

## פתרון מיקרו מתוך אלקטרוניקה מחשבים קיץ 2006

### שאלה 6

לפניך תת-שגרה, הכתובה בשפת הסף של המיקרו מעבד 8086/88 .

1. CHANGE: MOV SI , 20 H
2. MOV CL , 4 H
3. MOV BL , 6 H
4. CHG: MOV AL , [SI]
5. ROL AL , CL
6. MOV [SI] , AL
7. INC SI
8. DEC BL
9. JNZ CHG
10. RET

א. הסבר את ההוראות שמספריהן 4 , 5 , 9 , 10 .

ב. הסבר מה מבצעת תת-השגרה.

ג. להלן התכנים של תאי הזיכרון  $20 H + 25 H$  לפני ביצוע תת-השגרה:

25 H	24 H	23 H	22 H	21 H	20 H	<b>כתובת התא</b>
37 H	21 H	87 H	56 H	48 H	35 H	<b>תוכן התא</b>

מה יהיו תכני התאים  $20 H + 25 H$  לאחר ביצוע תת-השגרה?

### פתרון שאלה 6

א.

chg: mov al,[si]

העבר אל רגיסטר AL נתון בגודל ביית מהזיכרון מהכתובת עליה מצביע רגיסטר SI . chg היא תווית.

rol al,cl

סובב את רגיסטר AL שמאלה כמות פעמים לפי הנתון שברגיסטר CL . כל סיבית ברגיסטר AL זהו שמאלה כאשר סיבית ה MSB עוברת לסיבית ה LSB וגם לדגל הנשא. פעולה זו מתבצעת מספר פעמים לפי הנתון שב CL . בתרגיל כאן היות ו CL שווה 4 אז AL מסובב שמאלה 4 פעמים.

jnz chg

קפוץ אל התווית chg אם התוצאה של הפעולה האריתמטית או הלוגית האחרונה איננה 0 . כלומר אם בדגל האפס יש 0.

ret

פקודת חזרה (return) מפרוצדורה. כתובת החזרה נשלפת מהמחסנית וטוענת את רגיסטר IP - במקרה של פרוצדורה קרובה ) או את IP ו CS - במקרה של פרוצדורה רחוקה.

ב.

התכנית פועלת על בלוק בן 6 בתים החל מכתובת 20h ועד 25h . הנתון שבכל כתובת מועבר מהזיכרון אל AL ומסובב 4 פעמים שמאלה - פעולה זו מחליפה את 4 הסיביות הנמוכות עם 4 הגבוהות -

ומוחזר בחזרה אל הזיכרון. לדוגמא אם הנתון בכתובת 20h הוא 35h הרי לאחר פעולת הסיבוב הנתון יהיה 53h. נתון זה יוחזר לכתובת 20h. פעולה דומה עושים גם על 5 הכתובות הבאות. הרגיסטר SI מצביע על הכתובות בזיכרון, הרגיסטר BL משמש כמונה לולאה ומראה את כמות הפעמים שיש לחזור על הלולאה.

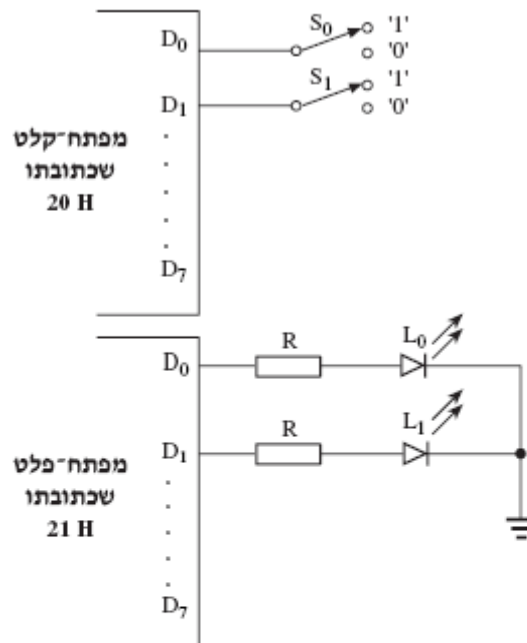
ג.

תוכן התאים אחרי התוכנית :

25H	24H	23H	22H	21H	20H	כתובת התא
73H	12H	78H	65H	84H	53H	תוכן התא

### שאלה 7

באיור לשאלה 7 נתונים מפתח-קלט, הנמצא בכתובת 20 H, ומפתח-פלט, הנמצא בכתובת 21 H.



איור לשאלה 7

כתוב תכנית בשפת ASM-86 שתבדוק את המצב של כל אחד מן המתגים  $S_0$  ו- $S_1$ , ובהתאם לו תקבע את מצב הנוריות  $L_0$  ו- $L_1$  כמפורט בטבלה שלהלן:

$S_1$	$S_0$	$L_1$	$L_0$
0	0	ON	ON
0	1	ON	OFF
1	0	OFF	ON
1	1	OFF	OFF

### פתרון שאלה 7

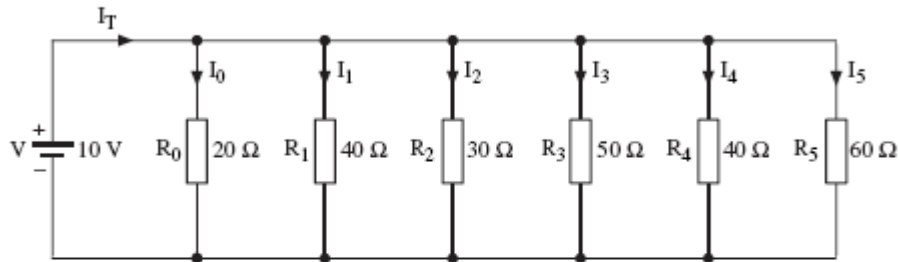
באיור לשאלה רואים שהלדים מופעלים ע"י 1 לוגי. כמו כן רואים מהטבלה שכל לד מופעל הפוך מהמפסק שלו כלומר אם מהמפסק נכנס 0 אז הלד דולק ולהפך.

מכאן שהתוכנית תיראה :

In al, 20h ; הכנס את מצב המפסקים ;  
 Not al ; הפוך את הביטים ;  
 Out 21h,al ; הוצא את המצב אל הלדים ;  
 Ret

### שאלה 8

באיור לשאלה 8 נתון מעגל חשמלי.



איור לשאלה 8

כתוב תכנית (בשפת C או בשפת VB) שתבצע את הפעולות האלה:

1. תגדיר מערך, ותציב את התנגדויות הנגדים שבמעגל במערך הזה.
2. תחשב את הזרם בכל אחד מן הנגדים ותדפיס את ערכו.
3. תחשב את הזרם  $I_T$  ואת ההתנגדות השקולה  $R_T$  של הנגדים במעגל, ותדפיס את ערכיהם.

**הערה:** יש להשתמש בלולאה לביצוע הנדרש בתכנית.

### פתרון שאלה 8

את התוכנית נרשום בשפת C. כדי להשתמש בפונקציה printf() נרשום: #include <stdio.h>

1. הגדרת המערך :

```
int resistors[6] = {20,40,30,50,40,60};
```

2.

```
int i;
float current;
for(i=0;i<6;i++)
{
    current = (float)10/resistors[i];
    printf("\nThe current in R%d = %f ", i,current);
}
```

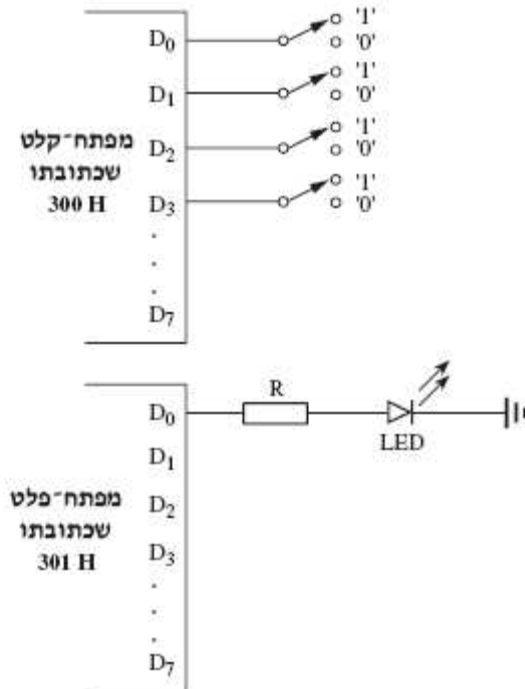
שים לב שעשינו הסבה מפורשת כדי לקבל ב current ערך ממשי.

3.

```
for (i=0,current=0; i<6;i++)
    current = current + (float)10/resistors[i];
printf("\nThe total current in the circuit = %f ",current);
printf ( "\nThe Equivalent Resistance is = %f ", 10/current);
```

שאלה 9

באיור לשאלה 9 נתונים מפתח-קלט שכתובתו 300 H , ומפתח-פלט שכתובתו 301 H .



איור לשאלה 9

כתוב תכנית (בשפת C או בשפת VB) המבצעת את הפעולות האלה:

1. קוראת נתון מספרי N ממפתח-הקלט.
  2. גורמת להבהוב של נורית ה-LED שבמוצא מפתח-הפלט N פעמים.
- הערה:** יש לבצע כל הבהוב של נורית ה-LED באופן הבא:  
הדלקת הנורית, השהיית התכנית למשך 0.2 sec , כיבוי הנורית והשהיית התכנית למשך 0.1 sec .

בהצלחה!

פתרון שאלה 9

את התוכנית נרשום בשפת C . נכלול את קובץ הכותרת dos.h בו יש פונקציות של קלט/פלט ופונקציות השהייה.

```
#include <dos.h>

1.
unsigned char n;
n=inportb(0x300);           ; הכנסת מצב המפסקים
n = n &0x0f;                ; ביטול 4 הביטים הגבוהים

2.
for( ; n>0 ; n--)
{
    outportb(0x301,1);
    delay(200);           ; 200MSsec = 0.2 second
    outportb(0x301,0);
    delay(100);
}
```