```
# define F_CPU 1000000UL
#define D4 eS_PORTD4
#define D5 eS_PORTD5
#define D6 eS_PORTD6
#define D7 eS_PORTD7
#define RS eS_PORTC6
#define EN eS_PORTC7


#include <avr/io.h>
#include <util/delay.h>
#include "lcd.h"

int i,val, d1,d2,fd1,fd2;
char str[4];
// adc funfion ************************************

void InitADC()
{
        ADMUX=(1<<REFS0);
        ADCSRA=(1<<ADEN)|(1<<ADPS2)|(1<<ADPS1)|(1<<ADPS0);
}

uint16_t ReadADC(uint8_t ch)
```

```
{
        //Select ADC Channel ch must be 0-7
        ch=ch&0b00000111;
        ADMUX&=0b11100000;
        ADMUX|=ch;

        //Start Single conversion
        ADCSRA|=(1<<ADSC);

        //Wait for conversion to complete
        while(!(ADCSRA & (1<<ADIF)));
        ADCSRA|=(1<<ADIF);
        return(ADC);
}


int main(void){

        DDRD = 0xFF;
        DDRC = 0xFF;
        DDRB = 0x00;
        DDRA= 0x00;  //for get cout
        PORTB = 0x00;
```

```
Lcd4_Init();
InitADC();
        int a, count1=0, count2=0, count3, temp;
        int b,c,d;

         Lcd4_Write_String("..Egg candler..");
         _delay_ms(500);
         Lcd4_Set_Cursor(2,0);
         Lcd4_Write_String("Process started");

         _delay_ms(1000);
         Lcd4_Clear();


        while(1){

                b= ReadADC(1);
                c= ReadADC(2);
                d= ReadADC(3);

                        if(b>500 ){
                        count1++;
                        //while(PINB & 0x01);
                        while (b>500 )
```

```
{
}

}

if(c>500 ){
        count2++;
        while (c>500 )
        {
        }

}
        //while(PINB & 0x02);

}


if(d>500  ){
        PORTC= (1<<PC0);
        _delay_ms(20000);
        temp=ReadADC(0);

        if (temp<400)
        {
```

```
                    PORTC= (1<<PC1);
                    _delay_ms(1500);
                }



            }else{
                PORTC= (0<<PC0);
                PORTC= (0<<PC1);
            }

Lcd4_Clear();

        count3=(count1-count2);

        itoa(count1,str,10);
        Lcd4_Write_String("input Egg= ");
        Lcd4_Write_String(str);

        Lcd4_Set_Cursor(2,0);
        itoa(count3,str,10);
        Lcd4_Write_String("out= ");
        Lcd4_Write_String(str);
        Lcd4_Write_String("  ");
```

```
itoa(count2,str,10);
Lcd4_Write_String("worst= ");
Lcd4_Write_String(str);

_delay_ms(200);



}
```