



Laboratorij za načrtovanje integriranih vezij

Univerza *v Ljubljani*
Fakulteta *za elektrotehniko*



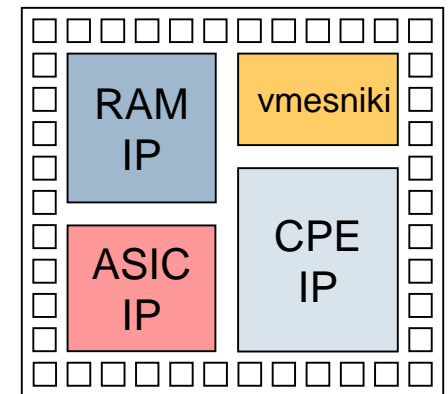
Digitalni Elektronski Sistemi

Digitalni sistemi in procesorji

digitalni sistem, zgradba mikroprocesorja

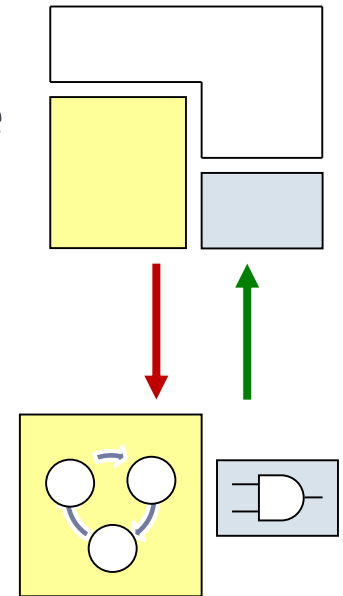
Digitalni sistemi

- ▶ Digitalni sistemi so kompleksna digitalna vezja iz različnih komponent:
 - ▶ centralnih procesnih enot (CPE) oz. mikroprocesorjev
 - ▶ namenskih digitalnih vezij (*Intellectual Property – IP core*)
 - ▶ vmesnikov in pomnilnikov
- ▶ Sistemi na čipu imajo integrirane vse komponente
- ▶ Sisteme razvijamo z uporabo obstoječih univerzalnih komponent in izdelavo novih namenskih vezij



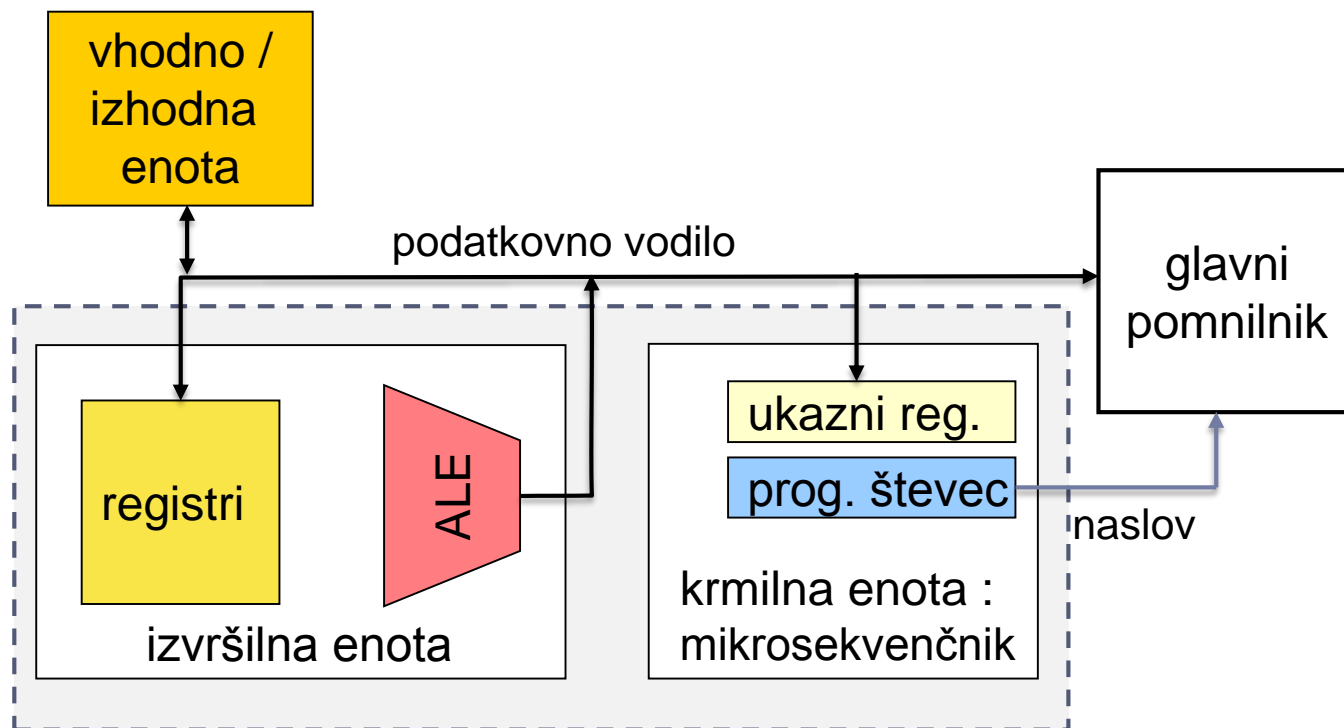
Načrtovanje digitalnih sistemov

- ▶ Tehnologija omogoča izdelavo vse kompleksnejši vezij
 - ▶ načrtovanje namenskih vezij na ravni registrov je zahtevno
 - ▶ povečuje se razkorak med tehnologijo (kaj lahko izdelamo) in učinkovitostjo razvoja (hitrost, cena razvoja vezja)
- ▶ Učinkovito načrtovanje
 - ▶ univerzalni gradniki (CPE) in že narejene namenske komponente (IP)
 - ▶ za razvoj novih uporabimo visokonivojski opis in hierarhično načrtovanje (*top-down design*)
 - ▶ vezje razdelimo na podvezja, ki jih postopoma dograjujemo
 - ▶ podatkovni in krmilni del vezja



Mikroprocesor

- ▶ Von Neumannov model: centralna procesna enota (CPE), glavni pomnilnik in vhodno/izhodna (V/I) enota



- ▶ delovanje CPE določa nabor ukazov
 - ▶ ukazi so prilagojeni programskemu jeziku (C/C++, Java)
 - ▶ kompleksen nabor ukazov (CISC) za zmogljive računalnike ali reduciran nabor ukazov (RISC) za vgrajene sisteme

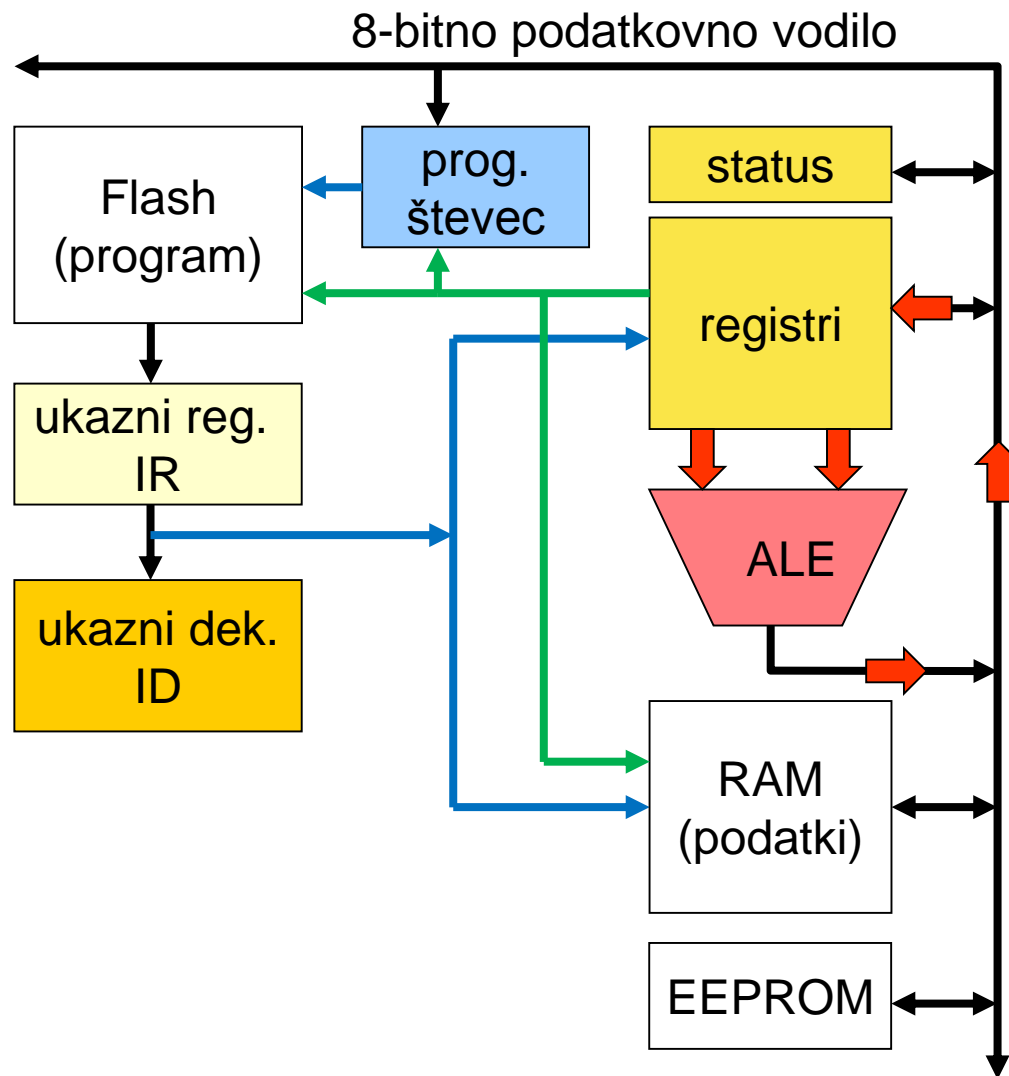
Osnovni gradniki mikroprocesorja

- ▶ Mikroprocesor na integriranem vezju vsebuje izvršilno in krmilno enoto
- ▶ Mikroprocesor potrebuje za delovanje:
 - ▶ zunanjo uro in reset
 - ▶ zunanji pomnilnik s programskimi ukazi in podatki
 - ▶ vhodno / izhodno (**periferno**) enoto za komunikacijo z okolico
 - ▶ periferna enota je lahko del pomnilnika (na določenih naslovih)
 - ▶ ali pa poteka komunikacija preko posebnih V/I ukazov
- ▶ V praksi potrebujemo vsaj dve vrsti pomnilnika
 - ▶ za program takšnega, ki ohranja vsebino (ROM, Flash)
 - ▶ za delovne podatke pa pomnilnik s hitrim branjem in pisanjem (RAM – *Random Access Memory*)

Mikroprocesorji za vgrajene naprave

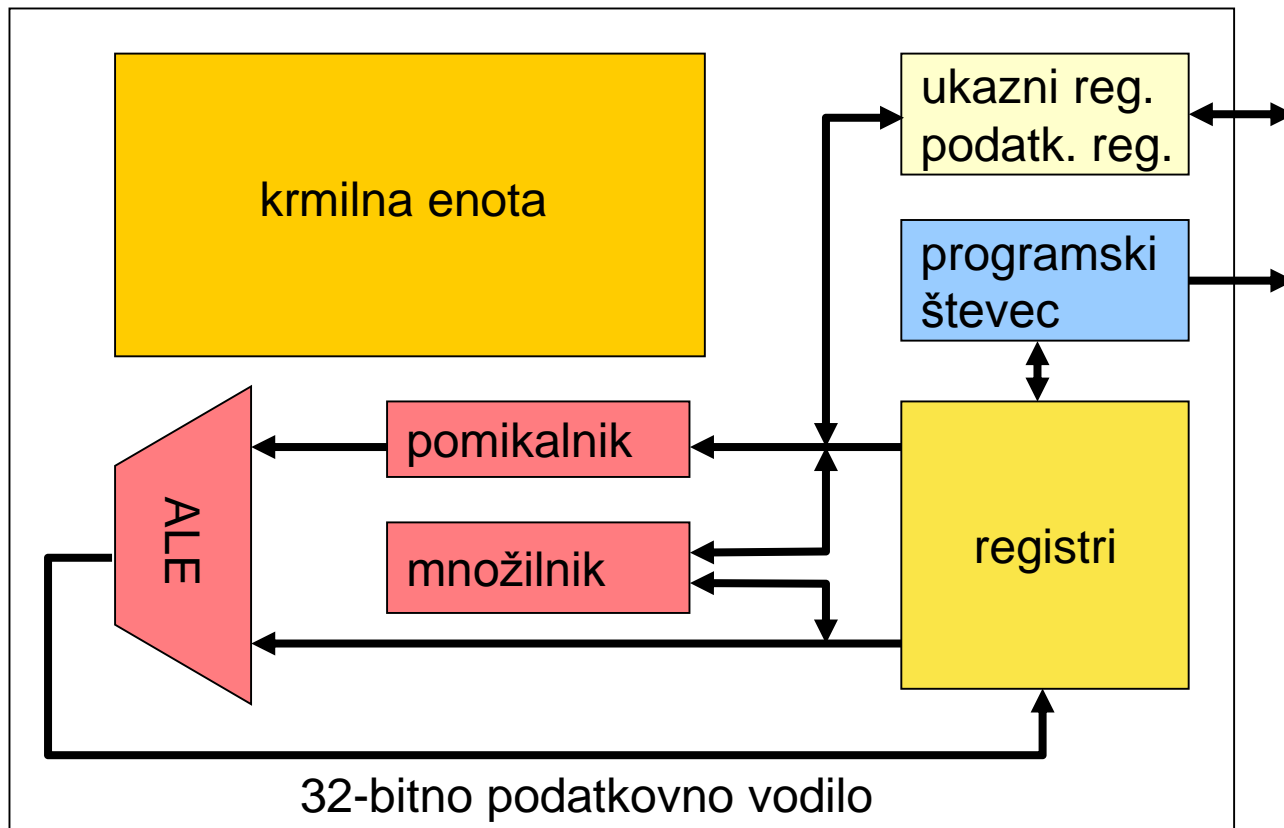
- ▶ Mikrokrmilniki so integrirana vezja, ki vsebujejo
 - ▶ mikroprocesorsko jedro (izvršilno in krmilno enoto),
 - ▶ programski in podatkovni pomnilnik,
 - ▶ ter različne V/I enote:
 - ▶ vzporedna vrata (Port)
 - ▶ zaporedne komunikacijske vmesnike: I2C, SPI, UART
 - ▶ analogno / digitalne (A / D) in D / A pretvornike
 - ▶ modulatorje (PWM), časovnike, števec...
 - ▶ komunikacijske krmilnike: Ethernet MAC, USB

Primer: centralna procesna enota AVR



- ▶ Atmel AVR, 8-bitni mikrokontroler na razvojnem sistemu Arduino

Primer: centralna procesna enota ARM-7



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

Mikroprocesorji 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 CPE4 (12-bit)

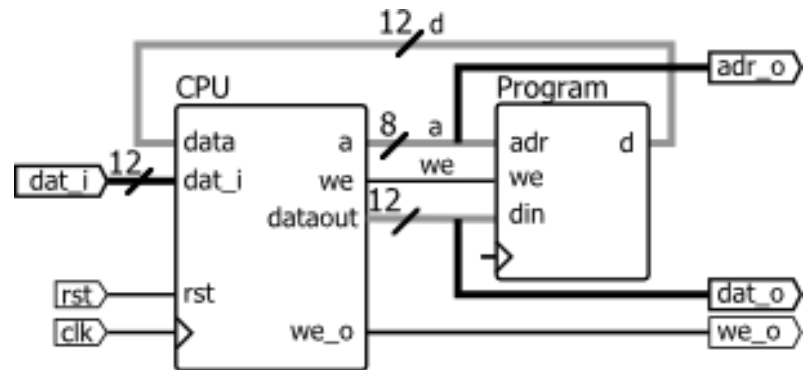
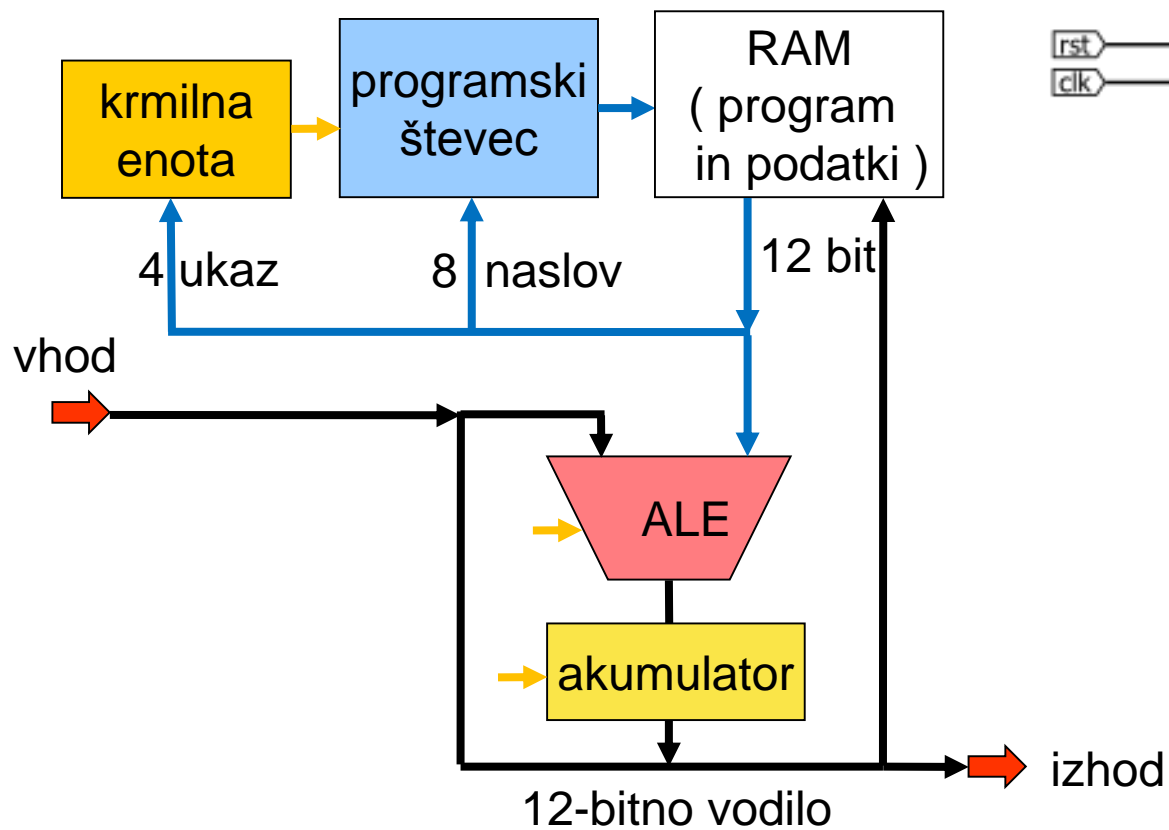
- ▶ izvedba z namenskim vezjem (*Hard Core*)
 - ▶ programirljivo vezje in mikroprocesor na istem čipu
 - ▶ bolj zmogljiva / manj prilagodljiva izvedba
 - ▶ ARM-9 (Altera in Xilinx)

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

Primer: učni mikroprocesor CPE4

- ▶ N-bitni procesor (N=12), 16 mikrooperacij (ukazov)
 - ▶ CPE in programski pomnilnik
 - ▶ ALE z akumulatorjem



Nabor ukazov

subtype koda is unsigned(3 downto 0);

constant lda: koda := "0001"; -- $a = [M]$

constant sta: koda := "0010"; -- $[M] = a$

constant add: koda := "0100"; -- $a = a + [M]$

constant sub: koda := "0101"; -- $a = a - [M]$

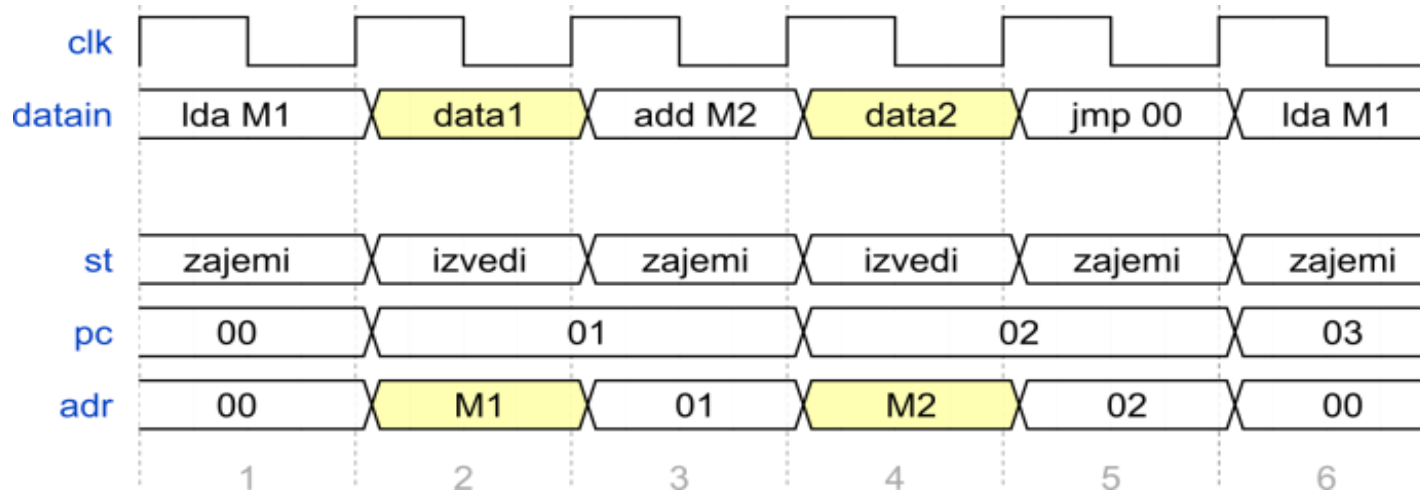
constant anda: koda := "0110"; -- $a = a \text{ and } [M]$

constant ora: koda := "0111"; -- $a = a \text{ or } [M]$

constant jmp: koda := "1000"; -- *jump*

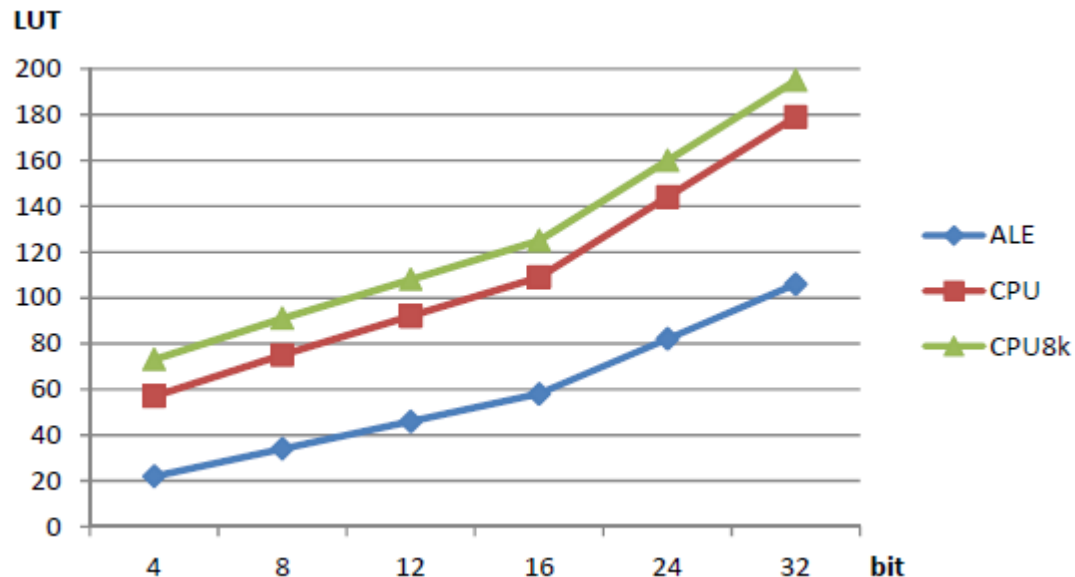
constant jze: koda := "1001"; -- jump if a=0

- ▶ potek izvajanja: dva cikla/ukaz – zajemi, izvedi



VHDL opis in sinteza procesorja

```
if st=zajemi then    -- shrani kodo ukaza
    code <= instr;
elsif st=izvedi then -- izvedi ukaz z akumulatorjem
    case code is
        when lda      => akum <= dat_i;
        when add      => akum <= akum + dat_i;
        when sub      => akum <= akum - dat_i;
        when anda     => akum <= akum and dat_i;
        when ora      => akum <= akum or dat_i;
        when others   => null;
    end case;
end if;
```



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

LNIV irtuaLAB

Home

Mikroprocesor CPU 12

Ukazi: NOTA, LDA, STA, INP, JMP, JZE, JCS, OUTP, ADD, SBT, CALL, RET, ANDA, ORA, SHL, SHR

Direktive: DB, DI, DO

Zbirni jezik:

```
start:  inp  vh      ;beri  vh
        jze  start   ;skoči nazaj dokler je vh=0
        add  vsota  ;prištej k vsoti in shrani
        sta  vsota
        outp izh   ;vsoto na izhod
        jmp  start   ;skok na začetek

vh     di  0
vsota  db  0
izh   di  1
```

Vhodi

vh	1

Simulacija

PC: 02

Akum: 01

Izhodi

IZH = 1
IZH = 2
IZH = 3

Vsebina pomnilnika:

@00	300	500	806	206	701	400	003	000
@08	000	000	000	000	000	000	000	000
@10	000	000	000	000	000	000	000	000
@18	000	000	000	000	000	000	000	000
@20	000	000	000	000	000	000	000	000
@28	000	000	000	000	000	000	000	000
@30	000	000	000	000	000	000	000	000
@38	000	000	000	000	000	000	000	000
@40	000	000	000	000	000	000	000	000
@48	000	000	000	000	000	000	000	000
@50	000	000	000	000	000	000	000	000
@58	000	000	000	000	000	000	000	000
@60	000	000	000	000	000	000	000	000
@68	000	000	000	000	000	000	000	000
@70	000	000	000	000	000	000	000	000
@78	000	000	000	000	000	000	000	000
@80	000	000	000	000	000	000	000	000
@88	000	000	000	000	000	000	000	000
@90	000	000	000	000	000	000	000	000
@98	000	000	000	000	000	000	000	000
@a0	000	000	000	000	000	000	000	000
@a8	000	000	000	000	000	000	000	000
@b0	000	000	000	000	000	000	000	000
@b8	000	000	000	000	000	000	000	000
@c0	000	000	000	000	000	000	000	000
@c8	000	000	000	000	000	000	000	000
@d0	000	000	000	000	000	000	000	000
@d8	000	000	000	000	000	000	000	000

Povzetek

- ▶ Opiši potek načrtovanja digitalnih sistemov.
 - ▶ Uporabljeni graniki sistema, delitev vezja.
- ▶ Navedi glavne gradnike mikroprocesorja.
 - ▶ Opiši nekaj glavnih sestavnih gradnikov.
 - ▶ V čem se zmogljivi procesorji razlikujejo od enostavnih ?
- ▶ Kako je sestavljen mikroprocesor za vgrajene naprave (mikrokrmilnik) ?