

ミニ温度計(7セグ表示)

概要

今迄に、温度に関する物を、多数製作してきました。

- 温度データロガー...☐S8100B☐-40°C~+100°C☐シリアルデータ出力
- 温度データロガーV2...☐LM60☐-25°C~+99°C☐EEPROM出力
- 温度データロガーV3...☐LM60☐-25°C~+99°C☐EEPROM出力
- 温度計(温度制御機能付き)...☐LM35DZ☐0°C~+100°C☐リレー出力☐LCD表示

今回は、出来るだけコンパクトを目指して、製作してみました。

- 温度センサ☐:LM61CIZ
- 測定範囲:-30 ~+100
- 温度係数:+10mV/°C
- 結果表示:7セグ(3桁)、小数点第一位まで表示(例 “ 12.3 “)
- 測定精度:2.44mV(0.41°C)
- 電源:単三電池2本(3V)

動作原理

ダイナミック点灯の基本的な原理は、ミニ周波数カウンタ(kHz表示)を参照してください。電源電圧の設定については、ミニ電圧&電流計(7セグ表示)を参照してください。

<温度の測定>

1. 温度センサーの出力電圧を、A/D変換で1000回測定し、平均電圧(V1)を求める。
2. 電圧(V1)データを、温度データに変換する。
$$\text{温度} = ((V1 - 300\text{mV}) \div 10\text{mV}) - 30^{\circ}\text{C}$$
3. 温度データが、マイナスであれば☐LEDを点灯させる。
4. 温度データを、7セグ(3桁)に表示する。
5. 1.へ戻る。

回路図


```
#define DATA1 0b00011000
#define DATA2 0b11010011
#define DATA3 0b11011010
#define DATA4 0b10111000
#define DATA5 0b11101010
#define DATA6 0b11101011
#define DATA7 0b01111000
#define DATA8 0b11111011
#define DATA9 0b11111010
#define DATA_SPACE 0b00000000

#define SPACE 10

#define NON_SEG 0b00000000
#define SEG1 PORTA.F0
#define SEG2 PORTA.F7
#define SEG3 PORTA.F6

#define ON 0
#define OFF 1

#define LED PORTA.F1

//*****
*

short seg_flg, data1, data2, data3, dot;
short tbl[11] = {DATA0, DATA1, DATA2, DATA3, DATA4, DATA5, DATA6,
DATA7, DATA8, DATA9, DATA_SPACE};

void interrupt()
{
    if (PIR1.CCP1IF == 1) {
        PIR1.CCP1IF = 0;
        //□□□□□桁)点灯処理
        switch (seg_flg) {
            case 0:
                seg_flg = 1;
                SEG3 = OFF;
                PORTB = (dot == 1) ? tbl[data1] | 0b00000100 : tbl[data1];
                SEG1 = ON;
                break;
            case 1:
                seg_flg = 2;
                SEG1 = OFF;
                PORTB = (dot == 2) ? tbl[data2] | 0b00000100 : tbl[data2];
                SEG2 = ON;
                break;
            case 2:
                seg_flg = 0;
                SEG2 = OFF;
```

```

        PORTB = (dot == 3) ? tbl[data3] | 0b00000100 : tbl[data3];
        SEG3 = ON;
        break;
    }
}

//*****
*

unsigned long measurement(unsigned short channel)
{
    static unsigned long dat;
    static unsigned int cnt;
    //
    dat = 0;
    for (cnt = 0; cnt < 1000; cnt++) {
        dat += Adc_Read(channel);
    }
    return (dat);
}

//*****
*

void main()
{
    static char buf[16];
    static double dat;
    //
    TRISA = 0b00111100;
    TRISB = 0b00000000;
    OSCCON = 0b01100000; // クロックを4Mhzに設定する。
    ANSEL = 0b00000100; // □□変換を使用する。
    ADCON1.VCFG1 = 1;
    ADCON1.VCFG0 = 0;
    // TIMER0の設定
    OPTION_REG.T0CS = 1;
    OPTION_REG.PSA = 0;
    OPTION_REG.PS2 = 0;
    OPTION_REG.PS1 = 1;
    OPTION_REG.PS0 = 0;
    // TIMER1の設定
    PIE1.TMR1IE = 0;
    PIR1.TMR1IF = 0;
    T1CON.T1CKPS0 = 1;
    T1CON.T1CKPS1 = 1;
    T1CON.TMR1ON = 0;
    TMR1L = 0;
    TMR1H = 0;
    // CCPの設定

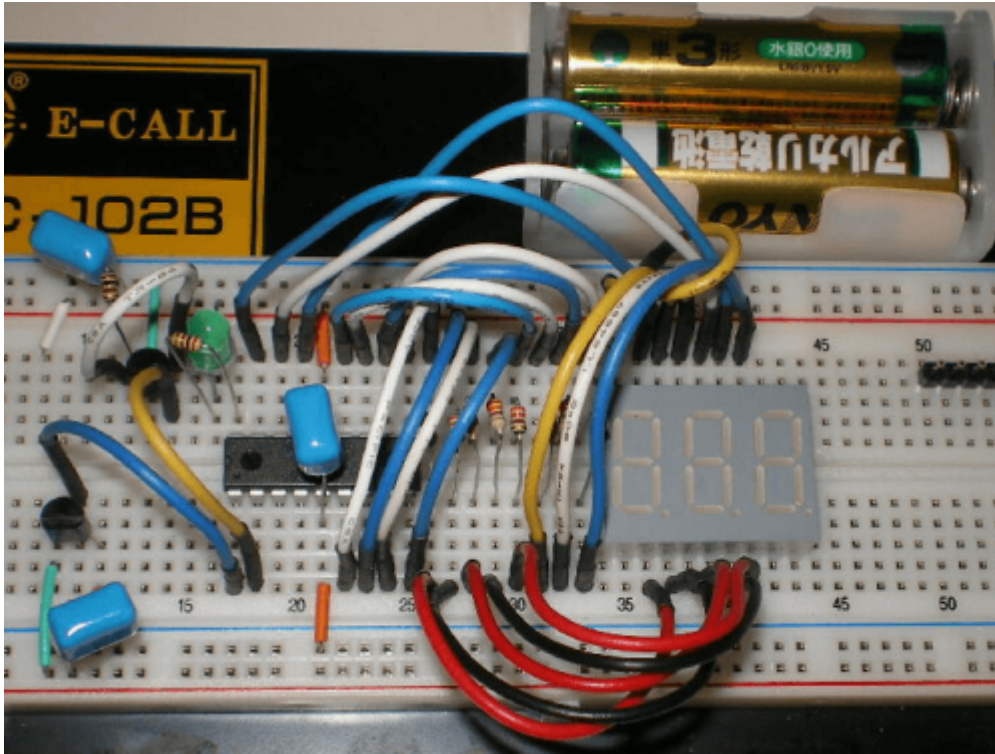
```

```

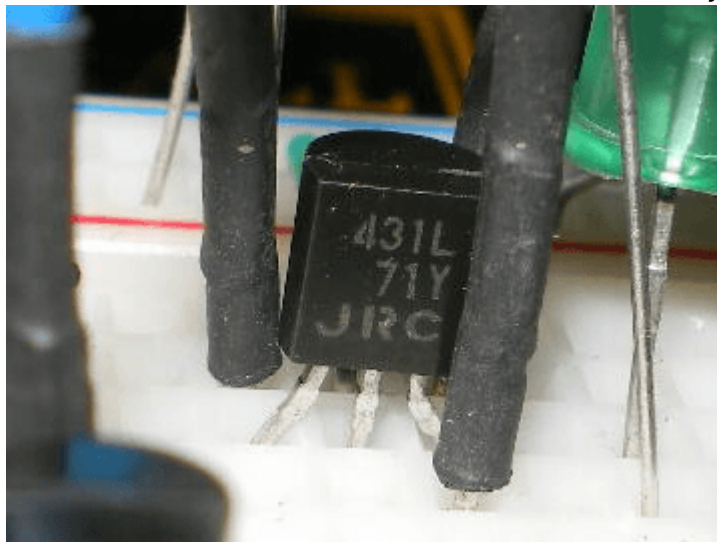
PIE1.CCP1IE = 1;
PIR1.CCP1IF = 0;
CCP1CON.CCP1M3 = 1;
CCP1CON.CCP1M2 = 0;
CCP1CON.CCP1M1 = 1;
CCP1CON.CCP1M0 = 1;
CCPR1L = 0x71;    // 5msec...(1÷4000000)*4*8*625
CCPR1H = 0x02;    //
//
SEG1 = OFF;
SEG2 = OFF;
SEG3 = OFF;
seg_flg = 0;
data1 = SPACE;
data2 = SPACE;
data3 = SPACE;
dot = 0;
// 割り込みを許可する。
INTCON.PEIE = 1;
INTCON.GIE = 1;
//
T1CON.TMR1ON = 1;
//
while (1) {
    dat = measurement(2);
    dat = (dat / 1000.0) * 2.4365234375;    //2.495V/1024...TL431
    dat = (((dat - 300.0) / 10.0) - 30.0) * 10.0;
    IntToStr(dat, buf);
    dot = 2;
    if (dat < 0)
        LED = ON;
    else
        LED = OFF;
    data1 = ((buf[3] == ' ') || (buf[3] == '-')) ? SPACE : buf[3] -
'0';
    data2 = ((buf[4] == ' ') || (buf[4] == '-')) ? SPACE : buf[4] -
'0';
    data3 = ((buf[5] == ' ') || (buf[5] == '-')) ? SPACE : buf[5] -
'0';
}
}

//*****
*
```

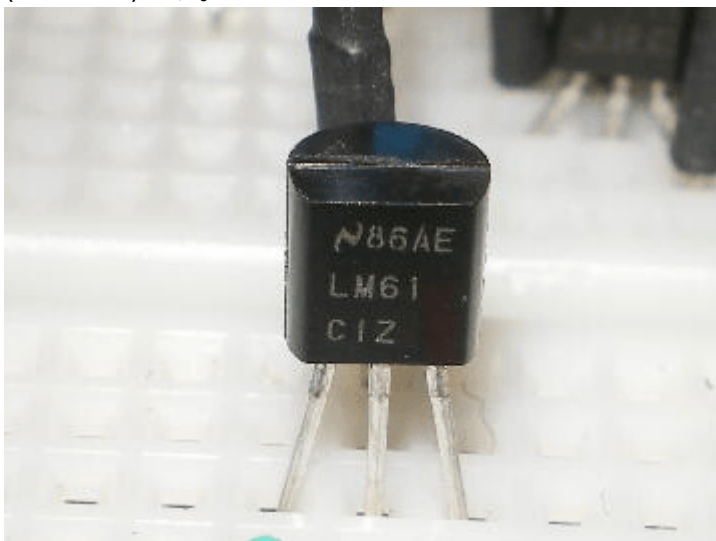
動作確認



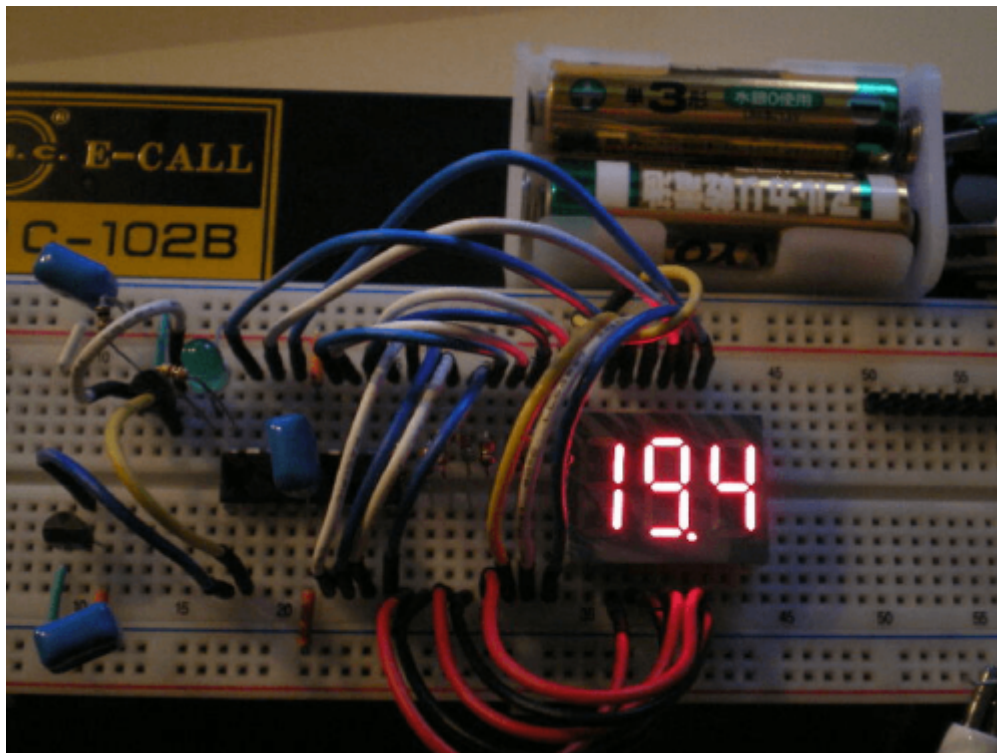
左側:A/D変換用の基準電圧(2.5V)を得るための、シャントレギュレータ(JRC431)です。右側:温度センサ□



(LM61CIZ)です。



私の作業小屋の温度を、測定してみました。



如何ですか? 蛇の目基板に実装すれば、かなりコンパクトに仕上がると思います。

著作権表示 copyright notice

このページは稲崎様の閉鎖したHPのコピーで、著作権は稲崎様にあります。詳細 This page is a copy of Mr. Inasaki's closed website, and the copyright is held by him. Details

From:

<http://deepsky.jp/wiki/> - うごくといいな

Permanent link:

<http://deepsky.jp/wiki/doku.php?id=elechobby:picdic:pic16f88:112>

Last update: 2020/05/01 18:08

