

簡易オシロ(グラフィック液晶表示)(PIC16F876)

概要

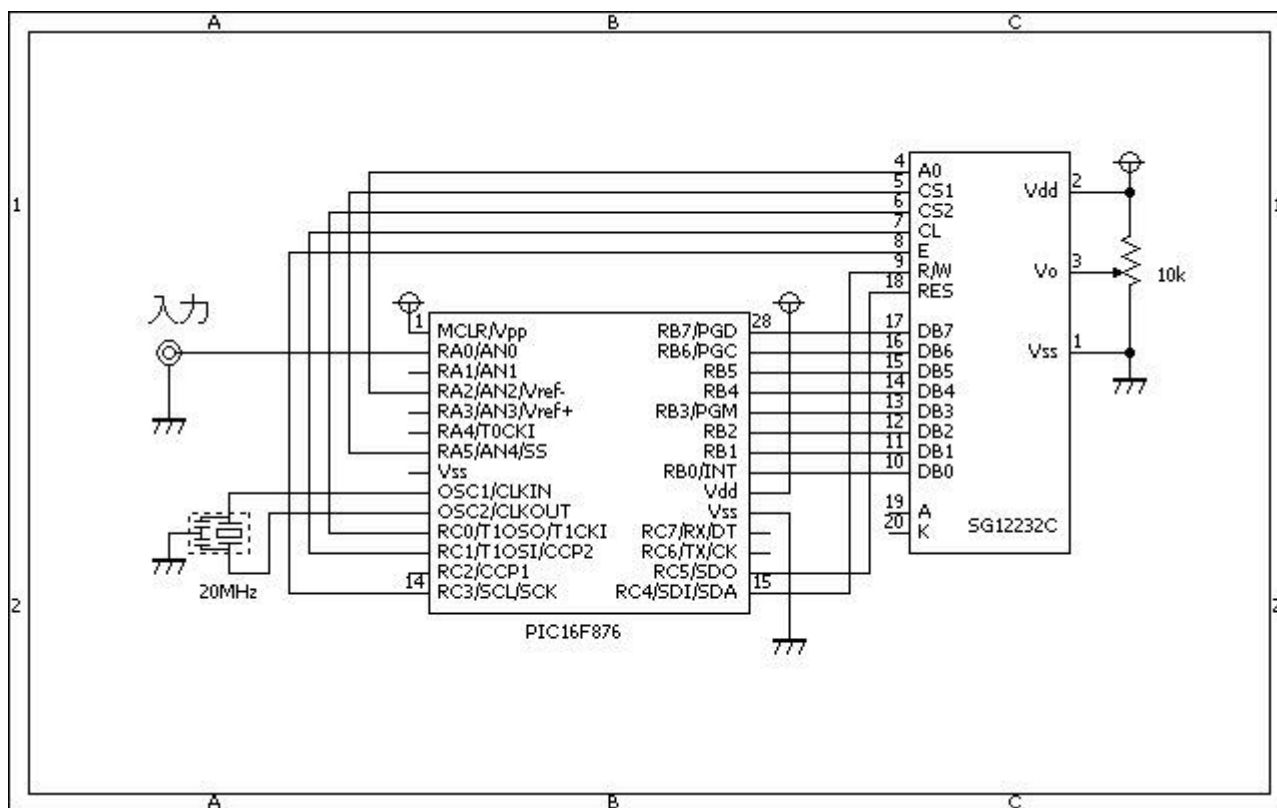
秋月電子通商で購入した「LCDグラフィック・ディスプレイモジュール(バックライト付)」を利用した簡易オシロを作成しました。

- SUNLIKE社グラフィックLEDモジュールです。
- 122×32ドット
- 漢字パターンを送れば、漢字 記号 図形を自由に表示できます。
- 5V単一電源
- H8マイコン・PICマイコンなどに直結できます。
- 制御ICはEPSONS1D1520準拠
- 液晶サイズ:60.5(W)×18.5(H)
- 接続端子:20ピン(2×10)

動作原理

PICのA/D変換機能でアナログデータを取り込みそれを単純に液晶に表示させるだけのものです。但し、PICのA/Dの分解能は10ビットなので5ビット分右にシフトして5ビット(32)にします。このモジュールは、液晶駆動クロック(2kHz~3kHz)を外部から供給する必要があります。今回はPICのPWM機能を使って供給しました。なので回路図としてはとてもシンプルです。

回路図



ソースコード

[gLcdDisp2.c](#)

```
//*****
*

#define    LcdA0          PORTA.F2
#define    LedPower      PORTA.F4
#define    LcdCS1        PORTA.F5
#define    LcdCS2        PORTC.F0
#define    LcdCL         PORTC.F1
#define    LcdRD         PORTC.F3
#define    LcdWR         PORTC.F4
#define    LcdRES        PORTC.F5
#define    LcdData       PORTB

static unsigned   char   temp;

//*****
*

void    gLcdBusy(unsigned char cs)
{
    TRISB = 0b11111111;
    LcdA0 = 0;
    if (cs == 1)
```

```
    LcdCS1 = 0;
else
    LcdCS2 = 0;
while (1) {
    LcdRD = 0;
    temp = LcdData;
    LcdRD = 1;
    if (temp.F7 == 0)
        break;
}

if (cs == 1)
    LcdCS1 = 1;
else
    LcdCS2 = 1;

TRISB = 0b00000000;
}

void    gLcdCmd(unsigned char cs, unsigned char cmd)
{
    gLcdBusy(cs);
    LcdA0 = 0;
    if (cs == 1)
        LcdCS1 = 0;
    else
        LcdCS2 = 0;
    LcdData = cmd;
    LcdWR = 0;
    LcdWR = 1;
    if (cs == 1)
        LcdCS1 = 1;
    else
        LcdCS2 = 1;
}

void    gLcdData(unsigned char cs, unsigned char data)
{
    gLcdBusy(cs);
    LcdA0 = 1;
    if (cs == 1)
        LcdCS1 = 0;
    else
        LcdCS2 = 0;
    LcdData = data;
    LcdWR = 0;
    LcdWR = 1;
    if (cs == 1)
        LcdCS1 = 1;
    else
        LcdCS2 = 1;
}
```

```
}

void  gLcdClr(unsigned char mode)
{
    unsigned    char    cnt, data;
    data = (mode == 1) ? 0xFF : 0x00;
    //
    gLcdCmd(1, 0b10111000);          // ページアドレス設定<0>
    gLcdCmd(2, 0b10111000);          // ページアドレス設定<0>
    gLcdCmd(1, 0b00000000);          // カラムアドレス設定<0>
    gLcdCmd(2, 0b00000000);          // カラムアドレス設定<0>
    for(cnt = 0; cnt < 61; cnt++) {
        gLcdData(1, data);
        gLcdData(2, data);
    }

    gLcdCmd(1, 0b10111001);          // ページアドレス設定<1>
    gLcdCmd(2, 0b10111001);          // ページアドレス設定<1>
    gLcdCmd(1, 0b00000000);          // カラムアドレス設定<0>
    gLcdCmd(2, 0b00000000);          // カラムアドレス設定<0>
    for(cnt = 0; cnt < 61; cnt++) {
        gLcdData(1, data);
        gLcdData(2, data);
    }

    gLcdCmd(1, 0b10111010);          // ページアドレス設定<2>
    gLcdCmd(2, 0b10111010);          // ページアドレス設定<2>
    gLcdCmd(1, 0b00000000);          // カラムアドレス設定<0>
    gLcdCmd(2, 0b00000000);          // カラムアドレス設定<0>
    for(cnt = 0; cnt < 61; cnt++) {
        gLcdData(1, data);
        gLcdData(2, data);
    }

    gLcdCmd(1, 0b10111011);          // ページアドレス設定<3>
    gLcdCmd(2, 0b10111011);          // ページアドレス設定<3>
    gLcdCmd(1, 0b00000000);          // カラムアドレス設定<0>
    gLcdCmd(2, 0b00000000);          // カラムアドレス設定<0>
    for(cnt = 0; cnt < 61; cnt++) {
        gLcdData(1, data);
        gLcdData(2, data);
    }
}

void  gLcdDotSet(unsigned char x, unsigned char y)    // x(0...121),
                                                    y(0...31)
{
    unsigned    char    cs, pa, ca, dt;
    if (x < 61)
        cs = 1;
}
```

```
    else
        cs = 2;
    //
    switch (y / 8) {
    case 0:
        pa = 0b10111000;
        break;
    case 1:
        pa = 0b10111001;
        break;
    case 2:
        pa = 0b10111010;
        break;
    case 3:
        pa = 0b10111011;
        break;
    }
    gLcdCmd(cs, pa);
    //
    ca = x - ((cs - 1) * 61);
    gLcdCmd(cs, ca);
    //
    switch (y / 8) {
    case 0:
        dt = y;
        break;
    case 1:
        dt = y - 8;
        break;
    case 2:
        dt = y - 16;
        break;
    case 3:
        dt = y - 24;
        break;
    }
    dt = 0b00000001 << dt;
    gLcdData(cs, dt);
}

void    gLcdInit()
{
    LcdA0 = 1;
    LcdCS1 = 1;
    LcdCS2 = 1;
    LcdRD = 1;
    LcdWR = 1;
    LcdRES = 0;
    gLcdCmd(1, 0b10101111);    // 表示ON
    gLcdCmd(2, 0b10101111);    // 表示ON
    gLcdClr(0);                // 画面クリア
}
```

```
}

//*****
*

void  initPwm()
{
    CCP1CON = 0b00001100;
    CCP2CON = 0b00001100;
    PR2 = 0x7F;
    T2CON = 0b00000110;
    CCPR1L = 0x00;
    CCPR2L = 0x3F;
}

//*****
*

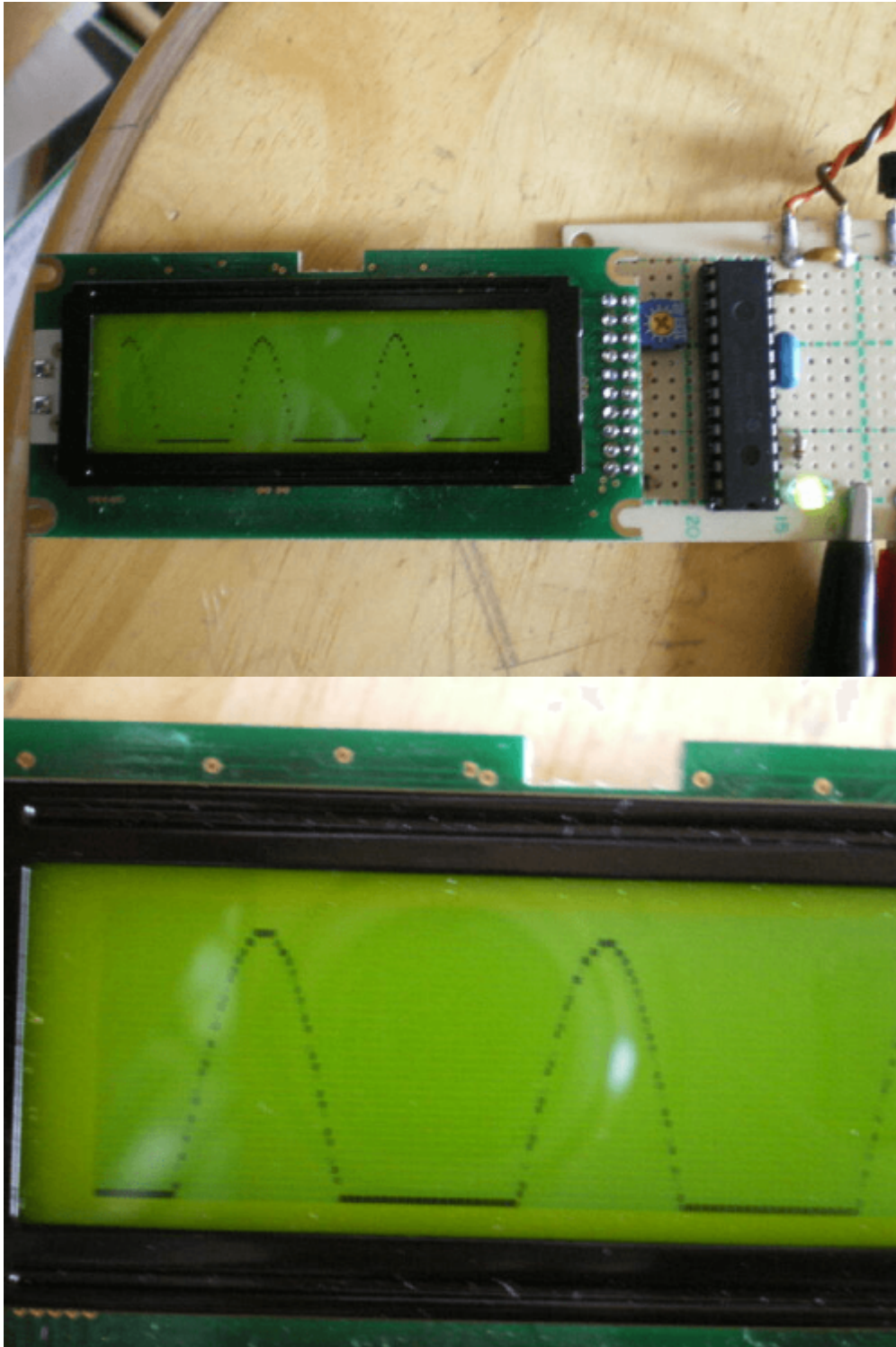
void main()
{
    unsigned char cnt, x, y;
    unsigned int ad0;
    // ポートの設定
    TRISA = 0b00001001;
    TRISB = 0b00000000;
    TRISC = 0b01000000;
    ADCON1 = 0b00001110;
    //
    initPwm();
    //
    gLcdInit();
    for (cnt = 0; cnt < 5; cnt++) {
        Delay_ms(300);
        gLcdClr(1);           // 画面クリア
        Delay_ms(300);
        gLcdClr(0);           // 画面クリア
    }
    //
    while (1) {
        for (x = 0; x < 122; x++) {
            ad0 = Adc_Read(0);
            ad0 = (ad0 >> 5) & 0b00011111;
            gLcdDotSet(x, ad0);
        }
        Delay_ms(100);
        gLcdClr(0);           // 画面クリア
    }
}

//*****
```

*

動作確認

100Hzの正弦波(半波)を入力してみました。



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

Permanent link:
<http://deepsky.jp/wiki/doku.php?id=elechobby:picdic:otherpic:164&rev=1588236393>

Last update: **2025/10/17 14:27**

