

Design and Technology Scaling Analysis of CMOS Priority Encoder: A Comparative Study Across TSMC 180nm, 130nm, 45nm, and 16nm Nodes

Smarajit paragranjan Mishra
dept. schoool of electronics sciences
Odisha University of Technology and
Research
Bhubaneswar, India
mishrasmarajit03@gmail.com

Abstract— The encoder is a combinational circuit that performs the inverse operation of a decoder. It translates input signals into binary-coded output lines. A priority encoder is a specific type that produces outputs based on the highest-priority active input. Compared to traditional logic gate-based implementations, the CMOS-based priority encoder demonstrates a significant advantage in power efficiency. The experimental analysis across various technology nodes — from 250nm to 16nm — shows a consistent and substantial reduction in average power consumption with technology scaling. This reduction validates the suitability of CMOS design for low-power applications, especially in modern high-density integrated circuits where minimizing power is crucial. The design also highlights how priority-based logic control can be optimized with advanced CMOS processes, ensuring both functional correctness and energy efficiency.

Keywords— CMOS, Priority Encoder, Power, 3-bit, LT Spice

I. INTRODUCTION

A priority encoder is a combinational logic circuit that outputs the binary representation of the highest-priority active input among several input lines. In an 8-to-3 priority encoder, eight input signals are encoded into a 3-bit binary output representing the index of the highest-priority active input. For example, if both input lines I_7 and I_5 are high simultaneously, the output reflects input 7 due to its higher priority. Priority encoders are widely used in applications such as flash analog-to-digital converters (ADCs), interrupt controllers, and data multiplexers, where efficient identification of the most significant active signal is crucial.

Power consumption has become a vital design consideration in modern digital systems. CMOS circuits offer very low static power dissipation, but dynamic power consumption—caused by the frequent switching of logic states—can dominate at high operating frequencies. This dynamic power is primarily due to the charging and discharging of load capacitances, and is generally proportional to the square of the supply voltage (V_{DD}^2) and switching frequency.

Technology scaling, transitioning from larger nodes like 250 nm to advanced nodes such as 16 nm, typically reduces dynamic power per gate. This is achieved through simultaneous reductions in both transistor dimensions and supply voltage. In this work, an 8-to-3 CMOS priority encoder is implemented and simulated using LTspice across four technology nodes: 250 nm, 180 nm, 45 nm, and 16 nm. The results show a consistent reduction in average power consumption—from 10.816 μ W at 250 nm to just 1.7141 μ W at 16 nm—demonstrating the effectiveness of scaling in achieving energy-efficient design. These findings support the

ongoing shift toward lower technology nodes in digital logic design, especially for power-sensitive applications.

II. DESIGN METHODOLOGY

The Design Methodology section outlines the systematic approach used to construct and analyze an 8-to-3 CMOS-based priority encoder across various technology nodes. The design begins with the implementation of the encoder logic using CMOS logic gates, focusing on efficient transistor-level design. The circuit is simulated in LTspice, and power consumption is evaluated across four process technologies—250 nm, 180 nm, 45 nm, and 16 nm—by scaling device parameters such as channel width, length, and supply voltage. For each node, dynamic power is calculated based on average current drawn from the supply, ensuring consistency in testing conditions. This methodology provides a comparative view of how technology scaling impacts power efficiency in modern CMOS encoder designs. Use the enter key to start a new paragraph. The appropriate spacing and indent are automatically applied.

A. 8-to-3 Encoder and parity encoder

a) *Encoder* : An encoder is a digital combinational circuit that transforms human-readable input into a binary coded format suitable for machine processing. It plays a crucial role in various digital systems such as data transmission, automation, communication, and signal processing. An encoder typically has 2^n input lines and n output lines, where it encodes the active input into an n -bit binary value. A common example is the 8-to-3 encoder, also known as an octal-to-binary encoder, which takes eight input lines and produces a 3-bit output based on the highest-priority active input. This type of encoder is widely used in systems requiring efficient data representation and control.

$$\begin{aligned} Y_0 &= I_1 + I_2 + I_3 + I_4 \\ Y_1 &= I_2 + I_3 + I_6 + I_7 \\ Y_2 &= I_4 + I_5 + I_6 + I_7 \end{aligned}$$

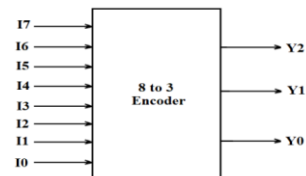


Fig.1. 8 TO 3 ENCODER

TABLE I. TRUTH TABLE FOR 8 TO 3 ENCODER

I7	I6	I5	I4	I3	I2	I1	I0	Y2	Y1	Y0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1

b) *Parity Encoder:* A priority encoder is a combinational logic circuit that identifies the highest-priority active input among multiple input lines and generates a binary output corresponding to that input. Unlike a multiplexer, which selects and transmits data from one input to the output, a priority encoder evaluates all inputs and encodes the position of the highest-priority logic '1' input. In essence, it converts multiple active inputs into a simpler binary representation based on priority.

Digital encoders, often referred to as binary encoders, typically feature 2^n input lines and n output lines. Depending on the configuration, standard encoder types include 4-to-2, 8-to-3, and 16-to-4 encoders. These are designed to produce binary or binary-coded decimal (BCD) outputs from active logic-high inputs. A common IC example is the TTL 74LS148, an 8-to-3 priority encoder that accepts eight active-low inputs and outputs a 3-bit binary code representing the highest-priority active input.

For instance, if multiple inputs such as D2, D3, and D5 are active simultaneously, the encoder will prioritize and generate the output code for D5 (binary 101), as it has the highest precedence. When D5 is deactivated, the next highest active input, D3, will be encoded (binary 011), and so on. This makes priority encoders extremely useful in systems like interrupt controllers, data selectors, and communication protocols where signal arbitration is crucial.

$$\begin{aligned}
 Q_2 &= D_7 + D_6 + D_5 + D_4 \\
 Q_1 &= D_7 + D_6 + (\bar{D}_5 \bar{D}_4 (D_3 + D_2)) \\
 Q_0 &= D_7 + (\bar{D}_6 (D_5 + \bar{D}_4 D_3 + \bar{D}_4 \bar{D}_3 \bar{D}_2 D_1)) \\
 V &= D_7 + D_6 + D_5 + D_4 + D_3 + D_2 + D_1 + D_0
 \end{aligned}$$

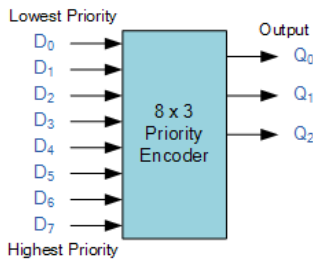


Fig.3. 8 TO 3 PARITY ENCODER

TABLE II. TRUTH TABLE FOR 8 TO 3 PARITY ENCODER

D7	D6	D5	D4	D3	D2	D1	D0	Q2	Q1	Q0	V
0	0	0	0	0	0	0	0	X	X	X	0
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	1	X	0	0	1	1
0	0	0	0	0	1	X	X	0	1	0	1
0	0	0	0	1	X	X	X	0	1	1	1
0	0	0	1	X	X	X	X	1	0	0	1
0	0	1	X	X	X	X	X	1	0	1	1
0	1	X	X	X	X	X	X	1	1	0	1
1	X	X	X	X	X	X	X	1	1	1	1

c) *8 to 3 Parity Encoder using CMOS logic:* In this design, the outputs of the 8-to-3 priority encoder (A, B, and C) are implemented using CMOS logic. For each output, PMOS transistors form the pull-up network and NMOS transistors form the pull-down path. The idea is simple: whenever more than one input is active, the output should reflect only the highest priority input, so the lower priority ones get blocked automatically. The logic is arranged in such a way that D7 has the highest priority and D0 the lowest.

The architecture for output A checks inputs D7 to D4, for output B checks D7 to D3, and for output C checks D7 to D1. The transistors are cascaded according to the priority sequence, so the output value is always correct even if multiple inputs are high. A capacitor of 100μF is connected at each output to measure switching energy and get realistic power readings during simulation.

Similarly, the Valid (V) output is designed using CMOS logic to indicate if any input is active. If even one of the inputs D7–D0 is '1', then V becomes '1', meaning the output is meaningful. If all inputs are 0, then V stays 0.

Overall, CMOS logic reduces power because only a small amount of charge shifts during switching, and static leakage is extremely low. The design works efficiently across all technology nodes used in this project.

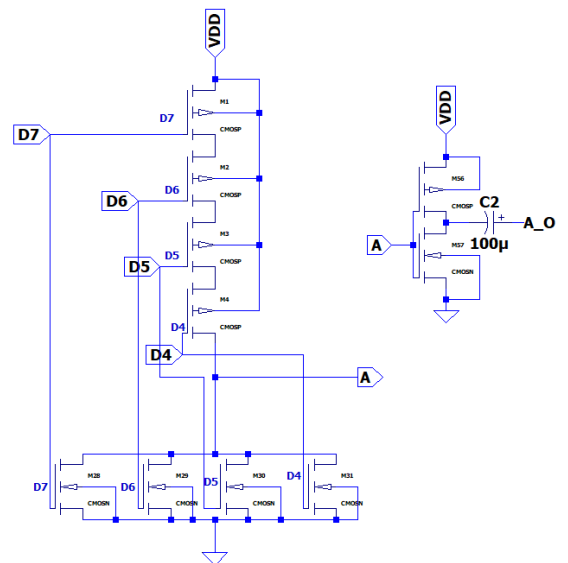


Fig.4. CMOS circuit for A output

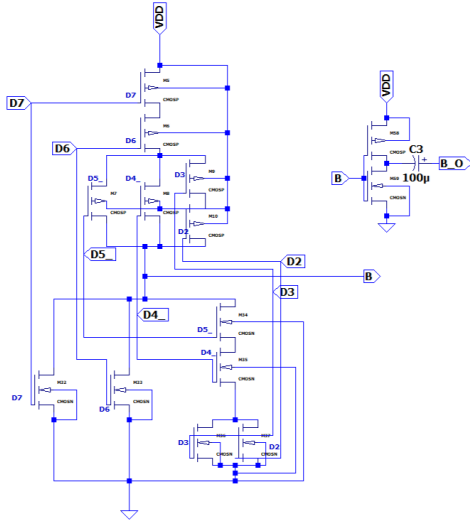


Fig.5. CMOS circuit for B output

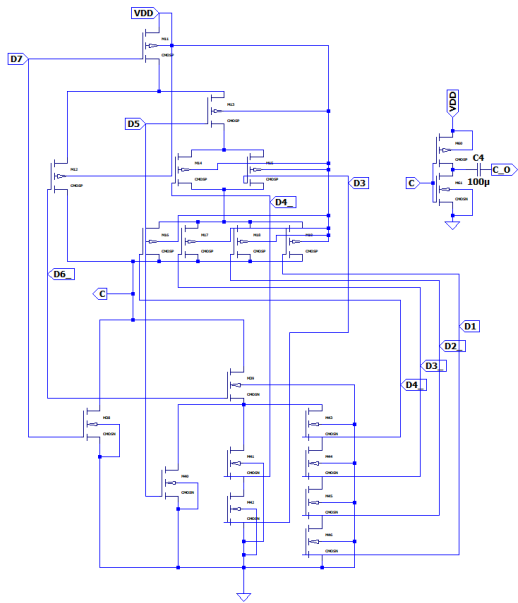


Fig.6. CMOS circuit for C output

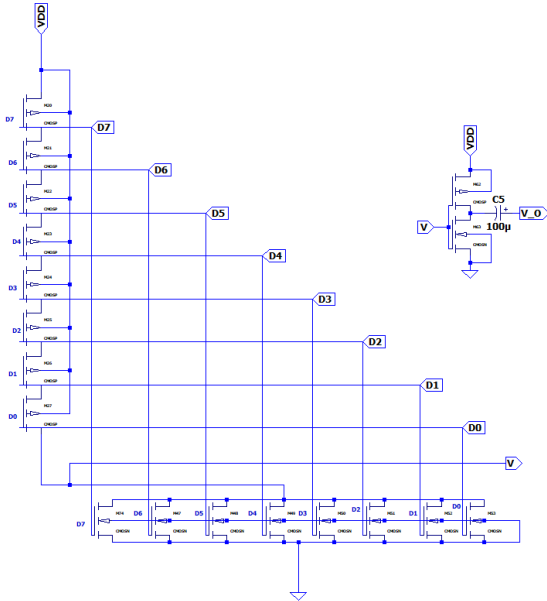


Fig.7. CMOS circuit for V output

III. RESULTS AND ANALYSIS

In this work, power consumption values of the CMOS-based 8-to-3 priority encoder were evaluated across different technology nodes. LTspice transient simulations were used to measure the average power drawn from the supply (VDD). A capacitive load of $100\mu\text{F}$ was used at each output to reflect real switching behavior.

The results clearly show that as the technology shrinks from 250 nm to 16 nm, the power consumption decreases significantly. This is because both the supply voltage and the effective switching capacitance reduce at smaller feature sizes. The measured results obtained from simulation are summarized in Table X.

The reduction in power is almost linear with scaling, indicating that CMOS implementation remains energy-efficient even in deep sub-micron and nano-scale nodes. This analysis confirms that CMOS logic-based design is suitable for low-power digital applications, especially where multiple logic units operate simultaneously.

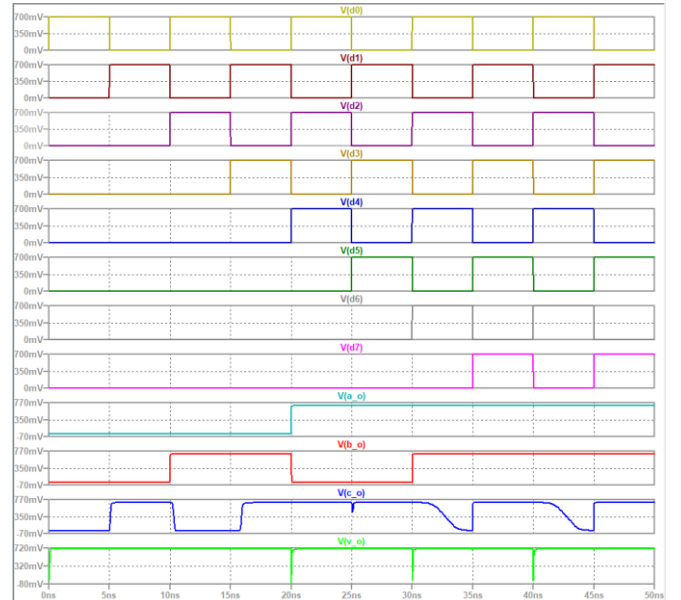


Fig.9. Simulated waveform output using 16 nm node.

When D0 is high, the output C stays low as expected since it has the lowest priority. As inputs like D1 and D2 go high, the encoder output changes to reflect the higher priority values. When D4 becomes high, output C takes a moment to respond because it must override the effect of lower priority inputs. Around 20–25 ns, C momentarily holds a high level even though it should go low; this happens due to internal wire delay. After this short delay, output C settles to the correct value, confirming proper priority functionality.

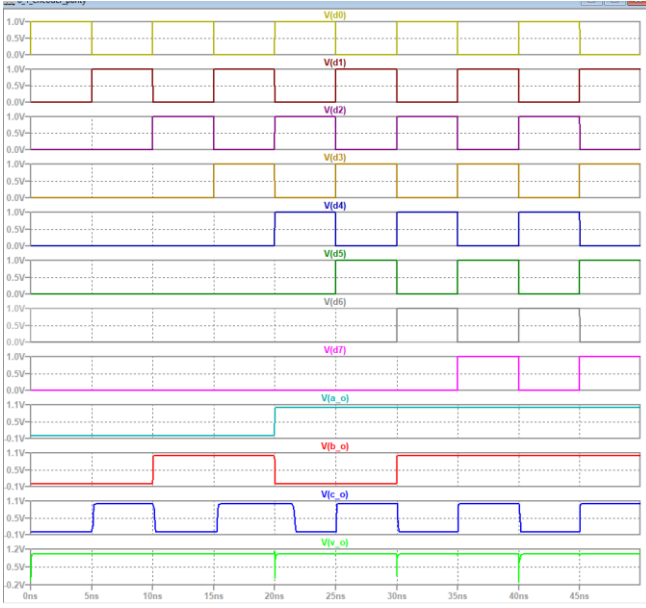


Fig.10. Simulated waveform output using 45 nm node.

When the input signals switch in the 45 nm simulation, all the outputs follow the expected priority encoder behavior. Output C correctly changes whenever a higher-priority input becomes active. The only small issue appears around 20–25 ns, where C shows a short delay before settling to the new value. This happens because the circuit momentarily holds the previous state while the internal nodes and wiring capacitances discharge. After this brief delay, the output becomes stable again, and the rest of the waveform behaves normally.

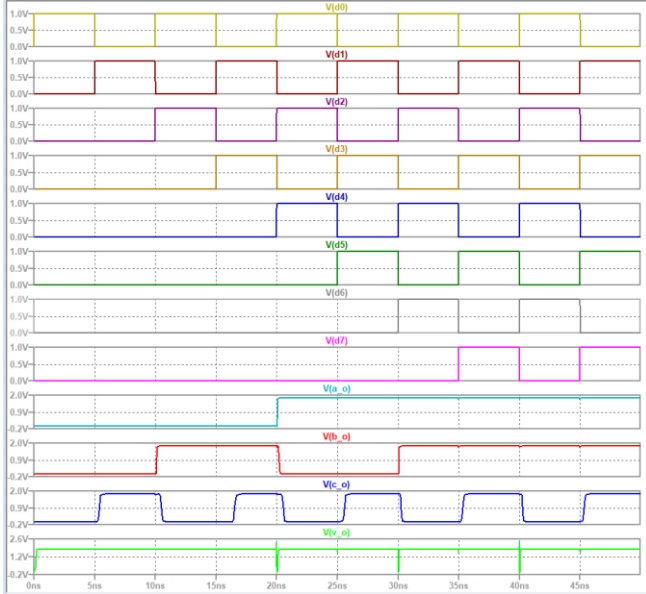


Fig.11. Simulated waveform output using 180 nm node.

For the 180 nm technology node, the output behavior is correct and follows the priority rules. However, small timing delays are observed in the C output whenever the input changes, particularly around 5 ns, 10 ns, and 15 ns transitions. This delay is likely due to circuit-level switching and internal capacitance effects. After a brief settling period, the output continues in a delayed way.

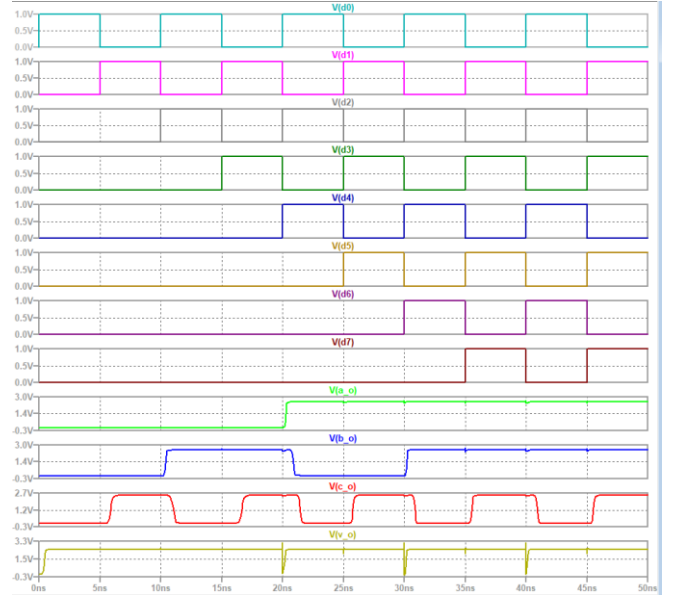


Fig.12. Simulated waveform output using 250 nm node.

In the 250 nm waveform, all the outputs (A, B, C, and V) follow the expected priority logic pattern. Whenever inputs D0 to D7 change, the outputs update correctly. However, both output B and output C show a small delay at multiple transitions. For output B, we can notice this delay around 10 ns, 20 ns, and 30 ns before it settles to the correct level. Output C shows this effect more frequently: at 5 ns, 10 ns, 15 ns, 20 ns, and even at 25 ns, the output holds briefly at its previous state before switching. This happens because, in 250 nm technology, the transistor rise/fall times are slower and the load capacitance is higher, which causes a short “hold” before the output stabilizes. Despite these delays, the circuit behavior is correct and all outputs eventually reach the proper logic levels according to priority rules

TABLE III. Power Comparison of Priority Encoder Across Different Technology Nodes

<i>Technology node</i>	<i>Vdd(v)</i>	<i>Width(w)</i>	<i>Length(l)</i>	<i>Average power consumption</i>
250 nm	2.5 V	360 nm	240 nm	10.816 μ W
180 nm	1.8 V	270 nm	180 nm	7.529 μ W
45 nm	1.0 V	90 nm	45 nm	4.0687 μ W
16 nm	0.7 V	32 nm	16 nm	1.7141 μ W

IV. CONCLUSION

The 8-to-3 priority encoder was successfully implemented using CMOS technology at four different process nodes—250 nm, 180 nm, 45 nm, and 16 nm—and the power characteristics were analyzed through SPICE simulations. The results clearly show that power consumption reduces significantly with technology scaling: from 10.816 μ W at 250 nm down to only 1.714 μ W at 16 nm. Although the waveforms exhibit minor propagation delays, especially in the 250 nm and 180 nm designs, the overall functionality remains correct across all nodes. The 45 nm and 16 nm implementations demonstrate excellent stability with minimal delay, proving that lower technology nodes not only reduce dynamic power but also enhance signal settling. Thus, CMOS-based priority encoders benefit greatly from scaling, making deep-submicron implementations highly suitable for low-power digital systems and VLSI applications.

REFERENCES

- [1] Jinn-Shyan Wang and Chung-Hsun Huang, "High-speed and low-power CMOS priority encoders," in *IEEE Journal of Solid-State Circuits*, vol. 35, no. 10, pp. 1511-1514, Oct. 2000, doi: 10.1109/4.871331.
keywords: {CMOS logic circuits;Signal design;Switching circuits;Digital systems;Delay;Costs;Energy consumption;Equations},
- [2] Balobas, Dimitrios & Konofaos, Nikos. (2017). High-Performance and Energy-Efficient 256-Bit CMOS Priority Encoder. 122-127. 10.1109/ISVLSI.2017.30.
- [3] Y. S. Goud, K. S. V. Prasad, S. V. Kumar and S. Shaikshavali, "Design and Implementation of Low Power 3-Bit Encoder Using Memristor on LT Spice," 2024 IEEE International Conference on Information Technology, Electronics and Intelligent Communication Systems (ICITEICS), Bangalore, India, 2024, pp. 1-6, doi: 10.1109/ICITEICS61368.2024.10625200. keywords: {Semiconductor device modeling;Analytical models;Memristors;Logic gates;Energy efficiency;Encoding;Circuit synthesis;Low Power;3-bit Encoder;Pseudo NMOS logic;CMOS logic;Circuit Design;Energy Efficiency;Power Analysis;Combinational Circuits},