

Embedded System: Programming Practice with Embedded C 21092020

Program No.1: Write an embedded C program to count from 0 to 255

```
#include <reg51.h>
```

```
void my_delay(unsigned int);
```

```
unsigned int i,j,k;
```

```
void main()
```

```
{
```

```
P1=00;
```

```
for (i=0; i<=255; i++)
```

```
{
```

```
P1 ++;
```

```
my_delay(2000);
```

```
}
```

```
}
```

```
void my_delay ( unsigned int my_time)
```

```
{
```

```
for ( j=0 ; j<my_time; j++)
```

```
for ( k=0; k<1275; k++);
```

```
}
```

Program No.2 : Write an embedded C program to take data from p1 and transfer to p2

```
#include <reg51.h>
```

```
void my_delay(unsigned int);
```

```
unsigned int i,j,k;
```

```
void main()
```

```
{
```

```
unsigned char data_byte;
```

```
P1= 0xFF;
```

```
while (1)
```

```
{
```

```
data_byte= P1;
```

```
my_delay(200);
```

```
P2= data_byte;
```

```
}
```

```
}
```

```
void my_delay(unsigned int my_time)
```

```
{
```

```
for (j=0; j<my_time; j++)
```

```
for (k=0; k<1275; k++);
```

```
}
```

Program No.3: Write an embedded C program to toggle p0, p1 and p2
PROGRAM TO TOGGLE P0, P1, P2

```
#include <reg51.h>
void add_delay(unsigned int);
unsigned int i,j,k;
sfr P0 = 0x80;
sfr P1 = 0x90;
sfr P2 = 0xA0;
void main()
{
while(1)
{
P0 = 0xAA;
P1 = 0xAA;
P2 = 0xAA;
add_delay(2);
P0 = 0x55;
P1 = 0x55;
P2 = 0x55;
add_delay(2);
}
}
void add_delay(unsigned int my_time)
{
for (j=0; j<my_time; j++)
for (k=0; k<1275; k++);
}
```

Program No.4 : Write an embedded C program to toggle single bit

```
#include <reg51.h>
void add_delay(unsigned int);
unsigned int i,j,k;
sbit data_bit = P1^2;
void main()
{
while(1)
{
data_bit = 1;
add_delay(2);
data_bit = 0;
}
}
void add_delay(unsigned int my_time)
{
for (j=0; j<my_time; j++)
for (k=0; k<1275; k++);
}
```

Write an 8051 C program to toggle P1.0 bit one time.

Answer :

```
#include <reg51.h>
sbit data1 = P1^0;
void main (void)
{
while(1)
{
    unsigned char i;

    for (i = 0; i <=5; i++) // Delay
        {
            data1=0;
        }

    for (i = 0; i<=5; i++) // Delay
        {
            data1=1;
        }
}
}
```