# 第 13 章 使用 VB.net 2008 程式來 讀/寫 Win-GRAF 變數

本章以 Visual Studio .NET 2008 開發工具建立一個範例程式的方式來說明,範例程式可以在

XP-8xx8-CE6, WP-8xx8, WP-8xx8-CE7, VP-x2x8-CE7, WP-5xx8-CE7 產品盒內附的 CD-ROM 內找到。

### VB.NET 範例:

光碟:\napdos\Win-GRAF\demo-project\vb.net\_2008\_demo\

demo\_vb01: 數位 I/O 範例,搭配 I-87055W 模組 (於 Slot 0)

- demo\_vb02: 類比 I/O 範例, 搭配 I-87024W (Slot 1) 與 I-8017HW (Slot 2) 模組
- demo\_vb03: 讀/寫 Win-GRAF Internal Integers, Timer, Real 以及 String 變數 (無需 I/O 模組)
- demo\_vb04: 讀/寫 Win-GRAF Internal String 變數 (無需 I/O 模組)

#### Win-GRAF 範例:

光碟:\napdos\Win-GRAF\demo-project\

"demo\_vb01.zip", "demo\_vb02.zip", "demo\_vb03.zip", "demo\_vb04.zip"

# 13.1 如何回存 Win-GRAF 專案?

以下介紹如何把 Win-GRAF 範例程式,回存到 Win-GRAF Workbench 中。

首先,滑鼠點選工具列 "File" -> "Add Exiting Project" -> "FromZip...",並選擇想要回存的 Win-GRAF 專案 (zip 檔),即可回存該專案。之後,請執行編譯並將專案下載到 PAC 內。



Win-GRAF 使用手冊, 1.07 版, 2019 年 1 月 ICP DAS 13-1

# 13.2 如何開放 Win-GRAF 變數給 .NET 程式使用?

Win-GRAF 中,除了字串 (String) 變數之外,其它欲開放的變數需在 "Binding" 設定中產生一個位址 才能提供.NET 程式進行存取。以下將示範如何開放變數:

- 1. 滑鼠點選工具列上的 "Open Binding Configuration" 按鈕來開啟 "Binding" 視窗。
- 2. 點選 "PUBLIC (:9000)" 來設定要公開的資料, "Address" 欄位無需填寫, "Port" 欄位固定為 "9000", 請勿更動。

Wm Win-GRAF - Test_3			Open Binding Configuration				
<u>File Edit V</u> iew Insert Project <u>T</u> ools <u>W</u> indow <u>H</u> elp			Sinuing Co	iingu	Tation		
BBBBBBBBB	🗙 🔍 🖕 🗢 🗠 🕌 🏭 品 🖲	0 🖸 E	§9 🗗 🎽	ا <b>لا</b> کر 🖁			
Workspace	Binding	-					XX
⊟ 🗿 Test_3	A D C Network	Name	Value		🍸 Name	Туре	1
🗄 🛅 Exception programs	📃 🧮 RUBLIC (:9000)	Name	PUBLIC		🚮 GI	obal variables	^
📩 🛁 Programs		Address			🔛 RE	ETAIN variables	
🖭 🎟 Main		Port	9000		🗋 M:	ain	
📥 🛅 Watch (for debugging)					P:: pC	)nBadIndex	~
Soft Scope						-	
🛅 Initial values	Name Identifier	POBLIC	" 區 个 必	ister	Name	 戀數區	
Binding Configuration		填寫 Ad	dress °	Stor.	Name		
Line 🔁 Types							
	<			>			
	Binding Variables				1		

 在設定公開資料之前,您必須在變數區先建立好要公開的變數。滑鼠點選 "Global variables" 再按 "Ins" 鍵來新增變數項目,下表為 "Test\_3" 範例所使用的變數,您可依實際需求來設定,設定完 成後,畫面如下。

變數名稱	資料型態
Public_BOOL	BOOL
Public_INT	INT
Public_DINT	DINT
Public_WORD	WORD
Public_REAL	REAL

			ΖX
Y	Name	Туре	D
	🗉 🚮 Global va	riables 📐	^
	Public_BOOL	BOOL	
	Public_INT	INT	
	Public_DINT	DINT	
	Public_WORD	WORD	
	Dublic DEAL	DEAL	
	Public_REAL		
	📄 Main 🗖		
		Enter 鍵	

4. 如下圖, 滑鼠點選在 "PUBLIC(:9000)", 接著選取變數區內欲公開的變數資料, 並將其拖曳到
 "Name" 區域。"Identifier" 欄位會自動產生編號, 若其他 VB 或 .Net 程式想取用變數資料, 需
 設定一樣的 ID 編號。

注意: "PUBLIC" 最多可使用 8192 個變數,"Identifier" 編號只能是 "1~8192"。

E		Name	Value	7 Name		Туре	
🗄 🚠 PUBI	.IC (:9000)	Name	PUBLIC	🗏 🔂 🖓 G	lobal varia	bles	
0 states		Address		Public	BOOL	BOOL	٦
	•	Port	9000	Public Public Public Public		INT DINT WORD REAL	
Name	Identifier		ositive hysteresis				>
Public_BOOL	1			Name	Valu	le	
Public_INT	2						
Public_DINT	3						
Public_WORD	4						
Public BEAL	5						
1 GENO_THEFTE							

另外,如需開放 Win-GRAF 的 String 變數,可使用 "Pub\_String" 函式。在 Win-GRAF 專案的程式裡,加入此段 ST 語言程式。(您可回存 Win-GRAF demo\_vb04 範例程式,來查看以下程式內容)

Pub\_string(Address, String\_val);

Address: 開放的位址編號,範圍可以是 1~1024 String\_val: String 變數的名稱

### 變數說明:

名稱	型態	說明
Init	BOOL	初始化用·初值設定為 True。
Tmp_val	ROOL	判斷開放位址是否成功。
	BOOL	"True" 表示成功."False" 表示失敗。
msg1	STRING, 字串長度為 100	
msg2	STRING, 字串長度為 32	
msg3	STRING, 字串長度為 60	│ <u>/注息</u> :子中 <b>反</b> 归以定 1 ~255 ◎

### 程式內容:

If init then Init := false; (\* 開放位址為1的字串值\*) Tmp\_val := pub\_string(1,msg1);

(\* 開放位址為 2 的字串值 \*) Tmp\_val := pub\_string(2,msg2);

(\* 開放位址為 3 的字串值 \*)

Tmp\_val := pub\_string(3,msg3);

End\_if;

# 13.3 建立 VB.NET 新專案

1. 開啟微軟 Visual Studio .NET 2008 軟體,點選 [File] > [New Project]。



 點選 [Smart Device] > [.NET frame work 3.5] > [Smart Device Project], 在下方輸入專案名稱 (本例: project1), 然後按 "OK"。

Project types:		Templates:	.NET Framework 3.5	
<ul> <li>■ Visual Basic</li> <li>■ Windows</li> <li>■ Web</li> <li>■ Smart Devin</li> <li>■ Office</li> <li>■ Database</li> <li>■ Reporting</li> <li>■ Test</li> <li>■ WCF</li> <li>■ Workflow</li> <li>■ Other Language</li> <li>■ Other Project Ty</li> <li>■ Test Projects</li> </ul>	es ypes	Visual Studio installed templates		2
A project for Smart l	Device applications. C	Choose target platform, Framework version, and template :	in the next dialog box.	
<u>N</u> ame:	project1			
			OK	Cancel

3. 點選 [Device Application] > [Windows CE] > [.NET Compact Framework Version 3.5], 然後按 "OK"。

Target <u>p</u> latform:	Windows CE		¥
.NET Compact Framework version:	.NET Compact Framework	k Version 3.5	2
<u>T</u> emplates:	NET Compact Framework	x Version 2.0 x Version 3.5	
Device Application 1 Class Library Ag Class Library Ag Class Library Ag	Console Control pplication Library	Empty Project	Description: A project for creating a .NET Compact Framework 3.5 forms application for Windows CE Platform
			OK Cancel

Win-GRAF 使用手冊, 1.07 版, 2019 年 1 月 ICP DAS 13-4

# 13.3.1 加入專案參考 (Project Reference)

"UserShareNet"程式庫 (Library) 中包含了所有要跟 Win-GRAF 專案交換資料用的函式 (Function),而 在程式裡使用 "UserShare" 關鍵字之前,您必須在程式的參考清單裡加入參考: "UserShareNet.dll"。

- 請於出貨光碟中 (\napdos\Win-GRAF\demo-project\demo\_vb01\vb01\),將 "UserShareNet.dll" 檔 複製到目前專案的位置底下(例如: "C:\project1")。
- 7. 滑鼠右鍵點擊 "Solution Explorer" 視窗中的專案名稱 (例如: "project1"),再選擇 "Add Reference ..."。



3. 點選 "Browse" 頁籤,並切換到目前的專案位置 (例如: "C:\project1"),再選擇 "UserShareNet.dll",並按 "OK" 即完成了。

NET Projects Browse Recent 搜尋位置①: C project	
i My Project i obj □ W W W W	切換到目前的專案位置 (例如: "C:\project1")。
檔案名稱(11):	×
檔案類型( <u>T</u> ): Component Files (	*.dll;*.tlb;*.olb;*.ocx;*.exe)
	OK Cancel

註: 加入 "UserShareNet.dll" 後,下次需開發新專案時,即可從先前開發的專案目錄中 (例如: "C:\project1"),將此檔案複製並加入到新的專案中,或是,可預先從光碟中複製 "UserShareNet.dll" 到固定路徑中 (例如: "C:\dll\_lib")。 4. 加入 "UserShareNet.dll" 後,請雙擊專案的 "My Project",確認是否已加入 "UserShareNet.dll"。

	Solution Explorer       中 ×         Image: Solution Explorer       中 ×         Image: Solution Explorer       Image: Solution Explorer         Image: Solution Explorer       Image: Solution Explorer
Application	Configuration: N/A V Platform: N/A V
Compile	
Debug	References: Unused References Reference Paths
References	Reference Name         Type         Ver         Copy Local         Path           System         NET         2.0.00         False         C:\Program Files\Microsoft.NET\SDK\CompactFrameworkw2.0\W           System         NET         2.0.00         False         C:\Program Files\Microsoft.NET\SDK\CompactFrameworkw2.0\W           System         NET         2.0.00         False         C:\Program Files\Microsoft.NET\SDK\CompactFrameworkw2.0\W
Resources	System Drawing NET 2.0.0.0 False C. Program Files Microsoft NET SDK CompactFramework 2.0.W System Windows For NET 2.0.0.0 False C. Program Files Microsoft NET SDK CompactFramework 2.0.W System Windows For NET 2.0.0.0 False C. Program Files Microsoft NET SDK CompactFramework 2.0.W
Signing	System Xml NFI Z000 Faise C. Program FilesMicrosoft NFI SUKV ompactbrameworkWZ0WU UserShareNet .NET 1.0.1.0 True C:Documents and Settings/user/桌面/wingrafdoc/Wapdos/Win-GR
Devices	
	<>

5. 以滑鼠右鍵點選 "Form1.vb",選擇 "View Code",並在第一、二行插入 "Option Explicit On" 與 "Imports UserShareNet" (如下圖)。

Solution Explorer 🛛 🚽 🗙	Form1.vb
🔓 😰 🖬 🗉 🖧	�;Form1
project1     My Project     Form1.vb	Option Explicit On Imports UserShareNet
右鍵 ① Open Ope <u>n</u> With ② View Code	∃Public Class Form1  End Class

接著,就可以在您的 VB 表格中設計所需的物件與動作了。請參考 <u>13.5 節</u> 的說明來使用 "UserShareNet.dll"內的函式,來讀/寫 Win-GRAF 專案內的變數資料。

程式編寫完成後,請依照以下步驟來編譯 (Build) 並產生一個執行檔 (.exe)。

1. 請記得隨時按下 "Save All" 工具按鈕來存檔。

😤 pro	ject1 -	Microso	oft Visual (	Studio		
Eile	<u>E</u> dit	∐iew	Project	<u>B</u> uild	Debug	D
: 🛅	۵ 🔮	i 🛅	- 📃 🚺	¥,	<b>b B</b>	
Toolbo	X Dovid	e Con	→ I trole v2	Save A	llorm1.vt	*
Per Pe	ointer				Form1	
🛛 🙄 Bi	ndinoS	ource				

2. 點選 "Build" > "Build project1" 來編譯此專案 (project1),於 "Error List" 視窗中會顯示編譯結果。

🔞 project1 - Microsoft Visual Stud	0
Eile Edit View Project Bui	ld <u>D</u> ebug Data F <u>o</u> rma
🚰 📎 📂 🖽 🗕 🗿 🏧	Build project1
Toolbox 👻 🕈	Rebuild project1
All Device Controls v2	Deploy p Error List
🖓 Fonnier 🃅 BindingSource	Clean prc 📀 0 Errors 🚹 0 Warnings 🕕 0 Messages
	Description
	Build succeeded

3. 编譯完成後,執行檔會存放在以下目錄中。



請將此執行檔 (例如: "project1.exe") 複製到 PAC 的 \System\_Disk\Win-GRAF\ 目錄下來執行。

### <u>注意:</u>

使用者可複製 VB.net 執行檔到其他目錄下執行,但請記得同時複製相關的 DLL 檔案,否則執行 會有錯誤。例如: 要在 \Micro\_SD\ 目錄下執行 "project1.exe",該目錄必須有以下 3 個檔案,即 "project1.exe"、"UserShareNet.dll"、"Quicker.dll" 檔。

(可在光碟的 "\System\_disk\Win-GRAF\" 目錄下取得 "UserShareNet.dll" 與 "Quicker.dll" 檔案)

# 13.5 "UserShareNet.DLL" 內的函式說明

本節著重於 "UserShareNet.DLL" 内的函式 (Function) 來進行說明。

"UserShareNet.DLL"提供了許多函式,可用來讀/寫 Win-GRAF 專案裡的變數,以下分為幾類:

- 1. 讀/寫 Boolean
- 2. 讀/寫 8-bit 整數
- 3. 讀/寫 16-bit 整數
- 4. 讀/寫 32-bit 整數
- 5. 讀/寫 64-bit 整數
- 6. 讀/寫 32-bit 實數
- 7. 讀/寫 64-bit 實數

※請參考附錄 A 來查看 Win-GRAF 變數的資料型態與範圍

# 13.5.1 讀/寫 Boolean 的函式

# Set\_BOOL

### 說明:

設定指定位址編號的 Boolean 變數值。

# 語法:

UserShare.Set\_BOOL ( iUserAddress As System.UInt16 , ByVal iStatus As byte) as Byte

### 參數:

iUserAddress: 指定變數的位址編號。(1~8192) iStatus: 設定變數的狀態;例如: iStatus = 1 表示 "True", iStatus = 0 表示 "False"

### 範例:

"設定位址 "1"的 Win-GRAF Boolean 變數為 "True". UserShare.Set\_BOOL(Convert.ToUInt16(1), 1)

### 範例程式:

光碟: \napdos\Win-GRAF\demo-project\vb.net\_2008\_demo\demo\_vb01



# 說明:

讀出指定位址編號的 Boolean 變數值。

# 語法:

UserShare.Get\_BOOL ( iUserAddress As System.UInt16 , ByRef iStatus As byte)

### 參數:

iUserAddress: 指定變數的位址編號 (1~8192)

iStatus: 取得變數的狀態; iStatus = 1 表示 "True", iStatus = 0 表示 "False"

# 範例:

\* 取得 Win-GRAF Boolean 位址編號 "1" 的變數狀態.
 Dim iStatus As Byte
 UserShare.Get\_BOOL(Convert.ToUInt16(1), iStatus)

### 範例程式:

光碟: \napdos\Win-GRAF\demo-project\vb.net\_2008\_demo\demo\_vb01

# 13.5.2 讀/寫 整數 的函式

### Set\_SINT Set\_INT Set\_DINT Set\_LINT

## 說明:

設定指定位址編號的 Win-GRAF 8-bit Integer, 16-bit Integer, 32-bit Integer 及 64-bit Integer 變數值。

### 語法:

UserShare.Set\_SINT (ByVal *iUserAddress* As System.UInt16, ByVal *iStatus* As SByte) As <u>Byte</u> UserShare.Set\_INT (ByVal *iUserAddress* As System.UInt16, ByVal *iStatus* As Short) As <u>Byte</u> UserShare.Set\_DINT (ByVal *iUserAddress* As System.UInt16, ByVal *iStatus* As <u>Integer</u>) As <u>Byte</u> UserShare.Set\_LINT (ByVal *iUserAddress* As System.UInt16, ByVal *iStatus* As <u>long</u>) As <u>Byte</u>

### 參數:

iUserAddress: 指定變數的位址編號 (1~8192) iStatus : 設定 8-bit Integer, 16-bit Integer, 32-bit Integer 及 64-bit Integer 值

### 範例:

(設定 32-bit integer 值 "1234567" 到位址編號 "1" 的變數. UserShare.Set\_DINT(Convert.ToUInt16(1), Convert.ToInt32(1234567))

\* 設定 integer 值 "-1234" 到位址編號 "2" 的變數. UserShare.Set\_INT(Convert.ToUInt16(3), Convert.ToInt16(-1234))

'設定 64-bit integer 值 "123456789012345" 到位址編號 "3" 的變數. UserShare.Set\_LINT(Convert.ToUInt16(3), Convert.ToInt64(123456789012345))

'設定 8-bit integer 值 "125" 到位址編號 "4" 的變數. UserShare.Set\_SINT(Convert.ToUInt16(3), Convert.ToSByte(125) )

### 範例程式:

- 1. 讀/寫 類比 I/O: \napdos\Win-GRAF\demo-project\vb.net\_2008\_demo\demo\_vb02
- : 讀/寫 Internal long integer 
   Timer 及 Real (浮點數)的值:
   \napdos\Win-GRAF\demo-project\vb.net\_2008\_demo\demo\_vb03

#### 📕 Get\_SINT 📕 Get\_INT 📕 Get\_DINT 📕 Get\_LINT

### 說明:

讀出指定位址編號的 Win-GRAF 8-bit Integer, 16-bit Integer, 32-bit Integer 及 64-bit Integer 變數值。

# 語法:

UserShare. Get\_SINT (ByVal *iUserAddress* As System.UInt16 , ByRef *iStatus* As <u>SByte</u>) As <u>Byte</u> UserShare. Get\_INT (ByVal *iUserAddress* As System.UInt16 , ByRef *iStatus* As <u>Short</u>) As <u>Byte</u> UserShare.Get\_DINT (ByVal *iUserAddress* As System.UInt16 , ByRef *iStatus* As <u>Integer</u>) As <u>Byte</u>

UserShare. Get LINT (ByVal iUserAddress As System.UInt16, ByRef iStatus As long) As Byte

#### 參數:

iUserAddress: 指定變數的位址編號 (1~8192) iStatus: 取得 8-bit Integer, 16-bit Integer, 32-bit Integer 及 64-bit Integer 值

### 範例:

Dim Dlong\_val As Int64 Dim short\_val As Int16 Dim long\_val As Int32 Dim Sbyte\_val as sbyte

"取得位址編號 "7"的 64-bit 整數的變數值. UserShare.Get\_LINT(Convert.ToUInt16(7), Dlong\_val)

'取得位址編號 "8"的 32-bit 整數的變數值. UserShare.Get\_DINT(Convert.ToUInt16(8), long\_val)

'取得位址編號 "9" 的 16-bit 整數的變數值. UserShare.Get\_INT(Convert.ToUInt16(9), short\_val)

'取得位址編號 "10" 的 8-bit 整數的變數值. UserShare.Get\_SINT(Convert.ToUInt16(9), sbyte\_val)

#### 範例程式:

- . 讀/寫 類比 I/O: \napdos\Win-GRAF\demo-project\vb.net\_2008\_demo\demo\_vb02
- 2. 讀/寫 Internal long integer 、 Timer 及 Real (浮點數) 的值: \napdos\Win-GRAF\demo-project\vb.net\_2008\_demo\demo\_vb03

# 13.5.3 讀/寫 實數 的函式

Get\_REAL Get\_LREAL

### 說明:

讀出指定位址編號的 Win-GRAF 32-bit Real、64-bit Real 變數值。

### 語法:

UserShare. Get\_REAL (ByVal iUserAddress As System.UInt16 , ByRef iStatus As Single) As Byte

UserShare. Get\_LREAL(ByVal iUserAddress As System.UInt16 , ByRef iStatus As Double) As Byte

### 參數:

iUserAddress: 指定變數的位址編號 (1~8192) iStatus: 取得浮點數數值

### 範例:

Dim float\_val As Single Dim double\_val As Double

\*取得位址編號 "7" 的 double 變數值.UserShare.Get\_LREAL(Convert.ToUInt16(7), double\_val)

'取得位址編號 "8"的 Single 變數值. UserShare.Get\_REAL(Convert.ToUInt16(8), float\_val)

### 範例程式:

- . 讀/寫 類比 I/O: \napdos\Win-GRAF\demo-project\vb.net\_2008\_demo\demo\_vb02
- 2. 讀/寫 Internal long integer 、Timer 及 Real (浮點數) 的值: \napdos\Win-GRAF\demo-project\vb.net\_2008\_demo\demo\_vb03

### Set\_REAL Set\_LREAL

### 說明:

寫入指定位址編號的 Win-GRAF 32-bit Real、64-bit Real 變數值。

# 語法:

UserShare. Set\_REAL (ByVal iUserAddress As System.UInt16, ByVal iStatus As Single) As Byte

UserShare. Set\_LREAL(ByVal iUserAddress As System.UInt16, ByVal iStatus As Double) As Byte

### 參數:

- iUserAddress: 指定變數的位址編號 (1~8192)
- iStatus: 設定浮點數數值

# 範例:

'寫入 11234.234567 到位址編號 "7" 的變數 UserShare.Set\_LREAL(Convert.ToUInt16(7),Convert.ToDouble(11234.234567))

'寫入 123.12 到位址編號 "8" 的變數 UserShare.Set\_REAL(Convert.ToUInt16(8), Convert.ToSingle (123.12))

### 範例程式:

- 1. 讀/寫 類比 I/O: \napdos\Win-GRAF\demo-project\vb.net\_2008\_demo\demo\_vb02
- 讀/寫 Internal long integer 、Timer 及 Real (浮點數) 的值: \napdos\Win-GRAF\demo-project\vb.net\_2008\_demo\demo\_vb03

# 13.5.4 讀/寫 字串 的函式

### Get\_STRING

# 說明:

讀取指定位址編號的 Win-GRAF 字串變數值。

### 語法:

UserShare. Get\_STRING (ByVal iUserAddress As System.UInt16, ByVal msg() As Byte) As Byte

# 參數:

iUserAddress: 指定變數的位址編號 (1~1024) msg(): 取得/設定 Win-GRAF 的字串值

# 範例:

Dim str\_val As String Dim msg() As Byte '取得位址編號 "7"的 String 變數值. UserShare.Get\_STRING(Convert.ToUInt16(7),msg) str\_val= byte\_array\_to\_unicode(msg)

Private Function byte\_array\_to\_unicode(ByVal buf() As Byte) As String Dim tmpmsg As String If buf.Length > 255 Then Return Nothing End If tmpmsg = System.Text.Encoding.GetEncoding("UTF-8").GetString(buf, 0, buf.Length)

Return tmpmsg End Function

### 範例程式:

光碟:

 讀/寫 STRING 的值: \napdos\Win-GRAF\demo-project\vb.net\_2008\_demo\demo\_vb04

### Set\_STRING

### 說明:

寫入指定位址編號的 Win-GRAF 字串值。

# 語法:

UserShare. Set\_STRING (ByVal iUserAddress As System.UInt16, ByVal msg() As Byte) As Byte

# 參數:

iUserAddress: 指定變數的位址編號 (1~1024) msg(): 設定 Win-GRAF 的字串值

### 範例:

Dim str\_val As String="Hello World" Dim msg() As Byte

msg= unicode\_to\_byte\_array(str\_val) ' 寫入位址編號 "7"的 String 變數值. UserShare.Set\_STRING(Convert.ToUInt16(7),msg )

'轉換字串成 byte array

Private Function unicode\_to\_byte\_array(ByVal msg As String) As Byte() Dim tmpbuf() As Byte If msg.Length > 255 Then Return Nothing End If tmpbuf = System.Text.Encoding.GetEncoding("UTF-8").GetBytes(msg) Return tmpbuf

End\_Function

### 範例程式:

光碟:

 讀/寫 STRING 的值: \napdos\Win-GRAF\demo-project\vb.net\_2008\_demo\demo\_vb04

# 13.5.5 如何讓 VB.NET 程式讀取 Win-GRAF 字串變數?

.NET 應用程式如果要寫入字串變數,必須根據目前使用的語言編碼 (例如: UTF-8) 轉換成 byte 陣列 才能寫入,若是讀出字串變數的陣列內容,則需要根據語言編碼轉換成字串。以下提供 VB.NET 程 式轉換的範例

(編碼為 UTF-8):

### String 轉成 byte 陣列

Private Function unicode\_to\_byte\_array(ByVal msg As String) As Byte() Dim tmpbuf() As Byte If msg.Length > 255 Then Return Nothing End If

tmpbuf = System.Text.Encoding.GetEncoding("UTF-8").GetBytes(msg)
Return tmpbuf
End Function

#### byte 陣列轉字串

Private Function byte\_array\_to\_unicode(ByVal buf() As Byte) As String Dim tmpmsg As String If buf.Length > 255 Then Return Nothing End If tmpmsg = System.Text.Encoding.GetEncoding("UTF-8").GetString(buf, 0, buf.Length)

Return tmpmsg End Function

# 第 14 章 使用 C# 程式來 讀/寫 Win-GRAF 變數

本章以 Visual Studio .NET 2008 開發工具建立一個範例程式的方式來說明,範例程式可以在 XP-8xx8-CE6, WP-8xx8, WP-8xx8-CE7, VP-x2x8-CE7, WP-5xx8-CE7 產品盒內附的 CD-ROM 內找到。

### C# 範例:

光碟:\napdos\Win-GRAF\demo-project\CSharp.net\_2008\_demo\ demo\_CSharp01: 數位 I/O 範例,搭配 I-87055W 模組 (於 Slot 0) demo\_CSharp02: 類比 I/O 範例,搭配 I-87024W (Slot 1) 與 I-8017HW (Slot 2) 模組 demo\_CSharp03: 讀/寫 Win-GRAF Internal Integer, Timer,及 Real 變數 (無需 I/O 模組) demo CSharp04: 讀/寫 Win-GRAF String 變數 (無需 I/O 模組)

### Win-GRAF 範例:

光碟 : \napdos\Win-GRAF\demo-project\ "demo\_vb01.zip", "demo\_vb02.zip", "demo\_vb03.zip", "demo\_vb04.zip"

# 14.1 如何回存 Win-GRAF 專案?

請參考 13.1 節 來回存 Win-GRAF 專案。

# 14.2 如何開放 Win-GRAF 變數給 C# 程式使用?

請參考 <u>13.2 節</u> - 開放 Win-GRAF 變數給 .NET 程式使用的方式。

# 14.3 建立 C# 新專案

1. 開啟微軟 Visual Studio .NET 2008, 點選功能表 [File] > [New Project]。

📽 Start Page - Microsoft Visual Studio							
E	ile	<u>E</u> dit <u>V</u> iew	<u>T</u> ools	Te <u>s</u> t	Window	Help	
		New Project		Ctrl	+N	L  🔜   🗏 😫   り ▾ (པ ▾ 🕮 ▾ 🖳   🕨 🗉	
5	0	New <u>W</u> eb Site	Sh	ift+Alt	+N Pa	age	
[ ]		<u>N</u> ew File					
	ĩ	Open <u>P</u> roject		Ctrl	+0	Visual Studio 2008	

2. 點選 [Smart Device] > [.NET frame work 3.5] > [Smart Device Project] · 並輸入專案名稱 (例如: "project1") · 然後點選 "OK" 。

Project types:	Templates:	.NET Framework 3.5
Database     Reporting     Test     Workflow     Other Languages     Windows     Web     Smart Device     Office     Database     Reporting     Test     WCF     Workflow     Window	Visual Studio installed templates Smart Device Project My Templates Search Online Templates	
A project for Smart Device applications. C	hoose target platform, Framework version, and template	in the next dialog box.
Name: project1		
		OK Cancel

3. 點選 [Device Application] > [Windows CE] > [.NET Compact Framework Version 3.5], 然後點選 "OK"。

Target platform: .NET <u>C</u> ompact Framework version: <u>T</u> emplates:	Windows CE .NET Compact Framework .NET Compact Framework .NET Compact Framework	ork Version 3.5 ork Version 2.0 ork Version 3.5	×
Device Application Class Library Ap	Console Control Library	Empty Project	Description: A project for creating a .NET Compact Framework 3.5 forms application for Windows CE Platform
Download additonal emulator images	and smart device SDKs		OK Cancel

# 14.3.1 加入 C# 專案參考

"UserShareNet"程式庫 (Library) 中包含了所有要跟 Win-GRAF 專案交換資料用的函式,而在程式裡使用 "UserShare" 關鍵字之前,您必須在應用程式的參考清單裡加入參考: "UserShareNet.dll"。

- 請於出貨光碟中 (\napdos\Win-GRAF\demo-project\demo\_CSharp01\demo\_CSharp01\), 將 "UserShareNet.dll" 檔複製到目前專案的位置底下(例如: "C:\project1")。
- 7. 滑鼠右鍵點選 "Solution Explorer" 視窗中的 專案名稱 (例如: "project1"), 再選擇 "Add Reference ..."。



3. 點選 "Browse" 頁籤,並切換到目前的專案位置 (例如: "C:\project1"),再選擇 "UserShareNet.dll",並按 "OK" 即完成了。

NET Projects Brows Recen 搜尋位置①: C project1	nt C 🎓 📂 🖽 -
Dobj Properties	切換到目前的專案位置 (例如: "C:\project1")。
檔案名稱(11):	×
檔案類型(I): Component Files	(*.dll;*.tlb;*.olb;*.ocx;*.exe)
	OK Cancel

 註:加入 "UserShareNet.dll" 後,下次需開發新專案時,即可從先前開發的專案目錄中 (例如: "C:\project1"),將此檔案複製並加入到新的專案中。或是,可預先從光碟中複製 "UserShareNet.dll"到固定路徑中 (例如: "C:\dll\_lib")。 4. 加入 "UserShareNet.dll" , 會出現在 "Solution Explorer" 視窗。



5. 以滑鼠右鍵點選 "Form1.cs",選擇 "View Code",將游標移到最上方,在第一個區段中加入 "using UserShareNet"。



接著,就可以在您的 C# 表單中設計所需的物件與動作了。請參考 14.5 節 的說明來使用 "UserShareNet.dll"內的函式,來讀/寫 Win-GRAF 內的變數資料。

程式編寫完成後,請依照以下步驟來編譯 (Build) 並產生一個執行檔 (.exe)。

1. 請記得隨時按下 "Save All" 工具按鈕來存檔。

🐏 project1 - Microsoft Visual Studio						
Eile	<u>E</u> dit	∐iew	Project	<u>B</u> uild	Debug	D
: 🛅	۵ 🔮	3 🛅 ·	- 📃 🚺	Υ.	<b>b B</b>	
Toolbo	x Devid	ce Cont	– ↓ trols v2	Save Al	l]orm1.vl	<b>b</b> * ]
► Pe	ointer ndingS	011708			Form1	

2. 點選 "Build" > "Build project1" 來編譯此專案 (project1),於 "Error List" 視窗中會顯示編譯結果。

🌀 project1 - Microsoft Visual St	ıdio			
Eile Edit View Project E	<u>duild D</u> ebug Data Forma			
i 🛐 💊 💕 🖽 🗕 🖪	🔛 Build project1 💊			
Toolbox 👻 🕂	Rebuild project1			
- All Device Controls v2	Deploy p Error List			
Reference BindingSource	Clean prc 📀 0 Errors 🔥 0 Warnings 🕕 0 Messages			
	Description			
	Build succeeded			

3. 编譯完成後,執行檔會存放在以下目錄中。

<您的 C# .net 專案資料夾> \bin\Release\ <project\_name>.exe

請將此執行檔 (例如: "project1.exe") 複製到 PAC 的 \System\_Disk\Win-GRAF\ 目錄下來執行。

### <u>注意:</u>

使用者可複製 C# .net 執行檔到其他目錄下執行,但請記得同時複製相關的 DLL 檔案,否則執 行會有錯誤。例如: 要在 \Micro\_SD\ 目錄下執行 "project1.exe",該目錄必須有以下 3 個檔案, 即 "project1.exe"、"UserShareNet.dll"、"Quicker.dll" 檔。

(可在光碟的 "\System\_disk\Win-GRAF\" 目錄下取得 "UserShareNet.dll" 與 "Quicker.dll" 檔案)

# 14.5 "UserShareNET.DLL" 內的函式說明

本節將針對 "UserShareNet.DLL" 内的函式 (Function) 來進行說明。

"UserShareNet.DLL"提供了許多函式,可用來讀/寫 Win-GRAF 專案裡的變數,以下分為幾類:

- 1. 讀/寫 Boolean
- 2. 讀/寫 8-bit 整數
- 3. 讀/寫 16-bit 整數
- 4. 讀/寫 32-bit 整數
- 5. 讀/寫 64-bit 整數
- 6. 讀/寫 32-bit 實數
- 7. 讀/寫 64-bit 實數

※請參考附錄 A 來查看 Win-GRAF 的變數資料型態與範圍

# 14.5.1 讀/寫 Boolean 的函式

Set\_BOOL

### 說明:

設定指定位址編號的 Win-GARF Boolean 變數值。

# 語法:

UserShare.Set\_BOOL(ushort iUserAddress, byte iStatus)

#### 參數:

iUserAddress: 指定變數的位址編號 (1~8192)

iStatus: 設定變數的狀態;例如: iStatus = 1 表示 "True", iStatus = 0 表示 "False"。

# 範例:

//設定位址 "1" 的 Win-GRAF 變數為 True.

UserShare.Set\_BOOL(Convert.ToUInt16(1), 1);

### 範例程式:

光碟 : \napdos\Win-GRAF\demo-project\CSharp.net\_2008\_demo\demo\_CSharp01



# 說明:

讀出指定位址編號的 Win-GARF Boolean 變數值。

# 語法:

UserShare.Get\_BOOL(ushort iUserAddress, out byte iStatus)

### 參數:

iUserAddress: 指定變數位址編號 (1~8192) iStatus: 取得變數的狀態; iStatus = 1 表示 "True", iStatus = 0 表示 "False"。

### 範例:

Byte iStatus=0; //取得位址編號 "1" 的變數狀態. UserShare.Get\_BOOL(Convert.ToUInt16(1),out iStatus);

# 範例程式:

光碟: \napdos\Win-GRAF\demo-project\CSharp.net\_2008\_demo\demo\_CSharp01

# 14.5.2 讀/寫 整數 的函式

### Set\_SINT Set\_INT Set\_DINT Set\_LINT

## 說明:

設定指定位址編號的 Win-GRAF 8-bit Integer, 16-bit Integer, 32-bit Integer 及 64-bit Integer 變數值。

### 語法:

UserShare.Set\_SINT(ushort iUserAddress , sbyte iStatus)

UserShare.Set\_INT(ushort iUserAddress , short iStatus)

UserShare.Set\_DINT(ushort iUserAddress, int iStatus)

UserShare.Set\_LINT(ushort iUserAddress, long iStatus)

# 參數:

iUserAddress: 指定變數的位址編號 (1~8192)

iStatus: 設定 8-bit Integer, 16-bit Integer, 32-bit Integer 及 64-bit Integer 值

### 範例:

//設定 32-bit integer 值 "1234567" 到位址編號 "1" 的變數. int temp1=1234567; UserShare.Set\_DINT(Convert.ToUInt16(1), temp );

//設定 16-bit Integer 值 "-1234" 到 Modbus 位址編號 "2" 的變數. short temp2= -1234; UserShare.Set\_INT(Convert.ToUInt16(2), temp2 );

//設定 64-bit Integer 值 "123456789012345" 到位址編號 "3" 的變數. long temp3=123456789012345; UserShare.Set\_LINT(Convert.ToUInt16(3), temp3 );

//設定 8-bit Integer "125" 值到位址編號 "4" 的變數. Sbyte temp4=125; UserShare.Set SINT(Convert.ToUInt16(4), temp4);

### 範例程式:

- 1. 讀/寫 類比 I/O: \napdos\Win-GRAF\demo-project\CSharp.net\_2008\_demo\demo\_CSharp02
- 2. 讀/寫 Internal long integer、Timer 及 Real (浮點數) 的值: \napdos\Win-GRAF\demo-project\CSharp.net\_2008\_demo\demo\_CSharp03

📕 Get\_SINT 📕 Get\_INT 📕 Get\_DINT 📕 Get\_LINT

### 說明:

讀出指定位址編號的 Win-GRAF 8-bit Integer, 16-bit Integer, 32-bit Integer 及 64-bit Integer 變數值。

### 語法:

UserShare.Get\_SINT(ushort iUserAddress, out sbyte iStatus) UserShare.Get\_INT(ushort iUserAddress, out short iStatus) UserShare.Get\_DINT(ushort iUserAddress, out int iStatus)

UserShare.Get\_LINT(ushort iUserAddress, out long iStatus)

# 參數:

iUserAddress: 指定變數的位址編號 (1~8192)

iStatus: 取得 8-bit Integer, 16-bit Integer, 32-bit Integer 及 64-bit Integer 值

# 範例:

Int64 Dlong\_val; Int16 short\_val; Int32 long\_val; sbyte sbyte\_val;

//取得位址編號 "7"的 64-bit 整數的變數值. UserShare.Get\_LINT(Convert.ToUInt16(7),out Dlong\_val);

//取得位址編號 "8"的 32-bit 整數的變數值. UserShare.Get\_DINT(Convert.ToUInt16(8),out long\_val);

//取得位址編號 "9" 的 16-bit 整數的變數值. UserShare.Get\_INT(Convert.ToUInt16(9),out short\_val);

//取得位址編號 "10" 的 8-bit 整數的變數值. UserShare.Get\_SINT(Convert.ToUInt16(9),out sbyte\_val)

# 範例程式:

- . 讀/寫 類比 I/O: \napdos\Win-GRAF\demo-project\CSharp.net\_2008\_demo\demo\_CSharp02
- 2. 讀/寫 Internal long integer 、 Timer 及 Real (浮點數) 的值: \napdos\Win-GRAF\demo-project\CSharp.net\_2008\_demo\demo\_CSharp03

# 14.5.3 讀/寫 實數 的函式

Get\_REAL Get\_LREAL

### 說明:

讀出指定位址編號的 Win-GRAF 32-bit Real、64-bit Real 變數值。

### 語法:

UserShare. Get\_REAL (System.UInt16 iUserAddress, out float iStatus)

UserShare. Get\_LREAL(ByVal iUserAddress As System.UInt16 , out Double iStatus)

# 參數:

iUserAddress: 指定變數的位址編號 (1~8192) iStatus: 取得浮點數數值

# 範例:

float float\_val; double double\_val;

// 取得位址編號 "7"的 double 變數值. UserShare.Get\_LREAL(Convert.ToUInt16(7),out double\_val);

//取得位址編號 "8"的 float 變數值. UserShare.Get\_REAL(Convert.ToUInt16(8),out float\_val);

### 範例程式:

- 1. 讀/寫 類比 I/O: \napdos\Win-GRAF\demo-project\CSharp.net\_2008\_demo\demo\_CSharp02
- 2. 讀/寫 Internal long integer、Timer 及 Real (浮點數)的值: \napdos\Win-GRAF\demo-project\CSharp.net 2008 demo\demo demo CSharp03

Set\_REAL Set\_LREAL

### 說明:

寫入指定的位址編號的 Win-GRAF 32-bit Real、64-bit Real 變數值。

### 語法:

UserShare. Set\_REAL ( ushort iUserAddress , float iStatus )

UserShare. Set\_LREAL( ushort *iUserAddress* , Double *iStatus*)

# 參數:

iUserAddress: 指定變數的位址編號 (1~8192)

iStatus: 設定浮點數數值

# 範例:

// 寫入 11234.234567 到位址編號 "7" 的變數 UserShare.Set\_LREAL(Convert.ToUInt16(7),Convert.ToDouble(11234.234567));

//寫入 123.12 到位址編號 "8" 的變數 UserShare.Set\_REAL(Convert.ToUInt16(8), Convert.ToSingle (123.12));

### 範例程式:

- 1. 讀/寫 類比 I/O: \napdos\Win-GRAF\demo-project\CSharp.net\_2008\_demo\demo\_CSharp02
- 2. 讀/寫 internal long integer、Timer 及 Real (浮點數) 的值: \napdos\Win-GRAF\demo-project\CSharp.net\_2008\_demo\demo\_CSharp03

# 14.5.4 讀/寫 字串的函式

### Set\_STRING

## 說明:

寫入指定位址編號的 Win-GRAF 字串值。

### 語法:

UserShare.Set\_STRING (ushort addr , <u>Byte</u> [] msg)

# 參數:

- addr: 指定變數的位址編號 (1~1024)
- msg[]: 設定 Win-GRAF String 變數值。

### 範例:

String str\_val; Byte[] msg;

```
// 寫入位址編號 "7"的 String 變數值.
msg= unicode_to_byte_array(str_val);
UserShare.Set_STRING(Convert.ToUInt16(7),msg);
```

//String 轉成 byte 陣列

```
private byte[] unicode_to_byte_array(string msg)
```

```
{
```

```
byte[] tmpbuf;
if (msg.Length > 255)
return null;
tmpbuf = Encoding.GetEncoding("UTF-8").GetBytes(msg);
return tmpbuf;
```

```
}
```

### 範例程式:

光碟:

 . 讀/寫 STRING 的值: \napdos\Win-GRAF\demo-project\CSharp.net\_2008\_demo\demo\_CSharp04

### Get\_STRING

### 說明:

讀取指定位址編號的 Win-GRAF 字串變數值。

# 語法:

### UserShare.Set\_STRING (ushort addr , Byte [] msg)

### 參數:

addr : 指定變數的位址編號 (1~1024) msg[] : 設定 Win-GRAF String 變數值。

# 範例:

String str\_val= "Hello World"; Byte[] msg;

// 設定位址編號 "7"的 String 變數值. UserShare.Get\_STRING(Convert.ToUInt16(7),msg); str\_val= byte\_array\_to\_unicode(msg);

```
//byte 陣列轉字串
private string byte_array_to_unicode(byte[] buf)
{
    string tmpmsg;
    if (buf.Length > 255)
        return null;
    tmpmsg = Encoding.GetEncoding("UTF-8").GetString(buf, 0, buf.Length);
    return tmpmsg;
    }
```

### 範例程式:

光碟:

1. 讀/寫 STRING 的值: \napdos\Win-GRAF\demo-project\CSharp.net\_2008\_demo\demo\_CSharp04

# 14.5.5 如何讓 C # 程式讀取 Win-GRAF 字串變數?

.NET 程式如果要寫入字串變數,必須根據目前使用的語言編碼 (例如: UTF-8) 轉換成 byte 陣列才能 寫入,若是讀出字串變數的陣列內容,則需要根據語言編碼轉換成字串。以下提供 C# 程式轉換的 範例 (編碼為 UTF-8):

# //String 轉成 byte 陣列

```
private byte[] unicode_to_byte_array(string msg)
```

```
{
    byte[] tmpbuf;
    if (msg.Length > 255)
        return null;
    tmpbuf = Encoding.GetEncoding("UTF-8").GetBytes(msg);
    return tmpbuf;
}
```

# //byte 陣列轉字串

```
private string byte_array_to_unicode(byte[] buf)
{
    string tmpmsg;
    if (buf.Length > 255)
        return null;
    tmpmsg = Encoding.GetEncoding("UTF-8").GetString(buf, 0, buf.Length);
    return tmpmsg;
}
```