这一节我们来学习更多的关于系统的学习

# 学习大纲

|  |
| --- |
|  |

## GetSystemDirectory语法

|  |
| --- |
| UINT GetSystemDirectoryA(  [out] LPSTR lpBuffer,  [in] UINT uSize  ); 参数 [out] lpBuffer  指向要接收路径的缓冲区的指针。 除非系统目录是根目录，否则此路径不会以反斜杠结尾。 例如，如果系统目录在驱动器 C 上名为 Windows\System32，则此函数检索的系统目录路径为 C：\Windows\System32。  [in] uSize  缓冲区的最大大小（以 **TCHAR 为单位**）。 返回值 如果函数成功，则返回值是复制到缓冲区的字符串的长度（ **以 TCHAR 为单位**），不包括终止 null 字符。 如果长度大于缓冲区的大小，则返回值是保存路径所需的缓冲区大小，包括终止 null 字符。  如果函数失败，则返回值为零。 要获得更多的错误信息，请调用 GetLastError。 注解 应用程序不应在系统目录中创建文件。 如果用户运行的是共享版本的操作系统，则应用程序对系统目录没有写入访问权限。 示例 有关示例，请参阅 [获取系统信息](https://learn.microsoft.com/zh-cn/windows/desktop/SysInfo/getting-system-information)。  **备注**  sysinfoapi.h 标头将 GetSystemDirectory 定义为别名，该别名根据 UNICODE 预处理器常量的定义自动选择此函数的 ANSI 或 Unicode 版本。 将非特定编码别名的使用与非非特定编码的代码混合使用可能会导致不匹配，从而导致编译或运行时错误。 有关详细信息，请参阅 [**函数原型的约定**](https://learn.microsoft.com/zh-cn/windows/win32/intl/conventions-for-function-prototypes)。 |

## GetWindowsDirectory语法

|  |
| --- |
| UINT GetWindowsDirectoryA(  [out] LPSTR lpBuffer,  [in] UINT uSize  ); 参数 [out] lpBuffer  指向接收路径的缓冲区的指针。 除非 Windows 目录是根目录，否则此路径不会以反斜杠结尾。 例如，如果 Windows 目录在驱动器 C 上名为 Windows，则此函数检索到的 Windows 目录的路径为 C：\Windows。 如果系统安装在驱动器 C 的根目录中，则检索的路径为 C：。  [in] uSize  *lpBuffer* 参数指定的缓冲区的最大大小（以 **TCHAR 为单位**）。 此值应设置为 **MAX\_PATH**。 返回值 如果函数成功，则返回值是复制到缓冲区的字符串的长度（以 **TCHAR** 为单位），不包括终止 null 字符。  如果长度大于缓冲区的大小，则返回值是保存路径所需的缓冲区的大小。  如果函数失败，则返回值为零。 要获得更多的错误信息，请调用 GetLastError。 注解 Windows 目录是某些旧版应用程序在其中存储初始化和帮助文件的目录。 新应用程序不应将文件存储在 Windows 目录中;相反，它们应将系统范围的数据存储在应用程序的安装目录中，并将特定于用户的数据存储在用户的配置文件中。  如果用户运行的是共享版本的系统，则保证 Windows 目录对于每个用户都是私有的。  如果应用程序创建希望按用户存储的其他文件，则应将它们放在 HOMEPATH 环境变量指定的目录中。 如果管理员通过用户管理器管理工具指定，则每个用户的此目录将有所不同。 HOMEPATH 始终指定用户的主目录（保证每个用户是专用目录）或默认目录 (例如 C：\USERS\DEFAULT) 用户将拥有所有访问权限。  **终端服务：**如果应用程序在终端服务环境中运行，则每个用户都有一个专用的 Windows 目录。 系统还有一个共享的 Windows 目录。 如果应用程序是终端服务感知 (在映像标头) 中设置了 **IMAGE\_DLLCHARACTERISTICS\_TERMINAL\_SERVER\_AWARE** 标志，则此函数将返回系统 Windows 目录的路径，就像 [GetSystemWindowsDirectory](https://learn.microsoft.com/zh-cn/windows/desktop/api/sysinfoapi/nf-sysinfoapi-getsystemwindowsdirectorya) 函数一样。 否则，它将检索用户的专用 Windows 目录的路径。 示例 有关示例，请参阅 [获取系统信息](https://learn.microsoft.com/zh-cn/windows/desktop/SysInfo/getting-system-information)。  **备注**  sysinfoapi.h 标头将 GetWindowsDirectory 定义为别名，该别名根据 UNICODE 预处理器常量的定义自动选择此函数的 ANSI 或 Unicode 版本。 将非特定编码别名的使用与非非特定编码的代码混合使用可能会导致不匹配，从而导致编译或运行时错误。 有关详细信息，请参阅 [**函数原型的约定**](https://learn.microsoft.com/zh-cn/windows/win32/intl/conventions-for-function-prototypes)。 |

## GetUserName函数用法

|  |
| --- |
| BOOL GetUserNameA(  [out] LPSTR lpBuffer,  [in, out] LPDWORD pcbBuffer  ); 参数 [out] lpBuffer  指向用于接收用户登录名的缓冲区的指针。 如果此缓冲区的大小不足以包含整个用户名，则函数将失败。 缓冲区大小为 (UNLEN + 1) 字符将保存最大长度用户名，包括终止 null 字符。 UNLEN 在 Lmcons.h 中定义。  [in, out] pcbBuffer  输入时，此变量指定 *lpBuffer* 缓冲区的大小（以 **TCHAR 为单位**）。 在输出时，变量接收复制到缓冲区的 **TCHAR** 数，包括终止 null 字符。  如果 *lpBuffer* 太小，则函数将失败， [GetLastError](https://learn.microsoft.com/zh-cn/windows/desktop/api/errhandlingapi/nf-errhandlingapi-getlasterror) 将返回ERROR\_INSUFFICIENT\_BUFFER。 此参数接收所需的缓冲区大小，包括终止 null 字符。 返回值 如果函数成功，则返回值为非零值，*lpnSize* 指向的变量包含复制到 *lpBuffer* 指定的缓冲区的 **TCHAR** 数，包括终止 null 字符。  如果函数失败，则返回值为零。 要获得更多的错误信息，请调用 GetLastError。 注解 如果当前线程正在模拟另一个客户端， **则 GetUserName** 函数将返回该线程正在模拟的客户端的用户名。  如果从在“网络服务”帐户下运行的进程调用 **GetUserName** ，则 *lpBuffer* 中返回的字符串可能因 Windows 版本而异。 在 Windows XP 上，返回“NETWORK SERVICE”字符串。 在 Windows Vista 上，返回“<HOSTNAME>$”字符串。 示例 有关示例，请参阅 [获取系统信息](https://learn.microsoft.com/zh-cn/windows/desktop/SysInfo/getting-system-information)。  **备注**  winbase.h 标头将 GetUserName 定义为别名，该别名根据 UNICODE 预处理器常量的定义自动选择此函数的 ANSI 或 Unicode 版本。 将非特定编码别名与非非特定编码的代码混合使用可能会导致不匹配，从而导致编译或运行时错误。 有关详细信息，请参阅 [**函数原型的约定**](https://learn.microsoft.com/zh-cn/windows/win32/intl/conventions-for-function-prototypes)。 |

### 注意：这里有坑，第二个参数不能直接写常量定义定义一个变量，然后传递他的地址

## GetComputerName函数用法

|  |
| --- |
| BOOL GetComputerNameA(  [out] LPSTR lpBuffer,  [in, out] LPDWORD nSize  ); 参数 [out] lpBuffer  指向接收计算机名称或群集虚拟服务器名称的缓冲区的指针。 缓冲区大小应足够大，以包含MAX\_COMPUTERNAME\_LENGTH + 1 个字符。  [in, out] nSize  输入时，指定缓冲区的大小（以 **TCHAR 为单位**）。 输出时，复制到目标缓冲区的 **TCHAR** 数，不包括终止 null 字符。  如果缓冲区太小，函数会失败， [GetLastError](https://learn.microsoft.com/zh-cn/windows/desktop/api/errhandlingapi/nf-errhandlingapi-getlasterror) 将返回ERROR\_BUFFER\_OVERFLOW。 *lpnSize* 参数指定所需的缓冲区的大小，包括终止 null 字符。 返回值 如果函数成功，则返回值为非零值。  如果函数失败，则返回值为零。 要获得更多的错误信息，请调用 GetLastError。 注解 **GetComputerName** 函数检索在系统启动时建立的 NetBIOS 名称。 在用户重新启动计算机之前， [SetComputerName](https://learn.microsoft.com/zh-cn/windows/desktop/api/sysinfoapi/nf-sysinfoapi-setcomputernamea) 或 [SetComputerNameEx](https://learn.microsoft.com/zh-cn/windows/desktop/api/sysinfoapi/nf-sysinfoapi-setcomputernameexa) 函数所做的名称更改不会生效。  如果调用方在客户端会话下运行，则此函数返回服务器名称。 若要检索客户端名称，请使用 [WTSQuerySessionInformation](https://learn.microsoft.com/zh-cn/windows/desktop/api/wtsapi32/nf-wtsapi32-wtsquerysessioninformationa) 函数。 示例 有关示例，请参阅 [获取系统信息](https://learn.microsoft.com/zh-cn/windows/desktop/SysInfo/getting-system-information)。  **备注**  winbase.h 标头将 GetComputerName 定义为别名，该别名根据 UNICODE 预处理器常量的定义自动选择此函数的 ANSI 或 Unicode 版本。 将非特定编码别名的使用与非非特定编码的代码混合使用可能会导致不匹配，从而导致编译或运行时错误。 有关详细信息，请参阅 [**函数原型的约定**](https://learn.microsoft.com/zh-cn/windows/win32/intl/conventions-for-function-prototypes)。 |

### 注意：这里有坑，第二个参数不能直接写常量定义定义一个变量，然后传递他的地址，

## SysParametersInfo函数用法

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| C++复制  BOOL SystemParametersInfoA(  [in] UINT uiAction,  [in] UINT uiParam,  [in, out] PVOID pvParam,  [in] UINT fWinIni  ); 参数 [in] uiAction  类型： **UINT**  要检索或设置的系统范围参数。 可能的值按以下相关参数表进行组织：   * 辅助功能参数 * 桌面参数 * 图标参数 * 输入参数 * 菜单参数 * 电源参数 * 屏幕保护程序参数 * 超时参数 * UI 效果参数 * 窗口参数   下面是辅助功能参数。  展开表   |  |  | | --- | --- | | **辅助功能参数** | **含义** | | **SPI\_GETACCESSTIMEOUT**  0x003C | 检索有关与辅助功能关联的超时期限的信息。 *pvParam* 参数必须指向接收信息的 [ACCESSTIMEOUT](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-accesstimeout) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(ACCESSTIMEOUT)。 | | **SPI\_GETAUDIODESCRIPTION**  0x0074 | 确定是启用还是禁用音频说明。 *pvParam* 参数是指向 [AUDIODESCRIPTION](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-audiodescription) 结构的指针。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(AUDIODESCRIPTION)。  虽然有视觉障碍的用户可能会听到视频内容中的音频，但视频中有很多没有相应音频的操作。 视频中发生情况的特定音频说明可帮助这些用户更好地了解内容。 此标志使你能够确定是否启用了音频说明以及使用哪种语言。  **Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETCLIENTAREAANIMATION**  0x1042 | 确定是启用还是禁用动画。 *pvParam* 参数必须指向一个 **BOOL** 变量，该变量在启用动画时接收 **TRUE**，否则为 **FALSE**。  显示功能（如闪烁、闪烁、闪烁和移动内容）可能会导致照片敏感癫痫用户癫痫发作。 通过此标志，可以确定是否在工作区中禁用了此类动画。  **Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETDISABLEOVERLAPPEDCONTENT**  0x1040 | 确定是启用还是禁用重叠内容。 *pvParam* 参数必须指向一个 **BOOL** 变量（如果启用）接收 **TRUE**;否则，该变量必须指向 **FALSE**。  背景图像、纹理背景、文档上的水印、alpha 混合和透明度等显示功能会降低前景和背景之间的对比度，使视力不佳的用户更难看到屏幕上的对象。 此标志使你能够确定此类重叠内容是否已禁用。  **Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETFILTERKEYS**  0x0032 | 检索有关 FilterKeys 辅助功能的信息。 *pvParam* 参数必须指向接收信息的 [FILTERKEYS](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-filterkeys) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(FILTERKEYS)。 | | **SPI\_GETFOCUSBORDERHEIGHT**  0x2010 | 检索使用 [DrawFocusRect](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-drawfocusrect) 绘制的焦点矩形的上边缘和下边缘的高度（以像素为单位）。 *pvParam* 参数必须指向 **UINT** 值。  **Windows 2000：**不支持此参数。 | | **SPI\_GETFOCUSBORDERWIDTH**  0x200E | 检索使用 [DrawFocusRect](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-drawfocusrect) 绘制的焦点矩形的左右边缘的宽度（以像素为单位）。 *pvParam* 参数必须指向 **UINT**。  **Windows 2000：**不支持此参数。 | | **SPI\_GETHIGHCONTRAST**  0x0042 | 检索有关 HighContrast 辅助功能的信息。 *pvParam* 参数必须指向接收信息的 [HIGHCONTRAST](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-highcontrasta) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(HIGHCONTRAST)。  有关一般讨论，请参阅备注。 | | **SPI\_GETLOGICALDPIOVERRIDE**  0x009E | 检索一个值，该值确定 Windows 8 是使用硬件的默认缩放平台显示应用，还是转到下一个较高的平台。 此值基于当前“放大屏幕上的所有内容”设置，该设置位于**电脑设置的“轻松使用**”部分：1 表示打开，0 表示关闭。  应用可以为以下每种缩放平台提供文本和图像资源：100%、140% 和 180%。 提供针对特定规模优化的单独资源可避免因调整大小而失真。 Windows 8 根据多种因素（包括屏幕大小和像素密度）确定适当的缩放平台。 当选择“放大屏幕上的所有内容” (SPI\_GETLOGICALDPIOVERRIDE 返回值 1) 时，Windows 将使用下一个较高平台中的资源。 例如，在 Windows 确定应使用 [SCALE\_100\_PERCENT](https://learn.microsoft.com/zh-cn/windows/desktop/api/shtypes/ne-shtypes-device_scale_factor)刻度的硬件的情况下，此替代会导致 Windows 使用 [SCALE\_140\_PERCENT](https://learn.microsoft.com/zh-cn/windows/desktop/api/shtypes/ne-shtypes-device_scale_factor) 缩放值，前提是它不违反其他约束。  **注意** 不应使用此值。 它在后续版本的 Windows 中可能已更改或不可用。 请改用 [**GetScaleFactorForDevice**](https://learn.microsoft.com/zh-cn/windows/desktop/api/shellscalingapi/nf-shellscalingapi-getscalefactorfordevice) 函数或 [**DisplayProperties**](https://learn.microsoft.com/zh-cn/uwp/api/Windows.Graphics.Display.DisplayProperties) 类来检索首选比例系数。 桌面应用程序应使用桌面逻辑 DPI，而不是比例系数。 可以通过 [**GetDeviceCaps**](https://learn.microsoft.com/zh-cn/windows/desktop/api/wingdi/nf-wingdi-getdevicecaps) 函数检索桌面逻辑 DPI。 | | **SPI\_GETMESSAGEDURATION**  0x2016 | 检索应显示通知弹出窗口的时间（以秒为单位）。 *pvParam* 参数必须指向接收消息持续时间的 **ULONG**。  有视觉障碍或认知障碍（如 ADHD 和阅读障碍）的用户可能需要更长的时间才能阅读通知消息中的文本。 此标志使你能够检索消息持续时间。  **Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | SPI\_GETMOUSECLICKLOCK  0x101E | 检索鼠标 ClickLock 功能的状态。 *pvParam* 参数必须指向一个 **BOOL** 变量（如果启用）接收 **TRUE**;否则，该变量必须指向 **FALSE**。 有关详细信息，请参阅 [鼠标输入概述](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/about-mouse-input)。  **Windows 2000：**不支持此参数。 | | SPI\_GETMOUSECLICKLOCKTIME  0x2008 | 检索锁定主鼠标按钮之前的时间延迟。 *pvParam* 参数必须指向接收时间延迟（以毫秒为单位）的 **DWORD**。 仅当 **SPI\_SETMOUSECLICKLOCK** 设置为 **TRUE** 时，才会启用此功能。 有关详细信息，请参阅 [鼠标输入概述](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/about-mouse-input)。  **Windows 2000：**不支持此参数。 | | **SPI\_GETMOUSEKEYS**  0x0036 | 检索有关 MouseKeys 辅助功能的信息。 *pvParam* 参数必须指向接收信息的 [MOUSEKEYS](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-mousekeys) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(MOUSEKEYS)。 | | SPI\_GETMOUSESONAR  0x101C | 检索鼠标声纳功能的状态。 *pvParam* 参数必须指向一个 **BOOL** 变量，该变量在启用时接收 **TRUE**，否则接收 **FALSE**。 有关详细信息，请参阅 [鼠标输入概述](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/about-mouse-input)。  **Windows 2000：**不支持此参数。 | | SPI\_GETMOUSEVANISH  0x1020 | 检索鼠标消失功能的状态。 *pvParam* 参数必须指向一个 **BOOL** 变量，该变量在启用时接收 **TRUE**，否则接收 **FALSE**。 有关详细信息，请参阅 [鼠标输入概述](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/about-mouse-input)。  **Windows 2000：**不支持此参数。 | | **SPI\_GETSCREENREADER**  0x0046 | 确定屏幕审阅者实用工具是否正在运行。 屏幕审阅者实用工具将文本信息定向到输出设备，例如语音合成器或盲文显示器。 设置此标志后，应用程序应在以图形方式呈现信息的情况下提供文本信息。  *pvParam* 参数是指向 **BOOL** 变量的指针，该变量在屏幕审阅者实用工具正在运行时接收 **TRUE**，否则接收 **FALSE**。  **注意** Windows 附带的屏幕阅读器“讲述人”不会设置 **SPI\_SETSCREENREADER** 或 **SPI\_GETSCREENREADER** 标志。 | | **SPI\_GETSERIALKEYS**  0x003E | 不支持此参数。  **Windows Server 2003 和 Windows XP/2000：**用户应通过控制面板控制此设置。 | | **SPI\_GETSHOWSOUNDS**  0x0038 | 确定“显示声音”辅助功能标志是打开还是关闭。 如果它处于打开状态，则用户要求应用程序在仅以有声形式呈现信息的情况下以可视方式呈现信息。 *pvParam* 参数必须指向一个 **BOOL** 变量，该变量在功能处于打开状态时接收 **TRUE**;如果关闭，则为 **FALSE**。  使用此值等效于使用 **SM\_SHOWSOUNDS** 调用 [GetSystemMetrics](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-getsystemmetrics)。 这是建议的调用。 | | **SPI\_GETSOUNDSENTRY**  0x0040 | 检索有关 SoundSentry 辅助功能的信息。 *pvParam* 参数必须指向接收信息的 [SOUNDSENTRY](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-soundsentrya) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(SOUNDSENTRY)。 | | **SPI\_GETSTICKYKEYS**  0x003A | 检索有关 StickyKeys 辅助功能的信息。 *pvParam* 参数必须指向接收信息的 [STICKYKEYS](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-stickykeys) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(STICKYKEYS)。 | | **SPI\_GETTOGGLEKEYS**  0x0034 | 检索有关 ToggleKeys 辅助功能的信息。 *pvParam* 参数必须指向接收信息的 [TOGGLEKEYS](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-togglekeys) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(TOGGLEKEYS)。 | | **SPI\_SETACCESSTIMEOUT**  0x003D | 设置与辅助功能关联的超时期限。 *pvParam* 参数必须指向包含新参数的 [ACCESSTIMEOUT](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-accesstimeout) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(ACCESSTIMEOUT)。 | | **SPI\_SETAUDIODESCRIPTION**  0x0075 | 打开或关闭音频说明功能。 *pvParam* 参数是指向 [AUDIODESCRIPTION](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-audiodescription) 结构的指针。  虽然视力受损的用户可能会听到视频内容中的音频，但视频中有很多没有相应音频的操作。 视频中发生情况的特定音频说明可帮助这些用户更好地了解内容。 此标志使你能够启用或禁用其提供的语言的音频说明。  **Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETCLIENTAREAANIMATION**  0x1043 | 打开或关闭工作区动画。 *pvParam* 参数是 **BOOL** 变量。 将 *pvParam* 设置为 **TRUE** 以在工作区中启用动画和其他暂时性效果，或将 **FALSE** 设置为禁用它们。  显示功能（如闪烁、闪烁、闪烁和移动内容）可能会导致照片敏感癫痫用户癫痫发作。 此标志使你能够启用或禁用所有此类动画。  **Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETDISABLEOVERLAPPEDCONTENT**  0x1041 | 打开或关闭重叠内容 (，例如背景图像和水印) 。 *pvParam* 参数是 **BOOL** 变量。 将 *pvParam* 设置为 **TRUE** 可禁用重叠内容，将 **FALSE** 设置为启用重叠内容。  背景图像、纹理背景、文档上的水印、alpha 混合和透明度等显示功能会降低前景和背景之间的对比度，使视力不佳的用户更难看到屏幕上的对象。 此标志使你能够启用或禁用所有此类重叠内容。  **Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETFILTERKEYS**  0x0033 | 设置 FilterKeys 辅助功能的参数。 *pvParam* 参数必须指向包含新参数的 [FILTERKEYS](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-filterkeys) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(FILTERKEYS)。 | | **SPI\_SETFOCUSBORDERHEIGHT**  0x2011 | 将 [使用 DrawFocusRect](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-drawfocusrect) 绘制的焦点矩形的上边缘和下边缘的高度设置为 *pvParam* 参数的值。  **Windows 2000：**不支持此参数。 | | **SPI\_SETFOCUSBORDERWIDTH**  0x200F | 将 [使用 DrawFocusRect](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-drawfocusrect) 绘制的焦点矩形的左边缘和右边缘的高度设置为 *pvParam* 参数的值。  **Windows 2000：**不支持此参数。 | | **SPI\_SETHIGHCONTRAST**  0x0043 | 设置 HighContrast 辅助功能的参数。 *pvParam* 参数必须指向包含新参数的 [HIGHCONTRAST](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-highcontrasta) 结构。 将此结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(HIGHCONTRAST)。 | | **SPI\_SETLOGICALDPIOVERRIDE**  0x009F | 请勿使用。 | | **SPI\_SETMESSAGEDURATION**  0x2017 | 设置通知弹出窗口应显示的时间（以秒为单位）。 *pvParam* 参数指定消息持续时间。  有视觉障碍或认知障碍（如 ADHD 和阅读障碍）的用户可能需要更长的时间才能阅读通知消息中的文本。 使用此标志可以设置消息持续时间。  **Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | SPI\_SETMOUSECLICKLOCK  0x101F | 打开或关闭鼠标 ClickLock 辅助功能。 当单击鼠标主按钮并按住 **SPI\_SETMOUSECLICKLOCKTIME**指定的时间时，此功能会暂时锁定该按钮。 *pvParam* 参数为 on 指定 **TRUE**，为 off 指定 **FALSE**。 默认值为 off。 有关详细信息，请参阅备注和 [AboutMouse 输入](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/about-mouse-input)。  **Windows 2000：**不支持此参数。 | | SPI\_SETMOUSECLICKLOCKTIME  0x2009 | 调整锁定主鼠标按钮之前的时间延迟。 *uiParam* 参数应设置为 0。 *pvParam* 参数指向指定以毫秒为单位的时间延迟的 **DWORD**。 例如，为 1 秒延迟指定 1000。 默认值为 1200。 有关详细信息，请参阅 [鼠标输入概述](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/about-mouse-input)。  **Windows 2000：**不支持此参数。 | | **SPI\_SETMOUSEKEYS**  0x0037 | 设置 MouseKeys 辅助功能的参数。 *pvParam* 参数必须指向包含新参数的 [MOUSEKEYS](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-mousekeys) 结构。 将此结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(MOUSEKEYS)。 | | SPI\_SETMOUSESONAR  0x101D | 打开或关闭 Sonar 辅助功能。 当用户按下并释放 Ctrl 键时，此功能简要显示鼠标指针周围的几个同心圆。 *pvParam* 参数为 on 指定 **TRUE**，为 off 指定 **FALSE**。 默认值为 off。 有关详细信息，请参阅 [鼠标输入概述](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/about-mouse-input)。  **Windows 2000：**不支持此参数。 | | SPI\_SETMOUSEVANISH  0x1021 | 打开或关闭“消失”功能。 此功能在用户键入时隐藏鼠标指针;当用户移动鼠标时，指针再次出现。 *pvParam* 参数为 on 指定 **TRUE**，为 off 指定 **FALSE**。 默认值为 off。 有关详细信息，请参阅 [鼠标输入概述](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/about-mouse-input)。  **Windows 2000：**不支持此参数。 | | **SPI\_SETSCREENREADER**  0x0047 | 确定屏幕评审实用工具是否正在运行。 *uiParam* 参数为 on 指定 **TRUE**，为 off 指定 **FALSE**。  **注意** Windows 附带的屏幕阅读器“讲述人”未设置 **SPI\_SETSCREENREADER** 或 **SPI\_GETSCREENREADER** 标志。 | | **SPI\_SETSERIALKEYS**  0x003F | 不支持此参数。  **Windows Server 2003 和 Windows XP/2000：**用户应通过控制面板控制此设置。 | | **SPI\_SETSHOWSOUNDS**  0x0039 | 打开或关闭 ShowSounds 辅助功能。 *uiParam* 参数为 on 指定 **TRUE**，为 off 指定 **FALSE**。 | | **SPI\_SETSOUNDSENTRY**  0x0041 | 设置 **SoundSentry** 辅助功能的参数。 *pvParam* 参数必须指向包含新参数的 [SOUNDSENTRY](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-soundsentrya) 结构。 将此结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(SOUNDSENTRY)。 | | **SPI\_SETSTICKYKEYS**  0x003B | 设置 StickyKeys 辅助功能的参数。 *pvParam* 参数必须指向包含新参数的 [STICKYKEYS](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-stickykeys) 结构。 将此结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(STICKYKEYS)。 | | **SPI\_SETTOGGLEKEYS**  0x0035 | 设置 ToggleKeys 辅助功能的参数。 *pvParam* 参数必须指向包含新参数的 [TOGGLEKEYS](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-togglekeys) 结构。 将此结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(TOGGLEKEYS)。 |     下面是桌面参数。  展开表   |  |  | | --- | --- | | **桌面参数** | **含义** | | **SPI\_GETCLEARTYPE**  0x1048 | 确定是否启用 ClearType。 *pvParam* 参数必须指向一个 **BOOL** 变量，如果启用 ClearType，则接收 **TRUE**;否则，该变量必须指向 **FALSE**。  ClearType 是一种软件技术，可提高液晶显示器 (液晶显示器) 显示器上文本的可读性。  **Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETDESKWALLPAPER**  0x0073 | 检索桌面壁纸的位图文件的完整路径。 *pvParam* 参数必须指向缓冲区才能接收以 null 结尾的路径字符串。 将 *uiParam* 参数设置为 *pvParam* 缓冲区的大小（以字符为单位）。 返回的字符串不超过 **MAX\_PATH** 个字符。 如果没有桌面壁纸，则返回的字符串为空。 | | **SPI\_GETDROPSHADOW**  0x1024 | 确定是否启用投影效果。 *pvParam* 参数必须指向**一个 BOOL** 变量，如果启用，则返回 **TRUE**;如果禁用，则返回 **FALSE**。  **Windows 2000：**不支持此参数。 | | **SPI\_GETFLATMENU**  0x1022 | 确定本机用户菜单是否具有平面菜单外观。 *pvParam* 参数必须指向一个 **BOOL** 变量，如果设置了平面菜单外观，则返回 **TRUE**，否则返回 **FALSE**。  **Windows 2000：**不支持此参数。 | | **SPI\_GETFONTSMOOTHING**  0x004A | 确定是否启用字体平滑功能。 此功能使用字体抗锯齿，通过绘制不同灰色级别的像素，使字体曲线看起来更平滑。  *pvParam* 参数必须指向一个 **BOOL** 变量，该变量在启用该功能时接收 **TRUE**;如果功能未启用，则为 **FALSE**。 | | **SPI\_GETFONTSMOOTHINGCONTRAST**  0x200C | 检索 [在 ClearType](https://learn.microsoft.com/zh-cn/typography/cleartype/) 平滑处理中使用的对比度值。 *pvParam* 参数必须指向接收信息的 **UINT**。 有效的对比度值为 1000 到 2200。 默认值为 1400。  **Windows 2000：**不支持此参数。 | | **SPI\_GETFONTSMOOTHINGORIENTATION**  0x2012 | 检索字体平滑方向。 *pvParam* 参数必须指向接收信息的 **UINT**。 可能 **的值FE\_FONTSMOOTHINGORIENTATIONBGR (** 蓝-绿-红) 和 **FE\_FONTSMOOTHINGORIENTATIONRGB** (红-绿-蓝) 。  **Windows XP/2000：**在 Windows XP SP2 之前不支持此参数。 | | **SPI\_GETFONTSMOOTHINGTYPE**  0x200A | 检索字体平滑的类型。 *pvParam* 参数必须指向接收信息的 **UINT**。 可能的值为 **FE\_FONTSMOOTHINGSTANDARD** 和 **FE\_FONTSMOOTHINGCLEARTYPE**。  **Windows 2000：**不支持此参数。 | | **SPI\_GETWORKAREA**  0x0030 | 检索主显示器上工作区的大小。 工作区域是系统任务栏或应用程序桌面工具栏未遮挡的屏幕部分。 *pvParam* 参数必须指向接收工作区坐标的 [RECT](https://learn.microsoft.com/zh-cn/windows/desktop/api/windef/ns-windef-rect) 结构，以物理像素大小表示。 调用方的任何 DPI 虚拟化模式都不会影响此输出。  若要获取主显示监视器以外的监视器的工作区域，请调用 [GetMonitorInfo](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-getmonitorinfoa) 函数。 | | **SPI\_SETCLEARTYPE**  0x1049 | 打开或关闭 ClearType。 *pvParam* 参数是 **BOOL** 变量。 将 *pvParam* 设置为 **TRUE** 以启用 ClearType，或 **将 FALSE** 设置为禁用它。  ClearType 是一种软件技术，可提高 LCD 监视器上文本的可读性。  **Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETCURSORS**  0x0057 | 重新加载系统游标。 将 *uiParam* 参数设置为零，将 *pvParam* 参数设置为 **NULL**。 | | **SPI\_SETDESKPATTERN**  0x0015 | 通过使 Windows 从 WIN.INI 文件中读取 **Pattern=** 设置来设置当前桌面模式。 | | **SPI\_SETDESKWALLPAPER**  0x0014 | **注意** 使用 **SPI\_SETDESKWALLPAPER** 标志时， **SystemParametersInfo** 返回 **TRUE** ，除非 (出现错误，例如) 指定的文件不存在时。 | | **SPI\_SETDROPSHADOW**  0x1025 | 启用或禁用投影效果。 将 *pvParam* 设置为 **TRUE** 可启用投影效果，将 **设置为 FALSE** 以禁用它。 还必须在窗口类样式中 **CS\_DROPSHADOW** 。  **Windows 2000：**不支持此参数。 | | **SPI\_SETFLATMENU**  0x1023 | 启用或禁用本机用户菜单的平面菜单外观。 将 *pvParam* 设置为 **TRUE** 可启用平面菜单外观，将 **设置为 FALSE** 以禁用它。  启用后，菜单栏将 **COLOR\_MENUBAR** 用于菜单栏背景， **COLOR\_MENU** 用于菜单弹出背景， **COLOR\_MENUHILIGHT** 填充当前菜单选择， **COLOR\_HILIGHT** 当前菜单选择的轮廓。 如果禁用，则使用与 Windows 2000 中相同的指标和颜色绘制菜单。  **Windows 2000：**不支持此参数。 | | **SPI\_SETFONTSMOOTHING**  0x004B | 启用或禁用字体平滑功能，该功能使用字体抗锯齿功能，通过绘制不同灰度级别的像素，使字体曲线看起来更平滑。  若要启用该功能，请将 *uiParam* 参数设置为 **TRUE**。 若要禁用该功能，请将 *uiParam* 设置为 **FALSE**。 | | **SPI\_SETFONTSMOOTHINGCONTRAST**  0x200D | 设置 [ClearType](https://learn.microsoft.com/zh-cn/typography/cleartype/) 平滑中使用的对比度值。 *pvParam* 参数是对比度值。 有效的对比度值为 1000 到 2200。 默认值为 1400。  **SPI\_SETFONTSMOOTHINGTYPE** 还必须设置为 **FE\_FONTSMOOTHINGCLEARTYPE**。  **Windows 2000：**不支持此参数。 | | **SPI\_SETFONTSMOOTHINGORIENTATION**  0x2013 | 设置字体平滑方向。 *pvParam* 参数**FE\_FONTSMOOTHINGORIENTATIONBGR (蓝**-绿-红) 或**FE\_FONTSMOOTHINGORIENTATIONRGB** (红-绿-蓝) 。  **Windows XP/2000：**在 Windows XP SP2 之前不支持此参数。 | | **SPI\_SETFONTSMOOTHINGTYPE**  0x200B | 设置字体平滑类型。 如果使用标准抗锯齿，则 *pvParam* 参数为**FE\_FONTSMOOTHINGSTANDARD**，如果使用 [ClearType](https://learn.microsoft.com/zh-cn/typography/cleartype/)，则**为FE\_FONTSMOOTHINGCLEARTYPE**。 默认值为 **FE\_FONTSMOOTHINGSTANDARD**。  **还必须设置SPI\_SETFONTSMOOTHING** 。  **Windows 2000：**不支持此参数。 | | **SPI\_SETWORKAREA**  0x002F | 设置工作区的大小。 工作区域是系统任务栏或应用程序桌面工具栏未遮挡的屏幕部分。 *pvParam* 参数是指向 [RECT](https://learn.microsoft.com/zh-cn/windows/desktop/api/windef/ns-windef-rect) 结构的指针，该结构指定以虚拟屏幕坐标表示的新工作区矩形。 在具有多个显示监视器的系统中， 函数设置包含指定矩形的监视器的工作区域。 |     下面是图标参数。  展开表   |  |  | | --- | --- | | **图标参数** | **含义** | | **SPI\_GETICONMETRICS**  0x002D | 检索与图标关联的指标。 *pvParam* 参数必须指向接收信息的 [ICONMETRICS](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-iconmetricsa) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(ICONMETRICS)。 | | **SPI\_GETICONTITLELOGFONT**  0x001F | 检索当前图标标题字体的逻辑字体信息。 *uiParam* 参数指定 [LOGFONT](https://learn.microsoft.com/zh-cn/windows/desktop/api/wingdi/ns-wingdi-logfonta) 结构的大小，*pvParam* 参数必须指向要填充的 **LOGFONT** 结构。 | | **SPI\_GETICONTITLEWRAP**  0x0019 | 确定是否启用图标标题环绕。 *pvParam* 参数必须指向一个 **BOOL** 变量（如果启用）接收 **TRUE**;否则，该变量必须指向 **FALSE**。 | | **SPI\_ICONHORIZONTALSPACING**  0x000D | 设置或检索图标单元格的宽度（以像素为单位）。 系统使用此矩形在大图标视图中排列图标。  若要设置此值，请将 *uiParam* 设置为新值，并将 *pvParam* 设置为 **NULL**。 不能将此值设置为小于 **SM\_CXICON**。  若要检索此值， *pvParam* 必须指向接收当前值的整数。 | | **SPI\_ICONVERTICALSPACING**  0x0018 | 设置或检索图标单元格的高度（以像素为单位）。  若要设置此值，请将 *uiParam* 设置为新值，并将 *pvParam* 设置为 **NULL**。 不能将此值设置为小于 **SM\_CYICON**。  若要检索此值， *pvParam* 必须指向接收当前值的整数。 | | **SPI\_SETICONMETRICS**  0x002E | 设置与图标关联的指标。 *pvParam* 参数必须指向包含新参数的 [ICONMETRICS](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-iconmetricsa) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(ICONMETRICS)。 | | **SPI\_SETICONS**  0x0058 | 重新加载系统图标。 将 *uiParam* 参数设置为零，将 *pvParam* 参数设置为 **NULL**。 | | **SPI\_SETICONTITLELOGFONT**  0x0022 | 设置用于图标标题的字体。 *uiParam* 参数指定 [LOGFONT](https://learn.microsoft.com/zh-cn/windows/desktop/api/wingdi/ns-wingdi-logfonta) 结构的大小，*pvParam* 参数必须指向 **LOGFONT** 结构。 | | **SPI\_SETICONTITLEWRAP**  0x001A | 打开或关闭图标标题环绕。 *uiParam* 参数为 on 指定 **TRUE**，为 off 指定 **FALSE**。 |     下面是输入参数。 它们包括与键盘、鼠标、触摸板、笔、输入语言和警告蜂鸣器相关的参数。  展开表   |  |  | | --- | --- | | **输入参数** | **含义** | | **SPI\_GETBEEP**  0x0001 | 确定警告蜂鸣器是否已打开。  *pvParam* 参数必须指向一个 **BOOL** 变量，该变量在蜂鸣器处于打开状态时接收 **TRUE**;如果关闭，则为 **FALSE**。 | | **SPI\_GETBLOCKSENDINPUTRESETS**  0x1026 | 检索一个 **BOOL** ，指示应用程序是否可以通过调用 [SendInput](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-sendinput) 函数来模拟键盘或鼠标输入来重置屏幕保护程序计时器。 如果模拟输入将被阻止，*pvParam* 参数必须指向接收 **TRUE** 的 **BOOL** 变量;否则，该变量必须指向 **FALSE**。 | | **SPI\_GETCONTACTVISUALIZATION**  0x2018 | 检索当前联系人可视化设置。 *pvParam* 参数必须指向接收设置的 **ULONG** 变量。 有关详细信息，请参阅 [联系人可视化](https://learn.microsoft.com/zh-cn/windows/desktop/winmsg/contact-visualization)。 | | **SPI\_GETDEFAULTINPUTLANG**  0x0059 | 检索系统默认输入语言的输入区域设置标识符。 *pvParam* 参数必须指向接收此值的 **HKL** 变量。 有关详细信息，请参阅 [语言、区域设置和键盘布局](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/about-keyboard-input)。 | | **SPI\_GETGESTUREVISUALIZATION**  0x201A | 检索当前手势可视化设置。 *pvParam* 参数必须指向接收设置的 **ULONG** 变量。 有关详细信息，请参阅 [手势可视化](https://learn.microsoft.com/zh-cn/windows/desktop/winmsg/gesture-visualization)。 | | **SPI\_GETKEYBOARDCUES**  0x100A | 确定菜单访问键是否始终带有下划线。 *pvParam* 参数必须指向**一个 BOOL** 变量，该变量在菜单访问键始终带有下划线时接收 **TRUE**;如果仅在键盘激活菜单时才为 FALSE，则为 **FALSE**。 | | **SPI\_GETKEYBOARDDELAY**  0x0016 | 检索键盘重复延迟设置，该值范围为 0 (大约 250 毫秒延迟) 到 3 (大约 1 秒延迟) 。 与每个值关联的实际延迟可能因硬件而异。 *pvParam* 参数必须指向接收设置的整数变量。 | | **SPI\_GETKEYBOARDPREF**  0x0044 | 确定用户是否依赖于键盘而不是鼠标，并希望应用程序显示本来隐藏的键盘界面。 如果用户依赖键盘，*pvParam* 参数必须指向接收 **TRUE** 的 **BOOL** 变量;否则为 **FALSE**。 | | **SPI\_GETKEYBOARDSPEED**  0x000A | 检索键盘重复速度设置，该值的范围是从 0 (大约 2.5 次每秒重复) 到 31 (大约每秒 30 次重复) 。 实际重复率取决于硬件，可能与线性比例相差多达 20%。 *pvParam* 参数必须指向接收设置的 **DWORD** 变量。 | | **SPI\_GETMOUSE**  0x0003 | 检索两个鼠标阈值和鼠标加速。 *pvParam* 参数必须指向接收这些值的三个整数数组。 有关详细信息 [，请参阅mouse\_event](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-mouse_event) 。 | | **SPI\_GETMOUSEHOVERHEIGHT**  0x0064 | 检索鼠标指针必须保留的矩形的高度（以像素为单位），以便 [TrackMouseEvent](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-trackmouseevent) 生成 [WM\_MOUSEHOVER](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/wm-mousehover) 消息。 *pvParam* 参数必须指向接收高度的 **UINT** 变量。 | | **SPI\_GETMOUSEHOVERTIME**  0x0066 | 检索鼠标指针必须停留在悬停矩形中的时间（以毫秒为单位），以便 [TrackMouseEvent](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-trackmouseevent) 生成 [WM\_MOUSEHOVER](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/wm-mousehover) 消息。 *pvParam* 参数必须指向接收时间的 **UINT** 变量。 | | **SPI\_GETMOUSEHOVERWIDTH**  0x0062 | 检索鼠标指针必须保留的矩形的宽度（以像素为单位），以便 [TrackMouseEvent](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-trackmouseevent) 生成 [WM\_MOUSEHOVER](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/wm-mousehover) 消息。 *pvParam* 参数必须指向接收宽度的 **UINT** 变量。 | | **SPI\_GETMOUSESPEED**  0x0070 | 检索当前鼠标速度。 鼠标速度根据鼠标移动的距离确定指针移动的距离。 *pvParam* 参数必须指向一个整数，该整数接收的值范围为 1 (最慢) 到 20 (最快) 。 默认值为 10。 该值可由最终用户使用鼠标控制面板应用程序设置，也可以由使用 **SPI\_SETMOUSESPEED**的应用程序设置。 | | **SPI\_GETMOUSETRAILS**  0x005E | 确定是否启用鼠标跟踪功能。 此功能通过简要显示光标的踪迹并快速擦除它们来提高鼠标光标移动的可见性。  *pvParam* 参数必须指向接收值的整数变量。 如果值为零或 1，则禁用该功能。 如果值大于 1，则启用该功能，该值指示在跟踪中绘制的游标数。 不使用 *uiParam* 参数。  **Windows 2000：**不支持此参数。 | | **SPI\_GETMOUSEWHEELROUTING**  0x201C | 检索鼠标滚轮输入的路由设置。 路由设置确定是将鼠标滚轮输入发送到具有焦点 (前台) 的应用，还是将鼠标光标下的应用发送到应用。  *pvParam* 参数必须指向接收路由选项的 **DWORD** 变量。 不使用 *uiParam* 参数。  如果值为零 (MOUSEWHEEL\_ROUTING\_FOCUS) ，则鼠标滚轮输入将传送到具有焦点的应用。 如果值为 1 (MOUSEWHEEL\_ROUTING\_HYBRID) ，鼠标滚轮输入将传送到具有焦点的应用 (桌面应用) 或鼠标指针下的应用 (Windows 应用商店应用) 。  **从 Windows 10 开始：** 如果值为 2 (MOUSEWHEEL\_ROUTING\_MOUSE\_POS) ，则鼠标滚轮输入将传送到鼠标指针下的应用。 这是新的默认值，MOUSEWHEEL\_ROUTING\_HYBRID在“设置”中不再可用。 | | **SPI\_GETPENVISUALIZATION**  0x201E | 检索当前笔手势可视化设置。 *pvParam* 参数必须指向接收设置的 **ULONG** 变量。 有关详细信息，请参阅 [触控笔可视化](https://learn.microsoft.com/zh-cn/windows/desktop/winmsg/pen-visualization)。 | | **SPI\_GETSNAPTODEFBUTTON**  0x005F | 确定是否启用对齐到默认按钮功能。 如果启用，鼠标光标会自动移动到对话框的默认按钮，例如 **“确定”** 或 **“应用**”。 *pvParam* 参数必须指向一个 **BOOL** 变量，如果功能处于打开状态，则接收 **TRUE**;如果关闭，则为 **FALSE**。 | | **SPI\_GETSYSTEMLANGUAGEBAR**  0x1050 | **从 Windows 8 开始：** 确定是启用或禁用系统语言栏。 如果启用语言栏，*pvParam* 参数必须指向接收 **TRUE** 的 **BOOL** 变量;否则，该变量必须指向 **FALSE**。 | | **SPI\_GETTHREADLOCALINPUTSETTINGS**  0x104E | **从 Windows 8 开始：** 确定活动输入设置是否具有每个线程的本地 (、 **TRUE**) 或全局 (会话、 **FALSE**) 范围。 *pvParam* 参数必须指向 **BOOL** 变量。 | | **SPI\_GETTOUCHPADPARAMETERS**  0x00AE | **从 Windows 11 版本 24H2 开始：** 检索有关精确式触摸板的详细信息，包括与触摸板相关的用户设置和系统信息。  pvParam 参数必须指向[**TOUCHPAD\_PARAMETERS**](https://learn.microsoft.com/zh-cn/windows/win32/api/winuser/ns-winuser-touchpad_parameters)结构。  uiParam 参数必须指定结构的大小。  必须将 TOUCHPAD\_PARAMETERS 结构中的 versionNumber 字段的值设置为所使用的结构版本的适当值。 | | **SPI\_GETWHEELSCROLLCHARS**  0x006C | 检索移动水平鼠标滚轮时要滚动的字符数。 *pvParam* 参数必须指向接收行数的 **UINT** 变量。 默认值为 3。 | | **SPI\_GETWHEELSCROLLLINES**  0x0068 | 检索移动垂直鼠标滚轮时要滚动的行数。 *pvParam* 参数必须指向接收行数的 **UINT** 变量。 默认值为 3。 | | **SPI\_SETBEEP**  0x0002 | 打开或关闭警告蜂鸣器。 *uiParam* 参数为 on 指定 **TRUE**，为 off 指定 **FALSE**。 | | **SPI\_SETBLOCKSENDINPUTRESETS**  0x1027 | 通过调用 [SendInput](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-sendinput) 函数来模拟键盘或鼠标输入，确定应用程序是否可以重置屏幕保护程序计时器。 如果屏幕保护程序不会通过模拟输入停用， *则 uiParam* 参数指定 **TRUE** ;如果屏幕保护程序将由模拟输入停用，则指定 **FALSE** 。 | | **SPI\_SETCONTACTVISUALIZATION**  0x2019 | 设置当前联系人可视化设置。 *pvParam* 参数必须指向标识设置的 **ULONG** 变量。 有关详细信息，请参阅 [联系人可视化](https://learn.microsoft.com/zh-cn/windows/desktop/winmsg/contact-visualization)。  **注意** 如果禁用联系人可视化效果，则无法启用手势可视化效果。 | | **SPI\_SETDEFAULTINPUTLANG**  0x005A | 设置系统 shell 和应用程序的默认输入语言。 指定的语言必须使用当前系统字符集显示。 *pvParam* 参数必须指向包含默认语言的输入区域设置标识符的 **HKL** 变量。 有关详细信息，请参阅 [语言、区域设置和键盘布局](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/about-keyboard-input)。 | | **SPI\_SETDOUBLECLICKTIME**  0x0020 | 将鼠标的双击时间设置为 *uiParam* 参数的值。 如果 *uiParam* 值大于 5000 毫秒，系统会将双击时间设置为 5000 毫秒。  双击时间是双击的第一次和第二次单击之间可能发生的最大毫秒数。 还可以调用 [SetDoubleClickTime](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-setdoubleclicktime) 函数来设置双击时间。 若要获取当前双击时间，请调用 [GetDoubleClickTime](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-getdoubleclicktime) 函数。 | | **SPI\_SETDOUBLECLKHEIGHT**  0x001E | 将双击矩形的高度设置为 *uiParam* 参数的值。  双击矩形是一个矩形，双击的第二次单击必须属于该矩形，才能将其注册为双击。  若要检索双击矩形的高度，请使用 **SM\_CYDOUBLECLK** 标志调用 [GetSystemMetrics](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-getsystemmetrics)。 | | **SPI\_SETDOUBLECLKWIDTH**  0x001D | 将双击矩形的宽度设置为 *uiParam* 参数的值。  双击矩形是一个矩形，双击的第二次单击必须属于该矩形，才能将其注册为双击。  若要检索双击矩形的宽度，请使用 **SM\_CXDOUBLECLK** 标志调用 [GetSystemMetrics](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-getsystemmetrics)。 | | **SPI\_SETGESTUREVISUALIZATION**  0x201B | 设置当前手势可视化设置。 *pvParam* 参数必须指向标识设置的 **ULONG** 变量。 有关详细信息，请参阅 [手势可视化](https://learn.microsoft.com/zh-cn/windows/desktop/winmsg/gesture-visualization)。  **注意** 如果禁用联系人可视化效果，则无法启用手势可视化效果。 | | **SPI\_SETKEYBOARDCUES**  0x100B | 设置菜单访问键字母的下划线。 *pvParam* 参数是 **BOOL** 变量。 将 *pvParam* 设置为 **TRUE** 以始终为菜单访问键添加下划线，将 **FALSE** 设置为仅在从键盘激活菜单时为菜单访问键添加下划线。 | | **SPI\_SETKEYBOARDDELAY**  0x0017 | 设置键盘重复延迟设置。 *uiParam* 参数必须指定 0、1、2 或 3，其中 0 设置最短延迟约 250 毫秒) ,3 设置最长延迟 (大约 1 秒) 。 与每个值关联的实际延迟可能因硬件而异。 | | **SPI\_SETKEYBOARDPREF**  0x0045 | 设置键盘首选项。 如果用户依赖于键盘而不是鼠标，并且希望应用程序显示本来隐藏的键盘接口， *则 uiParam* 参数指定 **TRUE** ; *否则，uiParam* 为 **FALSE** 。 | | **SPI\_SETKEYBOARDSPEED**  0x000B | 设置键盘重复速度设置。 *uiParam* 参数必须指定介于 0 (大约每秒 2.5 次重复) 到 31 (大约每秒 30 次重复) 范围内的值。 实际重复率取决于硬件，并且可能从线性刻度变化多达 20%。 如果 *uiParam* 大于 31，则 参数设置为 31。 | | **SPI\_SETLANGTOGGLE**  0x005B | 设置用于在输入语言之间切换的热键集。 不使用 *uiParam* 和 *pvParam* 参数。 值通过再次读取注册表来设置键盘属性表中的快捷键。 必须先设置注册表，然后才能使用此标志。 注册表中的路径为 **HKEY\_CURRENT\_USER**\**键盘布局**\**切换**  . 有效值为“1”= Alt+SHIFT，“2”= CTRL+SHIFT，“3”= none。 | | **SPI\_SETMOUSE**  0x0004 | 设置两个鼠标阈值和鼠标加速。 *pvParam* 参数必须指向指定这些值的三个整数数组。 有关详细信息 [，请参阅mouse\_event](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-mouse_event) 。 | | **SPI\_SETMOUSEBUTTONSWAP**  0x0021 | 交换或还原鼠标左键和右键的含义。 *uiParam* 参数指定 **TRUE** 以交换按钮的含义，或**指定 FALSE** 以还原其原始含义。  若要检索当前设置，请使用 **SM\_SWAPBUTTON** 标志调用 [GetSystemMetrics](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-getsystemmetrics)。 | | **SPI\_SETMOUSEHOVERHEIGHT**  0x0065 | 设置鼠标指针必须保留的矩形的高度（以像素为单位），以便 [TrackMouseEvent](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-trackmouseevent) 生成 [WM\_MOUSEHOVER](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/wm-mousehover) 消息。 将 *uiParam* 参数设置为新高度。 | | **SPI\_SETMOUSEHOVERTIME**  0x0067 | 设置鼠标指针必须停留在悬停矩形中的时间（以毫秒为单位），以便 [TrackMouseEvent](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-trackmouseevent) 生成 [WM\_MOUSEHOVER](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/wm-mousehover) 消息。 仅当在调用 **TrackMouseEvent** 时在 *dwHoverTime* 参数中传递**HOVER\_DEFAULT**时，才使用此选项。 将 *uiParam* 参数设置为新时间。  指定的时间应介于 **USER\_TIMER\_MAXIMUM** 和 **USER\_TIMER\_MINIMUM** 之间。 如果 *uiParam* 小于 **USER\_TIMER\_MINIMUM**，函数将使用 **USER\_TIMER\_MINIMUM**。 如果 *uiParam* 大于 **USER\_TIMER\_MAXIMUM**，则将 **USER\_TIMER\_MAXIMUM**函数。  **Windows Server 2003 和 Windows XP：**在 Windows Server 2003 SP1 和 Windows XP 与 SP2 之前，操作系统不会强制使用 **USER\_TIMER\_MAXIMUM** 和 **USER\_TIMER\_MINIMUM** 。 | | **SPI\_SETMOUSEHOVERWIDTH**  0x0063 | 设置鼠标指针必须保留的矩形的宽度（以像素为单位），以便 [TrackMouseEvent](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-trackmouseevent) 生成 [WM\_MOUSEHOVER](https://learn.microsoft.com/zh-cn/windows/desktop/inputdev/wm-mousehover) 消息。 将 *uiParam* 参数设置为新的宽度。 | | **SPI\_SETMOUSESPEED**  0x0071 | 设置当前鼠标速度。 *pvParam* 参数是介于 1 (最慢) 到 20 (最快) 之间的整数。 默认值为 10。 此值通常是使用鼠标控制面板应用程序设置的。 | | **SPI\_SETMOUSETRAILS**  0x005D | 启用或禁用鼠标跟踪功能，该功能通过简要显示光标跟踪并快速擦除它们来提高鼠标光标移动的可见性。  若要禁用该功能，请将 *uiParam* 参数设置为零或 1。 若要启用该功能，请将 *uiParam* 设置为大于 1 的值，以指示在跟踪中绘制的游标数。  **Windows 2000：**不支持此参数。 | | **SPI\_SETMOUSEWHEELROUTING**  0x201D | 设置鼠标滚轮输入的路由设置。 路由设置确定是将鼠标滚轮输入发送到具有焦点 (前台) 的应用，还是将鼠标光标下的应用发送到应用。  *pvParam* 参数必须指向接收路由选项的 **DWORD** 变量。 将 *uiParam* 参数设置为零。  如果值为零 (MOUSEWHEEL\_ROUTING\_FOCUS) ，则鼠标滚轮输入将传送到具有焦点的应用。 如果值为 1 (MOUSEWHEEL\_ROUTING\_HYBRID) ，鼠标滚轮输入将传送到具有焦点的应用 (桌面应用) 或鼠标指针下的应用 (Windows 应用商店应用) 。  **从 Windows 10 开始：** 如果值为 2 (MOUSEWHEEL\_ROUTING\_MOUSE\_POS) ，则鼠标滚轮输入将传送到鼠标指针下的应用。 这是新的默认值，MOUSEWHEEL\_ROUTING\_HYBRID在“设置”中不再可用。 | | **SPI\_SETPENVISUALIZATION**  0x201F | 设置当前笔手势可视化设置。 *pvParam* 参数必须指向标识设置的 **ULONG** 变量。 有关详细信息，请参阅 [触控笔可视化](https://learn.microsoft.com/zh-cn/windows/desktop/winmsg/pen-visualization)。 | | **SPI\_SETSNAPTODEFBUTTON**  0x0060 | 启用或禁用快照到默认按钮功能。 如果启用，鼠标光标会自动移动到对话框的默认按钮，例如 **“确定”** 或 **“应用**”。 将 *uiParam* 参数设置为 **TRUE** 以启用该功能，将 **设置为 FALSE** 以禁用该功能。 应用程序在显示对话框时应使用 [ShowWindow](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-showwindow) 函数，以便对话管理器可以定位鼠标光标。 | | **SPI\_SETSYSTEMLANGUAGEBAR**  0x1051 | **从 Windows 8 开始：** 打开或关闭旧语言栏功能。 *pvParam* 参数是指向 **BOOL** 变量的指针。 将 *pvParam* 设置为 **TRUE** 以启用旧语言栏，或 **将 FALSE** 设置为禁用它。 标志在 Windows 8 上受支持，其中旧语言栏被输入切换器替换，因此默认处于关闭状态。 打开旧语言栏是出于兼容性原因而提供的，对输入切换器没有影响。 | | **SPI\_SETTHREADLOCALINPUTSETTINGS**  0x104F | **从 Windows 8 开始：** 确定活动输入设置是否具有每个线程的本地 (、 **TRUE**) 或全局 (会话、 **FALSE**) 范围。 *pvParam* 参数必须是由 PVOID 强制转换的 **BOOL** 变量。 | | **SPI\_SETTOUCHPADPARAMETERS**  0x00AF | **从 Windows 11 版本 24H2 开始：** 设置有关精确式触摸板的详细信息，包括与触摸板相关的用户设置和系统信息。  pvParam 参数必须指向[**TOUCHPAD\_PARAMETERS**](https://learn.microsoft.com/zh-cn/windows/win32/api/winuser/ns-winuser-touchpad_parameters)结构。  uiParam 参数必须指定结构的大小。  必须将 TOUCHPAD\_PARAMETERS 结构中的 versionNumber 字段的值设置为所使用的结构版本的适当值。 | | **SPI\_SETWHEELSCROLLCHARS**  0x006D | 设置移动水平鼠标滚轮时要滚动的字符数。 从 *uiParam* 参数设置字符数。 | | **SPI\_SETWHEELSCROLLLINES**  0x0069 | 设置移动垂直鼠标滚轮时要滚动的行数。 从 *uiParam* 参数设置行数。  行数是在不使用修饰键的情况下滚动鼠标滚轮时要滚动的建议行数。 如果数字为 0，则不应发生滚动。 如果要滚动的行数大于可查看的行数，特别是当它 **WHEEL\_PAGESCROLL** (#defined 为 **UINT\_MAX**) ，则滚动操作应解释为单击滚动条页面向下或向上页区域单击一次。 |     下面是菜单参数。  展开表   |  |  | | --- | --- | | **菜单参数** | **含义** | | **SPI\_GETMENUDROPALIGNMENT**  0x001B | 确定弹出菜单相对于相应的菜单栏项是左对齐还是右对齐。 *pvParam* 参数必须指向接收 **TRUE**（如果右对齐）的 **BOOL** 变量;否则，该变量必须指向 **FALSE**。 | | **SPI\_GETMENUFADE**  0x1012 | 确定是否启用菜单淡出动画。 *pvParam* 参数必须指向一个 **BOOL** 变量，该变量在启用淡出动画时接收 **TRUE**，在可禁用时接收 **FALSE**。 如果禁用淡出动画，菜单将使用幻灯片动画。 除非启用了菜单动画，否则将忽略此标志，可以使用 **SPI\_SETMENUANIMATION** 标志执行此操作。 有关详细信息，请参阅 [AnimateWindow](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-animatewindow)。 | | **SPI\_GETMENUSHOWDELAY**  0x006A | 检索鼠标光标位于子菜单项上时系统在显示快捷菜单之前等待的时间（以毫秒为单位）。 *pvParam* 参数必须指向接收延迟时间的 **DWORD** 变量。 | | **SPI\_SETMENUDROPALIGNMENT**  0x001C | 设置弹出菜单的对齐值。 *uiParam* 参数为右对齐指定 **TRUE**，为左对齐指定 **FALSE**。 | | **SPI\_SETMENUFADE**  0x1013 | 启用或禁用菜单淡出动画。 将 *pvParam* 设置为 **TRUE** 以启用菜单淡出效果，或 **将 FALSE** 设置为禁用它。 如果禁用淡出动画，菜单将使用幻灯片动画。 仅当系统的颜色深度超过 256 种颜色时，才能实现菜单淡出效果。 除非还设置了 **SPI\_MENUANIMATION** ，否则将忽略此标志。 有关详细信息，请参阅 [AnimateWindow](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-animatewindow)。 | | **SPI\_SETMENUSHOWDELAY**  0x006B | 将 *uiParam* 设置为鼠标光标位于子菜单项上时系统在显示快捷菜单之前等待的时间（以毫秒为单位）。 |     下面是电源参数。  从 Windows Server 2008 和 Windows Vista 开始，不支持这些电源参数。 相反，若要确定当前显示电源状态，应用程序应注册 **GUID\_MONITOR\_POWER\_STATE** 通知。 若要确定当前显示电源关闭超时，应用程序应注册 **GUID\_VIDEO\_POWERDOWN\_TIMEOUT** 电源设置更改通知。 有关详细信息，请参阅 [注册 Power Events](https://learn.microsoft.com/zh-cn/windows/desktop/Power/registering-for-power-events)。  **Windows Server 2003 和 Windows XP/2000：**若要确定当前显示电源状态，请使用以下电源参数。  展开表   |  |  | | --- | --- | | **Power 参数** | **含义** | | **SPI\_GETLOWPOWERACTIVE**  0x0053 | 不支持此参数。  **Windows Server 2003 和 Windows XP/2000：**确定是否启用了屏幕保存的低功耗阶段。 *pvParam* 参数必须指向接收 **TRUE**（如果已启用）的 **BOOL** 变量;如果禁用，则为 **FALSE**。 仅 32 位应用程序支持此标志。 | | **SPI\_GETLOWPOWERTIMEOUT**  0x004F | 不支持此参数。  **Windows Server 2003 和 Windows XP/2000：**检索屏幕保存的低功耗阶段的超时值。 *pvParam* 参数必须指向接收值的整数变量。 仅 32 位应用程序支持此标志。 | | **SPI\_GETPOWEROFFACTIVE**  0x0054 | 不支持此参数。 启用屏幕保存的关机阶段后， **GUID\_VIDEO\_POWERDOWN\_TIMEOUT** 电源设置大于零。  **Windows Server 2003 和 Windows XP/2000：**确定是否启用了屏幕保存的关机阶段。 *pvParam* 参数必须指向接收 **TRUE**（如果已启用）的 **BOOL** 变量;如果禁用，则为 **FALSE**。 仅 32 位应用程序支持此标志。 | | **SPI\_GETPOWEROFFTIMEOUT**  0x0050 | 不支持此参数。 请改为检查 **GUID\_VIDEO\_POWERDOWN\_TIMEOUT** 电源设置。  **Windows Server 2003 和 Windows XP/2000：**检索屏幕保存的关机阶段的超时值。 *pvParam* 参数必须指向接收值的整数变量。 仅 32 位应用程序支持此标志。 | | **SPI\_SETLOWPOWERACTIVE**  0x0055 | 不支持此参数。  **Windows Server 2003 和 Windows XP/2000：**激活或停用屏幕保存的低功耗阶段。 将 *uiParam* 设置为 1 可激活，将零设置为停用。 *pvParam* 参数必须为 **NULL**。 仅 32 位应用程序支持此标志。 | | **SPI\_SETLOWPOWERTIMEOUT**  0x0051 | 不支持此参数。  **Windows Server 2003 和 Windows XP/2000：**为屏幕保存的低功耗阶段设置超时值（以秒为单位）。 *uiParam* 参数指定新值。 *pvParam* 参数必须为 **NULL**。 仅 32 位应用程序支持此标志。 | | **SPI\_SETPOWEROFFACTIVE**  0x0056 | 不支持此参数。 请改为设置 **GUID\_VIDEO\_POWERDOWN\_TIMEOUT** 电源设置。  **Windows Server 2003 和 Windows XP/2000：**激活或停用屏幕保存的关机阶段。 将 *uiParam* 设置为 1 可激活，将零设置为停用。 *pvParam* 参数必须为 **NULL**。 仅 32 位应用程序支持此标志。 | | **SPI\_SETPOWEROFFTIMEOUT**  0x0052 | 不支持此参数。 相反，请将 **GUID\_VIDEO\_POWERDOWN\_TIMEOUT** 电源设置设置为超时值。  **Windows Server 2003 和 Windows XP/2000：**为屏幕保存的关机阶段设置超时值（以秒为单位）。 *uiParam* 参数指定新值。 *pvParam* 参数必须为 **NULL**。 仅 32 位应用程序支持此标志。 |     下面是屏幕保护程序参数。  展开表   |  |  | | --- | --- | | **屏幕保护程序参数** | **含义** | | **SPI\_GETSCREENSAVEACTIVE**  0x0010 | 确定是否启用屏幕保存。 *pvParam* 参数必须指向一个 **BOOL** 变量，如果启用屏幕保存，则接收 **TRUE**;否则，该变量必须指向 **FALSE**。  **Windows 7、Windows Server 2008 R2 和 Windows 2000：**即使未启用屏幕保存，函数也会返回 **TRUE** 。 | | **SPI\_GETSCREENSAVERRUNNING**  0x0072 | 确定屏幕保护程序当前是否在调用进程的窗口工作站上运行。 如果屏幕保护程序当前正在运行，*pvParam* 参数必须指向接收 **TRUE** 的 **BOOL** 变量，否则为 **FALSE**。 请注意，只有交互式窗口工作站 WinSta0 可以运行屏幕保护程序。 | | **SPI\_GETSCREENSAVESECURE**  0x0076 | 确定屏幕保护程序是否需要密码才能显示 Windows 桌面。 如果屏幕保护程序需要密码，*pvParam* 参数必须指向接收 **TRUE** 的 **BOOL** 变量，否则为 **FALSE**。 *忽略 uiParam* 参数。  **Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETSCREENSAVETIMEOUT**  0x000E | 检索屏幕节省程序超时值（以秒为单位）。 *pvParam* 参数必须指向接收值的整数变量。 | | **SPI\_SETSCREENSAVEACTIVE**  0x0011 | 设置屏幕保护程序的状态。 *uiParam* 参数指定 **TRUE** 以激活屏幕保存，或**指定 FALSE** 以停用屏幕保存。  如果计算机已进入省电模式或系统锁定状态，则会发生ERROR\_OPERATION\_IN\_PROGRESS异常。 | | **SPI\_SETSCREENSAVESECURE**  0x0077 | 设置屏幕保护程序是否要求用户输入密码才能显示 Windows 桌面。 *uiParam* 参数是 **BOOL** 变量。 *忽略 pvParam* 参数。 将 *uiParam* 设置为 **TRUE** 以要求密码，将 **FALSE** 设置为不需要密码。  如果计算机已进入省电模式或系统锁定状态，则会发生ERROR\_OPERATION\_IN\_PROGRESS异常。  **Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETSCREENSAVETIMEOUT**  0x000F | 将屏幕节省程序超时值设置为 *uiParam* 参数的值。 此值是系统在激活屏幕保护程序之前必须处于空闲状态的时间量（以秒为单位）。  如果计算机已进入省电模式或系统锁定状态，则会发生ERROR\_OPERATION\_IN\_PROGRESS异常。 |     下面是应用程序和服务的超时参数。  展开表   |  |  | | --- | --- | | **超时参数** | **含义** | | **SPI\_GETHUNGAPPTIMEOUT**  0x0078 | 检索线程在系统认为消息无响应之前无需调度消息即可完成的毫秒数。 *pvParam* 参数必须指向接收值的整数变量。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETWAITTOKILLTIMEOUT**  0x007A | 检索系统在终止不响应关闭请求的应用程序之前等待的毫秒数。 *pvParam* 参数必须指向接收值的整数变量。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETWAITTOKILLSERVICETIMEOUT**  0x007C | 检索服务控制管理器在终止不响应关闭请求的服务之前等待的毫秒数。 *pvParam* 参数必须指向接收值的整数变量。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETHUNGAPPTIMEOUT**  0x0079 | 将挂起的应用程序超时设置为 *uiParam* 参数的值。 此值是线程在系统认为消息无响应之前无需分派消息即可到达的毫秒数。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETWAITTOKILLTIMEOUT**  0x007B | 将应用程序关闭请求超时设置为 *uiParam* 参数的值。 此值是系统在终止不响应关闭请求的应用程序之前等待的毫秒数。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETWAITTOKILLSERVICETIMEOUT**  0x007D | 将服务关闭请求超时设置为 *uiParam* 参数的值。 此值是系统在终止不响应关闭请求的服务之前等待的毫秒数。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 |     下面是 UI 效果。 **SPI\_SETUIEFFECTS**值用于同时启用或禁用所有 UI 效果。 此表包含 UI 效果值的完整列表。  展开表   |  |  | | --- | --- | | **UI 效果参数** | **含义** | | **SPI\_GETCOMBOBOXANIMATION**  0x1004 | 确定是否启用组合框的滑动打开效果。 *pvParam* 参数必须指向一个 **BOOL** 变量，该变量对于 enabled 接收 **TRUE**，对于 disabled 接收 **FALSE**。 | | **SPI\_GETCURSORSHADOW**  0x101A | 确定光标周围是否有阴影。 *pvParam* 参数必须指向一个 **BOOL** 变量，该变量在启用阴影时接收 **TRUE**;如果禁用，则为 **FALSE**。 仅当系统的颜色深度超过 256 种颜色时，才会显示此效果。 | | **SPI\_GETGRADIENTCAPTIONS**  0x1008 | 确定是否启用窗口标题栏的渐变效果。 *pvParam* 参数必须指向一个 **BOOL** 变量，该变量对于 enabled 接收 **TRUE**，对于 disabled 接收 **FALSE**。 有关渐变效果的详细信息，请参阅 [GetSysColor](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-getsyscolor) 函数。 | | **SPI\_GETHOTTRACKING**  0x100E | 确定是否启用用户界面元素的热跟踪，例如菜单栏上的菜单名称。 *pvParam* 参数必须指向一个 **BOOL** 变量，该变量对于 enabled 接收 **TRUE**，对于 disabled 接收 **FALSE**。  热跟踪意味着，当光标移动到某个项上时，它将被突出显示，但未选中。 可以查询此值，以确定是否在应用程序的用户界面中使用热跟踪。 | | **SPI\_GETLISTBOXSMOOTHSCROLLING**  0x1006 | 确定是否启用列表框的平滑滚动效果。 *pvParam* 参数必须指向一个 **BOOL** 变量，该变量对于 enabled 接收 **TRUE**，对于 disabled 接收 **FALSE**。 | | **SPI\_GETMENUANIMATION**  0x1002 | 确定是否启用菜单动画功能。 必须打开此主控开关才能启用菜单动画效果。 *pvParam* 参数必须指向一个 **BOOL** 变量，该变量在启用动画时接收 **TRUE**;如果禁用动画，则为 **FALSE**。  如果启用了动画， **SPI\_GETMENUFADE** 指示菜单是使用淡出动画还是幻灯片动画。 | | **SPI\_GETMENUUNDERLINES**  0x100A | 与 **SPI\_GETKEYBOARDCUES** 相同。 | | **SPI\_GETSELECTIONFADE**  0x1014 | 确定是否启用选择淡化效果。 *pvParam* 参数必须指向一个 **BOOL** 变量，如果启用，则接收 **TRUE**;如果禁用，则接收 **FALSE**。  选择淡出效果会导致用户选择的菜单项暂时停留在屏幕上，同时在菜单关闭后逐渐消失。 | | **SPI\_GETTOOLTIPANIMATION**  0x1016 | 确定是否启用工具提示动画。 *pvParam* 参数必须指向一个 **BOOL** 变量，如果启用，则接收 **TRUE**;如果禁用，则接收 **FALSE**。 如果启用了工具提示动画， **SPI\_GETTOOLTIPFADE** 指示工具提示是使用淡出动画还是幻灯片动画。 | | **SPI\_GETTOOLTIPFADE**  0x1018 | 如果启用了**SPI\_SETTOOLTIPANIMATION，SPI\_GETTOOLTIPFADE**指示工具提示动画是使用淡出效果还是幻灯片效果。 *pvParam* 参数必须指向一个 **BOOL** 变量，该变量对于淡出动画接收 **TRUE或 FALSE**（对于幻灯片动画）。 有关幻灯片和淡化效果的详细信息，请参阅 [AnimateWindow](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-animatewindow)。 | | **SPI\_GETUIEFFECTS**  0x103E | 确定是启用或禁用 UI 效果。 *pvParam* 参数必须指向一个 **BOOL** 变量，如果启用所有 UI 效果，则接收 **TRUE**;如果禁用，则为 **FALSE**。 | | **SPI\_SETCOMBOBOXANIMATION**  0x1005 | 启用或禁用组合框的幻灯片打开效果。 将 *pvParam* 参数设置为 **TRUE** 以启用渐变效果，将 **设置为 FALSE** 以禁用渐变效果。 | | **SPI\_SETCURSORSHADOW**  0x101B | 启用或禁用光标周围的阴影。 *pvParam* 参数是 **BOOL** 变量。 将 *pvParam* 设置为 **TRUE** 以启用阴影，将 **FALSE** 设置为禁用阴影。 仅当系统的颜色深度超过 256 种颜色时，才会显示此效果。 | | **SPI\_SETGRADIENTCAPTIONS**  0x1009 | 启用或禁用窗口标题栏的渐变效果。 将 *pvParam* 参数设置为 **TRUE** 以启用它，将 **设置为 FALSE** 以禁用它。 仅当系统的颜色深度超过 256 种颜色时，才有可能产生渐变效果。 有关渐变效果的详细信息，请参阅 [GetSysColor](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-getsyscolor) 函数。 | | **SPI\_SETHOTTRACKING**  0x100F | 启用或禁用用户界面元素（例如菜单栏上的菜单名称）的热跟踪。 将 *pvParam* 参数设置为 **TRUE** 以启用它，将 **设置为 FALSE** 以禁用它。  热跟踪意味着当光标移动到某个项上时，它将被突出显示，但未选中。 | | **SPI\_SETLISTBOXSMOOTHSCROLLING**  0x1007 | 启用或禁用列表框的平滑滚动效果。 将 *pvParam* 参数设置为 **TRUE** 以启用平滑滚动效果，或 **将 FALSE** 设置为禁用它。 | | **SPI\_SETMENUANIMATION**  0x1003 | 启用或禁用菜单动画。 此主控开关必须处于打开状态，才能发生任何菜单动画。 *pvParam* 参数是 **BOOL** 变量;将 *pvParam* 设置为 **TRUE** 以启用动画，将 **FALSE** 设置为禁用动画。  如果启用了动画， **SPI\_GETMENUFADE** 指示菜单是使用淡出动画还是幻灯片动画。 | | **SPI\_SETMENUUNDERLINES**  0x100B | 与 **SPI\_SETKEYBOARDCUES** 相同。 | | **SPI\_SETSELECTIONFADE**  0x1015 | 将 *pvParam* 设置为 **TRUE** 以启用选择淡化效果，将 **设置为 FALSE** 以禁用它。  选择淡出效果会导致用户选择的菜单项暂时停留在屏幕上，同时在菜单关闭后逐渐消失。 仅当系统的颜色深度超过 256 种颜色时，才能实现选择淡出效果。 | | **SPI\_SETTOOLTIPANIMATION**  0x1017 | 将 *pvParam* 设置为 **TRUE** 以启用工具提示动画，或 **将 FALSE** 设置为禁用它。 如果启用，可以使用 **SPI\_SETTOOLTIPFADE** 指定淡出动画或幻灯片动画。 | | **SPI\_SETTOOLTIPFADE**  0x1019 | 如果启用了 **SPI\_SETTOOLTIPANIMATION** 标志，请使用 **SPI\_SETTOOLTIPFADE** 来指示 ToolTip 动画是使用淡化效果还是幻灯片效果。 对于淡化动画，请将 *pvParam* 设置为 **TRUE** ，将幻灯片动画设置为 **FALSE** 。 仅当系统的颜色深度超过 256 种颜色时，才能实现工具提示淡出效果。 有关幻灯片和淡化效果的详细信息，请参阅 [AnimateWindow](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-animatewindow) 函数。 | | **SPI\_SETUIEFFECTS**  0x103F | 启用或禁用 UI 效果。 将 *pvParam* 参数设置为 **TRUE** 可启用所有 UI 效果，将 **FALSE** 设置为禁用所有 UI 效果。 |     以下是窗口参数。  展开表   |  |  | | --- | --- | | **Window 参数** | **含义** | | **SPI\_GETACTIVEWINDOWTRACKING**  0x1000 | 确定活动窗口跟踪 (激活鼠标打开的窗口，) 打开还是关闭。 *pvParam* 参数必须指向一个 **BOOL** 变量，该变量对于 on 接收 **TRUE**，对于 off 接收 **FALSE**。 | | **SPI\_GETACTIVEWNDTRKZORDER**  0x100C | 确定是否会将通过活动窗口跟踪激活的窗口带到顶部。 *pvParam* 参数必须指向一个 **BOOL** 变量，该变量对于 on 接收 **TRUE**，对于 off 接收 **FALSE**。 | | **SPI\_GETACTIVEWNDTRKTIMEOUT**  0x2002 | 检索活动窗口跟踪延迟（以毫秒为单位）。 *pvParam* 参数必须指向接收时间的 **DWORD** 变量。 | | **SPI\_GETANIMATION**  0x0048 | 检索与用户操作关联的动画效果。 *pvParam* 参数必须指向接收信息的 [ANIMATIONINFO](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-animationinfo) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(ANIMATIONINFO)。 | | **SPI\_GETBORDER**  0x0005 | 检索确定窗口大小边框宽度的边框乘数因子。 *pvParam* 参数必须指向接收此值的整数变量。 | | **SPI\_GETCARETWIDTH**  0x2006 | 检索编辑控件中的插入点宽度（以像素为单位）。 *pvParam* 参数必须指向接收此值的 **DWORD** 变量。 | | **SPI\_GETDOCKMOVING**  0x0090 | 确定将窗口移动到监视器或监视器数组的上边缘、左边缘或右边缘时，窗口是否停靠。 *pvParam* 参数必须指向一个 **BOOL** 变量（如果启用）接收 **TRUE**;否则，该变量必须指向 **FALSE**。  使用 **SPI\_GETWINARRANGING** 确定是否启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETDRAGFROMMAXIMIZE**  0x008C | 确定拖动标题栏时是否还原最大化窗口。 *pvParam* 参数必须指向一个 **BOOL** 变量（如果启用）接收 **TRUE**;否则，该变量必须指向 **FALSE**。  使用 **SPI\_GETWINARRANGING** 确定是否启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETDRAGFULLWINDOWS**  0x0026 | 确定是否启用拖动整个窗口。 *pvParam* 参数必须指向一个 **BOOL** 变量（如果启用）接收 **TRUE**;否则，该变量必须指向 **FALSE**。 | | **SPI\_GETFOREGROUNDFLASHCOUNT**  0x2004 | 检索 [SetForegroundWindow](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-setforegroundwindow) 在拒绝前台切换请求时闪烁任务栏按钮的次数。 *pvParam* 参数必须指向接收值的 **DWORD** 变量。 | | **SPI\_GETFOREGROUNDLOCKTIMEOUT**  0x2000 | 检索用户输入后的时间（以毫秒为单位），在此期间，系统不允许应用程序强制自己进入前台。 *pvParam* 参数必须指向接收时间的 **DWORD** 变量。 | | **SPI\_GETMINIMIZEDMETRICS**  0x002B | 检索与最小化窗口关联的指标。 *pvParam* 参数必须指向接收信息的 [MINIMIZEDMETRICS](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-minimizedmetrics) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(MINIMIZEDMETRICS)。 | | **SPI\_GETMOUSEDOCKTHRESHOLD**  0x007E | 检索使用鼠标将窗口拖动到监视器或监视器阵列边缘触发停靠行为的阈值（以像素为单位）。 默认阈值为 1。 *pvParam* 参数必须指向接收值的 **DWORD** 变量。  使用 **SPI\_GETWINARRANGING** 确定是否启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETMOUSEDRAGOUTTHRESHOLD**  0x0084 | 检索使用鼠标将窗口从监视器边缘或监视器数组向中心拖动来触发取消停靠行为的阈值（以像素为单位）。 默认阈值为 20。  使用 **SPI\_GETWINARRANGING** 确定是否启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETMOUSESIDEMOVETHRESHOLD**  0x0088 | 从监视器或监视器数组的顶部检索阈值（以像素为单位），其中使用鼠标拖动时将还原垂直最大化的窗口。 默认阈值为 50。  使用 **SPI\_GETWINARRANGING** 确定是否启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETNONCLIENTMETRICS**  0x0029 | 检索与非最小化窗口的非工作区相关联的度量值。 *pvParam* 参数必须指向接收信息的 [NONCLIENTMETRICS](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-nonclientmetricsa) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(NONCLIENTMETRICS)。  **Windows Server 2003 和 Windows XP/2000：**请参阅 [NONCLIENTMETRICS 的](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-nonclientmetricsa#remarks)备注。 | | **SPI\_GETPENDOCKTHRESHOLD**  0x0080 | **检索使用笔将窗口拖动到监视器或监视器阵列边缘触发停靠行为的阈值（以像素为单位）。 默认值为 30。**  使用 **SPI\_GETWINARRANGING** 确定是否启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETPENDRAGOUTTHRESHOLD**  0x0086 | 检索使用笔将窗口从监视器或监视器阵列的边缘拖动到其中心，从而触发取消停靠行为的阈值（以像素为单位）。 默认阈值为 30。  使用 **SPI\_GETWINARRANGING** 确定是否启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETPENSIDEMOVETHRESHOLD**  0x008A | 检索监视器或监视器数组顶部的阈值（以像素为单位），其中使用鼠标拖动时将还原垂直最大化的窗口。 默认阈值为 50。  使用 **SPI\_GETWINARRANGING** 确定是否启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETSHOWIMEUI**  0x006E | 确定输入法状态窗口是否以每个用户) (可见。 *pvParam* 参数必须指向一个 **BOOL** 变量，该变量在状态窗口可见时接收 **TRUE**;如果状态窗口不可见，则为 **FALSE**。 | | **SPI\_GETSNAPSIZING**  0x008E | 确定窗口在调整到监视器或监视器数组的顶部或底部时是否垂直最大化。 *pvParam* 参数必须指向一个 **BOOL** 变量（如果启用）接收 **TRUE**;否则，该变量必须指向 **FALSE**。  使用 **SPI\_GETWINARRANGING** 确定是否启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_GETWINARRANGING**  0x0082 | 确定是否启用窗口排列。 *pvParam* 参数必须指向一个 **BOOL** 变量（如果启用）接收 **TRUE**;否则，该变量必须指向 **FALSE**。  窗口排列方式通过简化拖动或调整窗口大小时的默认行为，减少了移动顶级窗口和调整窗口大小所需的鼠标、笔或触摸交互次数。  以下参数检索单个窗口排列设置：  **SPI\_GETDOCKMOVING**  **SPI\_GETMOUSEDOCKTHRESHOLD**  **SPI\_GETMOUSEDRAGOUTTHRESHOLD**  **SPI\_GETMOUSESIDEMOVETHRESHOLD**  **SPI\_GETPENDOCKTHRESHOLD**  **SPI\_GETPENDRAGOUTTHRESHOLD**  **SPI\_GETPENSIDEMOVETHRESHOLD**  **SPI\_GETSNAPSIZING**  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETACTIVEWINDOWTRACKING**  0x1001 | 设置活动窗口跟踪 (激活鼠标打开的窗口) 打开或关闭。 对于 on，将 *pvParam* 设置为 **TRUE** ，将 **FALSE** 设置为 off。 | | **SPI\_SETACTIVEWNDTRKZORDER**  0x100D | 确定是否应将通过活动窗口跟踪激活的窗口带到顶部。 对于 on，将 *pvParam* 设置为 **TRUE** ，将 **FALSE** 设置为 off。 | | **SPI\_SETACTIVEWNDTRKTIMEOUT**  0x2003 | 设置活动窗口跟踪延迟。 将 *pvParam* 设置为激活鼠标指针下的窗口之前要延迟的毫秒数。 | | **SPI\_SETANIMATION**  0x0049 | 设置与用户操作关联的动画效果。 *pvParam* 参数必须指向包含新参数的 [ANIMATIONINFO](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-animationinfo) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(ANIMATIONINFO)。 | | **SPI\_SETBORDER**  0x0006 | 设置确定窗口大小边框宽度的边框乘数因子。 *uiParam* 参数指定新值。 | | **SPI\_SETCARETWIDTH**  0x2007 | 设置编辑控件中的插入点宽度。 将 *pvParam* 设置为所需的宽度（以像素为单位）。 默认值和最小值为 1。 | | **SPI\_SETDOCKMOVING**  0x0091 | 设置当窗口移动到监视器或监视器阵列上的顶部、左侧或右侧停靠目标时，窗口是停靠的。 对于 on，将 *pvParam* 设置为 **TRUE** ，将 **FALSE** 设置为 off。  **SPI\_GETWINARRANGING** 必须为 **TRUE** 才能启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETDRAGFROMMAXIMIZE**  0x008D | 设置拖动最大化窗口标题栏时是否还原其标题栏。 对于 on，将 *pvParam* 设置为 **TRUE** ，将 **FALSE** 设置为 off。  **SPI\_GETWINARRANGING** 必须为 **TRUE** 才能启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETDRAGFULLWINDOWS**  0x0025 | 设置打开或关闭整个窗口的拖动。 *uiParam* 参数为 on 指定 **TRUE**，为 off 指定 **FALSE**。 | | **SPI\_SETDRAGHEIGHT**  0x004D | 设置用于检测拖动操作开始的矩形的高度（以像素为单位）。 将 *uiParam* 设置为新值。 若要检索拖动高度，请使用**SM\_CYDRAG**标志调用 [GetSystemMetrics](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-getsystemmetrics)。 | | **SPI\_SETDRAGWIDTH**  0x004C | 设置用于检测拖动操作开始的矩形的宽度（以像素为单位）。 将 *uiParam* 设置为新值。 若要检索拖动宽度，请使用**SM\_CXDRAG**标志调用 [GetSystemMetrics](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-getsystemmetrics)。 | | **SPI\_SETFOREGROUNDFLASHCOUNT**  0x2005 | 设置 [SetForegroundWindow](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-setforegroundwindow) 在拒绝前台切换请求时闪烁任务栏按钮的次数。 将 *pvParam* 设置为闪烁的次数。 | | **SPI\_SETFOREGROUNDLOCKTIMEOUT**  0x2001 | 设置用户输入后的时间（以毫秒为单位），在此期间，系统不允许应用程序强制自己进入前台。 将 *pvParam* 设置为新的超时值。  调用线程必须能够更改前台窗口，否则调用将失败。 | | **SPI\_SETMINIMIZEDMETRICS**  0x002C | 设置与最小化窗口关联的指标。 *pvParam* 参数必须指向包含新参数的 [MINIMIZEDMETRICS](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-minimizedmetrics) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(MINIMIZEDMETRICS)。 | | **SPI\_SETMOUSEDOCKTHRESHOLD**  0x007F | 设置使用鼠标将窗口拖动到监视器或监视器数组边缘时触发停靠行为的阈值（以像素为单位）。 默认阈值为 1。 *pvParam* 参数必须指向包含新值的 **DWORD** 变量。  **SPI\_GETWINARRANGING** 必须为 **TRUE** 才能启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETMOUSEDRAGOUTTHRESHOLD**  0x0085 | 设置使用鼠标将窗口从监视器或监视器阵列边缘拖动到其中心时触发取消停靠行为的阈值（以像素为单位）。 默认阈值为 20。 *pvParam* 参数必须指向包含新值的 **DWORD** 变量。  **SPI\_GETWINARRANGING** 必须为 **TRUE** 才能启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETMOUSESIDEMOVETHRESHOLD**  0x0089 | 设置从监视器顶部开始的阈值（以像素为单位），在使用鼠标拖动时，垂直最大化窗口将还原到该窗口。 默认阈值为 50。 *pvParam* 参数必须指向包含新值的 **DWORD** 变量。  **SPI\_GETWINARRANGING** 必须为 **TRUE** 才能启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETNONCLIENTMETRICS**  0x002A | 设置与非小窗口的非工作区关联的指标。 *pvParam* 参数必须指向包含新参数的 [NONCLIENTMETRICS](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-nonclientmetricsa) 结构。 将此 结构的 **cbSize** 成员和 *uiParam* 参数设置为 sizeof(NONCLIENTMETRICS)。 此外，[LOGFONT](https://learn.microsoft.com/zh-cn/windows/desktop/api/wingdi/ns-wingdi-logfonta) 结构的 **lfHeight** 成员必须是负值。 | | **SPI\_SETPENDOCKTHRESHOLD**  0x0081 | 设置使用笔将窗口拖动到监视器或监视器阵列边缘时触发停靠行为的阈值（以像素为单位）。 默认阈值为 30。 *pvParam* 参数必须指向包含新值的 **DWORD** 变量。  **SPI\_GETWINARRANGING** 必须为 **TRUE** 才能启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETPENDRAGOUTTHRESHOLD**  0x0087 | 设置使用笔将窗口从监视器或监视器阵列边缘拖动到其中心时触发取消停靠行为的阈值（以像素为单位）。 默认阈值为 30。 *pvParam* 参数必须指向包含新值的 **DWORD** 变量。  **SPI\_GETWINARRANGING** 必须为 **TRUE** 才能启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETPENSIDEMOVETHRESHOLD**  0x008B | 设置监视器顶部的阈值（以像素为单位），在使用笔拖动时，垂直最大化窗口将还原到该窗口。 默认阈值为 50。 *pvParam* 参数必须指向包含新值的 **DWORD** 变量。  **SPI\_GETWINARRANGING** 必须为 **TRUE** 才能启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETSHOWIMEUI**  0x006F | 设置输入法状态窗口是否以每个用户为基础可见。 *uiParam* 参数为 on 指定 **TRUE**，为 off 指定 **FALSE**。 | | **SPI\_SETSNAPSIZING**  0x008F | 设置窗口在调整到监视器的顶部或底部时是垂直最大化的。 对于 on，将 *pvParam* 设置为 **TRUE** ，将 **FALSE** 设置为 off。  **SPI\_GETWINARRANGING** 必须为 **TRUE** 才能启用此行为。  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 | | **SPI\_SETWINARRANGING**  0x0083 | 设置是否启用窗口排列。 对于 on，将 *pvParam* 设置为 **TRUE** ，将 **FALSE** 设置为 off。  窗口排列方式通过简化拖动或调整窗口大小时的默认行为，减少了移动顶级窗口和调整窗口大小所需的鼠标、笔或触摸交互次数。  以下参数设置单个窗口排列设置：  **SPI\_SETDOCKMOVING**  **SPI\_SETMOUSEDOCKTHRESHOLD**  **SPI\_SETMOUSEDRAGOUTTHRESHOLD**  **SPI\_SETMOUSESIDEMOVETHRESHOLD**  **SPI\_SETPENDOCKTHRESHOLD**  **SPI\_SETPENDRAGOUTTHRESHOLD**  **SPI\_SETPENSIDEMOVETHRESHOLD**  **SPI\_SETSNAPSIZING**  **Windows Server 2008、Windows Vista、Windows Server 2003 和 Windows XP/2000：**不支持此参数。 |   [in] uiParam  类型： **UINT**  一个参数，其用法和格式取决于要查询或设置的系统参数。 有关系统范围参数的详细信息，请参阅 *uiAction* 参数。 如果未另行指示，则必须为此参数指定零。  [in, out] pvParam  类型： **PVOID**  一个参数，其用法和格式取决于要查询或设置的系统参数。 有关系统范围参数的详细信息，请参阅 *uiAction* 参数。 如果未另行指示，则必须为此参数指定 **NULL** 。 有关 **PVOID** 数据类型的信息，请参阅 [Windows 数据类型](https://learn.microsoft.com/zh-cn/windows/desktop/WinProg/windows-data-types)。  [in] fWinIni  类型： **UINT**  如果正在设置系统参数，则指定是否要更新用户配置文件，如果是，则指定是否将 [WM\_SETTINGCHANGE](https://learn.microsoft.com/zh-cn/windows/desktop/winmsg/wm-settingchange) 消息广播到所有顶级窗口，以通知他们更改。  如果不想更新用户配置文件或广播 [WM\_SETTINGCHANGE](https://learn.microsoft.com/zh-cn/windows/desktop/winmsg/wm-settingchange) 消息，此参数可以为零，也可以是以下一个或多个值。  展开表   |  |  | | --- | --- | | **值** | **含义** | | **SPIF\_UPDATEINIFILE** | 将新的系统范围参数设置写入用户配置文件。 | | **SPIF\_SENDCHANGE** | 在更新用户配置文件后广播 [WM\_SETTINGCHANGE](https://learn.microsoft.com/zh-cn/windows/desktop/winmsg/wm-settingchange) 消息。 | | **SPIF\_SENDWININICHANGE** | 与 **SPIF\_SENDCHANGE** 相同。 |  返回值 类型： **BOOL**  如果函数成功，则返回值为非零值。  如果函数失败，则返回值为零。 要获得更多的错误信息，请调用 GetLastError。 注解 此函数适用于允许用户自定义环境的应用程序。  键盘布局名称应派生自对应于布局的语言标识符的十六进制值。 例如，美国英语的语言标识符为 0x0409，因此美国英语的主要布局名为“00000409”。 美国英语版式的变体（如 Dvorak 布局）名为“00010409”、“00020409”等。 有关构成语言标识符的主要语言标识符和子语言标识符的列表，请参阅 **MAKELANGID** 宏。  高对比度配色方案和高对比度模式之间存在差异。 高对比度配色方案将系统颜色更改为具有明显对比度的颜色;通过使用控制面板中的“显示选项”切换到此配色方案。 高对比度模式使用 **SPI\_GETHIGHCONTRAST** 和 **SPI\_SETHIGHCONTRAST**，建议应用程序修改视觉障碍用户的外观。 它涉及使用控制面板) 中的辅助功能选项 (向用户发出声音警告和自定义配色方案等内容。 有关详细信息，请参阅 [HIGHCONTRAST](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/ns-winuser-highcontrasta)。 有关常规辅助功能的详细信息，请参阅 [辅助功能](https://learn.microsoft.com/zh-cn/windows/desktop/accessibility)。  在按住主按钮以激活鼠标 ClickLock 功能期间，用户可以移动鼠标。 锁定主按钮后，释放主按钮不会导致 **WM\_LBUTTONUP** 消息。 因此，它会在应用程序上显示主按钮仍处于关闭状态。 任何后续按钮消息都会释放主按钮，向应用程序发送 **WM\_LBUTTONUP** 消息，因此可以通过编程方式或通过用户单击任何按钮来解锁该按钮。  此 API 不可识别 DPI，如果调用线程按监视器感知 DPI，则不应使用此 API。 有关此 API 的 DPI 感知版本，请参阅 [SystemParametersInfoForDPI](https://learn.microsoft.com/zh-cn/windows/desktop/api/winuser/nf-winuser-systemparametersinfofordpi)。 有关 DPI 感知的详细信息，请参阅 [Windows 高 DPI 文档。](https://learn.microsoft.com/zh-cn/windows/desktop/hidpi/high-dpi-desktop-application-development-on-windows) 示例 以下示例使用 **SystemParametersInfo** 将鼠标速度加倍。  C++复制  #include <windows.h>  #include <stdio.h>  #pragma comment(lib, "user32.lib")  void main()  {  BOOL fResult;  int aMouseInfo[3]; // Array for mouse information    // Get the current mouse speed.  fResult = SystemParametersInfo(SPI\_GETMOUSE, // Get mouse information  0, // Not used  &aMouseInfo, // Holds mouse information  0); // Not used    // Double it.  if( fResult )  {  aMouseInfo[2] = 2 \* aMouseInfo[2];    // Change the mouse speed to the new value.  SystemParametersInfo(SPI\_SETMOUSE, // Set mouse information  0, // Not used  aMouseInfo, // Mouse information  SPIF\_SENDCHANGE); // Update Win.ini  }  }  **备注**  winuser.h 标头将 SystemParametersInfo 定义为别名，该别名根据 UNICODE 预处理器常量的定义自动选择此函数的 ANSI 或 Unicode 版本。 将非特定编码别名的使用与非非特定编码的代码混合使用可能会导致不匹配，从而导致编译或运行时错误。 有关详细信息，请参阅 [**函数原型的约定**](https://learn.microsoft.com/zh-cn/windows/win32/intl/conventions-for-function-prototypes)。 |

## 这个函数c#也可以调用

view plaincopy to clipboardprint?  
[DllImport("user32.dll", CharSet = CharSet.Auto)]    
public static extern int SystemParametersInfo(int uAction, int uParam, string lpvParam, int fuWinIni);

其中各参数的意义如下表:

•uAction Long，指定要设置的参数。参考uAction常数表  
•uParam Long，参考uAction常数表  
•lpvParam Any，按引用调用的Integer、Long和数据结构。  
•fuWinIni 这个参数规定了在设置系统参数的时候，是否应更新用户设置参数  
•下面是部分uAction参数，和使用它们的方法：  
•参数    意义和使用方法    
•6    设置视窗的大小，SystemParametersInfo(6, 放大缩小值, P, 0)，lpvParam为long型  
•17    开关屏保程序，SystemParametersInfo(17, False, P, 1)，uParam为布尔型  
•13，24    改变桌面图标水平和垂直间距，uParam为间距值(像素)，lpvParam为long型  
•15    设置屏保等待时间，SystemParametersInfo(15, 秒数, P, 1)，lpvParam为long型  
•20    设置桌面背景墙纸，SystemParametersInfo(20, True, 图片路径, 1)  
•93    开关鼠标轨迹，SystemParametersInfo(93, 数值, P, 1)，uParam为False则关闭  
•97    开关Ctrl+Alt+Del窗口，SystemParametersInfo(97, False, A, 0)，uParam为布尔  
•spi\_seticontitlelogfont lpvparam是一个logfont结构，用于定义图标标题采用的字体。uparam指定logfont结构的大小。如两者均为null，则采用系统启动时定义的字体  spi\_setkeyboarddelay uparam代表新的键盘重复延时

•spi\_setkeyboardpref 若uparam为true，表明用户相比键盘和鼠标更喜欢用前者。参考spi\_getkeyboardpref

•spi\_setkeyboardspeed uparam代表新的键盘重复速率

•spi\_setlangtoggle 令窗口从系统注册表中重新装载热键信息，以便在键盘布局之间交换

•spi\_setlowpoweractive uparam为true时允许节能屏幕保护程序模式。只适用于win95

•spi\_setlowpowertimeout uparam代表新的节能屏幕保护程序延时。只适用于win95

•spi\_setmenudropalignment uparam为false时设置弹出菜单左对齐，true则右对齐

•spi\_setminimizedmetrics 参数与spi\_getminimizedmetrics一样

•spi\_setmouse lpvparam是一个三元素整数数组的头一个条目，其中包含了鼠标设置。参考spi\_getmouse

•spi\_setmousebuttonswap uparam为true时交换左右鼠标按钮的功能；如果为false，则使用原来的值

•spi\_setmousekeys 参数与spi\_getmousekeys一样

•spi\_setnonclientmetrics 参数与spi\_getnonclientmetrics一样

•spi\_setpenwindows uparam为true时载入笔窗口，false则卸载

•spi\_setpoweroffactive uparam为true时允许停电屏幕保护模式。只适用于win95

•spi\_setpowerofftimeout uparam代表新的停电屏幕保护程序延时。只适用于win95

•spi\_setscreenreader uparam为true时表示正在运行一个屏幕阅读程序。参考spi\_getscreenreader

•spi\_setscreensaveactive uparam为true时激活屏幕保护程序；false表示禁止

•spi\_setscreensavetimeout uparam代表新的屏幕保护延迟时间，以秒为单位

•spi\_setserialkeys 参数与 spi\_getserialkeys 相同

•spi\_setshowsounds uparam为true时允许窗口开启声音提示辅助访问特性。参考spi\_getshowsounds

•spi\_setsnaptodefbutton 如鼠标应自动移到新对话框的默认按钮，则uparam为true。只适用于nt 4.0

•spi\_setsoundsentry 与 spi\_getsoundsentry 相同

•spi\_setstickykeys 与 spi\_getstickykeys 相同

•spi\_settogglekeys 与 spi\_gettogglekeys 相同

•spi\_setworkarea 与 spi\_getworkarea 相同

# 演练

## 项目1，获取系统信息

## 1.在项目的根目录F:\Projects\_f\Learn-win-API下面新建一个文件夹Lesson24-more-sys-info

|  |
| --- |
|  |

## 2.打开vs2010，在Lessons4-more-sys-info里面新建一个项目，取名：01\_windir-and-sysdir,是一个空项目，然后新建一个cpp文件取名：windir-and-sysdir，然后添加一些骨架代码

|  |
| --- |
| #include<Windows.h>  #include<stdio.h>  #include<stdlib.h>  int main()  {    system("pause");  return 0;  } |

## 3.然后我们添加一些获取系统信息的代码如下

|  |
| --- |
| #include<Windows.h>  #include<stdio.h>  #include<stdlib.h>  int main()  {  DWORD dwPcNameLength = MAX\_COMPUTERNAME\_LENGTH+1;  TCHAR szSysDir[MAX\_PATH],szWinDir[MAX\_PATH],szPCName[MAX\_COMPUTERNAME\_LENGTH+1];  DWORD dwUserNameLength = 64;  TCHAR szUserName[64];  GetSystemDirectory(szSysDir,MAX\_PATH);  GetWindowsDirectory(szWinDir,MAX\_PATH);  GetComputerName(szPCName,&dwPcNameLength);//这里有坑，需要注意  GetUserName(szUserName,&dwUserNameLength);  printf("系统目录：%s\n",szSysDir);  printf("Windows目录：%s\n",szWinDir);  printf("电脑名称：%s\n",szPCName);  printf("当前用户名：%s\n",szUserName);  system("pause");  return 0;  } |

### 效果如下

|  |
| --- |
|  |

## 项目2.获取外设信息

## 1.新建一个常规空项目，取名02\_device\_info,然后新建一个cpp文件，取名device\_info.cpp,添加一些骨架代码

|  |
| --- |
|  |

## 2.添加获取鼠标消息的代码如下

|  |
| --- |
| #include<Windows.h>  #include<stdio.h>  #include<stdlib.h>  int main()  {  BOOL fResult;  int mouseInfo[3];  fResult = SystemParametersInfo(SPI\_GETMOUSE,0,&mouseInfo,0);  if(!fResult)  {  printf("Get Mouse Info Failed:%d\n",GetLastError());  return 1;  }  printf("first element:%d\n",mouseInfo[0]); //6  printf("second element:%d\n",mouseInfo[1]);//10  printf("third element:%d\n",mouseInfo[2]);//1,这个在指鼠标移动速度  system("pause");  return 0;  } |

### 效果

|  |
| --- |
|  |

# 扩展，这些函数也有对应的Set版本，我们可以自己学习一下