—: MICROCONTROLLERS

1) Drive a stepper motor interface to rotate the motor in Anti-clockwise by N steps. Introduce suitable delay between successive steps

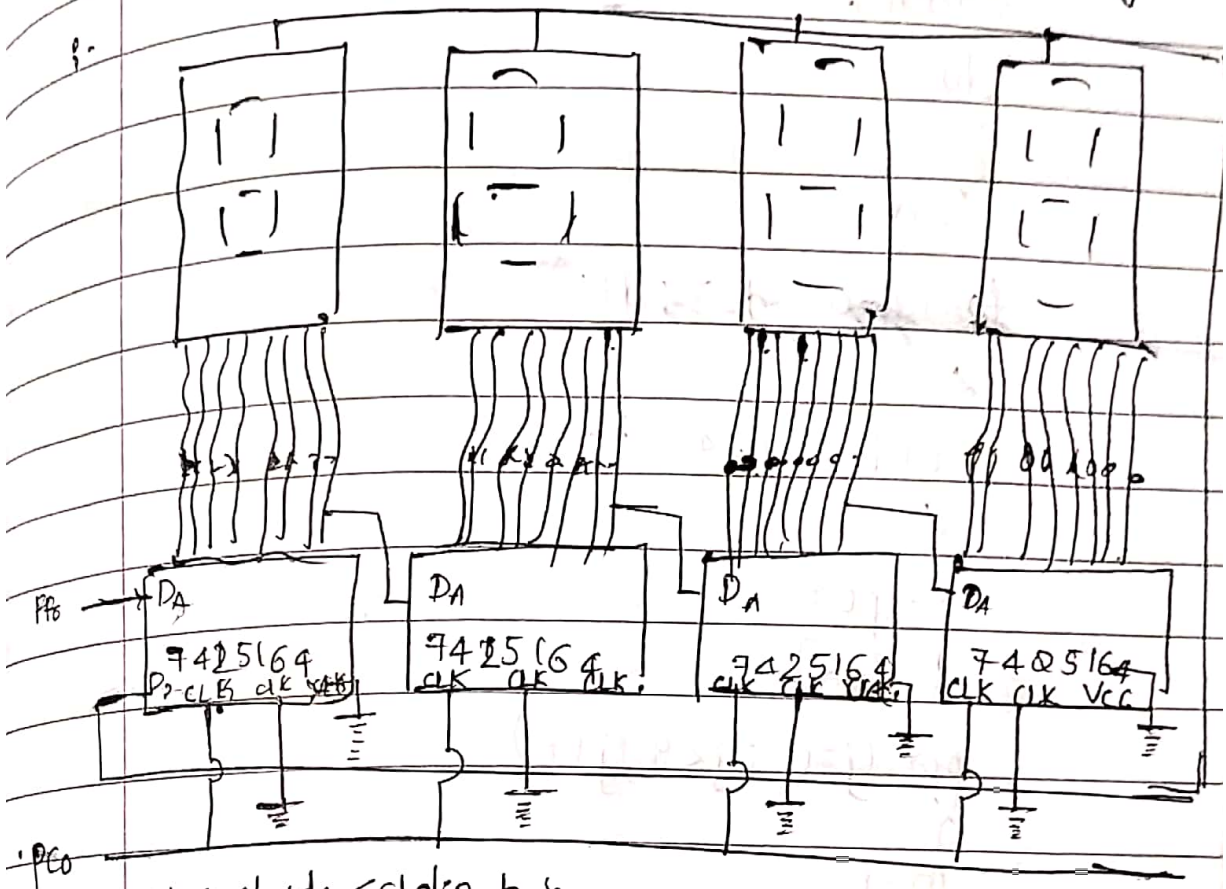
```
=> #Include <stdio.h>
#include <reg51.h>
char xdata port1 at 0xe803;
char xdata port at 0xe800;
char id at acc at 0x30;
delay();
{
int j;
for (j=0; j<800; j++)
 {
}
void main(){
port = 0x80;
while(1)
{
acc = 0x11;
porta = acc
delay();
acc = 0x22;
porta = acc;
delay();
acc = 0x44;
porta = acc;
delay();
acc = 0x88;
```

```
P0 = ~acc;
delay();
}
}
```

2) Clockwise rotation :-

```
#include<reg51.h>
#include<stdio.h>
void main();
{
void delay();
while(1)
{
P0 = 0x06;
delay();
P(0)=0x0c;
delay()
P0 = 0x09;
delay();
P0 = 0x03;
delay()
}
void delay()
{
unsigned char cnt, cnt1;
for(cnt=0; cnt<=254; cnt++)
for(cnt1=0; cnt1<254; cnt1++)
}
```

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

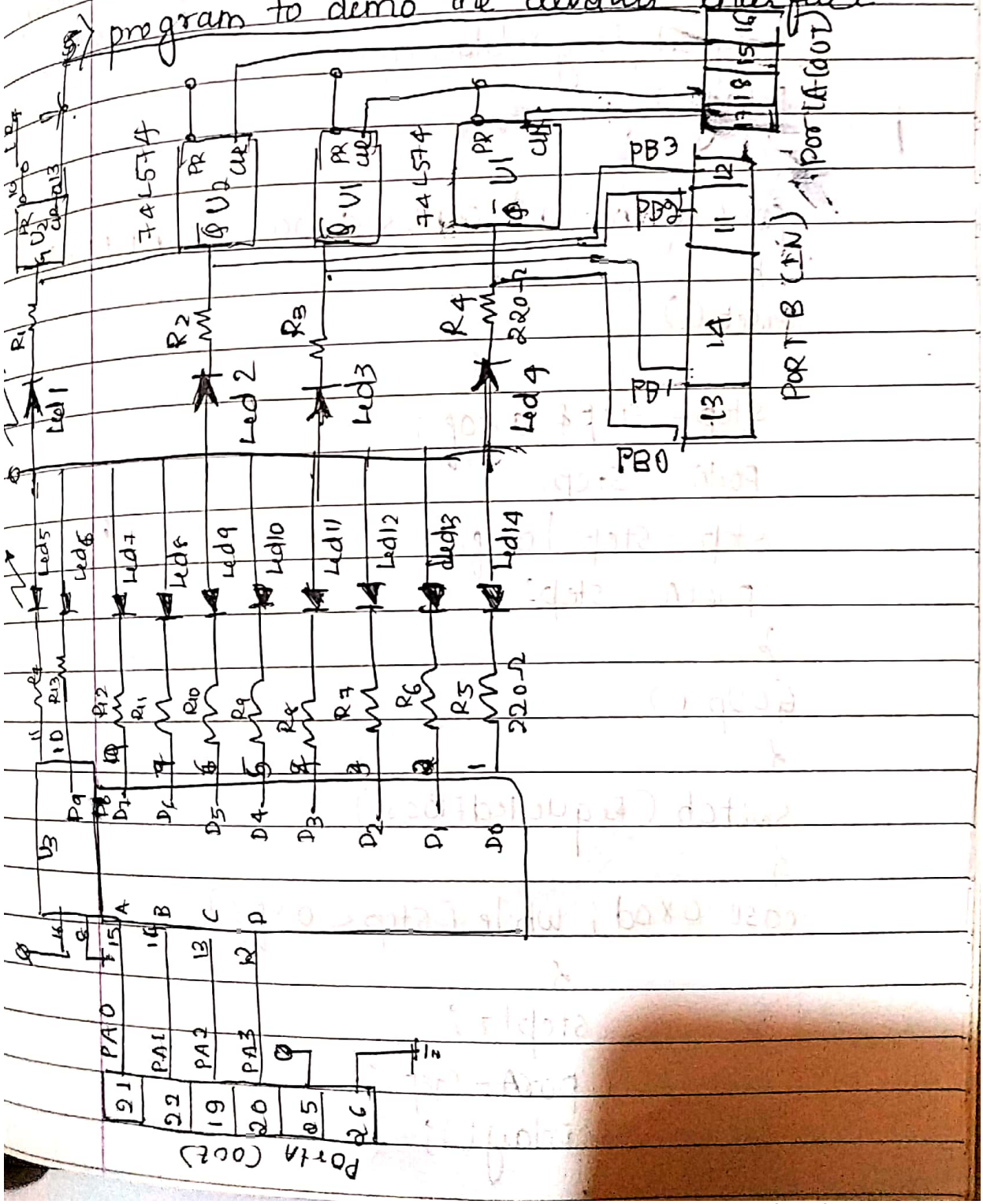



```
#include <stdio.h>
#include <reg51.h>
char xdatacommand_at_0xe803;
char xdataportB_at_0xe801;
char xdataportC_at_0xe802;
char port[20]={0x8e, 0xf9, 0xde, 0x86, 0xff, 0xff, 0xff,
               0x89, 0x86, 0xc7, 0x8c}, i;
delay()
{
long u;
for(u=0; u<800; u++)
{
```

```
Void main()
{
int d,b,j,m;
unsigned char k;
commWl= 0x80;
do
{
 i=0;
for (d=0 ; d<3 ; 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;
 }
 port C=0x01;
 port C = 0x00;
 k= m;
 K<<=1;
}
```

```
    }
    delay();
  }
  while(1);
}
```

5) program to demo the elevator interface



```
#include <stdio.h>
#include <reg51.h>
unsigned char xdataCommandWord_at_0xc8
Unsigned char xdataCommandWord_at_0xc803;
Unsigned char xdataportB_at_0xc801;
Unsigned char xdatapresentFloor, RequestedFloor,
                                    step=0xf0;
Unsigned long xdatacount, i;
Delay()
{
for(count=0; count<=4500; count++);
}
Reset()
{
 step= step& 0x0f;
 PortA = step;
 step = step |0xf0;
 portA = step;
}
GoUp()
{
switch(RequestedFloor)
{
case 0xd; while(step<0xf3)
          {
          step++;
          porA=step;
          Delay();
```

```
Reset();
break;
case 0x0b: while (step < 0x f6)
                {
                step++;
               portA = step;
                Delay();
               }
               Reset();
               Break;
case 0x07: while (step < 0x f9)
               {
               step ++;
               portA = step;
                Delay
                 }
               Reset();
               break;
  }
}
GoDown()
{
switch (RequestedFloor)
{
case 0x0d; while (step > 0f3)
                {
                step ++;
                portA = step;
                 Delay();
                 }
```

```
Reset();
break;
case 0xab; while (step > 0xf6)
{
step --;
portA = step;
Delay();
}
Reset();
break;
case 0xae; while(step > 0xf0)
{
step --;
portA = step;
Delay();
}
Reset();
break;
}
}
Void main()
{
Commandword = 0x82;
portA = 0xf0;
presentFloor = 0xae;
while(1)
{
RequestedFloor = portB;
```

```
RequestedFloor = RequestedFloor & 0x0f;
if (RequestedFloor != 0x0f && RequestedFloo
        -r != presentFloor)
{
if (RequestedFloor < presentFloor)
    GoUp();
else
    GODown();
    presentFloor = RequestedFloor;
}
RequestedFloor = PortB;
}
```

4) Seven-Segment-display - display 'Banglore' in rolling fashion.

=>
```
#include<stdio.h>
#include<conio.h>
char xdata CommW at 0xe803;
char xdata portB at 0xe801;
char xdata portc at 0xe802;
char port[20] = {0xff, 0xff, 0xff, 0xff, 0x83,
        0x88, 0x88, 0xc8, 0x82, 0x88, 0xc7, 0xc0, 0x8
        , 0x86), i;
  delay()
  {
   long u;
  for (u=0; u<4000; u++);
  }
```

```
void main()
{
int b,d,j,m,i;
unsigned char k;
control W = 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==0)
portB=0x00;
else
portB=0x01;
}
portC=0x01;
portC=0x00;
k=m;
k<<=1;
```

```
}
delay();
}
}
while(1);
}
```