

emwin 2-day quick tutorial 001_多个页面窗口切换

用 emWin 做界面, 可以简单理解为每个 C 文件即对应一个页面, 如果要设计多个页面的界面怎么办? 生成多个界面 C 文件并在回调函数的操作触发消息中用隐藏/显示页面函数: WM_HideWindow() 和 WM_ShowWindow() 实现界面切换程序即可.

当界面有多个页面需要切换显示时, 有两种方式可以实现: (1) 隐藏/显示方式: 程序启动时创建所有页面窗口, 程序运行时隐藏当前无需显示的页面, 同时显示当前需要显示的页面; (2) 删除/创建方式: 程序启动时创建开机需要显示的页面窗口, 程序运行时删除当前无需显示的页面, 同时创建当前需要显示的页面.

这两种方式的根本区别是: 第(1)种方式由于所有页面都是真实存在的, 因此占用更多的内存. 对于界面简单的应用, 或者 MCU 有足够 RAM 的情况下, 建议使用隐藏/显示方式, 因为删除/创建方式涉及到动态内存的申请和释放, 处理不当或程序不严谨容易出现内存错误造成死机现象, 而且删除/创建方式由于切换页面时需要创建窗口, 因此切换页面时速度比隐藏/显示方式更慢些.

通过隐藏/显示的方式做页面窗口切换, 使用到 emwin 两个窗口管理 API 函数:

```
void WM_HideWindow(WM_HWIN hWin); //隐藏窗口函数, hWin 是窗口的句柄
```

```
void WM_ShowWindow(WM_HWIN hWin); //显示窗口函数, hWin 是窗口的句柄
```

通过删除/创建的方式做页面窗口切换, 使用到 emwin 两个窗口管理 API 函数:

```
void WM_DeleteWindow(WM_HWIN hWin);
```

```
WM_HWIN CreatePage2(void); //即 1 个页面的创建函数, GUIBuilder 会自动生成
```

```
//操作触发消息处理(操作屏幕程序会跑到这里), 比如点击按键、点击编辑框(任何的操作)等等.....
```

```
//touch event message(The program runs here when operating the touch screen.), for example, click on the button, etc.
```

```
case WM_NOTIFY_PARENT:
```

```
Id = WM_GetId(pMsg->hWinSrc);
```

```
NCode = pMsg->Data.v;
```

```
switch(Id)
```

```
{
```

```
case ID_BUTTON_0:
```

```
switch(NCode)
```

```
{
```

```
case WM_NOTIFICATION_CLICKED://BUTTON_0 按下动作消息 BUTTON_0 pressed message
```

```
break;
```

```
case WM_NOTIFICATION_RELEASED://BUTTON_0 弹起动作消息 BUTTON_0 released message
```

```
//隐藏/显示方式
```

```
WM_HideWindow(hPage[1]); //隐藏 Page1
```

```
WM_ShowWindow(hPage[2]); //显示 Page2
```

```
WM_SetFocus(hPage[2]); //emwin will focus on the original widget
```

```
//删除/创建方式
```

```
WM_DeleteWindow(hPage[1]); hPage[1] = 0;
```

```
if(!hPage[2]) hPage[2] = CreatePage2(hPage[0]);
```

```
WM_ShowWindow(hPage[2]);
```

```
WM_SetFocus(hPage[2]); //emwin will focus on the original widget
```

```
break;
```

```
}
```

```
break;
```

```
}
```

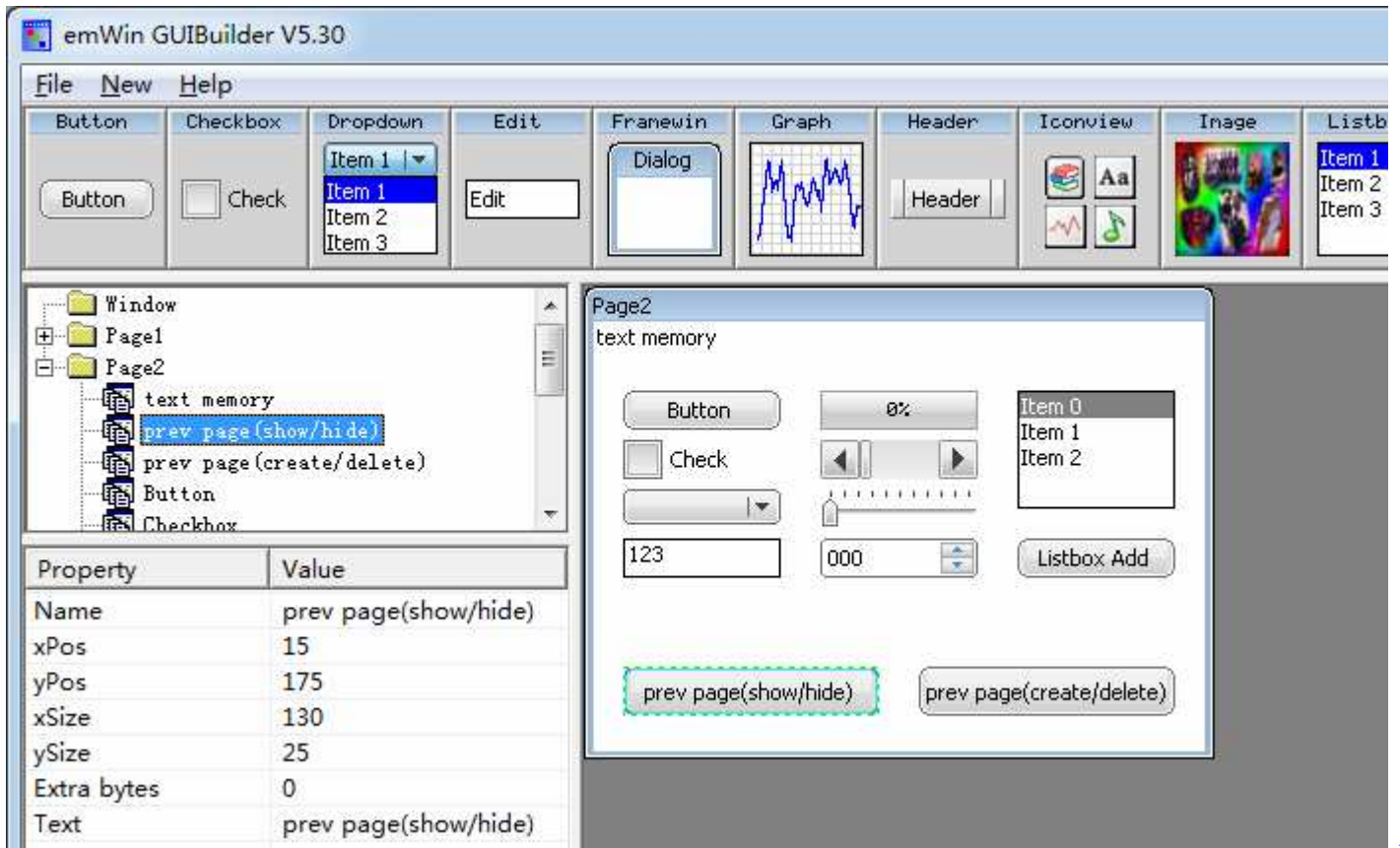
```
break;
```

```
.....
```

这几个函数使用比较简单, 看源代码即可. 值得注意的是, 在创建窗体时最好先用 if(!hPage[2]) 判断句柄是否为 0, 在删除窗体后将句柄清 0, 这样做的好处是避免因为某种主观忽略或客观的原因在没删除窗体的情况下又重复创建该窗体, 比如使用实体按键因为抖动等原因.

在源代码中还加入了实时监控内存 RAM 的使用情况, 防止删除/创建窗体处理不当造成内存错误, 不断切换窗口时如果占用内存不断增加, 即可断定是哪里搞错了.

页面加入了 Listbox 控件, 当连续点击” Listbox Add” 按键时, Listbox 会连续增加一行, emWin 占用的内存会不断增加.



小

技巧:放置 xxxx 控件之后不要用鼠标移动,用上下左右键移动更容易对齐(步进是 5).

备注:GUIBuilder 生成的 c 文件,用户代码最好加在“USER START”和“USER END”之间,其他地方除了数字以外,不要做任何修改,否则 GUIBuilder 将无法再次打开此 c 文件;另外, GUIBuilder 再次打开编辑并保存时,“USER START”和“USER END”之间的内容是不会被更改的;还有 c 文件不要在 GUIBuilder 打开状态去直接修改 c 代码,否则点 GUIBuilder 保存之后修改的内容将会丢失。

