

A dark blue vertical bar on the left side of the page, with a blue arrow pointing right from its center.

# Elevator Task

Using 8051 (C Language)

Several thin, curved lines in dark blue and light grey originating from the bottom left corner of the page.

## Team Members:

Mohammed Almotasem

Remon Alber

Omar Abdelzaher

Galal Hossam

Table of Contents:

**Task Description .....2**

**Schematic .....3**

**C Code Samples .....4**

    To Run The File ..... 5

## Task Description

- The building consists of four floors.
- Each floor has two buttons one for going up, and one for going down except for the ground and the last floor have one button only.

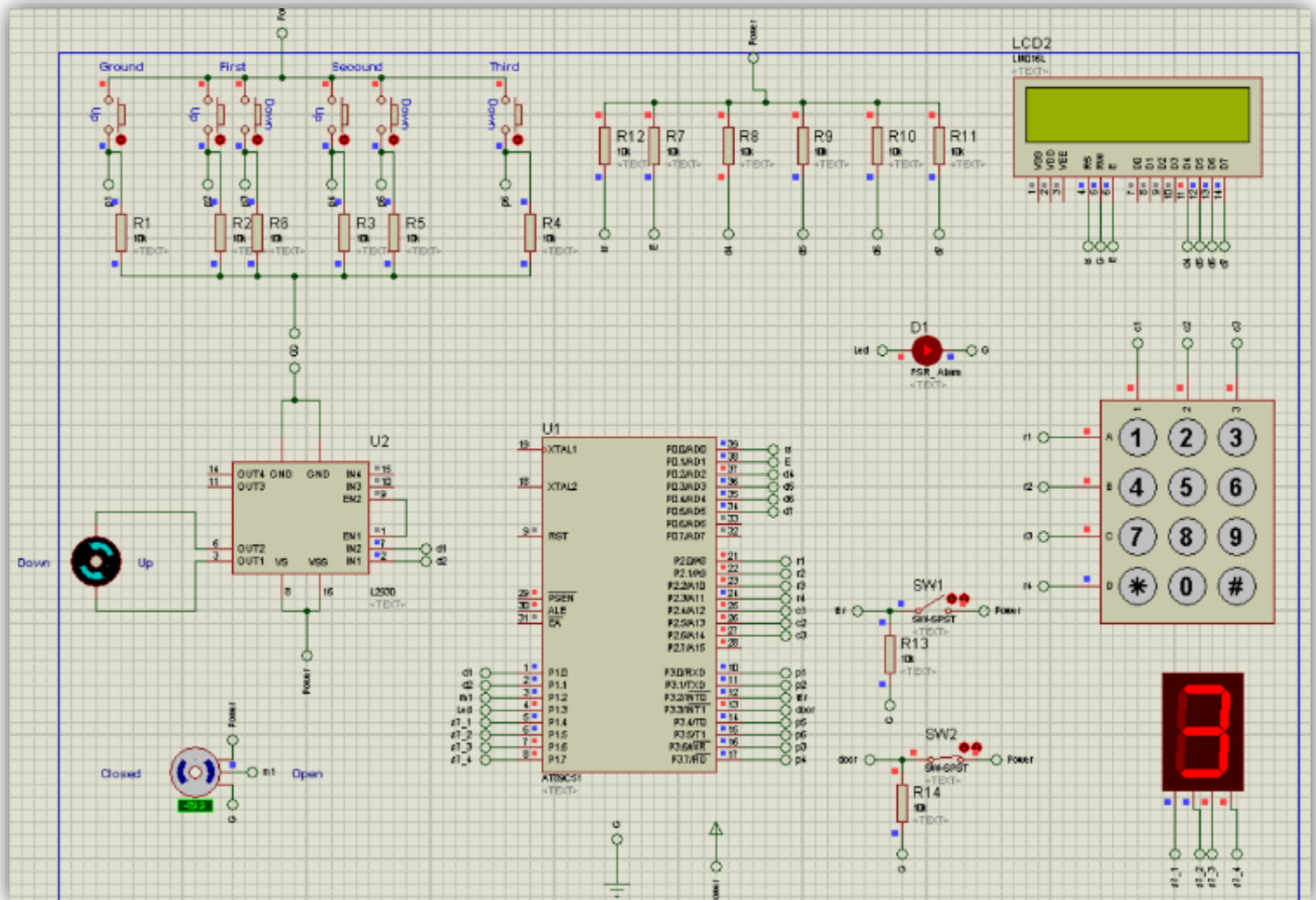
The buttons are: “Ground, First, Second, Third”

- There are five buttons inside the elevator cart, one for each floor.

The previous buttons + “#” button in keypad (Open button).

- The elevator door waits for 5 seconds before closing, and the door can be stopped from closing by pushing an Open button “#”. Also, the elevator door opens if someone blocks the door. That happens by opening a switch (SW2) which calls the ISR to open the door as a result of blocking it.
- If the elevator is going up, it does not stop for a “going down” request, and vice versa: if it is going down it does not stop for a “going up request”.
- The elevator system saves the requests, for example, if the elevator is going up, and you pressed on the “going down” button, the elevator ignores the request initially, but when it reaches the top floor it goes back to pick up that one who wants to go down.
- The maximum load is four persons. If the number of persons exceeds four, a switch (SW1) is closed and a Red Led (Alarm) is ON and the elevator will not move as an ISR is called.
- The floor number is displayed on a 7-segment.
- We used a servo motor to move the elevator door.
- We used a DC motor to move the elevator box.
- We used an LCD to show the status of the elevator.
  - Going to which floor.
  - The door is closing or opening.
- We used a speaker inside the elevator box to play a certain song or a prayer.

## Schematics



**Note:** The IDE used is MikroC PRO for 8051



# C Code Samples

```
void door_open() iv IVT_ADDR_EX1 ilevel 0 ics ICS_AUTO {  
  
    P1.B2=1;  
    delay_ms(1);  
    P1.B2=0;  
    delay_ms(5000);  
}  
void fsr_open() iv IVT_ADDR_EX0 ilevel 0 ics ICS_AUTO {  
    P1.B3 = 1;  
    delay_ms(2000);  
    P1.B3 = 0;  
}  
void leds_check() iv IVT_ADDR_ET0 ilevel 0 ics ICS_AUTO  
{  
  
    if(P3.B0)  
    {  
        flag[0] = 1;  
        wanted = 0;  
    }  
  
    if(P3.B1)  
    {  
        flag[1] = 1;  
        wanted = 1;  
        state = 1;  
    }  
}
```

```
void main()  
{  
    P1 = 0x00;  
    P3=0x00;  
    P2=0xFF;  
    TMOD=0x01;  
    IE=0x87;  
    SCON = 0x50;  
    TL0 = 0xE5;  
    TH0 = 0xBE;  
    TCON.B0 = 0;  
    TCON.B2 = 1;  
    TCON.TR0 = 1;  
    Lcd_Init();  
    delay_ms(10);  
    Lcd_Cmd(_LCD_CURSOR_OFF);  
    while(1)  
    {  
        lcd_Cmd(_LCD_CLEAR);  
        delay_ms(1000);  
        for(i = 0; i < 4 ; i++)  
        {  
            if(flag[i] == 1)  
            {  
                moving(i);  
            }  
        }  
    }  
}
```

```

void close_door(void)
{
    Lcd_Out(1,1, "Close");
    P1.B2=1;
    delay_ms(2);
    P1.B2=0;
    delay_ms(2);
    IE = 0x82;
}

void open_door(void)
{
    Lcd_Out(1,1, "Open ");
    P1.B2=1;
    delay_ms(1);
    P1.B2=0;
    delay_ms(2);
    IE = 0x87;
}

void move_elevat(signed char direction)
{
    if(direction > 0)
    {
        Lcd_Out(1,1, "Up   ");
        P1.B0=1;
        P1.B1=0;
    }
}

```

```

void Sevenseg(char x)
{
    P1 &= 0x0f;
    switch(x)
    {
        case 1:
        {
            P1|=0b10000000;
        }break;
        case 2:
        {
            P1|=0b01000000;
        }break;
        case 3:
        {
            P1|=0b11000000;
        }break;
        case 0:
        {
            P1|=0b00000000;
        }break;
    }
}

```

## To Run The File:

```
#include<stdio.h>

char flag[] = {0, 0, 0, 0};

char flag2 = 1;

char state = 1;

char current, wanted, time, direc = 0;

void moving(int index);

void open_door(void);

void close_door(void);

void Sevensseg(char x);

void stringtxt(char x);

void delay1(void);

void deduce_want(char index);

void move_elevat(signed char direction);


sbit LCD_RS at P0_0_bit;

sbit LCD_EN at P0_1_bit;

sbit LCD_D7 at P0_5_bit;

sbit LCD_D6 at P0_4_bit;

sbit LCD_D5 at P0_3_bit;

sbit LCD_D4 at P0_2_bit;


void door_open() iv IVT_ADDR_EX1 ilevel 0 ics ICS_AUTO {

    P1.B2=1;

    delay_ms(1);

    P1.B2=0;

    delay_ms(5000);

}

void fsr_open() iv IVT_ADDR_EX0 ilevel 0 ics ICS_AUTO {
```



```

    P1.B3 = 1;

    delay_ms(2000);

    P1.B3 = 0;
}

void leds_check() iv IVT_ADDR_ET0 ilevel 0 ics ICS_AUTO
{

    if(P3.B0)
    {
        flag[0] = 1;
        wanted = 0;
    }

    if(P3.B1)
    {
        flag[1] = 1;
        wanted = 1;
        state = 1;
    }

    if(P3.B6)
    {
        flag[1] = 1;
        wanted = 1;
        state = -1;
    }

    if(P3.B7)
    {

```

```
flag[2] = 1;
wanted = 2;
state = 1;
}
```

```
if(P3.B4)
{
    flag[2] = 1;
    wanted = 2;
    state = -1;
}
```

```
if(P3.B5)
{
    flag[3] = 1;
    wanted = 3;
}
```

```
P2.B0=P2.B1=P2.B2=P2.B3=1;
```

```
    P2.B0=0;
    if(P2.B4==0)
    {

        flag[1] = 1;
    }
```

```
P2.B0=P2.B1=P2.B2=P2.B3=1;
```

```
    P2.B0=0;
    if(P2.B5==0)
    {
```

```

    flag[2] = 1;
}
P2.B0=P2.B1=P2.B2=P2.B3=1;
P2.B0=0;
if(P2.B6==0)
{

    flag[3] = 1;
}

P2.B0=P2.B1=P2.B2=P2.B3=1;
P2.B3=0;
if(P2.B5==0)
{
    flag[0] = 1;
}
P2.B0=P2.B1=P2.B2=P2.B3=1;
P2.B3=0;
if(P2.B6==0)
{
    flag2 = 1;
}
TCON.TF0=0;
TCON.TR0=1;

}
char i = 0;
char j =0;
char z =0;

```

```

char w =0;

char wanted2 = 0;

char direc1, direc2 =0;

void main()
{
    P1 = 0x00;

    P3=0x00;

    P2=0xFF;

    TMOD=0x01;

    IE=0x87;

    SCON = 0x50;

    TLO = 0xE5;

    TH0 = 0xBE;

    TCON.B0 = 0;

    TCON.B2 = 1;

    TCON.TR0 = 1;

    Lcd_Init();

    delay_ms(10);

    Lcd_Cmd(_LCD_CURSOR_OFF);

    while(1)
    {
        Lcd_Cmd(_LCD_CLEAR);

        delay_ms(1000);

        for(i = 0; i <4 ; i++)
        {
            if(flag[i] == 1)
            {
                moving(i);
            }
        }
    }
}

```

```

    }
}

void deduce_want(char index){
    for(z=0; z<4; z++)
    {
        if(flag[z]==1 && z!=index)
            wanted2 = z;
    }
    if((wanted - current) > 0)
        direc1 = 1;
    else
        direc1 = -1;
    if((wanted2 - current) > 0)
        direc2 = 1 * state;
    else
        direc2 = -1 * state;
    if(direc1 == direc2)
    {
        wanted = wanted2;
        time = abs(wanted - current);
    }
}

void moving (int index)
{
    Sevensseg(current);
    wanted = index;
    flag2 = 1;
    time = abs(wanted - current);
    if((wanted - current) > 0)
        direc = 1;

```

```

else
    direc = -1;
//lcd_Out("5sec");
delay1();
close_door();
delay_ms(1000);
for( j = 0; j < time; j++)
{
    deduce_want(index);
    stringtxt(wanted);
    move_elevat(direc);
    current+= direc;
    Sevensseg(current);
    delay_ms(500);
}
P1.B0 =0;
P1.B1 =0;
flag[wanted] = 0;
delay_ms(1000);
open_door();
delay_ms(500);
}

```

```

void close_door(void)
{
    Lcd_Out(1,1, "Close");
    P1.B2=1;
    delay_ms(2);
    P1.B2=0;
    delay_ms(2);
}

```

```

    IE = 0x82;

}

void open_door(void)
{
    Lcd_Out(1,1, "Open ");
    P1.B2=1;
    delay_ms(1);
    P1.B2=0;
    delay_ms(2);
    IE = 0x87;

}

void move_elevat(signed char direction)
{
    if(direction > 0)
    {
        Lcd_Out(1,1, "Up ");
        P1.B0=1;
        P1.B1=0;
        delay_ms(2000);
        //    P1.B0=0;

    }
    else
    {
        Lcd_Out(1,1, "Down ");
        P1.B0=0;
        P1.B1=1;
    }
}

```

```
    delay_ms(2000);  
    //    P1.B1=0;  
  
    }  
  
}
```

```
void Sevensseg(char x)  
{  
    P1 &= 0x0f;  
    switch(x)  
    {  
        case 1:  
        {  
            P1|=0b10000000;  
        }break;  
        case 2:  
        {  
            P1|=0b01000000;  
        }break;  
        case 3:  
        {  
            P1|=0b11000000;  
        }break;  
        case 0:  
        {  
            P1|=0b00000000;  
        }break;  
    }  
}
```



```

void stringtxt(char x)
{
    switch(x)
    {
        case 1:
        {
            Lcd_Out(2,1, "Go to 1");
        }break;
        case 2:
        {
            Lcd_Out(2,1, " Go to 2");
        }break;
        case 3:
        {
            Lcd_Out(2,1, "Go to 3");
        }break;
        case 0:
        {
            Lcd_Out(2,1, "Go to 0");
        }break;
    }
}

void delay1(void)
{
    while(flag2)
    {
        flag2 = 0;
        delay_ms(3000);
    }
}

```