



МИСиС

Лекции

Микроконтроллеры

Обратная задача

2016

Обратная задача:

- 1) Читаем и изучаем код
- 2) Дописываем комментарии
- 3) Составляем блок схему
- 4) Составляем схему включения.

Разбираем код

```
#include <pic.h>
```

```
__CONFIG(0x184);
```

```
unsigned char mas8[10] = {
```

```
    0b00001000, //0
```

```
    0b01101011, //1
```

```
    0b00010010, //2
```

```
    0b01000010, //3
```

```
    0b01100001, //4
```

```
    0b01000100, //5
```

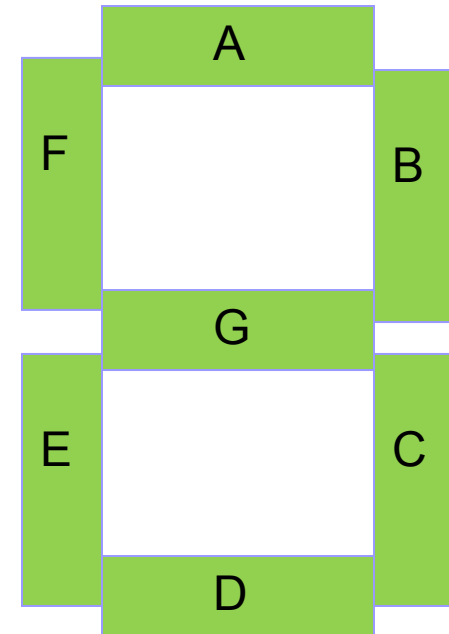
```
    0b00000100, //6
```

```
    0b01101010, //7
```

```
    0b00000000, //8
```

```
    0b01000000 }; //массив кодов семисегментника
```

```
unsigned char counter = 0; //
```



Разбираем код

```
void Delay(int count)
```

```
{  
    int i;  
    for(i = 0; i < count; i++)  
    {  
        i++;  
        i--;  
    }  
}
```

```
void Display(void)
```

```
{  
    unsigned char digit;  
    unsigned char code;  
    digit = counter % 10;  
    code = mas8[digit];  
    RC4 = 1; //ст.цифра  
    RC5 = 0; //мл.цифра  
    PORTA = ((PORTA & 0b11111000)|(code & 0b00000111));  
    PORTC = ((PORTC & 0b11110000)|(code>>3));  
}
```

Разбираем кодс

```
Delay(100);
```

```
    digit = (counter/10) % 10;
```

```
    code = mas8[digit];
```

```
    RC4 = 0;//ст.цифра
```

```
    RC5 = 1;//мл.цифра
```

```
    PORTA = ((PORTA & 0b11111000)|(code & 0b00000111));
```

```
    PORTC = ((PORTC & 0b11110000)|(code>>3));
```

```
    Delay(100);
```

```
}
```

Разбираем кодс

```
void main(void)
{
    char prevstat=0;
    char currstat=0;

    TMR0=0;
    OPTION= 208; //timer 1:2
    INTCON= 0;
    OSCCAL= 128;
    TRISA=56;//1-in, 0-out
    TRISC=0;

    counter = 0;
```

Разбираем код

```
while(1)
{
currstat=(PORTA>>3)&0x07;
if(prevstat == 1 && currstat == 0)
counter = counter+1;
if(prevstat == 1 && currstat == 3)
counter = counter-1;
if(prevstat == 1 && currstat == 5)
counter = 0;

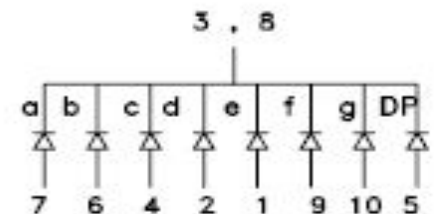
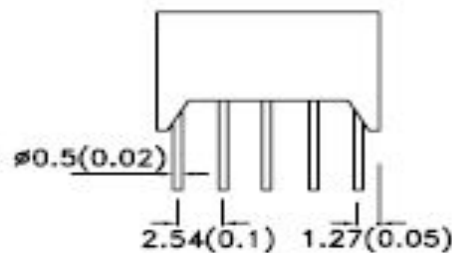
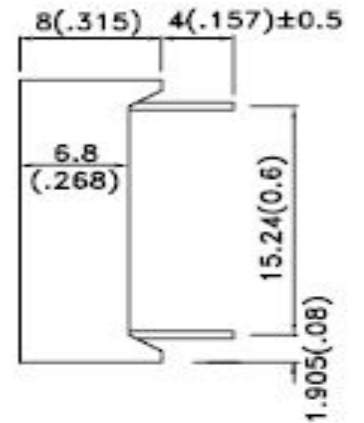
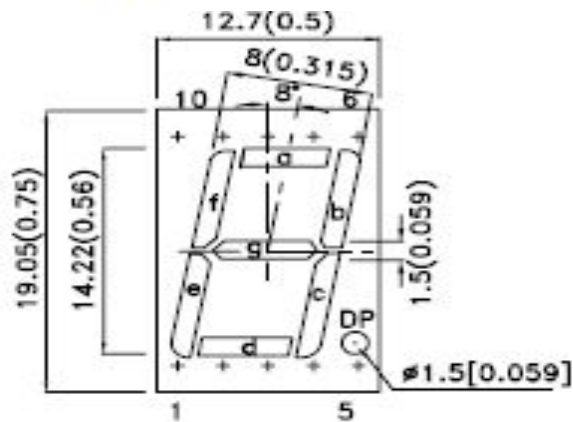
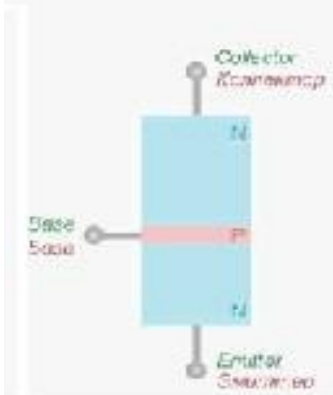
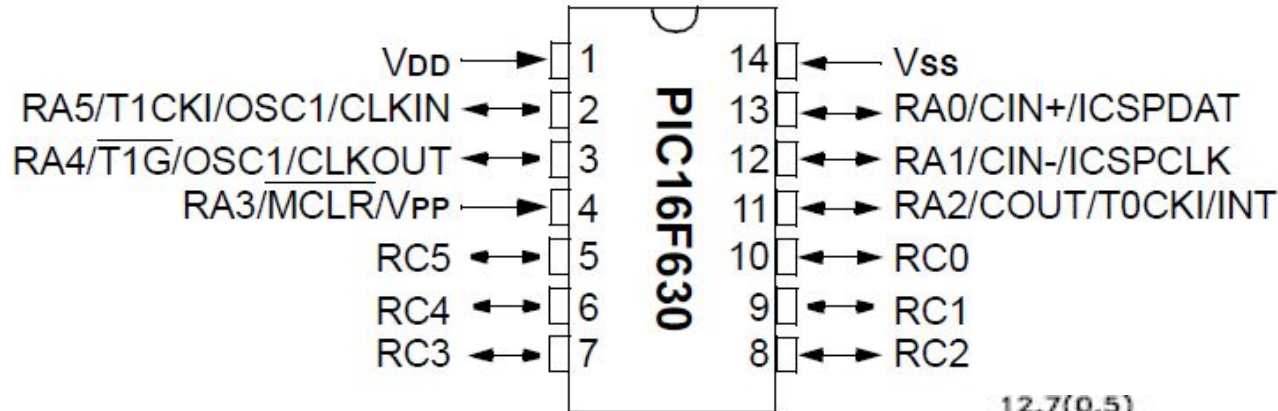
prevstat = currstat;

    if (counter == 100)
        counter = 0;

    if (counter == 0xff)
        counter = 99;

    Display();
}
}
```

PDIP, SOIC



7-сегментник

- `0b00001000, //0`
- `0b01101011, //1`
- `0b00010010, //2`
- `0b01000010, //3`
- `0b01100001, //4`
- `0b01000100, //5`
- `0b00000100, //6`
- `0b01101010, //7`
- `0b00000000, //8`
- `0b01000000 }; //9`
- `0bxEDCGBFA`

