Anticlockwise

1.
```
#include <stdio.h>
#include <reg51.h>
char xdata    port_at_0xe808;
char xdata    porta_at_0xe800;
char idata acc_at_0x30;
delay ()
{

    int j;
    for (j=0; j<800; j++)
    { }

}

void main ()
{

    port = 0x80;
    while (1)
    {

        acc = 0x11;
        porta = acc;
        delay ();
        acc = 0x22;
        porta = acc;
        delay ();
        acc = 0x44;
        porta = acc;
        delay ();
        acc = 0x88
        porta = acc;
        delay ();

    }

}
```

2. Display FIRE {HELP alternatily with flickering effects on a 7 segment display.

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

Char xdata CommW _at_ 0xe803;
Char xdata portB _at_ 0xe801;
Char x data portC _at_ 0xe802.
Char port [20] = {0x8e, 0xf9, 0xde, 0x86, 0xff, 0xff,
                  0xff, 0xff, 0x89, 0x86, 0xc7, 0x8c };
delay ()
{
    long u;
        for (u=0; u< 8000; u++);
}
void main ()
{
    unit d, b, j, m;
    unsigned char;
    CommW = 0x80;
    do
    {
    i=0;
    for ( d=0; d<8; d++)
    {
        for (b =0; b<4; b++)
        {
            k = por
            t[i++];
            for (j=0; j<8; j++)
            {
                m=k;
                k=k & 0x80;
                {
```
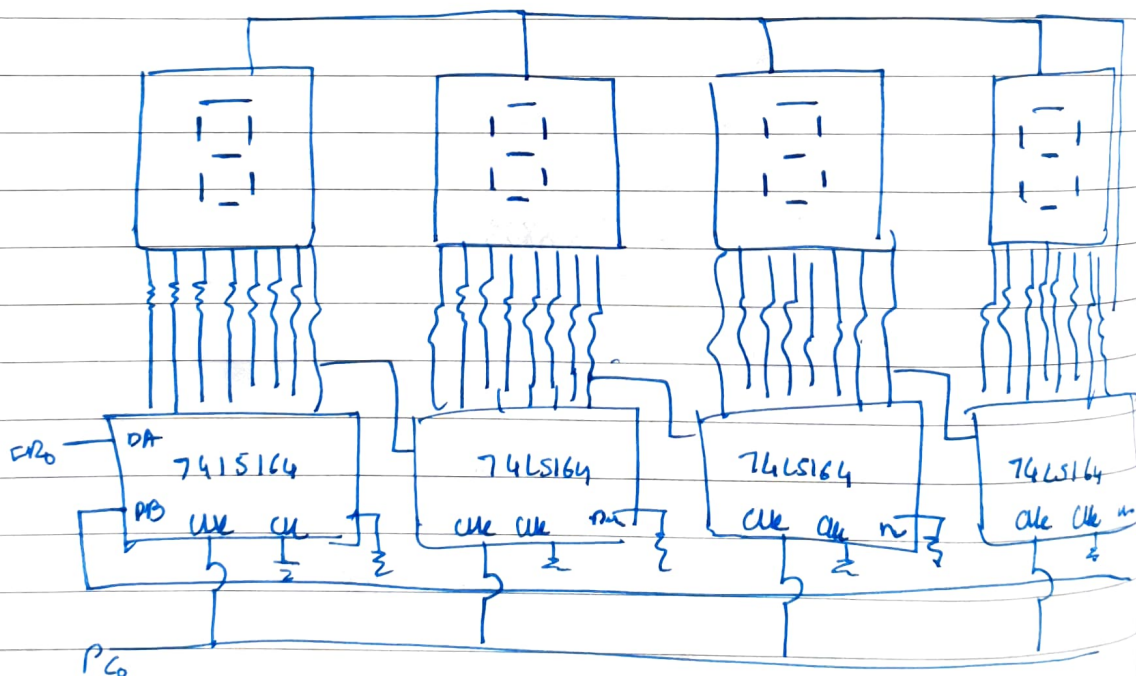
```
        if (k == 00)
                portB = 0x00;
        else
                portB = 0x01;
        }
        portc = 0x01;
        port c = 0x00;
        k = m;
        k <<= 1;
            }
        }
delay ();
    }
}
while (1);
}
```

Display Bangalore:

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

char xdata Commw _at_ 0xe808;
char xdata portB _at_ 0xe801;
char xdata portc _at_ 0xe802;
char port[20] = {0xff, 0xff, 0xff, 0x83, 0x88, 0xc8,
                 0x82, 0x88, 0xc7, 0xc0, 0xaf, 0x86};

void delay()
{
    long v;
    for(v=0; v<8000; v++)
    { }
}

void main()
{
    uint d, b, j, m, i;
    unsigned char k;
    Commw = 0x80;
    do
    {
        i=0;
        for(d=0; d<1; d++)
        {
            for(b=13, i>0; i--)
            {
                delay();
                k= port[i++];
                for(j=0; j<9; j++)
                {
                    m=k;
                    K= k+0x80;
```

```
            if (k==0)
                        portB = 0x80;
            else
                        port B = 0x01;
            portc = 0x01;
            port G = 0x00;
            K=m;
            k<<=1;
                }
        }
        delay();
    }
}while(1);
}
```

4) Clockwise Stepper motor

```
#include <stdio.h>
#include <reg51.h>
Char xdata port _at_ 0xe803;
Char xdata porta _at_ 0xe800;
Char idata acc _at_ 0x30;
void delay()
{
        int j;
        for(j=0; j<800; j++)
            { }
}
void main()
{
        port = 0x80;
        while(1)
            {
```

```
        acc = 0x11;
        porta = acc;
        delay ();
       acc = 0x22;
       porta = acc;
       delay ();
      acc = 0x44;
     porta = acc;
     delay ();
     acc = 0x88;
     porta = acc;
     delay ();
    }
  }
```
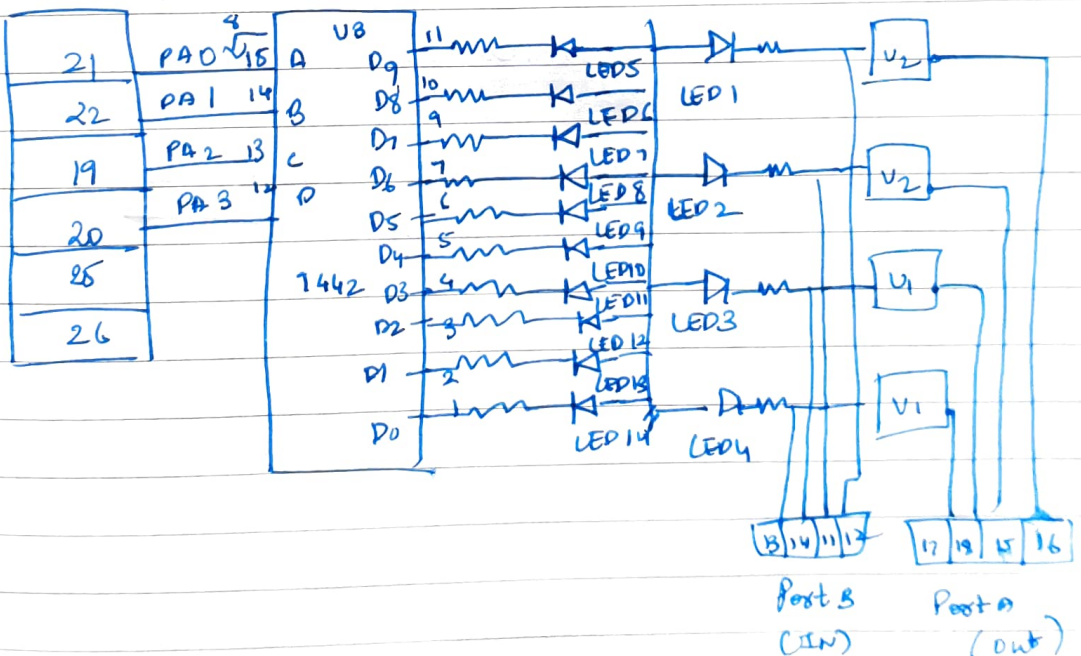
5) Elevator program :


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


unsigned    char xdata   Command Word _at: 0xc803
unsigned    char xdata PortA _at_ 0xc800;
```

```
unsigned char xdata Port B _at_ 0xe801;
unsigned char xdata Present floor, Requested floor,
                         Step = 0xf0;
unsigned long xdata Count, i;
Delay ()
{
    for (Count = 0; Count <= 4500; Count++);
}

Reset ()
{
        Step = Step & 0x0f;
        Port A = Step;
        Step = Step | 0xf0;
        Port A = Step;
}

GoUp()
{
    Switch ( Requested Floor)
    {
                Case 0x0d: while (Step < 0xf3)
                {
                        Step++;
                        Port A = Step;
                        Delay();
                }
                Reset();
                break;
    Case 0x0b: while (step < 0xf6)
                {
                        Step++;
                        Port A = Step;
                        Delay ();
                }
                Reset();
                break;
```

```
Case  0x07 : while (step < 0xfa)
                  {
                       Step++;
                       Port A = step;
                       Delay ();
                  }
                  Reset ();
                  break;

      }
 }

Go Down ()
 {

    switch (Requested Floor)
      {
          Case  0x0d :   while (step > 0xf3)
                            {
                                 Step--;
                                 Port A = step;
                                 Delay ()
                            }
                            Reset ();
                            break;
          Case  0x0b :   while (step > 0xf6)
                            {
                                 Step--;
                                 Port A = step;
                                 Delay ();
                            }
                            Reset ();
                            break;

          Case  0x0c :   while (step > 0xf0)
                            {
                                 Step -- ;
                                 Port A = step;
```

```c
            Delay();
        }
        Reset();
        break;
    }
}

void main()
{
    Command Word = 0x82
    PortA = 0xf0);
    PresentFloor = 0x0e;
    while (1) {
    Requested Floor = PortB;
    Requested Floor = Requested Floor & 0x0f;

    if (Requested Floor != 0x0f && Requested Floor != Present
                                                      Floor){

    if (RequestedFloor < PresentFloor)
            GoUp();
        else
                GoDown();
    Present Floor = Requested Floor;
    }
    RequestedFloor = PortB;
```