International Journal of Research in Engineering, Science and Management
Volume-1, Issue-11, November-2018
www.ijresm.com | ISSN (Online): 2581-5792

Design of Low-Power High-Performance 2-4 and 4-16 Mixed-Logic Line Decoders

M. Sai Prasanna[1], K. Seetharam[2]
[1]*M. Tech. Student, Dept. of VLSI System Design, Chaitanya Inst. of Technology and Science, Warangal, India*
[2]*Associate Professor, Dept. of VLSI System Design, Chaitanya Inst. of Technology and Science, Warangal, India*

***Abstract*: This brief introduces a mixed-logic design method for line decoders, combining transmission gate logic, pass transistor dual-value logic, and static complementary metal-oxide semiconductor (CMOS). Two novel topologies are presented for the 2–4 decoder: a 14-transistor topology aiming on minimizing transistor count and power dissipation and a 15-transistor topology aiming on high power-delay performance. Both normal and inverting decoders are implemented in each case, yielding a total of four new designs. Furthermore, four new 4–16 decoders are designed by using mixed-logic 2–4 predecoders combined with standard CMOS postdecoder. All proposed decoders have full-swinging capability and reduced transistor count compared to their conventional CMOS counterparts. Finally, a variety of comparative spice simulations at 32 nm shows that the proposed circuits present a significant improvement in power and delay, outperforming CMOS in almost all cases.**

***Keywords*: Semiconductor and voltages**

1. Introduction

Staticcmos circuits are used for the majority of logic gates in integrated circuits. They consist of complementary N-type metal-oxide-semiconductor (nMOS) pulldown and P-type metal-oxide semiconductor (pMOS) pullup networks and present good performance as well as resistance to noise and device variation. Therefore, complementary metal-oxide semiconductor (CMOS) logic is characterized by robustness against voltage scaling and transistor sizing and thus reliable operation at low voltages and small transistor sizes. Input signals are connected to transistor gates only, offering reduced design complexity and facilitation of cell-based logic synthesis and design. Pass transistor logic (PTL) was mainly developed in the 1990s, when various design styles were introduced, aiming to provide a viable alternative to CMOS logic and improve speed, power, and area. Its main design difference is that inputs are applied to both the gates and the source/drain diffusion terminals of transistors. Pass transistor circuits are implemented with either individual nMOS/pMOS pass transistors or parallel pairs of nMOS and pMOS called transmission gates.Line decoders are fundamental circuits, widely used in the peripheral circuitry of memory arrays (e.g., SRAM). This brief develops a mixed-logic methodology for their implementation, opting for improved performance compared to single-style design. The rest of this brief is organized as follows: Section II provides a brief overview of the examined decoder circuits, implemented with conventional CMOS logic. Section III introduces the new mixed-logic designs. Section IV conducts a comparative simulation study among the proposed and conventional decoders, with a detailed discussion on the derived results. Section V provides the summary and final conclusions of the work presented.

TABLE I
TRUTH TABLE OF THE 2–4 DECODER

A	B	D_0	D_1	D_2	D_3
0	0	**1**	0	0	0
0	1	0	**1**	0	0
1	0	0	0	**1**	0
1	1	0	0	0	**1**

TABLE II
TRUTH TABLE OF THE INVERTING 2–4 DECODER

A	B	I_0	I_1	I_2	I_3
0	0	**0**	1	1	1
0	1	1	**0**	1	1
1	0	1	1	**0**	1
1	1	1	1	1	**0**

2. Overview of line decoder circuits

In digital systems, discrete quantities of informationare represented by binary codes. An n-bit binary code can represent up to 2n distinct elements of coded data. A decoder is a combinationalcircuit that converts binary information fromn input lines to a maximum of 2n unique output lines or fewer if the n-bit coded information has unused combinations. The circuits examined. Here are n-to-mline decoders, which generate the m = 2n minterms of n input variables.

A. 2–4 Line Decoder

A 2–4 line decoder generates the 4 minterms D0−3 of 2 input variables A and B. Its logic operation is summarized in Table I. Depending on the input combination, one of the 4outputs selected and set to 1, while the others are set to 0. An inverting 2–4 decoder generates the complementary minterms I0−3, thus the selected output is set to 0 and the rest are set to as shown in Table II. In conventional CMOS design, NAND and NOR gates are preferred to AND and OR, since they can be implemented with 4 transistors, as opposed to 6, therefore implementing logic functions with higher efficiency. A 2–4 decoder can be implemented with 2 inverters and 4 NOR gates Fig. 1(a),

whereas an inverting decoder requires 2 inverters and4 NAND gates Fig. 1(b), both yielding 20 transistors.

B. 4–16 Line Decoder with 2–4 Predecoders

A 4–16 line decoder generates the 16 minterms D0−15 of4 input variables A, B, C, and D, and an inverting 4–16 linedecoder generates the complementary minterms I0−15. Suchcircuits can be implemented using a predecoding technique,according to which blocks of n address bits can be predecodedinto 1-of-2n predecoded lines that serve as inputs to the finalstage decoder. Therefore, a 4–16 decoder can be implementedwith 2 2–4 inverting decoders and 16 2-input NOR gates [Fig. 2(a)], and an inverting one can be implemented with 22–4decoders and 16 2-input NAND gates [Fig. 2(b)]. In CMOSlogic, these designs require 8 inverters and 24 2-input gates,yielding a total of 104 transistors each.

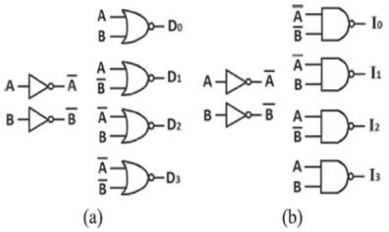

Fig. 1. 20-transistor 2–4 line decoders implemented with CMOS logic. (a) Non-inverting NOR-based decoder. (b) Inverting NAND-based decoder

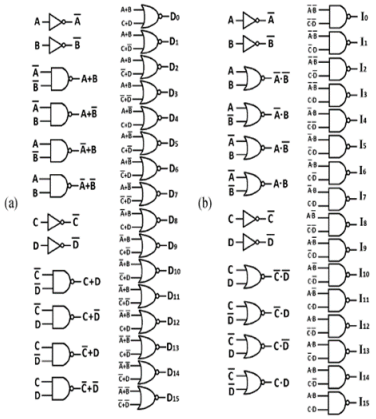

Fig. 2. 104-transistor 4–16 line decoders implemented with CMOS logic and predecoding. (a) Noninverting decoder implemented with two 2–inverting Pre decoders and a NOR-based postdecoder. (b) Inverting decoder implementedwith two 2–4 noninverting predecoders and a NAND-based post decoder.

3. New mixed-logic designs

Transmission gate logic (TGL) can efficiently implement AND/OR gates, thus it can be applied in line decoders. The2-input TGL AND/OR gates are shown in Fig. 3(a) and (b), respectively. They are full-swinging, but not restoring for all input combinations. Regarding PTL, there are two main circuit styles: those that use nMOS-only pass transistor circuits, like CPL, and those that use both nMOS and pMOS pass transistors, like DPL and DVL. The style we consider in this work is DVL, which preserves the full swing operation of DPL with reduced transistor count. The 2-input DVL AND/OR gates are shown in Fig.

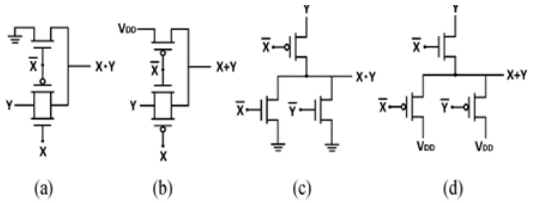

Fig. 3. Three-transistor AND/OR gates considered in this work. (a) TGL AND gate. (b) TGL OR gate. (c) DVL AND gate. (d) DVL OR gate

3(c) and (d), respectively. They are full swinging but non-restoring, as well. Assuming that complementary inputs are available, the TGL/DVL gates require only 3 transistors. Decoders are high fan-out circuits, where few inverters can be used by multiple gates, thus using TGL and DVL can result to reduced transistor count. An important common characteristic of these gates is their asymmetric nature, ie the fact that they do not have balanced input loads. As shown in Fig. 3, we labeled the 2gate inputs X and Y. In TGL gates, input X controls the gate terminals of all 3 transistors, while input Y propagates to the output node through the transmission gate. In DVL gates, input X controls 2 transistor gate terminals, while input Y controls 1gate terminal and propagates through a pass transistor to the output. We will refer to X and Y as the control signal and propagate signal of the gate, respectively. Using a complementary input as the propagate signal is not a good practice, since the inverter added to the propagation path increases delay significantly. Therefore, when implementing the inhibition (A' B) or implication (A' + B) function, it is more efficient to choose the inverted variable as control signal. When implementing the AND (AB) or OR (A + B) function, either choice is equally efficient. Finally, when implementing the NAND (A'+ B') or NOR (A'B') function, either choice results to a complementary propagate signal, perforce.

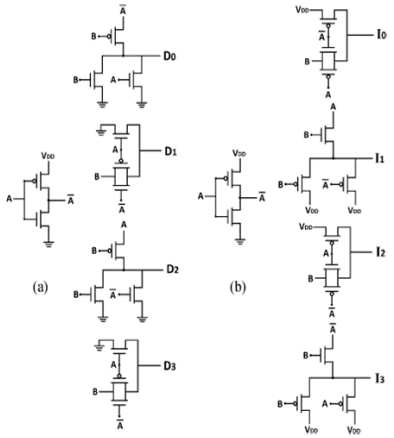

Fig. 4. New 14-transistor 2–4 line decoders. (a) 2–4LP. (b) 2–4LPI

A. 14-Transistor 2–4 Low-Power Topology

Designing a 2–4 line decoder with either TGL or DVL gates would require a total of 16 transistors (12 for AND/OR gates

and 4 for inverters). However, by mixing both AND gate types into the same topology and using proper signal arrangement ,it is possible to eliminate one of the two inverters, therefore reducing the total transistor count to 14.Let us assume that, out of the two inputs, namely, A and B, we aim to eliminate the B inverter from the circuit. The Do mine term (A'B') is implemented with a DVL gate, where A is used as the propagate signal. The D1 min term (AB') is implemented with a TGL gate, where B is used as the propagate signal. The D2 min term (A'B) is implemented with a DVL gate, where A is used as the propagate signal. Finally, TheD3 minterm (AB) is implemented with a TGL gate, where is used as the propagate signal. These particular choices completely avert the use of the complementary B signal.

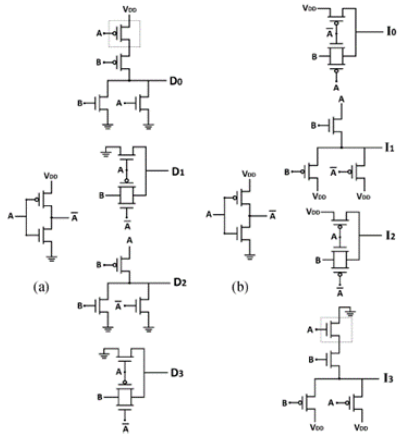

Fig. 5. New 15-transistor 2–4 line decoders. (a) 2–4HP. (b) 2–4HPI

Therefore, the B inverter can be eliminated from the circuit, resulting in a 14-transistor topology (9 nMOS and 5 pMOS).Following a similar procedure with OR gates, a 2–4 inverting line decoder can be implemented with 14 transistors (5 nMOSand 9 pMOS) as well: I0 and I2 are implemented with TGL (using as the propagate signal), and I1 and I3 are implemented with DVL (using A as the propagate signal). The B inverter can once again be elided. Inverter elimination reduces the transistor count, logical effort and overall switching activity of the circuits, thereby reducing power dissipation. The two new topologies are named"2–4LP" and "2–4LPI," where "LP" stands for "low power "and "I" for "inverting." Their schematics are shown in Fig. 4(a) and (b), respectively.

B. 15-Transistor 2–4 High-Performance Topology

The low-power topologies presented above have a drawback regarding worst case delay, which comes from the use of A as the propagate signal in the case of D0and I3. However, D0 and I3 can be efficiently implemented using static CMOS gates, without using complementary signals. Specifically, D0 can be implemented with a CMOS NOR gate and I3 with a CMOS NAND gate, adding one transistor to each topology. The new 15T designs present a significant improvement in delay while only slightly increasing power dissipation. They are named "2–4HP" (9 nMOS, 6 pMOS) and "2–4HPI" (6 nMOS, 9 pMOS), where "HP" stands for "high-performance" and "I "stands for "inverting." The 2–4HP and2–4HPI schematics are shown in Fig. 5(a) and (b), respectively.

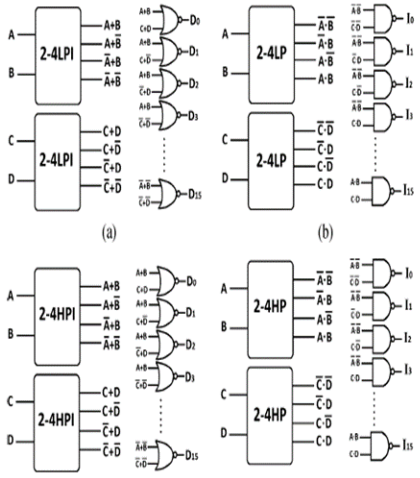

Fig. 6. New 4–16 line decoders. (a) 4–16LP. (b) 4–16LPI. (c) 4–16HP. (d) 4–16HPI

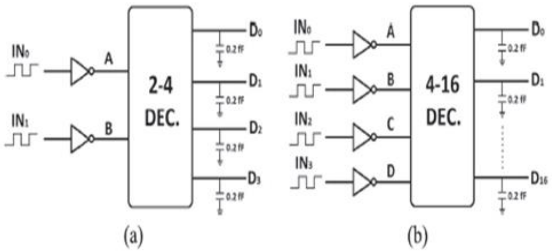

Fig. 7. Simulation setup regarding input/output loading conditions. (a) 2–4 decoders. (b) 4–16 decoders.

C. Integration in 4–16 Line Decoders

PTL can realize logic functions with fewer transistors and smaller logical effort than CMOS. However, cascading PTL circuits may cause degradation in performance due to the lack of driving capability. Therefore, a mixed-topology approach, i.e., alternating PTL and CMOS logic, can potentially deliver optimum results. We implemented four 4–16 decoders by using the four new2–4 as pre decoders in conjunction with CMOS NOR/NAND gates to produce the decoded outputs. The new topologies derived from this combination are the following: 4–16LP [Fig. 6(a)],which combines two 2–4LPI pre decoders with a NOR-based post decoder; 4–16HP [Fig. 6(b)], which combines two 2–4HPIpredecoderswith a NOR-based post decoder; 4–16LPI [Fig. 6(c)],which combines two 2–4LP predecoders with a NAND-based post decoder; and, finally, 4–16HPI [Fig. 6(d)], which combines two 2–4HP predecoders with a NAND-based post decoder. The "LP" topologies have a total of 92 transistors, while the "HP" ones have 94, as opposed to 104 with pure CMOS.

4. Simulation results

A. Proposed mixed

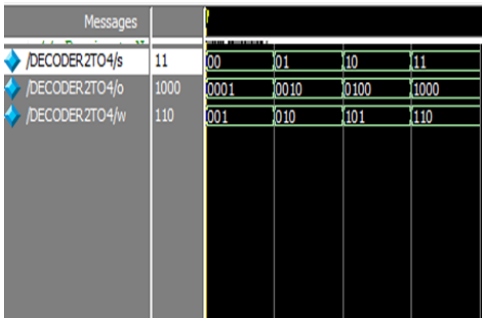

Fig. 8. 2–4LP line decoder

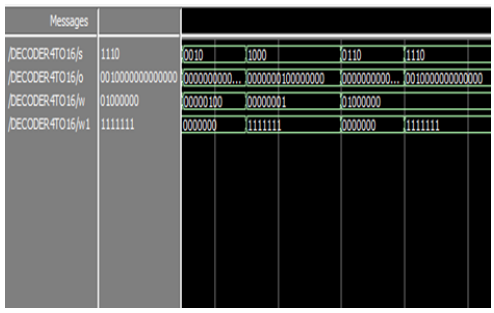

Fig. 9. 4-16LP line decoder

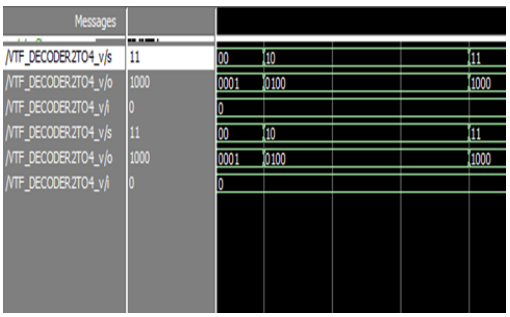

Fig. 10. 2–4HP line decoder

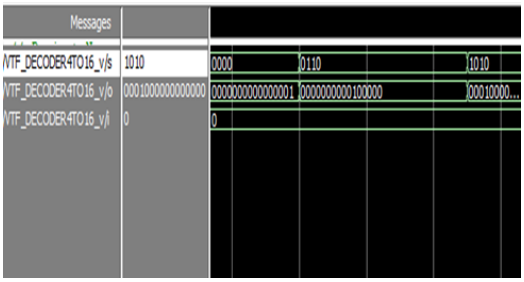

Fig. 11. 4-16HP line decoder

5. Conclusion

This brief has introduced an efficient mixed-logic design for decoder circuits, combining TGL, DVL and static CMOS. By using this methodology, we developed four new 2–4 line decoder topologies, namely 2–4LP, 2–4LPI, 2–4HP and 2–4HPI, which offer reduced transistor count and improved power delay performance in relation to conventional CMOS decoders. Furthermore, four new 4–16 line decoder topologies were presented, namely 4–16LP, 4–16LPI, 4–16HP and 4–16HPI, realized by using the mixed-logic 2-4 decoders as pre decoding circuits, combined with post decoders implemented in static CMOS to provide driving capability. A variety of comparative spice simulations was performed at32 nm, verifying, in most cases, a definite advantage in favor of the proposed designs. The 2–4LP and 4–16LPI topologies are mostly suitable for applications where area and power minimization is of primary concern. The 2–4LPI, 2–4HP, and2–4HPI, as well as the corresponding 4–16 topologies (4–16LP,4–16HPI, and 4–16HP), proved to be viable and all-around efficient designs; thus, they can effectively be used as building blocks in the design of larger decoders, multiplexers, and other combinational circuits of varying performance requirements.

References

[1] L. Jamal, M. Shamsujjoha, and H. M. HasanBabu, “Design of optimal reversible carry look-ahead adder with optimal garbage and quantum cost,” International Journal of Engineering and Technology, vol. 2, pp. 44–50, 2012.

[2] C. H. Bennett, “Logical reversibility of computation,” IBM J. Res. Dev., vol. 17, no. 6, pp. 525–532, Nov. 1973.

[3] M. Nielsen and I. Chuang, Quantum computation and quantum information. New York, NY, USA: Cambridge University Press, 2000.

[4] M. P. Frank, “The physical limits of computing,” Computing in Science and Engg., vol. 4, no. 3, pp. 16–26, May 2002.

[5] A. K. Biswas, M. M. Hasan, A. R. Chowdhury, and H. M. HasanBabu,“Efficient approaches for designing reversible binary coded decimal ladders,” Microelectron. J., vol. 39, no. 12, pp. 1693–1703, Dec. 2008.

[6] M. Perkowski, “Reversible computation for beginners,” 2000, lecture series, 2000, Portland state university.

[7] S. N. Mahammad and K. Veezhinathan, “Constructing online test table circuits using reversible logic,” *IEEE Transactions on Instrumentation and Measurement*, vol. 59, pp. 101–109, 2010.

[8] W. N. N. Hung, X. Song, G. Yang, J. Yang, and M. A. Perkowski,“Optimal synthesis of multiple output boolean functions using a set ofquantum gates by symbolic reachability analysis*,” IEEE Trans. on CAD of Integrated Circuits and Systems*, vol. 25, no. 9, pp. 1652–1663, 2006.