

# USB Port Alarm

2014.11.11 mtsw

**概要**：USB電源が切れるとアラーム音を発する。

**欠点**：PCがシャットダウンされると発報してしまう。

**回路動作**：

- USB電源の5Vを抵抗分割して、感知用電圧として用いる。
- 感知電圧をPICで監視し、およそ1V※以下になったら、“SOS”信号をPIC12F675のGP5(A5)端子の電圧変化として発生する。
- GP5には、電子ブザーと発光ダイオードを接続しているため、音響と光で“SOS”信号を発する。
- PIC用電源はUSBポートから受電し、整流して電気二重層コンデンサ(C2)に充電し、これを電源として用いる。  
C2のESRはおよそ10Ωであり、R2が160Ωの時、満充電に要する時間は20分となる。その状態で連続発報時間は約20分であり、発報停止PIC電圧は2.3Vであった。
- 電子ブザーPB04-SE12SHPRの消費電流は約10mAであり、発報プログラム次第で、さらに時間を延長できる。  
PIC自体は約100μAである。

**備考**：実験課題「マイクロプロセッサ」がUSBポートの使用を禁止しているが、無視して使用した時の発見機器の要求があった。

※ADCのVrefはPICのVccとしたので、若干の差が生ずる。

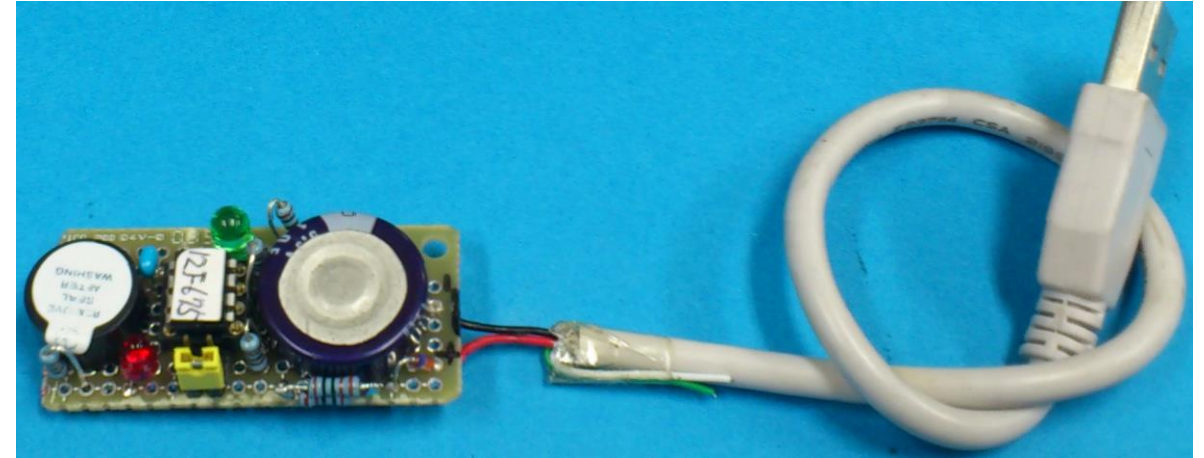


図1 USBポートアラーム試作

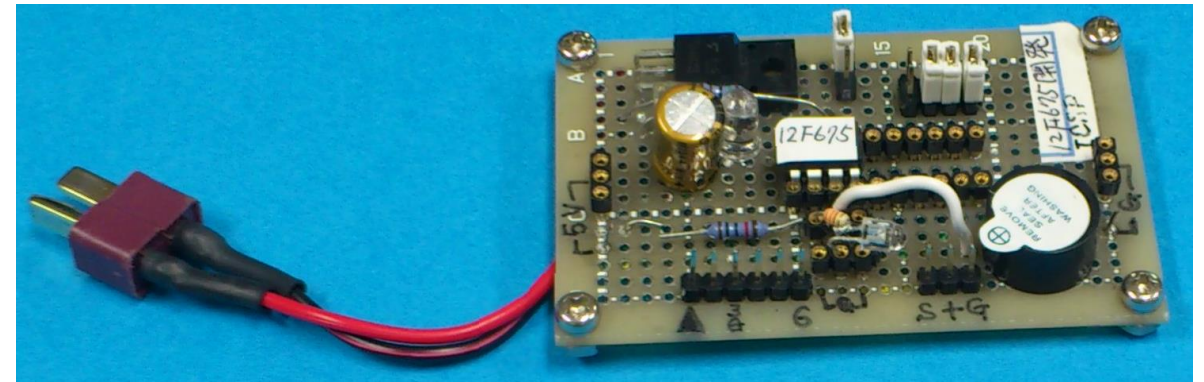


図2 PIC12F675 プログラム（書き込み）回路

Alarm 回路の発想から試作版作製まではおおよそ2時間程度であった。

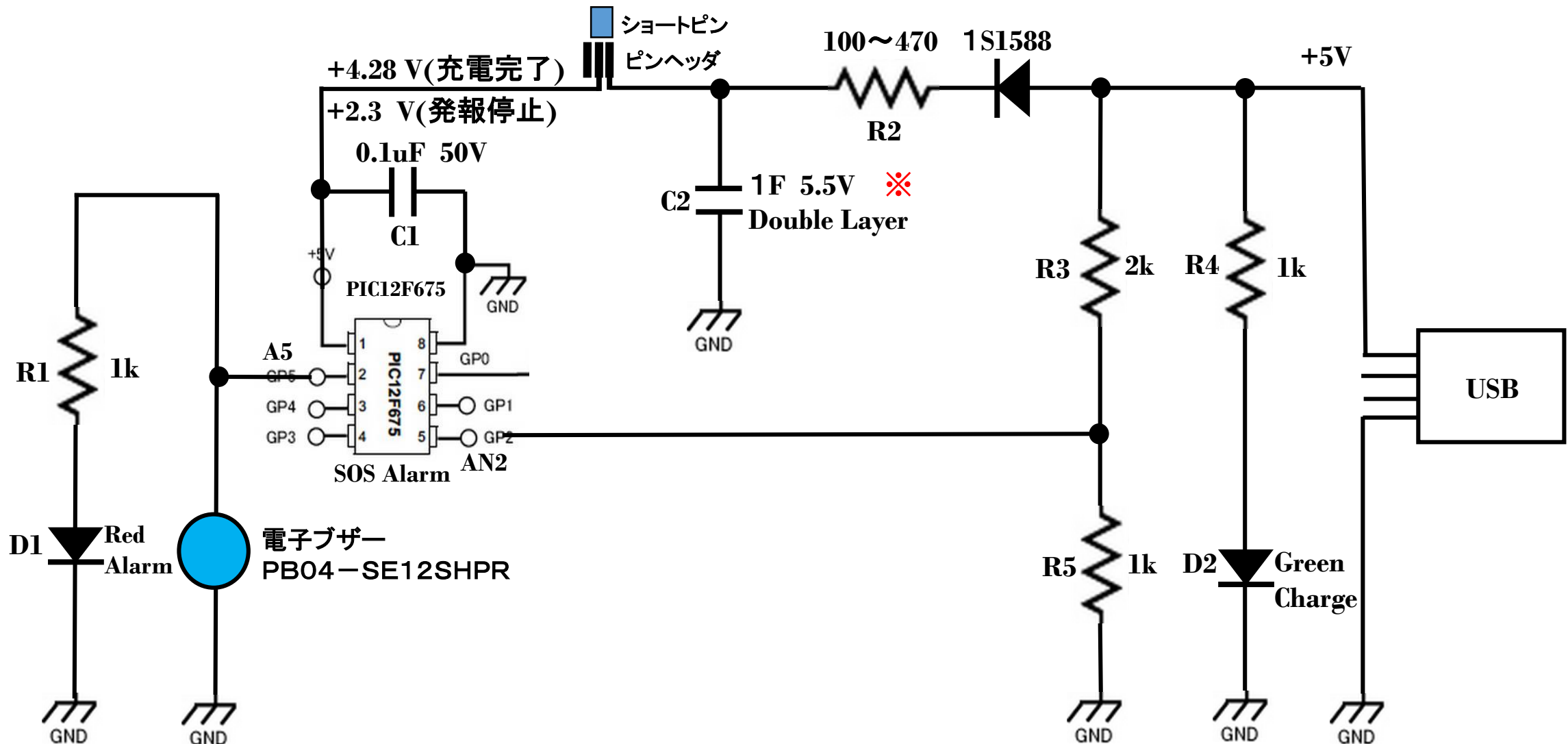


図3 USB Port Alarm回路

$\times$  電気二重層大容量コンデンサ ESR $\approx$ 10 $\Omega$  (1kHz)  
 満充電はおよそ20分 (R2=160 $\Omega$ )

グラフイメージデータ抜粋↓  
<http://www.robotsfx.com/robot/robohow/RoboHow41/RoboHow41.html>

# USB Port Alarm Control Program

```
// コンパイラはCCS C

#include <12F675.h>
#fuses INTRC_IO,NOWDT,NOPROTECT,NOMCLR,BROWNOUT
#use delay(clock=4000000)
##device ADC=10
##use rs232(baud=9600,xmit=PIN_A5,rcv=PIN_A3,parity=N,bits=8)

int To(int x)
{
    output_high(PIN_A5);    // PIN 2
    delay_ms(70);
    output_low(PIN_A5);
    delay_ms(70);
}

int Tu(int x)              // Tu
{
    output_high(PIN_A5);
    delay_ms(140);
    output_low(PIN_A5);
    delay_ms(70);
}

int TuTuTu(int x)        // TuTuTu
{
    Tu(0);
    Tu(0);
    Tu(0);
}

int ToToTo(int x)        // ToToTo
{
    To(0);
    To(0);
    To(0);
}

int SOS(int x)           // ToToTo TuTuTu ToToTo
{
    To(0);To(0);To(0);
    Tu(0);Tu(0);Tu(0);
    To(0);To(0);To(0);
}

void main(void){
    long ADC;

    while(1){
        setup_adc_ports(0x80);           // All ANALOG
        setup_adc_ports(2);             // PIN 5
        setup_adc(ADC_CLOCK_INTERNAL);  //ADC clock is included OSC
        set_adc_channel(AN2_ANALOG);    //Select Pin No5
        delay_us(100);
        ADC = read_adc(ADC_START_AND_READ);
        delay_us(100);
        if(ADC < 50){                   // 50 = 1V
            SOS(0);
            delay_ms(1000);
        }
    }
}
```