



Számítógép Architektúrák (MIKNB113A)

6. előadás: Vezérlő egységek II. -
Programozható logikai eszközök

Előadó: Dr. Vörösházi Zsolt
voroshazi.zsolt@virt.uni-pannon.hu

Jegyzetek, segédanyagok:

- Könyvfejezetek:

- <http://www.virt.uni-pannon.hu>

- Oktatás → Tantárgyak → Számítógép
Architektúrák

- ([chapter05.pdf](#))

- Fóliák, óravázlatok .ppt (.pdf)

- Feltöltésük folyamatosan

Ismétlés: Vezérlő egységek fajtái:

- Eddig: I. Huzalozott (klasszikus módszerrel):
 - Mealy-modell,
 - Moore-modell
- II. Mikroprogramozott (reguláris vezérlési szerkezettel):
 - Horizontális mikrokódos vezérlő,
 - Vertikális mikrokódos vezérlő.
- **III. Programozható logikai eszközök (PLD-k = Programmable Logic Devices):**
 - **PLA, PAL, PROM, GAL, CPLD;**
 - (FPGA – nem tárgyaljuk)

Ismétlés: Kombinációs hálózatok

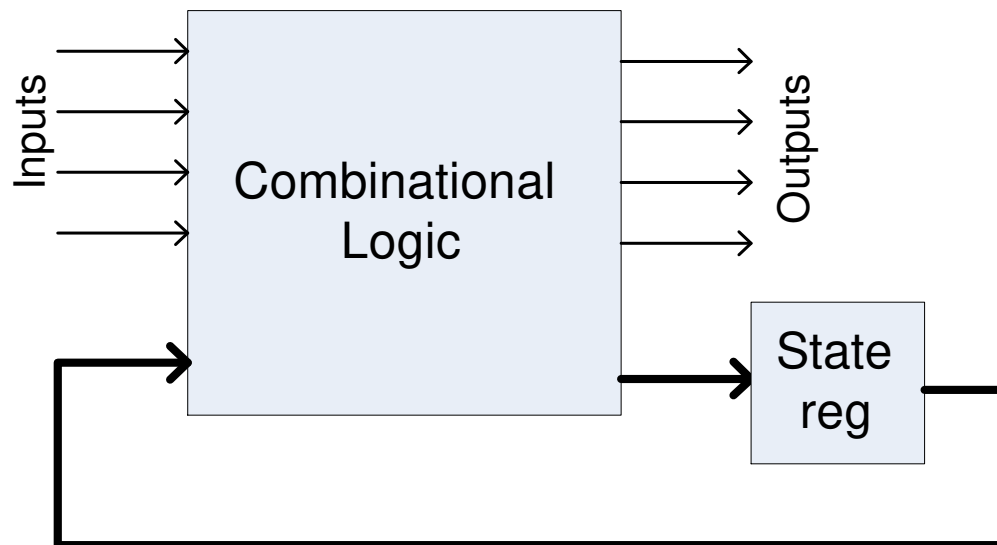
- **(K.H.) Kombinációs logikai hálózatról** beszélünk: ha a mindenkori kimeneti kombinációk értéke csupán a bemeneti kombinációk – „elsődleges kombinációk” – pillanatnyi értékétől függ (tároló „kapacitás”, vagy memória nélküli hálózatok).



Sorrendi hálózatok:

- **(S.H.) Sorrendi (szekvenciális) logikai hálózatról** beszélünk: ha a mindenkori kimeneti kombinációt, nemcsak a pillanatnyi bemeneti kombinációk, hanem a korábban fennállt bemeneti kombinációk és azok sorrendje is befolyásolja. (A szekunder/másodlagos kombinációk segítségével az ilyen hálózatok képessé válnak arra, hogy az ugyanolyan bemeneti kombinációkhoz más-más kimeneti kombinációt szolgáltatassanak, attól függően, hogy a bemeneti kombináció fellépésekor, milyen értékű a szekunder kombináció, pl. a State Register tartalma)

Vezérlő
egységek
alapjául szolgáló
hálózat!

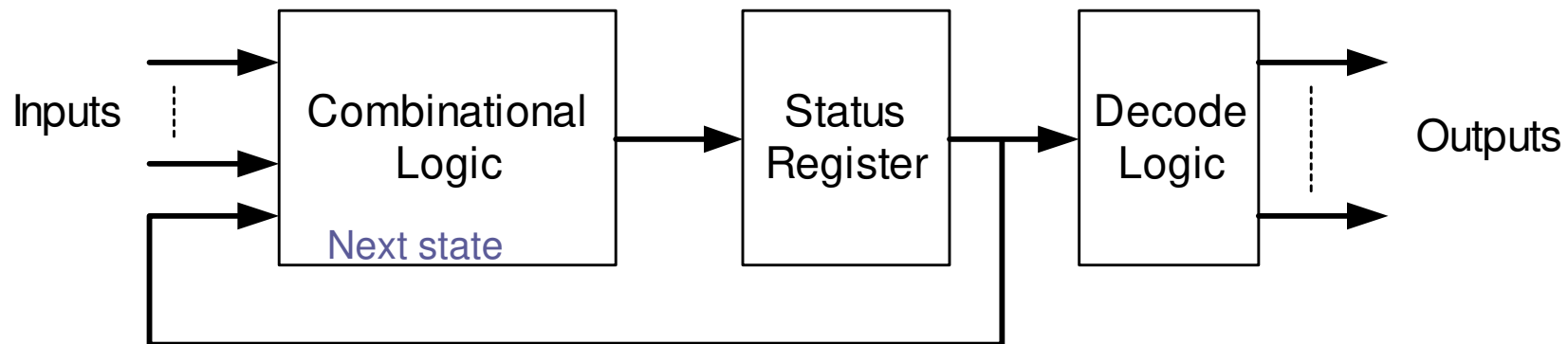




Programozható logikai eszközök (PLD)

Állapotgép (FSM) tervezés tulajdonságai

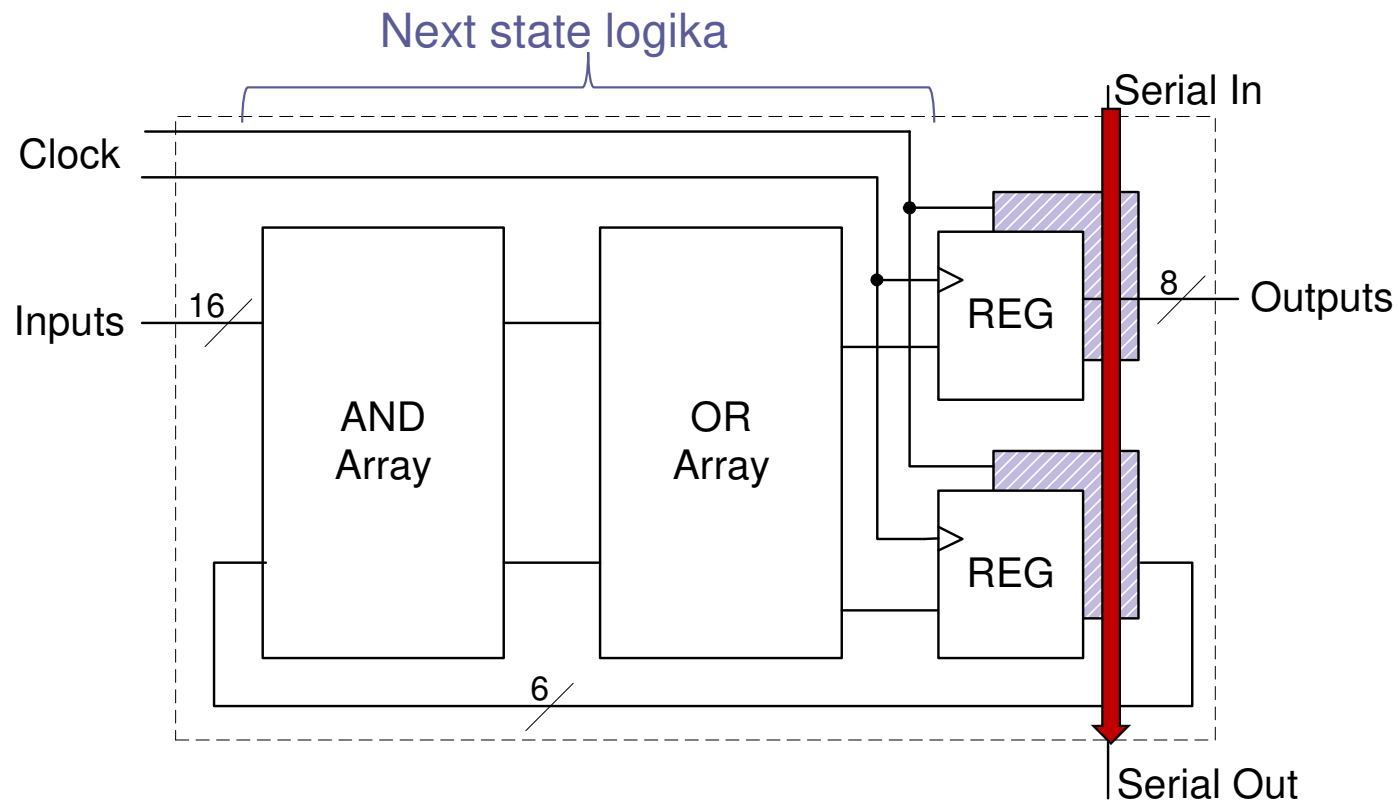
- Két kombinációs logikai hálózatról és egy regiszterből áll
- Tervezés során az állapot-átmeneteket vesszük figyelembe, **DE**
- Hibavalószínűség nagy,
- Szimulációs eszközök (CAD Tools) hiánya,
- Hibák lehetségesek a prototípus fejlesztése során is,
- Könnyen konfigurálható / flexibilis eszközök szükségesek
→ mindezek használunk programozható alkatrészeket



Field Programmable Logic Sequencer (FPLS) – Logikai sorrendvezérlő

- Egy vezérlő egység Next-State logikai blokkjának megvalósítása a tervező, gyártó feladata. Általában változó logikát használnak a függvények megvalósításánál.
- A felhasználó által programozható logikai sorrendvezérlő (**Field Programmable Logic Sequencer**) programozható alkatrészekből építhető fel, amelyek a következőkben részletesen ismertetésre kerülnek.

Field Programmable Logic Sequencer (FPLS) – Logikai sorrendvezérlő

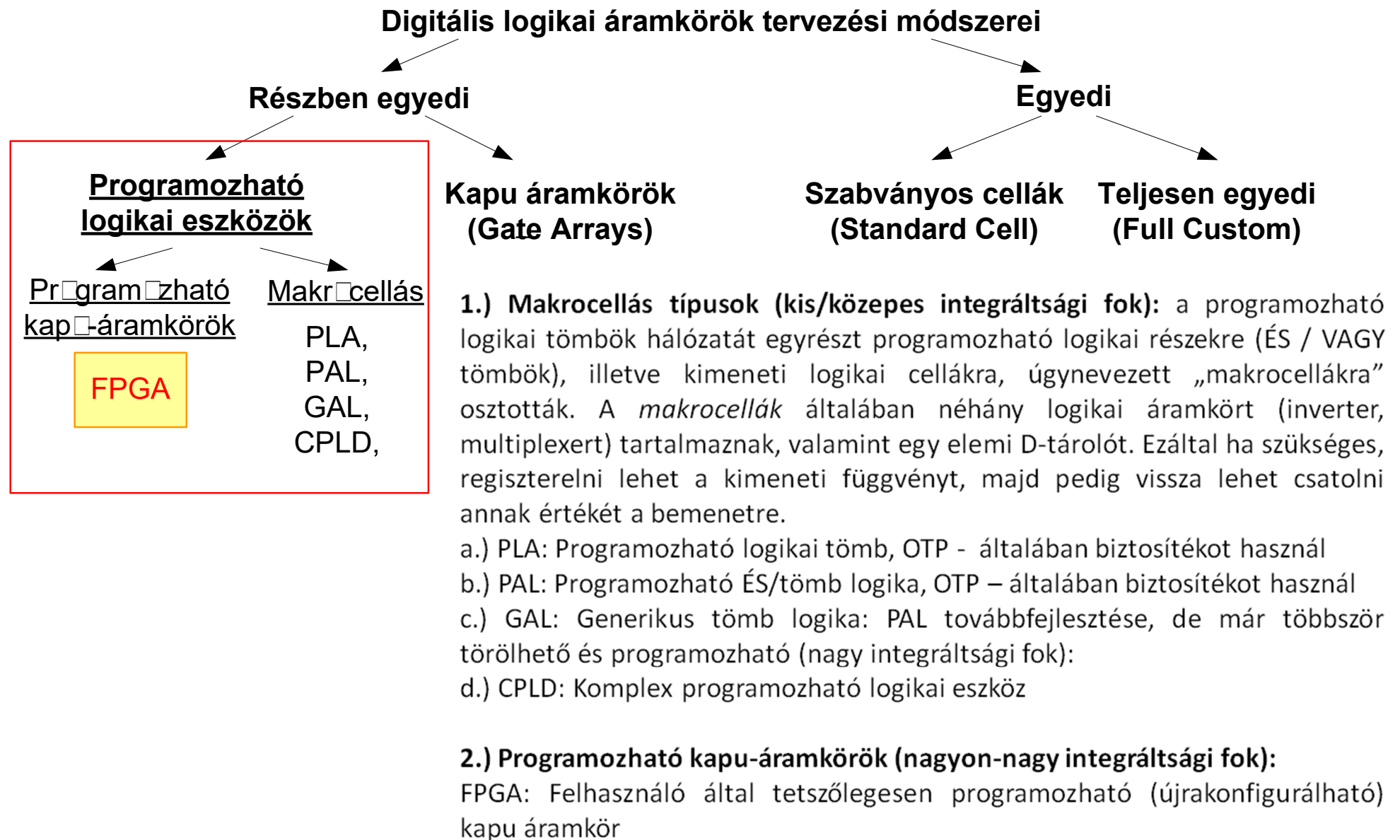


Működése:

- Normál (D-FF),
- Debug (Shift Reg.)

A logikai sorrendvezérlőnek 16 külső bemeneti vonala van, 1 RESET és 1 kimenet-engedélyező vonala, ill. 8 kimeneti vonala. A regiszterek egy-egy állapotot tárolnak, amelyek az órajel hatására a kimenetre íródnak, vagy a 6 belső, *visszacsatoló* vonalon keresztül visszacsatolódnak. A Next-State ill. a kimeneti szintek meghatározásánál programozható AND/OR tömböket használnak.

Tervezési módszerek



PLD – Programozható logikai eszközök

- A Programozható logikai áramköröket (PLD: Programmable Logic Devices) általánosan a kombinációs logikai hálózatok és sorrendi hálózatok tervezésére használhatjuk. Azonban míg a hagyományos kombinációs logikai hálózatok dedikált összeköttetésekkel, illetve kötött funkcióval (kimeneti függvény) rendelkeznek, addig a programozható logikai eszközökben pontosan ezek változtathatók, az alábbi lehetséges módokon:
 - A felhasználó által *egyszer programozható* / konfigurálható logikai eszközök (**OTP**: One Time Programmable), amelynél a gyártás során nem definiált funkció egyszer még megváltoztatható (ilyenek pl. a korai PAL, PLA eszközök)
 - **Többször**, akár *tetszőleges módon* programozható logikai eszközök = *újraconfigurálható* (ilyenek pl. a korábbi GAL, vagy a mai modern CPLD, **FPGA** eszközök)

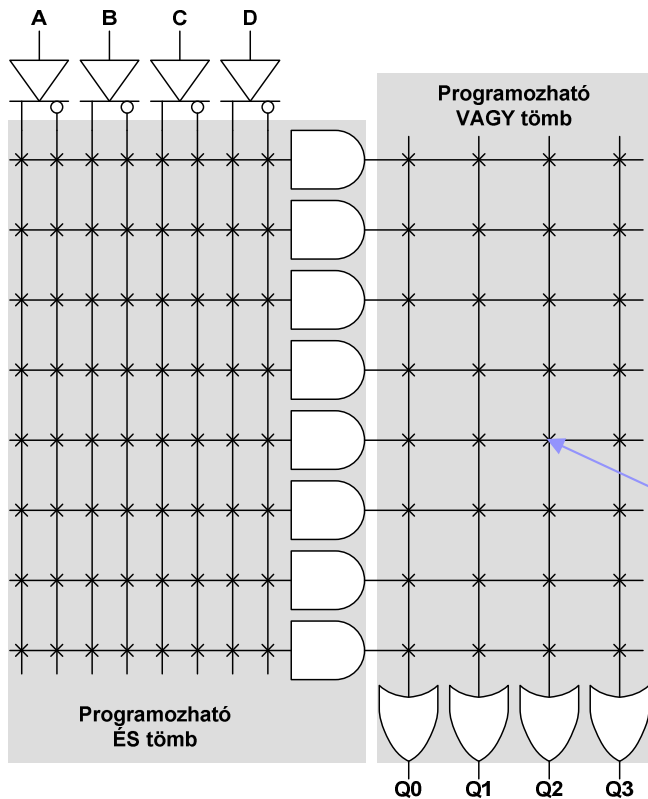
Programozható logikai eszközök (PLD-k) két fő típusa:

- 1.) **Makrocellás** PLD-k: (Programmable Logic Devices):
 - PLA
 - PAL
 - GAL
 - CPLD
- 2.) **FPGA** (Field Programmable Gate Array): Programozható Gate Array áramkörök
 - **XILINX** (Spartan, Virtex, Kintex, Artix) ~ 53 % !
 - **Altera/Intel-FPGA** (Stratix, Arria, Cyclone), ~ 36 %
 - MicroSemi-Actel (pl. úrkutatásban),
 - Lattice sorozatok,
 - QuickLogic,
 - További kisebb gyártók termékei.

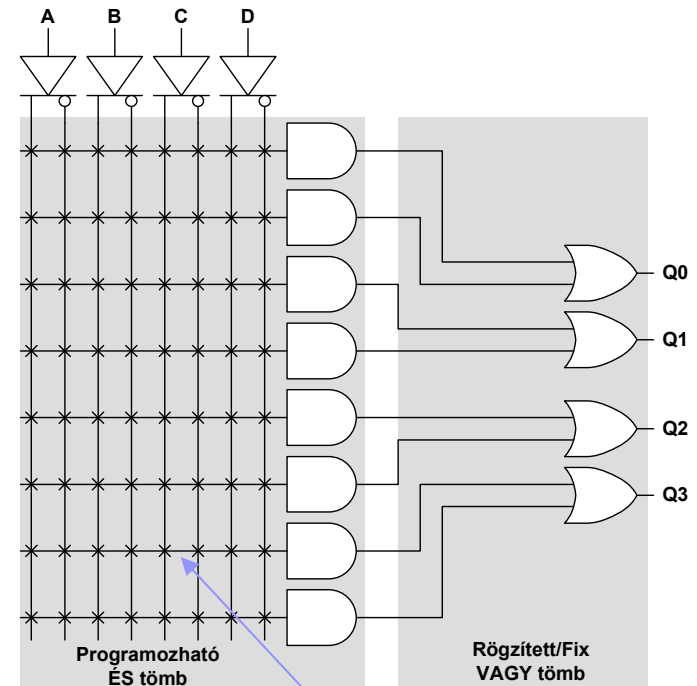


Makrocellás PLD-k

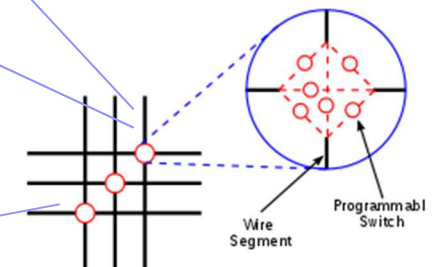
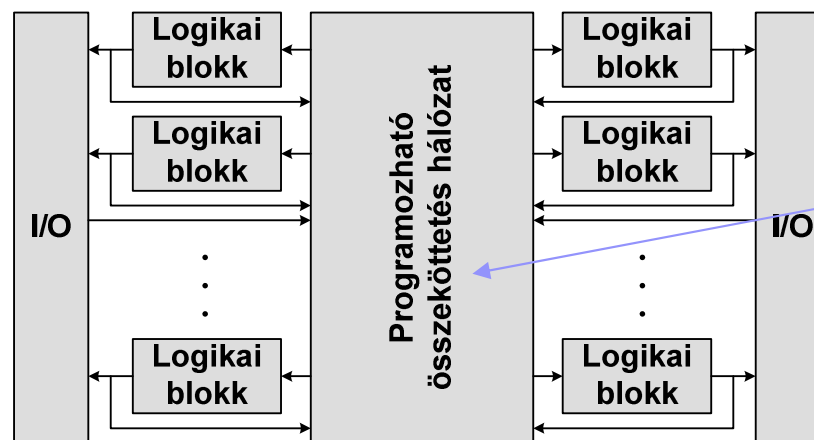
PLA



PAL

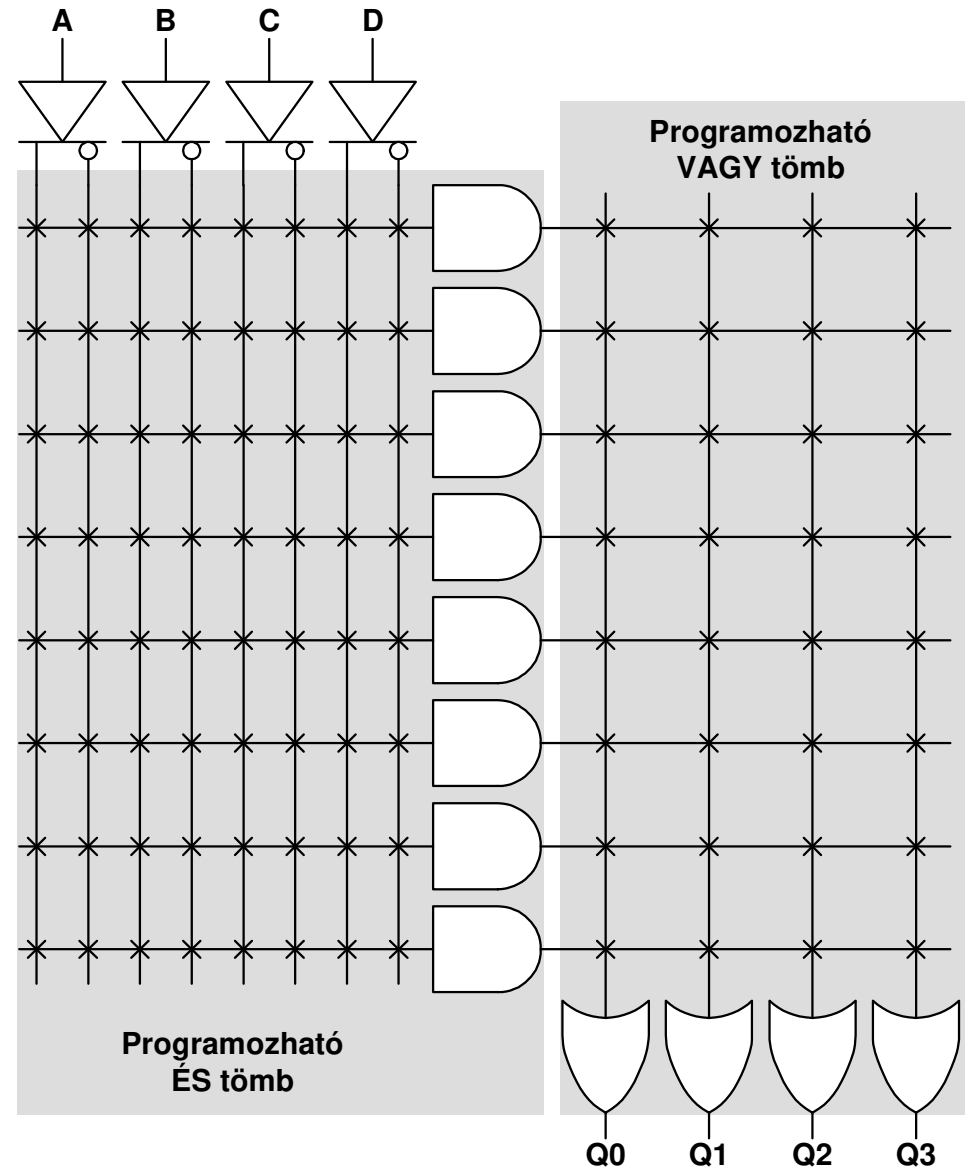


CPLD



Programmable Logic Array (PLA)

- Mindkét része (AND, OR) programozható
- Bármely kombinációja az AND / OR-nak előállítható
- Mintermek OR kapcsolata (DNF)
- Programozható kapcsolók a horizontális/ vertikális vonalak metszésében
- Q_n Kimeneteken D tárolók! (visszacsat. a bemenetekre)



PLA

- 1970-ben, a TI (Texas Instruments) által kifejlesztett eszköz mindkét részhálózata (ÉS, illetve VAGY tömb) programozható összeköttetéseket tartalmazott, amelyek segítségével tetszőleges mintermek tetszőleges VAGY kapcsolata előállítható (DNF alakot), ezáltal bármilyen kombinációs logikai hálózat realizálható volt (természetesen adott bemenet, ill. kimenet szám mellett).
- A programozható ÉS / VAGY tömbökben úgynevezett „programozható kapcsolók” vannak elhelyezve a horizontális/ vertikális vonalak metszéspontjában.
- Amennyiben a Q_n kimenet(ek)re tárolókat kötünk (pl. egyszerű D tárolót), majd pedig visszacsatoljuk a programozható logikai hálózat bemenete(i)re akár egy *sorrendi hálózati viselkedést* is meghatározhatunk.

Programozásuk (Fuse) biztosítékok segítségével

- Az összeköttetés mátrix metszéspontjaiban akár kis biztosítékok (fuse) helyezkednek el. Gyárilag logikai '1'-est definiál, tehát vezetőképes. Ha valamilyen spec. programozó eszközzel, a küszöbnél nagyobb feszültséget kapcsolunk rá, átégethető, tehát szigetelővé (nem-vezető) válik, és logikai '0'-át fog reprezentálni.
- A biztosíték átégetése, csak egyszer lehetséges, utána már csak a programozott állapotot fogja tárolni (**OTP** – One time programmable IC).

Példa: PLA tervezése

- Realizálja a következő függvényeket:

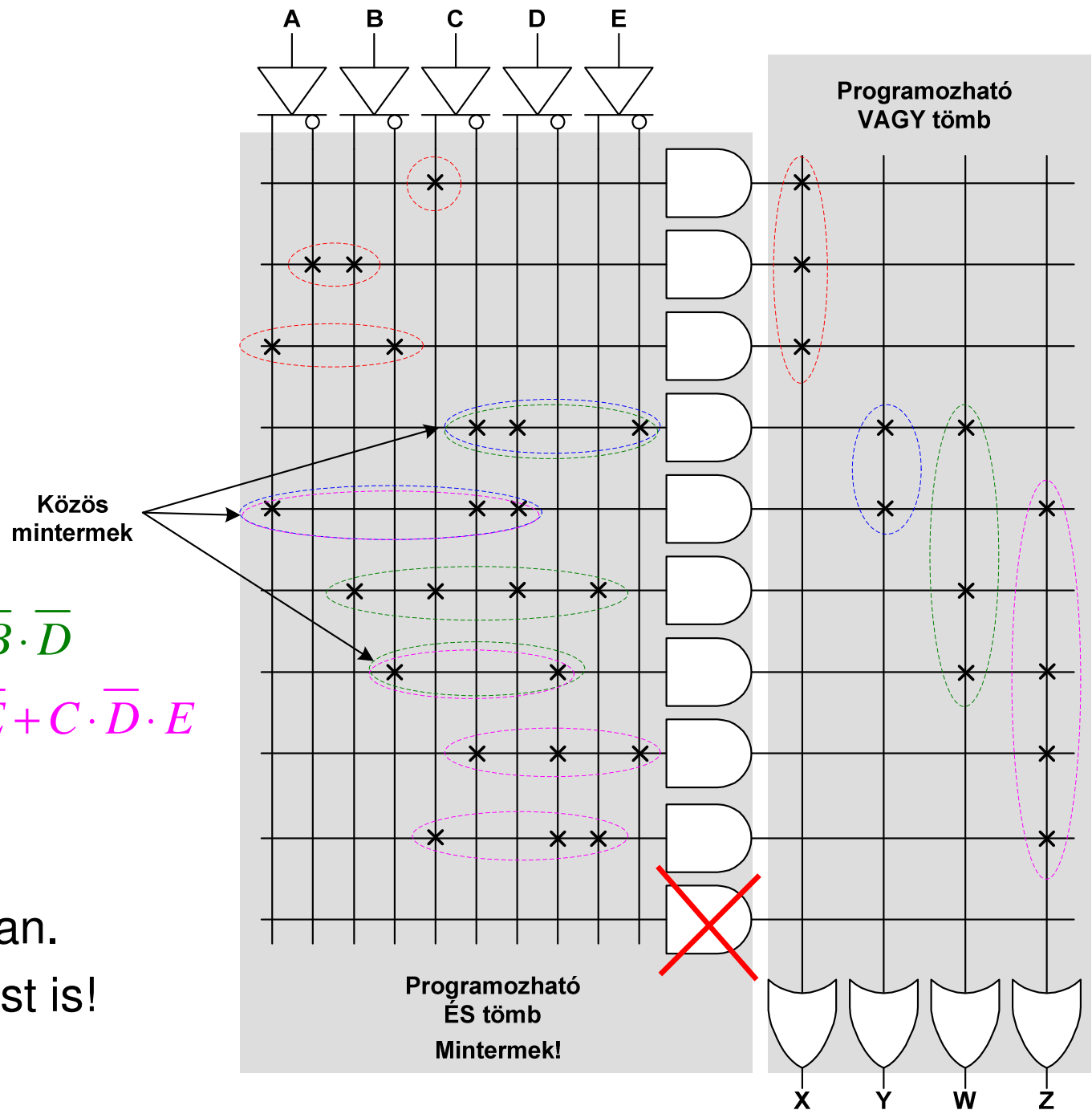
$$X = C + \bar{A} \cdot B + A \cdot \bar{B}$$

$$Y = \bar{C} \cdot D \cdot \bar{E} + A \cdot \bar{C} \cdot D$$

$$W = \bar{C} \cdot D \cdot \bar{E} + B \cdot C \cdot D \cdot E + \bar{B} \cdot \bar{D}$$

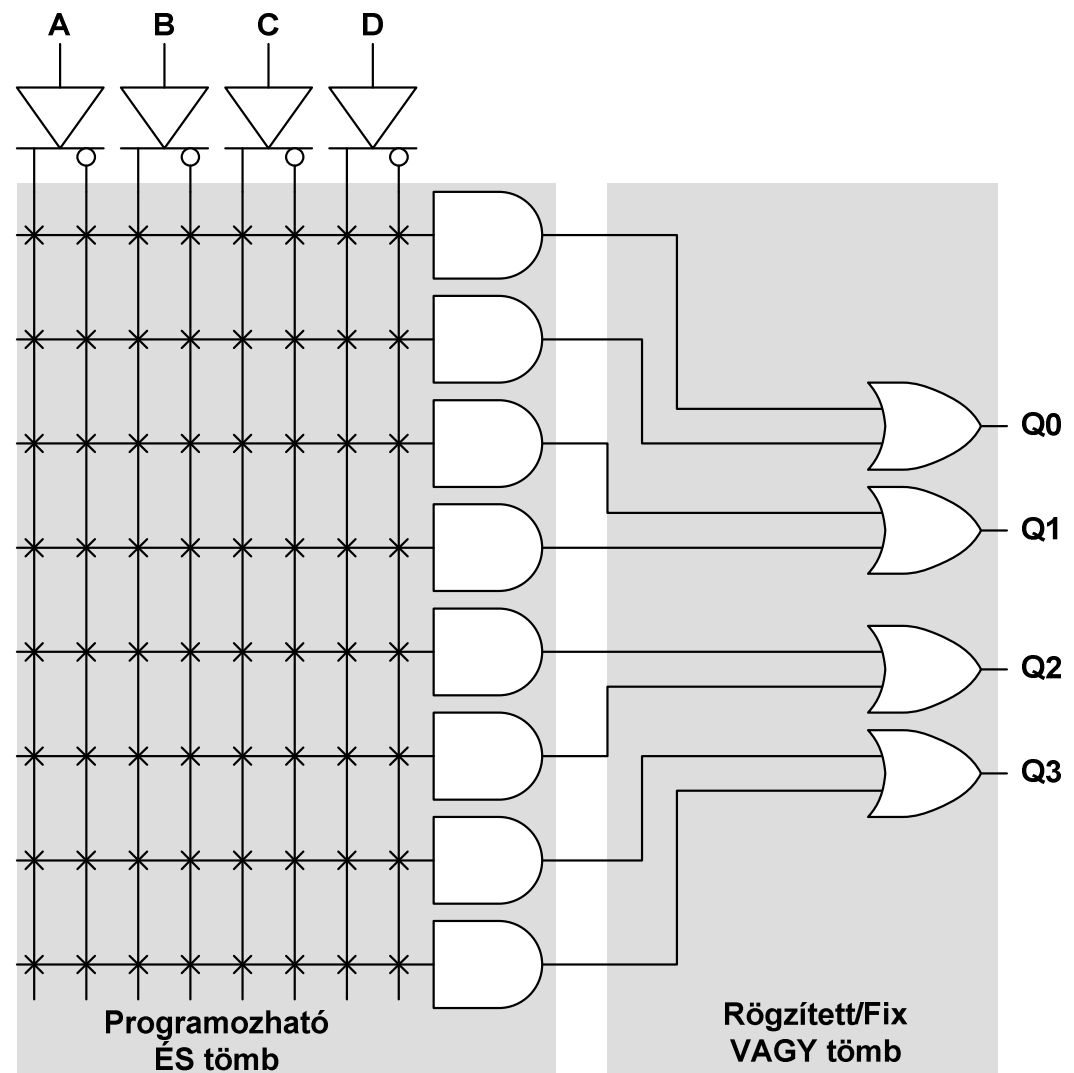
$$Z = A \cdot \bar{C} \cdot D + \bar{B} \cdot \bar{D} + \bar{C} \cdot \bar{D} \cdot \bar{E} + C \cdot \bar{D} \cdot E$$

- Tehát 5 bemenete (A,B,C,D,E) és
- 4 kimenete (X,Y,W,Z) van.
- Rajzoljuk fel a kapcsolást is!



Programmable AND Logic (PAL)

- Egy programozható rész - AND / míg az OR fix
- Véges kombinációja áll elő az AND / OR kapcsolatoknak
- Metszéspontokban kevesebb kapcsoló szükséges
- Gyorsabb, mint a PLA
- Q_n kimeneteken D tárolók (visszacsatolódhatnak a bemenetekre)

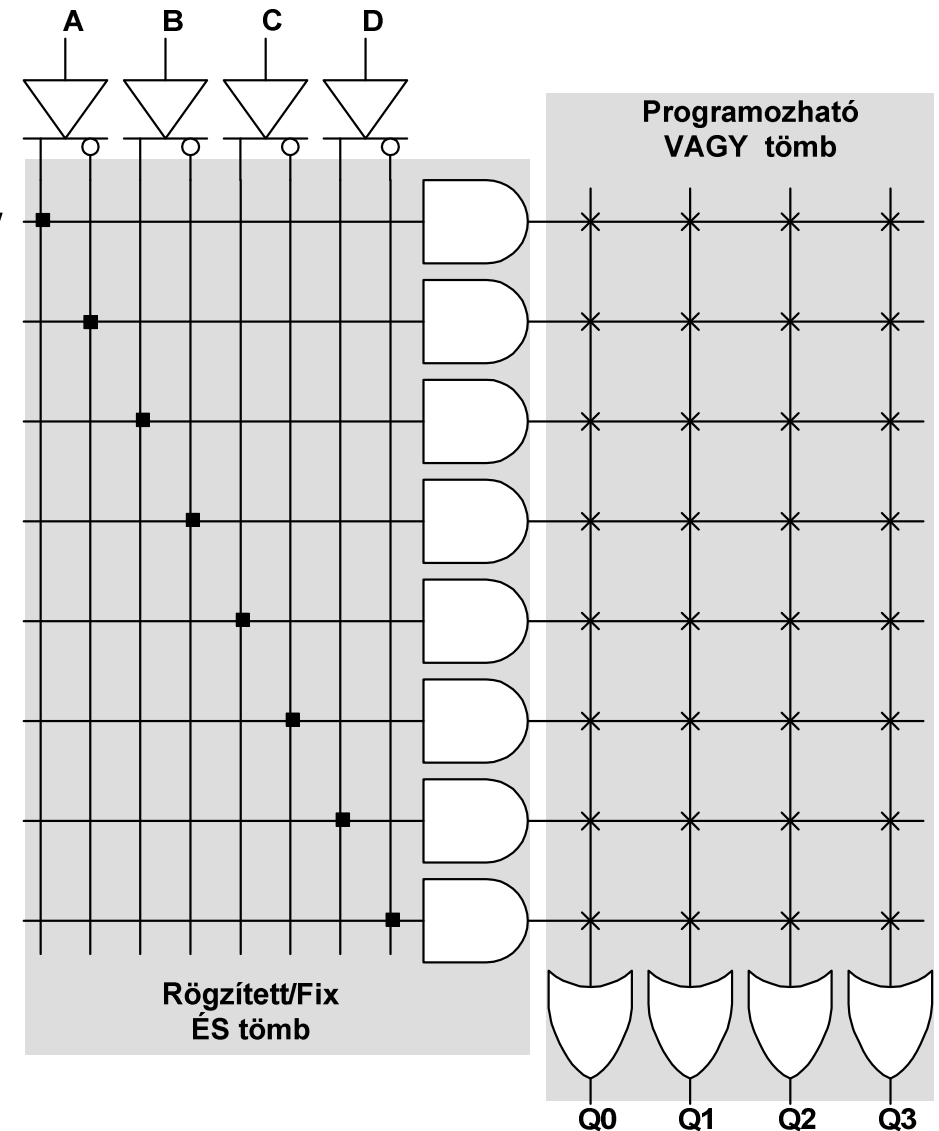


PAL

- Elsőként, 1978-ban az MMI (Monolithic Memories Inc.) jelent meg ilyen programozható eszközökkel, majd pedig későbbi jogutódja a Lattice Semiconductor, illetve az AMD a 80'-as évek végén.
- A PAL hálózatban a programozható részt az ÉS tömb jelenti, míg az VAGY tömb fix/rögzített. Így a tetszőleges mintermeknek csak egy véges kombinációja (VAGY) állítható elő: a lehetséges kimeneti függvények variálhatóságából veszítünk, cserébe viszont a VAGY részek dedikált útvonalainak jelterjedési sebessége nagyobb, míg az eszköz mérete kisebb és ezáltal olcsóbb is lesz.
 - Ezáltal a metszéspontokban kevesebb kapcsoló szükséges („gyorsabb”, mint a PLA). Hasonlóan a PLA-khoz, amennyiben a Q_n kimenet(ek)re tárolókat kötünk (pl. egyszerű D tárolót), majd pedig visszacsatoljuk a programozható PAL logikai hálózat bemenete(i)re akár sorrendi hálózati viselkedést is könnyen valósíthatunk.

Programmable Read Only Memory (PROM)

- Egy programozható rész - OR / míg az AND fix (vmilyen struktúra szerint)
- Véges kombinációja áll elő az AND / OR kapcsolatoknak
- Metszéspontokban kevesebb kapcsoló szükséges
- Gyorsabb, mint a PLA
- Q_n kimeneteken D tárolók! (visszacsatolódhatnak a bemenetekre)



x: programozható
■: fix/rögzített

GAL (Generic Array Logic): Általános tömb logika

- 1985-ben a Lattice Semiconductor fejlesztette ki elsőként,
 - amely a *PAL*-nak egy továbbfejlesztett változatát képviseli.
 - Ugyanolyan belső struktúrával rendelkezik, mint egy *PAL* áramkör,
 - azonban többször programozható: tehát törölhető és újraprogramozható eszköz.
 - EEPROM technológiát (lásd. lebegő-gate) alkalmaz. Később a National Semiconductor, és AMD is megjelent saját *GAL* sorozataival a piacon.

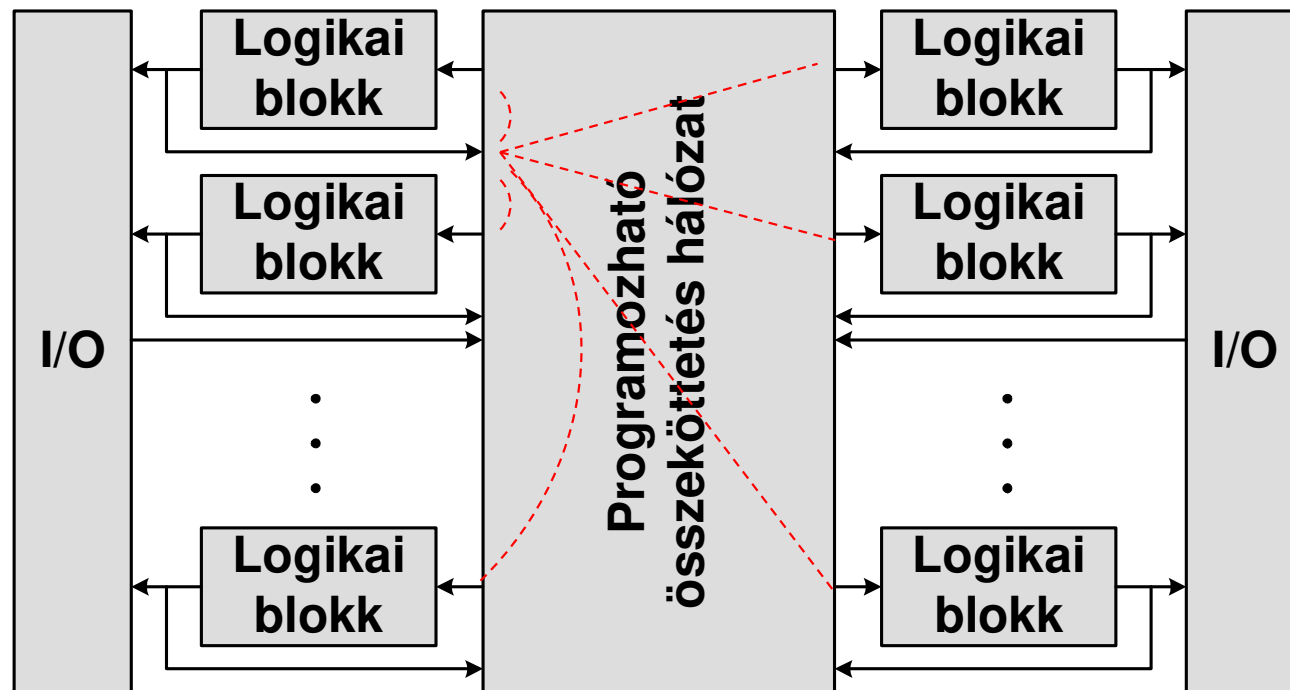
CPLD

CPLD (Complex Programmable Logic Devices): Komplex-programozható Logikai eszközök

- Valójában *átmenetet* képeznek a kis/közepes integráltsági fokú makrocellás PLD-k GAL/PAL áramkörei, illetve a nagy integráltsági fokú FPGA kapu-áramkörök között.
- A GAL/PAL áramköröktől architektúráisan annyiban különbözik, hogy ki lett bővítve: nem egy-, hanem több logikai cellamátrixot tartalmaz, amelyek konfigurálható blokkok reguláris struktúrájában vannak elrendezve. A mai modern FPGA áramköröktől viszont az különbözteti meg felépítésben, hogy *nem tartalmaz dedikált* erőforrásokat (pl. szorzók, memória blokkok).
- a legnagyobb gyártók, amelyek jelenleg is aktív szereplői a CPLD-k piacának a következők: Xilinx, Altera, Lattice Semiconductor, Atmel stb.

Complex Programmable Logic Device (CPLD)

- 1 Logikai Blokkon belül:
 - ~ PAL / PLA (K.H.)
 - Regiszterek (D-FF)
- I/O Blokkok
- Programozható összeköttetések (PI: Programmable Interconnection)
 - Teljes (Full-crossbar), vagy
 - Részleges összeköttetés hálózat



CPLD (folyt)

A CPLD-kben található Logikai Blokk-ok (=makrocellák):

- egyrészt *logikai kapuk* tömbjeit tartalmazzák (hasonlóan a PAL/GAL áramkörök felépítéséhez – DNF alak),
 - másrészt *regisztereket* (D-tárolókból) tartalmaznak a logikai tömbök által előállított kimenetek átmeneti tárolásához, valamint
 - *multiplexereket*, mellyel a programozható összeköttetés hálózatra, vagy I/O blokkok celláihoz lehet továbbítani a belső Logikai Blokkok által előállított kimeneti értékeket. Ezáltal nemcsak logikai kombinációs hálózatokat, hanem akár sorrendi hálózatokat is egyszerűen megvalósíthatunk CPLD-k segítségével
-
- A CPLD-kben található *Programozható összeköttetés hálózat*
 - teljes összeköttetést (mindenki-mindekivel), vagy
 - részleges összeköttetést (valamilyen struktúra szerint, pl. bemenetet – kimenettel, főként régi CPLD típusok esetén) biztosít az egyes blokkok között.
 - Kikapcsoláskor a CPLD konfigurációs memóriája megtartja értékét (non-volatile típus), ezért nem kell egy külső pl. ROM memóriát használni az inicializációs minták tárolásához, bekapcsoláskor ezek automatikusan betöltésre kerülnek. A CPLD-eket közkedvelten alkalmazzák különböző interfészek jeleinek összekapcsolásához (*glue-logic*), amennyiben a jeleken átalakításra is szükség van, továbbá áraik az FPGA-k árainál jóval kedvezőbbek



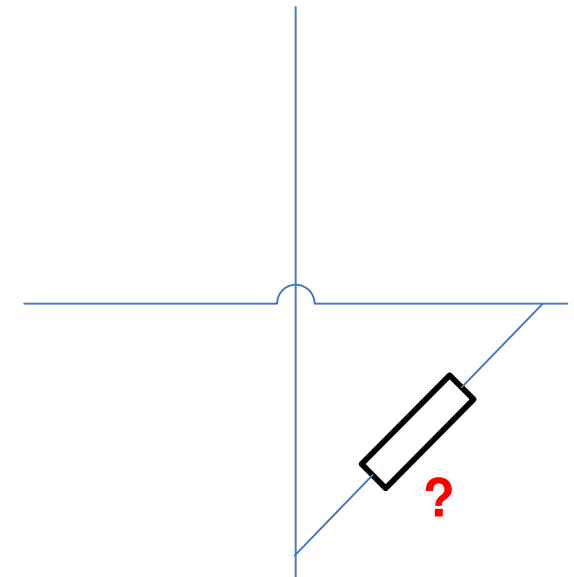
Hogyan programozhatók a
VLSI alkatrészek?

Programozási technikák

Programozási technikák

Mi van a programozható összeköttetések csomópontjaiban, illetve milyen módszerrel programozhatóak?

- a.) **SRAM**
- b.) **MUX**
- c.) **Antifuse**
- d.) **Floating Gate**
- e.) **EPROM/EEPROM/Flash**



$$\tau = R \cdot C$$
$$[ns / ps]$$

Programozás = konfigurálás

Konfigurálás (PLD/FPGA esetén) – mielőtt az eszközt használni szeretnénk egy speciális (manapság általában JTAG szabványú) programozó segítségével „fel kell programozni”: le kell tölteni a konfigurációs állományt (bitfájl, vagy object fájl). A programozás a legtöbb PLD esetében a belső programozható összeköttetésének fizikai típusától függően azok beállításával történik. A programozható összeköttetésekben a következő lehetséges alkatrészek találhatók:

- **Biztosíték (Fuse):** átégetésük után nem visszafordítható a programozási folyamat (OTP). Korábban a PAL eszközök népszerű kapcsoló elemeként használták.
- **Antifuse technológia:** (OTP), az antifuse-os kristályszerkezetű kapcsoló elem 'átolvasztása' után egy nagyon stabilan működő összeköttetést kapunk, amely sajnos szintén nem visszafordítható folyamatot jelent. A technológia drága, az előállításához szükséges maszk-rétegek nagy száma miatt, nagyon jó zavarvédetség elérése érdekében használják (pl. úrkutatás).
- **SRAM cella + tranzisztor:** tetszőlegesen programozható (FPGA-k esetén legelterjedtebb kapcsolás-technológia), az SRAM-ban tárolt inicializáló értéktől függően vezéri a tranzisztor gate-elektrodáját
- **SRAM cella + multiplexer:** tetszőlegesen programozható az SRAM cellában tárolt értéktől függően (kiválasztó jel) vezérelhető a multiplexer
- **Lebegő kapus tranzisztor (Floating Gate) technológia:** elektromosan tetszőlegesen programozható, a mai EEPROM/Flash technológia alapja. **27**