

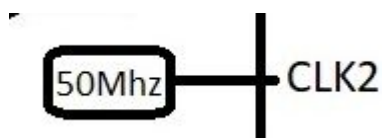
Zadanie c. 6



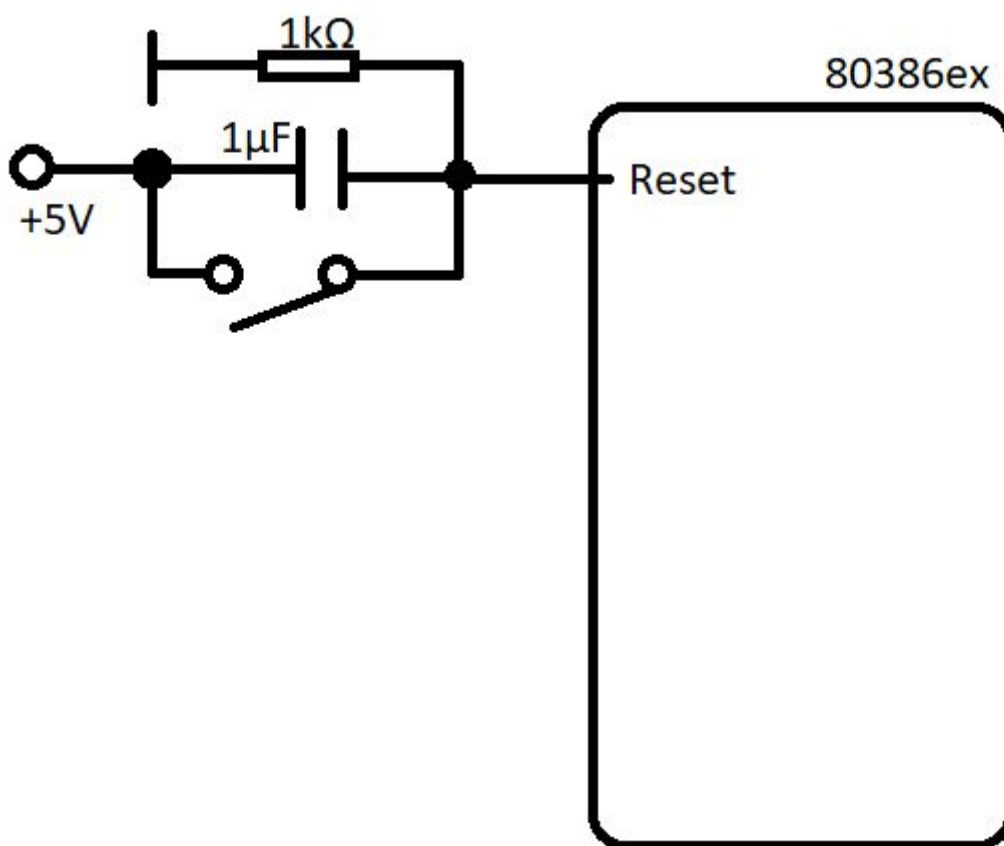
Úloha č. 1: Mikropočítač na báze mikroprocesora 80386EX

Navrhnete schému mikropočítača na báze mikroprocesora 80386EX v reálnom režime s nasledovnými parametrami:

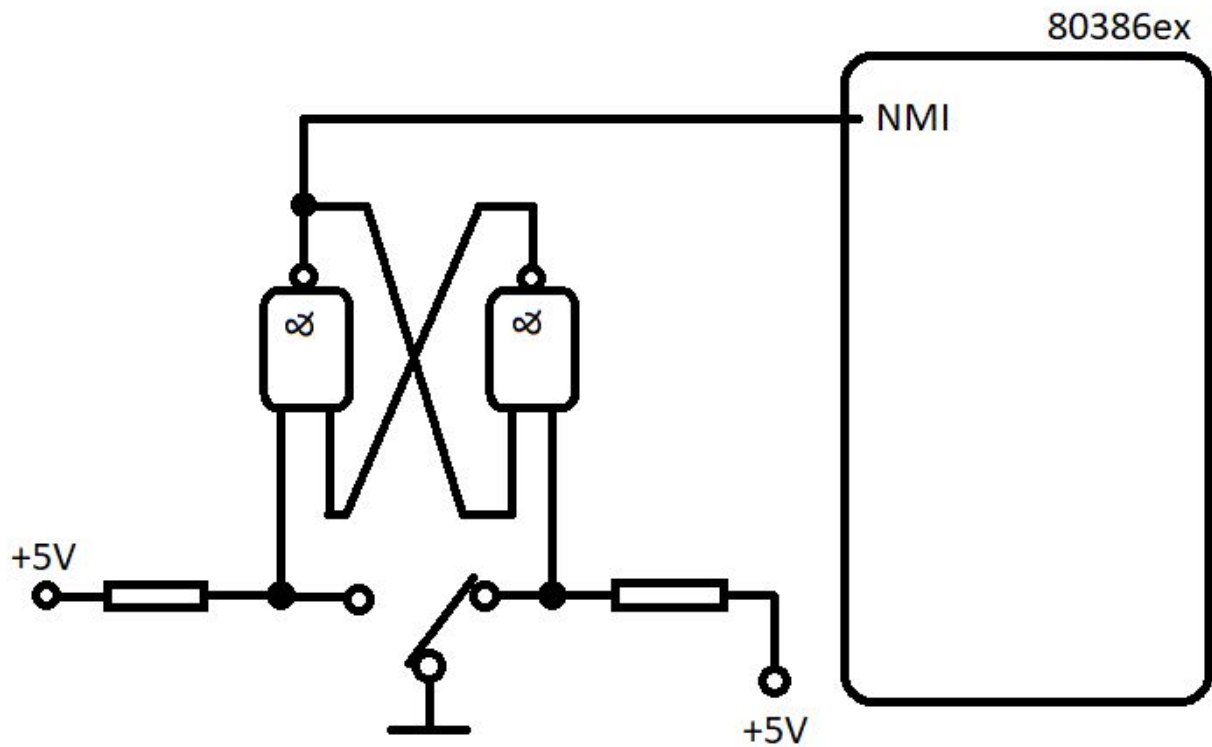
- frekvencia procesora 25 MHz. Nájdite vhodný integrovaný oscilátor.



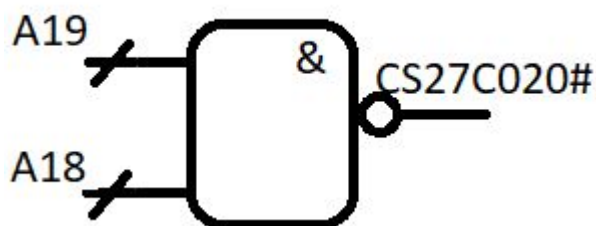
- signál RESET bude generovaný automaticky po pripojení napájacieho napätia alebo tlačidlom. Parametre RC člena: $R = 1\text{ k}\Omega$, $C = 10\text{ }\mu\text{F}$.



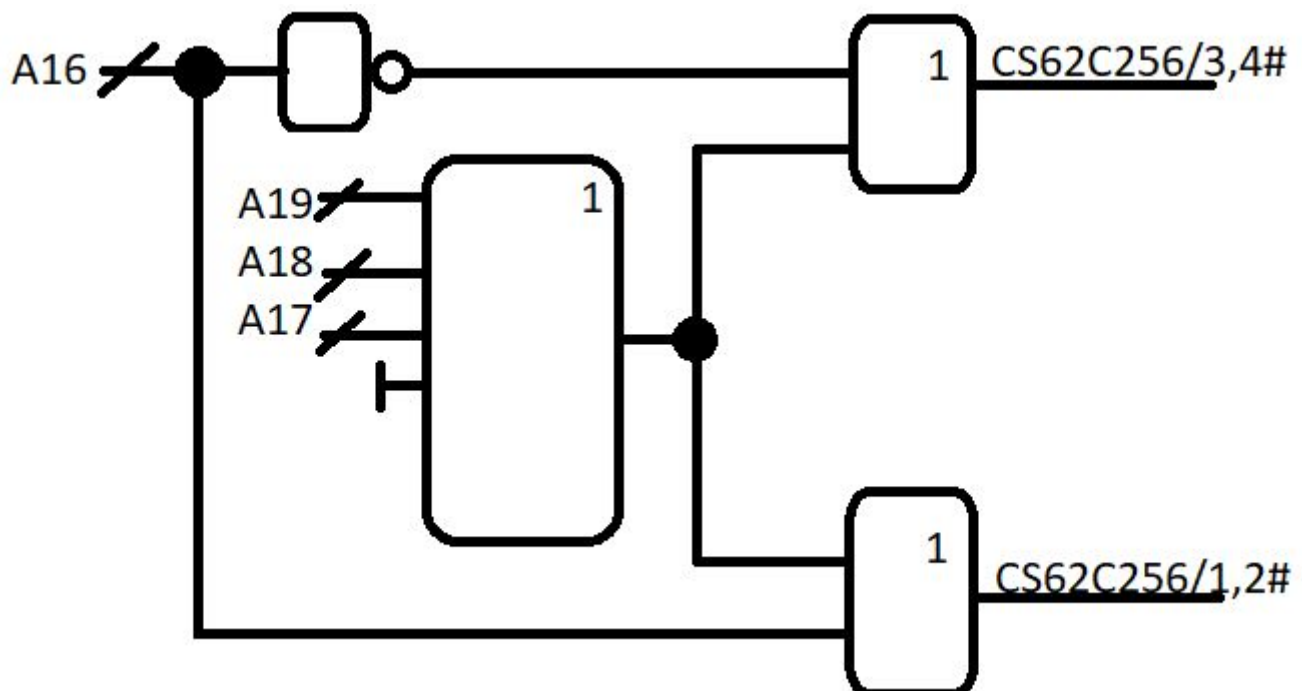
- signál NMI bude generovaný tlačidlom, ošetreným proti zákmitom



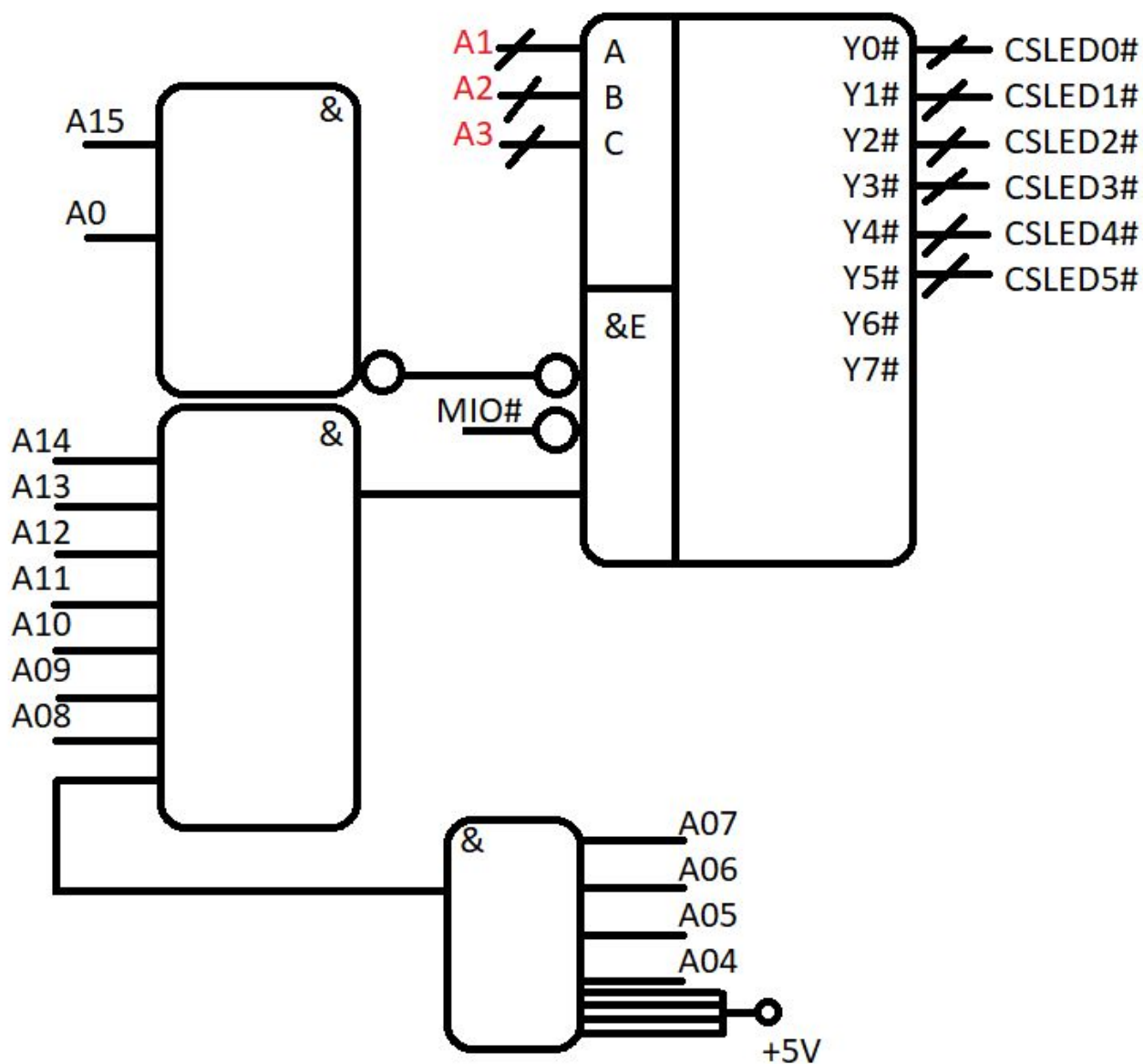
- 8-bitová bootovacia pamäť EPROM 256 kB (1x 27C020). Navrhňte externý úplný dekódér.



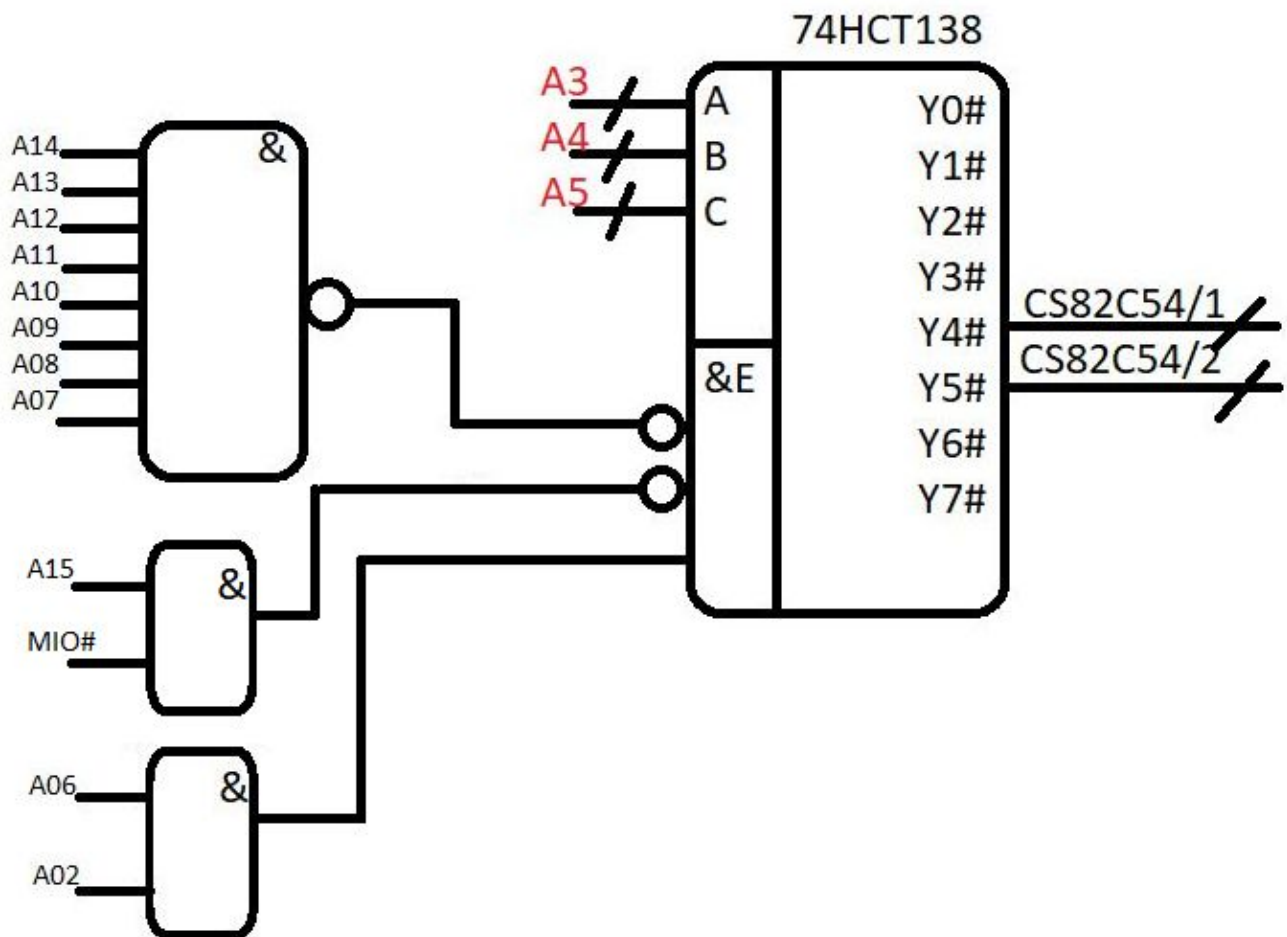
- 16-bitová pamäť RWM 128 kB (4x 62C256), umiestnená od fyzickej adresy 00000H. Navrhňte externý úplný dekódér.



- 6-miestny statický LED displej. Prúd cez jednotlivé segmenty každej číslice bude 8 mA. Jednotlivé číslice budú ovládané obvodmi 74HCT573. Adresy jednotlivých číslic: 7FF0H, 7FF2H, 7FF4H, 7FF6H, 7FF8H, 7FFAH. Navrhnete externý úplný dekódér.

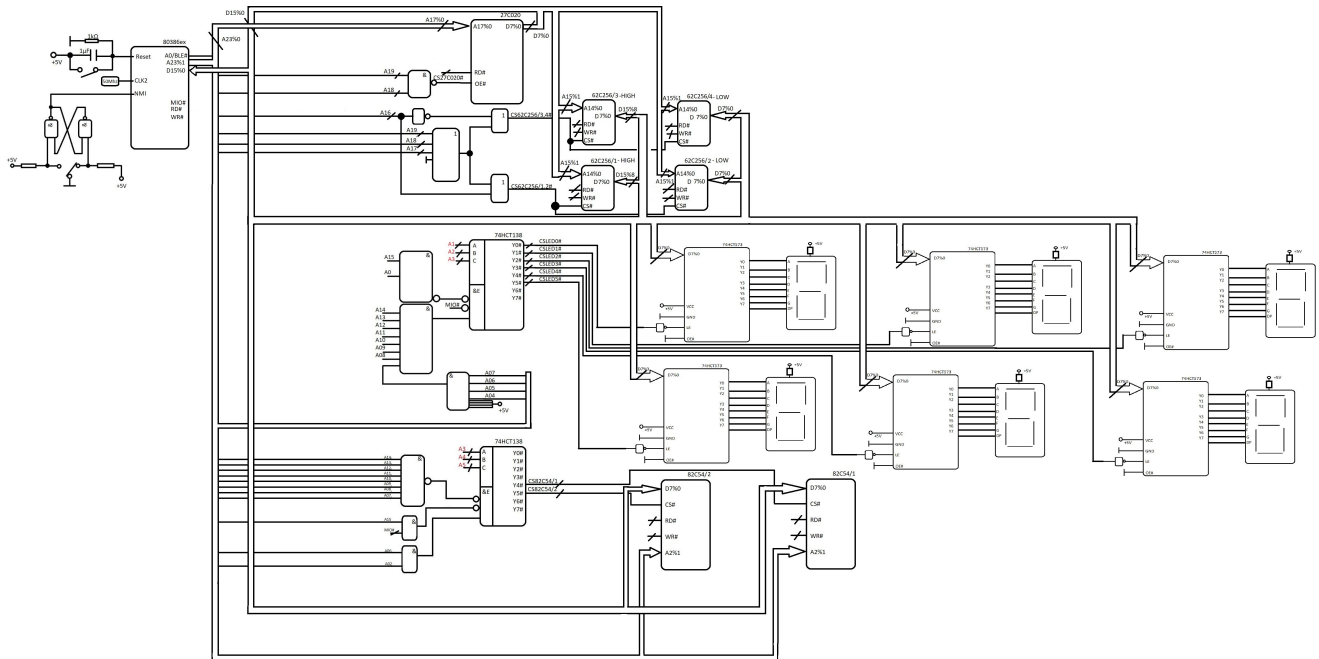


- 2 externé obvody 82C54. Adresy prvého obvodu: 7FE1H, 7FE3H, 7FE5H, 7FE7H. Adresy druhého obvodu: 7FE9H, 7FEBH, 7FEDH, 7FEFH.



Nizšie je uvedená tabuľka, ktorá je základom vytvorenia jednotlivých dekoderov pre posledné 4 body prvej ulohy, spolu so súhrnným diagramom celého obvodu.

25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	SAB
0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	27C020
0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	<00C0000H; 00FFFFFFH>
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	X 62C256/1 - HIGH <0000000H; 000FFFFFFH>
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	X 62C256/2 - LOW <0000000H; 000FFFFFFH>
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	X 62C256/3 - HIGH <0010000H; 001FFFFFFH>
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	X 62C256/4 - LOW <0010000H; 001FFFFFFH>
										15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	LED
										0	1	1	1	1	1	1	1	1	1	1	1	X	X	X	0	Rozsah - parne
										0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	7FF0H
										0	1	1	1	1	1	1	1	1	1	1	1	0	0	1	0	7FF2H
										0	1	1	1	1	1	1	1	1	1	1	1	0	1	0	0	7FF4H
										0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	0	7FF6H
										0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	7FF8H
										0	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	7FFAH
										15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	82C54/1
										0	1	1	1	1	1	1	1	1	1	1	0	0	X	X	1	Rozsah - neparne
										0	1	1	1	1	1	1	1	1	1	1	0	0	0	0	1	7FE1H
										0	1	1	1	1	1	1	1	1	1	1	0	0	0	1	1	7FE3H
										0	1	1	1	1	1	1	1	1	1	1	0	0	1	0	1	7FE5H
										0	1	1	1	1	1	1	1	1	1	1	0	0	1	1	1	7FE7H
										15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	82C54/2
										0	1	1	1	1	1	1	1	1	1	1	0	1	X	X	1	Rozsah - neparne
										0	1	1	1	1	1	1	1	1	1	1	0	1	0	0	1	7FE9H
										0	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	7FEBH
										0	1	1	1	1	1	1	1	1	1	1	0	1	1	0	1	7FEDH
										0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	7FEFH



Úloha č. 2: Inicializačný program

Napište inicializačný program v jazyku assembler 86, ktorý sa bude skladať z dvoch segmentov. Prvý segment bude umiestnený od fyzickej štartovacej adresy a bude obsahovať iba ďaleký skok na začiatok druhého

segmentu, ktorý bude umiestnený na začiatku pamäte EPROM. Určíte fyzickú adresu druhého segmentu v tvare SEGMENT:0000H.

```
; Fyzicka adresa druhého segmetu: 000D:0000
```

```
.code
```

```
    jmp 0x0000:0x000C
```

```
.code
```

```
    mov bx, 0x0
```

```
    mov cx, 0xff
```

```
loop:
```

```
    cmp [ebp + bx], cx
```

```
    jnz wrong
```

```
    dec cx, 1
```

```
    add bx, 1
```

```
    cmp bx, 0xff
```

```
    jb  loop
```

```
    mov dx, 0x0000
```

```
    exit 1
```

```
wrong:
```

```
    mov dx, 0xffff
```

```
    mov ds, 0x0000
```

```
    mov si, bx
```