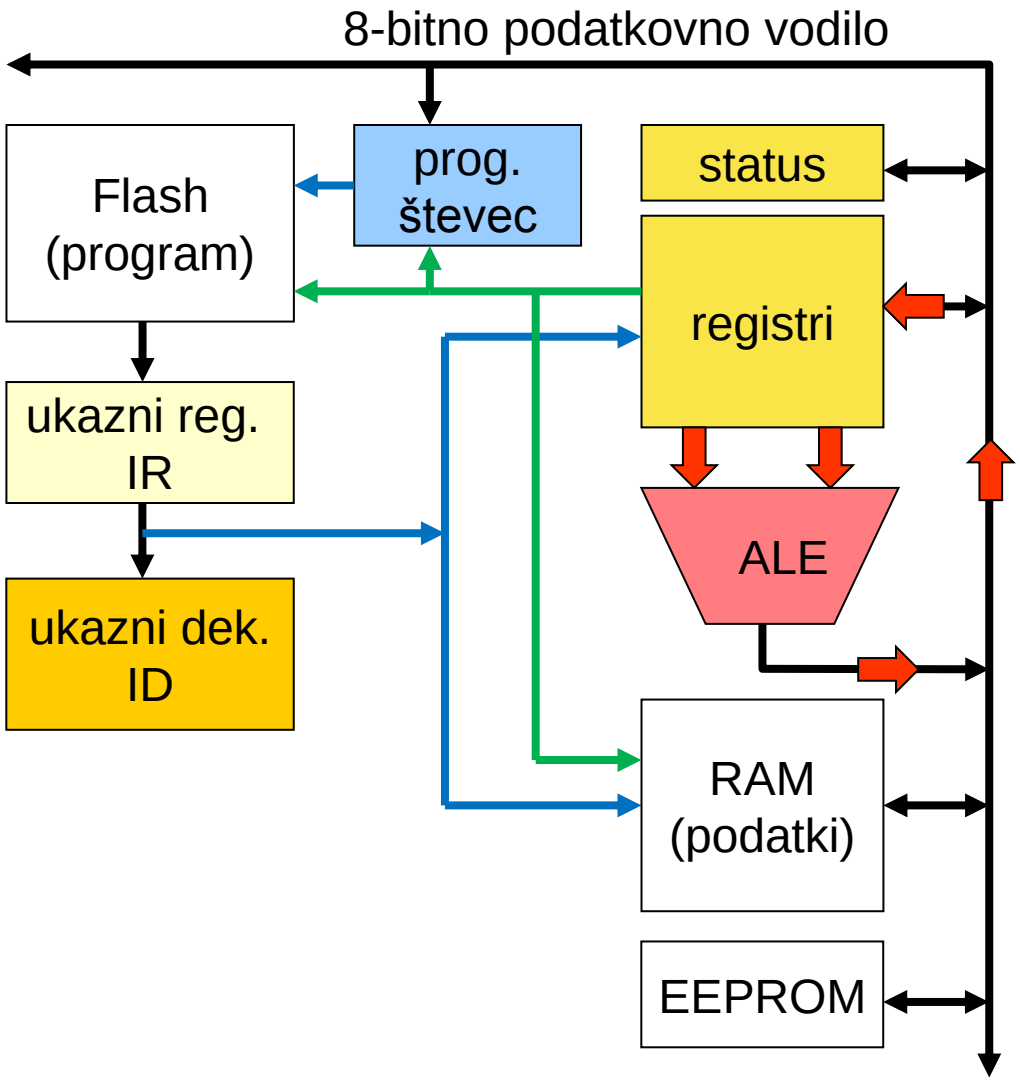
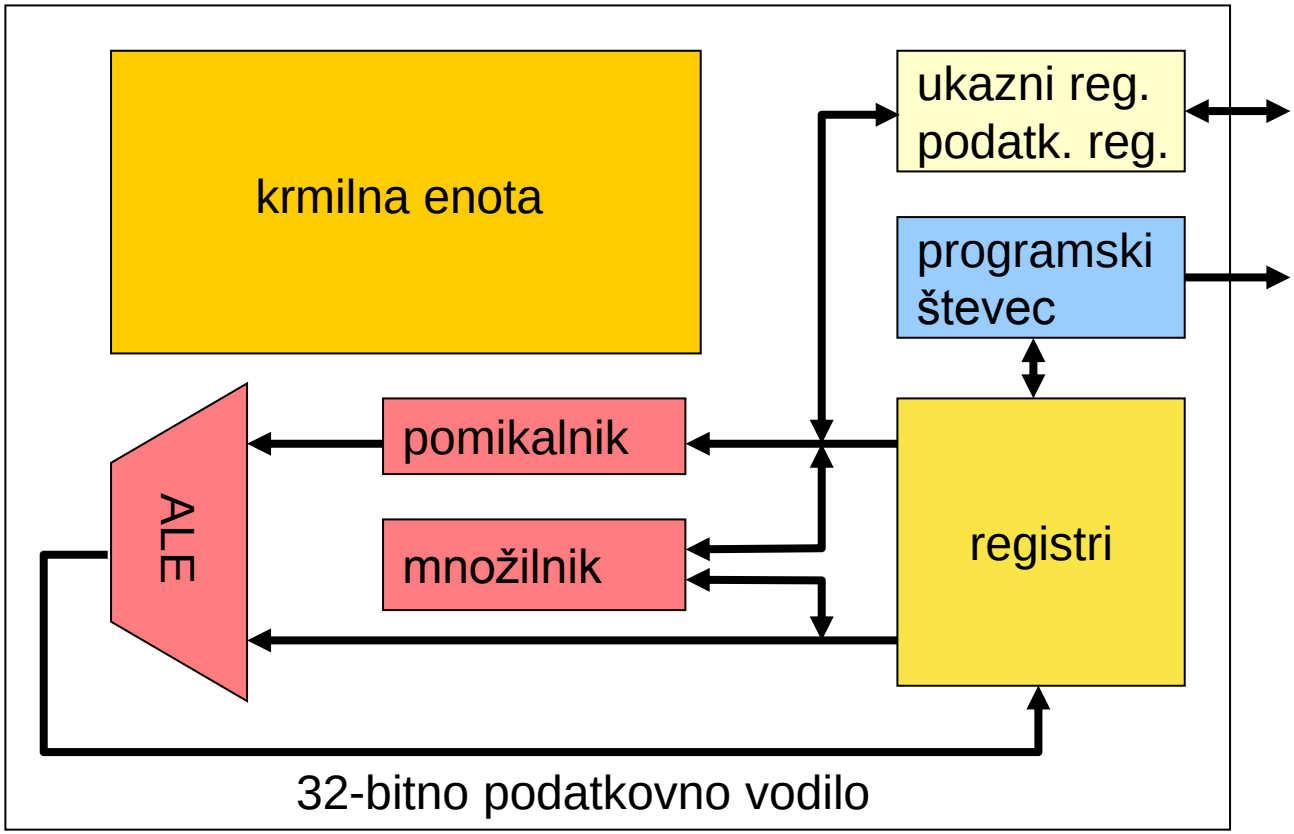
Centralna procesna enota AVR



Centralna procesna enota ARM-7



- Zmogljivi procesorji vsebujejo dodatne računske enote
  - ciklični pomikalnik, množilnik, enote za plavajočo vejico

Procesorji v programirljivih vezjih

- izvedba s program. logiko (Soft Core)
  - vnaprej pripravljene komponente intelektualne lastnine
  - procesor prevedemo (sinteza, tehnološka preslikava)
  - zgradbo procesorja in število jeder prilagodimo aplikaciji
  - Altera Nios (32-bit), LNIV CPE (12-bit)

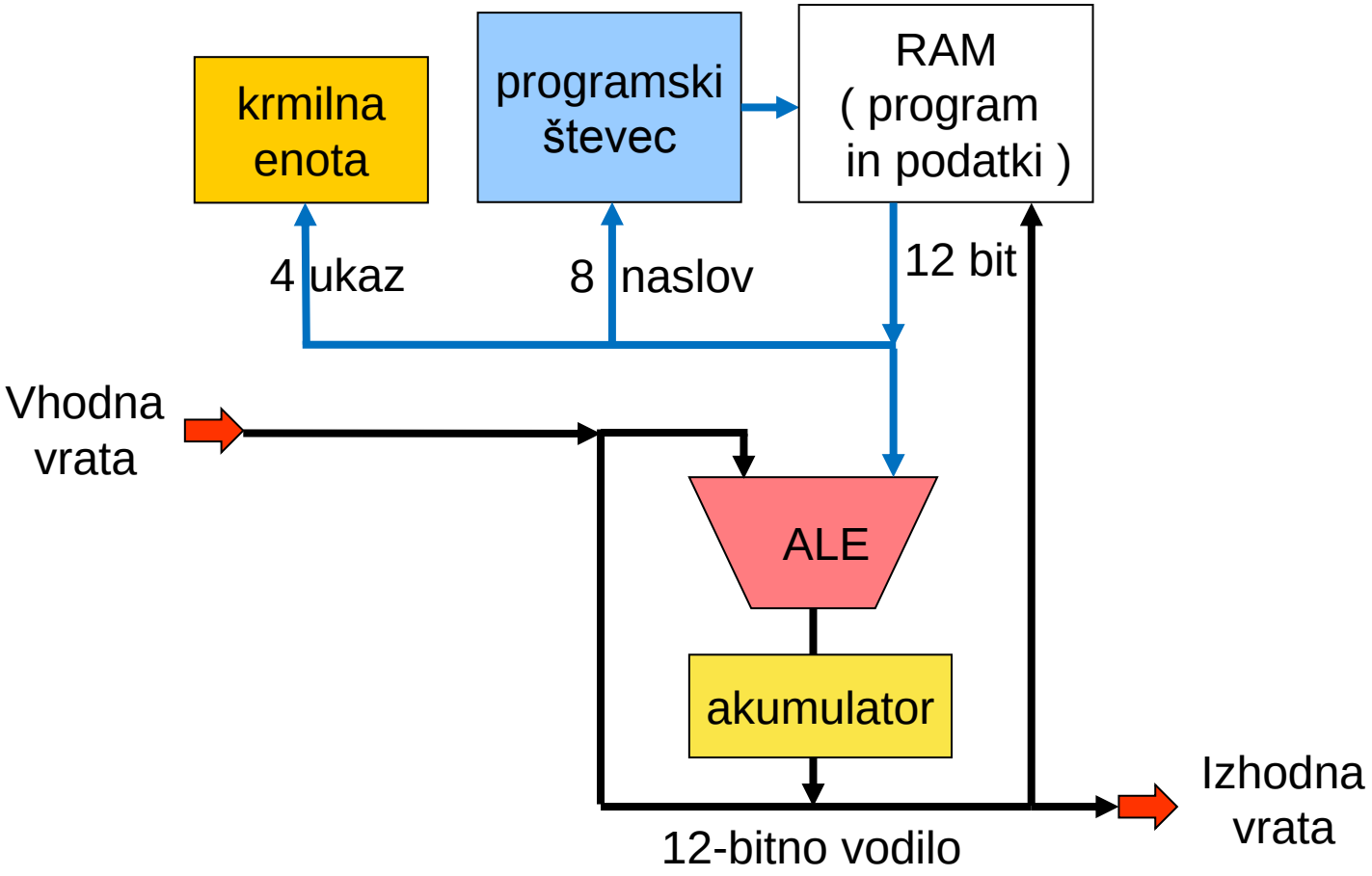
- izvedba z namenskim vezjem (Hard Core)
  - programirljivo vezje in mikroprocesor na istem čipu
  - Bolj zmogljiva / manj prilagodljiva izvedba
  - ARM-9 (Altera in Xilinx)

Prednosti procesorjev v tehnologiji FPGA

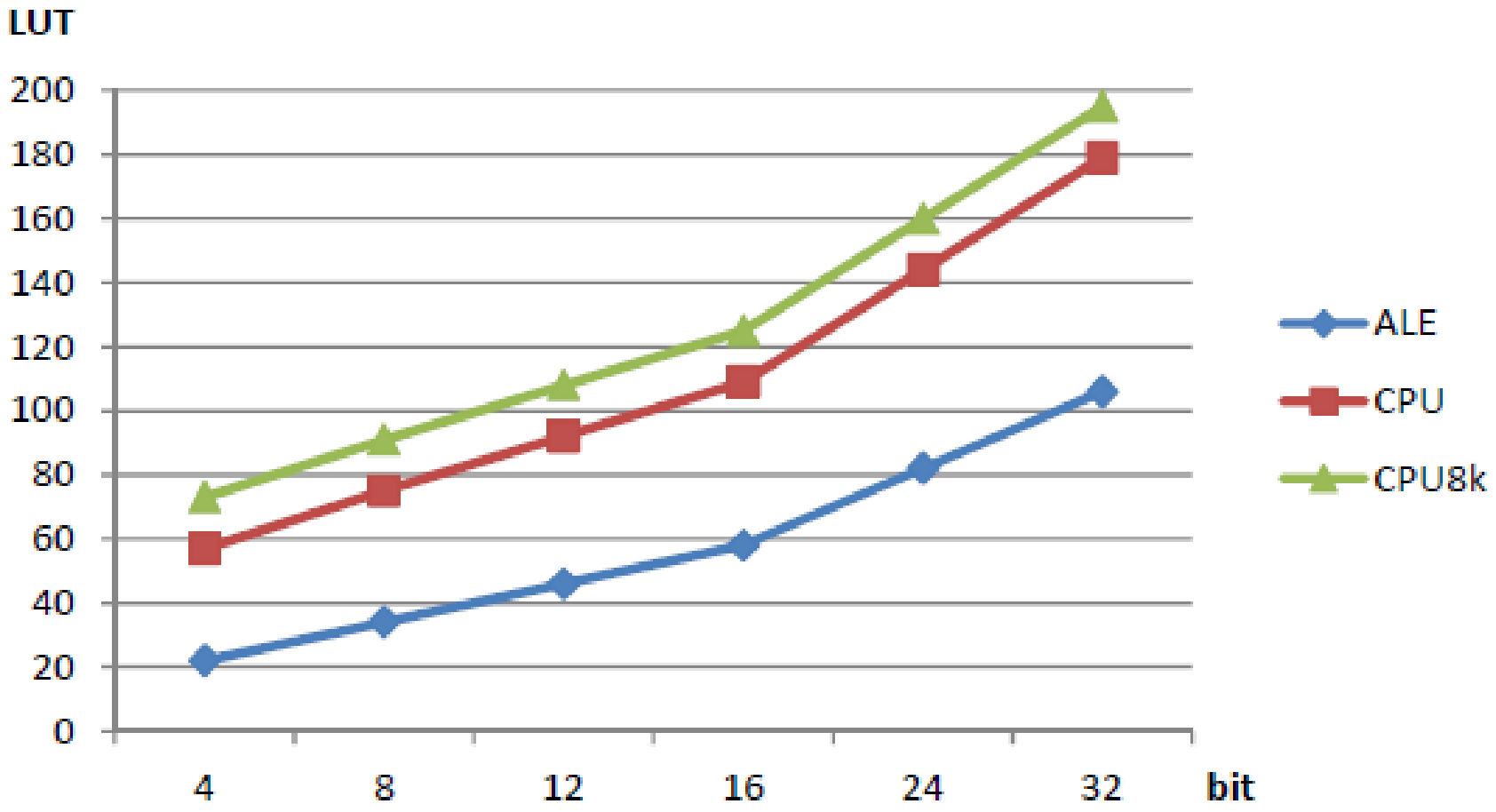
- Sodobna vezja FPGA so dovolj velika za CPE
  - zapletena opravila izvajamo na procesorju
- Na FPGA lahko naredimo celoten sistem
  - manj integriranih vezij v napravi
  - nismo odvisni od proizvajalcev procesorjev
- Prilagodljivost
  - CPE popolnoma prilagodimo aplikaciji

Enostaven mikroprocesor za FPGA

- N-bitni procesor, npr. N=12



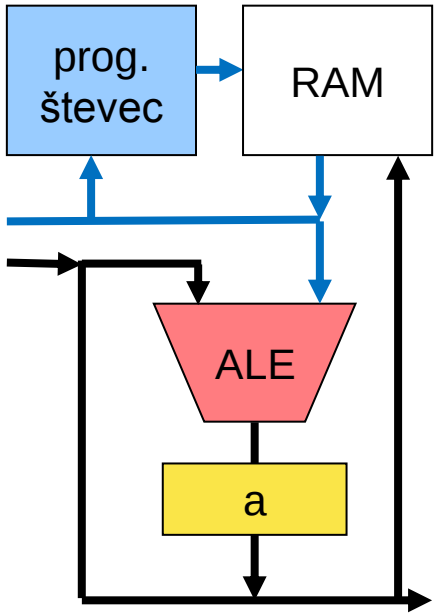
Procesor zasede malo celic v vezju FPGA



Osnovni nabor ukazov

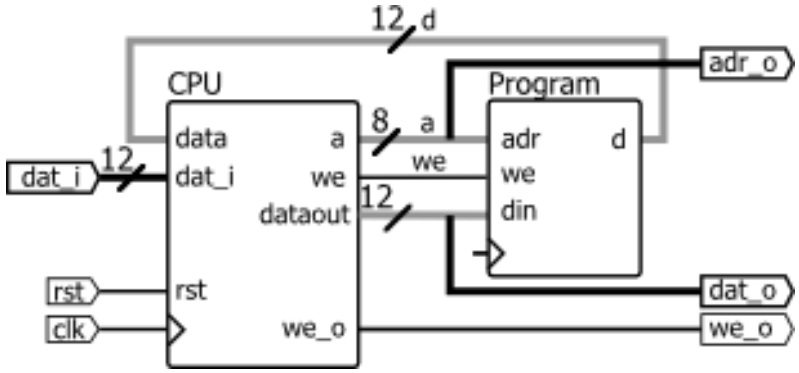
lda koda := "0001"; -- *naloži iz RAM a = [M]*
sta koda := "0010"; -- *shrani v RAM [M] = a*
add koda := "0100"; -- *prištej a = a + [M]*
sub koda := "0101"; -- *odštej a = a - [M]*
anda koda := "0110"; -- *logična a = a and [M]*
ora koda := "0111"; -- *logična a = a or [M]*
jmp koda := "1000"; -- *skok na nov naslov*
jze koda := "1001"; -- *skok, če je* a=0

inp koda := "0011"; -- *naloži iz vhoda a = IN*
outp koda := "0111"; -- *shrani na izhod OUT = a*



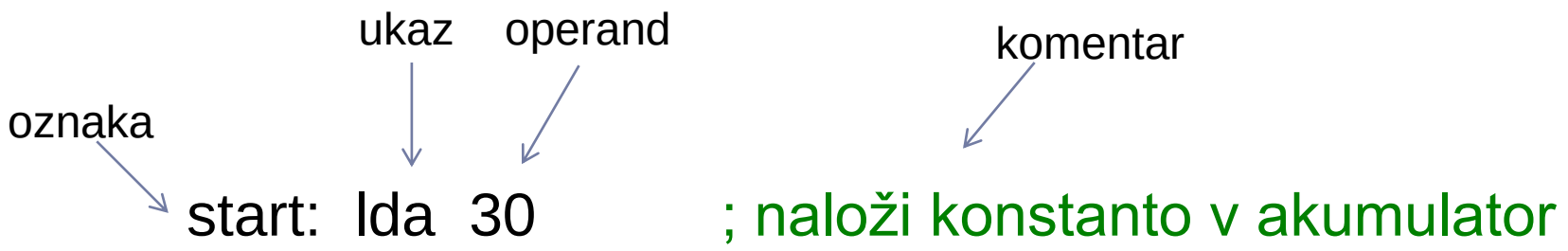
Priključki mikroprocesorja

signal	število bitov	opis
clk	1	Sistemska ura.
rst	1	Reset signal.
a	8	Naslov pomnilnika.
data	12	Podatkovni vhod.
dataout	12	Podatkovni izhod.
we	1	Pisanje v pomnilnik.
dat_i	12	Vhodna vrata.
dat_o	12	Izhodna vrata.
we_o	1	Pisanje v izhodna vrata.
adr_o	1	Vhodno-izhodni naslov.



Programiranje v zbirniku

- Zbirnik je vez med programskimi jeziki (npr. C) in strojno kodo, ki je odvisna od arhitekture
- Opis majhne in učinkovite programske kode za
  - nizkonivojske rutine (dostop do perifernih enot)
  - računsko zahtevna opravila (matematika, grafika)
- Pri velikih projektih je delo v zbirniku težko, nefleksibilno, prevajalniki jezika C pa so dobro optimizirani

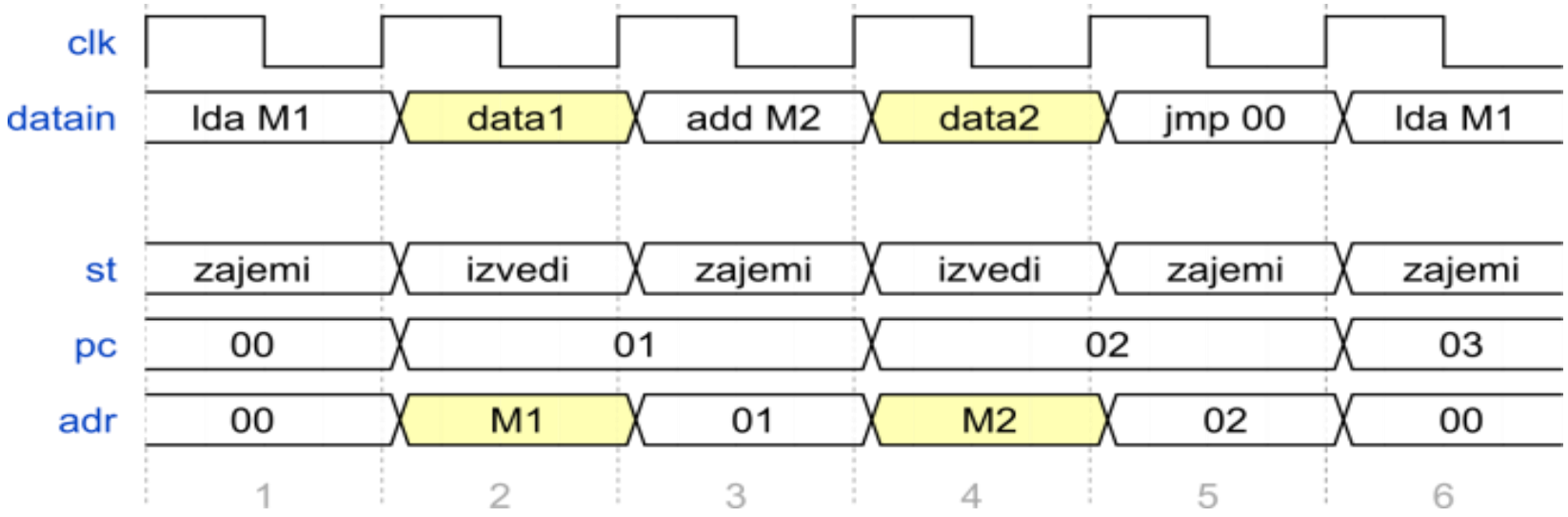


Zbirnik

- program, ki računa: 2+2+2+2+...

```
start:  lda vred     ; naloži vred
        add vred     ; prištej vred
        jmp start    ; skok nazaj
vred    db      2    ; direktiva
```

```
0:  0001 0000 0011
1:  0100 0000 0011
2:  1000 0000 0000
3:  0000 0000 0010
```



Simulator: http://lniv.fe.uni-lj.si/cpu.html