

פתרון מבחן מיקרו 2017**שאלה 5**

להלן שתי תכניות בשפת C של המיקרו-בקר 8051, המפיקות גל מרובע בהדק P_{2,0} שלו. תדר הגביש של המיקרו-בקר הוא 12 MHz.

תכנית 2

```

1. #include <8051.h>
2. unsigned int d;
3. sbit P2_0 = 0xA0;
4. void main (void)
5. {
6.     while (1)
7.     {
8.         P2_0=0;
9.         for(d=0;d<400;d++);
10.        P2_0=1;
11.        for(d=0;d<400;d++);
12.    }
13. }
```

תכנית 1

```

1. #include <8051.h>
2. sbit P2_0 = 0xA0;
3. void main (void)
4. {
5.     TMOD=0x01;
6.     TH0=0xF0;
7.     TL0=0x00;
8.     TR0=1;
9.     P2_0=0;
10.    while (1)
11.    {
12.        TF0=0;
13.        P2_0=!P2_0;
14.        TH0=0xF0;
15.        TL0=0x00;
16.        while (TF0==0);
17.    }
18. }
```

- א. הסבר את ההוראות שבשורות 2, 10, 13 ו-16 בתכנית 1.
- ב. מעוניינים להקטין פי 2 את זמן-המחזור של האות המופק בהדק P_{2,0}. רשום את הנדרש כדי לבצע זאת בתכנית 2.
- ג. יש צורך להפיק אות, מדויק ככל שניתן, שתדרו יהיה 1 kHz ומחזור הפעולה שלו יהיה 50%.
1. קבע באיזו משתי התכניות נכון יותר להשתמש כדי להפיק תדר מדויק. נמק את תשובתך.
2. שנה את קוד התכנית שבחרת כדי להפיק אות בתדר הזה.

תשובה 5

א.

```
sbit P2_0 = 0xA0;
```

הגדרה של ביט באזור ה SFR ששמו P2_0 וכתובתו A0 בהקסה דצימאלי

```
while ( 1)
```

לולאת while הבודקת האם מה שנמצא בתוך הסוגריים שמימין למילה while הוא TRUE. היות ו הערך 1 הוא TRUE מתבצעת הלולאה – השורות 11 עד 17 – אין סוף פעמים.

P2_0 = !P2_0 ;

העבר להדק P2_0 את המצב ההפוך למצב שיש בהדק. במילים אחרות הפוך את מצב הדק P2_0 .
אם בהדק היה 0 יהיה עכשיו 1 ולהפך .

while(TF0 == 0) ;

לולאת while הבודקת האם הביט TF0 ברגיסטר ה TCON שווה 0 . אם כן חוזרים לבדוק את מצב הביט . הלולאה מסתיימת ועוברים לשורה הבאה כאשר הביט עולה ל 1 . זה קורה כשטיימר 1 מסיים לספור ומעלה את הביט ל 1 . (הוא נותן בקשת פסיקה של טיימר 0) . בתרגיל כאן לא עובדים עם פסיקה אלא עובדים עם שאילתה - polling - על הביט .

ב.

בתכנית 2 קובעת כל אחת משתי לולאות ה for את הזמן שבו הדק P2_0 יהיה במצב 0 והזמן שהוא יהיה ב 1 . כדי להקטין את זמן המחזור פי 2 יש להקטין בכל אחת מהלולאות את הזמן של המשתנה d פי 2 . כל לולאת for תיראה כך :

for (d=0; d < 200 ; d++);

ג.

1. יש להשתמש בתכנית 1 כי עובדים עם טיימר שידועים כמה זמן הוא סופר במדויק. בתכנית 2 מבצעים השהייה עם משתנה d וזמן ההשהיה תלוי בקומפיילר שמתרגם את התכנית. כל קומפיילר יכול לתרגם את לולאת ה for בצורה שונה ומכאן שנקבל זמנים שונים כתלות בקומפיילר.

2.

לגל מרובע סימטרי בתדר 1 קילו הרץ יש זמן מחזור של 1 מילי שנייה או 1000 מיקרו שניות. מכאן שיש לדאוג שההדק P2_0 יהיה לזמן של 500 מיקרו שניות במצב 0 ועוד 500 מיקרו לזמן של 1 .

היות ותדר הגביש הוא 12 מגה הרץ ובכניסה למונה יש חלוקה ב 12 , אז הדפקים המגיעים לטיימר לספירה הם בתדר של 1 מגה הרץ. לתדר זה יש זמן מחזור של 1 מיקרו שניות. כלומר טיימר 0 צריך לספור 500 דפקים . הטיימר צריך לספור מהמספר 65036 לפי החישוב : $65036 - 500 = 65536$.

המספר 65036 הוא בהקסה : FE0C . את החלק הגבוה של הטיימר TH0 נטען ל FEh ואת החלק הנמוך TL0 נטען ל 0Ch . יש לשנות בהתאמה את השורות 6,7 ו 14 , 15 בהתאמה.

שאלה 6

פלינדרום הוא מספר, מילה, משפט או כל רצף סמלים אחר, שניתן לקרוא משני הכיוונים, מימין לשמאל או משמאל לימין, ולקבל תוצאה זהה. למשל:

מספר פלינדרומי - 10100101, 01100110, 10000001

מילה פלינדרומית - שמש, תות, מילים

משפט פלינדרומי - רבע הלילה עבר

כתוב תת-סגרה בשפת C של המיקרו-בקר 8051, שתבצע את הפעולות שלהלן:

1. תבדוק אם תוכנו של התא שכתובתו 40H בזיכרון הפנימי של המיקרו-בקר 8051 הוא פלינדרום.
2. אם תוכנו של התא שכתובתו 40H הוא פלינדרום - היא תשאיר את תוכן התא ללא שינוי. אם לא - היא תאפס את תוכן התא.

תשובה 6

נעביר את הנתון שבכתובת 40h אל האקומולטור ונשווה את הביטים המתאימים. נשתמש בתו ^ בכתיבת התוכנה עם MicroVision של קייל. התכנית נראית:

```
#include <AT89X51.h>

unsigned char palindrom _at_ 0x40 ;

void main()

{

    ACC=palindrom;

    if (!(ACC^0==ACC^7 && ACC^1==ACC^6 && ACC^2==ACC^5 && ACC^3==ACC^4 ))

        palindrom=0;

}
```

בכתיבת התכנה עם קומפיילר SDCC יש לרשום במקום התו ^ את התו נקודה. לדוגמה: במקום ACC^0 נרשום ACC.0 וכו'.

שאלה 7

להלן תת-שגרה הכתובה בשפת-הסף של המיקרו-בקר 8051 :

```

1.  START : MOV A , 40H
2.          MOV B, #100
3.          DIV AB
4.          MOV 52H , A
5.          MOV A, #10
6.          MOV R5,A
7.          MOV A,B
8.          MOV B,R5
9.          DIV AB
10.         MOV 51H , A
11.         MOV A,B
12.         MOV 50H , A
13.         RET

```

א. הסבר את ההוראות שבשורות 3, 5, 7 ו-10.

ב. להלן טבלת-מעקב אחר תוכני האוגרים A ו-B ותוכני תאי-הזיכרון שכתובותיהם 51H, 52H ו-50H לאחר ביצוע כל אחת מן ההוראות 12 ÷ 1 בתת-השגרה.

נתון שתוכן התא שכתובתו 40H לפני ביצוע התת-שגרה הוא $(129)_{10}$.

תוכן האוגר A	תוכן האוגר B	תוכן התא שכתובתו 52H	תוכן התא שכתובתו 51H	תוכן התא שכתובתו 50H	השורה בתת-השגרה
					1
					2
					.
					.
					.
					12

העתק את הטבלה למחברתך, והשלם בה את תכני האוגרים ותאי-הזיכרון לאחר ביצוע כל אחת מן ההוראות 12 ÷ 1.

ג. הסבר מה מבצעת התת-שגרה.

תשובה 7

א.

DIV AB

חלק את הנתון שבאקומולטור עם הנתון שברגיסטר B . התוצאה תהיה באקומולטור והשארית ברגיסטר B. לדוגמה : A=129 ו B=100 . החלוקה תיתן ב A את המספר 1 ו 29 ב B .

MOV A,#10

העבר אל האקומולטור את הנתון 10 (עשרוני ולא הקסה).

MOV A,B

העבר את הנתון שברגיסטר B אל האקומולטור.

MOV 51H , A

העבר את הנתון שבאקומולטור אל כתובת 51H בזיכרון RAM הנתונים הפנימי.

ב. בתרגיל כאן כדאי לעבוד עם מספרים עשרוניים. (הוספנו עמודה עבור R5 להקל על ההבנה).

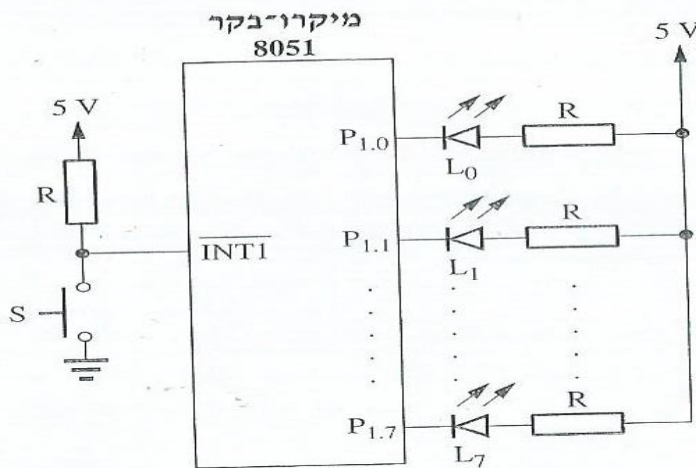
R5	A	B	52H	51H	50H	השורה
	129					1
		100				2
	1	29				3
			1			4
	10					5
10						6
	29					7
		10				8
	2	9				9
				2		10
	9					11
					9	12

ג.

התכנית לוקחת נתון מכתובת 40H בזיכרון הנתונים הפנימי ומפרידה אותו ל 3 הספרות עשרוניות שלו (ספרת מאות עשרות ויחידות) ושומרת את הספרות בתאים 52H (את ספרת המאות) 51H (את ספרת העשרות) ו 50H (ספרת האחדות) .

שאלה 8

באיור לשאלה 8 נתון מיקרו־בקר 8051, שלהדק הפסיקה $\overline{INT1}$ שלו חובר לחצן S. נוריות ה־LED, $L_0 \div L_7$, חוברו בהתאמה להדקים $P_{1.0} \div P_{1.7}$ של המיקרו־בקר.



איור לשאלה 8

להלן המבנה הכללי של תכנית בשפת C של המיקרו־בקר 8051, הכוללת פונקציה ראשית ופונקציית־פסיקה:

```
#include <8051.h>
void main (void)
{
    ...
}
void int1() interrupt 2
{
    ...
}
```

כתוב פונקציה ראשית main, שתאפשר פסיקה במבוא $\overline{INT1}$ בירידת השעון של המיקרו־בקר 8051, תדליק את הנורית L_0 בלבד, ותמתין לפסיקות נוספות. כתוב פונקציית־פסיקה, שתדליק בכל פעם שהלחצן S נסגר נורית LED נוספת לאלו שכבר דולקות, החל מ־ L_1 ועד L_7 . הפסיקה תיחסם כאשר כל נוריות ה־LED, $L_0 \div L_7$, דולקות.

א

ב

תשובה 8**.א.**

```
#include <AT89X51.h>

void main()
{
    P1=0xFE; // led in P1.0 is ON ---- P1.0 הדלקת הLED ב
    IE=0x84; // Enable int1 1 אפשר פסיקה חיצונית
    IT1=1; // interrupt 1 works on EDGE ( not LEVEL) - 0 ל 1 עבודה על שפה – ירידה מ
    while(1); // לולאה החוזרת על עצמה עד שיש פסיקה ובסיום הפסיקה חוזרים ללולאה.
}

```

.ב.

נרשום פונקציית השהייה כדי להתגבר על ניתור המגעות בלחיצה על המפסק (debounce) :

```
void delay ()
{
    for ( int i=0; i<10000;i++); // לולאה המתבצעת 10000 פעמים
}

```

תכנית הפסיקה :

```
void int1( ) interrupt 2
{
    P1=P1<<1; // mov P1 to the left
    delay(); // debounce פונקציית השהייה להתגבר על ניתור מגעות
    if(P1==0) // האם כל הLEDים דולקות ?
        IE=0; // Disable all interrupts חסום את כל הפסיקות
}

```