

```

// LC meter Arduino 2020.07.09, 20200716 f1 ok, 20200717 f2 ok
// 20200718 xxxxxxxpF xxxxxuH OK
// 20200718 if (test_value<1.000E+2) xxxxxnH OK
// 2020.06.25 Frequency counter 次のライブラリーを参照しました。ご協力感謝します。
// サンプルコード (FreqCount_2017-0207-001-ok)
// "http://interface.khm.de/index.
php/lab/interfaces-advanced/arduino-frequency-counter-library/"
// Using Counter1 for counting Frequency on T1 / PD5 / digitalPin 5
// Using Timer2 for Gatetime generation

// D5:入力、D6:標準C駆動リレー、D7:L/Cスイッチ PINMODE設定
// LCD関数の呼び方変更 i2cLCD

// Based on PIC1827_main_c
&

/*****
LC METER by PIC
1602 display by 4 bit with PIC16F1827
By nobcha all right reserved

Ver 1.0 09/29/2010 for PIC16F88
Ver 2.0 for PIC16F428
Ver 3.0 02/10/2012 for PIC16F1827
Hitech C & MPLAB PIC16F1827 + LCD
MPLAB IDE V8.73a HiTECH C V9.83
*****/

#include <FreqCounter.h>
#include <Wire.h>
#include <skI2CLCDlib.h>

int PIN_Led = 13; // Gate点滅LEDのポート
boolean LED_Stat = 1; // Gate点滅LEDの状態
int SEL_SW = 7; // L/C select SW
boolean SEL_SW_Stat = 1;
int CAL_ON = 6; // Caliblation relay
int ON = 1;
int OFF = 0;

//lcd instance
skI2CLCDlib LCD(0x3E, 16); // LCDのi2cアドレス、画面カラム数16文字

// initialize Serial, i2cLCD,
void setup() {
  Serial.begin(9600); // connect to the serial port
  Serial.println("LCM on i2cLCD v1.2");

```

```

pinMode(PIN_Led, OUTPUT);
pinMode(SEL_SW, INPUT);
pinMode(CAL_ON, OUTPUT);

// LCDモジュールの初期化処理
// ICON OFF,コントラスト(0-63),VDD=5Vで使う
LCD.Init(LCD_NOT_ICON,32,LCD_VDD5V) ; //5Vへ変更20200612
// 書き込む
LCD.SetCursor(0,0) ; // 表示位置を1行目1列[00H]に設定
LCD.Puts("LCM METER ") ; // [00H]から書込まれる(表示する)
// Ask cal SW
LCD.SetCursor(0,1) ; // 表示位置を2行目1列[40H]に設定
LCD.Puts(" i2cLCD v1.2 ") ; // [40H]から書込まれる(表示する)
delay(20);
}

volatile unsigned long freq_d;
volatile unsigned long freq;
volatile boolean err;
char charbuf[16]; // sprintf関数変換時に使う文字配列
volatile unsigned char i, l_power, l_digi, l_unit, c_power, c_digi, c_unit;
volatile unsigned long freq1, freq2, freq3;
volatile float c_int, l_int, l_inv, f_sq;
volatile float c_cal=1.000e+3;
volatile float test_value;

volatile unsigned long freq_count()
{
  FreqCounter::f_comp = 8; // Set compensation to 12
  FreqCounter::start(1000); // Start counting with gatetime of 100ms
  while (FreqCounter::f_ready == 0); // wait until counter ready
  freq_d = FreqCounter::f_freq; // read result
// Serial.println(freq_d); // 20200716 no print freq_d

  if(freq_d<10000) err=1; // ??
  return freq_d; // ??
}

float freq_cal(unsigned long f1, unsigned long f2){ // Coefficient
calculating
return((float)(f1)/(float)(f2)*(float)(f1)/(float)(f2)-1);
}

void loop()
{
// START SWITCH CHECK
// If calibrlation

```

```

Serial.println("WAIT SEL SW ON");
while( digitalRead(SEL_SW) == 0 ){ } // CAL switch check
delay(20);
while( digitalRead(SEL_SW) == 0 ){ } // Check again
delay(20);

/* Get frequency 1 and display */
Serial.println("SEL SW ON");
freq1=freq_count(); // F1 get
freq1=freq_count(); // F1 get again
Serial.println("F1 gotten");
Serial.println(freq1);

LCD.SetCursor(3,0); // 表示位置を1行目4列に設定
LCD.Puts(" f1=");
sprintf(charbuf,"%ld", freq1, 7);
LCD.Puts(charbuf);
LCD.Puts("Hz "); //

// Relay on f2 getting
digitalWrite(CAL_ON, ON); // Calibration capasitor on
delay(100); // Wait 100ms
freq2=freq_count(); // Get F2
freq2=freq_count(); // Get F2
digitalWrite(CAL_ON, OFF); // Calibration capasitor off

Serial.println("F2 gotten");
Serial.println(freq2);

/* F2 displaying */
LCD.SetCursor(0,1); // Move cursur to the top of 2nd line
LCD.Puts("f2=");
sprintf(charbuf,"%ld", freq2, 7);
LCD.Puts(charbuf);
LCD.Puts("Hz "); //
delay(1000);

/* Calculating C and L */
c_int=(float)(c_cal)/freq_cal(freq1,freq2); // Calcurate c_int from F
and F2 pF
f_sq=(float)(freq1)*(float)(freq1)/+1.000E+2; // (F1*F1)

Serial.println("freq_cal= ");
Serial.println(freq_cal(freq1,freq2));

l_inv=(+3.9438E+1)*f_sq*c_int; //
l_int=(+1.000E+16)/l_inv; // uH

/* for debugging */
Serial.println("C/L unit gotten");

```

```

Serial.println("C=");
Serial.println(c_int);
Serial.println("L=");
Serial.println(l_int);

LCD.SetCursor(0,1);          // Move cursur to the top of 2nd line
LCD.Puts("C=");
sprintf(charbuf,"%d", (int)c_int);
LCD.Puts(charbuf);
LCD.Puts("pF ");           //
LCD.Puts("L=");
sprintf(charbuf,"%d", (int)l_int);
LCD.Puts(charbuf);
LCD.Puts("uH ");           //

// Calibration finished

Serial.println("Calibration end ");
LCD.SetCursor(3,0) ;        // 表示位置を1行目4列に設定
LCD.Puts("Calibrated ");
delay(1000);

// Measurement starting check
while( digitalRead(SEL_SW) == 1 ){ } // Wait sw status changed

while (1) // Testing forever
{
digitalWrite(PIN_Led, LED_Stat=!LED_Stat); // LED on/off
delay(500);

LCD.SetCursor(4,0);        // Move cursur to the 8th of 1st line
LCD.Puts("SET TEST "); // Ask setting L or C
delay(100);

freq3=freq_count();        // Get F3
freq3=freq_count();        // Get F3

Serial.println("F3 gotten");
Serial.println(freq3);
LCD.SetCursor(3,0);        // Move cursur to the top of 2nd line
LCD.Puts(" f3=");
sprintf(charbuf,"%ld", freq3, 7);
LCD.Puts(charbuf);
LCD.Puts("Hz");           //
delay(200);

/* displaying value */
LCD.SetCursor(0,1);        // Move cursur to the top of 2nd line

```

```

/* Metric unit changing further study */
/* uH/mH l_power=1; mH, l_power=1000; uH,
   pF/uF c_power=1; pF, c_power=1000; nF,
*/

l_power=1;
l_digi=1;
l_unit='u';
c_power=1;
c_digi=0;
c_unit='p';

/* Calculating value whether C position or L */
if(digitalRead(SEL_SW) == 0) // If SW==0, start L measuring
{
test_value= ((freq_cal(freq1,freq3))*l_int);

int f_rate = (freq_cal(freq1,freq3)*1000);
Serial.println("f_rate");
Serial.println(f_rate);

LCD.Puts("L=");
sprintf(charbuf, "%d", (long) test_value );
LCD.Puts(charbuf);
LCD.Puts("uH ");

if (test_value<1.000E+2)
{
LCD.SetCursor(0,1); // Move cursur to the top of 2nd
line
LCD.Puts("L=");
sprintf(charbuf, "%d", (long) (test_value*1.0E+3 ));
LCD.Puts(charbuf);
LCD.Puts("nH ");
}

Serial.println("L=");
Serial.println( test_value);
Serial.println("uH");
}
else
{
test_value=((freq_cal(freq1,freq3))*c_int/c_power);

Serial.println("C=");
Serial.println( test_value);
Serial.println("pF");
LCD.Puts("C=");
sprintf(charbuf, "%u", (long) (test_value ));
}

```

```
LCD.Puts(charbuf);
```

```
LCD.Puts("pF      ");
```

```
}
```

```
delay(2000);
```

```
}
```

```
}
```