

Forum: [16-bit PIC/dsPIC](#)

Topic: 演算法

Subject: Re: 演算法

作者: Ryang

2017年01月06日 11:49:06

補充一下，C18 有提供一些轉換函數如 itoa( )，atoi( ) ... 可以參考一下 W401 教育訓練課程裡的第六章節，如下圖所示：

實際應用 ADC 的讀值顯示電壓值，可以參考 "W301 Advance PICC Application" 裡的練習六的寫法，如下：

程式裡顯示電壓值到 mV 是怎樣做到的？

$\text{Disp\_Temp} = (((\text{unsigned long}) \text{Temp\_Buff}) * 4883) / 100 ;$

$5V / 1024 = 4.883 \text{ mV}$ ，要顯示 4.9mV 的精度

不要使用浮點數運算，只要用整數運算即可

```
Read_ADC(); // 讀取 ADC (VR2) 的轉換值
Temp_Buff= ADC_Result.Word & 0x3FF; // 將無用位元遮銷 0
Disp_Temp=(((unsigned long)Temp_Buff)*4883)/100; // 10-bit 解析度, 1
LSB=4.883mV

    DisplayDigit1(Disp_Temp/10000); // 顯示 V LCD 第一數字上
Disp_Temp=Disp_Temp%10000; // 取出餘數

    DisplayDigit2(Disp_Temp/1000 ); // 顯示 100mV
Disp_Temp=Disp_Temp%1000;

    DisplayDigit3(Disp_Temp/100); // 顯示 10mV
Disp_Temp=Disp_Temp%100;

    DisplayDigit4(Disp_Temp/10); // 顯示 mV
Disp_Temp=Disp_Temp%10;

    DisplayDigit5(Disp_Temp); // 顯示 0.1mV

    SEG_4DP=1 ; // 點小數點

    SEG_m = !SEG_m; // 顯示 mV 的符號
SEG_V= !SEG_V;
// SEG_BAT=!SEG_BAT;

Disp_Count = 20; //
```

擷取1.jpg(124.09 KB)



## 標準函數庫

### 數字與文字的轉換功能(二)

- **btoa()**

- 將8位元有號整數轉換成數字字串後存到指標指到的位址
- 例: 0x80 -->"-128", 100-->"100", 199-->"-57"

- **itoa()**

- 將16位元有號整數轉換成數字字串後存到指標指到的位址
- 例: 0x1000-->"4096", 1000-->"1000"

- **ltoa()**

- 將32位元有號整數轉換成數字字串後存到指標指到的位址

```
#include <p18c452.h>
#include <stdlib.h>

char string[12];
int code=-1000;

void main(void)
{
    itoa(code,string);
    while(1);
}
```

擷取.JPG(123.73 KB)

# 標準函數庫

## 數字與文字的轉換功能(一)

- **atob()** :

- 將指標指到的數字字串轉換成8位元的有號整數
- 例: “100” --> 0x64 , ”255”-->0xFF
- ”-128”-->0x80 , ”-2”-->0xFE

- **atoi()**

- 將指標指到的數字字串轉換成16位元的有號整數
- 例: “1000” --> 0x03E8=1000 , ”-1000”-->0xFC18=-1000

- **atol()**

- 將指標指到的數字字串轉換成32位元的有號整數
- 例: “1234567890” --> 0x0499602D2
- “-1234567890” --> 0xB669FD2E