* **API Reference**

L398介面：(可與E550/E560/L370/L398/M398共用)

[**PPL398\_InitialDevice**](#PPL398_InitialDevice)

[**PPL398\_InitialDeviceByWId**](#PPL398_InitialDeviceByWId)

[**PPL398\_UnInitialDevice**](#PPL398_UnInitialDevice)

[**PPL398\_OpenLCD**](#PPL398_OpenLCD)

[**PPL398\_Clear**](#PPL398_Clear)

[**PPL398\_GetHWPenID**](#PPL398_GetHWPenID)

[**PPL398\_GetHWPadID**](#PPL398_GetHWPadID)

[**PPL398\_GetHWVersionID**](#PPL398_GetHWVersionID)

[**PPL398\_DisplayHWVersion**](#PPL398_DisplayHWVersion)

[**PPL398\_SetPenStyle**](#PPL398_SetPenStyle)

[**PPL398\_SetPenWidth**](#PPL398_SetPenWidth)

[**PPL398\_SignatureStatusCallback**](#PPL398_SignPadEventCallback)

[**PPL398\_GetTotalPacketsNumber**](#PPL398_GetTotalInkPointsNumber)

[**PPL398\_getPackets**](#PPL398_GetInkPointData)

[**PPL398\_GetInkPointTime**](#PPL398_GetInkPointDataTimeInfo)

[**PPL398\_FreeInkPointTime**](#PPL398_FreeInkPointDataTimeInfo)

[**PPL398\_SaveDrawingImage**](#PPL398_SaveDrawingImage)

[**PPL398\_SetSaveCenterImageClip**](#PPL398_SetSaveClipImageMargin)

[**PPL398\_getDeviceInfo**](#PPL398_GetDeviceInfo)

[**PPL398\_AboutBox**](#PPL398_AboutBox)

[**PPL398\_AboutBoxByWId**](#PPL398_AboutBoxByWId)

[**PPL398\_PacketsBase64Encode**](#PPL398_Base64Encode)

[**PPL398\_PacketsBase64EncodeFree**](#PPL398_Base64EncodeFree)

[**PPL398\_Base64Decode**](#PPL398_Base64Decode)

[**PPL398\_GetTotalDecodePacketsNumber**](#PPL398_GetTotalDecodeInkPointsNumber)

[**PPL398\_GetDecodePacketsData**](#PPL398_GetDecodeInkPointData)

[PPL398\_SetPenColor](#PPL398_SetPenColor)

[PPL398\_RealTimeSignatureDataCallback](#PPL398_RealTimeSignatureDataCallback)

L398介面: (可與E550/E560/L370/L398/M398共用)

**PPL398\_InitialDevice**

設定L398簽名板，初始化簽名視窗以及設定簽名畫布大小。

bool PPL398\_InitialDevice(int nDeviceType, QWidget

\*pParentWidget, int nX, int nY, int nWidth, int nHeight)

**Parameter**

***nDeviceType***

[in] 設備類型ID。

設備ID定義：

enum DEVICE\_TYPE\_ID

{

PPL398\_DEVICE = 2

};

***pParentWidget***

[in] 父視窗指標。

***nX***

[in] 簽名視窗起始點x座標。

***nY***

[in] 簽名視窗起始點y座標。

***nWidth***

[in] 簽名視窗寬度。

***nHeight***

[in] 簽名視窗高度。

**Remark**

1. 可使用PPL398\_InitialDevice去設定畫布大小。
2. 如需重新設定新畫布大小時，必須先使用PPL398\_UnInitialDevice後，再使用PPL398\_InitialDevice設置一次。

**Return Value**

參閱回傳值定義

**PPL398\_InitialDeviceByWId**

設定L398簽名板，初始化簽名視窗以及設定簽名畫布大小。

bool PPL398\_InitialDeviceByWId(int nDeviceType, WId

\*parentWId, int nX, int nY, int nWidth, int nHeight)

**Parameter**

***nDeviceType***

[in] 設備類型ID。

設備ID定義：

enum DEVICE\_TYPE\_ID

{

PPL398\_DEVICE = 2

};

***parentWId***

[in] 父視窗指標WId。

***nX***

[in] 簽名視窗起始點x座標。

***nY***

[in] 簽名視窗起始點y座標。

***nWidth***

[in] 簽名視窗寬度。

***nHeight***

[in] 簽名視窗高度。

**Remark**

1. 可使用PPL398\_InitialDeviceByWId去設定畫布大小。
2. 如需重新設定新畫布大小時，必須先使用PPL398\_UnInitialDevice後，再使用PPL398\_InitialDeviceByWId設置一次。

**Return Value**

參閱回傳值定義

**PPL398\_UnInitialDevice**

卸載L398簽名板，關閉簽名視窗，釋放資源。

int PPL398\_UnInitialDevice(int nDeviceType)

**Parameter**

***nDeviceType***

[in] 設備類型ID。

**Return value**

參閱回傳值定義

**PPL398\_OpenLCD**

打開L398簽名板LCD

int PPL398\_OpenLCD(int nDeviceType, bool bOn)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***bOn***

【打開LCD】：bOn = true

【關閉LCD】：bOn = false

**Return Value**

參閱回傳值定義

**PPL398\_Clear**

清除L398簽名板畫布上的筆跡紀錄。

int PPL398\_Clear (int nDeviceType)

**Parameter**

***nDeviceType***

[in] 設備類型ID

**Return value**

參閱回傳值定義

**PPL398\_GetHWPenID**

獲取L398簽名板Pen ID

int PPL398\_GetHWPenID(int nDeviceType, char\* pPenID, int

nMaxCount)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***pPenID***

[out] 輸出Pen ID字串指標

***nMaxCount***

[in] 輸出Pen ID字串的最大長度

**Return Value**

參閱回傳值定義

**PPL398\_GetHWPadID**

獲取L398簽名板Pad ID

int PPL398\_GetHWPadID(int nDeviceType, char\* pPadID, int

nMaxCount)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***pPadID***

[out] 輸出Pad ID字串指標

***nMaxCount***

[in] 輸出Pad ID字串的最大長度

**Return Value**

參閱回傳值定義

**PPL398\_GetHWVersionID**

獲取L398簽名板的版本ID

int PPL398\_GetHWVersionID(int nDeviceType, char\* pVersion, int

nMaxCount)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***pVersion***

[out] 輸出版本ID字串指標

***nMaxCount***

[in] 輸出版本ID字串的最大長度

**Return Value**

參閱回傳值定義

**PPL398\_DisplayHWVersion**

開關L398簽名板上版本資訊的顯示

Int PPL398\_DisplayHWVersion(int nDeviceType, bool bOn)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***bOn***

【顯示】：bOn = true

【關閉】：bOn = false

**Return Value**

參閱回傳值定義

**PPL398\_SetPenStyle**

設定L398簽名板畫筆種類。

int PPL398\_SetPenStyle (int nDeviceType, bool bPressureOn)

**Parameter**

***nDeviceType***

[in] 設備類型ID。

***bPressureOn***

[in] 畫筆種類。

【壓感筆】：bPressureOn = true

【定寬筆】：bPressureOn = false

**Return value**

參閱回傳值定義

**PPL398\_SetPenWidth**

設定L398簽名板畫筆寬度。

int PPL398\_SetPenWidth (int nDeviceType, int nPenWidth)

**Parameter**

***nDeviceType***

[in] 設備類型ID。

***nPenWidth***

[in] 畫筆寬度值。(1～10)

**Return value**

參閱回傳值定義

**PPL398\_SignatureStatusCallback**

設定L398簽名板上按【確定】或【取消】時的回傳函數

LONG PPL398\_SignatureStatusCallback)(int nDeviceType,

NOTIFYSIGNSTATUS pfnSignatureStatusCallback)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***pfnSignatureStatusCallback***

[in] 按【確定】或【取消】時的回傳函數位址

回傳函式定義：

typedef void (\*NOTIFYSIGNSTATUS)(int nEventID);

**Remark**

當L398簽名板按【確定】或【取消】時會調用pfnSignatureStatusCallback。

**Return Value**

參閱回傳值定義

**PPL398\_GetTotalPacketsNumber**

取得L398簽名板目前筆跡陣列點紀錄數量。

int PPL398\_GetTotalPacketsNumber(int nDeviceType)

**Parameter**

***nDeviceType***

[in] 設備類型ID

**Return value**

回傳筆跡陣列點數量, 但如果筆跡點為空或者其他錯誤的話, 將會回傳負的

錯誤值

**PPL398\_getPackets**

獲取L398簽名板單點筆跡資料

Int PPL398\_getPackets(int nDeviceType, PACKETS&

currentInkPoint)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***currentInkPoint***

[out] 筆跡資料結構體。

筆跡資料結構體:

typedef struct \_PACKETS

{

int nIndex;

short nX;

short nY;

short nPressureValue;

bool bStrokeEnd;

} PACKETS, \*PPACKETS;

**Return Value**

參閱回傳值定義

**PPL398\_GetInkPointTime**

獲取L398簽名板單點筆跡的時間資訊。

char\* PPL398\_GetInkPointTime(int nDeviceType, int nIndex)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***nIndex***

[in] 單點筆跡序號

**Return value**

單點筆跡時間資訊字串。

**PPL398\_FreeInkPointTime**

釋放L398簽名板單點筆跡時間資訊字串。

void PPL398\_FreeInkPointTime(int nDeviceType, char\*

pInkPointTime)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***pInkPointTime***

[in] 單點筆跡時間資訊字串

**Return value**

無。

**PPL398\_SaveDrawingImage**

儲存L398簽名板簽名筆跡內容至支援的影像檔案。

int PPL398\_SaveDrawingImage(int nDeviceType, int

nImageTypeID, wchar\_t\* pImageFilePath, int

nDPILevel)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***nImageTypeID***

[in] 儲存影像檔案類型

1 : BMP

2 : JPG

3 : PNG

4 : PDF

5 : SVG

***pImageFilePath***

[in] 儲存影像檔案名稱

***nDPILevel***

[in] 保存影像解析度等級。(0～1)

0：150 DPI

1：300 DPI

**Remark**

支援的影像檔案類型總共有BMP/JPG/PNG/PDF/SVG等五種格式，pImageFilePath為儲存影像的完整檔案名稱。

**Return value**

參閱回傳值定義

**PPL398\_SetSaveCenterImageClip**

設定裁切L398簽名板簽名筆跡影像的水平和垂直邊界。

int PPL398\_SetSaveCenterImageClip(int nDeviceType, bool

bClipImage, int nClipMarginX, int nClipMarginY)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***bClipImage***

[in] 是否要裁切筆跡影像?

【是】：bClipImage = true

【否】：bClipImage = false

***nClipMarginX***

[in] 裁切筆跡的同時, 並額外保留的水平邊界。

***nClipMarginY***

[in] 裁切筆跡的同時, 並額外保留的垂直邊界。

**Remark**

1) nClipMarginX，nClipMarginY要有作用，必須bClipImage等於Ture。

2)需搭配PPL398\_SaveDrawingImage API才能儲存裁切筆跡影像。

**Return value**

參閱回傳值定義

**PPL398\_getDeviceInfo**

獲取L398簽名板相關資訊

int PPL398\_getDeviceInfo(int nDeviceType, int nIndex, void\*

pOutput)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***nIndex***

[in] 設備資訊序號

1 ：設備當前狀態

2 ：簽名設備的X的座標範圍

3 ：簽名設備的Y的座標範圍

4 ：簽名設備的最大壓力值

5 ：簽名設備的廠商名字

6 ：簽名設備的設備名字

7 ：簽名設備的SDK版本

***pOutput***

[out] 獲取的設備資訊

**Return Value**

參閱回傳值定義

**PPL398\_AboutBox**

顯示SDK版本資訊。

void PPL398\_AboutBox(QWidget \*pParent)

**Parameter**

***pParent***

[in] 父視窗指標。

**Return value**

無。

**PPL398\_AboutBoxByWId**

顯示SDK版本資訊。

void PPL398\_AboutBoxByWId(WId parentWId)

**Parameter**

***pParent***

[in] 父視窗指標WId。

**Return value**

無。

**PPL398\_PacketsBase64Encode**

將L398簽名板簽名筆跡以指定的影像格式轉換為Base64編碼的字串。

char\* PPL398\_PacketsBase64Encode (int nDeviceType, int nImageTypeID)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***nImageTypeID***

[in] 欲編碼的影像格式。

1 : BMP

2 : JPG

3 : PNG

4 : INK

**Return value**

回傳編碼後的字串, 若失敗會回傳NULL

**PPL398\_PacketsBase64EncodeFree**

釋放Base64編碼佔用的記憶體，刪除編碼資料。

void PPL398\_PacketsBase64EncodeFree(int nDeviceType, char\* pData)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***pData***

[in] Base64編碼資料指標

**Return value**

無

**PPL398\_Base64Decode**

將Base64編碼的字串解碼為原始筆跡資料。

int PPL398\_Base64Decode(int nDeviceType, int nImageTypeID,

const char\* pEncodedData, wchar\_t\* pImageFilePath)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***nImageTypeID***

[in] 編碼的影像格式。

1 : BMP

2 : JPG

3 : PNG

4 : INK

***pEncodedData***

[in] 編碼的資料字串

***pImageFilePath***

[in] 解碼資料儲存完整檔案名稱

**Return Value**

參閱回傳值定義

**PPL398\_GetTotalDecodePacketsNumber**

獲取解碼後得到的筆跡點數。

int PPL398\_GetTotalDecodePacketsNumber (int nDeviceType)

**Parameter**

***nDeviceType***

[in] 設備類型ID

**Return value**

回傳筆跡陣列點數量, 但如果筆跡點為空或者其他錯誤的話, 將會回傳負的

錯誤值

**PPL398\_GetDecodePacketsData**

獲取解碼後得到的單點筆跡資料。

int PPL398\_GetDecodePacketsData (int nDeviceType, int nIndex,

int nTypeID)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***nIndex***

[in] 筆跡點在陣列中的序號

***nTypeID***

[in] 取值種類。

1：取得X座標

2：取得Y座標

3：取得是否為每筆劃結束點狀態

**Return Value**

參閱回傳值定義

**PPL398\_SetPenColor**

設定L398畫筆顏色。

int PPL398\_SetPenColor (int nDeviceType, int nR, int nG, int nB)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***nR***

[in] 畫筆紅色比例值。 (0~255)

***nG***

[in] 畫筆綠色比例值。 (0~255)

***nB***

[in] 畫筆藍色比例值。 (0~255)

**Return value**

參閱回傳值定義

**PPL398\_RealTimeSignatureDataCallback**

設定獲取L398簽名板上簽名即時數據的回傳函數

bool PPL398\_RealTimeSignatureDataCallback)(int nDeviceType,

NOTIFYREALTIMESIGNDATA pfnRealTimeSignDataCallback)

**Parameter**

***nDeviceType***

[in] 設備類型ID

***pfnRealTimeSignDataCallback***

[in] 獲取L398簽名板上簽名即時數據的回傳函數位址

回傳函式定義：

typedef void (\*NOTIFYREALTIMESIGNDATA)(WPARAM wParam, LPARAM lParam);

**Remark**

L398簽名板上簽名即時數據會調用pfnSignatureStatusCallback。

**Return Value**

參閱回傳值定義

**回傳值:**

|  |  |  |
| --- | --- | --- |
| 定義名稱 | 定義值 | 描述 |
| ARI\_SUCCESS | 0 | 成功 |
| ARI\_FAIL | -1 | 失敗 |
| ARI\_INVALID\_PARAMETERS | -2 | 傳入參數錯誤 |
| ARI\_DRIVER\_INIT\_FAIL | -3 | 【驅動】初始化失敗 |
| ARI\_DRIVER\_OPEN\_DEVICE\_FAIL | -4 | 【驅動】打開設備失敗 |
| ARI\_DRIVER\_CLAIM\_INTERFACE\_FAIL | -5 | 【驅動】宣告介面失敗 |
| ARI\_DRIVER\_CREATE\_THREAD\_FAIL | -6 | 【驅動】建立Thread失敗 |
| ARI\_DRIVER\_SENT\_CMD\_FAIL | -7 | 【驅動】下命令失敗 |
| ARI\_INK\_EMPTY | -8 | 筆跡陣列點為空 |