



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

Digitális technika (VIMIAA03)

5. gyakorlat és 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)
 - LED villogtatása 3-szor nyomógomb lenyomására
- A feladatokhoz szükséges új Verilog ismeretek:
 - ***always*** blokk szinkron sorrendi hálózatok leírásához
 - Lokális paraméterek (***localparam***)
 - Relációs operátorok

Verilog HDL ismeretek

Szinkron sorrendi hálózatok leírása

- A szinkron sorrendi hálózatok csak **az órajel felfutó élére vagy lefutó élére váltanak állapotot, memória (tároló) tulajdonsággal rendelkeznek**
 - Feltételezzük a vezérlő bemenetek szinkron működését
- Szinkron sorrendi hálózat leírása **reg** típusú jellel
 - Az **always** blokk érzékenységi listájában csak az órajel felfutó (**posedge**) vagy lefutó (**negedge**) éle szerepelhet
 - Nem kell, hogy a vizsgált feltételek minden kombinációja esetén legyen kimeneti értékadás
 - **if** utasítás: nem kötelező az **else** ág megadása
 - **case** utasítás: nem kell felsorolni minden alternatívát, illetve nem kell használni a **default** alternatívát
 - Ha valamilyen vezérlési kombináció esetén nincs kimeneti értékadás, akkor a szinkron sorrendi hálózat megtartja az aktuális állapotát (tároló tulajdonság)

Verilog HDL ismeretek

Lokális paraméterek, relációs operátorok

- A lokális paraméterek deklarálásának szintaxisa

localparam [azonosító] = [konstans];

- A lokális paraméterekkel azonosítóhoz konstans érték rendelhető
- Egyik felhasználási esete az állapotgépek állapotainak megadása, így az állapotkódok közvetlen használata helyett egy jóval szemléletesebb, jobban értelmezhető szöveges elnevezés is használható

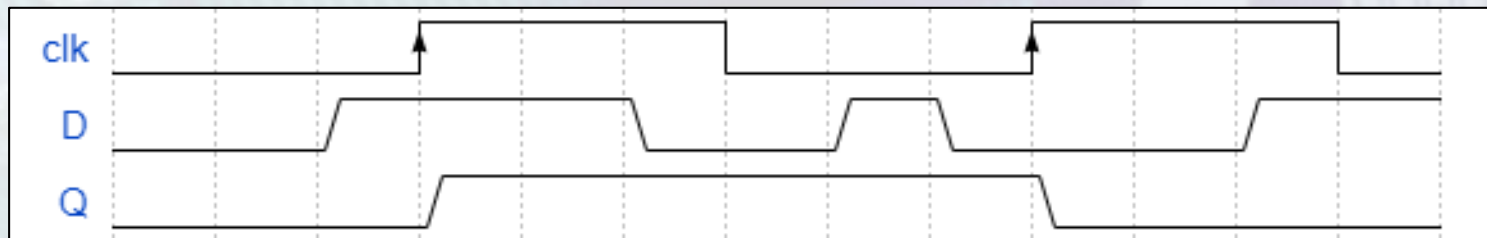
- Relációs operátorok

== (egyenlő), **!=** (nem egyenlő), **<** (kisebb), **>** (nagyobb),
<= (kisebb vagy egyenlő), **>=** (nagyobb vagy egyenlő)

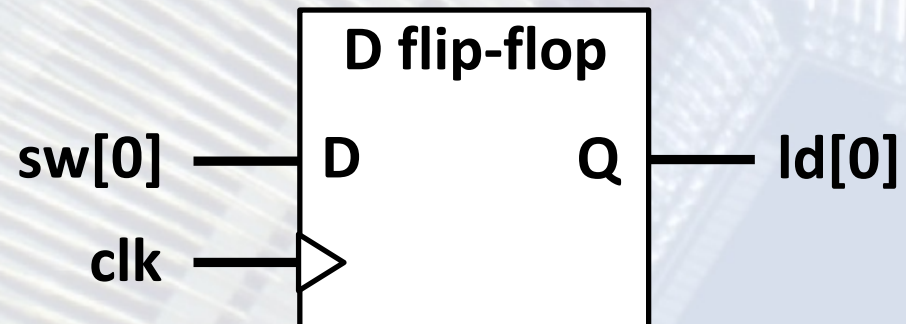
- Operandusok száma: 2
- Az eredmény mindig egybites: 0 (hamis) vagy 1 (igaz)
- Az egyenlő és a nem egyenlő reláció kapus logikára, a kisebb és a nagyobb reláció jellemzően aritmetikai funkcióra képződik le

1.a feladat: D flip-flop

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

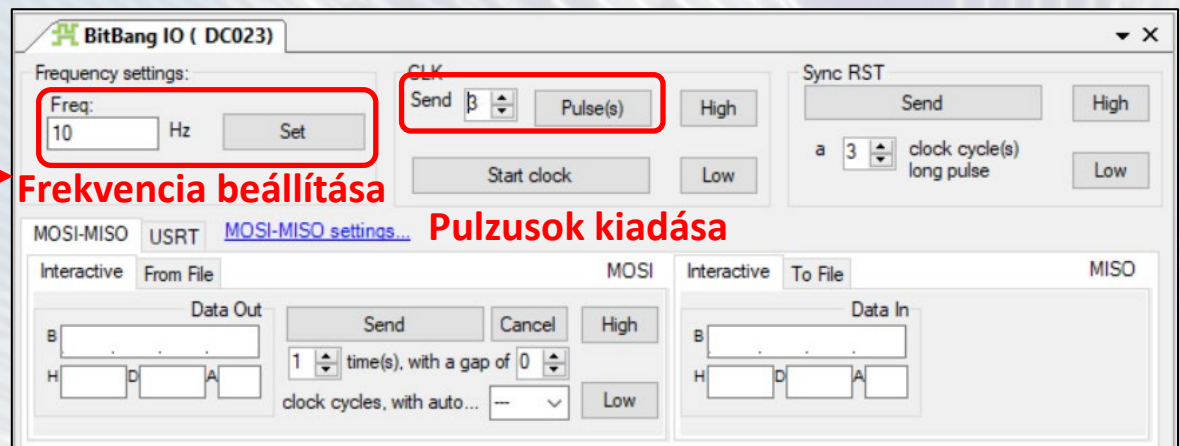
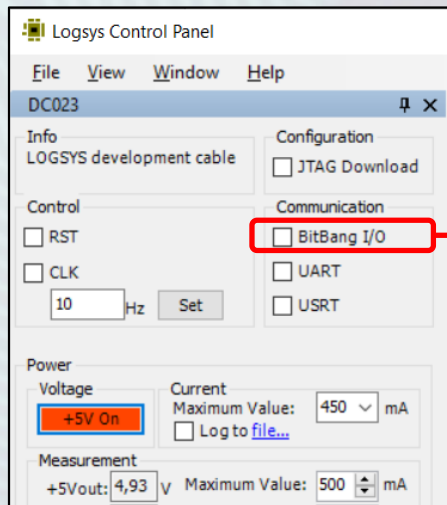


- A bemenetek és kimenetek bekötése
 - Órajel (clk): a letöltőkábeltől (*bitbang I/O*)
 - Bemenet (d): sw[0]
 - Kimenet (q): ld[0]



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

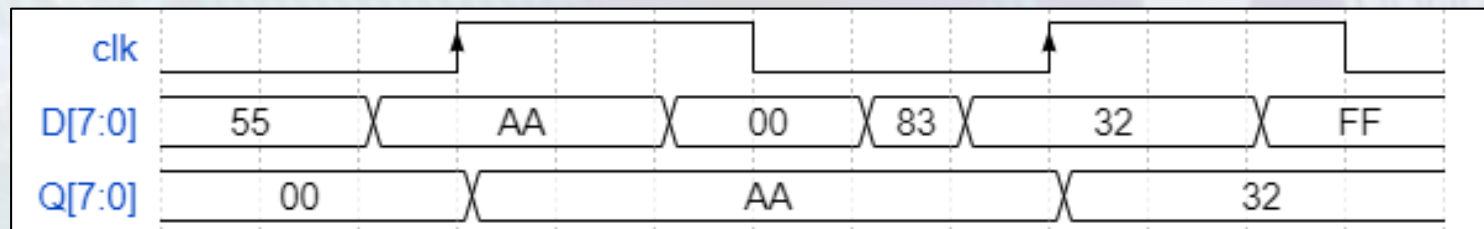


Frekvencia beállítása

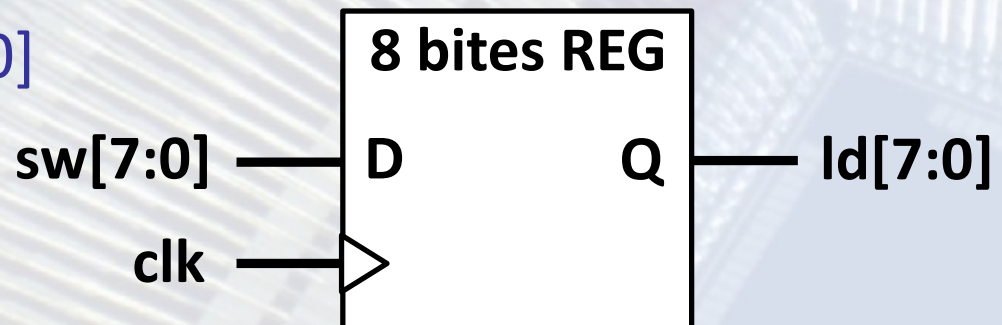
Pulzusok kiadása

1.b feladat: 8 bites regiszter

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

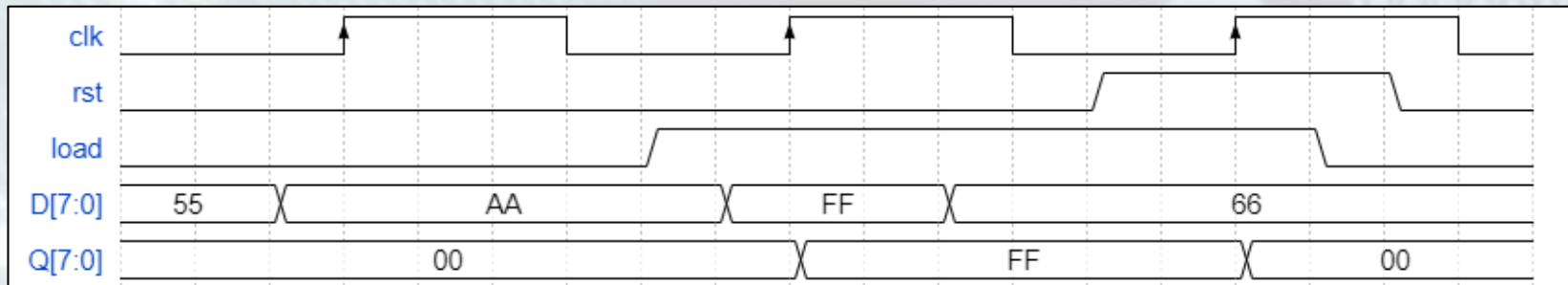


- A bemenetek és kimenetek bekötése
 - Órajel (clk): a letöltőkábeltől (*bitbang I/O*)
 - Bemenet (d): sw[7:0]
 - Kimenet (q): ld[7:0]



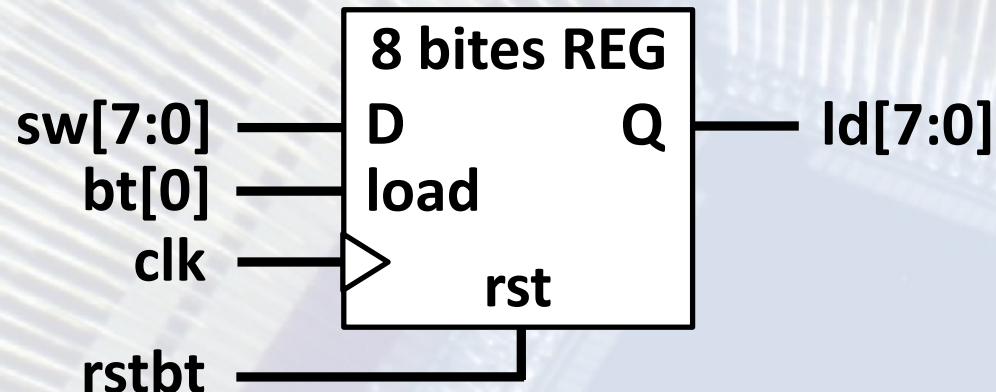
1.c feladat: törlés és engedélyezés

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



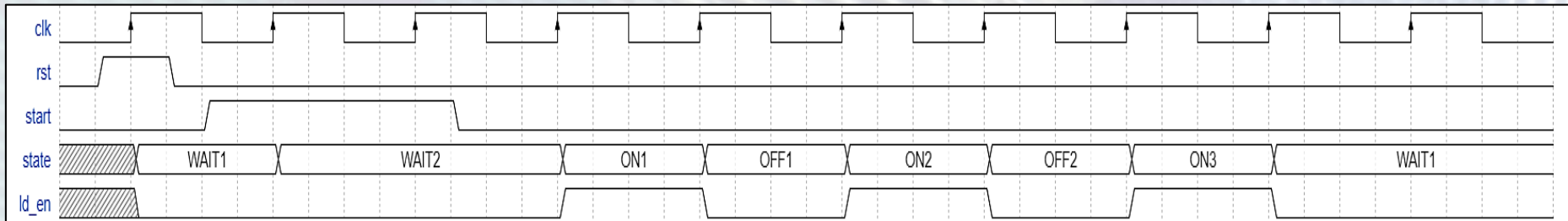
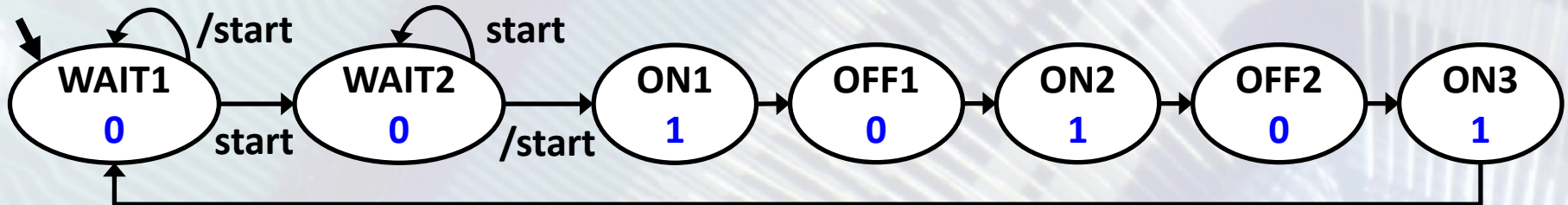
- A bemenetek és kimenetek bekötése

- 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 3-szori villogtatása egy nyomógomb megnyomásának, majd felengedésének hatására
- Megvalósítás állapotgéppel
 - Első 2 állapot: várakozás a nyomógomb (**start** bemenet) magas, majd alacsony szintjére
 - Következő 5 állapot: LED-ek (**ld_en** kimenet) 3-szori villogtatása (10101)
 - Milyen modell szerint működik ez az automata?
 - Milyen Verilog nyelvi elemekkel célszerű leírni az állapotgépet?

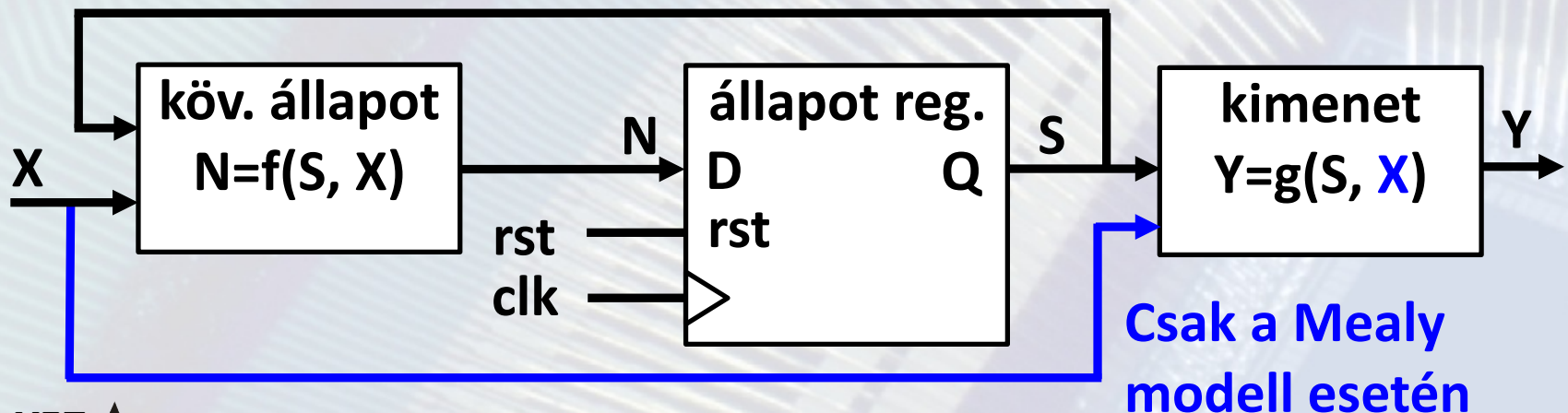


2. feladat: LED villogtató FSM

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
 - A villogás indítása a BTN0 nyomógombbal lehetséges
 - Az aktuális állapot kódja megjelenik a kijelzőn
- Az órajelet a *Bitbang I/O* funkcióval állítsuk elő

