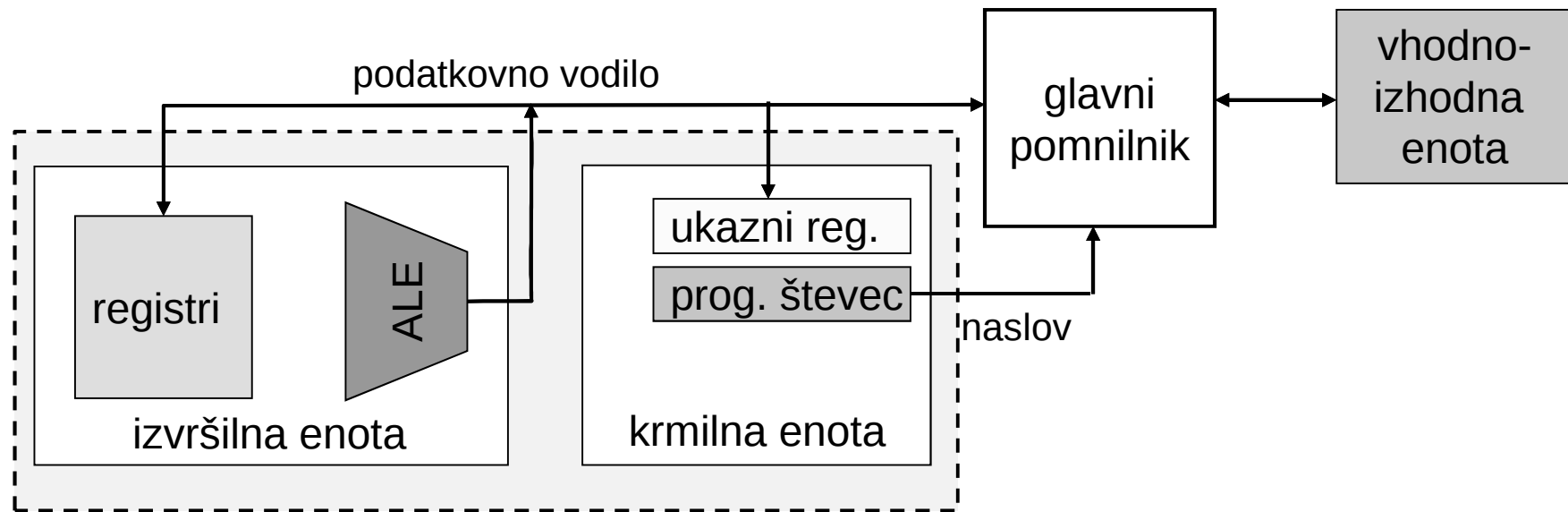


Digitalni sistemi z mikroprocesorjem

- ▶ Mikroprocesor na integriranem vezju vsebuje izvršilno in kontrolno enoto
- ▶ Mikroprocesor potrebuje za delovanje:
 - ▶ zunanjo uro in reset
 - ▶ zunanji pomnilnik s programskimi ukazi in podatki
 - ▶ vhodno in 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)

Von Neumannov model računalnika

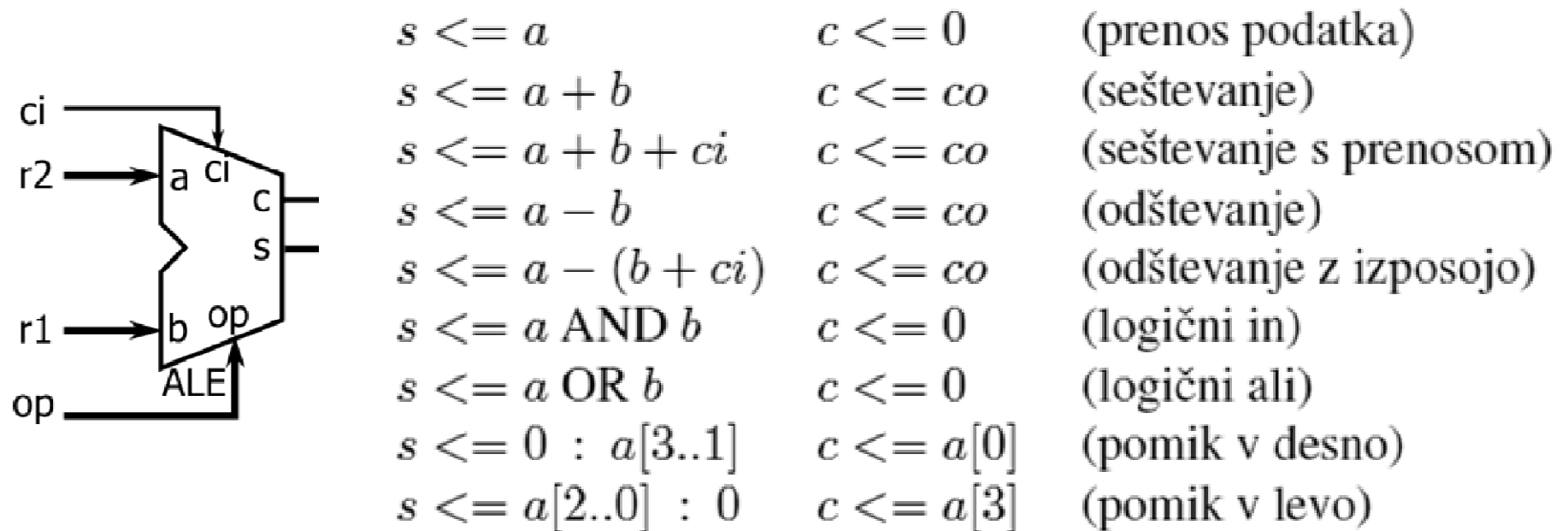
- ▶ Centralno procesna enota (CPE) in pomnilnik
 - ▶ vhodno – izhodna enota skrbi za komunikacijo z zunanjo svetom



- ▶ delovanje CPE določa nabor ukazov
 - ▶ ukazi so prilagojeni programskemu jeziku (C/C++)

Aritmetično logična enota – ALE

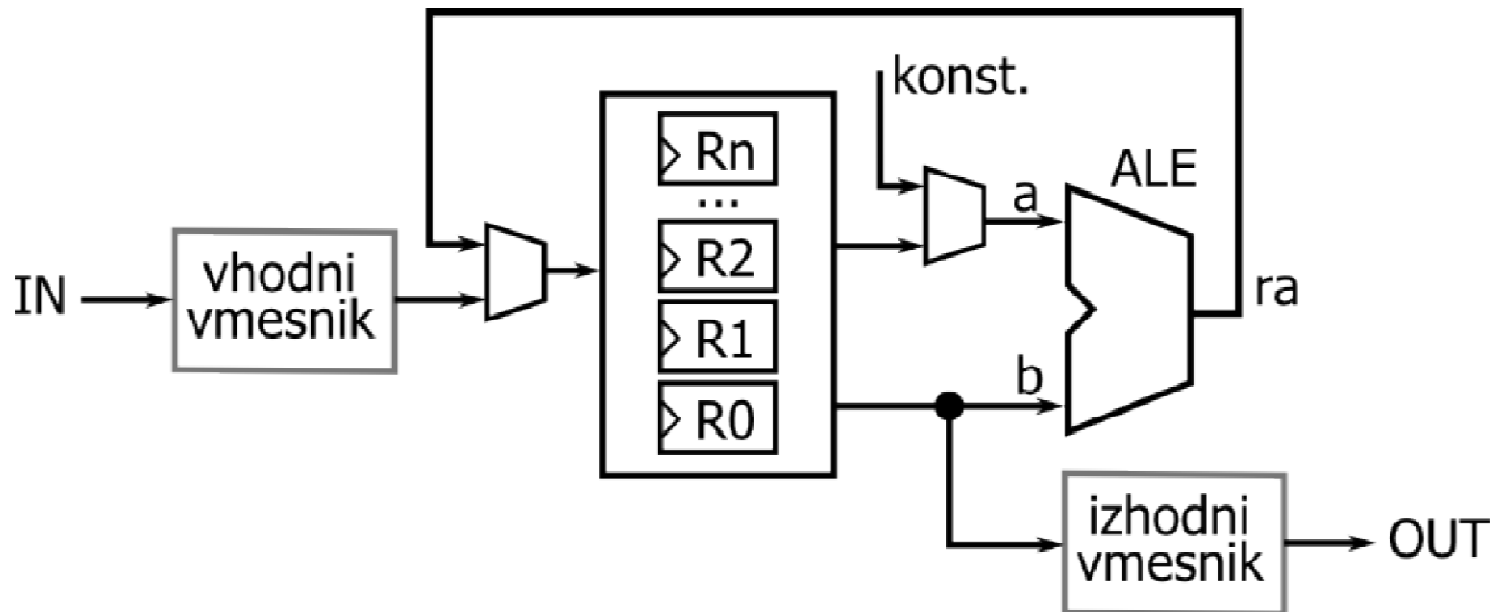
- ▶ ALE izvaja različne mikrooperacije, kot npr:



- ▶ ALE je narejena s povezavo več registrskih celic
 - ▶ npr. 8-bitna ALE vsebuje 8 registrskih celic

Podatkovna pot z registri

- ▶ Vhodni podatki za ALE pridejo iz registrov ali vhodnega vmesnika, en podatek je lahko konstanta



Mikrooperacije v zbirniku

- Ukazi v zbirniku predstavljajo procesorske mikrooperacije

mikrooperacija	zbirnik	pomen
$ra \leq rb$	LOAD ra, rb	prenos podatka med registri
$ra \leq k$	LOAD ra, K	prenos konstante K iz pomnilnika
$ra \leq ra + rb$	ADD ra, rb	vsota dveh registrov
$ra \leq ra + k$	ADD ra, K	vsota registra in konstante
$ra \leq ra \text{ OR } rb$	OR ra, rb	logični ali med registri
$ra \leq 0 : ra[3..1]$	SR0 ra	pomik registra v desno

- Prenos podatkov iz perifernih enot

$ra \leq IN$	IN ra, ID	prenos iz vhodne enote ID
$OUT \leq rb$	OUT rb, ID	prenos iz registra na izhodno enoto ID

Program v zbirniku

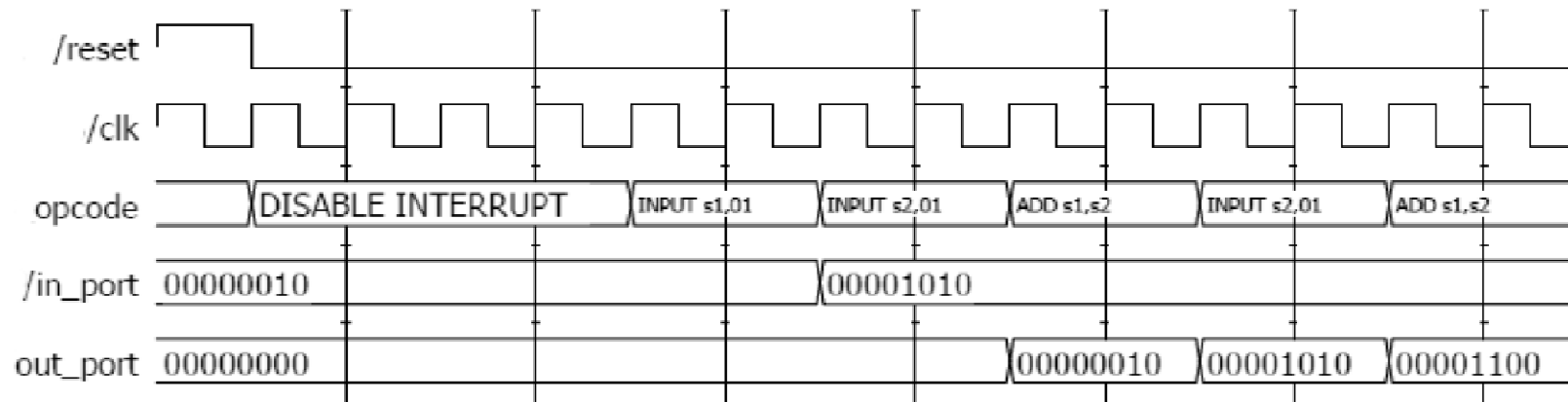
► Program izračuna povprečje štirih vrednosti

```
start:      IN      s1 , VHD    ; beri 1. podatek
            IN      s2 , VHD    ; 2. podatek
            ADD     s1 , s2      ; vsota
            IN      s2 , VHD    ; 3. podatek
            ADD     s1 , s2      ; vsota
            IN      s2 , VHD    ; 4. podatek
            ADD     s1 , s2      ; vsota
            SR0     s1           ; deli z 2
            SR0     s1           ; deli z 2
            OUT     s1 , IZH     ; na izhod
```

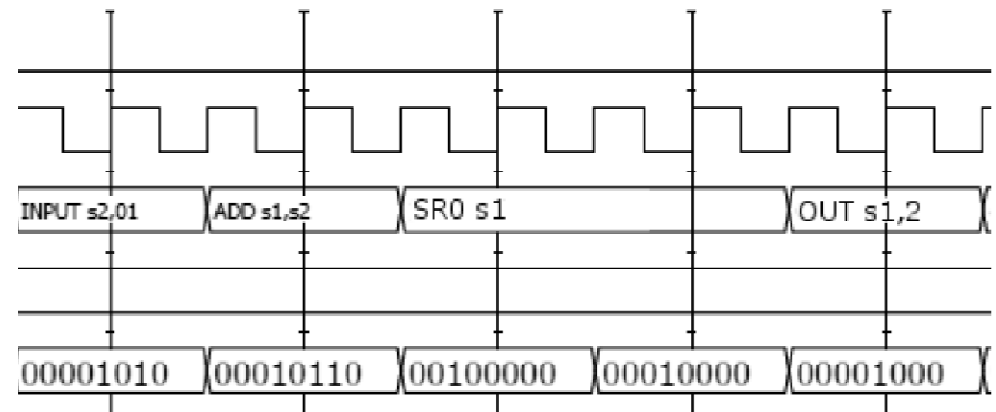
- Podatke beremo iz vhodne enote, izračunamo vsoto in jo delimo s 4 z dvema zaporednima operacijama pomika
 - pomik v desno SR0 predstavlja deljenje z 2

Simulacija procesorja

- ▶ Delovanje procesorja lahko opazujemo na simulatorju
 - ▶ primer na procesorju Picoblaze v programirljivem vezju

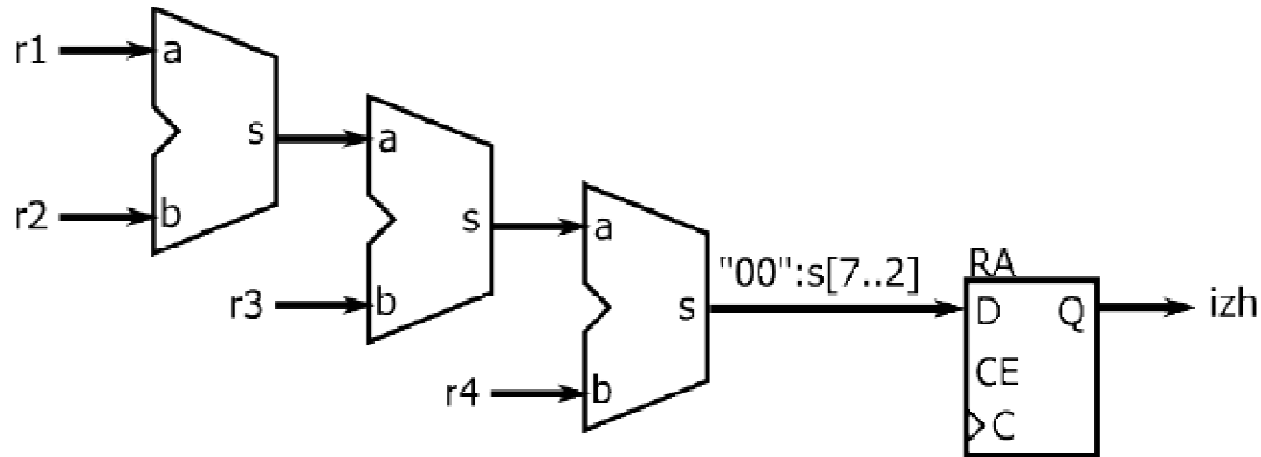


- ▶ Procesor izračuna izhod v 20 urnih ciklih



Izračun povprečja z digitalnim vezjem

- ▶ V vezju lahko paralelno izvedemo več mikrooperacij
- ▶ Porazdeljena obdelava podatkov
- ▶ Rezultat dobimo v enem urnem ciklu !



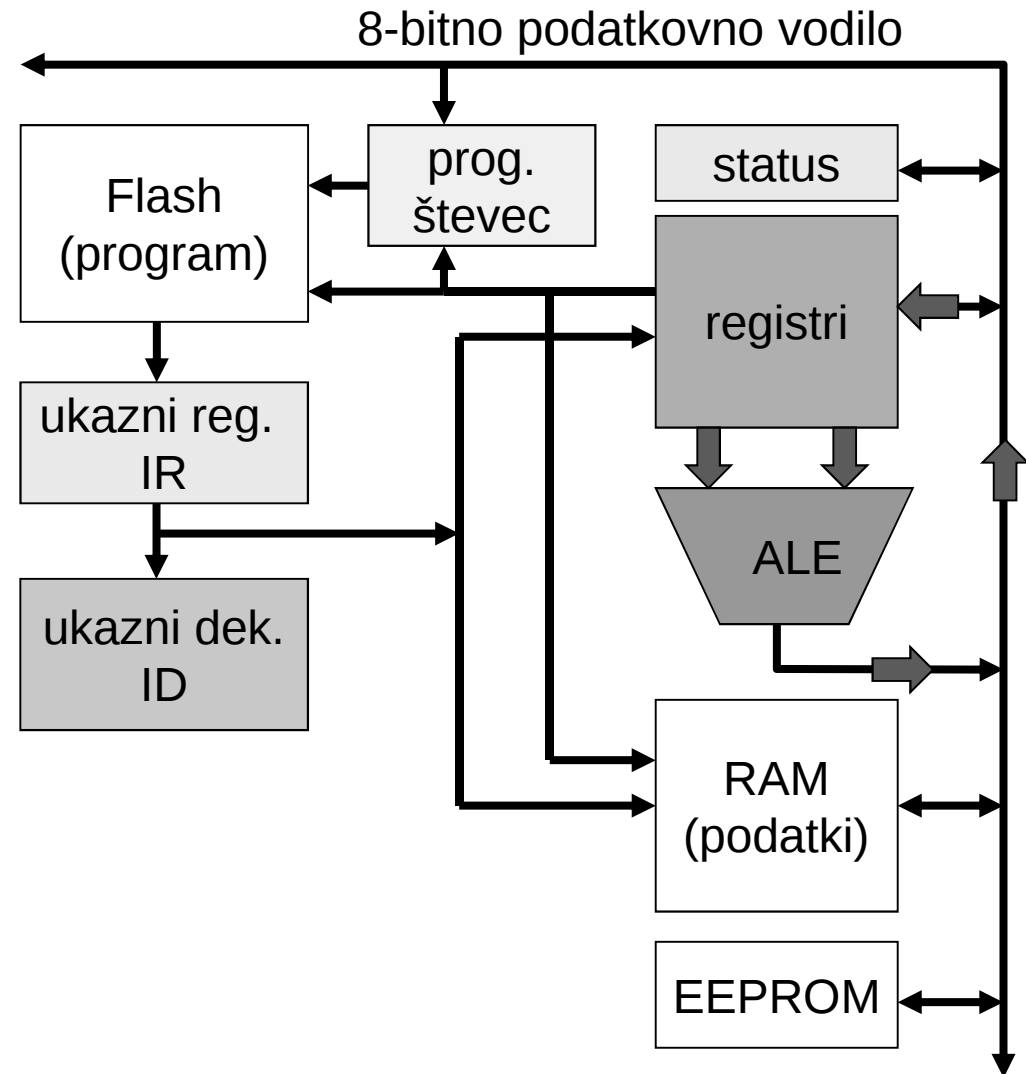
- ▶ Prednost porazdeljene obdelave podatkov: hitrost
- ▶ Slabost: velikost vezja, pri vsaki spremembi je potrebno razviti novo vezje (razvoj programa je hitrejši)

Mikroprocesorji za vgrajene naprave

- ▶ Mikrokrmilniki vsebujejo;
 - ▶ mikroprocesorsko jedro (izvršilno in krmilno enoto),
 - ▶ programski in delovni pomnilnik,
 - ▶ ter različne V/I vmesnike:
 - ▶ 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 mikrokrmilnikov
 - ▶ Atmel AVR, Microchip PIC, Intel 8051: 8-bitni procesor
 - ▶ ARM-7, ARM-9: 32-bitni procesor

Procesorsko jedro AVR

- ▶ Kontrolni del: Flash progr. pomnilnik
- ▶ programski števec vsebuje naslov naslednjega ukaza
- ▶ Registri in ALE tvorijo podatkovno pot ➡⬆
- ▶ RAM shranjuje podatke
- ▶ EEPROM trajno shrani podatke
 - ▶ počasen dostop, omejeno število vpisov v pomnilnik



Procesorsko jedro ARM-7

- Ukaze izvršuje v treh ciklih:

