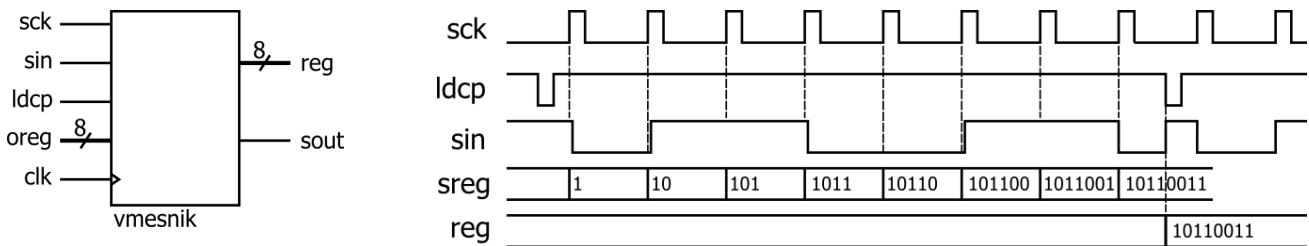


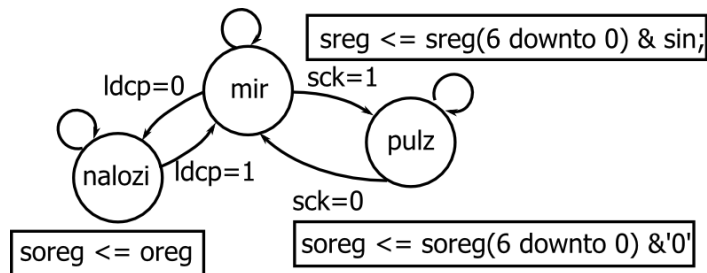
6. vaja: Zaporedni vmesnik

V jeziku VHDL naredi logični vmesnik za branje in zapisovanje podatkov na dodatni modul za Red Pitayo. Na modulu se nahajata pomikalna registra 74165 (PISO) in 74595 (SIPO). Vmesnik dobi na vhodu uro za prenos podatkov (**sck**) in kontrolni signal za nalaganje registrov (**ldcp**), zaporedni podatkovni vhod (**sin**), 8-bitni register (**oreg**), ki ga bomo prenesli na izhod in sistemsko uro (**clk**). Izhod vmesnika pa je 8-bitni register (**reg**) in zaporedni podatkovni izhod (**sout**).



Vhodni podatki se pomikajo ob prvi fronti ure **sck** v notranji pomikalni register **sreg**. Prehod signala **ldcp** na 0 sproži nov cikel prenosa in kopiranje podatkov iz pomikalnega registra v register na izhodu vmesnika (**reg**).

Vmesnik bomo naredili kot sekvenčni stroj, ki ga opišemo z diagramom stanj. Stanja prehajajo ob sistemski uri **clk**, ki je precej hitrejša kot ura za prenos podatkov, zato lahko v stanjih kar vzorčimo kontrolna signala **sck** in **ldcp**. Ko v stanju *mir* zaznamo prehod ure **sck** na 1 (prva fronta), se premaknemo v stanje *pulz* in hkrati pomaknemo vrednost iz vhoda **sin** v register **sreg**.



Ob zadnji fronti ure **sck** pa pomaknemo vrednost v notranjem pomikalnem registru **soreg**. Vrednost tega registra naj se naloži iz vhoda **oreg**, ko smo v stanju *nalozi*.

Na začetku opisa vezja deklariraj nov podatkovni tip za stanja in notranji signal:

```

type stanja is (mir, pulz, nalozi);
signal st: stanja;
    
```

Opiši delovanje vmesnika v sinhronem procesu s stavkom **case**. Za simulacijo vmesnika uporabi vnaprej pripravljeno testno strukturo.