

第5章 显示数值

前面章节说明了如何在屏幕上显示字符串。当然你也可以用字符串和标准C库函数显示数值。

然而，有时候这会是一件困难的事。通常最容易（最有效）的是调用一个函数显示你所需要结构的数值。 μ C/GUI提供了不同的十进制，十六进制和二进制输出函数。在这一章对这些函数进行单独的说明。

所有的函数不需要使用浮点库，并对速度和大小进行了优化。当然“Sprintf”可以用于任何系统。使用本章所述的函数有时能在ROM的存储空间和执行时间上进行精简。

5.1 数值API

下表列出了与数值处理相关的函数，在各自的类型中按字母顺序进行排列。函数的详细描述后面列出。

函 数	说 明
显示十进制数值	
GUI_DispDec()	在当前坐标显示指定数量字符的十进制数值
GUI_DispDecAt()	在指定坐标显示指定数量字符的十进制数值
GUI_DispDecMin()	在当前坐标显示最少数量字符的十进制数值
GUI_DispDecShift()	在当前坐标显示指定数量字符的长型十进制数值
GUI_DispDecSpace()	在当前坐标显示指定数量字符的十进制数值，用空格代替首位的0
GUI_DispsDec()	在当前坐标显示指定数量字符的十进制数值及显示符号
GUI_DispsDecShift()	在当前坐标显示指定数量字符的长型十进制数值及显示符号
显示浮点数值	
GUI_DispFloat()	在当前坐标显示指定数量字符的浮点数值
GUI_DispFloatFix()	显示浮点数值，指定小数点右边数字数量
GUI_DispFloatMin()	在当前坐标显示最少数量字符的浮点数值
GUI_DispsFloatFix()	显示浮点数值，指定小数点右边数字数量及显示符号
GUI_DispsFloatMin()	在当前坐标显示最少数量字符的浮点数值及显示符号
显示二进制数值	
GUI_DispBin()	在当前坐标显示二进制数值
GUI_DispBinAt()	在指定坐标显示二进制数值
显示十六进制数值	
GUI_DispHex()	在当前坐标显示十六进制数值
GUI_DispHexAt()	在指定坐标显示十六进制数值

5.2 显示十进制数值

GUI_DispDec()

描述

在当前视窗的当前文本坐标，使用当前字体显示一个十进制数值，指定显示字符的数量。

函数原型

```
void GUI_DispDec(I32 v, U8 Len);
```

参 数	含 意
<code>v</code>	用于显示的数值。 最小值为：-2147483648 (-2^{31}) 最大值为：2147483647 ($2^{31}-1$)
<code>Len</code>	显示的数字的数量（最大为9）

附加信息

不支持首位为0的格式（如0）。

如果数值为负，则会显示一个减号。

范例

// 以分秒的格式显示时间

```
GUI_DisString("Min:");
GUI_DisDec (Min, 2);
GUI_DisString("Sec:");
GUI_DisDec (Sec, 2);
```

相关主题

`GUI_DisSDec()`，`GUI_DisDecAt()`，`GUI_DisDecMin()`，`GUI_DisDecSpace()`

GUI_DisDecAt()

描述

在当前视窗的当前文本坐标，使用当前字体显示十进制数值，指定显示字符的数量。

函数原型

```
void GUI_DisDecAt(I32 v, I16P x, I16P y, U8 Len);
```

参 数	含 意
<code>v</code>	用于显示的数值。 最小值为：-2147483648 (-2^{31}) 最大值为：2147483647 ($2^{31}-1$)
<code>x</code>	写入客户视窗的X坐标（以像素为单位）
<code>y</code>	写入客户视窗的Y坐标（以像素为单位）
<code>Len</code>	显示的数字的数量（最大为9）

附加信息

不支持首位为0的格式。如果数值为负，则会显示一个减号。

范例

```
// 在左上角更新秒
GUI_DisDecAT(Sec, 200, 0, 2);
```

相关主题

GUI_DisDec(), GUI_DisDecSpace(), GUI_DisDecMin(), GUI_DisDecSpace()

GUI_DisDecMin()

描述

在当前视窗的当前文本坐标，使用当前字体显示十进制数值。不需要指定长度；自动使用最小的长度值。

函数原型

```
void GUI_DisDecMin(I32 v);
```

参 数	含 意
v	用于显示的数值。 最小值为：-2147483648 (-2^{31}) 最大值为：2147483647 ($2^{31}-1$) 能显示的数字的最大数量为9。

附加信息

如果数值必须要对齐，但是数字的数量不一样，使用该函数不是一个好的选择，应该使用一个能够指定数字数量的函数。

范例

```
// 显示结果
GUI_DisString("The result is :");
GUI_DisDecMin(Result);
```

相关主题

GUI_DispDec(), GUI_DispDecAt(), GUI_DispSDec(), GUI_DispDecSpace()

GUI_DispDecShift()**描述**

在当前视窗的当前文本坐标，使用当前字体显示一个长型十进制数值（用小数点作分隔符），指定显示字符的数量及使用小数点。

```
void GUI_DispDecShift(I32 v, U8 Len, U8 Shift);
```

参 数	含 意
v	用于显示的数值。 最小值为：-2147483648 (-2^{31}) 最大值为：2147483647 ($2^{31}-1$)
Len	显示的数字的数量（最大为9）
Shift	小数点右边数字的数量

附加信息

注意显示的最大字符数量为9（包括符号及小数点）

GUI_DispDecSpace()**描述**

在当前视窗的当前文本坐标，使用当前字体显示一个十进制数值，禁止首位的0（用空格代换）。

函数原型

```
void DispDecSpace(I32 v, U8 MaxDigits);
```

参 数	含 意
v	用于显示的数值。 最小值为：-2147483648 (-2^{31}) 最大值为：2147483647 ($2^{31}-1$)

MaxDigits	显示的数字数量，包括首位空格。 最大显示的数字的数量为9（包括首位空格）
-----------	---

附加信息

如果数值必须要对齐，但是数字的数量不一样，使用该函数是一个好的选择。

范例

// 显示结果

```
GUI_DisString("The result is :");
GUI_DisDecSpace(Result, 200);
```

相关主题

GUI_DisDec(), GUI_DisDecAt(), GUI_DisSDec(), GUI_DisDecMin()

GUI_DisSDec()

描述

在当前视窗的当前文本坐标，使用当前字体显示一个十进制数值（包括符号），并指定显示字符的数量。

函数原型

```
void GUI_DisSDec(I32 v, U8 Len);
```

参 数	含 意
V	用于显示的数值。 最小值为：-2147483648 (-2^{31}) 最大值为：2147483647 ($2^{31}-1$)
Len	显示的数字的数量（最大为9）

附加信息

不禁止首位为0的格式。

该函数与GUI_DisDec类似，但是在显示数值的前面总带有符号，即使这个数值是正的。

相关主题

GUI_DispDec(), GUI_DispDecAt(), GUI_DispDecMin(), GUI_DispDecSpace()

GUI_DispSDecShift()

描述

在当前视窗的当前文本坐标，使用当前字体显示一个长型十进制数值（包括符号，用小
数点作分隔符），指定数字的数量及使用小数点。

函数原型

```
void GUI_DispSDecShift(I32 v, U8 Len, U8 Shift);
```

参 数	含 意
<code>v</code>	用于显示的数值。 最小值为：-2147483648 (-2^{31}) 最大值为：2147483647 ($2^{31}-1$)
<code>Len</code>	显示的数字的数量（最大为9）
<code>Shift</code>	小数点右边数字的数量

附加信息

在数值前面总带有一个符号。注意显示的最大字符数量为9（包括符号及小数点）。

范例

```
void DemoDec(void)
{
    long l = 12345;
    GUI_Clear();
    GUI_SetFont(&GUI_Font8x8);
    GUI_DispStringAt("GUI_DispDecShift:\n", 0, 0);
    GUI_DispSDecShift(l, 7, 3);
    GUI_SetFont(&GUI_Font6x8);
    GUI_DispStringAt("Press any key", 0, GUI_VYSIZE-8);
    WaitKey();
}
```

下图为上面范例程序运行结果的屏幕截图



5.3 显示浮点数

GUI_DispFloat()

描述

在当前视窗的当前文本坐标，使用当前字体显示一个浮点数，指定显示字符数量。

函数原型

```
void GUI_DispFloat(float v, char Len);
```

参 数	含 意
<code>v</code>	用于显示的数值。 最小值为：1.2E-38 最大值为：3.4E38
<code>Len</code>	显示的数字的数量（最大为9）

附加信息

不支持首位为0的格式。小数点当作一个字符处理。如果数值为负数，会显示一个减号。

范例

```
/* 浮点数显示的所有特点的演示 */
void DemoFloat(void)
{
    float f = 123.45678;
    GUI_Clear();
    GUI_SetFont(&GUI_Font8x8);
    GUI_DisStringAt("GUI_DispFloat:\n", 0, 0);
    GUI_DispFloat(f, 9);
}
```



```
GUI_GotoX(100);
GUI_DisFloat (-f, 9);
GUI_DisStringAt("GUI_DisFloatFix:\n", 0, 20);
GUI_DisFloatFix (f, 9, 2);
GUI_GotoX(100);
GUI_DisFloatFix (-f, 9, 2);
GUI_DisStringAt("GUI_DisSFloatFix:\n", 0, 40);
GUI_DisSFloatFix (f, 9, 2);
GUI_GotoX(100);
GUI_DisSFloatFix (-f, 9, 2);
GUI_DisStringAt("GUI_DisFloatMin:\n", 0, 60);
GUI_DisFloatMin (f, 3);
GUI_GotoX(100);
GUI_DisFloatMin (-f, 3);
GUI_DisStringAt("GUI_DisSFloatMin:\n", 0, 80);
GUI_DisSFloatMin (f, 3);
GUI_GotoX(100);
GUI_DisSFloatMin (-f, 3);
GUI_SetFont(&GUI_Font6x8);
GUI_DisStringAt("Press any key", 0, GUI_VYSIZE-8);
WaitKey();
}
```

下图为上面范例程序运行结果的屏幕截图



```
GUI_DisFloat:
123.45678      -123.4568
GUI_DisFloatFix:
000123.46     -00123.46
GUI_DisSFloatFix:
+00123.46     -00123.46
GUI_DisFloatMin:
123.457      -123.457
GUI_DisSFloatMin:
+123.457     -123.457

Press any key
```

GUI_DisFloatFix()

描述

在当前视窗的当前文本坐标，使用当前字体显示一个浮点数，指定总的显示字符的数量及小数点右边字符的数量。

函数原型

```
void GUI_DisFloatFix (float v, char Len, char Decs);
```

参 数	含 意
v	用于显示的数值。 最小值为：1.2E-38 最大值为：3.4E38
Len	显示的所有数字的数量（最大为9）
Decs	小数点右边数字的数量

附加信息

不禁止首位为0的格式。如果数值为负，则显示一个减号。

GUI_DisFloatMin()

描述

在当前视窗的当前文本坐标，使用当前字体显示一个浮点数，小数点右边十进制数的数量为一个最小值。

函数原型

```
void GUI_DisFloatMin(float f, char Fract);
```

参 数	含 意
v	用于显示的数值。 最小值为：1.2E-38 最大值为：3.4E38
Fract	显示字符的最小数量

附加信息

不支持首位为0的格式。如果数值为负，则显示一个减号。

长度不需要指定，自动使用最小的长度。如果数值必须要对齐，但是数字的数量不一样，使用该函数不是一个好的选择，应该使用一个能够指定数字数量的函数。

GUI_DispSFloatFix()

描述

在当前视窗的当前文本坐标，使用当前字体显示一个浮点数（包括符号），指定总的显示字符的数量及小数点右边字符的数量。

函数原型

```
void GUI_DispSFloatFix(float v, char Len, char Decs);
```

参 数	含 意
v	用于显示的数值。 最小值为：1.2E-38 最大值为：3.4E38
Len	显示的所有数字的数量（最大为9）
Decs	小数点右边数字的数量

附加信息

不禁止首位为0的格式。数值的前面总会显示一个符号。

GUI_DispSFloatMin()

描述

在当前视窗的当前文本坐标，使用当前字体显示一个浮点数（包括符号），小数点右边数字使用最小数量。

函数原型

```
void GUI_DispSFloatMin(float f, char Fract);
```

参 数	含 意
v	用于显示的数值。 最小值为：1.2E-38 最大值为：3.4E38
Fract	显示数字的最小数量

附加信息

不支持首位为0的格式

数值的前面总会显示一个符号。

长度不需要指定，自动使用最小的长度。如果数值必须要对齐，但是数字的数量不一样，使用该函数不是一个好的选择，应该使用一个能够指定数字数量的函数。

5.4 显示二进制数值

GUI_DispBin()

描述

在当前视窗的当前文本坐标，使用当前字体显示一个二进制数。

函数原型

```
void GUI_DispBin(U32 v, U8 Len);
```

参 数	含 意
<code>v</code>	用于显示的数值，32位。
<code>Len</code>	显示的数字的数量（包括首位的0）

附加信息

与十进制及十六进制一样，最低有效位在最右边。

范例

```
// 显示二进制数“7”，结果为：000111
U32 Input = 0x7;
GUI_DispBin(Input, 6);
```

相关主题

GUI_DispBinAt()

GUI_DispBinAt()

描述

在当前视窗的指定的文本坐标处，使用当前字体显示一个二进制数。

函数原型

```
void DispBinAt(U32 v, I16P y, I16P x, U8 Len);
```

参 数	含 意
<code>v</code>	用于显示的数值，32位。
<code>x</code>	写入客户视窗的X轴坐标（以像素为单位）
<code>y</code>	写入客户视窗的Y轴坐标（以像素为单位）
<code>Len</code>	显示的数字的数量（包括首位的0）

附加信息

十进制及十六进制一样，最低有效位在最右边。

```
// 显示二进制输入状态
```

```
GUI_DispBinAt(Input, 0, 0, 8);
```

相关主题

GUI_DispBin(), GUI_DispHex()

5.5 显示十六进制数值

GUI_DispHex()

描述

在当前视窗的当前文本坐标，使用当前字体显示一个十六进制数值。

函数原型

```
void GUI_DispHex(U32 v, U8 Len);
```

参 数	含 意
<code>v</code>	用于显示的数值，16位。

Len	显示的数字的数量
-----	----------

附加信息

与十进制及二进制一样，最低有效位在最右边。

范例

```
/* 显示一个AD转换器的数值 */
```

```
GUI_DispHex(Input, 4);
```

相关主题

GUI_DispDec(), GUI_DispBin(), GUI_DispHexAt()

GUI_DispHexAt()

描述

在当前视窗的指定的文本坐标处，使用当前字体显示一个十六进制数。

函数原型

```
void GUI_DispHexAt(U32 v, I16P x, I16P y, U8 Len);
```

参 数	含 意
v	用于显示的数值，32位。
x	写入客户视窗的X轴坐标（以像素为单位）
y	写入客户视窗的Y轴坐标（以像素为单位）
Len	显示的数字的数量（包括首位的0）

附加信息

与十进制及二进制一样，最低有效位在最右边。

范例

```
// 在指定坐标显示一个AD转换器数值
```

```
GUI_DispHexAt(Input, 0, 0, 4);
```

相关主题

GUI_DispNet(), GUI_DispBin(), GUI_DispHex()

5.6 μ C/GUI 的版本

GUI_GetVersionString()

描述

返回一个字符串包含当前 μ C/GUI的版本

函数原型

```
const char * GUI_GetVersionString(void);
```

范例

```
// 在当前光标位置显示当前版本  
GUI_DispNet(GUI_GetVersionString());
```