

多機能交流電圧計

概要

交流(正弦波)信号をテストで測定すると、実効値(RMS:root mean square value)を得ることが出来ます。しかし交流には、実効値の他に、平均値、最大値、P-P値(Peak to Peak)等があります。また、自作したアンプの出力を求めるには、一般的には、テストで実効値を測定して、手で計算を行うことになります。そこでこれらの値をPICで自動計算させて表示させることにしました。

動作原理

出来るだけ非測定物に影響を与えないように、入力インピーダンスは1MΩと高くしました。そして約200分の1に下げた電圧を、LM386を使用して約20倍に増幅します。その出力をPICのA/D変換機能で取り込んで計算を行い結果をLCDに表示します。 <計算式>

- 実効値(Ve)
 $Ve = Vm \div \sqrt{2}$
- 平均値(Vav)
 $Vav = (2 \times Vm) \div \pi$
- 最大値(Vm)
0ボルトから波形のピークまでの電圧
- P-P値(Vpp)
波形の最小から最大までの電圧
- 出力電力値(W)
 $W = V \times I = V \times V \div R$

<仕様>

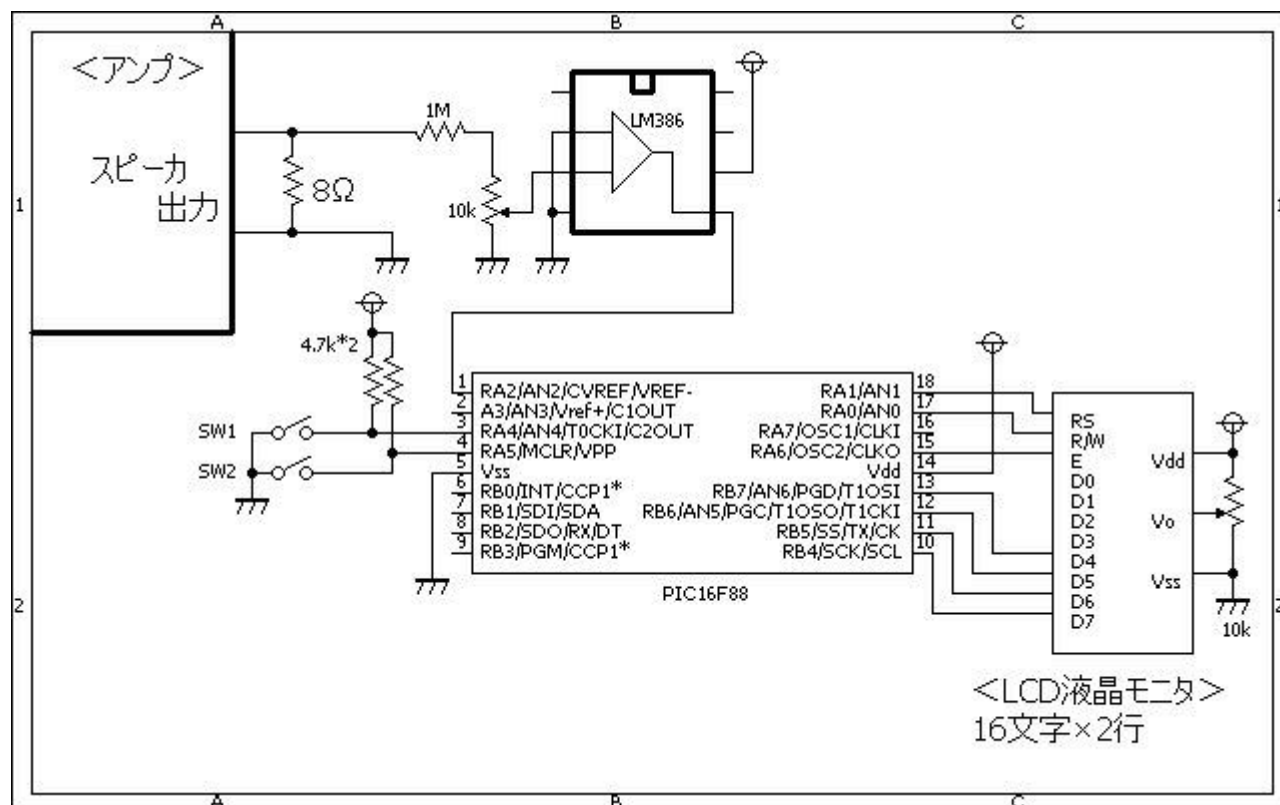
- 測定可能な周波数範囲は、約10Hz~100kHz迄の正弦波です。
- P-P値が、最大約40Vですので、8Ω負荷のアンプですと約12Wくらいまで測定が出来ます。

<調整>

- 実効値の分かっている正弦波(5V~30Vくらい)の信号を入力します。
- 10kΩのボリュームを廻してLCDの実効値の表示内容が同じ値を示すようにします。

回路図

- モード1 SW1=OFF SW2=OFF
 - 実行値Ve(mV)
 - 平均値Vav(mV)
- モード2 SW1=ON SW2=OFF
 - 最大値Vm(mV)
 - P-P値Vpp(mV)
- モード3 SW2=ON
 - アンプ出力値Pw(mW)8Ω 負荷
 - アンプ出力値Pw(mW)16Ω負荷



ソースコード

acVoltmeter.c

```
//*****
*
/*
< 交流電圧計 >

交流電圧入力における下記項目を表示します。

モード□□□□□□□□□□□□□□□□
実行値□□□□□□□□
平均値□□□□□□□□

モード□□□□□□□□□□□□□□□□
最大値□□□□□□□□
□□□□◎□□□値□□□□□□□□

モード□□□□□□□□□□
アンプ出力値□□□□□□□□Ω 負荷
アンプ出力値□□□□□□□□Ω 負荷

□□□□□□□□□□□□□□
□□□□□□□□□□□□□□

測定可能な周波数範囲は、約 1□□□□□□□□□□迄の正弦波です。
```

```
*/
//*****
*

unsigned int measurement()
{
    unsigned int ad, max, min, cnt;
    //
    ad = 0;
    max = 0;
    min = 1024;
    for (cnt = 0; cnt < 2000; cnt++) {
        ad = Adc_Read(2);
        max = ad > max ? ad : max;
        min = ad < min ? ad : min;
    }
    return (max - min);
}

//*****
*

void main()
{
    static unsigned char buf[6], cnt, *msg1, *msg2, *msg3,
    *msg4;
    static unsigned int vm, ve, vav, pw, tmp;
    static double Dvm;
    //
    msg1 = "mV ";
    msg2 = "mW(8) ";
    msg3 = "mW(16) ";
    msg4 = " Pw=";
    //
    OSCCON = 0b01110000; // クロックは8Mhz
    CMCON = 0b00000111; // コンパレータは使用しない。
    // □□□変換を使用する。
    ANSEL = 0b00000100;
    // ポートを初期化する。
    TRISA = 0b10111100;
    TRISB = 0b00001111;
    // □□□を初期化する。
    Lcd_Custom_Config(&PORTB, 4, 5, 6, 7, &PORTA, 1, 0, 6);
    Lcd_Custom_Cmd(LCD_CURSOR_OFF);
    Lcd_Custom_Cmd(LCD_CLEAR);
    //
    while (1) {
        tmp = 0;
        for (cnt = 0; cnt < 5; cnt++) {
            tmp += measurement();
        }
    }
}
```

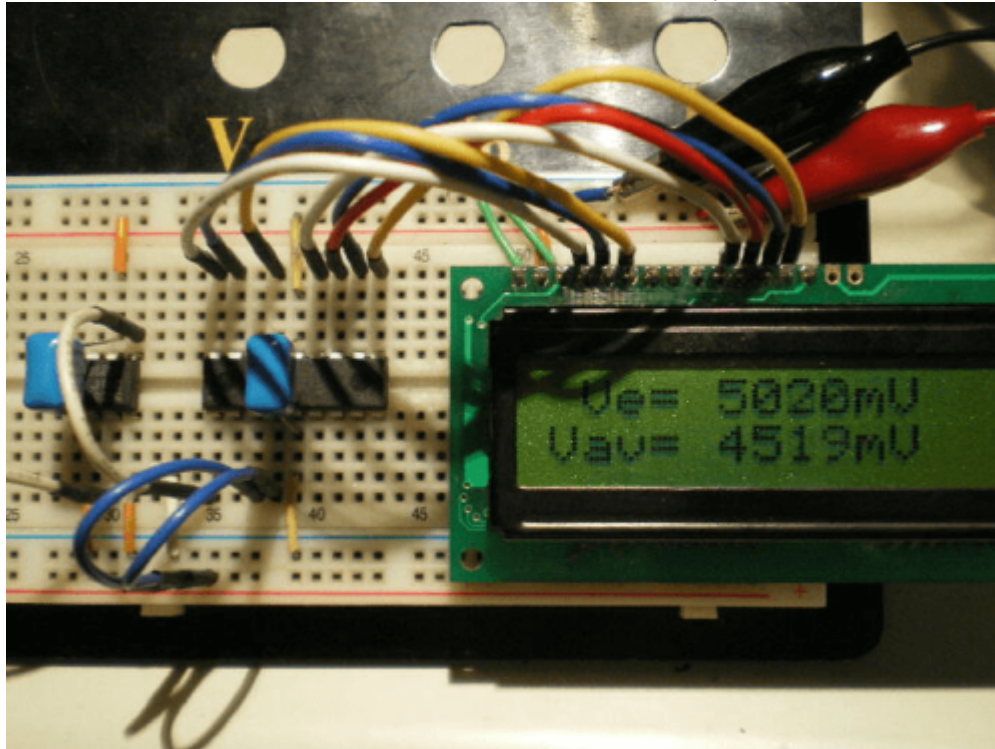
```
}
Dvm = (double)tmp * 4.8828125;
//
if (PORTA.F5 == 0) {
    Dvm = Dvm / 1.41421356;
    Dvm = Dvm * Dvm;
    //
    pw = Dvm / 8000.0;
    WordToStr(pw, buf);
    Lcd_Custom_Out(1, 1, msg4);
    Lcd_Custom_Out(1, 5, buf);
    Lcd_Custom_Out(1, 10, msg2);
    //
    pw = Dvm / 16000.0;
    WordToStr(pw, buf);
    Lcd_Custom_Out(2, 1, msg4);
    Lcd_Custom_Out(2, 5, buf);
    Lcd_Custom_Out(2, 10, msg3);
    //
    continue;
}
//
if (PORTA.F4 == 1) {
    ve = Dvm / 1.41421356;
    WordToStr(ve, buf);
    Lcd_Custom_Out(1, 1, " Ve=");
    Lcd_Custom_Out(1, 5, buf);
    Lcd_Custom_Out(1, 10, msg1);
    //
    vav = (Dvm * 2.0) / 3.14159265;
    WordToStr(vav, buf);
    Lcd_Custom_Out(2, 1, "Vav=");
    Lcd_Custom_Out(2, 5, buf);
    Lcd_Custom_Out(2, 10, msg1);
} else {
    vm = Dvm;
    WordToStr(vm, buf);
    Lcd_Custom_Out(1, 1, " Vm=");
    Lcd_Custom_Out(1, 5, buf);
    Lcd_Custom_Out(1, 10, msg1);
    //
    WordToStr(vm * 2, buf);
    Lcd_Custom_Out(2, 1, "Vpp=");
    Lcd_Custom_Out(2, 5, buf);
    Lcd_Custom_Out(2, 10, msg1);
}
}
}

//*****
```

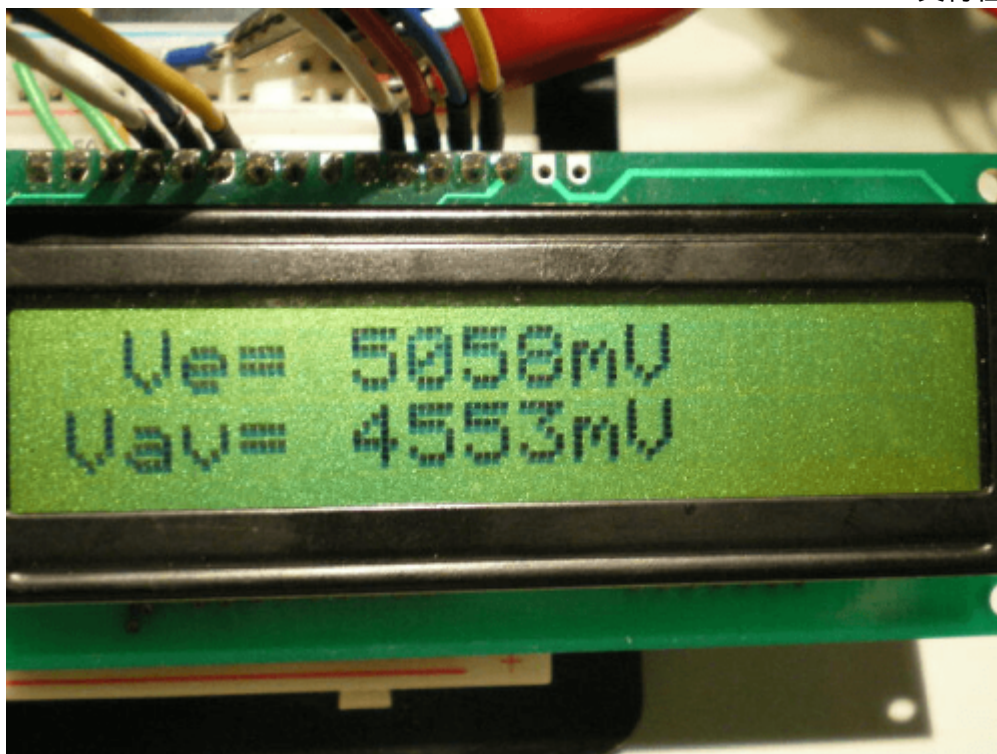
*

動作確認

いつものようにブレッドボードで動作確認をしました。

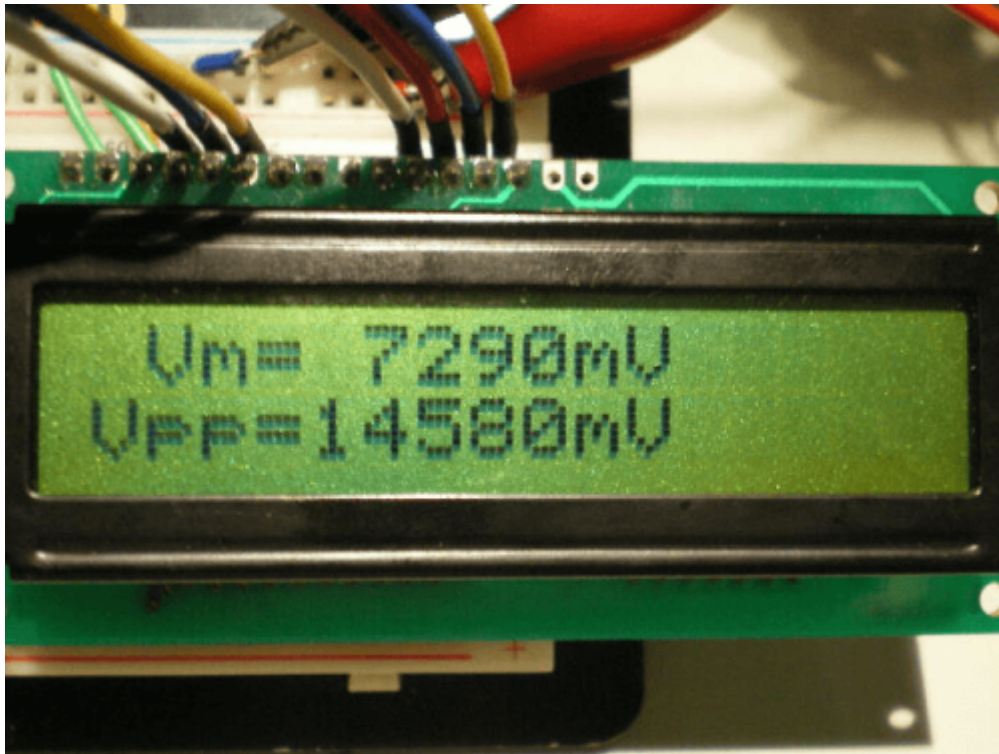


実行値(V_e)と平均値(V_{av})



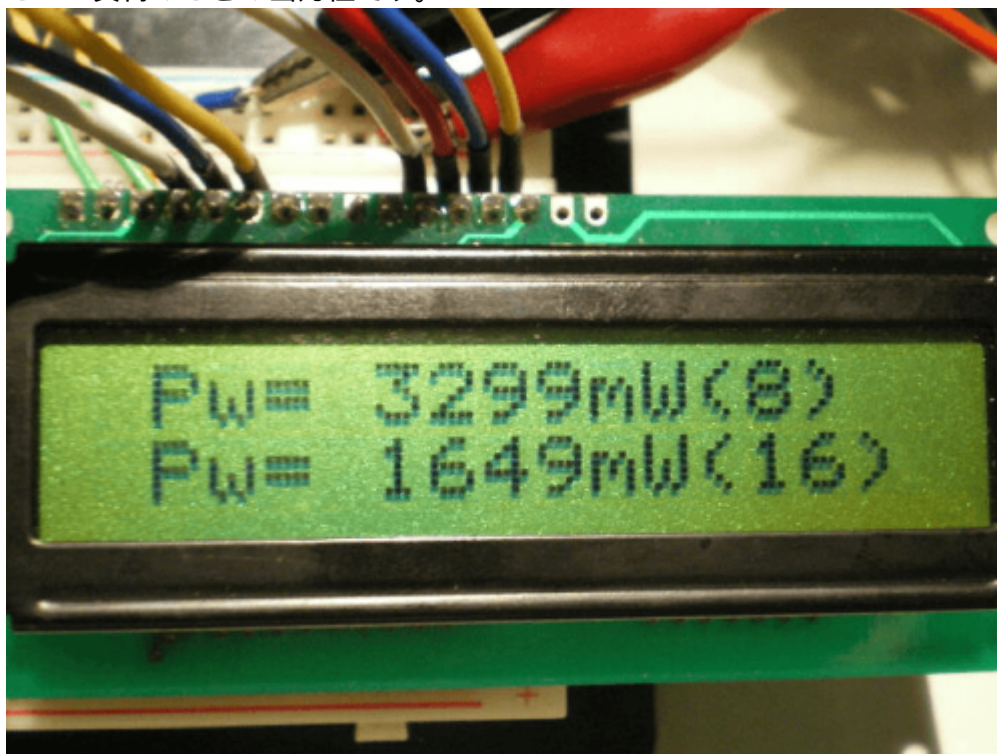
です。

最大値(V_m)とP-P



値(Vpp)です。
と16Ω負荷のときの出力値です。

8Ω負荷



如何ですか? これをケース
に入れて乾電池駆動できるようにして手元に置いてあれば何かと便利ですね。

From:
<http://deepsky.jp/wiki/> - うごくといいな

Permanent link:
<http://deepsky.jp/wiki/doku.php?id=elechobby:picdic:pic16f88:66&rev=1588151927>

Last update: 2025/10/17 14:28

