

ミニ電圧&電流計(7セグ表示)

概要

以前にも、電圧計&電流計アダプタ(LCD表示)を製作しました。今回は、測定結果の表示を、7セグ4桁(ダイナミック点灯)にした、とてもコンパクトな電圧&電流計を製作します。<仕様>

- 電圧測定:最大27.5Vとする。(分解能 $26.9\text{mV}=27.5\text{V}\div 1024$)
- 電流測定:最大2.5Aとする。(分解能 $2.4\text{mA}=2.5\text{A}\div 1024$)

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。

<電圧の測定>

- 抵抗R3とR4で、1/11に分圧した電圧(V_0)を測定する。
 $V = V_0 \times 11$ 倍

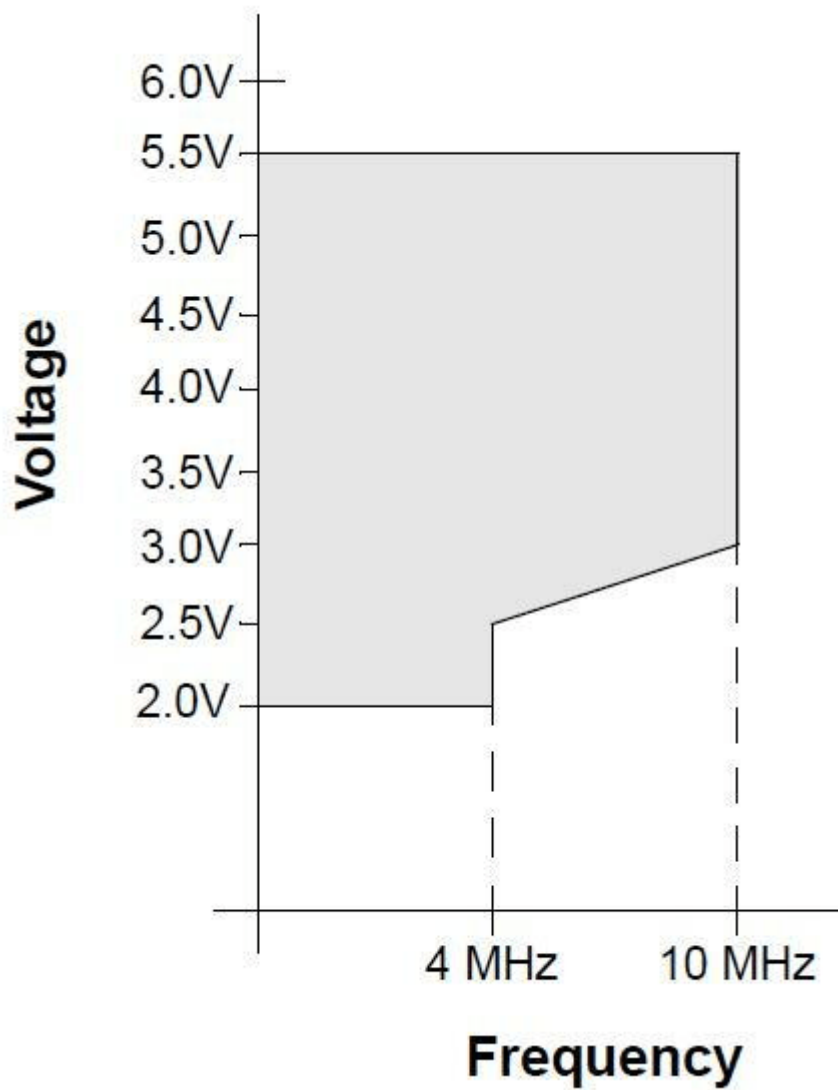
<電流の測定>

- R1の両端電圧(V_0)より電流を測定する。
 $I = V_0 \div 1\Omega$
- 1Ωのワット数は、出来るだけ大きな値のものにして下さい。

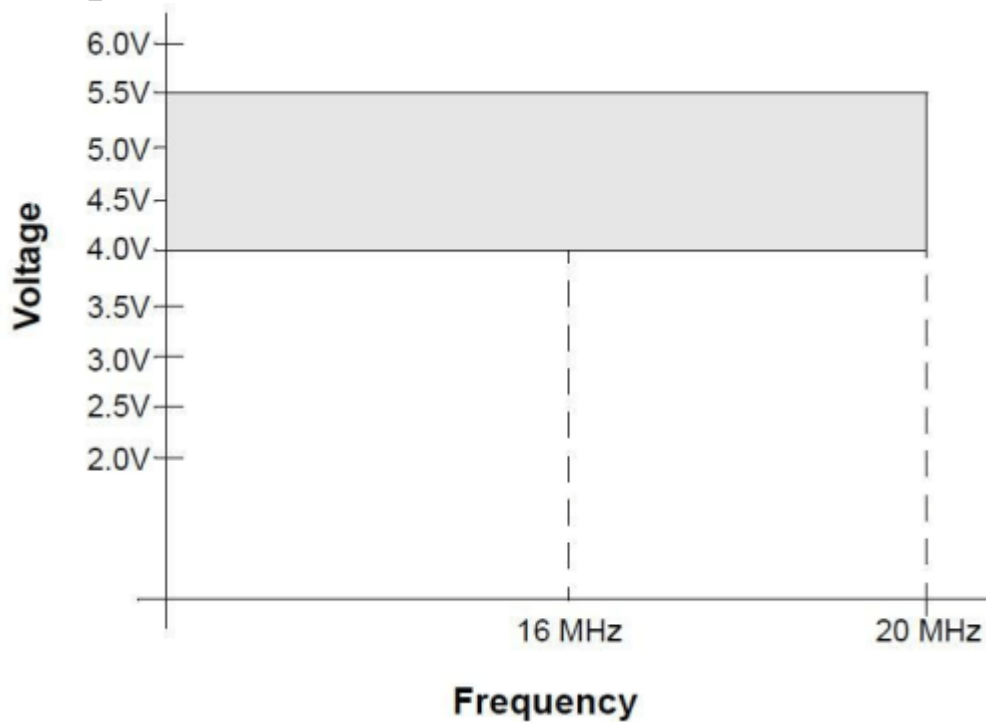
<電源電圧>

- PIC16F88の電源電圧は、規格では $2.0\text{V}\sim 5.5\text{V}$ になっています。但し、使用するクロック周波数によって、最低動作電圧は異なります。(下図参照)
- 10MHz以下の動作電圧
計算式は、
 $F_{\text{MAX}} = (12\text{MHz/V}) \times (V_{\text{DDAPP MIN}} - 2.5\text{V}) + 4\text{MHz}$
で求めることが出来ます。

クロック周波数が、8MHzの場合には、約2.83Vとなり、単三電池2本でも動作させることが出来ます。
 $8\text{MHz} = 12\text{MHz} \times (2.83\text{V} - 2.5\text{V}) + 4\text{MHz}$

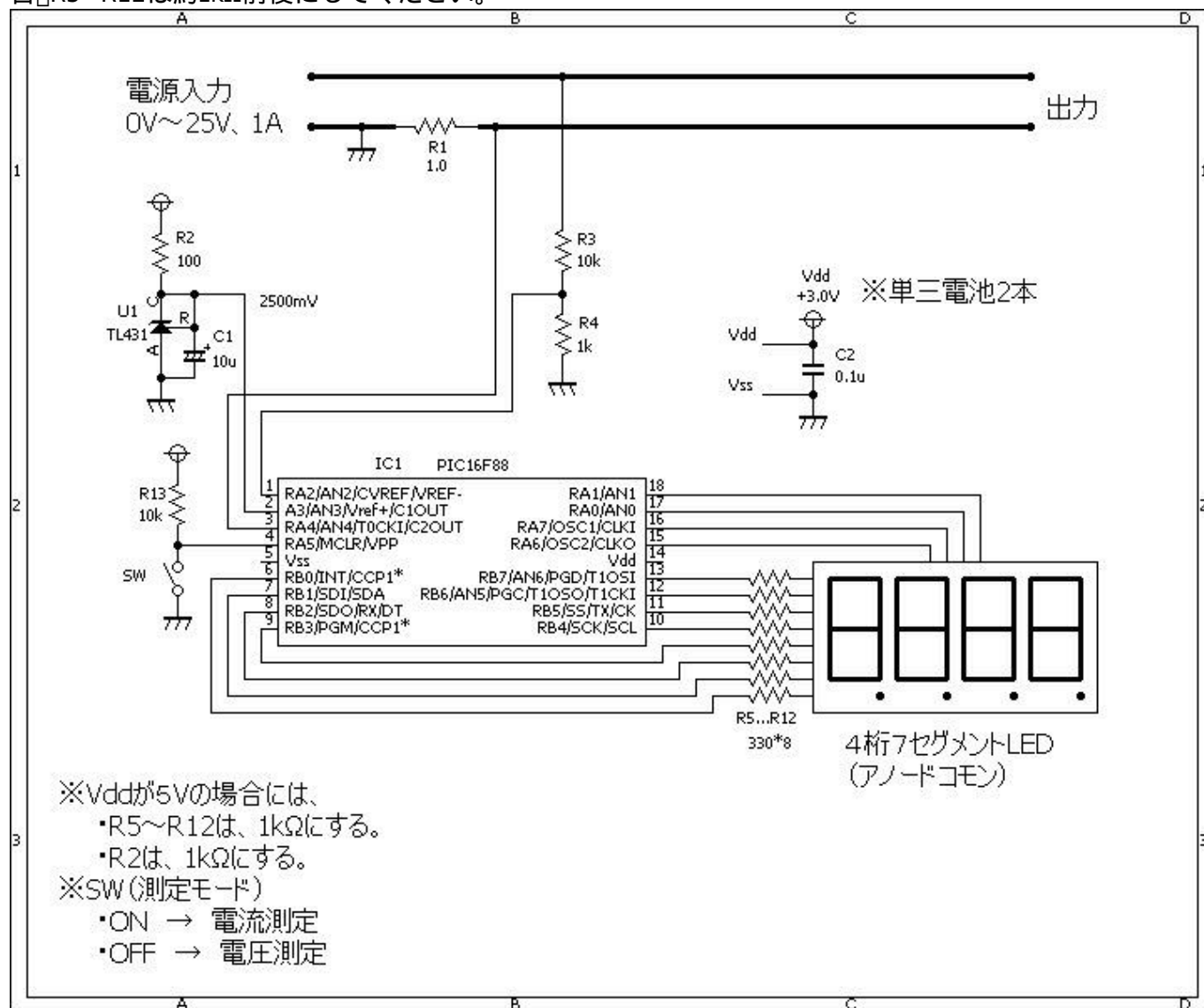


- 16MHz~20MHzの動作電圧

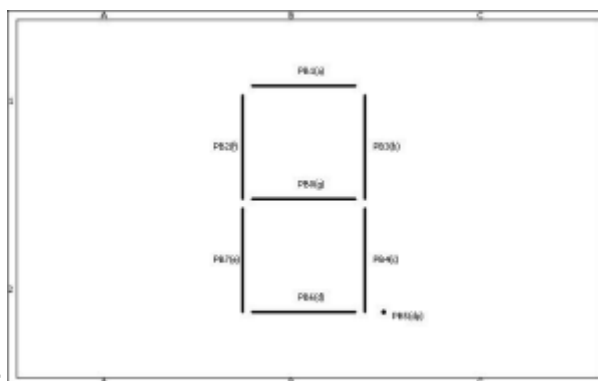


回路図

電源が、単三電池(約3.0V)の場合R5~R12は約330Ω前後にしてください。電源が、5.0Vの場合R5~R12は約1kΩ前後にしてください。



<各セグメントとポートBとの対応>



ソースコード

[vi_meter_7seg_4disp.c](#)

```
//*****
*
/*
    <ミニ電圧&電流計（7セグ表示）>
*/
//*****
*

#define      DATA0      0b00100001
#define      DATA1      0b11100111
#define      DATA2      0b00110100
#define      DATA3      0b10100100
#define      DATA4      0b11100010
#define      DATA5      0b10101000
#define      DATA6      0b00101000
#define      DATA7      0b11100001
#define      DATA8      0b00100000
#define      DATA9      0b10100000
#define      DATA_SPACE  0b11111111

#define      SPACE      10

#define      NON_SEG      0b00000000
#define      SEG1          PORTA.F6
#define      SEG2          PORTA.F7
#define      SEG3          PORTA.F0
#define      SEG4          PORTA.F1

#define      ON            1
#define      OFF           0

#define      SW            PORTA.F5

//*****
*

short  seg_flg, data1, data2, data3, data4, 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;
                SEG4 = OFF;
                PORTB = (dot == 1) ? tbl[data1] & 0b11011111 : tbl[data1];
```

```

        SEG1 = ON;
        break;
    case 1:
        seg_flg = 2;
        SEG1 = OFF;
        PORTB = (dot == 2) ? tbl[data2] & 0b11011111 : tbl[data2];
        SEG2 = ON;
        break;
    case 2:
        seg_flg = 3;
        SEG2 = OFF;
        PORTB = (dot == 3) ? tbl[data3] & 0b11011111 : tbl[data3];
        SEG3 = ON;
        break;
    case 3:
        seg_flg = 0;
        SEG3 = OFF;
        PORTB = (dot == 4) ? tbl[data4] & 0b11011111 : tbl[data4];
        SEG4 = 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 = 0b01110000; // クロックを8Mhzに設定する。
    ANSEL = 0b00010100; // □□□変換を使用する。

```

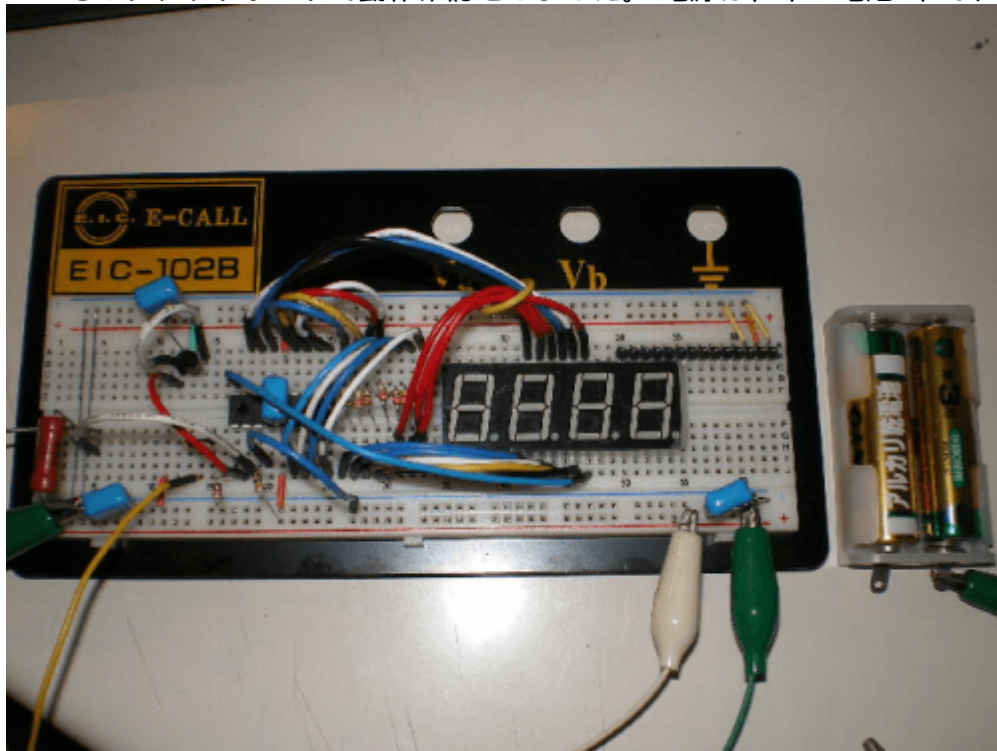
```
ADCON1.VCFG1 = 1;
ADCON1.VCFG0 = 0;
// TIMER0の設定
OPTION_REG.T0CS = 1;
OPTION_REG.PSA = 0;
OPTION_REG.PS2 = 0;
OPTION_REG.PS1 = 1;
OPTION_REG.PS0 = 0;
// TIMER1の設定
PIE1.TMR1IE = 0;
PIR1.TMR1IF = 0;
T1CON.T1CKPS0 = 1;
T1CON.T1CKPS1 = 1;
T1CON.TMR1ON = 0;
TMR1L = 0;
TMR1H = 0;
// CCPの設定
PIE1.CCP1IE = 1;
PIR1.CCP1IF = 0;
CCP1CON.CCP1M3 = 1;
CCP1CON.CCP1M2 = 0;
CCP1CON.CCP1M1 = 1;
CCP1CON.CCP1M0 = 1;
CCPR1L = 0xE2; // 5msec... (1÷8000000)*4*8*1250
CCPR1H = 0x04; //
//
SEG1 = OFF;
SEG2 = OFF;
SEG3 = OFF;
SEG4 = OFF;
seg_flg = 0;
data1 = SPACE;
data2 = SPACE;
data3 = SPACE;
data4 = SPACE;
dot = 0;
// 割り込みを許可する。
INTCON.PEIE = 1;
INTCON.GIE = 1;
//
T1CON.TMR1ON = 1;
//
while (1) {
    if (SW == 1) {
        dat = measurement(2);
        dat = ((dat / 1000.0) * 2.44140625) * 11.0;
        WordToStr(dat, buf);
        dot = 2;
        data1 = buf[0] == ' ' ? SPACE : buf[0] - '0';
        data2 = buf[1] == ' ' ? SPACE : buf[1] - '0';
```

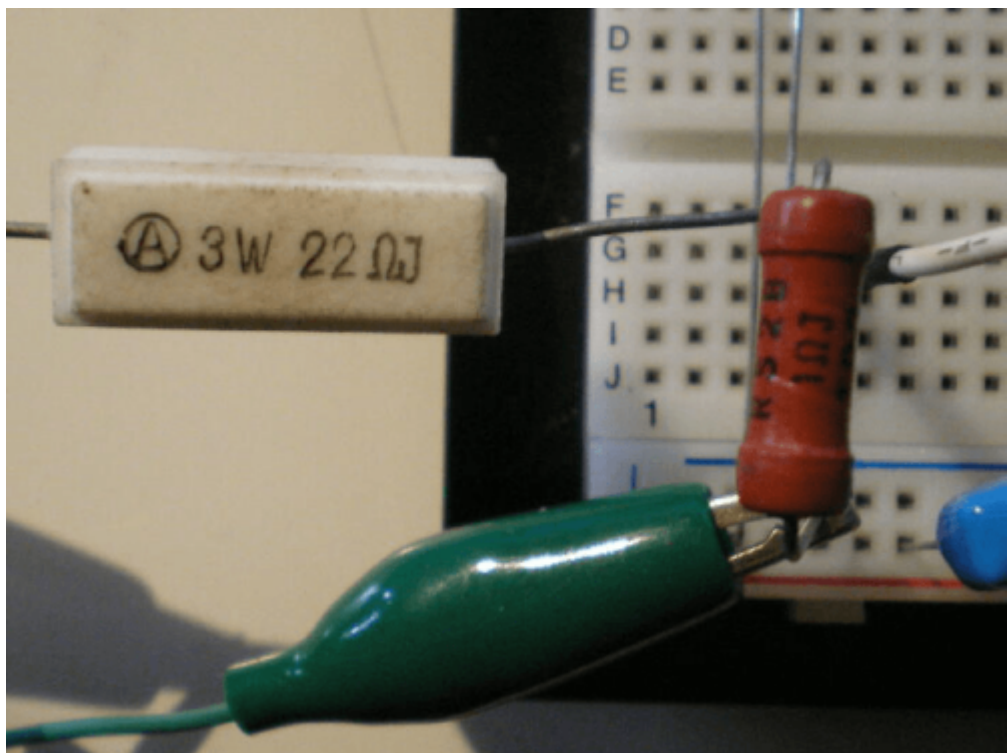
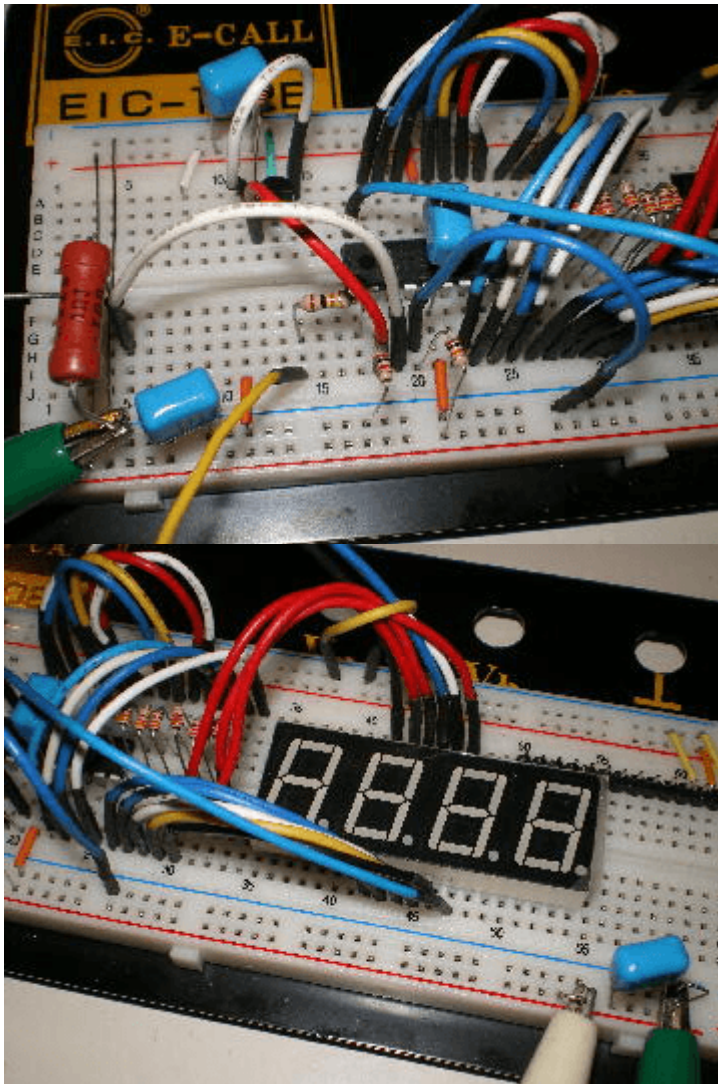
```
data3 = buf[2] == ' ' ? SPACE : buf[2] - '0';
data4 = buf[3] == ' ' ? SPACE : buf[3] - '0';
} else {
    dat = measurement(4);
    dat = ((dat / 1000.0) * 2.44140625) / 1.0;
    WordToStr(dat, buf);
    dot = 1;
    data1 = buf[1] == ' ' ? SPACE : buf[1] - '0';
    data2 = buf[2] == ' ' ? SPACE : buf[2] - '0';
    data3 = buf[3] == ' ' ? SPACE : buf[3] - '0';
    data4 = buf[4] == ' ' ? SPACE : buf[4] - '0';
}
}
}

//*****
*
```

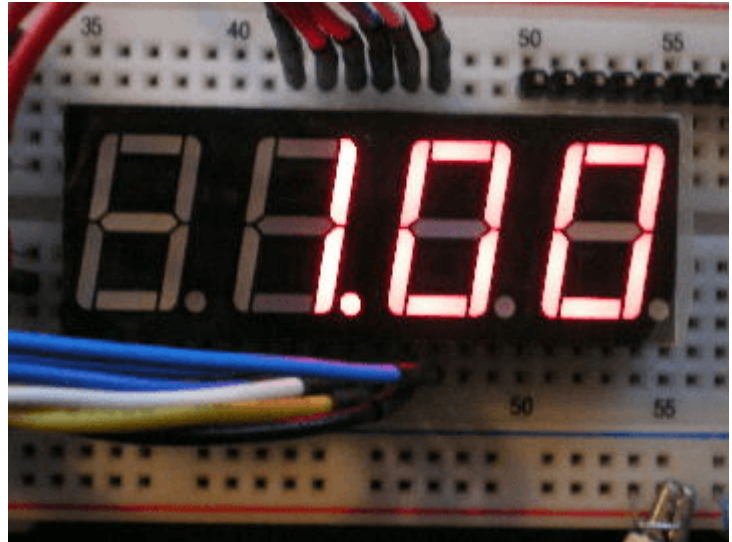
動作確認

いつものブレッドボードで動作確認をしました。電源は、単三電池2本です。

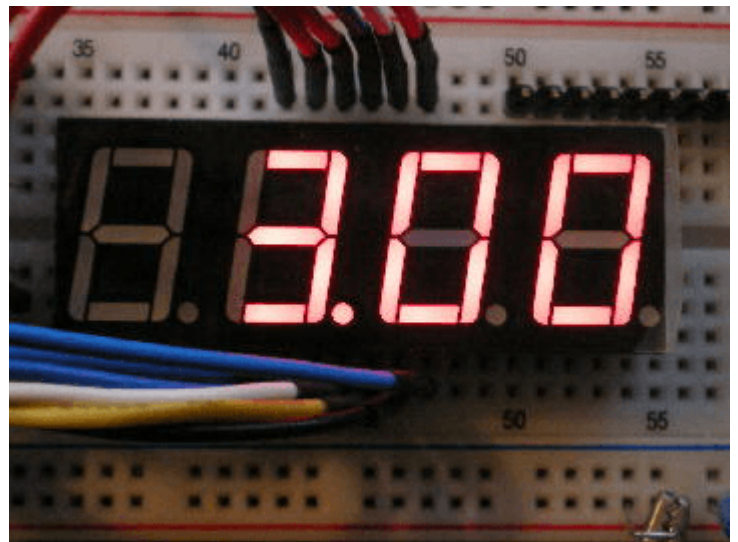
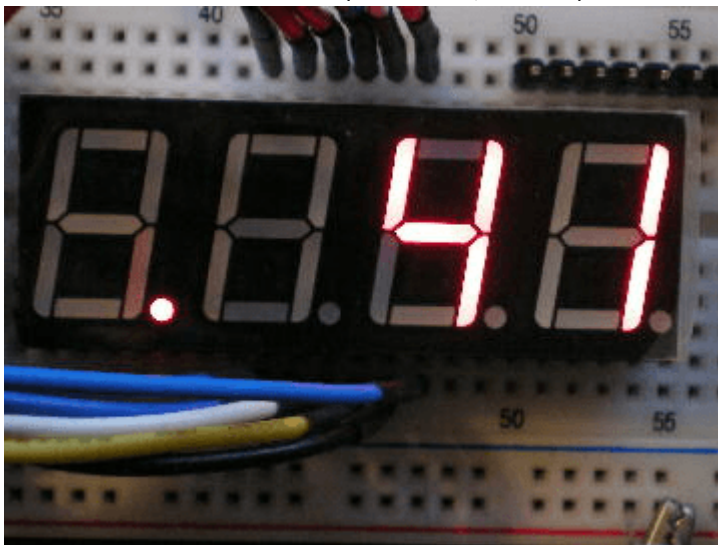




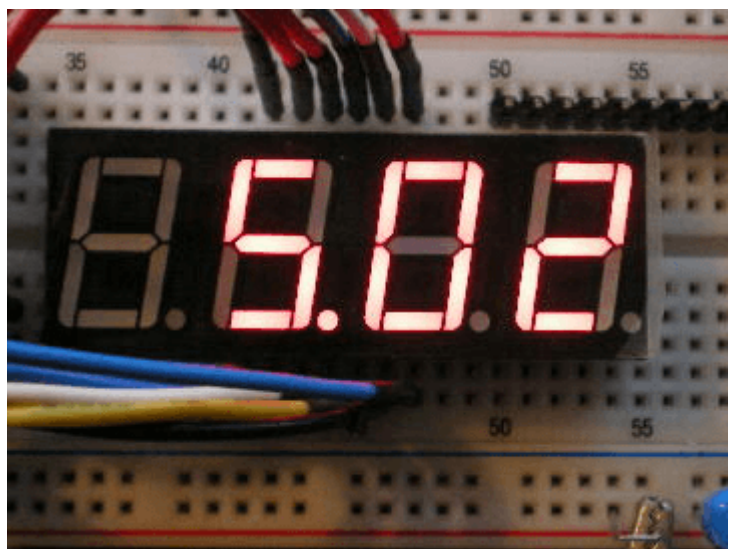
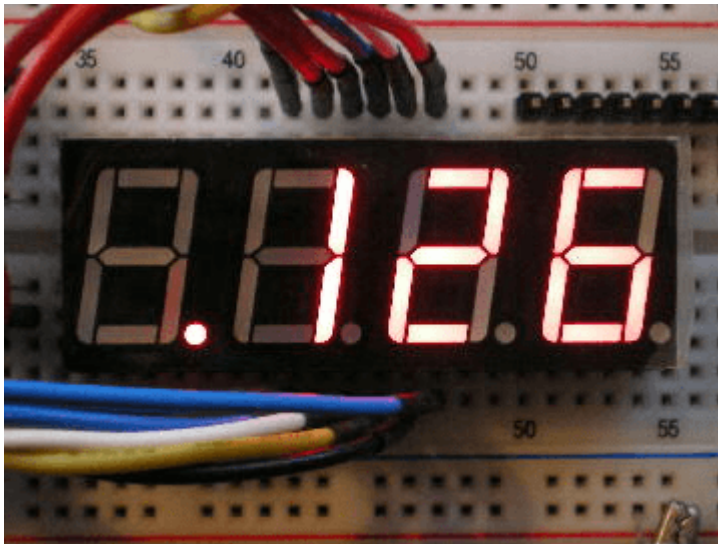
負荷:22Ω 検出抵抗:1Ω



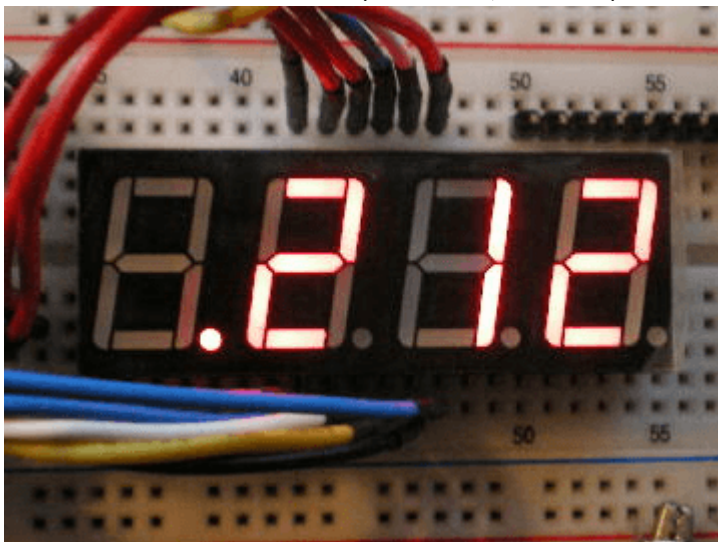
左側:1.0V入力 右側:41mA(計算上は、43mA)

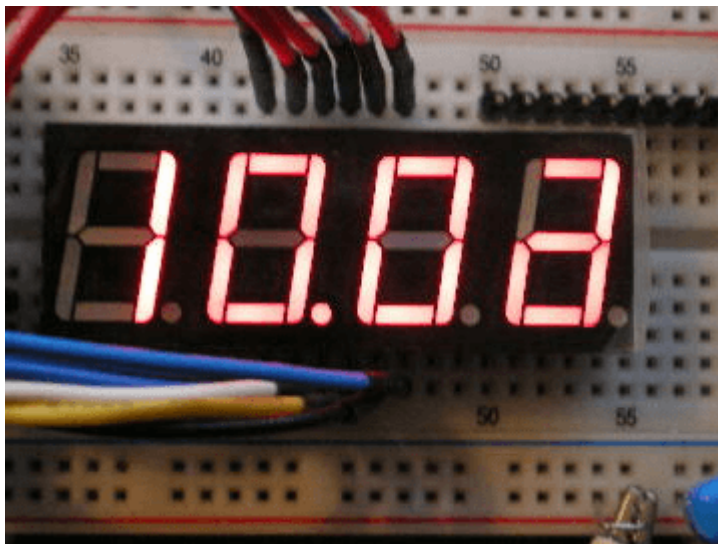


左側:3.0V入力 右側:126mA(計算上は、130mA)

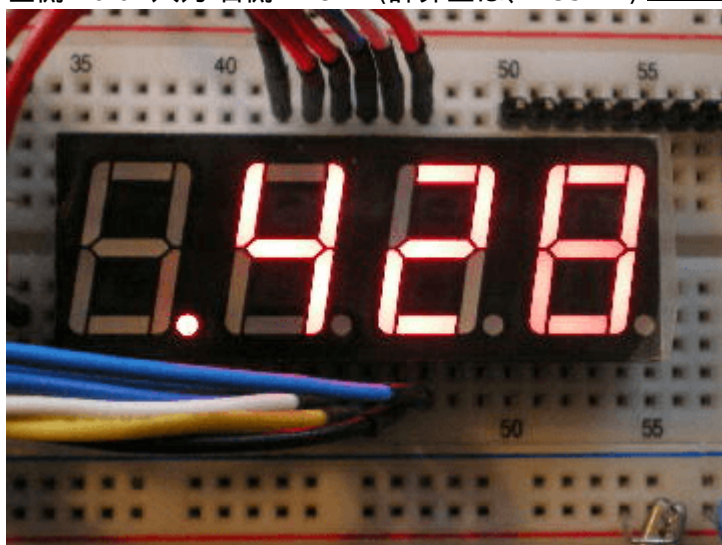


左側:5.0V入力 右側:212mA(計算上は、217mA)





左側:10.0V入力 右側:428mA(計算上は、435mA)



如何ですか? 単三電池2本+7セグ(4桁)なので、かなりコンパクトに仕上がるのではと思っています。

電圧と電流の表示は、精度(分解能)を考慮すると、電圧表示:100mV単位 電流表示:10mA単位 にするほうが望ましいですね。

From:

<http://deepsky.jp/wiki/> - うごくといいな

Permanent link:

<http://deepsky.jp/wiki/doku.php?id=elechobby:picdic:pic16f88:111&rev=1588208412>

Last update: 2025/10/17 14:27

