

### 3. Vaja: svetlobna veriga

#### 1. Pomikalni register

Sekvenčna vezja vsebujejo pomnilne gradnike, ki ohranjajo notranja stanja. Programirljiva vezja FPGA in CPLD vsebujejo sinhrono sekvenčne gradnike (flip-flope).

**signal** stanje: std\_logic := '0';

...

P: **process**(clk)

**begin**

**if** rising\_edge(clk) **then**

stanje <= vhod;

...

**end if;**

**end process;**

---

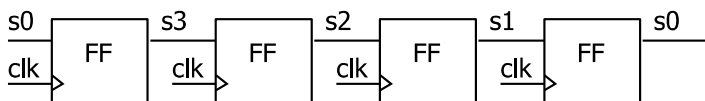
#### SINHRONI PROCES

---

Delovanje sinhronih vezij, ki spreminjajo izhode ob fronti ure, opišemo s procesom. Vsi stavki, ki opisujejo delovanje gradnika, so zapisani znotraj tega pogojnega stavka:

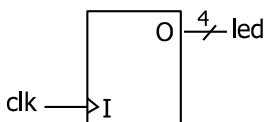
**if** rising\_edge(clk) **then...**

---



- a) Naredi vezje, ki je sestavljeno iz štirih D flip-flopov, kot prikazuje shema. Deklariraj notranje signale s0, s1, s2 in s3 in jim določi začetne vrednosti, tako da bo s0 na '1', vsi ostali pa na '0'. Preizkusi delovanje vezja na simulatorju !
- Ali je vrstni red zapisa prireditvenih stavkov pomemben ?
  - Združi vsa stanja v 4-bitni vektor, ki naj bo izhod vezja. Na katerem mestu je najbolje zapisati ta prireditveni stavek ?

#### 2. Svetlobna veriga



Naredi krmilnik za svetlobno verigo, ki v zaporedju prižiga eno od štirih LED na razvojnem sistemu. Uporabi ciklični pomikalni register, ki je nastavljen na začetno stanje 0001 in ima dodaten signal (*enable*) za omogočanje pomika. Ta signal naj pride iz delilnika frekvence.

- a) Opiši vezje v dveh procesih: en proces naj bo pomikalni register s signalom *enable*, drugi proces pa naj bo delilnik v obliki števca. Delilnik naj šteje od 0 do 10 in naredi impulz *enable* vsakokrat, ko se vrne na 0. Naredi simulacijo vezja.
- b) Na razvojnem sistemu imamo uro frekvence 50 MHz, zato bo potrebno večje delilno razmerje. Naredi delilnik, ki šteje do 10 000 000 in napravi sintezo vezja.
- c) Ustvari novo datoteko z definicijami priključkov (User Constraints File) in naredi implementacijo vezja ter preizkusi delovanje na razvojnem sistemu Basys2.

```
NET "clk" LOC = "B8" ;
NET "led<0>" LOC = "M5" ;
NET "led<1>" LOC = "M11" ;
NET "led<2>" LOC = "P7" ;
NET "led<3>" LOC = "P6" ;
```