



9. Vaja: mikroprocesor (3)

Model mikroprocesorja bomo nadgradili z vhodno izhodno enoto in dvema ukazoma:

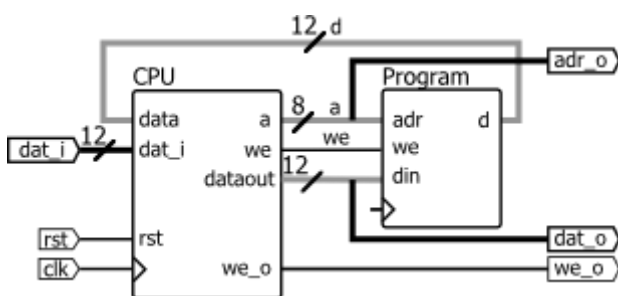
INP A – naloži v akumulator vrednost iz DAT_I na naslovu A

OUTP A – prenesi akumulator v DAT_O ob naslovu A

Simulator mikroprocesorja: cpu.html

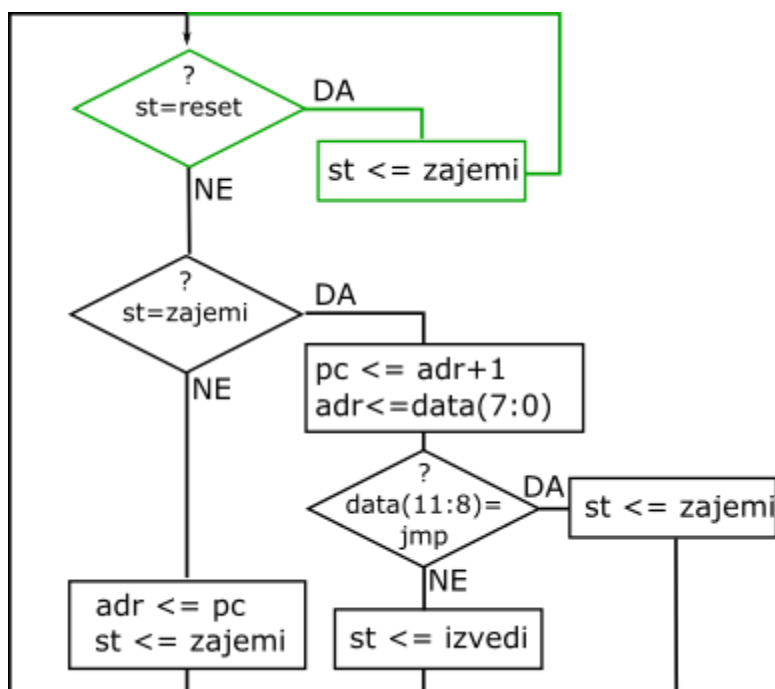
Naloga

Dodaj v Proc.vhd zunanje priključke (**port**) in povezave med komponentami, kot prikazuje shema:



Nadgradnja CPU

- Dodaj v kontrolno enoto procesorja stanje reset, ki naj bo prvo izmed naštetih stanj in dopolni delovanje po diagramu:
- `type` stanje `is` (reset, zajemi, izvedi);
- `signal` `st`: stanje;



d. Dodaj zunanje priključke: 12-bitni nepredznačeni vhod **dat_i** in enopbitni izhod **we_o**. Napiši logiko za ta dva signala:

- Ob ukazu INP naj se podatek iz vodila **dat_i** prenese v akumulator (podobno kot ukaz LDA)
- **we_o** dobi vrednost '1', kadar smo v stanju *zajemi* in je na podatkovnem vhodu ukaz
OUTP: **data(11:8) = outp**
- sicer naj ima **we_o** vrednost '0'

e. Naredi simulacijo s testnim programom:

```
type memory is array(0 to 255) of unsigned(11 downto
0);
signal m : memory := (
0=> inp & x"00",
1=> jze & x"00",
2=> lda & x"07",
3=> add & x"08",
4=> sta & x"07",
5=> outp & x"01",
6=> jmp & x"00",
7=> x"007",
8=> x"003",
others => x"000" );
```

Program v simulatorju:

[cpu.html](#)

```
start: inp vh
      jze start
      lda a
      add b
      sta a
      outp vh
      jmp start
vh     di 0
a      db 7
b      db 3
```

Predlagane razširitve nabora ukazov:

```
-- razširjen nabor ukazov za samostojno delo
constant jcs: koda := "0110"; -- skok, ce je prenos (carry=1)

constant call: koda := "1010"; -- skok na podprogram, shrani PC in akum
constant ret: koda := "1011"; -- vrni se, ponastavi PC in akum

constant nota: koda := "0000"; -- logicna negacija
constant anda: koda := "1100"; -- logicna IN
constant ora: koda := "1101"; -- logicna ALI

constant shl: koda := "1110"; -- pomik akum za eno v levo (deljenje z 2)
constant shr: koda := "1111"; -- pomik akum za eno v desno (mnozenje z 2)
```