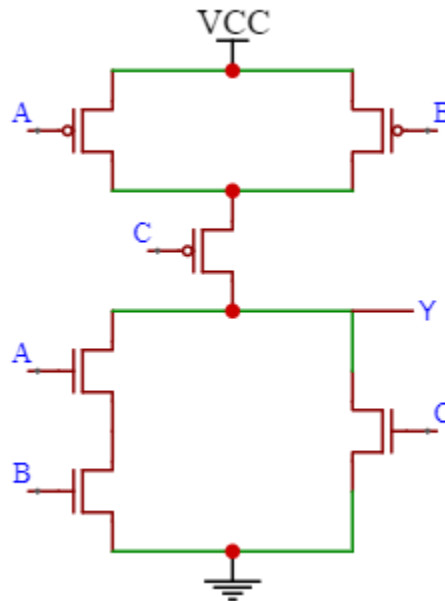


Elektronika alapjai 4. gyakorlat

A gyakorlaton megoldott feladatok

1. CMOS komplex kapuk megvalósítása

Milyen logikai függvényt valósít meg az alábbi CMOS kapu?



Hány tranzisztorral tudnánk megvalósítani ezt a kaput, "hagyományos" alapkapuk használatával?
Tervezze át a kaput, hogy az

$Y = \overline{(A + B)C}$ függvényt valósítsa meg!

A kapcsolás elemzése egyszerű, emlékezzünk arra, hogy a pull-down network vezet (az NMOS tranzisztorok), ha a kimenet logikai értéke 0 kell, hogy legyen, ha pedig a pull-up network vezet (a PMOS tranzisztorok), akkor pedig a kimenet értéke 1 lesz. A kimenet a kapcsolási rajz szerint kétféleképpen lehet 0:

- az A és B-hez tartozó NMOS tranzisztor egyszerre vezet, azaz a logika függvény A ÉS B
- a C-hez tartozó NMOS tranzisztor vezet

Logikai kifejezéssel leírva:

$$Y = \overline{AB + C}$$

ellenőrizzük, hogy a pull-up network jól működik-e. Két De Morgan átalakítás után

$$Y = \overline{AB + C} = \overline{AB} \cdot \overline{C} = (\overline{A} + \overline{B})\overline{C}$$

Tehát a kimenet akkor lesz logikai 1, ha $C = 0$ és A vagy B nulla. A kapcsolási rajzon pedig pontosan ezt látjuk.

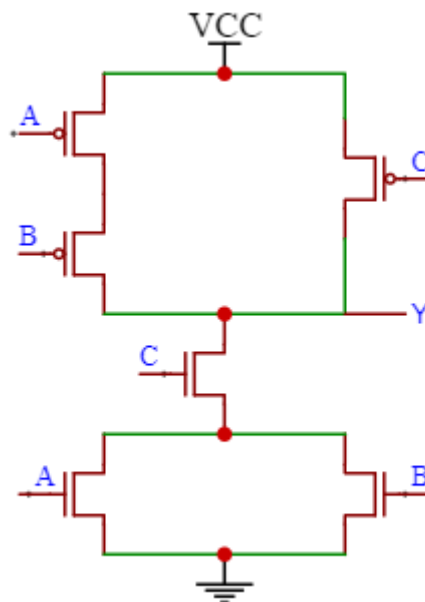
A hagyományos megvalósításhoz szükségünk lenne egy kétbemenetű NAND, kétbemenetű NOR és egy inverter kapura, amit összesen 10 db tranzisztorral tudunk megvalósítani. Látható, hogy a komplex kapuk használata előnyös. Ráadásul a többszintű megvalósítás esetén a jelek késleltetése nem lesz egyforma. [A kapu interaktív kapcsolási rajza](#).

Tervezzük át a kaput, hogy az $Y = \overline{(A + B)C}$ függvényt valósítsa meg!

A tervezés nagyon egyszerű, kezdjük a pull-down hálózattal! (mivel ennek logikai kifejezése rögtön adott) ÉS kapcsolatnak sorbakötött, VAGY kapcsolatnak párhuzamosan kötött hálózatrészek felelnek meg, azaz két tranzisztort kell párhuzamosan kötni az $A+B$ kifejezés megvalósításához, és ezzel pedig sorbakötni egy harmadikat. A pull-up network megvalósítására két út is van. A pull-up network rövidzár akkor, amikor a pull-down network szakad, illetve fordítva (ha nem így lenne, a kapu nem logikai kapuként, hanem rövidzárként működne). Azaz semmi mást nem kell tenni, mint a párhuzamos-soros kapcsolatokat megcserélni. (Ennek példáját láttuk az alapkapuk esetén.) A másik lehetőség, hogy a De Morgan azonosságokat használva:

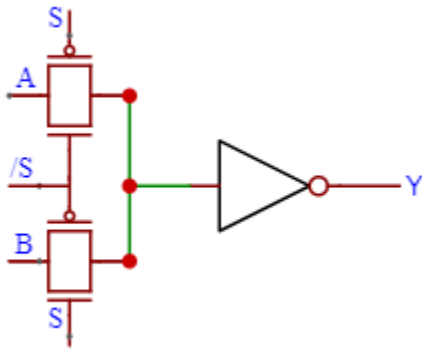
$$Y = \overline{(A + B)C} = \overline{A + B} + \overline{C} = \bar{A} \cdot \bar{B} + \bar{C}$$

Ez alapján a kapcsolási rajz:



2. CMOS transzfer kapu

Milyen logikai függvényt valósít meg az alábbi kapcsolás? A /S rajztechnikai okokból az S jel negáltját jelenti.



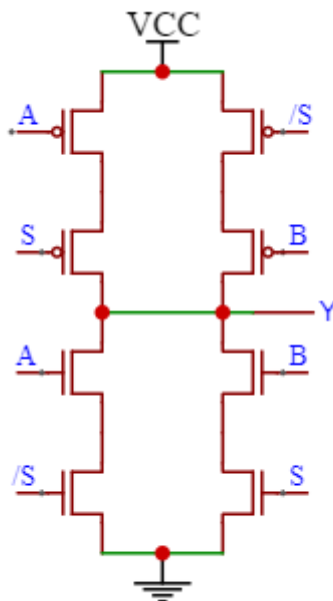
Egyszerűen végiggondolva azt, hogy mi történik, a következőkre juthatunk. Ha $S=0$, akkor az inverter bemenetére a "felső" transzfer kapu a A jelet kapcsolja, miközben az "alsó" transzfer kapu zárva van. Ha $S=1$, akkor az "alsó" transzfer kapu működik és az inverter bemenetére a B jelet kapcsolja rá.

Összefoglalva:

$$Y = \overline{A\bar{S}} + BS$$

A komplex kaput az előző feladat alapján könnyen megtervezhetjük.

Invertáló multiplexer komplex kapuval:



(az itt szereplő kapcsolási rajzban vagy egy apró "trükk". Egy összekötéssel kevesebb van, mégis helyes. Miért?)

A komplex kapuban 8 tranzisztor lesz, míg a transzfer kapus megvalósítás csak hatot tartalmazott. Tehát a transzfer kapuk használata is előnyös.

[A két megvalósítás megtekinthető itt](#)

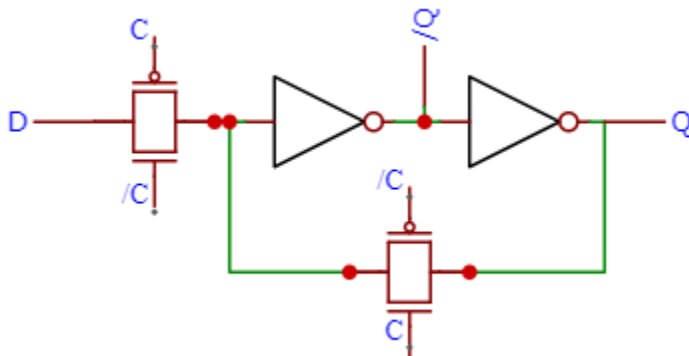
3. CMOS tárolók

Hogyan kellene átalakítani az előadáson látott latch-et úgy, hogy az órajel alacsony szintjén legyen átlátszó?

Fel kell cserélni az órajelek polaritását az előadáson szereplőhöz képest, semmi több. Az első komplex kapu órajel alacsony szintjén vezet, ekkor a visszacsatoló komplex kapu leválaszt. Ekkor a bemenet a két inverteren keresztül íródik a kimenetre, azaz a latch átlátszó.

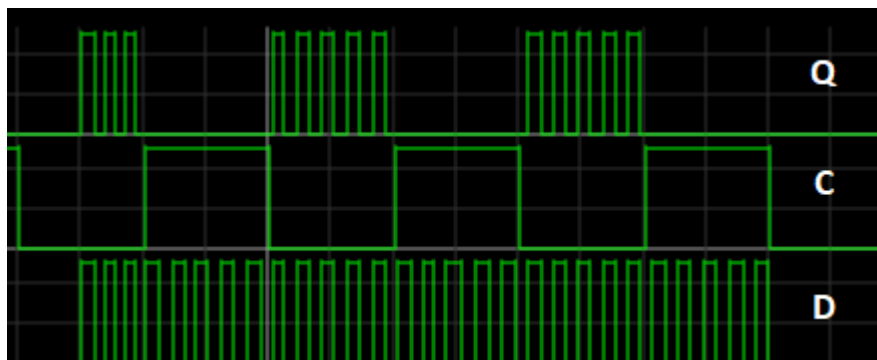
Ha az órajel magas, akkor az első komplex leválasztja a bemenetet, a második pedig létrehozza a két inverter között a visszacsatolást, azaz a 2. inverter kimenetét az első inverter bemenetére kapcsolja. Ez a rendszer stabil, amíg tápfeszültség van, megőrzi az állapotát.

A kapcsolási rajz tehát:



[A működés interaktívan.](#)

Egy jellemző hullámforma pedig:



4. CMOS teljesítményigény, fogyasztás

1. Egy otthoni router átlagos teljesítménye 5W. Mennyibe kerül a havi üzemeltetése, ha 1kWh kb. 40Ft? Mennyibe kerül egy átlagosan 70W-os PC alapú szerver 0-24 otthoni üzemeltetése havonta?

Egy óra alatt az elfogyasztott energia 0,005kWh. Egy átlagos 30 napos hónappal számítva $0,005 \cdot 24 \cdot 30 = 3,6\text{kWh}$

Ez kb. 144 forintot jelent. A PC-s szerver ezekkel az adatokkal számítva több mint 2000Ft havonta, ami azért már egy észrevehető összeg lesz majd az elszámoló számlában...

2. Egy CMOS technológiával készült SoC órajele 1GHz, tápfeszültsége 3V. A rendszer így teljesen feltöltött akkumulátorról 12 órát működik. Az órajelet felére csökkentjük.
 - a. Meddig fog teljesen feltöltött akkumulátorról működni?
 - b. Egy taszk az eredeti rendszeren 100s-ig futott és 1kJ energiát használt fel. Mennyi ideig fog futni a módosított rendszeren és mennyi energiát használ fel?
 - c. Egy folyamatos, felügyelet jellegű taszkot futtatunk, amely biz. események bekövetkezésekor 10ms múlva tud reagálni. Hogyan változik ez a reakcióidő a felére csökkentett órajelű rendszeren?
 - d. Hogyan alakulnak ezek az értékek, ha a tápfeszültséget is kétharmadára csökkentjük?
 - a. A csökkentett fogyasztás a fele lesz, így kétszer annyi ideig fog a rendszer akkumulátorról működni, azaz 24 órát.
 - b. A teljesítményt felére csökkentjük, de a taszk a fele akkora órajellel kétszerannyi ideig fog futni, tehát 200s-ig és ugyanúgy 1kJ energiát használ el. Az órajel csökkentése önmagában nem lesz energiahatékony, ahhoz a tápfeszültséget is csökkenteni kell!
 - c. A feleakkora órajel miatt kétszer annyi idő lesz, mire reagál, azaz 20ms
 - d. Ha a tápfeszültséget is változtatjuk, a fogyasztás: $P' = \frac{1}{2} \left(\frac{2}{3}\right)^2 P = \frac{2}{9} P$ módon alakul. Azaz 4,5x több ideig fog működni, 54 órát. Az órajelet felére csökkentettük, azaz a taszk kétszer annyi ideig fog futni, 200 másodpercig, de a teljesítményigény 2/9 részére csökken, így a felhasznált energia: $W = 1\text{kJ} \cdot \frac{2}{9} \cdot 2 = 444\text{J}$. A reakcióidő ugyanúgy 20ms lesz, mint az előző esetben.
3. Egy rendszerben a mikroprocesszor magfeszültsége 3GHz-en 1,1V. A rendszert kétprocesszorossá szereljük át és 1,5GHz frekvencián működtetjük, 700mV tápfeszültségről. Feltételezzük, hogy a processzor fogyasztásának nagy részét a töltéspumpálás okozza.

A szokásos képletet alkalmazva:

$$\frac{P_1}{P_2} = 2 \cdot \frac{1,5}{3} \cdot \frac{0,7^2}{1,1^2} = 0,404$$

Tehát a szükséges elektromos teljesítményigény KÉTÖTÖDRÉSÉRE csökken, míg a számítástechnikai értelemben vett teljesítménye (performance) változatlan. Azaz a kétprocesszoros, kisebb frekvencián működtetett rendszer energiahatékonyasága (performance/W) két és félszeres, feltéve, hogy a taszkokat hatékonyan el tudjuk osztani a processzorok között.

5. CMOS kapu kimenetének megváltozása

Egy kétbemenetű CMOS NAND kapu bemenete p valószínűséggel változik meg. Mekkora valószínűséggel változik meg a kapu kimenete?

Ki kell számolni minden lehetséges állapotban a megváltozás valószínűségét, majd összegezni. NAND esetén pl. ha a bemeneti kombináció 01 volt, a kimenet megváltozásához az kell, hogy A 1-be váltszon, és eközben B ne változzon, azaz $p(1 - p)$

Összeszedve és $1/4$ súllyal összegezve a valószínűségeket (sorban a 00 01 10 11 állapotokra):

$$p(ki) = \frac{1}{4}(p^2 + p(1 - p) + (1 - p)p + 1 - (1 - p)^2) = p - \frac{p^2}{2}$$

6. CMOS késleltetés, kritikus út

Egy teljes összeadó esetén a carry $100ps$, az összeg pedig a carry elkészülése után további $100ps$ után készül el. Mekkora lesz egy 12 bites, ripple carry összeadó legnagyobb késleltetése?

A kritikus útvonal a carry terjedése, ez fogja a teljes hálózat késleltetését és ebből következően a maximális órajelfrekvenciát meghatározni.

A késleltetés $t_{pd} = 12 \cdot 100ps + 100ps = 1300ps$ lesz.