

NC 编辑器控件应用文档

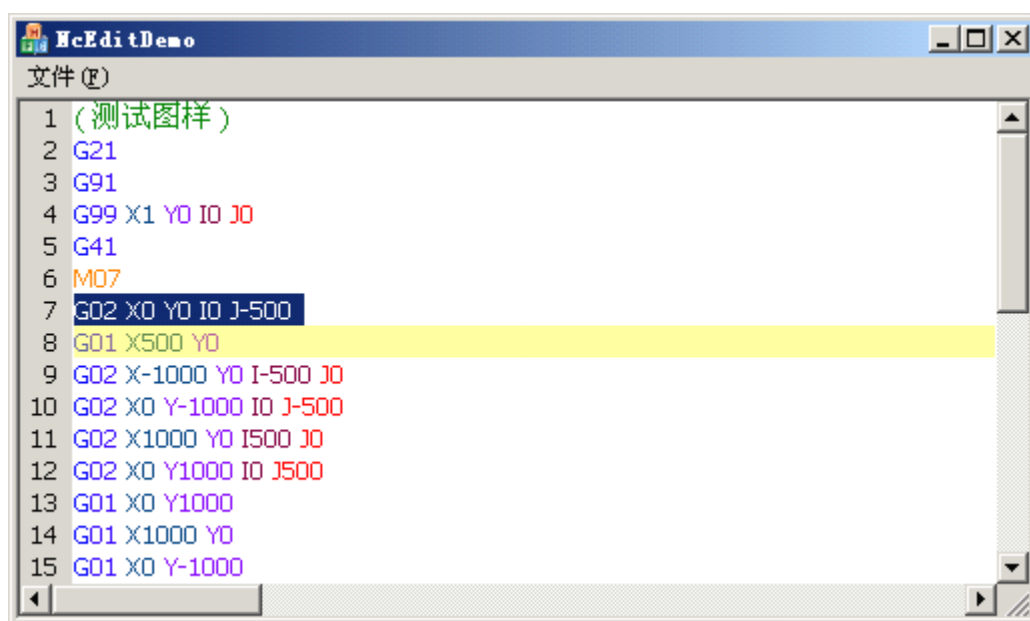
1. 概述

1.1 简介

这个 NC 编辑控件（以下简称“本控件”）基于开源项目 [Scintilla](#) 开发，它是一个简单的 Scintilla 封装类，类名为 CNcEditWnd。

Scintilla 是一款支持语法高亮的编辑控件，使用它，可以通过自己编写 Lexer 文件来添加对新的语言语法高亮的支持。

1.2 屏幕截图



2. 使用方法

2.1 添加文件到项目中

2.1.1 源代码文件

NcEditWnd.h
SciLexer.h

Scintilla.h
NcEditWnd.cpp

2.1.2 动态链接库文件

SciLexer.dll

2.2 包含 NcEditWnd.h

在准备作为 NC 编辑器窗体的对话框关联的类中包含 NcEditWnd.h，并定义变量：

```
CNcEditWnd *m_pNcEditWnd;
```

2.3 创建本控件

在该类的 OnCreate 函数中添加如下代码，用来创建本控件：

```
CRect rcWin; //本控件的位置以及本空间大小
//..
m_pNcEditWnd=new CNcEditWnd();
m_pNcEditWnd->Create(WS_EX_CLIENTEDGE,rcWin,this);
m_pNcEditWnd->InitNcEdit(12,"Tahoma");
m_pNcEditWnd->ShowWindow(SW_SHOW);
```

2.4 打开 NC 程序文件

```
m_pNcEditWnd->OpenFile("文件路径");
```

2.5 保存 NC 程序文件

```
m_pNcEditWnd->SaveFile("文件路径");
```

2.6 在该类中添加 OnNotify 函数，并添加处理程序

```
struct SCNotification* scn = (struct
SCNotification*)lParam;
LPNMHDR pnmh = (LPNMHDR)lParam;
switch(pnmh->code)
{
```

```
case SCN_MODIFIED: //修改了文件
case SCN_ZOOM:      //放大, 缩小
    m_pNcEditWnd->UpdateStyle();
    break;
case SCN_UPDATEUI: //界面更新(单击鼠标, 按下箭头等)
    int n=m_pNcEditWnd->GetCurrentLine();
    //当前行
    break;
}
```

2.7 在该类的析构函数中销毁该控件

```
if (m_pNcEditWnd!=NULL){
    delete m_pNcEditWnd;
}
```

3.CNcEditWnd 类参考

```
class CNcEditWnd : public CWnd
{
public:
    /**
     * 构造时加载 Scintilla.dll
     * @param void
     * @return
     * @see
     */
    CNcEditWnd(void);
    /**
     * 释放 Scintilla.dll
     * @param void
     * @return
     * @see
     */
    ~CNcEditWnd(void);
protected:
```

```

DECLARE_MESSAGE_MAP()

public:

    /**
     * 创建 NC 编辑器窗口。
     * @param DWORD dwExStyle: 窗口样式
     * @param const RECT & rect: 位置与大小
     * @param CWnd * pParentWnd: 父窗口
     * @return BOOL: 成功与否
     * @see
     */
    virtual BOOL Create(DWORD dwExStyle, const RECT& rect,
CWnd* pParentWnd);

    /**
     * 初始化编辑器窗口的各种属性。
     * @param int nSize: 字号
     * @param const TCHAR * face: 字体名称
     * @return void
     * @see
     */
    void InitNcEdit(int nSize, const TCHAR* face);

    /**
     * 打开 NC 程序
     * @param LPCTSTR lpFileName
     * @return BOOL
     * @see
     */
    BOOL OpenFile(LPCTSTR lpFileName);

    /**
     * 返回当前光标所在的行的行号
     * @param void
     * @return UINT
     * @see
     */

```

```
UINT GetCurrentLine(void);

/**
 * 刷新窗口样式，在父窗口的 OnNotify 中根据需要调用他。
 * @param void
 * @return void
 * @see
 */
void UpdateStyle(void);

/**
 * 自动换行
 * @param BOOL bWrapped
 * @return void
 * @see
 */
void SetWrap(BOOL bWrapped);

/**
 * 显示行尾标志
 * @param BOOL bShow
 * @return void
 * @see
 */
void ShowLineEndings(BOOL bShow);

/**
 * 将指定的行设置为选中状态
 * @param int nLine
 * @return void
 * @see
 */
void SelectLine(int nLine);

/**
 * 保存文件。 \n
 * 若不指定 strFileName 或者 strFileName 为空，则按原文件名存储。
```

```

    * @param CString & strFileName
    * @return BOOL
    * @see
    */
    BOOL SaveFile(CString strFileName=_T("")) ;

private:
    HMODULE _hSciLexerDll;
    CString _strFileName;
private:
    void SetDefaultColorFont(int nSize,const TCHAR* face);
    void UpdateLineNumberWidth(void);
    void SetNCEditStyle(void);
};

```