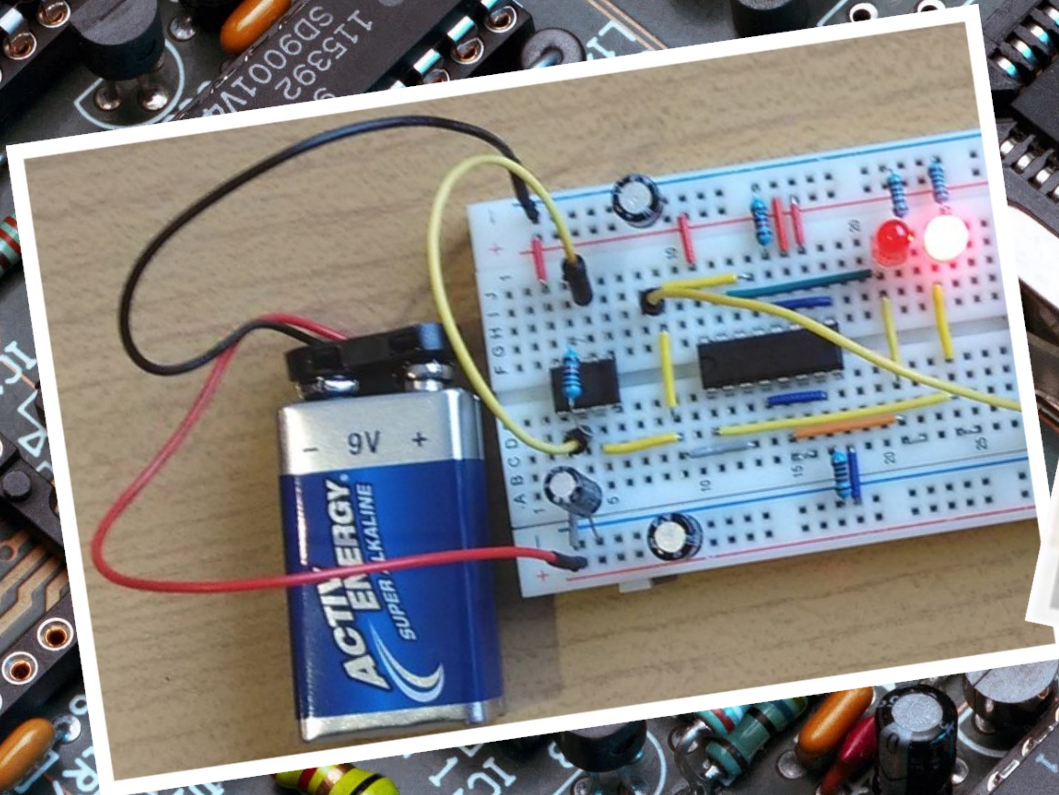


A digitális elektronika alapjai



2. Logikai kapuáramkörök

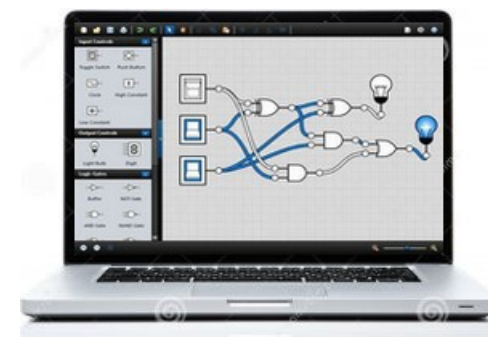
Felhasznált és ajánlott irodalom

- Gulyás Dénes: [Számítógép architektúrák](#) (*interaktív jegyzet*)
- Mike Gábor: [A digitális elektronika alapjai](#) (*jegyzet és videók*)
- Zalotay Péter: [Digitális technika](#)
- Végh János: [Ismerkedés a digitális elektronikával](#)
- Mészáros Miklós: [Logikai algebra alapjai, logikai függvények I.](#)
- Mingesz Róbert: [Digitális technikai tananyagok](#)
- F-alpha.net: [Digital Electronics](#)
- Electronics Tutorials: [Logic Gates](#)
- M. Morris Mano and Michael D. Ciletti:
[Digital Design - With an Introduction to the Verilog HDL, 5th. Edition](#)



Logikai áramkör szimulátorok

- LogiSim szimulátor: www.cburch.com/logisim/
- Falstad.com: [Circuit simulator](#)
- CircuitVerse: [Simulator](#)
- University of Genoa: [Deeds Simulator](#)
- Gatecat: [Breadboard Simulator v1.0](#)
- Logic.ly: [Logic.ly Simulator \(online demo\)](#)



Ismétlés: Boole-algebra

- A **Boole-algebra** egy olyan kétműveletes algebrai struktúra amely a **halmazműveletek**, a **logikai műveletek** és az **eseményalgebra** műveleteinek közös tulajdonságaival rendelkezik
- A kétértékű Boole-algebra az alábbi axiómákra épül:
 - 1) Az algebra kétértékű elemek halmazára értelmezett
 - 2) A halmaz minden elemének létezik **komplementese**, amely szintén eleme a kétértékű elemek halmazának ($x + x' = 1$; $x \cdot x' = 0$)
 - 3) Az elemek között végezhető műveletek a **konjunkció** (logikai ÉS, logikai „szorzás”: \cdot), illetve a **diszjunkció** (logikai VAGY, logikai „összeadás”: $+$) és a műveletek eredménye is eleme a kétértékű elemek halmazának
 - 4) A struktúra a logikai műveletekre nézve legyen: **Kommutatív, Asszociatív és Disztributív**
 - 5) A halmaz kitüntetett elemei:
 - ❖ **egység elem (1)** – ($x + 1 = 1$; $x \cdot 1 = x$)
 - ❖ **nulla elem (0)** – ($x + 0 = x$; $x \cdot 0 = 0$)

Ismétlés: A Boole-algebra műveletei

- A kétértékű **Boole-algebra** bináris (kétváltozós) **műveleteinek** eredménye a $\mathbf{B} = \{0,1\}$ halmaz valamelyik eleme lesz, a műveletben szereplő két logikai változó értékétől függően. A műveletvégzés szabályait táblázatos formában is megadhatjuk.

Igazságtáblázat

x	y	$x \cdot y$
0	0	0
0	1	0
1	0	0
1	1	1

x	y	$x + y$
0	0	0
0	1	1
1	0	1
1	1	1

x	x'
0	1
1	0

A 0 és 1 itt nem számok, hanem szimbólumok!

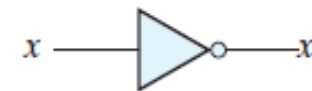
Logikai kapu rajzjele



Kétbemenetű ÉS kapu



Kétbemenetű VAGY kapu



NEM kapu (inverter)

- Az ábrán szereplő harmadik művelet a 2. axiómában definiált komplementum előállítására szolgál
- A logikai áramkörök építőkövei az úgynevezett **kapuáramkörök**, amelyek egy-egy elemi logikai művelet (NEM, ÉS, VAGY kapcsolat) elvégzésére képesek. A logikai áramköröknél a „0” pl. alacsony jelszinttel, az „1” pedig magas jelszinttel reprezentálható
- Egy logikai kifejezés több, különböző műveletet is tartalmazhat, ekkor a műveleti sorrend: 1. Zárójel, 2. Negáció (NEM), 3. ÉS művelet, 4. VAGY művelet

Ismétlés: A Boole-algebra azonosságai

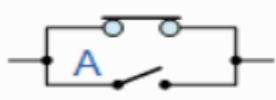
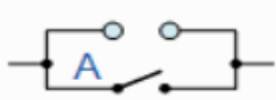
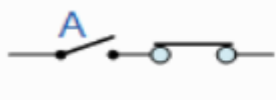
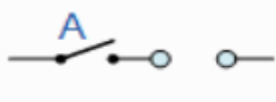
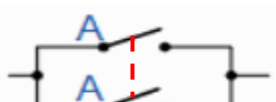
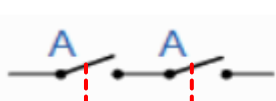
- **Dualitás elve:** Az (a) és (b) párok egyikből megkaphatjuk a másikat, ha a + és \cdot műveleteket felcseréljük, valamint a 0 helyébe 1-et, illetve az 1 helyébe 0-t írunk. Ez általános tulajdonsága a **Boole-algebrának**
- Az azonos átalakításokat az alábbi azonosságok szerint végezhetjük:

5. axióma	(a) $x + 0 = x$	(b) $x \cdot 1 = x$
2. axióma	(a) $x + x' = 1$	(b) $x \cdot x' = 0$
Idempotencia	(a) $x + x = x$	(b) $x \cdot x = x$
Korlátosság	(a) $x + 1 = 1$	(b) $x \cdot 0 = 0$
Involúció	$(x')' = x$	
Kommutativitás	(a) $x + y = y + x$	(b) $xy = yx$
Asszociativitás	(a) $x + (y + z) = (x + y) + z$	(b) $x(yz) = (xy)z$
Disztributivitás	(a) $x(y + z) = xy + xz$	(b) $x + yz = (x + y)(x + z)$
De Morgan tétel	(a) $(x + y)' = x'y'$	(b) $(xy)' = x' + y'$
Abszorpció	(a) $x + xy = x$	(b) $x(x + y) = x$

Ahol nem zavaró, ott a logikai szorzás jelét elhagytuk!

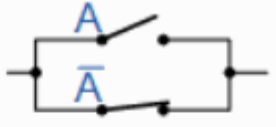
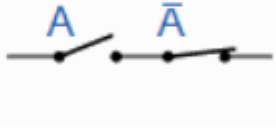
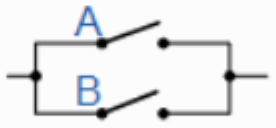
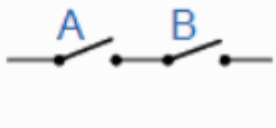
A Boole algebra azonosságai – kicsit másképp

- A már ismerős szabályokat kapcsolókkal is szemléltethetjük (forrás: [Electronics Tutorials: Laws of Boolean Algebra](#))

Boolean Expression	Description	Equivalent Switching Circuit	Boolean Algebra Law or Rule
$A + 1 = 1$	A in parallel with closed = "CLOSED"		Identity
$A + 0 = A$	A in parallel with open = "A"		Annulment
$A \cdot 1 = A$	A in series with closed = "A"		Identity
$A \cdot 0 = 0$	A in series with open = "OPEN"		Annulment
$A + A = A$	A in parallel with A = "A"		Idempotent
$A \cdot A = A$	A in series with A = "A"		Idempotent

A Boole algebra azonosságai – kicsit másképp

- A már ismerős szabályokat kapcsolókkal is szemléltethetjük (forrás: [Electronics Tutorials: Laws of Boolean Algebra](#))

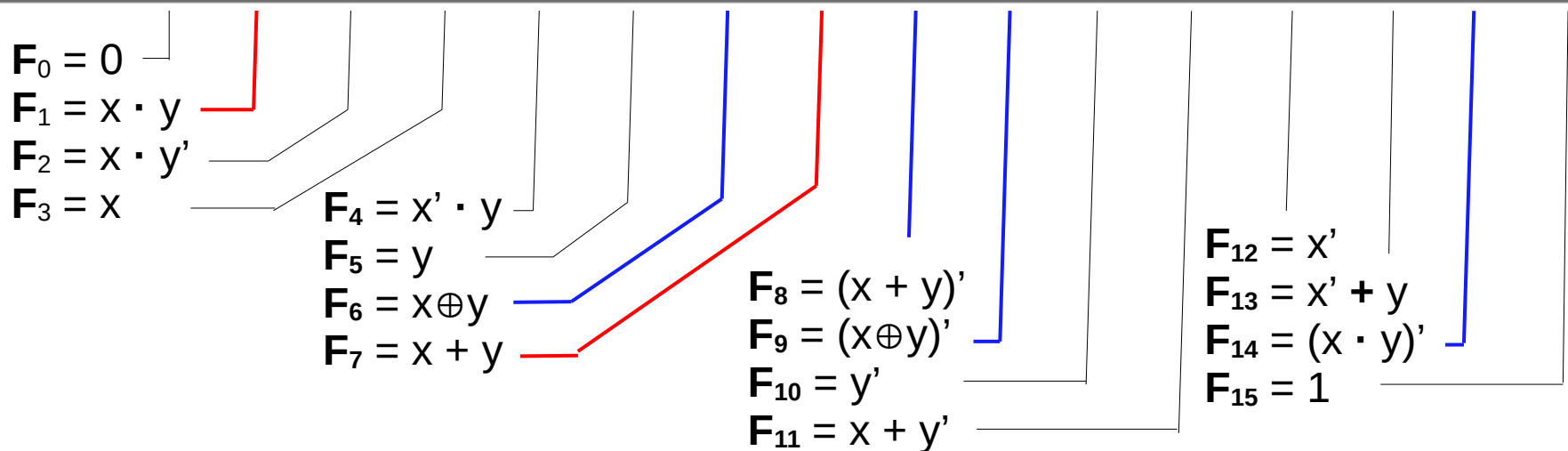
Boolean Expression	Description	Equivalent Switching Circuit	Boolean Algebra Law or Rule
$A + \bar{A} = 1$	A in parallel with NOT A = "CLOSED"		Complement
$A \cdot \bar{A} = 0$	A in series with NOT A = "OPEN"		Complement
$A + B = B + A$	A in parallel with B = B in parallel with A		Commutative
$A \cdot B = B \cdot A$	A in series with B = B in series with A		Commutative

További logikai műveletek

- A kétváltozós logikai függvények igazságtáblázatát az eddig tárgyalt **ÉS** és **VAGY** logikai műveleten kívül másképpen (összesen 16-féleképpen) is kitölthetjük. Ezeket, vagy ezek némelyikét újabb logikai műveletnek is tekinthetjük

Kétváltozós logikai függvények igazságtáblázatai

x	y	F_0	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1




További logikai műveletek

Boole függvény	Műveleti jel	Elnevezés	Megjegyzés
$F_0 = 0$		Zérus	konstans
$F_1 = xy$	$x \cdot y$	AND (ÉS)	x és y
$F_2 = xy'$	x/y	Inhibíció	x, de nem y
$F_3 = x$		Transzfer	x
$F_4 = x'y$	y/x	Inhibíció	y, de nem x
$F_5 = y$		Transzfer	y
$F_6 = xy' + x'y$	$x \oplus y$	XOR	x vagy y, de mindkettő nem
$F_7 = x + y$	$x + y$	OR (VAGY)	x vagy y
$F_8 = (x + y)'$	$x \downarrow y$	NOR	nem-VAGY
$F_9 = xy + x'y'$	$(x \oplus y)'$	Ekvivalencia (XNOR)	x egyenlő y-nal
$F_{10} = y'$	y'	Komplement	nem y
$F_{11} = x + y'$	$x \subset y$	Implikáció	ha y, akkor x
$F_{12} = x'$	x'	Komplement	nem x
$F_{13} = x' + y$	$x \supset y$	Implikáció	ha x, akkor y
$F_{14} = (xy)'$	$x \uparrow y$	NAND	nem-ÉS
$F_{15} = 1$		Egység	konstans


Megjegyzés: Az inhibíció és az implikáció nem kommutatív és nem asszociatív, nem érdemes használni.

Logikai kapuk


- Az előző oldalakon felsorolt 16 függvény közül csak az alábbi nyolc terjedt el általánosan a digitális tervezés gyakorlatában

AND  $F = x \cdot y$


x	y	F
0	0	0
0	1	0
1	0	0
1	1	1

NAND  $F = (xy)'$


x	y	F
0	0	1
0	1	1
1	0	1
1	1	0

OR  $F = x + y$


x	y	F
0	0	0
0	1	1
1	0	1
1	1	1

NOR  $F = (x + y)'$


x	y	F
0	0	1
0	1	0
1	0	0
1	1	0

XOR  $F = xy' + x'y = x \oplus y$


x	y	F
0	0	0
0	1	1
1	0	1
1	1	0

equivalence (XNOR)  $F = xy + x'y' = (x \oplus y)'$

x	y	F
0	0	1
0	1	0
1	0	0
1	1	1

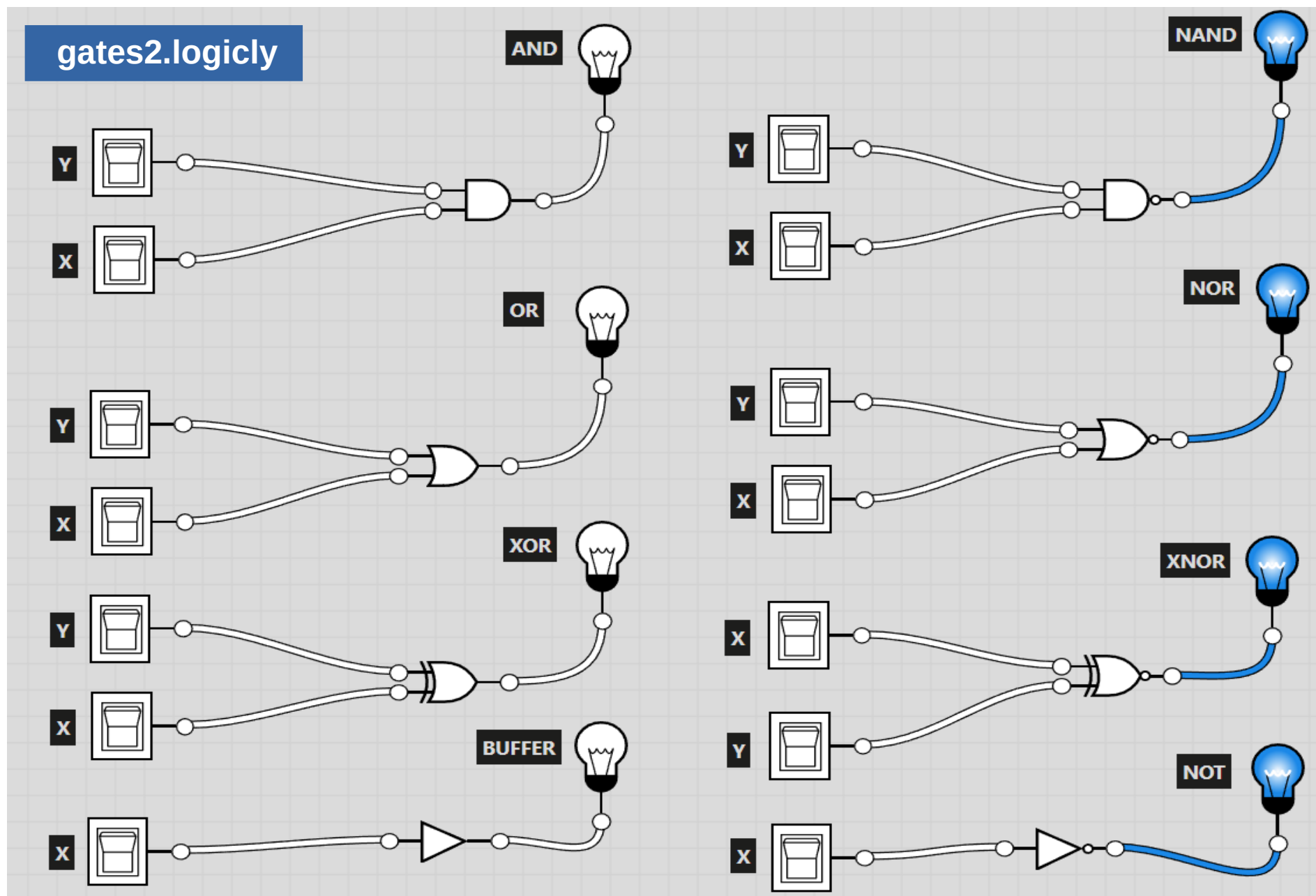
Buffer  $F = x$

x	F
0	0
1	1

Inverter  $F = x'$

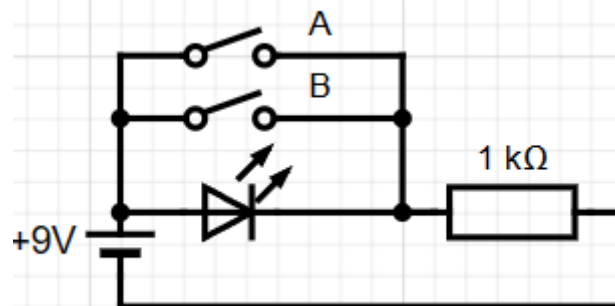
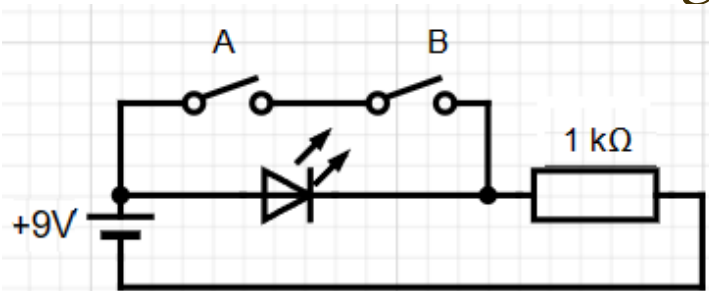
x	F
0	1
1	0

Vizsgáljuk meg a kapuk működését:



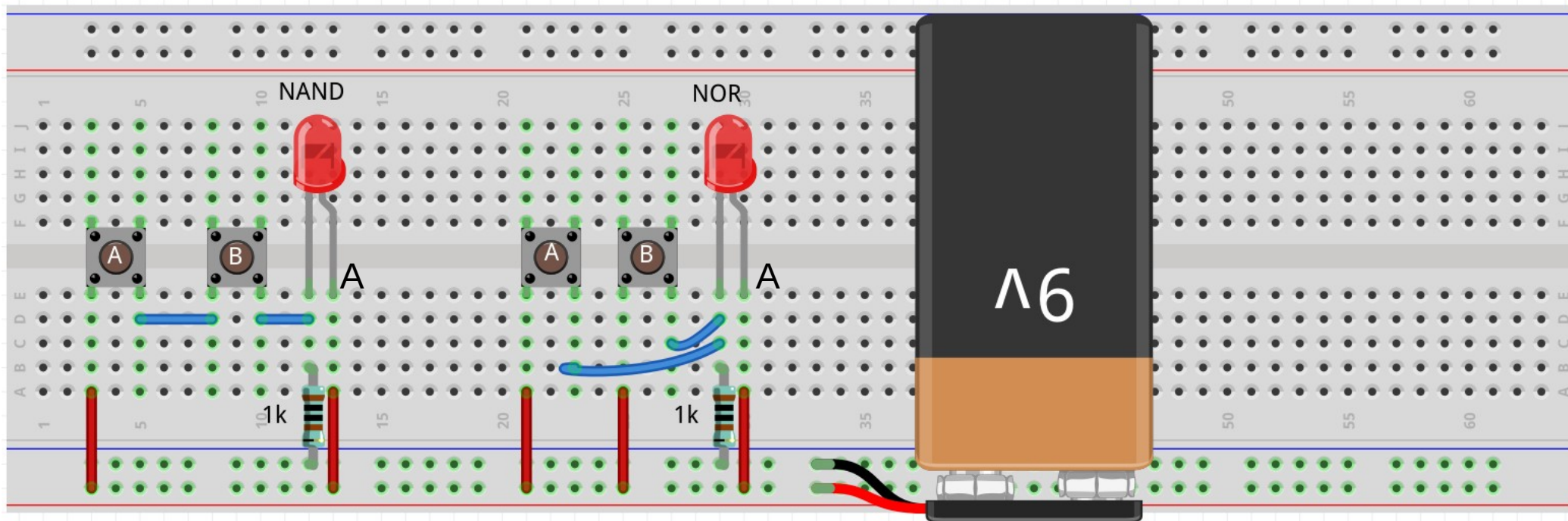
NAND és NOR kapcsolás nyomógombokkal

- A **NAND** és **NOR** kapcsolásnál a kapcsolók zárása nem felgyújtja, hanem lekapcsolja (rövidre zárja) a LED-et
Ne feledkezzünk meg a LED áramát korlátozó ellenállásról!



„1”, ha:

- a gomb lenyomva
- a LED világít

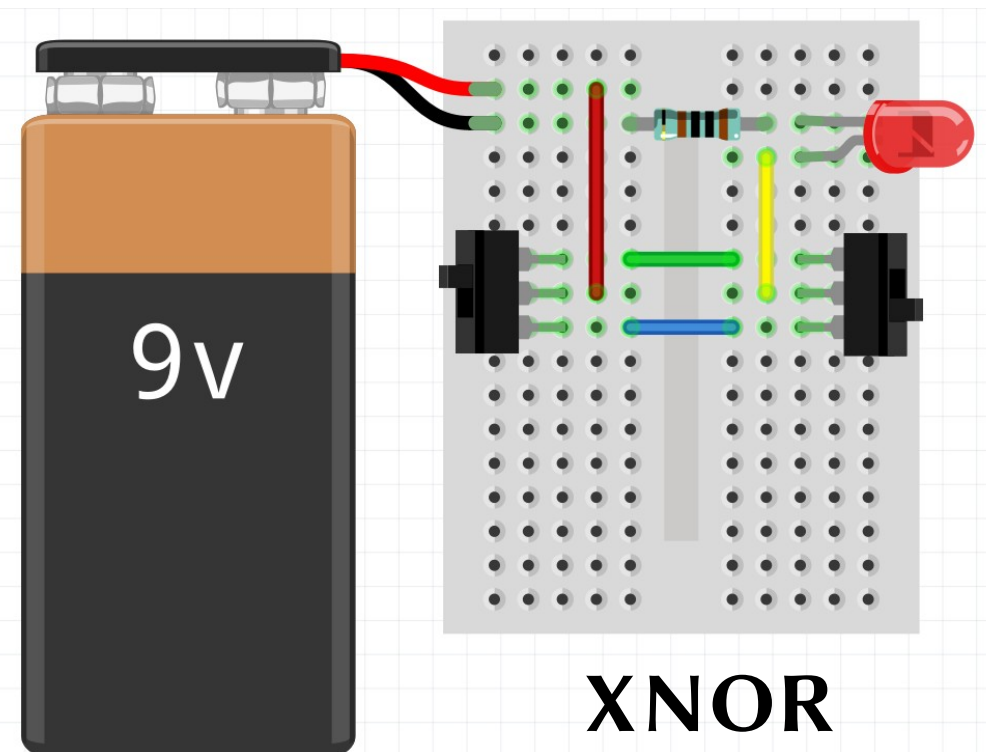
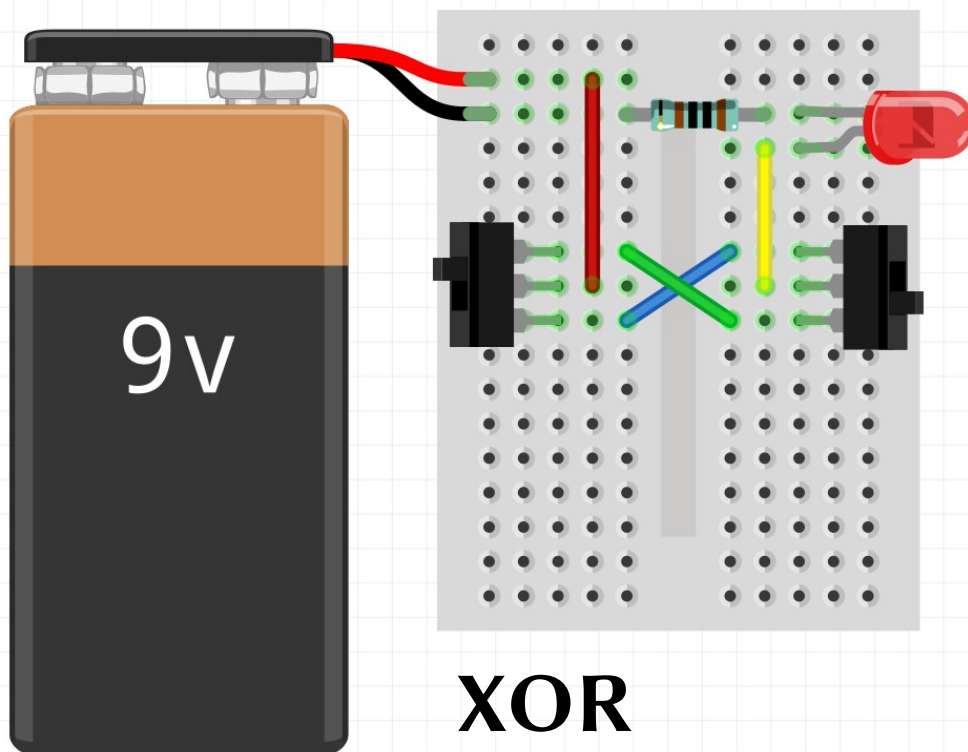


XOR és XNOR kapcsolás

- Az **XNOR** (vagy ekvivalencia) műveletet az ún. alternatív kapcsolással tudjuk szemléltetni (lásd pl. folyosó két végéről kapcsolható lámpa)
- Az **XOR** (kizáró vagy) művelet megvalósításához csak a kapcsolókat összekötő vezetékeket kell keresztbe kötni

„1”, ha:

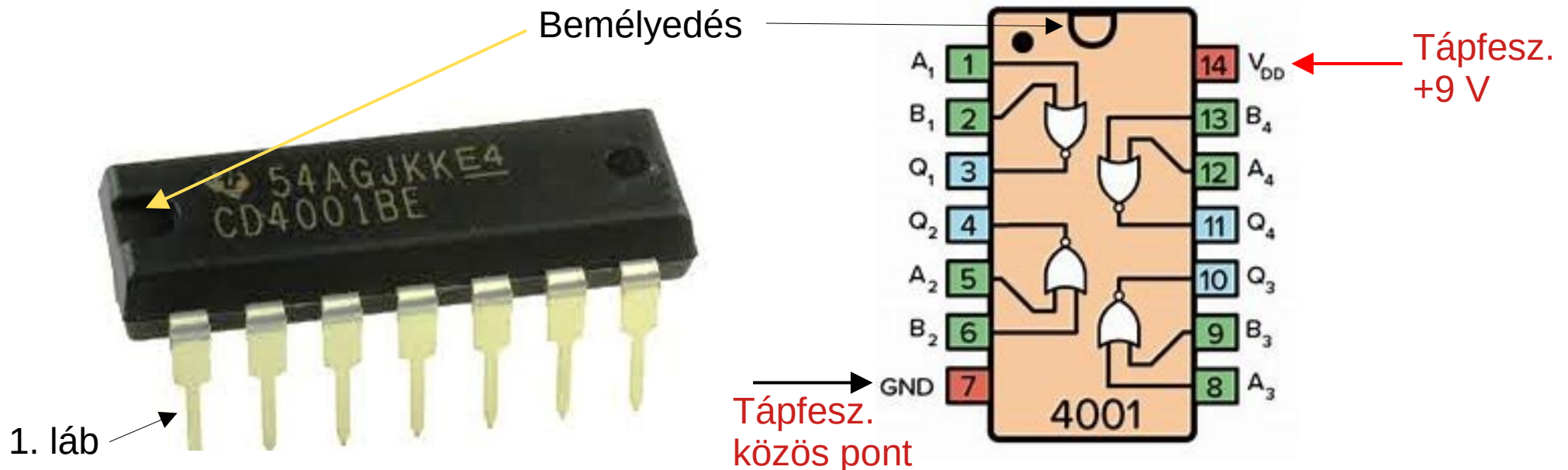
- a kapcsoló felső állásban van
- a LED világít





Integrált áramkörök

- Az integrált áramkörök (IC-k) nagyszámú miniatürizált tranzisztort és más elemi elektronikai alkatrészt egyesítenek egyetlen fizikai eszközben, amelyet egy adott célra vagy funkcióra terveztek, és amelynek legfőbb jellemzője az adott funkció teljesítése, nem pedig az eszközt felépítő alkotóelemek viselkedése
- Ilyenek például a mikrovezérlők, az akkumulátorfigyelők és -töltésvezérlők, az analóg-digitális átalakítók, a műveleti erősítők, a logikai kapuk, a feszültségszabályozók, a kapuvezérlők, a motorvezérlők és más IC-k
- Az általunk használt logikai IC-k DIP (dual in line package) tokozásúak, ahol az 1-es sorszámú láb helyét bemélyedés vagy egy kis pötty jelzi



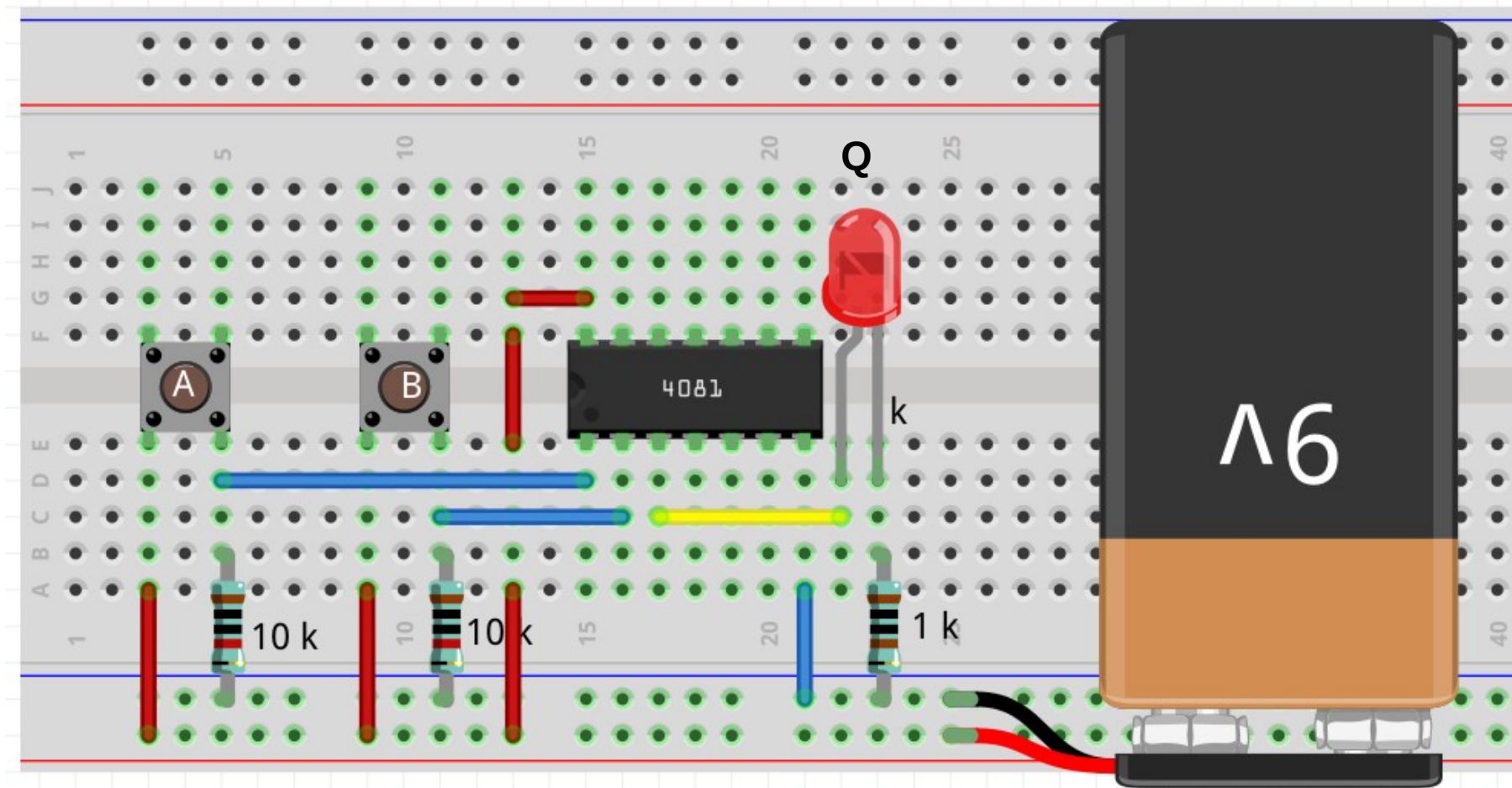
30 types 4000 Series CMOS Logic IC Kit

- A 30 darabos IC készletünk dobozában típusszám szerint növekvő sorrendben vannak elhelyezve az IC-k, tartsuk meg ezt a sorrendet!
- A következő oldalon található feladatban a képen bejelölt IC-kre lesz szükség (mindegyik kétbemenetű logikai kapuáramkörök tartalmaz, bekötésük egységes, megegyezik az előző oldalon bemutatottal)
- A feladatban szereplő **4077**-es típusszámú IC nincs benne a készletben (a későbbiekben nem is lesz rá szükségünk)



Kétbemenetű kapuk vizsgálata

- Vizsgáljuk meg a 4001, 4011, 4070, 4071, 4077 és 4081 típusszámú IC-k első kapuáramkörét (pin 1, 2, 3), s készítsük el az igazságtáblázatát!
- A felsorolt IC-k 4 db kétbemenetű kaput tartalmaznak
Láb kiosztás: 1: A, 2: B, 3: Q, 7: GND, 14: VCC



A vizsgált kapuk igazságtáblázata

■ 4001

A	B	Q
0	0	
0	1	
1	0	
1	1	

4011

A	B	Q
0	0	
0	1	
1	0	
1	1	

4070

A	B	Q
0	0	
0	1	
1	0	
1	1	

■ 4071

A	B	Q
0	0	
0	1	
1	0	
1	1	

4081

A	B	Q
0	0	
0	1	
1	0	
1	1	

4077

A	B	Q
0	0	
0	1	
1	0	
1	1	

A vizsgált kapuk igazságtáblázata

■ 4001

A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

4011

A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

4070

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

■ 4071

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

4081

A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

4077

A	B	Q
0	0	1
0	1	0
1	0	0
1	1	1

A vizsgált kapuk igazságtáblázata

■ 4001 NOR

A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

4011 NAND

A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

4070 XOR

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

■ 4071 OR

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

4081 AND

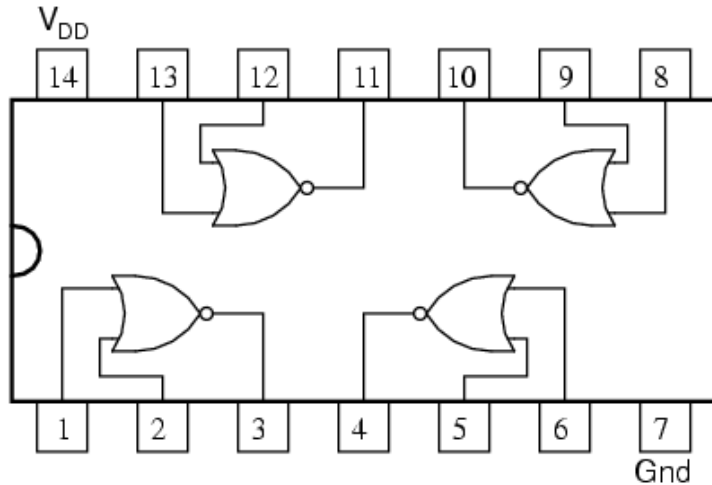
A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

4077 XNOR

A	B	Q
0	0	1
0	1	0
1	0	0
1	1	1

A 4000-es sorozat tipikus tagjainak lábkiosztása

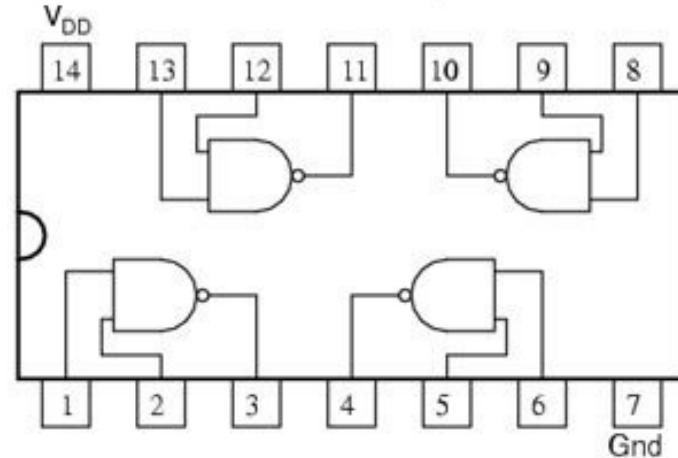
4001 Quad 2-Input NOR Gate



XOR

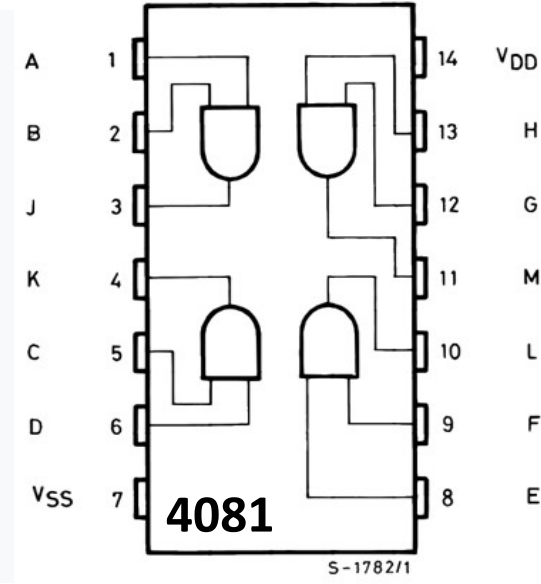
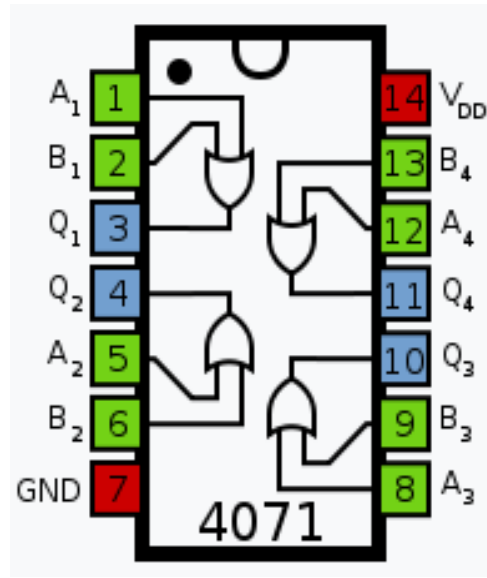
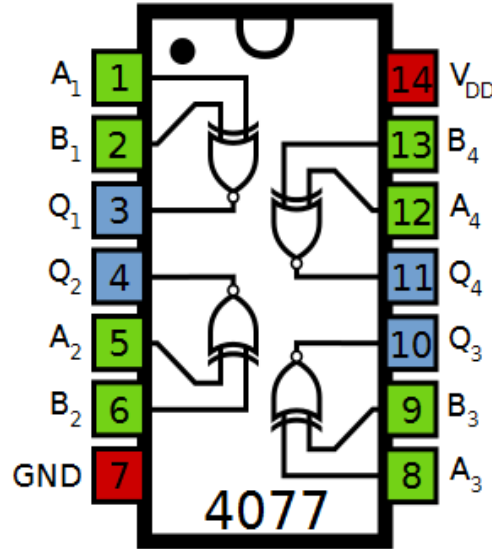
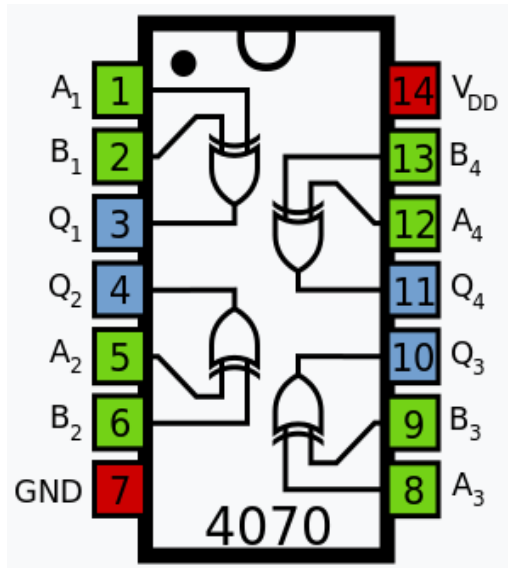
XNOR

4011 Quad 2-input NAND



OR

AND

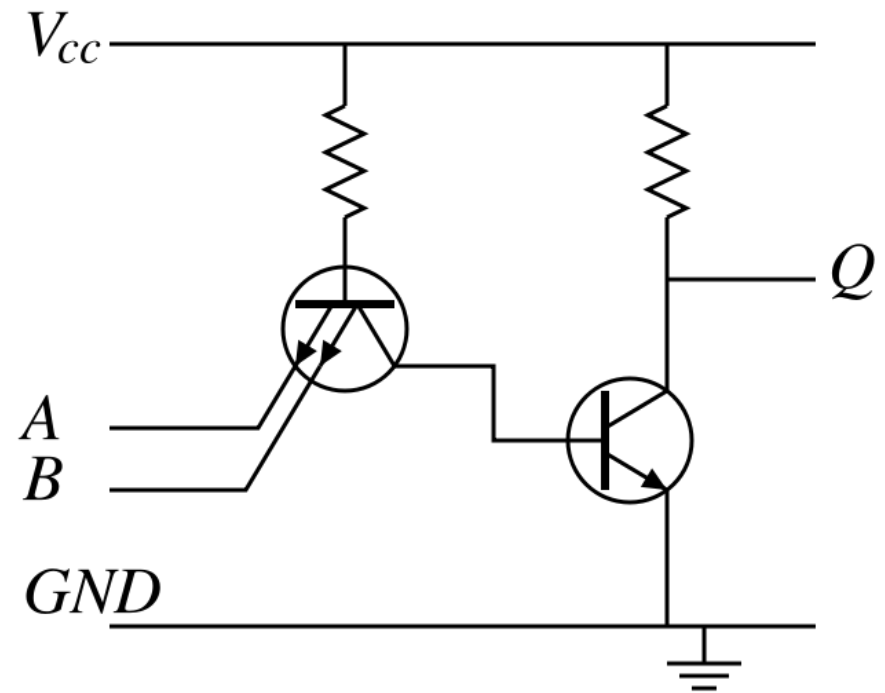


TTL Logikai áramkörök

A TTL (Transistor-Transistor Logic) kapcsolásban bipoláris tranzisztorok vannak, s a logikai bemenetek többemitteres tranzisztorok emitter kivezetései.

1. A több emitterrel rendelkező tranzisztorok működése egyenértékű azzal, ha több tranzisztor bázisát és kollektorát összekötjük.
2. Az ábrán látható kétbemenetű NEM-ÉS (NAND) áramkör ebben a leegyszerűsített formában nem mentes az RTL áramkörök ismert problémáitól.

A probléma a következő oldalon bemutatott totem-pole kimenettel orvosolható.



Két bemenetű NAND áramkör

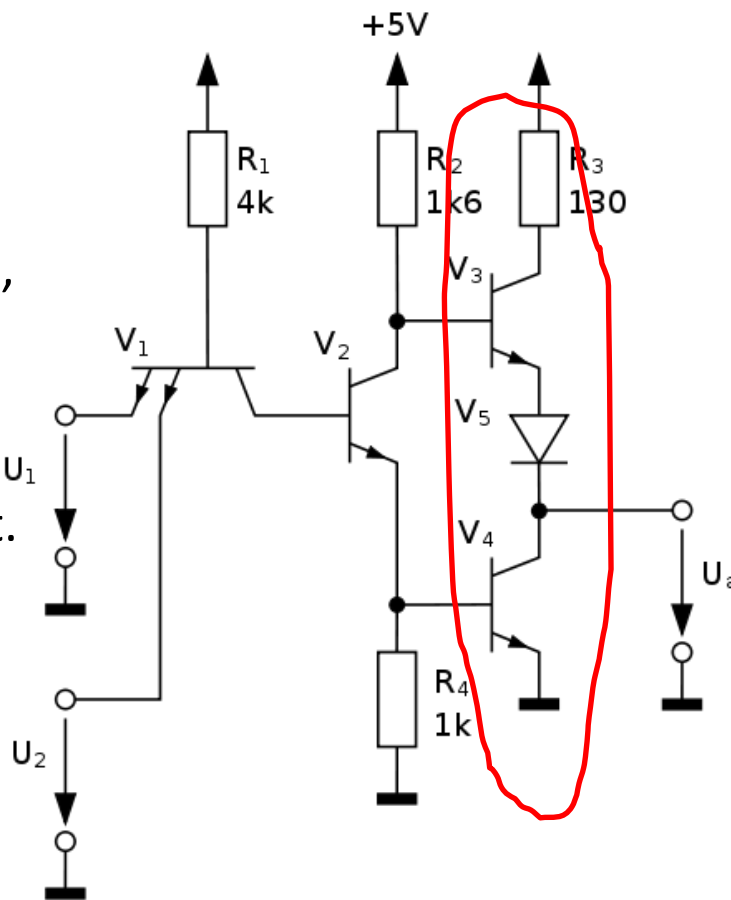
TTL kapu totem-pole kimenettel

Totem-pole = totem oszlop. Fából faragott oszlop, figurái egymás fölött helyezkednek el, „egymás fején ülnek” ... A TTL kimenet elnevezése onnan ered, hogy itt is egymás tetejére ültetett alkatrészeket látunk (az ábrán V3, V5 és V4).

Kétbemenetű TTL NAND áramkör totem-pole kimenettel

1. Alaphelyzetben V2 és V4 vezet, a kimenet alacsony szintű.
2. Ha valamelyik bemenetet lehúzzuk, V2 és V4 nem vezet, V3 vezet: magas kimeneti szint.

Hátrány: viszonylag alacsony kimeneti magas szint (~3,5 V)



TTL Inverter szimulációja

A <http://www.falstad.com/circuit/> címen elérhető áramkör szimulátor segítségével vizsgáljuk a kapcsolás működését!

A Circuits/Logic Families/TTL Inverter mintapéldát nézzük meg!

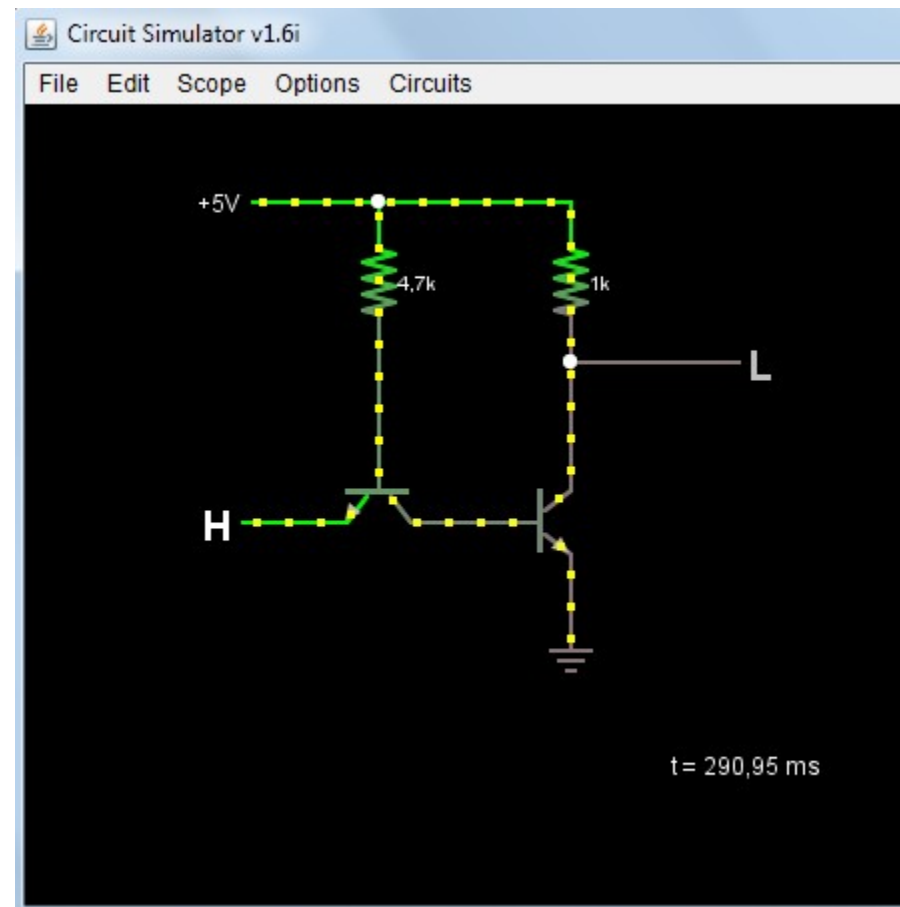
Baloldalt a bemenet állapota egérkattintással váltogatható:

H = magas szint (logikai 1)

L = alacsony szint (logikai 0)

Jobboldalt a kimenet állapota látható.

Len a be- és kimeneti feszültségek oszcilloszkópos megjelenítése látható.



TTL Inverter (NEM áramkör)

TTL NAND (NEM-ÉS)

A <http://www.falstad.com/circuit/> címen elérhető áramkör szimulátor segítségével vizsgáljuk a kapcsolás működését!

A Circuits/Logic Families/TTL NAND mintapéldát nézzük meg!

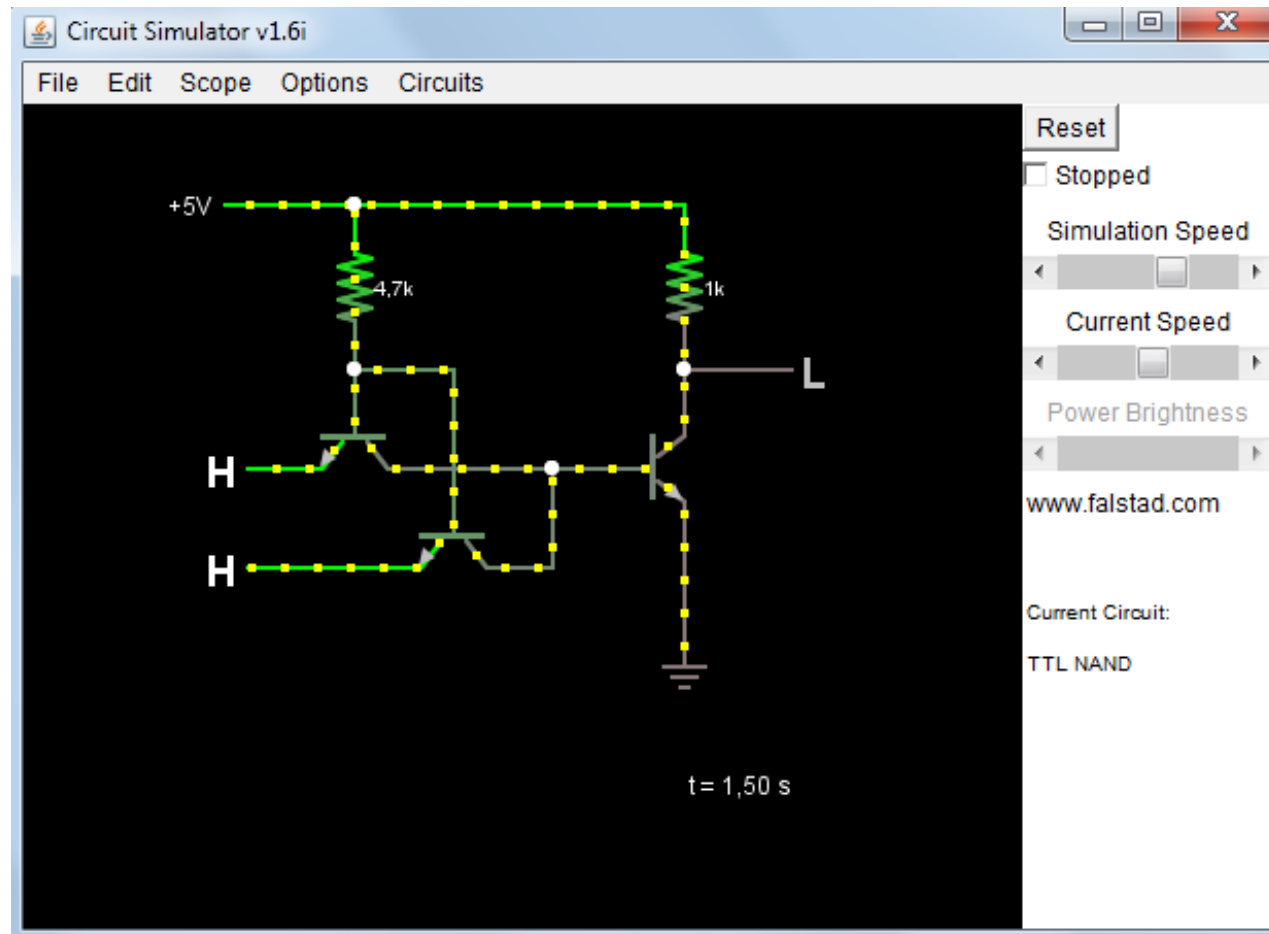


$$Y = \overline{A \cdot B}$$



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

A kimenet csak akkor alacsony, ha minden bemenet magas szinten van



2 bemenetű TTL NAND (NEM-ÉS)

TTL NOR (NEM-VAGY)

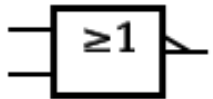
A <http://www.falstad.com/circuit/> címen elérhető áramkör szimulátor segítségével vizsgáljuk a kapcsolás működését!

A Circuits/Logic Families/TTL NOR mintapéldát nézzük meg!

Rajzjele:

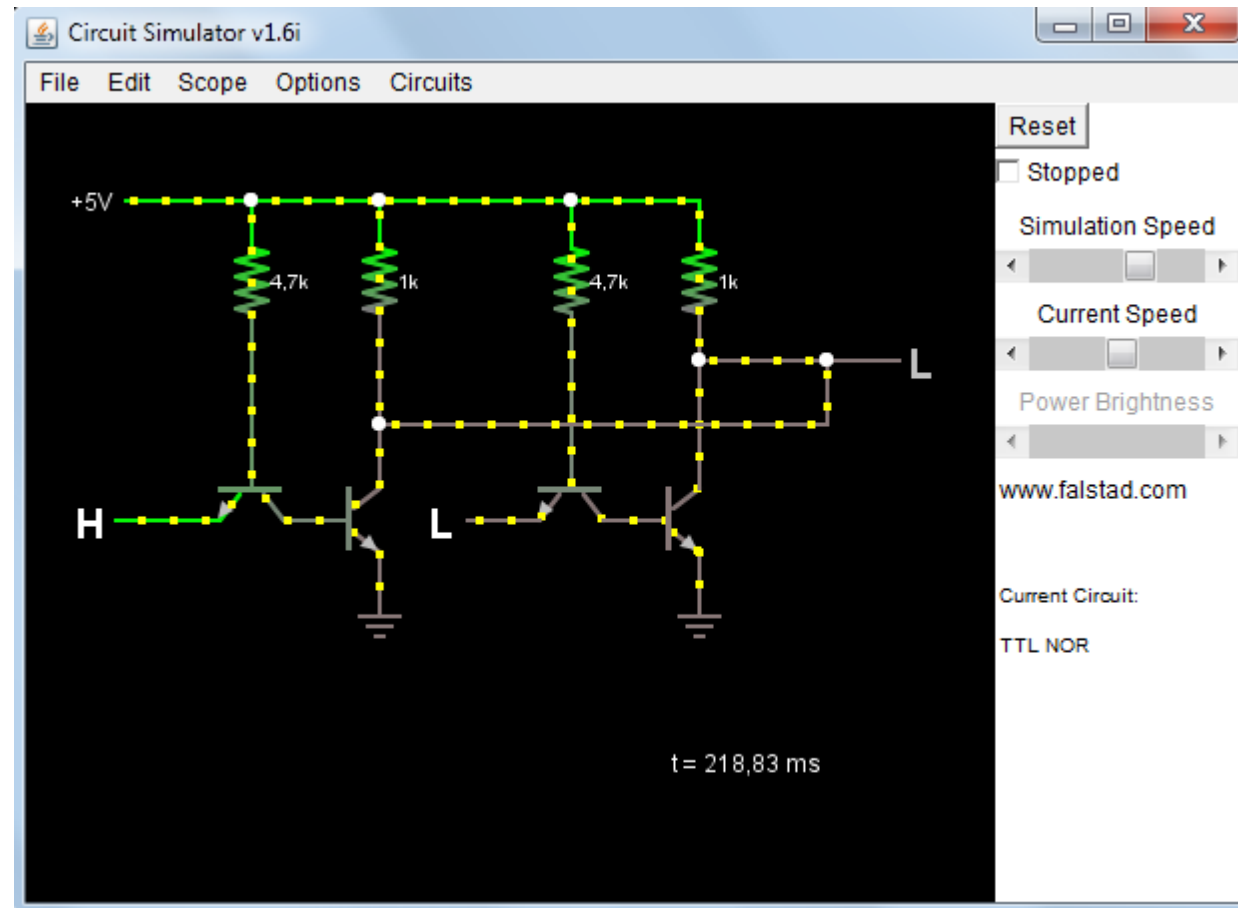


$$Y = \overline{A + B}$$



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

A kimenet csak akkor magas, ha minden bemenet alacsony szinten van

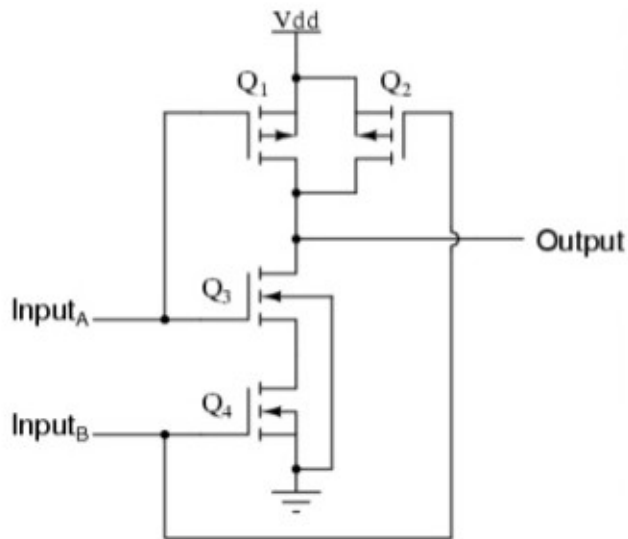


2 bemenetű TTL NOR (NEM-VAGY)

CMOS kapuk

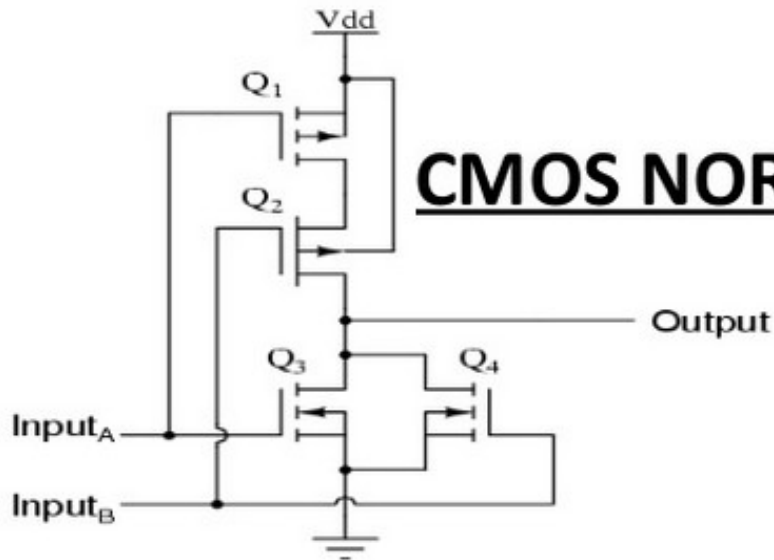
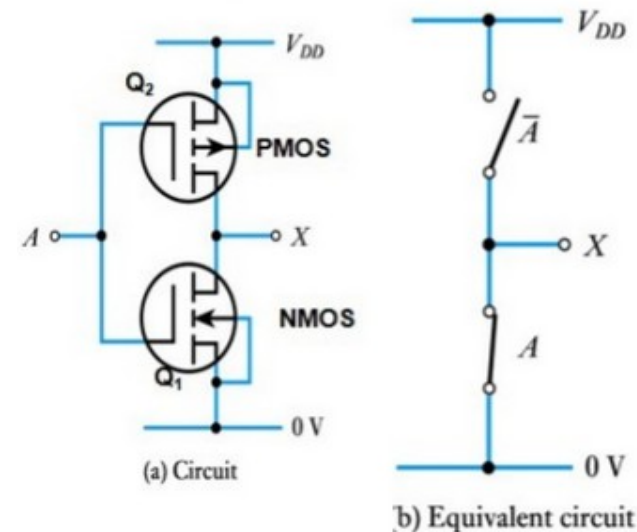
CMOS = Complementary MOS, p- és n-csatornás FET-ekből épülnek fel.

CMOS NAND



Input		Transistor				O/P
A	B	Q1	Q2	Q3	Q4	Y
0	0	ON	ON	OFF	OFF	1
0	1	ON	OFF	OFF	ON	1
1	0	OFF	ON	ON	OFF	1
1	1	OFF	OFF	ON	ON	0

CMOS inverter



CMOS NOR

Input		Transistor				Output
A	B	Q1	Q2	Q3	Q4	Y
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

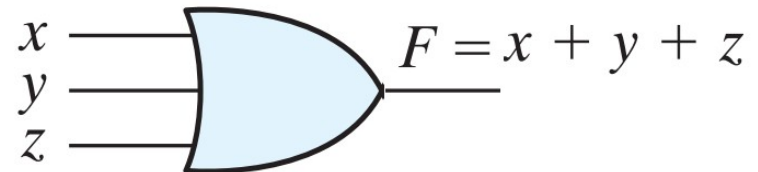
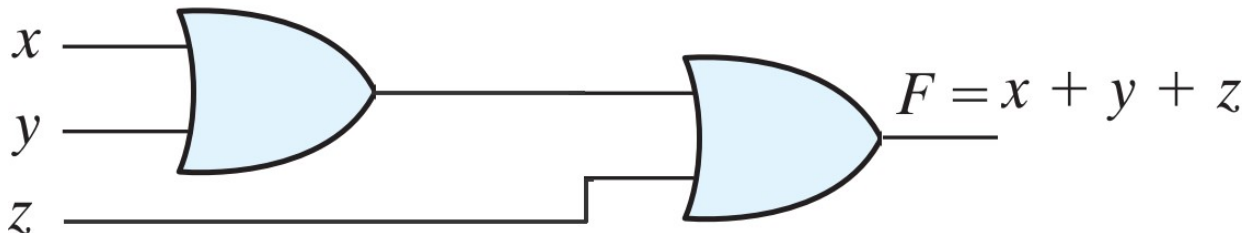
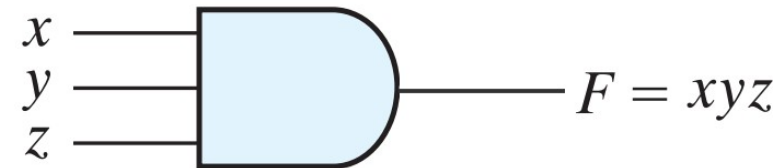
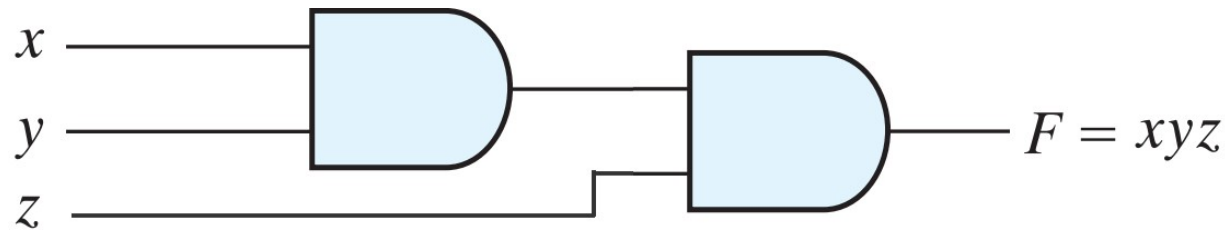
Több-bemenetű logikai kapuk

- Az **AND** (ÉS), illetve az **OR** (VAGY) műveletek asszociatív és kommutatív tulajdonságai miatt a kiterjesztés magától értetődő

Például az **OR** műveletre írhatjuk, hogy:

$$x + y = y + x \quad (\text{commutative})$$

$$(x + y) + z = x + (y + z) = x + y + z \quad (\text{associative})$$

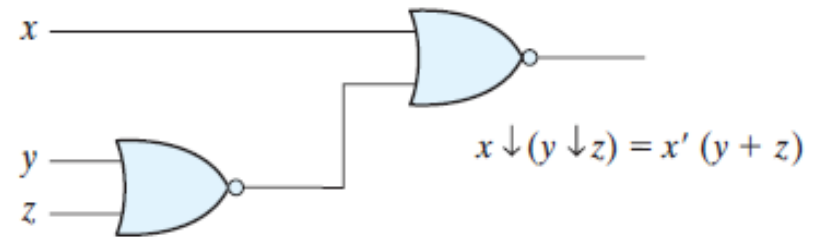
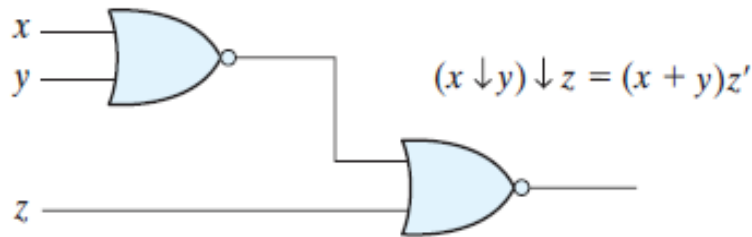


Több-bemenetű logikai kapuk

- A **NOR** és **NAND** műveleteknél, bár kommutatívak, gondot okoz, hogy nem asszociatívak. Például:

$$(x \downarrow y) \downarrow z = [(x + y)' + z]' = (x + y)z' = xz' + yz'$$

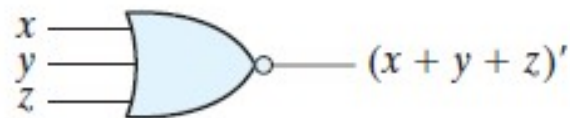
$$x \downarrow (y \downarrow z) = [x + (y + z)']' = x'(y + z) = x'y + x'z$$



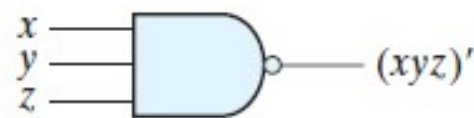
- Ezen a nehézségen úgy lehetünk úrrá, ha a több-bemenetű **NOR** (illetve **NAND**) kaput úgy definiáljuk, mint komplementált kimenetű, több-bemenetű **VAGY** (illetve **ÉS**) kapu:

$$x \downarrow y \downarrow z = (x + y + z)'$$

$$x \uparrow y \uparrow z = (xyz)'$$



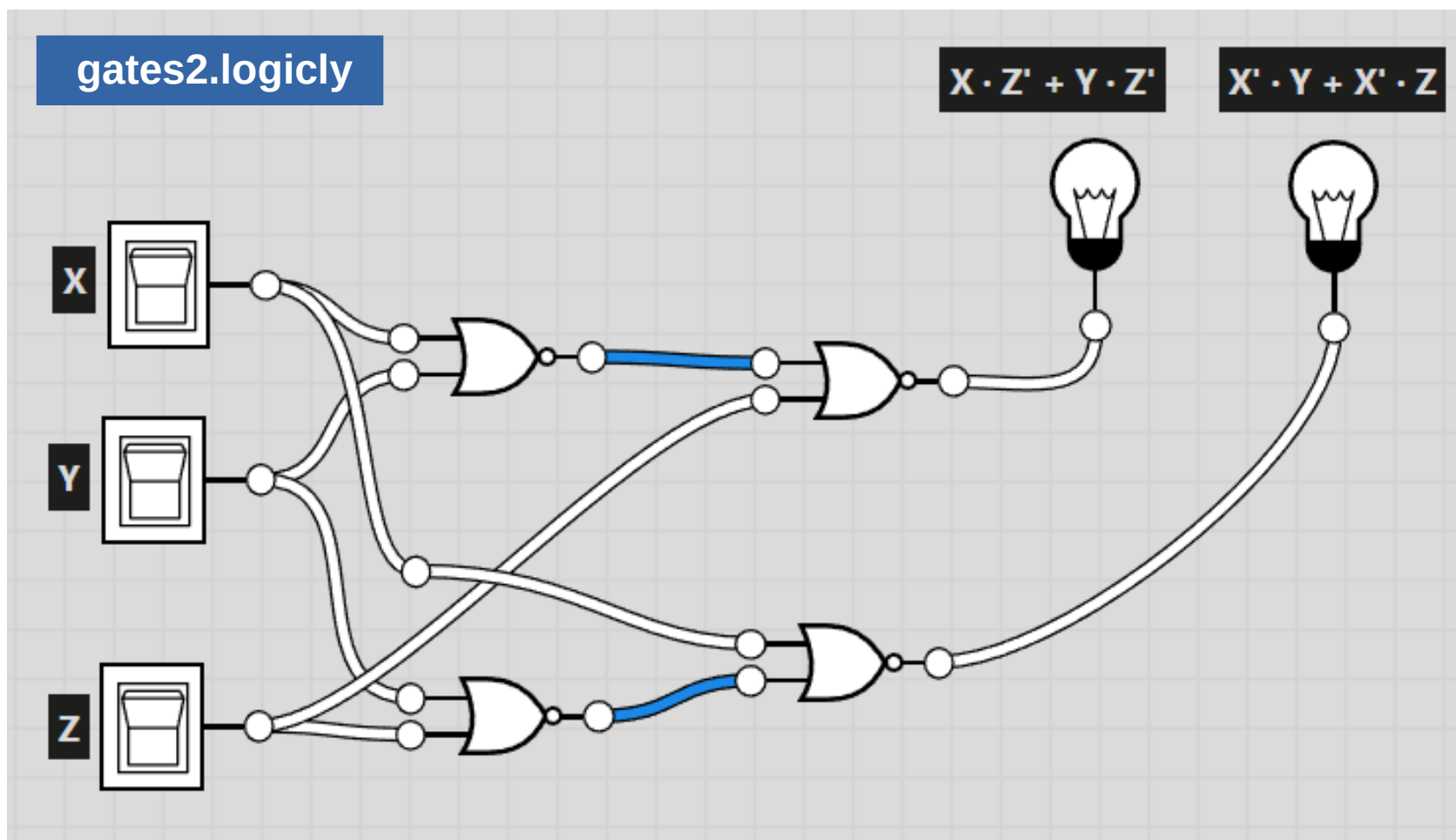
(a) 3-input NOR gate



(b) 3-input NAND gate

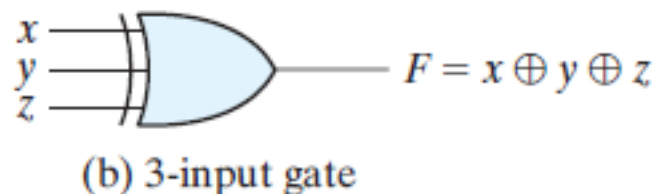
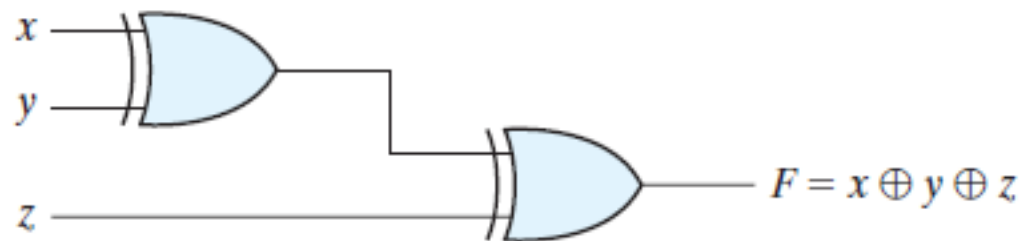
A NOR művelet nem asszociatív

- A **Logic.ly** szimulátor segítségével meggyőződhetünk róla, hogy az $((x \text{ NOR } y) \text{ NOR } z)$ és az $(x \text{ NOR } (y \text{ NOR } z))$ kifejezések különböző eredményre vezetnek



Több-bemenetű XOR kapu

- A **kizáró VAGY** (XOR) művelet kommutatív és asszociatív, így a több-bemenetű kiterjesztésének nincs elvi nehézsége. A gyakorlatban azonban nem terjedt el
- Azon ritka esetekben, amikor mégis szükség van rá, inkább több, kétbemenetű kapu összekapcsolásával valósítják meg
- Több változóra kiterjesztve az XOR művelet „páratlansági műveletként” értelmezhető: a kimenet '1', ha páratlan számú bemenet van '1' állapotban



x	y	z	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

(c) Truth table