14/12/20 Microcontrollers 1BM19EE025

1] Drive a Stepper Motor interface to rotate the motor in anti-clockwise by N-steps. Introduce suitable delay b/w successive steps .

```
#include <stdio.h>
#include <reg51.h>
char xdata port _at_ 0xe803;
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();
    }
```

3] Design the stepper motor to rotate the motor in clockwise direction.

```
#include <stdio.h>
# include < reg51. h>
char xdata port_at_ 0xe803;
char xdata porta _at_ 0xe800;
char idata acc _at_ 0x300;

void delay
{
     for (int i =0 ; i < 800 ; i++);
}
void main()
{
       port = 0x 80 ;
       while (1)
       {
            acc = 0x 88;
            porta = acc ;
            delay();
            acc = 0x44;
            port a =acc;
            delay ();
            acc = 0x 22;
            porta = acc ;
            delay ();

            acc = 0x11;
            porta =acc;
            delay();
       }
}
```

1BM19EE025

2] Display messages FIRE and HELP alternately 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>
char  xdata CommW _at_ 0xe803;
char xdata portB _at_ 0xe801;
char xdata portC _at_ 0xe802;
char port[20] = {0x8e, 0xf9, 0xde, 0x86, 0xff, 0xff, 0xff, 0xff, 0x89,
0x86, 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==0)
              portB=0x00;
              else
              PortB = 0x01;
            }
            portc = 0x01;
            portc = 0x00;
            k = m;
            k << =1;
        }
    }
  delay ();
  }
  }
  while (1);
}
```

4) To Display Bangalore in Rolling fashion.

```
#include <stdio.h>
#include <stdlib.h>
#include <reg51.h>
char xdata commw _at_ 0xe803;
char xdata portB _at_ 0xe800;
char idata portc _at_ 0xe802;
char port[20] = {0xff, 0xff, 0xff, 0xff, 0x83, 0x88, 0xC8, 0x82; 0x88, 0xC7,
                  0xC0, 0xaf, 0x86}, i;
void delay()

{ for(int i =0; i<4000; i++);

}
void main()
{   int , d, b, j, m; unsigned char k;
    CommW = 0x80;
    do{  i=0;
      for(d=0; d<1; d++)
      {  for(b=0; b<13; b++)
         { delay();
           k = port[i++];
           for (j=0; j<8; j++)
           {   m = k;
               k = k&0x80;
               if(k==0)
                  portB = 0x00;
               else
                  portB = 0x01;
           }
           portc=0x01; portc = 0x00; k=m;
           k<<=1; } { delay(); }} while(1);}
```

Program to demo the elevator interface.

```
# include <stdio.h>
# include <reg51.h>
unsigned char xdata CommandWord _at_ 0xe803;
unsigned char xdata PortA _at_ 0xe800;
unsigned char xdata PortB _at_ 0xe801;
 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;
```

```
cax  0x06 :  while(step < 0xf6)
             {
               step++;
               PortA = step;
               Delay();
               }
               Rest();
               break;

  Case  0x07;  while (step < 0xf9)
             {
               step++;
               PortA = step;
               Delay();
               }
               Reset ();
                break;


   }
 }
 go Down()
 {
  switch (RequestedFloor)
  {
       Case  0x0d :  while(step >0xf3)
                     { step --;
                       PortA = step;
                       Delay();
                     }
                     Reset();
                     break;
```

```
case  0x06 : while (step > 0xf6)
             {
                 step--;
                 PortA = step;
                 Delay();
             }
             Reset();
             break;

   case  0x0e : while (step > 0xf0)
                {
                    step--;
                    PortA = step;
                    Delay();
                }
                Reset();
                break;

   }
 }
void main()
{
   CommandWord = 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;

   }
}
```