

2015

GoDEX Mac OS X framework User Manual

透過程式控制 GoDEX 印表機

此手冊中將介紹如何透過 GoDEX Mac OS framework 來撰寫 GoDEX 印表機控制程式。透過簡單的範例，您將學習到如何設定印表機參數，列印文字，影像與條碼



GoDEX Mac OS X framework User Manual

目錄

1. 概觀說明.....	2
2. 函式清單.....	3
3. 函式參數說明.....	4
3.1 - (BOOL)OPENNET:(NSString *)ADDRESS PORT:(NSInteger)PORT.....	4
3.2 - (BOOL)OPENNET:(NSString *)ADDRESS PORT:(NSInteger)PORT TIMEOUT:(NSInteger)TIMEOUT.....	4
3.3 - (BOOL)OPENUSB:(NSString *)SERIALNUMBER.....	5
3.4 - (NSString *)FINDALLNET.....	5
3.5 - (NSString *)FINDALLUSB.....	5
3.6 - (VOID)CLOSE	5
3.7 - (BOOL)SENDCOMMAND:(NSString *)COMMAND	5
3.8 - (BOOL)SENDCOMMAND:(NSString *)COMMAND ENCODING:(NSInteger)ENCODE	6
3.9 - (NSString *)READ.....	6
3.10 - (NSString *)READ:(NSInteger)TIMEOUT.....	6
3.11 - (BOOL)SETUP:(NSInteger)HEIGHT GAP:(NSInteger)GAP TOP:(NSInteger)TOP DARK:(NSInteger)DARK SPEED:(NSInteger)SPEED MODE:(NSInteger)MODE	7
3.12 - (BOOL)PRINTCmdFILE:(NSString *)FILENAME	7
3.13 - (BOOL)LOADIMAGE:(NSString *)FILENAME IMAGEID:(NSString *)IMGAID	8
3.14 - (BOOL)PUTIMAGE:(NSString *)FILENAME POINTX:(NSInteger)PX POINTY:(NSInteger)PY	8
4. GODEX FRAMEWORK 使用範例.....	9
4.1 建立 OS X 的 APPLICATION	9
4.2 加入 GODEXT TOOL.FRAMEWORK.....	10
4.3 實作 GODEXT TOOL 應用程式範例.....	12

1. 概觀說明

godextool.framework 是由 GoDEX 提供的 Mac OS 函式庫。程式開發者可使用此函式庫在 Mac OS 上開發印表機的控制程式。

godextool.framework 提供了 USB 與網路通訊的控制函式。開發者在開始對印表機下指令前，必須建立 GodexPrinterControl 物件，然後呼叫 *openUSB* 或 *openNet* 函式來與印表機建立連線；在設定或列印完畢後，必須呼叫 *close* 函式來結束程式與印表機間的連線。

開發者可以透過 *sendCommand* 函式，傳送指令或資料給印表機；可透過 *read* 函式取得印表機送出的字串資料。開發者必須使用 EZPL 印表機語言來控制 GoDEX 印表機，如果需要更進階的控制方式，可以至 GoDEX 官網下載 [EZPL Programming Manual](#)。

範例：

1. 若要使用網路，在 EZPL 印出自我測試頁的指令為

```
openNet:@ "192.168.1.1" port:9100;  
sendCommand:@ "~V";  
close;
```

2. 若要使用 USB，在 EZPL 印出自我測試頁的指令為

```
openUSB:20140108;  
sendCommand:@ "~V";  
close;
```

3. 開發者可以使用 *putImage* 函式來列印圖片，須注意，在 EZPL 印表機語言在【列印】時，列印的內容必須在 ^L 與 E 中。

```
open:@ "192.168.1.1" port:9100;  
sendCommand:@ "^L";  
  
putImage:fileName pointX:0 pointY:0;  
  
sendCommand:@ "E";  
close;
```

2. 函式清單

Command	Description
<code>openNet</code>	開啟網路通訊介面與印表機建立連線，預設連線秒數為 2 秒。
<code>openNet</code>	開啟網路通訊介面與印表機建立連線，使用者設定連線秒數。
<code>openUSB</code>	開啟 USB 通訊介面與印表機建立連線。
<code>findAllNet</code>	搜尋相同網段上，所有 GoDEX 印表機。
<code>findAllUSB</code>	搜尋在 Mac OS X USB 接孔上，所有 GoDEX 印表機。
<code>close</code>	結束與印表機間的連線。
<code>sendCommand</code>	傳送字串資料給印表機(預設 UTF8 編碼)。
<code>sendCommand</code>	傳送字串資料給印表機(不同編碼方式)。
<code>read</code>	讀取印表機送出的字串，預設讀取字串的秒數為 3 秒。
<code>read</code>	讀取印表機送出的字串，使用者設定讀取字串的秒數。
<code>setup</code>	設定標籤紙種類、紙張高度、列印速度等參數。
<code>printCmdFile</code>	將命令檔傳送至印表機。
<code>loadImage</code>	上傳二進位圖檔至印表機的內部記憶體。
<code>putImage</code>	直接列印影像(不需上傳影像至印表機)。

3. 函式參數說明

使用函式前須先宣告 `godex` 物件和初始化設定，方可使用下列成員函式。

Objective-C

```
@property (strong, nonatomic) GoDEXPrinterControl *godex;
_godex = [[GoDEXPrinterControl alloc] init];
```

Swift

```
var godex = GoDEXPrinterControl ()
```

3.1 - (BOOL)openNet:(NSString *)address port:(NSInteger)port

函式描述 透過網路，建立與印表機的連線。

參數定義	address	類型：NSString 印表機無線網路之 IP Address。
	port	類型：NSInteger 印表機的端口。
回傳值	true	與印表機連線成功。
	false	與印表機連線失敗或超過與印表機的連線時間(預設 2 秒)。

3.2 - (BOOL)openNet:(NSString *)address port:(NSInteger)port timeout:(NSInteger)timeout

函式描述 透過指定通訊方式，建立與印表機的連線，使用者設定連線時間。

參數定義	address	類型：NSString 印表機無線網路之 IP Address。
	port	類型：NSInteger 印表機的端口。
	timeout	類型：NSInteger 嘗試與印表機連線的秒數時間。
回傳值	true	與印表機連線成功。
	false	與印表機連線失敗或超過與印表機的連線時間。

GoDEX Mac OS X framework User Manual

3.3 - (BOOL)openUSB:(NSString *)serialNumber

函式描述	透過指定通訊方式，建立與印表機的連線。	
	serialNumber	
參數定義	類型：NSString 印表機的序號。	
回傳值	true	與印表機連線成功。
	false	與印表機連線失敗。

3.4 - (NSString *)findAllNet

函式描述	搜尋相同網段上，所有 GoDEX 印表機。	
參數定義	None。	
回傳值	在相同網段上，所有 GoDEX 印表機(NSArray *)物件。每個搜尋到的印表機(NSArray *)物件包含印表機的名稱、IP 位址、端口。	

3.5 - (NSString *)findAllUSB

函式描述	搜尋在 Mac OS X USB 接孔上，所有 GoDEX 印表機。	
參數定義	None。	
回傳值	在 USB 接孔上，所有 GoDEX 印表機(NSArray *)物件。每個搜尋到的印表機(NSArray *)物件包含印表機的名稱、序號。	

3.6 - (void)close

函式描述	結束與印表機的連線。	
參數定義	None。	
回傳值	None。	

3.7 - (BOOL)sendCommand:(NSString *)command

函式描述	傳送字串資料給印表機。	
	command	
參數定義	類型：NSString 傳送給印表機的字串(預設 NSStringEncoding 編碼)資料。	
回傳值	true	字串資料傳送成功。
	false	字串資料傳送失敗。

GoDEX Mac OS X framework User Manual

3.8 - (BOOL)sendCommand:(NSString *)command encoding:(NSInteger)encode

函式描述	傳送字串資料給印表機，使用者設定字串編碼方式。	
參數定義	command	類型：NSString 傳送給印表機的字串資料。
	encode	類型：NSInteger 選擇 command 之編碼方式。 (例：NSUTF8StringEncoding 或 NSASCIIStringEncoding，請參照 Apple 所提供的說明文件中的 StringEncoding)。
	回傳值	true 字串資料傳送成功。 false 字串資料傳送失敗。

3.9 - (NSString *)read

函式描述	讀取印表機回傳的字串，預設讀取時間為 3 秒。	
參數定義	None。	
回傳值	印表機回傳的字串。	

3.10 - (NSString *)read:(NSInteger)timeout

函式描述	讀取印表機回傳的字串，使用者設定讀取時間。	
參數定義	timeout	類型：NSInteger 讀取印表機回傳的字串的秒數時間。
	回傳值	印表機回傳的字串。

GoDEX Mac OS X framework User Manual

3.11 - (BOOL)setup:(NSInteger)height gap:(NSInteger)gap top:(NSInteger)top dark:(NSInteger)dark speed:(NSInteger)speed mode:(NSInteger)mode

函式描述 設定印表機基本參數。

參數定義	height	類型：NSInteger 紙張高度，單位為 mm。
	gap	類型：NSInteger 標籤間距，單位為 mm。
	top	類型：NSInteger 黑線標記距離，單位為 mm。
	dark	類型：NSInteger 列印黑度，可設定範圍為 0~19。
	speed	類型：NSInteger 列印速度，單位為 IPS，可設定範圍為 2~7。 根據不同印表機有不同的範圍限制。
	mode	類型：NSInteger 紙張類型 0: 標籤紙 (label with gap)。 1: 連續紙 (plain paper)。 2: 黑線紙 (black mark label)。
	回傳值	true 印表機基本參數設定成功。 false 印表機基本參數設定失敗。

3.12 - (BOOL)printCmdFile:(NSString *)fileName

函式描述 將命令檔傳送至印表機。

參數定義	fileName	類型：NSString 命令檔路徑與檔名。
	回傳值	true 命令檔傳送成功。 false 命令檔傳送失敗。

GoDEX Mac OS X framework User Manual

3.13 - (BOOL)loadImage:(NSString *)fileName imageID:(NSString *)imageID

函式描述	上傳二進位圖檔至印表機的內部記憶體。	
參數定義	fileName	類型：NSString 圖檔路徑與檔名。
	imageID	類型：NSString imageID 代表上傳圖檔到印表機後，所重新命名的名稱，列印圖檔使用^Y 命令。
	true	圖檔傳送成功。
	false	圖檔傳送失敗。

3.14 - (BOOL)putImage:(NSString *)fileName pointX:(NSInteger)px pointY:(NSInteger)py

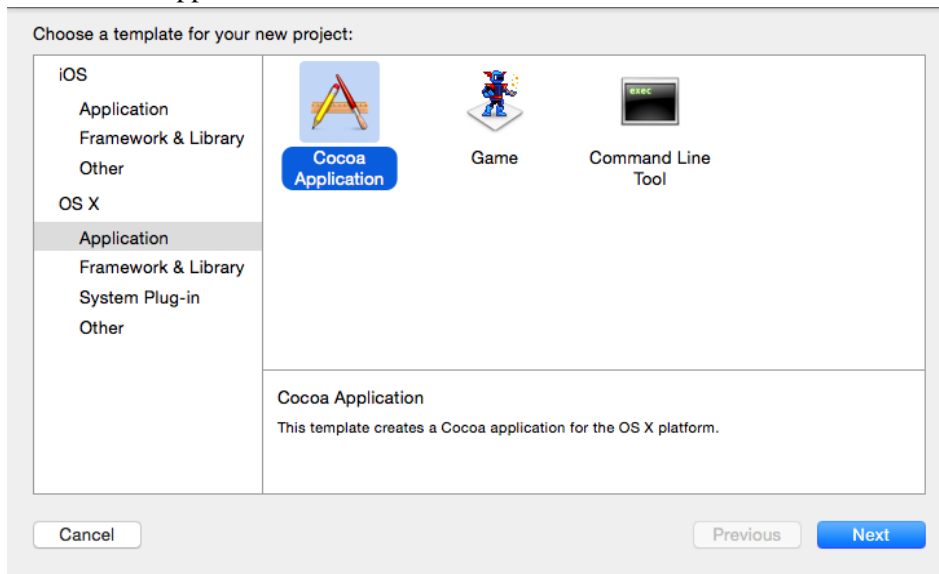
函式描述	直接列印影像(不需上傳影像至印表機)。	
參數定義	fileName	類型：NSString 欲列印之影像的檔案路徑。
	px	類型：NSInteger 圖形的 X 座標位置(左上方水平位置)，單位為 dot。
	py	類型：NSInteger 圖形的 Y 座標位置(左上方垂直位置)，單位為 dot。
	true	影像傳送成功。
回傳值	false	影像傳送失敗。

4. GoDEX framework 使用範例

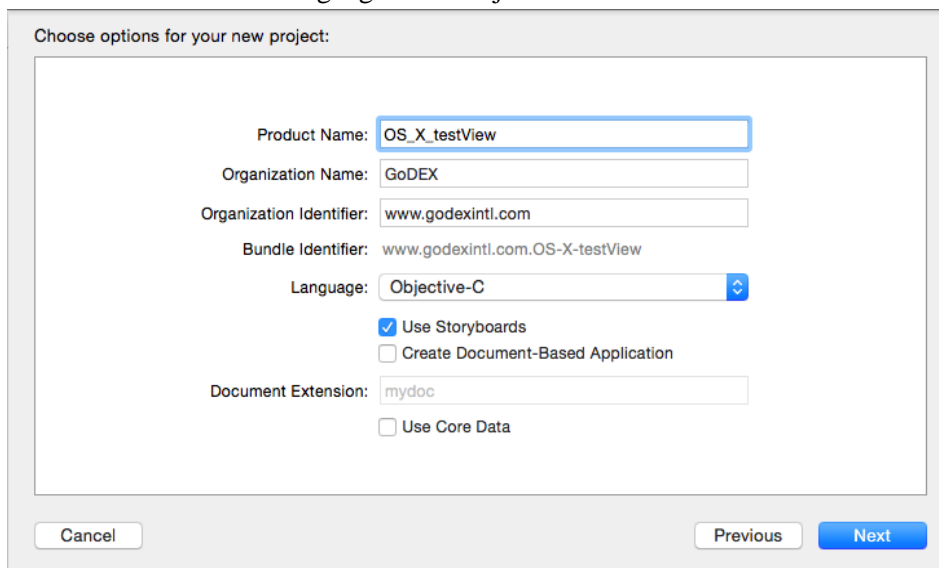
透過 4.1 和 4.2 步驟，將可以在 Objective-C 或 Swift 中建立出一個簡單控制 GoDEX 印表機的程式。
4.3 包含一個範例程式，供你參考。

4.1 建立 OS X 的 Application

選擇 Cocoa Application

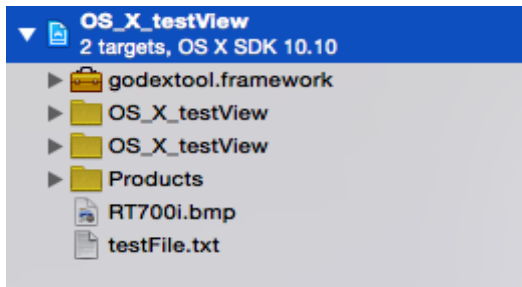


輸入 Product Name，Language 選擇 Objective-C 或是 Swift

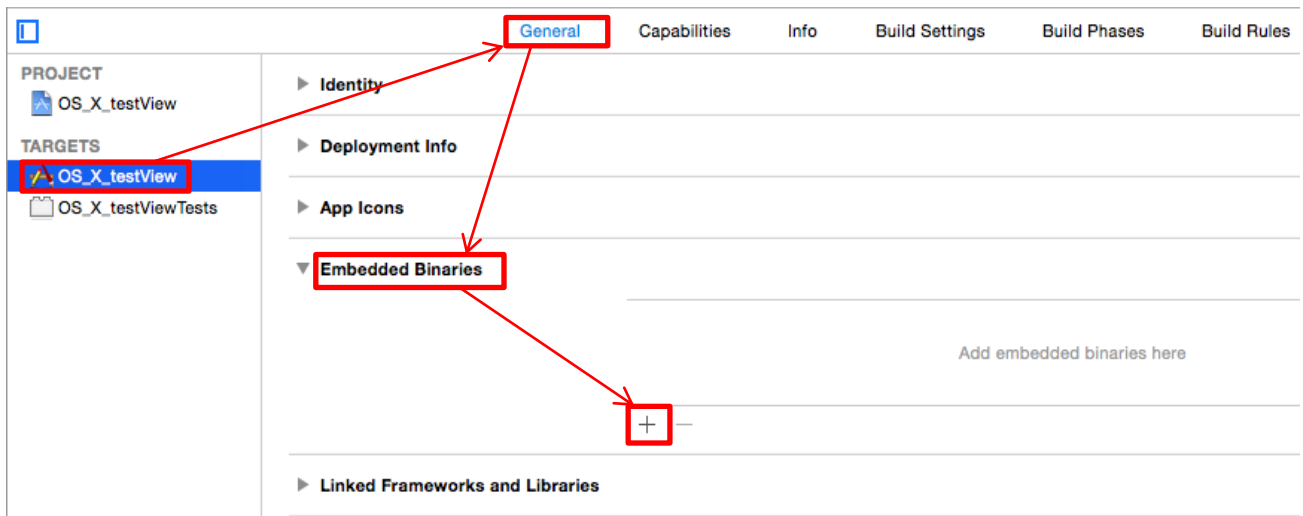


4.2 加入 godextool.framework

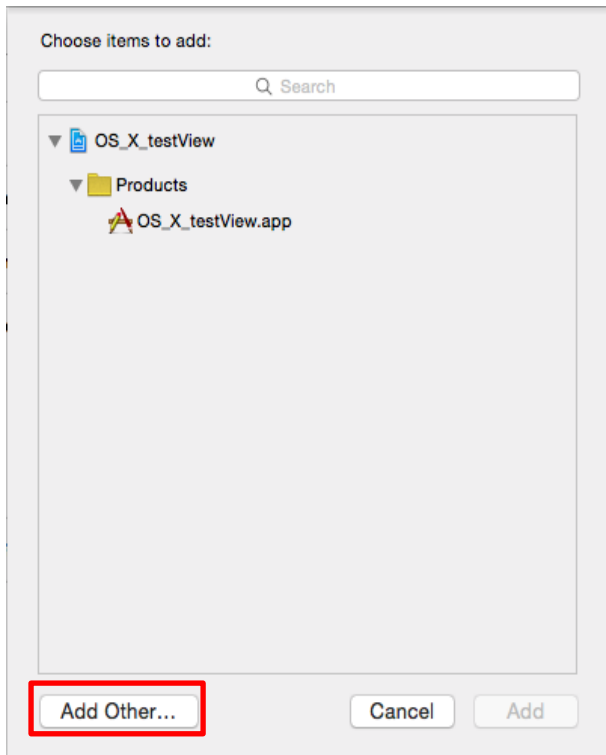
點選你的專案



開啟 TARGETS，選擇 General，在 Embedded Binaries 欄位點選 +

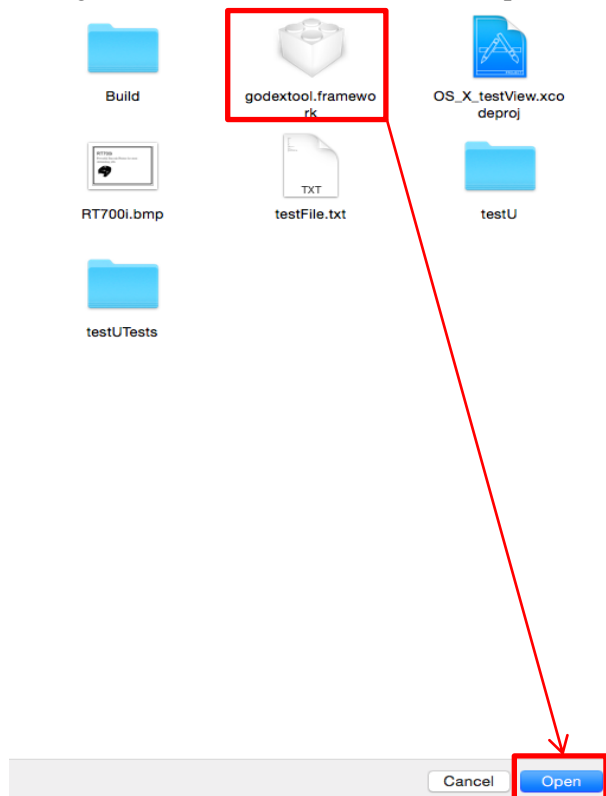


點選 Add Other...

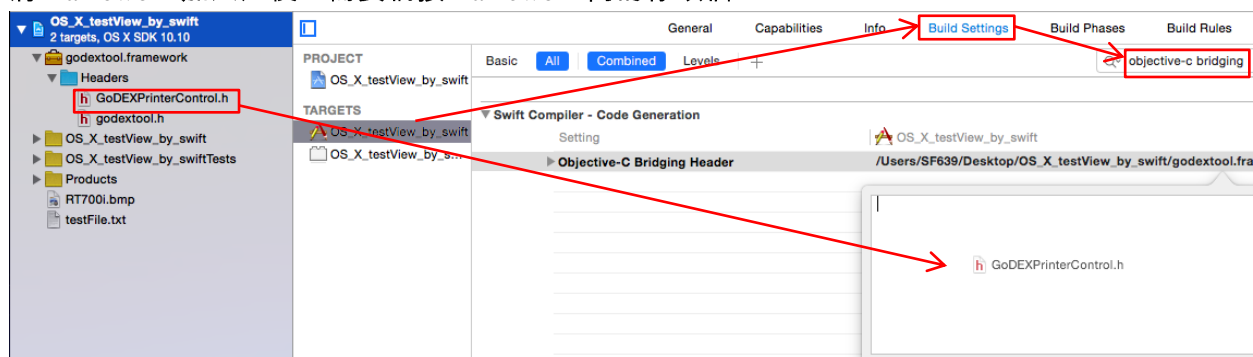


GoDEX Mac OS X framework User Manual

選擇 godextool.framework，之後點選 Open



由於 godextool.framework 是由 Objective-C 所撰寫，如果想要 Swift 中使用 godextool.framework，則將 framework 加入之後，需要橋接 framework 內的標頭檔。

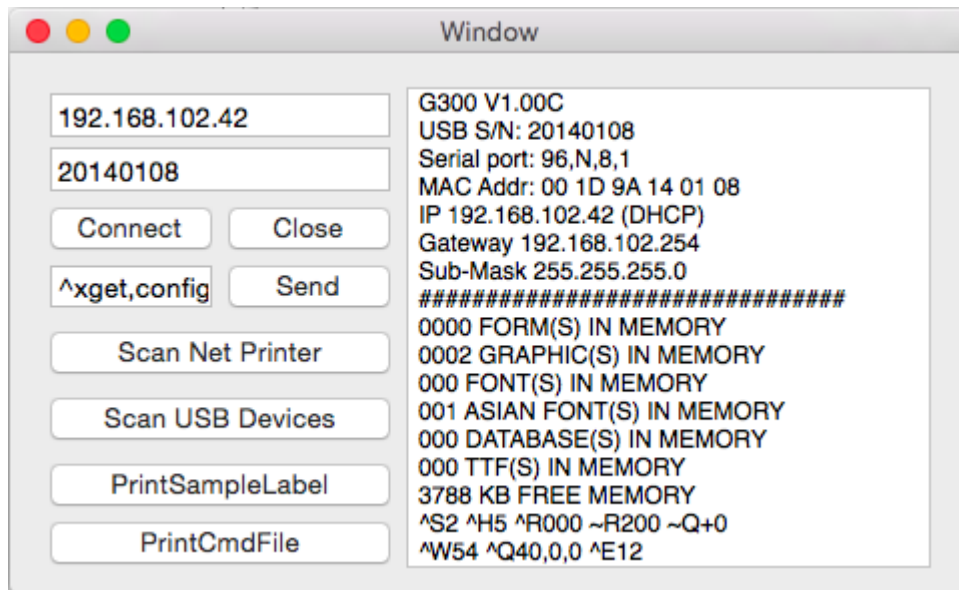


在 Swift 的專案中開起 TARGETS，選擇 Build Settings，在搜尋欄位輸入 Objective-C Bridging，在下面的 Objective-C Bridging Header 將 godextool.framework 內的 GoDEXPrinterControl.h 加入此中。

如果使用 Objective-C 開發，需要在 ViewController.h 將 `<godextool/GoDEXPrinterControl.h>` 加入。

4.3 實作 godextool 應用程式範例

此範例程式包含在 Mac_OS_X_SDK.zip 中，Readme 包含 framework 和此範例程式的相關資訊。之後編譯和執行此程式碼，將會顯示下面的畫面。



在此畫面上包含有三個 `NSTextField` 用於輸入 IP 位址、序號和印表機命令，一個 `NSTextView` 用於顯示印表機回傳的字串，`Connect`、`Close`、`Send`、`Scan Net Printer`、`Scan USB Devices`、`PrintSampleLabel`、`PrintCmdFile`、七個 `NSButton` 用於執行命令。