

Design of 64-bit Synchronous Content-Addressable Memory (CAM) on 3nm TSMC Technology

ECE Department, Georgia Tech April 2024

Benjamin Kubwimana

I. INTRODUCTION

This study implements a 64bit Content addressable memory (CAM) in a 3nm technology with the goal of creating a design optimized for minimum energy delay and area product. A CAM employs benefits of an static random access memory (SRAM) along with comparison capability. The ability of CAMs to perform parallel searches makes them valuable for accelerating data-intensive operations in processing-in-memory (PIM) architectures, as well as in diverse applications such as Translation Lookaside Buffers (TLBs) and network routers. This study contains three main parts, a preliminary evaluation of CAM architecture, schematic and layout design, and finally a simulation with 20 cycle test vector. The main advantage of CAM is its single cycle delay for searching addresses[1]. A 10-T NOR CAM bitcell architecture was chosen for its fast bit-matching as well as bigger noise margin [1]. Search-line (SL) conditioning was eliminated to reduce energy consumption, additionally, the write-enable signal was used to disable SLs during writing. Minimum sized transistors were used to optimize for area and energy.

II. DESIGN DESCRIPTION

A. BITCELL

Content-addressable memory (CAM) cells combine bit storage with comparison functionality. NOR and NAND cells are common types used in CAM design, both implementing cross-coupled inverters that create D and \bar{D} nodes for bit storage. Figure 2 shows a schematic of a NOR type CAM cell implemented in this study. Notice the bitlines (BL and complementary \bar{BL}) using the access NMOS transistors ($M7/M8$) are used in doing write operations to a cell. On the other hand, comparison operation is essentially an XOR operation between the stored bit and the search bit in both the NAND and NOR types with different implementation.

The reason for choosing a NOR type cell over the NAND type lies in its speedy operation. The goal of the study was to optimize the speed of the largest block of this study and seek to optimize for area and power in the remaining blocks. Thus when it comes to matching delays, a 10 T NOR types achieves faster matching than an equivalent 9T NAND type. Notice that the delay of a NAND cell as the number of cells increases is approximately quadratic, $O(n^2)$, while the NOR cell can be approximated to be linear $O(n)$. We observe from Figure 2 that the critical path during matchline (ML) evaluation is through $M5/M9$ or $M6/M10$ when a miss happens. Additionally, as noted by Pagiamtzis et al., the NAND type is susceptible to charge sharing which can lead to false matches when a sense amplifier is used and no recharge is implemented in intermediate (cell level) matchlines. On the other hand, adding precharging will increase power consumption.

This study implements a 10T NOR type, however, 9T type is another optional type which was disregarded in this study because it has a weak pull up path (up to VDD-Vtn, Vtn is threshold voltage of nMos) as it uses NMOS pass transistor for bit matching. Thus providing a low noise margin, a problem shared with NAND type cells.

There is several configurations for a 10T NOR type, the configuration shown in Figure 2 is chosen to minimize charge sharing

during ML evaluation that occurs due to dynamic nature of the searchlines that cause node X_L and X_R to share charge with ML if $M5/M6$ are directly connected to the ML. This study employs the conventional precharge-high technique that involves precharging match-line followed by the match-line evaluation phase.

B. ROW DECODER

A dynamic row decoder was chosen over hierarchical decoders due to the need for fewer transistors and less power. [2] A NOR was chosen instead of NAND for its speed advantage. Initially, all transistors were of minimum size. However, unselected writelines were failing to pull down below 0.3V, creating risk of unintentional writes. So, NMOS fin count was increased to 8, and unselected matchlines were pulled to <0.1V.

C. CONDITIONERS

Initially a 'current-race scheme' type ML conditioner, as described by Pagiamtzis et al., was selected because it eliminated the need for SL conditioning. Debug difficulty led to using simple pull up ML conditioner. However, prior to addition of the now-required SL conditioners, near correct circuit functionality was observed. So, SL conditioner functionality was omitted. A simple 'WENB gating' to the SLs and BLs was added to ensure that lines were only active when needed.

D. ENCODER + FOUND LOGIC

[3] The 8 to 3 bit encoder used was made using three 4 bit OR gates. The found logic used an 8 bit OR. The logic involved looked at each of the Match Lines in the input and if any ML flipped, then a found signal was asserted.

E. D-FLIP-FLOP

TSPC based Flip Flops were used because of the advantage of not having 2 clock phases and to eliminate clock-overlap issues[2]. Only minimum sized transistors were used for the design. Sizing up the transistors would have saved some glitches on the final output but had to be dropped because of time constraints.

III. RESULTS

Table I presents design statistics pertaining to power, area, and delay.

IV. CONCLUSION

This study presents a 64-bit Content Addressable Memory implemented on a 3nm technology using a ten-transistor NOR-type architecture. Schematic and layout were implemented and evaluated for correctness using DRC and LVS examination tools. Simulations were performed with a 20-cycle test vector. Energy-delay-area product was determined to be $5.20 \text{ pJ} * \text{ns} * \mu\text{m}^2$. Sink rise time, probed at clk input to DFFs, was measured to be 32.2 ps. Further design statistics are summarized in table 1.

ACKNOWLEDGMENT

This work was done in collaboration with Taylor Templeton and Francis John from North Carolina State University.

REFERENCES

- [1] K. Pagiamtzis and A. Sheikholeslami, "Content-addressable memory (CAM) circuits and architectures: a tutorial and survey," in IEEE Journal of Solid-State Circuits, vol. 41, no. 3, pp. 712-727, March 2006, doi: 10.1109/JSSC.2005.864128.
- [2] Class notes "ECE 546", Dr. W.R Davis.
- [3] microchip, The Configurable Logic Cell (CLC) module, 8-to-3 Binary Encoder, <https://onlinedocs.microchip.com/pr/GUID-054EB76B-0E74-464C-B77F-C328232E814B-en-US-2/index.html?>

APPENDIX

TABLE I
DESIGN STATISTICS

Metric	Value
Operating Supply voltage	0.8V
Minimum supply voltage	0.2V
Read access time	75ps
Area	$28.3 \mu m^2$
Transistor Count	726NMOS, 306PMOS
Transistors/Area	$36.5/\mu m^2$
Total Energy	2.45 pJ
EDA Product	$5.20 \text{ pJ}^*\text{ns}^*\mu m^2$
Design Time (Estimate)	168 hrs

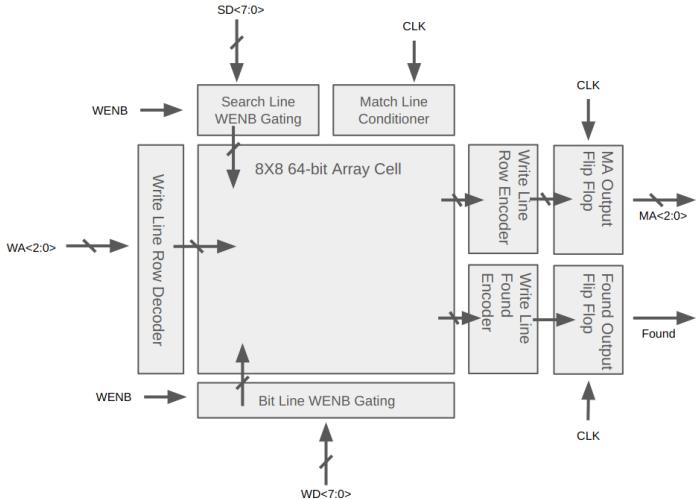


Fig. 1. Block Diagram

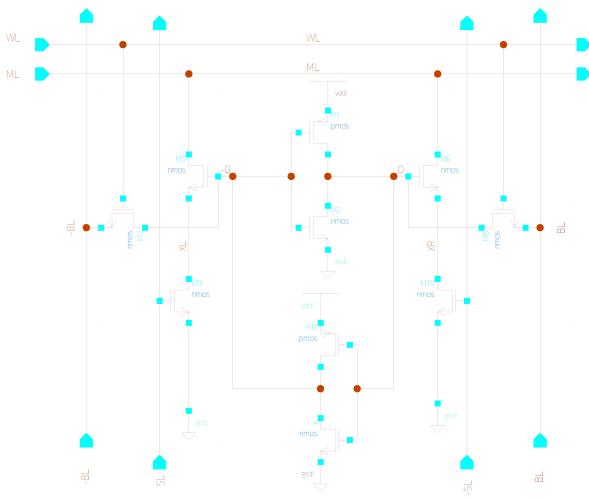


Fig. 2. Bitcell Schematic

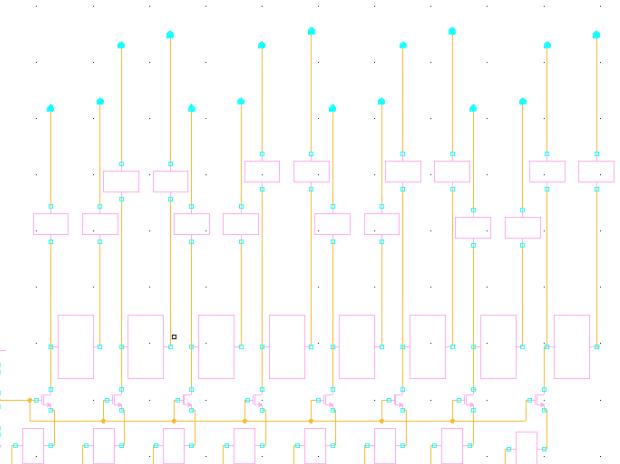


Fig. 3. Bit conditioner Schematic

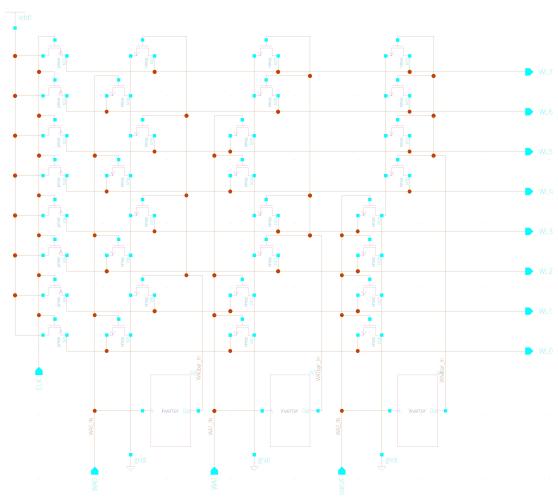


Fig. 5. Row Decoder Schematic

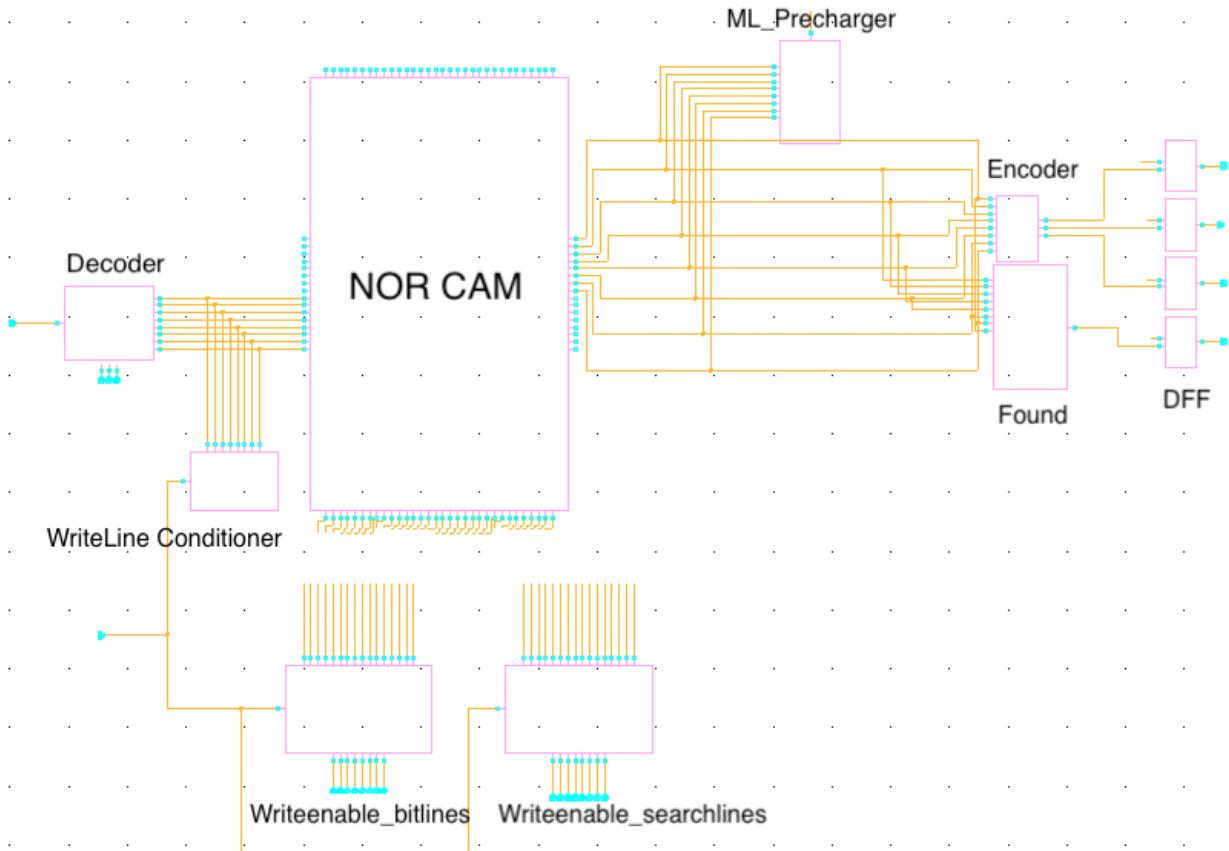


Fig. 4. CAM Schematic

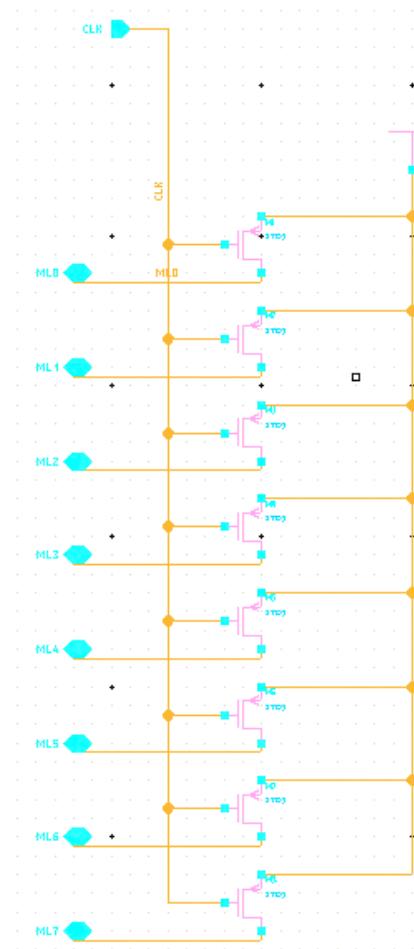


Fig. 6. Matchline precharger

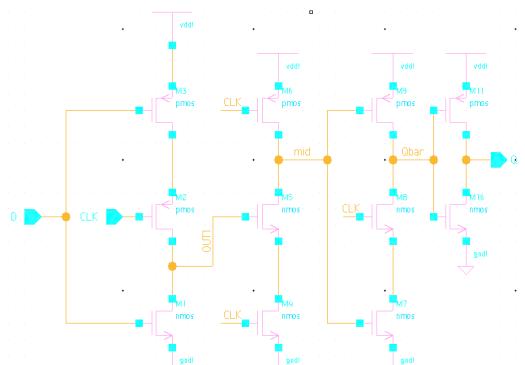


Fig. 7. DFF Schematic

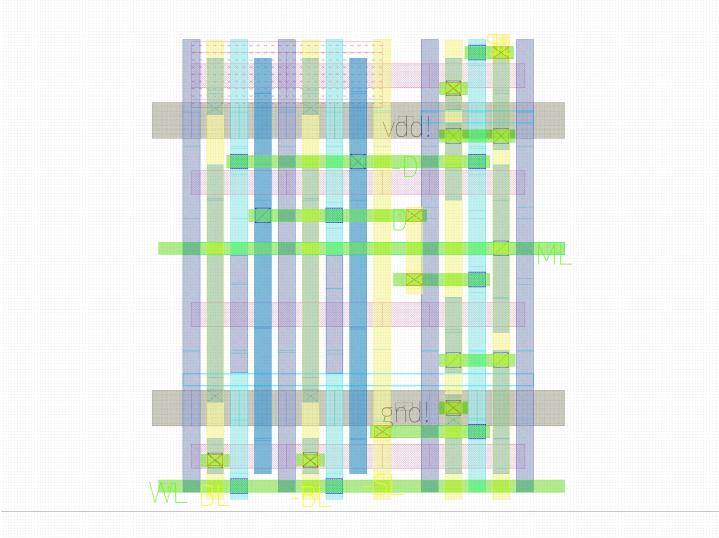


Fig. 8. 10T bitcell Layout

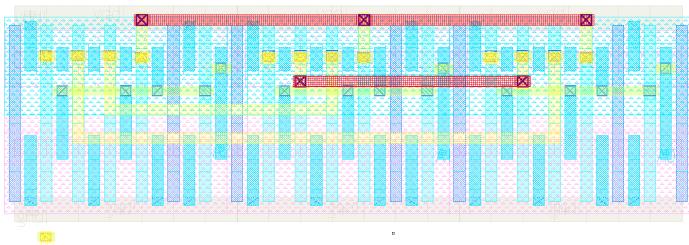


Fig. 9. 8-3 Bit Encoder Layout

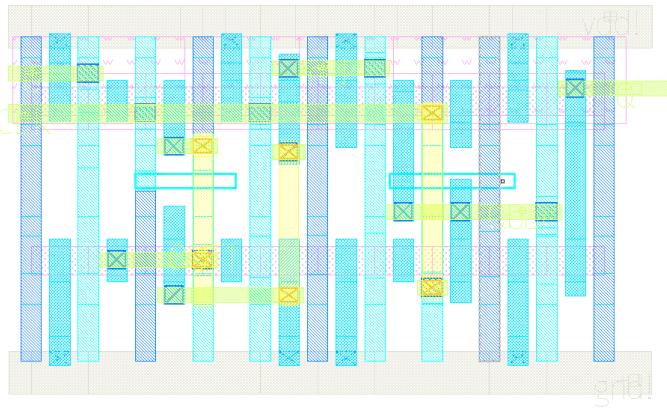


Fig. 10. DFF Layout

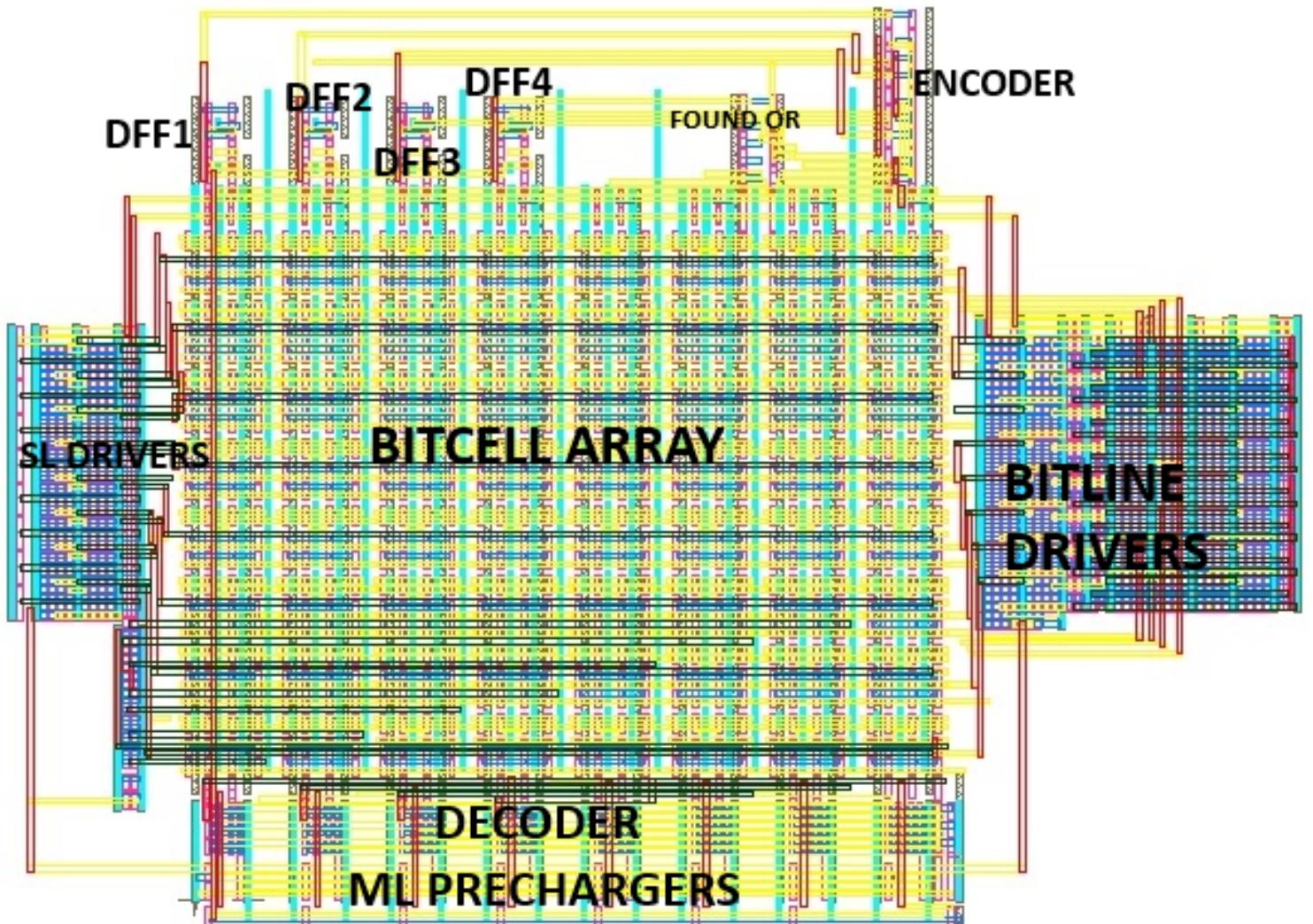


Fig. 11. 10-T NOR CAM



Fig. 12. 8T ML Precharger

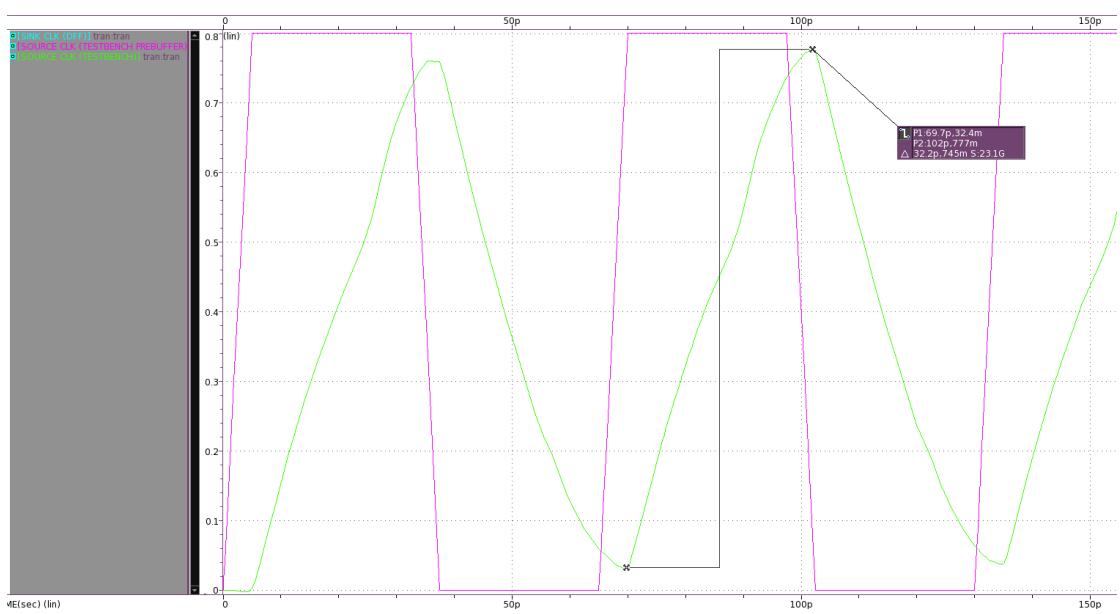


Fig. 13. Sink Rise Time Analysis

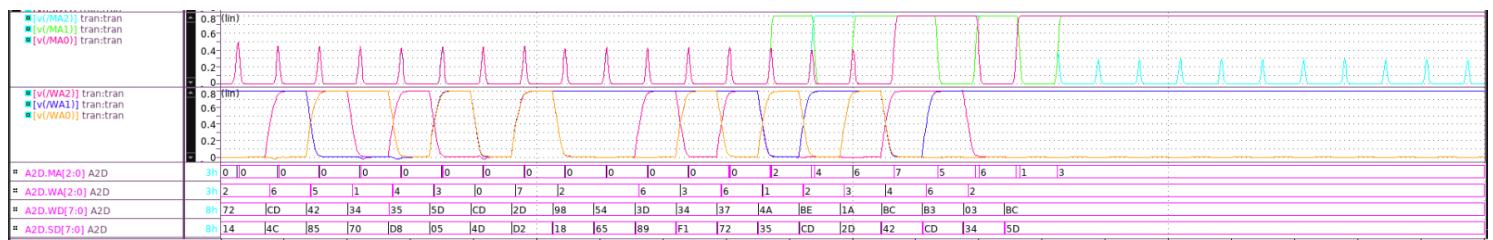


Fig. 14. Data Waveform