1 - LAB → CLOCKWISE

* MICROCONTROLLER : [8051]

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

void delay (int val) {
        int i,j;
    for(i=0 ; i<val; i++) {
        for (j=0; j<100 ; j++) }
        ;
}

void main () {
    do {
        P2 = 0x03;
        delay (1000);
        P2 = 0x06;
        delay (1000);
        P2 = 0x0C;
        delay (1000);
    }
    while(1);
}
```

2-LAB : ANTICLOCKWISE

```
#include <stdio.h>
#include <reg51.h>
void delay()
{
    int i,j;

    for(i=0; i<1000; i++)
    {
        for(j=0; j<1000; j++)
        {
        }
    }

}

void main()
{
    do
    {
        P2 = 0x09;
        delay();
        P2 = 0x0C;
        delay();
        P2 = 0x06;
        delay();
        P2 = 0x03;
        delay();
    } while(1);

}
```

3 - LAB PROGRAM

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

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

void main() {

    char xdata command-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,
                      0xbc };

    int i, j, d, b;
    char k, m;
    command = 0xb0;

    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(1);
 }
 swhile(1);
}
```

4 - LED * BENGLORE

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

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

void main()
{
    char port[] = {0x80, 0x88, 0xBB, 0x90,
                   0x88, 0x87, 0xBB, 0xLE,
                   0x86};

    int i, j, d;
    char k, m;

    do
    {
        i=8;
        for(d=0; d<10; d++)
        {
            m=k;
            k = k & 0x00;
            if(k==00)
            {
                portB = 0x00;
            }
            else
            {
                portB = 0x01;
            }
            portG = 0x01;
            portC = 0x00;
```

```
        K=m;
        K<<=1;
    }
    delay();
}
    delay();
}
while(1);
}
```

* LAB - 5 Program :-

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

unsigned char *dataCommandWord_at_ 0xe803;
unsigned char *dataPortB_at_ 0xe800;
unsigned char *dataPortB_at - 0xe801;
unsigned char *dataPresentFloor, RequsLFloor,
        step = 0x00;
unsigned long *dataCount, i;

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

```
Reset() {
        step = step & 0x0f;
        portB = step;
        step = step | 0xf0;
        PortA = step;
}

GoUP () {
        switch (Requested floor)
        {

case 0x0d : while (step < 0xf3)
            {
            step++;
            PortA = step;
            Delay();
            }
            Reset();
            break;

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

```
case 0x07 : while (step<0x09)
                {
                  step++;
                  PortA = step;
                  Delay();
                }
              Reset();
              break;
            }
          }

GoDown() {
          switch (RequestAFloor) {

case 0x0A : while(step>=0x03)
              {
                step--;
              PortA =step;
               Delay();
              }
              Rest();
              break;

case 0x0b : while (step>=0x06)
              {
                step--;
              PortA =step;
               Delay();
              }
              Reset();
              break;
```

```
case 0x0c : while (step > 0x[0)
            {
              step--;
            PortA = step;
              Delay();
            }
            ResetC();
            break;
        }
    }

void main() {
        CommandWord = 0x82;
        PortA = 0x[0;
        PresentFloor = 0x0c;

    while(1){
        RequestedFloor = PortB;
        RequestedFloor = RequestedFloor &
                         0x0f;

if (RequestedFloor != 0x0[ && RequestedFloor
                          != PresentFloor) {

    if (RequestedFloor < PresentFloor)
                    goUp();
    else
                    goDown();
    presentFloor = RequestedFloor;
    }
    RequestedFloor = PortB;
}
```