Trisha
1BM19CS214

1. Drive a Stepper Motor interface to rotate the motor in Anti-Clockwise by N steps. Introduce suitable delay between successive steps

```
#include <stdio.h>
#include <reg51.h>
char xdata port_at_0xe803;
char xdata porta_at_0xe800;
char idataacc_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();
  }
}
```

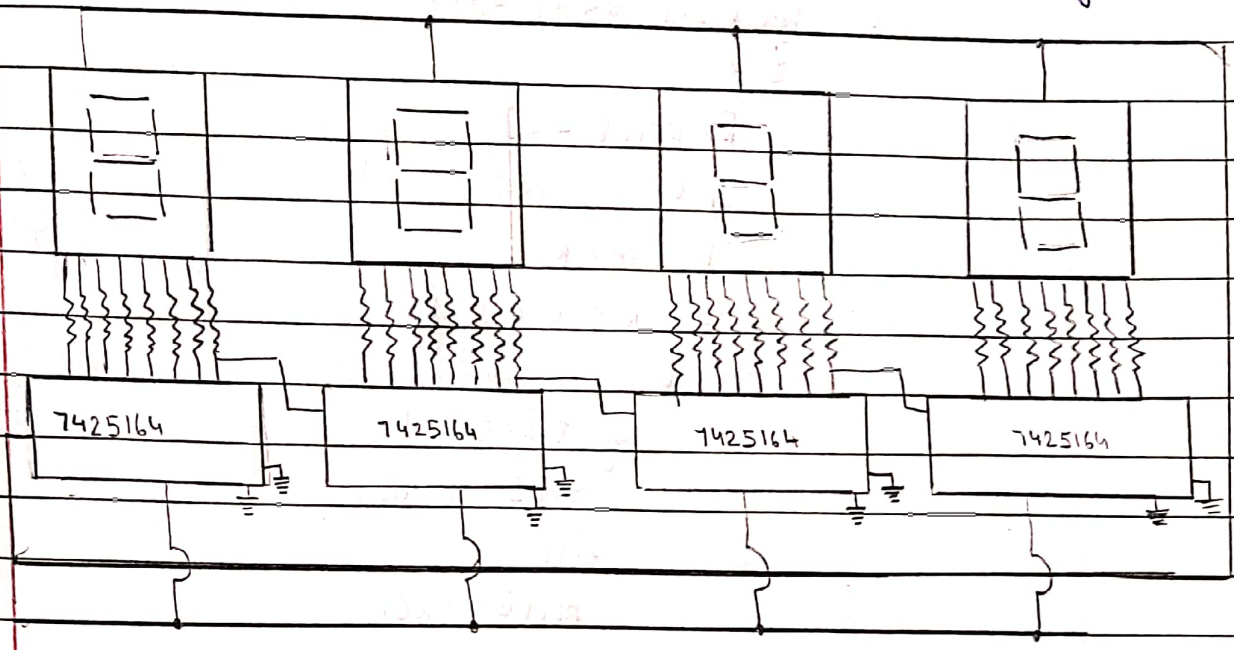
Trisha
IBM19CS214

Drive a Stepper Motor interface to rotate the motor in clockwise by N steps. Introduce suitable delay between successive steps.

```
#include <stdio.h>
#include <reg51.h>
chardata  port_at_ 0xc803;
chardata  porta_at_ 0xe800;
charidataacc_at_ 0x30;
delay()
{   int j;
    for( j=0 ; j<800; j++)
        {}
}
void main()
{   port = 0x80;
    while(1)
    {   acc = 0x88;
        porta = acc;
        delay();
        acc = 0x44;
        porta = acc;
        delay();
        acc = 0x22;
        porta = acc;
        delay();
        acc = 0x11;
        porta = acc;
        delay();
    }
}
```

Trisha
IBM19CS214

Display messages FIRE and HELP alternatively with flickering effects on a 7-segment display interface for a suitable period of time. Ensure a flashing rate that makes it easy to read both the messages.




```
#include <stdio.h>
#include <reg51.h>
charxdataCommW _at - 0xe803;
charxdataportB_at - 0xe801;
charxdataportC _at - 0xe802;
char port[20] = { 0x8e, 0xf9, 0xdc, 0x86, 0xff, 0xff, 0xff, 0xff,
                  0x89, 0x89, 0xc7, 0x8c }, i;

delay()
{   long u;
    for (u=0; u<8000; u++);
}
void main()
{   int d, b, j, m;
    unsigned char k,
    CommW = 0x80;
```

```
do {
    i = 0;
    for ( d = 0; d < 3; d++ )
    {   for (b = 0; b < 4; b++)
        {
          k = port [i++];
        for (j = 0; j < 8; j++)
          {   m = k;
              k = k & 0x80;
            {
              if ( k == 00)
                portB = 0x00;
              else
                portB = 0x01;
            }
        portc = 0x01;
        portc = 0x00;
         k = m
         k <<= 1;
        }
    }
    delay ();
    }
}
while (1);
}
```

Trisha
1BM19CS214

Display message BANGALORE in rolling fashion on a 7-segment display interface for a suitable period of time.

```
#include <stdio.h>
#include <reg51.h>
char xdata CommW at 0x803;
char xdata portB at 0xc801;
char xdata portC at 0xe802;
char port[20] = { 0xff, 0xff, 0xff, 0xff, 0x83, 0x88, 0xc8, 0x82,
                  0x88, 0xc7, 0xc0, 0xAF, 0x86 }, i;
delay()
{ long u;
  for (u=0; u<4000; u++);
}
void main()
{   int d, b, j, m;
    unsigned char k;
    CommW = 0x80;
  do
  {  i=0;
     for (d=0; d<1; d++)
     {
        for (b=13; b>0; b--)
        {  delay();
           k = port[i++];
           for (j=0; j<8; j++)
           {  m=k;
              k = k & 0x80;
              if (k==00)
              portB = 0x00;
              else
```

```
            portB = 0x01;
        }
        portC = 0x01;
        portC = 0x00;
         k= m
         k <<= 1;
      }
    }
   delay();
  }
}
 while(1);
 }
```

Trisha
1BM19CS214

Program to demo the elevator interface

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

unsigned char xdata CommandWord_at_0xe803;
unsigned char xdata PortA_at_0xc800;
unsigned char xdata PortB_at_0xc801;
unsigned char xdata PresentFloor, RequestedFloor, Step = 0xf0
unsigned long xdata Count, i;

Delay()
 {  for (Count=0; Count<=4500; Count++);
 }

Reset()
 {      Step= Step & 0x0f;
        PortA = Step;
        Step = Step | 0xf0;
        PortA = Step;
 }
GoUp()
 {  Switch (RequestedFloor)
        {   case 0x0d : while (Step < 0xf3)
                        {  Step++;
                           PortA = Step;
                           Delay();
                        }
                        Reset();
                        break;
```

```
case 0x0b : while ( step < 0xf6)
            { Step ++;
              Port A = step;
              Delay ();
            }
            Reset ();
            break;

case 0x07 : while (Step < 0xf9)
            { Step ++ ;
              Port A = Step;
              Delay ();
            }
            Reset();
            break;
        }
    }

GoDown ()
{ 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 0x0e: while (Step > 0xf0)
                { Step --;
                  PortA = Step;
                  Delay();
                } Reset();
                break.
        }
    }
void main()
  { CommonWord = 0x82;
    PortA = 0xf0;
    PresentFloor = 0x0e;
    while (1) {
    RequestedFloor = PortB;
    RequestedFloor = RequestedFloor & 0x0f;
    if (RequestedFloor != 0x0f && RequestedFloor != PresentFloor) {
        if (RequestedFloor < PresentFloor)
            GoUp();
        else
            GoDown();
        PresentFloor = RequestedFloor;
      }
    RequestedFloor = PortB;
  }.
```