

1. Napisati program za programiranje uC 8051 koji treba da generise povorku impulsa na pinu P1.0. Program napisati koristeći Tajmer 0 koji radi u modu 0. Perioda signala treba da bude 2ms tako da impuls traje 1ms, pauza 1ms.

Rad:

TL0 u Modu0 radi kao preskaler - tj broji 32 impulsa, od 0 do 31 (0 do 1F), I pri tranziciji od 31 na 0 inkrementira TH0 za 1. Na osnovu ovoga treba sračunati kako da se inicijalizuju tajmerski registri TL0 I TH0.

$1000 / 32 = 31$ I ostatak 8,

Dakle TH0 treba inicijalizovati tako da nakon 31 inkrementa dodje do vrednosti 0xFF a TL0 tako da nakon 8 inkrementa dodje do vrednosti 31 (0x1F). Tada ce nakon sto TL0 odbroji 8 impulsa na zadnji izvršiti tranziciju sa 31 na 0 I inkrementirati TH0 koji ce onda precizirati iz 0xFF u 0x00 I postaviti overflow flag TF0 koji setuje tajmerski interrupt.

$255 - 31 = 224$ (0xE0) - vrednost za TH0

$31 - 8 = 23$ (0x17) - vrednost za TL0

```
#include <reg51.h> // Include 8051 header file
```

```
sbit Test = 0x90; // P1.0
```

```
// funkcija za generisanje 1ms kasnjenja koriscenjem Timer 0 u Mode 0
```

```
void Delay()
{
    TMOD &= 0x00; // Timer 0 u Mod 0 (16-bit timer)
    TL0 = 0x17; // inicijalizacija TL0
    TH0 = 0xE0; // inicijalizacija TH0
    TF0 = 0; // ponisti se Timer 0 overflow flag
    TR0 = 1; // Start Timer 0

    // Wait until Timer 0 overflows
    while (!TF0)
    {
    }
    TR0 = 0; // Stop Timer 0
    TF0 = 0; // ponisti se Timer 0 overflow flag
}
```

```
void main()
```

```
{
    // konfigurisi P1 port kao izlazni
    P1 = 0x00; // Initialize P1
    Test = 0; // inicijalno postavljanje P1.0 na 0

    EA = 1; // globalna dozvola za interapt-e
    ET0 = 1; // dozvola za tajmerski interapt 0 (kad se desi TF0)
    while (1)
    {
        // Generisanje impulsa (1ms ON)
        Test = 1;
        Delay(); // cekanje 1ms

        // Pauza (1ms OFF)
        Test = 0;
        Delay(); // cekanje 1ms
    }
}
```

```
}
```

2. Napisati program za programiranje uC 8051 koji treba da generise povorku impulsa na pinu P1.0. Program napisati koristeći Tajmer 0 koji radi u modu 1. Perioda signala treba da bude 2ms tako da impuls traje 1ms, pauza 1ms.

Rad:

TL0 u Modu0 radi kao 16 bit - ni tajmer - broji od 0 do $2^{16} - 1 = 65535$ (0x0000 - 0xFFFF). Na osnovu ovoga treba sračunati kako da se inicijalizuju tajmerski registri TL0 i TH0.

$65536 - 1000 = 64536$ (0xFC17)

0xFC - vrednost za TH0

0x17 - vrednost za TL0

```
#include <reg51.h> // Include 8051 header file
```

```
sbit Test = 0x90; // P1.0
```

```
// funkcija za generisanje 1ms kasnjenja koriscenjem Timer 0 u Mode 0
```

```
void Delay()
```

```
{
    TMOD &= 0x00; // Timer 0 u Mod 0 (16-bit timer)
    TL0 = 0x17; // inicijalizacija TL0
    TH0 = 0xFC; // inicijalizacija TL1
    TF0 = 0; // ponisti se Timer 0 overflow flag
    TR0 = 1; // Start Timer 0

    // cekaj dok Timer 0 ne bude overflow
    while (!TF0)
    {
    }
    TR0 = 0; // Stop Timer 0
    TF0 = 0; // ponisti se Timer 0 overflow flag
}
```

```
void main()
```

```
{
    // konfigurisi P1 port kao izlazni
    P1 = 0x00; // Initialize P1
    Test = 0; // inicijalno postavljanje P1.0 na 0
    EA = 1; // globalna dozvola za interapt-e
    ET0 = 1; // dozvola za tajmerski interapt 0 (kad se desi TF0)
    while (1)
    {
        // Generisanje impulsa (1ms ON)
        Test = 1;
        Delay(); // cekanje 1ms

        // Pauza (1ms OFF)
        Test = 0;
        Delay(); // cekanje 1ms
    }
}
```