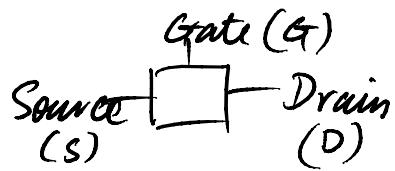
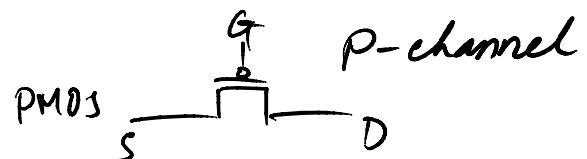
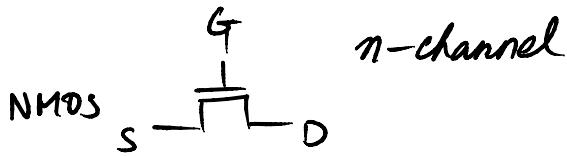


B8 MOSFET Transistors in Logic gates metal oxide semiconductor field effect transistor



Gate voltage controls on or off modes of the switch

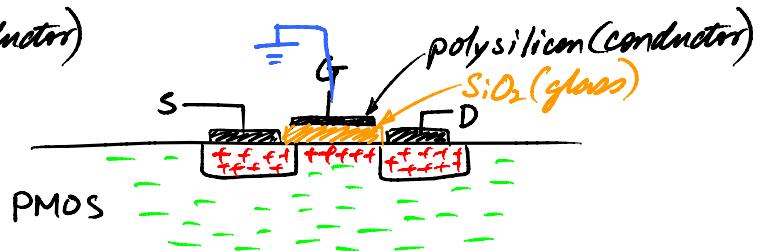
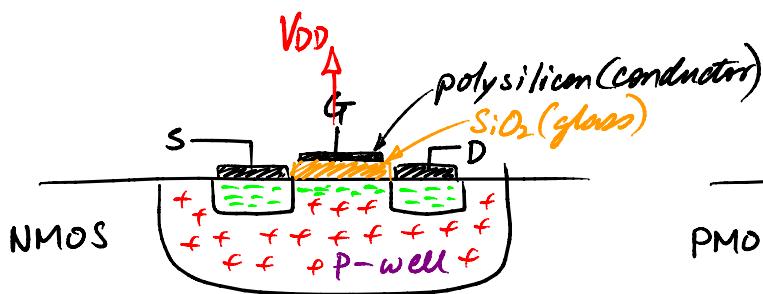


$G = 3V$ (logic 1) S — D
SW closed
(connection)

$G = 0V$ (logic 0) S — D
SW closed
(connection)

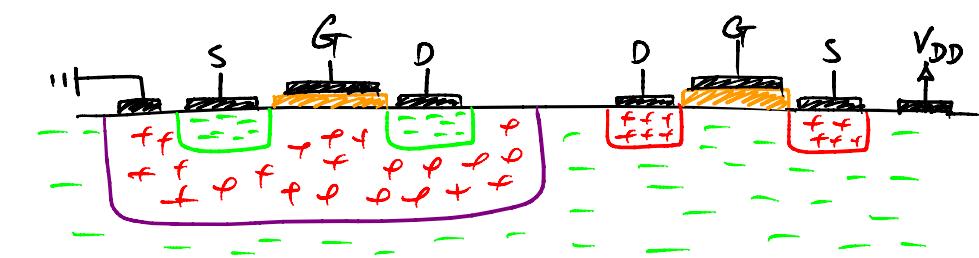
$G = 0V$ (logic 0) S — D
SW opened
(no connection)

$G = 3V$ (logic 1) S — D
SW opened
(no connection)

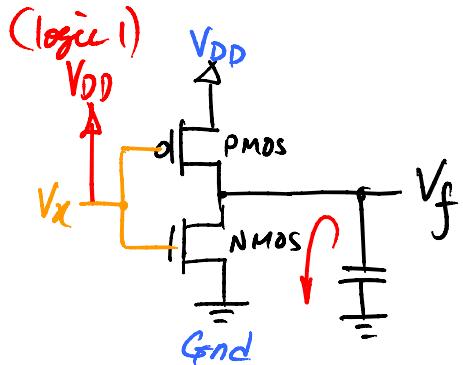
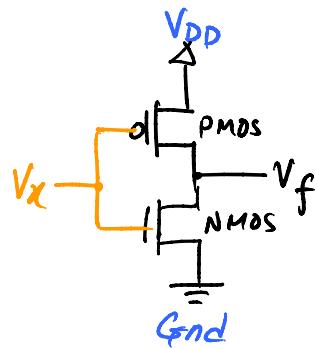
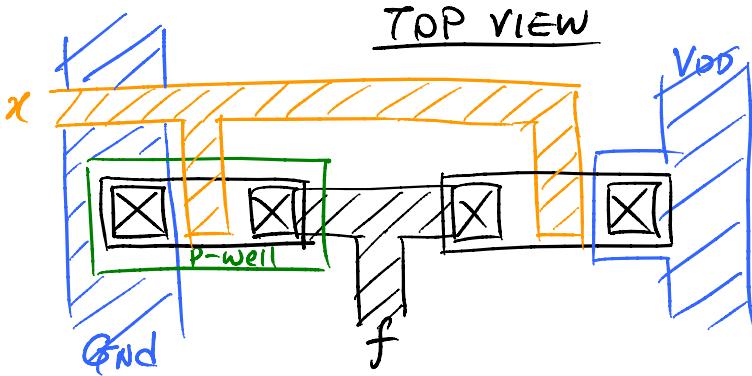


a channel is formed under the gate for electricity to flow when
 $V_G = V_{DD}$

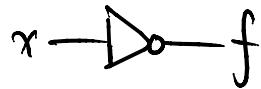
by setting $V_G = 0V$, electrons are pushed away while +ve charges are attracted under the gate (connecting source + drain)



CMOS - complementary MOS

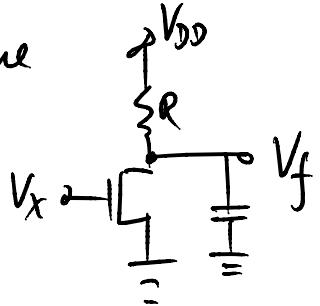


nmos turned on while
pmos is off, charges on the
cap. drains off through NMOS.
resulting in $V_f = 0V$, (logic 0)



| V_x | T_p | T_n | V_f |
|-------|-------|-------|-------|
| 0 | on | off | 1 |
| 1 | off | on | 0 |

alternative



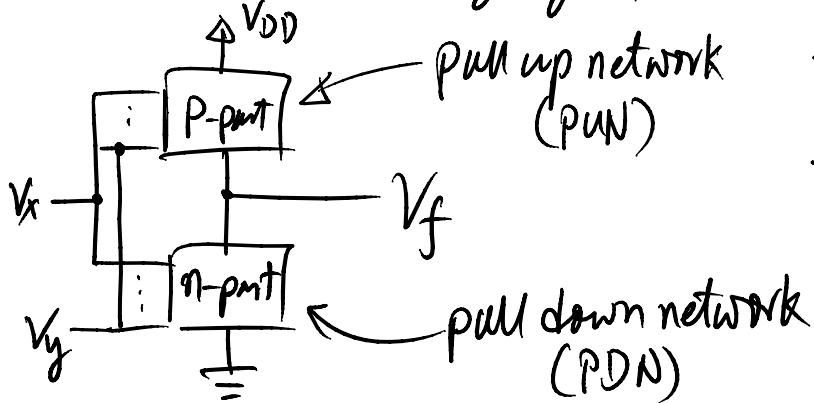
$V_x = 0V$, NMOS off,

V_f pulled up to V_{DD} through PMOS.

i.e. $R = 10k\Omega$, $V_{DD} = 2.5V$, then $I = \frac{2.5V}{10k\Omega} = 0.25mA$

millions of transistors $\Rightarrow 250mA$
(huge energy loss)

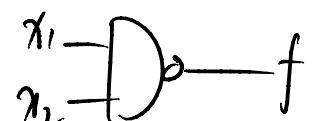
use CMOS to build logic gates/circuits



→ PMOS & NMOS always come in pairs

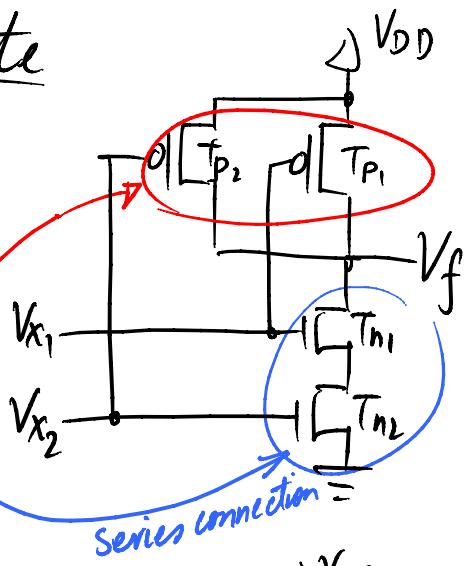
→ PUN on top and PDN at the bottom

CMOS NAND gate



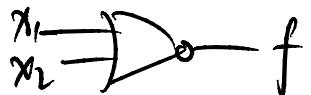
$$f = \overline{x_1 x_2}$$

$$= \overline{x_1} + \overline{x_2}$$



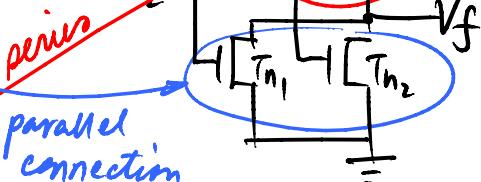
| | V_{x_1} | V_{x_2} | $T_{p_1}, T_{p_2}, T_{n_1}, T_{n_2}$ | V_f |
|--------------|-----------|-----------|--------------------------------------|-------|
| (\equiv) | 0 | 0 | on on off off | 1 |
| | 0 | 1 | on off off on | 1 |
| (V_{DD}) | 1 | 0 | off on on off | 1 |
| | 1 | 1 | off off on on | 0 |

CMOS NOR gate



$$f = \overline{x_1 + x_2}$$

$$= \overline{x_1} \cdot \overline{x_2}$$



| | V_{x_1} | V_{x_2} | $T_{p_1}, T_{p_2}, T_{n_1}, T_{n_2}$ | V_f |
|--|-----------|-----------|--------------------------------------|-------|
| | 0 | 0 | on on off off | 1 |
| | 0 | 1 | on off off on | 0 |
| | 1 | 0 | off on on off | 0 |
| | 1 | 1 | off off on on | 0 |

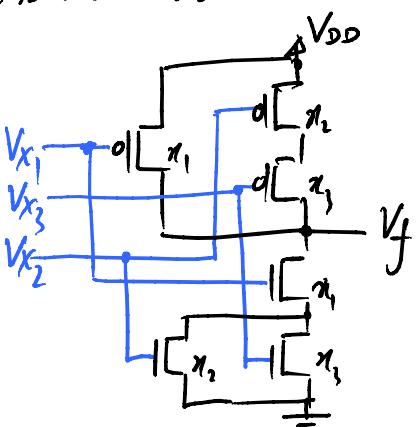
\Rightarrow logic AND corresponds to series connections
 .. OR .. " parallel "

\Rightarrow logic function \rightarrow PDN

\Rightarrow if PDN has parallel connections, then PUN will have series connections and the reverse is also true.

example! $f = \overline{x_1} + \overline{x_2} \cdot \overline{x_3}$

Option 1 - draw PUN first
 - then PDN.



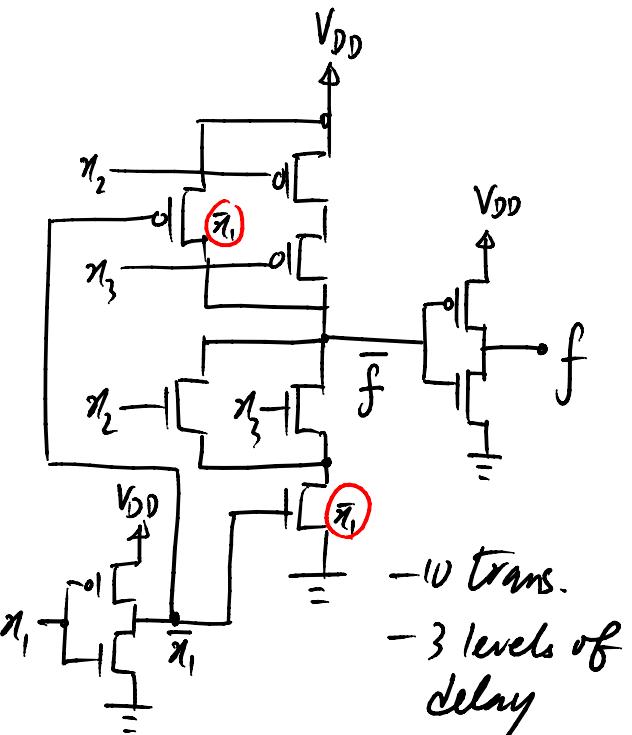
Option 2 - draw PDN first
- then PUN

$$\begin{aligned} f &= \bar{x}_1 + \bar{x}_2 \cdot \bar{x}_3 \\ &= \bar{x}_1 + \overline{x_2 + x_3} \\ &= \underline{\bar{x}_1 (\bar{x}_2 + \bar{x}_3)} \end{aligned}$$

→ PDN

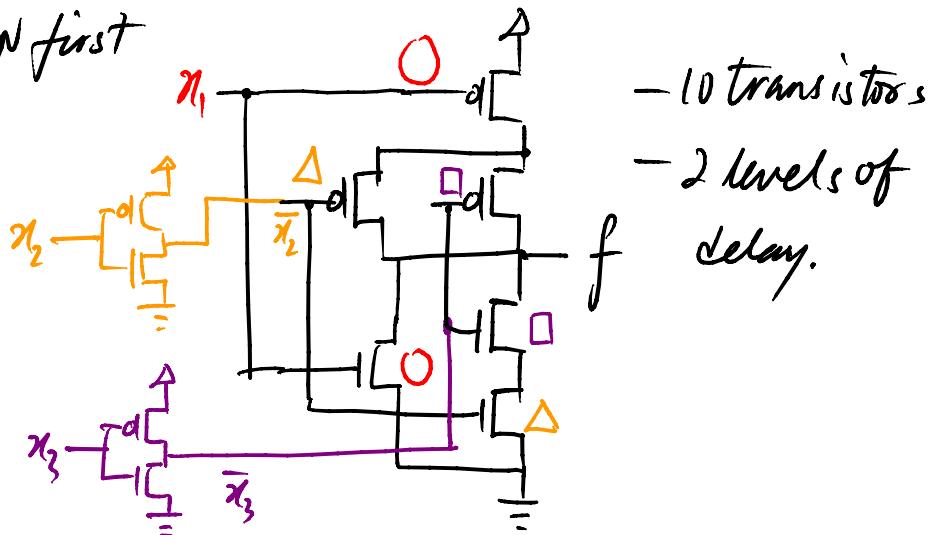
example 2 $f = \bar{x}_1 (\bar{x}_2 + \bar{x}_3)$

Option 1 → draw PDN first.
Drawing the cct. for \bar{f}
then add an inverter
 $\bar{f} = \underline{\bar{x}_1 (\bar{x}_2 + \bar{x}_3)}$

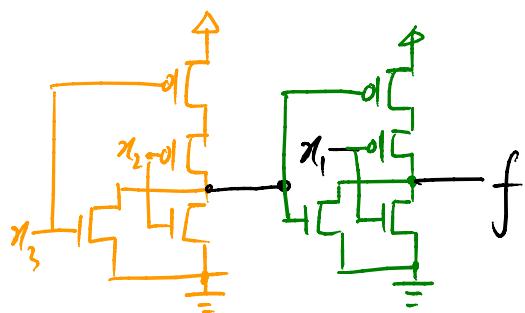


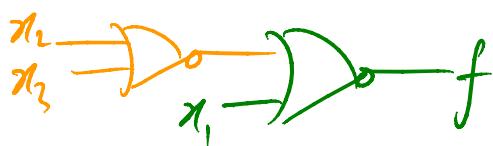
Option 2 → draw PUN first

$$f = \bar{x}_1 (\underline{\bar{x}_2 + \bar{x}_3})$$



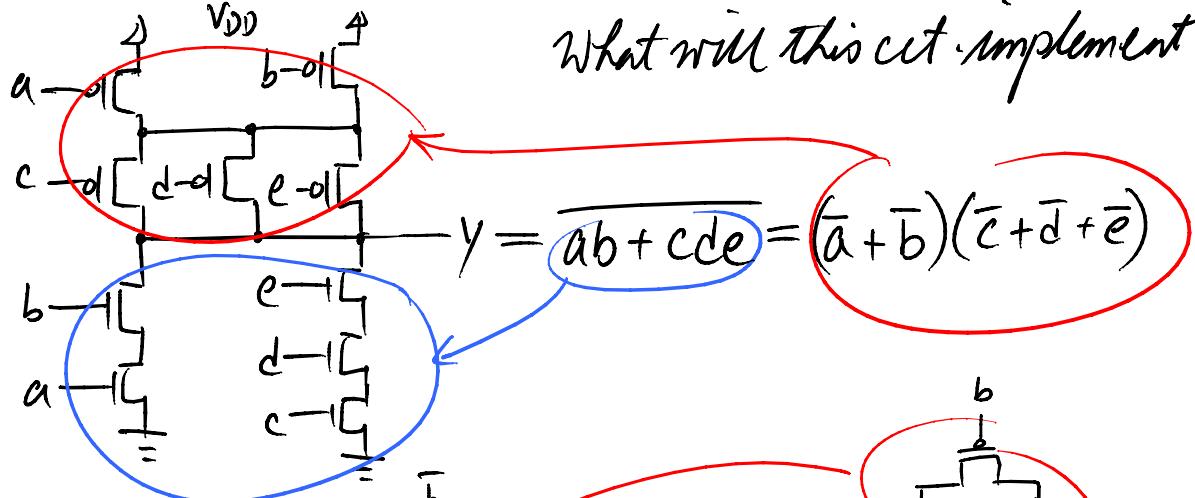
Option 3 $f = \underline{\bar{x}_1 (\bar{x}_2 + \bar{x}_3)}$
= $\underline{\bar{x}_1 + (\bar{x}_2 + \bar{x}_3)}$
NOR



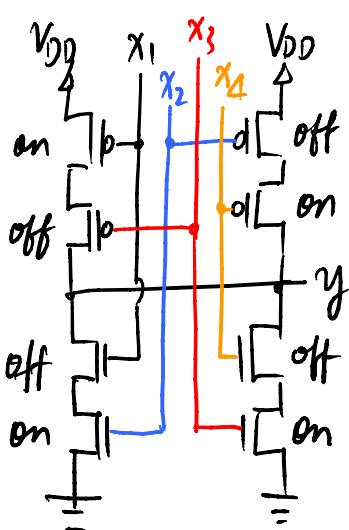
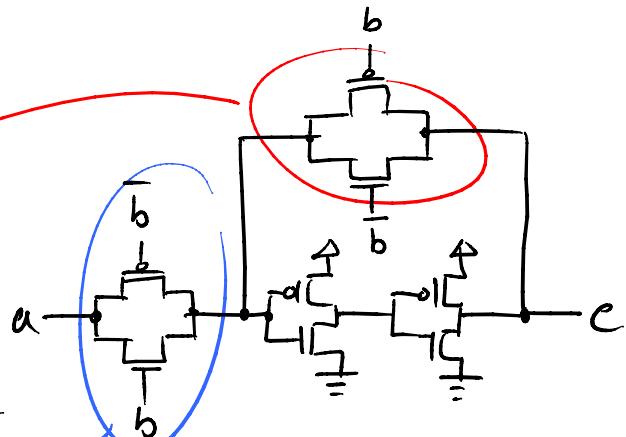
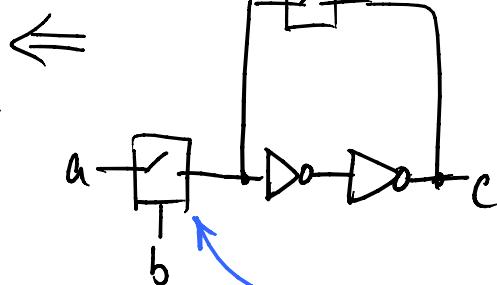
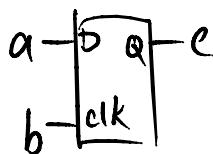


- 8 transistors
- 2 levels of delay.

\Rightarrow Some interesting CMOS ccts



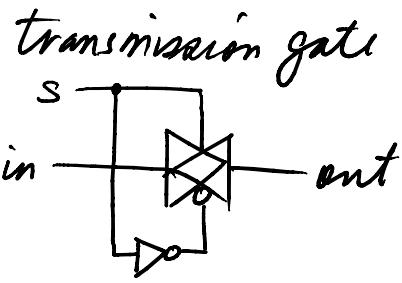
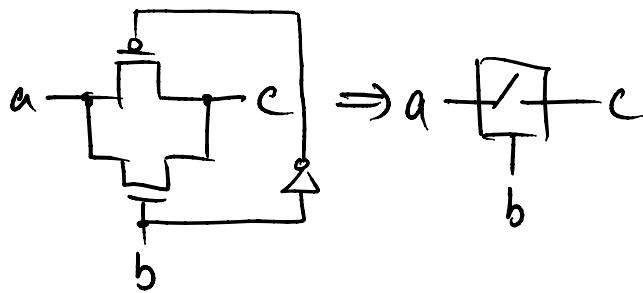
D-latch



PDN when $x_1, x_2 + x_3, x_4$
PUN when $\overline{x}_1, \overline{x}_3 + \overline{x}_2, \overline{x}_4$

neither a pull up
nor a pull down path.

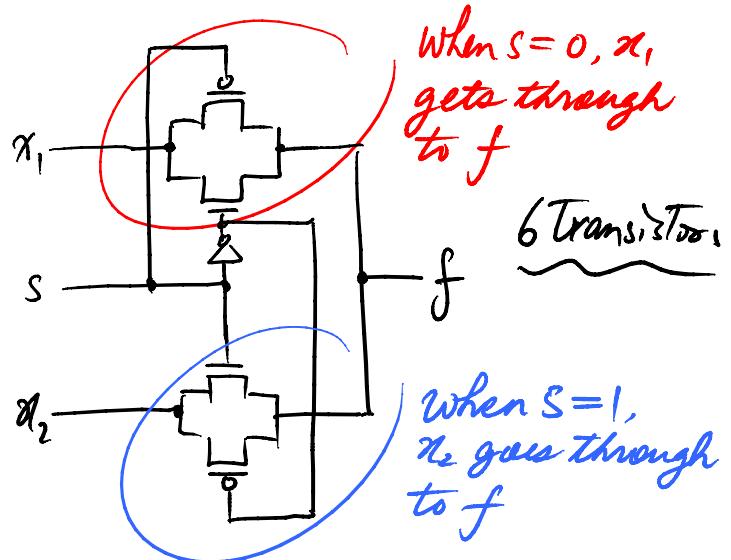
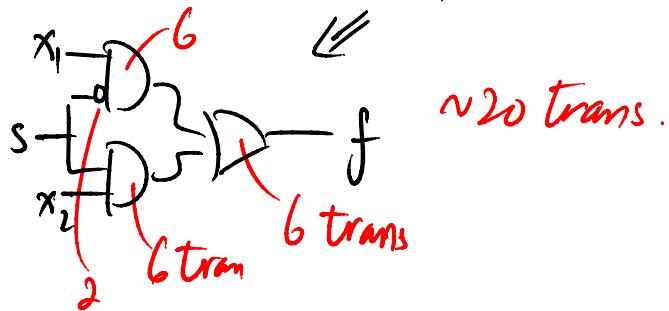
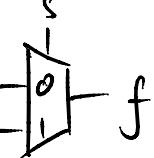
| x_1 | x_2 | x_3 | x_4 | y |
|-------|-------|-------|-------|-----|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |



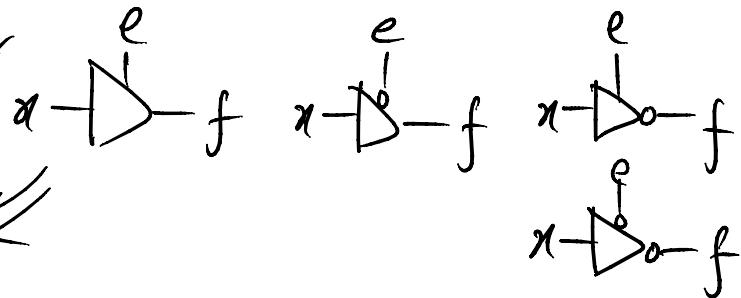
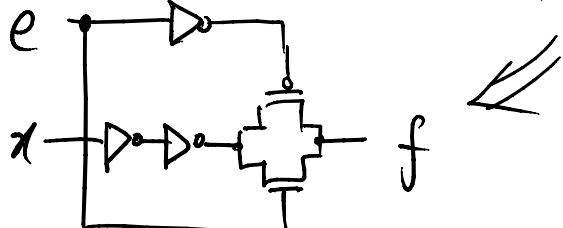
logic 1 goes through
logic 0 goes through

use transmission gates

try to implement



make a tri-state buffer



make an XOR gate

| x_1 | x_2 | f |
|-------|-------|-----|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

$\{ f = x_2 \}$ $\{ f = \bar{x}_2 \}$

