**Qt 之 QLCDNumber**

**简述**

QLCDNumber控件用于显示一个LCD数字。

它可以显示几乎任意大小的数字。可以显示十进制、十六进制、八进制或二进制数。很容易使用display()槽连接到数据源，这个槽可以被任何五个参数类型的数据源重载。

当显示范围以外的数据时，就会发射overflow()信号。返回可以通过setDigitCount()来设置，但是setSmallDecimalPoint() 也会影响它。如果显示设置为十六进制、八进制或二进制，就会显示整数等效值。

这些数字和其它符号可以显示：0/O、1、 2、 3、 4、5/S、 6、7、 8、9/g、负号、小数点、A、 B、 C、D、E、F、h、H、L、o、P、r、u、U、Y、冒号、度数符号（字符串中被指定为单引号）和空格。 QLCDNumber会将非法字符替代为空格。

获取QLCDNumber的内容是不可能的，虽然可以通过value()来获取（返回的double类型，如果显示其他格式怎么办）。所以如果真的需要文本，建议自己存储数据。

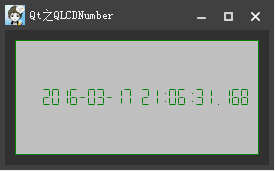
**|**版权声明：一去、二三里，未经博主允许不得转载。

**接口说明**

| **接口** | **描述** |
| --- | --- |
| setDigitCount(int numDigits) | 设置所显示的位数 |
| setBinMode() | 以二进制形式显示 |
| setOctMode() | 以八进制形式显示 |
| setHexMode() | 以十六进制形式显示 |
| setDecMode() | 以十进制形式显示（默认） |
| setSmallDecimalPoint(bool) | 其参数设置为true或者false，决定了小数点单独站一位空间还是在两个位之间。换句话说，如果参数为true，小数点将占用比平常更少的空间 |
| setSegmentStyle(SegmentStyle) | 改变现实数字的外观，包括：Outline、Filled、Flat |
| checkOverflow(double num) | 检查给定值是否可以在区域内显示（也会发射overflow()信号，可以将其连接到槽中处理） |

下面我们来实现一个更新时钟的效果。

**效果**



**QLCDNumber**

首先我们构建一个QLCDNumber对象，并设置显示位数、外观、样式等效果。

m\_pLCD = new QLCDNumber(this);

*// 设置能显示的位数*

m\_pLCD->setDigitCount(25);

*// 设置显示的模式为十进制*

m\_pLCD->setMode(QLCDNumber::Dec);

*// 设置显示外观*

m\_pLCD->setSegmentStyle(QLCDNumber::Flat);

*// 设置样式*

m\_pLCD->setStyleSheet("border: 1px solid green; color: green; background: silver;");

**定时器**

然后，构建一个定时器，每隔一秒来定时刷新QLCDNumber中的内容。

QTimer \*pTimer = new QTimer(this);

*// 设置定时间隔*

pTimer->setInterval(1000);

connect(pTimer, SIGNAL(timeout()), this, SLOT(onTimeOut()));

*// 启动定时器*

pTimer->start();

**槽函数**

获取当前日期时间，显示格式为：年-月-日 时:分:秒.毫秒。

void onTimeOut()

{

*// 获取系统当前时间*

QDateTime dateTime = QDateTime::currentDateTime();

*// 显示的内容*

m\_pLCD->display(dateTime.toString("yyyy-MM-dd HH:mm:ss.zzz"));

}