Logic Circuit Families (3)

Hanwool Jeong

hwjeong@kw.ac.kr

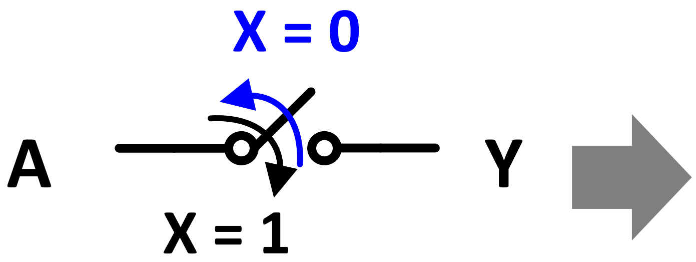
Contents

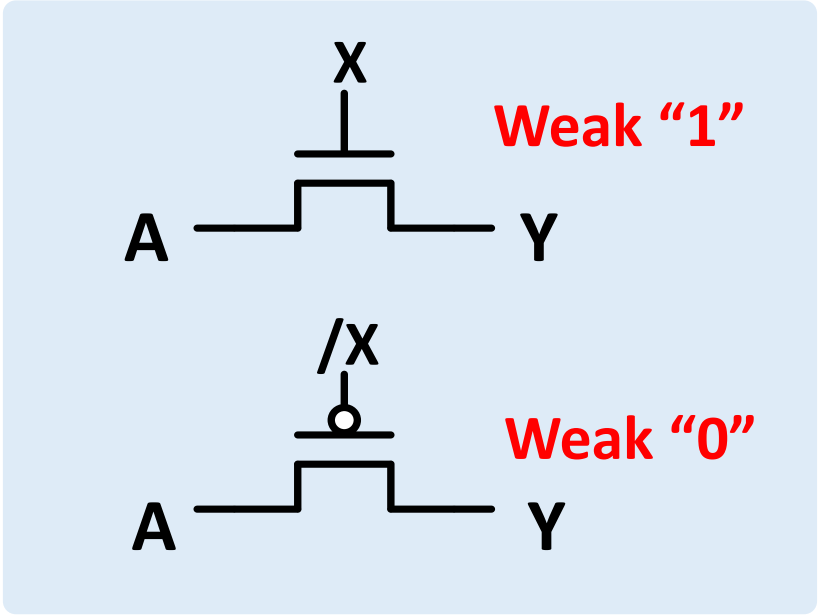
- Pseudo nMOS
- Cascode Voltage Switch Logic
- Dynamic Circuit and Domino Logic
- **Pass-Transistor Circuits**

Pass TR Logic Basic

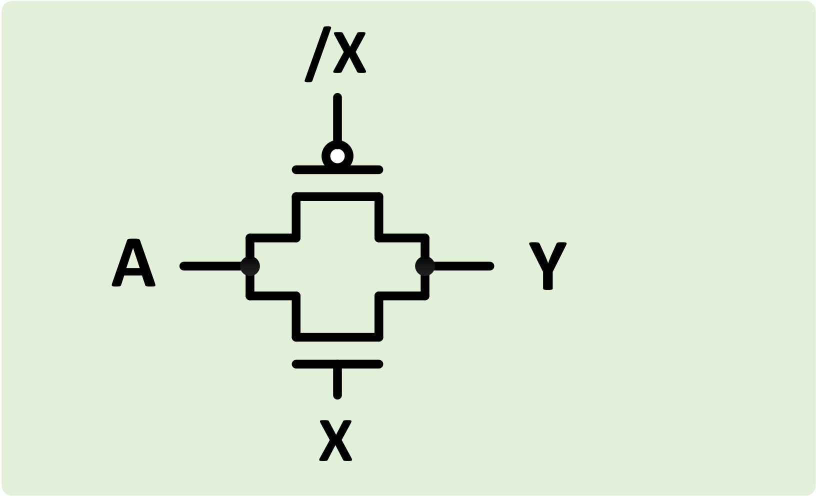
- ✓ Motivation for using pass TR circuits
- ✓ Applying to MUX design

Pass TR vs. Transmission Gate





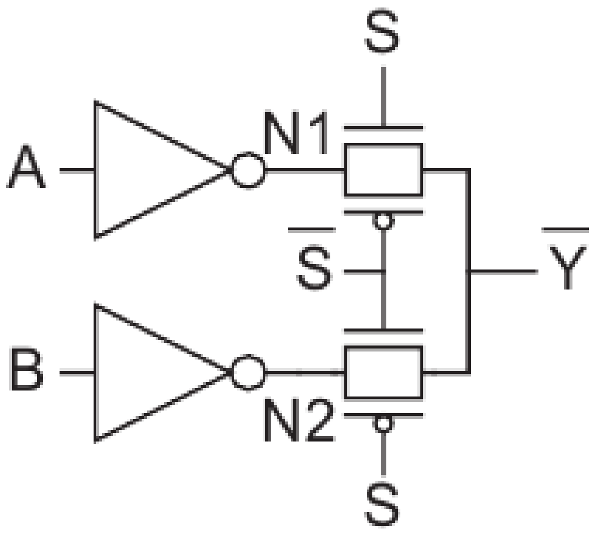
Pass Transistor

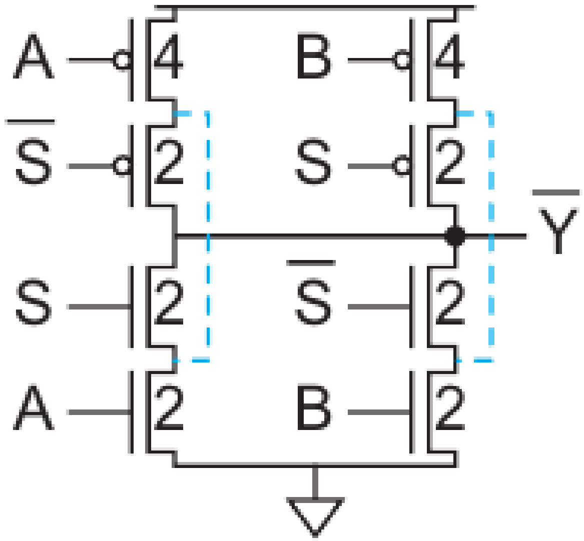


Transmission Gate

2:1 MUX Implementation

- Pass TR or TG can play a good role for achieving tri-state.

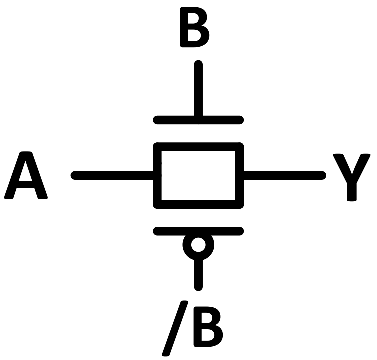




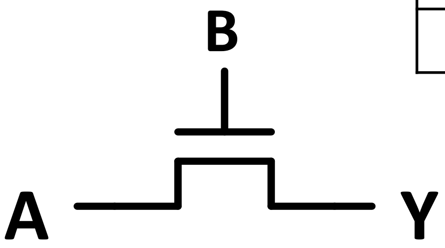
Key Difference of
Pass TR from Other Logic Families

- A single TR accepts 2 inputs to determine 1 output.

➔ **Chance** to reduce the number of TRs for **MUX** implementation.



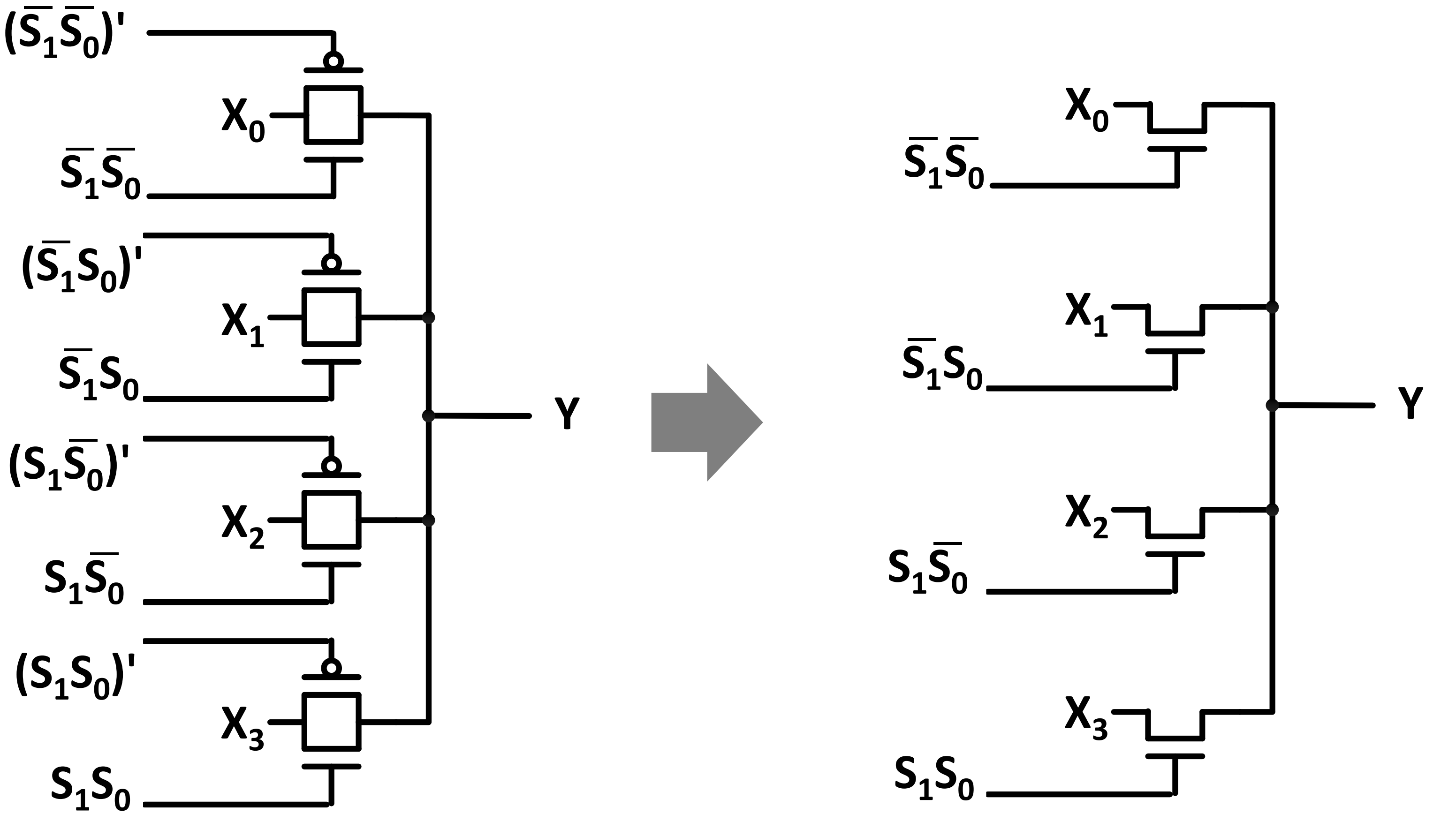
A	B	Y
0	0	Z
0	1	0
1	0	Z
1	1	1



A	B	Y
0	0	**Z**
0	1	**0**
1	0	**Z**
1	1	**1-**

Is there any way to restore 1-(V_{DD}-V_{th}) into full V_{DD}?

4:1 MUX Example

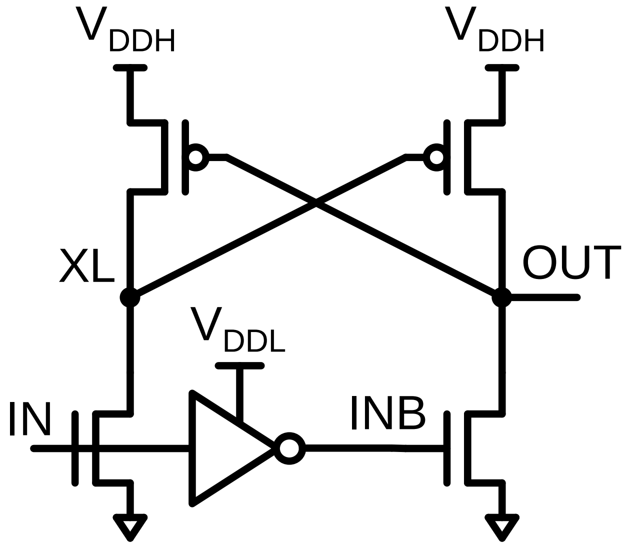


Revisit the Role of Cross-Coupled PFET

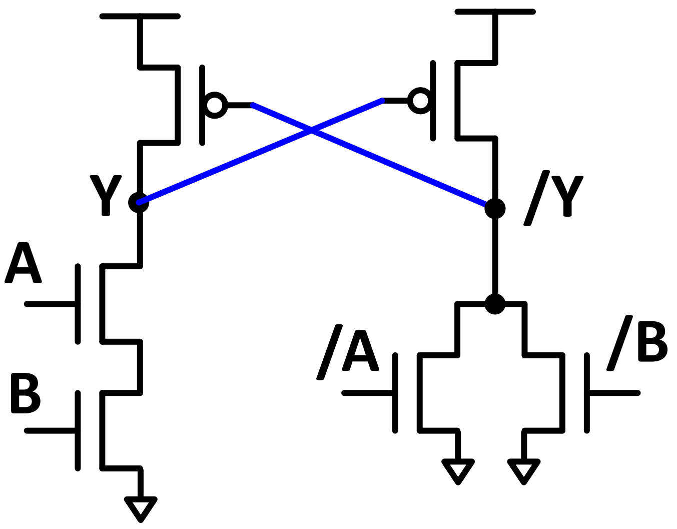
- You also saw this structure in the level shifter
- It operates in the positive feedback manner.

➔ **We can use cross-coupled pFETs for amplification/restoring**

➔ **This generalization thinking is so important!**



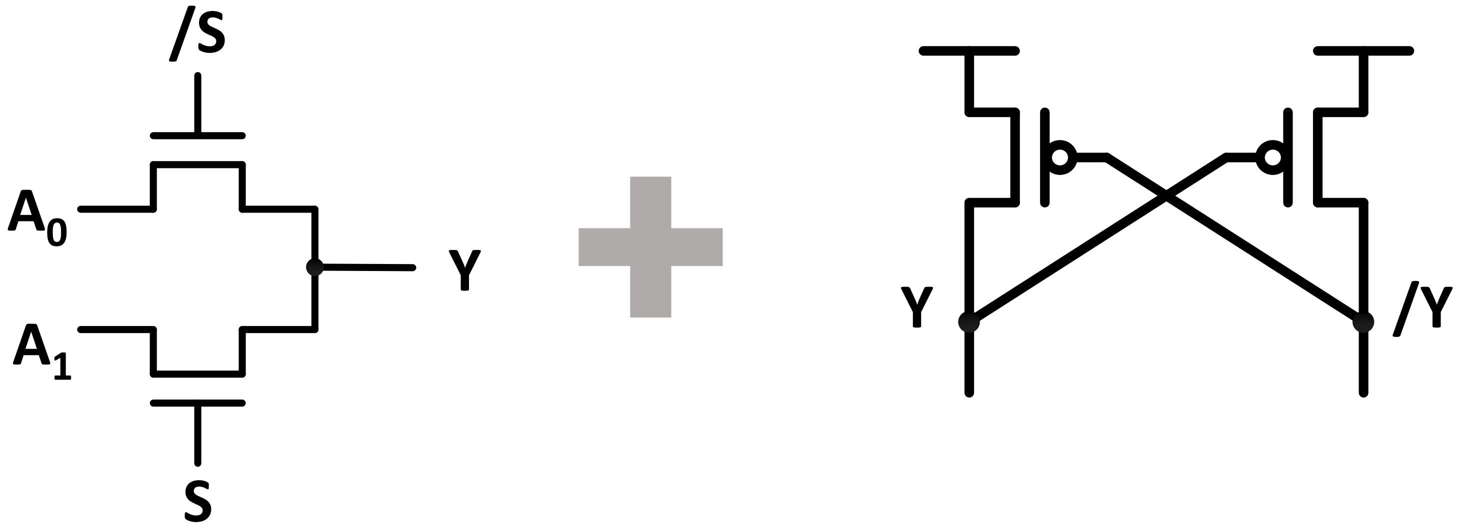
Level Shifter



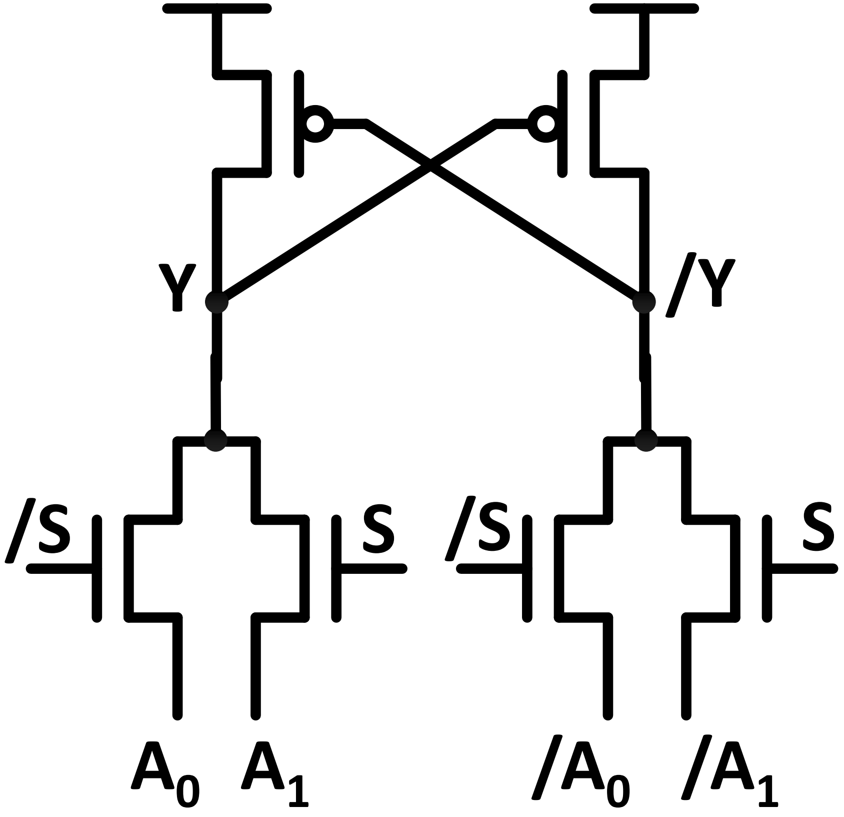
CVSL

Ideation Procedure

1) Replacing TG/tri-state inverter with pass TR in MUX **to reduce the number of transistors**
2) Employing the cross-coupled pFET **to restore weak “1” to strong “1”**



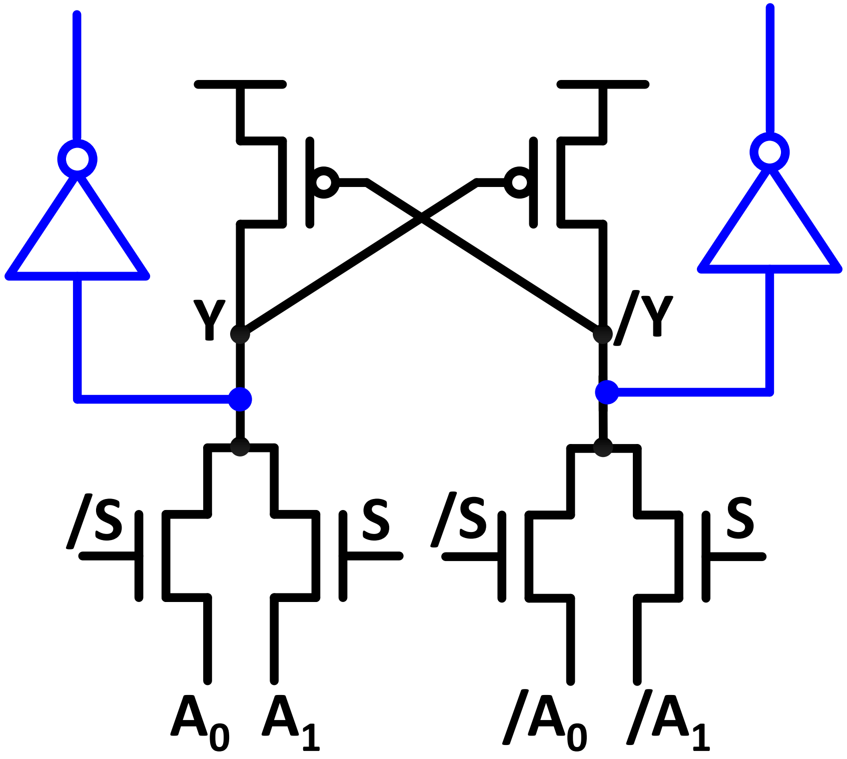
Design Procedure

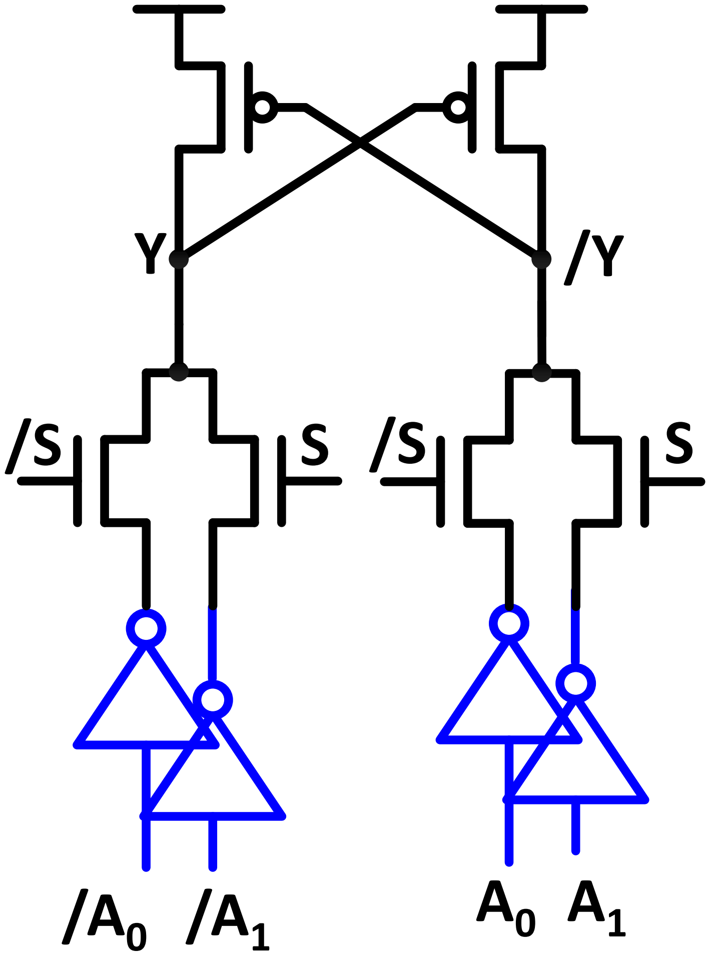


Complementary Pass Transistor Logic (CPL)

K. Yano, et al, "A 3.8-ns 16 × 16-b multiplier using complementary pass-transistor logic," JSSC, vol. 25, no. 2, Apr. 1990, pp. 388–395

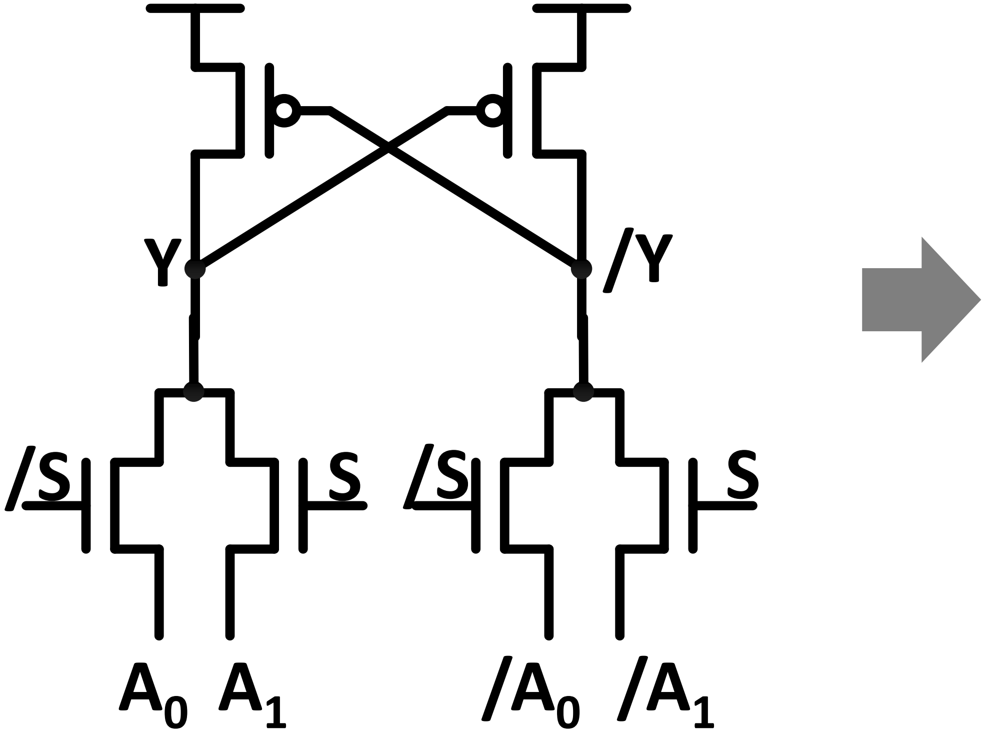
Mitigating Drivability Problem in CPL MUX





Can’t We Use PMOS Pass TR?

- Sure! Why not?



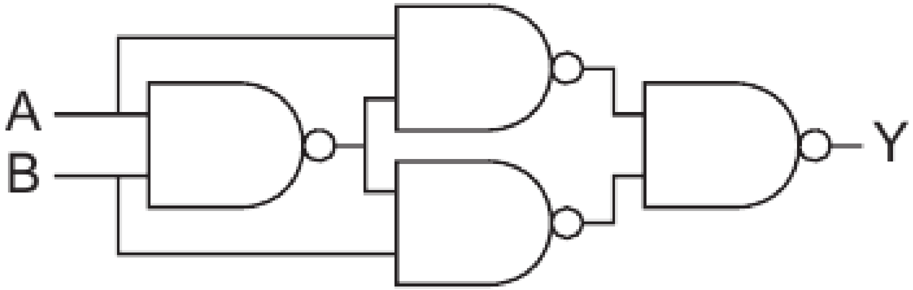
XOR-XNOR Design

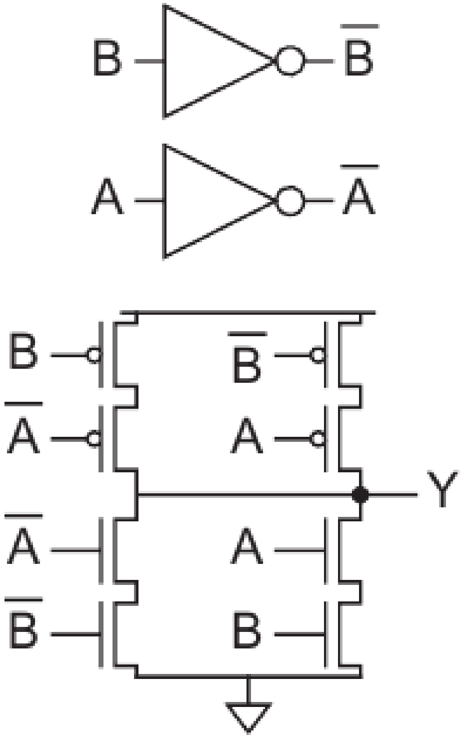
- ✓ Reducing the TR number thru Pass TR Logic
- ✓ Potentially decreases area, energy, and delay

Previous XOR2 Designs

A	B	Y
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**0**

- Y = $\overline{A}B + A\overline{B}$

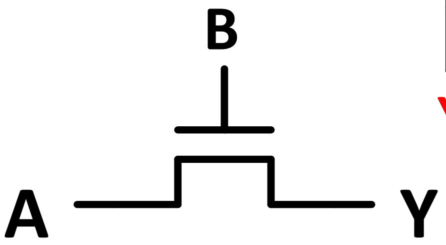




Key Difference of
Pass TR from Other Logic Families

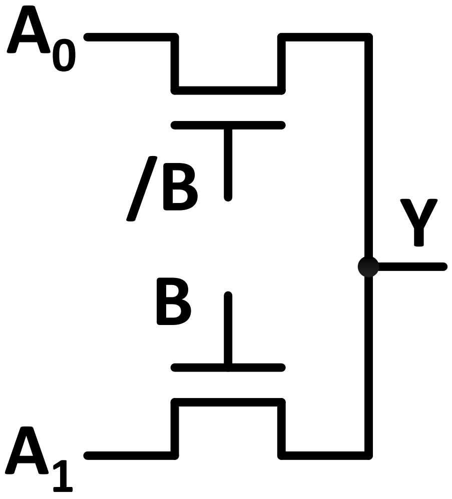
- A single TR accepts 2 inputs to determine 1 output.

➔ **Chance** to reduce the number of TRs for **logic** implementation.



A	B	Y
0	0	Z
0	1	0
1	0	Z
1	1	1-

Y = AB (when B=1)

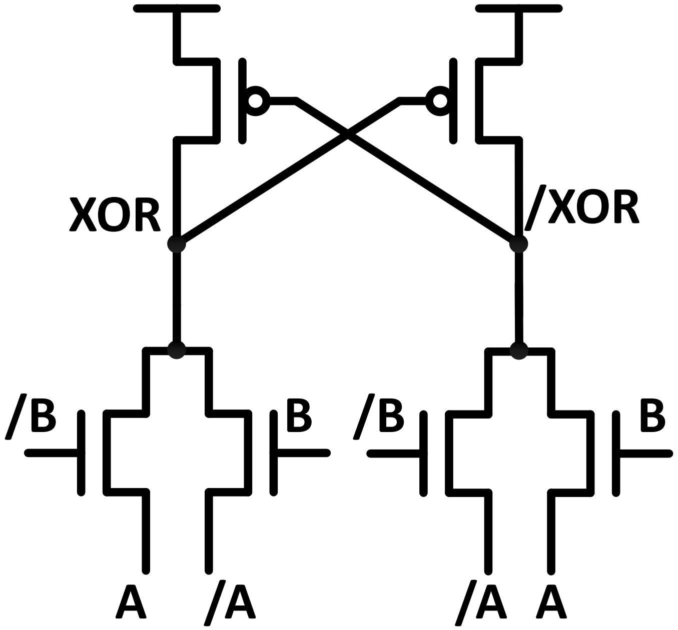


$$Y = A_0\bar{B} + A_1B$$

We can achieve XOR2!

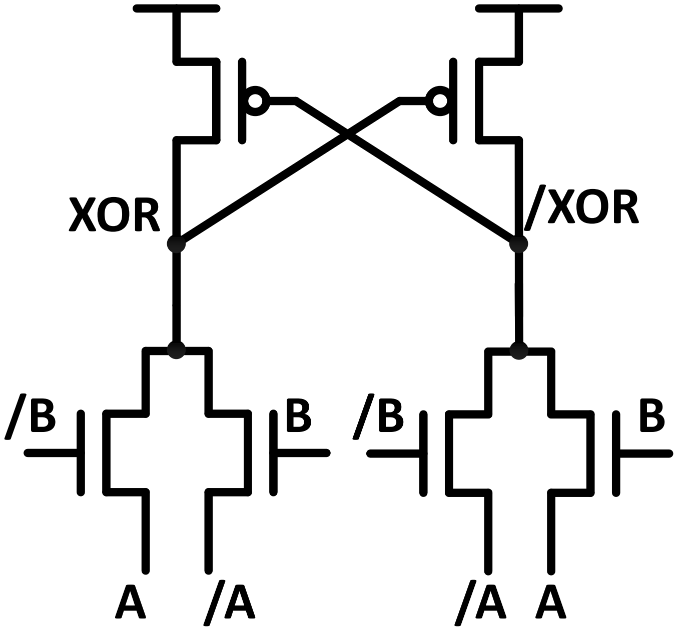
$Y = A\bar{B} + \bar{A}B$

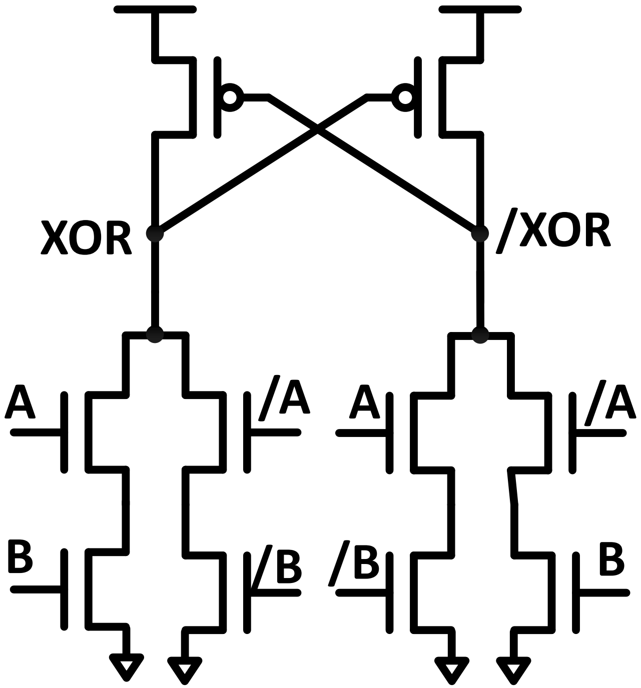
CPL based XOR Design



Comparing it With CVSL

- Could be faster in CPL (Why?)



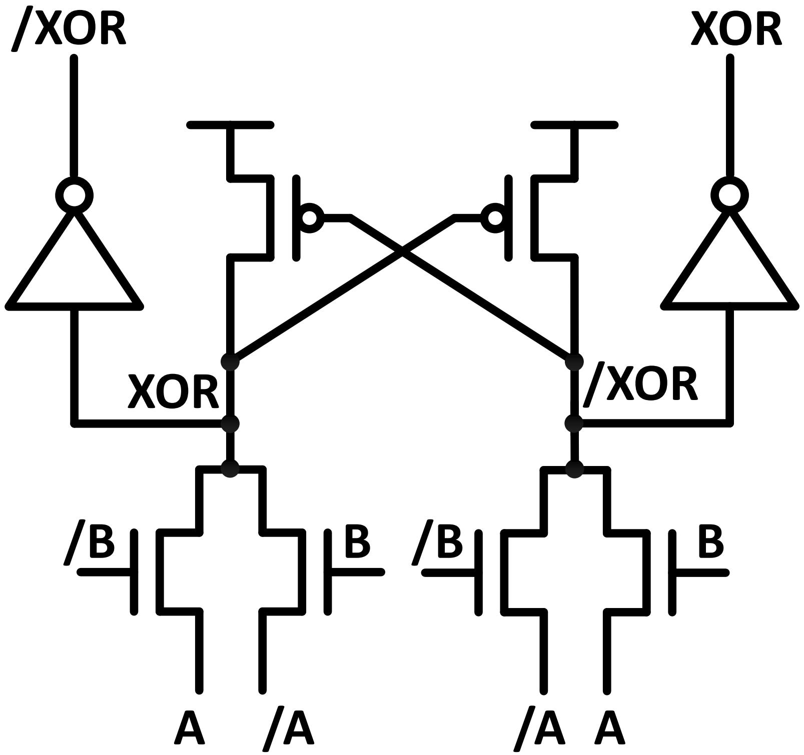


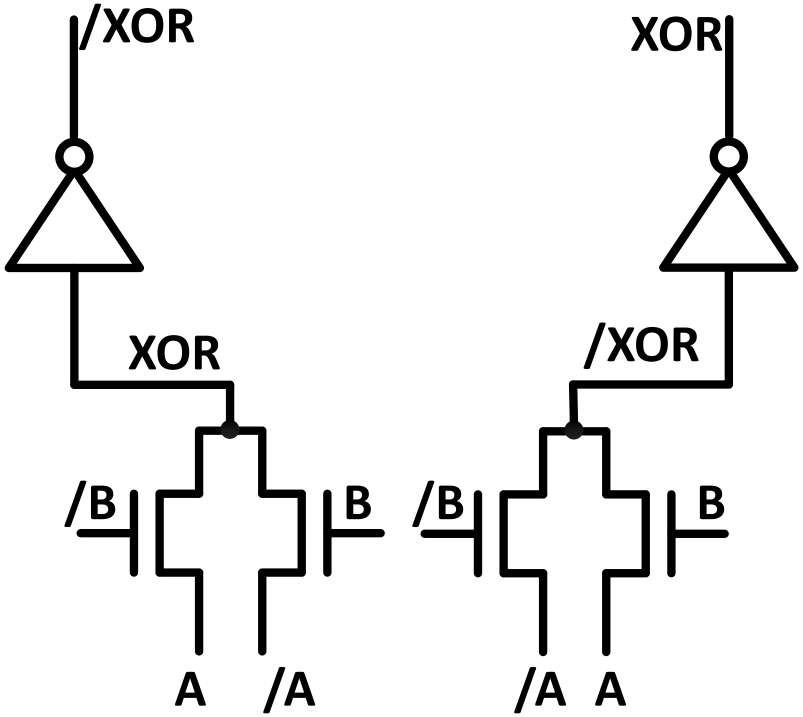
How About This?

- With the output driver (inverter), full VDD output can be achieved (Think of the restoring feature of the inverter!)

➔Yes.. but It incurs the short-circuit current at the output inverter.

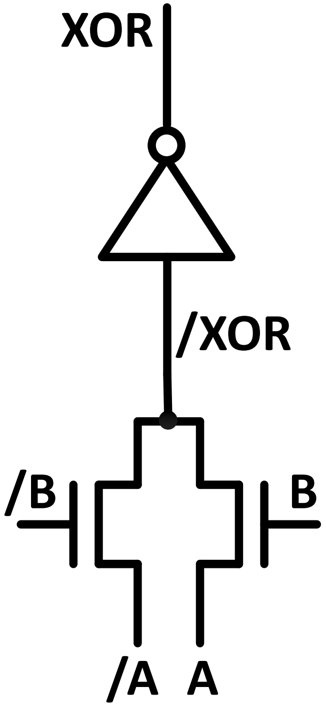
➔Can you improve it?

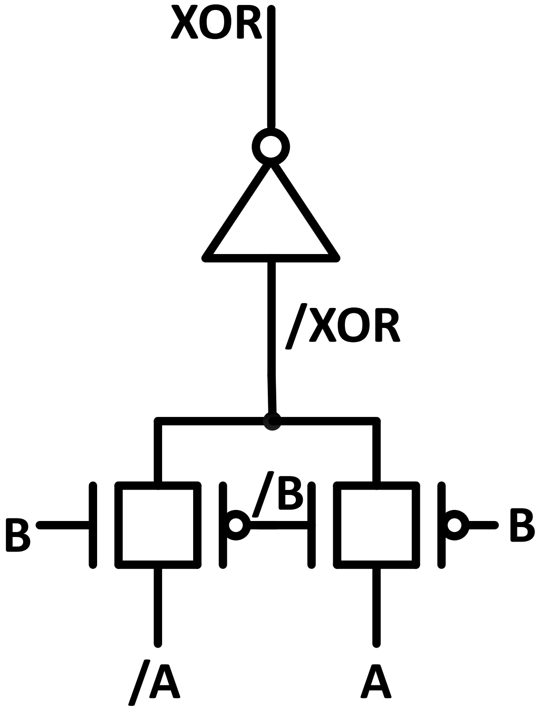




CMOSTG Design

- We can use TG instead of pass TR → CMOSTG
- **Is there any other way to make it full V_{DD}?**

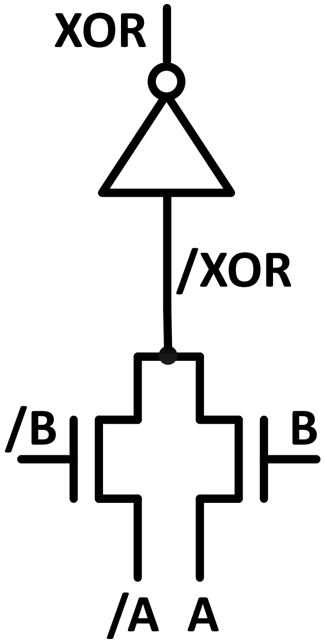


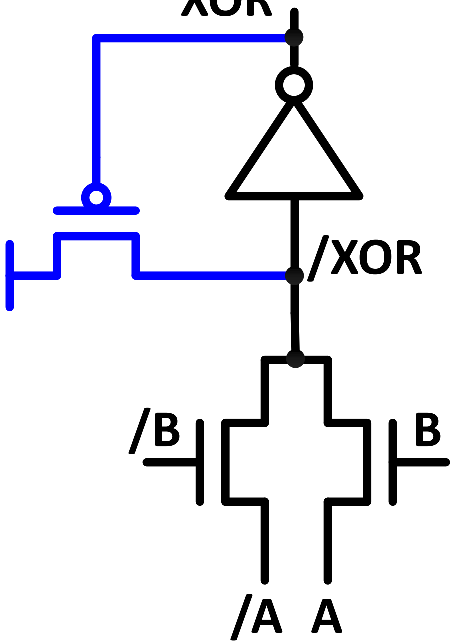


CMOSTG

Ideation

1) /XOR node should be pulled up ➔ We should utilize PMOS
2) When? Condition formed by A and B is meaningless ➔ Equivalent to CMOSTG
3) Again, when? utilizing XOR! When XOR = 0
4) Thus, we can add pull-up pMOS to /XOR that is controlled by XOR.





Lean Integration with Pass Transistors (LEAP)

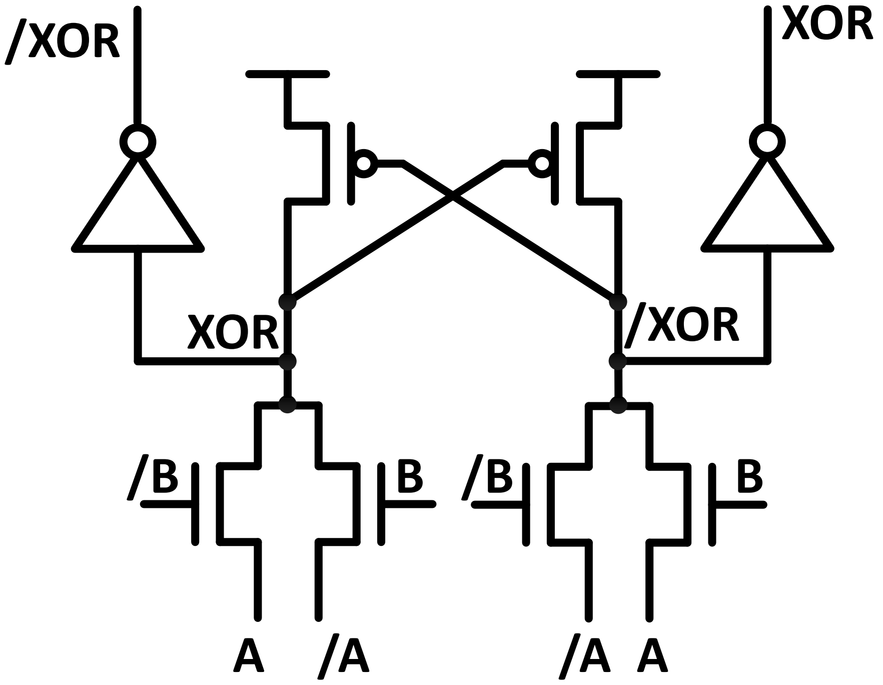
K. Yano, Y. Sasaki, K. Rikino, and K. Seki, "Top-down pass-transistor logic design," JSSC, vol. 31, no. 6, Jun. 1996, pp. 792–803.

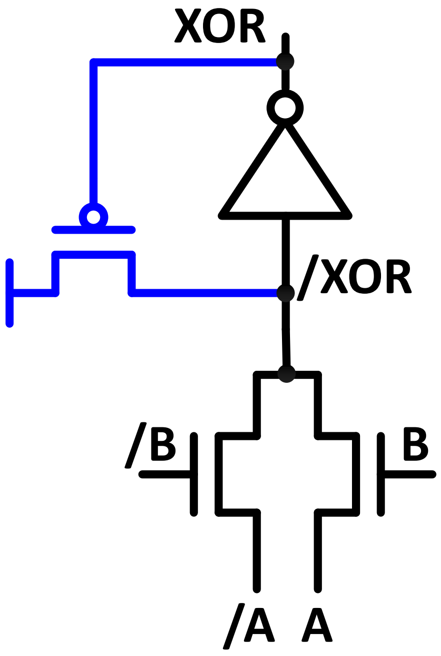
LEAP is Slow

1) Why? /XOR is pulled up solely by A or /A while, in the CPL, the opposite path aids to charge up /XOR.

2) In addition, complementary output is often useful.

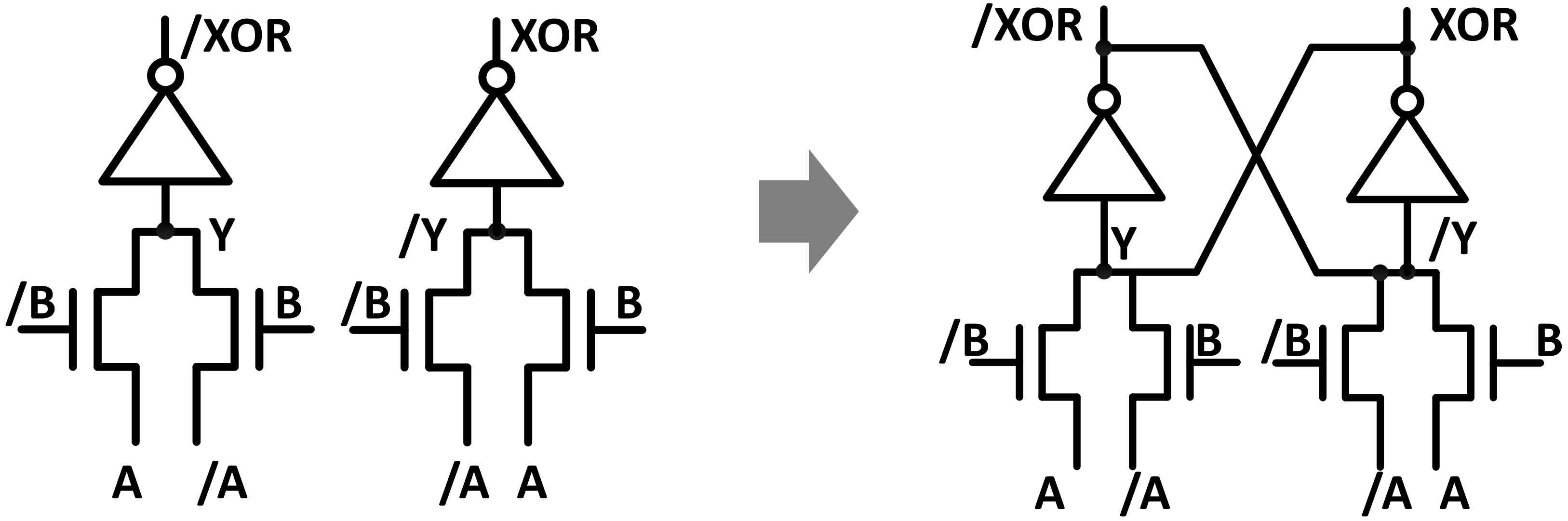
➔ Is there any way to reduce TR number of CPL without removing the operation complementary output?





Ideation

1) Cross-coupled pFET is removed.
2) Try to find which part can be utilized to replace the cross-coupled pFET
3) What is the role of the cross-coupled pFET?
   ➔ It is to charge up Y and /Y when required.
4) When? Y should be charged up when XOR is raised.

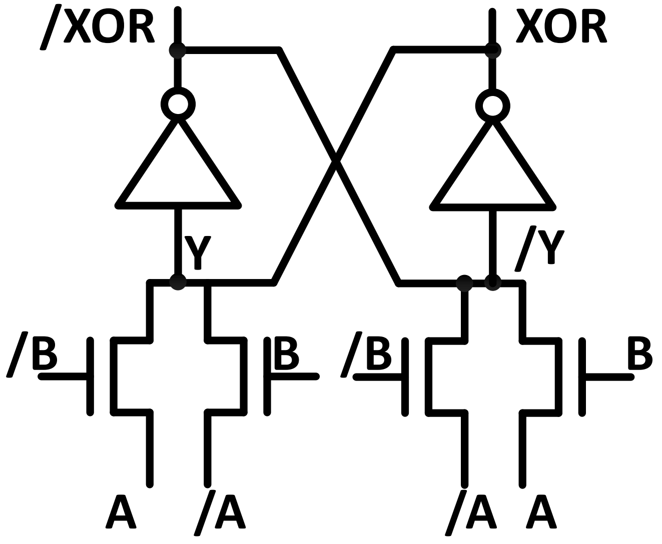


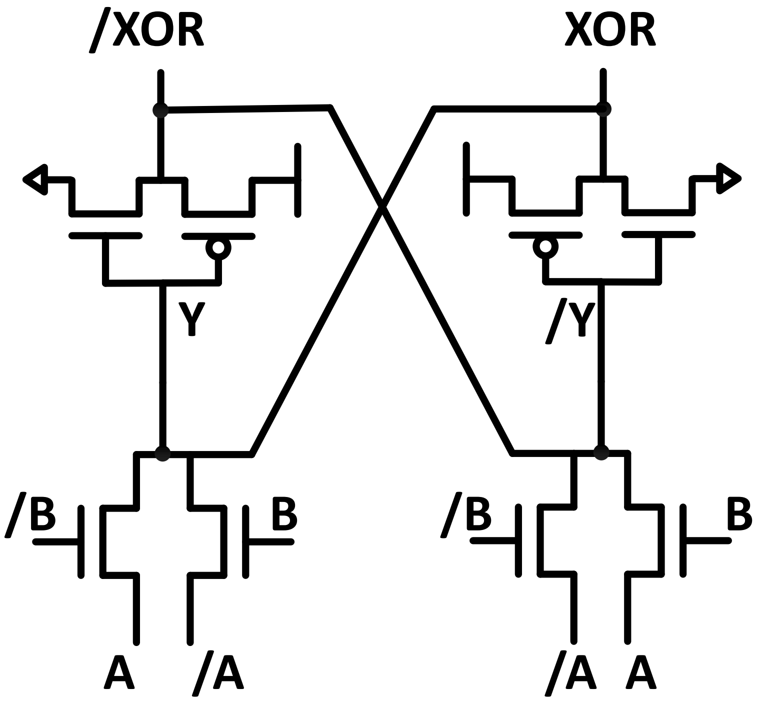
Swing-Restored Pass Transistor Logic (SRPL)

A. Parameswar, H. Hara, and T. Sakurai, "A swing restored pass-transistor logic-based multiply and accumulate circuit for multimedia applications," JSSC, vol. 31, no. 6, Jun. 1996, pp. 804–809.

Limitation of SRPL

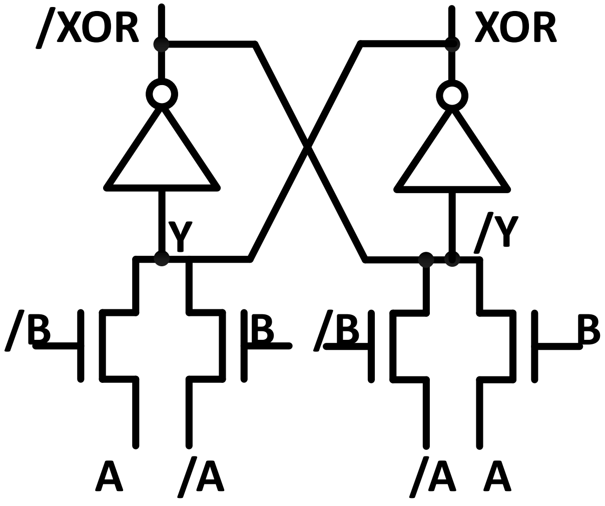
- Suppose that Y is supposed to be pulled down as A=0 and B=0, while (Y, /Y) initially is (1, 0)
- Contention occurs at the both sides, increasing the power and delay.

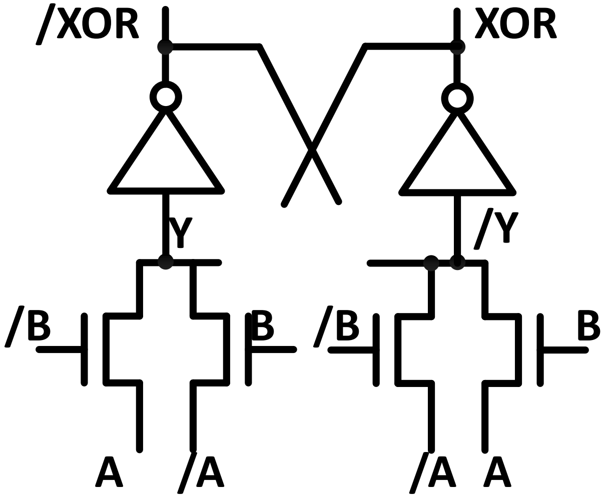


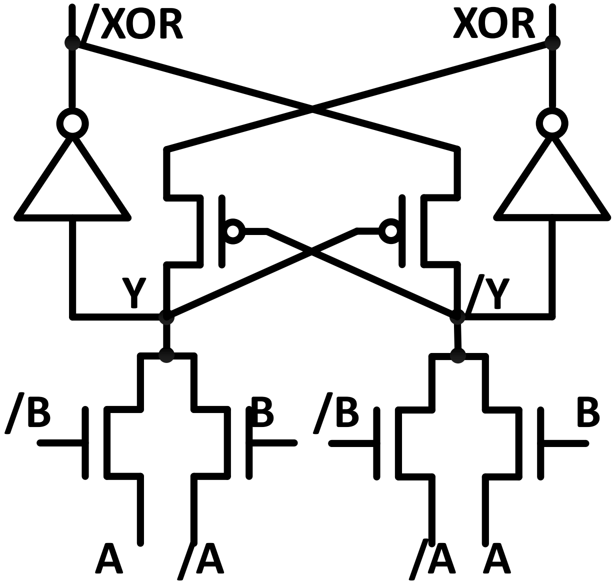


Ideation; Contention Mitigation

1) How can we make A or /A can easily pull down Y or /Y?
2) Not directly connect output of inverter to Y.
3) But we should keep the original goal being achieved, that is, pulling up Y to full VDD.
4) Is there any way pull-up VDD to full, while not disturbing Y pull-down?





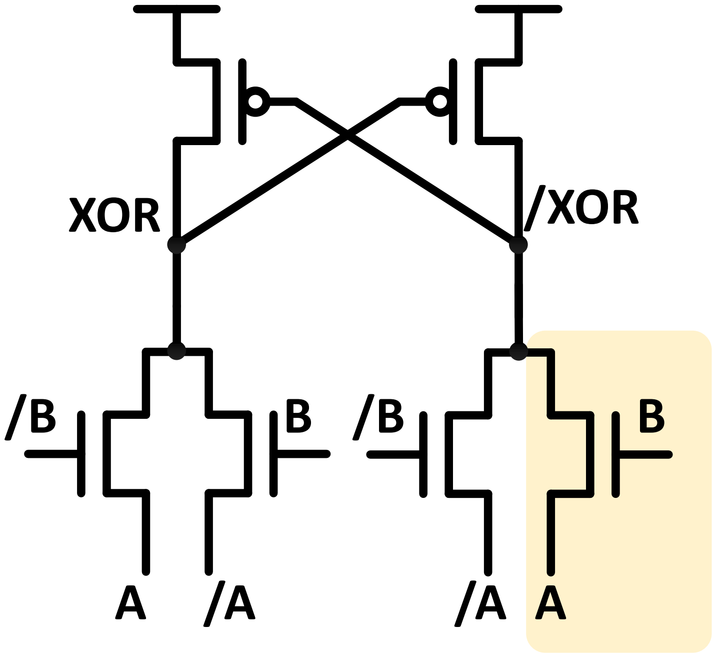


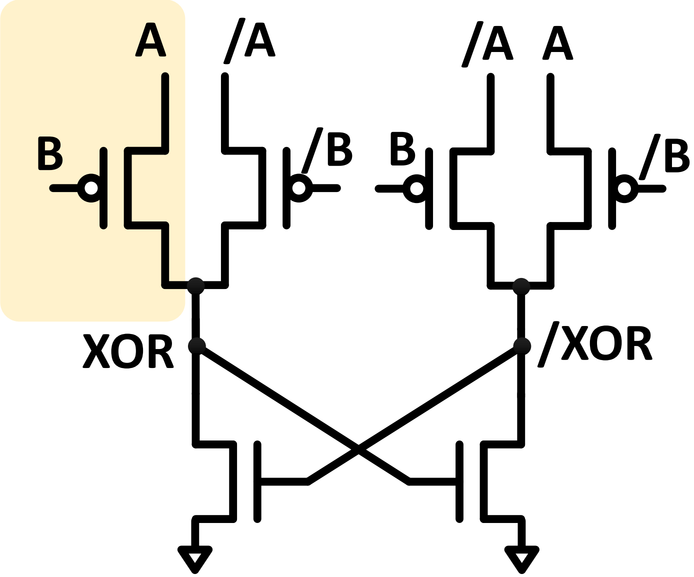
M. Song, G. Kang, S. Kim, and B. Kang, "Design methodology for high speed and low power digital circuits with energy economized pass-transistor logic (EEPL)," Proc. 22nd European SolidState Circuits Conf., 1996, pp. 120–123.

Energy Economized Pass Transistor Logic (EEPL)

PMOS Pass TR + Cross-Coupled NMOS

- **Can’t we make XOR/XNOR only with A and B? (i.e., without /A and /B)?**
- We can go on by relying on the facts that
  1) NMOS with A-B can generate XNOR and pMOS with A-B can generate XOR (highlighted below) → but output can be imperfect for some cases.
  2) XOR can be used to restore XNOR and vice versa.

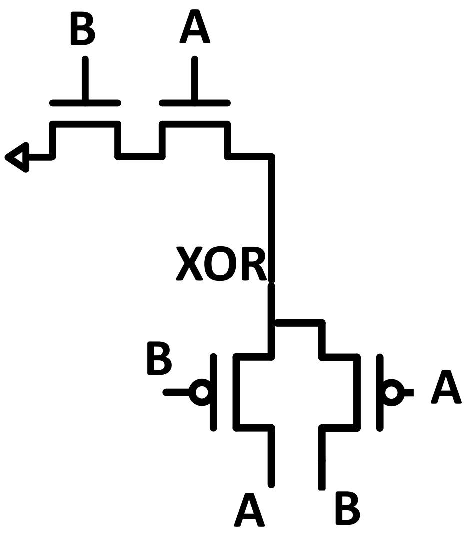


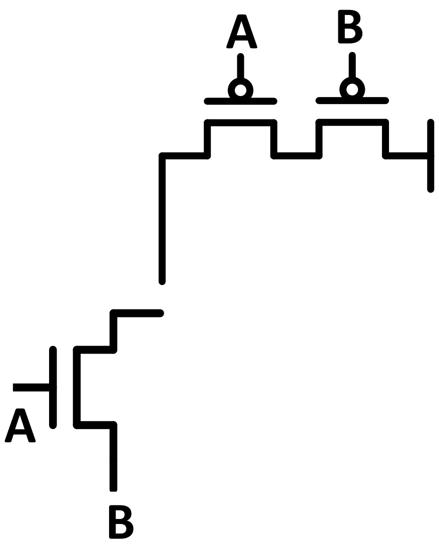


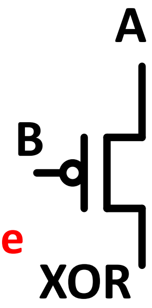
Ideation

- Constraint: Don’t use /A and /B

A	B	XOR	XNOR
0	0	**0**	**1**
0	1	**1**	**0**
1	0	**1**	**0**
1	1	**0**	**1**





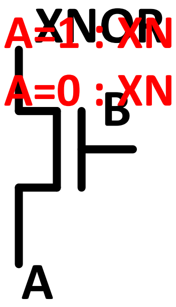


How about when B = 1?

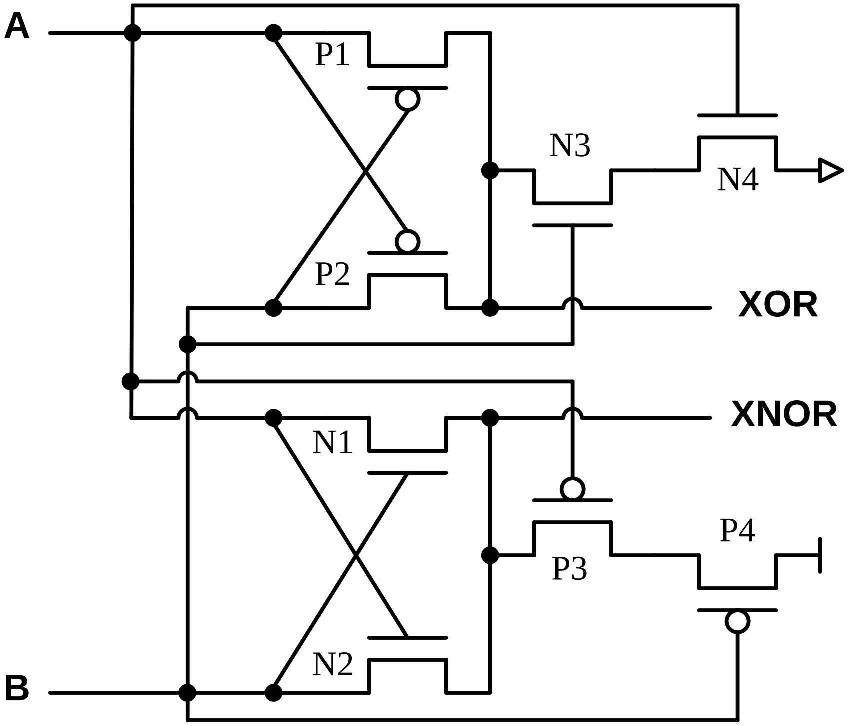
1) A=0 : XOR=1
2) A=1 : XOR=0

B =0 is problematic here

1) A=1 : XNOR = 0
2) A=0 : XNOR = 1

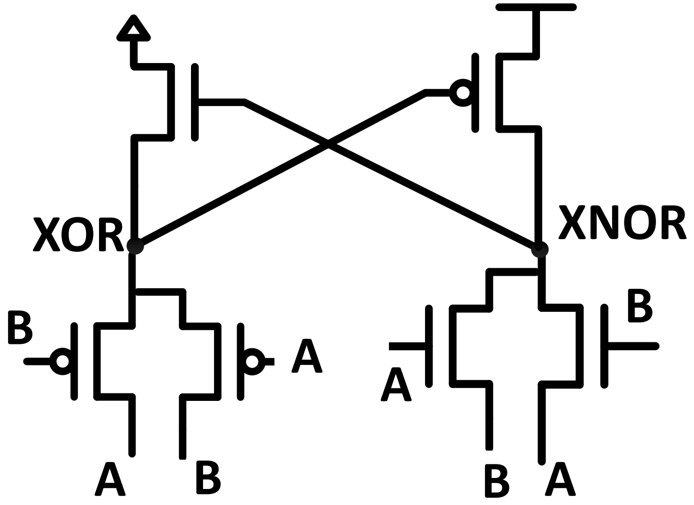


Resultant 8T XOR-XNOR



Jinaping Hu, International Journal of Advancements in Computing Technology, 2012

Ideation Again



A	B	XOR	XNOR
0	0	**0**	**1**
0	1	**1**	**0**
1	0	**1**	**0**
1	1	**0**	**1**

B = 1 : problem

1) A=0 : XOR=1
2) **A=1 : XOR=0**

➔ XNOR can restore this

B = 0 : problem

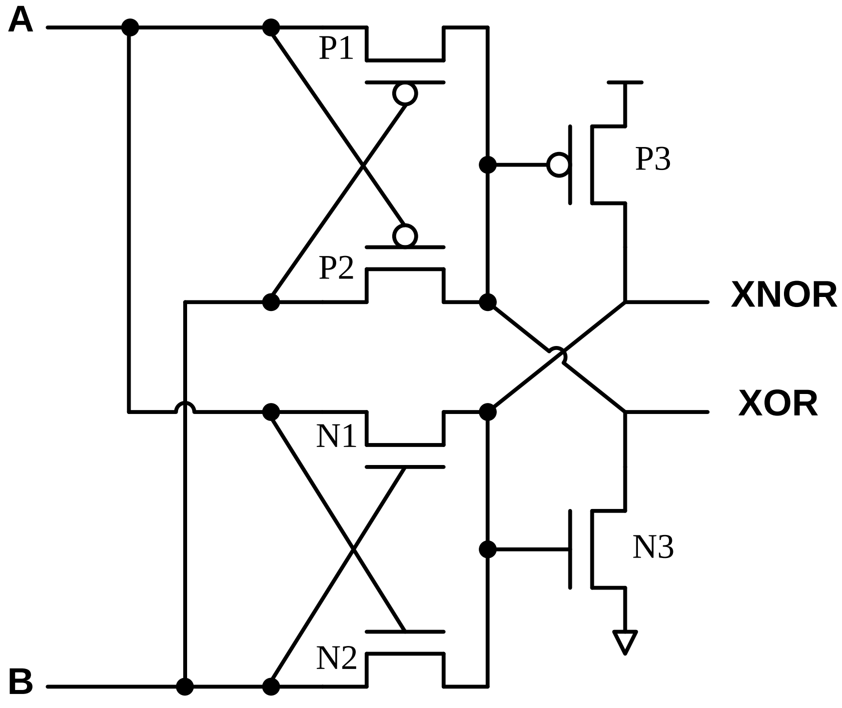
1) A=1 : XNOR=0
2) **A=0 : XNOR=1**

➔ XOR can restore this

Resultant 6T XOR-XNOR

- Can you find the limitation?

➔ Due to V_{th} drop of gate level of P3 and N3, restoring is very slow

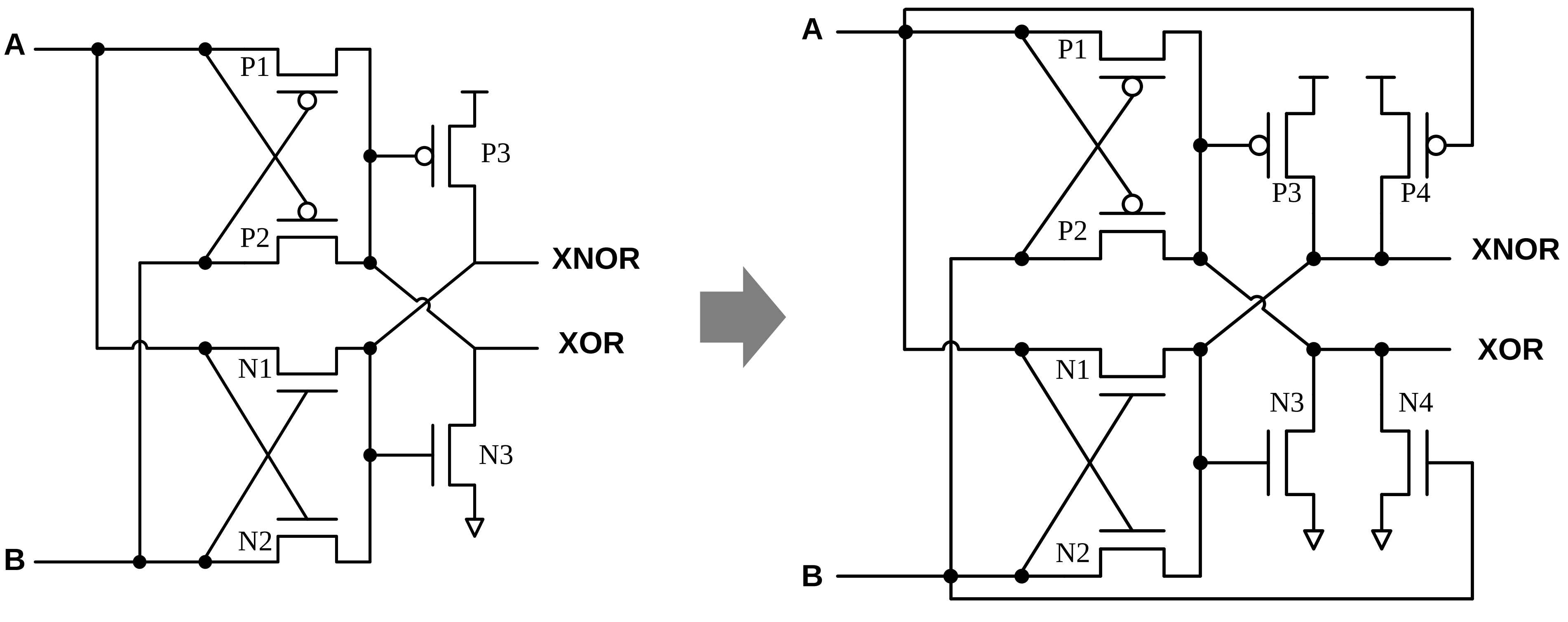


D.Radhakrishnan, IEEE Proc.-Circiiits Devices Syst., 2001

Accelerating Restoring

A	B	XOR	XNOR
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1

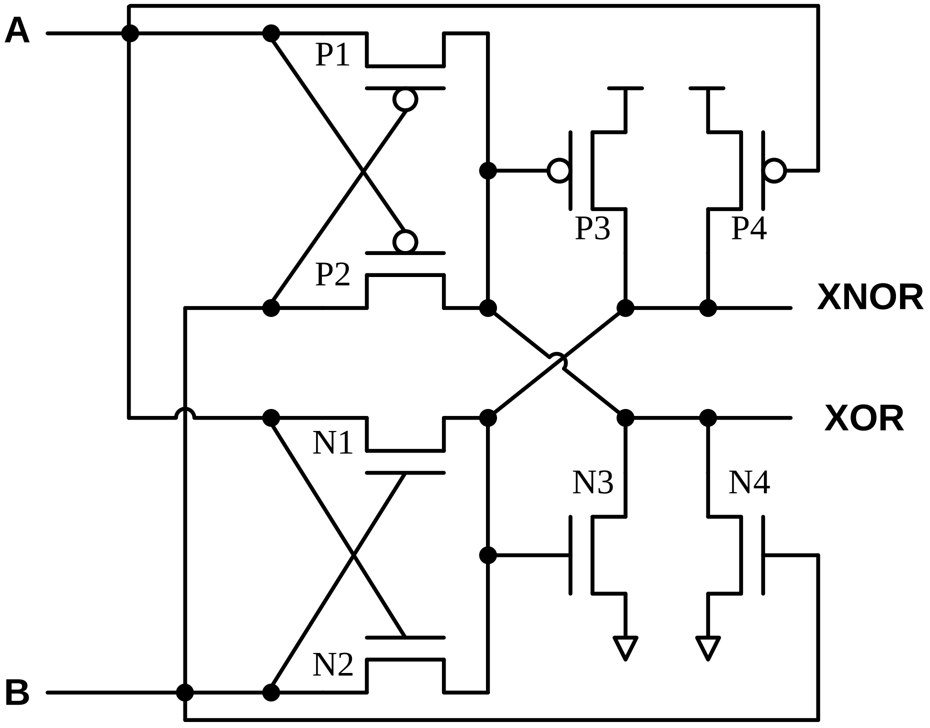
- Let’s focus on XOR.



Resultant 8T XOR-XNOR

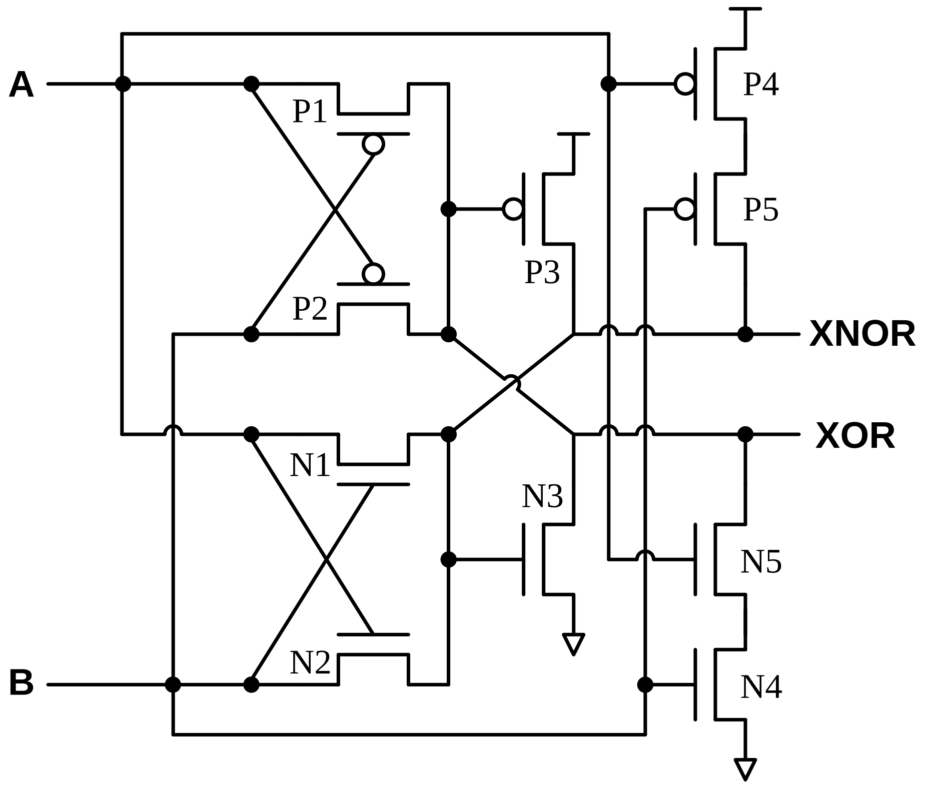
- Can you find the limitation?

➔ How can you solve the contention issue?



M. Elgamel, Great Lakes symposium on VLSI, 2003

Resultant 10T XOR-XNOR



M. Zhang, ISCAS2003

Summary

- Pass TR can be used to implement logic to reduce the number of transistor, leading to reduction in area/energy/delay.
- Various methods for designing XOR2 and XNOR2 are described to resolve the limitation of the previous structure in terms of area/energy/delay.
- Step by step ideation procedure is highly important.