National University of Sciences and Technology (NUST)
School of Electrical Engineering and Computer Science

Department of Electrical Engineering

EE-222: Microprocessor Systems

PROJECT REPORT

PROJECT TITLE: Automatic Car Parking System

Name	Reg.no
1.Saqib Javed	4964
2.Abbas Ahmed Siddiqui	5545
3.Muhammad Ibrahim Zafar	5770

Hardware Components.

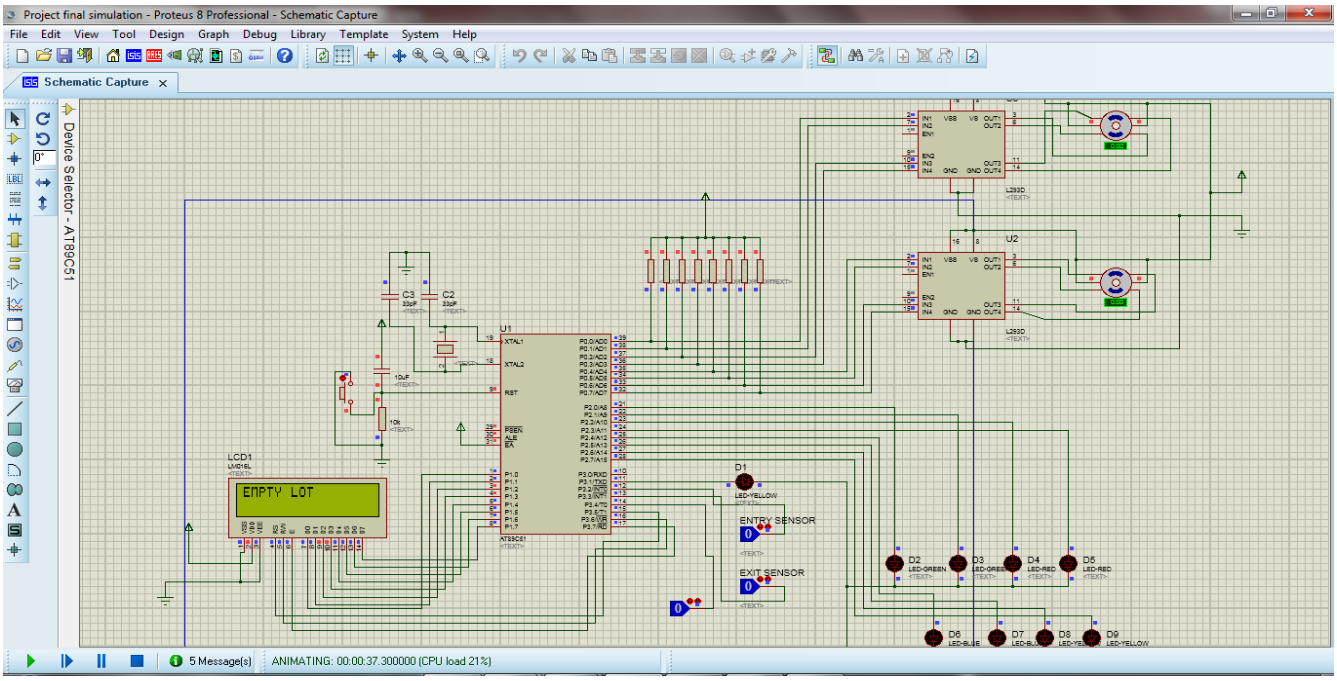
- 8051 Micro Controller.
- Battery
- 8 Pull up Resistors (10k).
- Other Resistors (10k for reset cicruit,100 for LED).
- Oscillator.
- LCD Display.
- Capacitors (33pF*2,10uF).
- 2 L293D IC.(motor driving ic)
- Stepper motor.
- ChipMAX.(for burning code in micro controller)
- 1 LDR
- 2 IR sensor
- LEDs

National University of Sciences and Technology (NUST)
School of Electrical Engineering and Computer Science

Project Simulation screen shots:

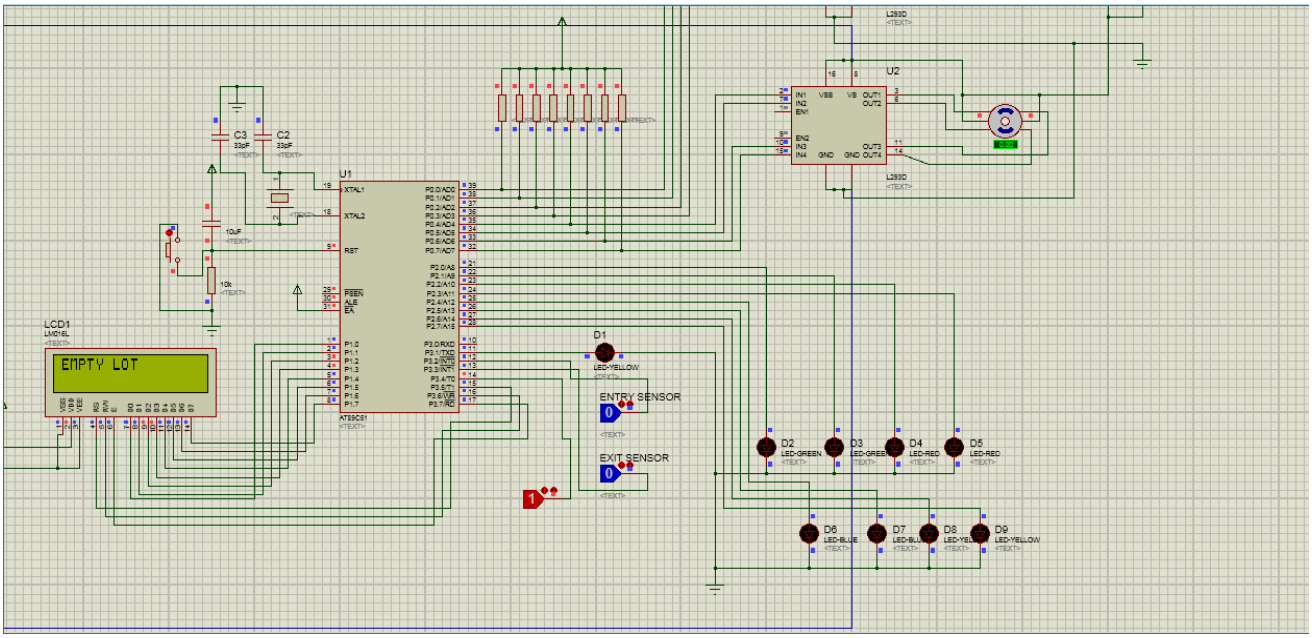
1. When code start:

No car and Lights Off



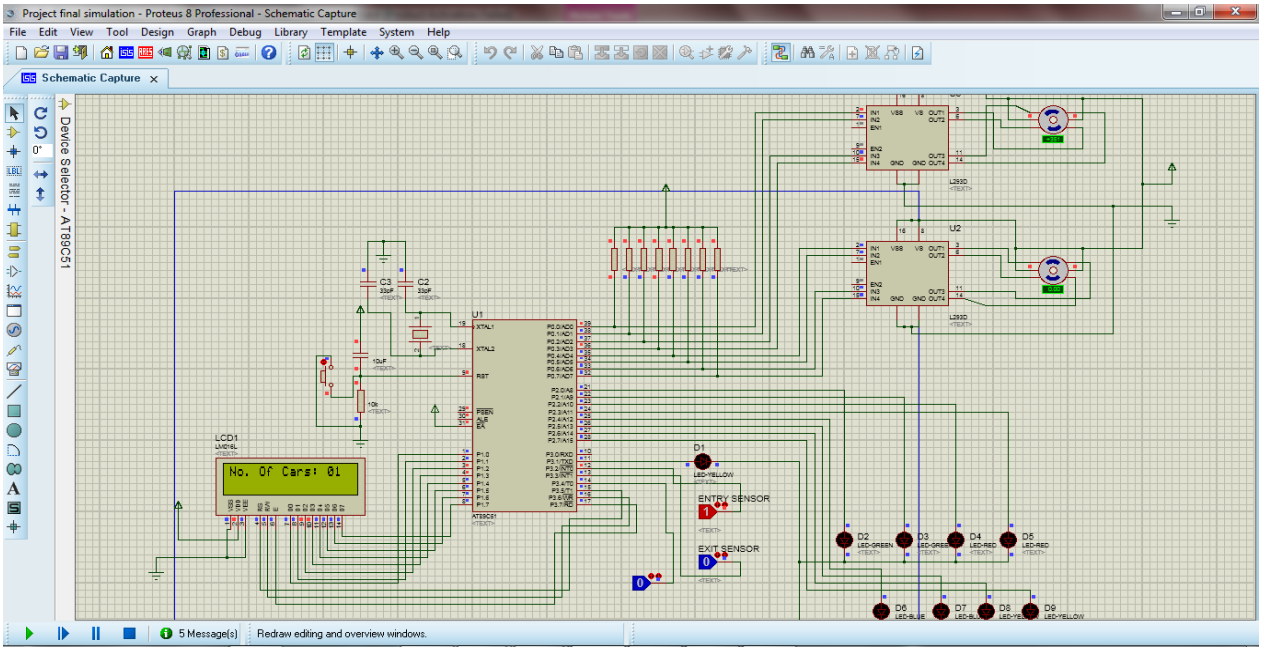
2. When empty lot and It's night:

No car so are Lights Off automatically even it's night.



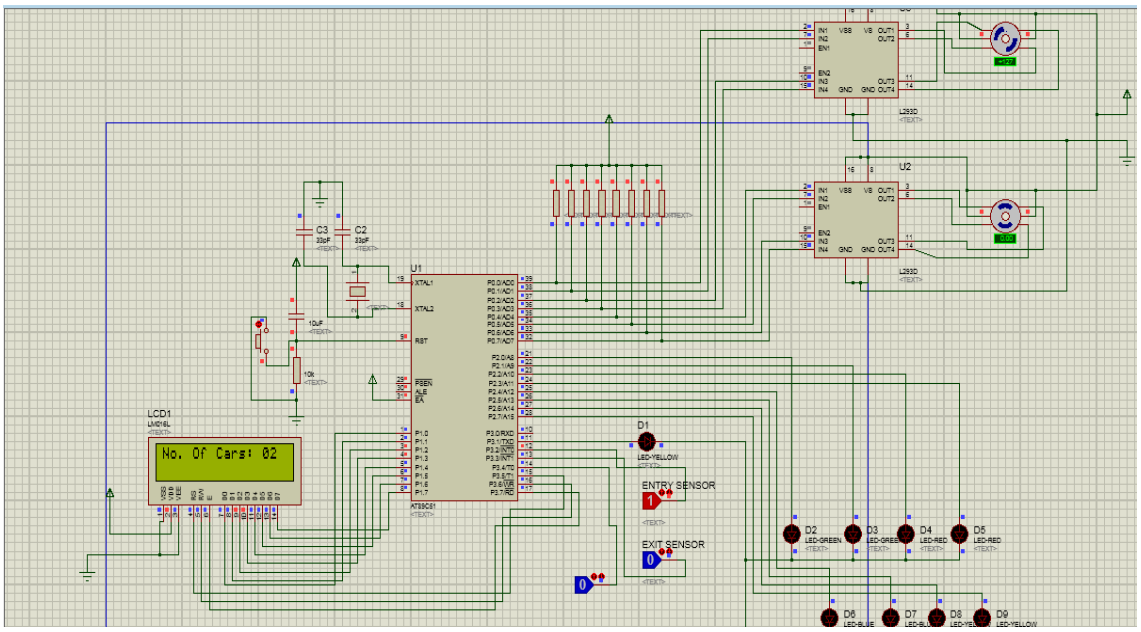
3. When car enters:

No. of cars++(incremented),entry motor rotates and Lights still Off as LDR input=0(day time)



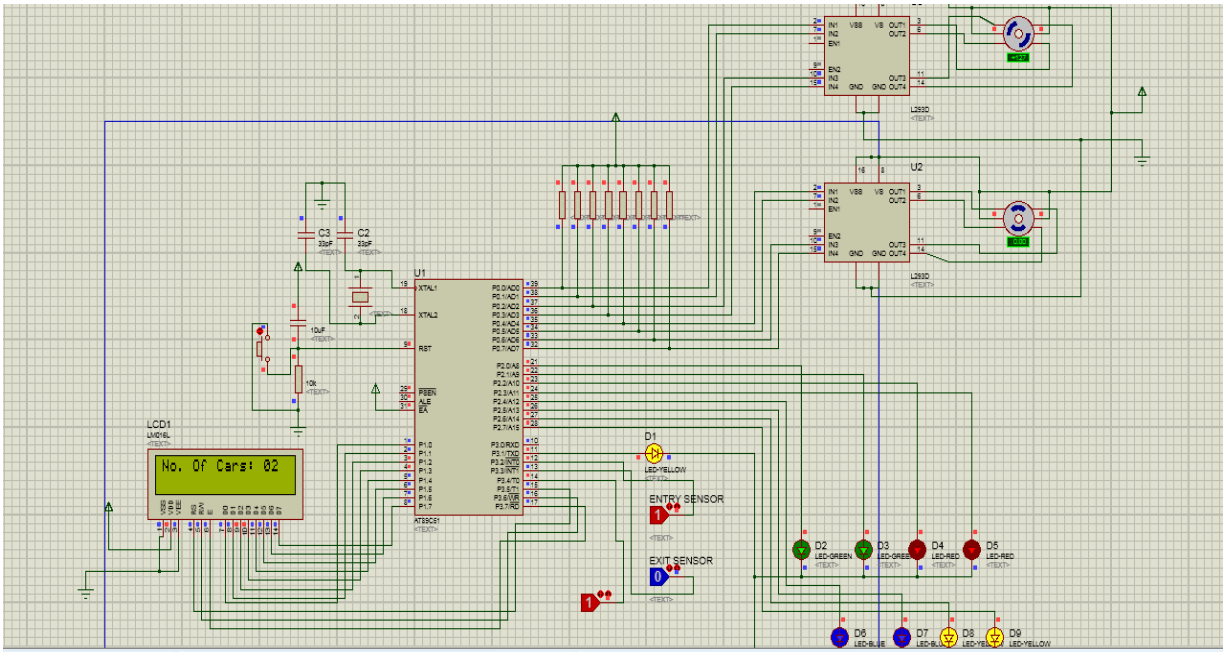
4. When another car enters:

No. of cars++, entry motor rotates and Lights still Off as LDR input=0(day time).



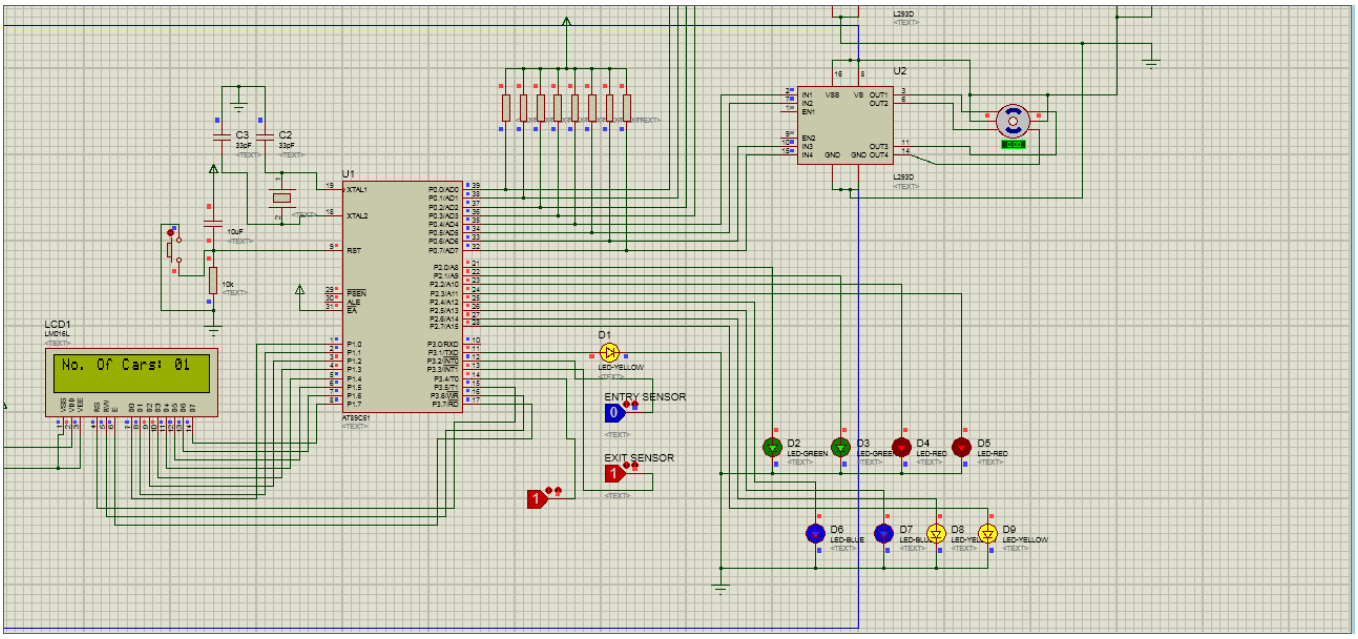
5. When there are cars and it's night:

Cars in parking and it's night, so lights ON.



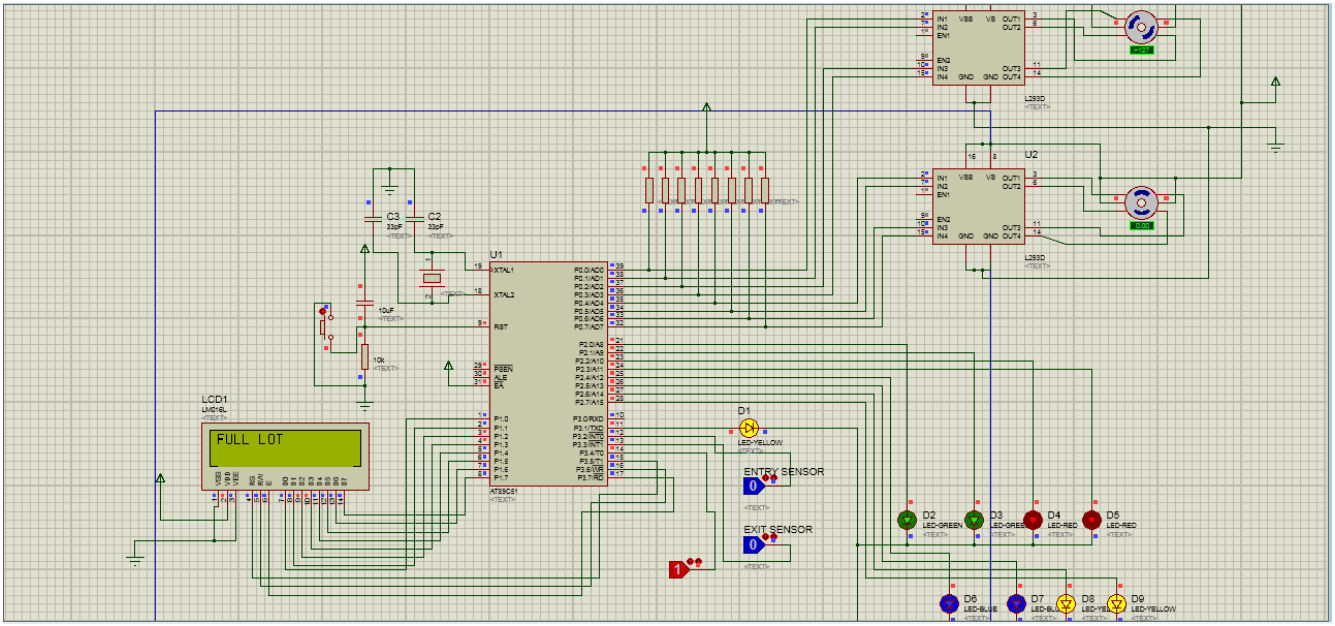
6. When a car exit:

No. of cars--, exit motor rotates and as there is one car remaining so lights still on.



7. When 16 cars had entered:

after 16 cars Parking Full.



CODE

```
#include <stdio.h>

#include <reg51.h>


unsigned int i,j;

unsigned char z;

unsigned char count=0;

unsigned char temp=0;

sbit LDR= P3^4;                                    //input from LDR

sbit Led=P3^1;                                     //Output for LED
```

```
sfr ldata = 0x90;
                                    //configuring port 1 for LCD

unsigned char emp1[]= "EMPTY LOT ";

unsigned char fu[]= "FULL LOT ";

unsigned char noc[]= "No. Of Cars: ";

int value[2] = {0};

int* div_num(unsigned int t)

{

   if(t<10)

   {

         value[0] = t;

         value[1] = 0;

   }

   if(t>=10)

   {

         value[1] = t/10;

         value[0] = t%10;

   }

   return value;

}

sbit rs = P3^5;  // declare P3.5 as rs pin

sbit en = P3^7;  // declare p3.7 as enable pin

sbit rw = P3^6;  // declare p3.6 as read/write pin

sbit b = P1^7;   // busy flag port 1.7


void Delay(unsigned int x)
```

```
 // Generic function for 10ms Delay
{
   for(i=0;i<x;i++)
   for (j=0;j<10000;j++);
}
void Delay123(unsigned int x)
// Generic function for 1ms Delay(for motor )
{
   for(i=0;i<x;i++)
   for (j=0;j<1000;j++);
}
void busy()
{
   en = 0;   // disable display
   P1 = 0xFF;  // configur P3 as input
   rs = 0;   // clear rs pin for command
   rw = 1;   // set rw pin to read
   while(b==1)
   {
         en=0;   // strob LCD till P3.7 is 1
         en=1;
   }
   en=0;
}
void lcdcommand(unsigned char a)
```

```
{
   busy();   // check for LCD is busy or not
   rs = 0;   // clear rs pin for command
   rw = 0;   // clear rw pin to write
   P1 = a;   // send command character
   en = 1;   // strob LCD
   en = 0;
}
void data1(unsigned char b)
{
   busy();   // check for LCD is busy or not
   rs = 1;   // set rs pin for data
   rw = 0;   // clear rw pin to write
   P1 = b;   // send data character
   en = 1;   // strob LCD
   en = 0;
}
//**********************Interrupt for enter******************//
void Gate_Enter() interrupt 0
//Interrupt INT0 p3.2, address=0003H
{
   unsigned char temp=count;
Delay(10);
   for(z=0;z<130;z++)
   {
   P0=0x03;          //00110000
```

```
 Delay123(3);
 P0=0x06;         //01100000
 Delay123(3);
 P0=0x0C;       //11000000
 Delay123(3);
 P0=0x09;      //10010000
  Delay123(3);
 }
 Delay(70);                          //Barrier opens for 10 seconds
for(z=0;z<130;z++)
 {
 P0=0x09;      //10010000
  Delay123(3);
 P0=0x0C;       //11000000
 Delay123(3);
 P0=0x06;         //01100000
 Delay123(3);
 P0=0x03;          //00110000
 Delay123(3);
 }              //Motor in opposite direction to close the barrier
  P0=0x00;
count++;
 temp=count+48;
switch (temp)
{
```

```
case (0):
{
lcdcommand(0x01);
for(z=0;z<9;z++)
{
          data1(emp1[z]);
}
    lcdcommand(0x0c);
break;
}
case (64):
{
//unsigned char ful[]= "    FULL LOT";
lcdcommand(0x01);
for(z=0;z<8;z++)
{
          data1(fu[z]);
}
    lcdcommand(0x0c);
break;
}
default:
{
      lcdcommand(0x01);
          for(z=0;z<13;z++)
```

```
                {
                      data1(noc[z]);
                }
                div_num(count);
                for(z= 0 ; z < 2 ; z++ )
                      data1(value[1-z]+48);
                lcdcommand(0x0c);
                break;
     }
    }
   }
 //******************Interrupt for exit******************//
void Gate_Exit() interrupt 2
//Interrupt INT1 p3.3 , address=0013H
{
unsigned char temp=count;
Delay(10);
   for(z=0;z<130;z++)
   {
   P0=0x30;          //00110000
Delay123(1);
   P0=0x60;         //01100000
   Delay123(1);
   P0=0xC0;       //11000000
   Delay123(1);
   P0=0x90;     //10010000
```

```
Delay123(1);
 }
 Delay(70);                          //Barrier opens for 10 seconds
for(z=0;z<130;z++)
 {
 P0=0x90;     //10010000
Delay123(1);
 P0=0xC0;      //11000000
 Delay123(1);
 P0=0x60;        //01100000
 Delay123(1);
 P0=0x30;         //00110000
 Delay123(1);
 }
 P0=0x00;
count--;
 temp=count+48;
switch (count)
{
case (0):
{
  lcdcommand(0x01);
  for(z=0;z<9;z++)
  {
            data1(emp1[z]);
```

```
}
    lcdcommand(0x0c);
break;
}
case (64):
{
          lcdcommand(0x01);

for(z=0;z<8;z++)
{
          data1(fu[z]);
}
    lcdcommand(0x0c);
break;
}

default:
{
          lcdcommand(0x01);

          for(z=0;z<13;z++)
          {
                data1(noc[z]);
          }
          div_num(count);
```

```
                for(z = 0 ; z<2 ; z++ )
                      data1(value[1-z]+48);
                lcdcommand(0x0c);
     break;
     }
 }
}
   //*************************main*******************************//
void main()
{
   unsigned char wel[]= "    WELCOME ";
  P0=0x00;        //output configure for motor
  P2=0x00;
  P1=0x00;                          // P3 and P2 as output ports
  P3=0x00;
  lcdcommand(0x3C);                  // initialize LCD
  lcdcommand(0x0E);
  lcdcommand(0x01);
for(z=0;z<12;z++)
{
 data1(wel[z]);
}
lcdcommand(0x0c);
Delay(60);
lcdcommand(0x01);
```

```
for(z=0;z<9;z++)
{
   data1(emp1[z]);
}
   lcdcommand(0x0c)
;
  TCON=0x05;                                  //Edge sensitive
  IE=0x85;                        //enabling interrupts
  LDR=1;       //Configure input
  while (1)
  {      if (count==0)
         {
               Led=0;
               P2=0x00;
         }
         else
         {
       if(LDR==1)            //Check for LDR Input, 1=high R =Night
               {     Led=1;
                     P2=0xFF;
               }
               else if(LDR==0)
               {     Led=0;
                     P2=0x00;
               }
```

```
            }

    }

}
```