



HID 範本應用指南

版本：V1.00 日期：2024-03-29

www.holtek.com

目錄

1. HID 範本簡介	3
2. HID 範本架構	3
3. HID 範本檔案說明	4
4. MCU 程式說明	5
4.1 程式主要流程	5
4.2 資料傳遞	6
4.3 define.h	8
4.4 mcu.h	9
5. 軟體使用說明	10
5.1 HID_INOUT	10
5.2 ConsoleApplicationCS	11
5.3 ConsoleApplicationVB	11
6. HIDAPI 動態連結庫函數說明	11
6.1 函數說明	11

1. HID 範本簡介

Holtek USB Code Library Generator 提供一個簡單又快速的 USB 解決方案 – HID 範本，幫助客戶避開底層複雜的 USB 通訊協定，也無需安裝驅動程式，只需透過所提供的電腦端介面函數及程式範例，即可完成 USB 裝置與電腦間的資料通訊。

2. HID 範本架構

HID 範本除了 MCU 程式之外，同時提供了與其溝通之軟體，包含 Windows 動態連結庫 (HIDAPI.DLL) 及連結庫使用範例。

產出專案後，裝置端檔案放在專案目錄內的 Device 目錄中，電腦端的檔案則放在 Host 目錄下。

此範本展示簡易的資料環回(loopback)功能，由電腦端軟體主動送出一筆 16 Bytes 資料，USB 裝置端再將收到的資料回傳給電腦，簡單地呈現 USB 主控與 USB 裝置間的資料傳遞。

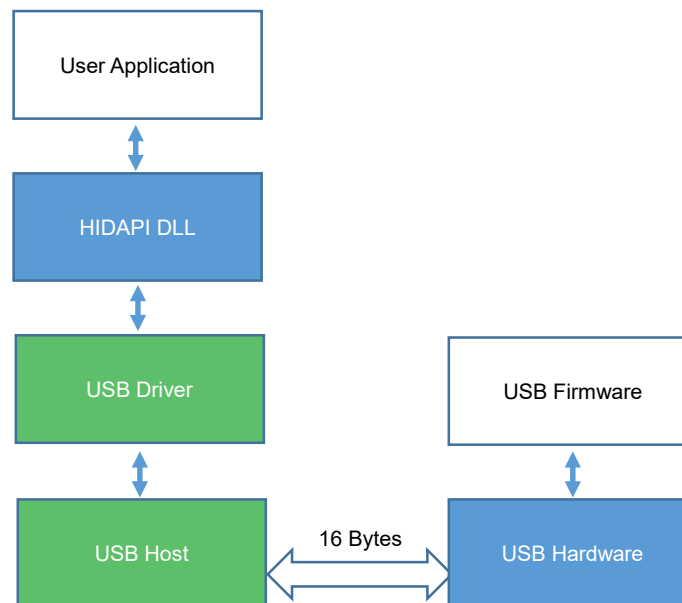


圖 1. USB 堆疊

3. HID 範本檔案說明

[Project Dir]	
-- [Device]	
-- [Source]	
-- [application]	
-- main.c	主程式
-- define.h	使用者定義變數
-- [hw]	
-- mcu.h	MCU 相關設定
-- [os]	
-- [usb]	
-- CLS	USB 類別 (class) 命令相關功能
-- STD	USB 列舉及標準命令相關功能
-- USB_DESC.h	USB 列舉資料
-- USB_INT	USB 中斷處理
-- USB_LIB	USB FIFO 存取相關功能
-- USB_FIFO.h	USB 相關定義
-- xxxx.pjtx	HT-IDE3000 專案檔
-- [Host]	
-- [HIDAPI]	HIDAPI.dll 及標頭檔
-- [HID_INOUT]	
-- HID_INOUT.sln	展示資料環回功能之軟體專案檔
-- [HID_INOUT]	展示資料環回功能之視窗軟體原始碼
-- [ConsoleApplicationCS]	以 C# 開發之 console 程式
-- [ConsoleApplicationVB]	以 VB 開發之 console 程式

圖 2. 目錄結構

4. MCU 程式說明

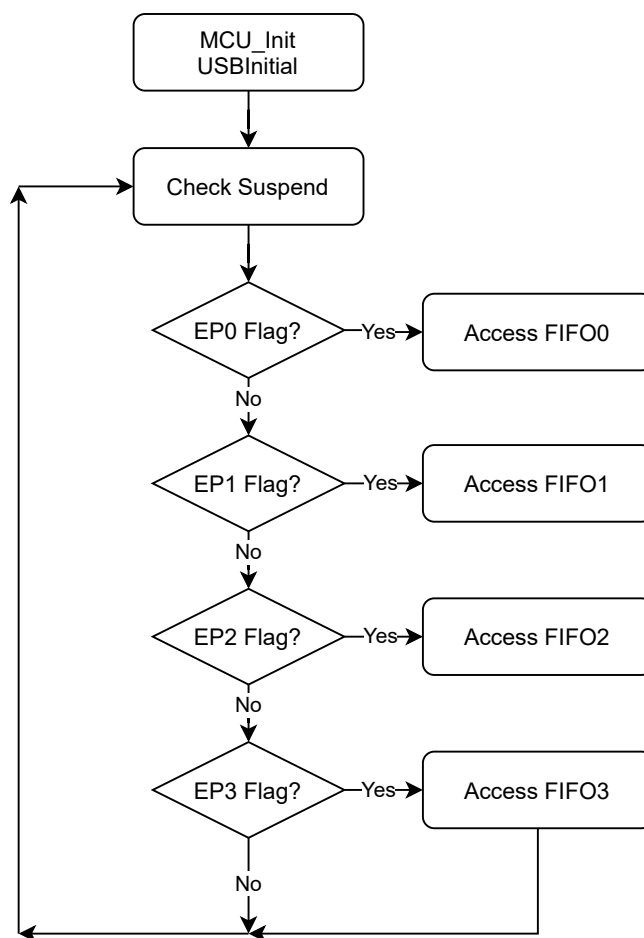


圖 3. 程式主要流程

4.1 程式主要流程

MCU 啟動後呼叫 MCU_Init 配置系統頻率及輸出輸入口後，接著呼叫 USBInitial 執行 USB 相關設定。

隨後不斷地檢查 USB 端點是否有資料需要存取，分別執行：

- | | |
|-------------|--|
| AccessFIFO0 | USB 裝置透過端點 0 完成列舉。本文不對 USB 規格進行說明，相關文件請參考 usb.org 網站。 |
| AccessFIFO1 | 透過端點 1 將資料上傳至主機。 |
| AccessFIFO2 | 透過端點 2 接收來自主機的資料。 |

4.2 資料傳遞

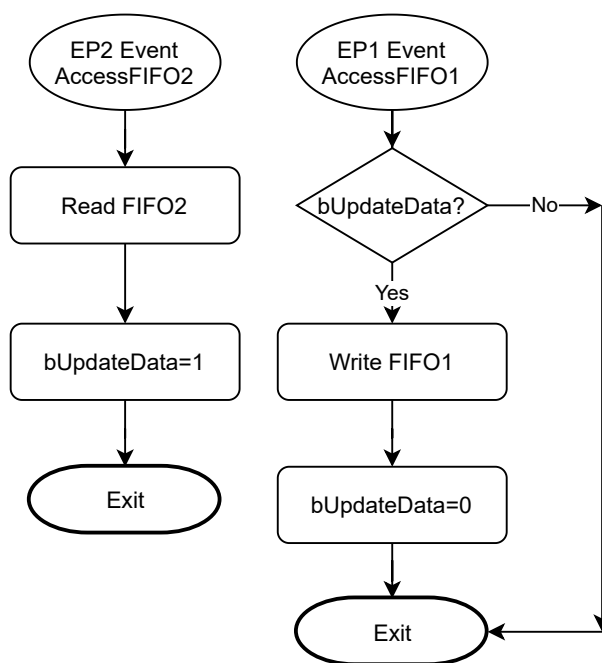


圖 4. 端點存取流程

4.2.1 USB 裝置被動等待主機的存取事件發生

偵測到端點 2 的存取事件時，將資料讀出，並將 bUpdateData 標誌設為 1。

偵測到端點 1 的存取事件時，若 bUpdateData 為 1，表示已經收到主機送來之資料，將資料回送後再將 bUpdateData 標誌設為 0 即可。

使用者可修改 AccessFIFO1 與 AccessFIFO2 (usb_int.c)，視功能需要加入產品的通訊命令協議。

4.2.2 SetFeature/GetFeature

USB HID 另外還有一個透過端點 0 傳送資料的路徑。讓使用者在自訂協議時可以更有彈性。例如：

USB 裝置可以用 SetFeature 接收來自主機的命令，AccessFIFO2 則用來接收資料。反之，用 GetFeatureReport 來傳送狀態，用 AccessFIFO1 傳送資料。

註：USB 命令之方向性都是從主機的角度來描述。Get 的方向為裝置至主機。Set 的方向為主機至裝置。

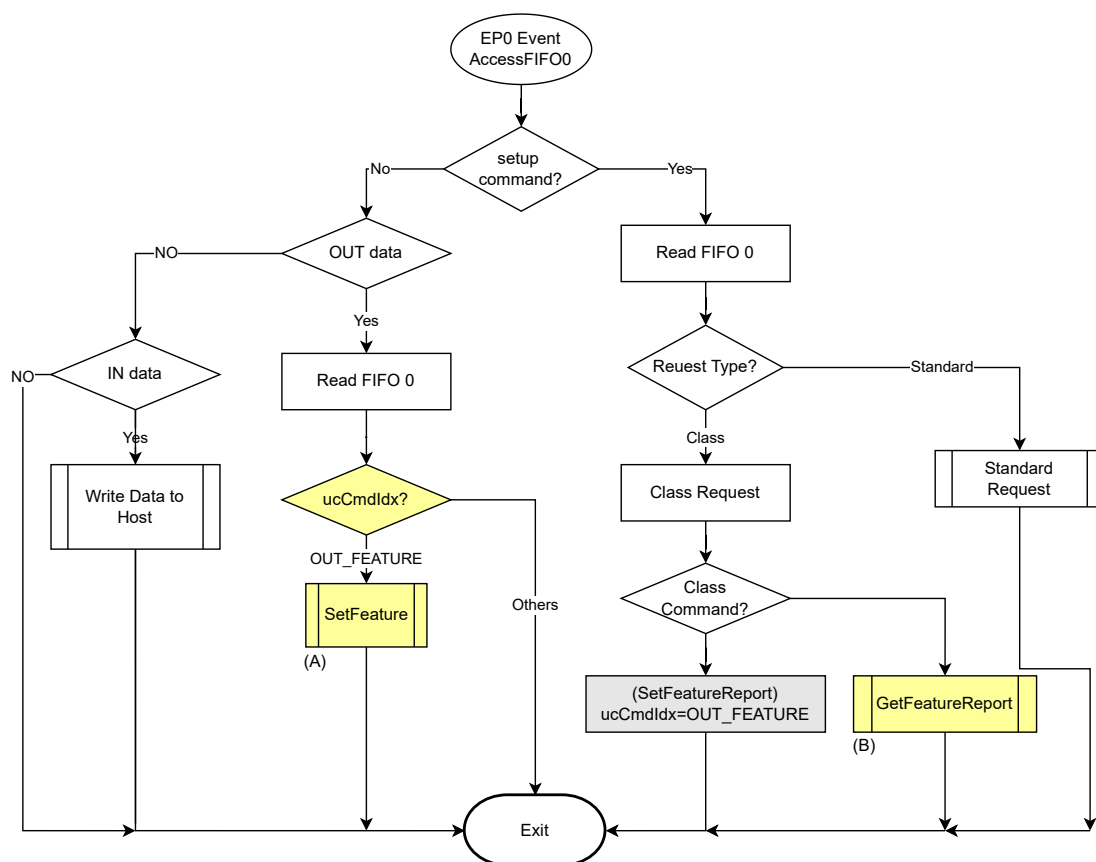


圖 5. 端點 0 存取流程

(A) SetFeature (usb_int.c)

當程式進入此函數時，主機送來之資料已經被讀取並置於 FIFO_OUT 中。

```

void SetFeature()
{
    /**modify code here
    /**data was ready in FIFO_OUT buffer
    |

    ucCmdIdx=0;
    WriteFIFO(0,0);           //handshake
}
  
```

使用者只要將 FIFO_OUT 中之資料取出並使用即可。

本範例之 Feature 長度為 2，若要改變 Feature 長度，請修改 define.h 中之定義。請參閱下一小節。

(B) GetFeature (cls.c)

當程式進入 cls.c 之 GetReport 函數時，代表主機已經下達 GetFeatureReport 命令。使用者只要將欲回傳主機之資料置於 FIFO_OUT 中即可。長度同樣為 2。

```
void GetReport()
{
    if((FIFO_OUT[FIFO_Type] & 0x01))
    {
        if(FIFO_OUT[FIFO_wValH]==1)
        {
            //GetFeatureReport();
            if(FIFO_OUT[FIFO_wLenH]!=0 ||
               FIFO_OUT[FIFO_wIdxL]!=0 ||
               FIFO_OUT[FIFO_wLenL]!=FEATURE_SIZE)
                _stli |= 0x01;
            else
            {
                /**add code here**
                /**put data in FIFO_OUT buffer**
                WriteFIFO(0,FEATURE_SIZE);
            }
        }
    }
    else
    {
        _stli |= 0x01;
    }
}
```

4.3 define.h

```
#define FEATURE_SIZE          2
#define DESC_IDVENDOR         (0x04D9)
#define DESC_IDPRODUCT        (0x806C)
#define DESC_BCDDEVICE         (0x0100)
```

define.h 檔案中提供下列參數讓使用者依需求做修改。

a. Feature 長度 (FEATURE_SIZE)

透過端點 0 傳送資料之長度。可設定之最大長度為 8。

透過端點 1 與端點 2 傳送資料之長度，目前固定為 16，不開放修改。

USB HID 類別不支援不定長度之資料傳送，也就是說每次傳送下來的長度固定為 FEATURE_SIZE (端點 0) 與 EP_LEN (端點 1 與 2)。

b. USB 供應商識別碼 (DESC_IDVENDOR)

0x04D9 為 Holtek 所屬之 USB ID。使用者可沿用 0x04D9，亦可修改為公司已申請之供應商 ID。

c. USB 產品識別碼 (DESC_IDPRODUCT)

不同的產品可以配置一個不同產品識別碼，軟體才能以此來辨識所搭配之裝置。

d. 程式版本號 (DESC_BCDDEVICE)

4.4 mcu.h

本範例之系統頻率預設採用的是內部振盪器。不同的 IC 使用外部振盪器有不同的設定方法，使用者可以修改 mcu.h 裡的參數來改變設定。

```
#define _HXT_      0
#define _HXTEN_   0
#define _PLL_     0

#define _LXT_      0
#define _LXTEN_   0
```

HXT 為 0，表示此型號 IC 不支持用程式將高頻設定為使用外部振盪器，或者不支持外部振盪器。

同樣的，_LXT_ 為 0，表示此型號 IC 不支持用程式將低頻設定為使用外部振盪器，或者不支持外部振盪器。

詳細使用方式請參考 IC 之 datasheet。

```
#define _HXT_      1
#define _HXTEN_   0
#define _hxp0_    _pds00
#define _hxp1_    _pds01
#define _hxp2_    _pds02
#define _hxp3_    _pds03

#define _PLL_     0

#define _LXT_      1
#define _LXTEN_   0
#define _lxp0_    _pes06
#define _lxp1_    _pes07
#define _lxp2_    _pes10
#define _lxp3_    _pes11
```

HXT 為 1 時，若高頻要使用外部振盪器，請將 _HXTEN_ 設為 1 即可。

同理，_LXTEN_ 設為 1，則低頻將來自外部振盪器。

mcu.h 裡的其他定義值，都與 IC 本身的特性有關，請不要任意改變。

5. 軟體使用說明

5.1 HID_INOUT

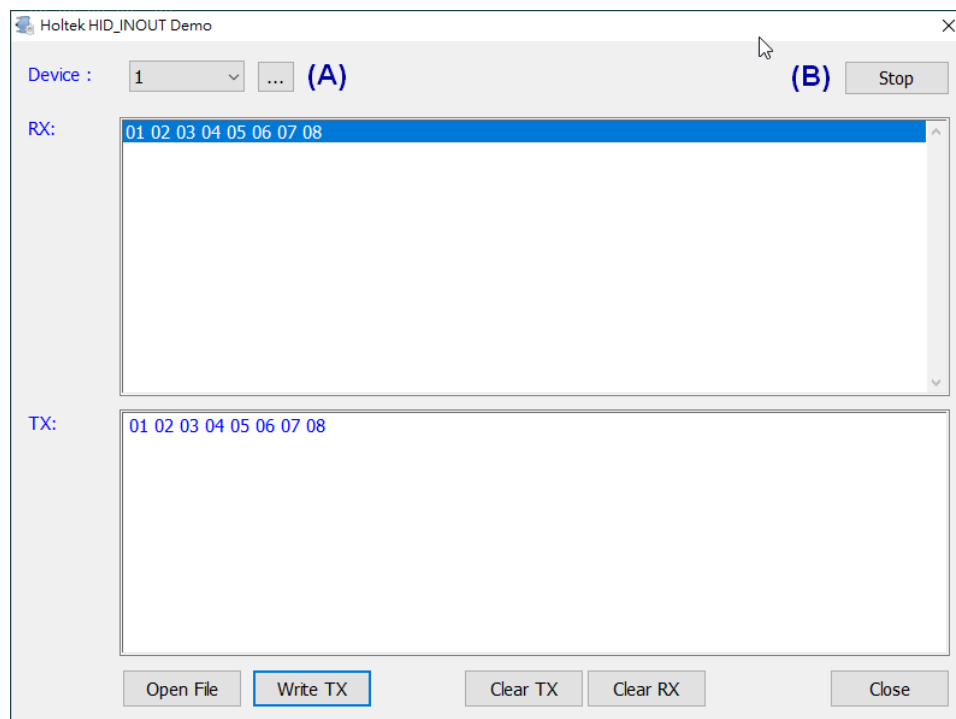


圖 6. Holtek HID Demo 資料環回測試

USB Code Library Generator 為 HID 範例提供了一個簡易視窗軟體，由 Visual Studio C++ 開發而成，用來展示資料環回的效果。

使用步驟如下：

(A) 按下 [...] 時，會呼叫 OpenFirstHIDDevice/OpenNextHIDDevice 來尋找主機中指定 VID/PID 所有的 USB HID 裝置。

(B) 同樣呼叫 OpenFirstHIDDevice/OpenNextHIDDevice 來正式開啟指定的 USB HID 裝置。並啟動背景執行緒，隨時監測回送的資料。

(TX) 於畫面下方的 TX 視窗輸入十六進制數字，並以空白隔開，再按下 [Write TX] 按鍵，此時軟體會呼叫 WriteFile (請參考 [\[HIDAPI 動態連結庫函數說明\]](#)) 將資料送到 USB HID 裝置 (AccesssFIFO2)。

(RX) USB HID 裝置透過 AccessFIFO1 回傳資料給軟體後，軟體透過 ReadFile (請參考 [\[HIDAPI 動態連結庫函數說明\]](#))，將讀回的資料顯示在 RX 視窗中。

需特別注意的是，USB HID 類別不支援不定長度之資料傳送，也就是說每次傳送下來的長度固定為 EP_LEN (端點 1 與 2)。

故 HID_INOUT 軟體將第一個位元組視為資料長度，來達成不定長度傳送的能力，一次能夠傳送的最大長度為 EP_LEN-1。

D0	D1	D2~D16
Report ID(0)	Len	Data

5.2 ConsoleApplicationCS

5.3 ConsoleApplicationVB

除了 C++ 之外，HIDAPI.dll 亦能給其他程式語言呼叫。USB Code Library Generator 提供了 C# 與 VB 二種程式語言之 console 程式，程式中僅簡單地開啟 USB 裝置 / 寫入 / 讀出 / 關閉 USB 裝置，主要用意為示範載入 DLL 函數並呼叫。

6. HIDAPI 動態連結庫函數說明

6.1 函數說明

函數	HANDLE OpenFirstHIDDevice(DWORD wVID, DWORD wPID, DWORD wUsagePage, DWORD wUsage, BOOL bSync); HANDLE OpenNextHIDDevice(DWORD wVID, DWORD wPID, DWORD wUsagePage, DWORD wUsage, BOOL bSync);	
參數	wVID	欲開啟裝置之 USB 供應商 ID (Vendor ID)。
	wPID	欲開啟裝置之 USB 產品 ID (Product ID)。
	wUsagePage wUsage	Report 描述元所採用的 UsagePage/Usage (請參考 HID 規格)。不指定請填 0。 同一個 VID/PID 擁有多個 USB 裝置界面時，需指定 UsagePage/Usage，否則只會傳回第一個裝置界面之代表值 (Handle)。
	bSync	採用同步傳輸或非同步傳輸。同步傳輸意指函數會等待 I/O 完成才回傳。
回傳值	開啟之裝置代表值 (Handle)，若開啟失敗則回傳 Null。	

說明：

當相同的 VID/PID USB 裝置同時存在時，用 OpenFirstHIDDevice 可取得第一個，接著用 OpenNextHIDDevice 繼續開啟，直到回傳值為 Null。

開啟後，可進行 ReadFile/WriteFile 的動作。ReadFile/WriteFile 為 Windows API 標準函數，詳細使用方式請參考 MSDN。

```

BOOL WriteFile(HANDLE          hFile,
               LPCVOID          lpBuffer,
               DWORD            nNumberOfBytesToWrite,
               LPDWORD          lpNumberOfBytesWritten,
               LPOVERLAPPED     lpOverlapped);

BOOL ReadFile(HANDLE          hFile,
              LPVOID          lpBuffer,
              DWORD            nNumberOfBytesToRead,
              LPDWORD          lpNumberOfBytesRead,
              LPOVERLAPPED     lpOverlapped);

```

讀出 (nNumberOfBytesToRead) / 寫入 (nNumberOfBytesToWrite) 之長度必須為韌體定義之長度加 1，在本範例中為 EP_LEN+1。

傳入之 lpBuffer 的第一個 Byte 必須設定為 Report ID，若無 Report ID 則填入 0 (Report ID 之應用請參考 HID 規格)。在本範例中為 0。

主機呼叫 WriteFile 後，裝置將於 AccessFIFO2 中取得資料。

主機呼叫 ReadFile 後，會取得裝置從 AccessFIFO1 中送出之資料。

函數	void CloseHIDDevice(HANDLE hDevice)	
參數	hDevice	欲關閉之 USB 裝置代表值。
回傳值	無	

說明：

關閉指定之 USB 裝置。

函數	BOOL SetFeature(HANDLE hDevice, LPVOID pData, DWORD nLen)	
參數	hDevice	已開啟之 USB 裝置代表值。
	pData	欲寫入 Feature 資料之 Buffer。第一個 Byte 必須填入 Report ID，若無 Report ID 的話，則必須填入 0。
	nLen	欲寫入 Feature 資料之長度，此長度必須為韌體定義之 Feature Report 長度加 1，在本範例中為 FEATURE_SIZE+1。
回傳值	1：寫入成功 0：寫入失敗	

說明：

主機呼叫 SetFeature 後，裝置可於 SetFeature 中取得資料，請參考 [\[MCU 程式說明 4.2.2 章節 \(A\)\]](#)。

函式	BOOL GetFeature(HANDLE hDevice, LPVOID pData, DWORD nLen)	
參數	hDevice	已開啟之 USB 裝置代表值。
	pData	讀出 Feature 資料存放之 Buffer。呼叫此函數前，第一個 Byte 必須填入 Report ID，若無 Report ID 則必須填入 0。
	nLen	欲讀出 Feature 資料之長度，此長度必須為韌體定義之 Feature Report 長度加 1，在本範例中為 FEATURE_SIZE+1。
回傳值	1：讀出成功 0：讀出失敗	

說明：

USB 裝置於 GetReport 中寫入之資料，主機可呼叫 GetFeature 取得。請參考 [\[MCU 程式說明 4.2.2 章節 \(B\)\]](#)。

Copyright© 2024 by HOLTEK SEMICONDUCTOR INC. All Rights Reserved.

本文件出版時 HOLTEK 已針對所載資訊為合理注意，但不保證資訊準確無誤。文中提到的資訊僅是提供作為參考，且可能被更新取代。HOLTEK 不擔保任何明示、默示或法定的，包括但不限於適合商品化、令人滿意的品質、規格、特性、功能與特定用途、不侵害第三人權利等保證責任。HOLTEK 就文中提到的資訊及該資訊之應用，不承擔任何法律責任。此外，HOLTEK 並不推薦將 HOLTEK 的產品使用在會因故障或其他原因而可能會對人身安全造成危害的地方。HOLTEK 特此聲明，不授權將產品使用於救生、維生或安全關鍵零組件。在救生 / 維生或安全應用中使用 HOLTEK 產品的風險完全由買方承擔，如因該等使用導致 HOLTEK 遭受損害、索賠、訴訟或產生費用，買方同意出面進行辯護、賠償並使 HOLTEK 免受損害。HOLTEK (及其授權方，如適用) 擁有本文件所提供資訊 (包括但不限於內容、資料、示例、材料、圖形、商標) 的智慧財產權，且該資訊受著作權法和其他智慧財產權法的保護。HOLTEK 在此並未明示或暗示授予任何智慧財產權。HOLTEK 擁有不事先通知而修改本文件所載資訊的權利。如欲取得最新的資訊，請與我們聯繫。