```
/*
* main.c
*
*  Created on: 25-Aug-2017
*      Author: utkarsh
*/

#define F_CPU 8000000UL
#define LCD_PRT PORTA
#define LCD_DDR DDRA
#define LCD_PIN PINA
#define LCD_RS 3
#define LCD_EN 2
#define LCD_RW 1
#include <avr/io.h>
#include <util/delay.h>
#include <stdlib.h>
#include <avr/interrupt.h>

volatile unsigned int counter=0;                    //Counter for Timer/Counter1
int day=0;                                          //Number of days passed
unsigned int month[6];                              //Array of monthly usage data
char month_no=0;
int count=0;                                        //PIR Count of people

void delay_us(int d)
{
        _delay_us(d)
}

void delay_ms(int d)
{
        _delay_ms(d)
}


// Functions of  Timer/Counter1
ISR(TIM1_OVF)                                       //Timer/Counter1 Overflow ISR
{
        counter++;                                  //+2 sec
        TCNT1H=0xC2;                                // Adding offset
        TCNT1L=0xF5;
}

void initT1()                                       //Initialization of Timer/Counter1
{
        TCNT1H=0xC2;                                // Adding offset
        TCNT1L=0xF5;
        TCCR1A=0x00;                                //Normal Mode
        TCCR1B=0x05;                                //prescaler of 1024
        TIMSK1 |= 1<<TOIE1;                         //enable Timer/Counter1 overflow interrupt
}

//Function of Pir Sensor
ISR(INT0)
{
        count++;
        GIMSK = 0x00;                                   //Disable External Interrupt Request
        delay_ms(120000);                               //Delay of 2 min
        GIMSK =(1<<INT0)|(0<<PCIE1)|(0<<PCIE0);         //External Interrupt Request 0
                                                        //Enable and Pin Change Interrupt
                                                        //Disable for 1 and 0
}

void initI0()                                           //Initialization Of External Interrupt 0
{
        GIMSK =(1<<INT0)|(0<<PCIE1)|(0<<PCIE0);         //External Interrupt Request 0
                                                        //Enable and Pin Change Interrupt
                                                        //Disable for 1 and 0
        MCUCR = (1<<ISC01)|(1<<ISC00);                  //The rising edge of INT0 generates
                                                        //an interrupt request
}

//LCD
void lcdCommand( unsigned char cmnd )
{
        LCD_PRT = (LCD_PRT & 0x0F) | (cmnd & 0xF0);
        LCD_PRT &= ~ (1<<LCD_RS);                       //RS = 0 for command
        LCD_PRT &= ~ (1<<LCD_RW);                       //RW = 0 for write
        LCD_PRT |= (1<<LCD_EN);                         //EN = 1 for H-to-L pulse
        delay_us(1);                                    //make EN pulse wider
        LCD_PRT &= ~ (1<<LCD_EN);                       //EN = 0 for H-to-L pulse

        delay_us(20);                                   //wait

        LCD_PRT = (LCD_PRT & 0x0F)|(cmnd<<4);           //Send low nibble to D4-D7
        LCD_PRT |= (1<<LCD_EN);                         //EN = 1 for H-to-L pulse
        delay_us(1);                                    //make EN pulse wider
        LCD_PRT &= ~ (1<<LCD_EN);                       //EN = 0 for H-to-L pulse
}

void lcdData( unsigned char data )
{
        LCD_PRT = (LCD_PRT & 0x0F) | (data & 0xF0);     //Send high nibble to D4-D7
        LCD_PRT |= (1<<LCD_RS);                         //RS = 1 for data
        LCD_PRT &= ~ (1<<LCD_RW);                       //RW = 0 for write
        LCD_PRT |= (1<<LCD_EN);                         //EN = 1 for H-to-L pulse
        delay_us(1);                                    //make EN pulse wider
        LCD_PRT &= ~ (1<<LCD_EN);                       //EN = 0 for H-to-L pulse
        LCD_PRT = (LCD_PRT & 0x0F) | (data <<4);        //Send low nibble to D4-D7
        LCD_PRT |= (1<<LCD_EN);                             //EN = 1 for H-to-L pulse
        delay_us(1);                                        //make EN pulse wider
        LCD_PRT &= ~ (1<<LCD_EN);                           //EN = 0 for H-to-L pulse
}

void lcd_init()
{
        LCD_DDR = 0x7F;

        LCD_PRT &=~(1<<LCD_EN);                             //LCD_EN = 0
        delay_us(2000);
        lcdCommand(0x33);                                   //send $33 for init.
        delay_us(100);
        lcdCommand(0x32);                                   //send $32 for init
        delay_us(100);
        lcdCommand(0x28);                                   //init. LCD 2 line,5*7 matrix
        delay_us(100);
        lcdCommand(0x0e);                                   //display on, cursor on
        delay_us(100);
        lcdCommand(0x01);                                   //clear LCD
        delay_us(2000);
        lcdCommand(0x06);                                   //Shift cursor right
        delay_us(100);
}

void lcd_gotoxy(unsigned char x, unsigned char y)           //function to goto x,y on LCD
{
        unsigned char firstCharAdr[]={0x80,0xC0,0x94,0xD4} ;
        lcdCommand(firstCharAdr[y-1] + x - 1);
        delay_us(100);
}

void lcd_print(char * str )                                 //function to print given str. on LCD
{
        unsigned char i = 0 ;
        while(str[i]!=0)
        {
                lcdData(str[i]);
                i++ ;
        }
}


//Main Code
int main(void)
{
        DDRA=0b00111111;                                    //LCD Interfacing
        PORTB2=1;                                           //pull up activated
        initI0();                                           //External Interrupt 0
        initT1();                                           //Timer/Counter1 to count time
```

```
    sei();                          //enable global interrupt
    while(1)
    {
        if(counter>=43200)          //No of days passed (30*60*24 = 43200 )
        {
            day++;                  //+day
            counter=counter-43200;  //counter remainder
        }
        if (day==30)
        {
            month[month_no]=((count/2)/30);  //Assigning monthly average data
            month_no++;                      //+month_no
            day=0;
        }
    }

    if ( PINA & 0b00000001)
    {
        //LCD code
        lcd_init();
        lcd_gotoxy(1,1);
        lcd_print(“%4d  %4d  %4d”,month[0], month[1], month[2]);
        lcd_gotoxy(1,2);
        lcd_print(“%4d  %4d  %4d”,month[3], month[4], month[5]);

    }

}
```