	**100%**	**100%**	**50%**	**33%**
Simulation	**75%**	**100%**	**100%**	**100%**	**25%**	**33%**

Table 1: Completion Percentage of Selected Project Categories

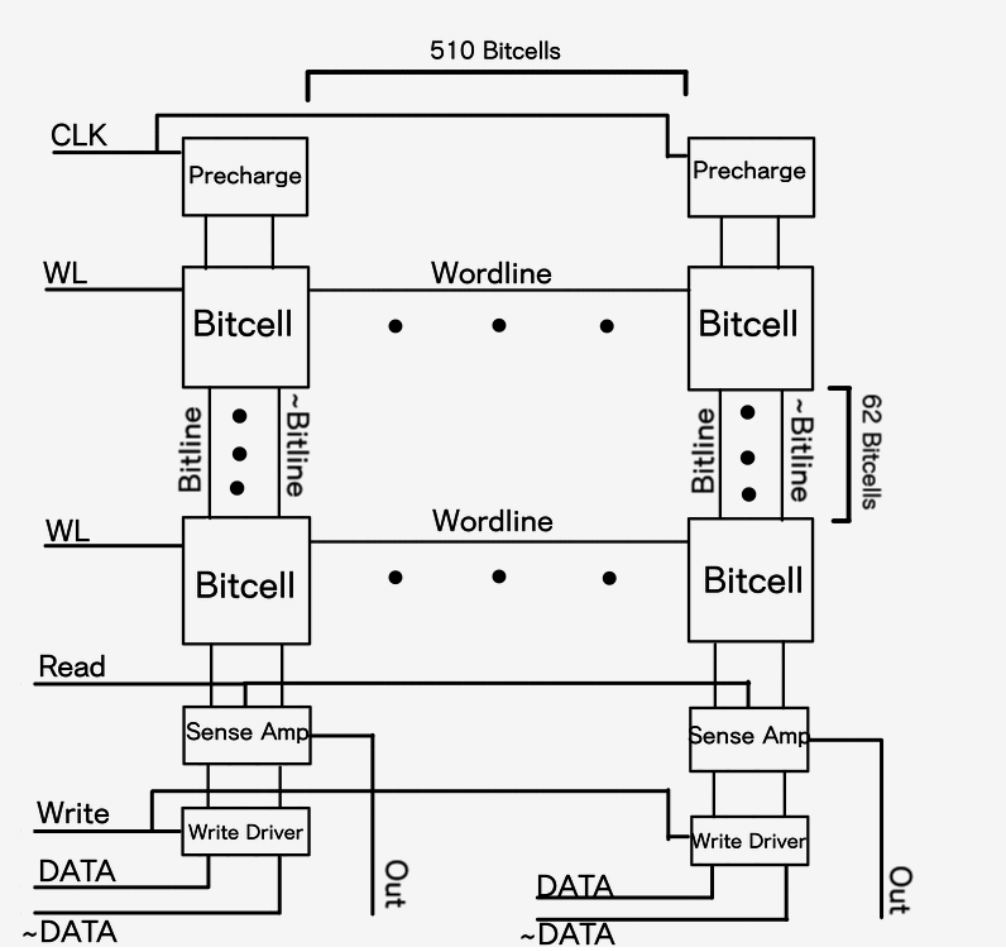


Figure 8: Block Schematic

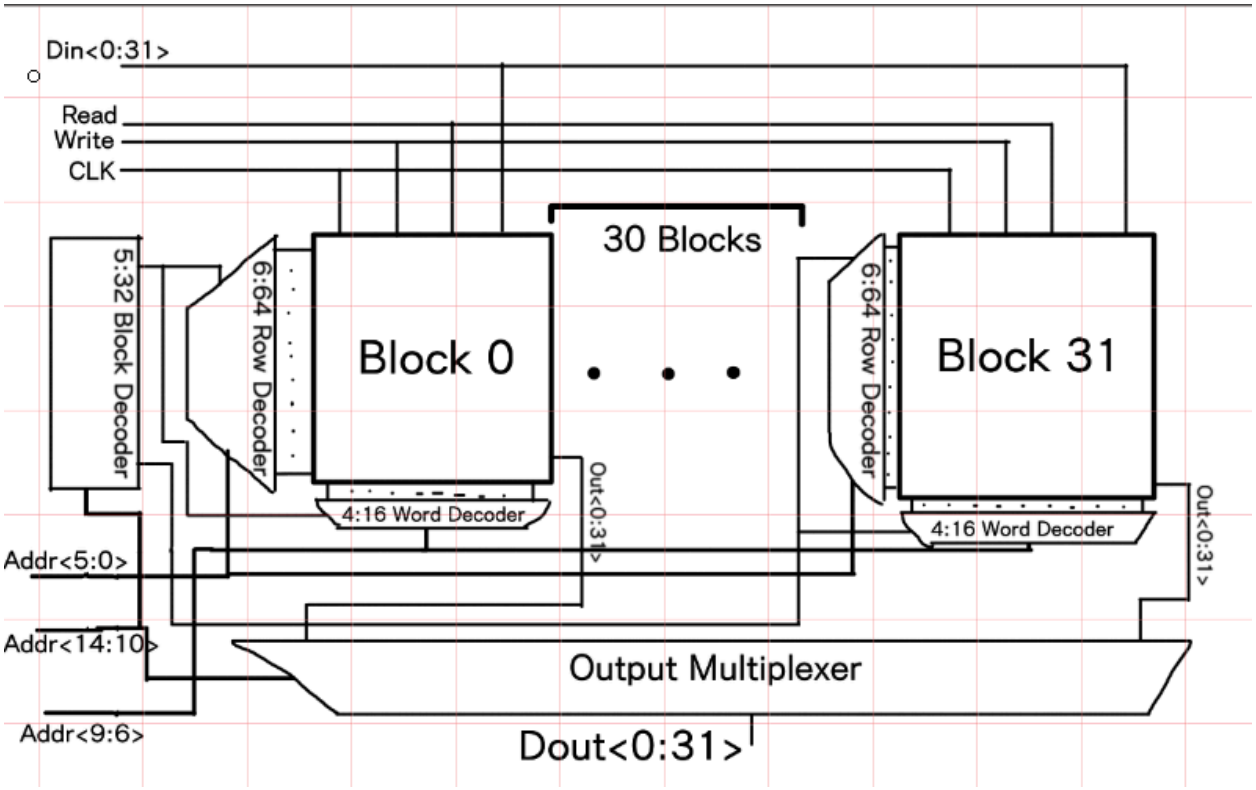


Figure 9: Full SRAM Schematic

We think that the above schematics represent an optimal low-power solution to the SRAM problem. We have used data to simulate the maximum reasonable size of our blocks, while still keeping our decoders and multiplexers of a small size (<= 6 bits). Additionally, because we picked such a large block width, we are able to use only 32 blocks in our final design. We have been and will continue to strive to realize this design.

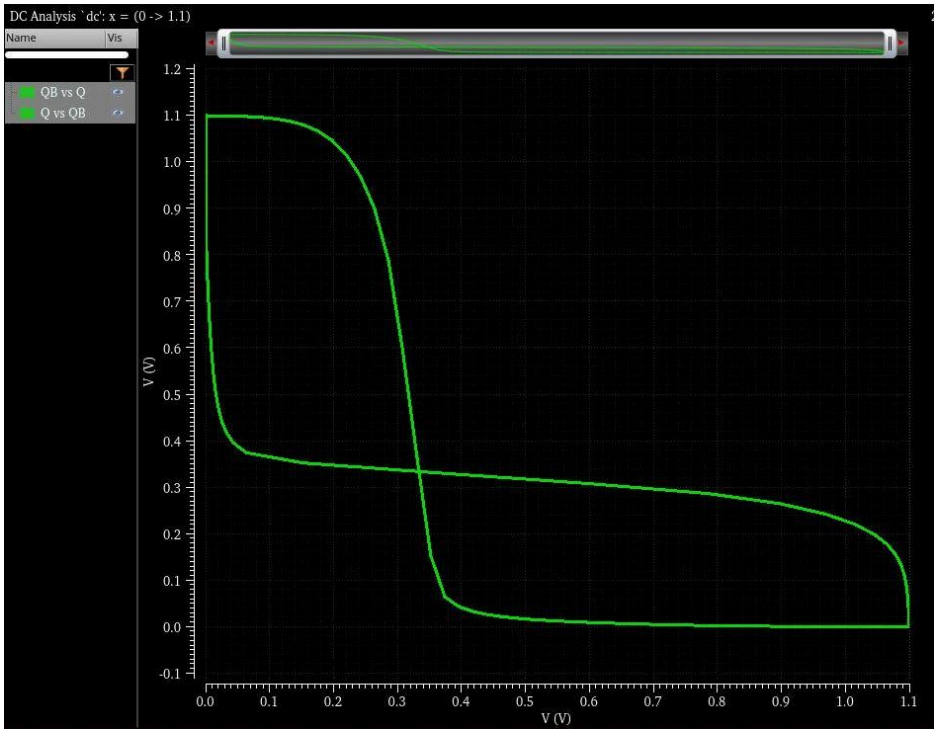
IV. Timeline

Below is the final timeline. Each part is due by the listed date at midnight.

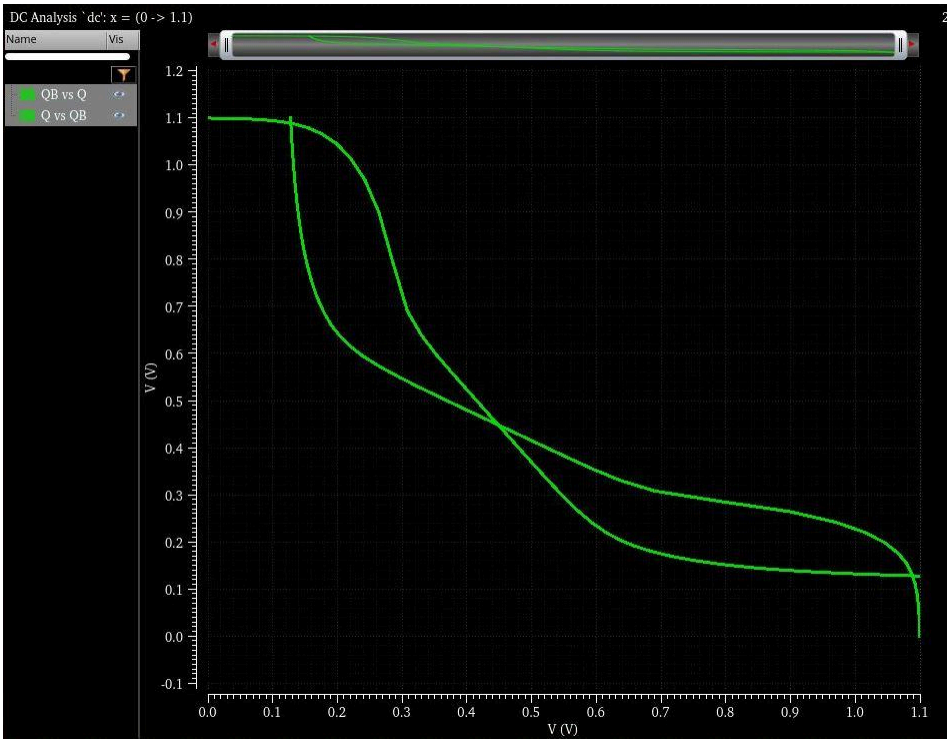
4/9: Test and simulate full SRAM (August)
4/16: Test efficacy of body bias and 7T (Robert, TJ)
4/16: Work to continue optimizing metrics (August)
4/19: Stop changing SRAM design, collect figures (August, TJ, Robert)

4/23: Complete paper and final presentation (August, Robert, TJ)

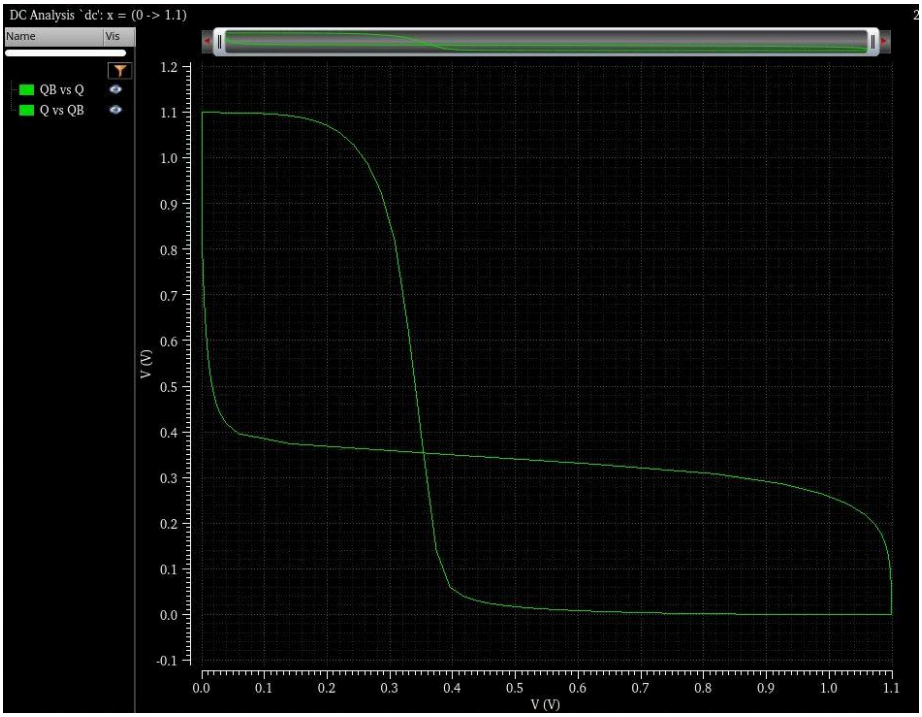
Appendix:



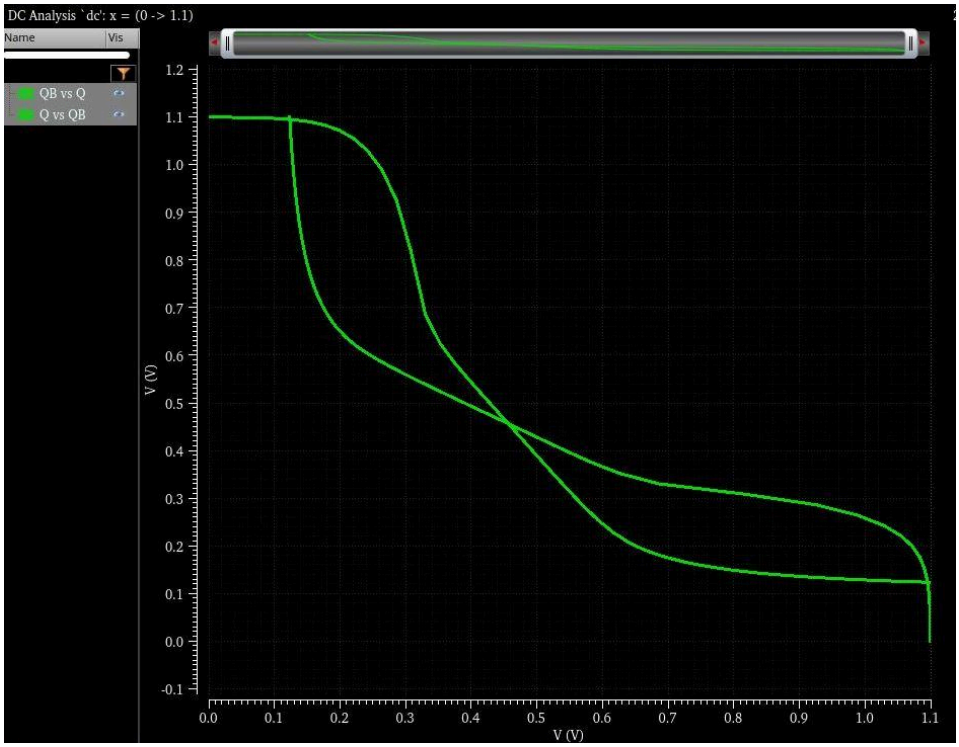
(i): Typical-Typical Static Noise Margin



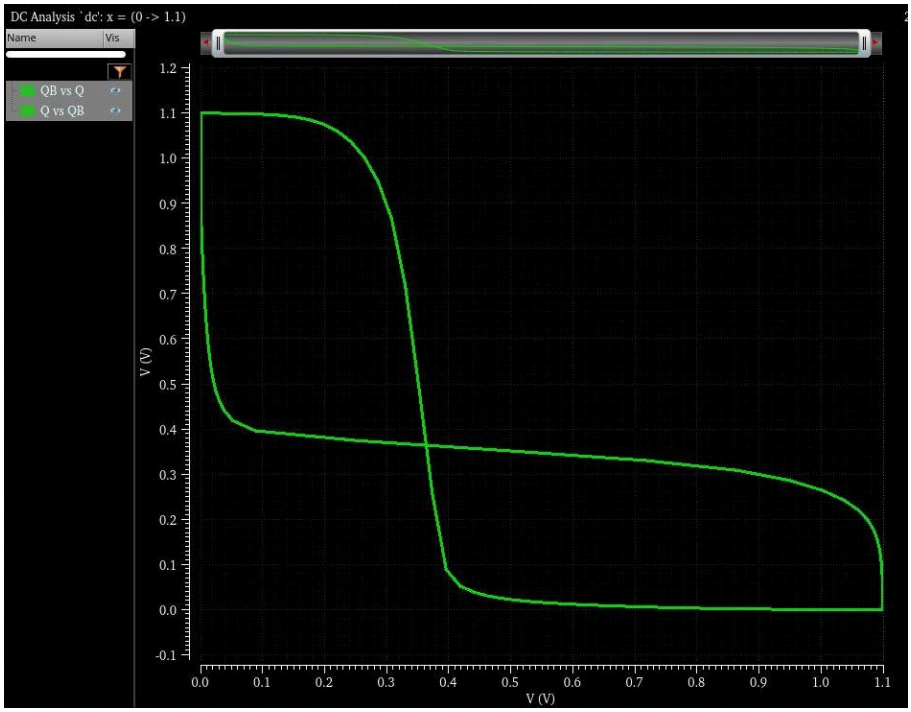
(ii): Typical-Typical Read Noise Margin



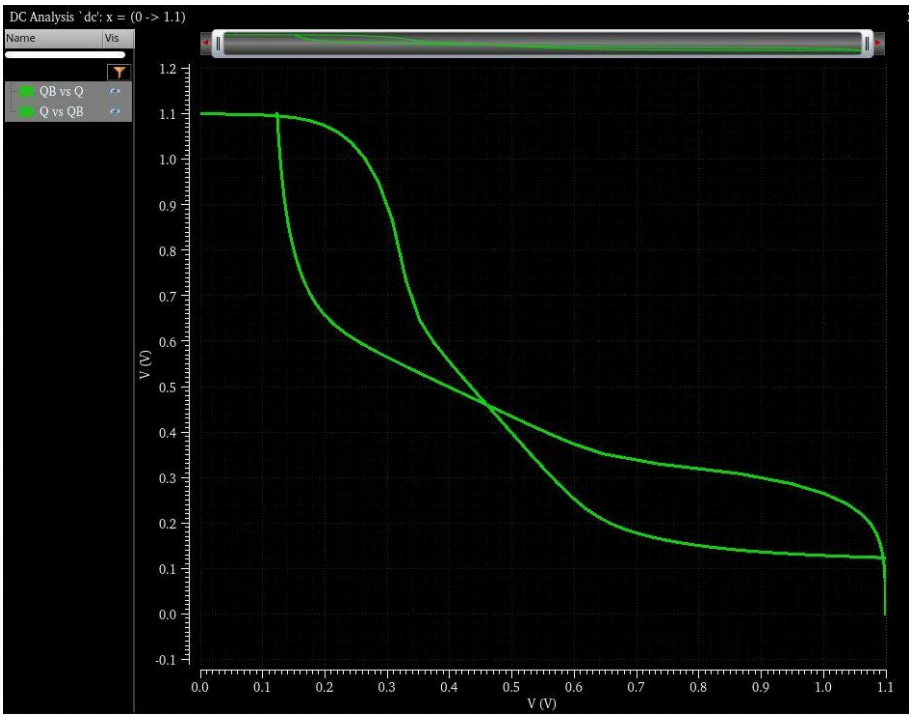
(iii): Slow-Slow Static Noise Margin



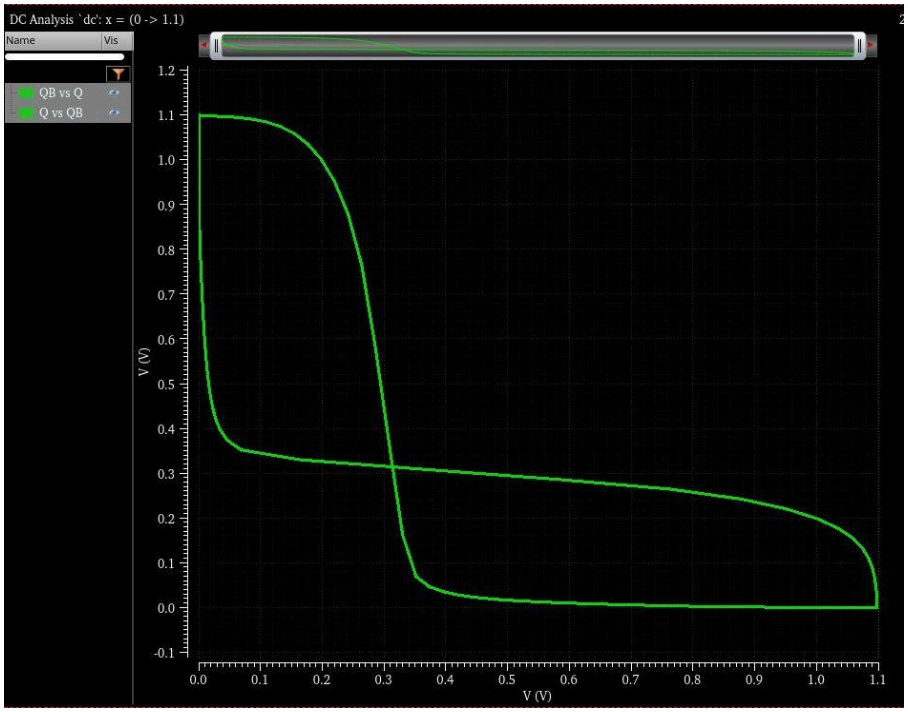
(iv): Slow-Slow Read Noise Margin



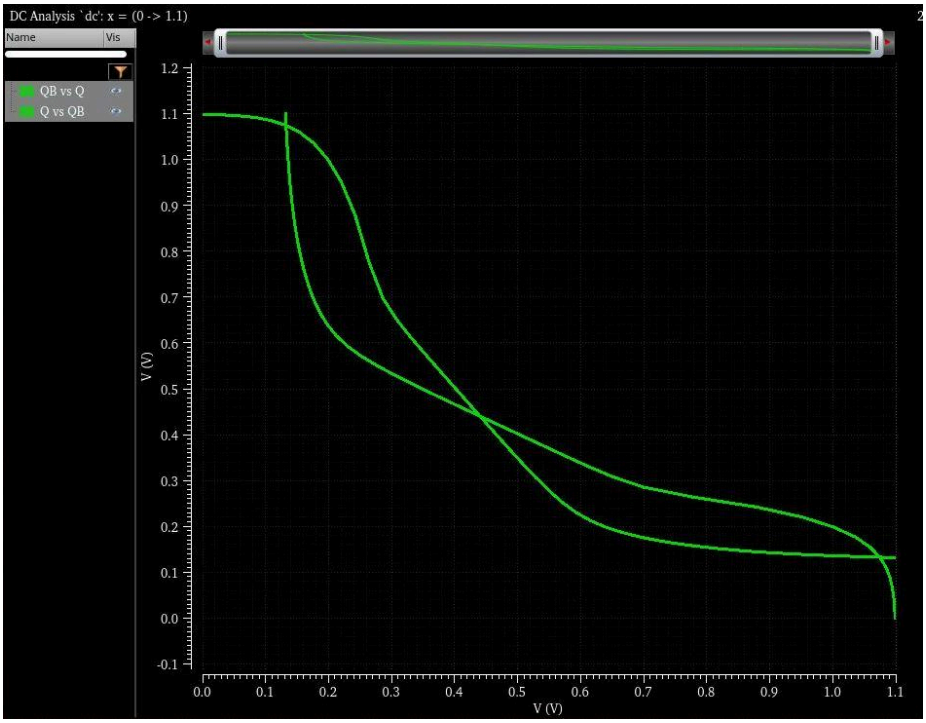
(v): Slow-Fast Static Noise Margin



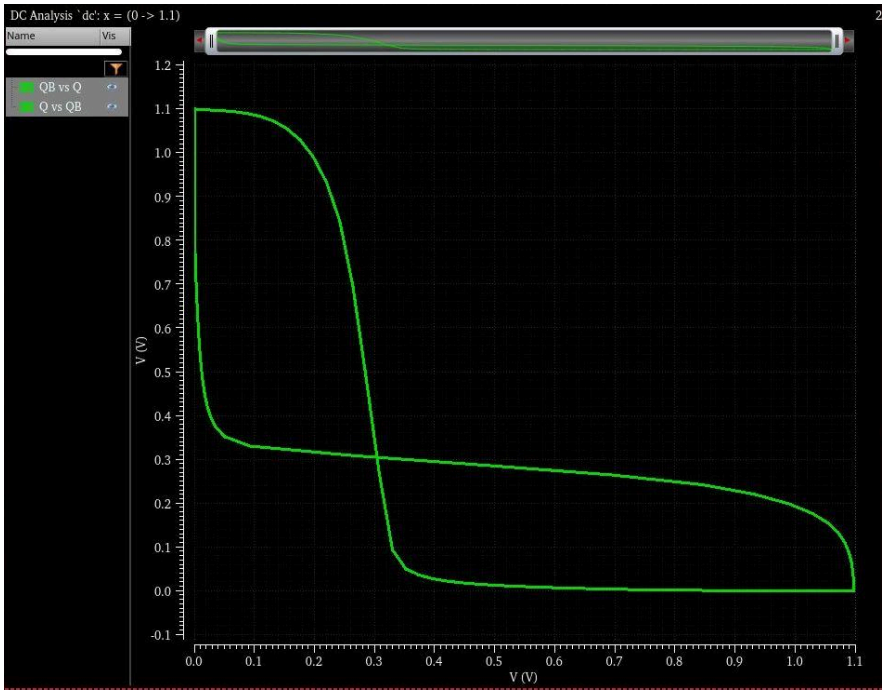
(vi): Slow-Fast Read Noise Margin



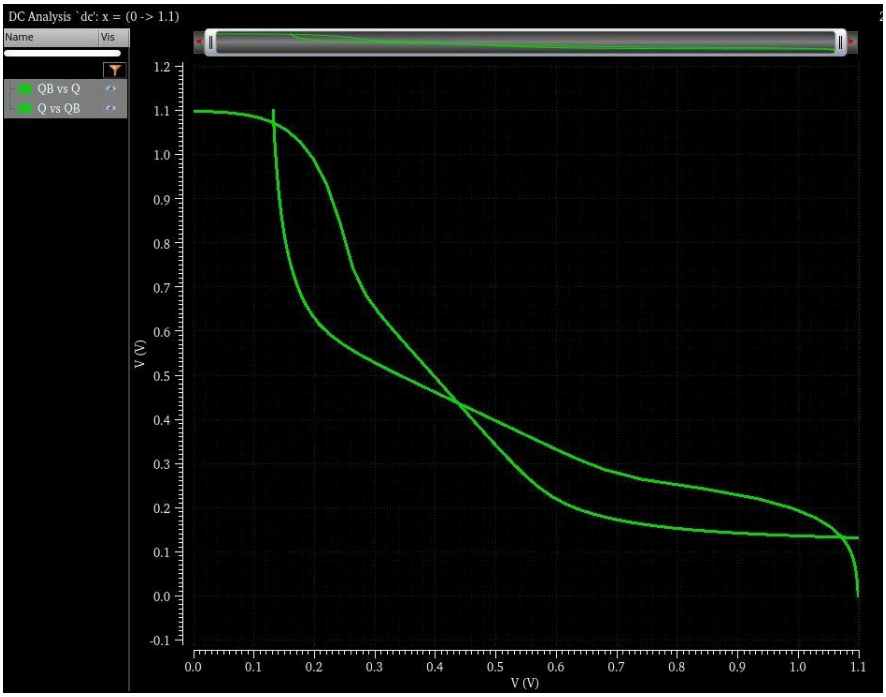
(vii): Fast-Fast Static Noise Margin



(vii): Fast-Fast Read Noise Margin



(ix): Fast-Slow Static Noise Margin



(x): Fast-Slow Read Noise Margin