**PenPower A6 Scan Control SDK**

**.NET API Reference**

版本1.0.2

**PPScanControl 命名空间**

**类别**

[ScanControl](#ScanControl_Class) Class

**方法：**

[GetApiVersion](#GetApiVersion)

[GetConnectedScanner](#GetConnetedScanner)

[GetOpenedScanner](#GetOpenedScanner)

[Open](#Open)

[IsCalibrateNeeded](#IsCalibrateNeeded)

[Calibrate](#Calibrate)

[StartScan](#StartScan)

[StartScanBlocking](#StartScanBlocking)

[Close](#Close)

**事件：**

[ScanCompleted](#ScanCompleted)

[ButtonPressed](#ButtonPressed)

[ScanCompletedEventArgs](#ScanCompletedEventArgs_Class) Class

[ButtonEventArgs](#ButtonEventArgs_Class) Class

**列举**

[ScannerType](#ScannerType_definition) Enum

[ImageType](#ImageType_definition) Enum

[ReturnCode](#Return_code_definition) Enum

**ScanControl 类别**

1. **GetApiVersion**

取得此SDK函式库的版本号码。

|  |
| --- |
| string GetApiVersion(); |

**Parameter: void**

**Return value: string**

回传SDK版本号码，以三个数字分隔的字符串格式(例：“1.0.0”)。

1. **GetConnetedScanner**

取得目前连接的扫描仪ID。

|  |
| --- |
| ScannerType GetConnectedScanner(); |

**Parameter: void**

**Return value: ScannerType**

回传目前计算机USB可侦测到的扫描仪ID。若侦测到一种以上扫描仪，则回传ID A6\_7P3。

1. **GetOpenedScanner**

取得目前已开启的扫描仪ID。

|  |
| --- |
| ScannerType GetOpenedScanner(); |

**Parameter: void**

**Return value: ScannerType**

回传目前SDK正在开启使用中的扫描仪ID。

1. **Open**

开启指定的扫描仪。

|  |
| --- |
| ReturnCode Open(ScannerType scannerType); |

**Parameters:**

**ScannerType scannerType**

[In] 指定要开启的扫描仪ID。

**Return value: ReturnCode**

参阅回传值定义。

1. **IsCalibrateNeeded**

检查目前开启的扫描仪是否需要校正。

|  |
| --- |
| ReturnCode IsCalibrateNeeded(); |

**Parameters: void**

**Return value: ReturnCode**

若扫描仪需要校正，则此方法回传NeedCalibrate。若扫描仪不需要校正，此方法回传Success。其他错误代码请参阅回传值定义。

1. **Calibrate**

校正目前开启的扫描仪。

|  |
| --- |
| ReturnCode Calibrate(); |

**Parameter: void**

**Return value: ReturnCode**

若校正正常结束，则此方法回传Success。若校正失败，则此方法回传NeedCalibrate。其他错误代码请参阅回传值定义。

**Remarks:**

此方法阻挡直到校正结束后回传。此方法没有互动UI。

1. **StartScan**

开始扫描流程。

|  |
| --- |
| ReturnCode StartScan(int dpi, ImageType type, string outputFileName); |

**Parameter:**

**int dpi**

[In] 指定扫描的DPI。扫描仪ID A6\_73F最低DPI是300，A6\_7P3最低DPI是200。

**ImageType type**

[In] 指定扫描的影像格式。

**string outputFileName**

[In] 指定输出图像文件案的完整路径。若此档案存在，将会被覆盖。若 outputFileName为null，SDK将会以内存方式回传影像，而不是磁盘上的档案。

**Return value: ReturnCode**

参阅回传值定义。

**Remarks:**

此方法呼叫后即回传。若此方法成功，扫描的结果则会经由[ScanCompleted](#ScanCompleted) 事件来触发。

1. **StartScanBlocking**

开始扫描流程。

|  |
| --- |
| ScanCompletedEventArgs StartScanBlocking(int dpi, ImageType type, string outputFileName); |

**Parameter:**

**int dpi**

[In] 指定扫描的DPI。扫描仪ID A6\_73F最低DPI是300，A6\_7P3最低DPI是200。

**ImageType type**

[In] 指定扫描的影像格式。

**string outputFileName**

[In] 指定输出图像文件案的完整路径。若此档案存在，将会被覆盖。若 outputFileName为null，SDK将会以内存方式回传影像，而不是磁盘上的档案。

**Return value: ScanCompletedEventArgs**

参阅[ScanCompletedEventArgs](#ScanCompletedEventArgs_Class)类别定义。

**Remarks:**

此方法会阻挡直到扫描完成。不会触发[ScanCompleted](#ScanCompleted) 事件。

1. **Close**

关闭已开启的扫描仪，中断扫描流程。将会释放所有资源，并卸除已连接的装置。

|  |
| --- |
| void Close(); |

**Parameter: void**

**Return value: void**

1. **ScanCompleted Event**

扫描完成后发生的事件。

|  |
| --- |
| event EventHandler<ScanCompletedEventArgs> ScanCompleted; |

**Remarks:**

此事件将在扫描成功后触发，或是扫描/处理图片发生错误时触发。请参考类别[ScanCompletedEventArgs](#ScanCompletedEventArgs_Class)的定义。

1. **ButtonPressed Event**

硬件按钮按下后发生的事件。

|  |
| --- |
| event EventHandler<ButtonEventArgs> ButtonPressed; |

**Remarks:**

此事件将在A6\_7P3按钮按下后触发，其他扫描仪不会触发此事件。请参考类别[ButtonEventArgs](#ButtonEventArgs_Class)的定义。

**ScanCompletedEventArgs 类别**

**字段**

**ReturnCode Result**

参阅回传值定义。

**string FileName**

输出的图像文件案。

若结果是Success，此字段包含[StartScan](#StartScan)方法所指定的outputFileName扫描影像完整路径。否则此字段是null。

**string ImageData**

输出的内存内影像，以Base 64格式编码的字符串。

若结果是Success，且扫描时outputFileName字段是null，此字段会包含影像数据。否则此字段是null。

**ButtonEventArgs 类别**

**字段**

**int ButtonId**

于A6\_7P3扫描仪上面的按钮代号。[Scan]按钮是1，而[Custom]按钮是2。

**回传值定义**

public enum ReturnCode

{

Success = 0,

Busy,

InvalidParameter,

IOException,

ScannerNotFound,

ScannerOpenFailed,

NoPaper,

PaperJam,

NeedCalibrate,

DllNotFound,

DriverError,

Uninitalized,

OtherError

}

|  |  |  |
| --- | --- | --- |
| Error code | Value | Explanation |
| Success | 0 | 操作成功 |
| Busy | 1 | SDK正在执行其他操作 |
| InvalidParameter | 2 | 指定了无效的参数 |
| IOException | 3 | 扫描中发生I/O错误 |
| ScannerNotFound | 4 | 指定的扫描仪无效或是不存在 |
| ScannerOpenFailed | 5 | 无法开启扫描仪 |
| NoPaper | 6 | 扫描仪纸匣内没有纸张 |
| PaperJam | 7 | 扫描仪卡纸 |
| NeedCalibrate | 8 | 扫描仪需要校正 |
| DllNotFound | 9 | 找不到相依的DLL档案 |
| DriverError | 10 | 扫描仪的驱动程序发生错误 |
| Uninitalized | 11 | 需要[Open](#Open)后才能执行此方法 |
| OtherError | 12 | 其他未定义错误 |

**ScannerType定义**

指定的扫描仪ID。A6\_7P3是2022年度黑色新版机器。A6\_73F是白色旧版机器。

public enum ScannerType

{

None = 0,

A6\_7P3,

A6\_73F

}

**ImageType定义**

public enum ImageType

{

BMP = 0,

JPG,

PNG

}