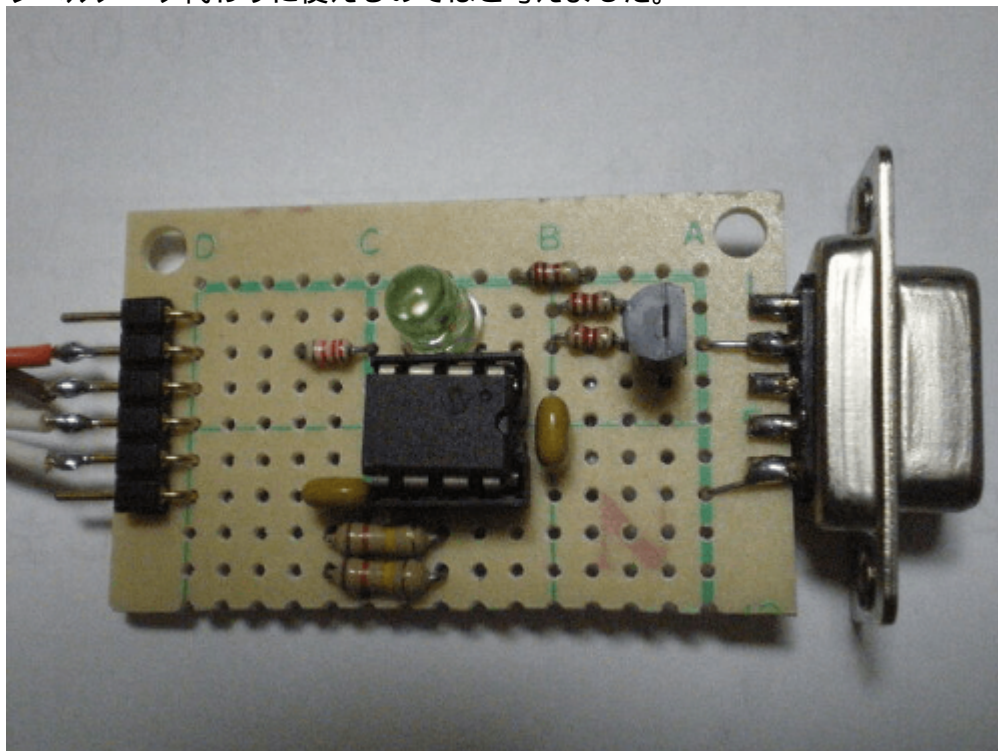


アナログデータロガー

概要

とても簡易（ハードウェアもソフトウェアも共に）なデータロガーです。アナログデータ（2チャンネル）を約1秒周期で収集しRS232C経由でPCに送るものです。アナログデータは、10ビット（1024）の精度でA/D変換されます。尚、1ビットあたりは約5mVの精度となります（ $5V \div 1024$ ）PCに送る際には、バー表示で送るモードと数値表示で送るモードの選択がSWで出来ます。バー表示はちょっとしたレベルメータ代わりに使えるのではと考えました。



回路図



```
void interrupt(){
    if (INTCON.T0IF == 1) {
        INTCON.T0IF = 0;
    }
    if (PIR1.TMR1IF == 1) {
        GPIO.F4 = ~GPIO.F4;
        PIR1.TMR1IF = 0;
    }
}

void initCcp()
{
    CCP1CON = 0b000001100;
    PR2 = 0xFF;
    T2CON = 0b000000101;
    CCPR1L = 0x00;
}
```

```
void    buzzer()
{
    CCPR1L = 0x7F;
    Delay_ms(200);
    CCPR1L = 0x00;
    Delay_ms(200);
    CCPR1L = 0x7F;
    Delay_ms(200);
    CCPR1L = 0x00;
    Delay_ms(200);
    CCPR1L = 0x7F;
    Delay_ms(200);
    CCPR1L = 0x00;
}

void    itoa(char *buf, int val)
{
    unsigned    a, b, c, d;
    a = val / 1000;
    b = (val - (a * 1000)) / 100;
    c = (val - (a * 1000) - (b * 100)) / 10;
    d = val - (a * 1000) - (b * 100) - (c * 10);
    buf[0] = '0' + a;
    buf[1] = '0' + b;
    buf[2] = '0' + c;
    buf[3] = '0' + d;
    buf[4] = 0;
}

void    Soft_Uart_Write_String(char *buf)
{
    int    len, i;
    len = strlen(buf);
    for (i = 0; i < len; i++) {
        INTCON.GIE = 0;
        Soft_Uart_Write(buf[i]);
        INTCON.GIE = 1;
    }
}

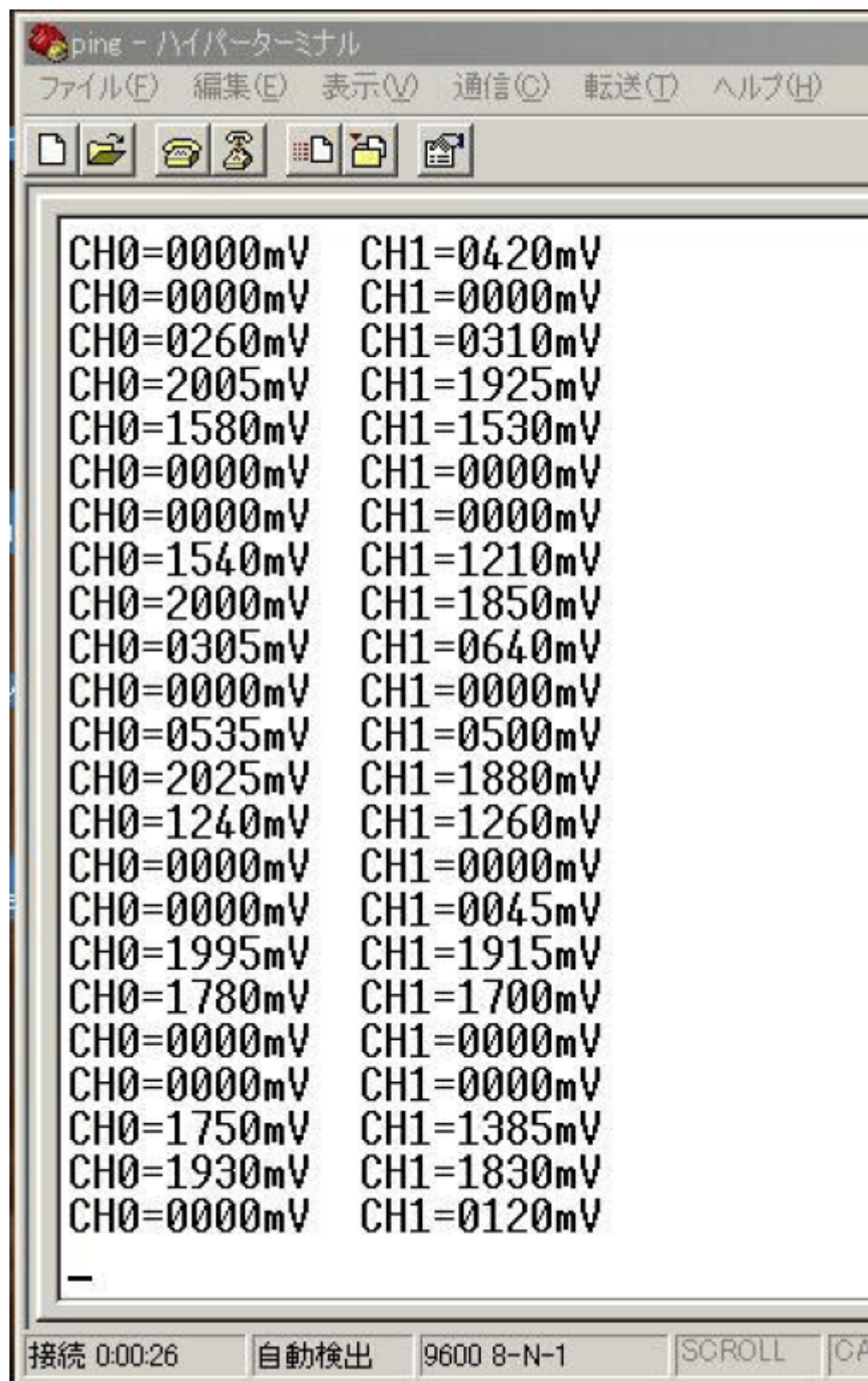
unsigned    char    buf[10];
unsigned    char    *text = "<A/D converter> R1\r\n";
unsigned    int    ad0, ad1, cnt;

void main()
{
    CMCON0 = 0b00000111;
    ANSEL = 0b00000011;
    TRISIO = 0b00001011;
    OSCCON = 0b01110000;
    OPTION_REG = 0b10000111;
```

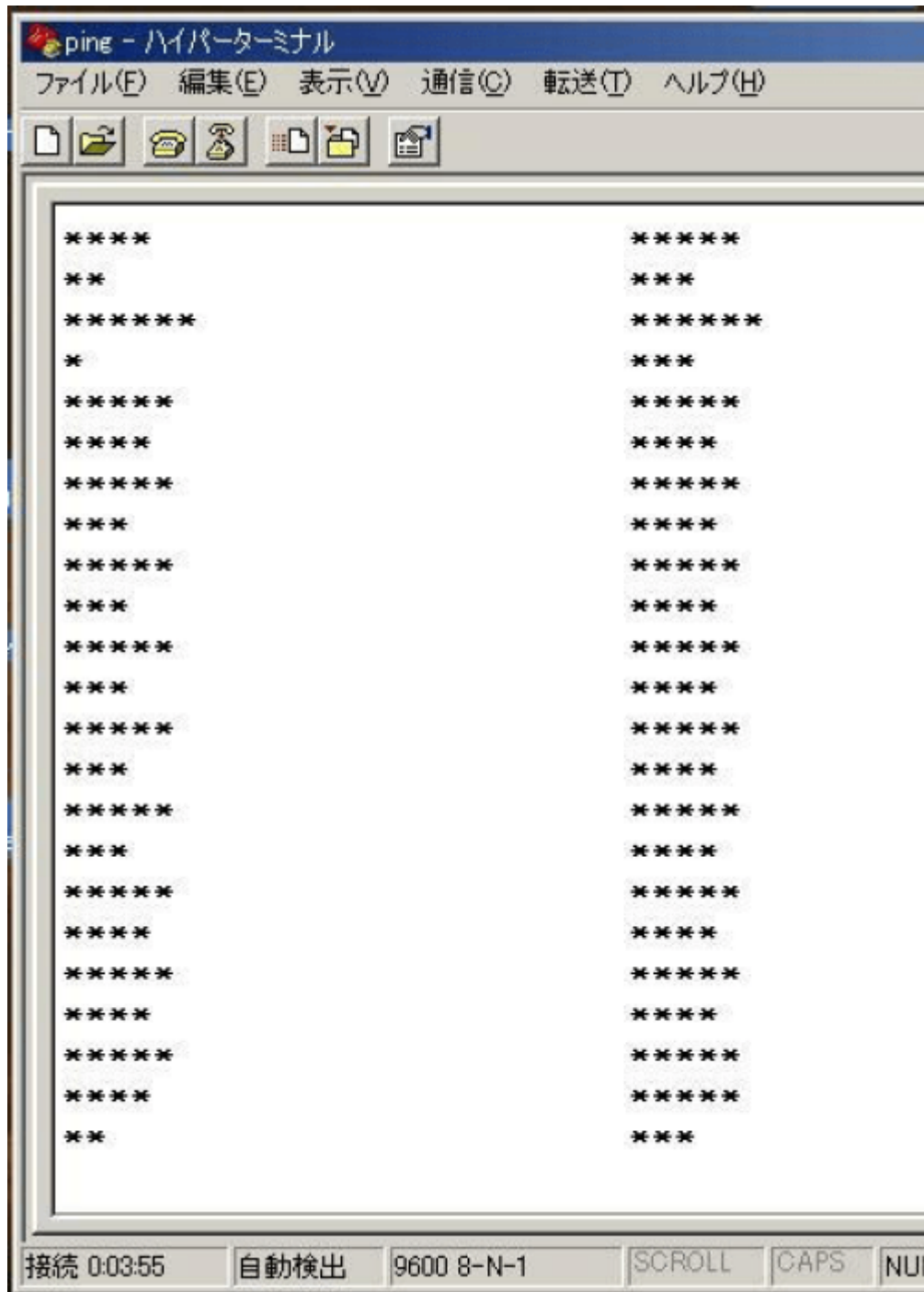
```
PIE1.TMR1IE = 1;
PIR1.TMR1IF = 0;
T1CON = 0b00110001;
INTCON = 0b11100000;
initCcp();
buzzer();
Soft_Uart_Init(GPIO, 3, 5, 9600, 0);
Soft_Uart_Write_String(text);
while (1) {
    ad0 = Adc_Read(0);
    ad1 = Adc_Read(1);
    if (GPIO.F3 == 0) {
        for (cnt = 0; cnt < (ad0 / 50); cnt++)
            Soft_Uart_Write_String("*");
        for (cnt = 0; cnt < ((1023 / 50) - (ad0 / 50) + 5); cnt++)
            Soft_Uart_Write_String(" ");
        for (cnt = 0; cnt < (ad1 / 50); cnt++)
            Soft_Uart_Write_String("*");
        Soft_Uart_Write_String("\r\n");
    } else {
        itoa(buf, ad0 * 5);
        Soft_Uart_Write_String("CH0=");
        Soft_Uart_Write_String(buf);
        Soft_Uart_Write_String("mV  CH1=");
        itoa(buf, ad1 * 5);
        Soft_Uart_Write_String(buf);
        Soft_Uart_Write_String("mV\r\n");
    }
    Delay_ms(1000);
}
}
```

動作確認

PCのハイパーターミナルを使って動作確認をしました。 <数値表示>



< バー表示 >



From:

<http://deepsky.jp/wiki/> - うごくといいな

Permanent link:

<http://deepsky.jp/wiki/doku.php?id=elechobby:picdic:pic12f683:2&rev=1588054335>

Last update: 2025/10/17 14:27

