

ICP DAS SmartQ SDK User Manual

[Version 1.0]



ICP DAS CO., LTD.

泓格科技股份有限公司

免責聲明 **Warning**

泓格科技股份有限公司對於因為應用本產品所造成的損害並不負任何法律上的責任。本公司保留有任何時間未經通知即可變更與修改本檔內容之權利。本文所含資訊如有變更，恕不予另行通知。

本公司盡可能地提供正確與可靠的資訊，但不保證此資訊的使用或其他團體在違反專利或權利下使用。此處包涵的技術或編輯錯誤、遺漏，概不負其法律責任。

版權 **Copyright**

© 2009 泓格科技股份有限公司保留所有權利。

商標識別 **Trademark**

本檔提到的所有公司商標、商標名稱及產品名稱分別屬於該商標或名稱的擁有者所有。

授權宣告 **License**

使用者僅被授權可以在單一電腦上與有限條件下使用、備份軟體與相關資料，不得同時于該單一電腦外使用本軟體。本公司仍保有此軟體與相關資料的著作權及其它智慧財產權。除非事先經過本公司的書面授權，否則禁止重制、傳送及散佈等方式取得部份或全部軟體或相關的複製品。

目錄：

1.	建置環境	4
1.1	C / C++	4
1.2	C# / VB .Net	5
2.	函式介面	7
2.1	自定義結構 PSPARMS:.....	7
2.2	函式：	10
3.	範例程式說明	16
3.1	Publish.c (Publish_WinCE.cpp)	16
3.2	SubScribe.c (SubScribe _WinCE.cpp)	18

1. 建置環境

在建置程式之前，建議使用者先參考WinPAC-8000 User Manual ([WinPAC 8x4x user manual v2.0.1.pdf](#))，初步了解整個開發環境架設與開發步驟。

手冊下載網址：

http://ftp.icpdas.com/pub/cd/winpac/napdos/wp-8x4x_ce50/document/winpac_8x4x_user_manual_v2.0.1.pdf

1.1 C / C++

● Windows

1. 建立一個Visual Studio C++的Win32 Console空專案。
2. 將Publish.c、SmartQAPI.h、MQTT.dll及MQTT.lib等4個檔案複製到該專案檔夾內。
3. 將SmartQAPI.h及Publish.c匯入該專案。
4. 於該項目的連接器中加入ws2_32.lib及MQTT.lib。
5. 將Publish.c的Precompiled Headers設定為Not using precompiled headers。
6. 重複1-5步驟，唯其中2、3、5步驟中Publish.c檔案改為SubScribe.c檔案。
7. 執行Publish及SubScribe程式。

● QP-8x1/QP-8x2

1. 選定已安裝eMbedded Visual C++的電腦為程式開發平臺，您也可以至Microsoft網站免費下載eMbedded Visual C++。
2. 安裝WinPAC SDK：pac270_sdk_20091006.msi (已放置於 SmartQ CD-ROM：\SmartQ SDK\ PAC270 SDK 目錄中)或以後的版本於此平臺。(請參考WinPAC-8000 User Manual 第4.2節 “Installing WinPAC SDKs”)
3. 建立一個eMbedded Visual C++的WCE Application專案。(請參考WinPAC-8000 User Manual 第4.6節 “Your First Program in eMbedded Visual C++”)
4. 將Publish_WinCE.cpp、SmartQAPI.h、WMQTT_CE_MT.dll、WMQTT_CE_MT.lib等4個檔案複製到該專案檔夾內。
5. 將SmartQAPI.h及Publish_WinCE.cpp匯入該專案。
6. 於該項目的連接器中加入WMQTT_CE_MT.lib。
7. 將Publish_WinCE.cpp的Precompiled Headers設定為Not using precompiled headers。

8. 於該專案的WCE Configuration工具列(tool bar)中將STANDARDSDK改為PAC270。
9. 並於主選單Tools\Configure Platform Manager中選取PAC270 Device，進入Device Properties頁面將Transport設定為TCP/IP Transport for Windows CE，以及Startup Server設定為Manual Server。
10. 執行程式，出現CEMGRC.EXE的cmd指令。
11. 將CEMGRC.EXE的cmd指令輸入QP-8x1/QP-8x2 WinCE中。
12. 執行QP-8x1/QP-8x2 WinCE內下載回來的執行檔程式(默認儲存於Micro_SD檔夾內，另外WMQTT_CE_MT.dll必須放入執行檔的目錄當中)。
13. 建立SubScribe專案：重複3-12步驟，唯其中4、5、7步驟中Publish_WinCE.cpp檔案改為SubScribe_WinCE.cpp檔案。
14. 執行Publish及SubScribe程式。

1.2 C# / VB .Net

● Windows

1. 建立一個Visual Studio C#(或VB)的Console應用程式(Console Application)空專案。
2. 將Publish.cs(或Publish.vb)、MQTT.dll與MQTTDotNet.dll等3個檔案到該專案檔案夾內。
3. 在方案總管” Solution Explorer” 視窗中，右鍵點選參考Reference，或點選專案屬性(properties)選擇參考頁籤，新增MQTTDotNet.dll的參考路徑。
4. 將Publish.cs(或Publish.vb)匯入該專案，確認檔案開頭有加入using ICPDAS.SmartQ(或Import ICPDAS.SmartQ)。
5. 重複1-4步驟，唯其中2、4步驟中Publish.cs(或Publish.vb)檔案改為SubScribe.cs(或SubScribe.vb)檔案。
6. 另外MQTT.dll與MOTTDotNet.dll必須放入執行檔的目錄當中。
7. 執行Publish及SubScribe程式。

● QP-8x1/QP-8x2

1. 安裝WinPAC SDK：pac270_sdk_20091006.msi (已放置於 SmartQ CD-ROM：\SmartQ SDK\ PAC270 SDK 目錄中)或以後的版本於此平臺。(請參考WinPAC-8000 User Manual 第4.2節 “Installing WinPAC SDKs”)
2. 建立一個Visual Studio C#(或VB)的Smart Device中Windows CE 5.0

的Console應用程式(Console Application)空專案。(請參考 WinPAC-8000 User Manual 第4.4節 “Your First Program in C#” 或第4.5節 “Your First Program in VB.net”)

3. 將Publish.cs(或Publish.vb)、WMQTTDotNetCF.dll、WMQTT_CE_MT.dll等3個檔案複製到該專案檔案夾內。
4. 在方案總管” Solution Explorer” 視窗中，在專案節點上按右鍵選擇參考Reference，或點選專案屬性(properties)選擇參考頁籤，新增WMQTTDotNetCF.dll的參考路徑。
5. 在項目節點上按右鍵選擇Change Target Platform，在Change To選項中，將平臺改爲PAC270。
6. 在設備(Device)頁籤中，將Target Device改選爲” PAC50-ARMV4:ARMV4I_Releas”
7. 將Publish.cs(或Publish.vb)匯入該專案，確認檔案開頭有using ICPDAS.SmartQ(或Import ICPDAS.SmartQ)。
8. 建立SubScribe專案：重複2-7步驟，唯其中3、7步驟中Publish.cs檔案改爲SubScribe.cs(或SubScribe.vb)檔案。
9. 參考QPAC硬體使用手冊，編譯完後將程式以FTP上傳到QPAC的存放目錄。
10. **WMQTTDotNetCF.dll、WMQTT_CE_MT.dll必須放入執行檔的目錄當中**
11. 執行Publish及SubScribe程式。

2. 函式介面

2.1 自定義結構 PSPARMS:

參數名稱	類型	用途
pBroker	char* / String	用以儲存SmartQ QP-500的網路位址或主機名稱。
Port	Int	SmartQ QP-8x1/QP-8x2 or Windows Client與SmartQ QP-500連接的通訊埠號碼，標準設置為1883。
Topic	char* / String	<p>SmartQ QP-8x1/QP-8x2 or Windows Client發佈至SmartQ QP-500的訊息主題或及SmartQ QP-8x1/QP-8x2 or Windows Client與SmartQ QP-500訂閱的訊息主題。一般而言，訊息主題常見三種特殊字元：</p> <p>分隔符號：(/)</p> <p>單一層級：(+)</p> <p>多層層級：(#)</p> <p>舉例說明：</p> <p>例1：topic = LEVEL1 訊息主題為LEVEL1</p> <p>例2：topic = LEVEL1/LEVEL2 訊息主題為LEVEL1/LEVEL2，分隔符號(/)在此可視為LEVEL2繼承LEVEL1主題，而非LEVEL1或LEVEL2主題。</p> <p>例3：topic = LEVEL1/LEVEL2/+ 對於LEVEL1/LEVEL2/及所有LEVEL1/LEVEL2/LEVEL3共三層的主題感興趣。</p> <p>例4：topic = LEVEL1/LEVEL2/# 對於所有繼承LEVEL1/LEVEL2的主題全部感興趣。</p>
qos	Int	用以設定SmartQ QP-8x1/QP-8x2 or Windows Client與SmartQ QP-500間訊息傳輸的服務品質，Qos值越高則傳輸品質越高，相對地網路負荷則越重。其值可

		<p>為0,1或2，預設值為0。</p> <p>當Qos值為：</p> <p>0：最多僅傳送一次，不論傳送是否成功皆不再傳送。</p> <p>1：至少傳送一次，直到確保SmartQ QP-500成功接收。</p> <p>2：準確地傳送一次，重複發送直到確認接收者接收到為止。</p> <p>建議連接同一個SmartQ QP-500的所有SmartQ QP-8x1/QP-8x2/QP-711/QP-712 or Windows Client，都使用相同的QOS值，以免發生不可預測的錯誤。</p>
retain	Int	Publish所傳輸的訊息是否被QP-500所保留，預設值為0(不保留)。
timeout	Int	程式接收訊息時的等待時間，單位為秒，預設值為10(秒)。
keepAliveTime	Unsigned short / ushort	SmartQ QP-8x1/QP-8x2 or Windows Client與SmartQ QP-500之間連線的存活時間。當SmartQ QP-8x1/QP-8x2 or Windows Client在此設定時間內都沒有發送訊息給SmartQ QP-500時，此連線就會被SmartQ QP-500強制中斷。
hConn	void* / IntPtr	用以紀錄有效連線的記憶體位址。
lastSentMsg	Long / ulong	用以紀錄處理的狀況。
clientId	char* / String	SmartQ QP-8x1/QP-8x2 or Windows Client用以連接至SmartQ QP-500的使用者名稱。不可以兩個以上的client同時有相同名稱。
lwtTopic	char* / String	離線主題，設定SmartQ QP-8x1/QP-8x2 or Windows Client離線後SmartQ QP-500會自動發送的訊息主題。
lwtData	char* / String	離線數據，設定SmartQ QP-8x1/QP-8x2 or Windows Client離線後SmartQ QP-500會自動發送的訊息資料。
lwtQos	int	遺言的QOS設定。

lwtRetain	int	設定遺言訊息是否會被QP-500保留。
persistDir	char* / String	設定硬碟暫存機制所使用路徑。 僅提供Windows版本MQTT SDK使用。
常數MQISDP_CLIENT_ID_LENGTH其值為23的長整數。		

2.2 函式：

ICPDAS_MQIsdpConnect:

`int ICPDAS_MQIsdpConnect (PSPARMS *psp);`

函示功能說明：

協助使用者透過此函式更簡易地來連接上SmartQ QP-500。

參數說明：

Inputs:

- `psp`：
儲存SmartQ QP-8x1/QP-8x2 or Windows Client與SmartQ QP-500連接所需的資訊，
PSPARMS結構定義請參考2.1節。

Returns:

- 回傳整數值：
 - 0 : OK
 - 1002 : HOSTNAME NOT FOUND
 - 1007 : PERSISTENCE FAILED
 - 1010 : NO WILL TOPIC
 - 1013 : DATA TOO BIG
 - 1014 : ALREADY CONNECTED
- `psp`：
如果ICPDAS_MQIsdpConnect回傳值為MQISDP_OK(0)，則`psp->hConn`將回傳為一個有效的記憶體位址，反之`psp->hConn`將被設定為NULL。

ICPDAS_MQIsdpDisconnect:

`int ICPDAS_MQIsdpDisconnect (PSPARMS *psp);`

函示功能說明：

協助使用者透過此函式更簡易地中斷與SmartQ QP-500連線。

參數說明：

Inputs:

<http://www.icpdas.com>

- **psp** :
儲存SmartQ QP-8x1/QP-8x2 or Windows Client與SmartQ QP-500連接所需的資訊，PSPARMS結構定義請參考2.1節。

Returns:

- 回傳整數值：
 - 0 : OK
 - 1007 : PERSISTENCE_FAILED
 - 1008 : CONN_HANDLE_ERROR
 - **psp** :
psp->hConn回傳為NULL表示invalid connection 處理。
-

ICPDAS_MQIsdpPublish:

```
int ICPDAS_MQIsdpPublish ( PSPARMS *psp,  
  
                           char *pData,  
                           int dataLength );
```

函示功能說明：

協助使用者透過此函式更簡易地經由SmartQ QP-500來發佈訊息。

參數說明：

Inputs:

- **psp** :
儲存SmartQ QP-8x1/QP-8x2 or Windows Client與SmartQ QP-500連接所需的資訊，PSPARMS結構定義請參考2.1節。使用者必須先將所欲發佈的topic設定在psp->topic當中。
- **pData** :
SmartQ QP-8x1/QP-8x2 or Windows Client欲發佈與SmartQ QP-500的字串資料。
- **dataLength** :
欲發佈字串資料的長度。

Returns:

- 回傳整數值：
 - 0 : OK
 - 1003 : Queue FULL
 - 1007 : PERSISTENCE FAILED
 - 1008 : CONN HANDLE ERROR
 - 1011 : INVALID STRUCTURE LENGTH

1013 : DATA TOO BIG
1017 : CONNECTION BROKEN

- **psp :**

psp->lastSentMsg回傳值 :

1 : DELIVERED
2 : RETRYING
3 : IN PROGRESS
4 : MSG HANDLE ERROR

ICPDAS_MQIsdpSubscribe:

[int](#) ICPDAS_MQIsdpSubscribe (**PSPARMS** *psp);

函示功能說明 :

協助使用者透過此函式更簡易地向SmartQ QP-500來訂閱訊息。

參數說明 :

Inputs:

- **psp :**

儲存SmartQ QP-8x1/QP-8x2 or Windows Client與SmartQ QP-500連接所需的資訊，PSPARMS結構定義請參考2.1節。使用者必須先將所欲訂閱的topic設定在psp->topic當中。

Returns:

- 回傳整數值 :

0 : OK
1003 : Queue FULL
1007 : PERSISTENCE FAILED
1008 : CONNECTION HANDLE ERROR
1011 : INVALID STRUCTURE LENGTH
1013 : DAT TOO BIG
1017 : CONNECTION BROKEN

- **psp :**

如果ICPDAS_MQIsdpSubscribe回傳值為OK(0)，則psp->lastSentMsg將回傳一個有效值，反之psp->lastSentMsg將被設定為(-1)，表示INVALID MSG HANDLE。

psp->lastSentMsg回傳值 :

1 : DELIVERED

- 2 : RETRYING
 - 3 : IN_PROGRESS
 - 4 : MSG_HANDLE_ERROR
-

ICPDAS_MQIsdpUnsubscribe:

`int ICPDAS_MQIsdpUnsubscribe (PPSARMS *psp);`

函示功能說明：

協助使用者透過此函式更簡易地向SmartQ QP-500取消訂閱訊息。

參數說明：

Inputs:

- `psp`：
儲存SmartQ QP-8x1/QP-8x2 or Windows Client與SmartQ QP-500連接所需的資訊，PPSARMS結構定義請參考2.1節。使用者必須先將所欲取消訂閱的topic設定在`psp->topic`當中。

Returns:

- 回傳整數值：
 - 0 : OK
 - 1003 : Queue FULL
 - 1007 : PERSISTENCE FAILED
 - 1008 : CONN HANDLE ERROR
 - 1011 : INVALID STRUCTUR LENGTH
 - 1013 : DATA TOO BIG
 - 1017 : CONNECTION BROKEN
 - `psp`：
如果ICPDAS_MQIsdpUnsubscribe回傳值為OK(0)，則`psp->lastSentMsg`將回傳一個有效值，反之`psp->lastSentMsg`將被設定為INVALID MSG HANDLE (-1)。
`psp->lastSentMsg`回傳值：
 - 1 : DELIVERED
 - 2 : RETRYING
 - 3 : IN PROGRESS
 - 4 : MSG HANDLE ERROR
-

ICPDAS_MQIsdpReceive:

```
int ICPDAS_MQIsdpReceive ( PPSARMS *psp,  
                           char *topic,  
                           char *data );
```

函示功能說明：

協助使用者透過此函式更簡易地向SmartQ QP-500接收訊息。

參數說明：

Inputs:

- **psp**：
儲存SmartQ QP-8x1/QP-8x2 or Windows Client 與SmartQ QP-500連接所需的資訊，
PPSARMS結構定義請參考2.1節。
- **topic**：
用以儲存SmartQ QP-8x1/QP-8x2 or Windows Client從SmartQ QP-500所接收到的訊息之
主題。
- **data**：
用以儲存SmartQ QP-8x1/QP-8x2 or Windows Client從SmartQ QP-500所接收到的訊息內
文資訊。

Returns:

- 回傳整數值：
0 : OK
1008 : CONNECTION HANDLE ERROR
1005 : PUBS AVAILABLE
1006 : NO PUBS AVAILABLE
1018 : DATA TRUNCATED
- **topic**：
用以儲存SmartQ QP-8x1/QP-8x2 or Windows Client從SmartQ QP-500所接收到的訊息之
主題。
- **data**：
用以儲存SmartQ QP-8x1/QP-8x2 or Windows Client從SmartQ QP-500所接收到的訊息內
文資訊。

ICPDAS_MQIsdpTerminate:

`void ICPDAS_MQIsdpTerminate(PSPARMS *psp);`

函示功能說明：

協助使用者透過此函式更簡易地釋放資源。

參數說明：

Inputs:

- `psp`：
儲存SmartQ QP-8x1/QP-8x2 or Windows Client與SmartQ QP-500連接所需的資訊，
PSPARMS結構定義請參考2.1節。

Returns:

- 無回傳。

3. 範例程式說明

3.1 Publish.c (Publish_WinCE.cpp)

3.1.1 定義結構 PPARMS。(PPARMS 詳細結構請見第二章節)

3.1.2 設定變數：

(必要設定)

```
strcpy( psParams.clientId, "Client_Pub");  
// 設定SmartQ QP-8x1/QP-8x2 or Windows Client ID為"Client_Pub"。  
psParams.pBroker = (char*)malloc( strlen("192.168.100.63") + 1);  
// SmartQ QP-500 IP位置字串空間配置  
psParams.topic = (char*)malloc( strlen("MyTopic")+ 1);  
// SmartQ QP-8x1/QP-8x2 or Windows Client 欲發佈至SmartQ QP-500的訊息主題字串空間配置  
strcpy( psParams.pBroker, "192.168.100.63");  
// 此程式目前暫定SmartQ QP-500 IP為192.168.100.63, 使用者可依實際狀況調整IP。  
psParams.port = 1883;  
// 設定與SmartQ QP-500連接的通訊埠號碼，標準設置為1883。  
strcpy(psParams.topic, "MyTopic");  
// 設定訊息發佈的主題為MyTopic。  
psParams.keepAliveTime = 32000;  
// 設定與SmartQ QP-500連線的存活時間為32000毫秒。
```

// persistDir僅提供Windows版本SmartQ SDK使用。

```
psParams.persistDir = (char*)malloc( strlen("C:\\temp\\wmqtt")+ 1);  
// SmartQ Windows Client的硬碟暫存路徑字串空間配置。  
strcpy( psParams.persistDir, "C:\\temp\\wmqtt");  
// 設定與SmartQ Windows Client的硬碟暫存路徑，可依使用者需求修改路徑。
```

(選擇設定)

```
psParams.qos = 1;  
// 設定傳輸品質QoS為1，如未設定預設值為0。  
psParams.retain = 0;  
// 設定發佈資訊於SmartQ QP-500部份不保留。  
psParams.lwtTopic = (char*)malloc( strlen("MyTopic")+ 1);  
strcpy(psParams.lwtTopic, "MyTopic");  
// 為遺言主題配置記憶體空間並將其字串設定為MyTopic，如不使用遺言則可將其字串設為NULL。  
psParams.lwtQos = 1;  
// 設定遺言的傳輸品質QoS為1，如未設定預設值為0。  
psParams.lwtRetain = 0;
```

```
// 設定遺言訊息於SmartQ QP-500部份不保留。  
psParms.lwtData = (char*)malloc( strlen("Client_Pub terminated.")+ 1 );  
strcpy(psParms.lwtData, "Client_Pub terminated. ");  
// 為遺言資料配置記憶體空間並將其字串設定為Client_Pub terminated.，如不使用遺言則可將其字串  
// 設為NULL。。
```

3.1.3 連線：

```
ICPDAS_MQIsdpConnect( &psParms );  
// SmartQ QP-8x1/QP-8x2 or Windows Client 呼叫ICPDAS_MQIsdpConnect函式與SmartQ QP-500  
// 進行連線，傳入參數為psParms，若回傳值為0則連線成功，反之則連線失敗並結束程式。
```

3.1.4 發佈：

```
pBuffer = (char*)malloc(bufLen);  
// 為pBuffer配置記憶體空間  
fgets(pBuffer, bufLen, stdin);  
// 讀取由為用戶輸入的訊息字串。  
if ( msgLen <=0 ) {  
    ICPDAS_MQIsdpPublish( &psParms, "\0", 1 );  
    break; }  
// 當用戶僅輸入ENTER時程式結束，並通知Subscribe程式結束。  
ICPDAS_MQIsdpPublish( &psParms, pBuffer, msgLen );  
// SmartQ QP-8x1/QP-8x2 or Windows Client呼叫ICPDAS_MQIsdpPublish函式將Hello字串發佈與  
// SmartQ QP-500，傳入參數為psParms、pBuffer及msgLen。
```

3.1.5 中斷連線：

```
ICPDAS_MQIsdpDisconnect( &psParms );  
// SmartQ QP-8x1/QP-8x2 or Windows Client 呼叫ICPDAS_MQIsdpDisconnect函式與SmartQ  
// QP-500中斷連線，傳入參數為psParms。
```

3.1.6 釋放記憶體：

```
ICPDAS_MQIsdpTerminate( &psParms );  
// SmartQ QP-8x1/QP-8x2 or Windows Client 呼叫ICPDAS_MQIsdpTerminate()，以結束所有工作及  
// 資源。  
free( pBuffer );  
// 釋放變數所配置的記憶體空間。
```

3.2 SubScribe.c (SubScribe _WinCE.cpp)

3.2.1 定義結構 PSPARMS。(PSPARMS 詳細結構請見第二章節)

3.2.2 設定變數：

(必要設定)

```
strcpy( psParams.clientId, "Client_Sub" );  
// 設定SmartQ QP-8x1/QP-8x2 or Windows Client ID為"Client_Sub"。  
psParams.pBroker = (char*)malloc( strlen("192.168.100.63") + 1 );  
// SmartQ QP-500 IP位置字串空間配置  
psParams.topic = (char*)malloc( strlen("MyTopic")+ 1 );  
// SmartQ QP-8x1/QP-8x2 or Windows Client欲發佈至SmartQ QP-500的訊息主題字串空間配置  
strcpy( psParams.pBroker, "192.168.100.63" );  
// 此程式目前暫定SmartQ QP-500 IP為192.168.100.63, 使用者可依實際狀況調整IP。  
psParams.port = 1883;  
// 設定與SmartQ QP-500連接的通訊埠號碼, 標準設置為1883。  
strcpy(psParams.topic, "MyTopic" );  
// 設定訊息訂閱的主題為MyTopic。  
psParams.keepAliveTime = 32000;  
// 設定與SmartQ QP-500連線的存活時間為32000毫秒。  
psParams.lastSentMsg = MQISDP_INV_MSG_HANDLE;  
// 預設處理狀況為-1。
```

// persistDir僅提供Windows版本SmartQ SDK使用。

```
psParams.persistDir = (char*)malloc( strlen("C:\\temp\\wmqtt")+ 1 );  
// SmartQ Windows Client的硬碟暫存路徑字串空間配置。  
strcpy( psParams.persistDir, "C:\\temp\\wmqtt");  
// 設定與SmartQ Windows Client的硬碟暫存路徑, 可依使用者需求修改路徑。
```

(選擇設定)

```
psParams.qos = 1;  
// 設定傳輸品質QoS為1, 如未設定預設值為0。  
psParams.timeout = 10000;  
// 設定等待接收資訊的時間為10000毫秒。  
psParams.lwtTopic = (char*)malloc( strlen("MyTopic")+ 1 );  
strcpy(psParams.lwtTopic, "MyTopic" );  
// 為遺言主題配置記憶體空間並將其字串設定為MyTopic, 如不使用遺言則可將其字串設為NULL。  
psParams.lwtQos = 1;  
// 設定遺言的傳輸品質QoS為1, 如未設定預設值為0。  
psParams.lwtRetain = 0;
```

```
// 設定遺言訊息於SmartQ QP-500部份不保留。  
psParms.lwtData = (char*)malloc( strlen("Client_Sub terminated.")+ 1 );  
strcpy(psParms.lwtData, "Client_Sub terminated. ");  
// 為遺言資料配置記憶體空間並將其字串設定為Client_Sub terminated.，如不使用遺言則可將其字串  
設為NULL。。
```

3.2.3 連線：

```
ICPDAS_MQIsdpConnect( &psParms );  
// SmartQ QP-8x1/QP-8x2 or Windows Client呼叫ICPDAS_MQIsdpConnect函式與SmartQ QP-500進  
行連線，傳入參數為psParms，若回傳值為0則連線成功，反之則連線失敗並結束程式。
```

3.2.4 訂閱：

```
ICPDAS_MQIsdpSubscribe( &psParms );  
// SmartQ QP-8x1/QP-8x2 or Windows Client呼叫ICPDAS_MQIsdpSubscribe函式以通知SmartQ  
QP-500其訂閱那些訊息主題，傳入參數為psParms。
```

3.2.5 接收：

```
rcvTopic = (char*)malloc( BUFFER_SIZE );  
rcvData = (char*)malloc( BUFFER_SIZE );  
// 為rcvTopic及rcvData配置記憶體空間。  
ICPDAS_MQIsdpReceive( &psParms, rcvTopic, rcvData );  
// SmartQ QP-8x1/QP-8x2 or Windows Client呼叫ICPDAS_MQIsdpReceive函式進行訂閱主題的訊息  
接收，傳入參數為psParms、rcvTopic及rcvData，並經由rcvTopic及rcvData取得所接收到的主題及  
資訊。  
// 若psParms.timeout = -1; 則此函式會在有接收到主題資訊後才會返回 (僅于Multi Task下支援)  
// 若psParms.timeout 其他資料(單位為ms); 則此函式會在有接收到主題資訊或Timeout時間到達而返  
回
```

3.2.6 取消訂閱：

```
ICPDAS_MQIsdpUnsubscribe( &psParms );  
// SmartQ QP-8x1/QP-8x2 or Windows Client呼叫ICPDAS_MQIsdpUnsubscribe函式以通知SmartQ  
QP-500其取消訂閱那些訊息主題，傳入參數為psParms。
```

3.2.7 中斷連線：

```
ICPDAS_MQIsdpDisconnect( &psParms );  
// SmartQ QP-8x1/QP-8x2 or Windows Client呼叫ICPDAS_MQIsdpDisconnect函式與SmartQ QP-500  
中斷連線，傳入參數為psParms。
```

3.2.8 釋放記憶體：

```
ICPDAS_MQIsdpTerminate ( &psParms );  
// SmartQ QP-8x1/QP-8x2 or Windows Client 呼叫ICPDAS_MQIsdpTerminate()，以結束所有工作及  
資源。  
free( rcvData );  
free( rcvTopic );
```

// 釋放變數所配置的記憶體空間。