

Zadatak 1. Bitovi podatka i koda su raspoređeni u 12-bitnoj reči 0010 1001 1111. Primenom Hamingovog koda, odrediti da li je došlo do greške u podatku.

Rešenje.

| | | | | | | | | | | | | |
|-----------------------|------|------|------|------|------|------|------|------|------|------|------|------|
| Pozicija bita | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| Broj pozicije | 1100 | 1011 | 1010 | 1001 | 1000 | 0111 | 0110 | 0101 | 0100 | 0011 | 0010 | 0001 |
| Bit | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| Bit podatka | D8 | D7 | D6 | D5 | | D4 | D3 | D2 | | D1 | | |
| Bit za proveru | | | | | C8 | | | | C4 | | C2 | C1 |

Bitovi koda koji se nalaze u podatku su:

$$C1 = 1$$

$$C2 = 1$$

$$C4 = 1$$

$$C8 = 1$$

Ispravni bitovi koda prema Hamingovom kodu su:

$$C1 = D1 \oplus D2 \oplus D4 \oplus D5 \oplus D7 = 1 \oplus 1 \oplus 0 \oplus 0 \oplus 0 = 0$$

$$C2 = D1 \oplus D3 \oplus D4 \oplus D6 \oplus D7 = 1 \oplus 0 \oplus 0 \oplus 1 \oplus 0 = 0$$

$$C4 = D2 \oplus D3 \oplus D4 \oplus D8 = 1 \oplus 0 \oplus 0 \oplus 0 = 1$$

$$C8 = D5 \oplus D6 \oplus D7 \oplus D8 = 0 \oplus 1 \oplus 0 \oplus 0 = 1$$

Formiranje sindroma:

$$\begin{array}{cccc}
 & C8 & C4 & C2 & C1 \\
 & 1 & 1 & 1 & 1 \\
 \oplus & 1 & 1 & 0 & 0 \\
 \hline
 & 0 & 0 & 1 & 1
 \end{array}$$

Pošto u sindromu postoje dve jedinice, to vrednost sindroma određuje broj pozicije koja je u grešci. U ovom slučaju, bit na poziciji 3, odnosno bit D1 je pogrešno prenet i treba da ima vrednost 0.

Zadatak 2. Hard disk ima 8652 cilindra, 12 staza/cilindru, 543 sektora/stazi, a veličina sektora je 512 B. Broj obrtaja magnetnog diska je 7200 obrtaja/min, pri čemu je minimalno vreme pozicioniranja 1,4 ms, a srednje vreme pozicioniranja 10,8 ms. Odrediti:

- Kapacitet diska u GB;
- Vreme potrebno za iščitavanje 512 kB podataka koji su sekvencijalno raspoređeni na disku;

c) Vreme potrebno za iščitavanje 512 kB podataka koji su slučajno raspoređeni na disku.

Rešenje:

a) $C = 8652 \cdot 12 \cdot 543 \cdot 512 = 28\,864\,733\,184 \text{ B} = 28\,188\,216 \text{ kB} = 27\,527,5547 \text{ MB} = 26,88 \text{ GB}$

b) $N_{sek} = \frac{512 \text{ kB}}{512 \text{ B}} = 1024 \rightarrow 512 \text{ kB podataka se nalazi na } 1024 \text{ sektora na hard disku}$

$N_{st} = \frac{1024 \text{ sektora}}{543 \text{ sektora/stazi}} = 1,88 \rightarrow 512 \text{ kB podataka se nalazi na } 1,88 \text{ stazi na disku}$

Vreme pristupa jednoj stazi se računa prema sledećem obrascu:

$$T_{pr} = T_{poz_{sr}} + \frac{1}{2r} + \frac{b}{rN}$$

$r = 7200 \text{ ob/min} = 120 \text{ ob/s}$

$T_{poz_{sr}} = 10,8 \text{ ms}$

$N = 543 \cdot 512 = 278\,016 \text{ B} = 271,5 \text{ kB} \rightarrow \text{kapacitet staze}$

$b = b_1 + b_2 \rightarrow \text{pročitaće se jedna cela staze i } 88\% \text{ treće staze}$

$b_1 = N = 271,5 \text{ kB}$

$b_2 = 0,88 \cdot N = 238,92 \text{ kB}$

Prema tome, ukupno vreme potrebno za iščitavanje 512 kB podataka je:

$$T_{pr} = T_{poz_{sr}} + \frac{1}{2r} + \frac{b_1}{rN} + T_{poz_{min}} + \frac{1}{2r} + \frac{b_2}{rN}$$

traženje
rotaciono
čitanje
prelazak
rotaciono
čitanje
prve
kašnjenje
prve
na
kašnjenje
druge
staze

staze
susednu

stazu

$T_{poz_{min}} = 1,4 \text{ ms}$

$$T_{pr} = 10,8 \text{ ms} + \frac{1}{2 \cdot 120} + \frac{271,5 \text{ kB}}{120 \cdot 271,5 \text{ kB}} + 1,4 \text{ ms} + \frac{1}{2 \cdot 120} + \frac{238,92 \text{ kB}}{120 \cdot 271,5 \text{ kB}}$$

$T_{pr} = 10,8 \text{ ms} + 0,0042 \text{ s} + 0,0083 \text{ s} + 1,4 \text{ ms} + 0,0042 \text{ s} + 0,0073 \text{ s}$

$T_{pr} = 10,8 \text{ ms} + 4,2 \text{ ms} + 8,3 \text{ ms} + 1,4 \text{ ms} + 4,2 \text{ ms} + 7,3 \text{ ms}$

$T_{pr} = 36,2 \text{ ms}$

$$c) N_{sek} = \frac{512 \text{ kB}}{512 \text{ B}} = 1024 \rightarrow 512 \text{ kB podataka se nalazi na 1024 sektora na hard disku}$$

$$b = 512 \text{ B}$$

$$N = 271,5 \text{ kB}$$

Pošto se podaci nalaze nasumično raspoređeni na disku, to znači da u najgorem slučaju moramo pročitati svih 1024 sektora na kojima se može naći po 512 B našeg podatka od 512 kB, tako da je srednje vreme pristupa:

$$T_{pr} = 1024 \cdot \left(T_{poz_{sr}} + \frac{1}{2r} + \frac{b}{rN} \right)$$

$$T_{pr} = 1024 \cdot \left(10,8 \text{ ms} + \frac{1}{2 \cdot 120 \text{ s}} + \frac{512}{120 \cdot 271,5 \text{ kB}} \right)$$

$$T_{pr} = 1024 \cdot (0,0108 \text{ s} + 0,0042 \text{ s} + 0,0000153 \text{ s}) = 15,3757 \text{ s}$$

Zadatak 3. Posmatra se procesor kod koga su i podaci i adrese širine 2 bajta, a niži bajt se smešta na višu adresu (*Big-Endian*). Širina memorijske lokacije iznosi 8 bita. Postoje tri linije za spoljašnje maskirajuće prekide (IRQ0, IRQ1, IRQ2), pri čemu su prioriteti za IRQ0, IRQ1 i IRQ2 01, 10 i 11, respektivno. Ne prihvata se prekid istog nivoa. Na prekid ne reaguju instrukcije: INTE, INTD, TRPE, TRPD, INT, RTI. Unutrašnji procesorski prekid (*Fault*) nastaje pri korišćenju neposrednog adresiranja za određeni operand. Pri prekidu na steku se čuva prvo PSW pa PC, a potom se brišu bitovi I i T u PSW registru. Stek raste od nižih ka višim adresama, a SP ukazuje na poslednju zauzetu lokaciju. Početna vrednost SP je 10F0. Struktura registra PSW je:

| | | | | | | | | | | | | | | | |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| I | T | L1 | L0 | P | - | - | - | - | - | - | - | V | C | Z | N |

Tabela prekidnih rutina započinje od memorijske adrese AF00 i ima 256 ulaza. Prekidne rutine pojedinačnih prekida su date u sledećoj tabeli:

| Prekid | Adresa prekidne rutine | Broj ulaza u IVT |
|--------------|------------------------|------------------|
| IRQ0 | A52Fh | 2 |
| IRQ1 | 25F1h | 3 |
| IRQ2 | 3904h | 4 |
| NMI | AB25h | 5 |
| Fault | A9C3h | 6 |
| Trap | 67E6h | 7 |

Adrese registara broja ulaza periferija IRQ0, IRQ1 i IRQ2 su FF00, FF10 i FF20, respektivno. Memorijski i ulazno-izlazni adresni prostor su razdvojeni.

- a) Napisati program kojim se periferijama dodeljuju brojevi ulaza u IV tabeli.
- b) Nacrtati izgled dela memorije koji sadrži tabelu prekidnih rutina.
- c) Posmatra se instrukcija dužine 4 bajta na adresi

100h: ADD #3h

Kontekst:

PSWI=1, PSWT=0, PSWL₁₀=01, IMR=111b

IRQ0=0, IRQ1=1, IRQ2=0, IRQN=0

Odrediti adresu naredne instrukcije. U slučaju da dolazi do prekida, objasniti da li se prekid prihvata i zašto, takođe, nacrtati i izgled steka u slučaju da se prekid prihvata. Napisati kontekst procesora nakon izvršenja navedene instrukcije.

- d) Posmatra se instrukcija dužine 2 bajta na adresi

50h: SUB #3h

Kontekst:

PSWI=1, PSWT=0, PSWL₁₀=10, IMR=111b

IRQ0=0, IRQ1=1, IRQ2=0, IRQN=1

Odrediti adresu naredne instrukcije. U slučaju da dolazi do prekida, objasniti da li se prekid prihvata i zašto, takođe, nacrtati i izgled steka u slučaju da se prekid prihvata. Napisati kontekst procesora nakon izvršenja navedene instrukcije.

- e) Posmatra se instrukcija dužine 2 bajta na adresi

50h: STORE #5h

Kontekst:

PSWI=1, PSWT=0, PSWL₁₀=01, IMR=111b

IRQ0=0, IRQ1=1, IRQ2=0, IRQN=1

Odrediti adresu naredne instrukcije. U slučaju da dolazi do prekida, objasniti da li se prekid prihvata i zašto, takođe, nacrtati i izgled steka u slučaju da se prekid prihvata. Napisati kontekst procesora nakon izvršenja navedene instrukcije.

f) Posmatra se instrukcija dužine 1 bajt na adresi

100h: INT #7h

Kontekst:

PSWI=1, PSWT=0, PSWL₁₀=01, IMR=111b

IRQ0=0, IRQ1=1, IRQ2=0, IRQN=0

Odrediti adresu naredne instrukcije. U slučaju da dolazi do prekida, objasniti da li se prekid prihvata i zašto, takođe, nacrtati i izgled steka u slučaju da se prekid prihvata. Napisati kontekst procesora nakon izvršenja navedene instrukcije.

g) Posmatra se instrukcija dužine 4 bajta na adresi

100h: ADD #3h

Kontekst:

PSWI=1, PSWT=0, PSWL₁₀=01, IMR=011b

IRQ0=0, IRQ1=1, IRQ2=1, IRQN=0

Odrediti adresu naredne instrukcije. U slučaju da dolazi do prekida, objasniti da li se prekid prihvata i zašto, takođe, nacrtati i izgled steka u slučaju da se prekid prihvata. Napisati kontekst procesora nakon izvršenja navedene instrukcije.

Rešenje:

a) **LOAD #2**
OUT FF00h
LOAD #3
OUT FF10h
LOAD #4
OUT FF20h

- b) Pošto su adrese širine 16 bita, a širina memorijske lokacije je 8 bita, to svaki ulaz u tabeli prekidnih rutina zauzima dve susedne lokacije u memoriji. Pošto se koristi *Big-endian* princip, niži bajt adrese se smešta na višu adresu u memoriji. Početna adresa tabele prekidnih rutina je AF00, a njen izgled je prikazan na sledećoj slici:

| Prekid | Broj ulaza | Adresa | Sadržaj memorije |
|--------|------------|--------|------------------|
| | 8 | AF11 | |
| | | AF10 | |
| Trap | 7 | AF0F | E6h |
| | | AF0E | 67h |
| Fault | 6 | AF0D | C3h |
| | | AF0C | A9h |
| NMI | 5 | AF0B | 25h |
| | | AF0A | ABh |
| IRQ2 | 4 | AF09 | 04h |
| | | AF08 | 39h |
| IRQ1 | 3 | AF07 | F1h |
| | | AF06 | 25h |
| IRQ0 | 2 | AF05 | 2Fh |
| | | AF04 | A5h |
| | 1 | AF03 | |
| | | AF02 | |
| | 0 | AF01 | |
| | | AF00 | |

- c) Prihvata se IRQ1, jer je I = 1, ima veći prioritet od trenutnog i drugi bit registra IMR je 1. Prelazi se na 25F1h.

Stek:

| | | |
|-----------|-----------|-----------------|
| 10F0 | | |
| 10F1 | 1001xxxxb | Viši bajt PSW-a |
| 10F2 | xxxxxxxxb | Niži bajt PSW-a |
| 10F3 | 01h | Viši bajt PC-a |
| SP → 10F4 | 04h | Niži bajt PC-a |

Kontekst nakon izvršenja instrukcije (menja se nivo trenutnog prioriteta i linija IRQ1 se postavlja na 0, kao i bit I):

PSWI=0, PSWT=0, PSWL₁₀=10, IMR=111b

IRQ0=0, IRQ1=0, IRQ2=0, IRQN=0

- d) Postoje dva prekida, usled NMI i IRQ1. Pošto NMI ima veći prioritet, on se izvršava. Prelazi se na AB25h.

Stek:

| | | |
|-----------|-----------|-----------------|
| 10F0 | | |
| 10F1 | 1001xxxxb | Viši bajt PSW-a |
| 10F2 | xxxxxxxxb | Niži bajt PSW-a |
| 10F3 | 00h | Viši bajt PC-a |
| SP → 10F4 | 52h | Niži bajt PC-a |

Kontekst nakon izvršenja instrukcije (linija NMI se postavlja na 0, kao i bit I):

PSWI=0, PSWT=0, PSWL₁₀=01, IMR=111b

IRQ0=0, IRQ1=1, IRQ2=0, IRQN=0

- e) Postoje tri prekida: *Fault*, NMI i IRQ1. *Fault* ima najveći prioritet, tako da se on izvršava. Prelazi se na A9C3.

Stek (pamti se adresa prekinute instrukcije):

| | | |
|-----------|-----------|-----------------|
| 10F0 | | |
| 10F1 | 1001xxxxb | Viši bajt PSW-a |
| 10F2 | xxxxxxxxb | Niži bajt PSW-a |
| 10F3 | 00h | Viši bajt PC-a |
| SP → 10F4 | 50h | Niži bajt PC-a |

Kontekst nakon izvršenja instrukcije (bit I se postavlja na nulu, a NMI i IRQ1 ostaju na 1 jer se njihov prekid nije obradio):

PSWI=0, PSWT=0, PSWL₁₀=01, IMR=111b

IRQ0=0, IRQ1=1, IRQ2=0, IRQN=1

- f) Postoje dva prekida, usled instrukcije INT i od IRQ1. Instrukcija INT ima najveći prioritet, tako da se ona izvršava. Adresni deo instrukcije određuje broj ulaza u tabeli prekidnih rutina odakle se čita adresa prekidne rutine. U ovom slučaju, radi se o ulazu broj 7. Prelazi se na 67E6.

Stek:

| | | |
|-----------|-----------|-----------------|
| 10F0 | | |
| 10F1 | 1001xxxxb | Viši bajt PSW-a |
| 10F2 | xxxxxxxxb | Niži bajt PSW-a |
| 10F3 | 01h | Viši bajt PC-a |
| SP → 10F4 | 01h | Niži bajt PC-a |

Kontekst nakon izvršenja instrukcije (bit I se postavlja na nulu):

PSWI=0, PSWT=0, PSWL₁₀=01, IMR=111b

IRQ0=0, IRQ1=1, IRQ2=0, IRQN=0

- g) Postoje prekidi usled IRQ1 i IRQ2. Pošto IRQ2 ima veći prioritet, on bi trebalo da se izvršava. Međutim, bit 2 u registru maske (IMR) je jednak nuli, tako da IRQ2 ne može da se izvrši. IRQ1 može da se izvrši, jer su ispunjeni svi uslovi: ne postoji prekid višeg prioriteta, I = 1, IMR₁ = 1, prioritet od IRQ1 (10) je veći od trenutnog (01). Prelazi se na 25F1.

Stek:

| | | |
|-----------|-----------|-----------------|
| 10F0 | | |
| 10F1 | 1001xxxxb | Viši bajt PSW-a |
| 10F2 | xxxxxxxxb | Niži bajt PSW-a |
| 10F3 | 01h | Viši bajt PC-a |
| SP → 10F4 | 04h | Niži bajt PC-a |

Kontekst nakon izvršenja instrukcije (menja se nivo trenutnog prioriteta i linija IRQ1 se postavlja na 0, kao i bit I):

PSWI=0, PSWT=0, PSWL₁₀=10, IMR=011b

IRQ0=0, IRQ1=0, IRQ2=1, IRQN=0

Zadatak 4. RAID sistem se sastoji od tri nezavisna diska sa sledećim karakteristikama:

| Disk | Kapacitet (GB) | Vreme pristupa (ms) | Pouzdanost (%) |
|------|----------------|---------------------|----------------|
| 1 | 100 | 10 | 98 |
| 2 | 100 | 10 | 98 |
| 3 | 250 | 8 | 97 |

Odrediti kapacitet, vreme pristupa i verovatnoću otkaza u slučaju:

- RAID 0,
- RAID 1,
- RAID 4, gde se poslednji disk koristi za skladištenje bita parnosti.

Rešenje:

a)

$$C_{RAID0} = \sum_{i=1}^3 C_i = 100 + 100 + 250 = 450 \text{ GB}$$

$$\frac{1}{t_{RAID0}} = \sum_{i=1}^3 \frac{1}{t_i} = \frac{1}{10} + \frac{1}{10} + \frac{1}{8} \Rightarrow t_{RAID0} = 3,077 \text{ ms}$$

Pošto je data pouzdanost, verovatnoća otkaza svakog od diskova se dobija kao razlika do 100%. Tako je verovatnoća otkaza prvog diska 2% ili 0,02, drugog 2% ili 0,02 i trećeg 3% ili 0,03.

$$p_{RAID0} = \sum_{i=1}^3 p_i = 0,02 + 0,02 + 0,03 = 0,07 = 7\%$$

b)

$$C_{RAID1} = \min(C_i) = 100 \text{ GB}$$

$$t_{RAID1} = \max(t_i) = 10 \text{ ms}$$

$$p_{RAID1} = \prod_{i=1}^3 p_i = 0,02 \cdot 0,02 \cdot 0,03 = 0,000012 = 0,0012\%$$

c)

$$C_{RAID4} = C_1 + C_2 = 200 \text{ GB}$$

$$\frac{1}{t_{RAID4}} = \frac{1}{t_1} + \frac{1}{t_2} = \frac{1}{10} + \frac{1}{10} \Rightarrow t_{RAID4} = 5 \text{ ms}$$

$$p_{RAID4} = p_1 \cdot p_2 \cdot (1 - p_3) + p_1 \cdot (1 - p_2) \cdot p_3 + (1 - p_1) \cdot p_2 \cdot p_3 + p_1 \cdot p_2 \cdot p_3 = 0,0016 = 0,16\%$$