

# 2011

## Grafik LCD Katmanı



**ROBOMORE**

## Arduino Uyumlu Grafik LCD Katmanı



Şekil 1 Grafik LCD Katmanı

Bu LCD4884 katmanında bulunan grafik LCD'nin boyutu 48x48'dir. Eğer sizin projeniz daha fazla bilgiyi görüntülemenizi gerektiriyorsa, bu LCD katmanı tam olarak sizin ihtiyaçlarınızı karşılayabilir. İngilizce, Çince ve hatta şekil görüntülenebilmektedir. Ayrıca 5 DOF joystick ile entegrasyon yapabilir. Bu grafik ekranı üzerinde 6 Dijital IO ve 5 Analog IO bulunmaktadır. Grafik LCD4884 katmanın güç kaynağı 5V olmakla birlikte üzerinde bir adet reset tuşu ile birlikte 5 derecelik bir adet Arduino Analog Pin 0 kullanan joystick bulunmaktadır. Arduino uyumlu grafik LCD katmanı Şekil 1'de gösterilmiştir.

### Teknik Özellikleri

- Besleme Voltajı : 5V
- Spı Arayüzü (Arduino Dijital Pin 2, 3, 4, 5 ve 6'yı kullanarak)
- Reset Tuşu
- 5 Derecelik Joystick
- Arka Işık Kontrolü

## Grafik LCD Kullanımı

```
#include "LCD4884.h"
#include "DFrobot_bmp.h"
#include "DFrobot_chinese.h"

//tuştakımı debounce parametreleri
#define DEBOUNCE_MAX 15
#define DEBOUNCE_ON 10
#define DEBOUNCE_OFF 3

#define NUM_KEYS 5

#define NUM_MENU_ITEM 4

// joystick numaraları
#define LEFT_KEY 0
#define CENTER_KEY 1
#define DOWN_KEY 2
#define RIGHT_KEY 3
#define UP_KEY 4

// menu başlangıç noktaları

#define MENU_X 10 // 0-83
#define MENU_Y 1 // 0-5

int adc_key_val[5]={50, 200, 400, 600, 800 };

// debounce sayıcıları
byte button_count[NUM_KEYS];
// tuş durumu - basıldı/bırakıldı
byte button_status[NUM_KEYS];
// tuş flagları
byte button_flag[NUM_KEYS];

// menu tanımı
char menu_items[NUM_MENU_ITEM][12]={
    "TEMPERATURE",
    "CHAR MAP",
    "BITMAP",
    "ABOUT"
};

void (*menu_funcs[NUM_MENU_ITEM])(void) = {
    temperature,
    charmap,
    bitmap,
    about
}
```

```
};

char current_menu_item;

void setup()
{
    // reset tuşu arrayleri
    for(byte i=0; i<NUM_KEYS; i++){
        button_count[i]=0;
        button_status[i]=0;
        button_flag[i]=0;
    }

    // timer2 -- Prescaler/256
    TCCR2A &= ~(1<<WGM21) | (1<<WGM20);
    TCCR2B &= ~(1<<WGM22);
    TCCR2B = (1<<CS22)|(1<<CS21);

    ASSR |= (0<<AS2);

    // Normal mod kullanımı
    TCCR2A =0;
    //Timer2 Overflow Interrupt
    TIMSK2 |= (0<<OCIE2A);
    TCNT2=0x6; // 6 dan saymaya başla;
    TIMSK2 = (1<<TOIE2);

    SREG|=1<<SREG_I;

    lcd.LCD_init();
    lcd.LCD_clear();

    //menu sıfırlama
    init_MENU();
    current_menu_item = 0;
}

/* döngü*/

void loop()
{
    byte i;
    for(i=0; i<NUM_KEYS; i++){
        if(button_flag[i] !=0){

            button_flag[i]=0; // reset tuşu flagi
```

```

switch(i){

    case UP_KEY:
        lcd.LCD_write_string(MENU_X, MENU_Y + current_menu_item,
menu_items[current_menu_item], MENU_NORMAL );
        current_menu_item -=1;
        if(current_menu_item <0) current_menu_item = NUM_MENU_ITEM -1;
        lcd.LCD_write_string(MENU_X, MENU_Y + current_menu_item,
menu_items[current_menu_item], MENU_HIGHLIGHT );
        break;
    case DOWN_KEY:
        lcd.LCD_write_string(MENU_X, MENU_Y + current_menu_item,
menu_items[current_menu_item], MENU_NORMAL );
        current_menu_item +=1;
        if(current_menu_item >(NUM_MENU_ITEM-1)) current_menu_item = 0;
        lcd.LCD_write_string(MENU_X, MENU_Y + current_menu_item,
menu_items[current_menu_item], MENU_HIGHLIGHT );
        break;
    case LEFT_KEY:
        init_MENU();
        current_menu_item = 0;
        break;
    case CENTER_KEY:
        lcd.LCD_clear();
        (*menu_funcs[current_menu_item])();
        lcd.LCD_clear();
        init_MENU();
        current_menu_item = 0;
        break;
}

}

}

}

/* menu fonksiyonları */

void init_MENU(void){

    byte i;

    lcd.LCD_clear();

    lcd.LCD_write_string(MENU_X, MENU_Y, menu_items[0], MENU_HIGHLIGHT );

    for (i=1; i<NUM_MENU_ITEM; i++){
        lcd.LCD_write_string(MENU_X, MENU_Y+i, menu_items[i], MENU_NORMAL);
    }

}

```

```

// orta tuşun basılmasını bekle
void waitfor_OKkey(){
    byte i;
    byte key = 0xFF;
    while (key!= CENTER_KEY){
        for(i=0; i<NUM_KEYS; i++){
            if(button_flag[i] !=0){
                button_flag[i]=0; // reset tuşu flagi
                if(i== CENTER_KEY) key=CENTER_KEY;
            }
        }
    }
}

void temperature()
{
    lcd.LCD_write_string_big(10, 1, "+12.30", MENU_NORMAL);
    lcd.LCD_write_string(78, 2, "C", MENU_NORMAL);
    lcd.LCD_write_string(38, 5, "OK", MENU_HIGHLIGHT );
    waitfor_OKkey();
}

void charmap(){
    char i,j;
    for(i=0; i<5; i++){
        for(j=0; j<14; j++){
            lcd.LCD_set_XY(j*6,i);
            lcd.LCD_write_char(i*14+j+32, MENU_NORMAL);
        }
    }

    lcd.LCD_write_string(38, 5, "OK", MENU_HIGHLIGHT );
    waitfor_OKkey();
}

void bitmap(){
    lcd.LCD_draw_bmp_pixel(0,0, DFrobot_bmp, 84,24);
    lcd.LCD_write_chinese(6,3, DFrobot_chinese,12,6,0,0);
    lcd.LCD_write_string(38, 5, "OK", MENU_HIGHLIGHT );
    waitfor_OKkey();
}

void about(){

    lcd.LCD_write_string( 0, 1, "LCD4884 Shield", MENU_NORMAL);
    lcd.LCD_write_string( 0, 3, "www.DFrobot.cn", MENU_NORMAL);
    lcd.LCD_write_string(38, 5, "OK", MENU_HIGHLIGHT );
}

```

```
waitfor_OKkey();

}

// ADC degerini tuş numarasına donustur
char get_key(unsigned int input)
{
    char k;

    for (k = 0; k < NUM_KEYS; k++)
    {
        if (input < adc_key_val[k])
        {
            return k;
        }
    }

    if (k >= NUM_KEYS)
        k = -1; // tusa basilmadi

    return k;
}

void update_adc_key(){
    int adc_key_in;
    char key_in;
    byte i;

    adc_key_in = analogRead(0);
    key_in = get_key(adc_key_in);
    for(i=0; i<NUM_KEYS; i++)
    {
        if(key_in==i) //1 tusa basildi
        {
            if(button_count[i]<DEBOUNCE_MAX)
            {
                button_count[i]++;
                if(button_count[i]>DEBOUNCE_ON)
                {
                    if(button_status[i] == 0)
                    {
                        button_flag[i] = 1;
                        button_status[i] = 1;
                    }
                }
            }
        }
    }
}
```

```
}
else
{
    if (button_count[i] > 0)
    {
        button_flag[i] = 0;
        button_count[i]--;
        if (button_count[i] < DEBOUNCE_OFF){
            button_status[i] = 0;
        }
    }
}
}
}

ISR(TIMER2_OVF_vect) {
    TCNT2 = 6;
    update_adc_key();
}
```