

**פתרון מבחן בגרות – מיקרו 2005**

**שאלה 6**

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

1. MB : MOV SI, 10 H
2. AG : MOV BX, [SI]
3. MOV CL, 8 H
4. ROL BX, CL
5. MOV [SI], BX
6. ADD SI, 2 H
7. CMP SI, 18 H
8. JNE AG
9. RET

להלן תכני התאים שכתובותיהם  $10 H \div 17 H$  בסגמנט הנתונים שבזיכרון, לפני ביצוע תת-השגרה:

17 H	16 H	15 H	14 H	13 H	12 H	11 H	10 H	<b>כתובת התא</b>
80 H	70 H	60 H	50 H	40 H	30 H	20 H	10 H	<b>תוכן התא</b>

- א. הסבר את ההוראות שמספריהן 2, 4, 7, 8.
- ב. הסבר מה מבצעת תת-השגרה, וציין מה יהיו ערכי התאים  $10 H \div 17 H$  לאחר ביצועה.
- ג. מחליפים את הוראה 4 בהוראה ROR BX, CL. האם ישפיע השינוי הזה על ערכי התאים  $10 H \div 17 H$  בסיום ביצוע תת-השגרה? נמק את תשובתך.

**פתרון שאלה 6**

א.

AG: MOV BX,[SI]

העבר לרגיסטר BX מילה מהזיכרון מהכתובת שרגיסטר SI מצביע עליה. למעשה מכתובת זו מועבר בית אל רגיסטר BL ומהכתובת [SI+1] מועבר בית אל BH. AG הוא תווית-LABEL.

ROL BX,CL

סובב שמאלה את הנתון שברגיסטר BX. כמות הסיבובים נתונה ברגיסטר CL. הזזה שמאלה פעם אחת אומרת שכל ביט ברגיסטר BX מוזז שמאלה כאשר ביט ה MSB מועבר אל ביט ה LSB וגם אל דגל הנשא. בפקודה כאן מזיזים את הנתון כמות פעמים לפי הנתון שב CL.

CMP SI,18H

בצע פעולת השוואה בין רגיסטר SI והנתון 18H . ההשוואה מתבצעת ע"י חיסור בין הנתון שברגיסטר SI לנתון 18H , התוצאה נשארת ביחידה האריתמטית לוגית והדגלים מושפעים. אם תוצאת החיסור היא 0 ( דגל ה 0 מקבל 1) יודעים ש SI שווה 18H.

#### JNE AG

קפוץ לתווית AG אם דגל האפס איננו '1' כלומר אם תוצאת הפעולה האריתמטית (או הלוגית) האחרונה לא נתנה אפס. במקרה שלנו תוצאת ההשוואה בין SI והנתון 18H אינם אפס, כלומר SI לא שווה 18H .

ב.

התוכנית פועלת על בלוק נתונים המתחיל בכתובת 10H ומסתיים בכתובת 17H (כולל). כל נתון בגודל מילה (2 בתים ) מועבר מהזיכרון אל רגיסטר BX מסובב שמאלה 8 פעמים ומוחזר בחזרה אל הזיכרון. הסיבוב שמאלה 8 פעמים מחליף את התכנים של שני הבתים. מה שהיה ברגיסטר BL עובר ל BH ולהיפך. הפעולה מתבצעת כך שרגיסטר CL מראה כמה פעמים יש לסובב את הנתון (8 פעמים בדוגמא כאן), SI הוא הרגיסטר בעזרתו פונים אל הזיכרון. הפרוצדורה מסתיימת כאשר SI הגיע לכתובת 18H .

מפת הזיכרון אחרי ריצת התוכנית :

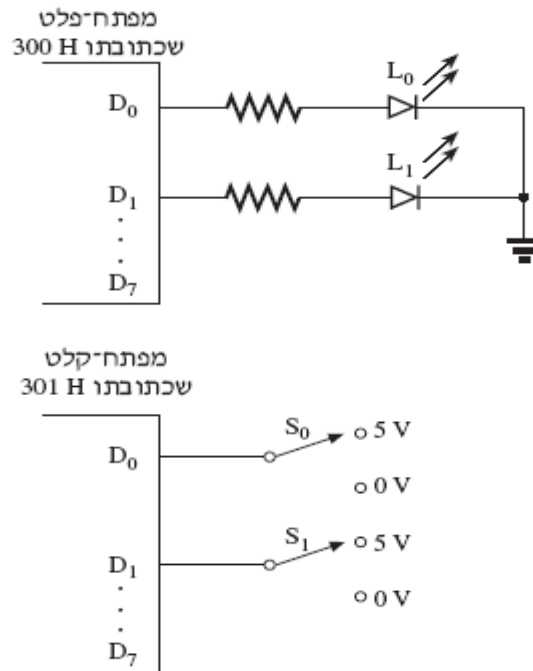
17H	16H	15H	14H	13H	12H	11H	10H	<b>כתובת התא</b>
70H	80H	50H	60H	30H	40H	10H	20H	<b>תוכן התא</b>

ג.

הפעולה מסובבת את הנתון שברגיסטר BX ימינה במקום שמאלה. היות ומסובבים 8 פעמים אז אין שינוי בנתון שב BX . מספר שונה מ 8 (או כפולות שלו) היה משנה את התוצאות.

**שאלה 7**

באיור לשאלה 7 מתוארים מפתח-פלט, שאליו מחוברות שתי נוריות LED ( $L_0$  ו- $L_1$ ), ומפתח-קלט, שאליו מחוברים שני מתגים ( $S_0$  ו- $S_1$ ).



**איור לשאלה 7**

כתוב תת-שגרה בשפת ASM-86, שתבדוק את מצב הסיביות D<sub>0</sub> ו- D<sub>1</sub> שבמפתח-הקלט. תת-השגרה תדליק את נוריות ה-LED ( $L_0$  ו- $L_1$ ) שבמפתח-הפלט, בהתאם לטבלה שלהלן:

D <sub>1</sub>	D <sub>0</sub>	L <sub>1</sub>	L <sub>0</sub>
0	0	ON	ON
0	1	ON	OFF
1	0	OFF	ON
1	1	OFF	OFF

**פתרון שאלה 7**

היות והלדים גדלקים כאשר נספק ליציאת פורט הפלט '1' (5 וולט) ולפי הטבלה רוצים להדליק אותן כאשר המפסק המתאים להן נמצא ב 0 אז יש להפוך את הביטים D<sub>0</sub> ו D<sub>1</sub>.

LEDS:

```
MOV DX,301H
IN AL,DX
NOT AL ; היפוך הביטים.
MOV DX,300H
OUT DX,AL
RET
```

## שאלה 8

בלוק נתונים, שכתובת ההתחלה שלו היא H 10 ואורכו שמונה תאים, מכיל נתונים מספריים. גודלו של כל נתון בבלוק הוא בית אחד (1 byte). כתוב תת-שגרה בשפת ASM-86, המונה את מספר התאים שערכם H 22 בבלוק, ומציבה את תוצאת המנייה בתא שכתובתו H 18.

## פתרון שאלה 8

COUNT22H:

MOV SI,10H	; כתובת התחלה למצביע הכתובות
MOV CX,8	; כמות הבתים שנבדוק
MOV BL,0	; מונה את כמות הפעמים שהנתון נמצא
CHK: MOV AL,[SI]	; העבר את הנתון מהזיכרון אל הרגיסטר
CMP AL,22H	; השווה עם 22 הקסה
JNZ CONT	; אם לא שווה
INC BL	; הגדל המונה אם הנתון שווה 22 הקסה
CONT: INC SI	; הגדל מצביע הכתובות
LOOP CHK	; חסר 1 מהרגיסטר ואם הוא לא שווה 0 קפוץ לתווית
MOV [SI],BL	; העבר את המספר שבמונה אל הזיכרון.
RET	

## שאלה 9

- א. הסבר את תפקידו של כל אחד מן ההדקים  $\overline{RD}$ , HOLD, NMI, ALE במיקרו-מעבד 8086.
- ב. השווה בין פסיקה לבין שאילתה מהבחינות האלה:
1. מהירות התגובה של המחשב.
  2. התוכנה הנדרשת.
- ג. לפניך קטע תכנית. ציין את תוכן האוגר AL ואת מצבם הלוגי של דגל הנשא ודגל האפס בסיום הביצוע של כל שורה.

1. MOV AL, 39 H
2. ADD AL, E8 H
3. DEC AL

## פתרון שאלה 9

א.

ALE - Address Latch Enable

אפשר נעילת כתובות. בעזרת רגל זו מודיע המעבד מתי יש בפס המרובב (כתובות ונתונים ברגלים AD0 – AD15 וכתובות וסטאטוס ברגלים A16-A19/S3-S6) כתובות ומתי נתונים (או סטאטוס ברגלים הגבוהות של הכתובות). כאשר יש '1' ברגל זו אז ברגליים המרובבות יש כתובות. רגל זו מתחברת בדרך כלל אל נועל כמו 373' אשר נועל את הכתובות ביציאה שלו כאשר מרגל ALE יש '1'. לאחר שהרגל חוזרת ל'0' הפס יכול להיות פס נתונים (או סטאטוס).

NMI – Non Mask able Interrupt

פסיקה שאיננה ניתנת למיסוך. זוהי אחת מרגלי הפסיקה של המעבד. כאשר ברגל זו יש '1' לוגי אז המעבד מסיים את הפקודה שבה הוא נמצא ונותן פסיקה מספר 2. פסיקה זו שימושית בדרך כלל בחירום ולא ניתן לחסום אותה בעזרת תוכנה (פקודת CLI) להבדיל מפסיקה שמתבקשת ברגל INTR של המעבד ואותה ניתן לחסום בעזרת תוכנה.

#### RD - READ

רגל זו היא רגל הקריאה של המעבד. היא פעילה בנמוך. כאשר המעבד מבצע מחזור קריאה מהזיכרון או מהתקן קלט הרגל יורדת ל '0' לזמן קצר ומתזמנת את פעולת הקריאה. רגל זו מתחברת לרגל הקריאה של רכיבי הזיכרון או התקני הקלט.

#### HOLD

רגל זו היא כניסה אל הרכיב. כאשר רכיב DMA (גישה ישירה אל הזיכרון Direct memory Access) רוצה לבצע פעולת DMA הוא שם ברגל זו '1'. כאשר המעבד נענה הוא מנתק את עצמו מהפסים ומודיע בעזרת רגל INTA שהוא נענה לפעולה. הרכיב שביקש את ה DMA מתחבר אל הפסים של המעבד ומבצע פעולת DMA. בסיום הפעולה הרכיב מנתק את עצמו מהפסים ומוריד את הרגל ל '0' ואז המעבד חוזר להתחבר אל הפסים שלו.

ב.

- שאילתא הוא מצב שבו המעבד פונה כל תקופת זמן אל התקן קלט או פלט ושואל אותו האם הוא צריך טיפול. פסיקה היא בקשה של רכיב שיטפלו בו בעזרת רגל INTR של המעבד.
1. יוזם הפנייה במקרה של שאילתא הוא המעבד. במקרה של פסיקה, הרכיב עצמו הוא זה שמבקש טיפול. מהירות הטיפול בפסיקה גבוהה מזה של שאילתא. כאשר יש בקשת פסיקה (ויש אפשר פסיקות) המעבד מסיים את הפקודה הנוכחית ומטפל במבקש הפסיקה. לעומת זאת בשאילתא נטפל ברכיב רק כאשר המעבד ייפנה אל הרכיב כאשר הוא יתפנה.
  2. התוכנה הנדרשת במקרה של שאילתא היא תוכנית המבצעת פניה (בדרך כלל מחזור קריאה) אל הרכיב ובודקת האם הוא צריך טיפול. אם לא אז פונים בשאילתא לרכיבים אחרים או חוזרים לתוכנית הראשית. אם כן עוברים לתוכנית המטפלת ברכיב. בפסיקה יש להכניס בטבלת וקטור הפסיקות, ב 4 כתובות עוקבות, את הכתובת שבה נמצאת התוכנית המטפלת ברכיב (התוכנית יכולה להיות אותה תוכנית כמו בשאילתא). פעולה זו נקראת התקנת DRIVER.

ג.

MOV AL, 39H

הדגלים לא מושפעים בפקודות העברת נתונים ונשארים במצב הקודם.  $AL = 39H$

ADD AL,E8H

$$AL = 39H + E8H = 21H, \quad CF = 1 \quad ZF = 0$$

DEC AL

$$AL = 20H \quad CF = 1 \quad ZF = 0$$

דגל הנשא לא מושפע מפעולת DEC ולכן נשאר 1 למרות שלא היה נשא בפקודה.

## שאלה 15

לפניך תת־שגרה בשפת ASM-86. תכני התאים 10 H ו־11 H לפני ביצוע התת־שגרה הם 35 H ו־48 H, בהתאמה.

```

1. SM_ON:  MOV  AL, 0 H
2.         MOV  CX, 8 H
3.         MOV  SI, 10 H
4.         MOV  BL, [SI]
5. ROT:    ROL  BL, 1 H
6.         JNC  NC
7.         INC  AL
8. NC:     DEC  CX
9.         JNZ  ROT
10.        MOV  [SI+1], AL
11.        RET
    
```

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

ב. הסבר מה מבצעת תת־השגרה.

ג. ציין את ערכי התאים 10 H ו־11 H לאחר ביצוע תת־השגרה.

## פתרון שאלה 15

א.

MOV BL,[SI]

העבר אל רגיסטר BL נתון בגודל בית מהזיכרון מהכתובת עליה מצביע רגיסטר SI.

ROT : ROL BL,1

סובב את סיביות הנתון שברגיסטר BL פעם אחת שמאלה. הסיבוב מתבצע כך שכל סיבית עוברת סיבית אחת שמאלה. סיבית ה MSB עוברת אל ה LSB וגם אל דגל הנשא.

JNZ ROT

אם בדגל האפס אין '1' אז קפוץ אל התווית ROT. בפקודה זו נבדק דגל האפס שמציין האם בפעולה האריתמטית או הלוגית האחרונה התוצאה הייתה אפס. אם כן הדגל מקבל 1. בפקודה כאן בודקים שאין '1' כלומר שהתוצאה לא הייתה אפס.

MOV [SI+1], AL

העבר את הנתון שב AL אל כתובת בזיכרון עליה מצביע הסכום של תוכן SI ועוד 1.

ב.

התוכנית מעבירה נתון מכתובת 10H אל רגיסטר BL. היא מסובבת אותו שמאלה 8 פעמים כאשר בכל פעולת סיבוב היא בודקת האם בדגל הנשא (המקבל את סיבית ה MSB) יש '1'. אם כן היא מגדילה את

המונה AL ב 1 . אחרי 8 סיבובים המונה AL יציין כמה סיביות של '1' יש בנתון שבכתובת 10H . מספר זה התוכנית מעבירה לכתובת 11H.

ג.

כתובת 10H איננה משתנה בתוכנית. כתובת 11H מקבלת את המספר 4 כי בנתון 35H → 00110101 יש 4 פעמים '1'.

## שאלה 16

כתוב תת־שגרה בשפת ASM-86 , שתציב בתאי הזיכרון שכתובותיהם 16 H ÷ 10 H את הערכים 0 H ÷ 60 H בהתאמה, כלומר: בתא שכתובתו 10 H יוצב הערך 0 H , בתא שכתובתו 11 H יוצב הערך 10 H , וכך הלאה, עד התא שכתובתו 16 H , שבו יוצב הערך 60 H .

## פתרון שאלה 16

PROC1:

MOV SI,10H	; מצביע הכתובות
MOV AL,0	; הערך שייטען לזיכרון
MOV CX,7	; כמות הבתים שיש להעביר אליהם נתונים
AG: MOV [SI],AL	; העבר הנתון לזיכרון
ADD AL,10H	; הגדל את הערך ב 10 הקסה
INC SI	; קדם את מצביע הכתובות
LOOP AG	; בצע את הלולאה 7 פעמים
RET	

## שאלה 17

מערך, שגודלו עשרה תאים, מכיל את המספרים הבאים: 3, 6, 18, 12, 15, 30, 60, 70, 11, 10 . המספרים הללו מייצגים ערכים של עשרה נגדים (ביחידות Ω), המחוברים בטור.

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

1. תגדיר את המערך ותציב את עשרת המספרים לתוכו.
2. תכניס למשתנה RT את ההתנגדות השקולה של עשרת הנגדים.
3. תדפיס את ערכי עשרת הנגדים ואת ערך ההתנגדות השקולה.

## פתרון שאלה 17

א.

```
int array1[10] = {3,6,18,12,15,30,60,70,11,10};
```

ב.

```
int x,RT;
for(x=0, RT=0;x<10;x++)
    RT = RT + array1[x];
```

.ג

```
for(x=0;x<10;x++)
    printf("\nThe Value of R%d = %d ",x,array1[x] );
printf("\nThe value of RT = %d",RT);
```

### שאלה 18

מערך, שגודלו עשרה תאים, מכיל את המספרים הבאים: -2, 3, 0, 9, -5, 4, 6, -8, -9, 0.  
כתוב תכנית (בשפת C או בשפת VB), שתבצע את הפעולות האלה:

1. תגדיר את המערך ותציב את עשרת המספרים לתוכו.
2. תמנה את מספר איברי המערך החיוביים.
3. תמנה את מספר איברי המערך השליליים.
4. תדפיס את תוצאות המנייה שהתקבלו בפעולות 2 ו-3.

### פתרון שאלה 18

```
int array[10] = {-2,3,0,9,-5,4,6,-8,-9,0};
int x,NumberOfPosit,NumberOfNegat;
for(x=NumberOfPosit=0;x<10;x++)
    if (array[x]>=0)
        NumberOfPosit++;
NumberOfNegat=10-NumberOfPosit;
printf("\nThe number of positive numbers is %d ",NumberOfPosit);
printf("\nThe number of negative numbers is %d ",NumberOfNegat);
```