

emwin 2-day quick tutorial 013_屏幕无操作进入待机状态的实现方法

(1) 在一些操作界面应用中, 需要实现待机功能, 当长时间不操作触摸屏时, 设备将进入待机模式, 比如关闭功耗较大的模块, 关闭背光等等...

(2) 示例中利用一个透明/灰色的 Window 窗体做为待机画面, 在这个待机 Window 窗体中放置一个定时器, 设定该定时器每 100MS 触发一次, 触发时 TimeD 变量加 1, 检测到点击触摸屏时 TimeD 清零, 当 TimeD 等于 30 时(即 3 秒), 触发待机事件(示例中只是用 WM_ShowWindow() 函数显示待机 Window 窗体并隐藏其他窗体), 其他事件用户自行添加.

(3) WindowStandbyDLG.c 下的 #define WSBY_transE (0) 宏定义用于设置在待机时待机窗体是灰色还是透明.

(4) 值得说明的是: GUI_PID_STATE statePID; 和 GUI_PID_GetCurrentState(&statePID); GUI_PID_STATE 是 PID 输入设备(比如键盘、鼠标、触摸屏等等)状态数组, GUI_PID_GetCurrentState() 是获取当前 PID 状态函数, 在一些特殊界面应用中这是非常有用的, 务必记住有这么一个东西:

26.2 Pointer input device API

The table below lists the pointer input device routines in alphabetical order. Detailed descriptions follow.

Note: This API is used by the PID-driver; if you use a PID-driver shipped with emWin, your code does not need to call these routines.

Routine	Description
GUI_PID_GetCurrentState()	Returns the most recently stored state from the PID.
GUI_PID_GetState()	Returns the state of the PID.
GUI_PID_IsEmpty()	Returns if the PID buffer is empty.
GUI_PID_IsPressed()	Returns if the most recent state of the PID is pressed.
GUI_PID_StoreState()	Stores the current state of the PID.

Table 26.1: PID API list

Data structure

The structure of type GUI_PID_STATE referenced by the parameter `pState` is filled by the routine with the current values. The structure is defined as follows:

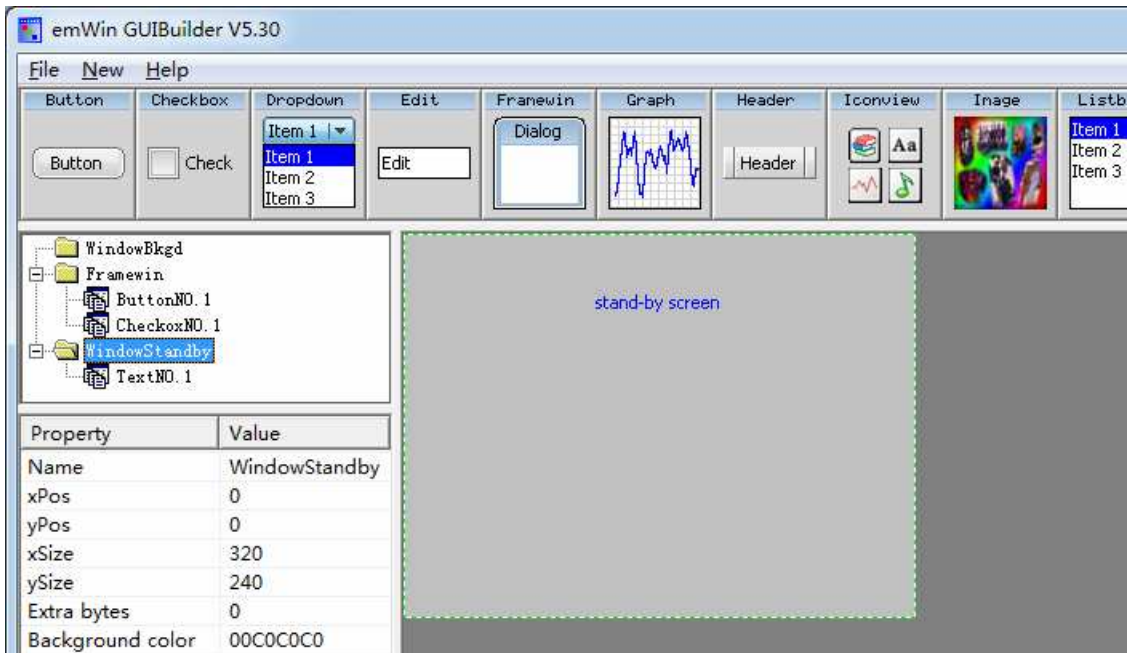
```
typedef struct {
    int x, y;
    U8 Pressed;
    U8 Layer;
} GUI_PID_STATE;
```

Elements of structure GUI_PID_STATE

Data type	Element	Description
int	x	X position of pointer input device.
int	y	Y position of pointer input device.
U8	Pressed	If using a touch screen this value can be 0 (unpressed) or 1 (pressed). If using a mouse bit 0 is used for the pressed state of the left button and bit 1 for the right button. The bits are 1 if the button is pressed and 0 if not.
U8	Layer	Describes the layer from which the PID state has been received

Table 26.2: GUI_PID_STATE element list

使用 GUIBuilder V5.30 创建如下页面并保存为 c 文件, 然后添加到 CodeBlocks 工程, 其中 WindowBkgd 是作为桌面背景用的, WindowStandby 是进入待机状态后的页面:



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”之间的内容将不会被更改; 还有 c 文件不要在 GUIBuilder 打开状态直接去修改 c 代码, 否则点 GUIBuilder 保存之后修改的内容将会丢失.

