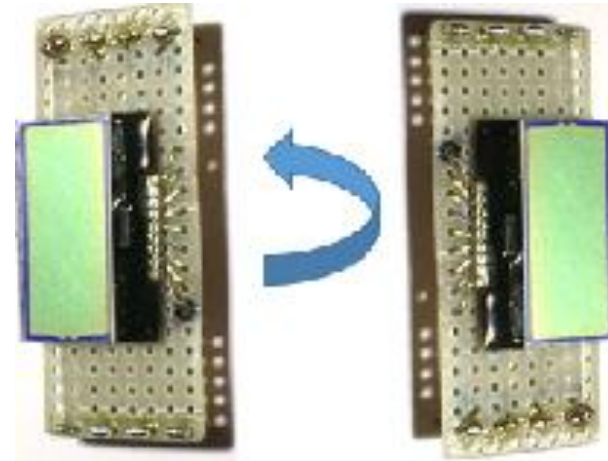
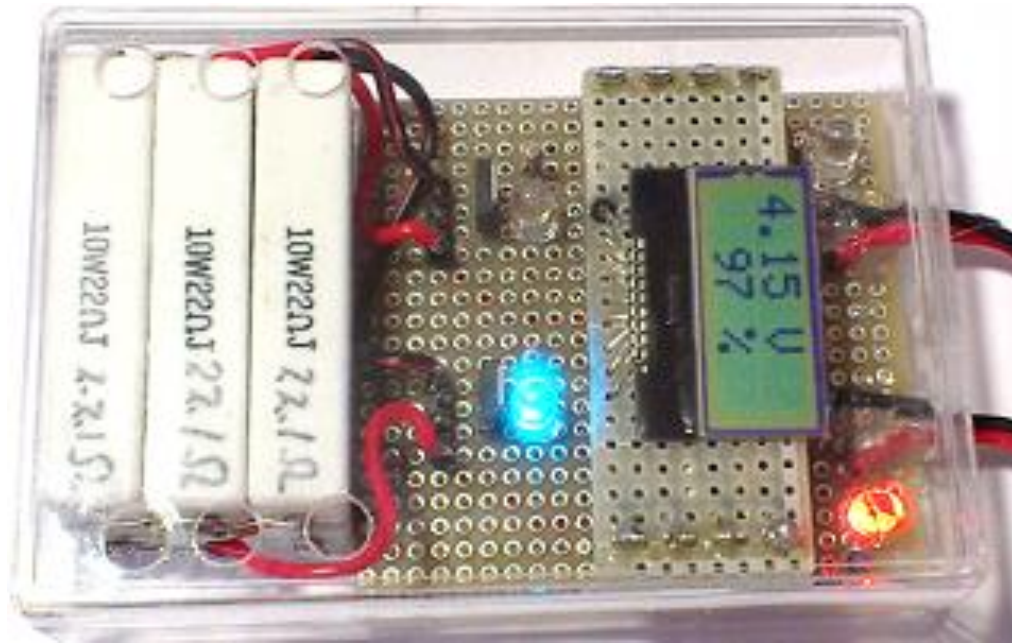


Lipo 1&2 cell Discharger with LCD AQM0802 V2

2015.11.23



LED動作 青色LED

点灯: 放電中

消灯: 放電終了(約50%)

LCD動作 1cellのみ

オープニングメッセージ

電圧表示 : 10mV 分解能

試作品仕様

- ・ 放電用抵抗
 - 1cell: 20Ω, 10W, 2個
 - 2cell: 20Ω, 10W, 4個
- ・ 電流断続用制御素子:
 - 60V,5A FET 2SK4017(Q)
- ・ 放電電圧: 約4.2~3.98V **1cell**
- ・ 放電電圧: 約8.4~7.96V **2cell**
- ・ 放電電流: 約400mA **1,2cell**
- ・ 動作電圧: 3.6~20V

設計仕様

- ・ 負荷は純抵抗:
 - 抗実質抵抗損失は定格の1/10以下
- ・ 残量, 約50%*****で自動停止
- ・ LCDによる電圧, 電気残量表示
- ・ LEDによる放電状態通知
- ・ 単純な構成と余裕の安全設計
 - 1,2cell完全独立回路,

メモ:

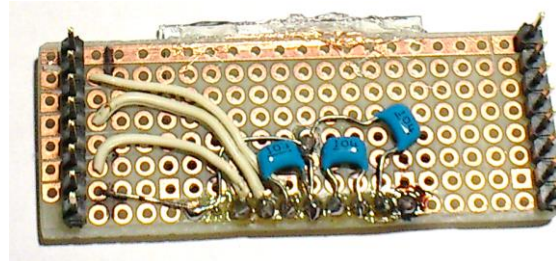
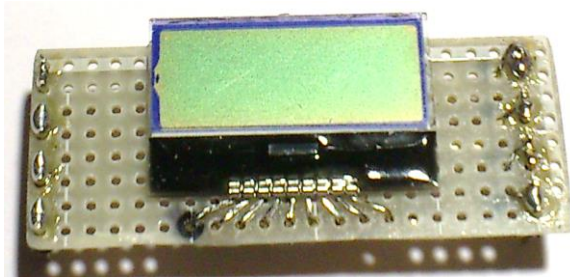
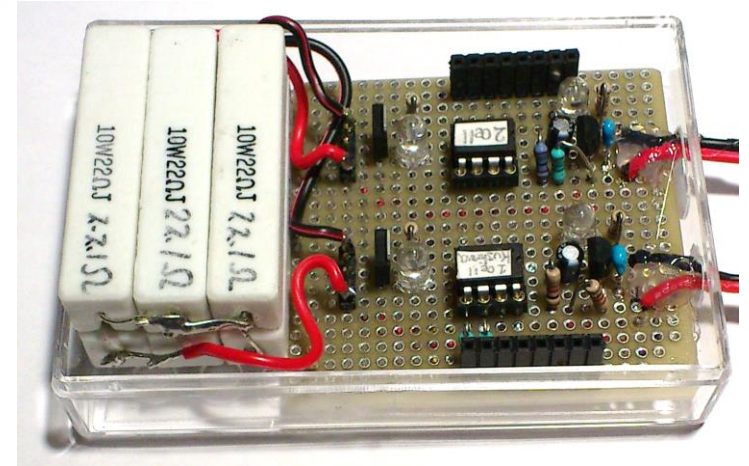
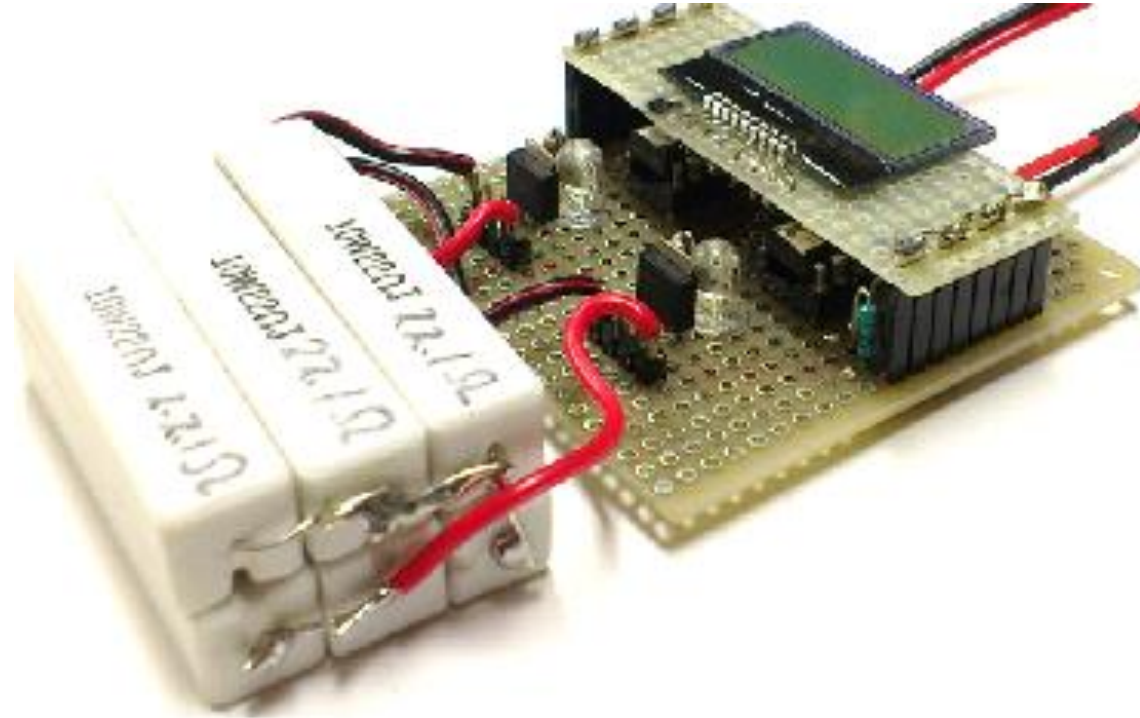
※1cell, 2cell 同時使用可
(Concurrent Use)

※ 50% ⇒ 3.85V

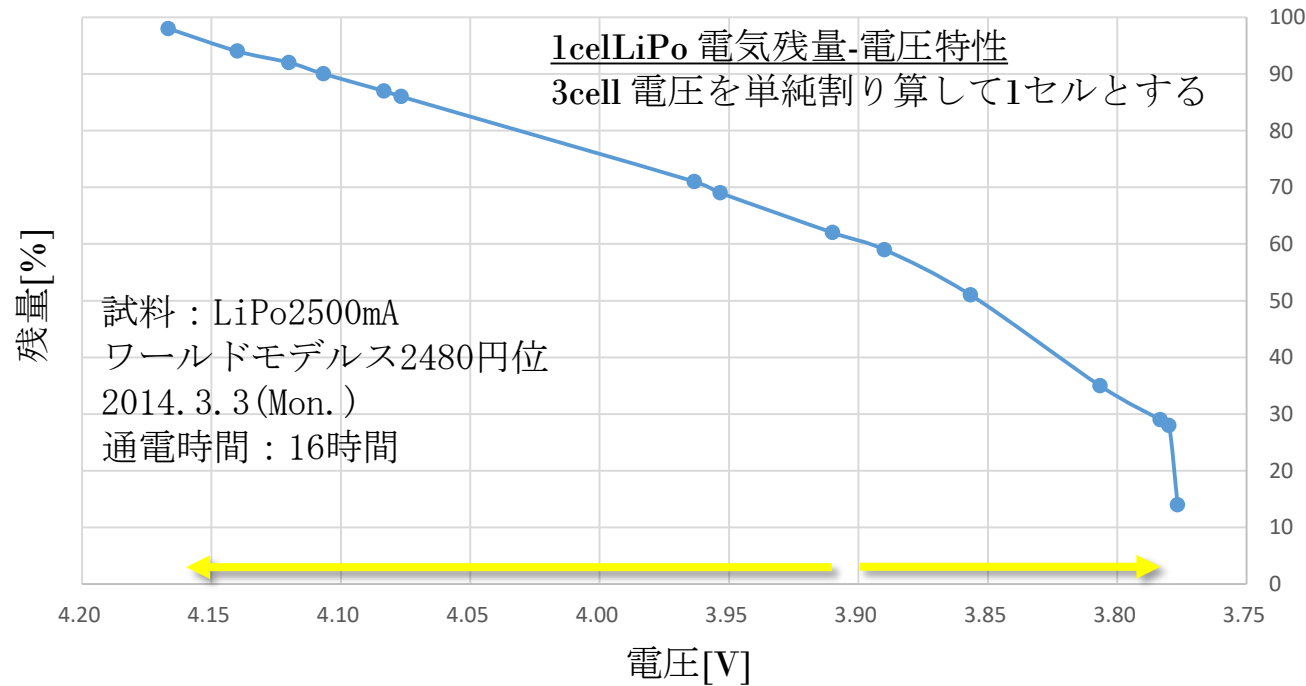
※ LCD 180度回転可

Lipo 1&2 cell Discharger with LCD AQM0802 V2

2015.11.23



Lipo 1&2 cell Discharger with LCD AQM0802



実測による放電特性を元に、
2つに分けて直線近似を行い、
電気残量を計算する。

電圧値の較正器：

Agilent 3458A 8 ½ 桁
Digital Multi Meter

負荷抵抗：10Ω, 10W, セメント

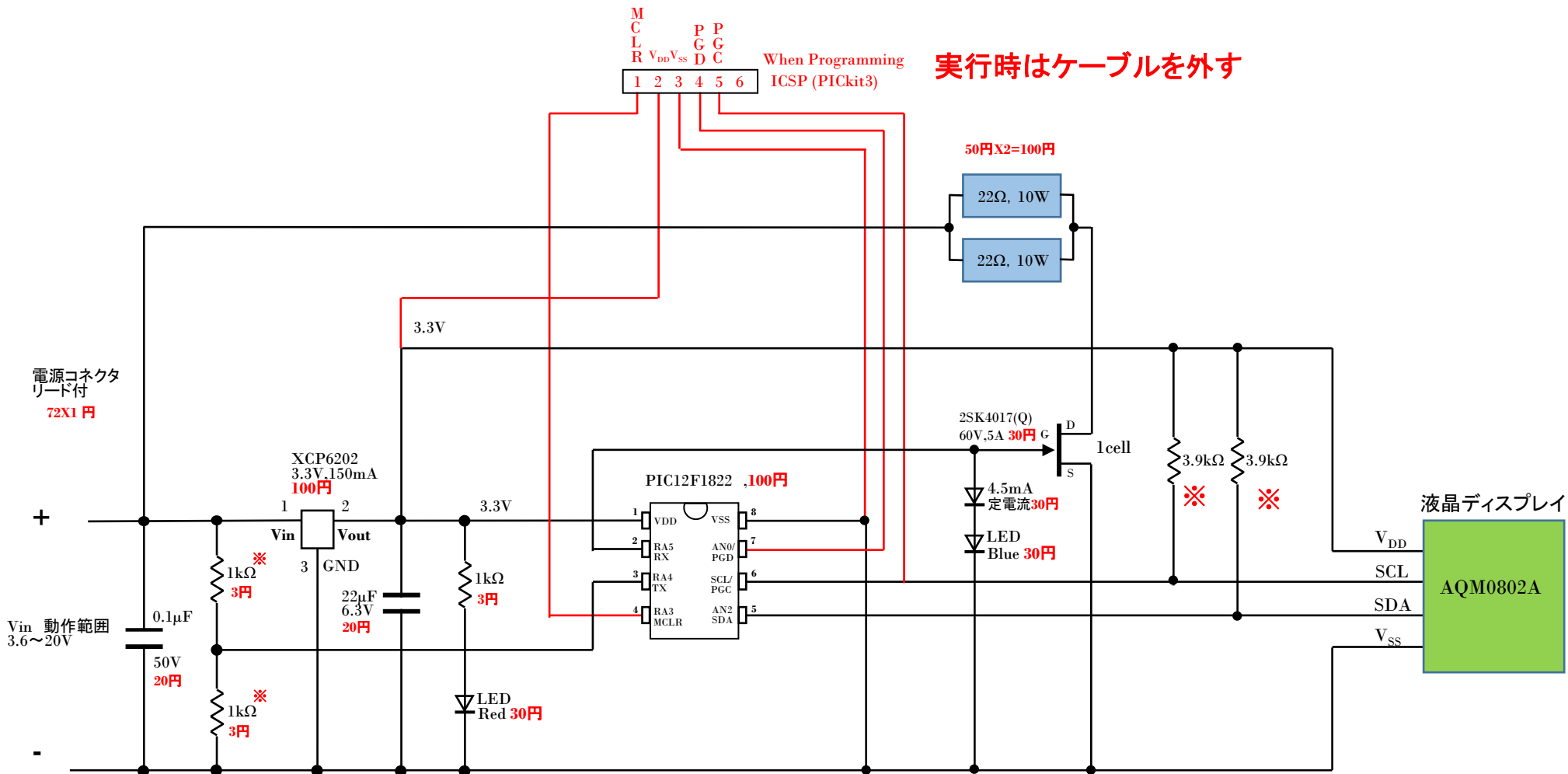
電気残量測定：

バッテリーチェッカー
Hyperion EOS SENTRY2



カットオフ電圧は電気残量
チェッカーの精度に支配される。

Lipo 1 cell Discharger with LCD AQM0802



※ 較正済金属被膜抵抗

I2Cのstart()が、応答しない時、ハンゲアップとなる。プルアップ抵抗が確実に入っているか確認のこと。

Lipo 1 cell Discharger with LCD AQM0802

```
// PIC 12F1822 2015.11.24(Mon.)
// LiPo 1 cell Discharge use 22 ohm 10 Watt resistor X6
// with LCD AQM0802

#include<12F1822.h>
//#device ADC=10 // 10 bit Left
//#include<math.h>
#fuses INTRC_IO,NOWDT,NOPROTECT,NOMCLR,BROWNOUT
#use delay(clock=16000000)
#use i2c(MASTER,SDA=PIN_A2,SCL=PIN_A1,FAST,NOFORCE_SW)

// AQM0802A
#define LCD_ADD_AQM 0x7C // AQM0802 Slave Address
// AQM W/R Mode val
#define LCD_CMD_AQM 0x80 // Instruction Write Mode
#define LCD_DAT_AQM 0xC0 // Data Write Mode
#define line_1_AQM 0x00 // first line
#define line_2_AQM 0xC0 // second line = 0x80 + 0x40

void LCD_com_AQM(unsigned char cmd);
void LCD_Clear_AQM();
void LCD_Setline_AQM(unsigned char line);
void LCD_init_AQM();
void LCD_data_AQM(unsigned char data);
void LCD_space_AQM(int8 n);

void disp_val(int16 x);
static int8 d1,d2,d3,d4,d5;

#define AD_PIN 3 // No 3
#define SW_PIN PIN_A5 // No 2
#define CutoffV 3870 // About 50% Remain
#define Keisuu 26.3
//
static long ADC2 = 0;
//
//void read_adc_ch2();
void LED_blink(int8);

////////////////////
void main(){
  char cnt;
  char open1[8] = "1c. Dis.";
  char open2[8] = "2015.11.";
  char open3[8] = "I am ";
  char open4[8] = "M Taka-.";

  int16 pcnt;

  delay_ms(1000);
  LCD_init_AQM();
}
```

```
LCD_Setline_AQM(line_1_AQM);
for(cnt=0 ; cnt < 8; cnt++){
  LCD_data_AQM( open1[cnt]);
}
LCD_Setline_AQM(line_2_AQM);
for(cnt=0 ; cnt < 8; cnt++){
  LCD_data_AQM( open2[cnt]);
}
delay_ms(1000);

LCD_Clear_AQM();
LCD_Setline_AQM(line_1_AQM);
for(cnt=0 ; cnt < 8; cnt++){
  LCD_data_AQM( open3[cnt]);
}
LCD_Setline_AQM(line_2_AQM);
for(cnt=0 ; cnt < 8; cnt++){
  LCD_data_AQM( open4[cnt]);
}
delay_ms(1000);

setup_adc_ports(0x80); // All ALALOG
setup_adc(ADC_CLOCK_INTERNAL);
set_adc_channel(AD_PIN);
delay_us(200);

while(1){
  ADC2 = Keisuu*read_adc(ADC_START_AND_READ);
  LCD_Clear_AQM();
  LCD_Setline_AQM(line_1_AQM);
  disp_val(ADC2);

  pcnt = 0;
  if( ADC2 > 4220 ){
    pcnt = 99;
  }
  else{
    if( ADC2 >= 3900 ){
      pcnt = ADC2*0.14 - 484;
    }
    if( ADC2 < 3900 ){
      //if( ADC < 3775){
      if( ADC2 < 3000){
        pcnt = 0;
      }
      else{
        pcnt = ADC2*0.233 - 852;
      }
    }
  }
}
```

```
d5 = pcnt % 10 + 0x30;
pcnt /= 10;
d4 = pcnt % 10 + 0x30;
pcnt /= 10;
d3 = pcnt % 10 + 0x30;
pcnt /= 10;
d2 = pcnt % 10 + 0x30;
pcnt /= 10;
d1 = pcnt % 10 + 0x30;

LCD_Setline_AQM(line_2_AQM);
LCD_data_AQM(" "); LCD_data_AQM(" ");
LCD_data_AQM(d4);
LCD_data_AQM(d5);
LCD_data_AQM(' ');
LCD_data_AQM('%');

if( ADC2 < CutoffV ){
  output_low(SW_PIN); // PIN 2 Load Resistor OFF
}
else{
  output_high(SW_PIN); // PIN 2 Load Resistor ON
}
set_adc_channel(AD_PIN);
delay_ms(400);
}

void LED_blink(int8 cnt){
  while(cnt-- != 0){
    output_high(SW_PIN); // PIN 2 Load Resistor ON
    delay_ms(500);
    output_low(SW_PIN); // PIN 2 Load Resistor OFF
    delay_ms(500);
  }
}

void disp_val(int16 x){ // Disp Digits
  int16 sw;

  d5 = x % 10 + 0x30;
  x /= 10;
  d4 = x % 10 + 0x30;
  x /= 10;
  d3 = x % 10 + 0x30;
  x /= 10;
  d2 = x % 10 + 0x30;
  x /= 10;
  d1 = x % 10 + 0x30;
```

```

sw = 0;
if(d1 == 0x30){
    d1 = 0x20;
}
else{
    sw = 1;
}
if( (d2 == 0x30)&&(sw == 0)){
    d2 = 0x20;
}
else{
    sw = 1;
}
if((d3 == 0x30)&&(sw == 0)){
    d3 = 0x20;
}
else{
    sw = 1;
}
if((d4 == 0x30)&&(sw == 0)){
    d4 = 0x20;
}
}
//printf("%1c%1c.%1c%1c%1c",d1,d2,d3,d4,d5);
//LCD_data_AQM(d1);
LCD_data_AQM(d2);
LCD_data_AQM('.');
LCD_data_AQM(d3);
LCD_data_AQM(d4);
LCD_data_AQM(' ');
LCD_data_AQM('V');
//LCD_data_AQM(d5);
}

// AQM0802 //
void LCD_com_AQM(unsigned char cmd){
    int16 u_t=10;
    i2c_start();    delay_ms(1);
    i2c_write(LCD_ADD_AQM); delay_us(u_t);    // Slave Address
    i2c_write(LCD_CMD_AQM);delay_us(u_t);    // Command Mode
    i2c_write(cmd);    delay_us(u_t);    // Send Command
    i2c_stop();    delay_ms(1);
}

void LCD_Clear_AQM(){
    LCD_com_AQM(0x01);
    delay_ms(20);
}

```

```

void LCD_Setline_AQM(unsigned char line) {
    LCD_com_AQM(line);
}

void LCD_init_AQM(){
    delay_ms(60);
    LCD_com_AQM(0x38);    // Function set
    delay_us(30);
    LCD_com_AQM(0x39);    // Function set
    delay_us(30);
    LCD_com_AQM(0x14);    // Internal OSC frequency
    delay_us(30);
    LCD_com_AQM(0x70);    // Contrast
    delay_us(30);
    LCD_com_AQM(0x56);    // Power/ICON/Contrast Control
    delay_us(30);
    LCD_com_AQM(0x6c);    // Follow Control
    delay_ms(300);
    LCD_com_AQM(0x38);    // Function set
    delay_us(30);
    LCD_com_AQM(0x0c);    // Display ON/OFF Control
    delay_us(30);
    LCD_com_AQM(0x01);    // Clear Display
    delay_ms(2);
}

void LCD_data_AQM(unsigned char data){
    int16 u_t=5;

    i2c_start();    delay_ms(1);
    i2c_write(LCD_ADD_AQM); delay_us(u_t);    // Slave Address
    i2c_write(LCD_DAT_AQM);delay_us(u_t);    // Data Out Code
    i2c_write(data);    delay_us(u_t);    // Send Command
    i2c_stop();    delay_ms(1);
}

void LCD_space_AQM(int8 n){
    int8 cnt;
    cnt = 0;
    while(cnt++ < n){
        LCD_data_AQM(' ');
    }
}

```