



**BUDAPESTI MŰSZAKI ÉS GAZDASÁGTUDOMÁNYI EGYETEM**  
**VILLAMOSMÉRNÖKI ÉS INFORMATIKAI KAR**  
**MÉRÉSTECHNIKA ÉS INFORMÁCIÓS RENDSZEREK TANSZÉK**

# **Digitális technika (VIMIAA02)**

## **4. laboratórium**

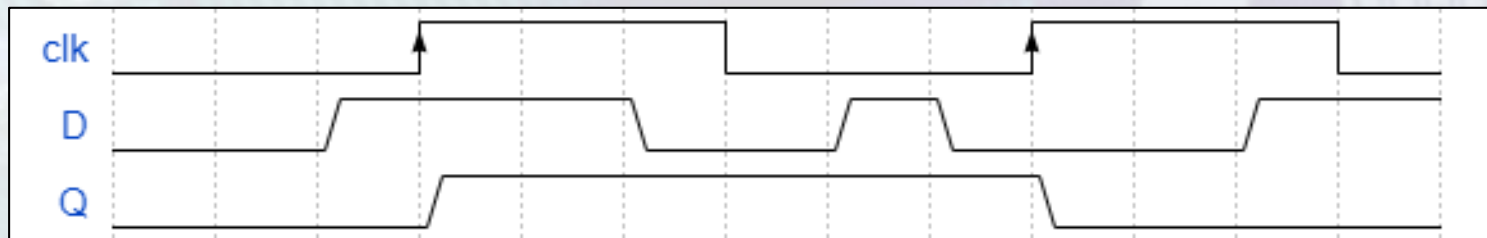
**Raikovich Tamás**  
**BME MIT**

# Regiszterek, állapotgépek (FSM)

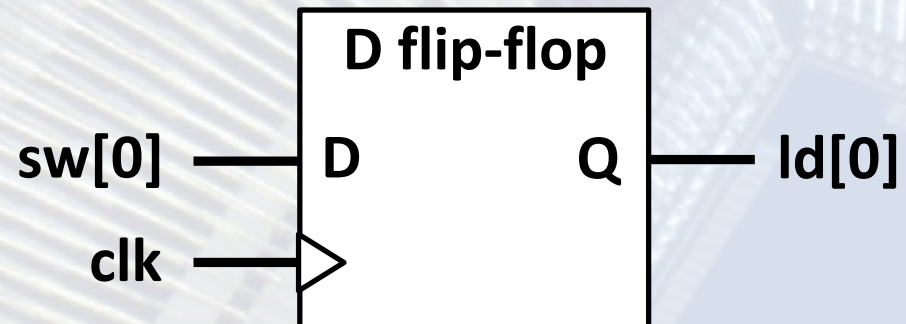
- **1. feladat: flip-flop, regiszterek**
  - a) D flip-flop
  - b) 8 bites regiszter
  - c) Törölhető és engedélyezhető 8 bites regiszter
- **2. feladat: állapotgép (FSM)**
  - a) Folyamatos LED villogtatás 1/3 kitöltési tényezővel
  - b) A villogás legyen engedélyezhető
- **A feladatokhoz szükséges új Verilog ismeretek:**
  - ***always*** blokk szinkron sorrendi hálózatok leírásához
  - Lokális paraméterek (***localparam***)
  - Relációs operátorok

# 1.a feladat: D flip-flop

- D flip-flop: élvezérelt tároló, a bemenet beírása az órajel felfutó élének hatására történik meg

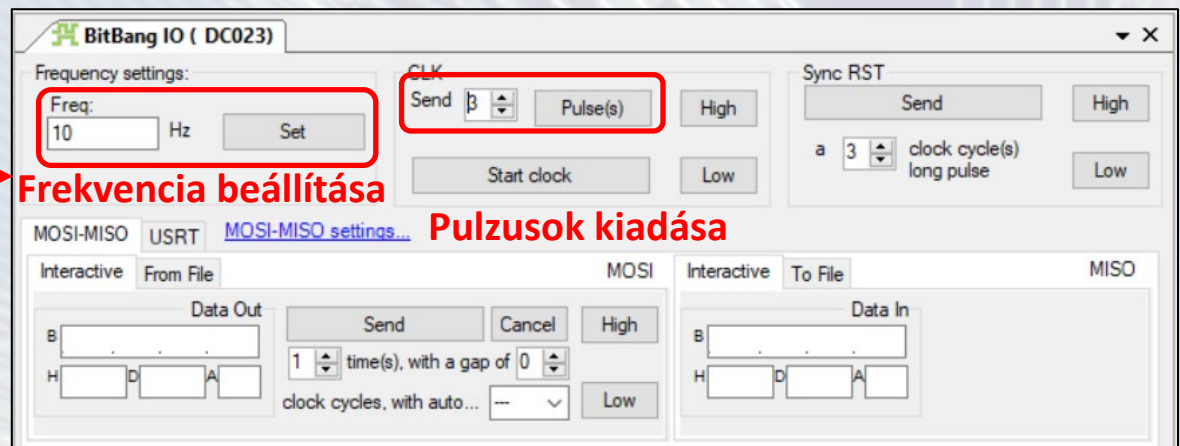
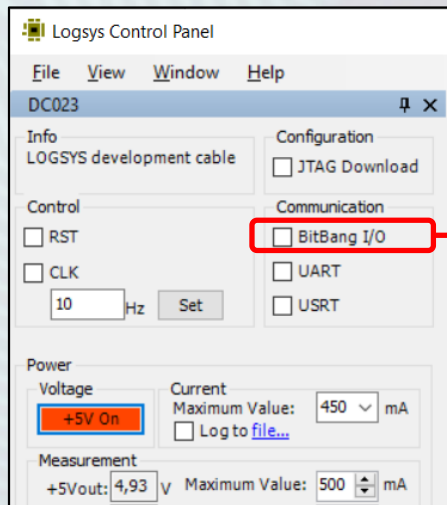


- A bemenetek és kimenetek bekötése
  - Órajel (clk): a letöltőkábeltől (*bitbang I/O*)
  - Bemenet (d): sw[0]
  - Kimenet (q): ld[0]



# 1.a feladat: D flip-flop

- Egészítsük ki a tesztkörnyezet vázát
  - Hogy tudunk itt órajelet előállítani?
  - Vegyük alapul az előző idődiagramot (1 osztás: 100 ns)
- Ellenőrizzük szimulátorban a D FF működését
- Próbáljuk ki az FPGA kártyán a D FF működését
  - Az órajelet a **Bitbang I/O** funkcióval állítsuk elő

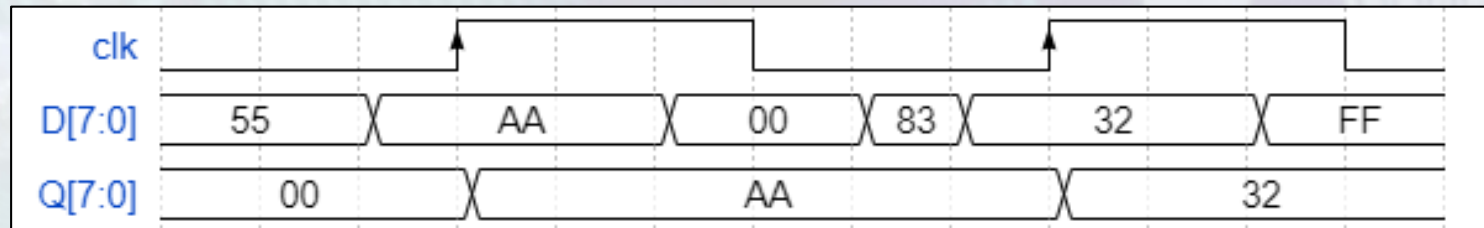


Frekvencia beállítása

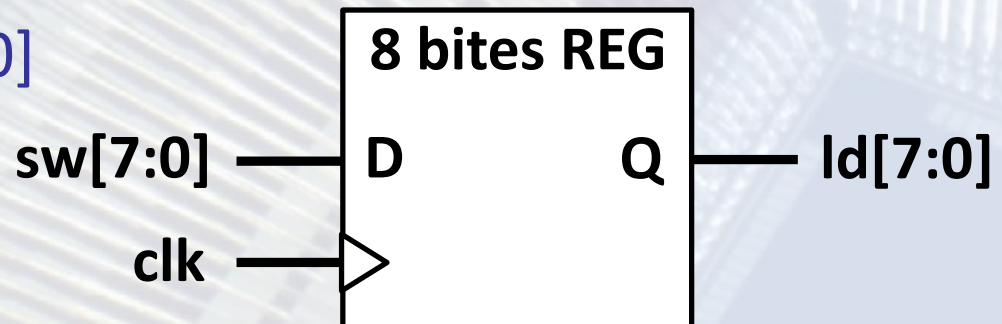
Pulzusok kiadása

# 1.b feladat: 8 bites regiszter

- Hogy módosítsuk a D flip-flop Verilog kódját, hogy az egy 8 bites regisztert valósítson meg?

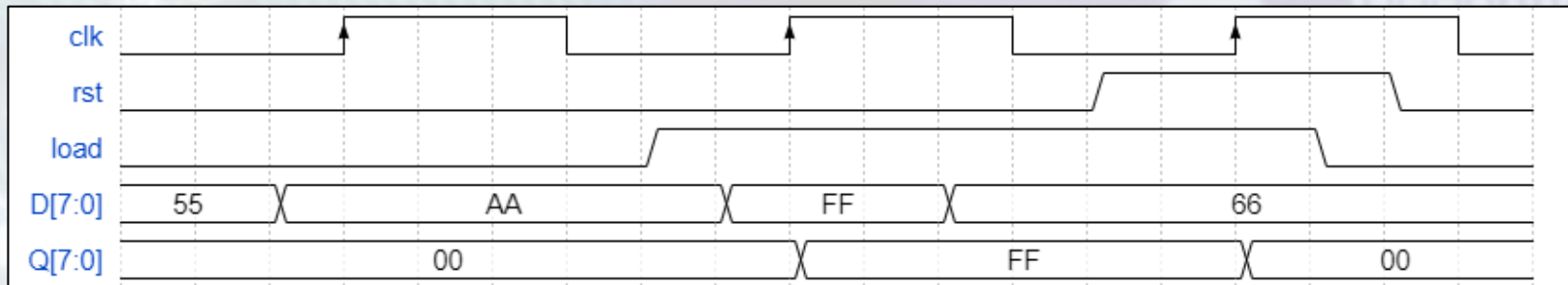


- A bemenetek és kimenetek bekötése
  - Órajel (clk): a letöltőkábeltől (*bitbang I/O*)
  - Bemenet (d): sw[7:0]
  - Kimenet (q): ld[7:0]



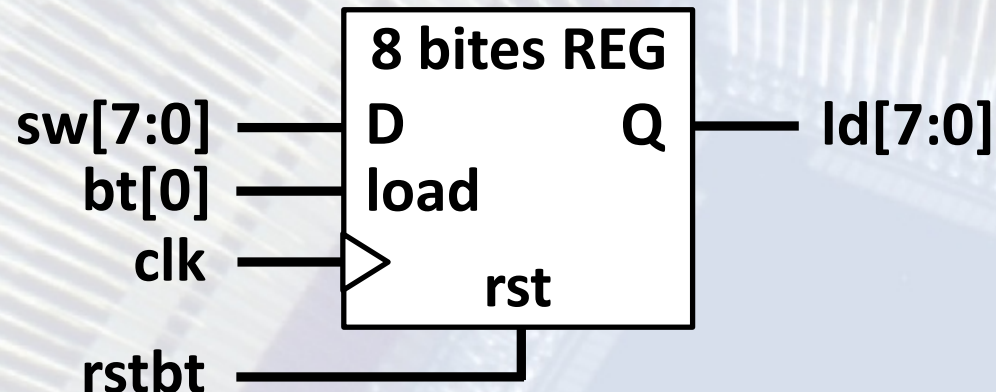
# 1.c feladat: törlés és engedélyezés

- Adjunk a 8 bites regiszterhez szinkron törlés és betöltés (engedélyezés) jeleket (a törlés a nagyobb prioritású)



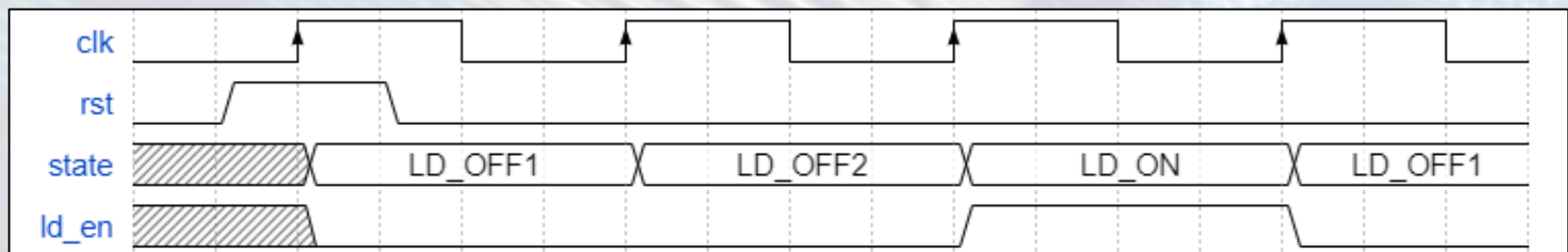
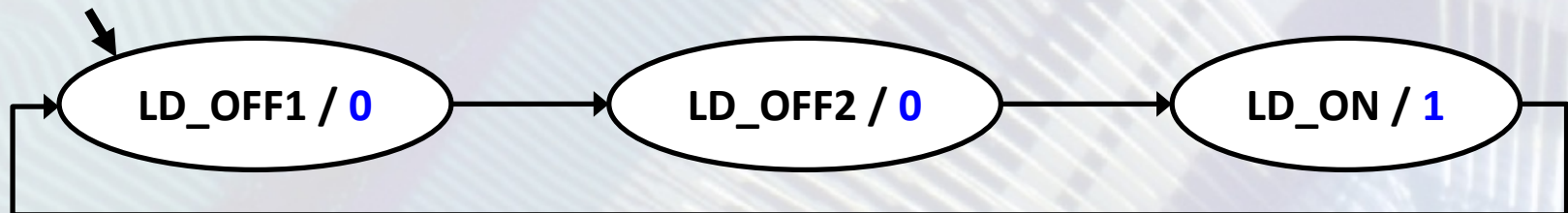
- A bemenetek és kimenetek bekötése**

- Órajel (clk): a letöltőkábeltől (*bitbang I/O*)
- Reset (rst): rstbt
- Betöltés (load): bt[0]
- Bemenet (d): sw[7:0]
- Kimenet (q): ld[7:0]



## 2. feladat: LED villogtató FSM

- A LED-ek villogtatása 1/3 kitöltési tényezővel
  - Minden 3. órajel periódusban bekapcsoljuk a LED-eket
- Megvalósítás állapotgéppel
  - 3 állapot: LD\_OFF1, LD\_OFF2, LD\_ON
  - Az LD\_ON állapotban bekapcsoljuk a LED-eket ( $ld\_en = 1$ )
  - Milyen modell szerint működik ez az automata?
  - Milyen Verilog nyelvi elemekkel célszerű leírni az állapotgépet?

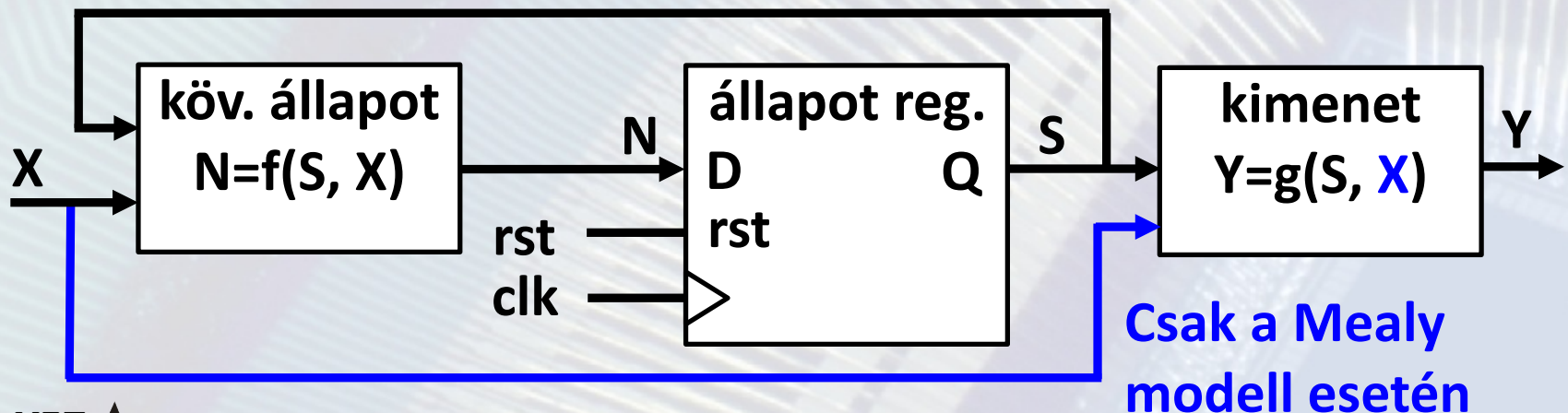


## 2. feladat: LED villogtató FSM

### FSM leírási lehetőségek

1. Az állapotregiszter és a következő állapot logika ( $f$ ) közös always blokkban vannak megadva
2. Az állapotregiszter és a következő állapot logika ( $f$ ) külön always blokkokban vannak megadva

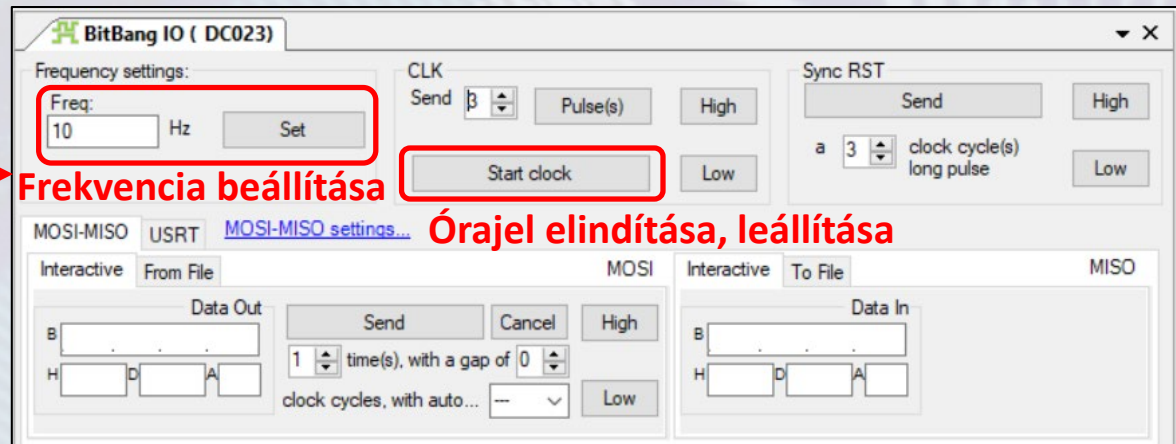
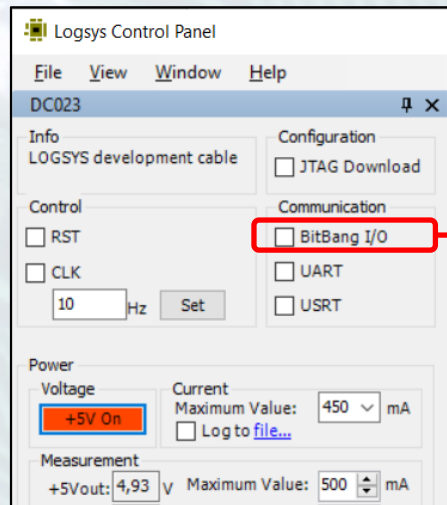
A kimeneti logikához ( $g$ ) mindig külön always blokkot vagy assign utasítást használunk





# 2. feladat: LED villogtató FSM

- Próbáljuk ki az FPGA kártyán az állapotgép működését
- Az órajelet a *Bitbang I/O* funkcióval állítsuk elő



## 2. feladat: LED villogtató FSM

Adjunk engedélyezési lehetőséget az FSM-hez

- Engedélyező bemenet: **enable** ( $\leftarrow bt[0]$ )
- Ha **enable** nem aktív, akkor LD\_OFF1-ben várakozunk

