LabVIEW

作者；曾智勇, chihyungtseng@hotmail.com

LabVIEW簡介

- 為什麼需要寫程式：
  - 為了讓工作內容能因程式而**提昇工作效率**，進而**提高個人附加價值**
  - 客戶需求不斷昇高，難度逐漸提高，而所給予的完成時間卻不斷縮短。因此需要藉由程式來幫你減輕負擔、加快任務執行進度。

- 為什麼要學LabVIEW：
  - 非資工背景人員都能容易上手的程式語言
  - 能快速開發及驗證想法的程式語言
  - 能夠方便控制儀器的程式語言
  - 能夠方便擷取感測器 (溫度、壓力、電壓等) 訊號的程式語言
  - 專注於重點技術而非開發工具 (程式撰寫)

- 麵粉→三明治(傳統程式語言 → 應用程式)
- 吐司→三明治(圖形化程式語言 → 應用程式)

作者；曾智勇, chihyungtseng@hotmail.com

LabVIEW簡介

麵粉→三明治

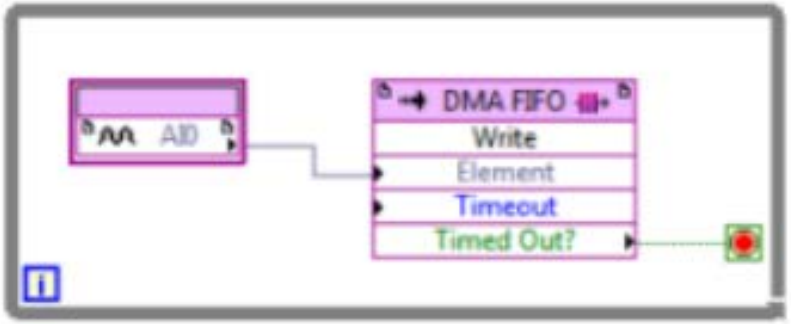
吐司→三明治

I/O With Direct Memory Access Transfer

VHDL

66 pages, ~4,000 lines

Graphical HLS Approach



作者；曾智勇, chihyungtseng@hotmail.com

LabVIEW簡介

作者；曾智勇, chihyungtseng@hotmail.com

LabVIEW簡介

- LabVIEW 的全名是實驗室虛擬儀控平台(Laboratory Virtual Instrument Engineering Workbench)，最初 LabVIEW 創建的宗旨是為了提供工程師及科學研究人員適合使用的程式語言，重點並不是在於要寫出多艱深的程式，而是要能夠快速驗證目前的想法。

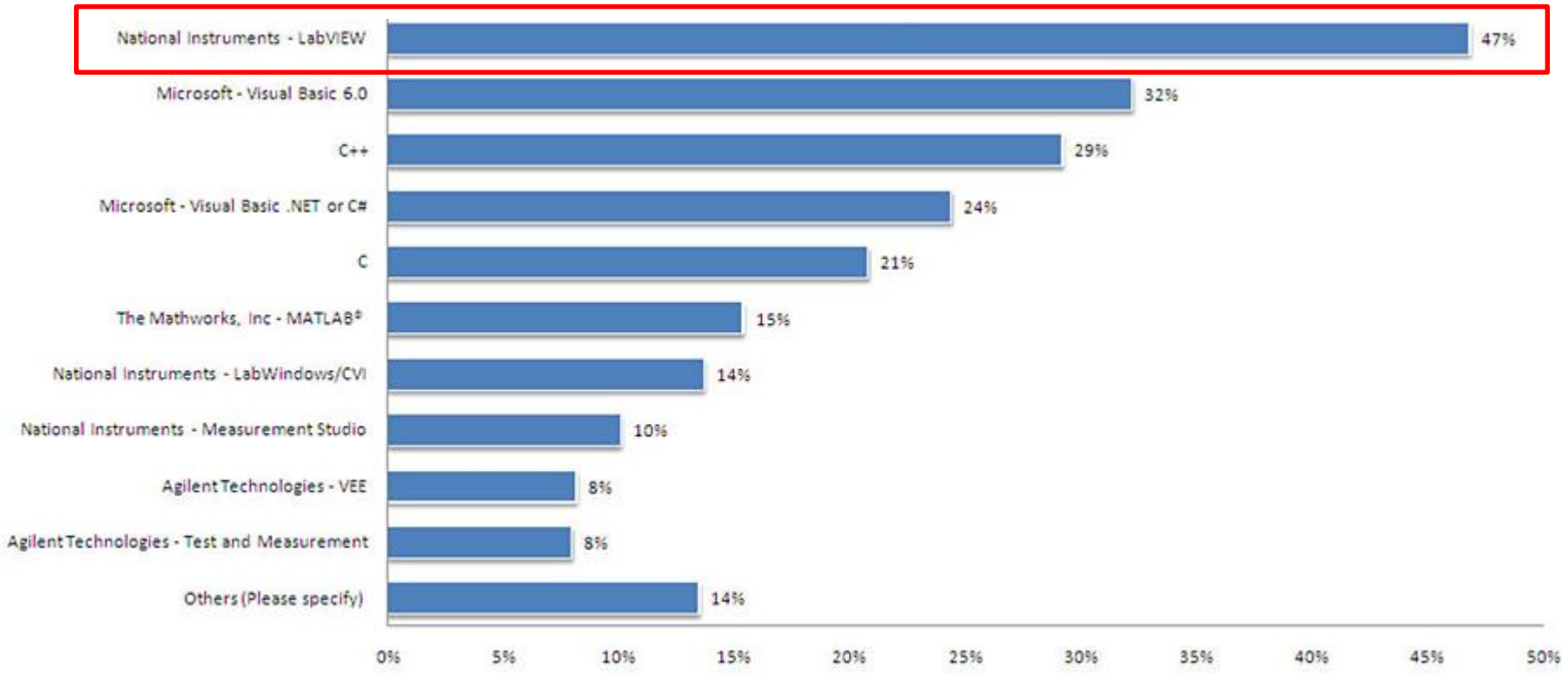
- 工程師和科學研究人員絕大多數都習慣使用圖形化的方式來思考，像是繪製及分析電路圖，或是繪製系統流程圖等，因此基於圖形化、訊號流概念，開發出LabVIEW程式語言。

- 圖形比文字來的容易記憶及理解，這也是目前 LabVIEW 大量被工程師與科學研究人員使用的原因，因為他採用了圖形化的方式來撰寫程式，具有完整的函式/元件，涵蓋各領域醫學、科學、數學、統計、訊號處理、通訊等，廣泛應用於不同領域、容易協同作業，快速開發提昇了工作效率。

作者；曾智勇,chihyungtseng@hotmail.com

LabVIEW簡介

- 針對於**量測及自動化領域**中，所使用程式語言的調查結果 ：



作者；曾智勇,chihyungtseng@hotmail.com

LabVIEW簡介

- 美國SpaceX公司，使用LabVIEW監控火箭發射程序

作者；曾智勇,chihyungtseng@hotmail.com

LabVIEW簡介

- 台灣廠商相關職缺：

仁寶	Labview軟體開發主管	1. 2年自動控制、程式設計、機電整合工作經驗 2. 具實際撰寫Labview完成專案之經驗 3. 持有 NI LabVIEW 專業認證資格尤佳 4. 有產線導入工作經驗 或 Windows Dll開發經驗 者尤佳
光寶	自動化檢測工程師	1.開發光學檢測設備，檢查產品外觀及尺寸,功能檢測及訊號處理 2.使用LABVIEW軟體進行光學自動化設備開發 3.自動化電路設計程式撰寫
日月光	測試工程師	1. IC CP/FT 軟、硬體測試開發、良率分析、維護 2.對光學相關軟體研發有興趣 3.新產品導入及流程驗證 4.具93K、J750、V50、D10測試經驗尤佳 5.擅長Labview
致茂	軟體應用工程師	1.客製化軟體應用設計與開發(LabVIEW)，及相關自動化技術研究 2.整合資源，提供客戶軟體測試解決方案。
捷安特	機電創新研發工程師	1. 電動載具驅動單元設計 2. 先進馬達技術可行性評估與導入、磁路構造設計開發 3/3. 無刷馬達控制技術開發及程式設計 4. LabView軟體進行產品開發、測試與驗證
富士康	自動化軟體工程師	1. C/C++ 或Labview 编程能力，实现RS-232,USB,TCP/IP通迅功能。 2. 熟悉一种UML设计工具，有一定的架构设计能力； 3. 熟悉数电、模电，有电子自动化测试经验者优先 4. 至少熟练掌握以下一种数据库：SQL Server,Oracle,DB2。
台達電	自動化電控工程師	1. 通过VB，VC，Labview等编程进行软件设计与测试程序开发 2. 可通过计算机软件编程去控制测试仪器和接收测试仪器传来的数据

作者；曾智勇,chihyungtseng@hotmail.com

LabVIEW簡介

- 監測軟體市場行情：

項次	品　名	商 牌/規 格	數量	單價(未稅)	總價(未稅)
1	太陽能壽命測試系統圖表顯示功能	圖表顯示主架構：			
		1. 即時運算分析系統 2. 提供報表列印及圖檔列印功能 3. 查看指標所顯示的所有數值 4. 可選擇報表要顯示的曲線 5. 提供報表 Zoom In 及 Zoom Out 功能 6. 選擇 X 軸的顯示選項 7. 選擇 Y 軸的顯示選項 以下空白 交貨時間：45 工作天	1	180,000	180,000
				小計 (未稅)：	180,000
				5%稅	9, 000
				合計：	189, 000

實際：
不用1天

提供程式一年免費保固服務
以 LabView 8.5 以上之版本撰寫，以 LabVIEW 圖控語言撰寫之可安裝執行檔

作者；曾智勇,chihyungtseng@hotmail.com

LabVIEW簡介

- NI LabVIEW 建議學習順序

作者；曾智勇, chihyungtseng@hotmail.com

LabVIEW簡介

- NI LabVIEW 認證及其人數 **(2007年 ~ 2017年Q1)**
  http://www.ni.com/services/certification.htm

紅框為本講義涵蓋教學範圍

作者；曾智勇, chihyungtseng@hotmail.com

LabVIEW簡介

- NI LabVIEW 教育訓練參考價格 (2016年官方報價單)

Description	Qty.	Unit Price	Net Price
Object-Oriented Design and Programming in LabVIEW (Loc) 2 Days 報名10/26-27台北場課程	1	31,200.00	31,200.00
Sub-Total:			**TWD 31,200.00**
VAT:			**TWD 1,560.00**
Total:			**TWD 32,760.00**

作者；曾智勇, chihyungtseng@hotmail.com

NI LabVIEW 教育訓練標準課程 (打勾部分為本講義涵蓋教學範圍)

	中文課程名稱	英文課程名稱	價格	課程時間	課程簡介	關鍵學習主題	必備條件
✓	LabVIEW 核心課程 1	LabVIEW Core 1	NT$39000 (未稅)	三天 (9:00am-5:00pm)	LabVIEW 核心課程 1 (LabVIEW Core 1) 為所有 LabVIEW 課程所必備。LabVIEW 核心課程 1 將透過實機操作的方式，介紹 LabVIEW 的環境、相關功能、資料流程式設計，與常見的 LabVIEW 架構。學員透過狀態機器設計模式而建立應用，可學習開發測試與量測、資料擷取、儀器控制、資料紀錄，還有量測分析的應用。	人機界面(Front Panel), 程式方塊圖(Block Diagram), Dataflow Programming, 迴圈(Loops), Case Structure, DAQmx API	熟悉微軟視窗軟體與撰寫演算法的經驗 (流程圖、程式圖)
✓	LabVIEW 核心課程 2	LabVIEW Core 2	NT$26000 (未稅)	二天 (9:00am-5:00pm)	LabVIEW 核心課程 2 (LabVIEW Core 2) 為 LabVIEW 核心課程 1 (LabVIEW Core 1) 的延伸，課程主題涵蓋最佳化應用效能的技術，如最佳化現有程式碼、使用檔案 I/O 功能、資料管理原則、事件程式設計，與錯誤處理實作。此教育訓練將銜接應用所需的 LabVIEW 功能，並可迅速入門應用開發。	事件規劃(Event Programming), 錯誤處理(Error Handling), File I/O, 改善現有 VI, 編寫執行檔案(EXE)	具備 Microsoft Windows 與 LabVIEW 核心課程 1 (Core 1) 等級的相關經驗
✓	LabVIEW 核心課程 3	LabVIEW Core 3	NT$39000 (未稅)	三天 (9:00am-5:00pm)	LabVIEW 核心課程 3 (LabVIEW Core 3) 將可實際練習設計、開發、測試，並佈署 LabVIEW 應用。學員將可了解常見的應用開發技術，如階層式 VI 開發、事件導向的架構、設計合適的使用者介面、錯誤處理，與有效的文件記錄方式。	使用者介面設計, 擬定VI測試, 評估並強化VI效能, 階層式 VI 開發, 有效的文件記錄	LabVIEW 核心課程 1 與 2 (LabVIEW Core 1, 2) 或同等經驗
✓	LabVIEW 網路連結	LabVIEW Connectivity	NT$26000 (未稅)	二天 (9:00am-5:00pm)	LabVIEW 網路連結 (LabVIEW Connectivity) 課程，是以 LabVIEW 核心課程 3 (LabVIEW Core 3) 為架構。可了解整合式系統的元件，並針對自己的應用建置網路連線技術。同時可透過如 DLL、ActiveX，與網際網路的技術，擴充應用功能並縮短開發時間，以妥善發揮其他應用的功能。	VI Server, 物件呼叫, 資料庫連結, 網路傳遞資料	LabVIEW 核心課程 1 與 2 (LabVIEW Core 1, 2) 或同等經驗
	LabVIEW 優化效能	LabVIEW Performance	NT$26000 (未稅)	二天 (9:00am-5:00pm)	LabVIEW 優化效能 (LabVIEW Performance) 課程，將可撰寫更優質的程式碼。此實機操作課程將協助學員找出 LabVIEW 程式碼的問題，並進一步提升效能。另可了解應如何設計自己的程式碼，避開導致效能低落的陷阱。	記憶體、執行緒，與 I/O 資源管理；VI Analyzer Toolkit, Desktop Execution Trace Toolkit	LabVIEW 核心課程 1 與 2 (LabVIEW Core 1, 2) 或同等經驗
	LabVIEW專案管理與設計	Managing Software in LabVIEW	NT$26000 (未稅)	二天 (9:00am-5:00pm)	LabVIEW 專案管理與設計課程 (Managing Software Engineering in LabVIEW)，可協助學員培養相關技術，於單一或多位開發者的環境中，有效管理並建立大型 LabVIEW 應用。此課程將針對管理大型的團隊應用開發專案，說明從應用規格到佈署作業的常見實例。於專案中整合這些應用開發實作之後，即可改良開發程序並最佳化系統資源，以大幅減少開發成本與時間。	擷取可追蹤的資訊, Requirements Gateway, Unit Test Framework Toolkit, 自動化 LabVIEW VI 的單元測試	LabVIEW 核心課程 3 (LabVIEW Core) 或相關經驗
	LabVIEW 高階架構	Advanced Architectures in LabVIEW	NT$39000 (未稅)	三天 (9:00am-5:00pm)	LabVIEW 高階架構 (Advanced Architectures for LabVIEW) 課程中，學員可進行討論並個別/共同學習應用的架構方式，並設計相關元件以支援該架構。此外，並可透過 NI LabVIEW 的高階設計模式 (如 Functional global variables、Plug-ins、X controls，與 subpanels)，獲得更多設計經驗。訓練課程將提供實作範例，並由講師指定高階系統需求，讓學員草擬系統架構並設計數項元件。	API設計, Control Design and Simulation Module, 良好架構的的錯誤處理系統(Error Handling)	LabVIEW 核心課程 3 (LabVIEW Core) 或相關經驗
✓	資料擷取與訊號處理	Data Acquisition and Signal Conditioning	NT$39000 (未稅)	三天 (9:00am-5:00pm)	本課程將教您如何利用 LabVIEW 搭配資料擷取卡 (DAQ) 與cDAQ 進行資料擷取。本課程包含資料擷取與訊號處理的基本原理、硬體與軟體架構與撰寫程式方法。能在課堂中學習到類比輸入、啟動、訊號處理、類比輸出、數位輸出 / 輸入和計數等功能。以 PC-based 作資料擷取和訊號處理。	DAQmx API, Analog Input/output, DIO, Signal Conditioning, Synchronization	LabVIEW 核心課程 1 與 2 (LabVIEW Core 1, 2) 或同等經驗
✓	儀器控制課程	LabVIEW Instrument Control	NT$26000 (未稅)	二天 (9:00am-5:00pm)	LabVIEW 儀器控制課程教授如何發展 LabVIEW 應用於裝置和控制 GPIB 系列和 PXI 儀器。學員使用信號介面可以裝置和控制 GPIB 系列、VXI 儀器的虛擬軟體 (VISA) 和具有轉換功能的虛擬儀器伺服器 (IVI)。	LabVIEW Plug & Play instrument drivers, VISA API, NI Spy, Instrument Driver	LabVIEW 核心課程 1 與 2 (LabVIEW Core 1, 2) 或同等經驗
	機器視覺與影像處理	Machine Vision and Image Processing	NT$26000 (未稅)	二天 (9:00am-5:00pm)	本課程包括機器視覺的基本原理、機器視覺的構成要素和設置適當相機、鏡頭以及照明設備等多種資源。您可運用視覺軟硬體擷取和校正影像。	Vision Acquisition SW, Vision Developmonet Module	LabVIEW 核心課程 1 與 2 (LabVIEW Core 1, 2) 或同等經驗
	馬達運動控制	LabVIEW Motion Control	NT$26000 (未稅)	二天 (9:00am-5:00pm)	配合使用 LabVIEW，馬達運動控制課程內容包含基礎運用和 NI 運動控制配備的運用。在本課程中，學員學習如何設定馬達運動控制系統、開發基礎的運動軌跡，並設計迴饋控制迴路，了解基本的運動控制方式並運用這些方式來設計運動控制的解決方案。	NI Motion, Motion Assistant, 馬達系統, 回饋, 運動控制	LabVIEW 核心課程 1 與 2 (LabVIEW Core 1, 2) 或同等經驗
	即時監控課程	LabVIEW Real-Time	NT$39000 (未稅)	三天 (9:00am-5:00pm)	LabVIEW即時課程提供實務訓練，以開發穩定、可靠及決定性的測量及控制系統。在課程結束時，你就能夠佈署一個可以穩定持續執行的LabVIEW Real-Time系統。本課程是學習穩定的系統技術、即時程式設計技術、以及節省時間的開發秘訣的最快速途徑。	Real-time System, Real-Time Architecture, Network Communication,	LabVIEW 核心課程 1 與 2 (LabVIEW Core 1, 2) 或同等經驗
	LabVIEW FPGA	LabVIEW FPGA	NT$26000 (未稅)	三天 (9:00am-5:00pm)	LabVIEW FPGA Module 課程可讓您使用 LabVIEW FPGA Module 與 NI 可重設 I/O (RIO) 硬體，進行設計、除錯，並建置有效率的應用。並可學習將 VI 佈署制 RIO 系統的方法，加以控系統作業的時脈、同步化，與優先性。	FPGA Module, Data sharing, SCTL, DMA Data Transfer, Modular Programming	LabVIEW 核心課程 1 與 2 (LabVIEW Core 1, 2) 或同等經驗

作者；曾智勇,chihyungtseng@hotmail.com

課程目標

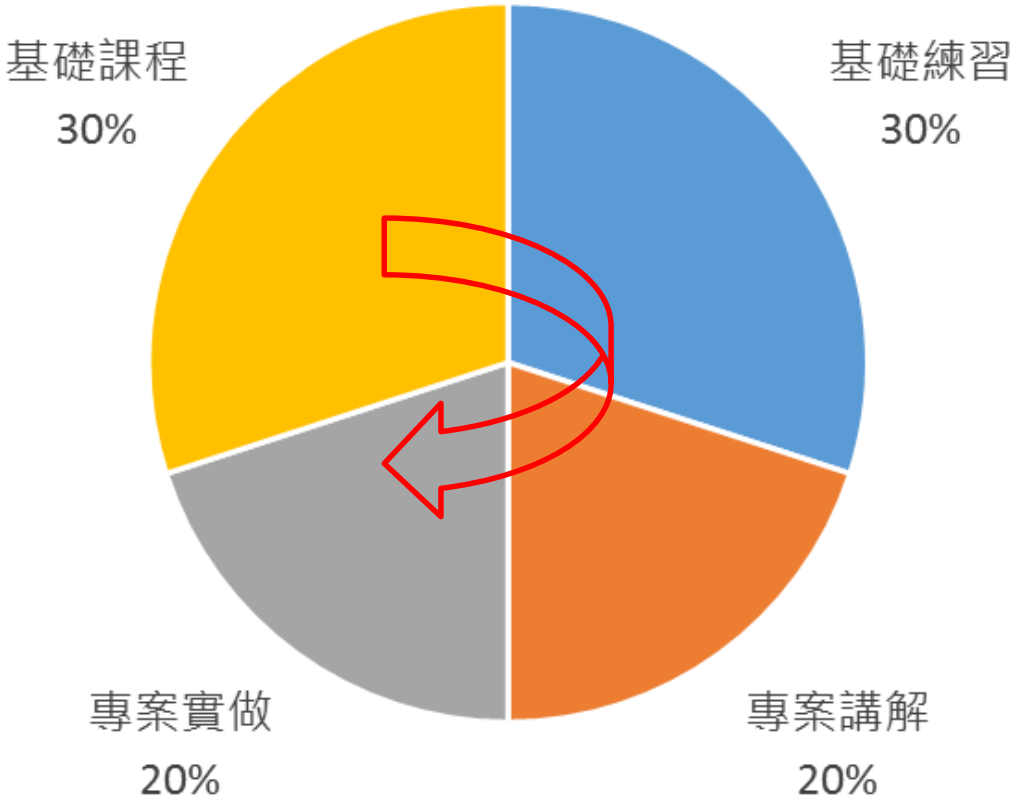
- 完成本課程您可以學習到的能力：

A. 獨立開發-單人小型專案
B. 協同開發-多人中型專案
C. 專案維護-CLD層級的 (< 100個VIs)
D. 儀器控制-RS232/485、USB、NI DAQ、TCP/IP
E. 整合應用-.NET物件、ActiveX物件、動態聯結檔
F. 遠端監控-Web Publishing、Web Service

作者；曾智勇,chihyungtseng@hotmail.com

課程時間

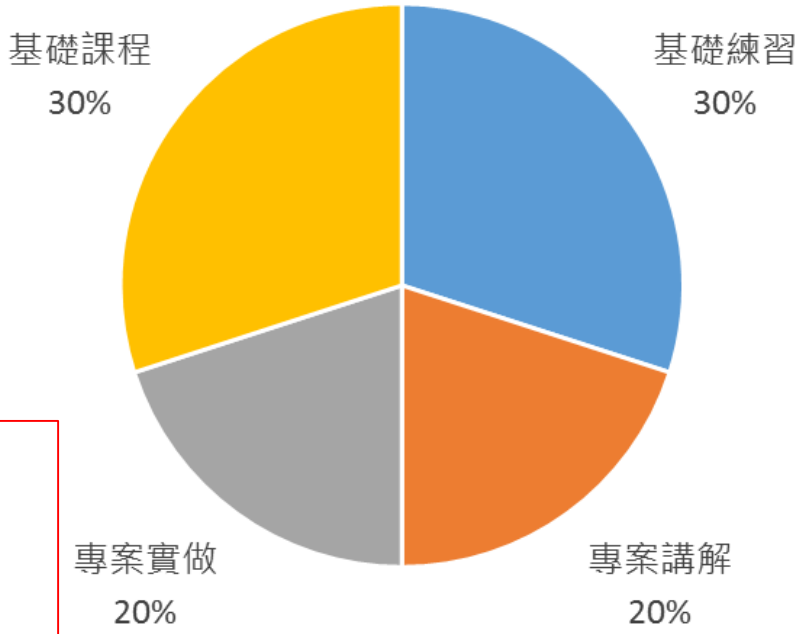
- 課程總時數 60小時~
- 進度分配：



作者；曾智勇, chihyungtseng@hotmail.com

課程內容

基礎操作、程式架構
數據擷取、儀器控制
資料紀錄、檔案存取
事件管理.....



時間管理、 事件管理
ini設定檔存取、
csv檔存取 、程序控制
綜合練習.....

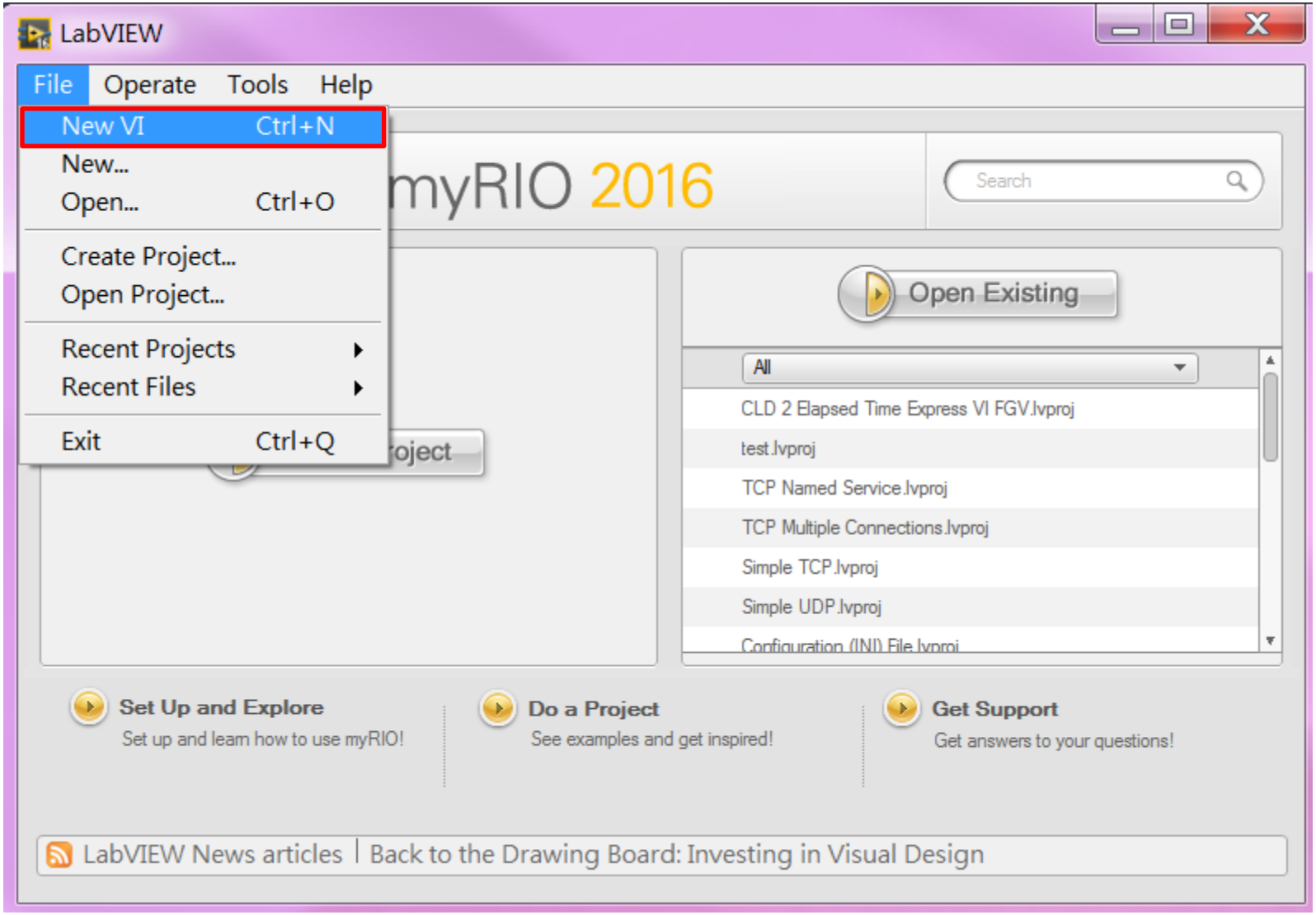
擇二：
Subpanel 程式開發
多台電腦連線報到系統
餐飲店點餐
變頻風機控制
其他.....

Subpanel架構
XControl架構
曲線圖表模組
.....

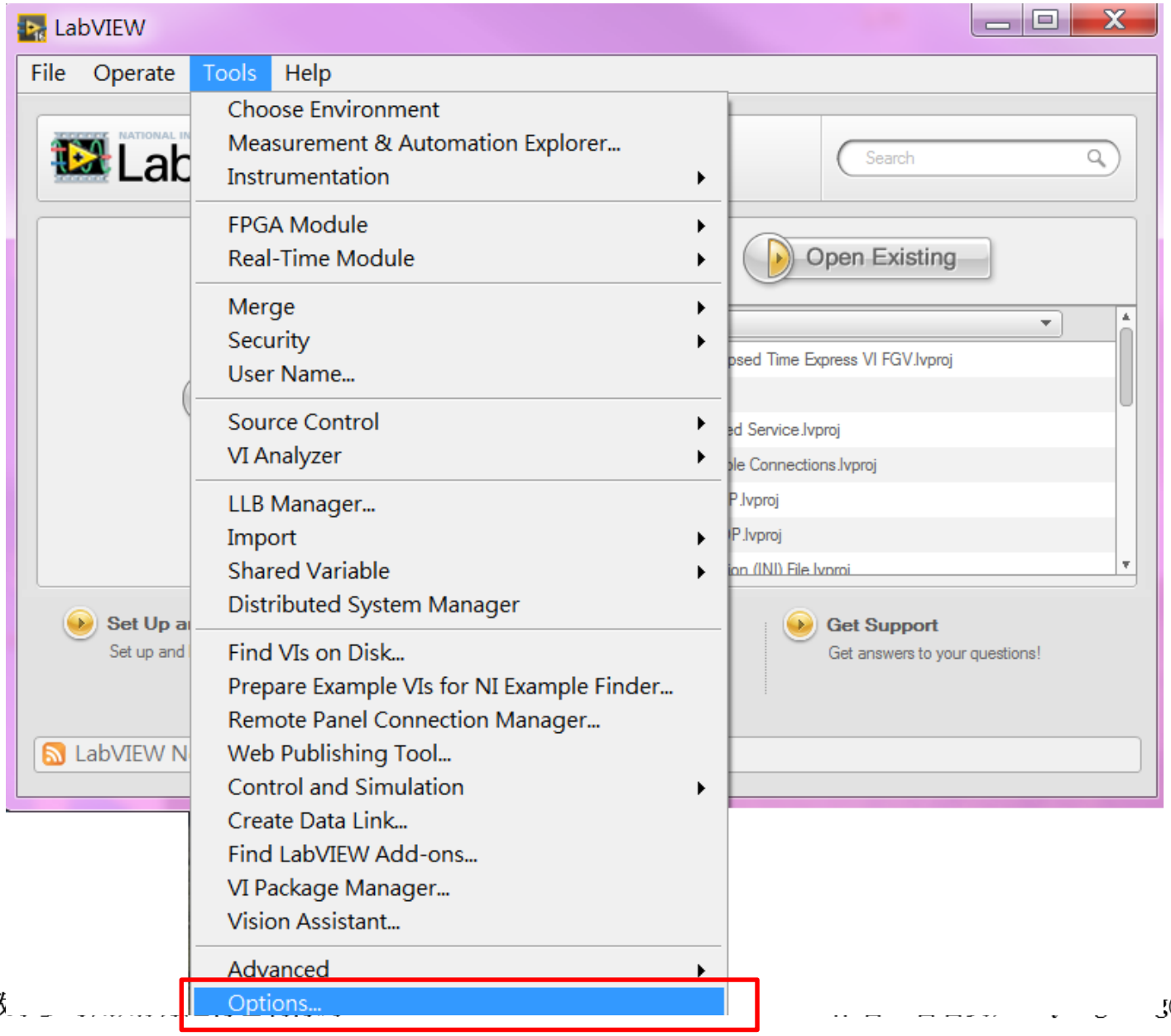
作者；曾智勇,chihyungtseng@hotmail.com

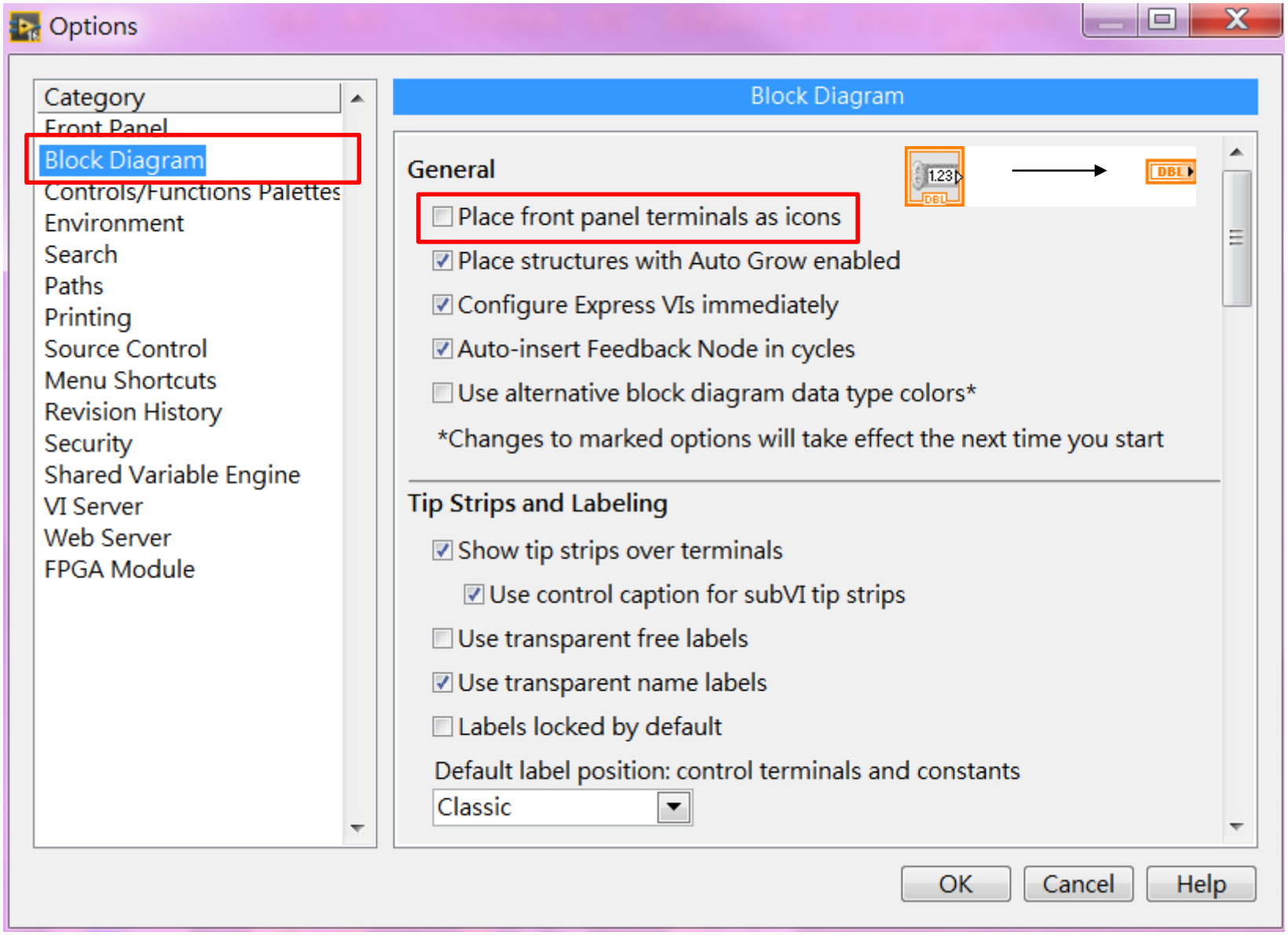
環境設定

作者；曾智勇, chihyungtseng@hotmail.com

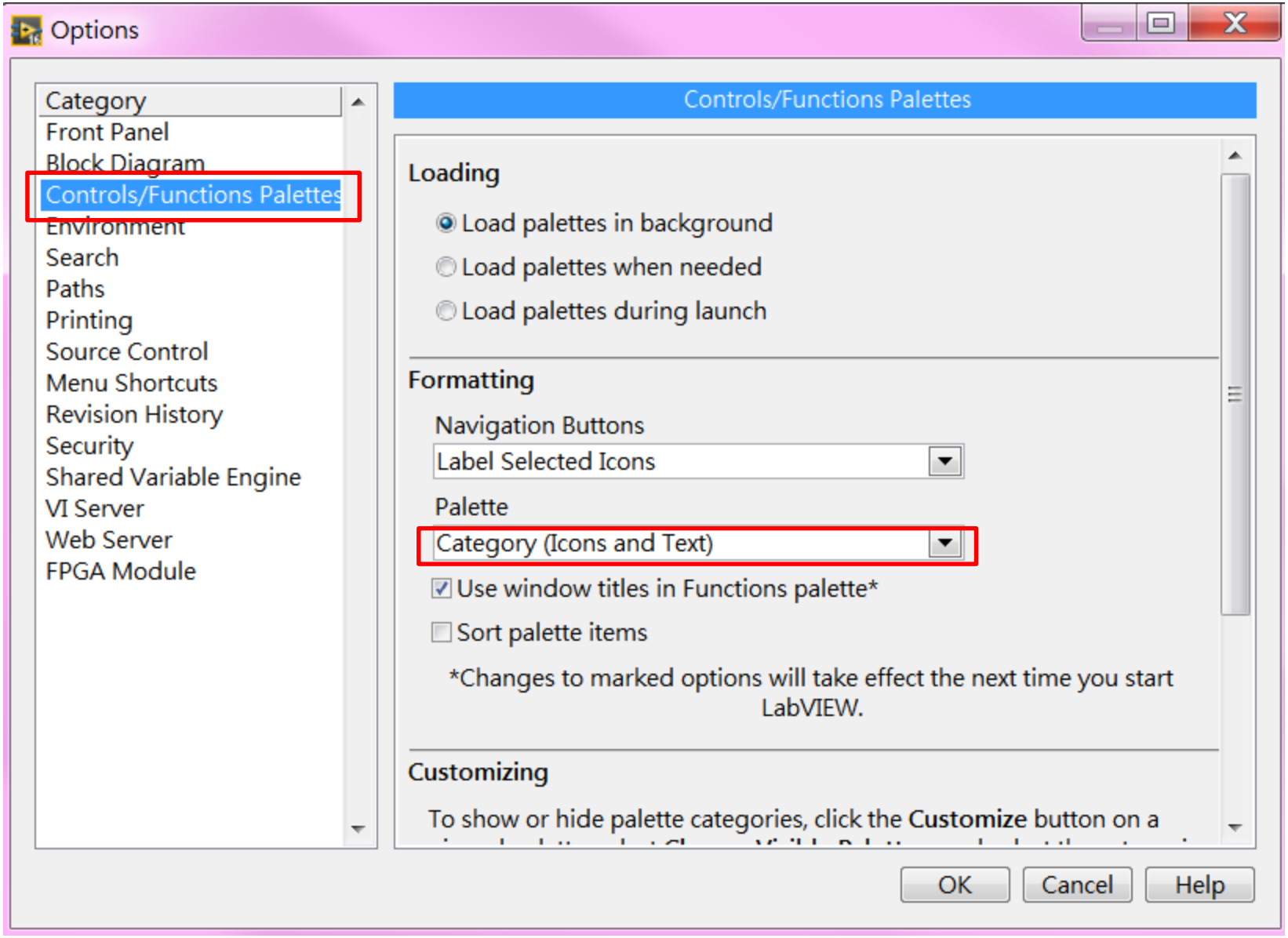


作者；曾智勇,chihyungtseng@hotmail.com

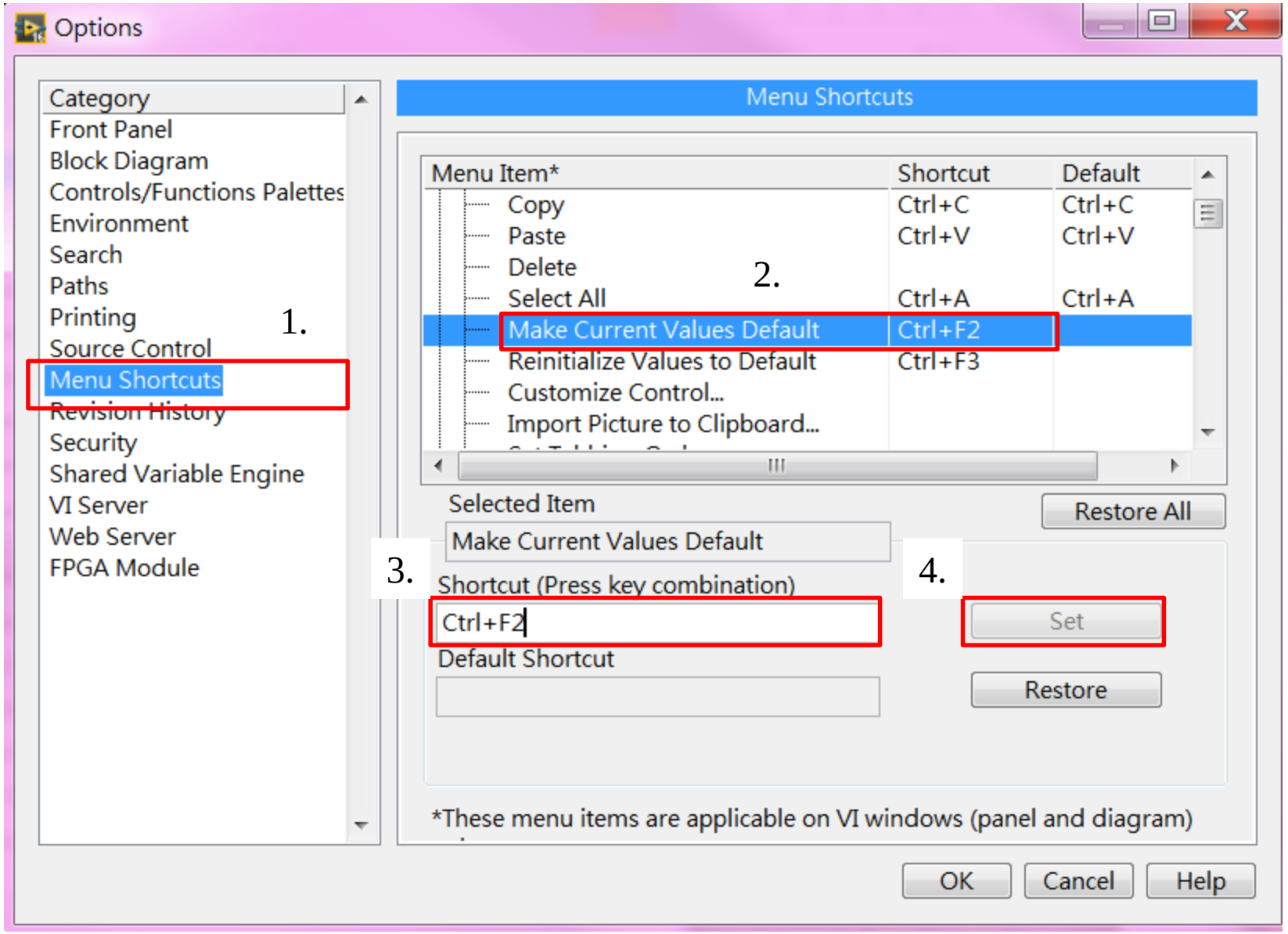




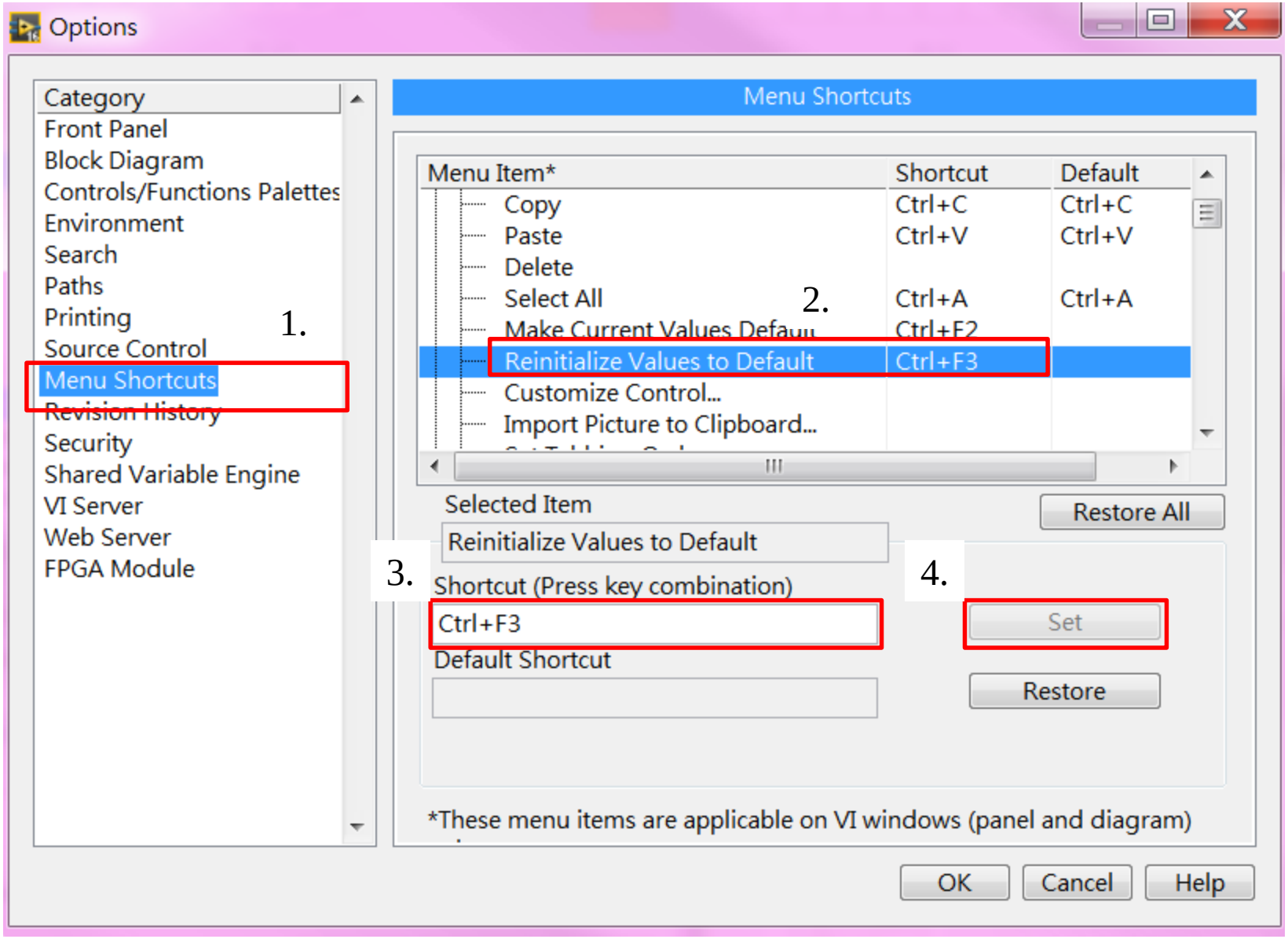
作者；曾智勇,chihyungtseng@hotmail.com



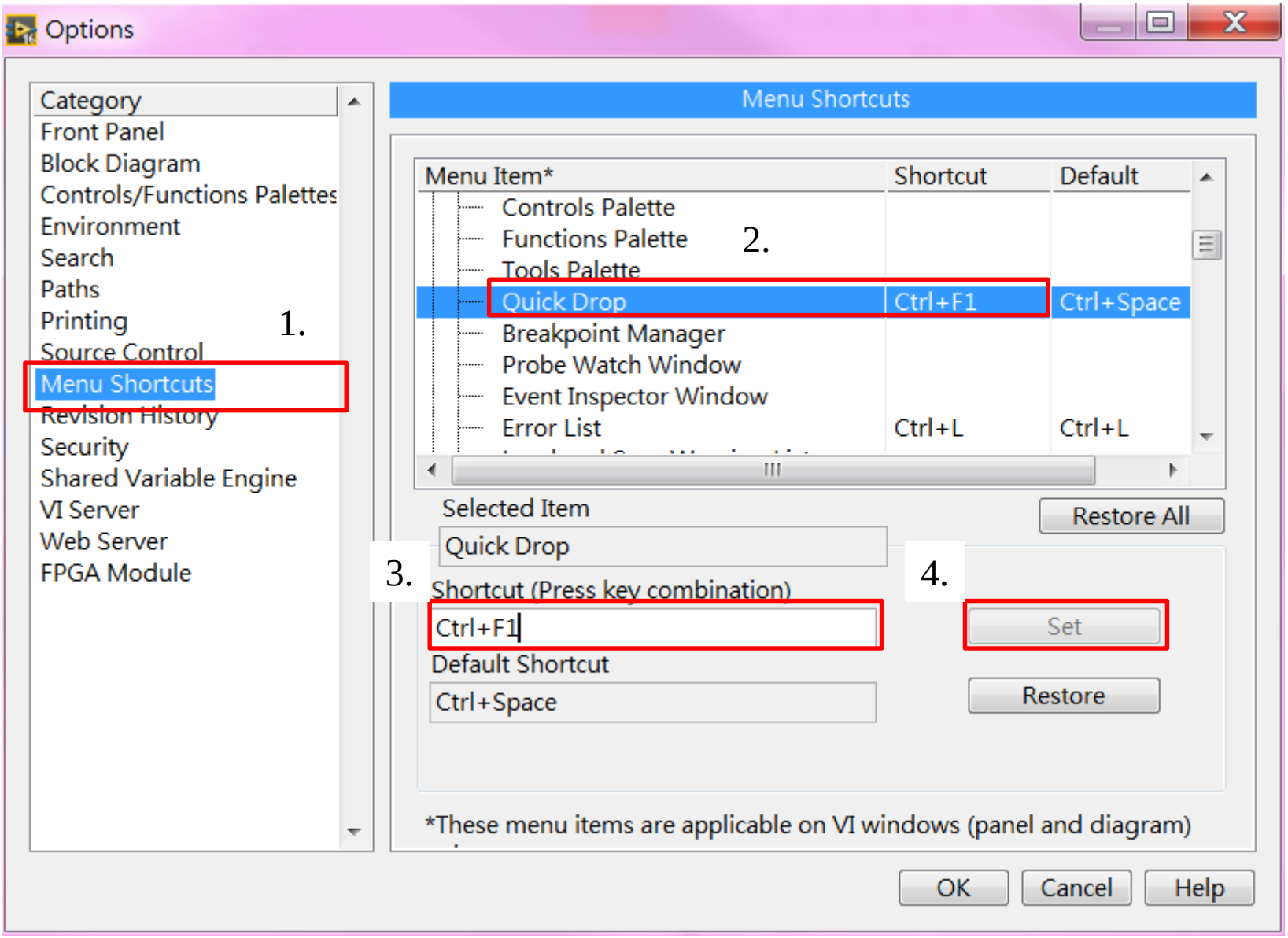
作者；曾智勇,chihyungtseng@hotmail.com



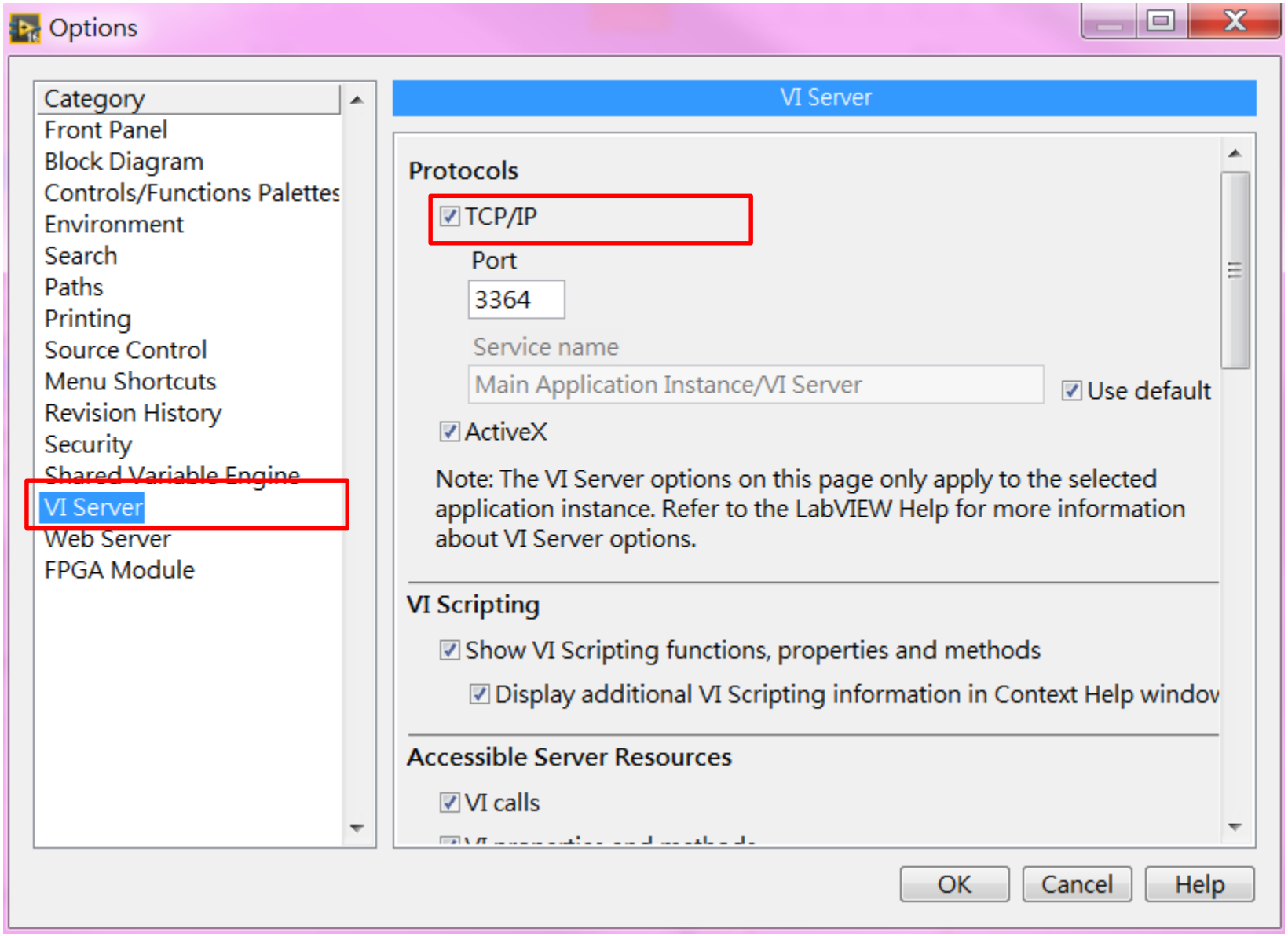
作者；曾智勇,chihyungtseng@hotmail.com



作者；曾智勇,chihyungtseng@hotmail.com

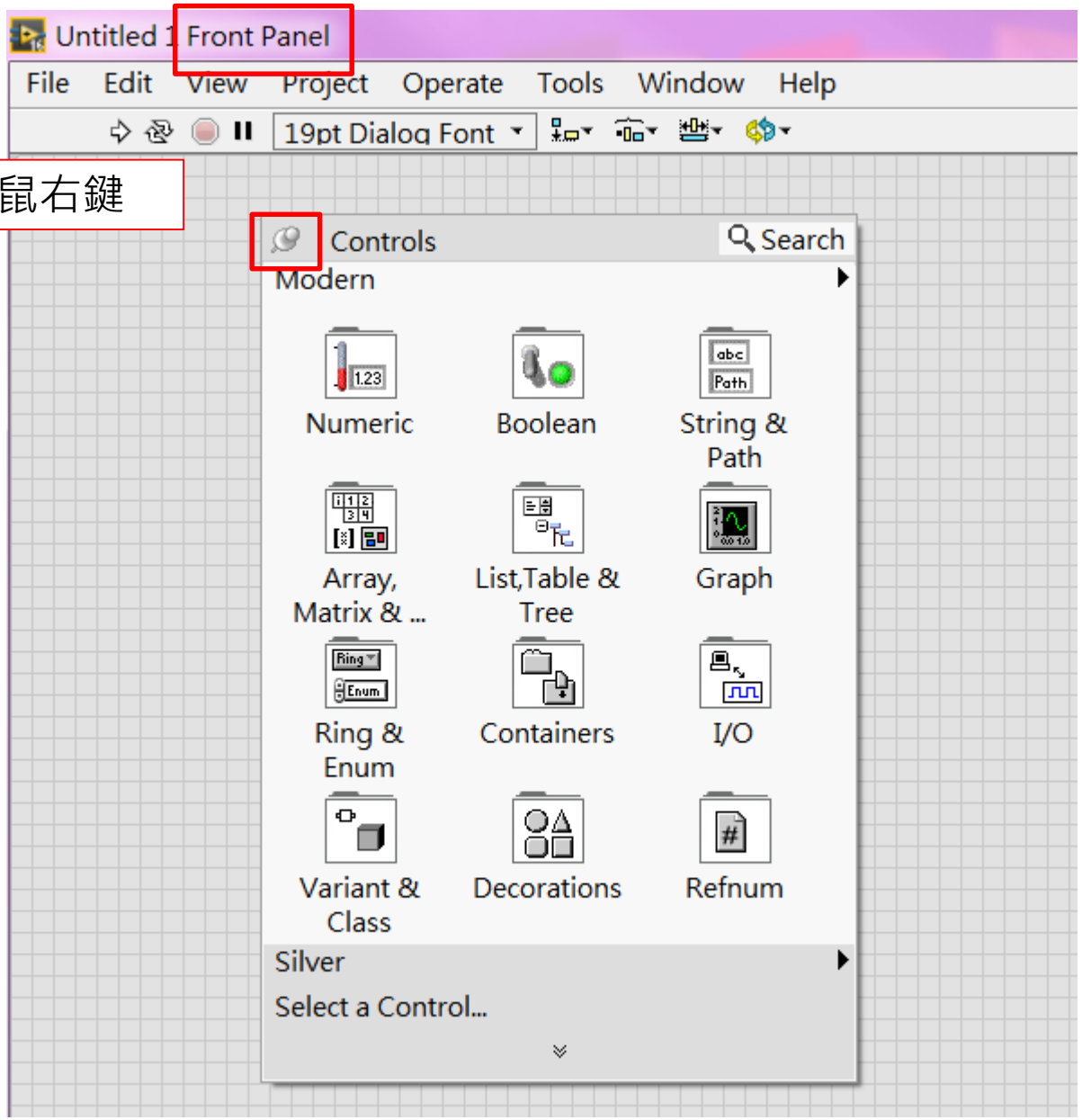


作者；曾智勇,chihyungtseng@hotmail.com

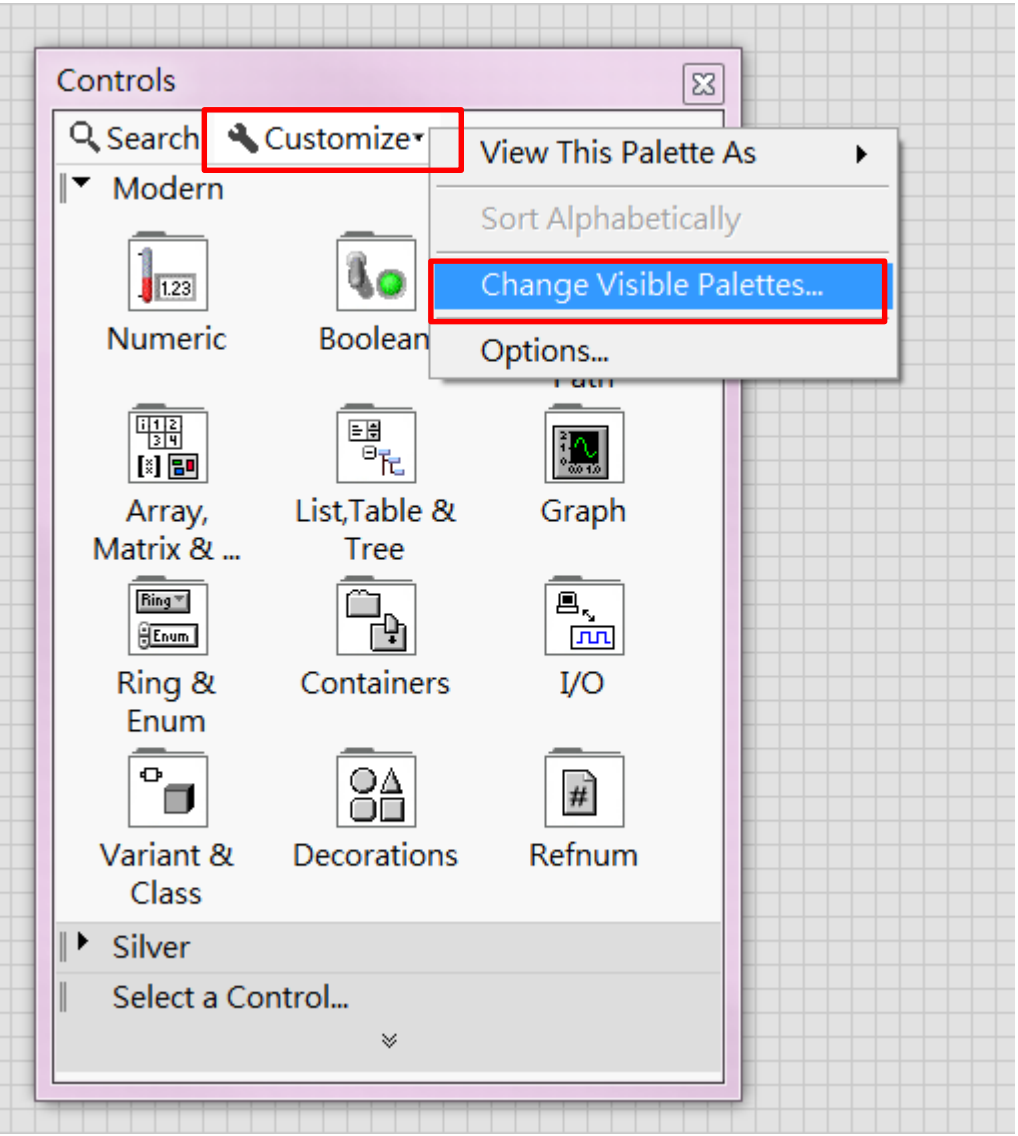


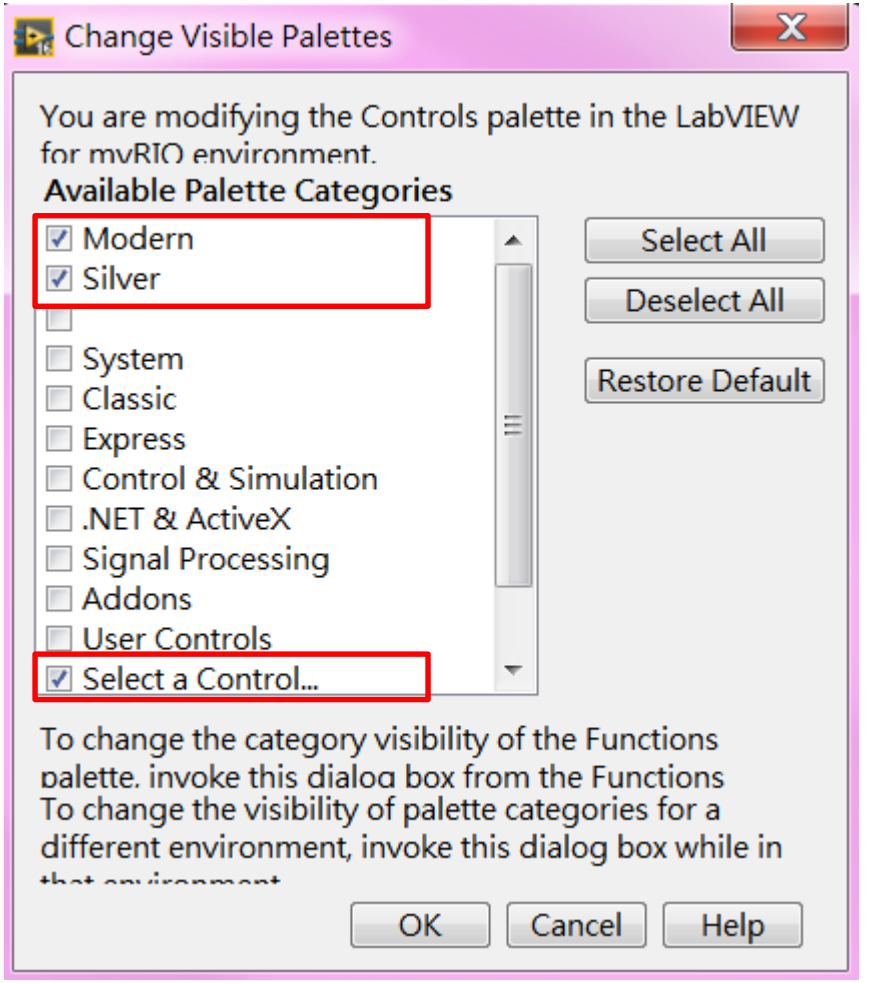
作者；曾智勇,chihyungtseng@hotmail.com

空白處點滑鼠右鍵



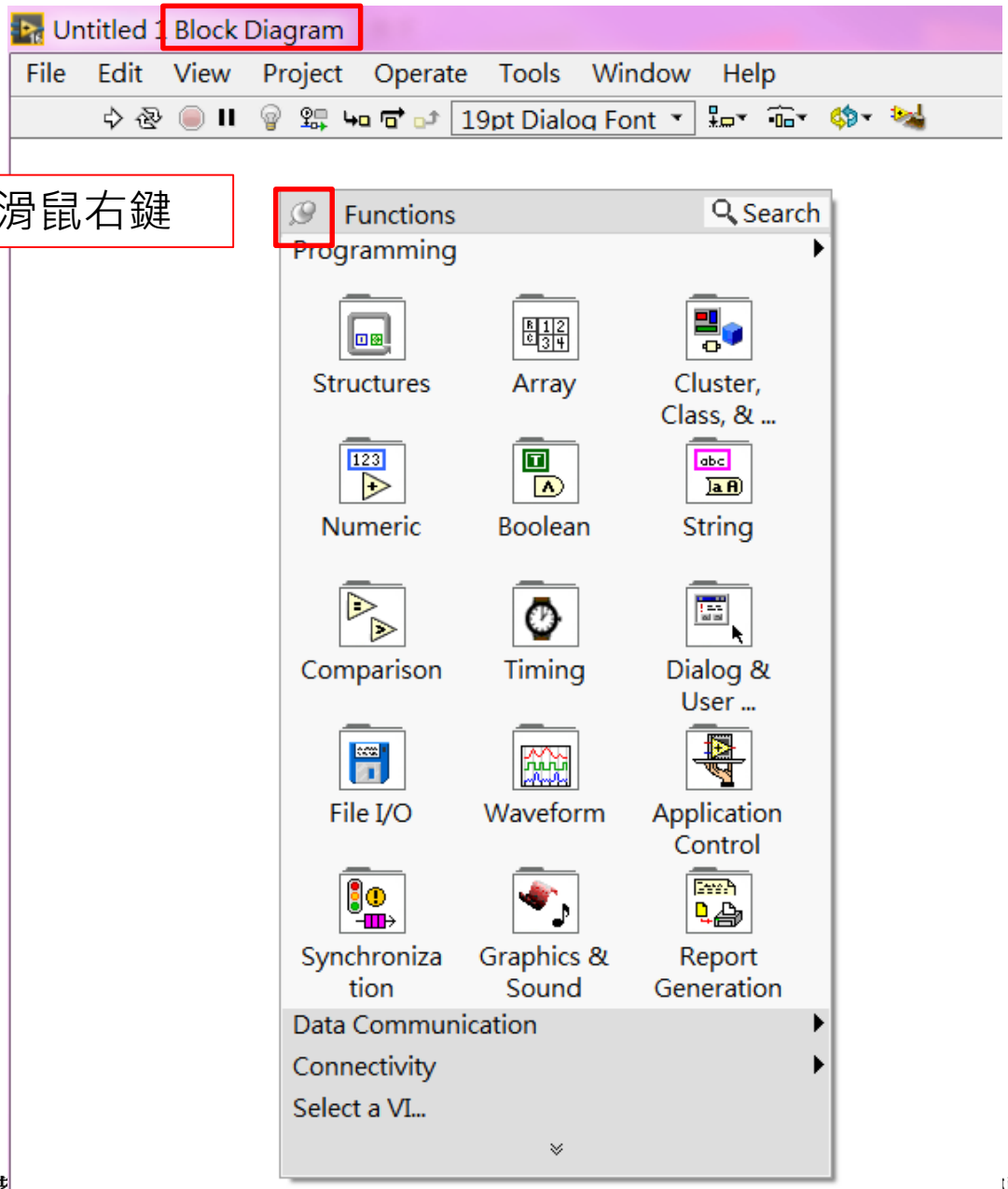
作者；曾智勇,chihyungtseng@hotmail.com



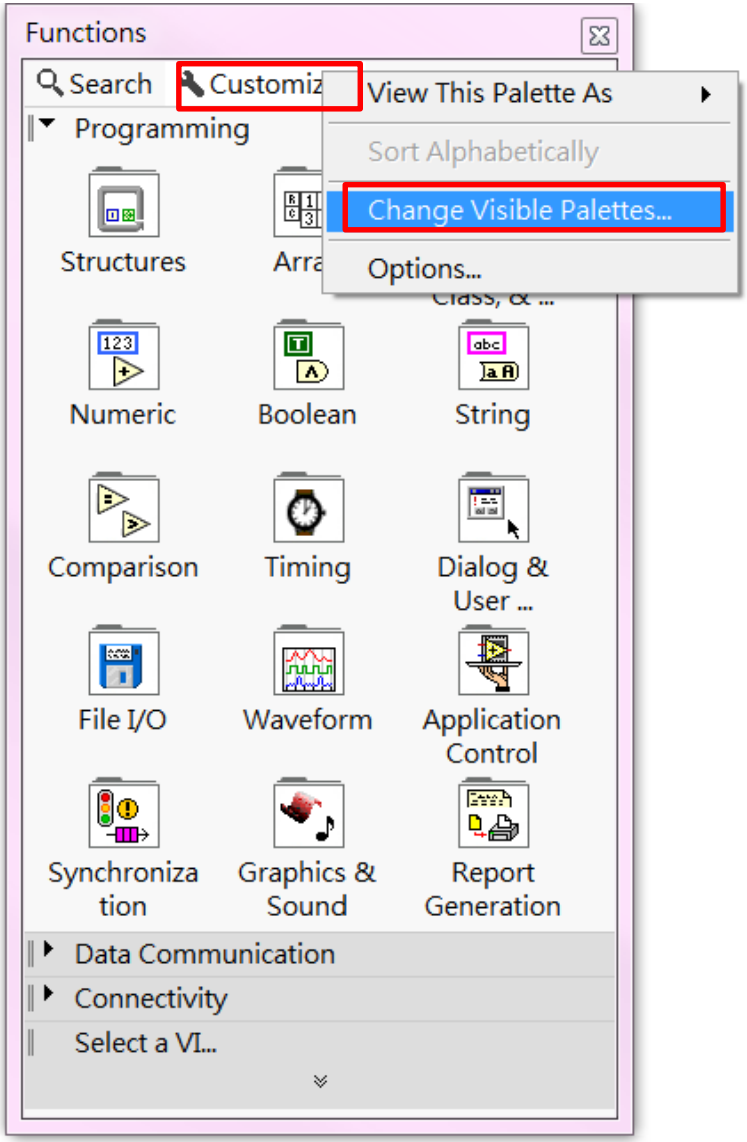


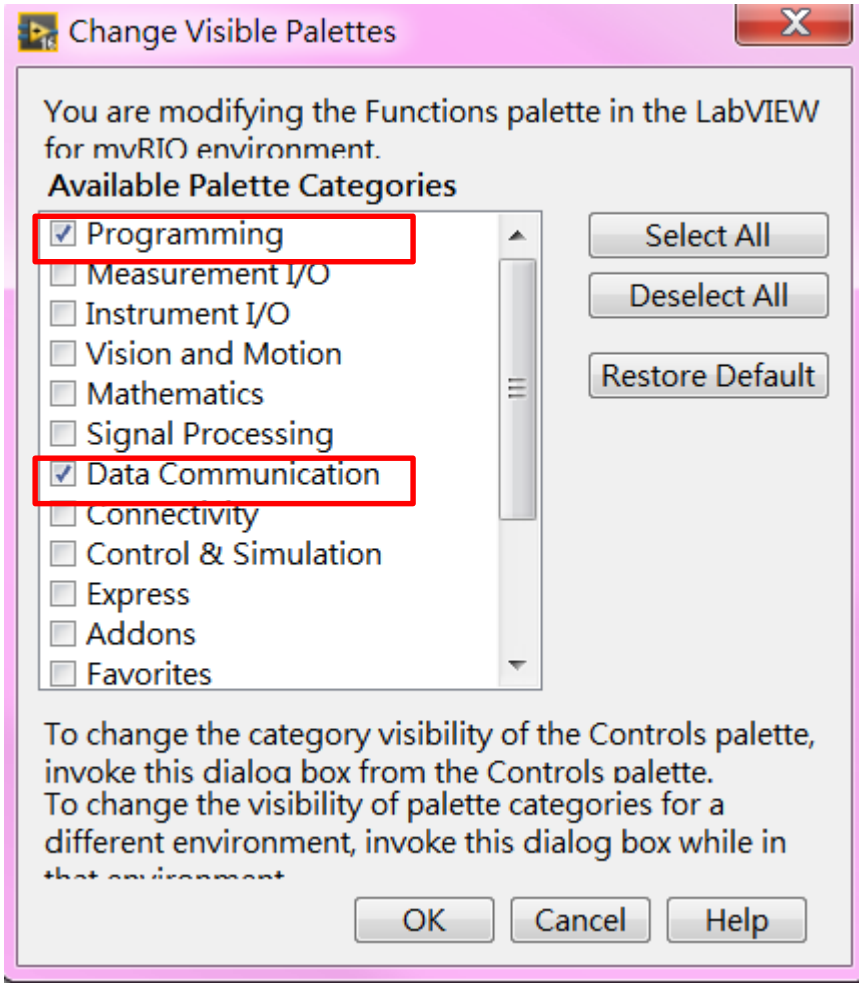
作者；曾智勇,chihyungtseng@hotmail.com

空白處點滑鼠右鍵



勇, chihyungtseng@hotmail.com



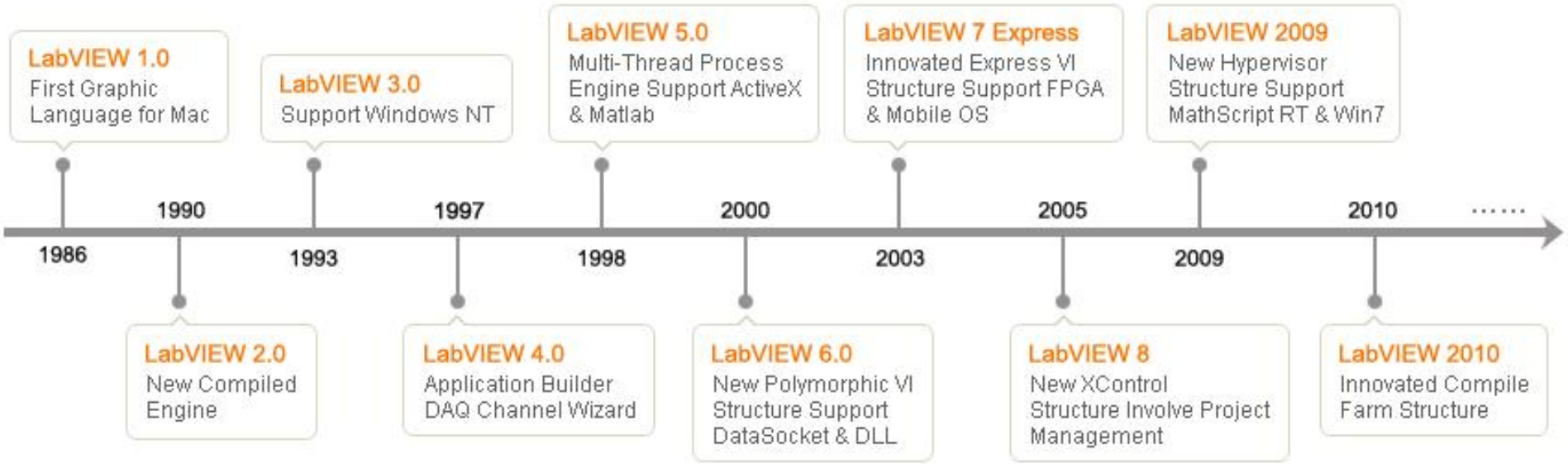


作者；曾智勇,chihyungtseng@hotmail.com

LabVIEW簡介

作者；曾智勇, chihyungtseng@hotmail.com

LabVIEW簡介



LabVIEW歷史演進

※近年來使用人數急遽攀升，2009年~2016年，每年更新，新增實用功能與運作效能。

作者；曾智勇,chihyungtseng@hotmail.com

LabVIEW簡介

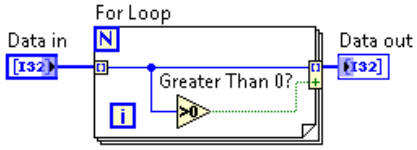
- LabVIEW 歷年重大新功能：

2009

- 儲存程式png圖
- 自動清理程式

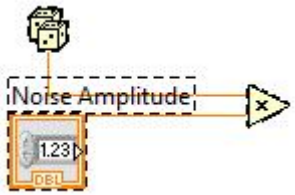
2012

- 條件化輸出



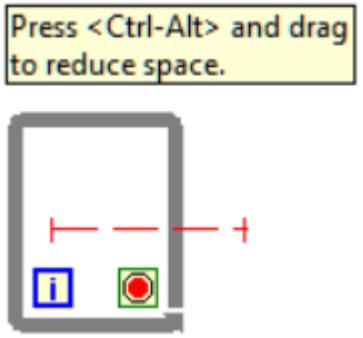
2014

- 自動接線 Ctrl+W@Quick Drop



2015

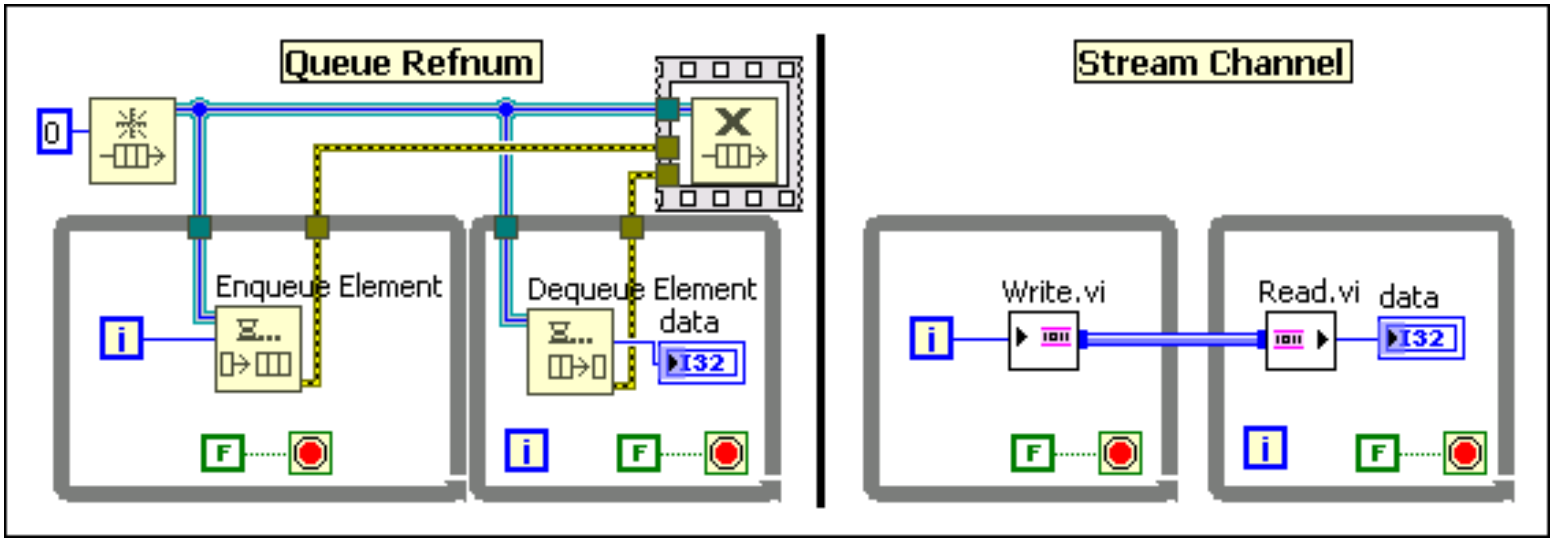
- 縮小工作區



作者；曾智勇, chihyungtseng@hotmail.com

LabVIEW簡介

- LabVIEW 2016年版新功能：迴圈通道(跨迴圈傳遞資料)



2016年，重大新功能，迴圈通道

作者；曾智勇,chihyungtseng@hotmail.com

LabVIEW簡介

- LabVIEW 具有相當多免費/付費擴充模組：
- 付費模組：
  - 數位濾波器設計工具組
  - PID 控制工具組
  - Sound and Vibration Measurement Suite
  - LabVIEW Embedded for ARM Microcontrollers Module
  - LabVIEW FPGA Module
  - LabVIEW MathScript RT Module
  - LabVIEW Real-Time Module
  - 機器人開發模組
  - 觸控式面板模組
  - NI LTE Measurement Suite
  - NI 頻譜量測工具組
  - 無線感測器網路 (WSN) 模組
  - ……

作者；曾智勇, chihyungtseng@hotmail.com

LabVIEW簡介

- **LabVIEW 2016** 擴充模組：

進階訊號處理工具組 (225 MB)	更多資訊
控制設計與模擬模組 (475 MB)	更多資訊
資料庫連結工具組 (83.4 MB)	更多資訊
DataFinder 工具組 (164 MB)	更多資訊
資料記錄與監控模組 (879 MB)	更多資訊
桌上型執行追蹤工具組 (346 MB)	更多資訊
數位濾波器設計工具組 (136 MB)	更多資訊
FPGA Compile Farm 工具組 (257 MB)	更多資訊
FPGA Module (442 MB)	更多資訊
MathScript RT Module (342 MB)	更多資訊
NI 運動小幫手 (477 MB)	更多資訊
NI SoftMotion Module (296 MB)	更多資訊
Real-Time Module 1 GB)	更多資訊
報表產生工具組 – 適用於 Microsoft Office (84.1 MB)	更多資訊
機器人模組 (296 MB)	更多資訊
聲音與振動量測套餐 (442 MB)	更多資訊
狀態圖模組 (101 MB)	更多資訊
系統識別工具組 (136 MB)	更多資訊
單元測試框架工具組 (102 MB)	更多資訊
VI 分析器工具組 (87.1 MB)	更多資訊
視覺開發模組 (2.04 GB)	更多資訊

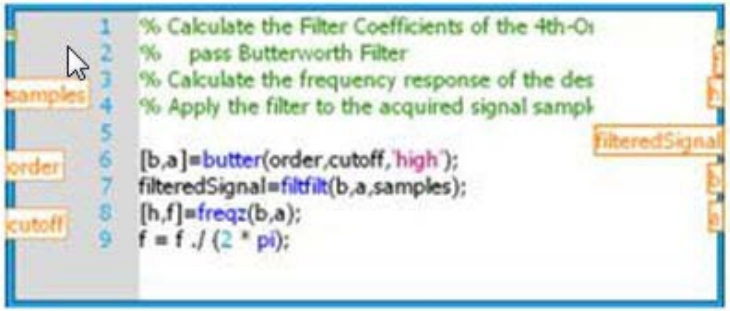
作者；曾智勇, chihyungtseng@hotmail.com

LabVIEW簡介

- LabVIEW 可以直接利用 MATLAB 的語法或 .m 檔案：

- 使用 MATLAB Script Node：
  MATLAB Script Node 可以讓使用者將 .m 檔案加入到 LabVIEW 。若要使用 MATLAB Script Node ，使用者必須要在電腦上安裝 MATLAB 6.5 或更新的版本。

- 使用 LabVIEW MathScript ：
  使用 .m 檔案的程式碼語法，包含可用於數學、訊號處理與分析作業的超過 700 筆常見函式。 可於 LabVIEW 中整合圖形化與文字式的程式碼。透過 MathScript Node ，則可直接輸入 .m 檔案程式碼文字，或從文字檔案匯入。基於此種相容性，可讓使用者繼續搭配先前所開發的 .m 檔案程式碼。



作者；曾智勇, chihyungtseng@hotmail.com

LabVIEW簡介

- LabVIEW 是一個標準的程式語言，它擁有著非常強大與硬體的整合性，所以只要硬體廠商有提供 Driver / API ，此儀器即可透過 LabVIEW 來使用。

- 在 LabVIEW 中使用儀器/硬體常見的方法有以下三種方法：
  1. 硬體廠商直接提供 LabVIEW 驅動程式(VI 檔)：
     許多硬體廠商已直接提供各種程式語言的驅動程式與說明文件，只要取得後即可方便地在 LabVIEW 中直接使用。

  2. 硬體廠商提供標準的 API (DLL)：
     LabVIEW 具有呼叫 DLL / ActiveX / .NET 等的功能，只要取得硬體裝置的 API 與說明，即可在 LabVIEW 中使用。

  3. 透過 Instrument Driver Network 取得驅動程式：
     NI 提供了一個叫做 Instrument Driver Network 的網頁，這個網頁整理了各家廠商的驅動程式。此網頁上提供超過 275 家廠商，超過 6000 種以上儀器的驅動程式。

作者；曾智勇,chihyungtseng@hotmail.com

LabVIEW簡介

- LabVIEW 可以在 Windows、Mac OS 與 Linux 上運作。除此之外，程式也可以在即時作業系統(RT)上執行，以達到決定性的控制；也可以在 FPGA 上執行，達到精準且快速的控制。

- 使用 LabVIEW Embedded 模組/DSP 模組(Digital Signal Processing Module)進行不應用的開發，例如PDA、DSP、ARM 等環境的程式。

- LabVIEW Mobile 模組讓程式可以在手持式裝置上執行。透過藍芽(bluetooth)、文字簡訊(SMS text message)、 802.11 等方式與 PDA 溝通，進行資料分析與顯示。

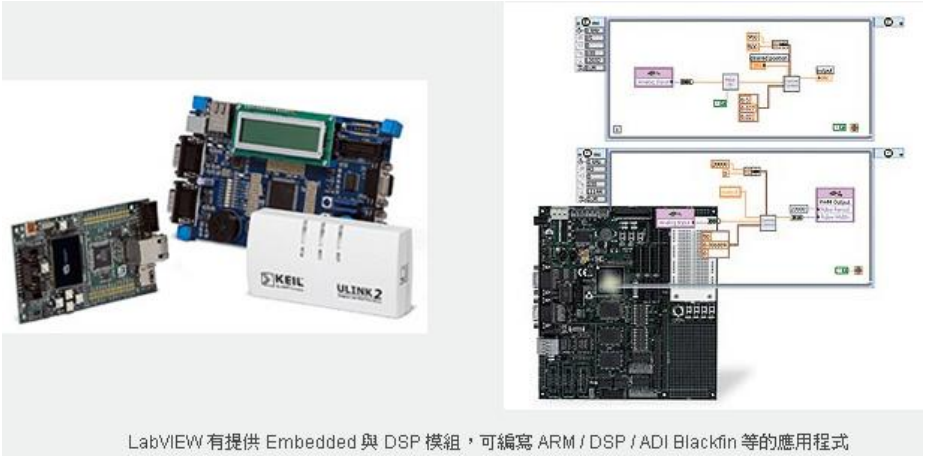
LabVIEW ARM開發板STM32 Cortex-M4

作者；曾智勇,chihyungtseng@hotmail.com

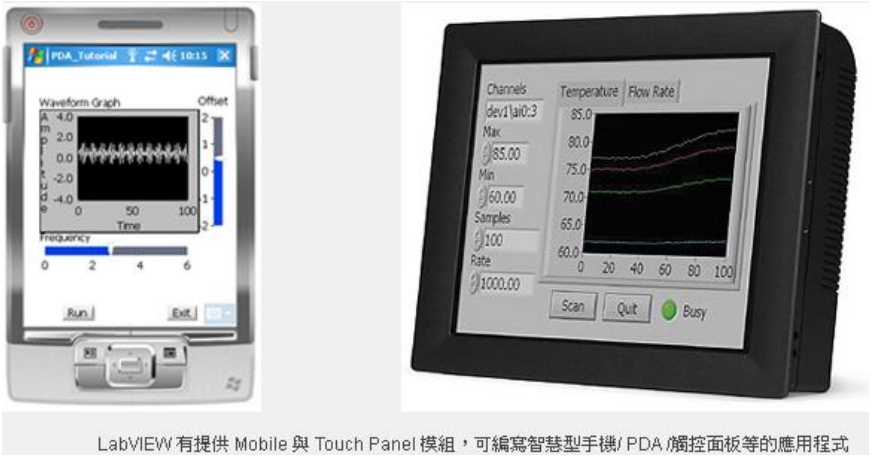
LabVIEW簡介

- LabVIEW Touch Panel 模組， LabVIEW 程式也可以在觸控式面板上執行。

- 其他諸如 ARM 微控制器等嵌入式系統，也可以使用 LabVIEW Embedded 模組來完成開發。 LabVIEW 也有 DSP 模組(Digital Signal Processing Module)，可以圖形程式設計快速開發 DSP 應用程式。



LabVIEW 有提供 Embedded 與 DSP 模組，可編寫 ARM / DSP / ADI Blackfin 等的應用程式

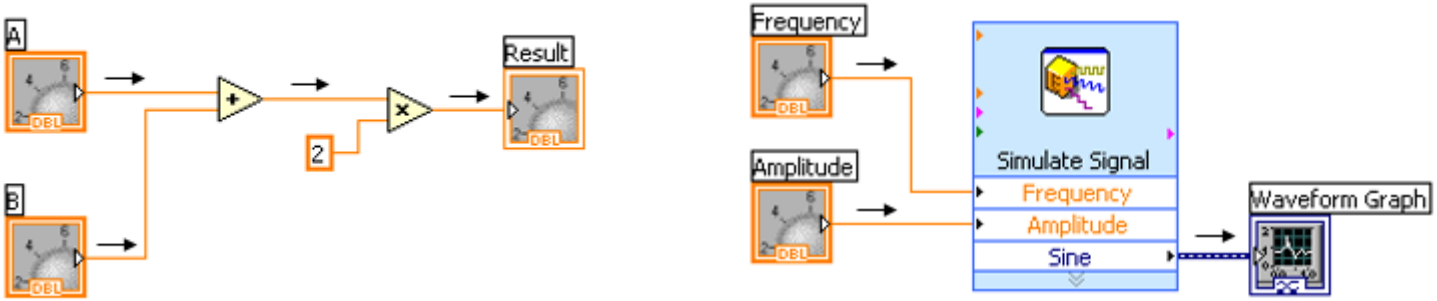


LabVIEW 有提供 Mobile 與 Touch Panel 模組，可編寫智慧型手機/ PDA /觸控面板等的應用程式

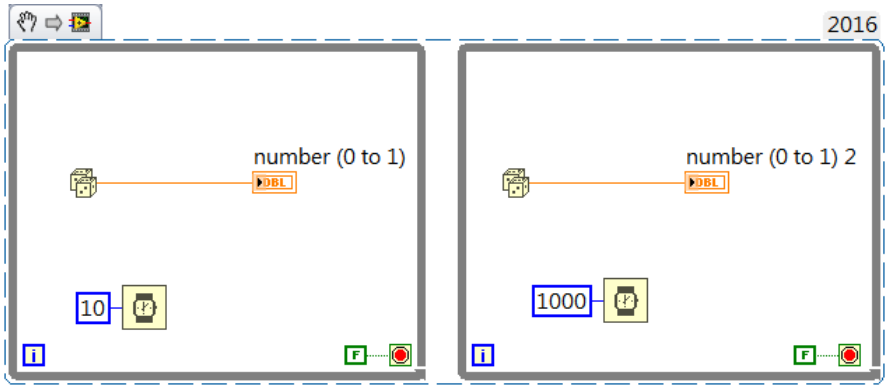
作者；曾智勇, chihyungtseng@hotmail.com

LabVIEW簡介

- LabVIEW是資料流的程式語言，可以很直覺的控制程式執行順序



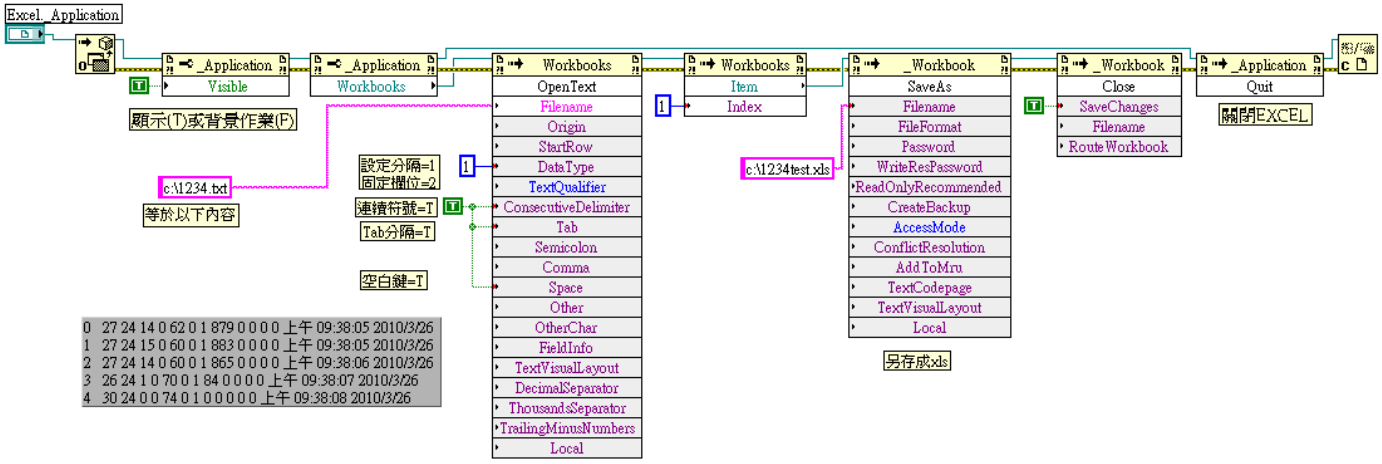
- LabVIEW可自動跑多執行緒，當While Loop之間沒有資料前後關聯性時，程式會自動跑多執行緒，不需額外設定



作者；曾智勇,chihyungtseng@hotmail.com

LabVIEW簡介

- LabVIEW 是一個開放的程式語言，可以與其他程式所建立的 Shared Library 溝通。
- 在 Windows 系統中定義了一種檔案型態，能夠動態地進行載入及執行，稱為動態函式庫DLL (Dynamic Linkable Library)；所以可以將重覆使用的程式編譯成 DLL 的格式，以LabVIEW呼叫使用。
- 在 Windows 中常見的 ActiveX 與 .NET 物件等，LabVIEW 也都提供了對應的函式來使用；至於其他不同的程式語言或特殊型態的動態函式庫，也都有其對應的呼叫方式。



互叫EXCEL程式，將TXT檔進行分隔、另存。

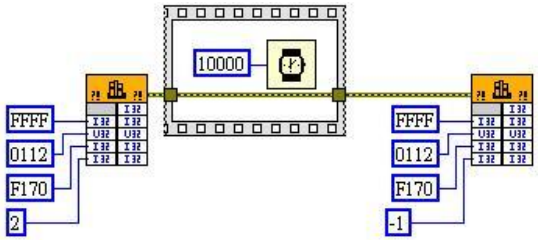
作者；曾智勇,chihyungtseng@hotmail.com

LabVIEW簡介

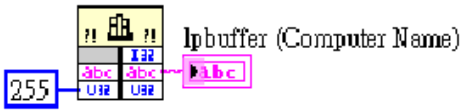
- LabVIEW 內建了高達 2000 個以上的函式，對於 DLL 的載入也提供了相關的元件。常見的方法有以下兩種：

1. 已知函式輸入與輸出資料型態：使用 Call Library Function Node。
呼叫外部程式所建立的 DLL / Windows API。在 Library name or path 的欄位選擇欲載入的 DLL 檔，然後使用 Function name 欄位去選擇欲使用的函式名稱，並設定正確的變數資料型態。

2. 具有函式.h檔：使用 Import Shared Library Wizard。
透過此精靈的一步步設定，即可將 DLL 內的各函式快速地轉變成各 LabVIEW 中的 VI 。使用者可以利用這種方式快速地將 DLL 匯入，再根據自己的需求針對個別 VI 作修改或調整。



使用User32.dll
送出螢幕開啟/關閉訊號



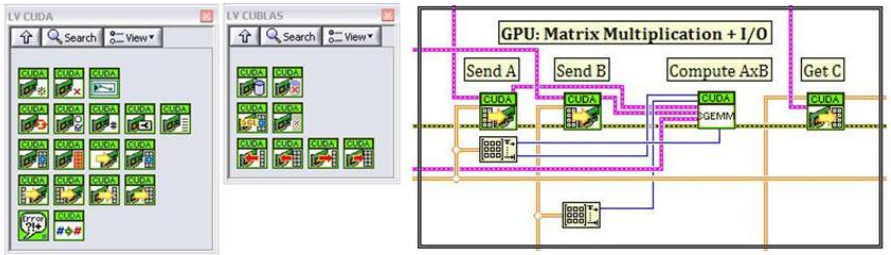
使用kernel32.dll中的函式取得目前電腦名稱。

作者；曾智勇, chihyungtseng@hotmail.com

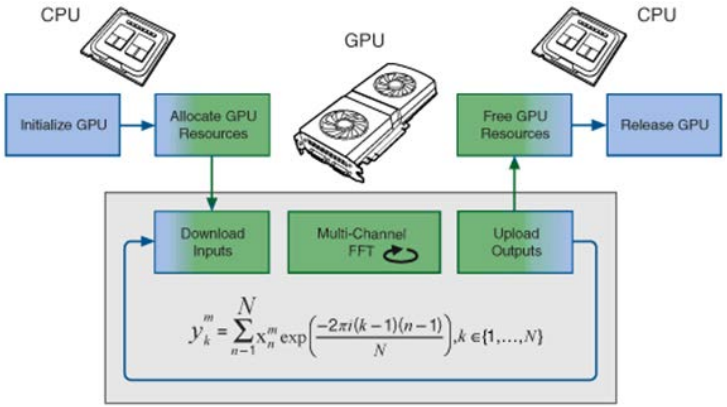
LabVIEW簡介

- LabVIEW GPU 分析工具組透過 LabVIEW 納入了某些函式庫內建的函式，所以開發人員可以運用重要的 NVIDIA CUDA(Compute Unified Device Architecture)資源，完整利用GPU 的運算功能。

- 開發人員可把重要的運算作業轉移到 GPU 並加以操作，享受前所未有的強大處理資源。除了 FPGA 與 CPU 之外，現在還可以使用 GPU 快速處理所擷取到的資料，大幅降低平行資料處理與轉換作業所帶來的運算成本。

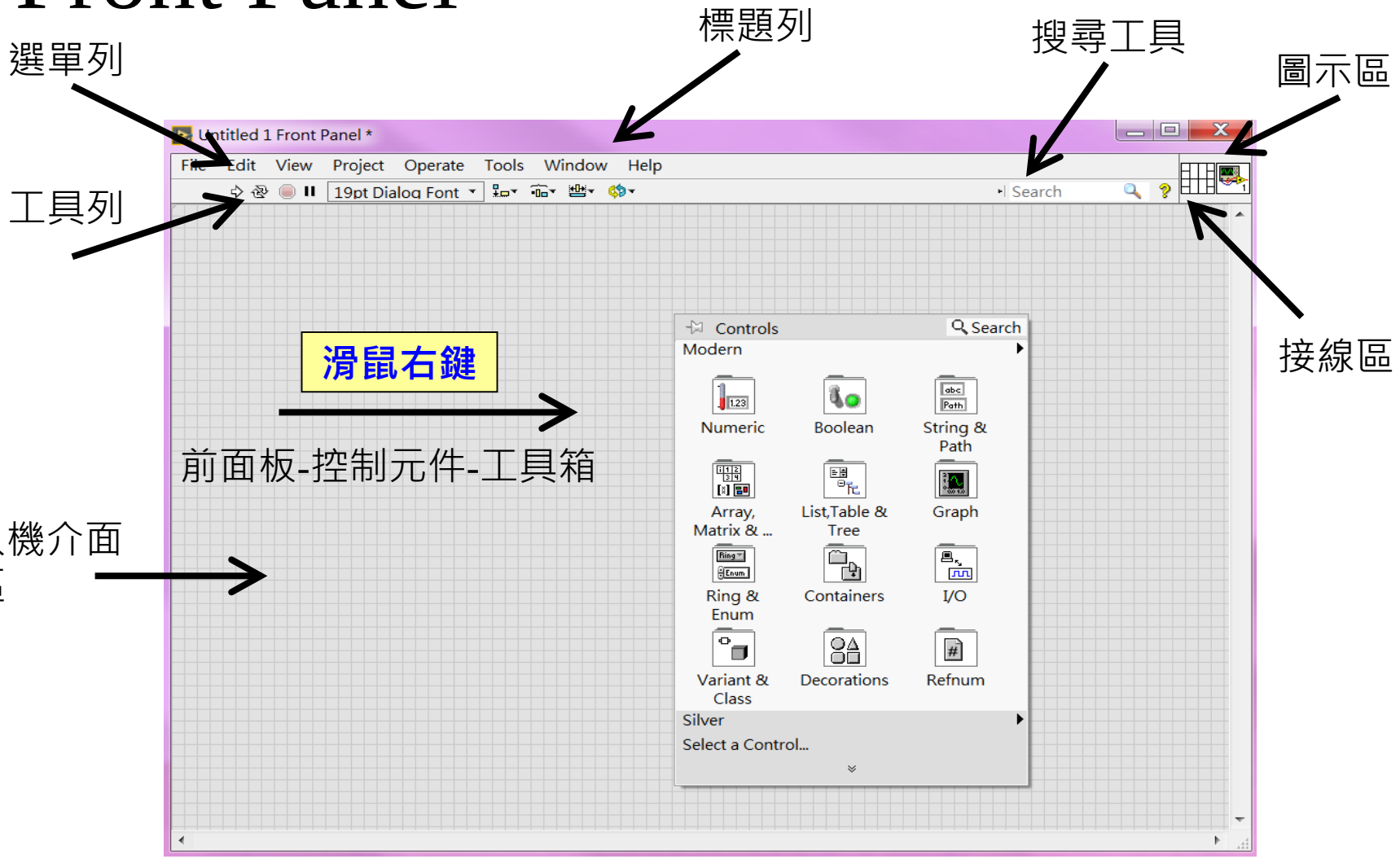


LabVIEW GPU運算模組



作者；曾智勇, chihyungtseng@hotmail.com

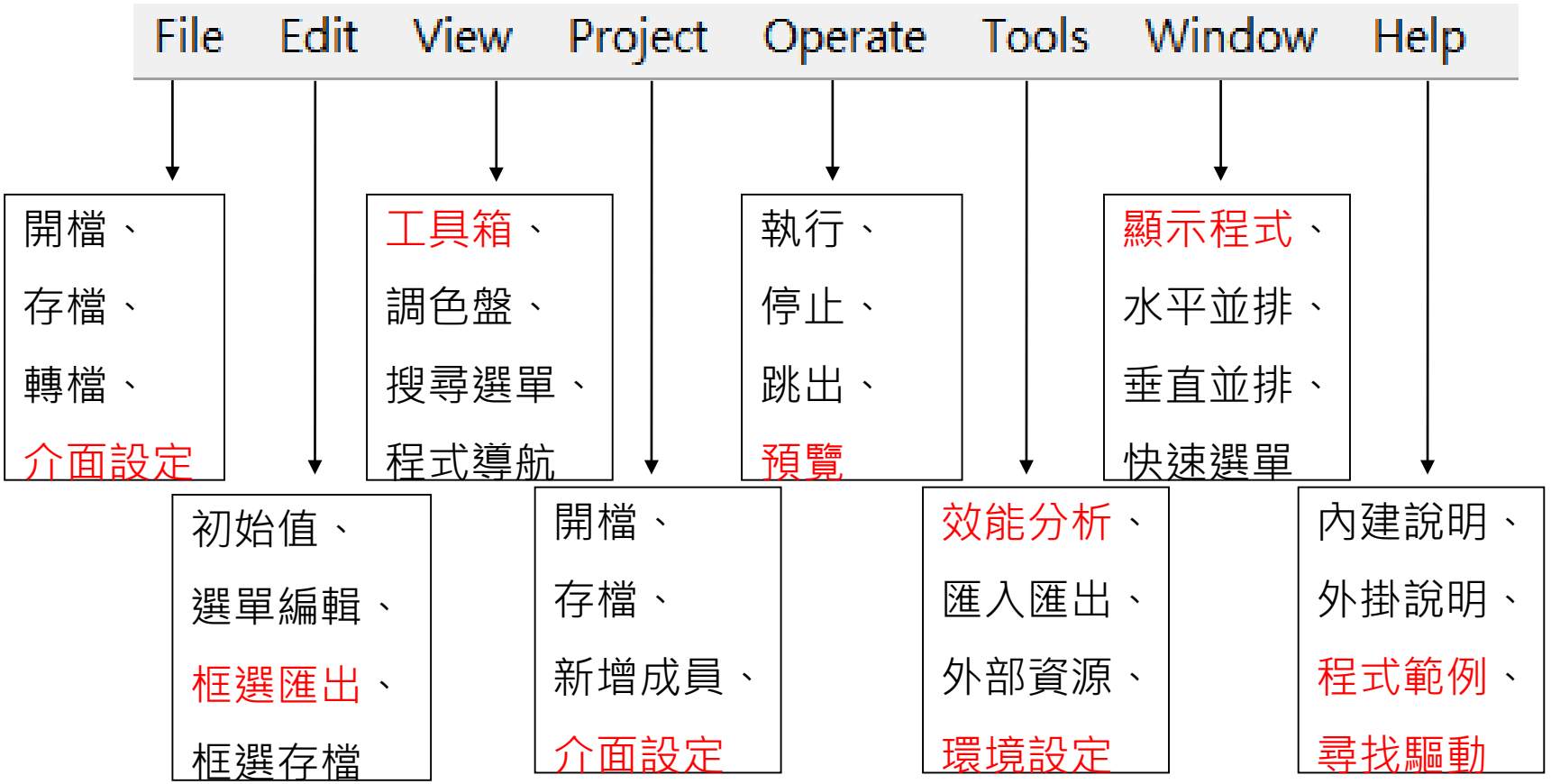
Front Panel



作者；曾智勇,chihyungtseng@hotmail.com

Front Panel

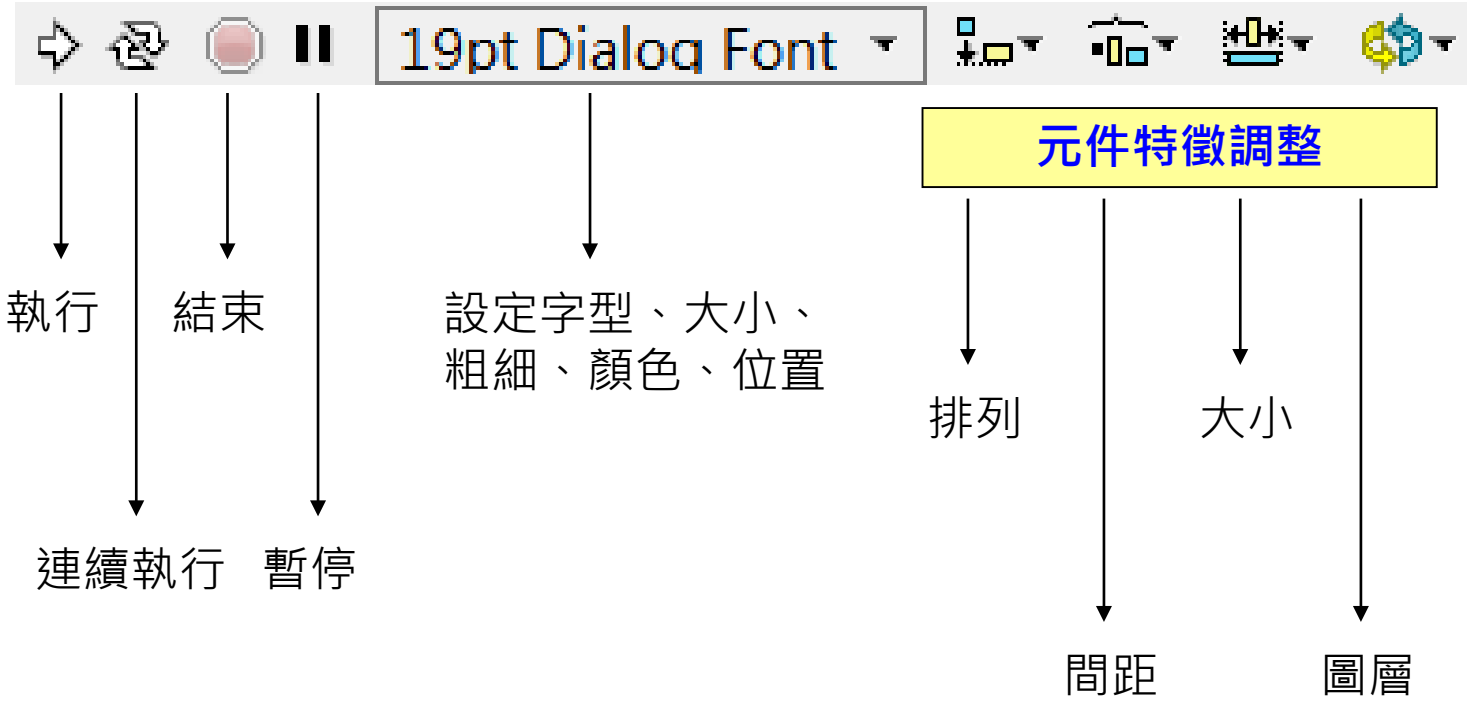
- 選單列：



作者；曾智勇,chihyungtseng@hotmail.com

Front Panel

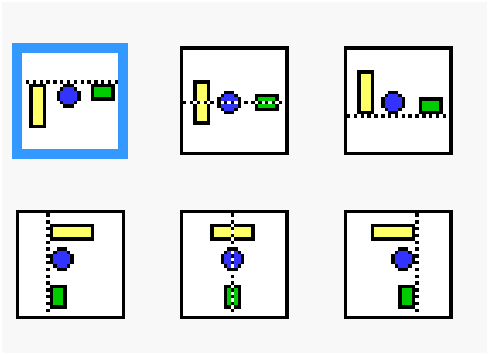
- 工具列：



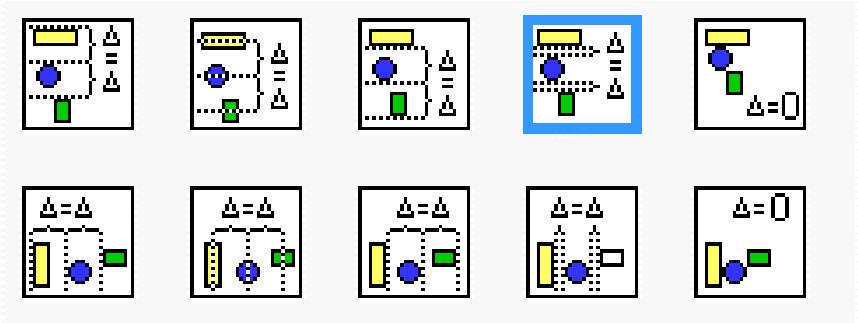
作者；曾智勇, chihyungtseng@hotmail.com

Front Panel

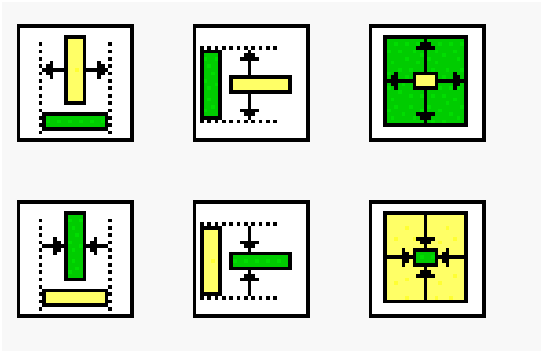
- 對齊工具：對齊框選的元件



- 間隔工具：均分框選元件間隔



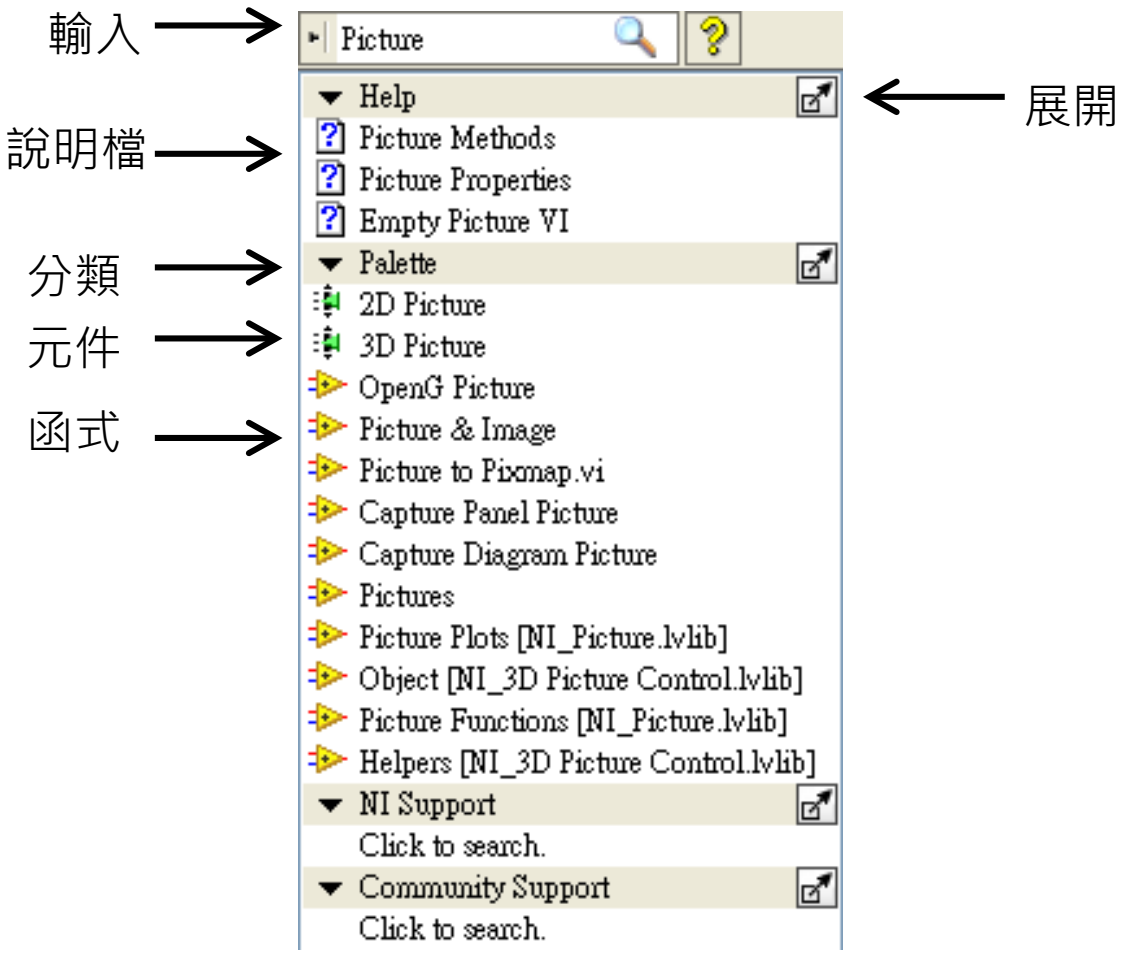
- 尺寸工具：統一框選元件尺寸



作者；曾智勇, chihyungtseng@hotmail.com

Front Panel

- 搜尋工具：

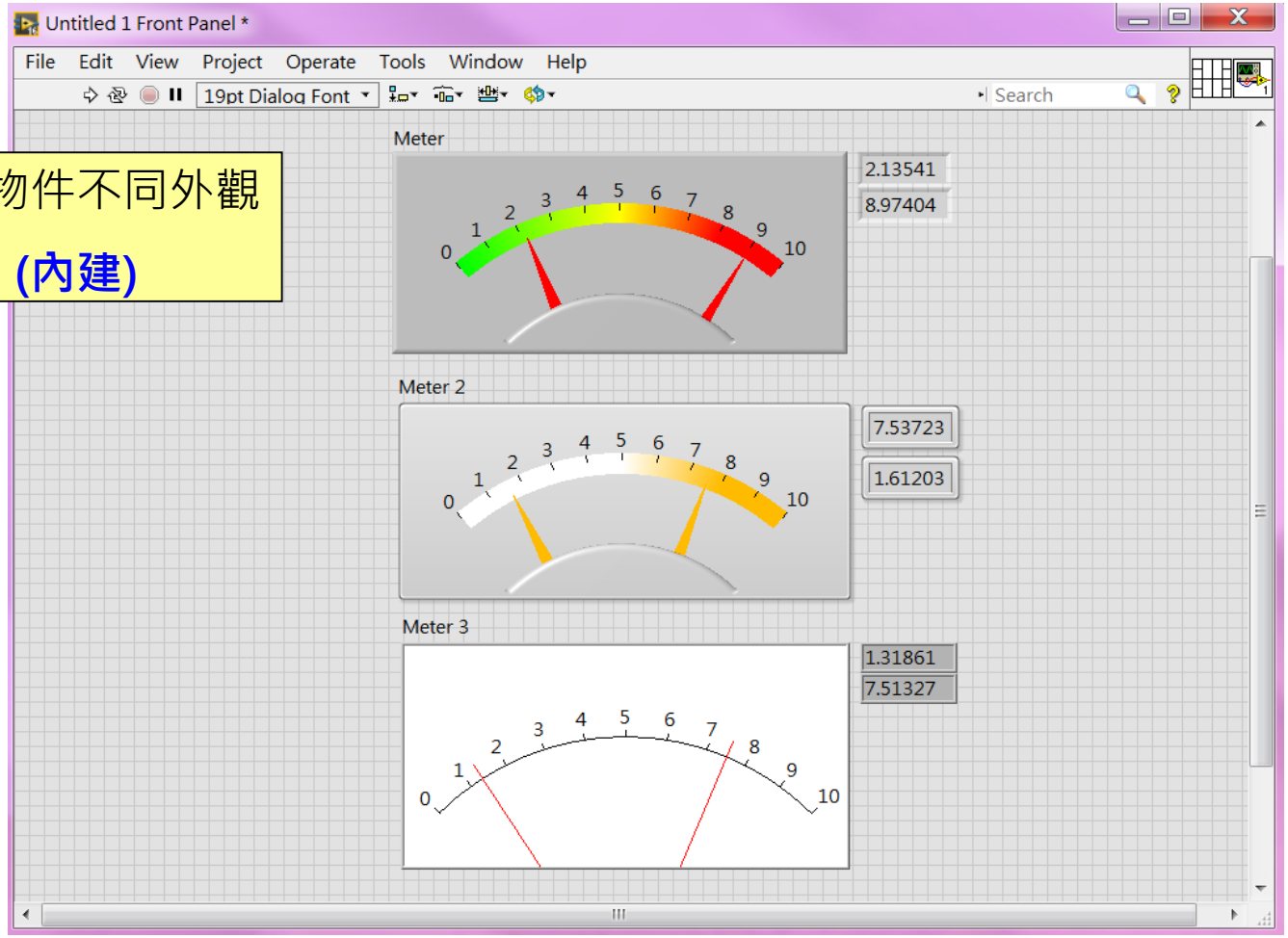


作者；曾智勇, chihyungtseng@hotmail.com

Front Panel

相同物件不同外觀

(內建)

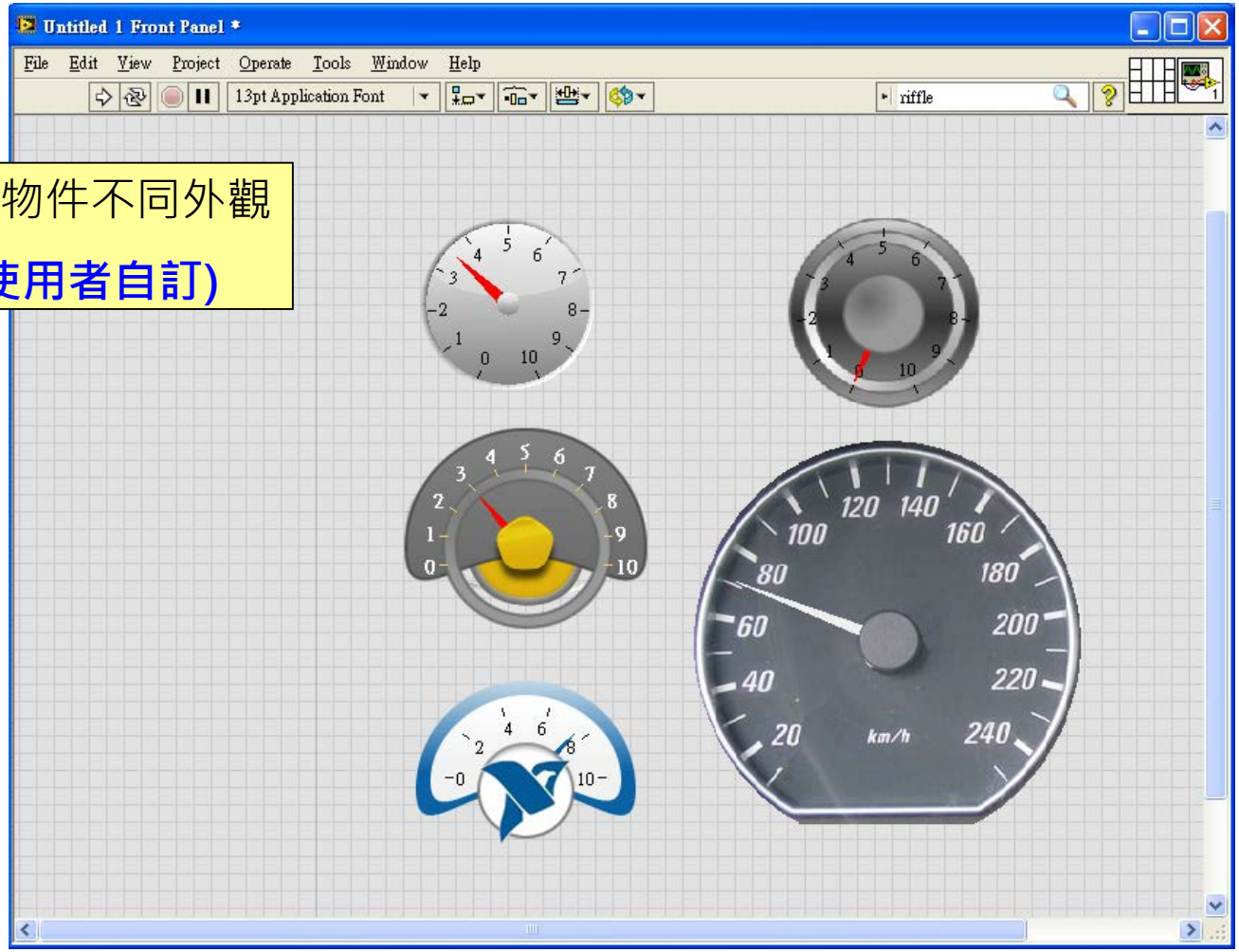


作者；曾智勇,chihyungtseng@hotmail.com

Front Panel

相同物件不同外觀

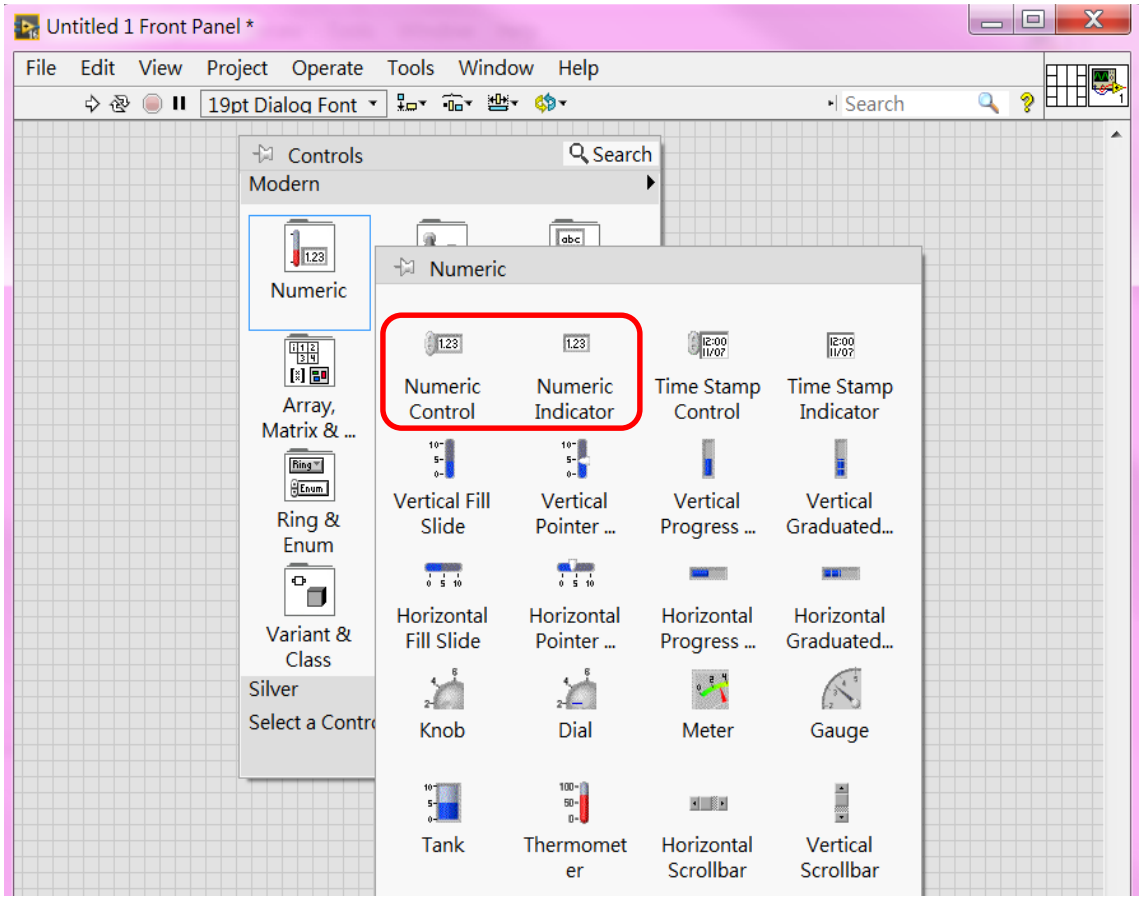
(使用者自訂)



作者；曾智勇, chihyungtseng@hotmail.com

Front Panel

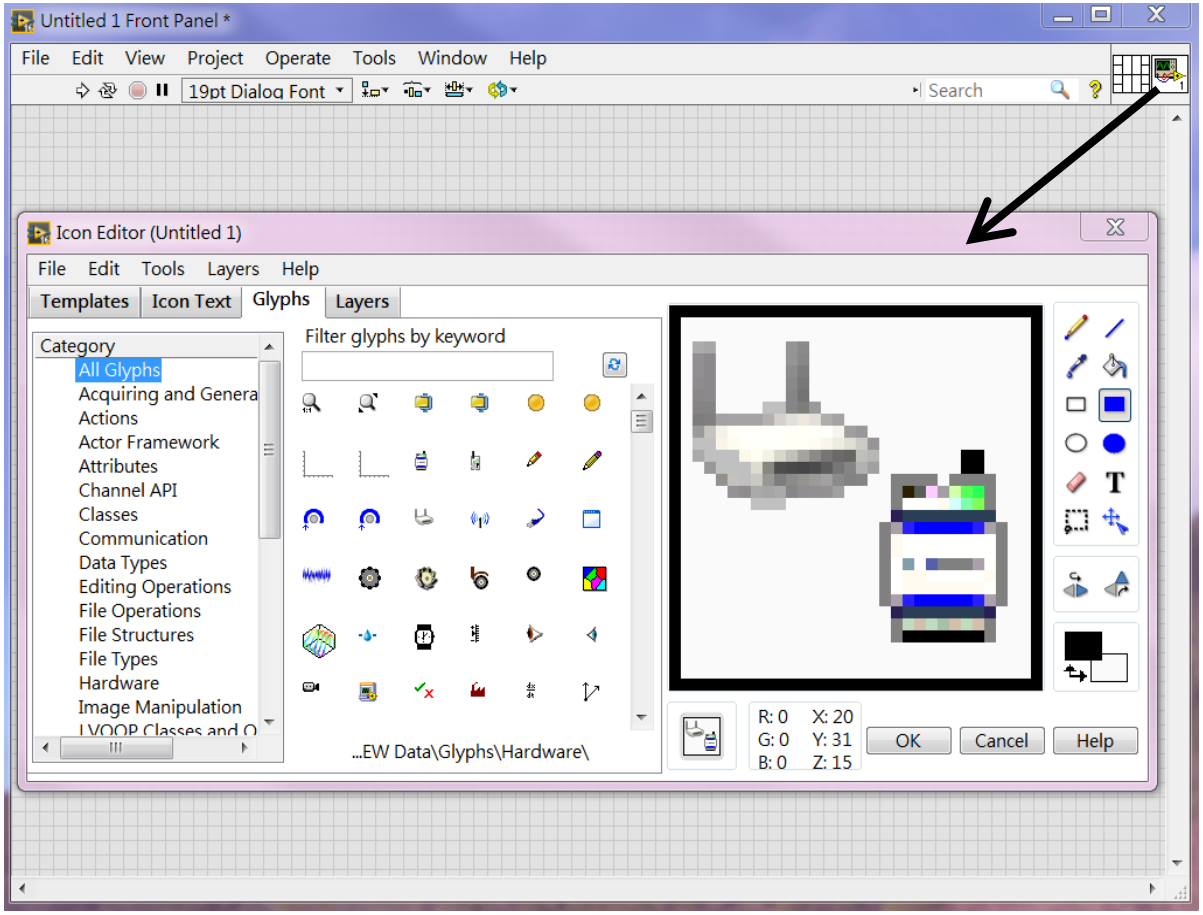
- Control = Input
- Indicator = Output



作者；曾智勇, chihyungtseng@hotmail.com

Front Panel

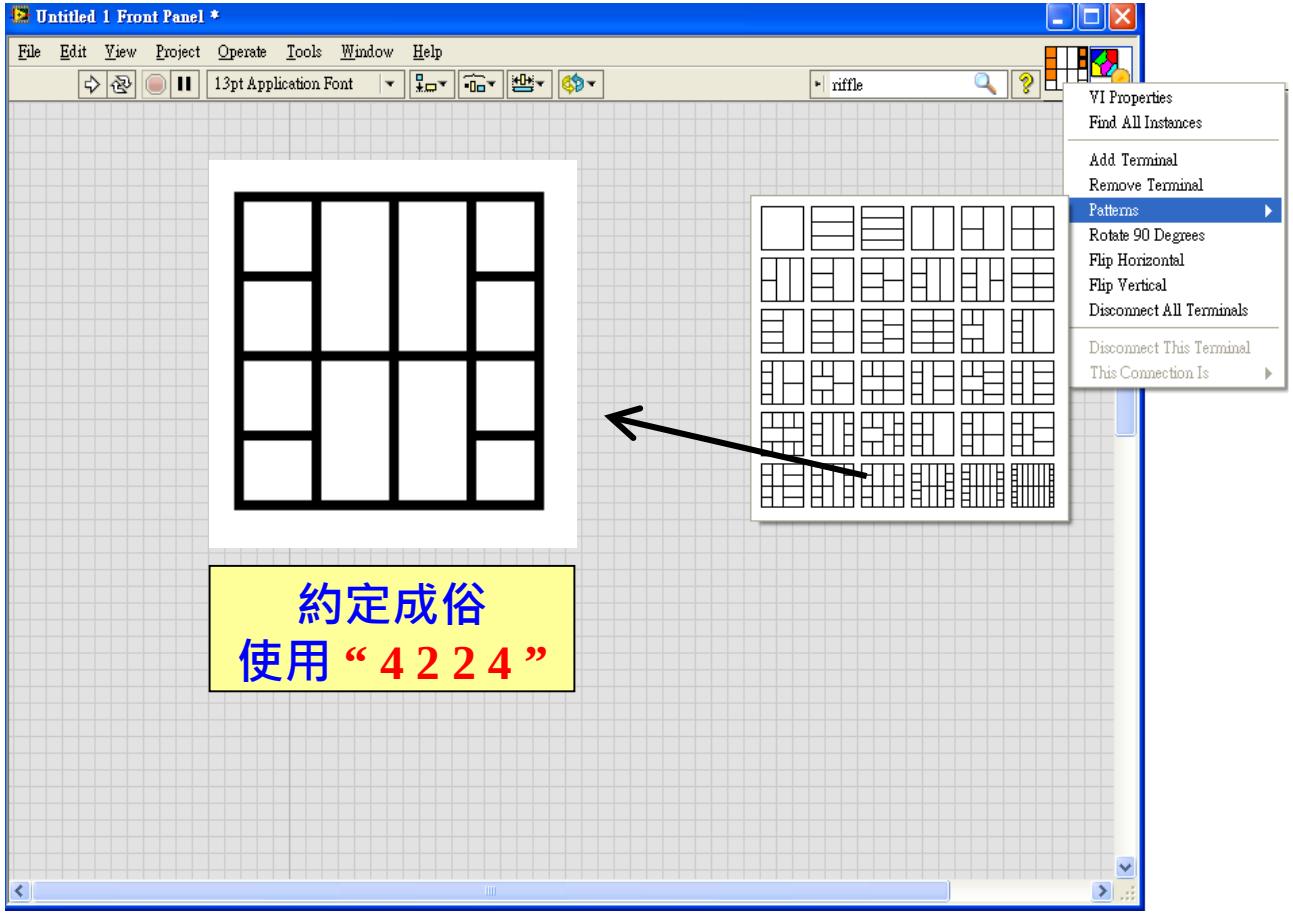
- 圖示編輯工具：



作者；曾智勇,chihyungtseng@hotmail.com

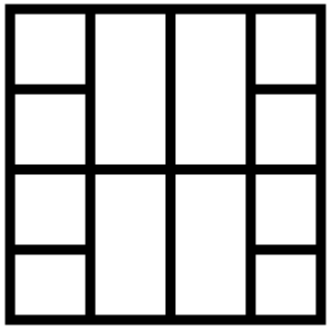
Front Panel

- 接線區編輯工具：

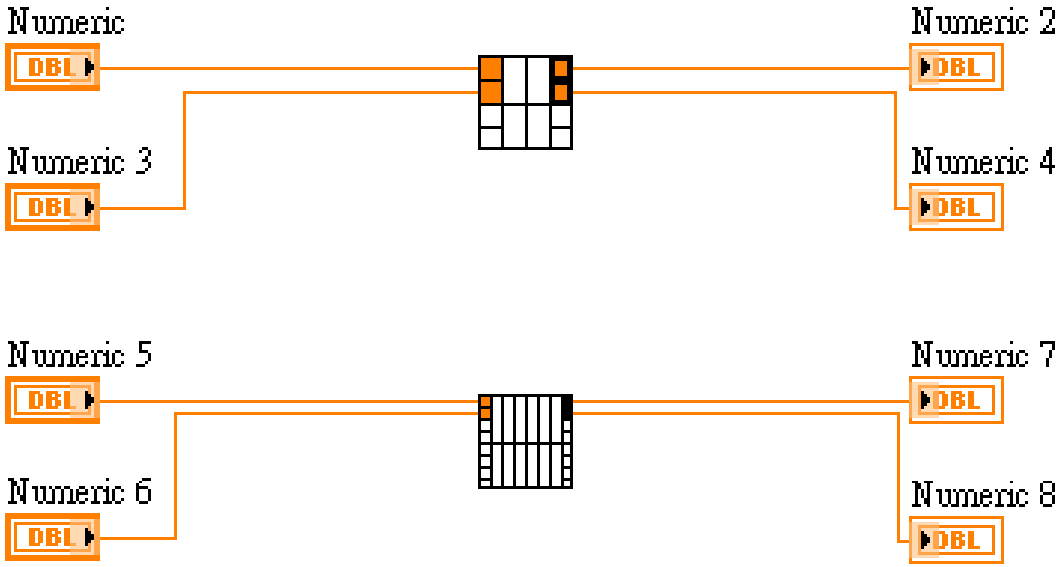


作者；曾智勇,chihyungtseng@hotmail.com

Front Panel

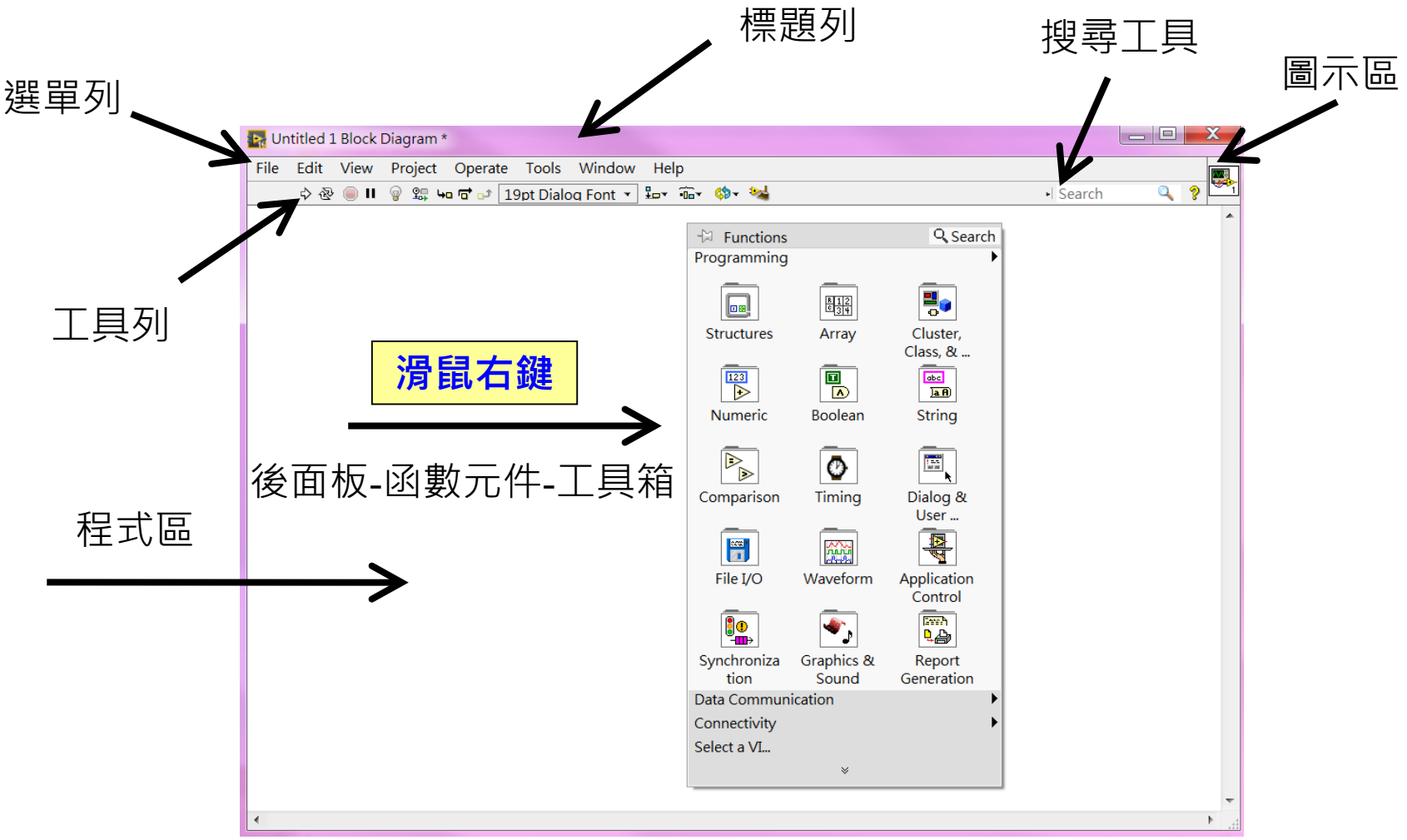


約定成俗
使用“4224”



作者；曾智勇,chihyungtseng@hotmail.com

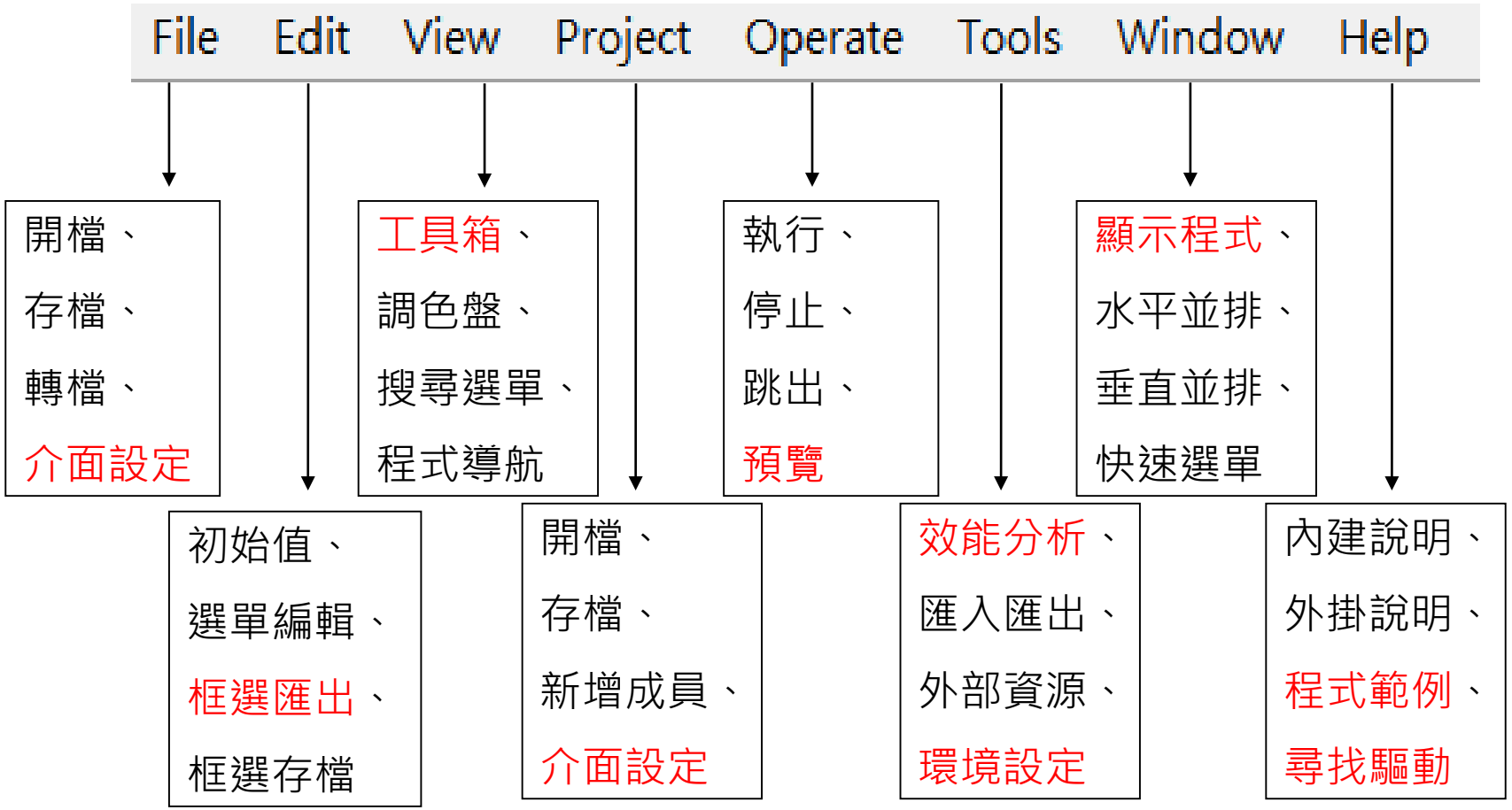
Block Diagram



作者；曾智勇,chihyungtseng@hotmail.com

Block Diagram

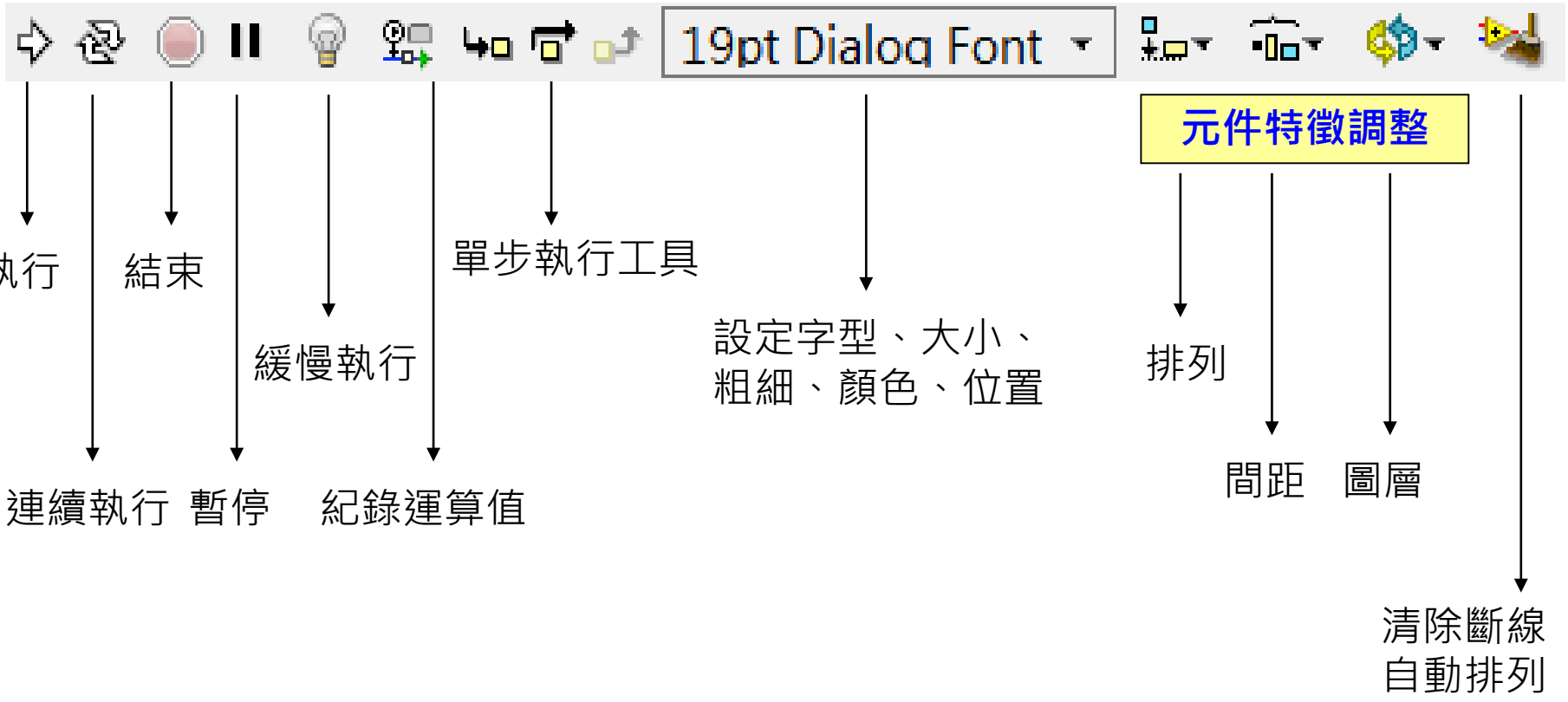
- 選單列：



作者；曾智勇,chihyungtseng@hotmail.com

Block Diagram

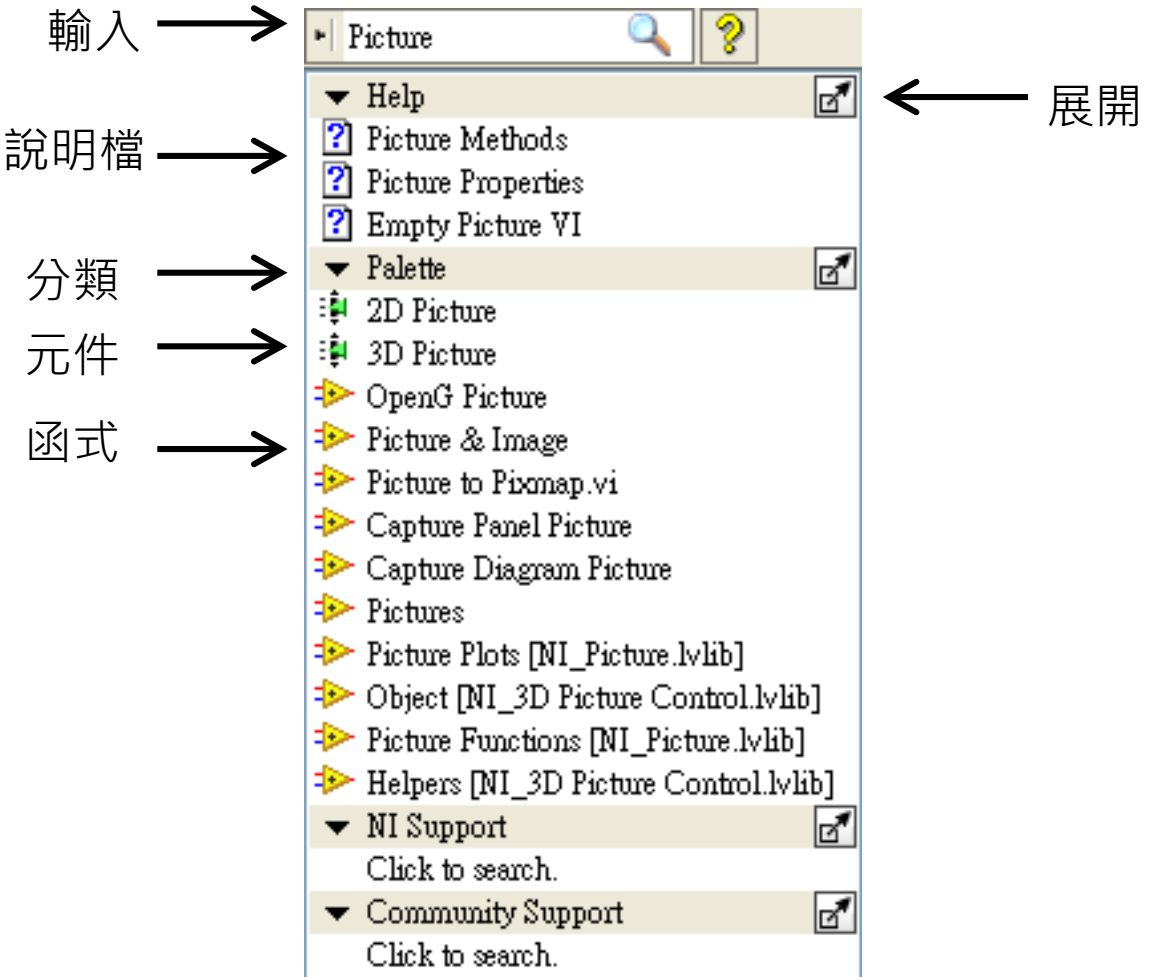
- 工具列：



作者；曾智勇, chihyungtseng@hotmail.com

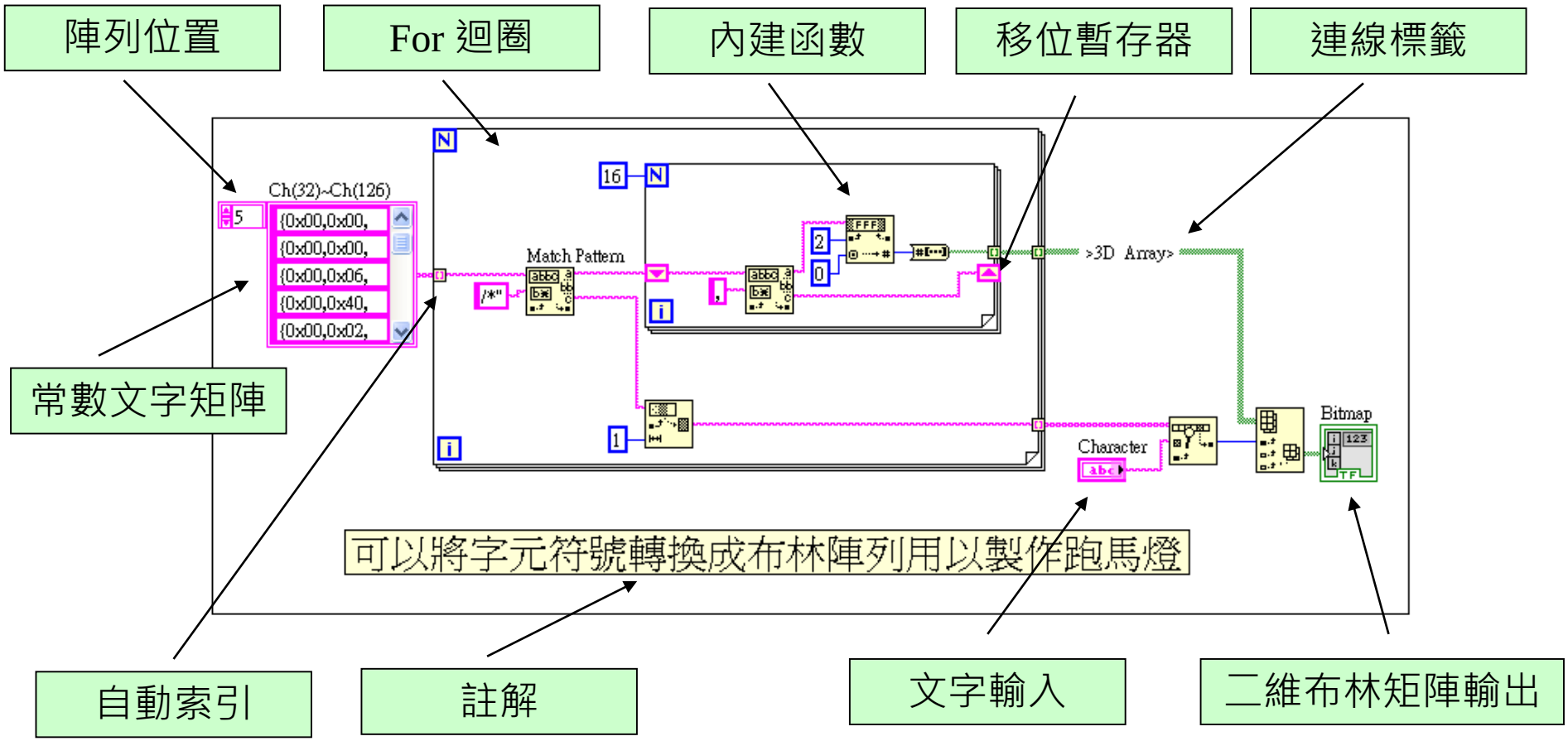
Block Diagram

- 搜尋工具：



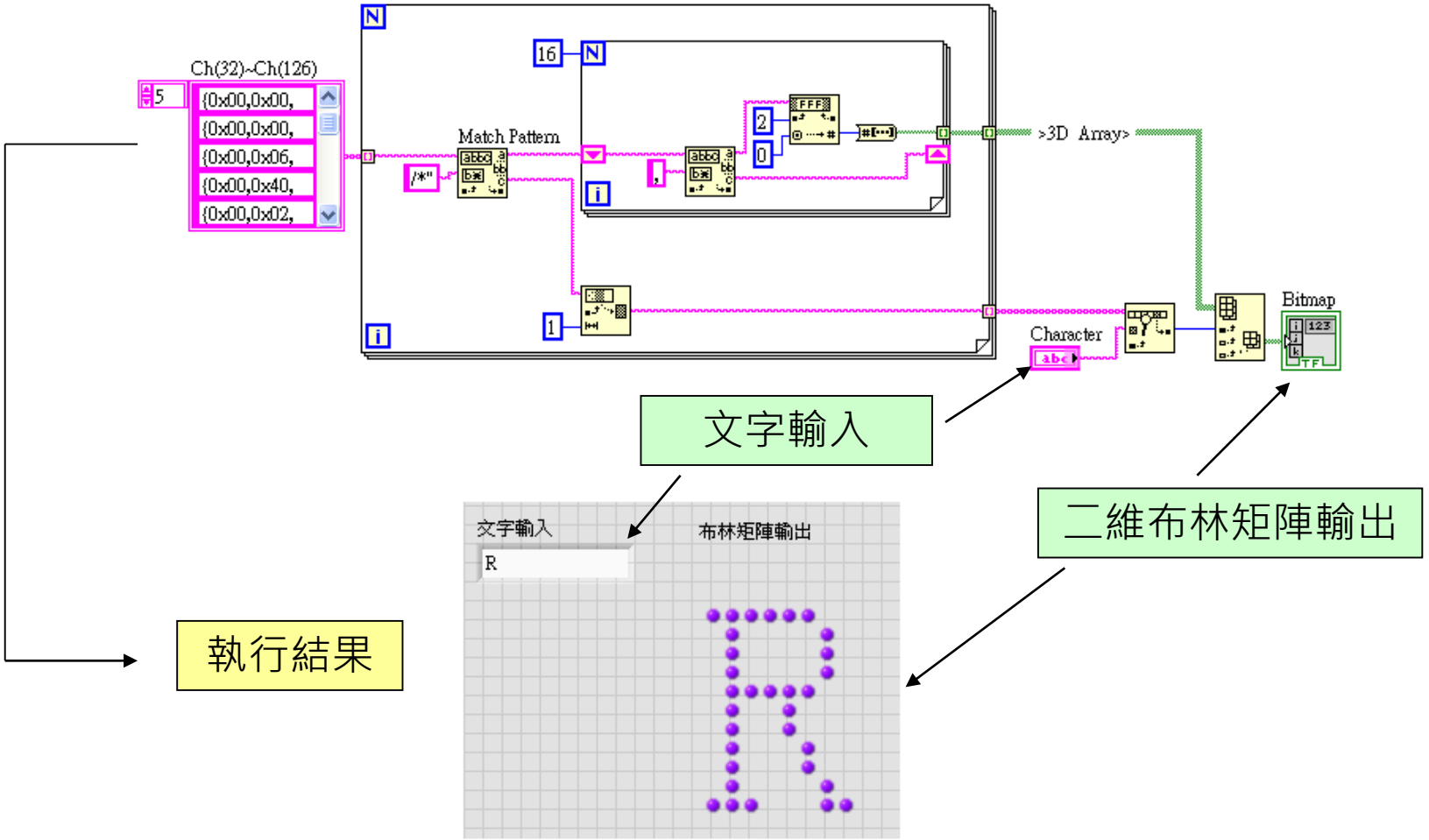
作者；曾智勇, chihyungtseng@hotmail.com

Block Diagram



作者；曾智勇, chihyungtseng@hotmail.com

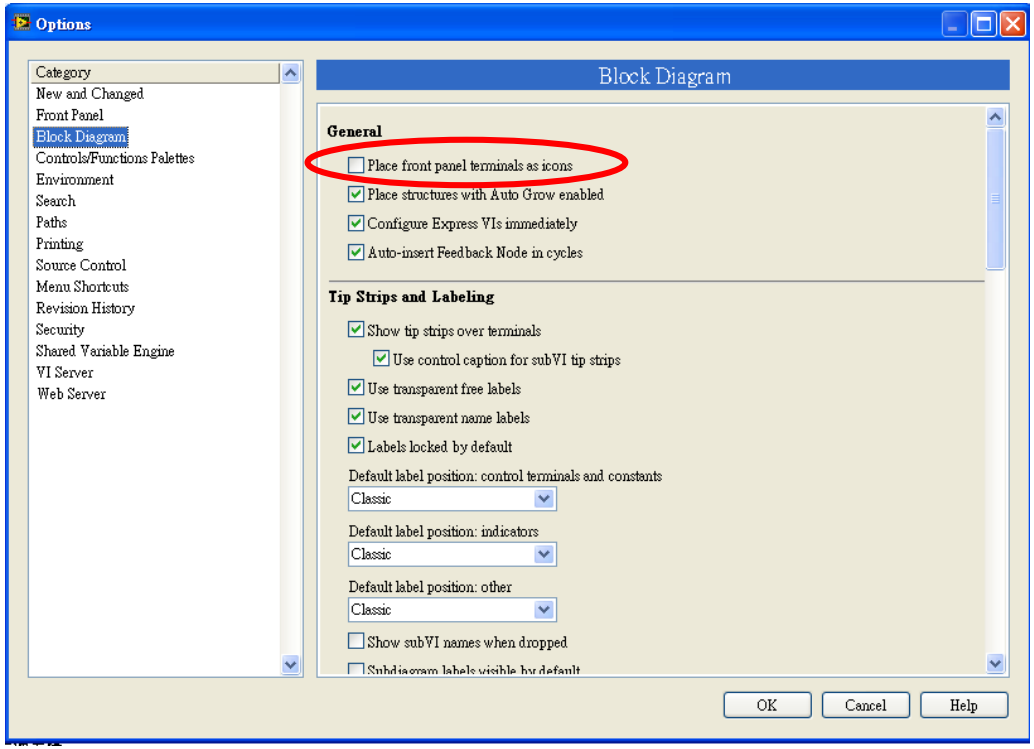
Block Diagram



作者；曾智勇,chihyungtseng@hotmail.com

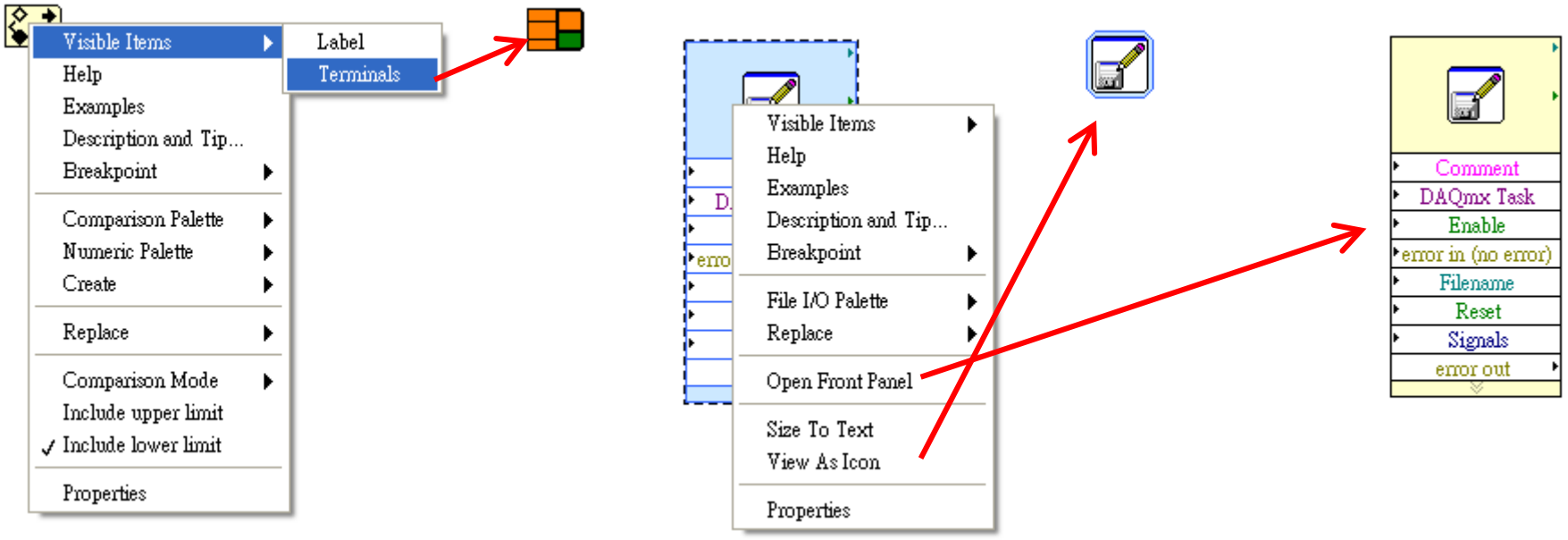
Block Diagram

- Options/ Block Diagram/ General/ □Place front panel terminals as icons：



作者；曾智勇,chihyungtseng@hotmail.com

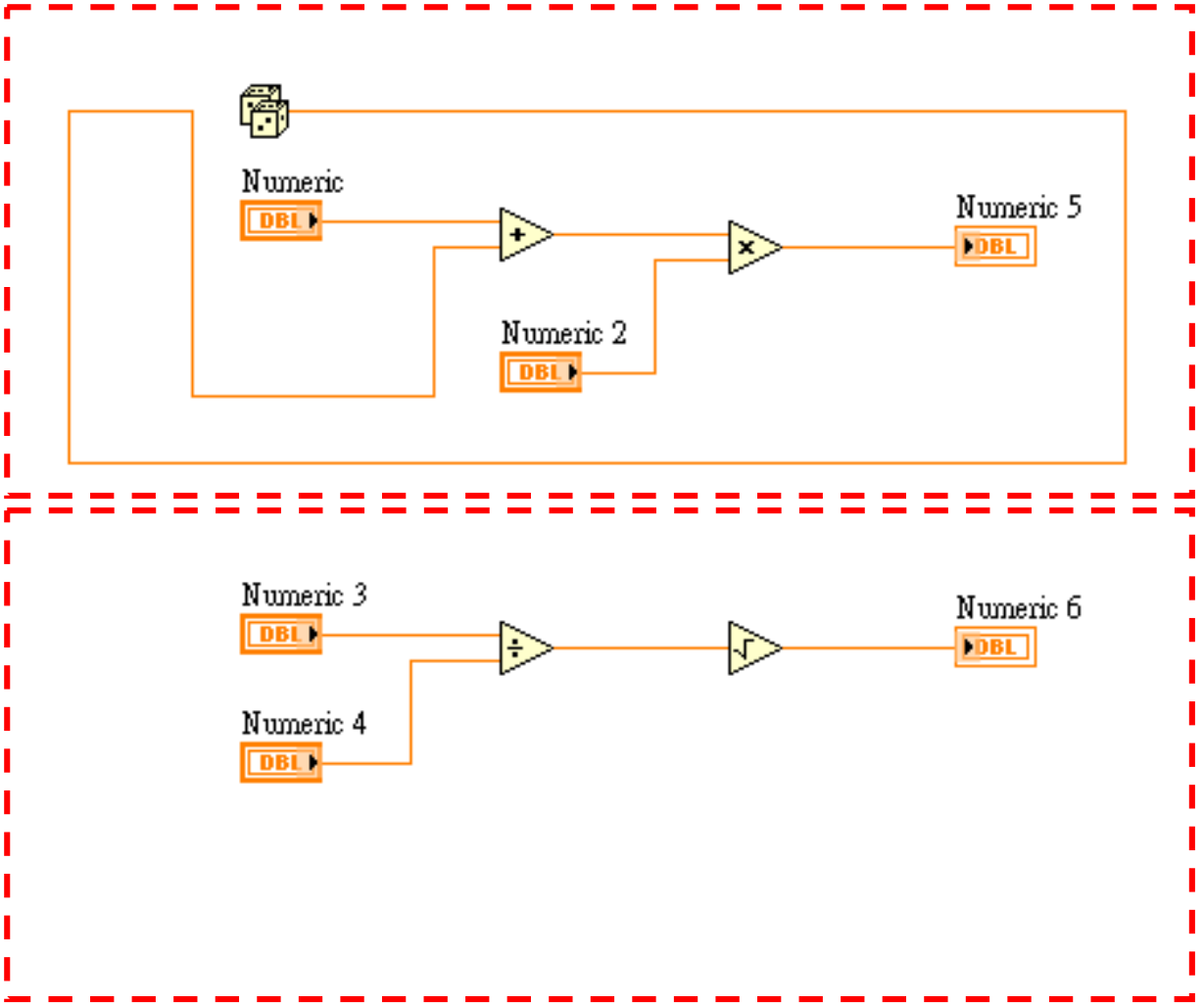
Block Diagram



作者；曾智勇,chihyungtseng@hotmail.com

Block Diagram

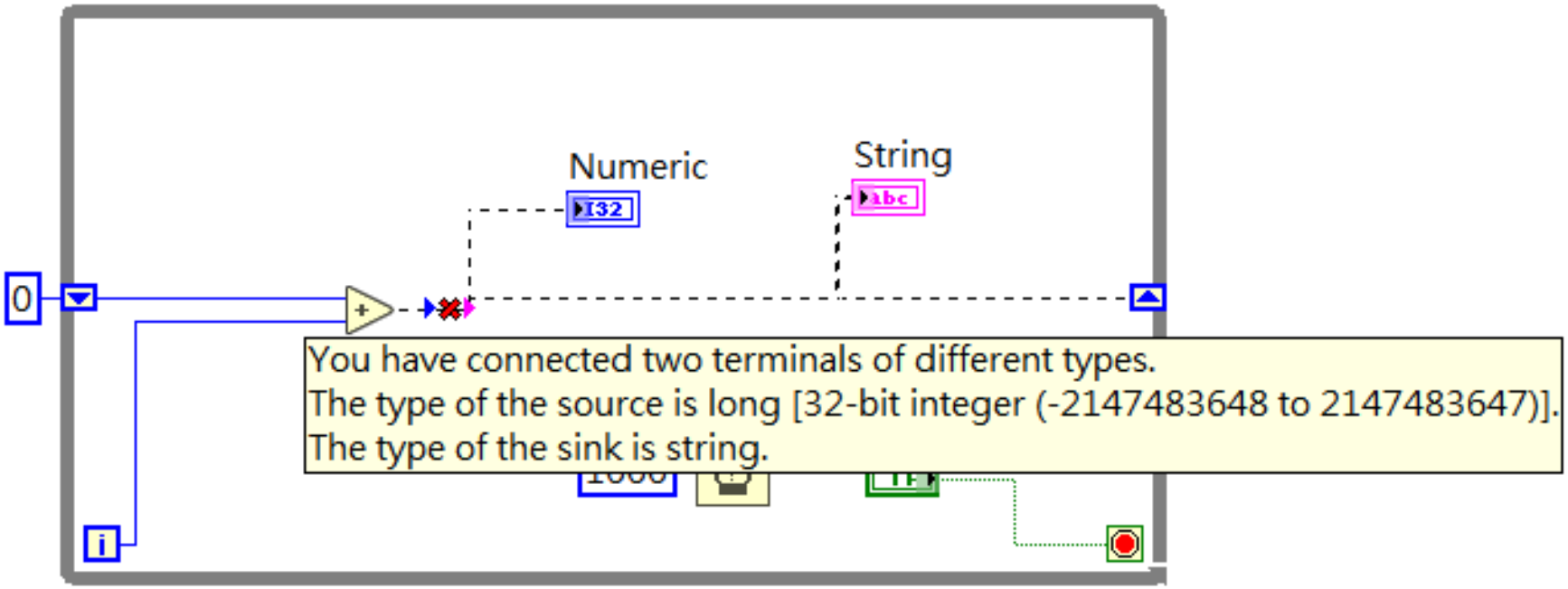
- 資料流：
  - 資料到，就執行
  - 無分上下、左右
  - 與線長無關
  - 多執行緒平行處理



作者；曾智勇,chihyungtseng@hotmail.com

除錯工具

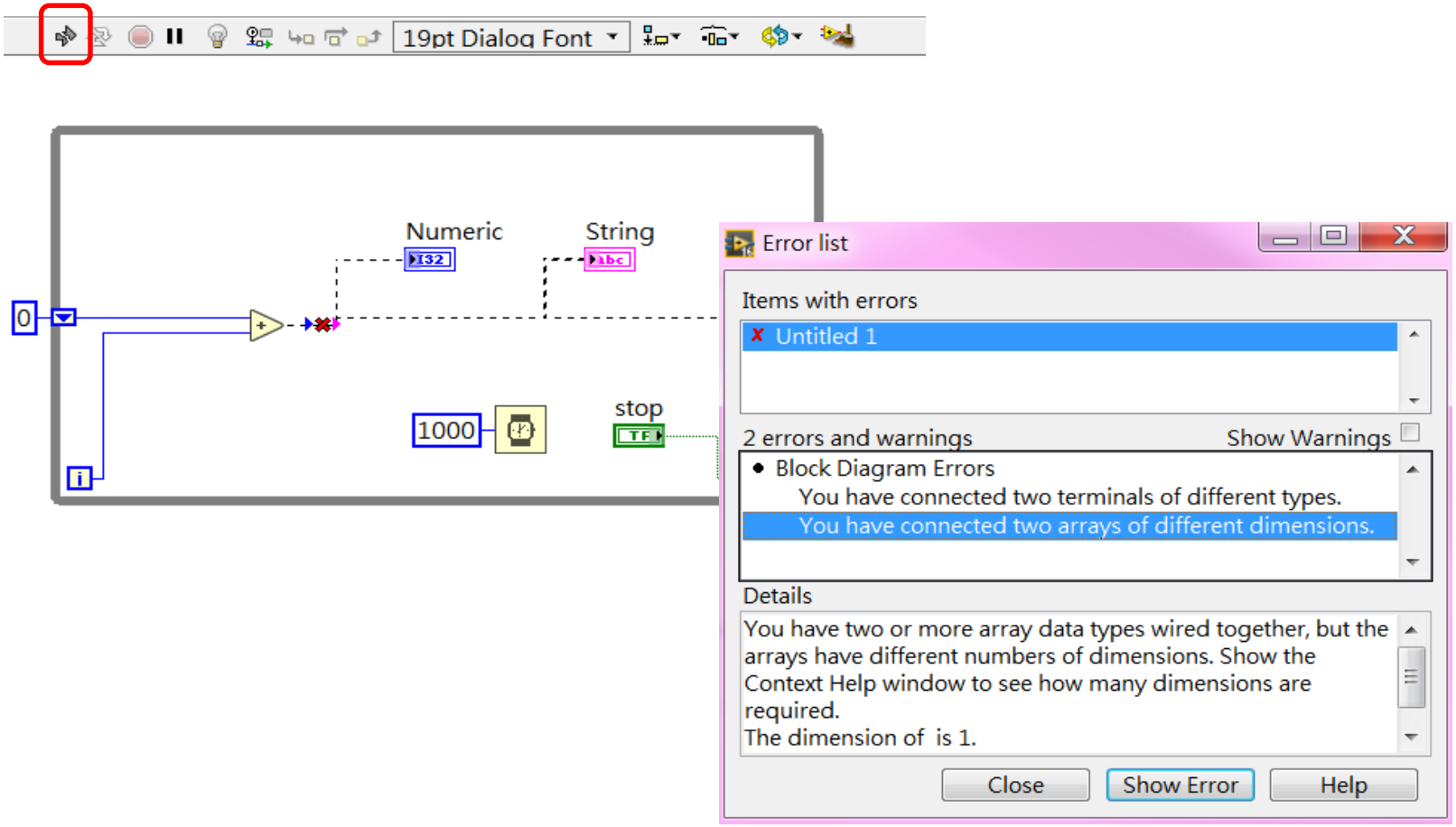
- 錯誤訊息提示，滑鼠游標移動到連接失敗的線上，顯示錯誤訊息提示



作者；曾智勇,chihyungtseng@hotmail.com

除錯工具

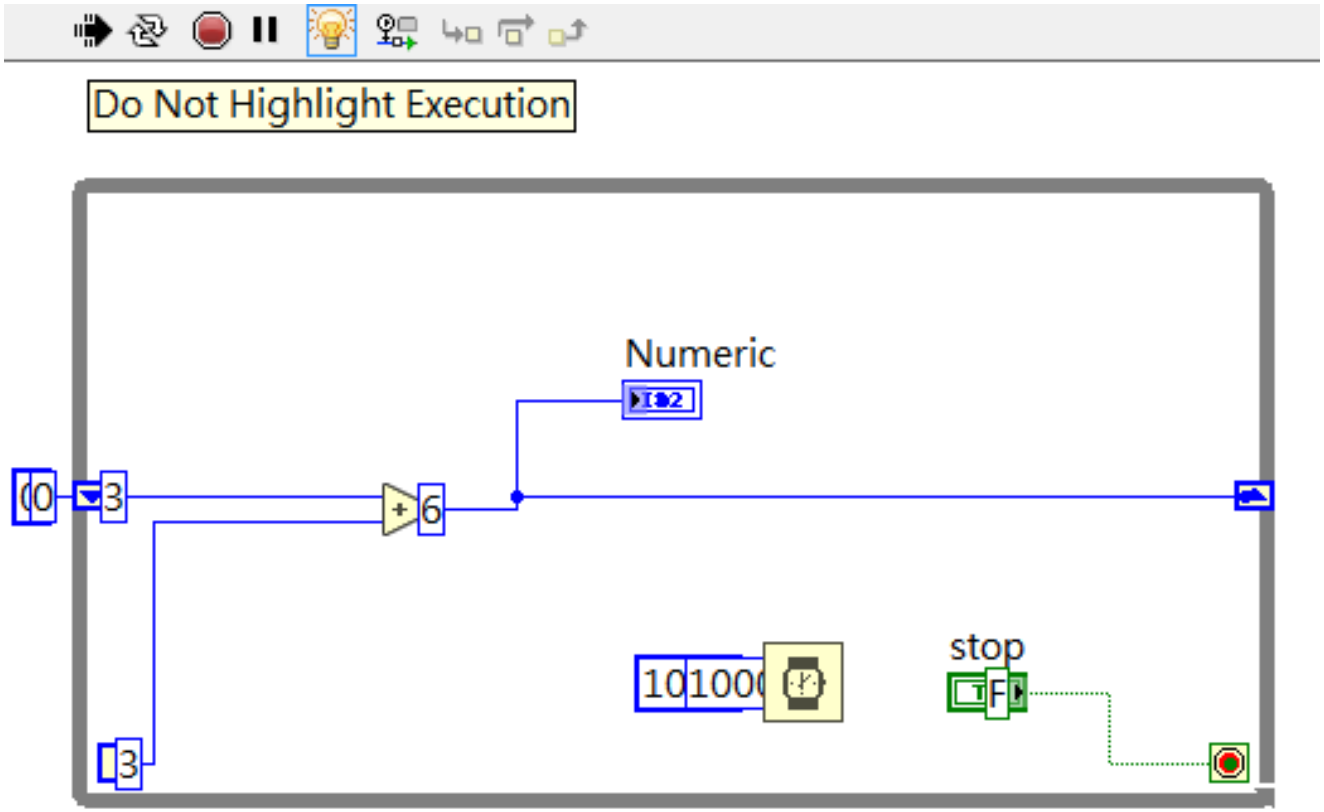
- 點擊破碎箭頭，顯示錯誤位置，了解詳細訊息與修復建議



作者；曾智勇,chihyungtseng@hotmail.com

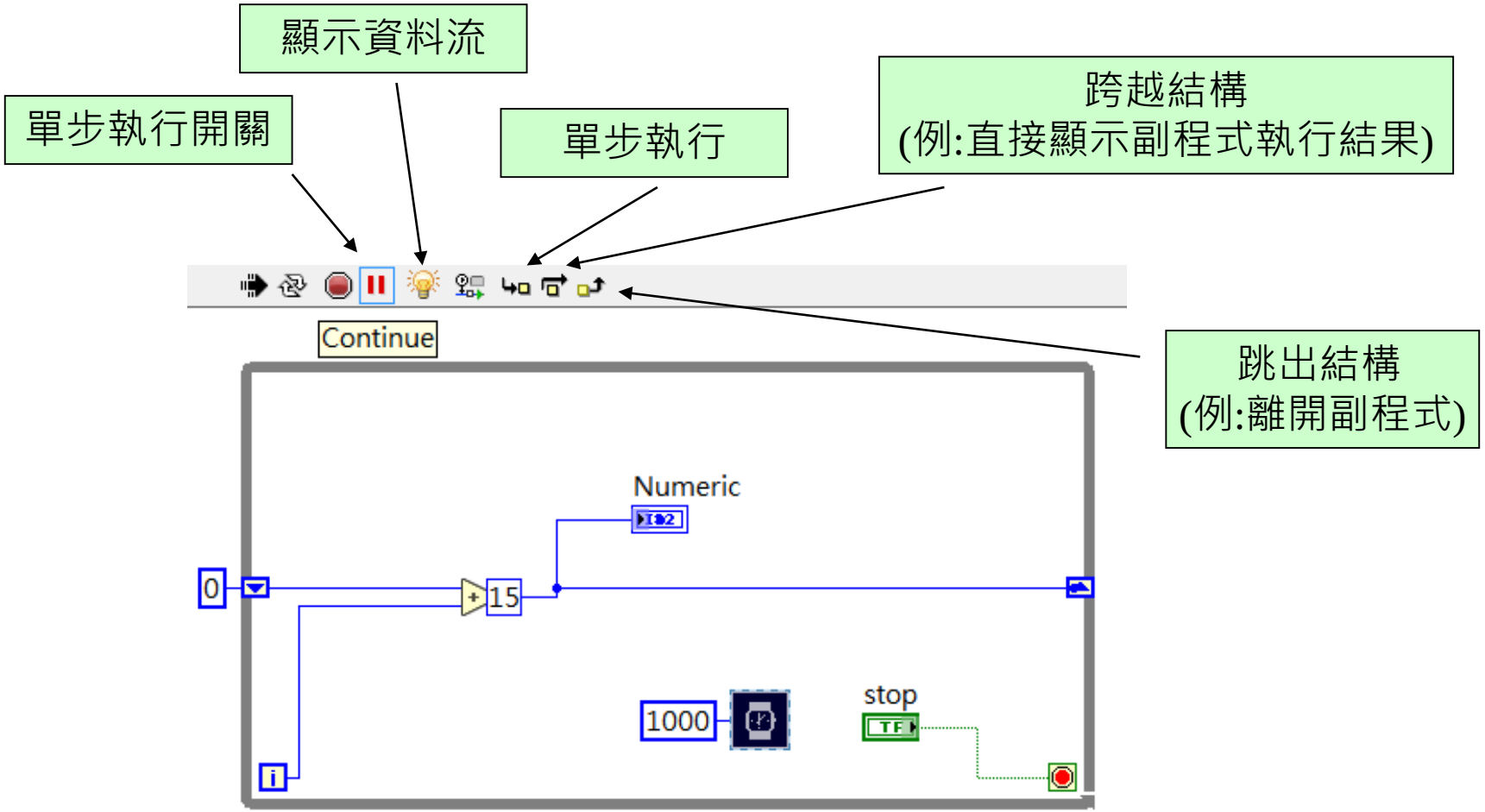
除錯工具

- 點擊燈泡圖示，啟動Highlight Execution，進行緩慢執行功能
  (自動顯示各連接線上的資料流狀態)



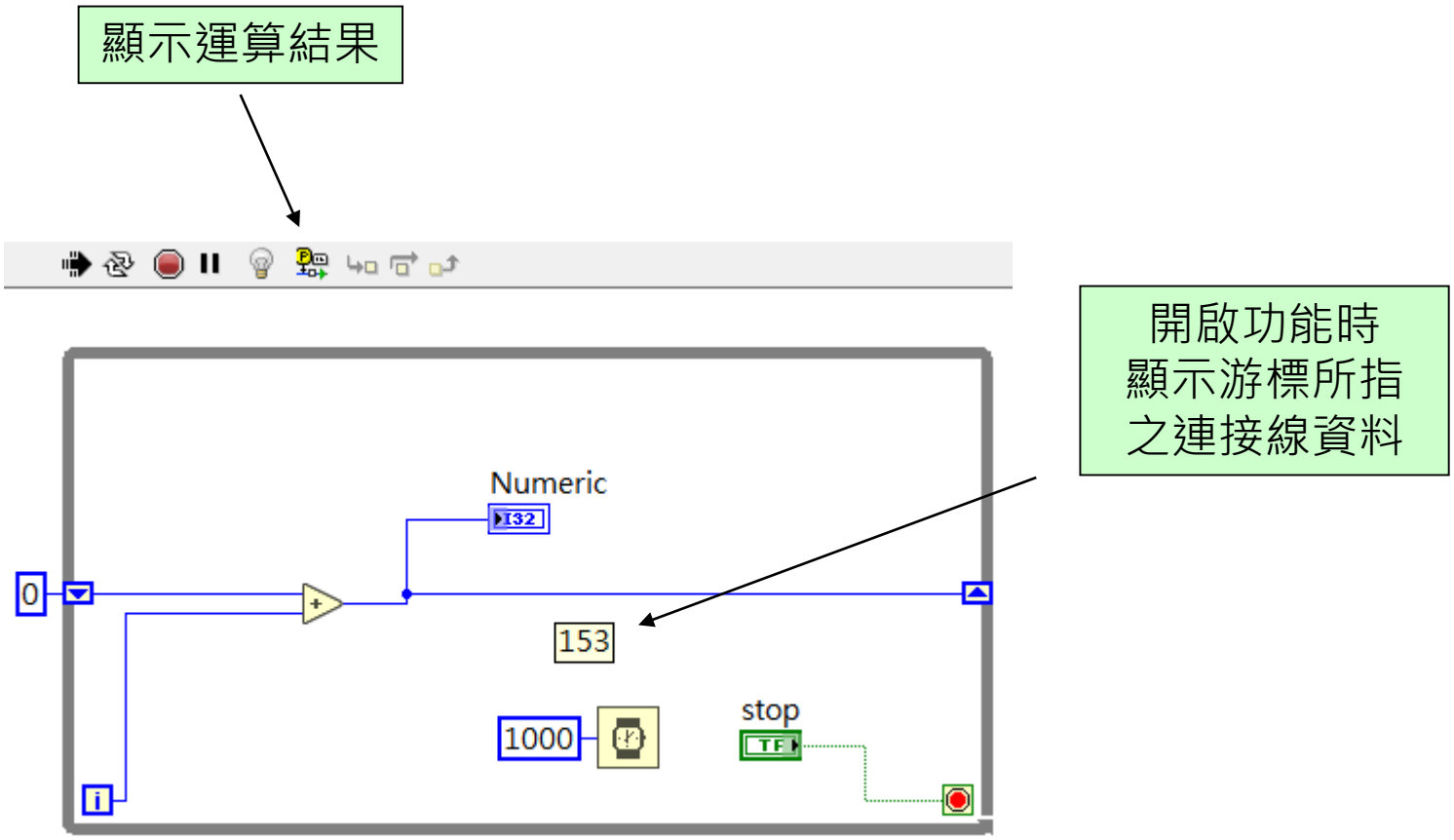
作者；曾智勇,chihyungtseng@hotmail.com

除錯工具



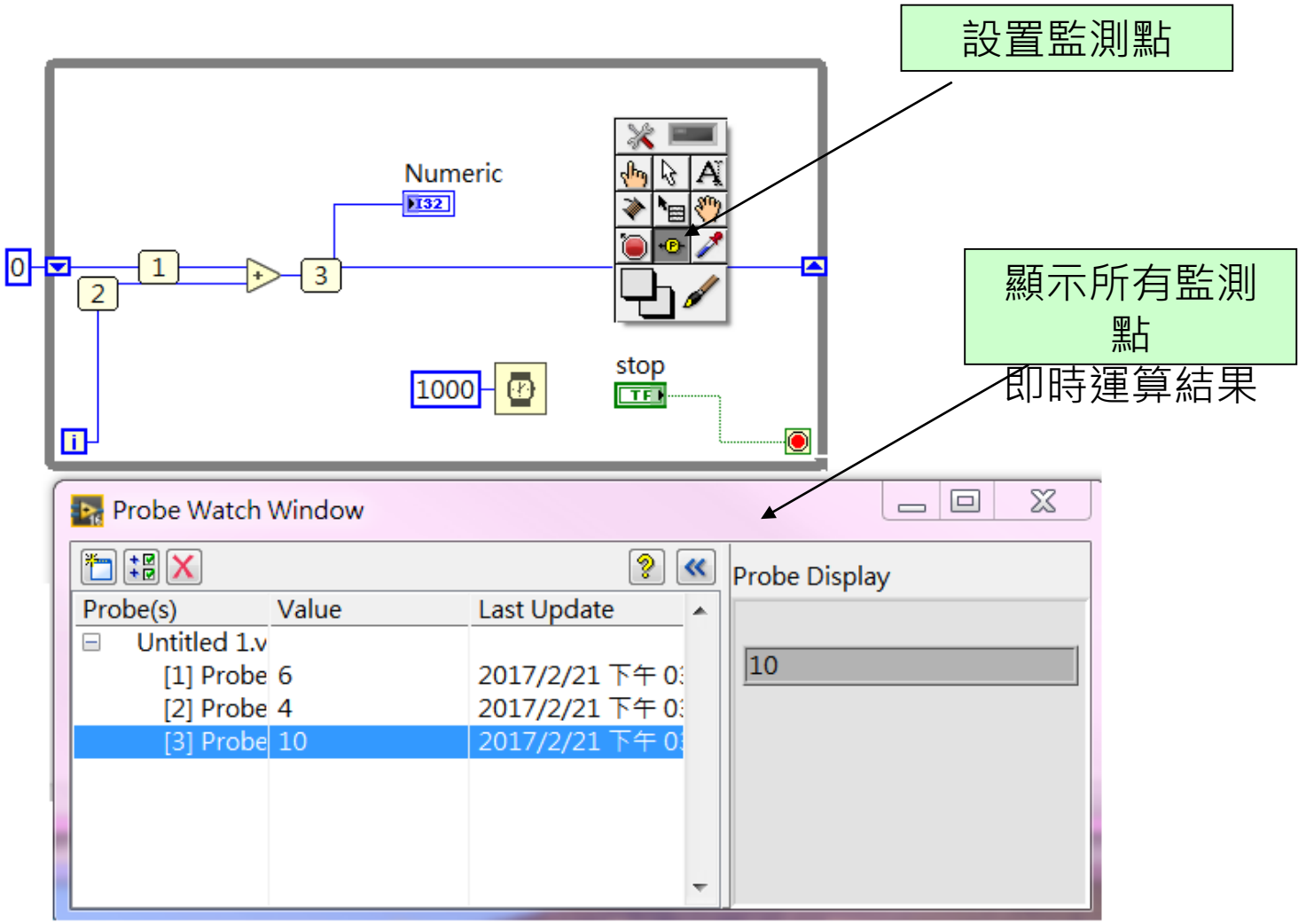
作者；曾智勇, chihyungtseng@hotmail.com

除錯工具



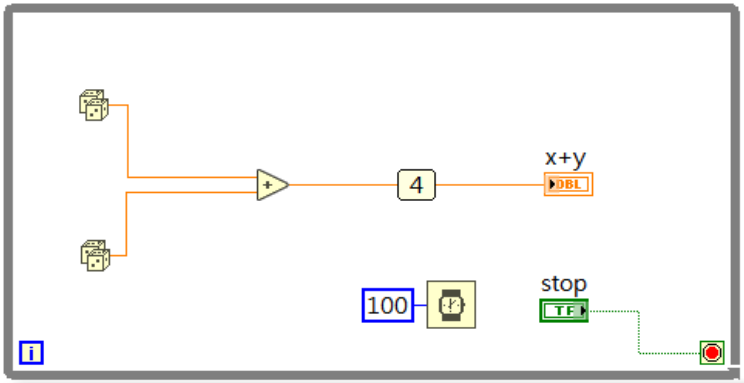
作者；曾智勇, chihyungtseng@hotmail.com

除錯工具

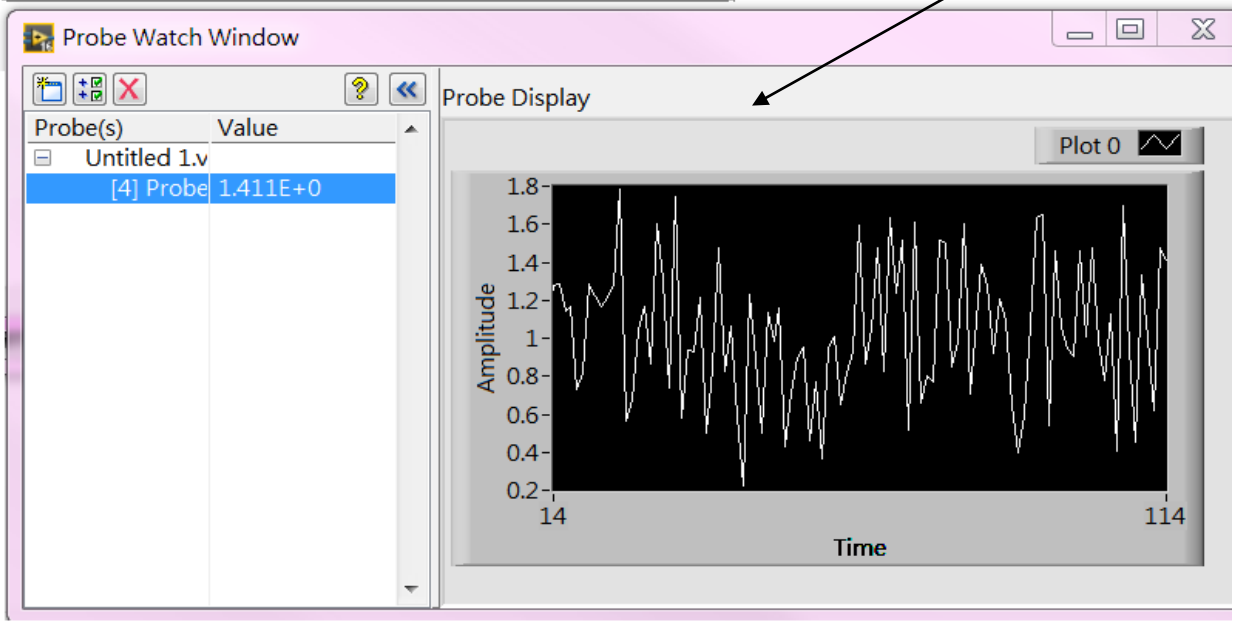


作者；曾智勇, chihyungtseng@hotmail.com

除錯工具



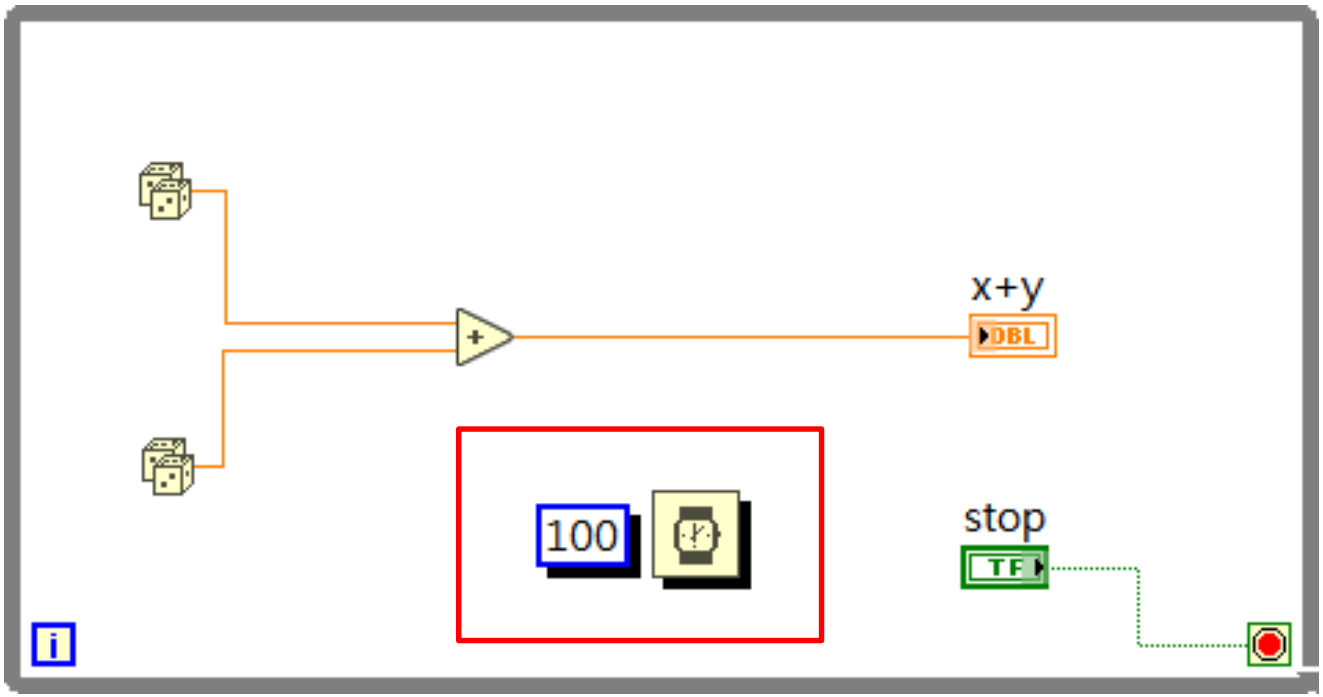
以曲線圖示顯示
即時監測結果



作者；曾智勇,chihyungtseng@hotmail.com

除錯工具

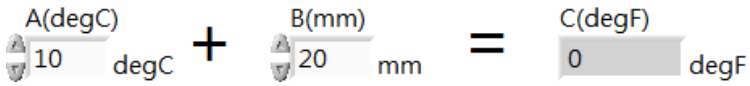
- 陰影，代表物件之間有上下圖層關係，物件未被正確放置於迴圈內

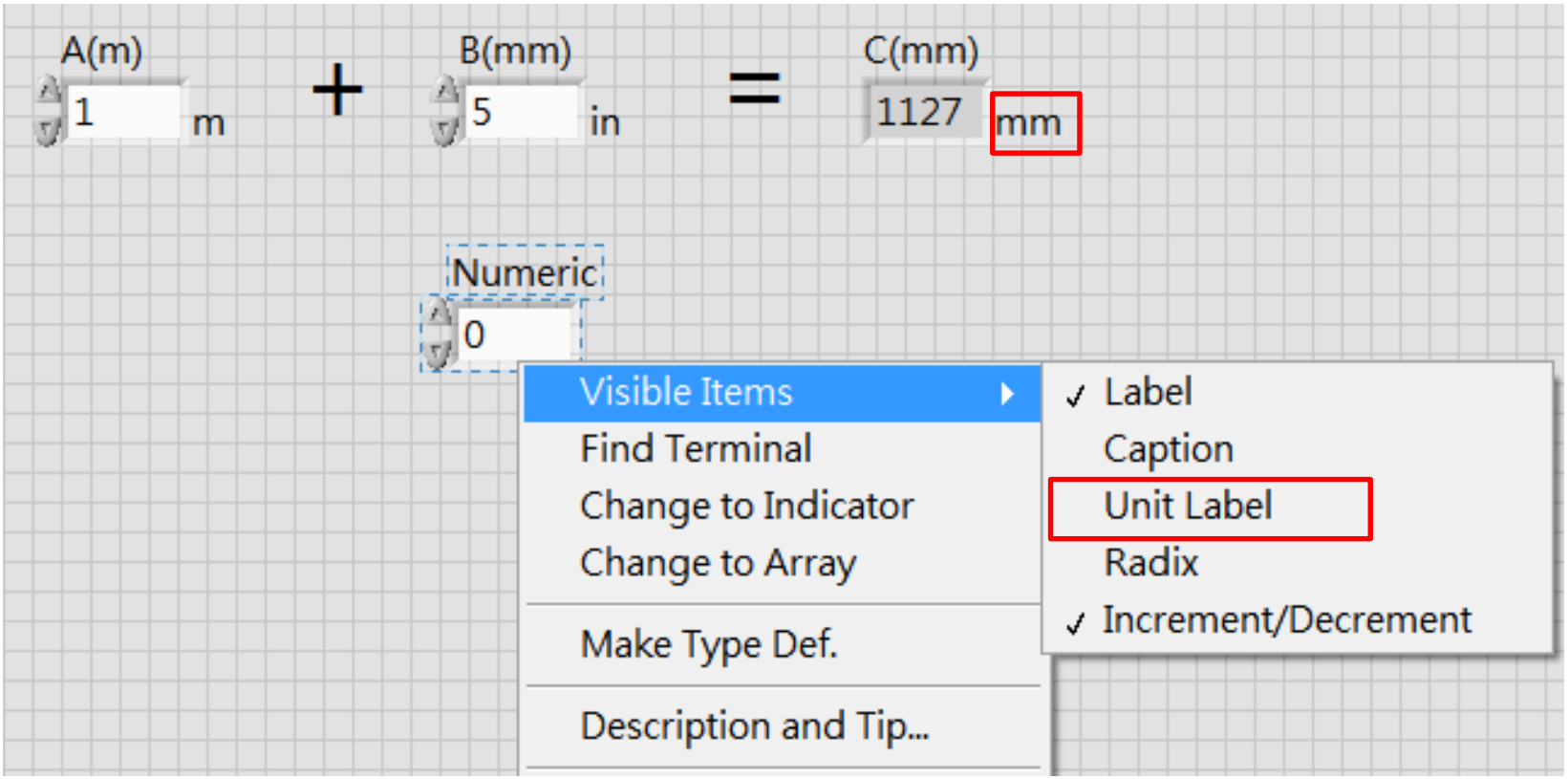


作者；曾智勇,chihyungtseng@hotmail.com

除錯工具

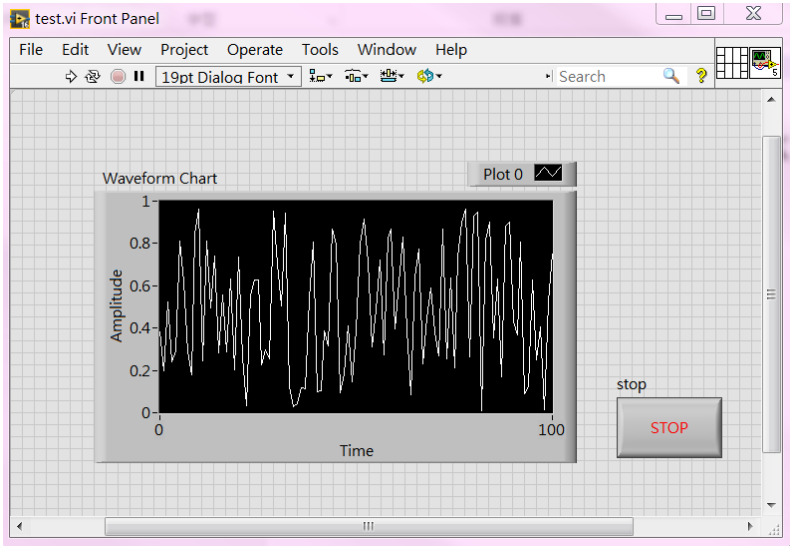
- 單位，數值物件可給定單位，並且程式會自動換算
- 當單位錯誤時程式無法執行，例如：

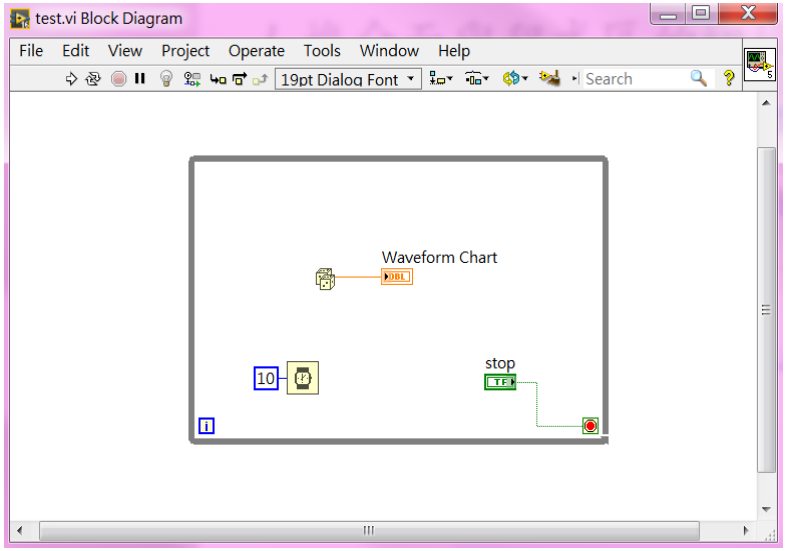




作者；曾智勇, chihyungtseng@hotmail.com

切換 Front Panel / Block Diagram





Ctrl+E：切換人機介面和程式區

作者；曾智勇,chihyungtseng@hotmail.com

常用快速鍵

- Ctrl+N：開新檔
- Ctrl+S：儲存
- Ctrl+X：剪下
- Ctrl+C：複製
- Ctrl+V：貼上
- Ctrl+A：全選
- Ctrl+Space：開啟Quick drop
- Ctrl+U：整理程式區
- Ctrl+B：清除斷線
- Ctrl+R：執行
- Ctrl+.：停止
- Ctrl+M：變更為執行模式
- Ctrl+E：顯示程式區＆介面區
- Ctrl+T：並列顯示

作者；曾智勇,chihyungtseng@hotmail.com

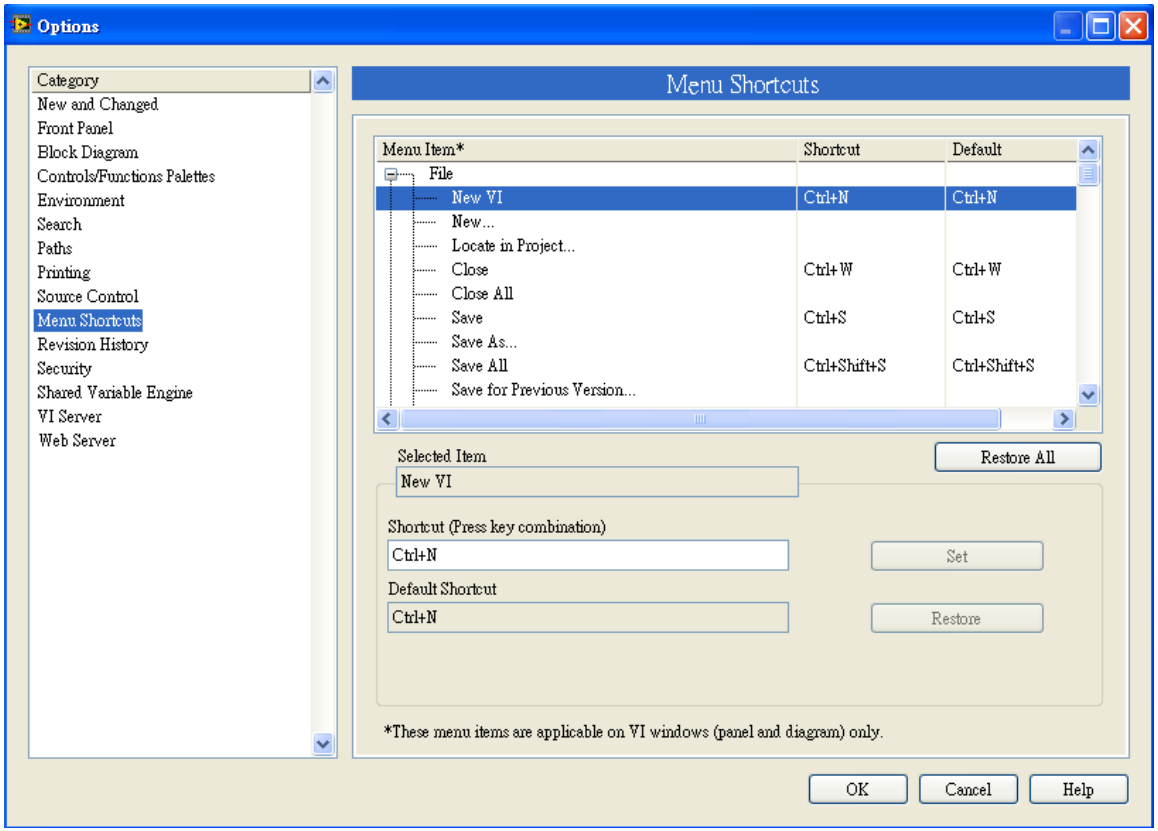
其他快速鍵

- Ctrl+W：關閉檔案
- Ctrl+Shift+S：全部儲存
- Ctrl+P：列印
- Ctrl+I：設定VI
- Ctrl+Q：結束程式
- Ctrl+Z：上一步
- Ctrl+shift+Z：下一步
- Ctrl+#：顯示程式區網格
- Ctrl+shift+W：顯示視窗選單
- Ctrl+H：顯示說明
- Ctrl+shift+L：鎖定說明
- Ctrl+?：呼叫說明文件
- Ctrl+/：最大化
- Ctrl+Shift+A：靠左貼齊(2個以上)
- Ctrl+D：平均間隔(3個以上)
- Ctrl+Y：紀錄
- Ctrl+F：尋找
- Ctrl+shift+F：顯示搜尋結果
- Ctrl+L：錯誤清單
- Ctrl+shift+E：開啟所屬專案
- Ctrl+shift+B：搜尋屬性/方法節點
- Ctrl+shift+N：程式導航
- Ctrl+↓：進入副程式
- Ctrl+→：跨過副程式
- Ctrl+↑：跳出副程式
- ……

作者；曾智勇,chihyungtseng@hotmail.com

快速鍵

- 可自定義快速鍵，Options/ Menu Shortcuts/ Menu Item：



作者；曾智勇,chihyungtseng@hotmail.com

特殊熱鍵

- **Shift+滑鼠右鍵：呼叫工具箱(Tool Palette)**
- **Shift+滑鼠左鍵：物件多選**
- 選擇物件+(↑→↓←)：細調位置
- 選擇物件+Shift+(↑→↓←)：粗調位置
- Tool Palette-Auto取消時，shift可連續切換功能
- 人機介面區時，

程式區時：

- **Ctrl+滑鼠左鍵+拖曳：複製物件於拖曳位置**
- **Ctrl+shift+滑鼠左鍵+拖曳：複製物件於平行位置**
- Ctrl+=：字體放大
- Ctrl+-：字體縮小

作者；曾智勇,chihyungtseng@hotmail.com

工具面板(Tool Palette)


自動選擇工具


點選


選取


修改文字


連線


開啟捷徑選單


捲動視窗


設定中斷點


建立資料偵測點

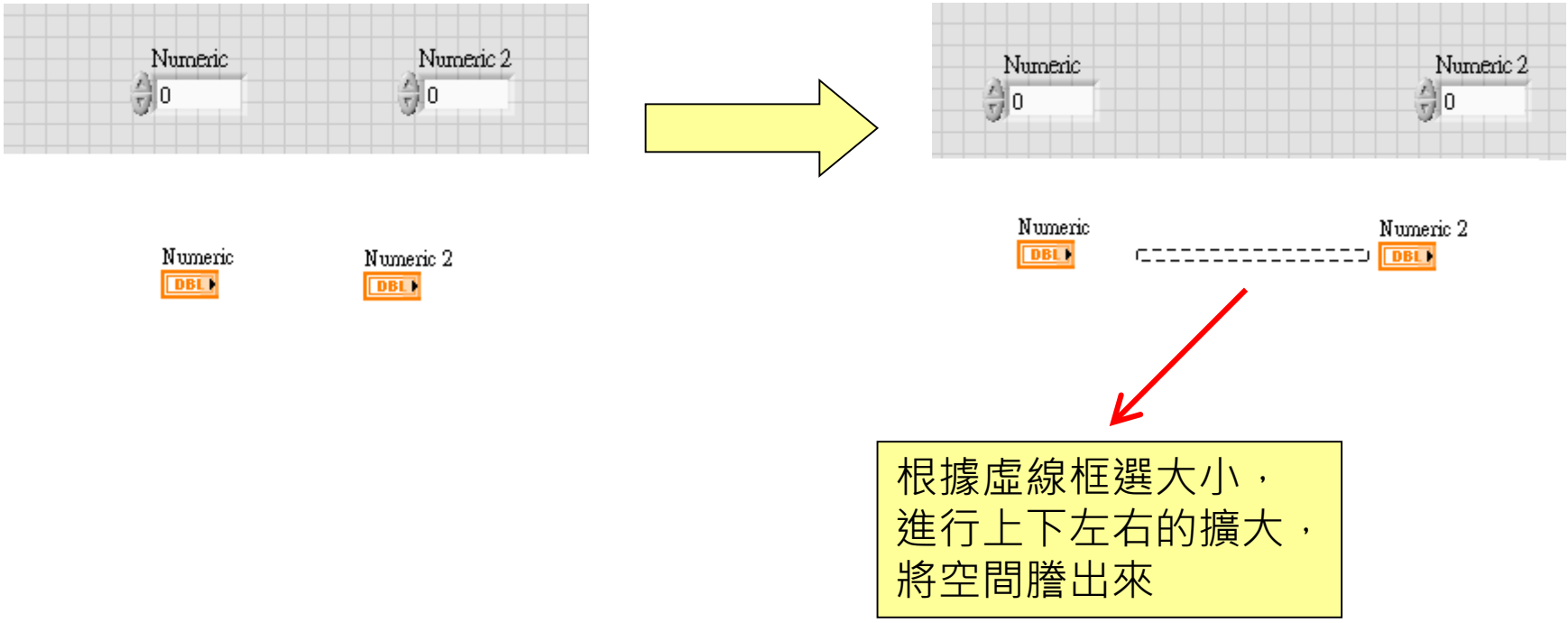

複製顏色


設定顏色

作者；曾智勇, chihyungtseng@hotmail.com

特殊熱鍵

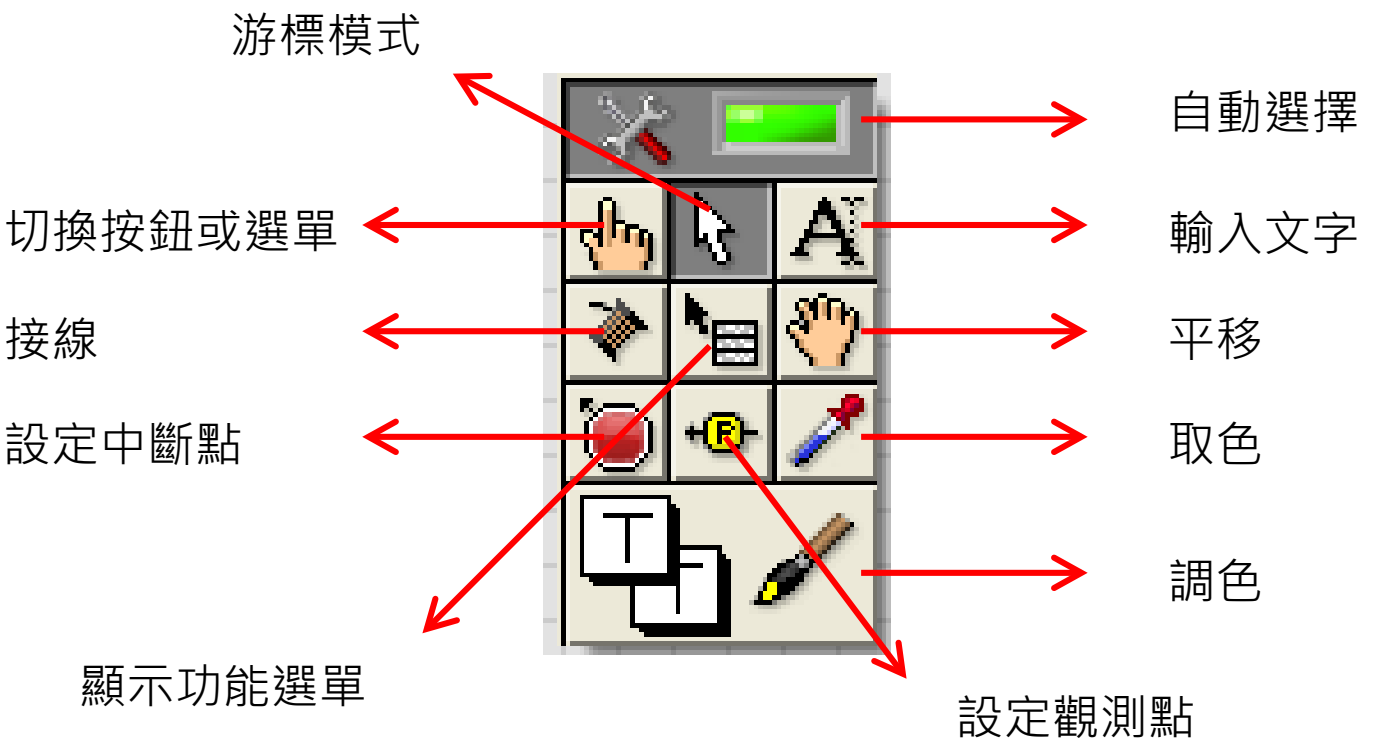
- 在人機介面區或程式區，空白處+Ctrl+滑鼠左鍵+拖曳：擴大區域
- 空白處+Ctrl+ALT+滑鼠左鍵+拖曳：縮小區域(2015版以後才有此功能)



作者；曾智勇,chihyungtseng@hotmail.com

Tools Palette

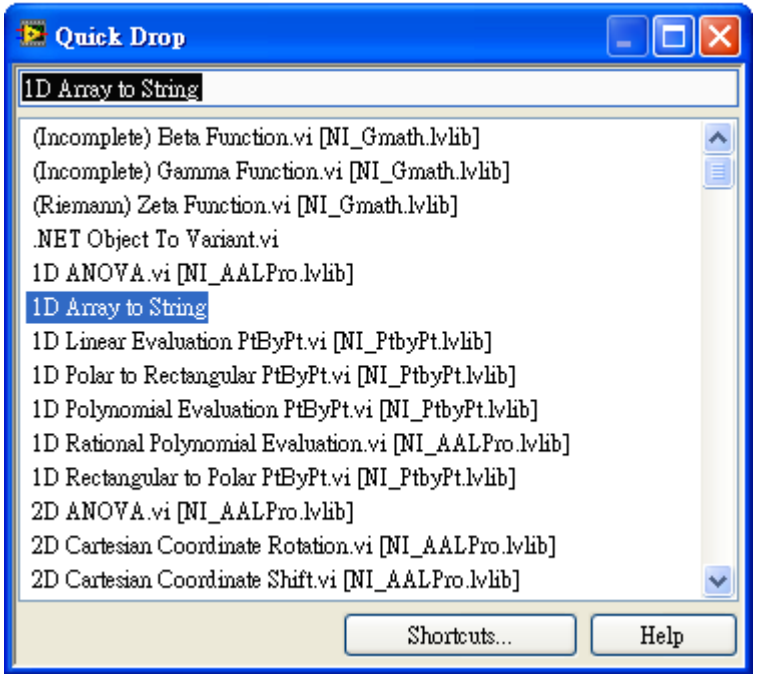
- Menu/ View/ Tool Palette
- Shift+滑鼠右鍵：呼叫工具箱(Tool Palette)
- Options/ Menu Shortcuts/ Menu Item/ Tool Palette：自定義熱鍵



作者；曾智勇, chihyungtseng@hotmail.com

Quick drop

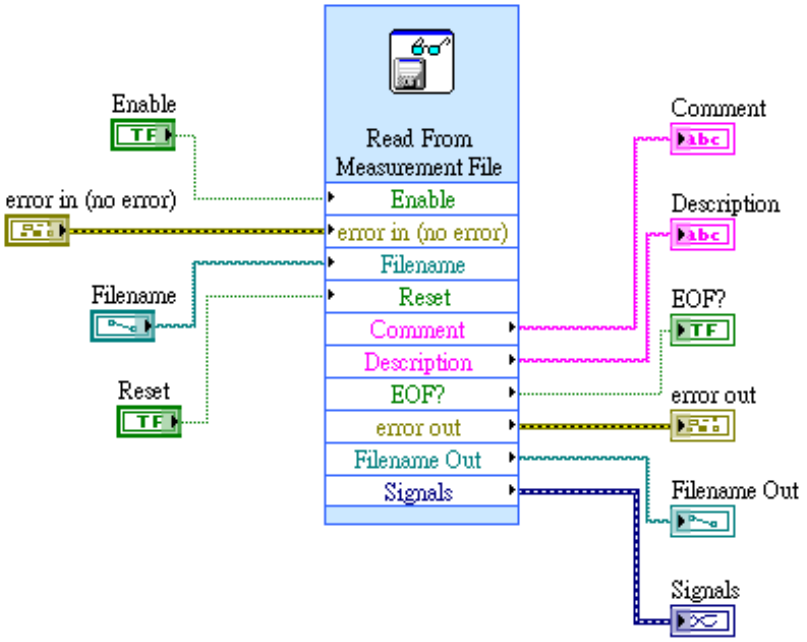
- Menu/ View/ Quick drop
- Ctrl+Space：呼叫搜尋選單(Quick drop)
- Options/ Menu Shortcuts/ Menu Item/ Quick drop ：自定義熱鍵



作者；曾智勇,chihyungtseng@hotmail.com

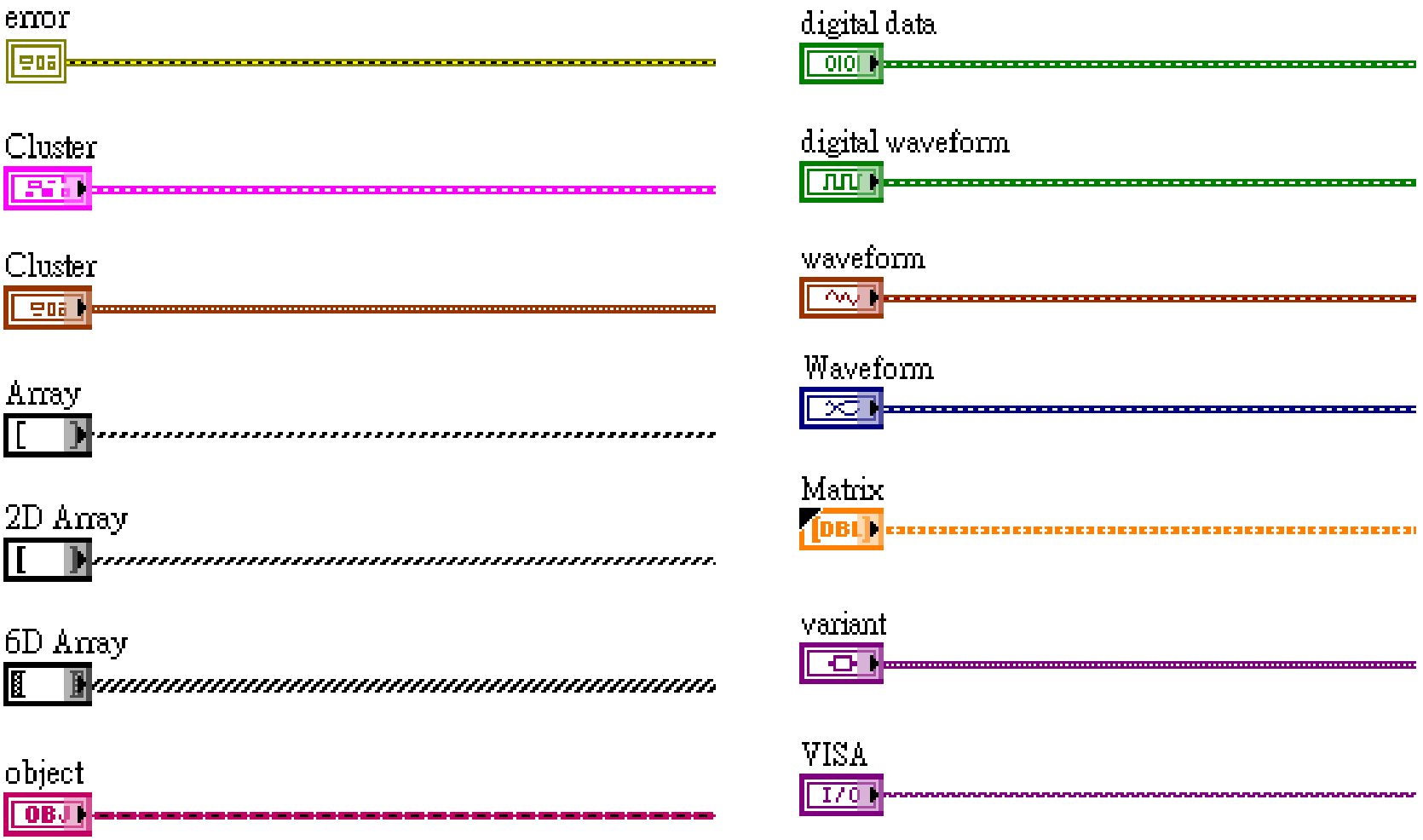
Quick drop

- 特殊功能：
  - 點選物件後，
  - 再開啟Quick drop，
  - 再按Ctrl+D，可自動產生輸入與輸出。
  - 再按Ctrl+W，可自動接線。
  - **可自訂副程式與熱鍵，執行自訂運作事項(VI Scripting)。**



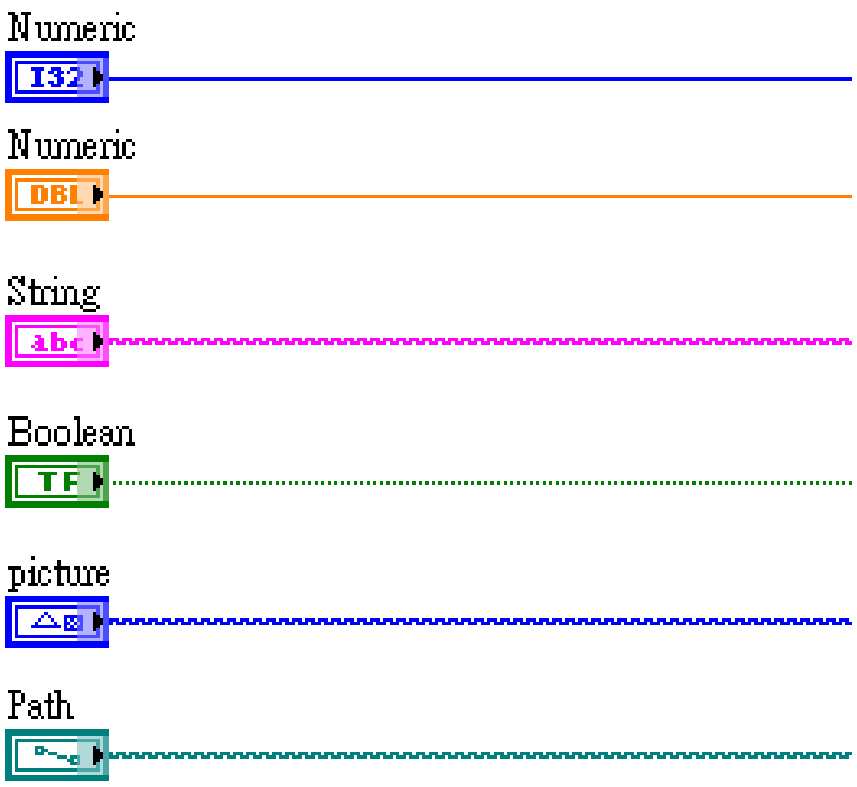
曾智勇, chihyungtseng@hotmail.com

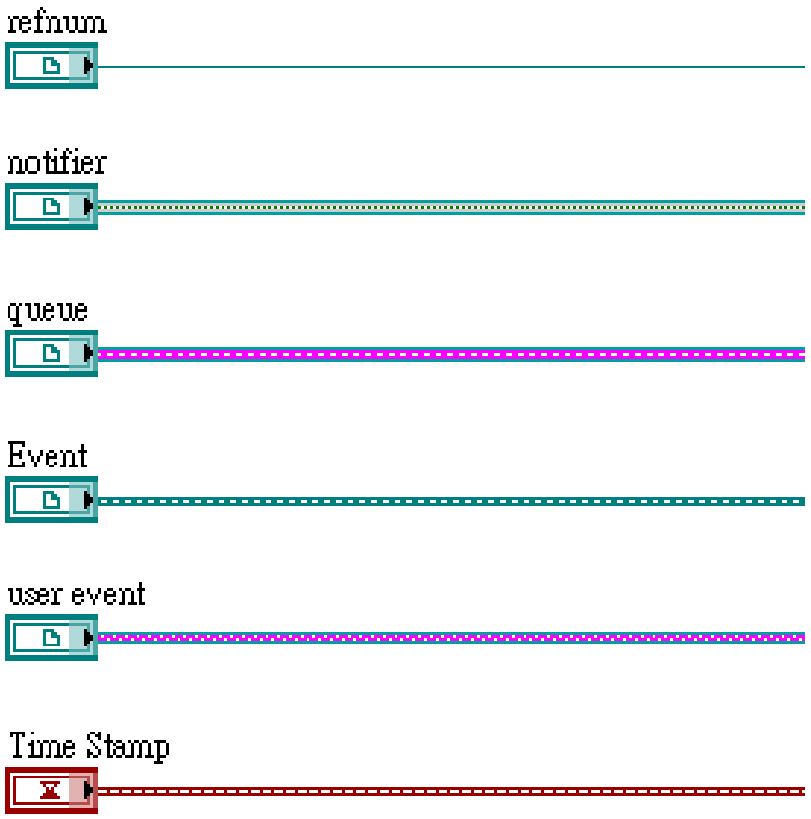
資料型態



作者；曾智勇, chihyungtseng@hotmail.com

資料型態





作者；曾智勇,chihyungtseng@hotmail.com

資料型態

- 資料連接線分為三種形式，不同粗細代表不同維度
- 2個維度以上，資料線粗細固定，不再增加

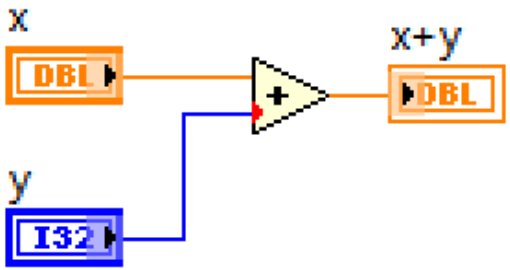
Wire Type	Scalar	1D Array	2D Array	Color
Numeric				Orange (floating-point), Blue (integer)
Boolean				Green
String				Pink

作者；曾智勇,chihyungtseng@hotmail.com

資料型態

- 當資料型態不符 / 接線過程中斷，線會變成虛線，並有紅色“X”且白色運行箭頭變成破碎的灰色箭頭

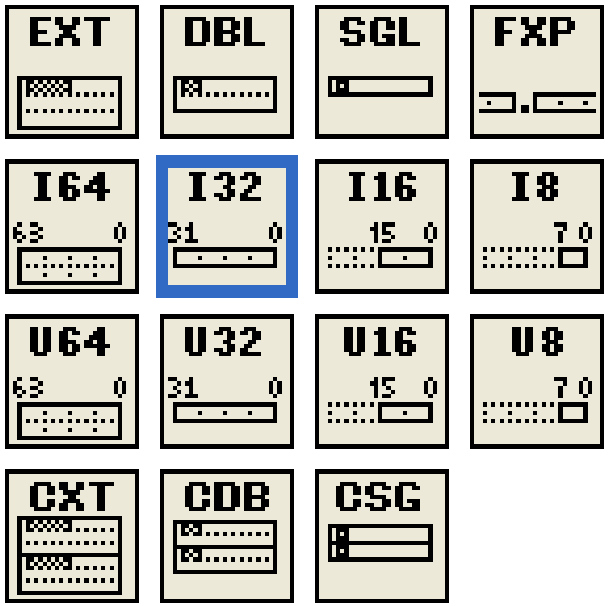
- 當資料型態不符，但仍可接線，則會出現強制轉換點(coercion dot)代表資料經過自動的強制型別轉換

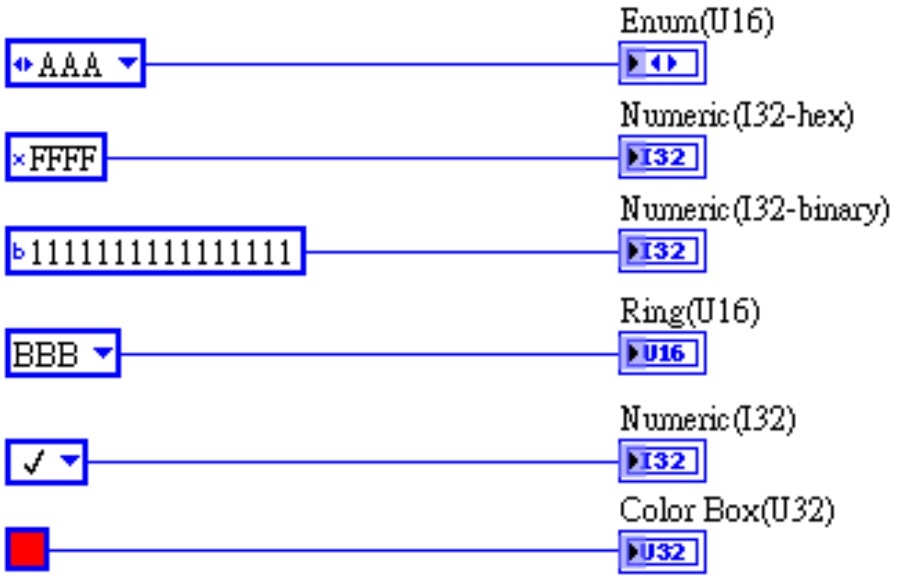


作者；曾智勇,chihyungtseng@hotmail.com

數值

- 數值：在數值元件上點滑鼠右鍵可切換成各種數值資料型態
- 數值可能為各種顯示類型，選單、符號、顏色，不限定以數字顯示





作者；曾智勇, chihyungtseng@hotmail.com

數值

- 數值：

Terminal	Numeric Data Type	Bits of Storage on Disk	Approximate Number of Decimal Digits	Approximate Range
SGL	Single-precision, floating-point	32	6	Minimum positive number: 1.40e-45 Maximum positive number: 3.40e+38 Minimum negative number: -1.40e-45 Maximum negative number: -3.40e+38
DBL	Double-precision, floating-point	64	15	Minimum positive number: 4.94e-324 Maximum positive number: 1.79e+308 Minimum negative number: -4.94e-324 Maximum negative number: -1.79e+308
EXT	Extended-precision, floating-point	128	varies from 15 to 20 by platform	Minimum positive number: 6.48e-4966 Maximum positive number: 1.19e+4932 Minimum negative number: -6.48e-4966 Maximum negative number: -1.19e+4932
CSG	Complex single-precision, floating-point	64	6	Same as single-precision, floating-point for each (real and imaginary) part
CDB	Complex double-precision, floating-point	128	15	Same as double-precision, floating-point for each (real and imaginary) part
CXT	Complex extended-precision, floating-point	256	varies from 15 to 20 by platform	Same as extended-precision, floating-point for each (real and imaginary) part

作者；曾智勇,chihyungtseng@hotmail.com

數值

- 數值：

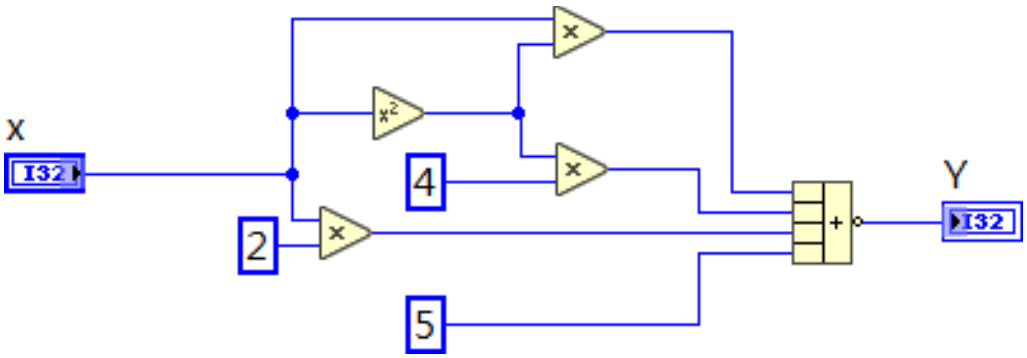
FXP	Fixed-point	64, or 72 if you include an overflow status	varies by user configuration	varies by user configuration
I8	Byte signed integer	8	2	-128 to 127
I16	Word signed integer	16	4	-32,768 to 32,767
I32	Long signed integer	32	9	-2,147,483,648 to 2,147,483,647
I64	Quad signed integer	64	18	-1e19 to 1e19
U8	Byte unsigned integer	8	2	0 to 255
U16	Word unsigned integer	16	4	0 to 65,535
U32	Long unsigned integer	32	9	0 to 4,294,967,295
U64	Quad unsigned integer	64	19	0 to 2e19
X	128-bit time stamp	128	19	Minimum time: 01/01/1600 00:00:00 UTC maximum time: 01/01/3001 00:00:00 UTC

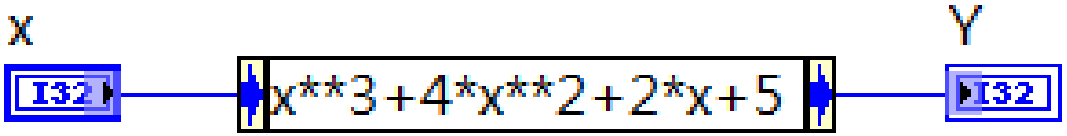
作者；曾智勇,chihyungtseng@hotmail.com

數值

- 範例：

$$Y=x^3+4x^2+2x+5$$

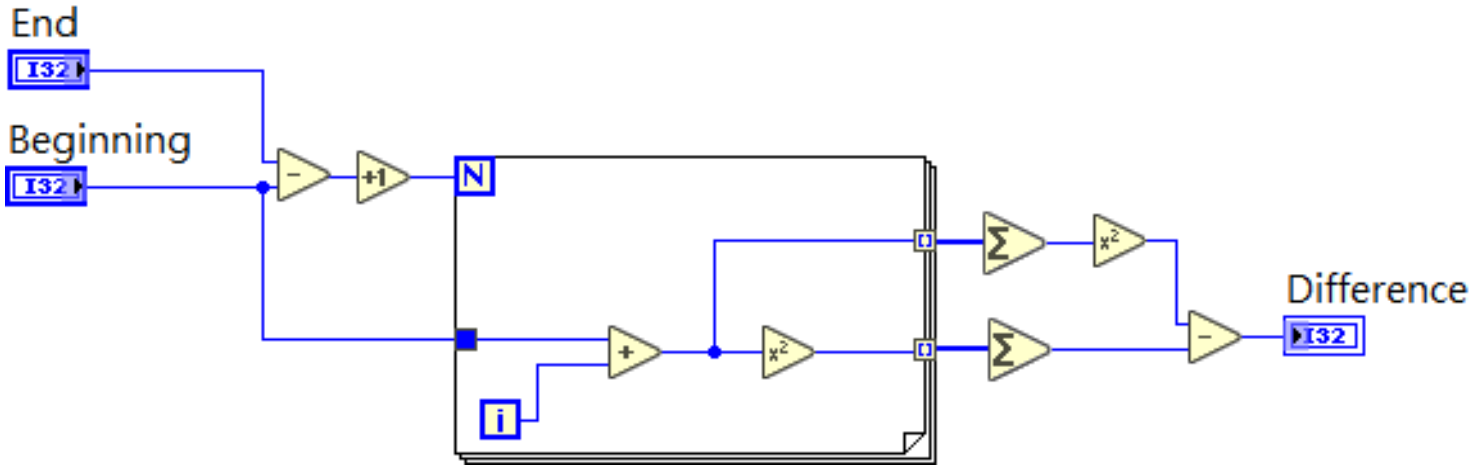




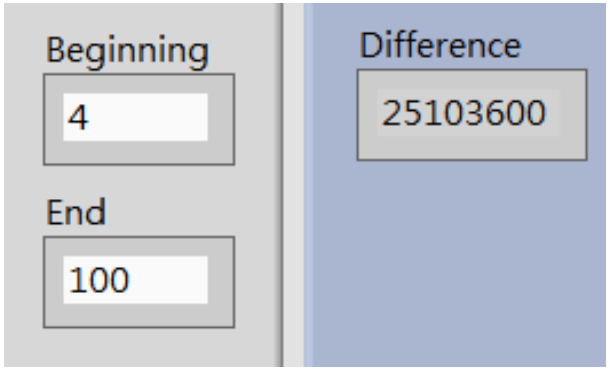
作者；曾智勇,chihyungtseng@hotmail.com

數值

- 範例：總和之平方 減去 平方之總和 $(0+1+2)^2-(0^2+1^2+2^2)$



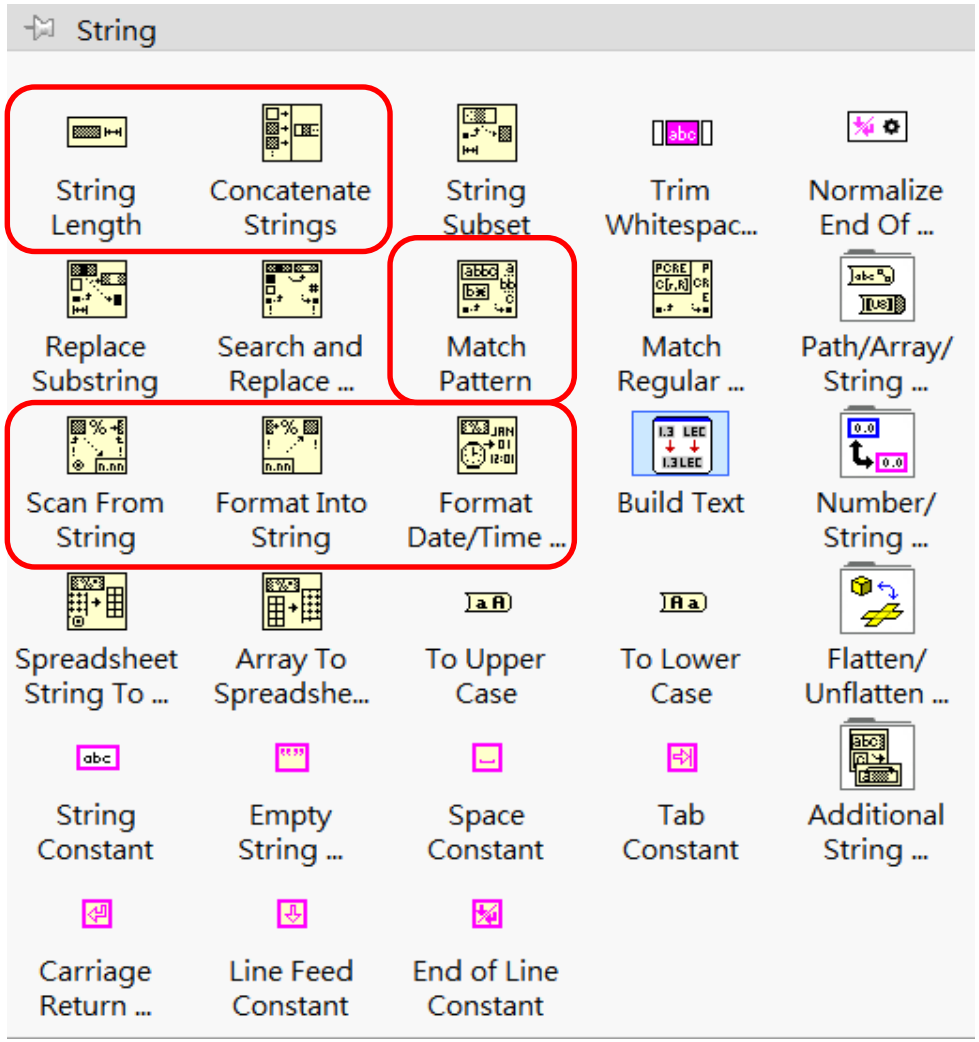
Beginning	End	Difference
0	2	4
-5	-4	40
10	13	1582
4	100	25103600



作者；曾智勇, chihyungtseng@hotmail.com

字串

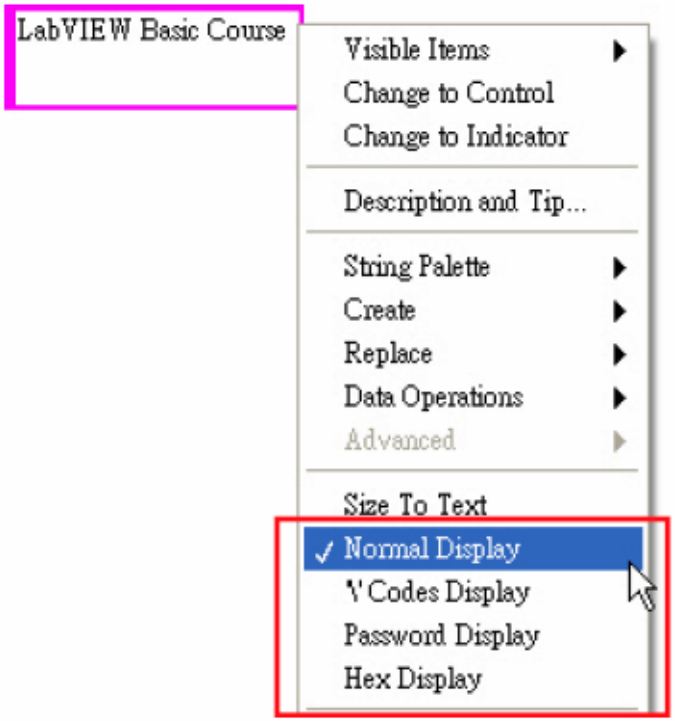
- 常用元件：

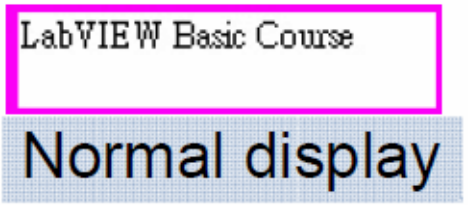


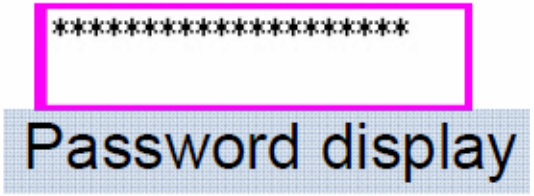
作者；曾智勇,chihyungtseng@hotmail.com

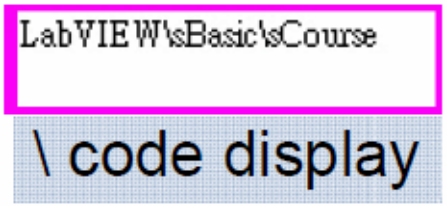
字串

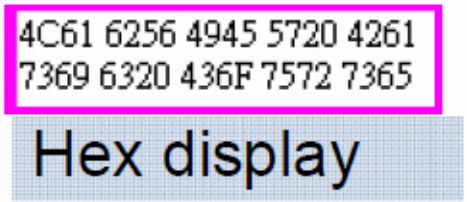
- 四種顯示模式：











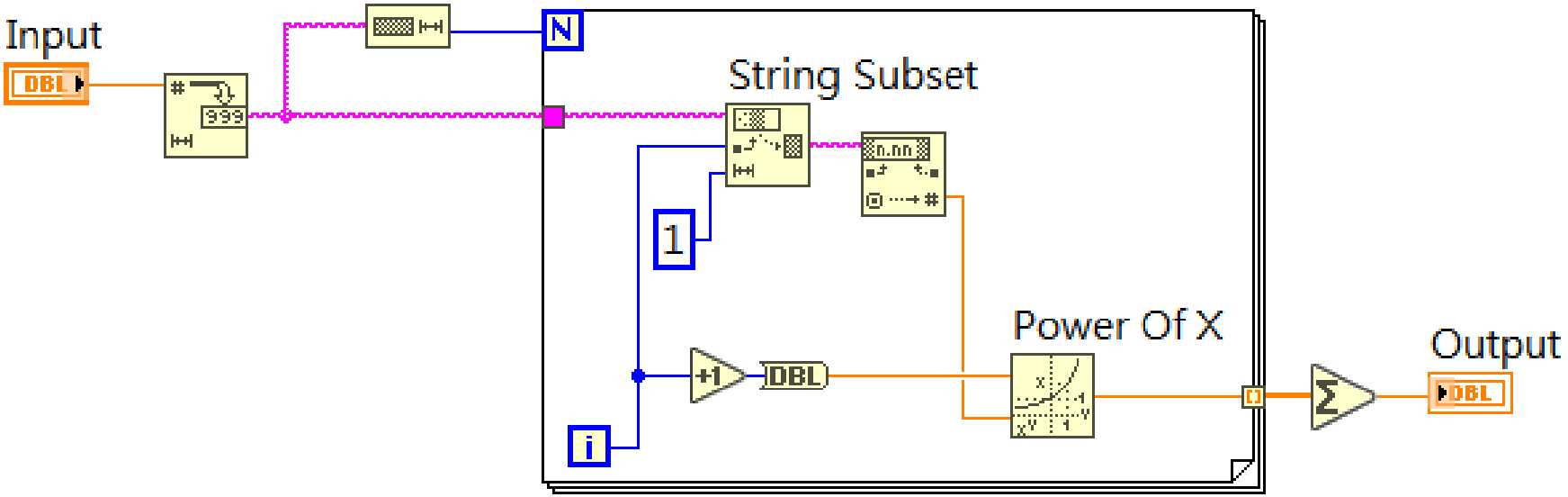
作者；曾智勇,chihyungtseng@hotmail.com

字串

- 範例：文字轉數值並進行運算

$678 \rightarrow 6 + 7^2 + 8^3$

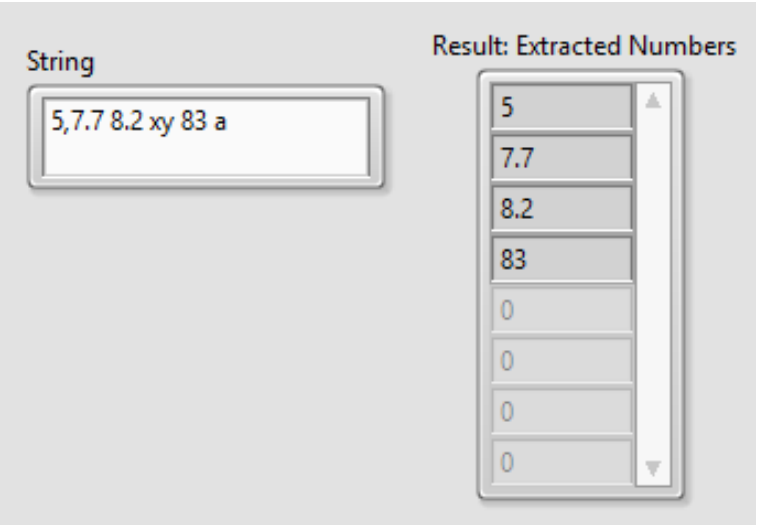
$3956 \rightarrow 3 + 9^2 + 5^3 + 6^4$

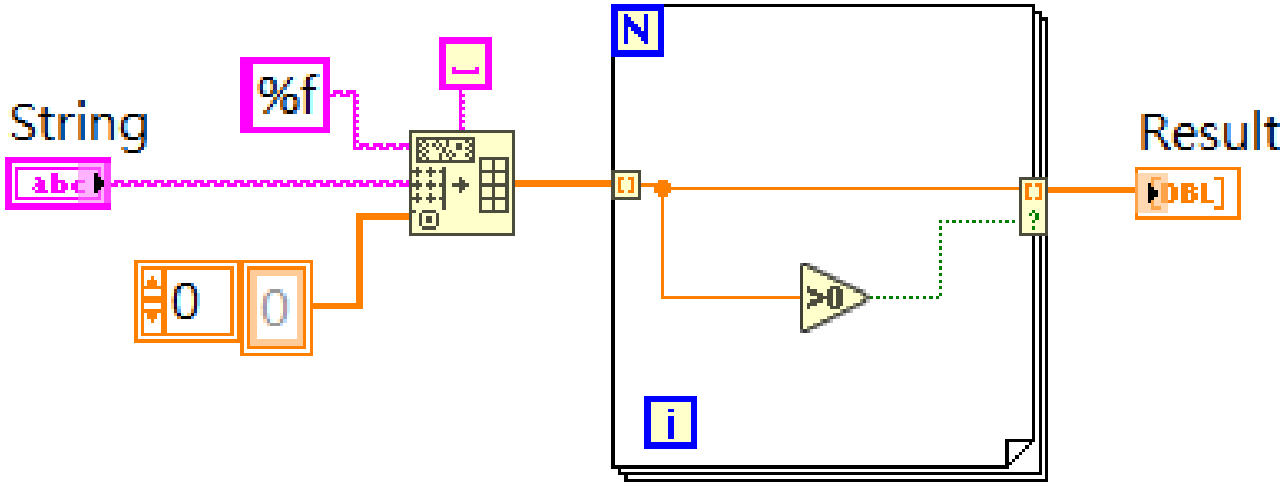


作者；曾智勇,chihyungtseng@hotmail.com

字串

- 範例：文字轉數值並進行過濾

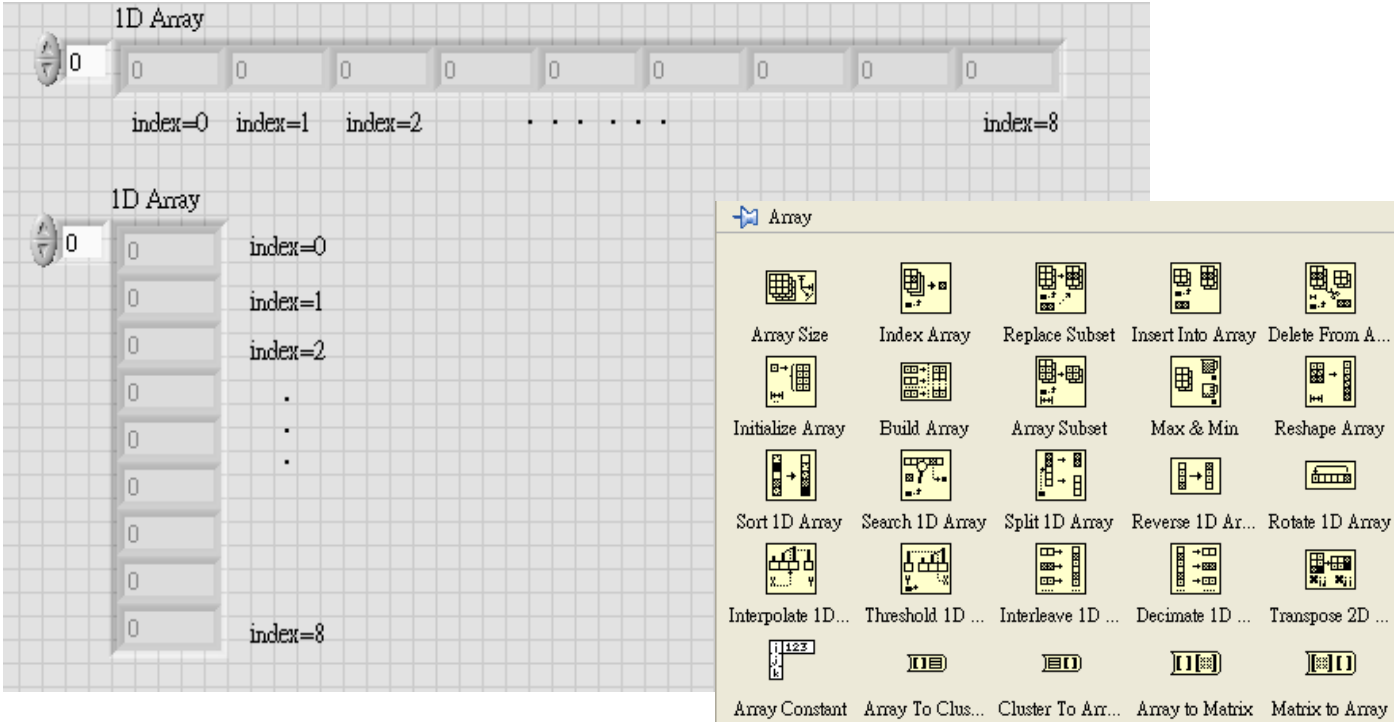




作者；曾智勇,chihyungtseng@hotmail.com

陣列

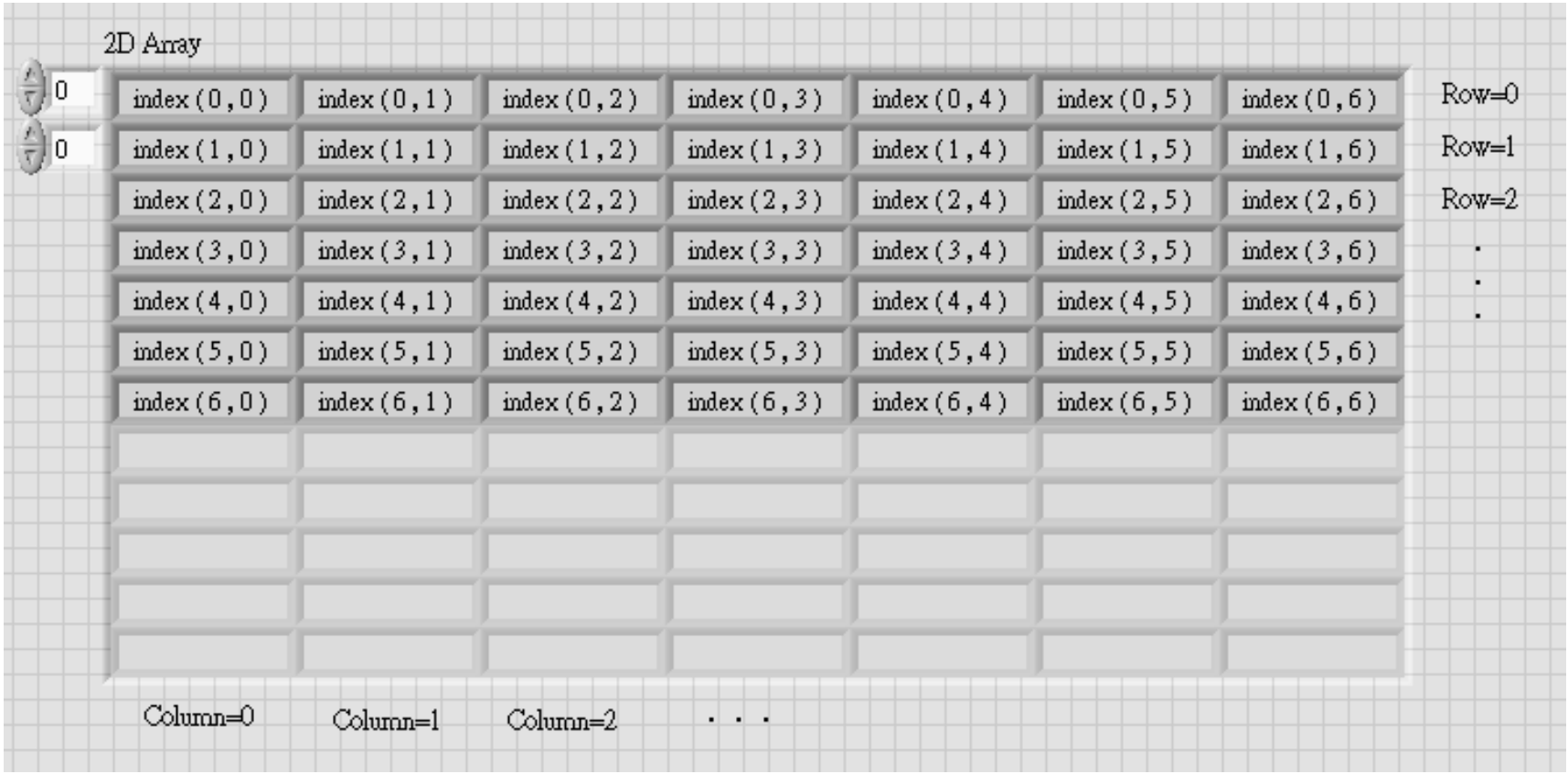
- 陣列：
  - 相同資料型態的元素，有序排列的群組
  - 可有一個或多個維度，每個維度可放至有2^{31}個元素
  - 可使用索引(index)存取陣列內元素，index起始值為0



作者；曾智勇,chihyungtseng@hotmail.com

陣列

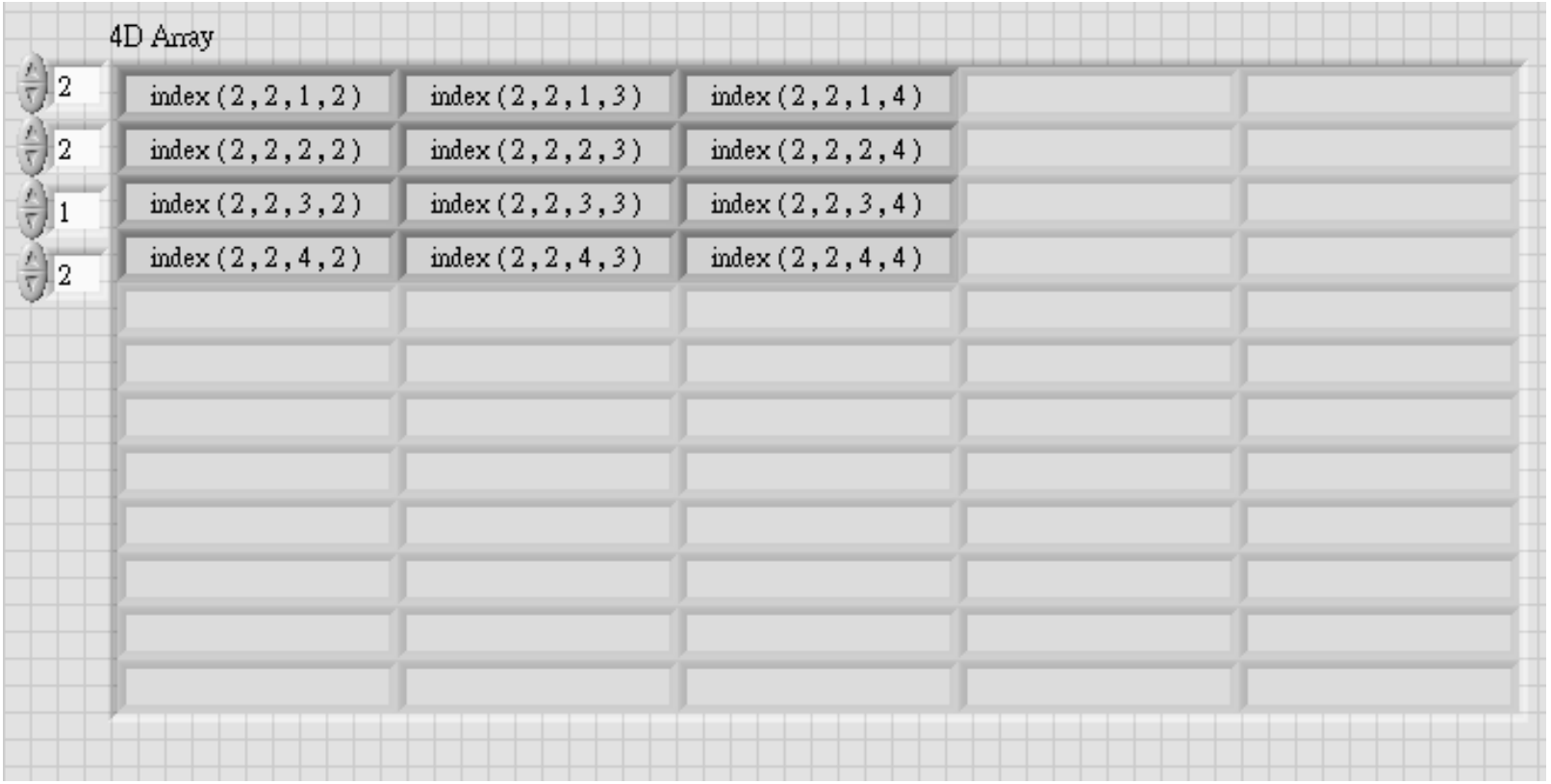
- 陣列：
  - 2D矩陣，橫向為列(Row)，縱向為欄(Column)



作者；曾智勇,chihyungtseng@hotmail.com

陣列

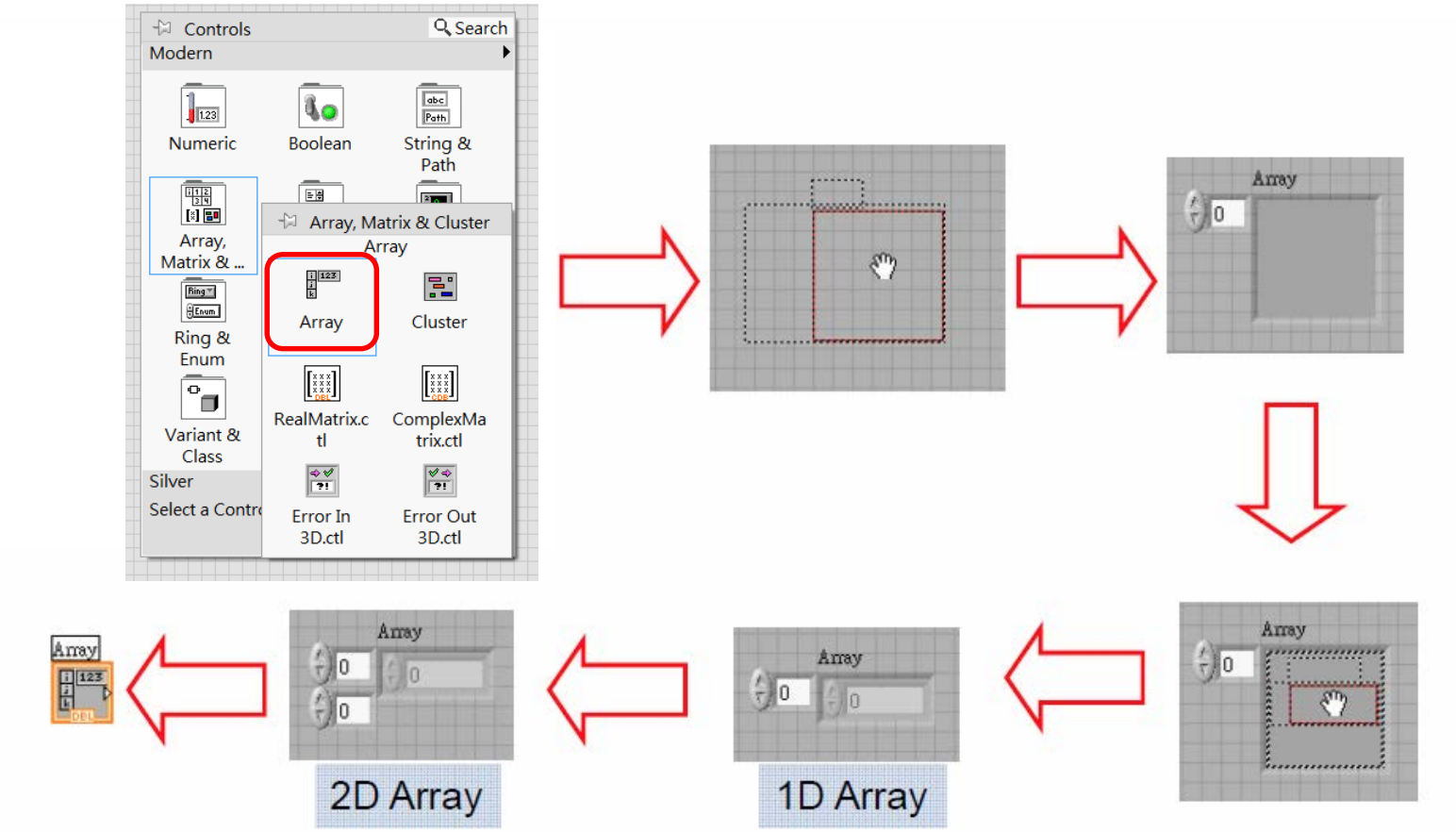
- 陣列：
  - N維度矩陣(含3維)，index欄位依序為N維、N-1維、…、Row、Column



作者；曾智勇,chihyungtseng@hotmail.com

陣列

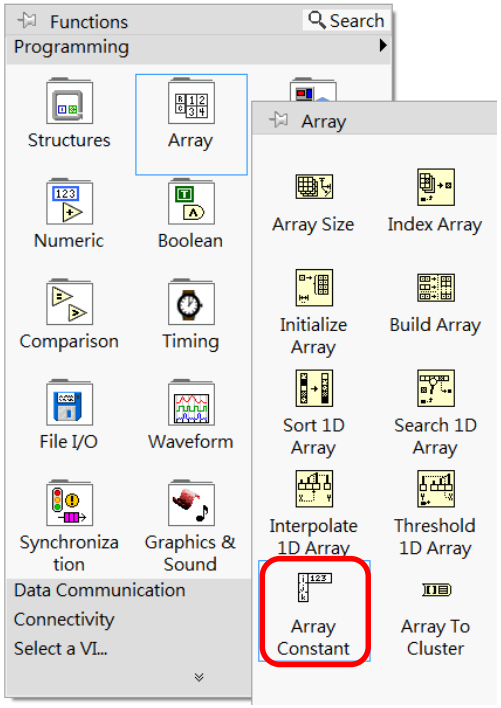
- 建立控制項或顯示項陣列：

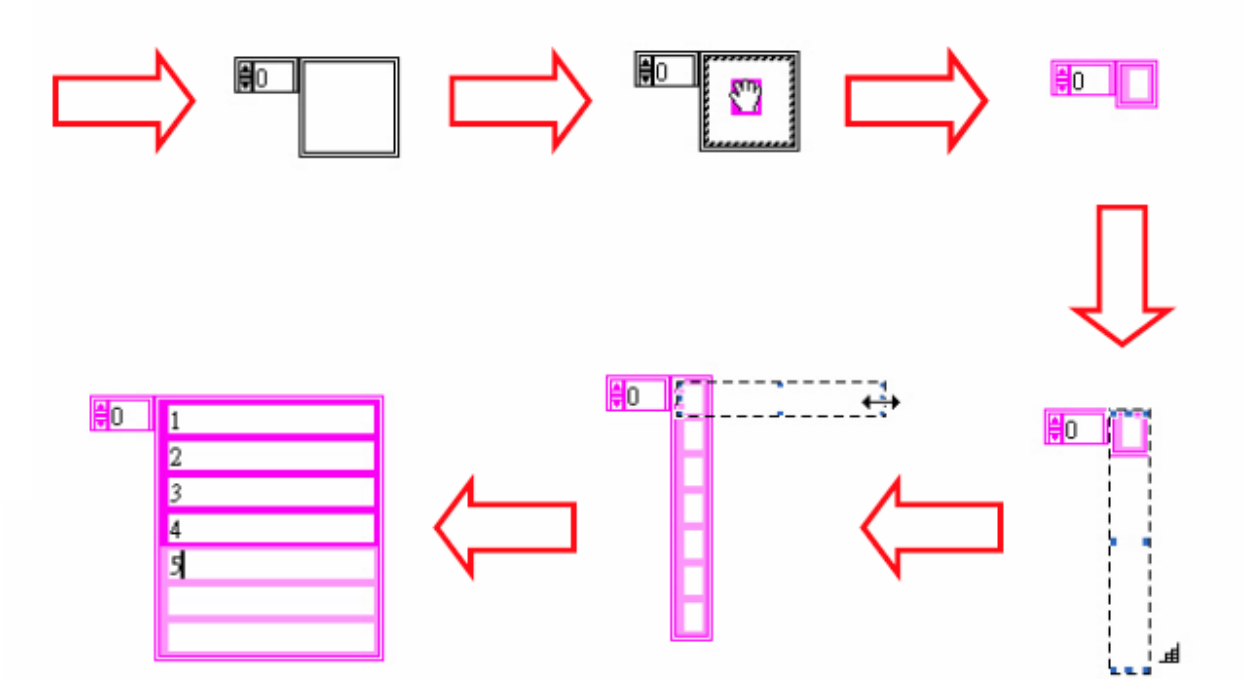


作者；曾智勇,chihyungtseng@hotmail.com

陣列

- 建立常數陣列：

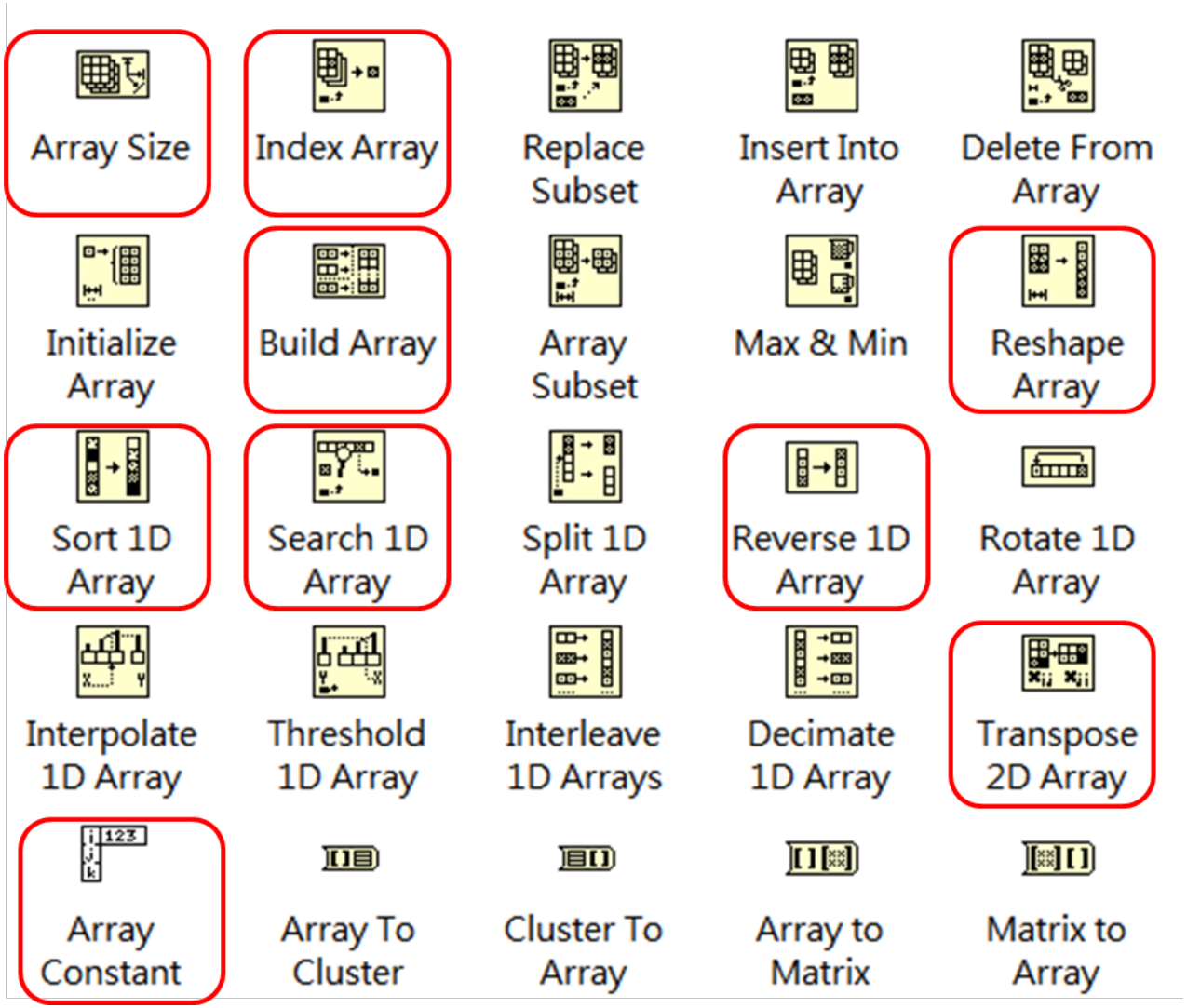




作者；曾智勇,chihyungtseng@hotmail.com

陣列

- 陣列常用元件：



作者；曾智勇, chihyungtseng@hotmail.com

陣列

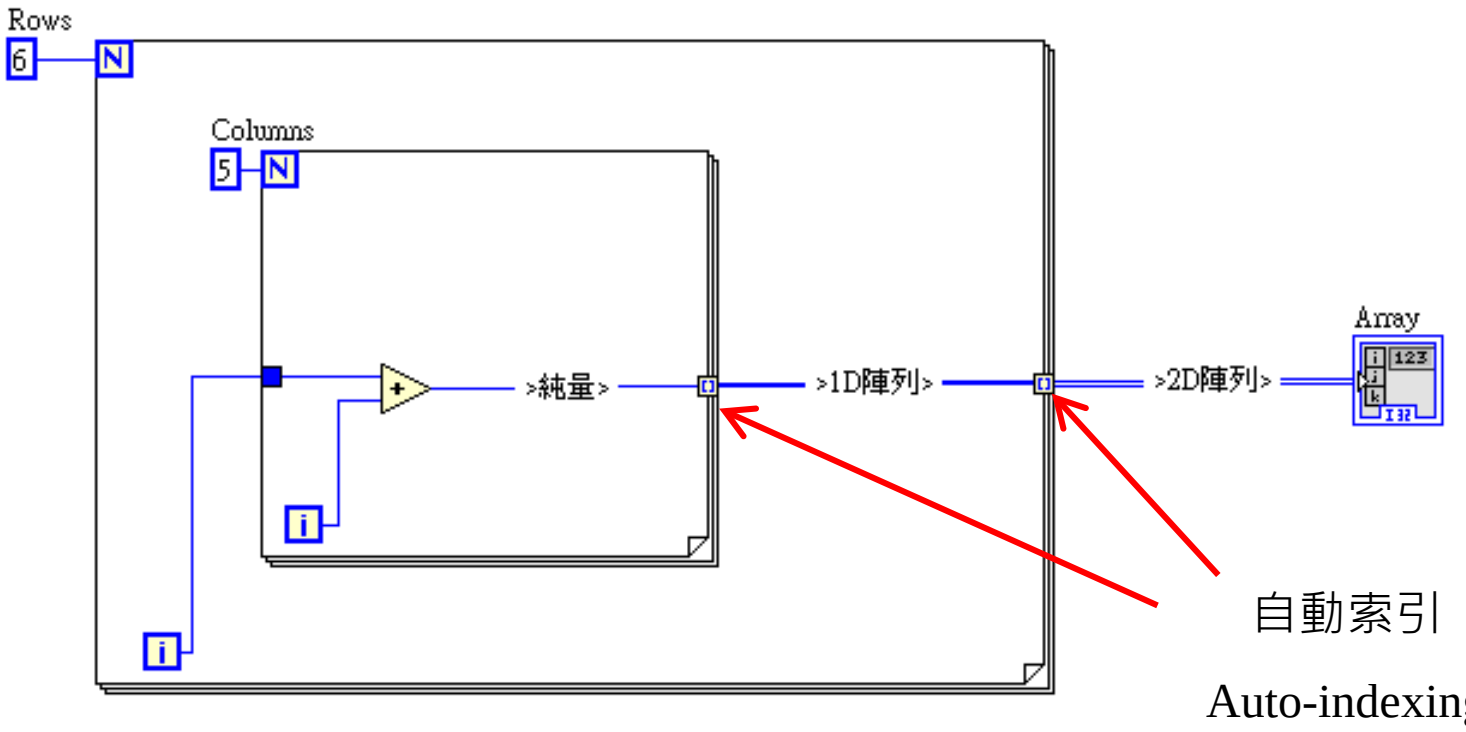
- 常見的五種產生陣列方法：
  - 人機介面直接產生空陣列，並拖曳所需的資料型態
  - 程式區直接產生空陣列，並拖曳所需的資料型態
  - 程式區於vi的陣列輸入/輸出點，使用滑鼠右鍵，Create Control/Indicator
  - 使用Auto-indexing產生輸出點，使用滑鼠右鍵，Create Control/Indicator
  - 使用初始化陣列vi

- 增加陣列維度：
  - 滑鼠拖曳下拉index指示器
  - 右鍵新增維度
  - 經過新的迴圈，使用Auto-indexing功能
  - 使用建立陣列vi

作者；曾智勇,chihyungtseng@hotmail.com

陣列

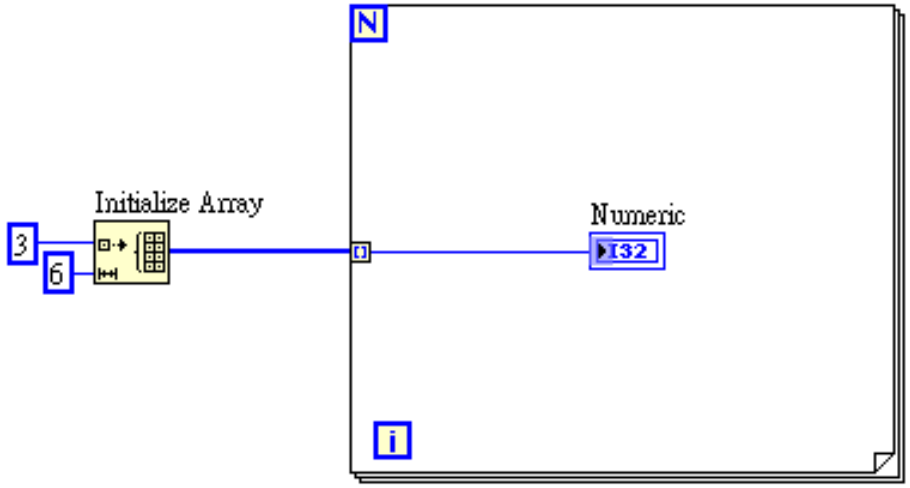
- 從迴圈產生陣列：

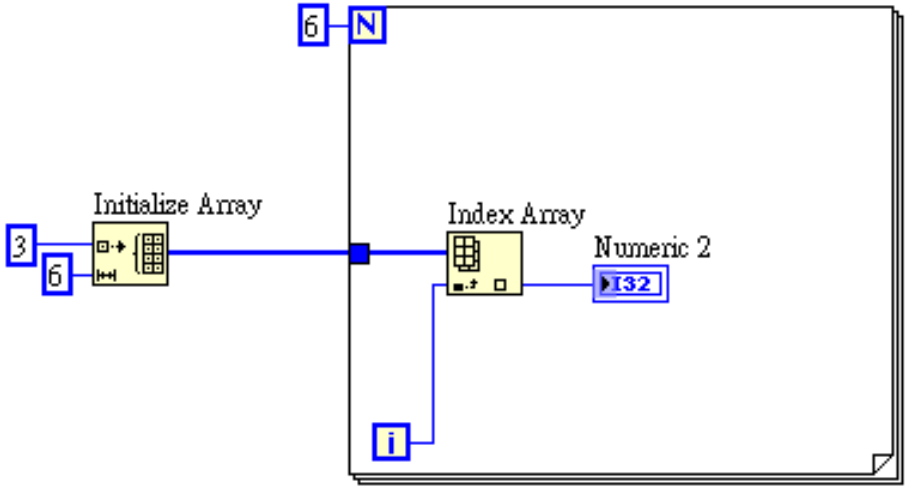


作者；曾智勇,chihyungtseng@hotmail.com

陣列

- 陣列特殊情況：
  - 未給For loop次數時，以auto-indexing執行
  - 給For loop次數時， auto-indexing需手動取消
  - 同時存在For loop次數與auto-indexing時，取兩者較小者為執行次數
  - 陣列可與常數進行運算
  - 陣列與陣列進行運算時，取元素數較小者為運算後之元素數
  - 資料經過While loop時， auto-indexing需手動開啟

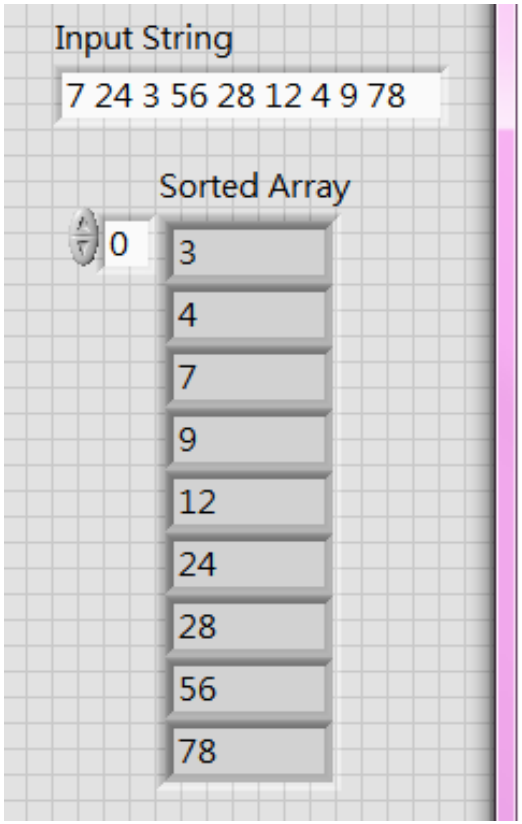


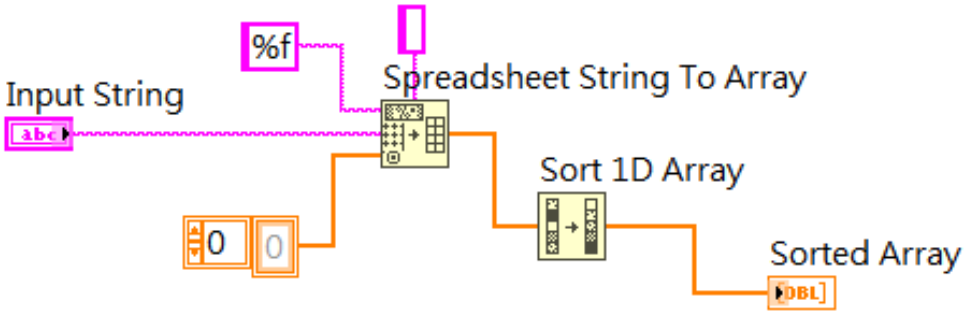


作者；曾智勇,chihyungtseng@hotmail.com

陣列

- 範例：字串轉數值陣列，並進行排序

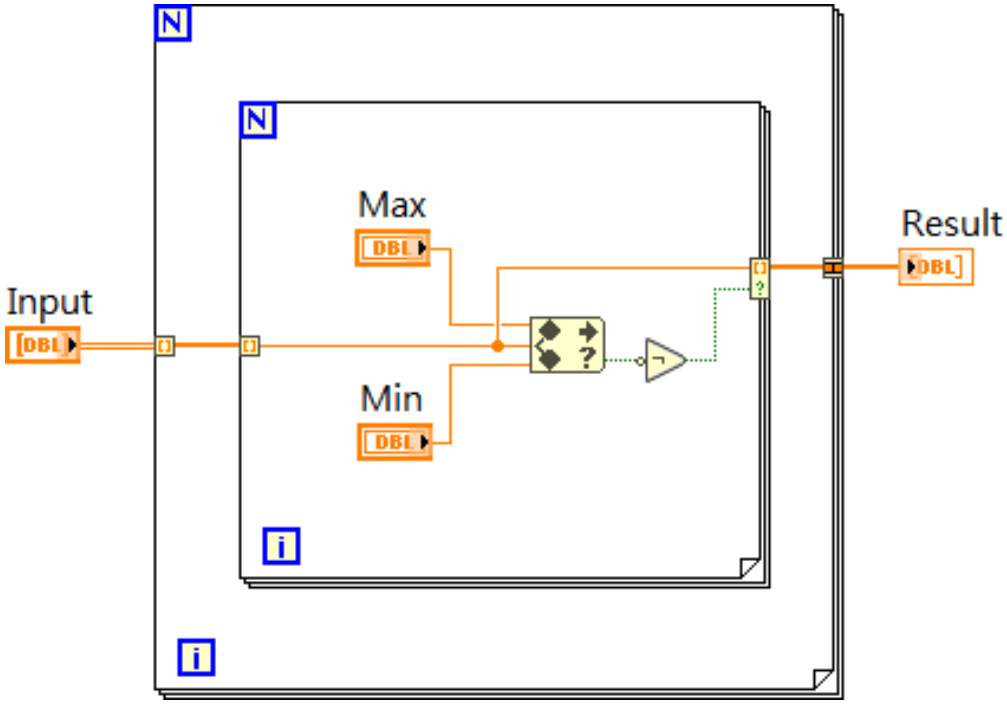


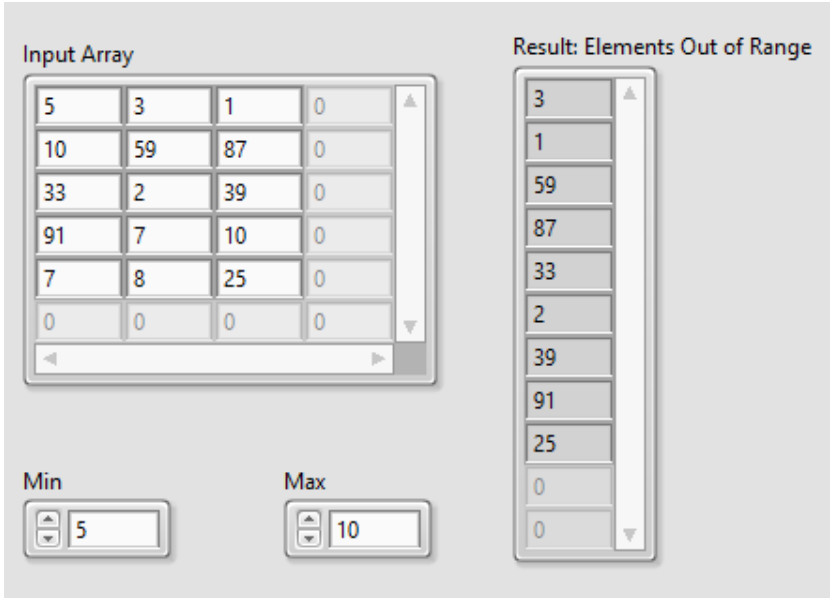


作者；曾智勇,chihyungtseng@hotmail.com

陣列

- 範例：將二維陣列中，範圍區間以外的值以一維陣列顯示

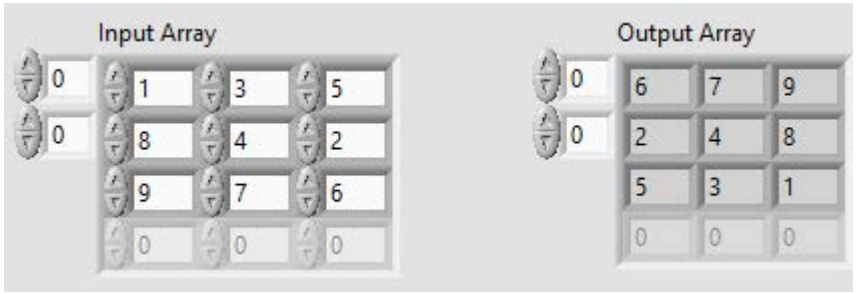


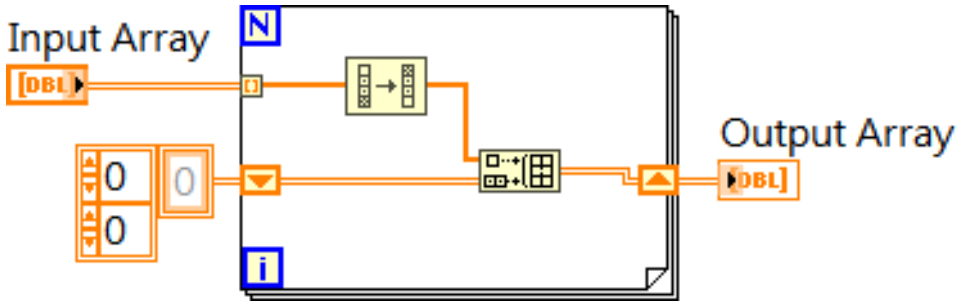


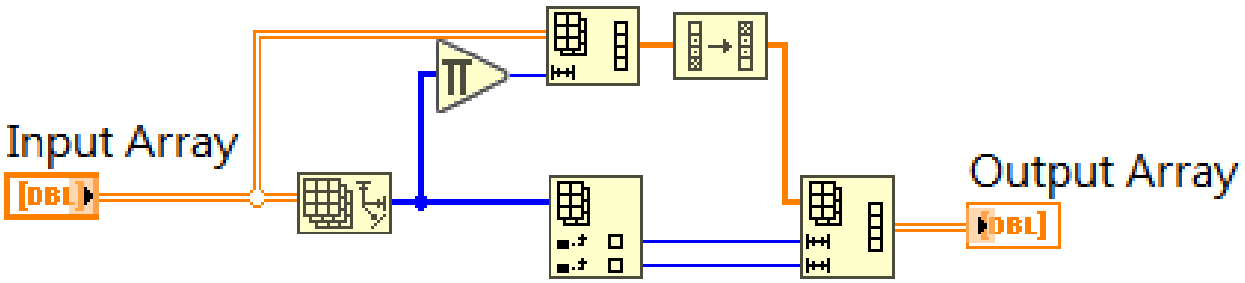
作者；曾智勇,chihyungtseng@hotmail.com

陣列

- 範例：二維陣列中，資料旋轉180度



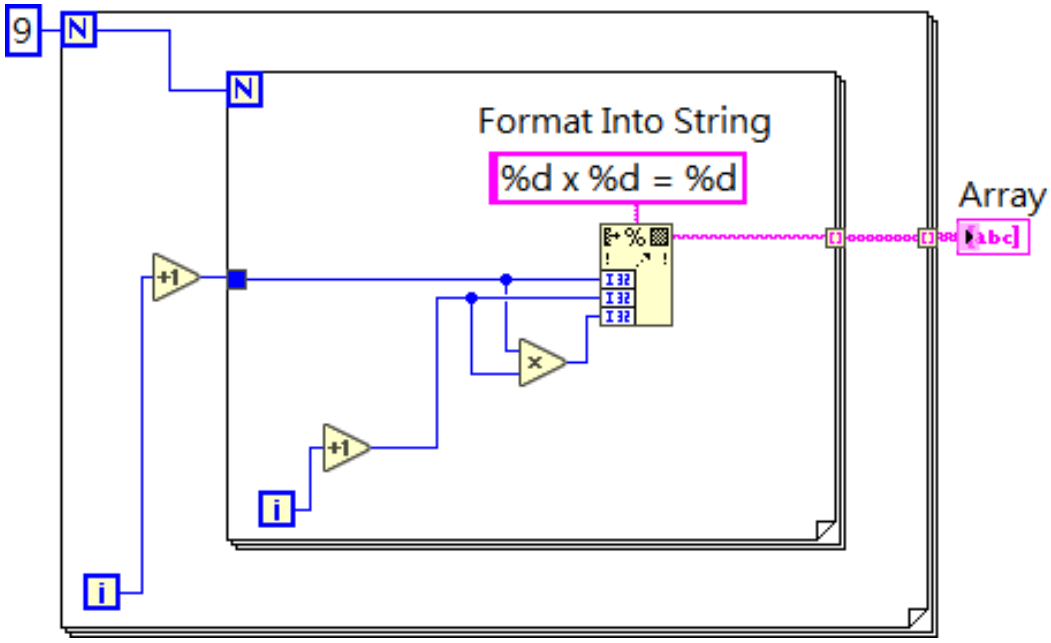




作者；曾智勇,chihyungtseng@hotmail.com

陣列

- 範例：九九乘法表

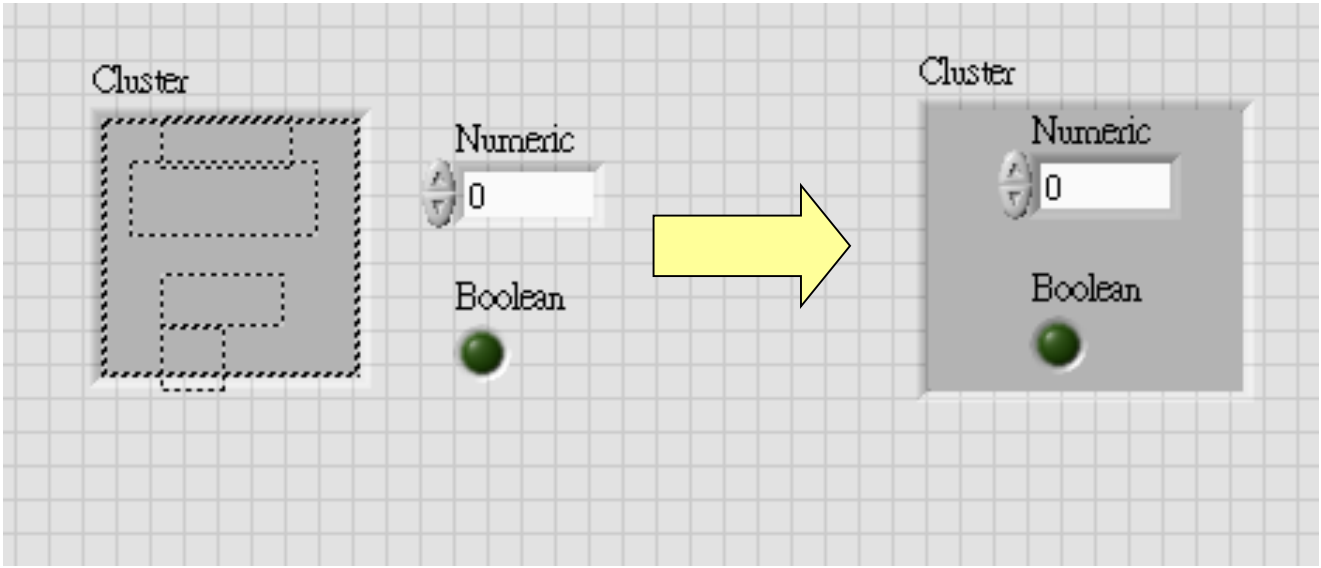


1 x 1 = 1	1 x 2 = 2	1 x 3 = 3	1 x 4 = 4	1 x 5 = 5	1 x 6 = 6	1 x 7 = 7	1 x 8 = 8	1 x 9 = 9
2 x 1 = 2	2 x 2 = 4	2 x 3 = 6	2 x 4 = 8	2 x 5 = 10	2 x 6 = 12	2 x 7 = 14	2 x 8 = 16	2 x 9 = 18
3 x 1 = 3	3 x 2 = 6	3 x 3 = 9	3 x 4 = 12	3 x 5 = 15	3 x 6 = 18	3 x 7 = 21	3 x 8 = 24	3 x 9 = 27
4 x 1 = 4	4 x 2 = 8	4 x 3 = 12	4 x 4 = 16	4 x 5 = 20	4 x 6 = 24	4 x 7 = 28	4 x 8 = 32	4 x 9 = 36
5 x 1 = 5	5 x 2 = 10	5 x 3 = 15	5 x 4 = 20	5 x 5 = 25	5 x 6 = 30	5 x 7 = 35	5 x 8 = 40	5 x 9 = 45
6 x 1 = 6	6 x 2 = 12	6 x 3 = 18	6 x 4 = 24	6 x 5 = 30	6 x 6 = 36	6 x 7 = 42	6 x 8 = 48	6 x 9 = 54
7 x 1 = 7	7 x 2 = 14	7 x 3 = 21	7 x 4 = 28	7 x 5 = 35	7 x 6 = 42	7 x 7 = 49	7 x 8 = 56	7 x 9 = 63
8 x 1 = 8	8 x 2 = 16	8 x 3 = 24	8 x 4 = 32	8 x 5 = 40	8 x 6 = 48	8 x 7 = 56	8 x 8 = 64	8 x 9 = 72
9 x 1 = 9	9 x 2 = 18	9 x 3 = 27	9 x 4 = 36	9 x 5 = 45	9 x 6 = 54	9 x 7 = 63	9 x 8 = 72	9 x 9 = 81

作者；曾智勇,chihyungtseng@hotmail.com

叢集

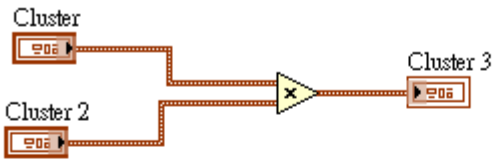
- 叢集(Cluster)：
  - 多種資料型態的集合群組
  - 類似C語言的結構(Structure)功能

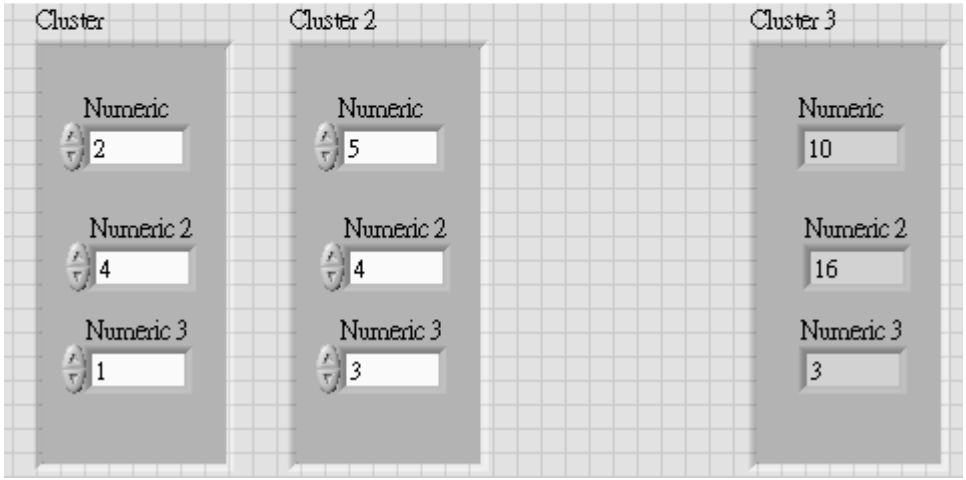


作者；曾智勇,chihyungtseng@hotmail.com

叢集

- 叢集特例：
  - 純數字的叢集元素，和純量或相同元素的叢集可進行四則運算

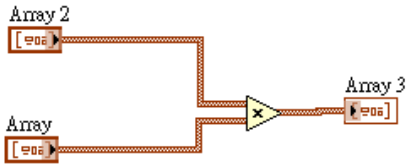


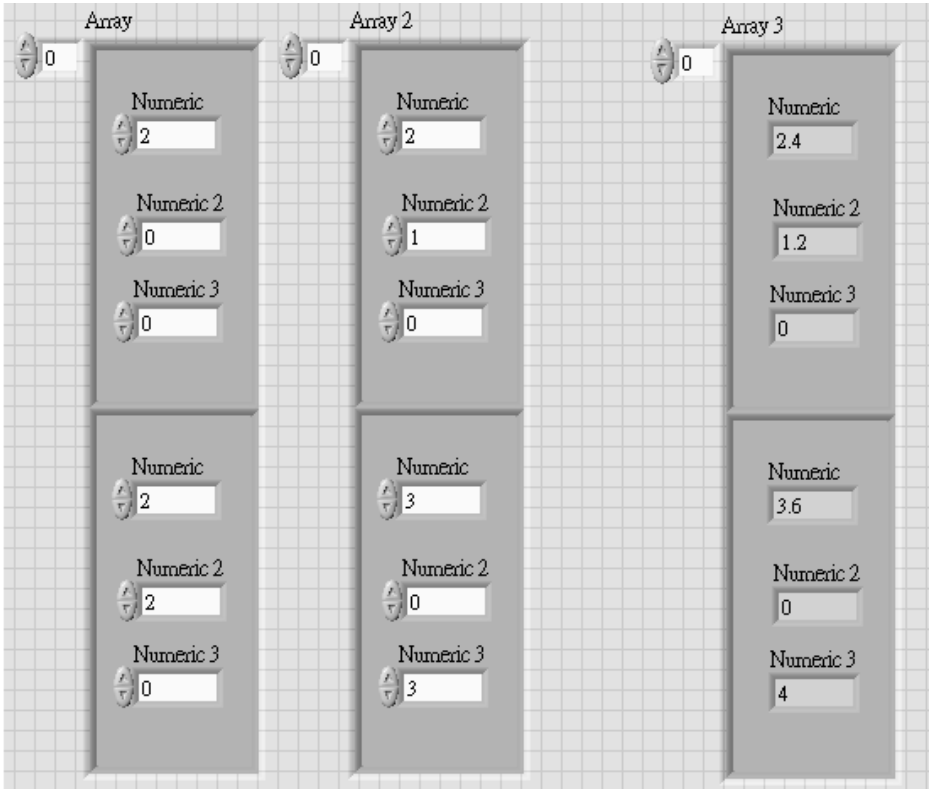


作者；曾智勇,chihyungtseng@hotmail.com

叢集

- 叢集特例：
  - 純數字的叢集陣列，和純量或相同元素的叢集也可進行四則運算

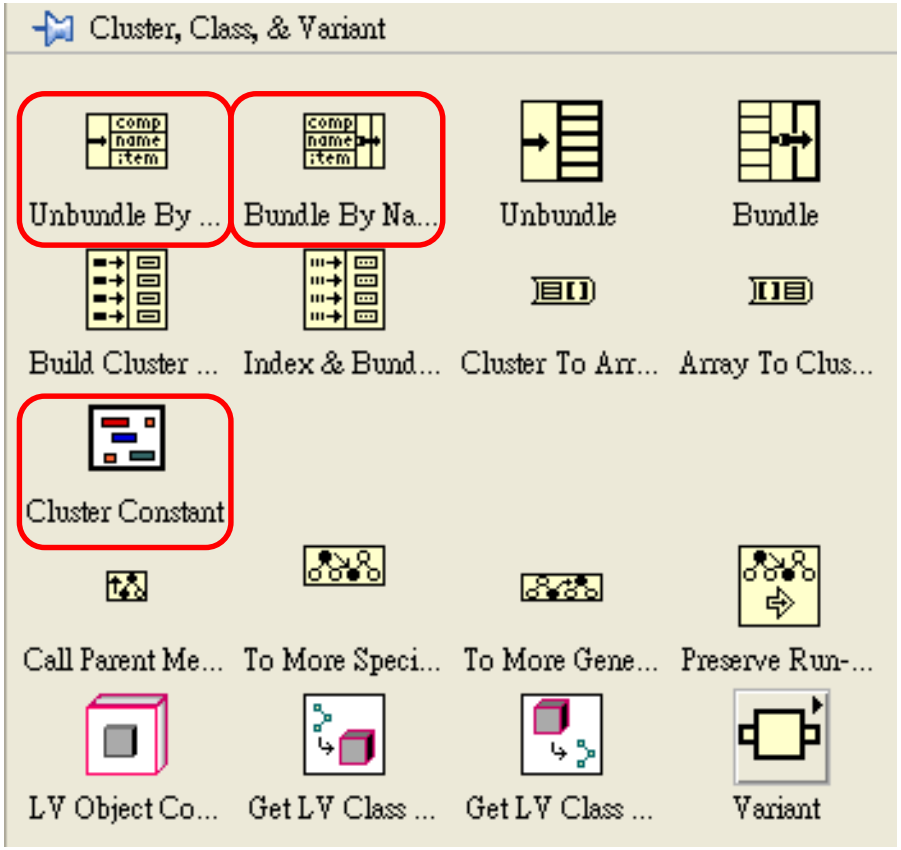


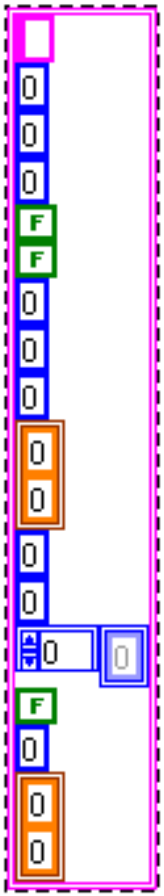


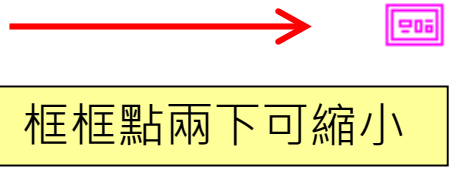
作者；曾智勇,chihyungtseng@hotmail.com

叢集

- 叢集：



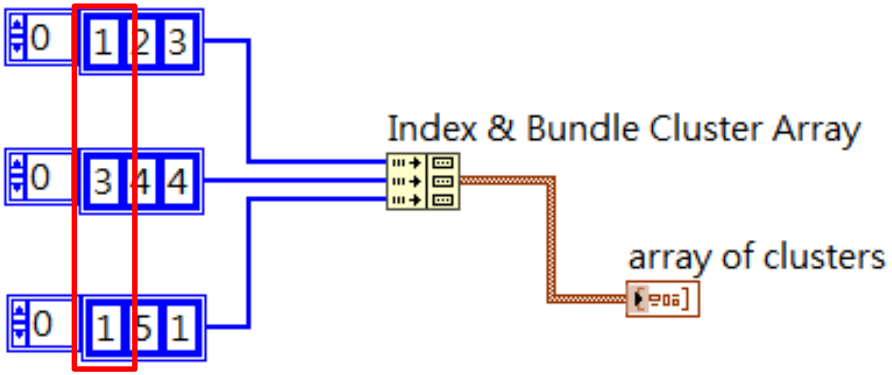


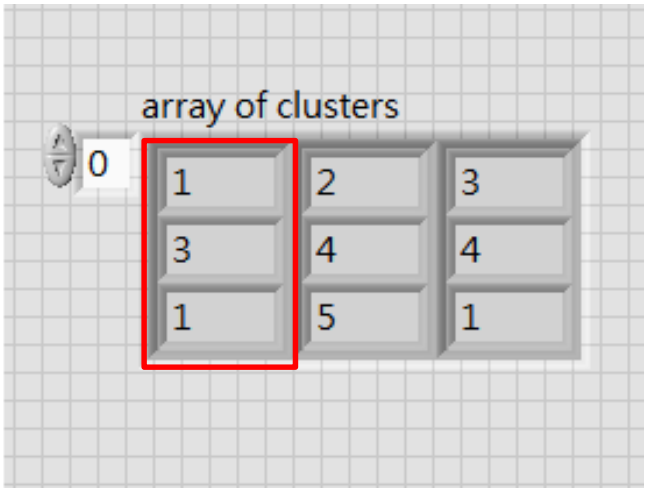


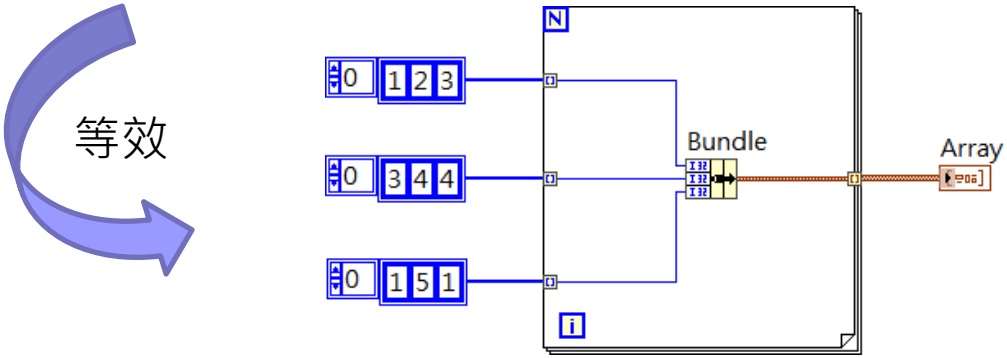
作者；曾智勇,chihyungtseng@hotmail.com

叢集

- 叢集特殊元件-Index & Bundle Cluster Array：
  將各矩陣的對應元素取出，Bundle成Cluster，再堆疊成Cluster陣列



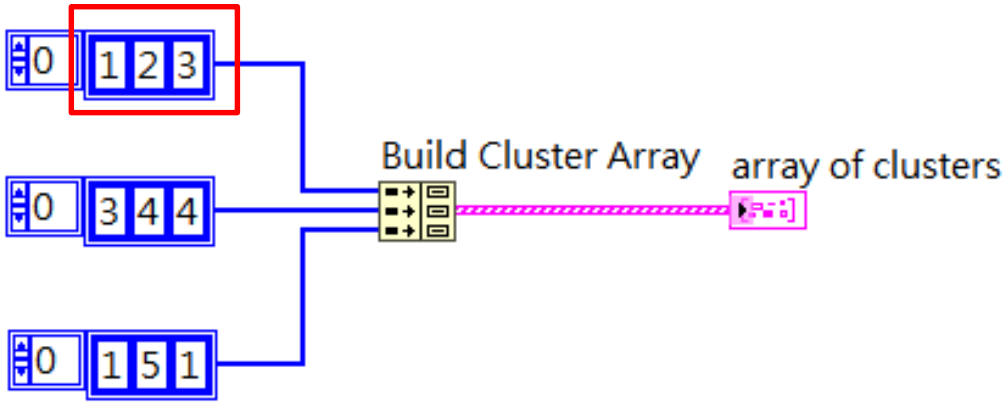


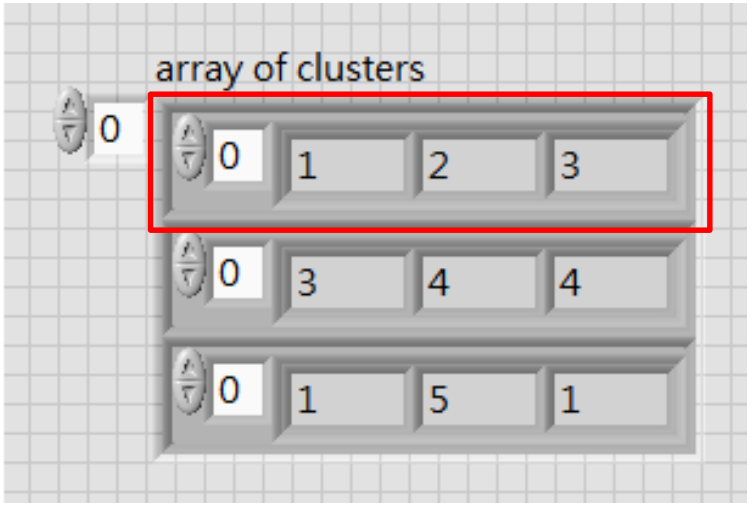


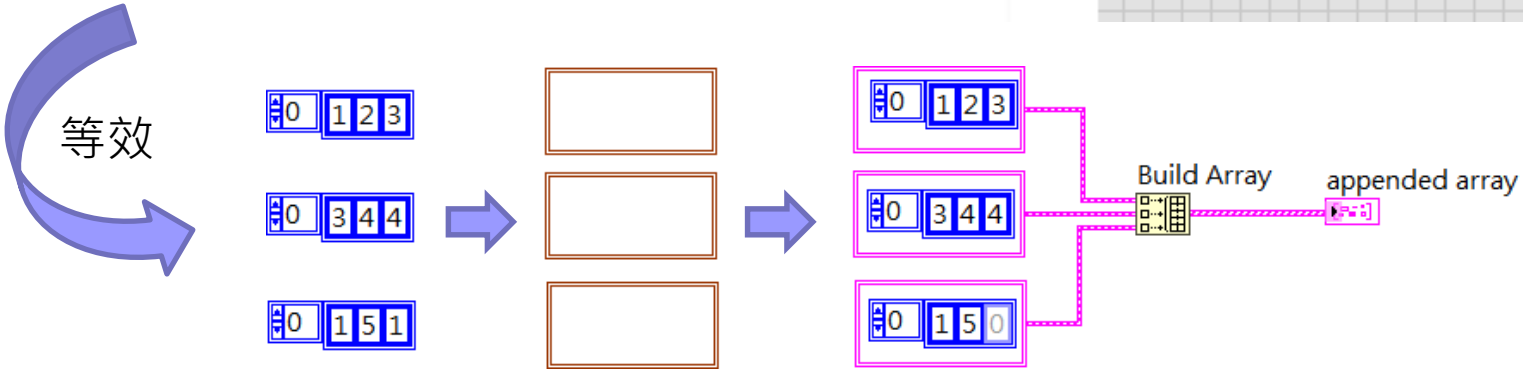
作者；曾智勇,chihyungtseng@hotmail.com

叢集

- 叢集特殊元件-Build Cluster Array：
  將矩陣的Bundle成Cluster，再堆疊成Cluster陣列



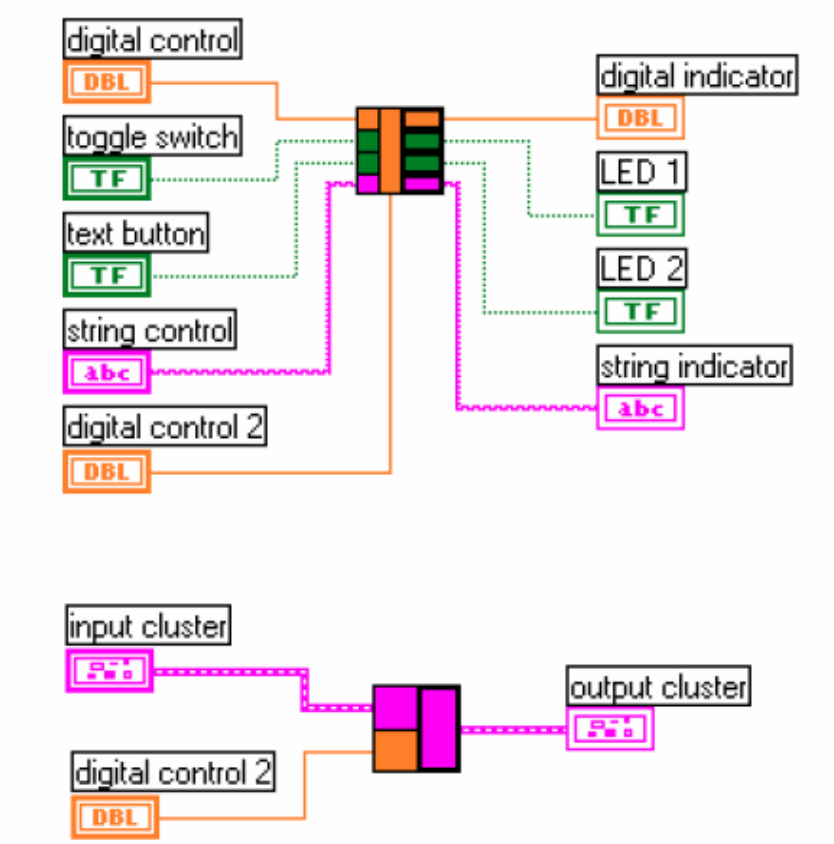


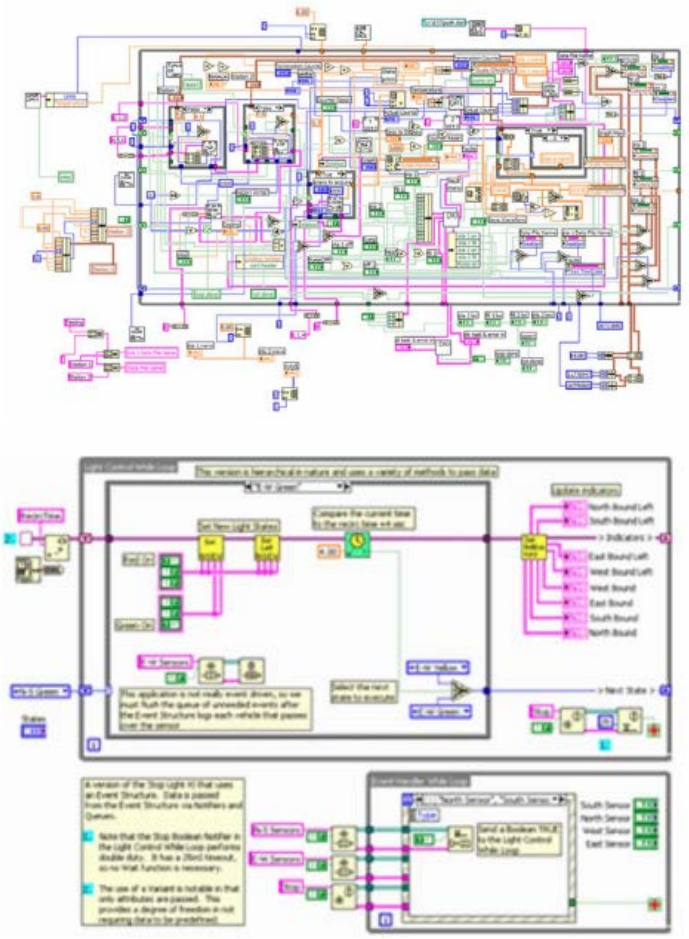


作者；曾智勇,chihyungtseng@hotmail.com

叢集

- 運用叢集，簡化接線，精簡畫面

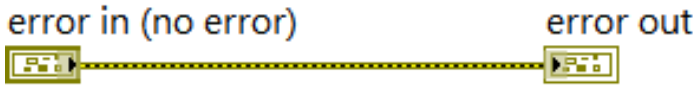




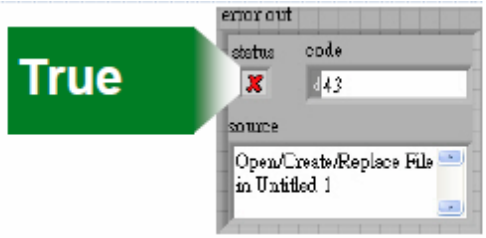
作者；曾智勇,chihyungtseng@hotmail.com

叢集

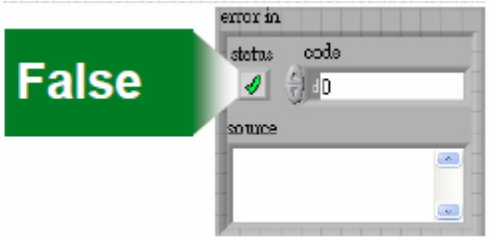
- 錯誤叢集(Error Cluster)：
  由布林(status)、數字(code)、文字(source)所組成的叢集特例
  元件與連接線顏色呈現金色



- **Status**為布林
  有錯誤=True=紅色打叉符號，
  沒錯誤=False=綠色打勾符號。
- **Code**為32 位元帶正負號的整數，以數字的方式來辨識錯誤。
- **Source**為字串，用於說明錯誤發生的位置與內容。



有錯誤

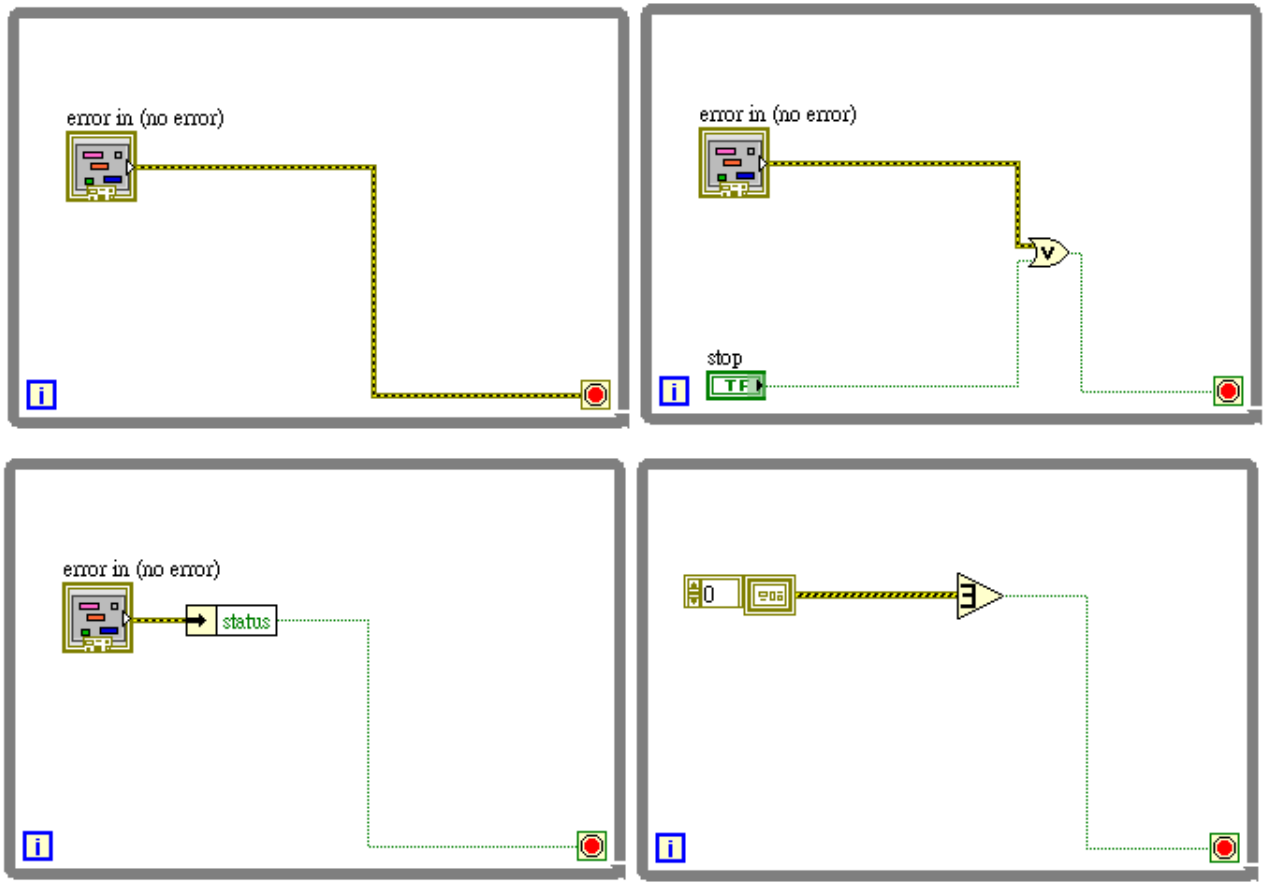


沒錯誤

作者；曾智勇,chihyungtseng@hotmail.com

叢集

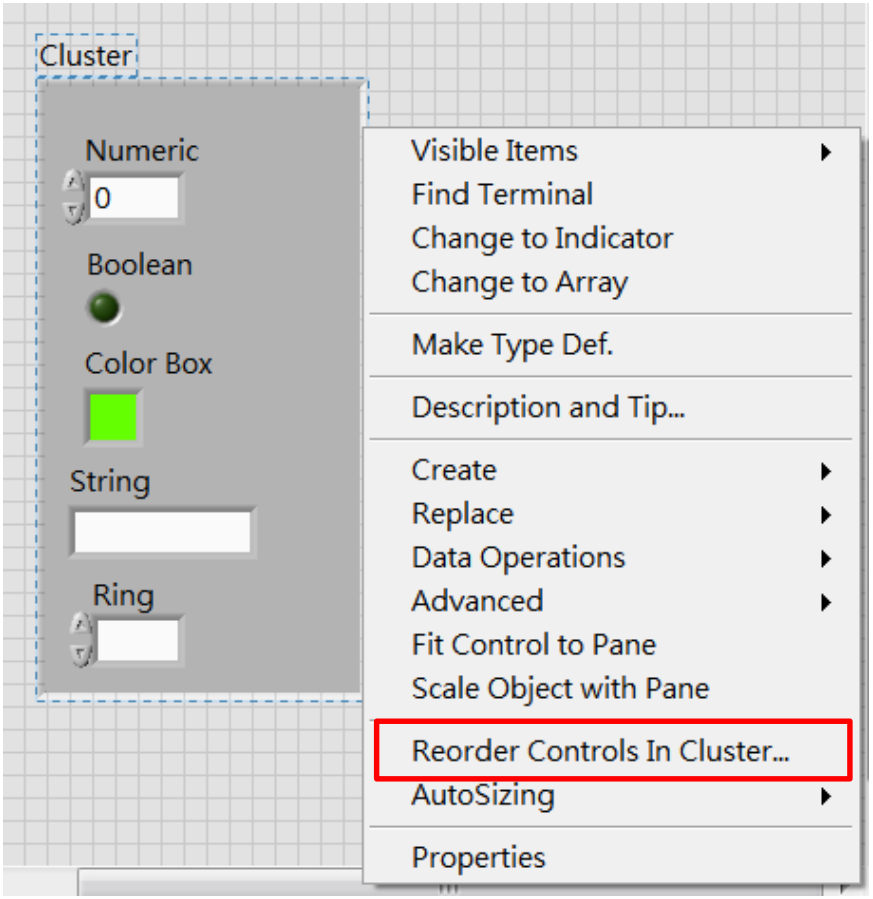
- 錯誤叢集應用於迴圈停止的四種方法

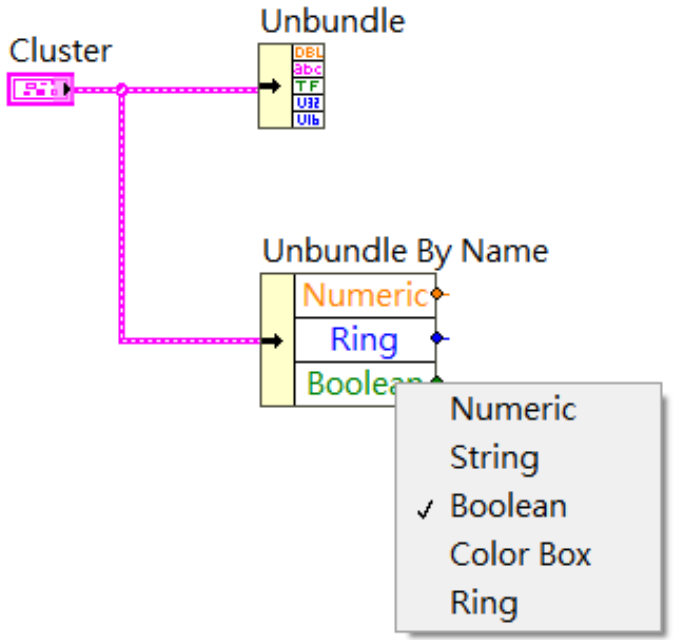


作者；曾智勇,chihyungtseng@hotmail.com

叢集

- 叢集順序：預設排序依照元件建立順序，可重新定義順序

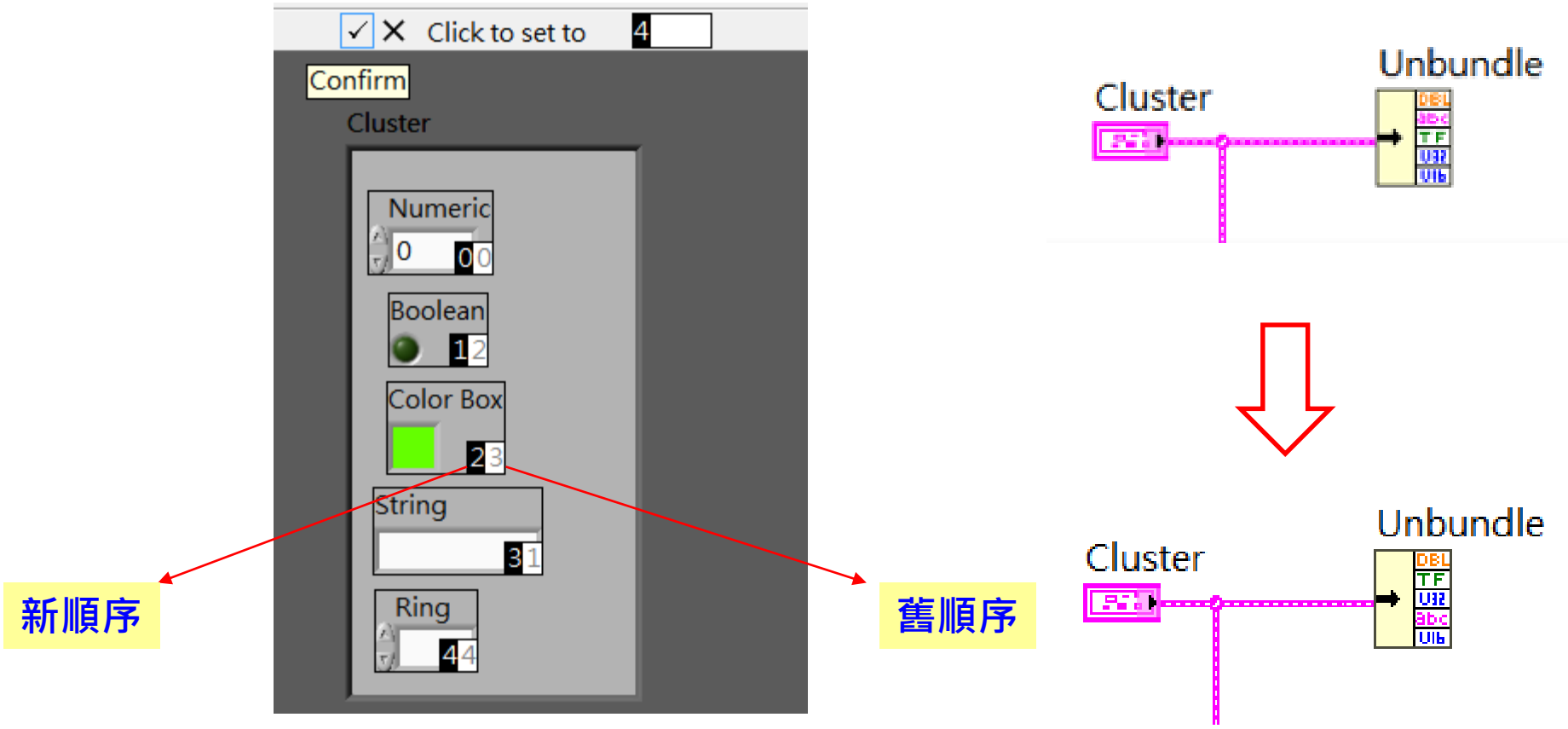




作者；曾智勇,chihyungtseng@hotmail.com

叢集

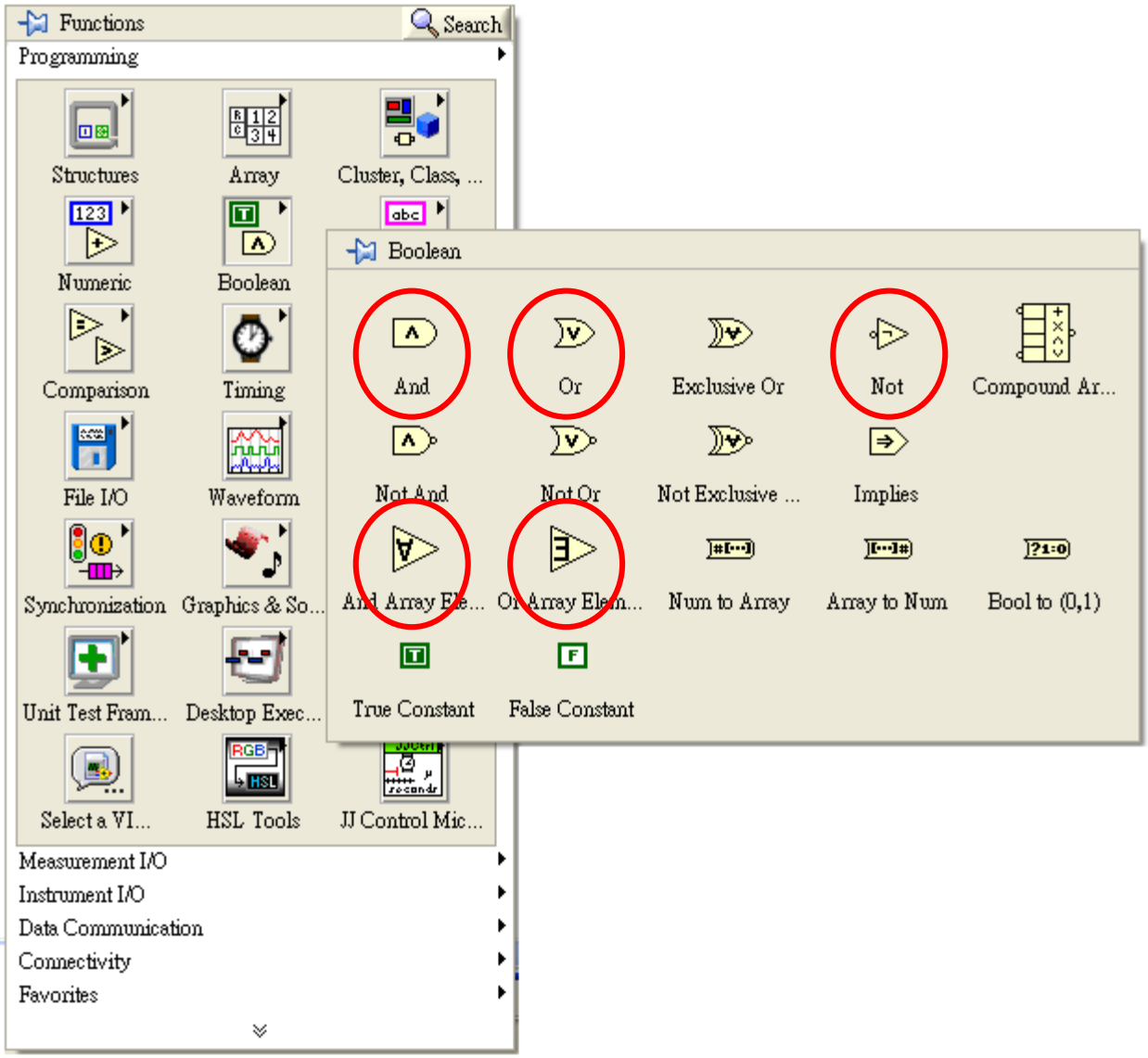
- 叢集順序：預設排序依照元件建立順序，可重新定義順序



作者；曾智勇,chihyungtseng@hotmail.com

布林

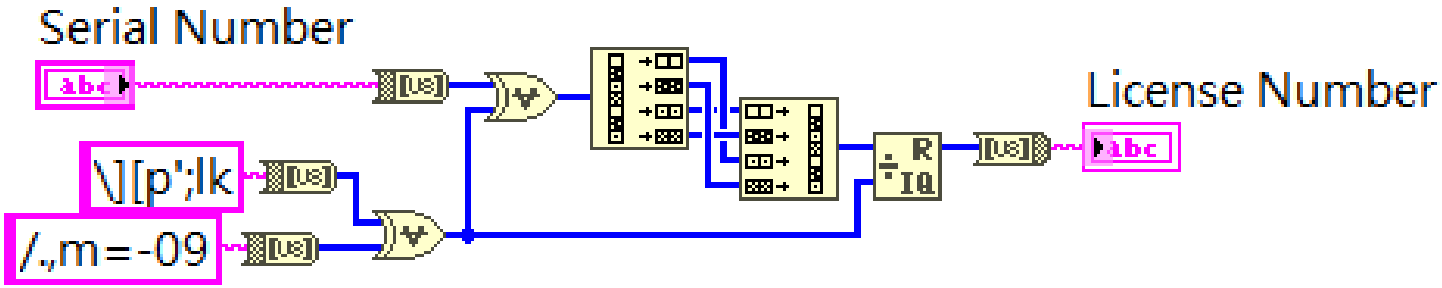
- 布林：

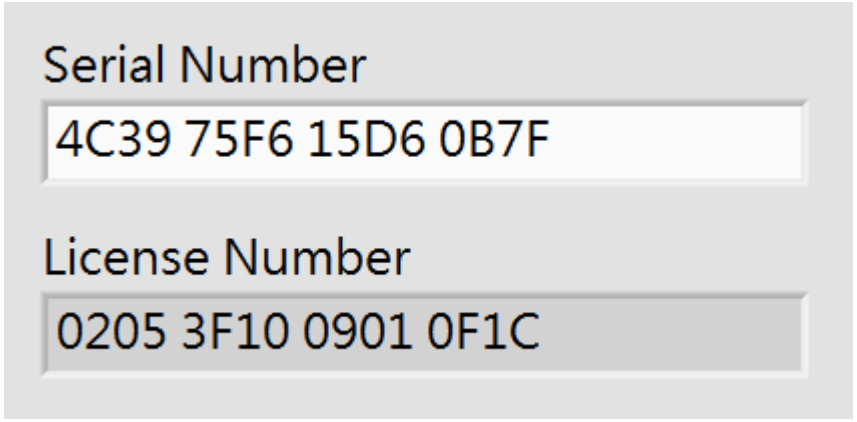


作者；曾智勇,chihyungtseng@hotmail.com

布林

- 布林：

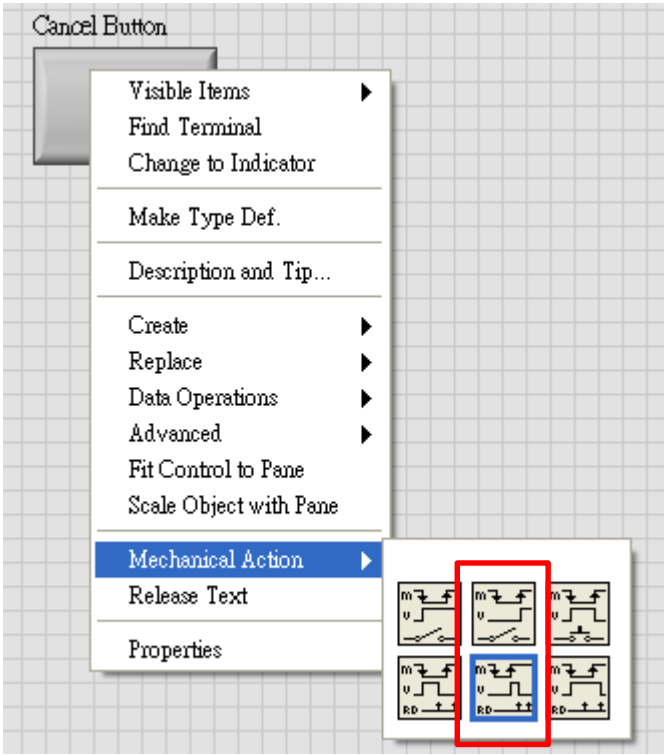


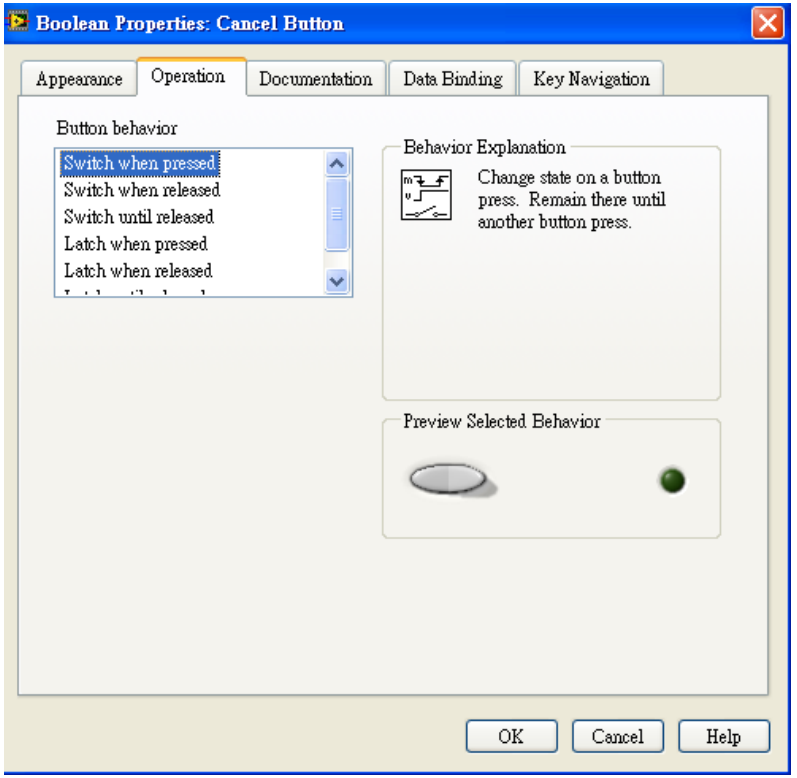


作者；曾智勇, chihyungtseng@hotmail.com

布林

- 布林按鈕：可切換6種不同觸發方式。

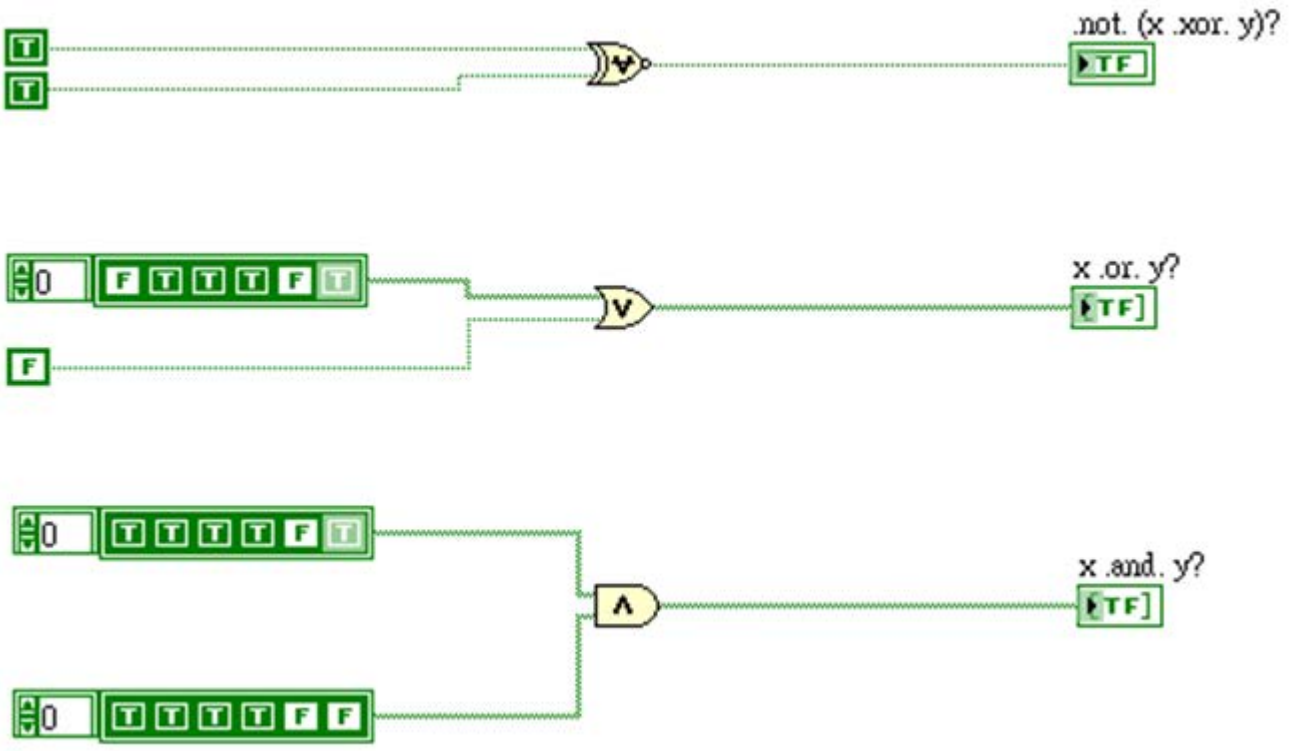




作者；曾智勇,chihyungtseng@hotmail.com

布林

- 布林運算：運算方式除了元素-元素，也可以元素-矩陣、矩陣-矩陣

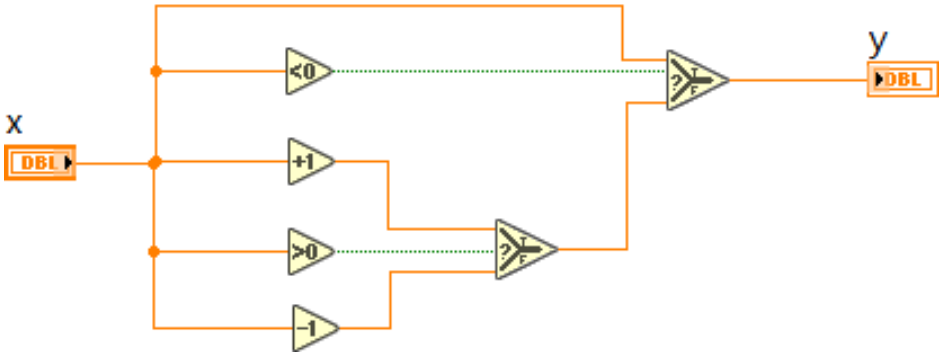


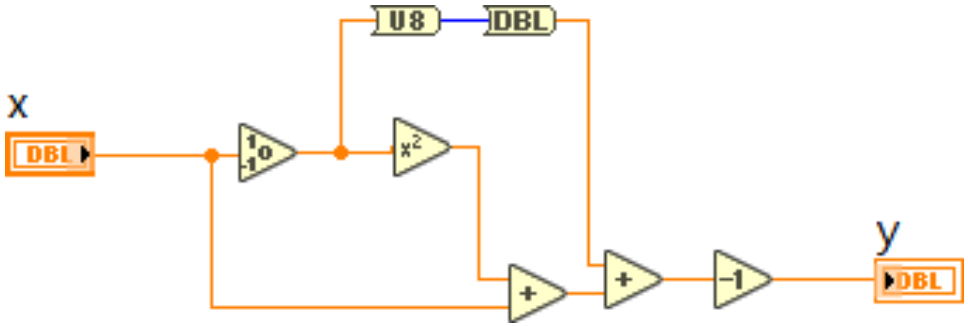
作者；曾智勇,chihyungtseng@hotmail.com

布林

- 範例：根據判斷式決定運算值

$$y = \begin{cases} x(-5 < x < 0) \\ x - 1(x = 0) \\ x + 1(0 < x < 10) \end{cases}$$

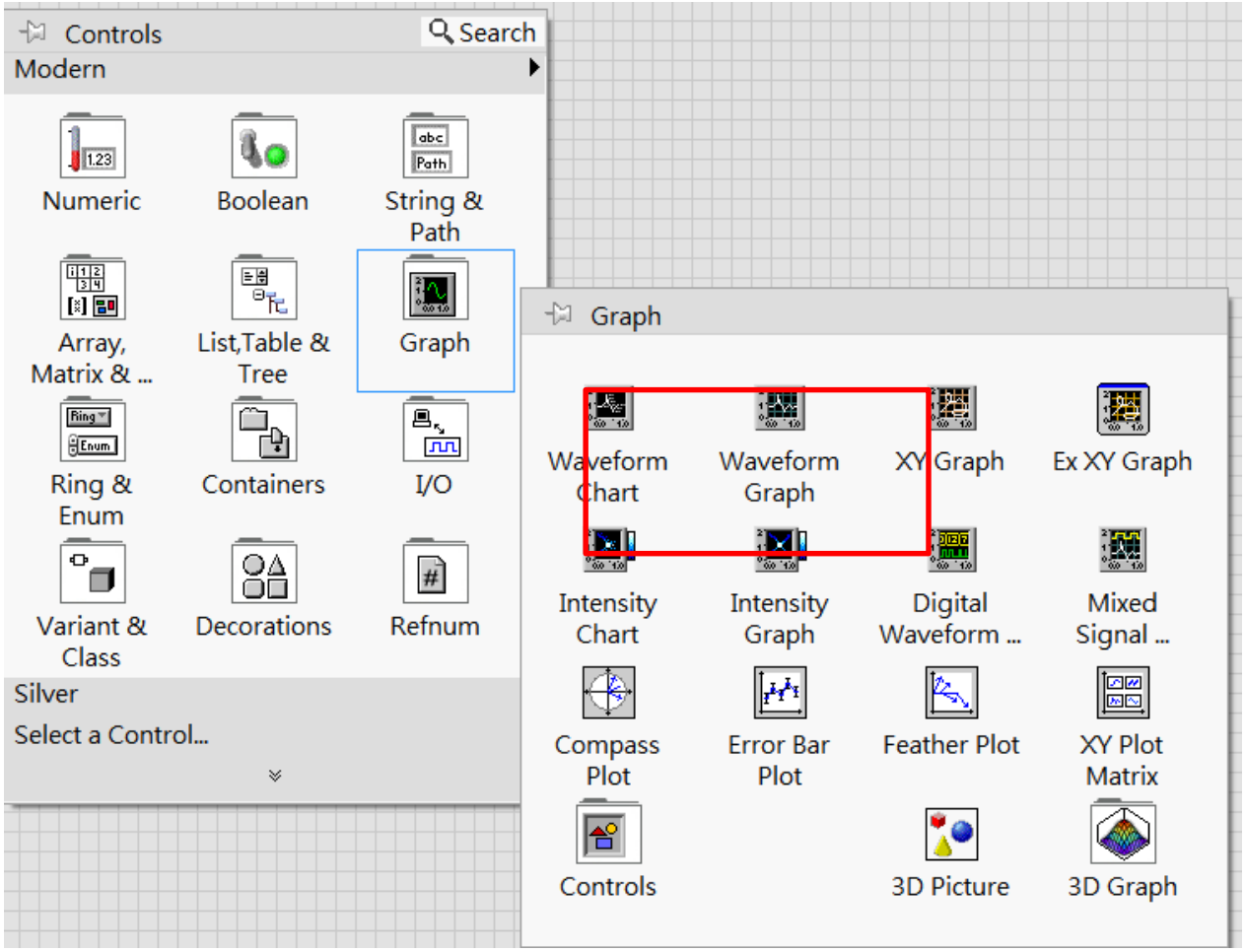




作者；曾智勇,chihyungtseng@hotmail.com

圖表

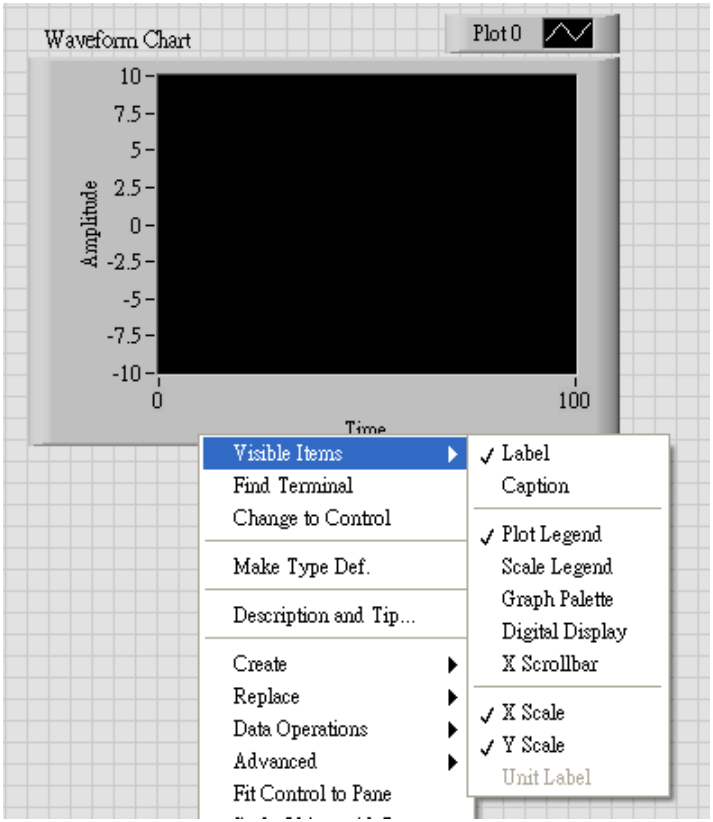
- 圖表位於：

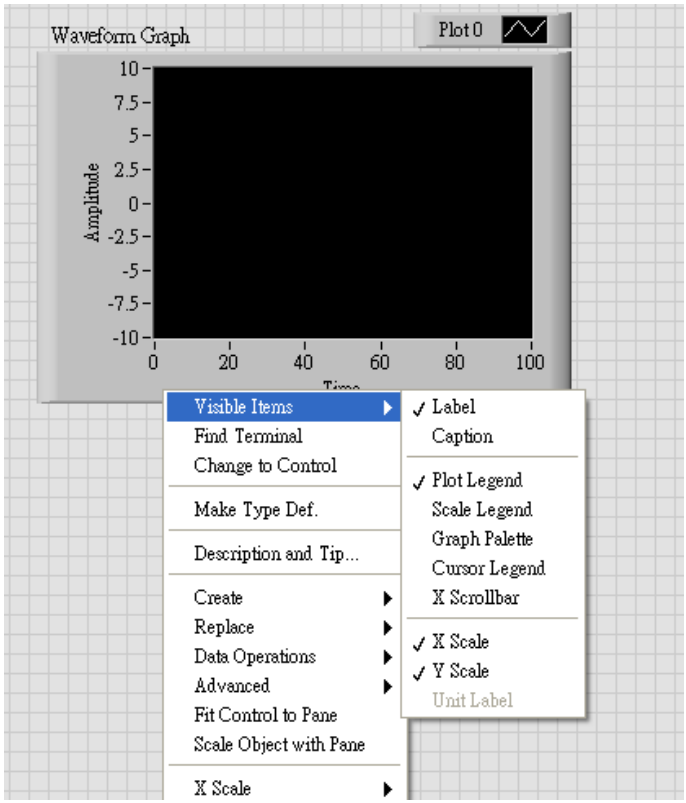


作者；曾智勇,chihyungtseng@hotmail.com

圖表

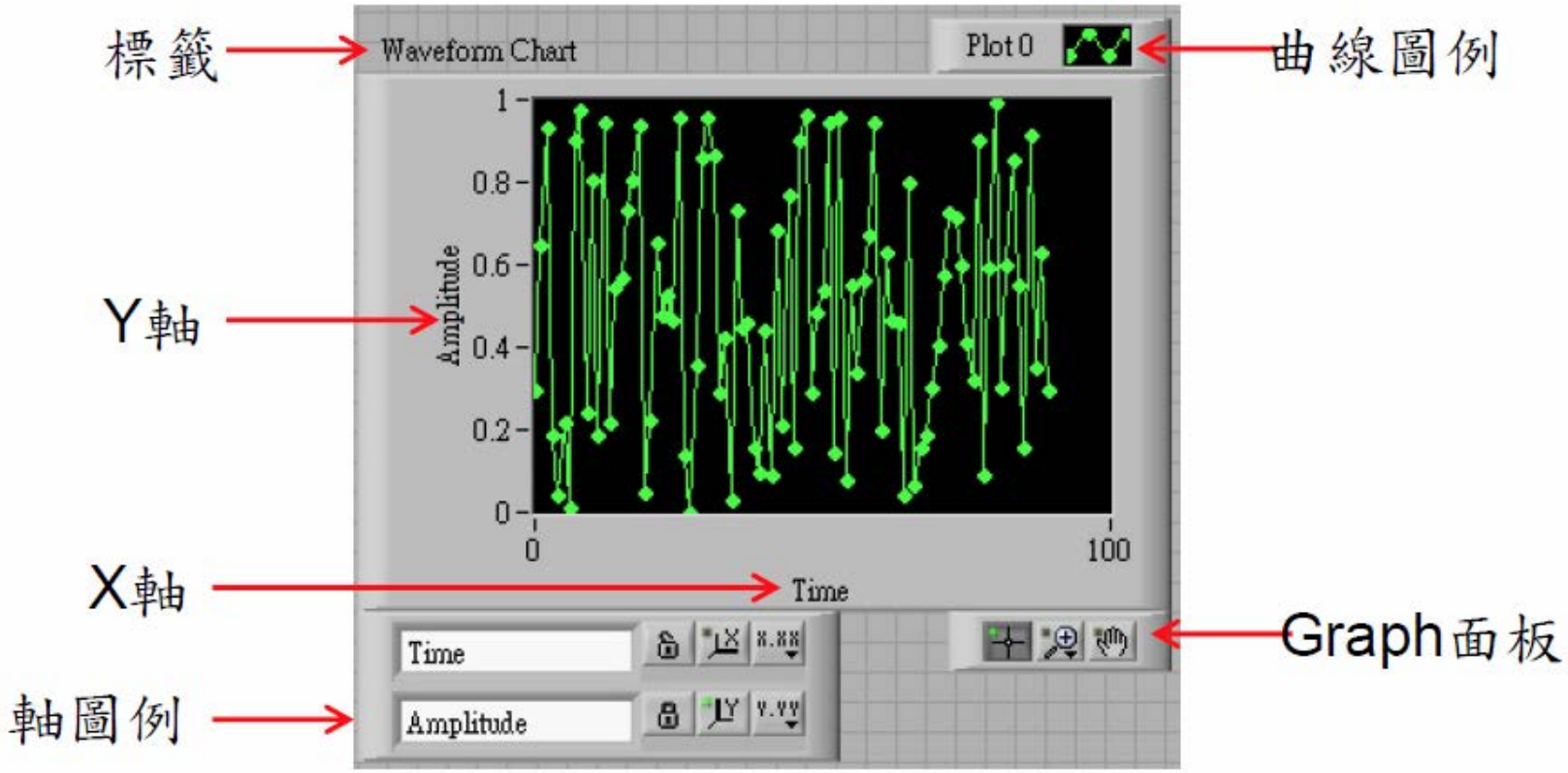
- 圖表大致分為兩類：元素輸入(Chart)，矩陣輸入(Graph)
- 暫存資料點上下限為10點 ~ 2,147,483,647點





作者；曾智勇,chihyungtseng@hotmail.com

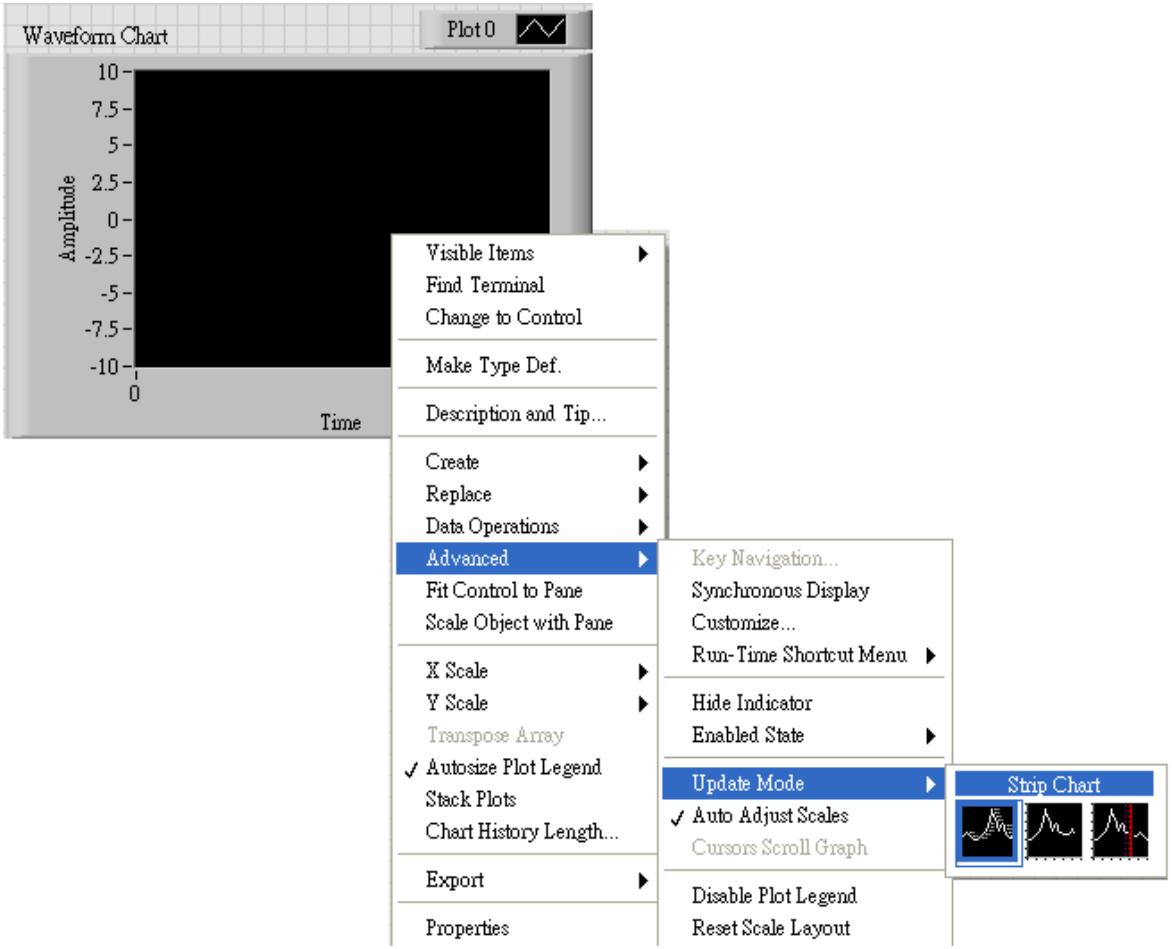
圖表



作者；曾智勇, chihyungtseng@hotmail.com

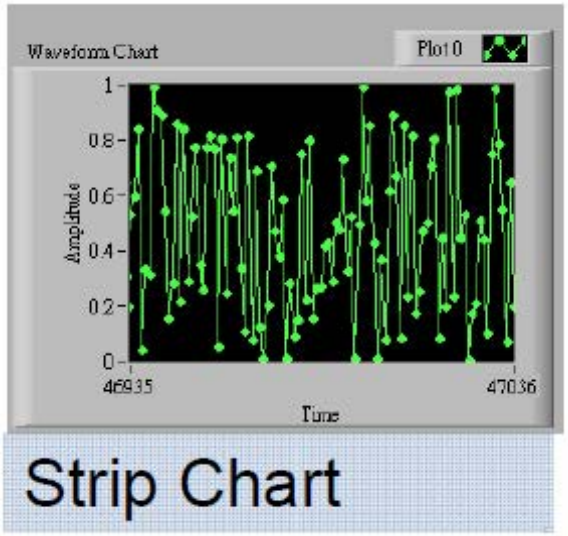
圖表

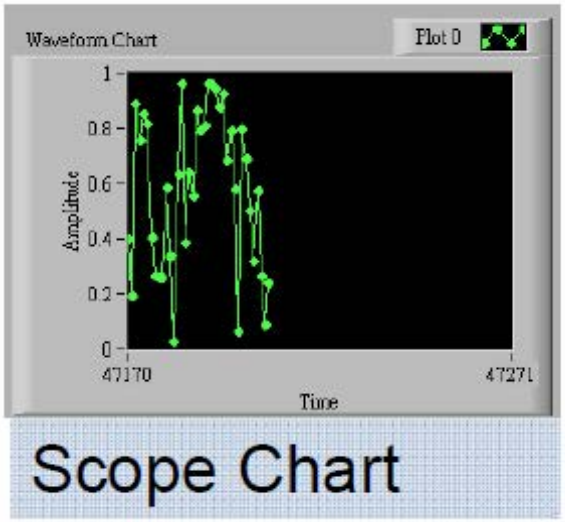
- Chart圖表，隨時間更新。具有三種資料更新方式。

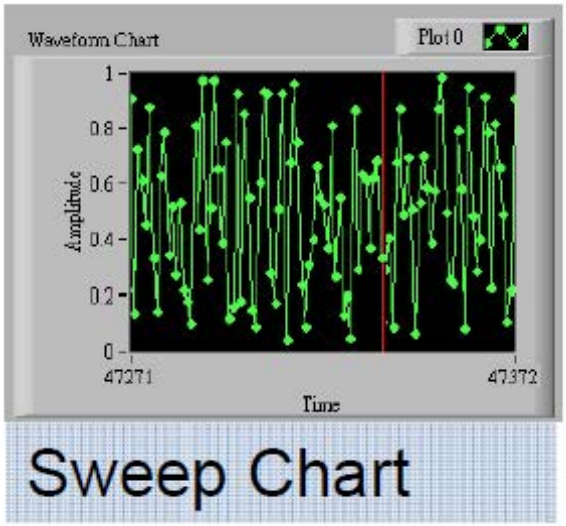


圖表

- 帶狀圖表(strip chart)：從左向右捲動圖表以持續顯示執行資料
- 範圍圖表(scope chart)：圖表清空後，資料從左側向右繪製，周而復始
- 掃瞄圖表(sweep chart)：運作類似範圍圖表，但是舊的資料不清空，以紅色垂直線做為區隔，新資料顯示在左側，舊資料顯示在右側



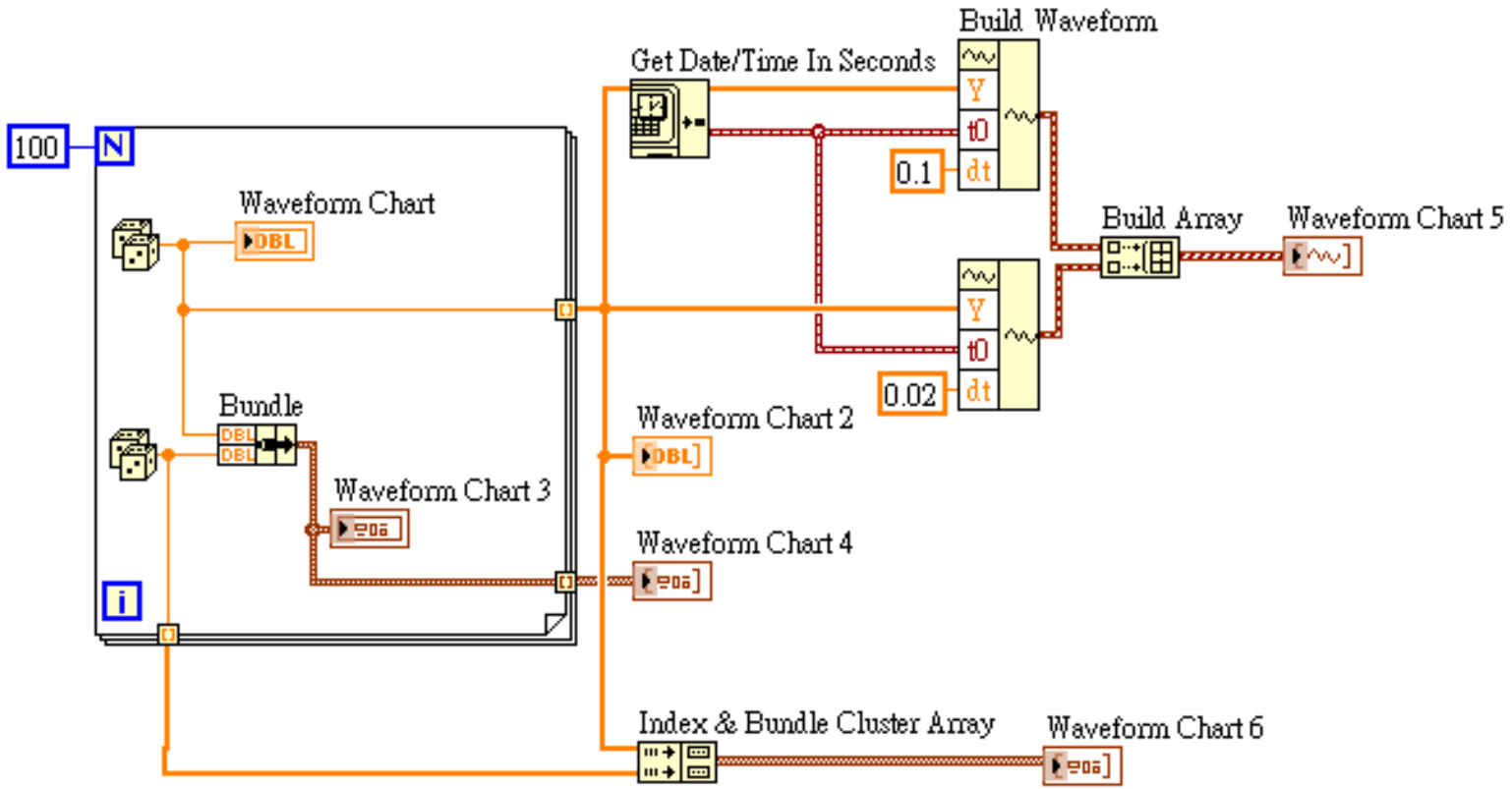




作者；曾智勇,chihyungtseng@hotmail.com

圖表

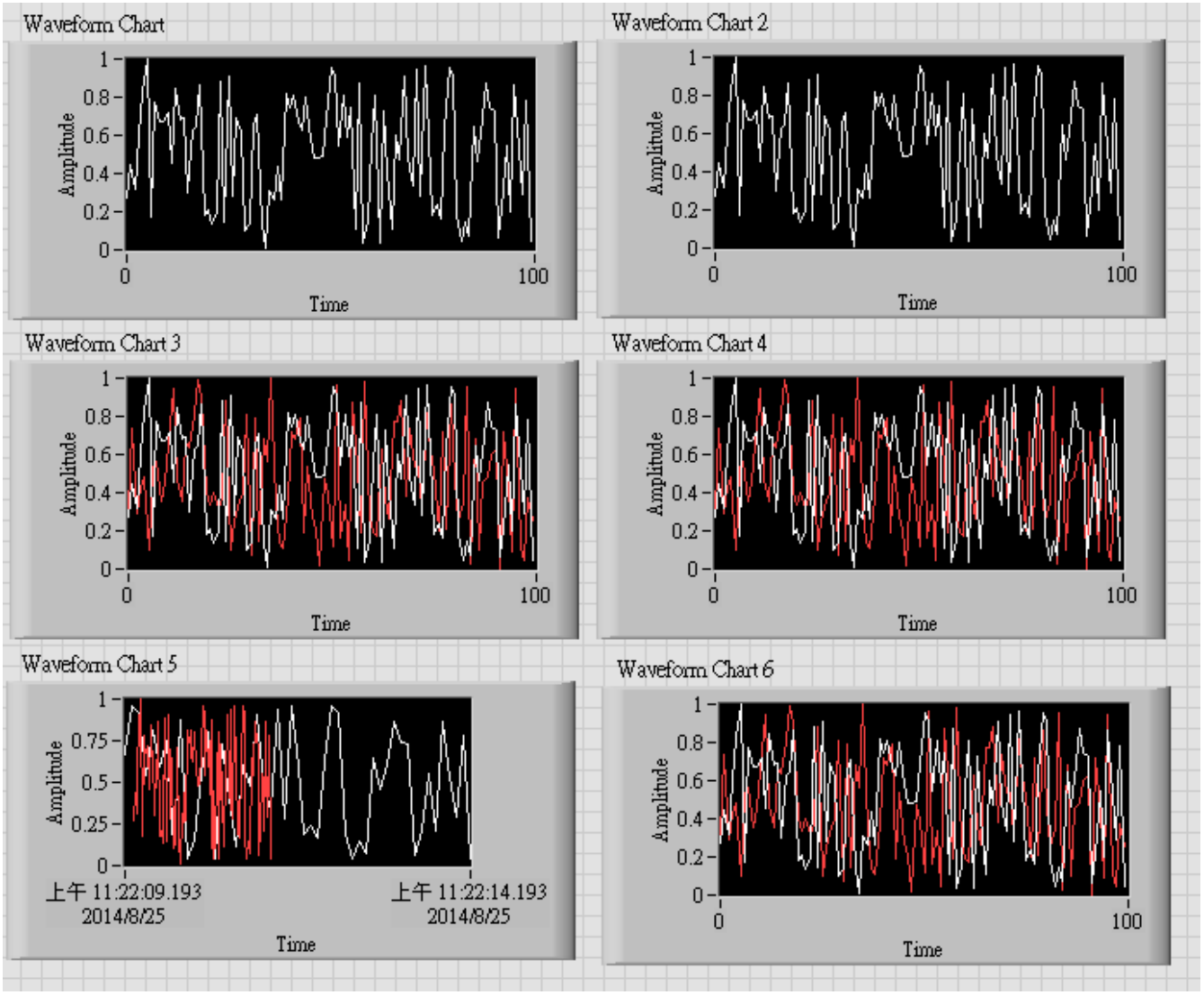
- Chart，各種做圖方法



作者；曾智勇, chihyungtseng@hotmail.com

圖表

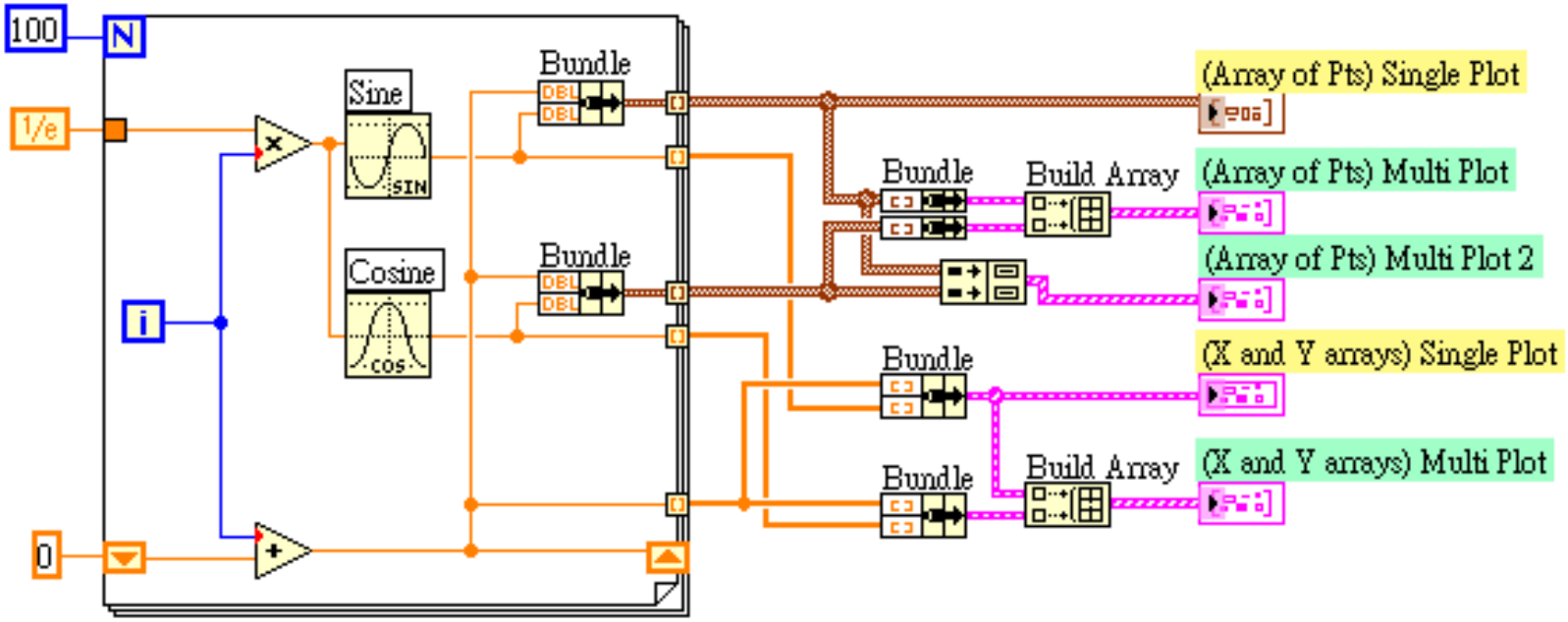
- Chart



作者；曾智勇, chihyungtseng@hotmail.com

圖表

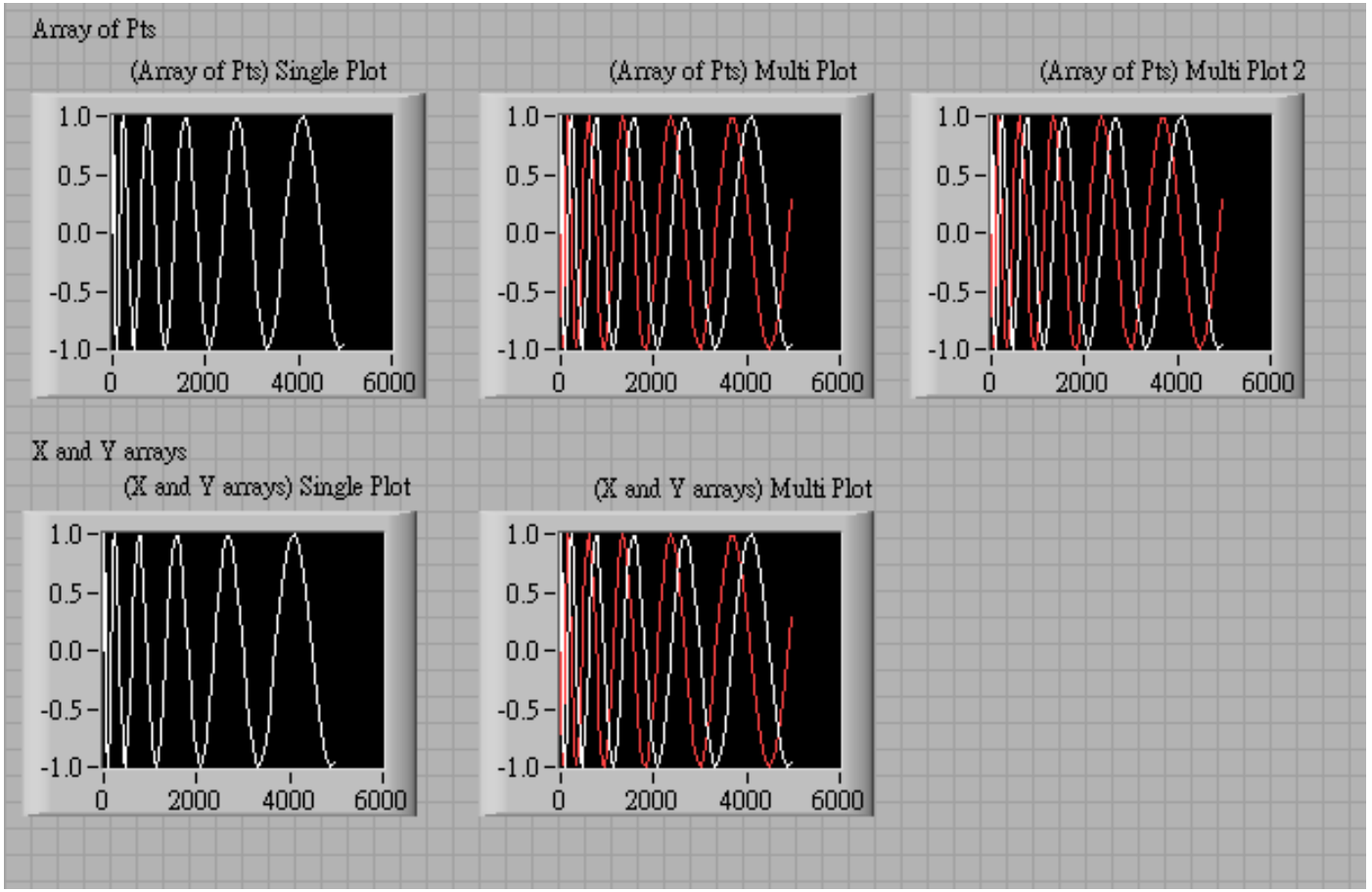
- Graph，各種做圖方法



作者；曾智勇,chihyungtseng@hotmail.com

圖表

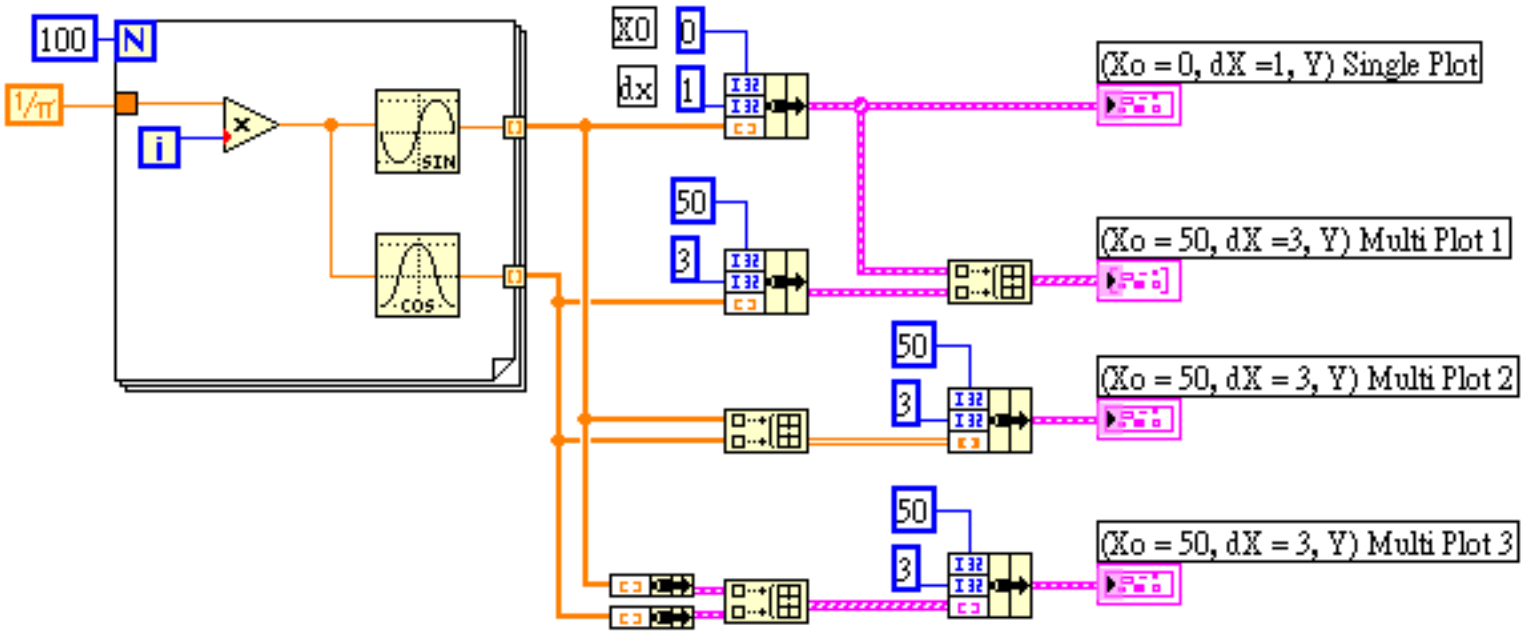
- Graph



作者；曾智勇, chihyungtseng@hotmail.com

圖表

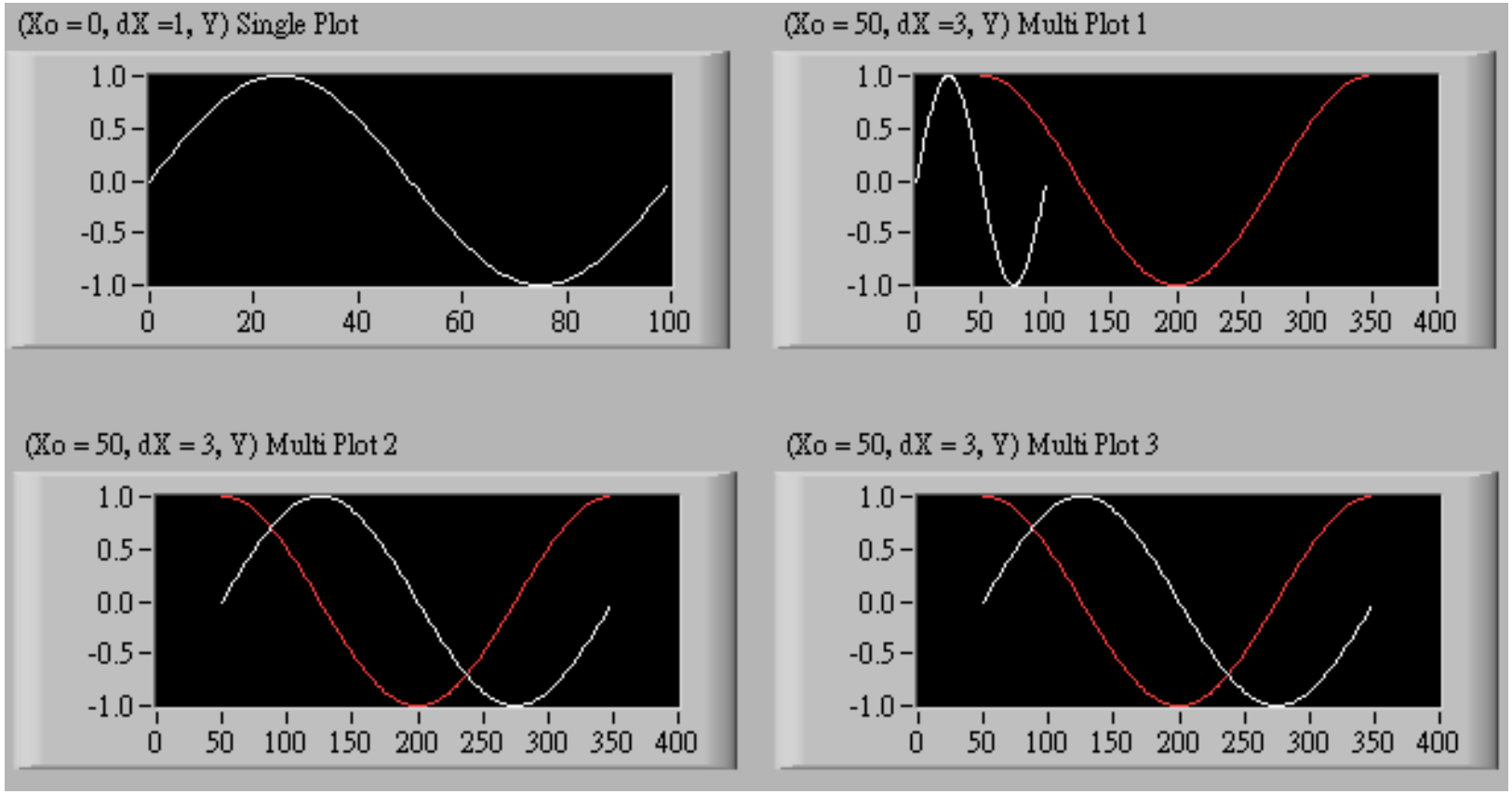
- Graph，各種做圖方法



作者；曾智勇, chihyungtseng@hotmail.com

圖表

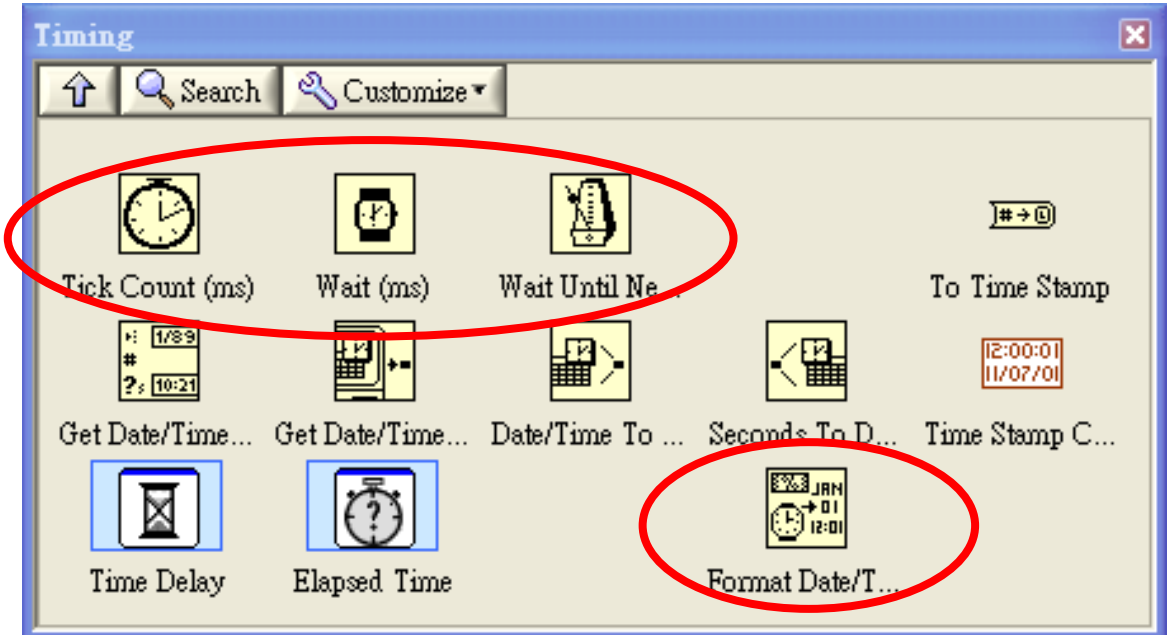
- Graph



作者；曾智勇, chihyungtseng@hotmail.com

時間

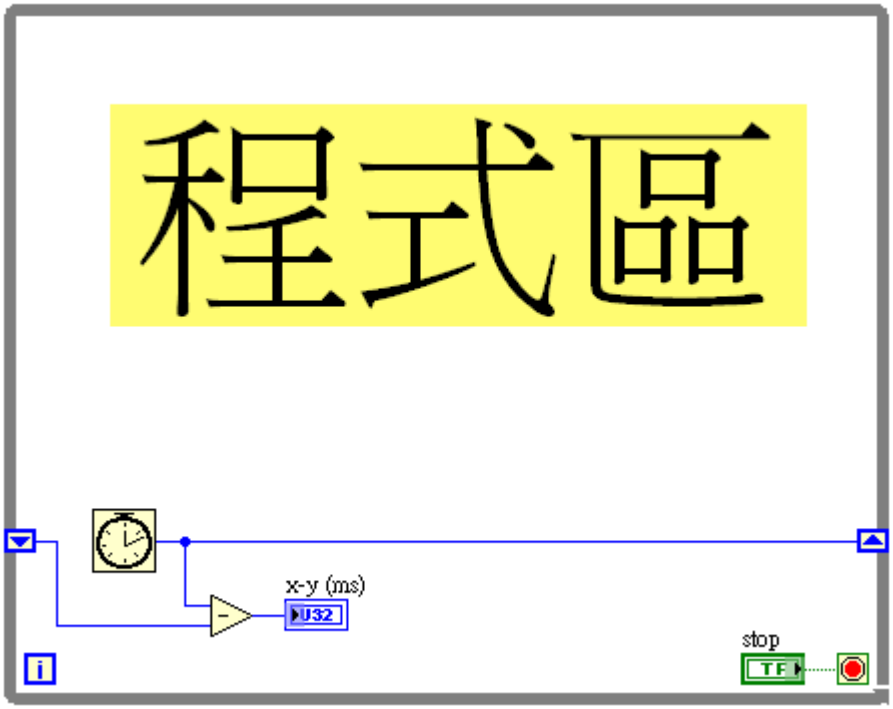
- 碼錶：計算程式運行時間。
- 計時器：迴圈或程式延遲用。
- 系統時間：用於計算時間經過、存檔、開檔。
- 常用函式：



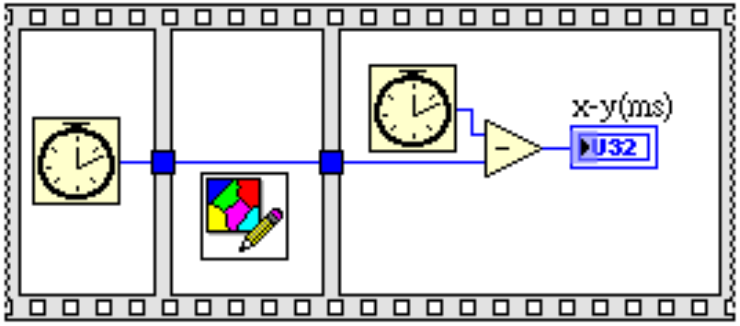
作者；曾智勇,chihyungtseng@hotmail.com

時間

- Tick Count (ms)：



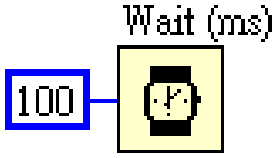
計算迴圈之間的時間差

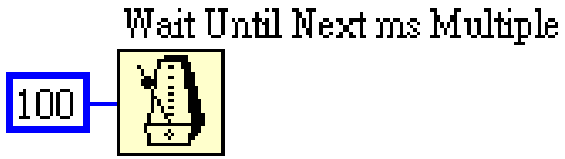


計算副程式執行所需時間

作者；曾智勇, chihyungtseng@hotmail.com

時間





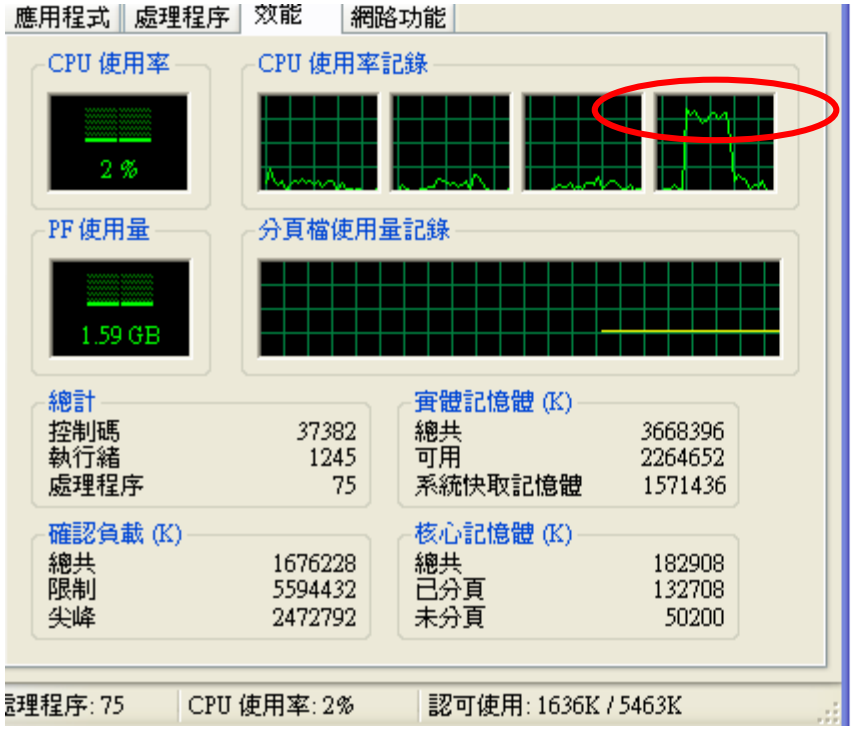
- Wait (ms)：
  Wait 元件，就字義上很容易了解到它的功能，就是一個等待的元件。當程式執行到 Wait 元件時，停留所設定的時間，單位是毫秒 (ms)，等到時間到達時，程式就會繼續往下執行。

- Wait Until Next ms Multiple (ms)：
  確認目前系統內部的計數值是否和 Wait Until Next ms Multiple 達到倍數關係，如果達到，即滿足等待的條件。舉例來說，目前系統的計數值是 2100 (ms)，如果你設 50 給 Wait Until Next ms Multiple，則程式執行到這個元件時，會等到 2150 (ms) 時才會繼續往下執行。

- Wait 及 Wait Until Next ms Multiple 的比較：
  這兩個元件最大的差別是 Wait Until Next ms Multiple會與CPU Timer對齊，因此時間誤差不會累積；但 Wait 不會對齊，因此長時間執行後誤差有可能會累積。此外，由於 Wait Until Next ms Multiple要對齊 CPU timer，因此第一個 Loop 的時間可能小於指定時間。

作者；曾智勇,chihyungtseng@hotmail.com

時間

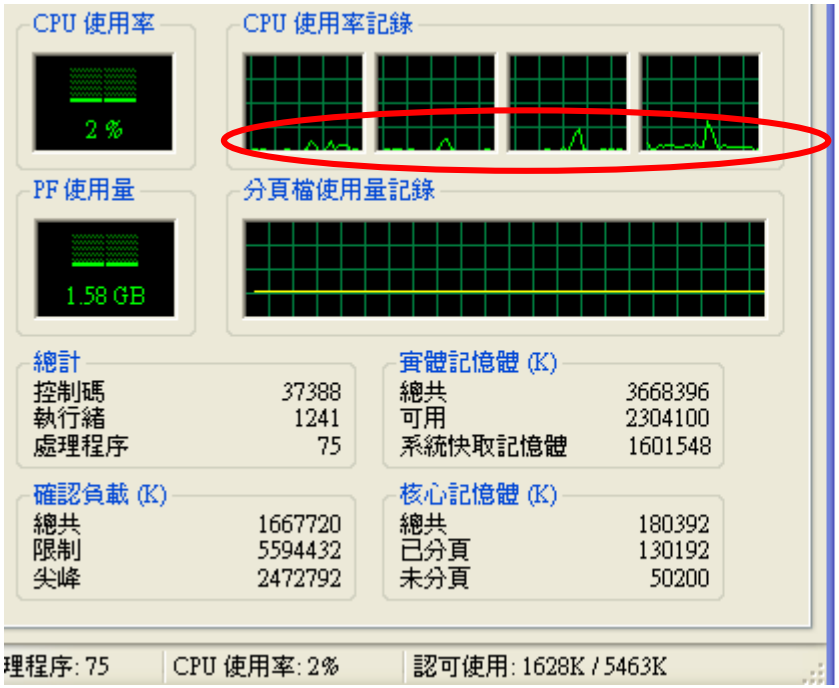


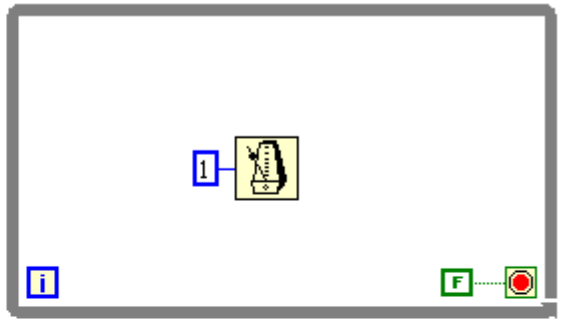
沒有延遲，則迴圈高速執行

CPU資源被完全佔用

作者；曾智勇,chihyungtseng@hotmail.com

時間





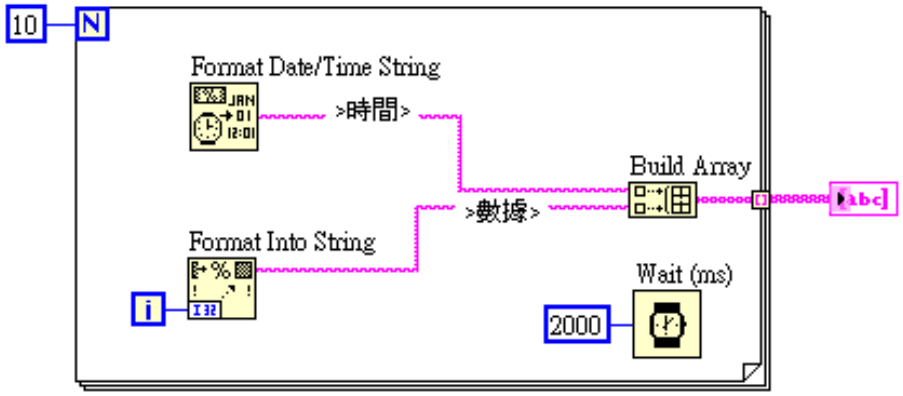
迴圈有1ms 延遲

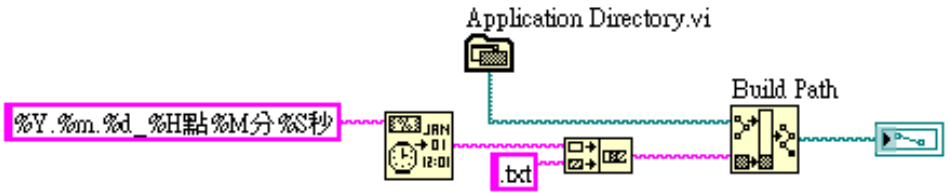
CPU資源可釋放

作者；曾智勇, chihyungtseng@hotmail.com

時間

- Format Date/Time String：
- 可用於(時間+數據)輸出，可用於(時間+檔名)開檔

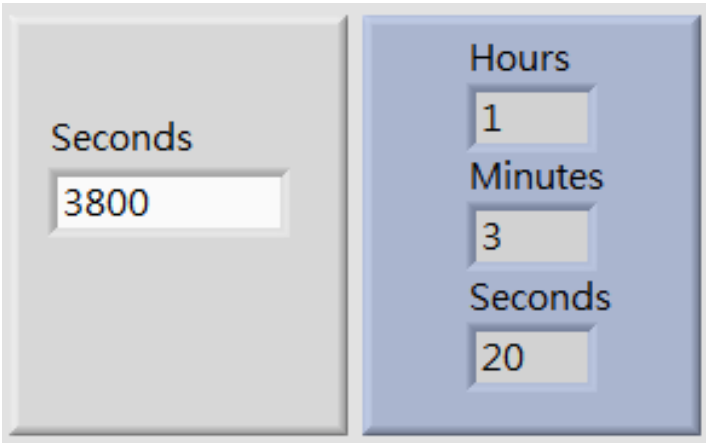


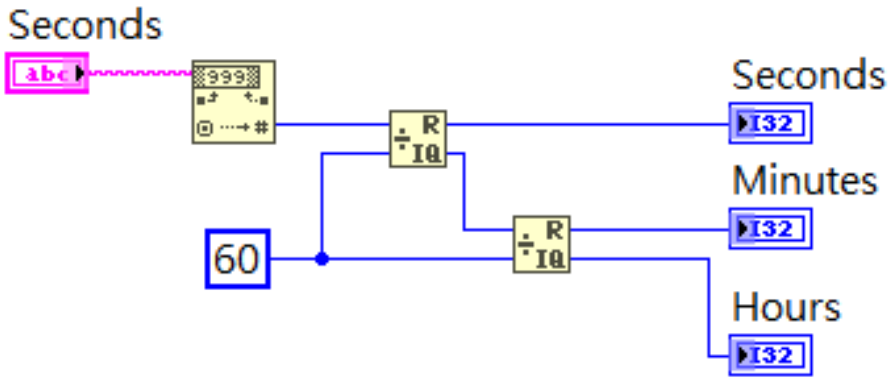


作者；曾智勇,chihyungtseng@hotmail.com

時間

- 範例：字串秒數轉數值，再計算小時、分鐘、秒鐘

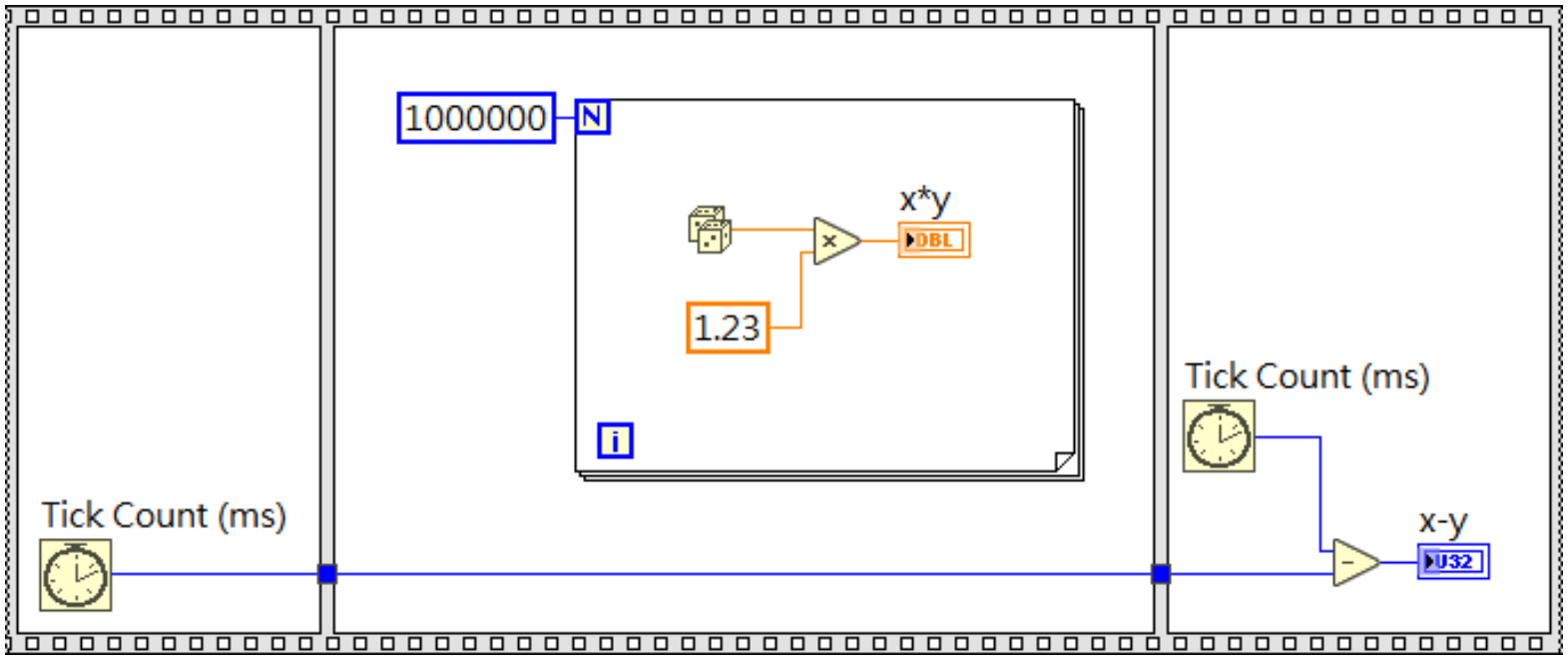




曾智勇, chihyungtseng@hotmail.com

時間

- 範例：計算程式執行時間，使用 Tick Count (ms)



作者；曾智勇,chihyungtseng@hotmail.com

元件編輯

- 使用LabVIEW元件編輯製做的iPhone手機介面

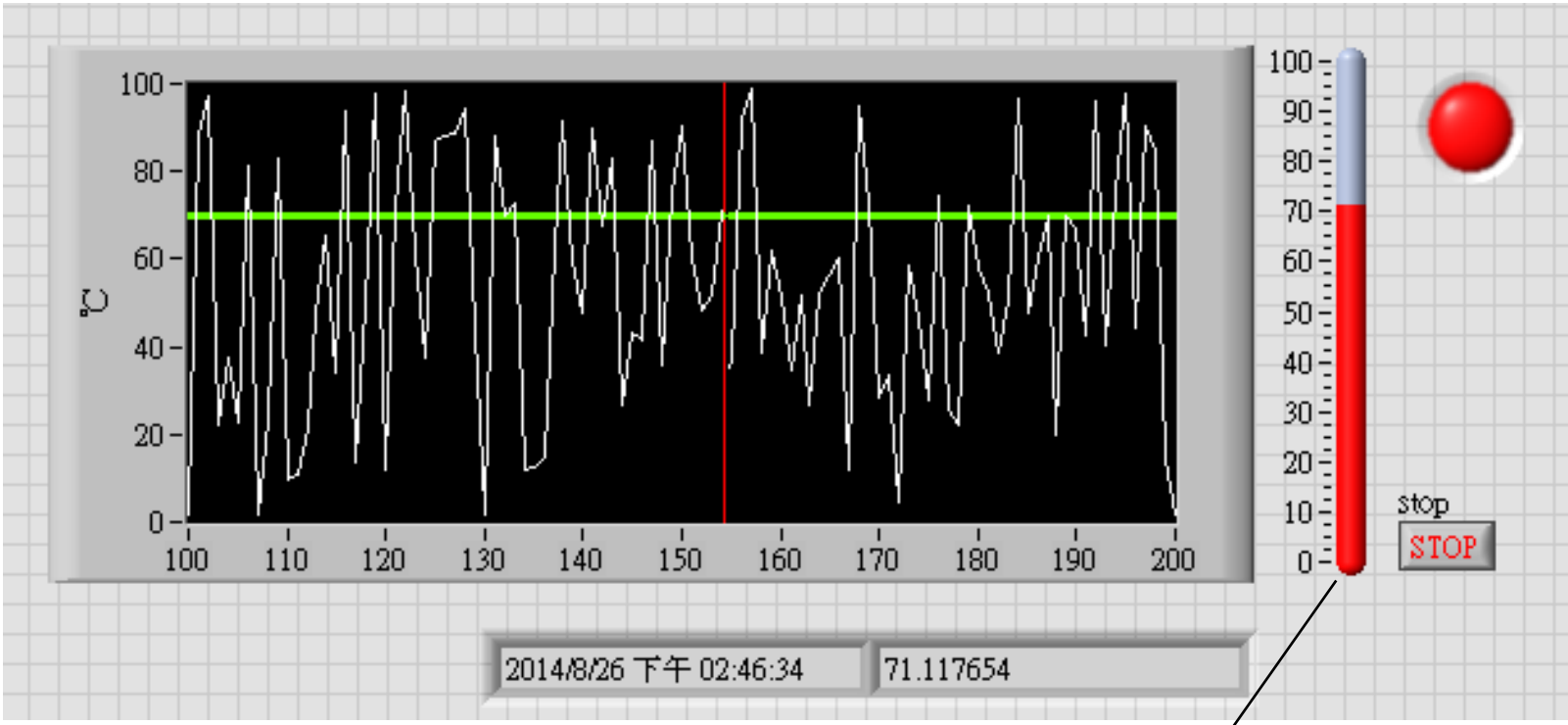
作者；曾智勇,chihyungtseng@hotmail.com

元件編輯

- 外觀：將元件設定成Control模式，資料與外觀只會被引用一次，之後編輯不會更新程式內容。
- 資料：將元件設定成**Type Definition模式(最常用)**，修改資料時，只要進行元件檔編輯，程式中所有使用該元件之資料一併更新。
- 外觀+資料：將元件設定成Strict Type Definitions模式，修改資料&外觀時，只要進行元件檔編輯，程式中所有使用該元件之資料&外觀一併更新。

作者；曾智勇,chihyungtseng@hotmail.com

元件編輯

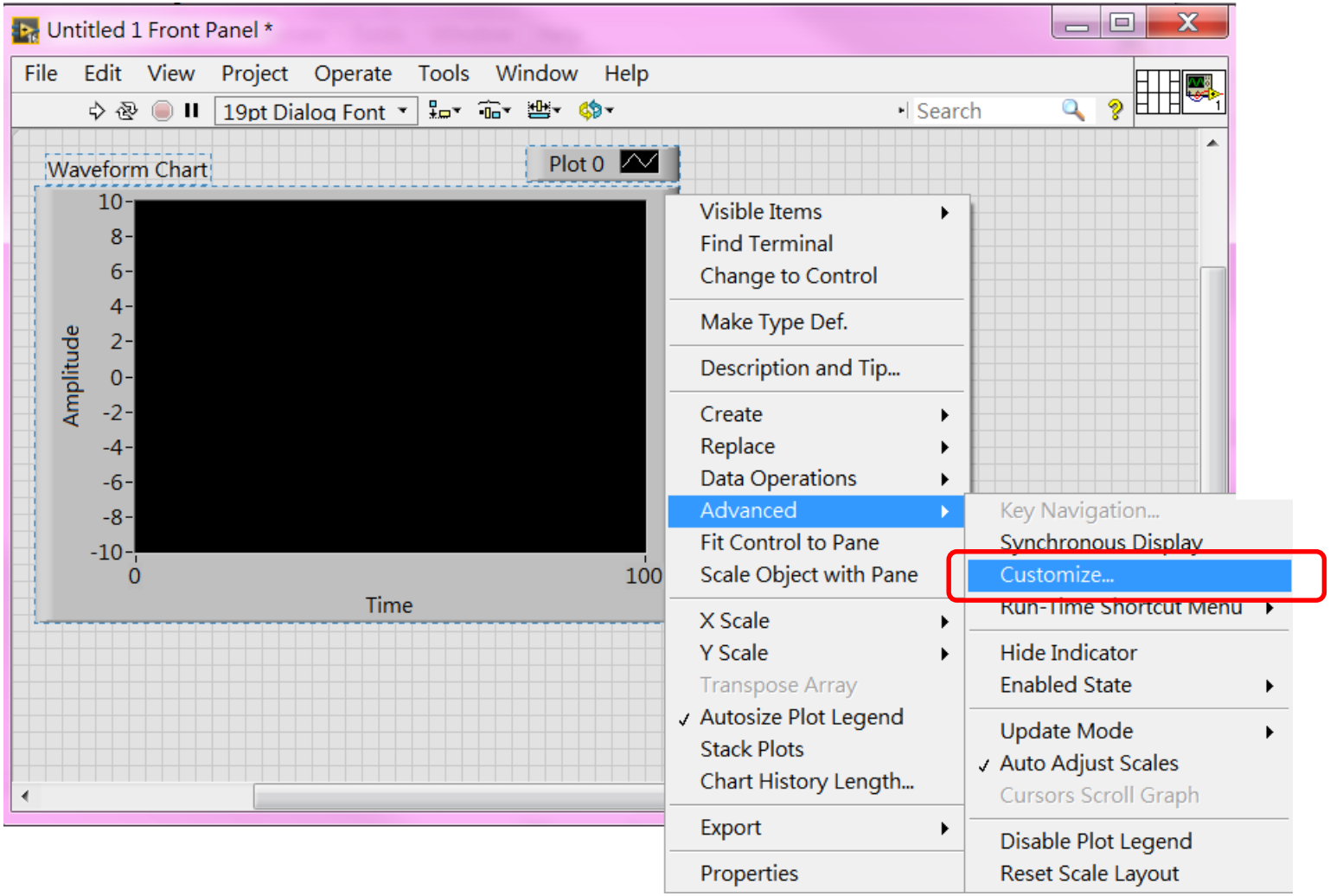


相同資料型態，數字顯示與溫度計圖示

同為數值資料型態，故可以透過元件編輯置換

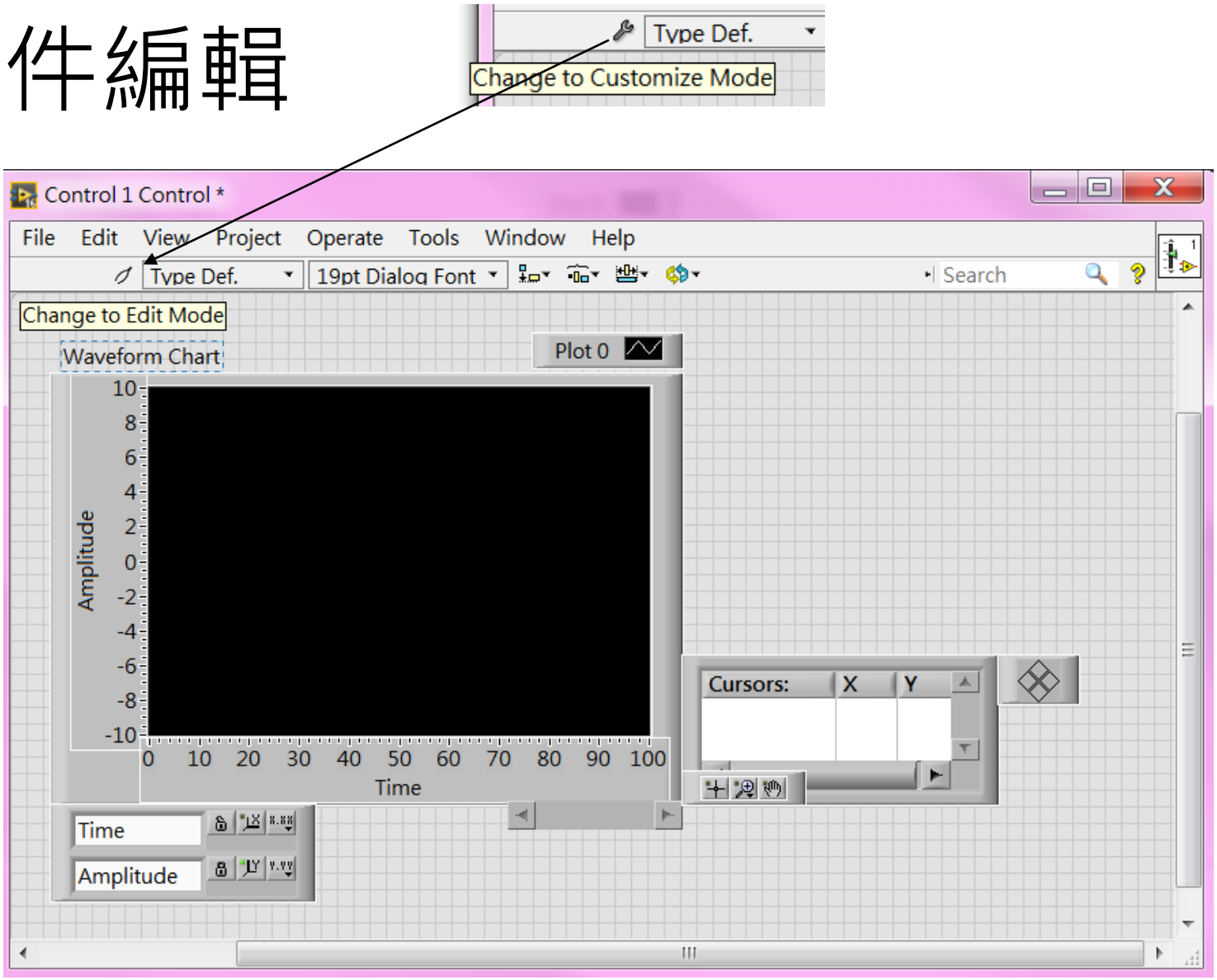
作者；曾智勇,chihyungtseng@hotmail.com

元件編輯



作者；曾智勇, chihyungtseng@hotmail.com

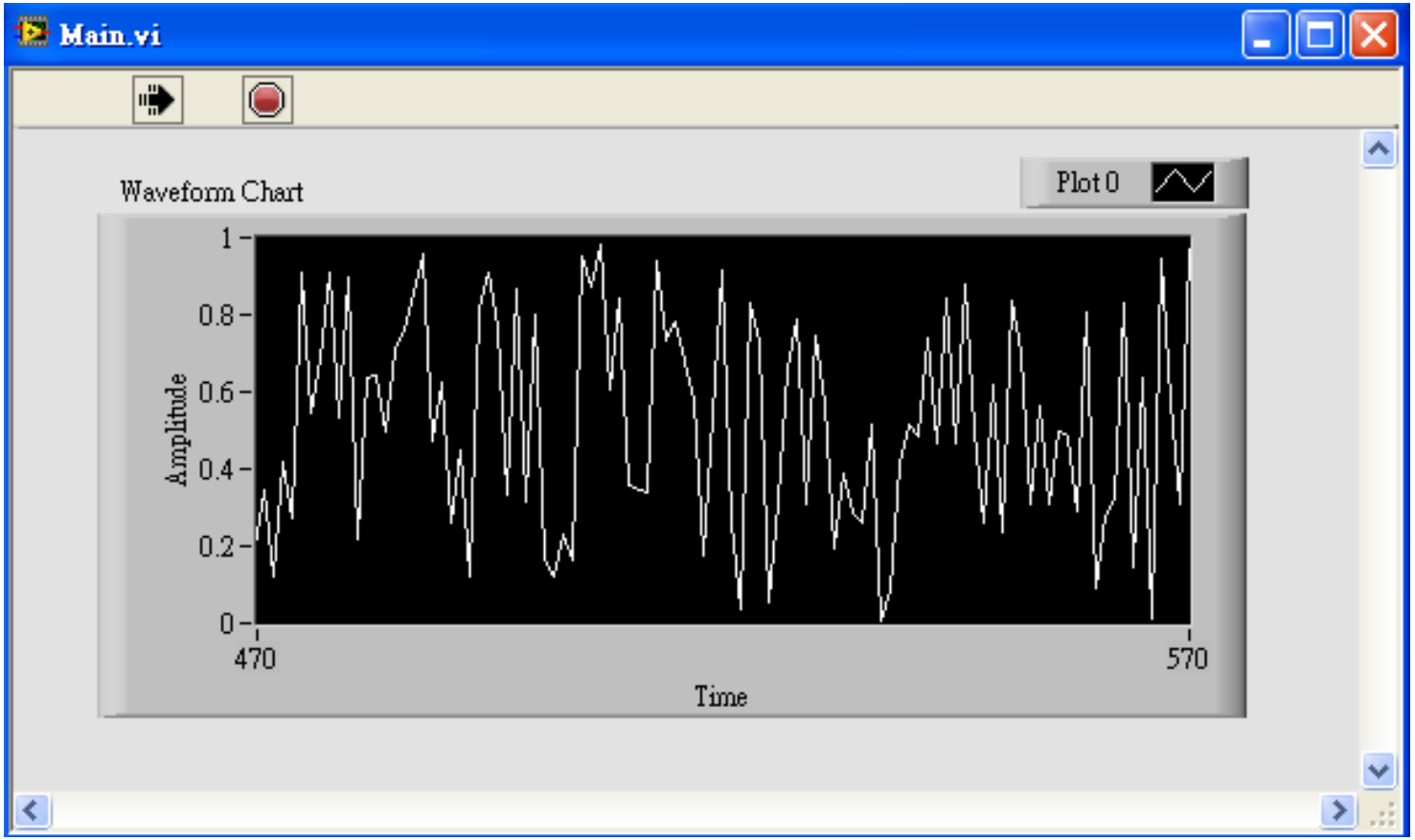
元件編輯



作者；曾智勇,chihyungtseng@hotmail.com

元件編輯

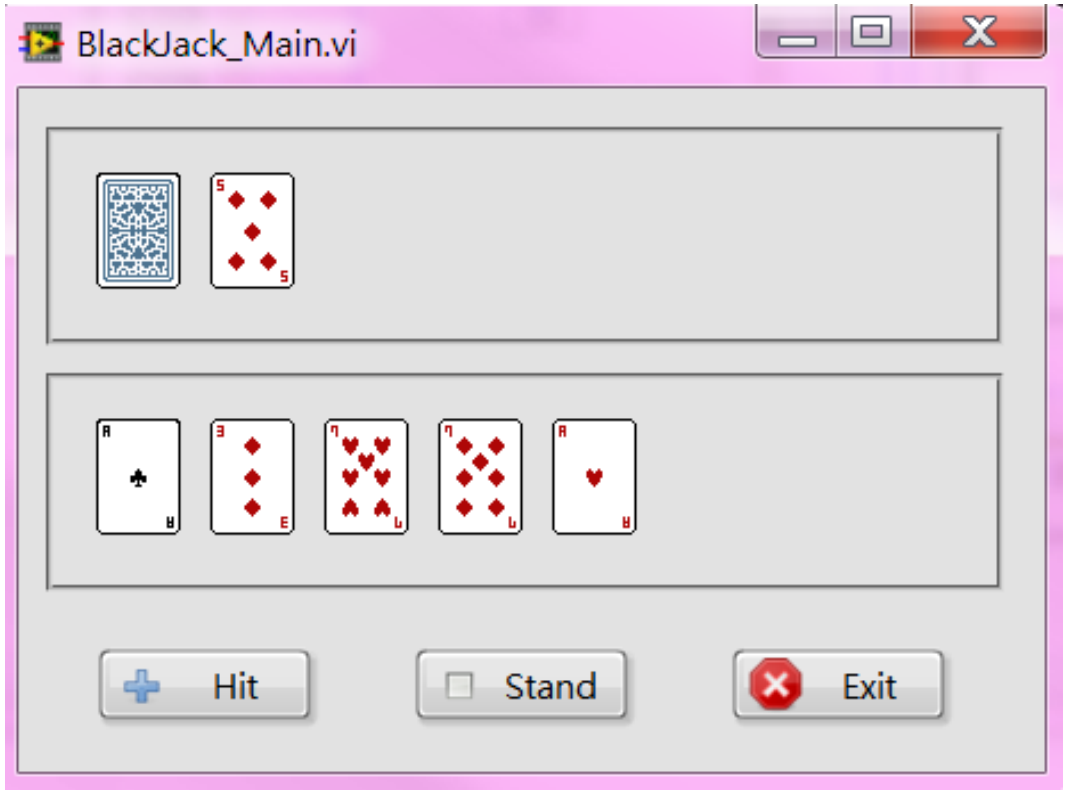
- 虛擬的Run & Stop 按鈕

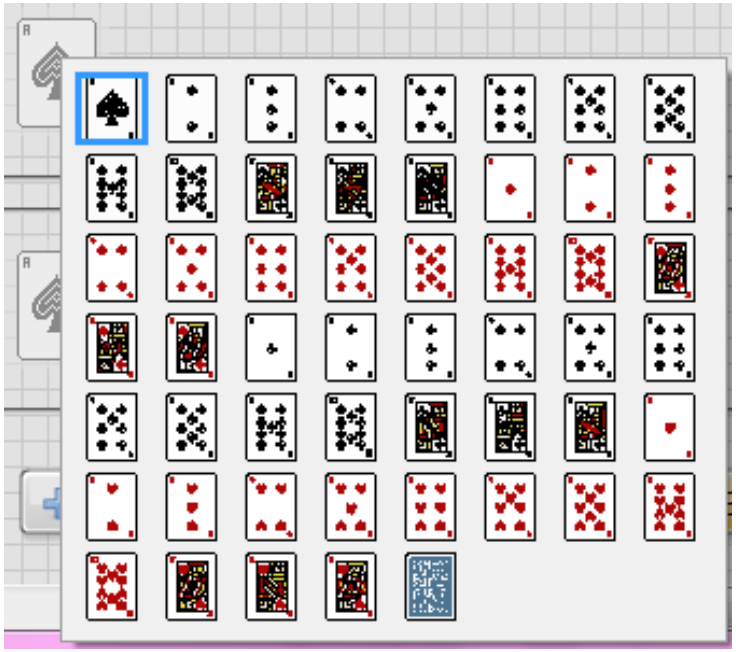


作者；曾智勇, chihyungtseng@hotmail.com

元件編輯

- 自定義圖片式選單

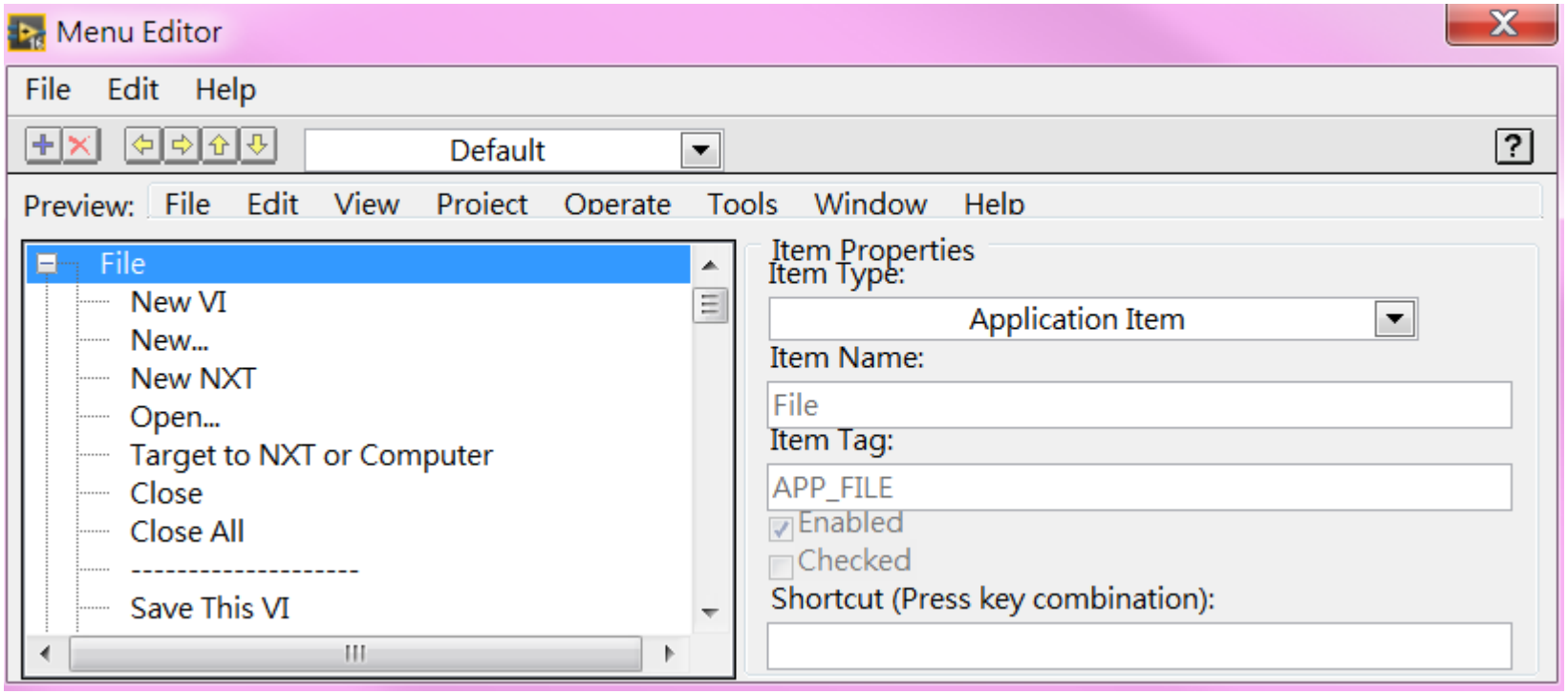




作者；曾智勇, chihyungtseng@hotmail.com

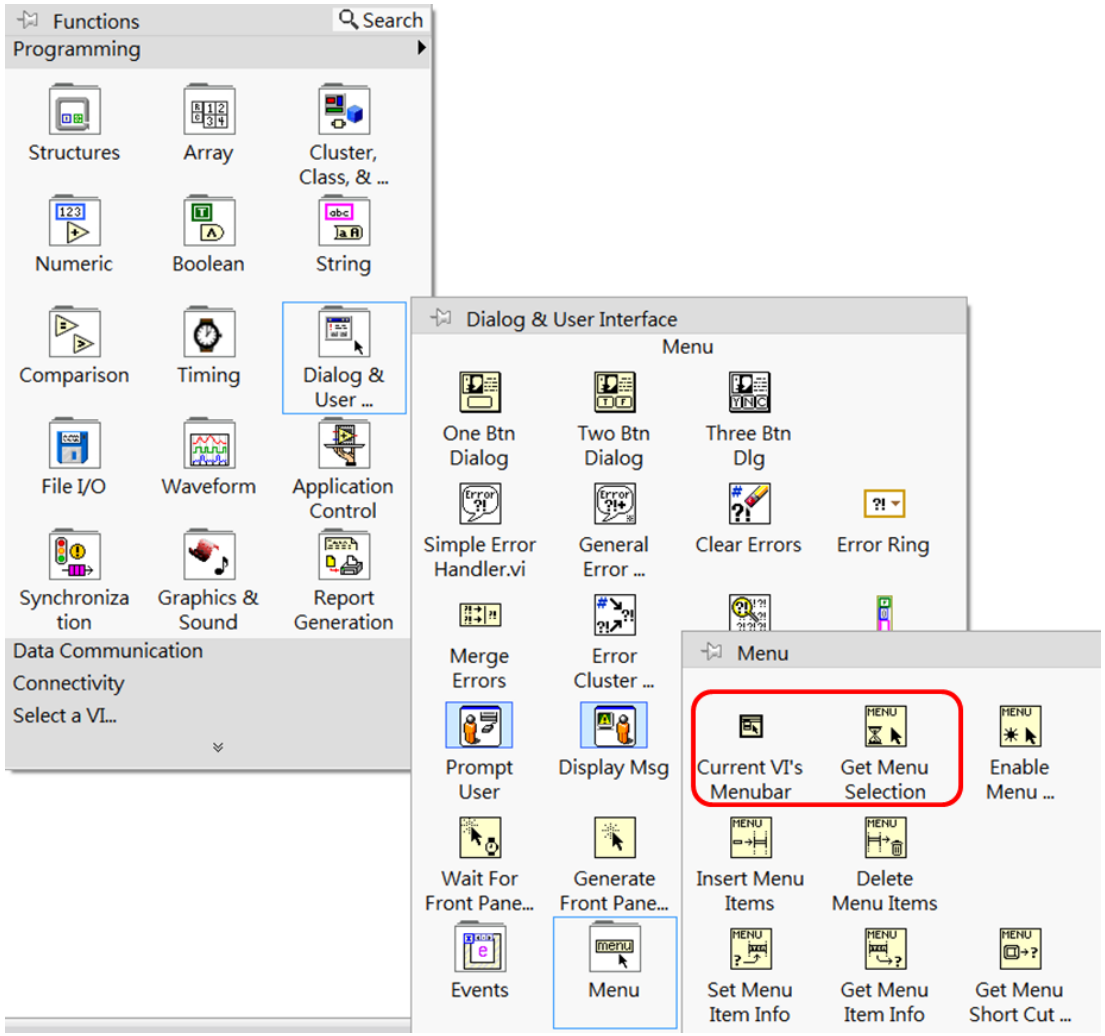
功能表編輯

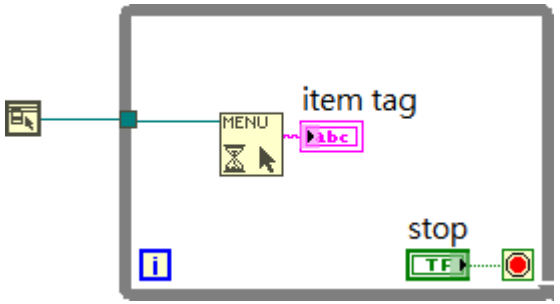
- 《Edit / Run-time Menu》



作者；曾智勇,chihyungtseng@hotmail.com

功能表編輯

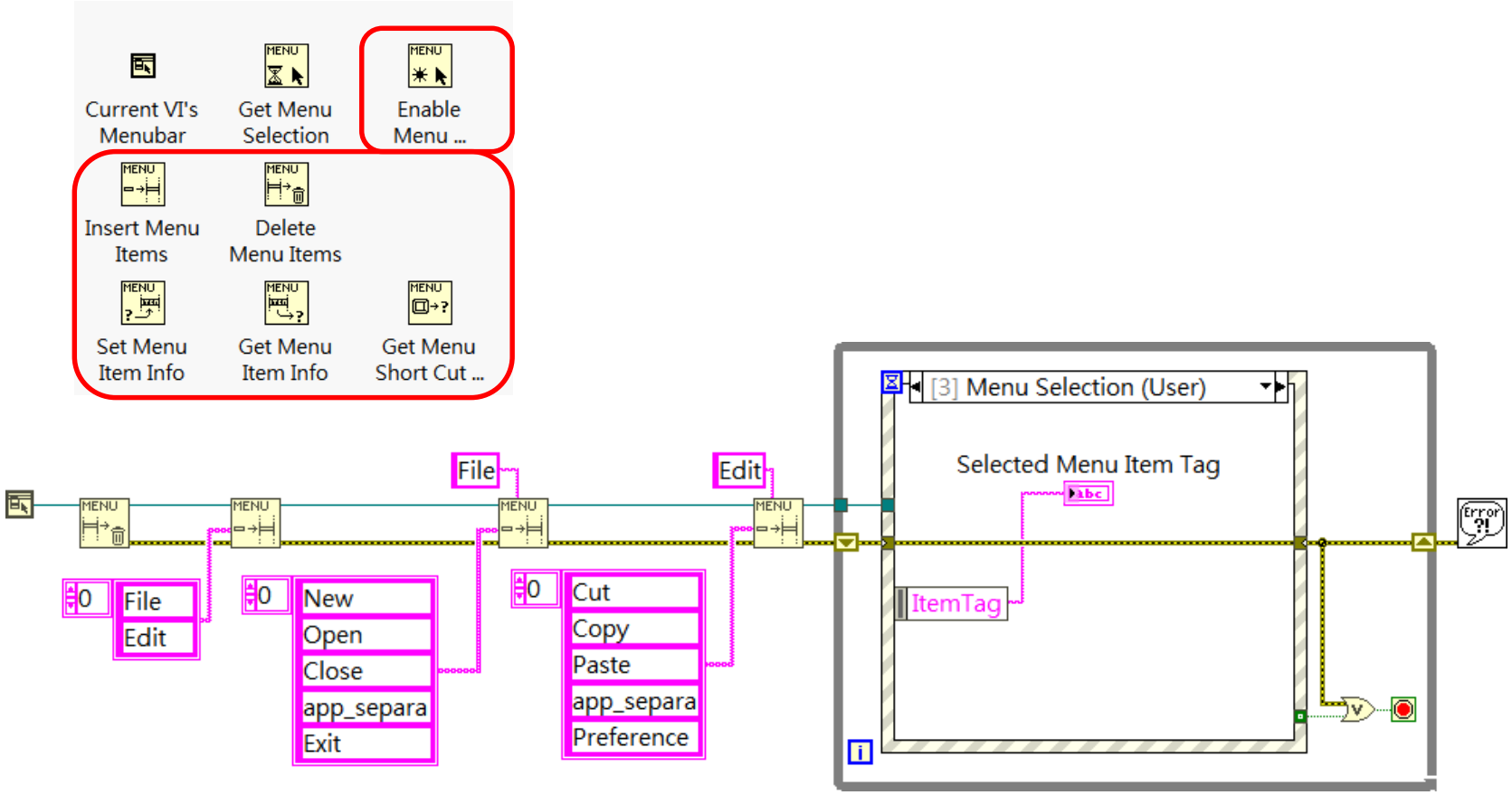




作者；曾智勇,chihyungtseng@hotmail.com

功能表編輯

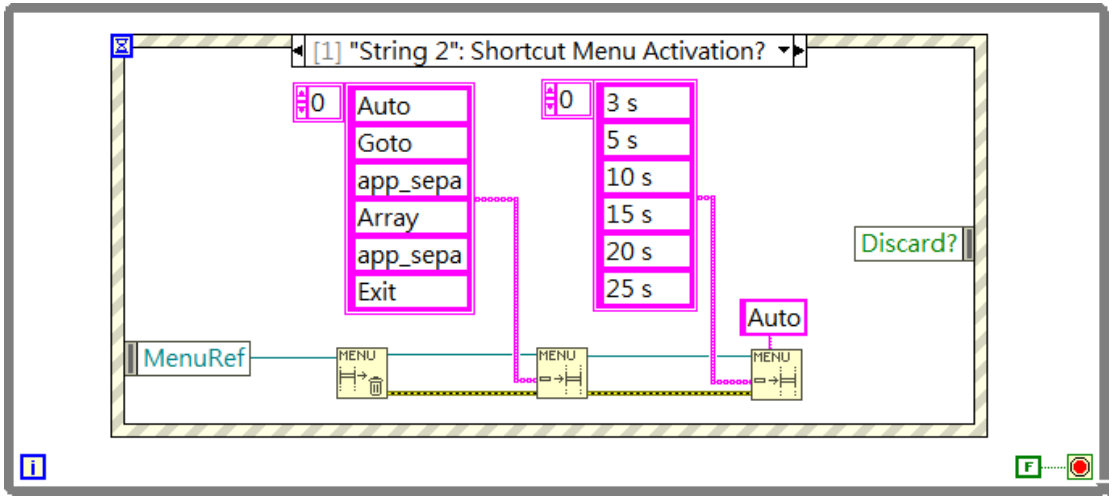
- 除了可以手動編輯，還可以在程式中 **動態編輯功能表**

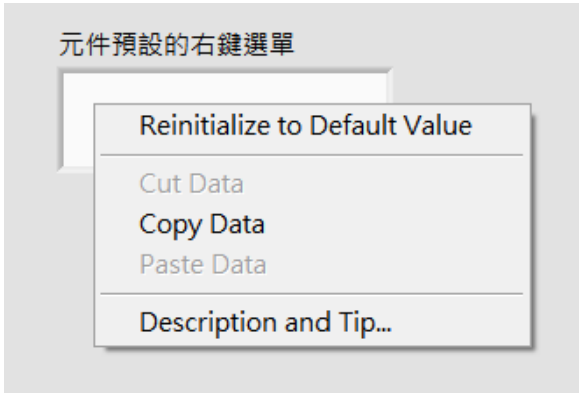


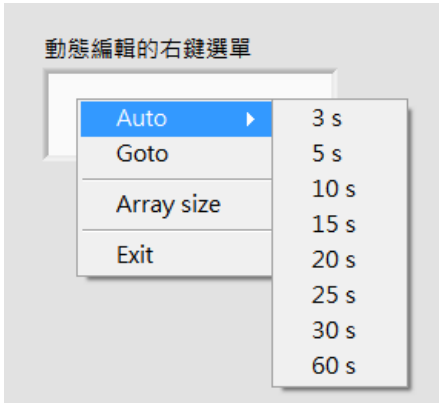
作者；曾智勇,chihyungtseng@hotmail.com

功能表編輯

- 除了可以編輯功能表，還可以針對各別元件 **動態編輯右鍵選單**



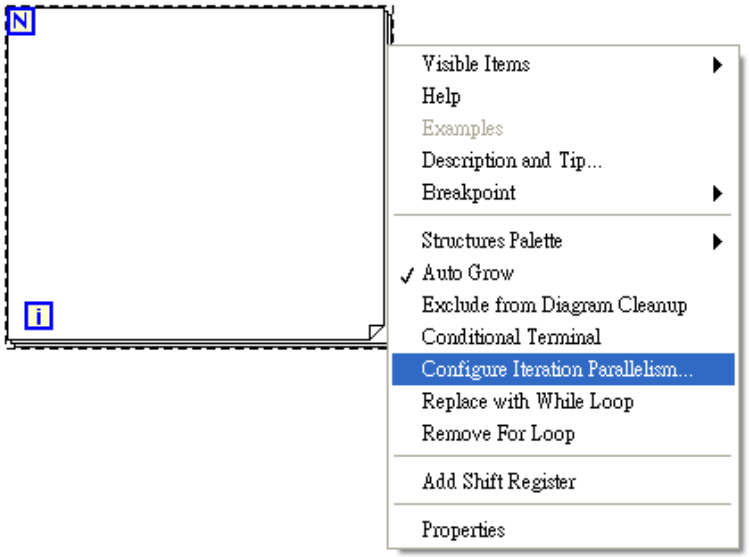


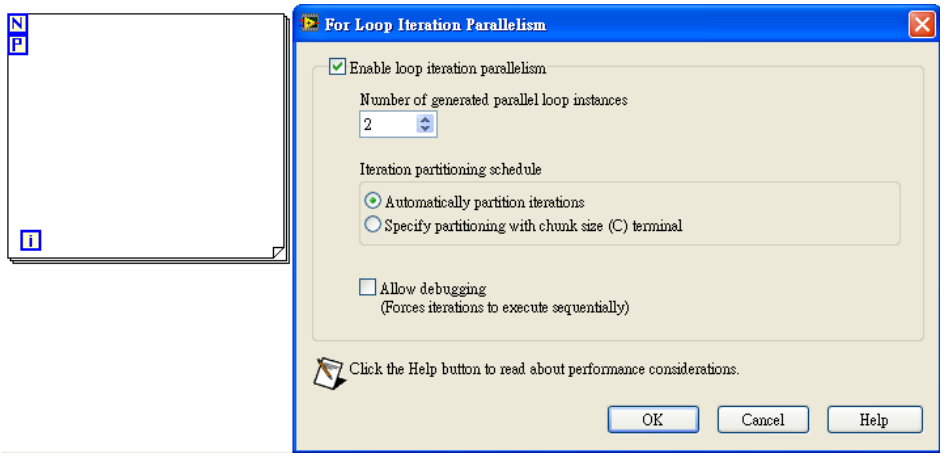


作者；曾智勇, chihyungtseng@hotmail.com

多執行緒

- 平行迴圈（Parallel For Loop）：
  100 次迴圈的 For Loop，開啟平行迴圈功能， CPU0 上執行 50 次，另 50 次則在 CPU1 上同時執行。使程式執行速度大幅的提升。

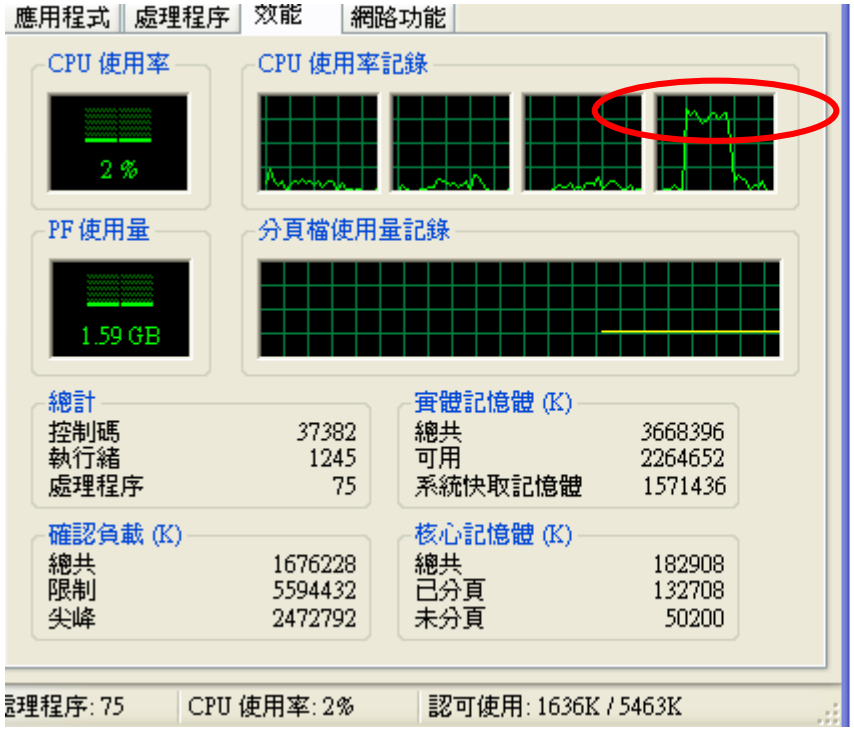




作者；曾智勇,chihyungtseng@hotmail.com

多執行緒

- 平行迴圈（While Loop）：
  當多個While Loop之間沒有任何結構限制時，自動跑多執行續。



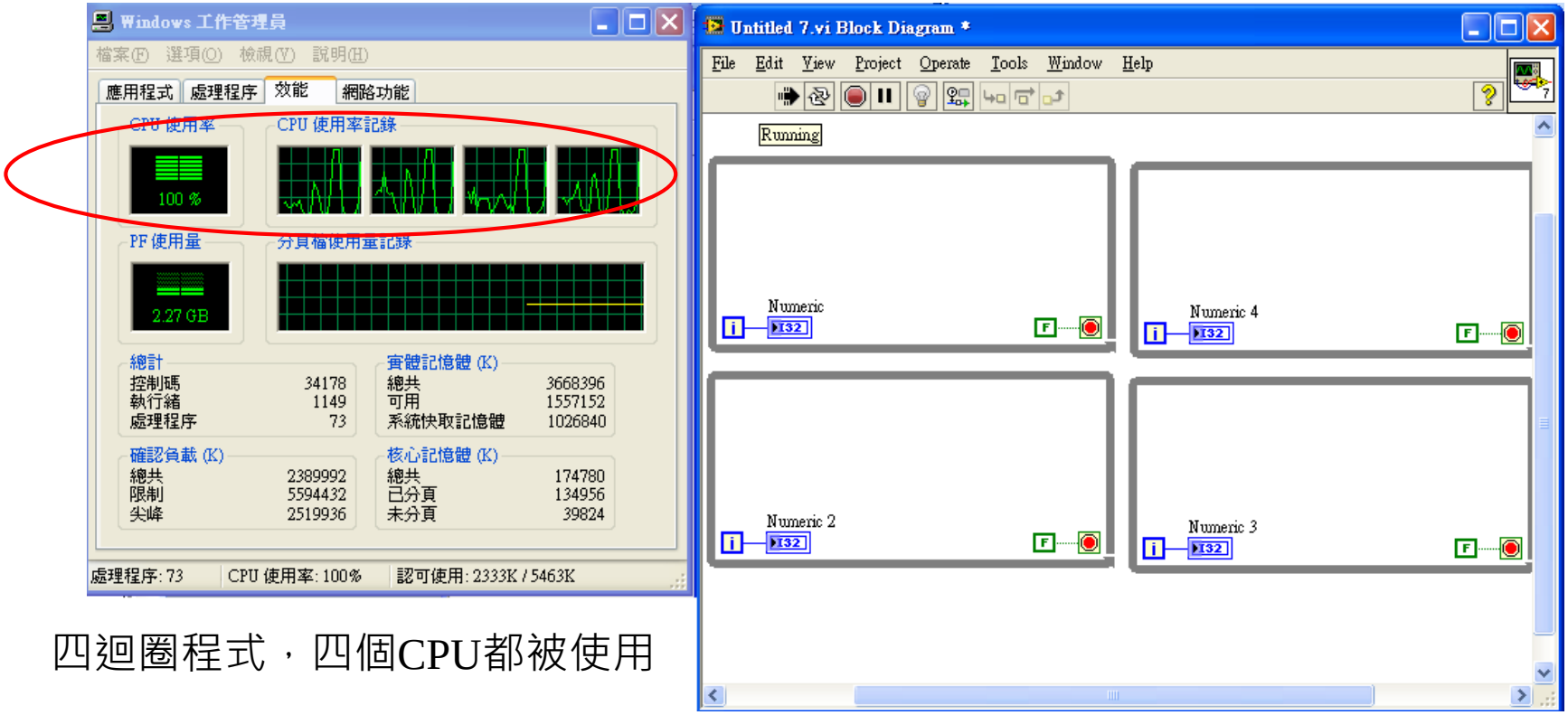
單迴圈程式，

只有一個CUP被使用。

作者；曾智勇,chihyungtseng@hotmail.com

多執行緒

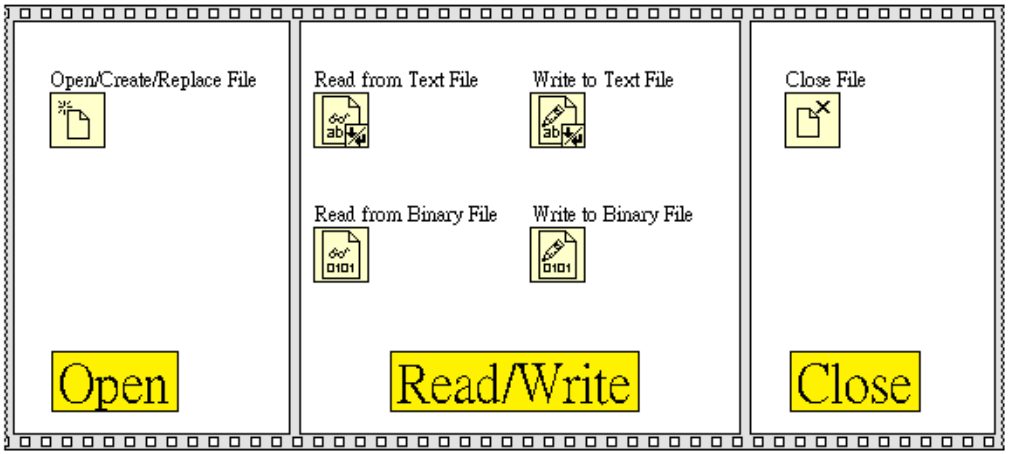
- 平行迴圈（While Loop）：
  當多個While Loop之間沒有任何結構限制時，自動跑多執行續。

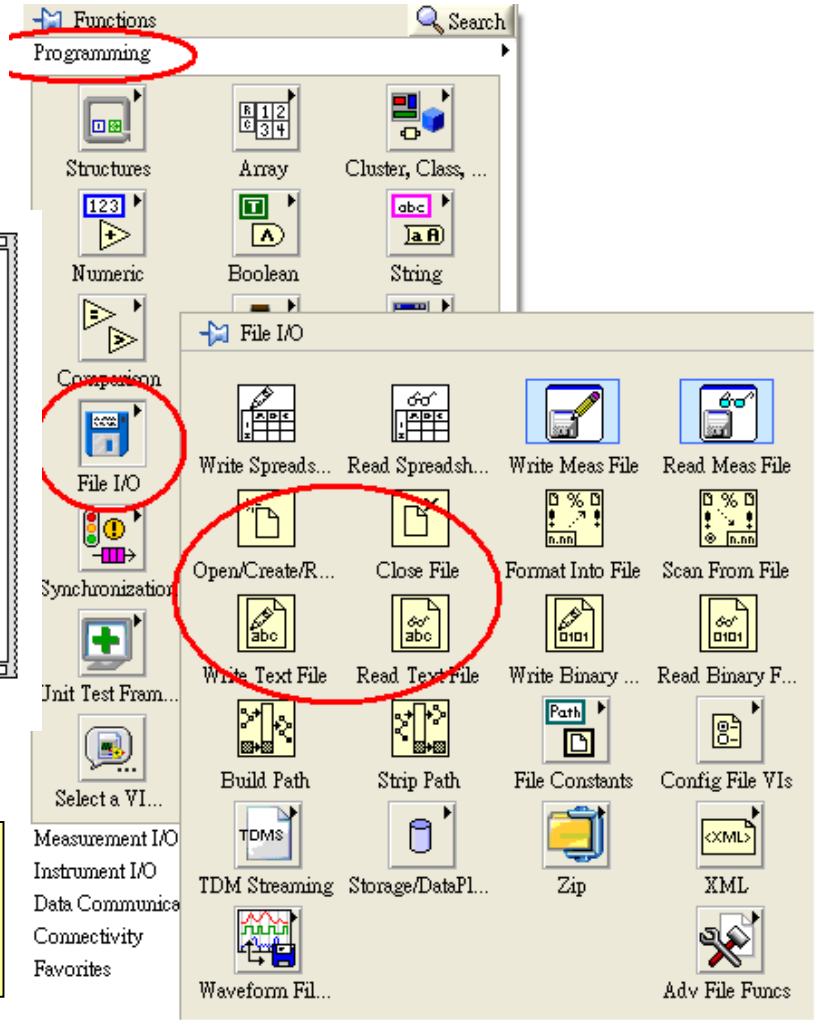


四迴圈程式，四個CPU都被使用

作者；曾智勇,chihyungtseng@hotmail.com

檔案輸入與輸出

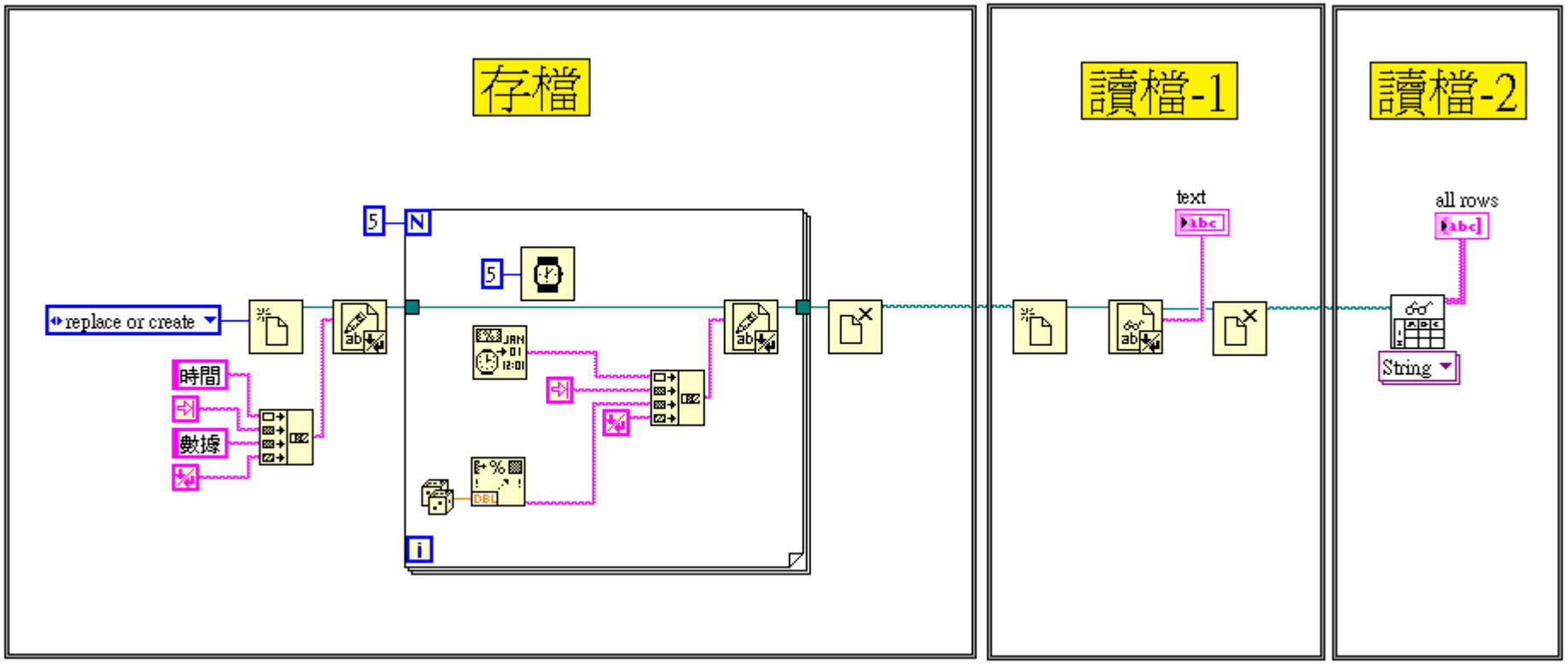




檔案請存成純文字檔
csv/txt/ini/xml

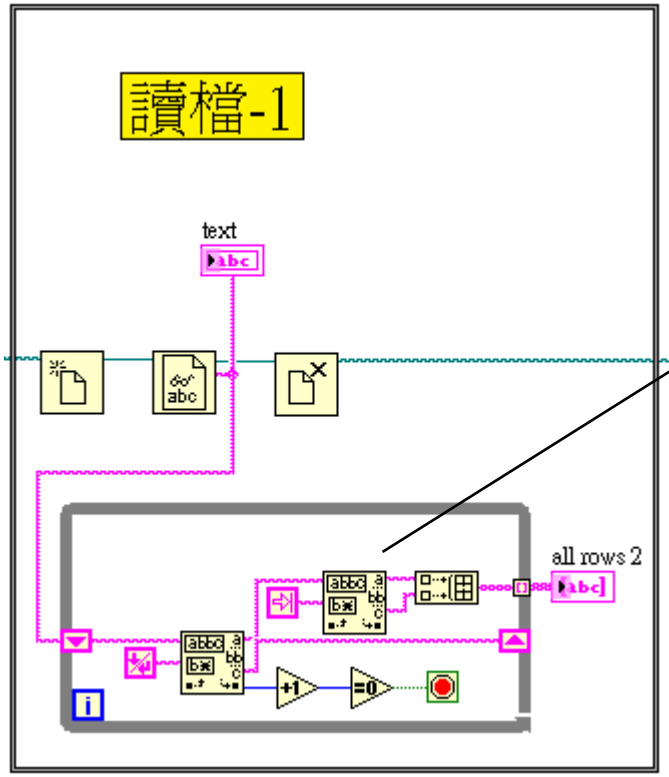
作者；曾智勇, chihyungtseng@hotmail.com

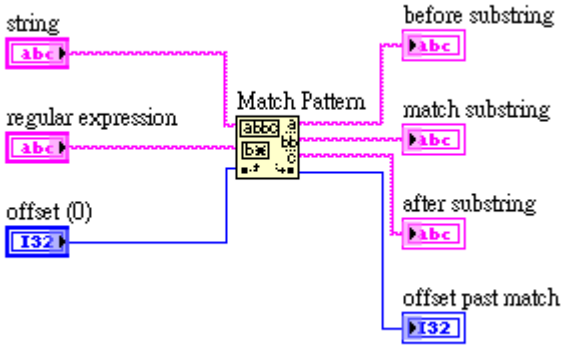
檔案輸入與輸出



作者；曾智勇,chihyungtseng@hotmail.com

檔案輸入與輸出



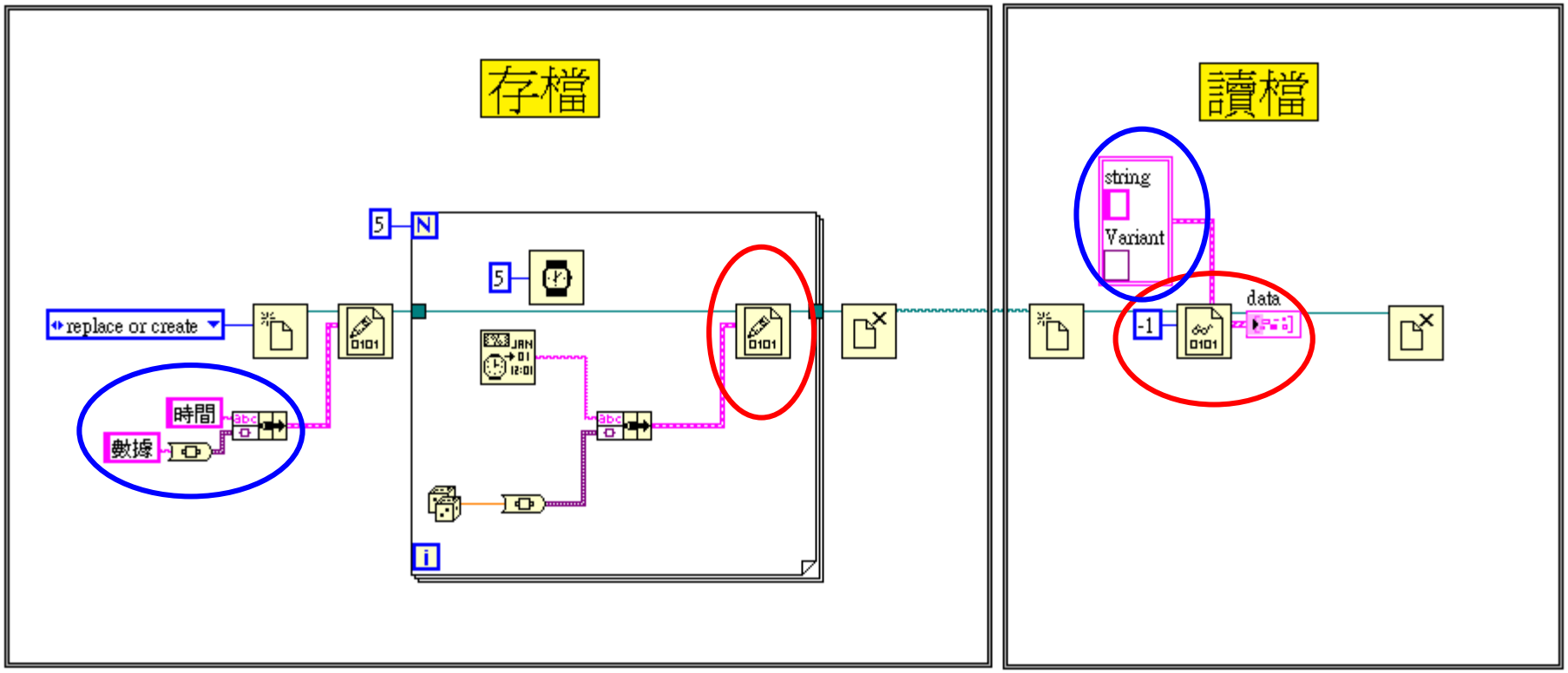


時間	數據
2014/8/26 上午	0.603970
2014/8/26 上午	0.761945
2014/8/26 上午	0.397379
2014/8/26 上午	0.287865
2014/8/26 上午	0.338905

作者；曾智勇, chihyungtseng@hotmail.com

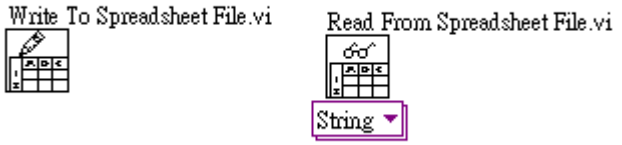
檔案輸入與輸出

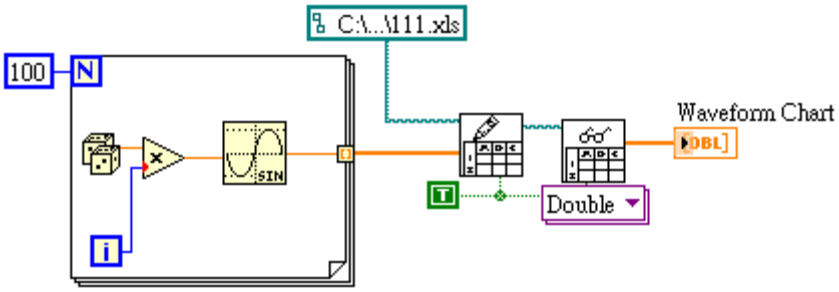
- 使用Binary模式存檔/讀檔，資料型態需一致，才能讀檔。

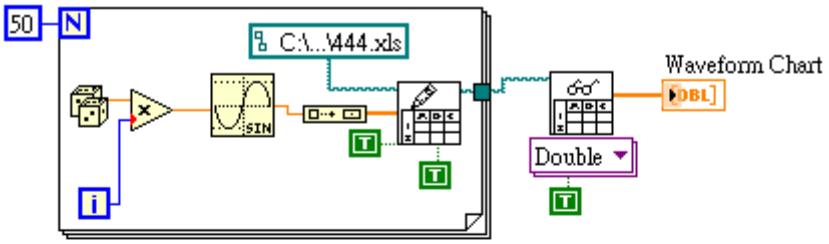


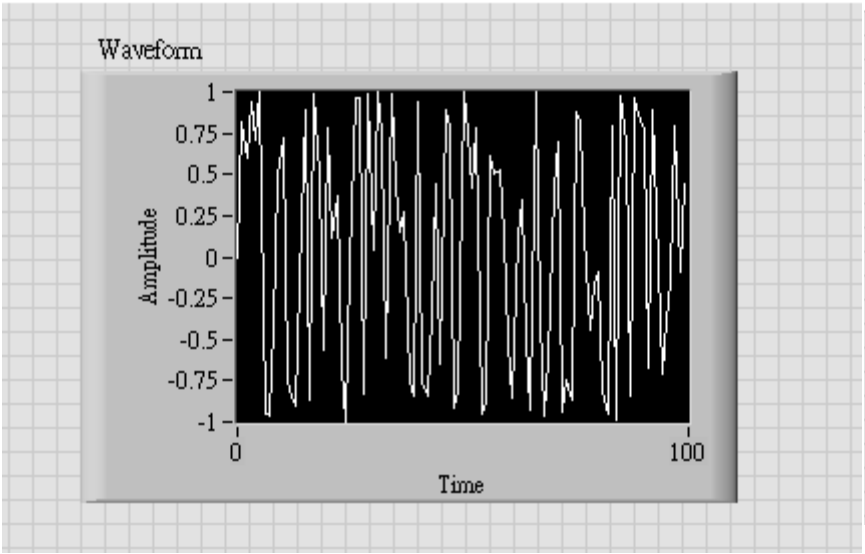
作者；曾智勇, chihyungtseng@hotmail.com

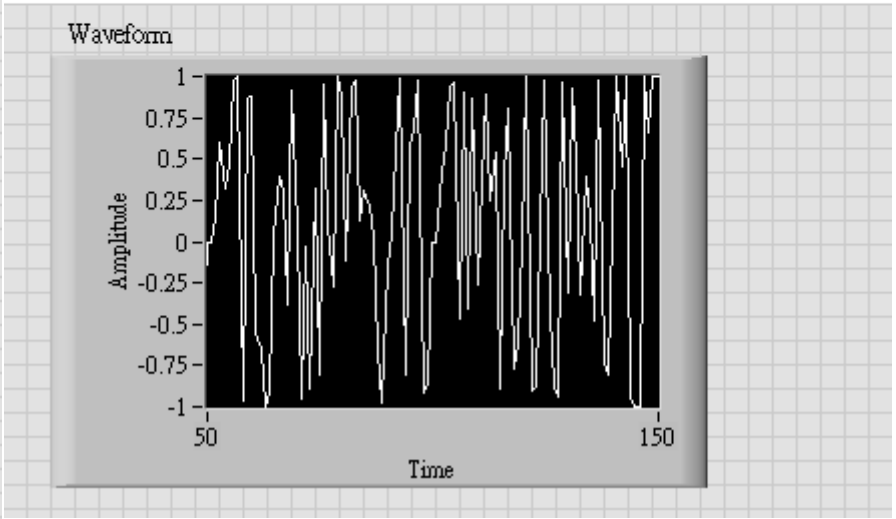
檔案輸入與輸出









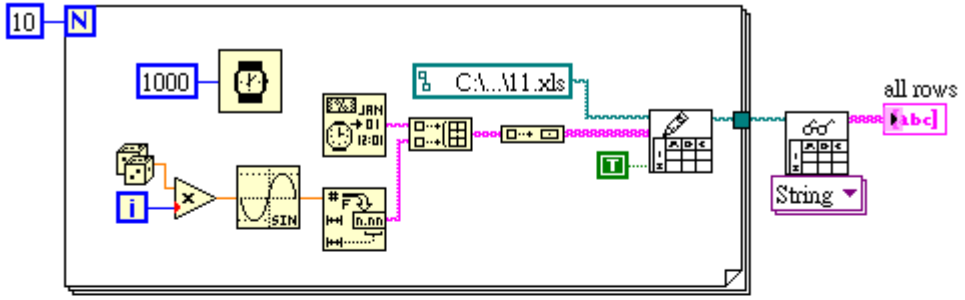


一次存檔（只進行一次開-寫-關）

迴圈內多次存檔（反覆進行開-寫-關）

作者；曾智勇,chihyungtseng@hotmail.com

檔案輸入與輸出



all rows

0	2014/8/25 上午 11:48:29	0.000000
0	2014/8/25 上午 11:48:30	0.606924
	2014/8/25 上午 11:48:31	0.103542
	2014/8/25 上午 11:48:32	0.743815
	2014/8/25 上午 11:48:33	0.399717
	2014/8/25 上午 11:48:34	0.460767
	2014/8/25 上午 11:48:35	-0.741281
	2014/8/25 上午 11:48:36	0.467628
	2014/8/25 上午 11:48:37	0.999575
	2014/8/25 上午 11:48:38	0.493749

Microsoft Excel - 11

檔案(F) 編輯(E) 檢視(V) 插入(I) 格式(O) 工具(T)

安全性...

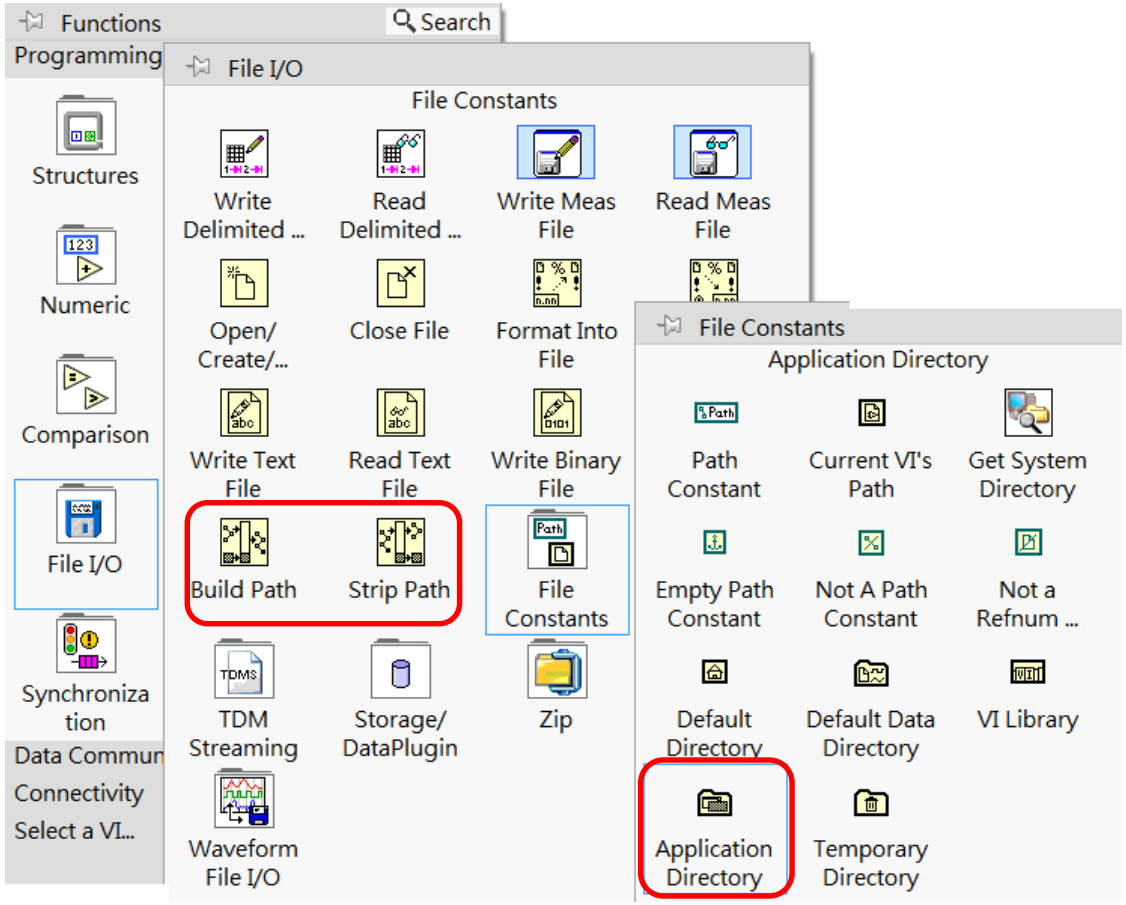
D7

	A	B	C
1	2014/8/25 上午 11:48:29	0	
2	2014/8/25 上午 11:48:30	0.606924	
3	2014/8/25 上午 11:48:31	0.103542	
4	2014/8/25 上午 11:48:32	0.743815	
5	2014/8/25 上午 11:48:33	0.399717	
6	2014/8/25 上午 11:48:34	0.460767	
7	2014/8/25 上午 11:48:35	-0.74128	
8	2014/8/25 上午 11:48:36	0.467628	
9	2014/8/25 上午 11:48:37	0.999575	
10	2014/8/25 上午 11:48:38	0.493749	
11			

高階、簡易型、快速完成，存/取函式。
（只能單一資料格式）

作者；曾智勇,chihyungtseng@hotmail.com

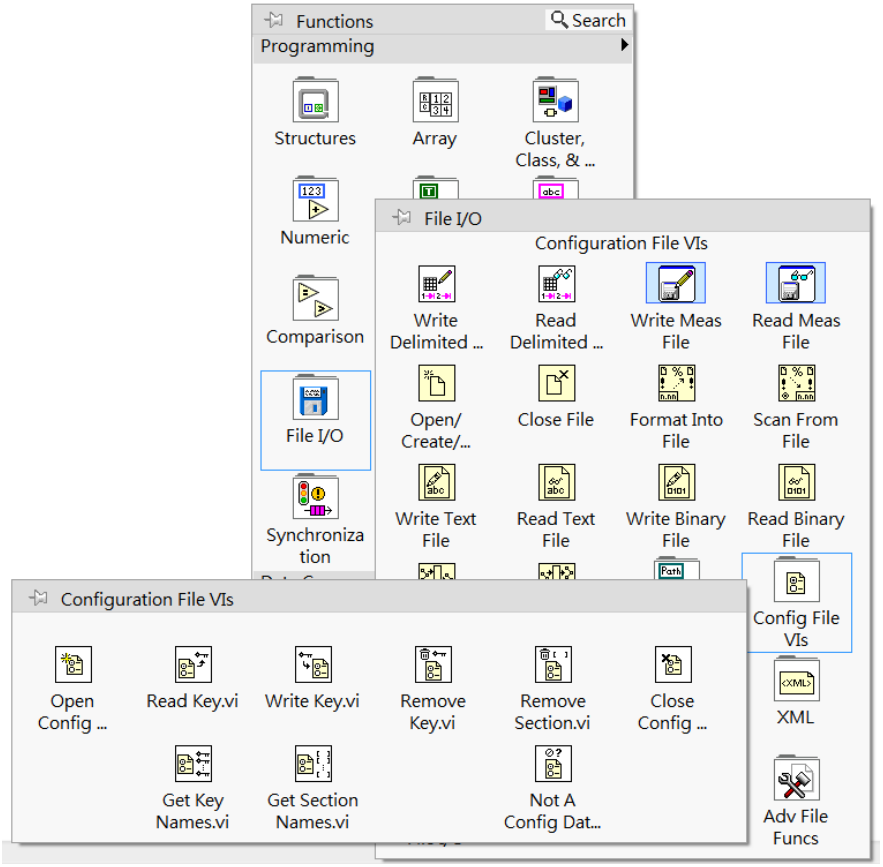
檔案輸入與輸出



路徑根目錄請使用" Application Directory"

作者；曾智勇, chihyungtseng@hotmail.com

檔案輸入與輸出

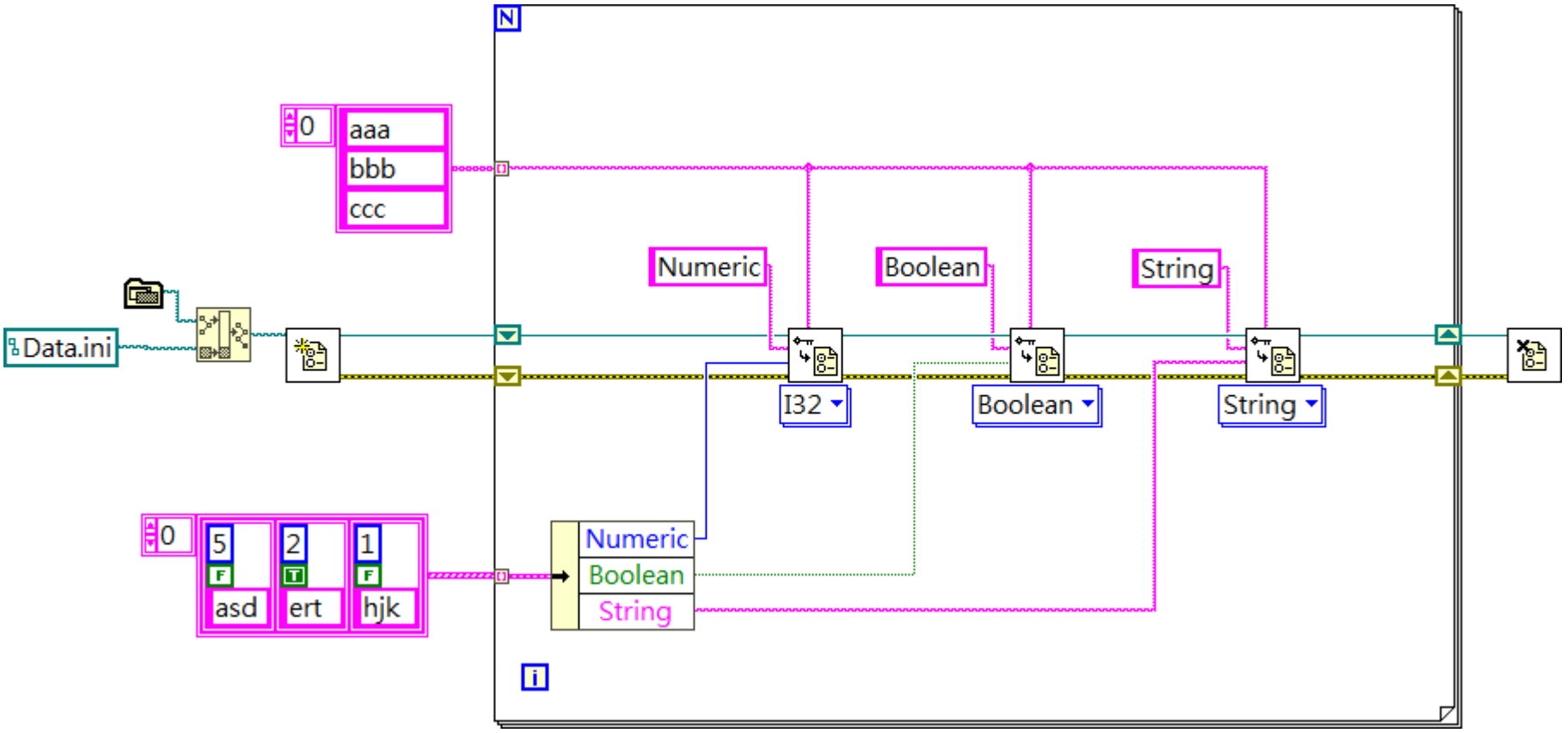


.ini檔存取用函式庫

Read	Write
Open Config Data.vi	Open Config Data.vi
Get Section Names.vi	Write Key.vi
Read Key.vi	
Close Config Data.vi	Close Config Data.vi

作者；曾智勇,chihyungtseng@hotmail.com

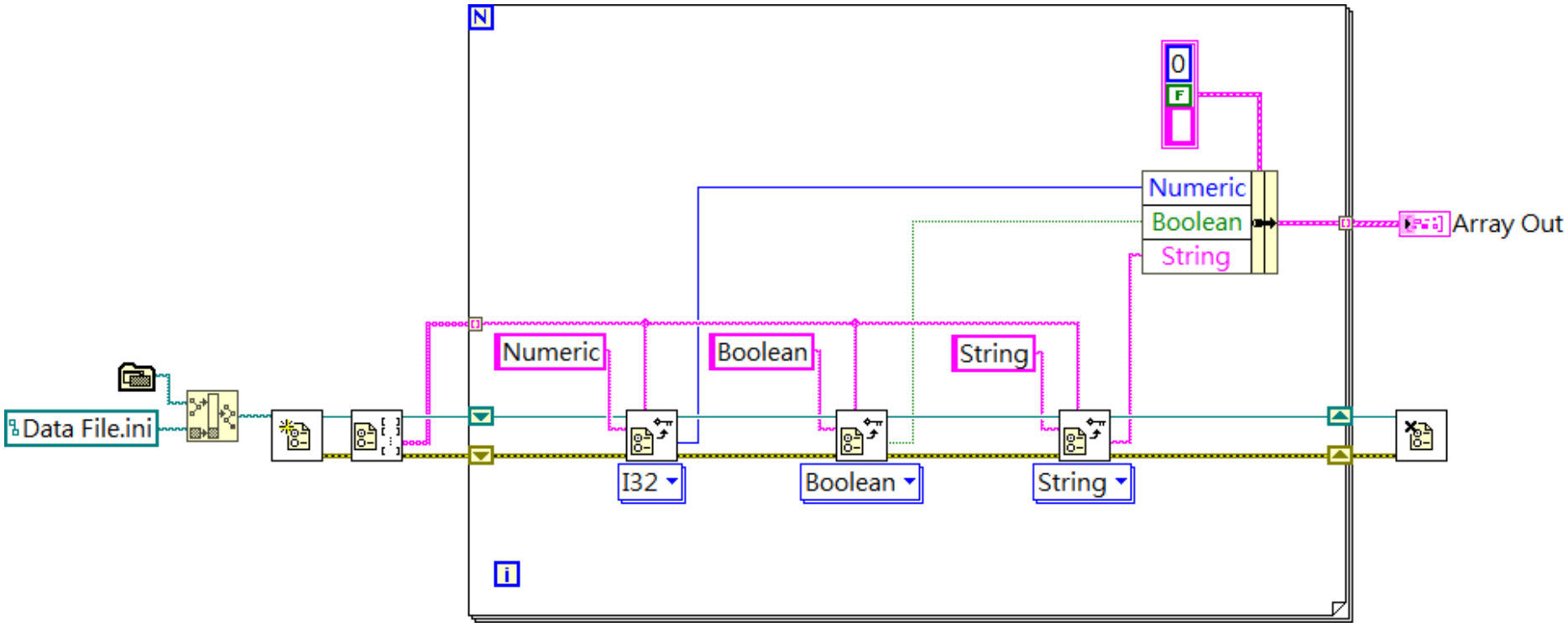
檔案輸入與輸出



.ini Write

作者；曾智勇, chihyungtseng@hotmail.com

檔案輸入與輸出

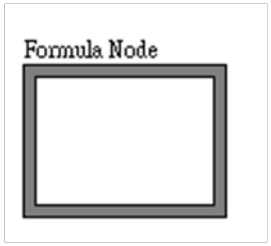


.ini Read

作者；曾智勇, chihyungtseng@hotmail.com

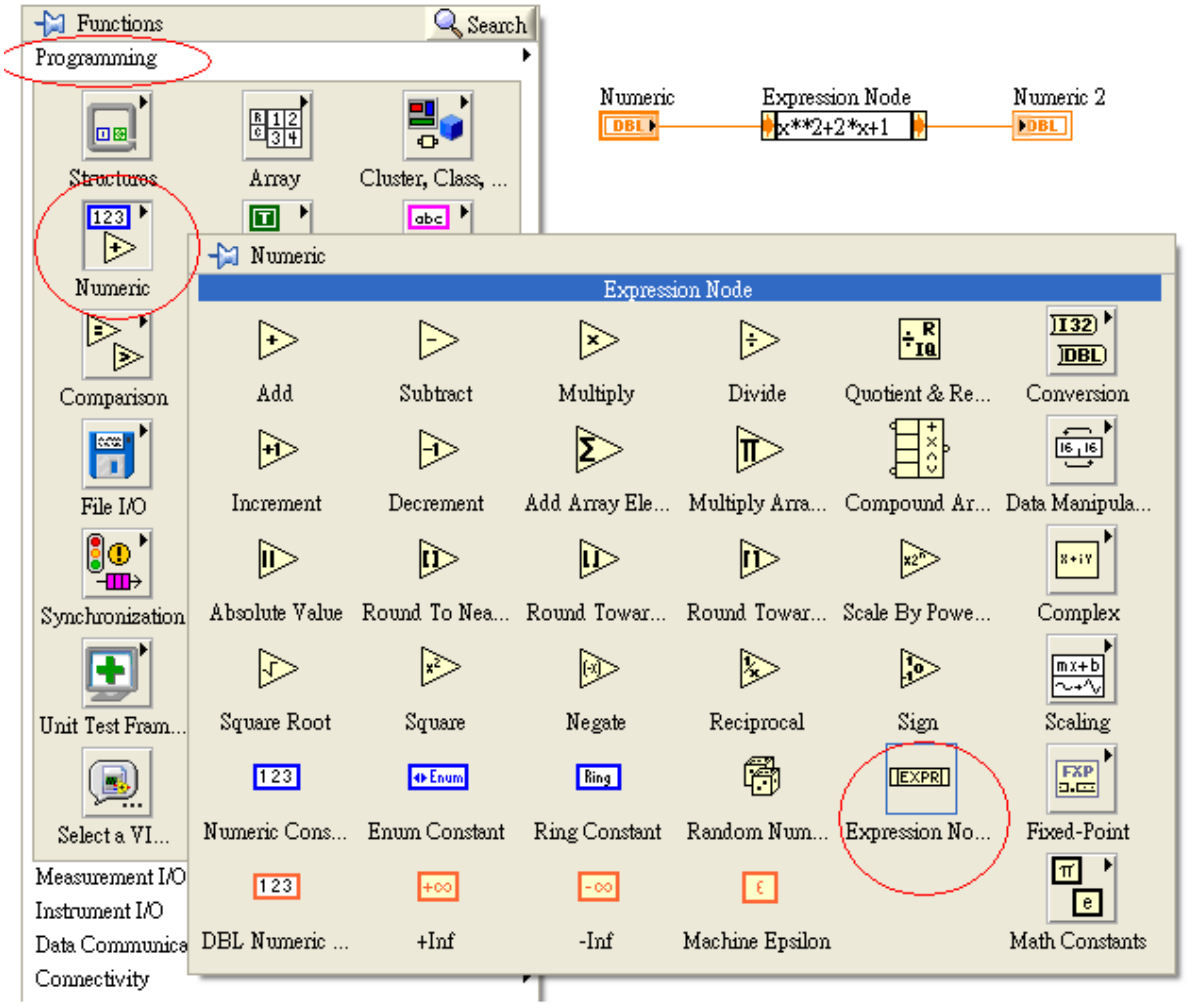
公式節點

- 節省空間、重複使用、語法不熟

一輸入+一輸出	
多輸入+一輸出	
多輸入+多輸出	

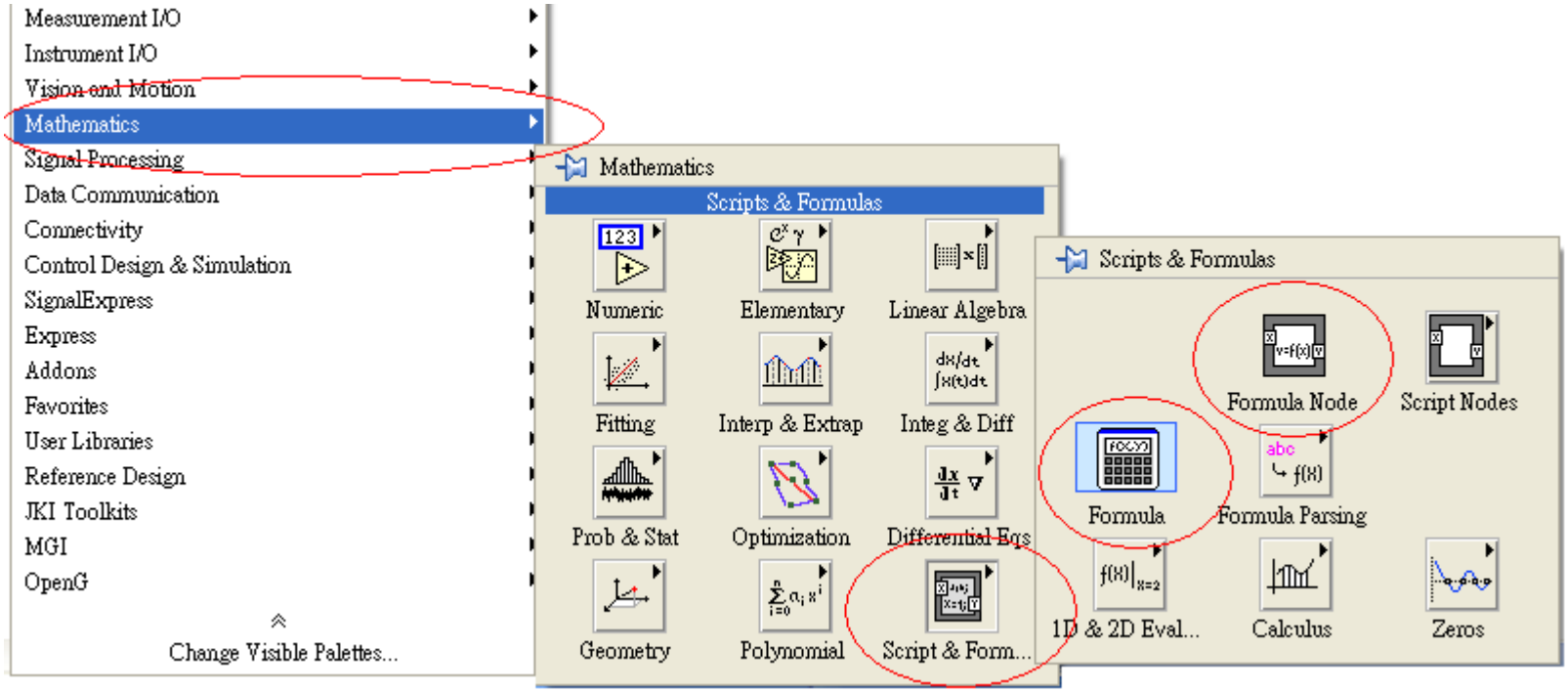
作者；曾智勇,chihyungtseng@hotmail.com

公式節點



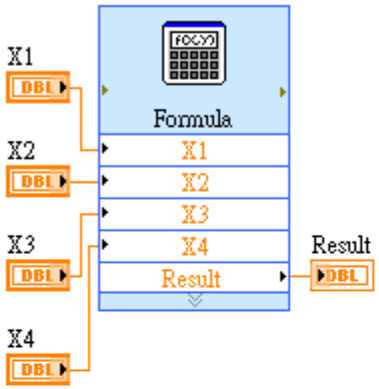
作者；曾智勇, chihyungtseng@hotmail.com

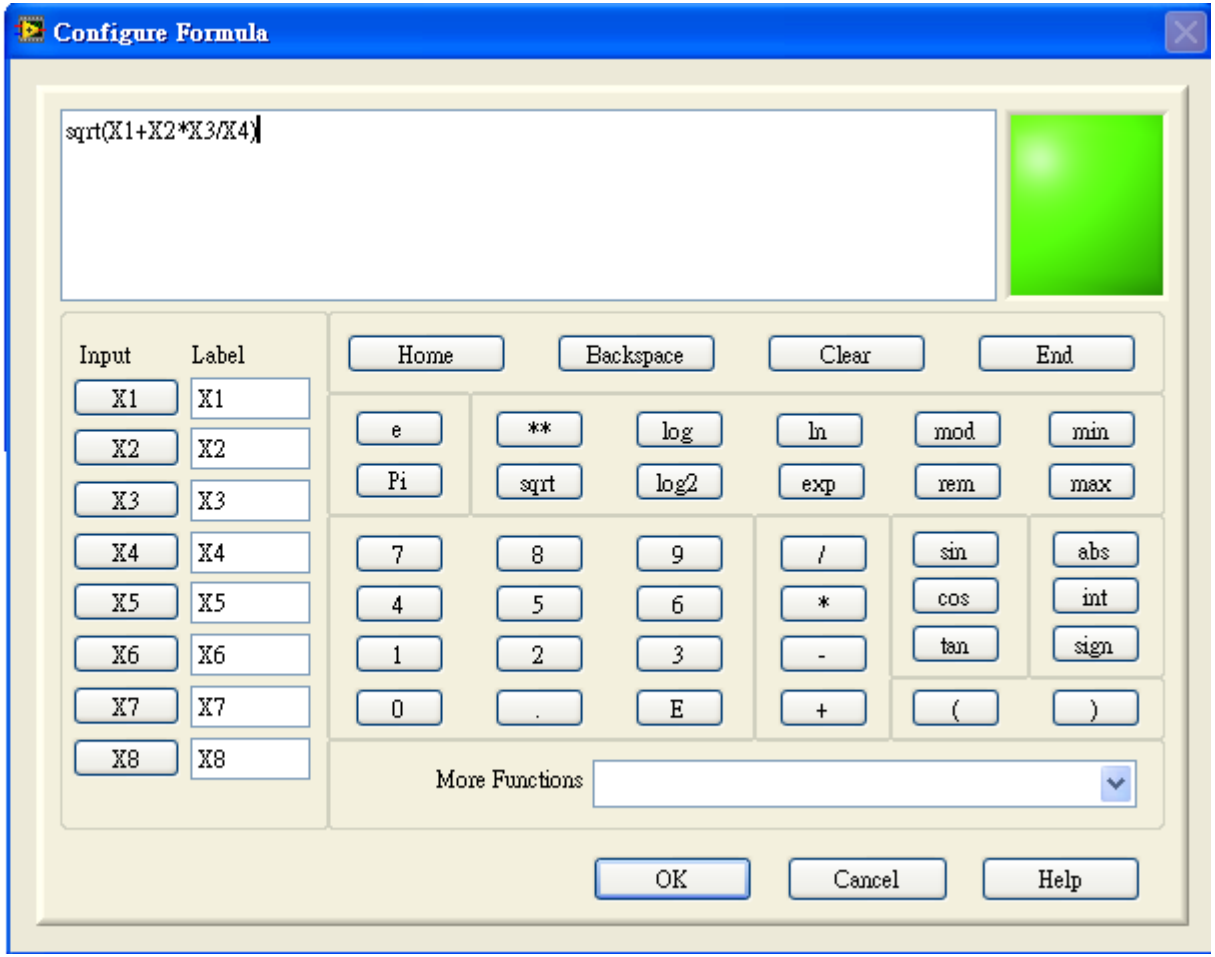
公式節點



作者；曾智勇,chihyungtseng@hotmail.com

公式節點

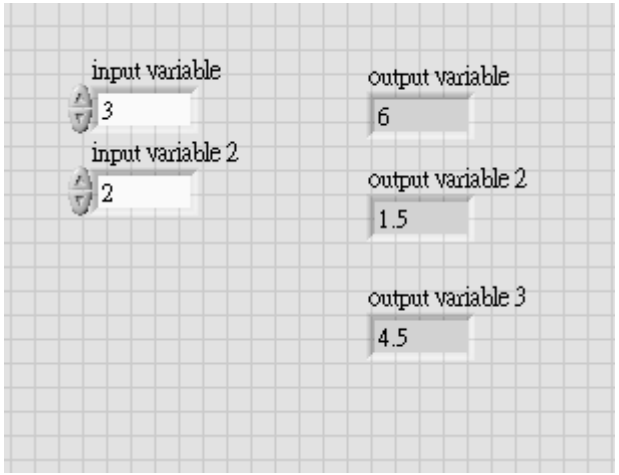


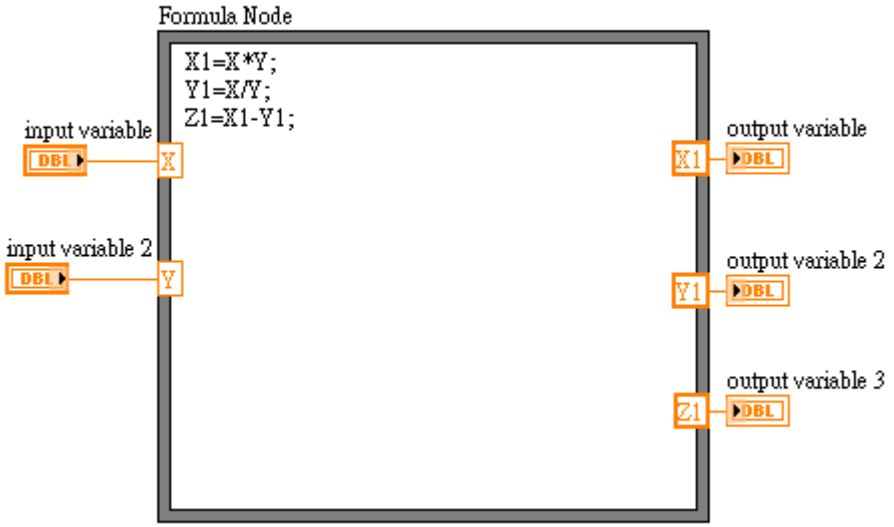


作者；曾智勇,chihyungtseng@hotmail.com

公式節點

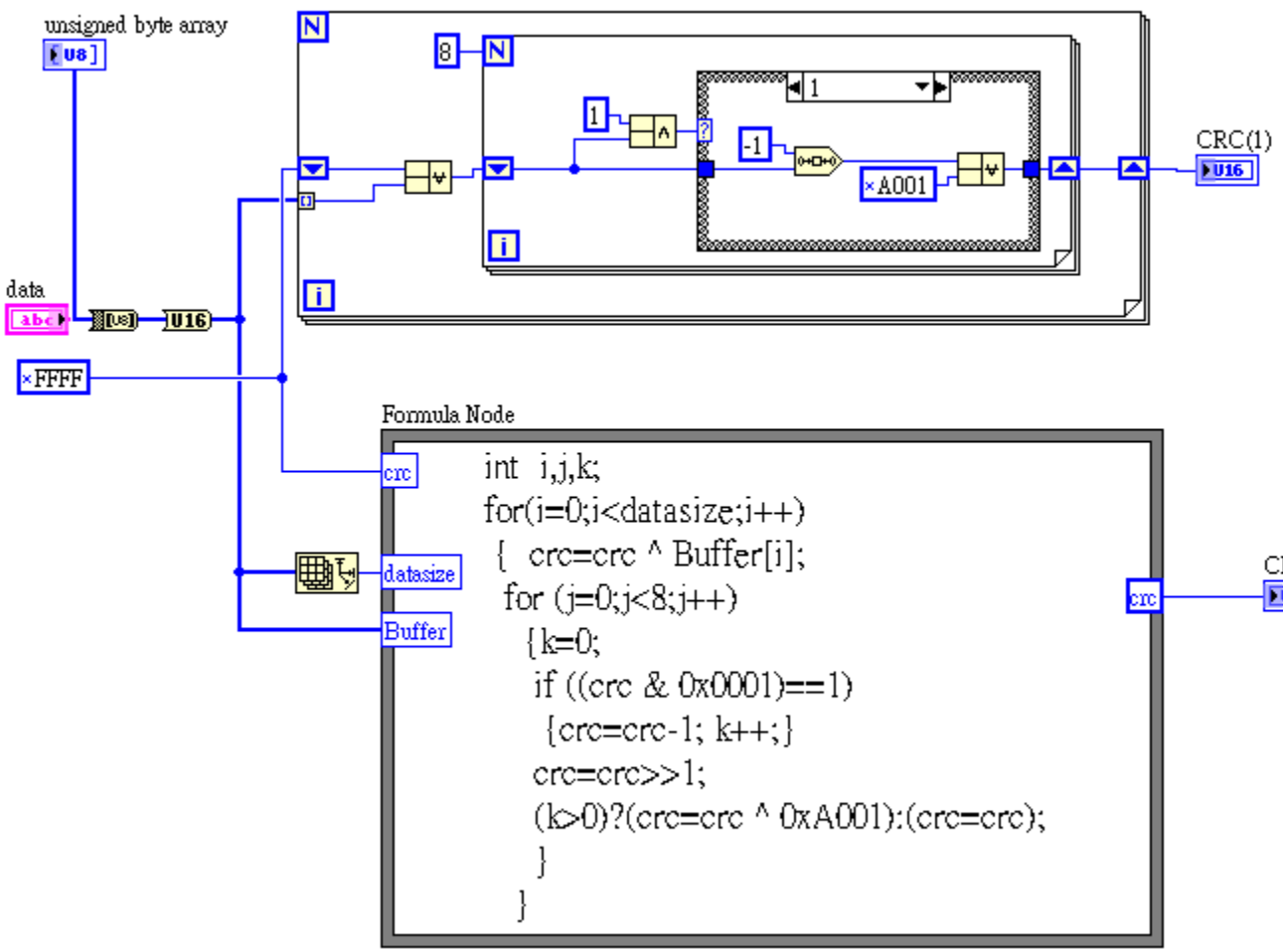
- Formula Node：
  - 多輸入+多輸出
  - 可以是矩陣，不一定要純量
  - 支援多種C語法、數學函數、布林運算

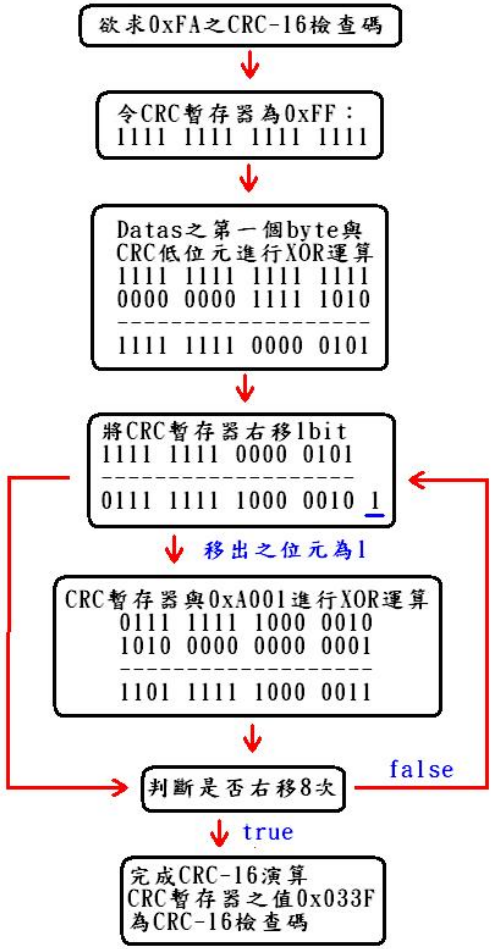




作者；曾智勇,chihyungtseng@hotmail.com

公式節點

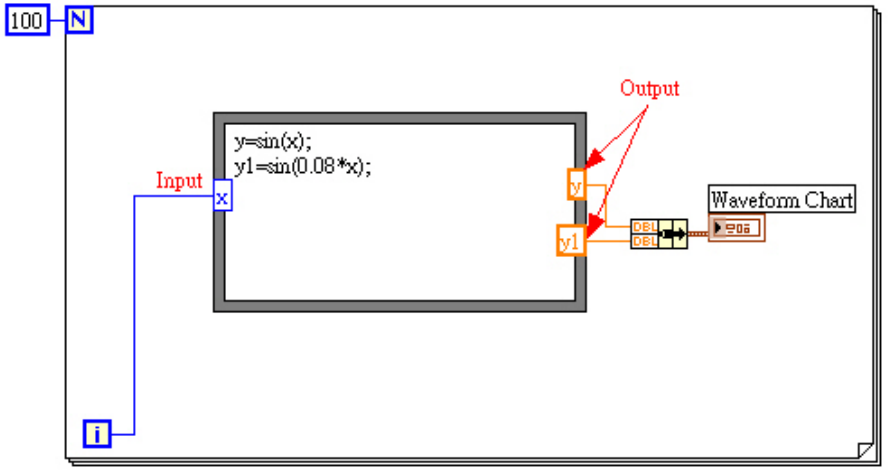


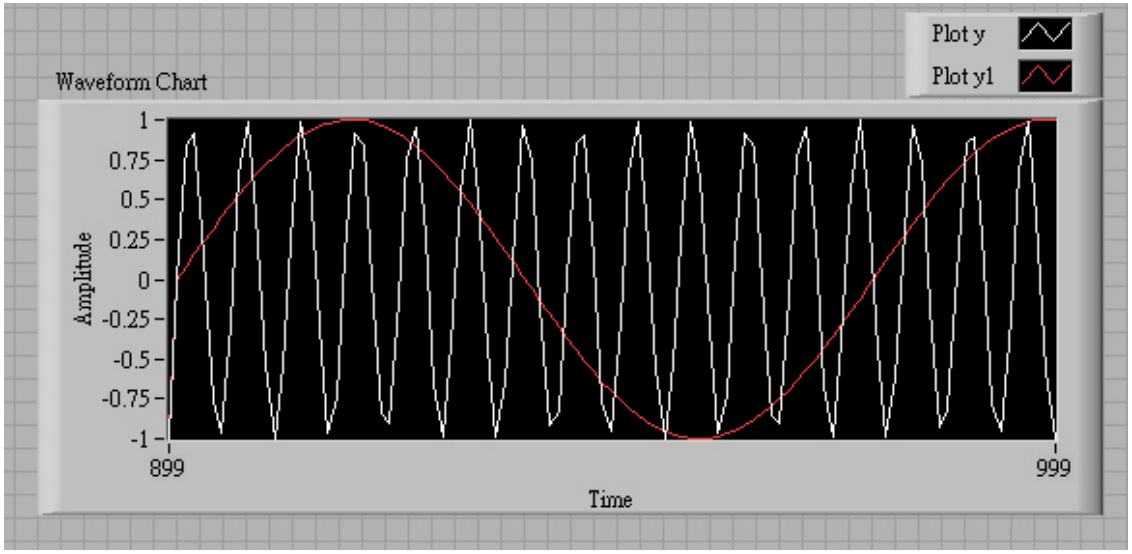


作者；曾智勇,chihyungtseng@hotmail.com

公式節點

- Formula Node：

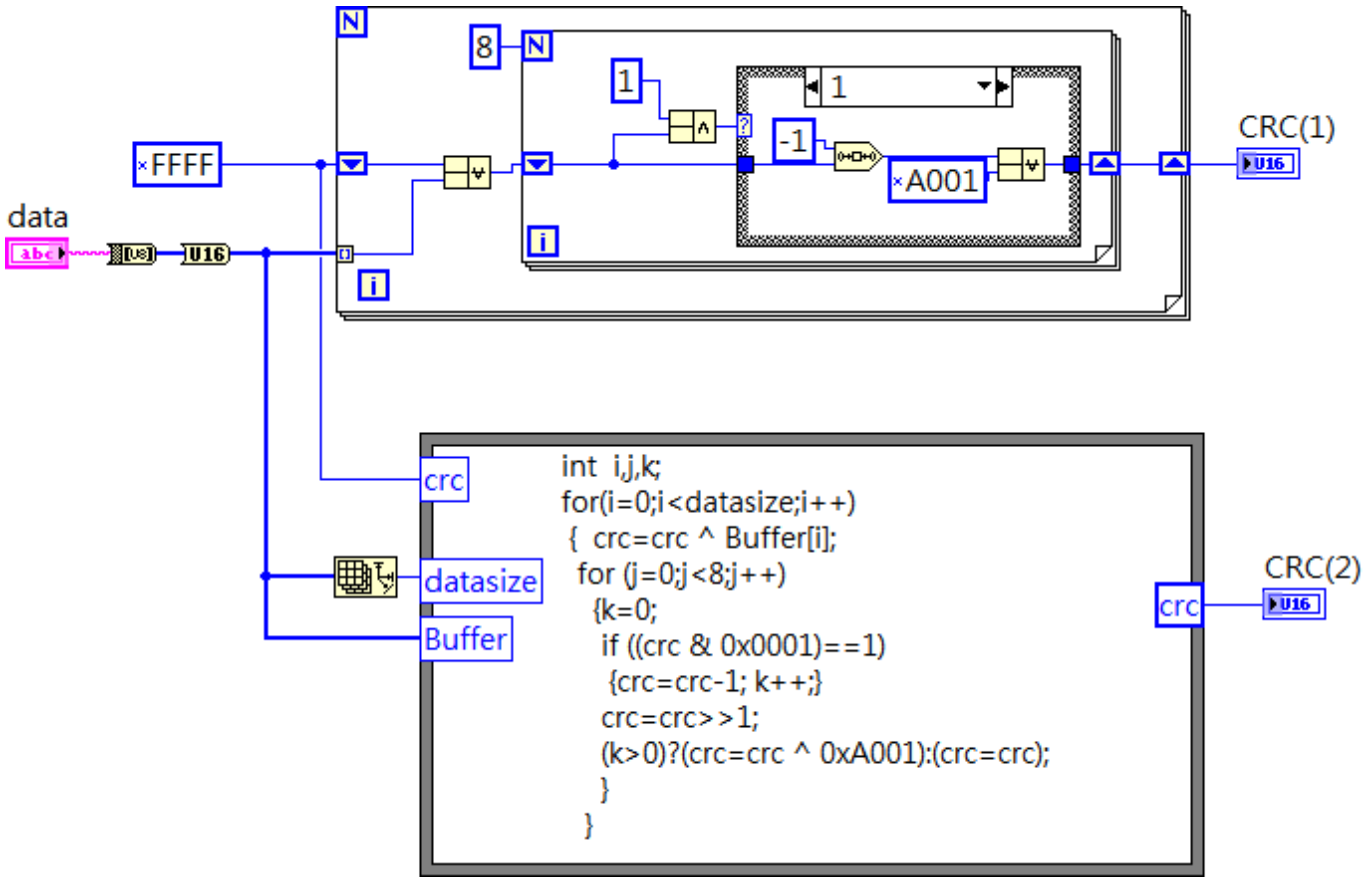




作者；曾智勇,chihyungtseng@hotmail.com

公式節點

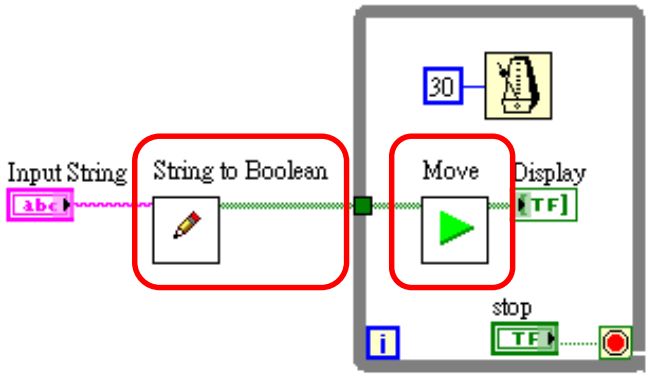
- Formula Node：

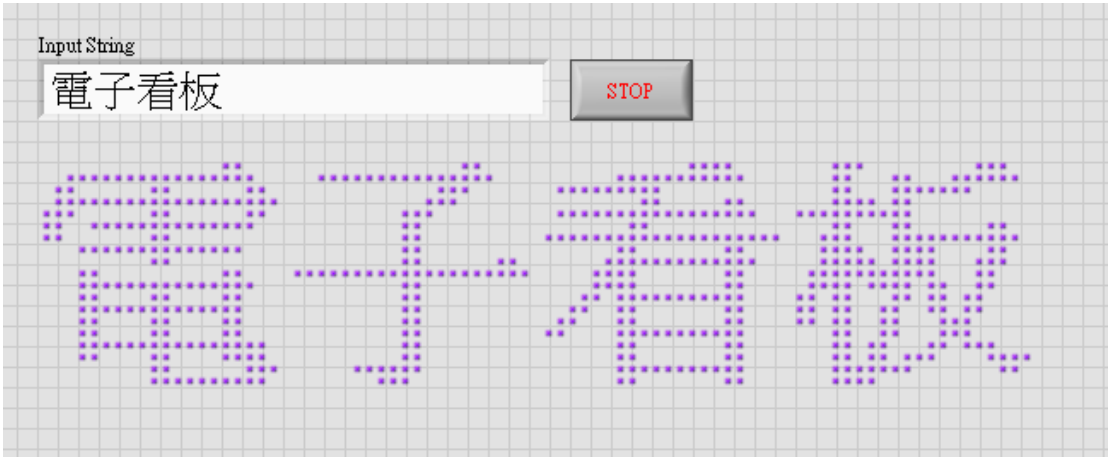


作者；曾智勇, chihyungtseng@hotmail.com

Sub VI

- Sub VI的功用：節省空間、重複使用、協力開發

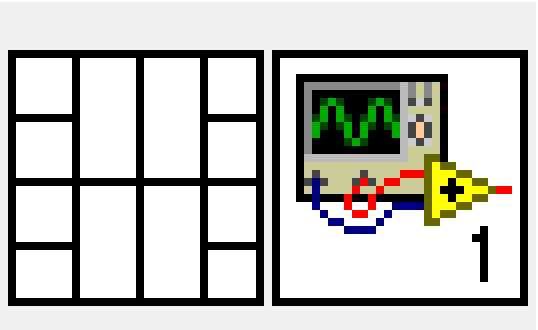


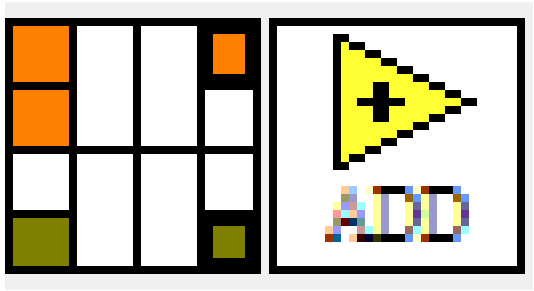


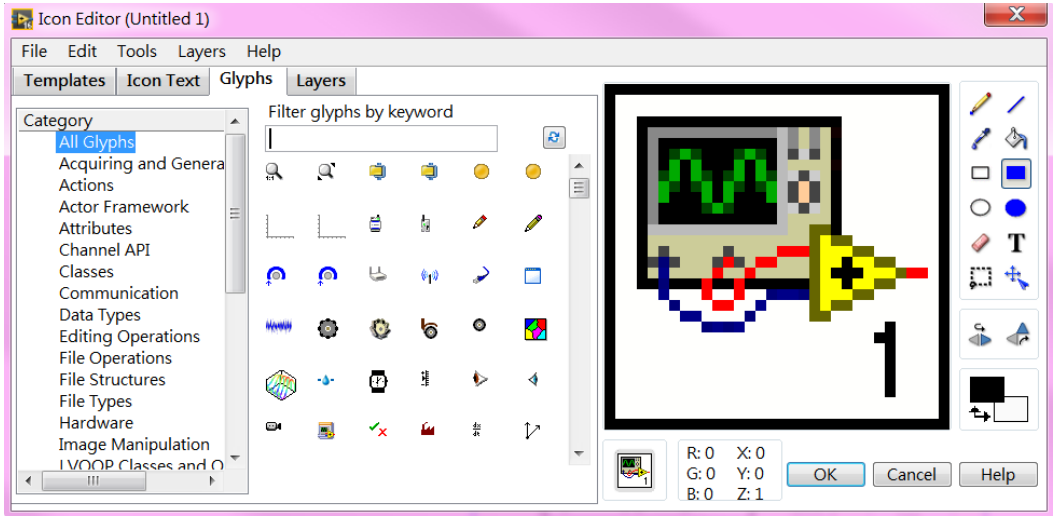
作者；曾智勇,chihyungtseng@hotmail.com

Sub VI

- 雙擊右上角的icon圖示，可進行編輯圖示或I/O接腳



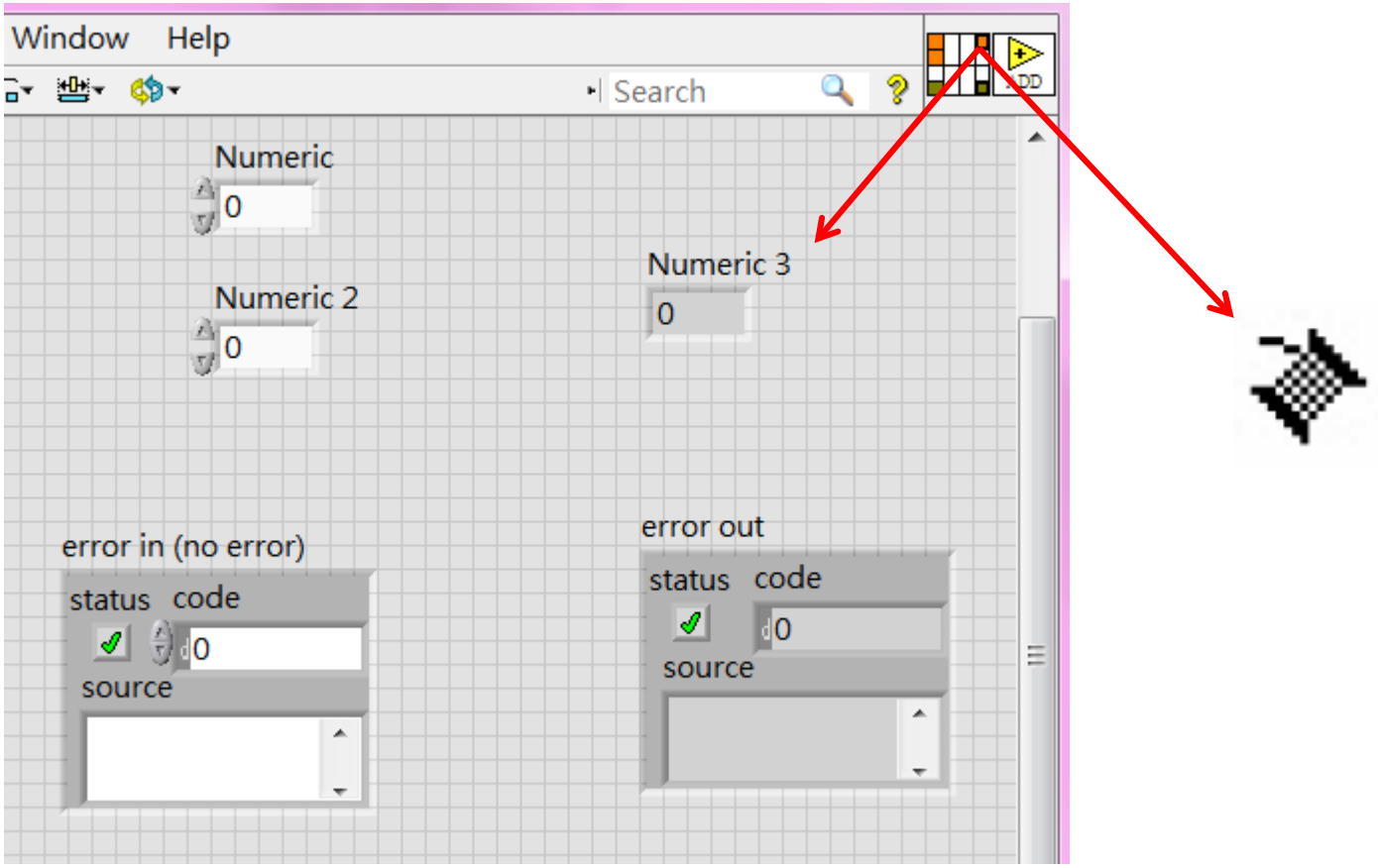




作者；曾智勇,chihyungtseng@hotmail.com

Sub VI

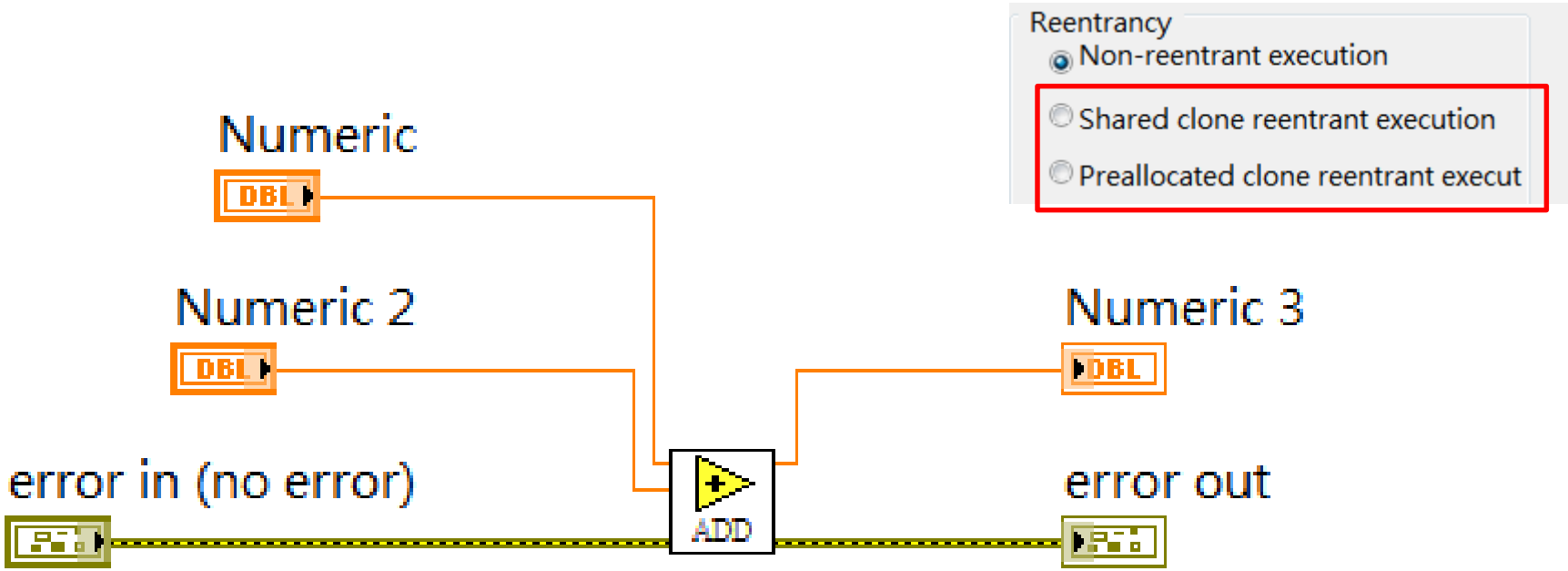
- 雙擊右上角的icon圖示，可進行編輯圖示或I/O接腳



作者；曾智勇,chihyungtseng@hotmail.com

Sub VI

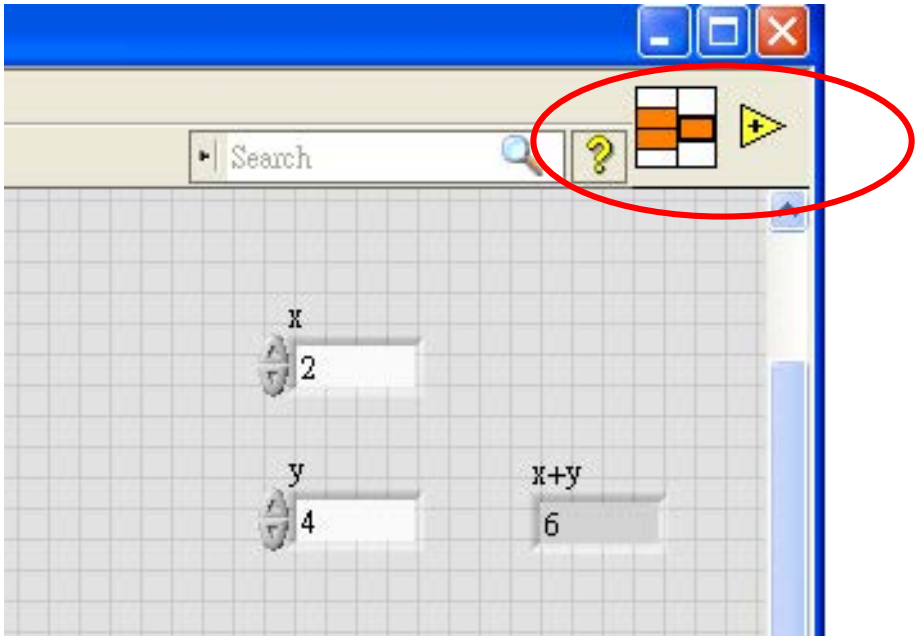
- 將被呼叫程式拖曳到呼叫程式之程式區，即可使用
- 單一個程式可重複被呼叫、可被多個程式呼叫
- 若要同時呼叫，需於VI屬性處設定為reentrant execution

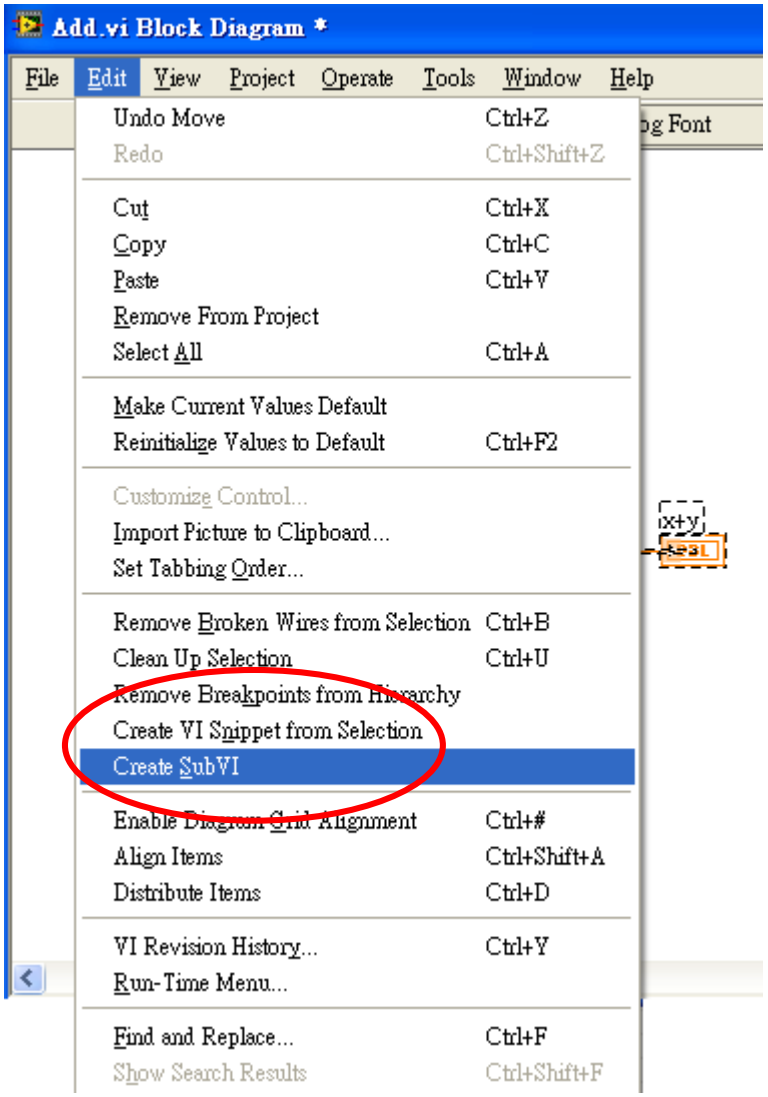


作者；曾智勇,chihyungtseng@hotmail.com

Sub VI

- 每一個VI都可以是Sub VI
- 可框選輸出成Sub VI



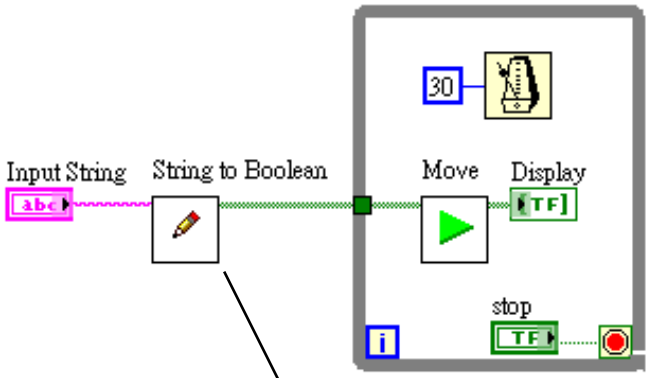


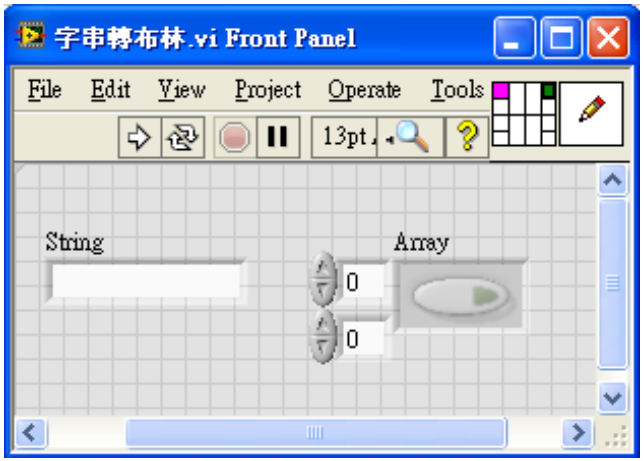
•定義輸入、輸出接腳（傳入/傳回值）

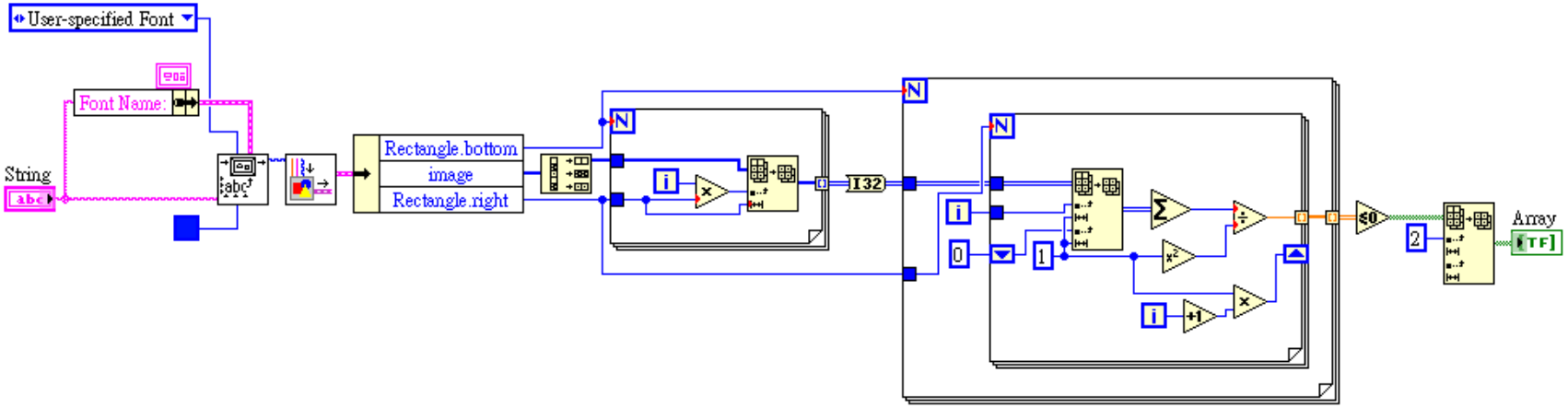
•繪製icon圖示，用以識別用

作者；曾智勇, chihyungtseng@hotmail.com

Sub VI

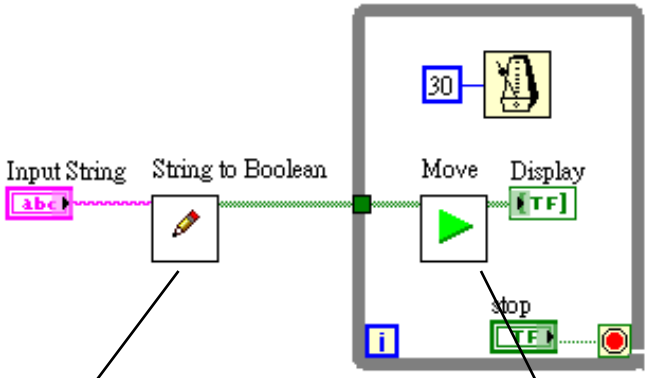


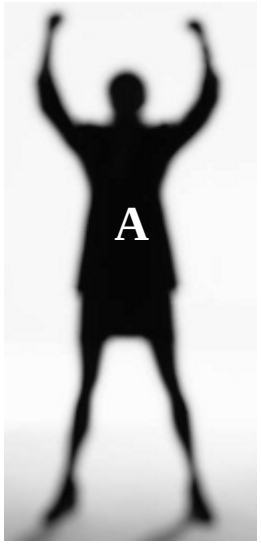


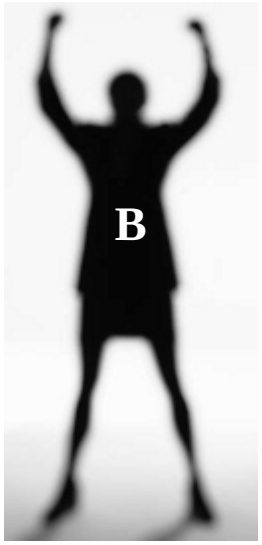


作者；曾智勇,chihyungtseng@hotmail.com

Sub VI







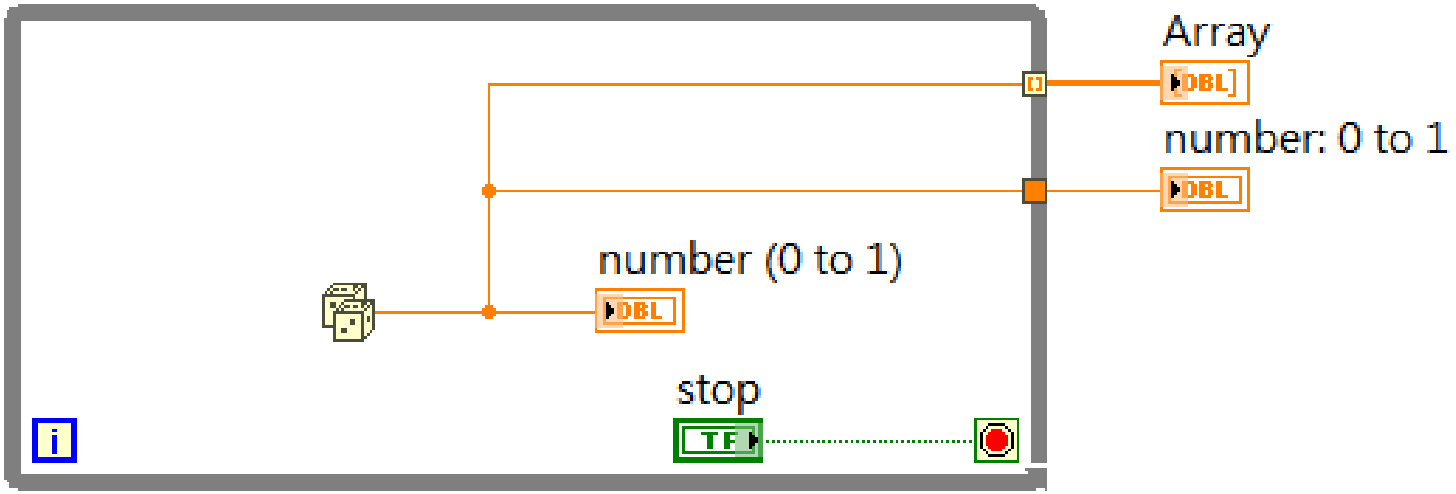
透過事先約定輸入、輸出資料格式，

進行工作分工（分別開發Sub VI）

作者；曾智勇, chihyungtseng@hotmail.com

Structure

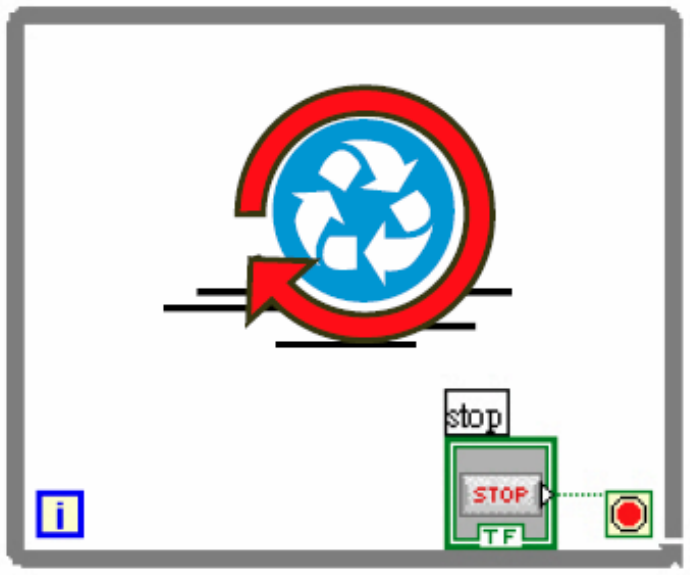
- While Loop： i 代表迴圈執行計數器， 代表迴圈停止條件接收器
- i 從0開始往上計數
- 資料經過While Loop邊界時，Indexing預設為OFF，開啟後可輸出陣列



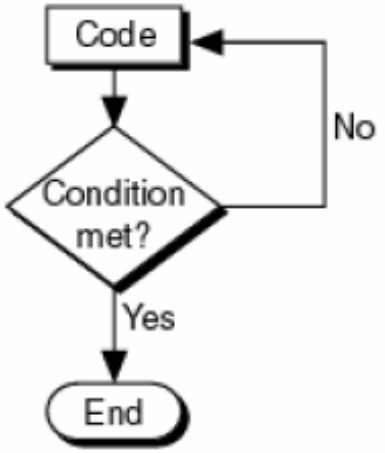
作者；曾智勇,chihyungtseng@hotmail.com

Structure

- While Loop：



LabVIEW While Loop



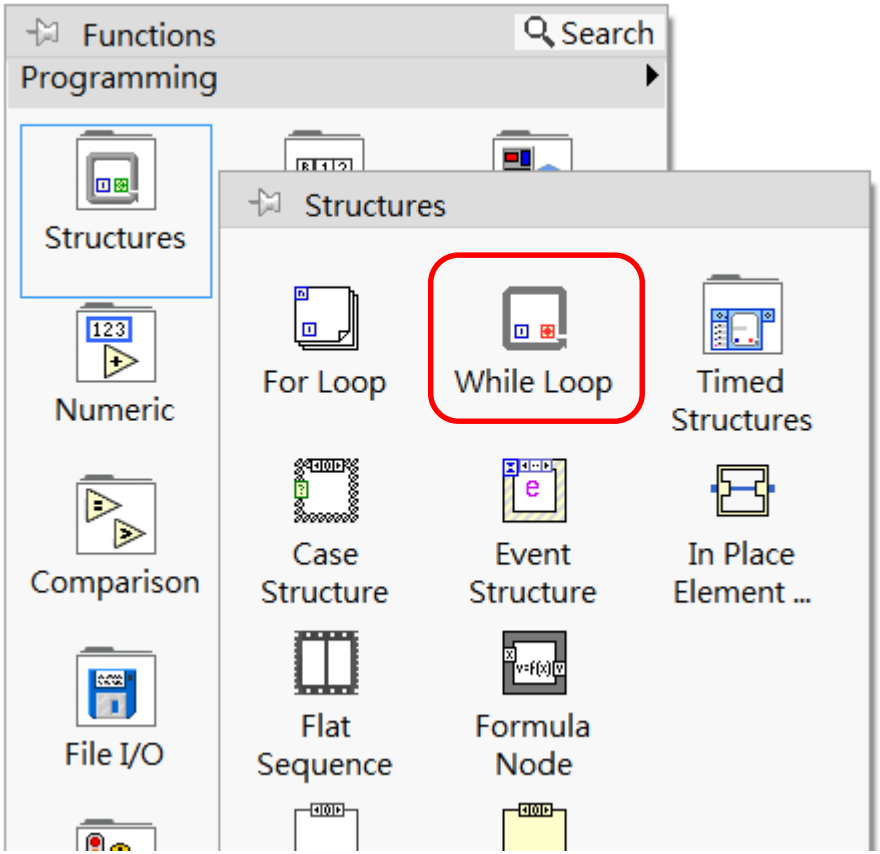
Flow Chart

While Loop 至少會執行一次

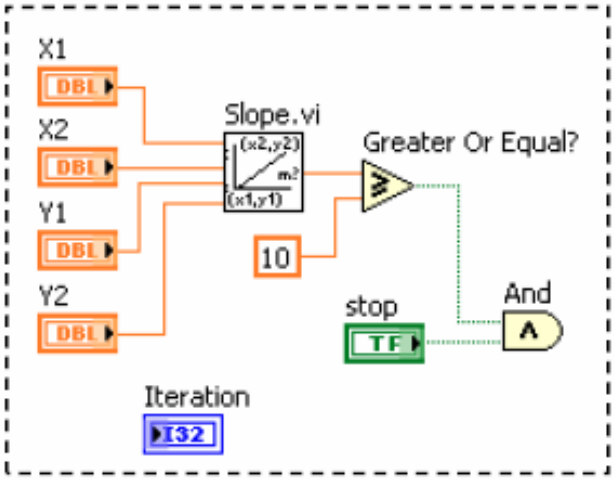
作者；曾智勇, chihyungtseng@hotmail.com

Structure

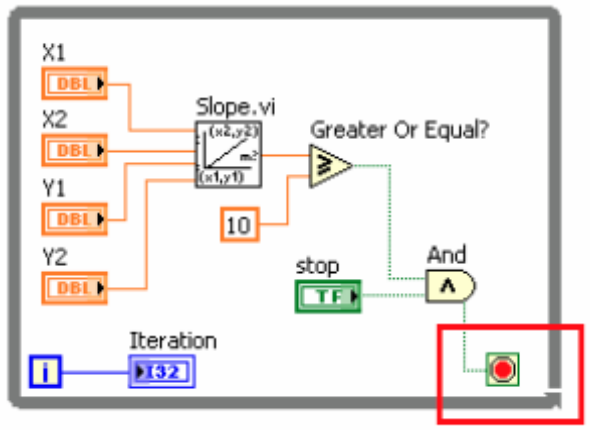
- While Loop：



框選重複執行區域



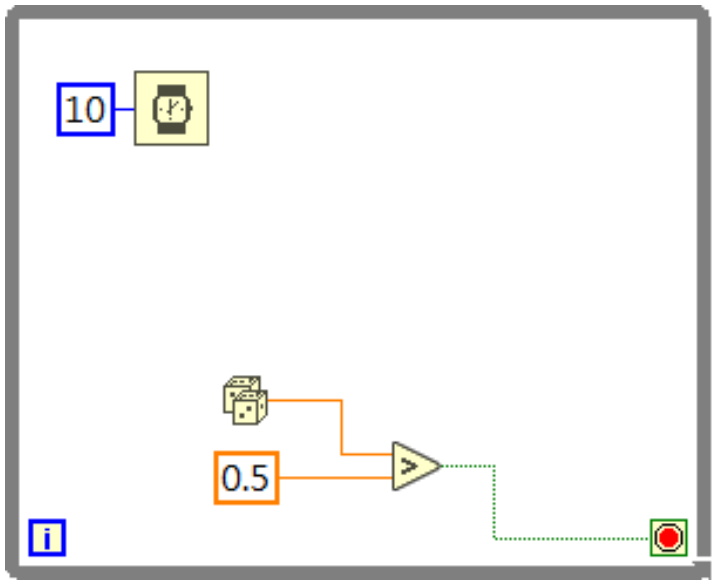
設定停止條件



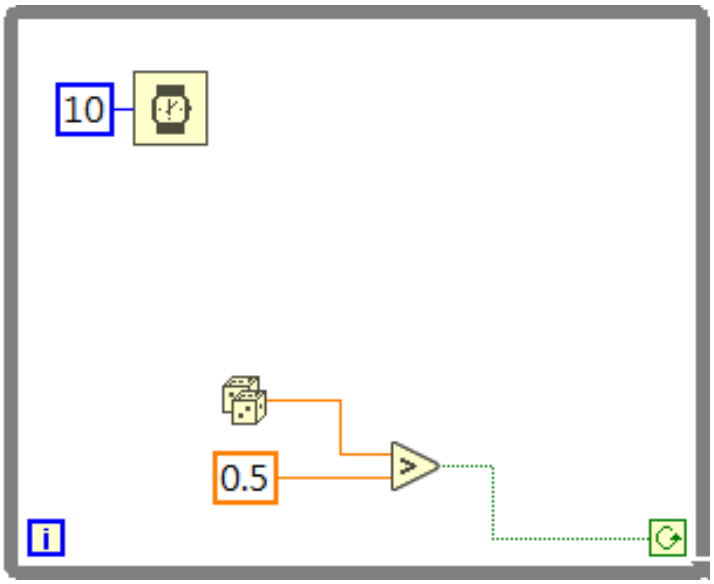
作者：曾智勇, chihyungtseng@hotmail.com

Structure

- While Loop：



大於0.5，滿足條件就**停止**



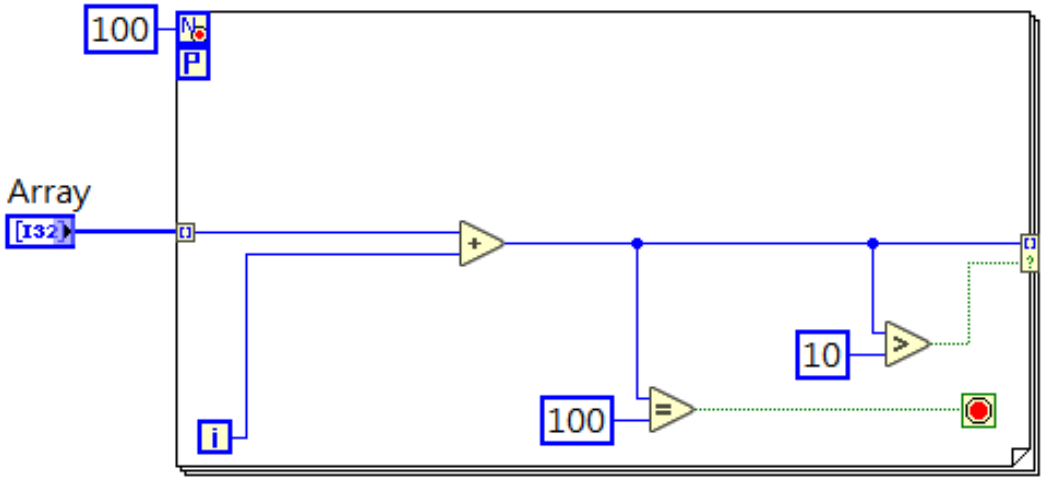
大於0.5，滿足條件就**繼續執行**

停止條件需明確，以避免形成無窮迴圈

作者；曾智勇,chihyungtseng@hotmail.com

Structure

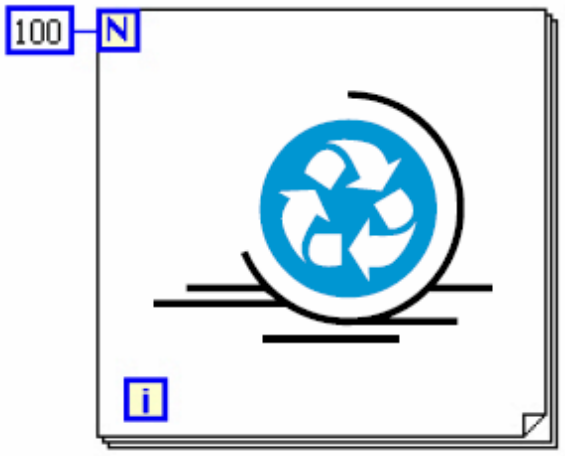
- For Loop： i 代表迴圈執行計數器， 代表迴圈停止條件接收器
- i 從0開始往上計數
- 預設為關閉，迴圈會跑到指定次數或由輸入資料Indexing決定
- P 預設為關閉，開啟後可跑多執行緒，迴圈資料須無關聯性
- 資料經過For Loop邊界時，Indexing預設為ON
- Indexing可開起Conditional功能(選擇性輸出)



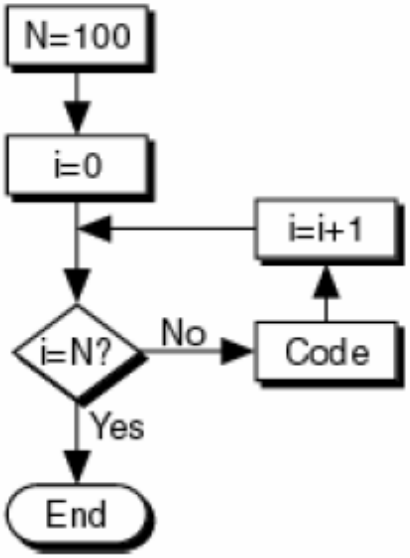
作者；曾智勇,chihyungtseng@hotmail.com

Structure

- For Loop：



For Loop



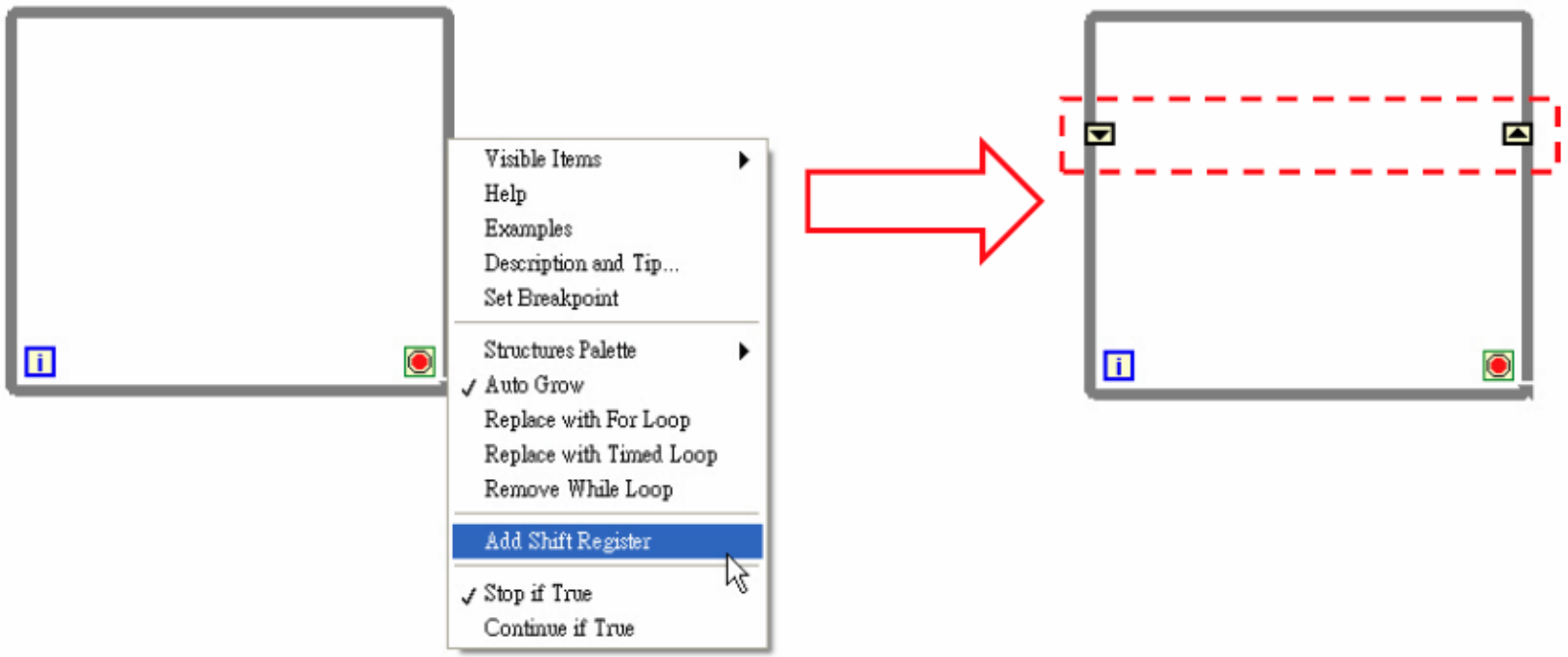
流程圖

For Loop有可能執行0次

作者；曾智勇, chihyungtseng@hotmail.com

Structure

- 移位暫存器(Shift register)：
  儲存前數次迴圈執行結果、程式執行過程資料保存

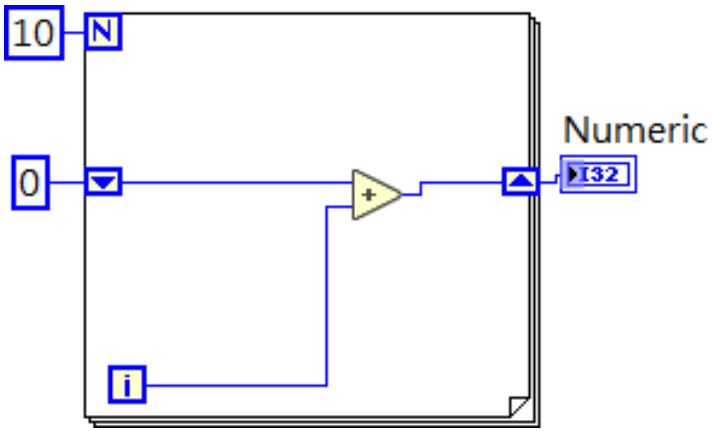


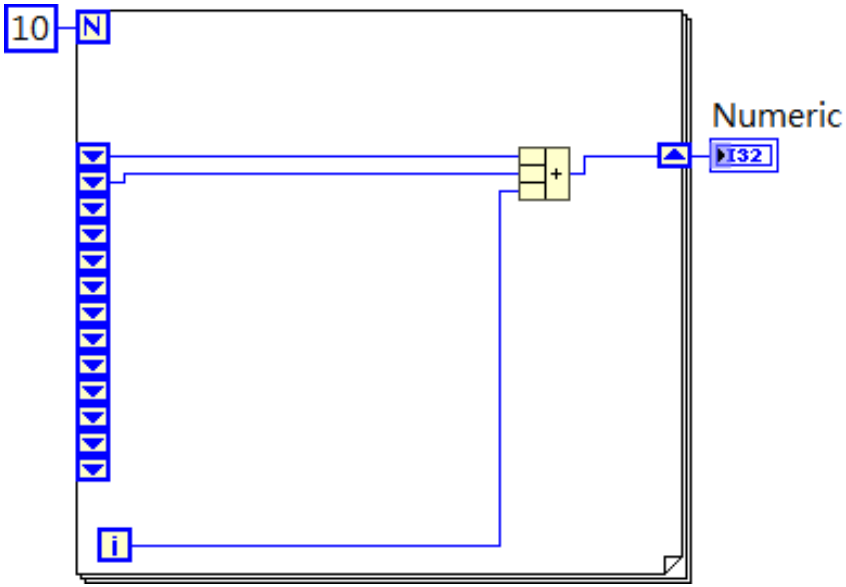
For Loop/ While Loop皆可使用

作者；曾智勇, chihyungtseng@hotmail.com

Structure

- 移位暫存器(Shift register)：
  儲存前數次迴圈執行結果、程式執行過程資料保存



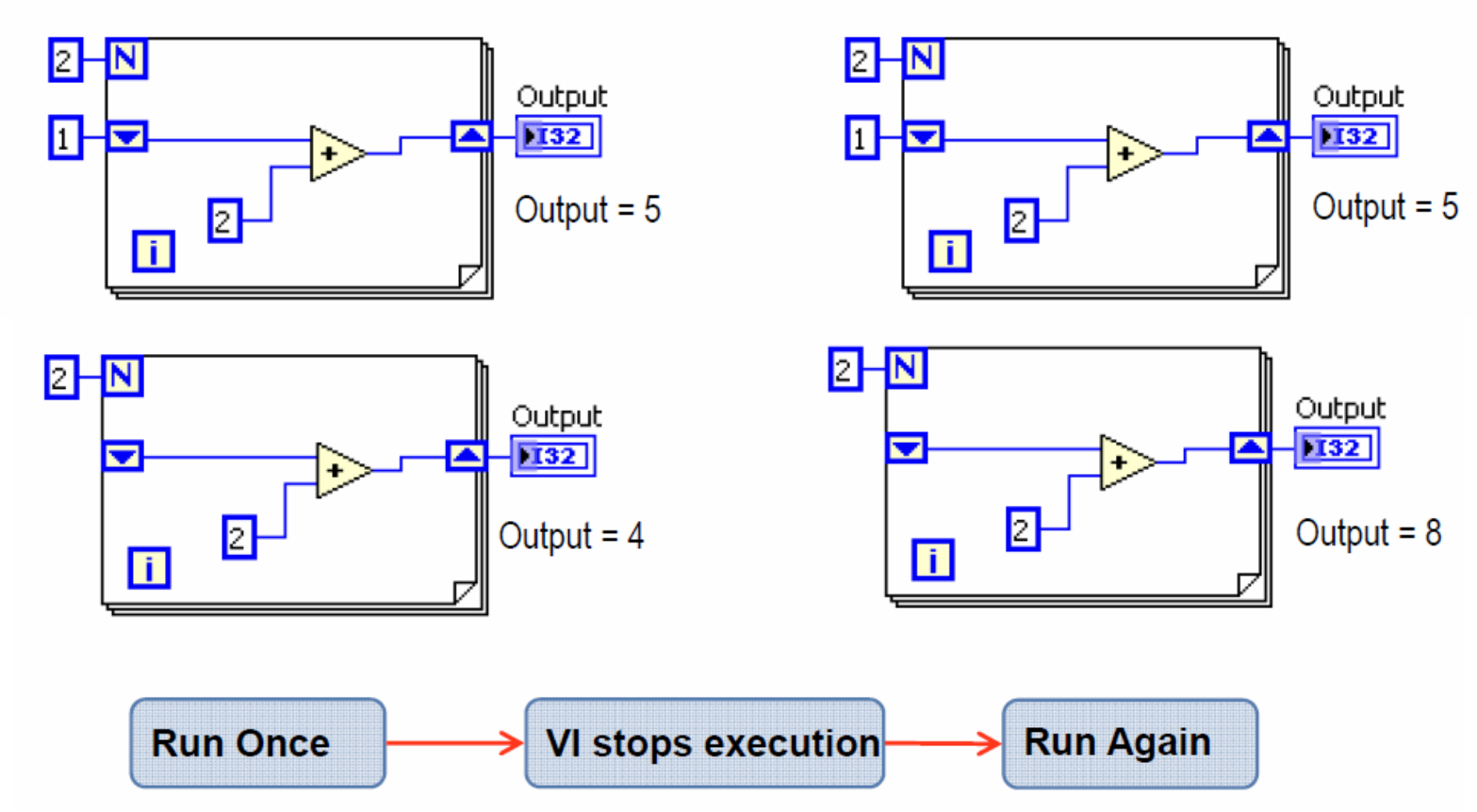


可保存多次迴圈前的運算值

作者；曾智勇,chihyungtseng@hotmail.com

Structure

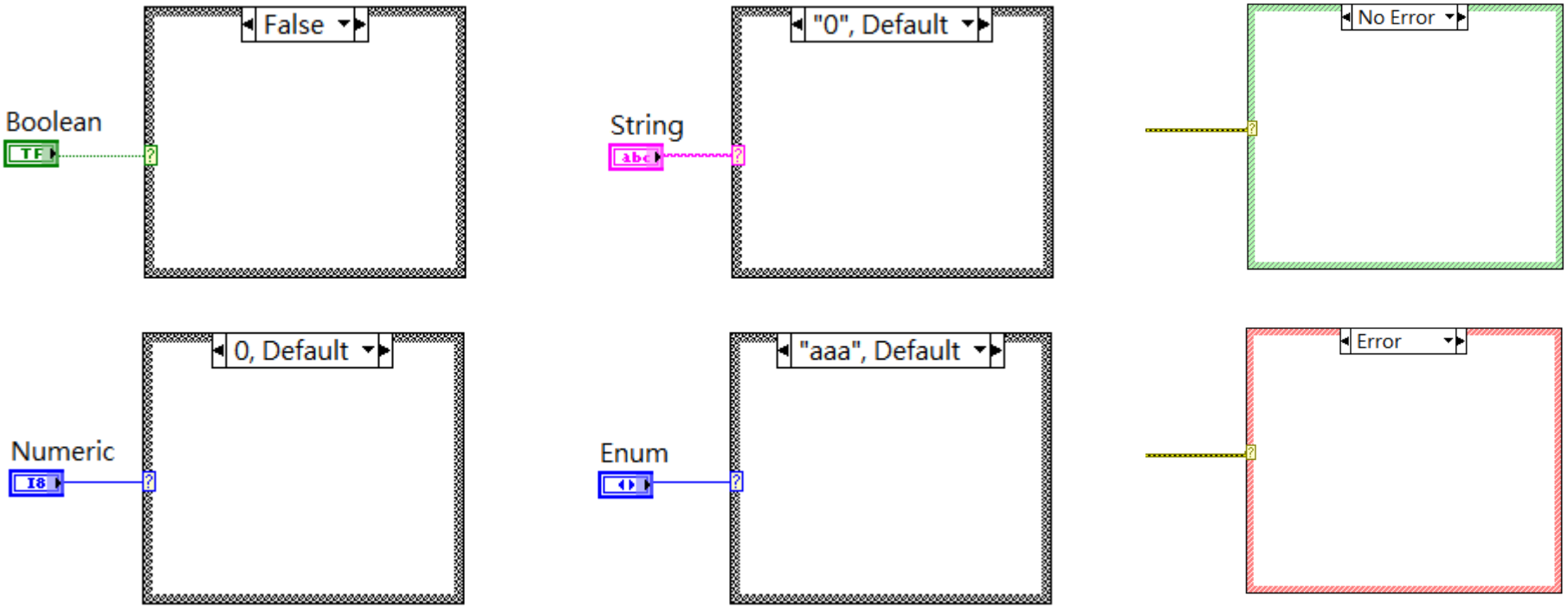
- 移位暫存器(Shift register)：
  儲存前數次迴圈執行結果、程式執行過程資料保存



作者；曾智勇, chihyungtseng@hotmail.com

Structure

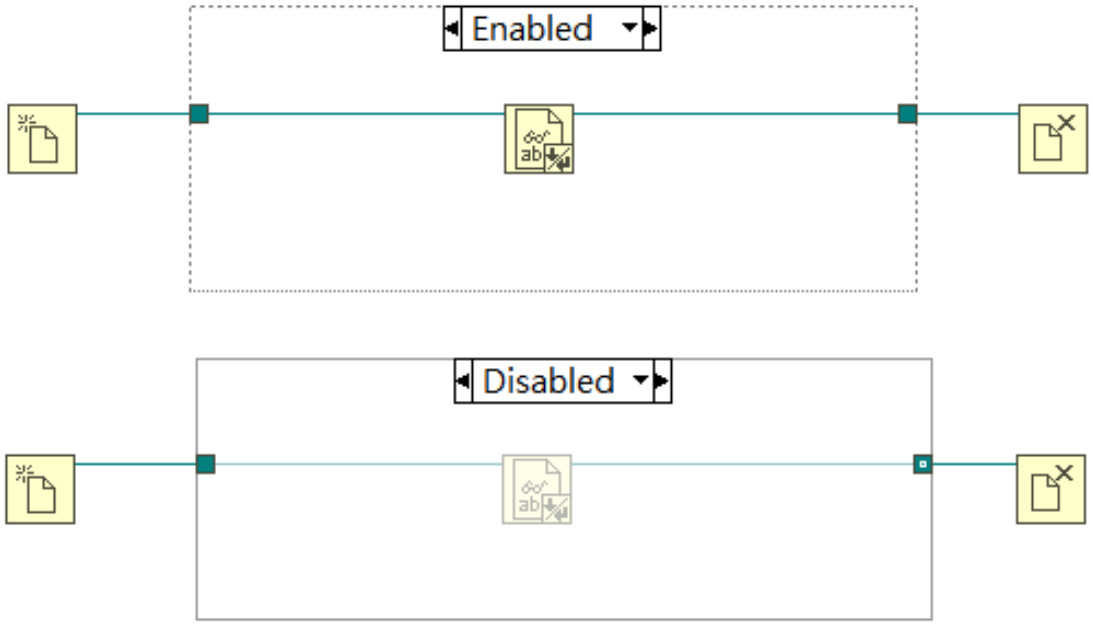
- Case：輸入條件可為布林、文字、數字、下拉選單、錯誤叢集



作者；曾智勇,chihyungtseng@hotmail.com

Structure

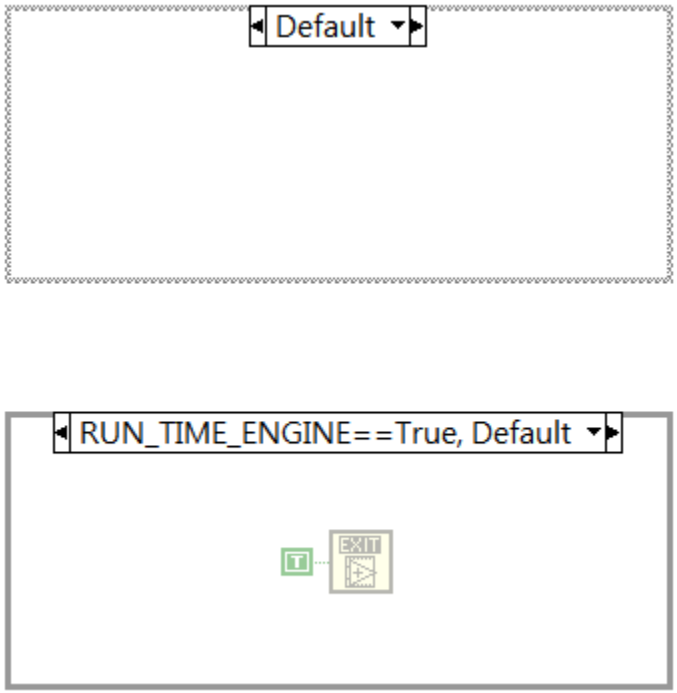
- Diagram Disable Structure：切換程式碼Enabled/Disabled
- 通常用於遮罩測試程式碼，或將來有可能會執行的程式碼



作者；曾智勇,chihyungtseng@hotmail.com

Structure

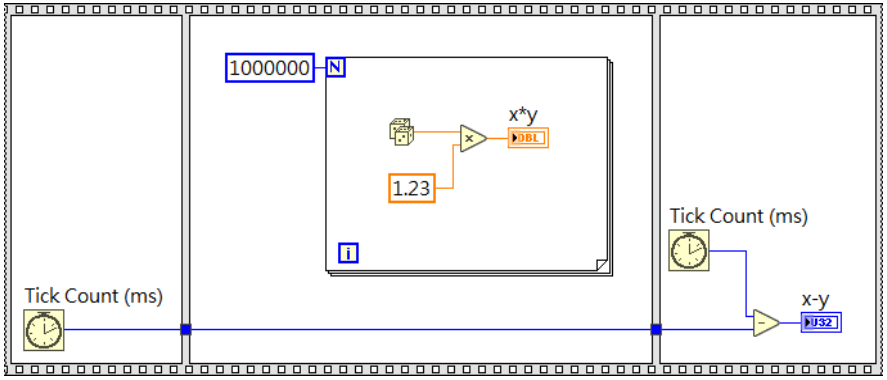
- Conditional Disable Structure：根據狀態決定程式碼Enabled/Disabled
- 通常用於執行檔階段要執行的程式碼

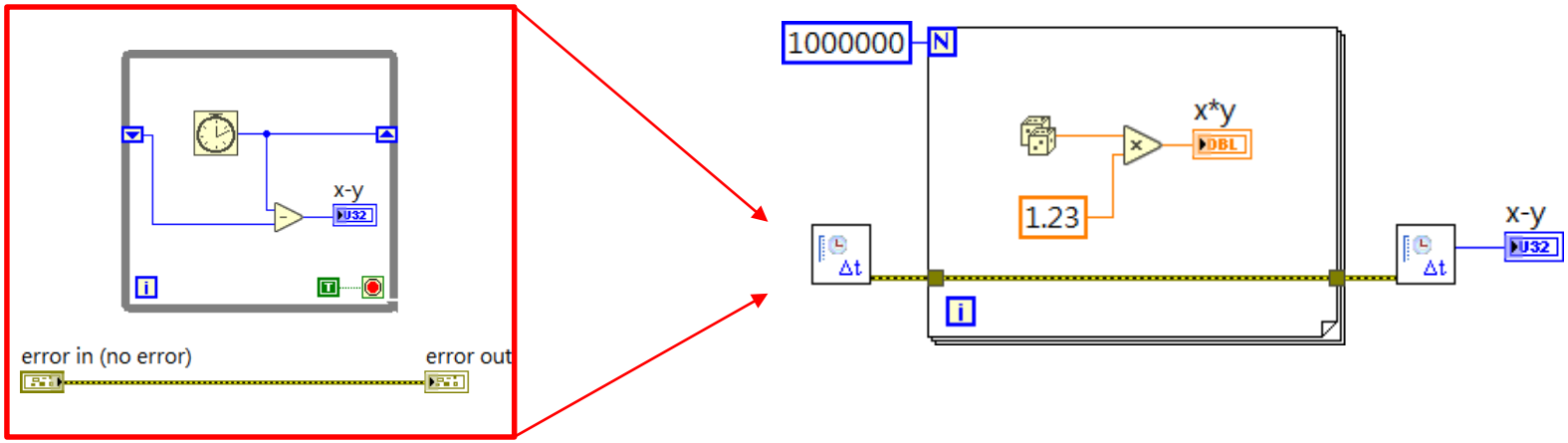


作者；曾智勇,chihyungtseng@hotmail.com

Structure

- 範例：使用副程式計算程式碼運行時間

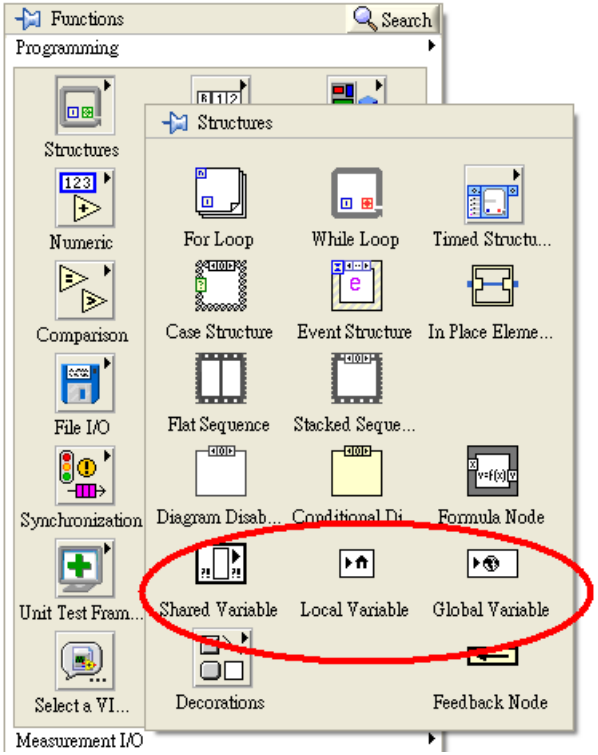


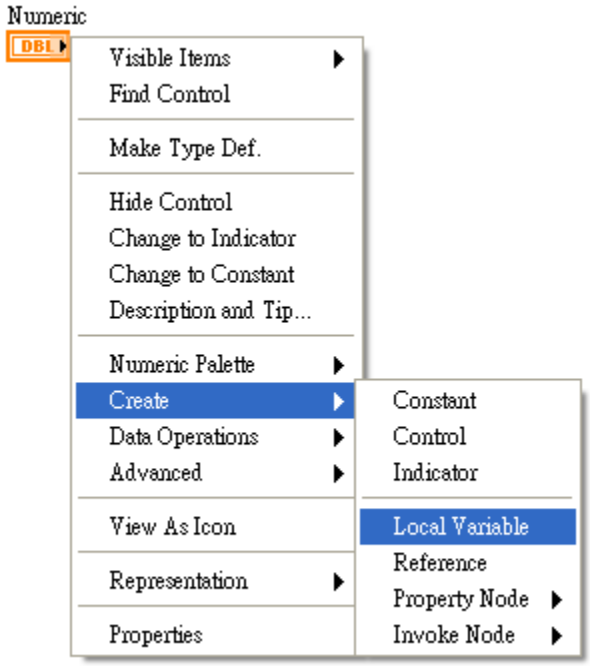


作者；曾智勇,chihyungtseng@hotmail.com

共變數

- Local Variable：同一個VI程式間傳遞變數
- Global Variable：不同VI程式間傳遞變數
- Shared Variable：不同電腦間傳遞變數

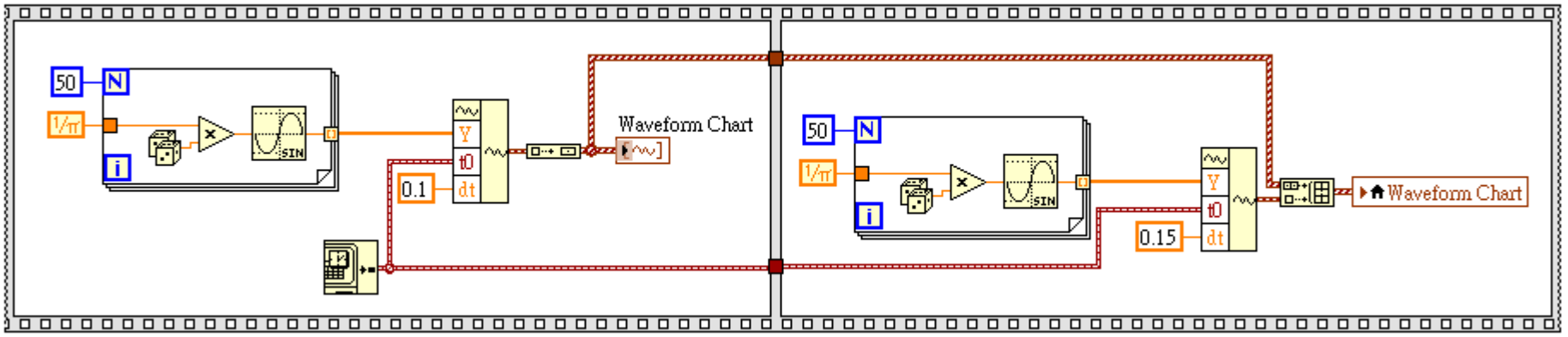




作者；曾智勇,chihyungtseng@hotmail.com

共變數

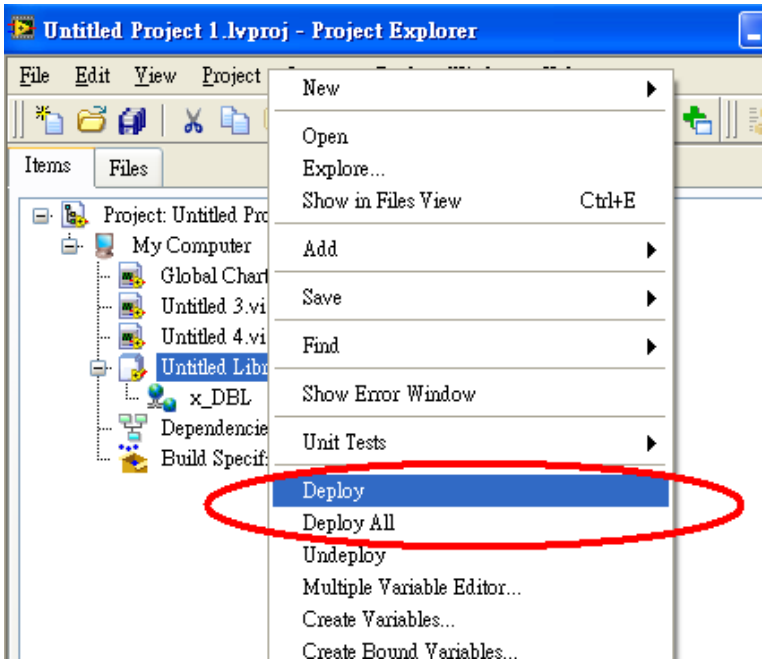
- 使用Local Variable先後繪製多層線

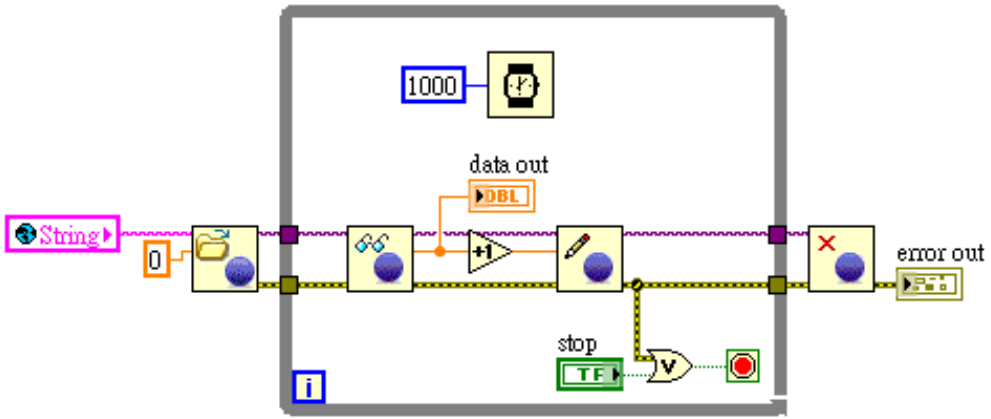


作者；曾智勇,chihyungtseng@hotmail.com

共變數

- Shared Variable：透過網路功能，在不同裝置間傳遞變數

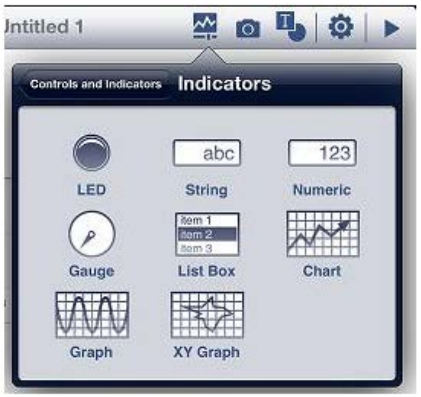




共變數

- Shared Variable：透過網路功能，在不同裝置間傳遞變數

Data Dashboard for LabVIEW 2.1 (http://www.ni.com/white-paper/14033/zht/)
(2013 年秋季推出，適用於 iOS 與 Android 平板電腦)



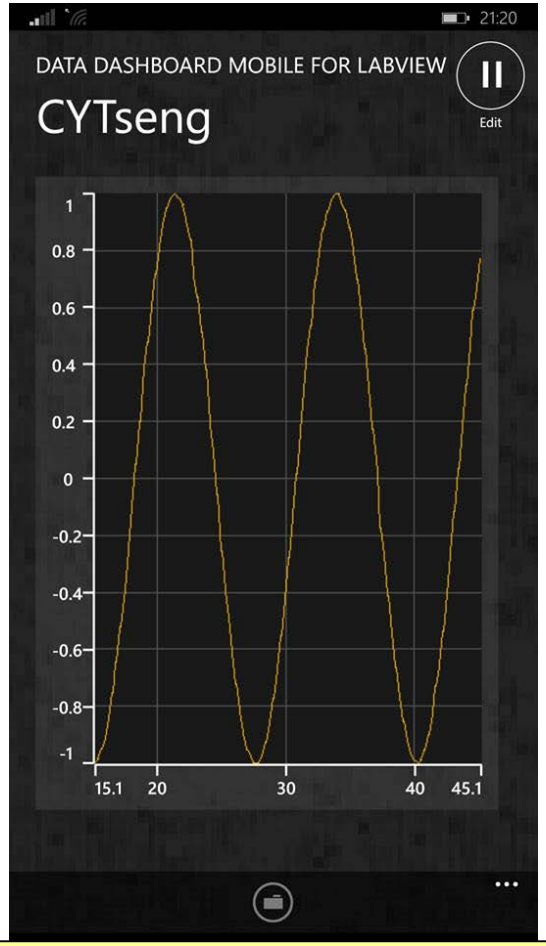
作者；曾智勇, chihyungtseng@hotmail.com

共變數

- 上述Data Dashboard功能僅限於 iOS 與 Android 平板電腦
- iPhone 與特定的 Android 智慧型手機、Windows Phone 智慧型手機與 Windows 8 電腦也提供了某些 Data Dashboard for LabVIEW 功能
- 目前這些平台所提供的功能如下：
  - 固定配置的介面
  - 可透過網路發佈共用變數或網路服務來存取資料
  - 包含圖形、字串與數字讀數在內的指示元件
  - 可存取內建裝置感測器 (例如加速規)

作者；曾智勇, chihyungtseng@hotmail.com

共變數



使用手機（WP 8.1）讀取**Shared variable**

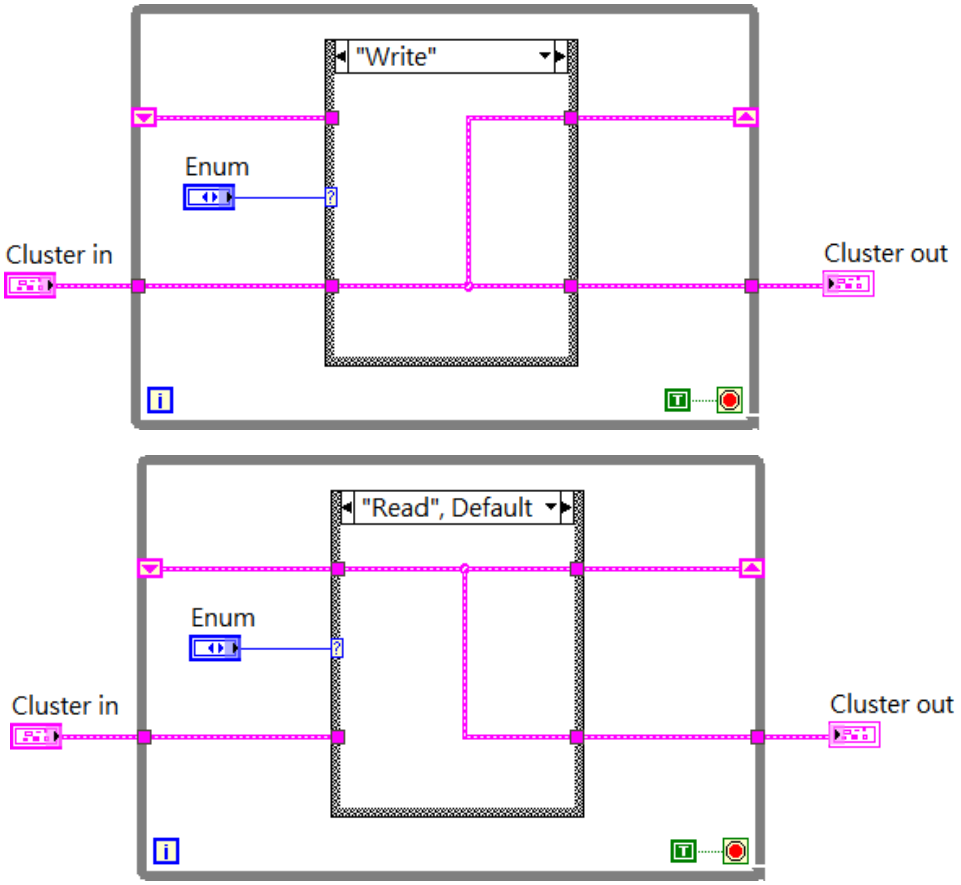
```
Result=
輸入單位：RT*kPa
輸出單位：kW*psi
輸入常數：12.43
輸出常數：6.34
```

使用手機（WP 8.1）存取**Web Service**

作者；曾智勇, chihyungtseng@hotmail.com

共變數

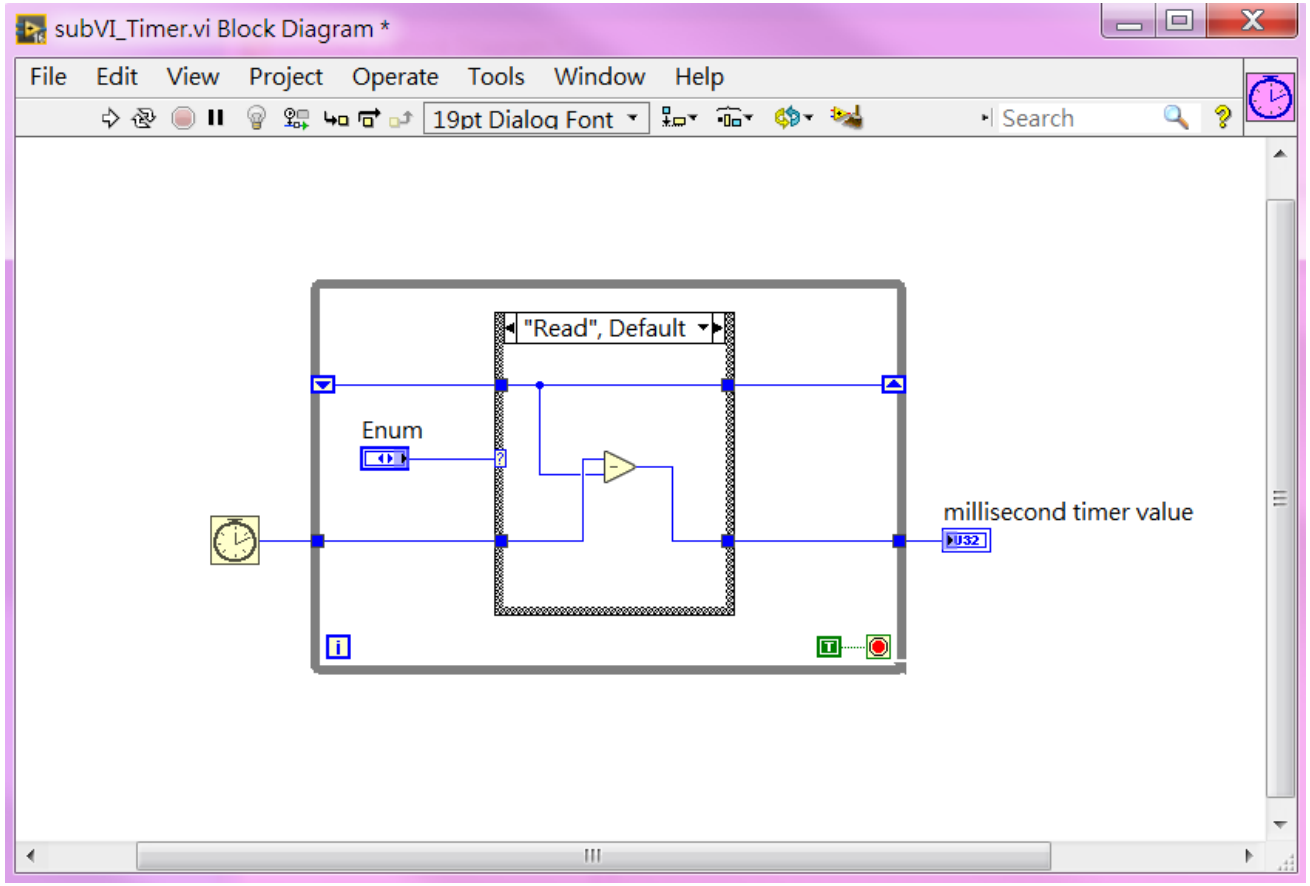
- Functional Global Variable, **FGV**
  - 執行一次的迴圈
    +
  - 沒有初始執的移位暫存器
    +
  - Case結構
    +
  - Enum選單
    +
  - 自定義的任意資料



作者；曾智勇,chihyungtseng@hotmail.com

共變數

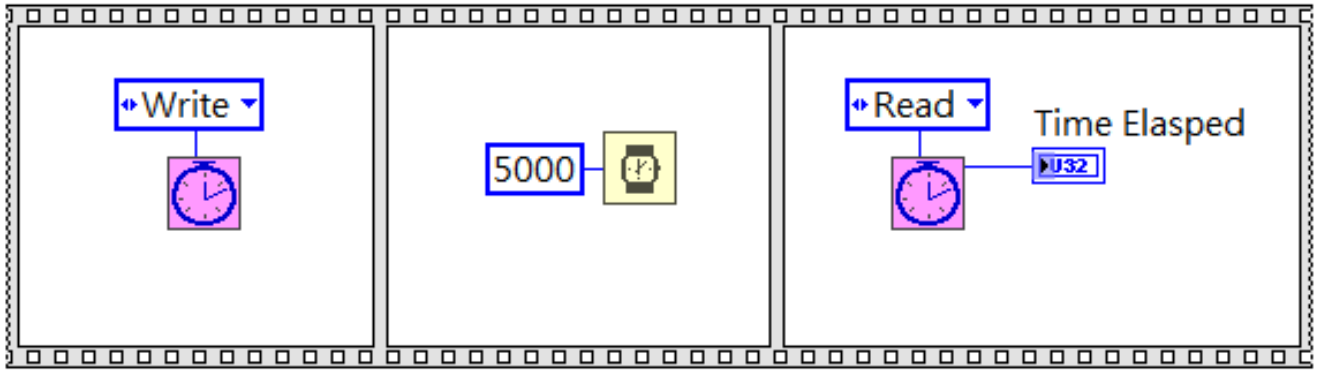
- Functional Global Variable, **FGV**

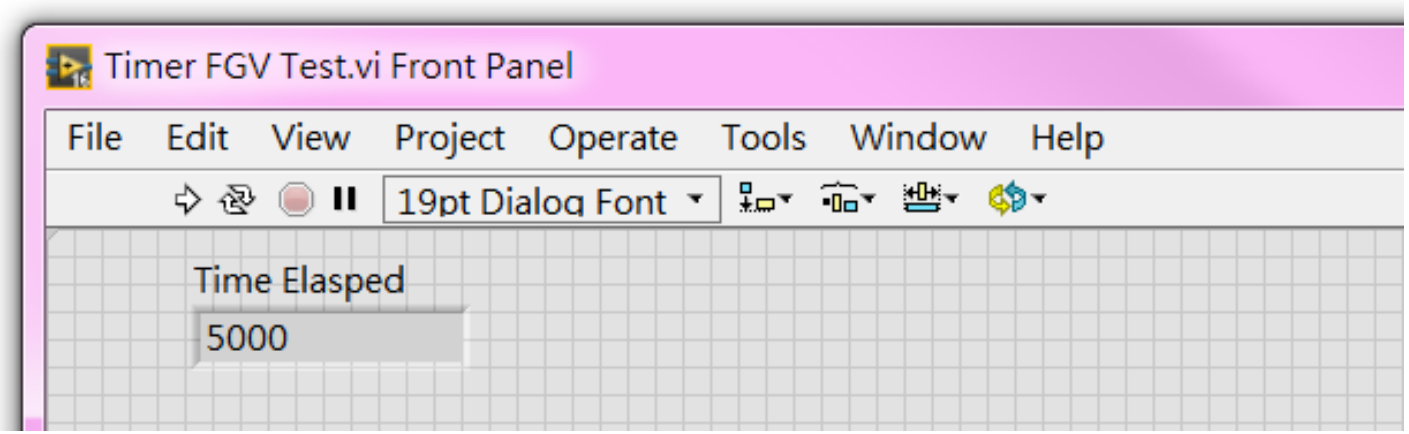


作者；曾智勇, chihyungtseng@hotmail.com

共變數

- Functional Global Variable, **FGV**

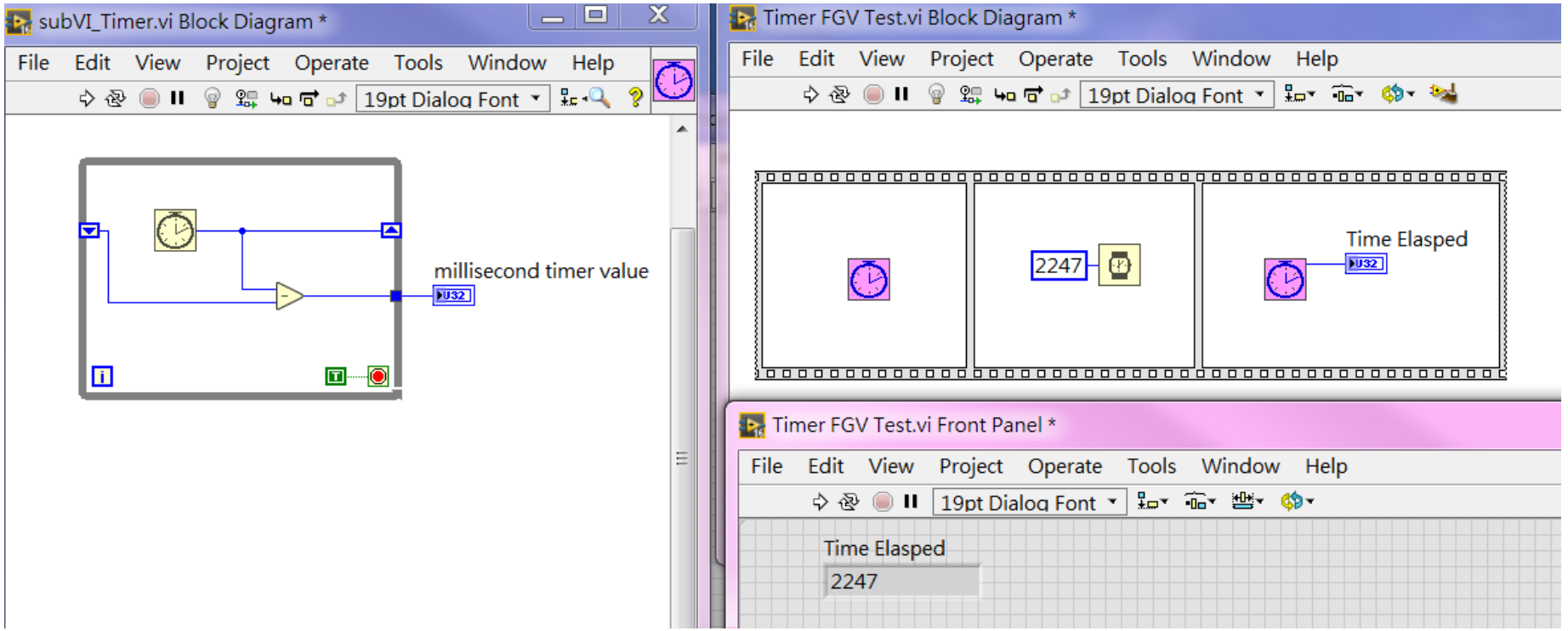




作者；曾智勇, chihyungtseng@hotmail.com

共變數

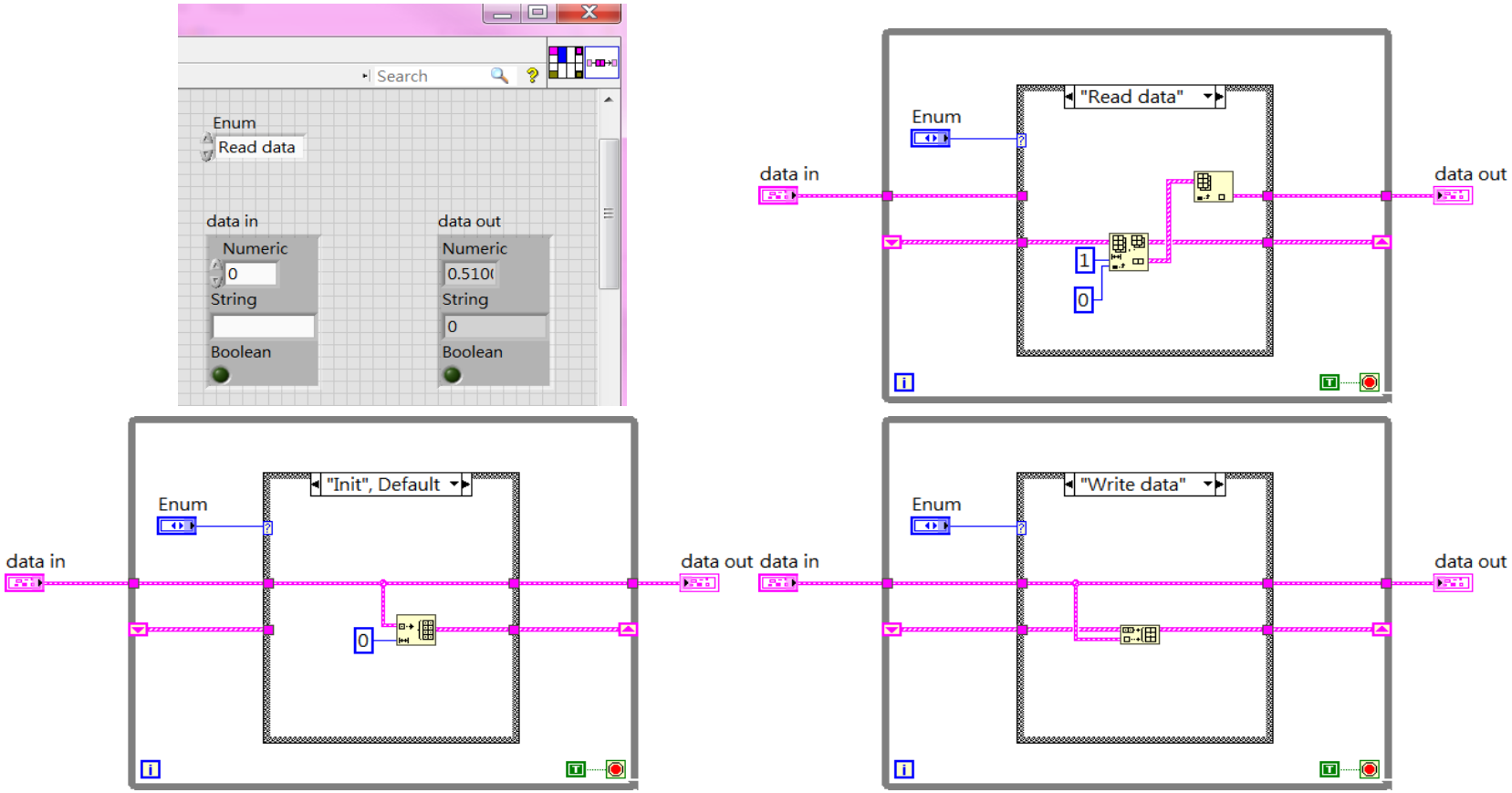
- **簡化版的FGV** (迴圈+移位暫存器+輸出)



作者；曾智勇, chihyungtseng@hotmail.com

共變數

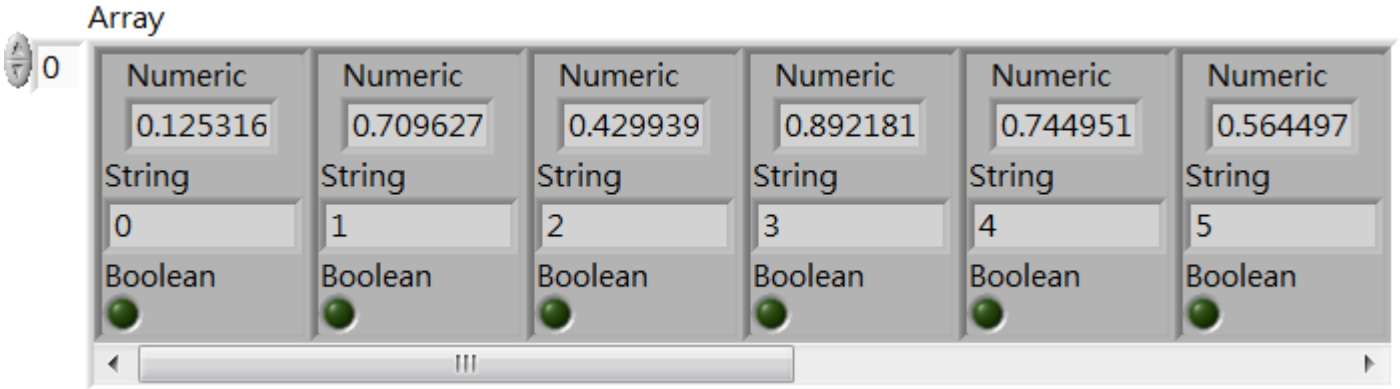
- **使用FGV製作FIFO** (先進先出的暫存器)

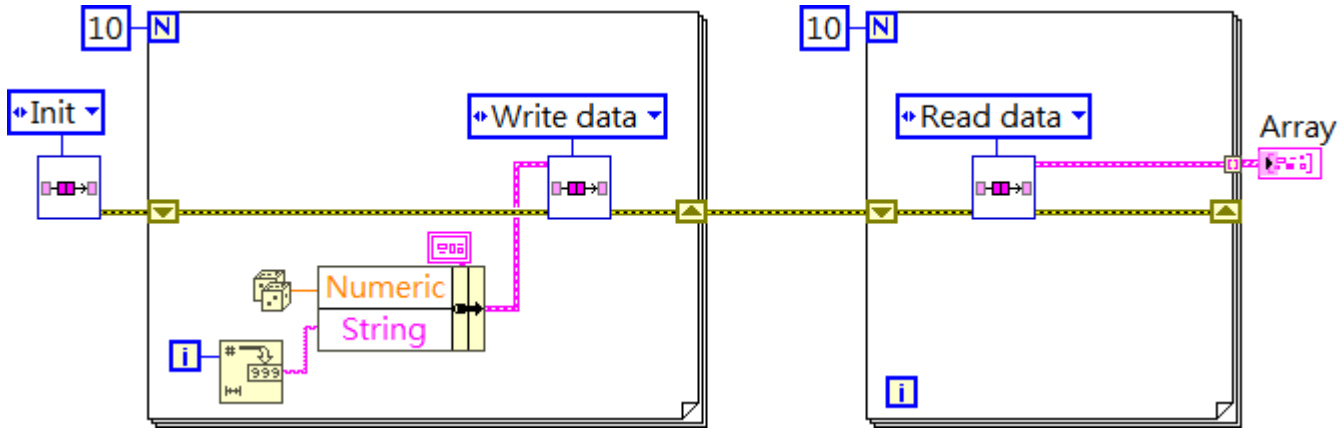


作者；曾智勇, chihyungtseng@hotmail.com

共變數

- **使用FGV製作FIFO** (先進先出的暫存器)





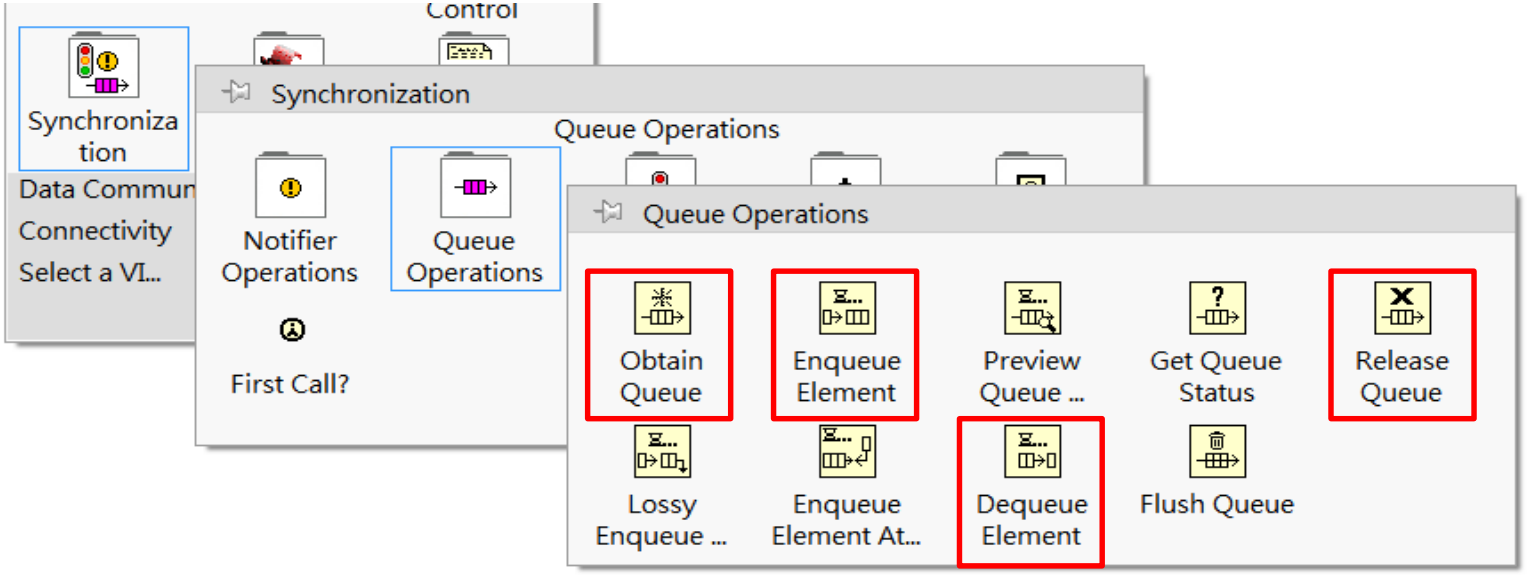
作者；曾智勇, chihyungtseng@hotmail.com

共變數

- **Queue** (內建的佇列式暫存器)

佇列，又稱為隊列（queue），是先進先出（FIFO, First-In-First-Out）的線性表。 在具體應用中通常用鍊表或者數組來實現。 佇列只允許在後端（稱為rear）進行插入操作，在前端（稱為front）進行刪除操作。

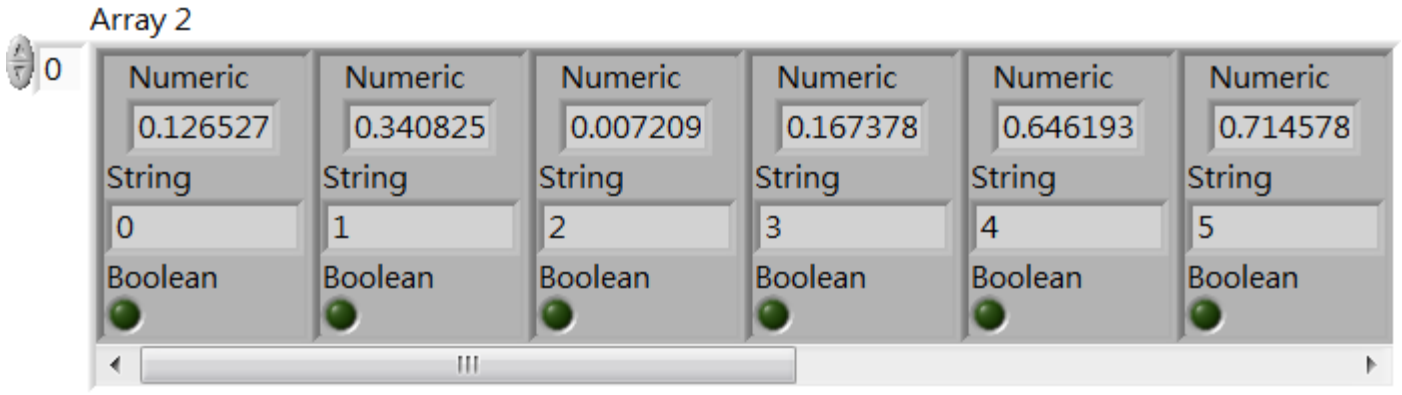
佇列- 維基百科，自由的百科全書 - Wikipedia
https://zh.wikipedia.org/zh-tw/队列

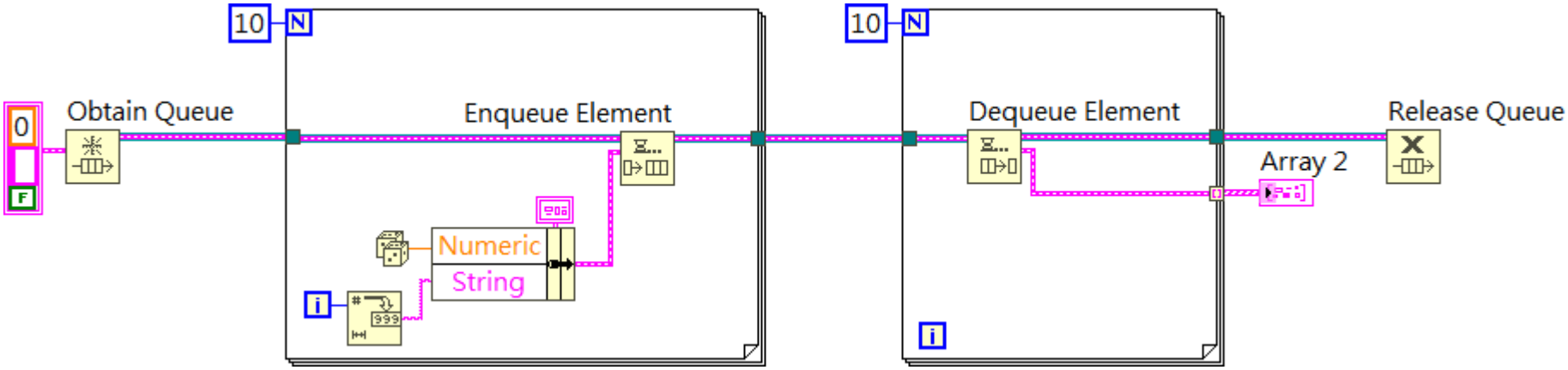


作者；曾智勇,chihyungtseng@hotmail.com

共變數

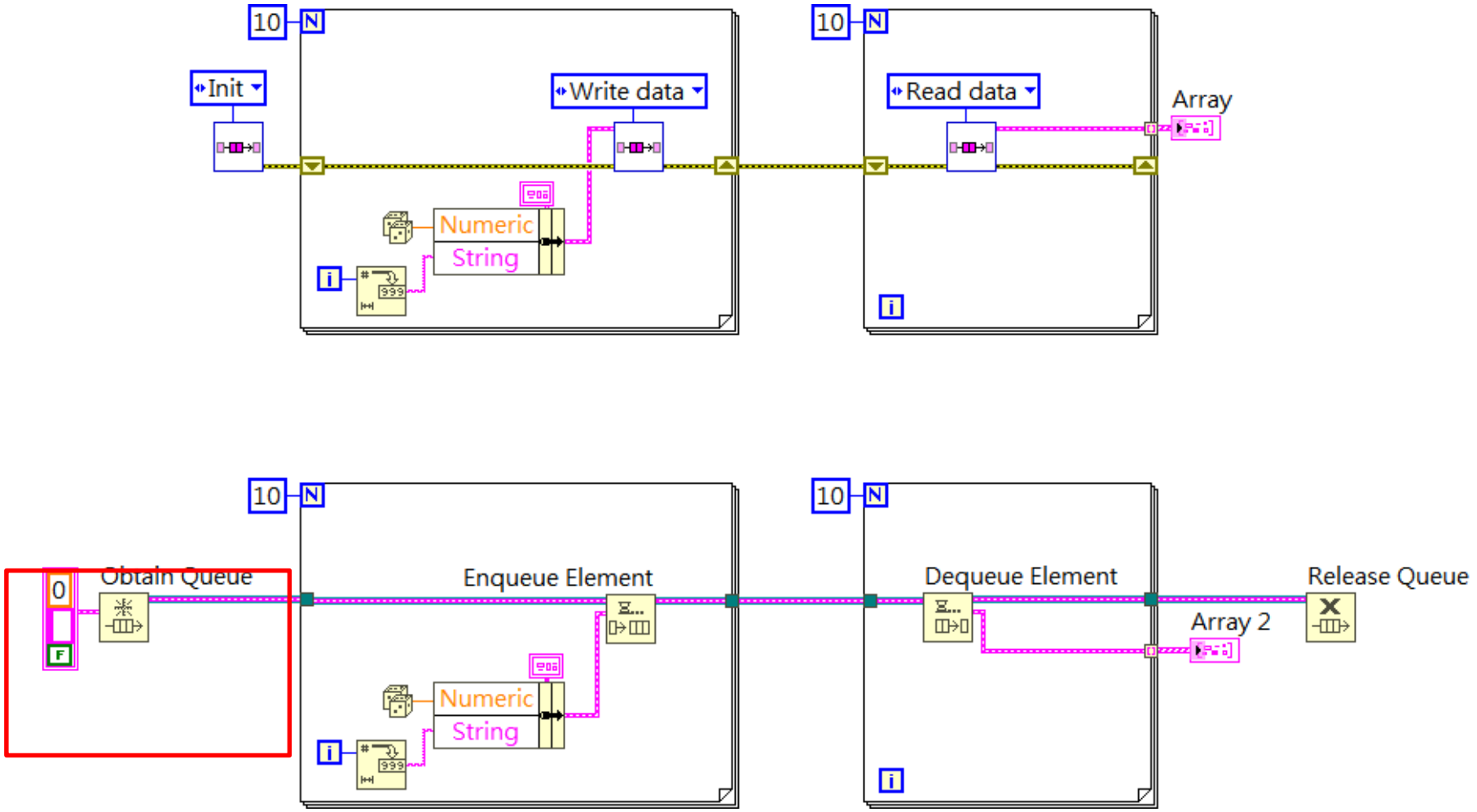
- **Queue** (內建的佇列式暫存器)





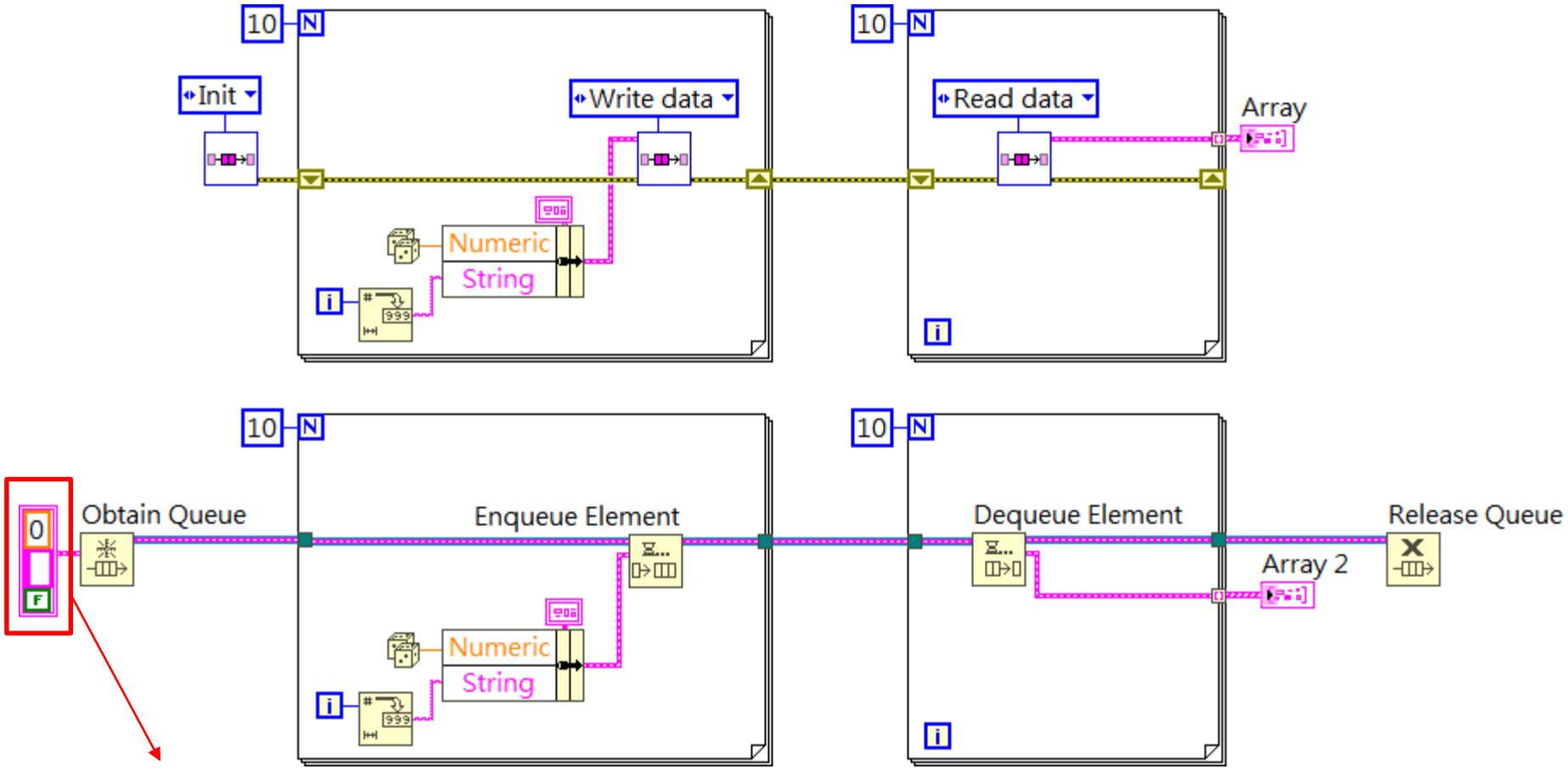
作者；曾智勇, chihyungtseng@hotmail.com

共變數



作者；曾智勇,chihyungtseng@hotmail.com

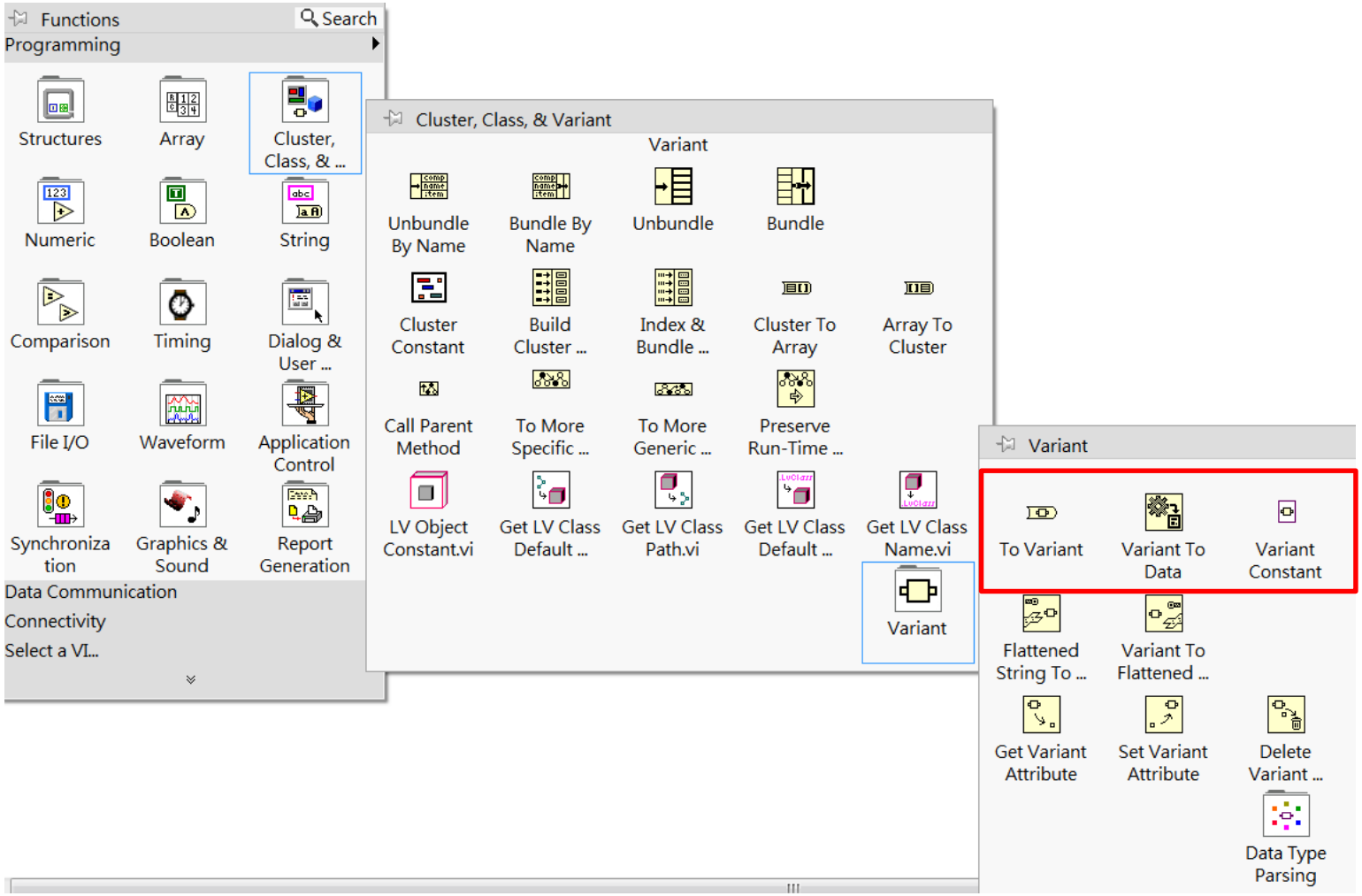
任意資料型態



可為任意資料型態

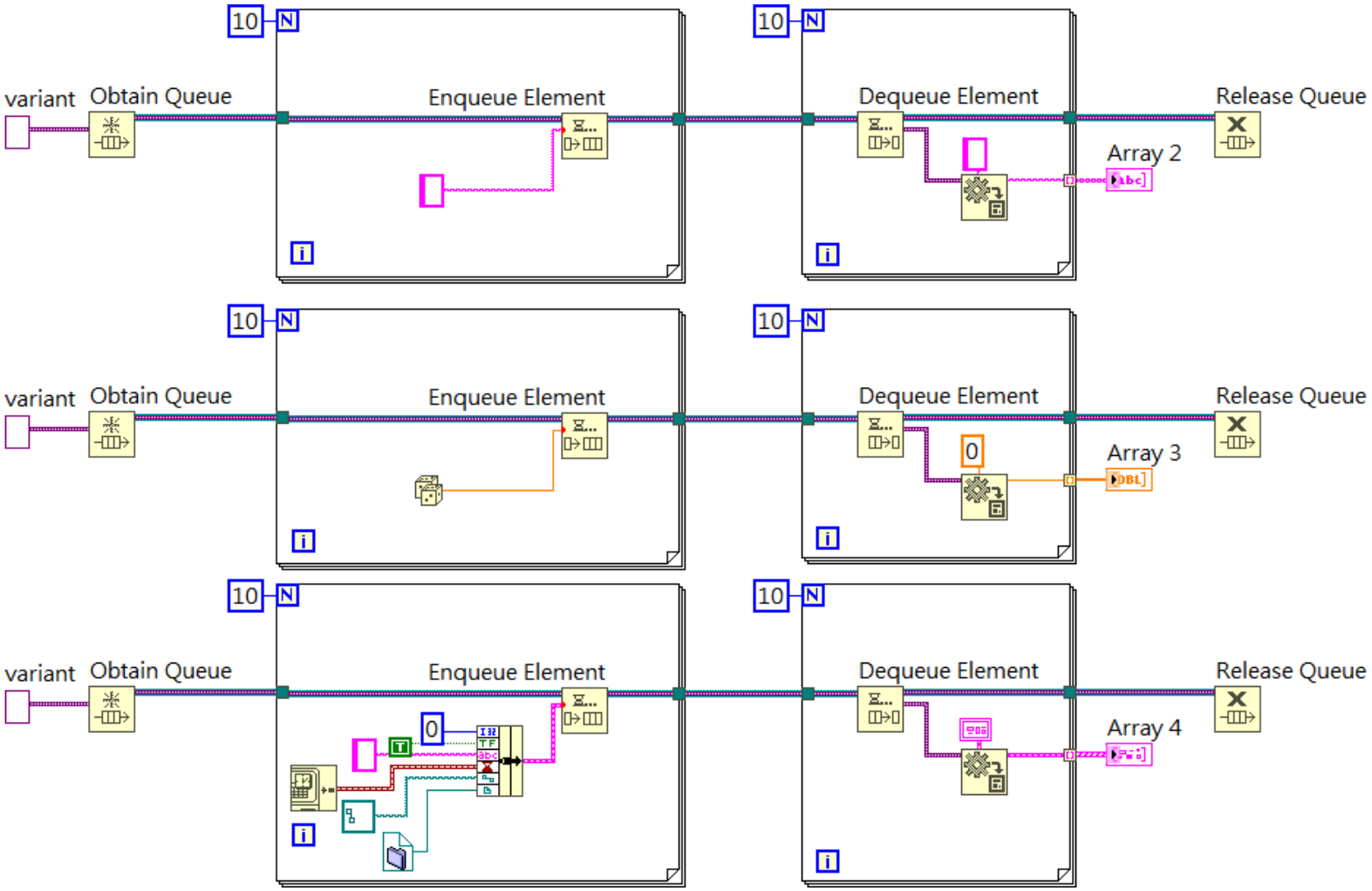
作者；曾智勇, chihyungtseng@hotmail.com

任意資料型態(Variant Data Type)



作者；曾智勇,chihyungtseng@hotmail.com

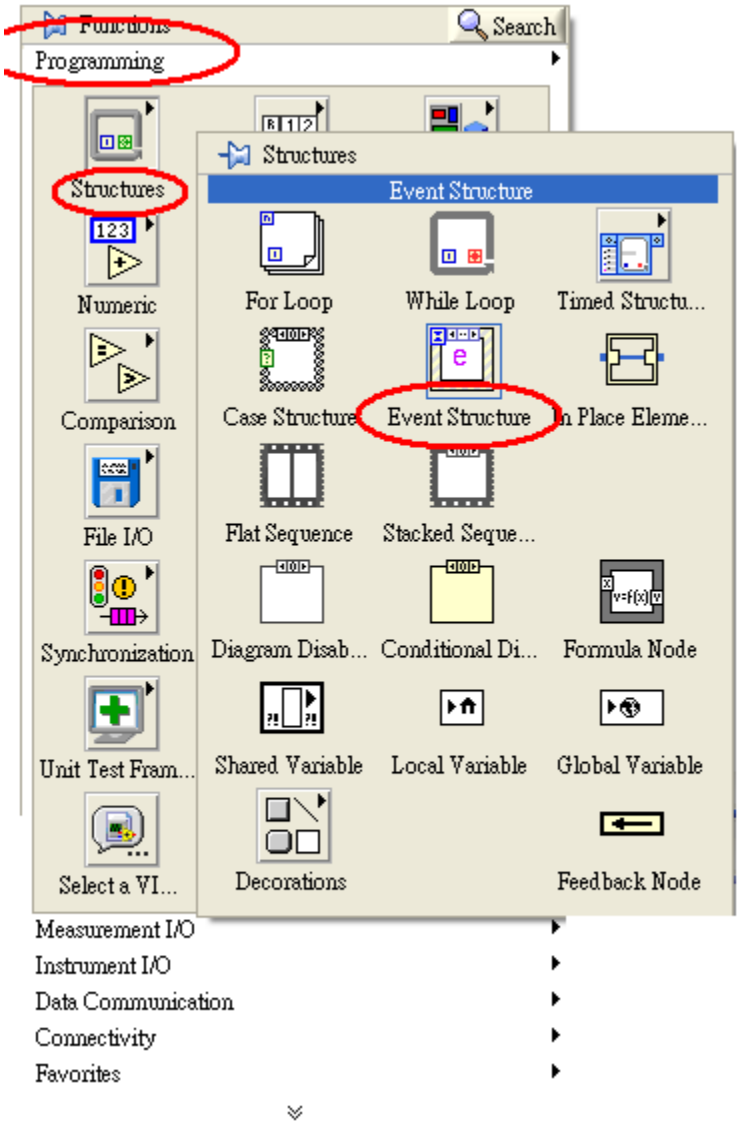
任意資料型態



作者；曾智勇,chihyungtseng@hotmail.com

Event Structure

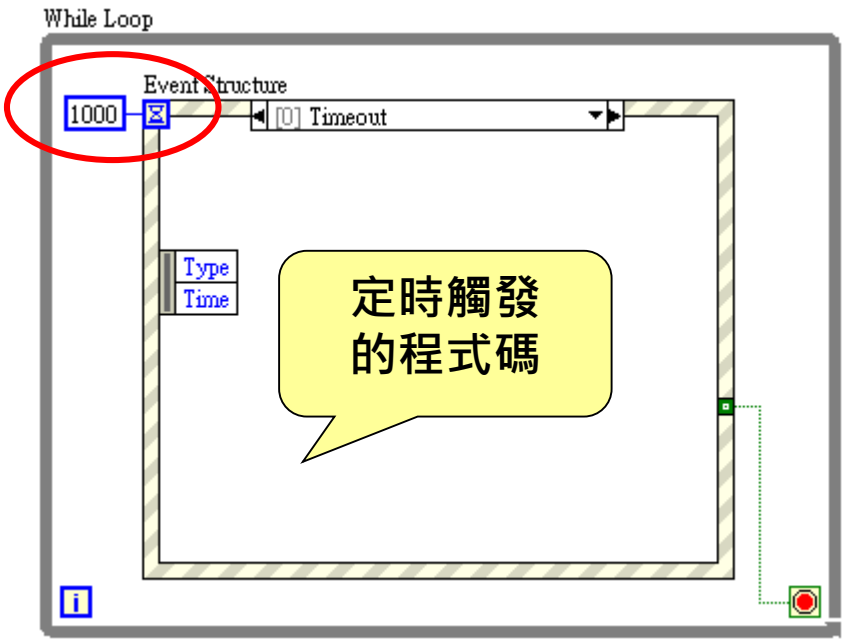
- 擷取人機介面的互動訊息後，觸發Event Stucture內的程式碼，以執行新的動作或工作程序。
- 除了元件的人機互動外，視窗互動、Menu互動、滑鼠、鍵盤皆可觸發Event Stucture。
- 只能接受人機介面產生的數值改變，使用Variable進行數值改變的事件，並不會觸發Event Structure內的程式碼。
- 必須放置在While Loop內，若在While Loop外只會觸發一次。

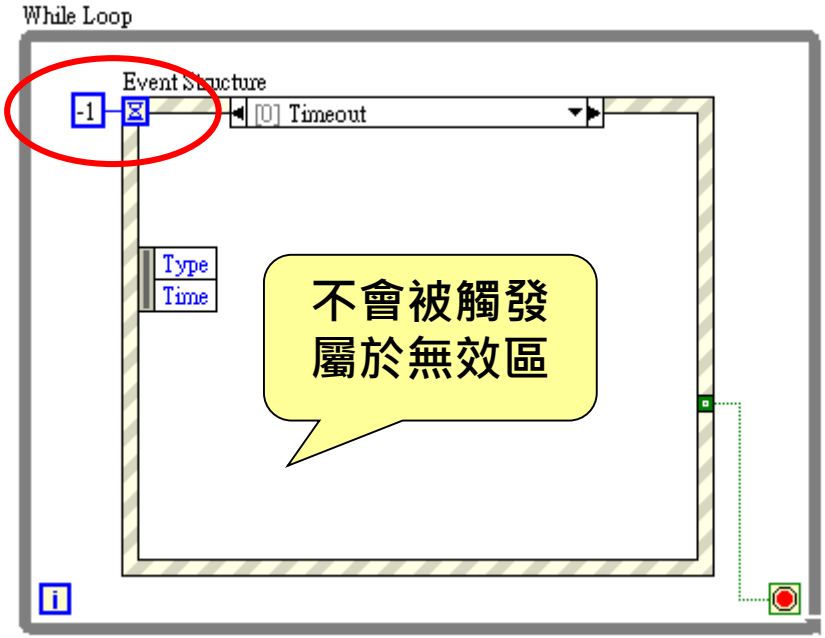


作者；曾智勇,chihyungtseng@hotmail.com

Event Structure

- 三種常見的架構
- (While Loop) + (Event Structure) + (Timeout 固定-1)
- (While Loop) + (Event Structure) + (Timeout 固定正整數)
- (While Loop) + (Event Structure) + (Timeout 隨程式架構變動)

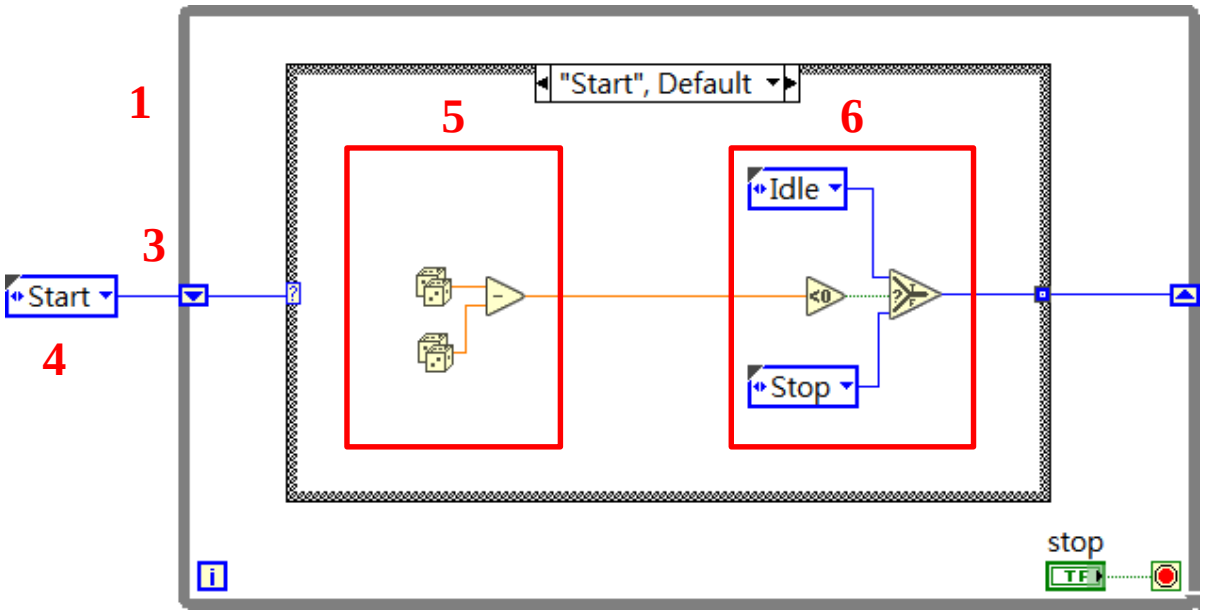




作者；曾智勇, chihyungtseng@hotmail.com

Design Pattern

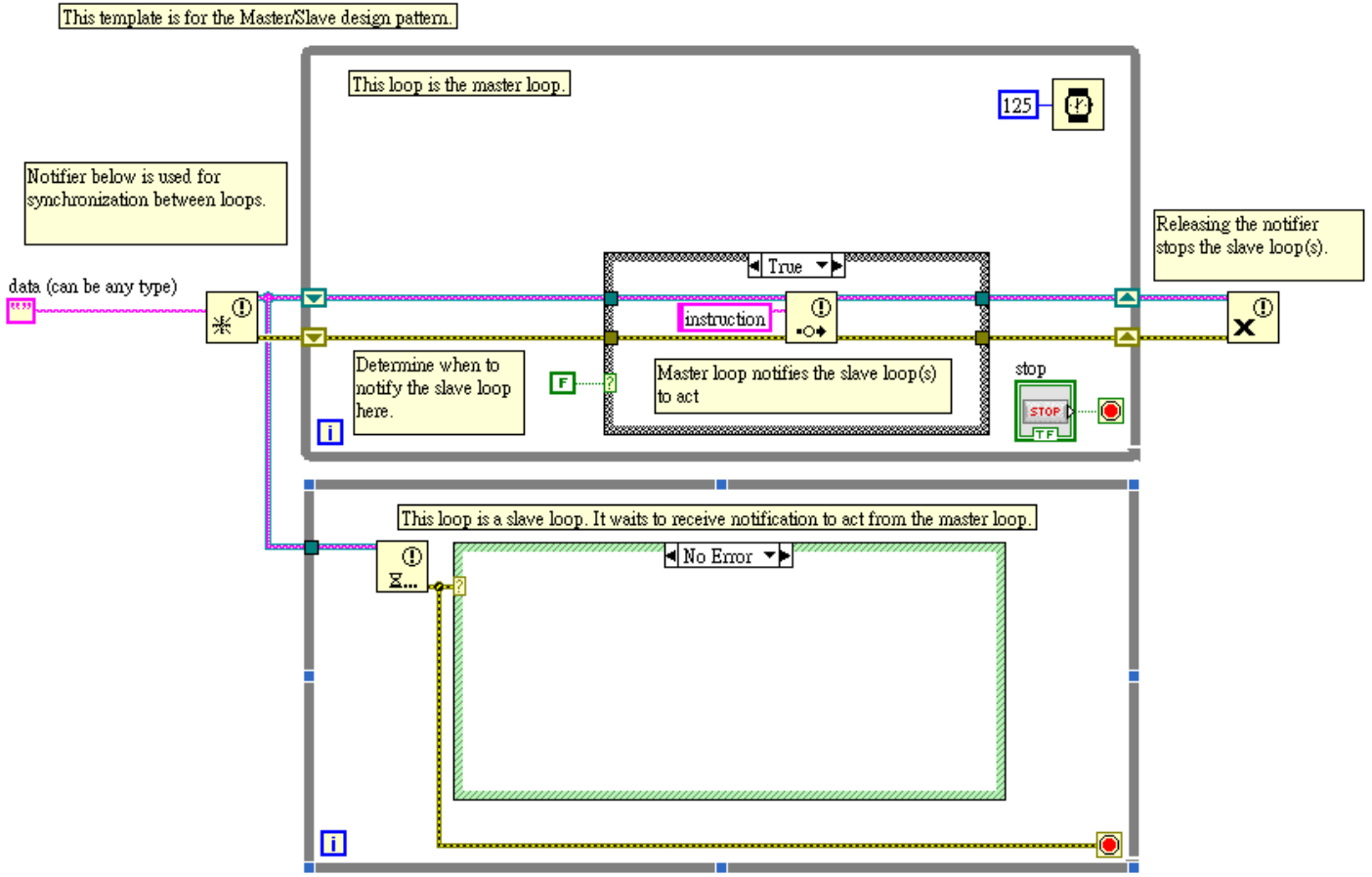
- State Machine
  - 1. 迴圈，While Loop
  - 2. 狀態，Case Structure
  - 3. 暫存器，Shift Register
  - 4. 下拉選單，Type def. Enum
  - 5. 狀態程式碼，State Functionality Code
  - 6. 下次狀態程式碼，Transition Code



作者；曾智勇,chihyungtseng@hotmail.com

Design Pattern

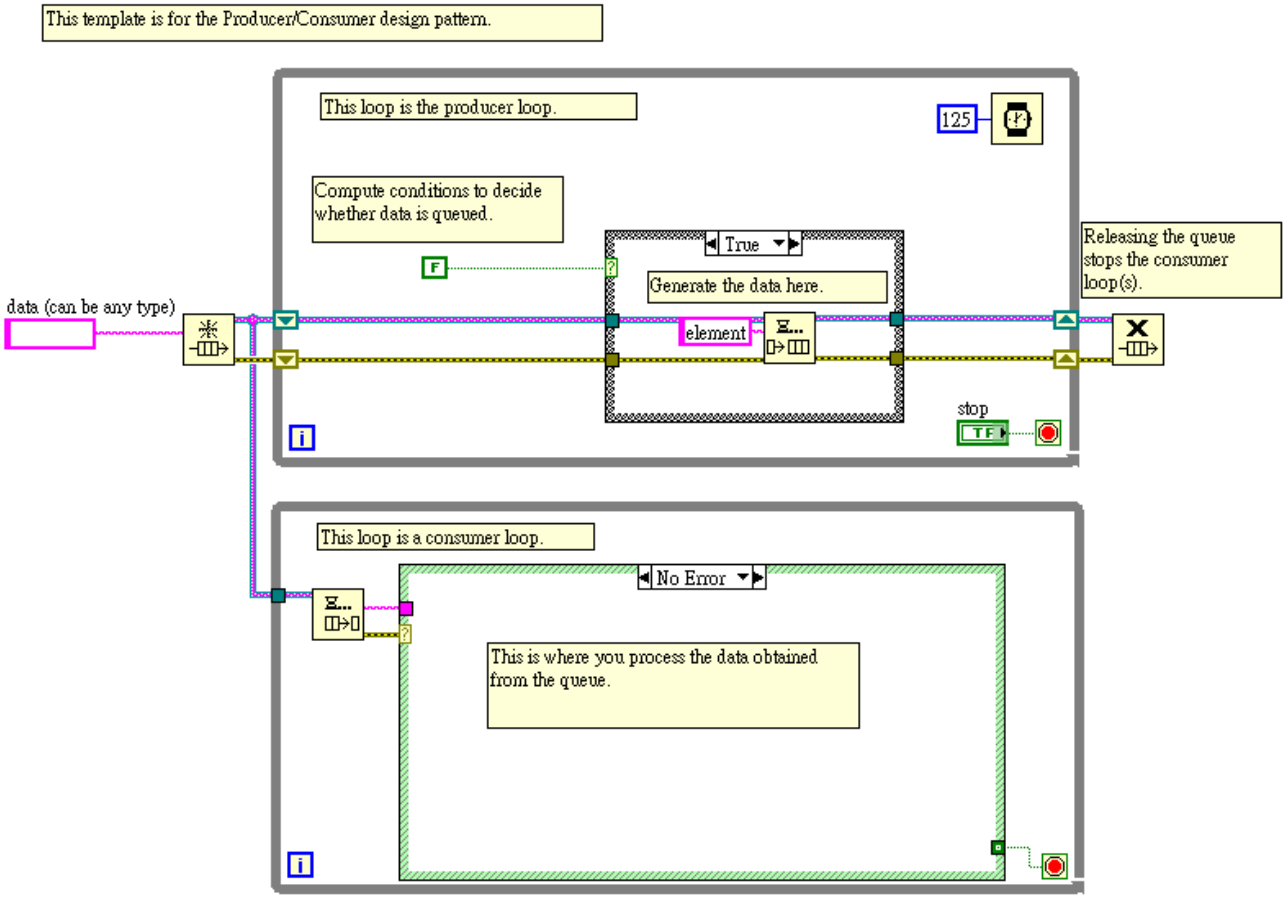
- Master-Slave



作者；曾智勇,chihyungtseng@hotmail.com

Design Pattern

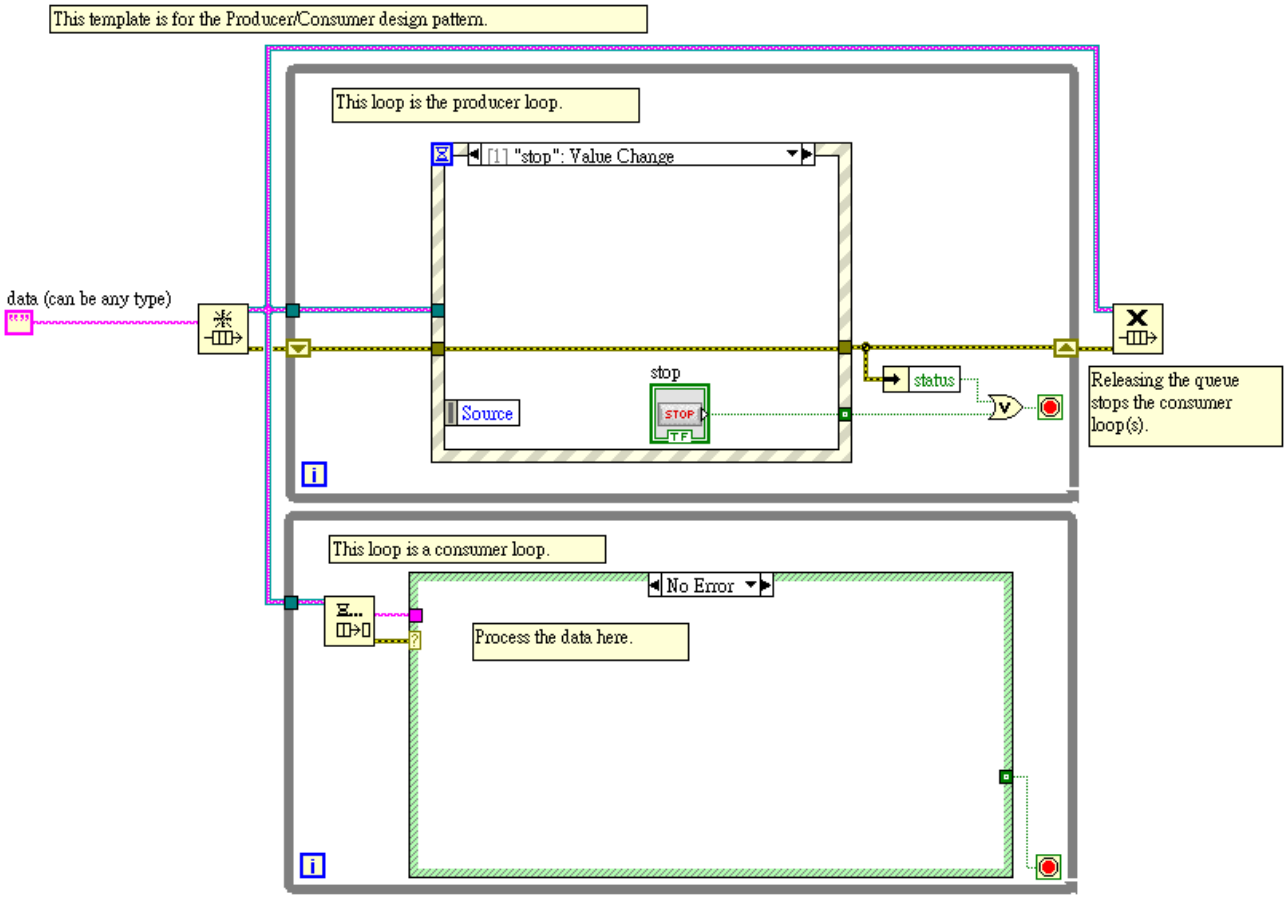
- Producer-Consumer(data)



作者；曾智勇,chihyungtseng@hotmail.com

Design Pattern

- Producer-Consumer(event)



作者；曾智勇,chihyungtseng@hotmail.com

儀器通訊

- 儀器通訊需考慮：
  - 介面、接腳（RS232、USB、GPIB、…、）
  - 通訊協定（SCPI、Modbus-RTU、…、）
  - 驅動程式（dll、VIs）

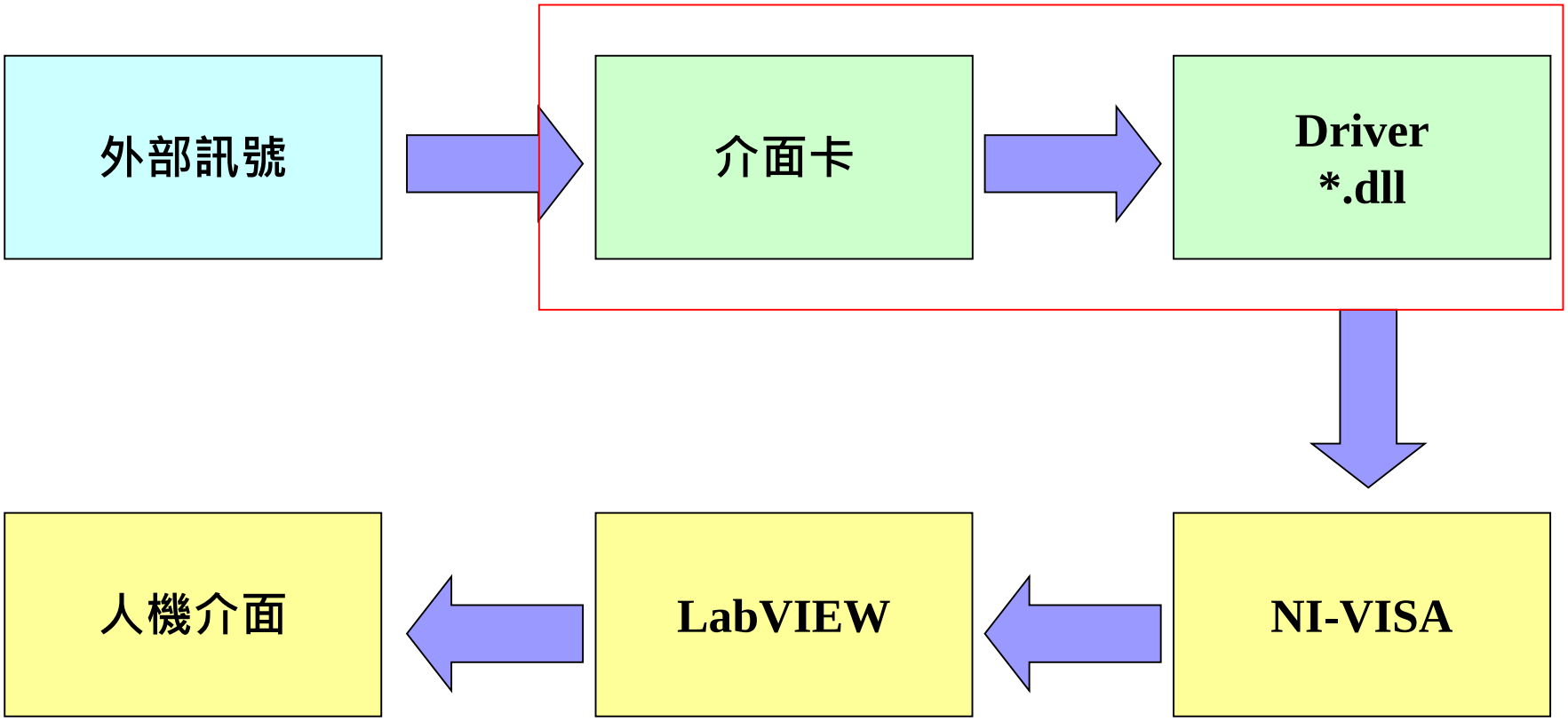
- SCPI可程式儀器標準命令（Standard Commands for Programmable Instruments，縮寫：SCPI）定義了一套用於控制可程式測試測量儀器的標準語法和命令。

- Modbus是一種串列通信協議，為使用可程式邏輯控制器（PLC）而發表的。允許多個設備連接在同一個網路上進行通信。MODBUS是工業領域通信協議的業界標準，並且現在是工業電子設備之間相當常用的連接方式。

作者；曾智勇,chihyungtseng@hotmail.com

儀器通訊

- 儀器通訊流程： NI 介面卡 + NI MAX（Measurement & Automation Explorer）



作者；曾智勇, chihyungtseng@hotmail.com

儀器通訊

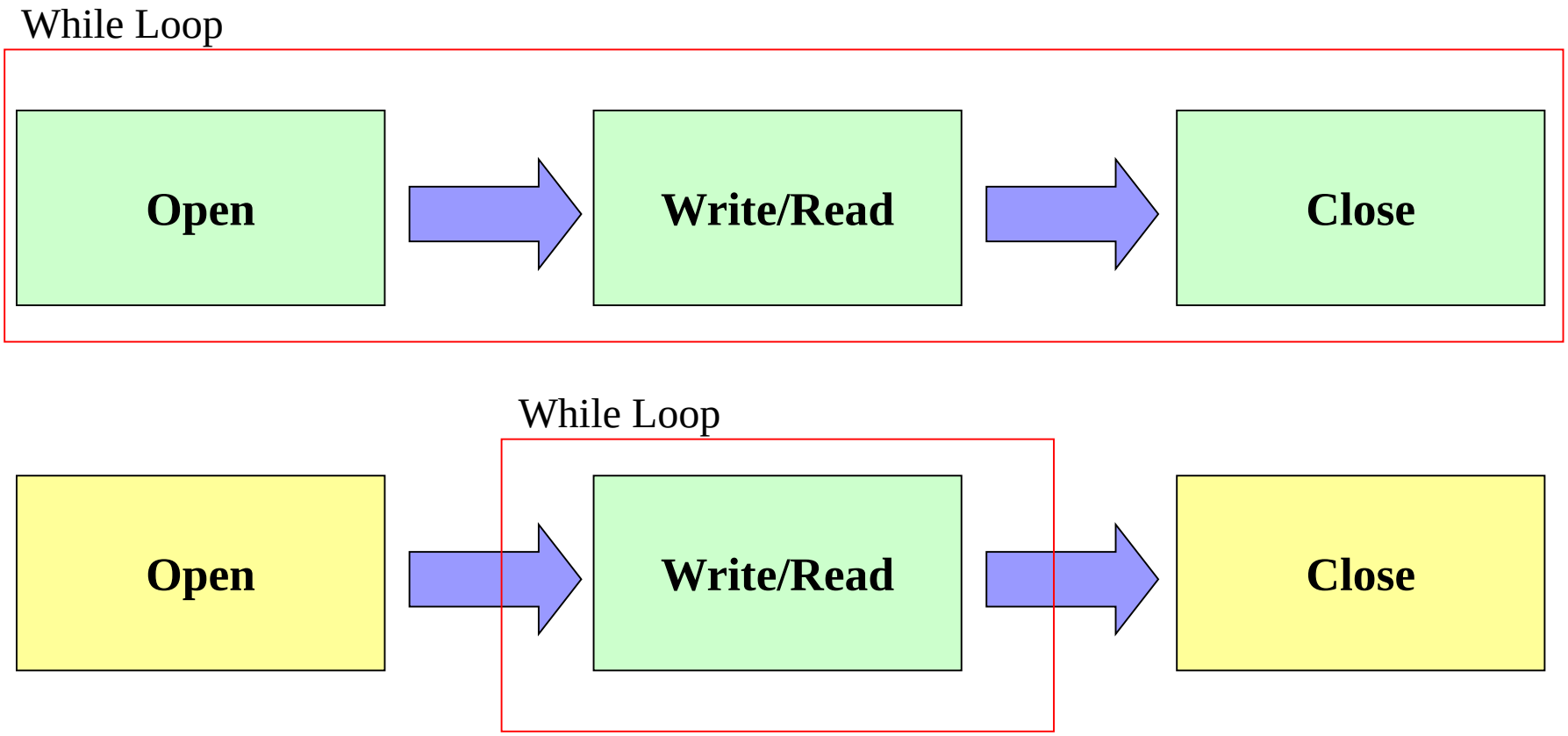
- VISA：虛擬儀器軟體架構(Virtual Instrument Software Architecture)，整合多種介面，在LabVIEW內，使用單一API進行儀器通訊工作。

Interface	Syntax	Classes
VXI INSTR	VXI[board]::VXI logical address[::INSTR]	Instr; VXI/GPIB-VXI MBD Instr; VXI/GPIB-VXI/VME RBD Instr
VXI MEMACC	VXI[board]::MEMACC	VXI/GPIB-VXI/VME MemAcc
VXI BACKPLANE	VXI[board][::VXI logical address]::BACKPLANE	VXI/GPIB-VXI Backplane
VXI SERVANT	VXI[board]::SERVANT	VXI Servant
GPIB-VXI INSTR	GPIB-VXI[board]::VXI logical address[::INSTR]	VXI/GPIB-VXI MBD Instr; VXI/GPIB-VXI/VME RBD Instr
GPIB-VXI MEMACC	GPIB-VXI[board]::MEMACC	VXI/GPIB-VXI/VME MemAcc
GPIB-VXI BACKPLANE	GPIB-VXI[board][::VXI logical address]::BACKPLANE	VXI/GPIB-VXI Backplane
GPIB INSTR	GPIB[board]::primary address[::GPIB secondary address][::INSTR]	GPIB Instr
GPIB INTFC	GPIB[board]::INTFC	GPIB BoardInterface
GPIB SERVANT	GPIB[board]::SERVANT	N/A
PXI INSTR	PXI[bus]::device[::function][::INSTR]	PXI Instr
PXI INSTR	PXI[interface]::[bus-]device[.function][::INSTR]	PXI Instr
PXI MEMACC	PXI[interface]::MEMACC	PXI MemAcc
Serial INSTR	ASRL[board][::INSTR]	Serial Instr
TCPIP INSTR	TCPIP[board]::host address[::LAN device name][::INSTR]	TCP/IP Instr
TCPIP SOCKET	TCPIP[board]::host address::port::SOCKET	TCP/IP Socket
USB INSTR	USB[board]::manufacturer ID::model code::serial number[::USB interface number][::INSTR]	USB Instr
USB RAW	USB[board]::manufacturer ID::model code::serial number[::USB interface number]::RAW	USB Raw

作者；曾智勇,chihyungtseng@hotmail.com

儀器通訊

- 兩種撰寫架構：



作者；曾智勇, chihyungtseng@hotmail.com

Property Node（屬性節點）

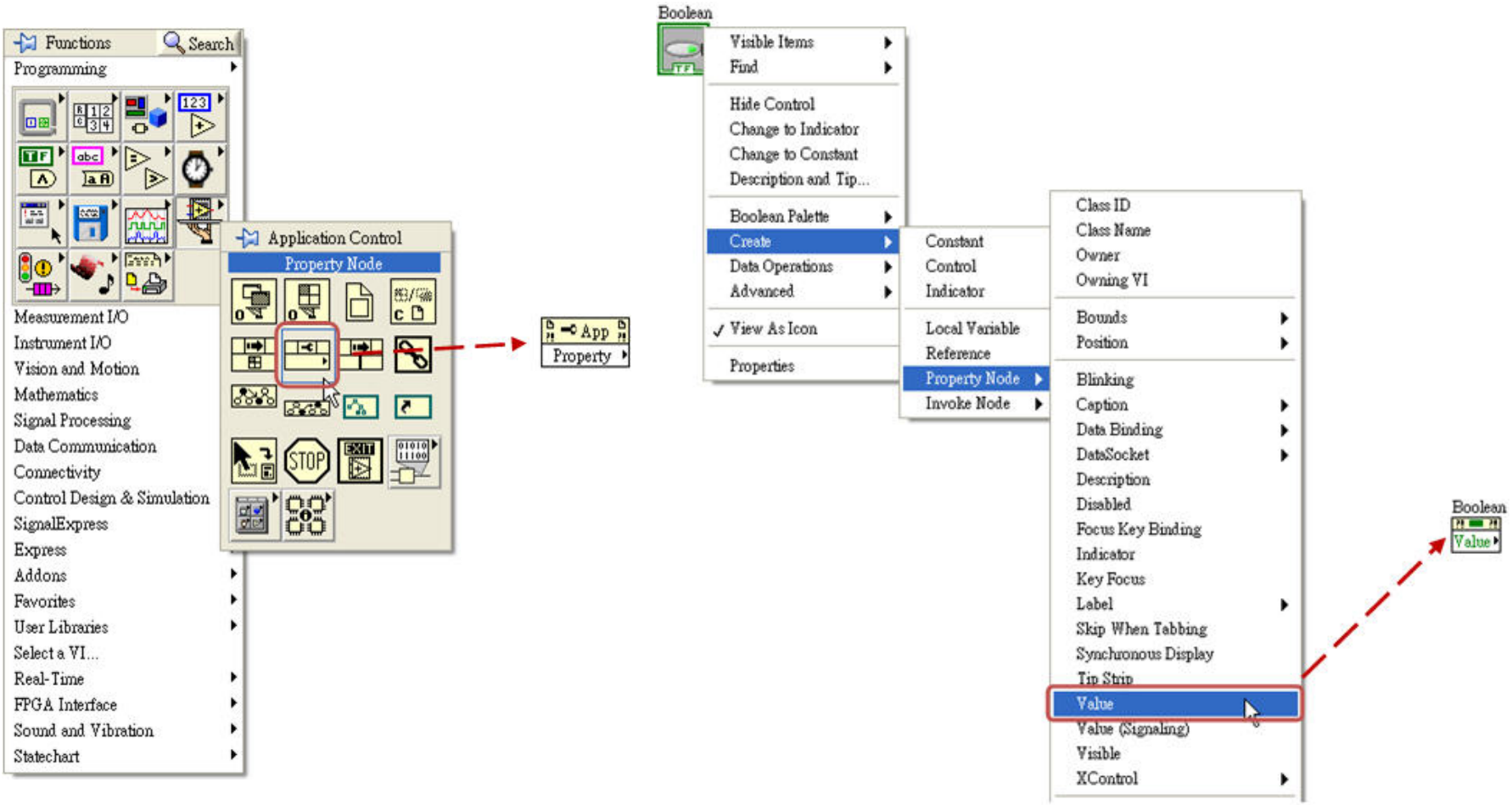
- 屬性節點算是LabVIEW中相當重要的元件，而且內容也相當的廣泛，幾乎所有的控制或是顯示元件都具有許多各式各樣的屬性可以使用，而且當你使用到ActiveX元件時，也多半使用屬性節點來控制。

- 通常我們會直接在元件上滑鼠右鍵選擇建立相對應會屬性，或者是先建立Property Node再利用Control Reference來指定屬性所連結的物件，Control Reference你可以想像成物件的指標，用來告訴屬性元件，目前所要被調整的元件。

- 如果你想要控制一個布林控制開關的值，你可以直接在物件上滑鼠右鍵選擇[Create]>>[Property Node]>>[Value]來建立Value這個屬性的節點。

作者；曾智勇,chihyungtseng@hotmail.com

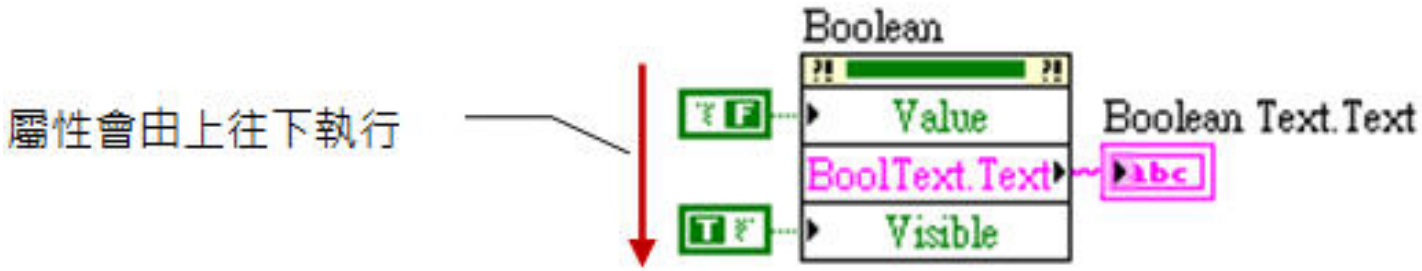
Property Node（屬性節點）



作者；曾智勇,chihyungtseng@hotmail.com

Property Node（屬性節點）

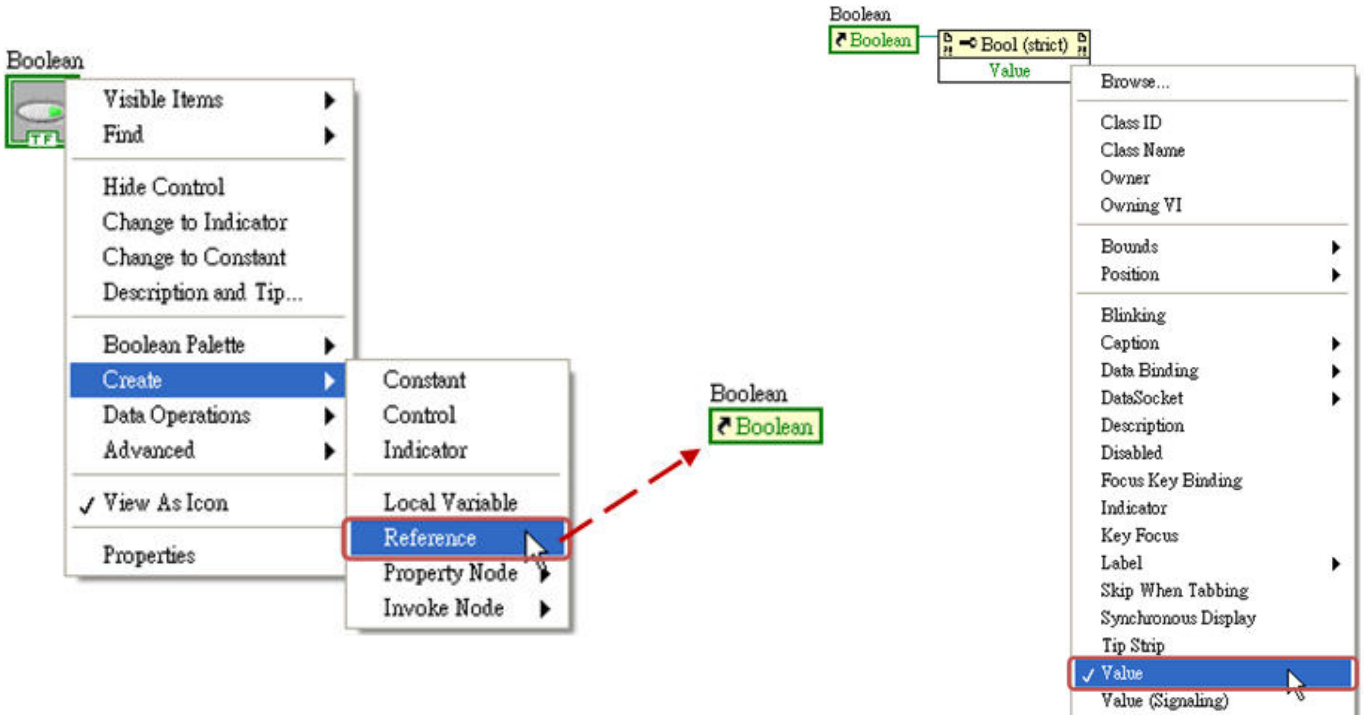
- 一個屬性節點可以使用下拉的方式來一次進行多個屬性操作，不過它的執行順序會由上往下執行，且當中途有任一個屬性操作的過程發生錯誤，則屬性節點會立即停止動作，然後將錯誤訊息丟出。



作者；曾智勇,chihyungtseng@hotmail.com

Property Node（屬性節點）

- 用Control Reference的方式來控制屬性，那你需要先建立物件的Control Reference，然後再建立一個Property Node元件，再將其連結在一起，接著你就可以用滑鼠去設定所要控制的屬性了。然後將Control Reference接在Property Node元件上，就可以用滑鼠選擇所需的屬性。



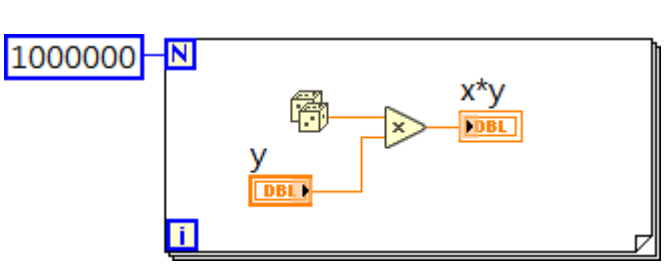
作者；曾智勇,chihyungtseng@hotmail.com

Property Node（屬性節點）

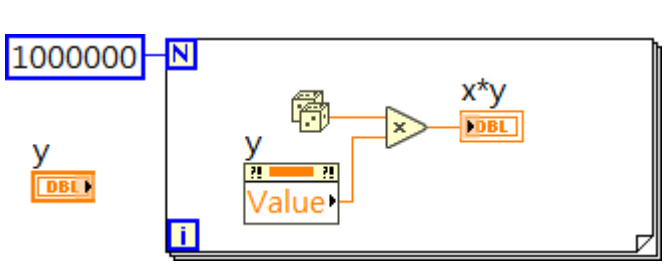
- 用於資料傳輸時的效能差異：

直接接線(最省資源) → 區域變數/全域變數 → 屬性節點(浪費資源)

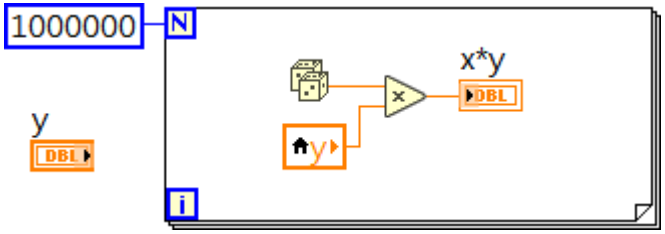
(Local / Global)　　(property node)



100 ms



43,000 ms



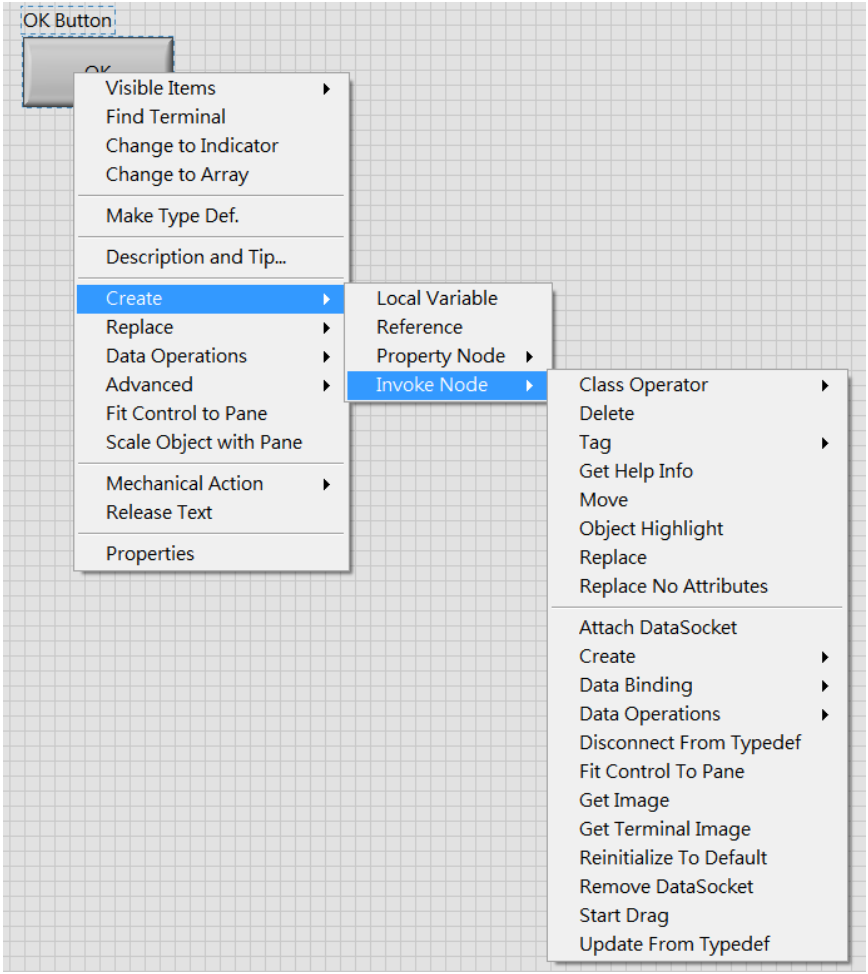
130 ms

作者；曾智勇, chihyungtseng@hotmail.com

Invoke Node（方法節點）

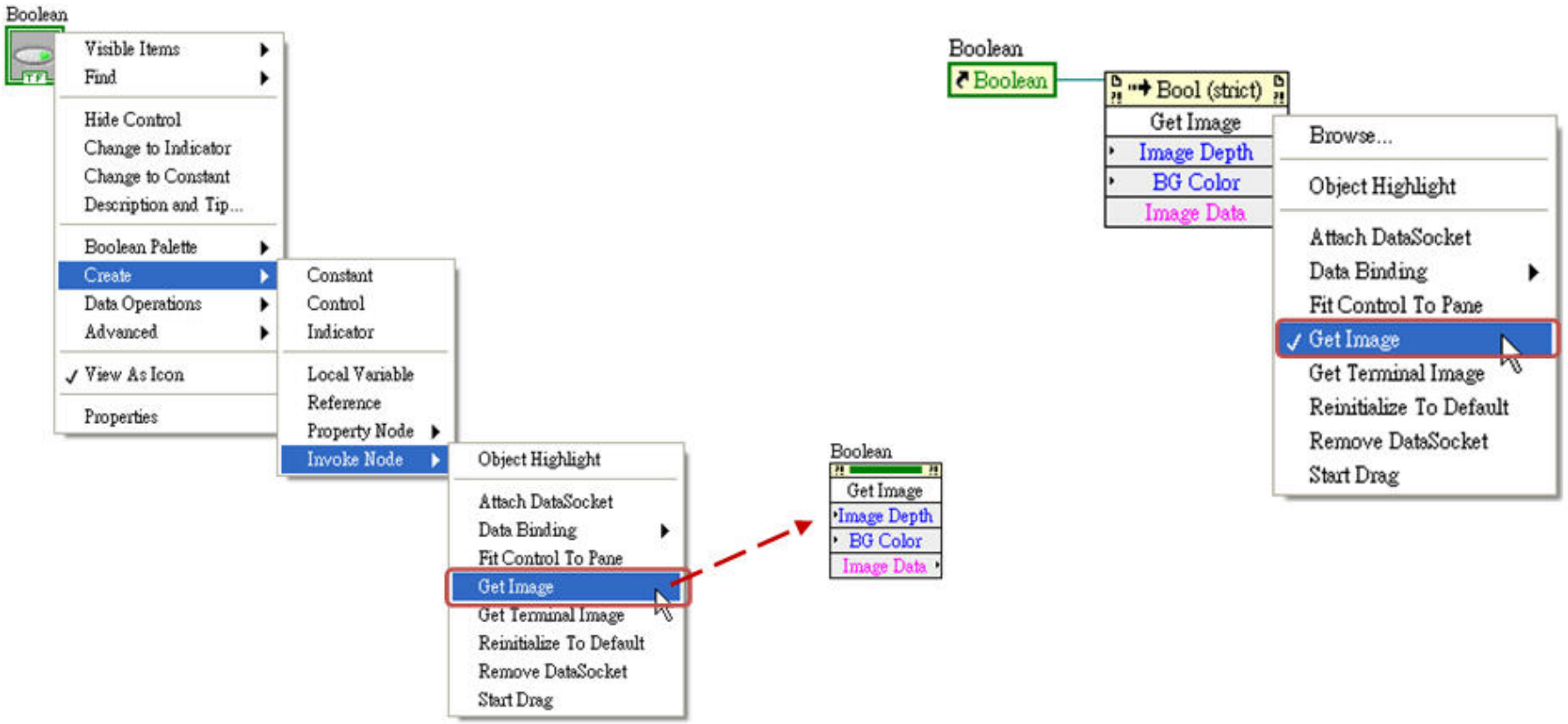
- 方法節點(Invoke Node)和屬性節點不同在於，方法節點用來呼叫物件的功能使用，屬性節點用來存取物件的屬性，假設想要讓程式自動置中或是將目前Front Panel存成圖片資訊，可以使用方法節點來達到此功能。

- 建立方式和屬性節點一樣，可以在元件上用右鍵選擇[Create]>>[Invoke Node]再選擇所需要的動作，或是使用Control Reference再配合Invoke Node來使用。



作者；曾智勇,chihyungtseng@hotmail.com

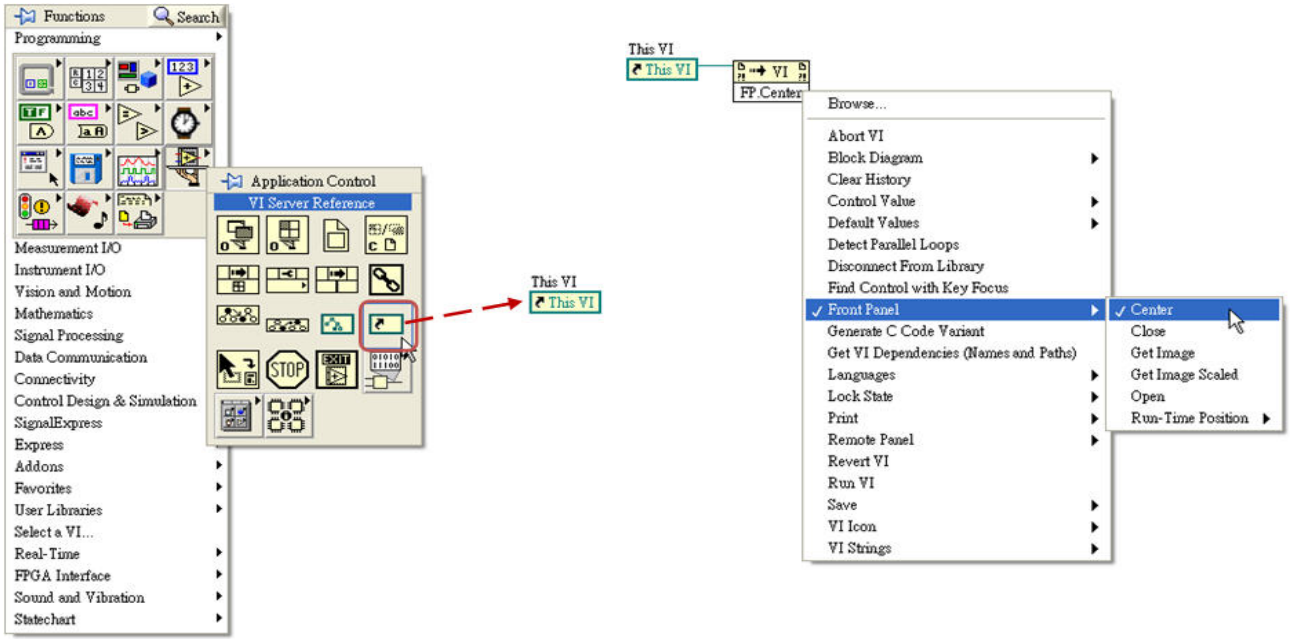
Invoke Node（方法節點）



作者；曾智勇,chihyungtseng@hotmail.com

Invoke Node（方法節點）

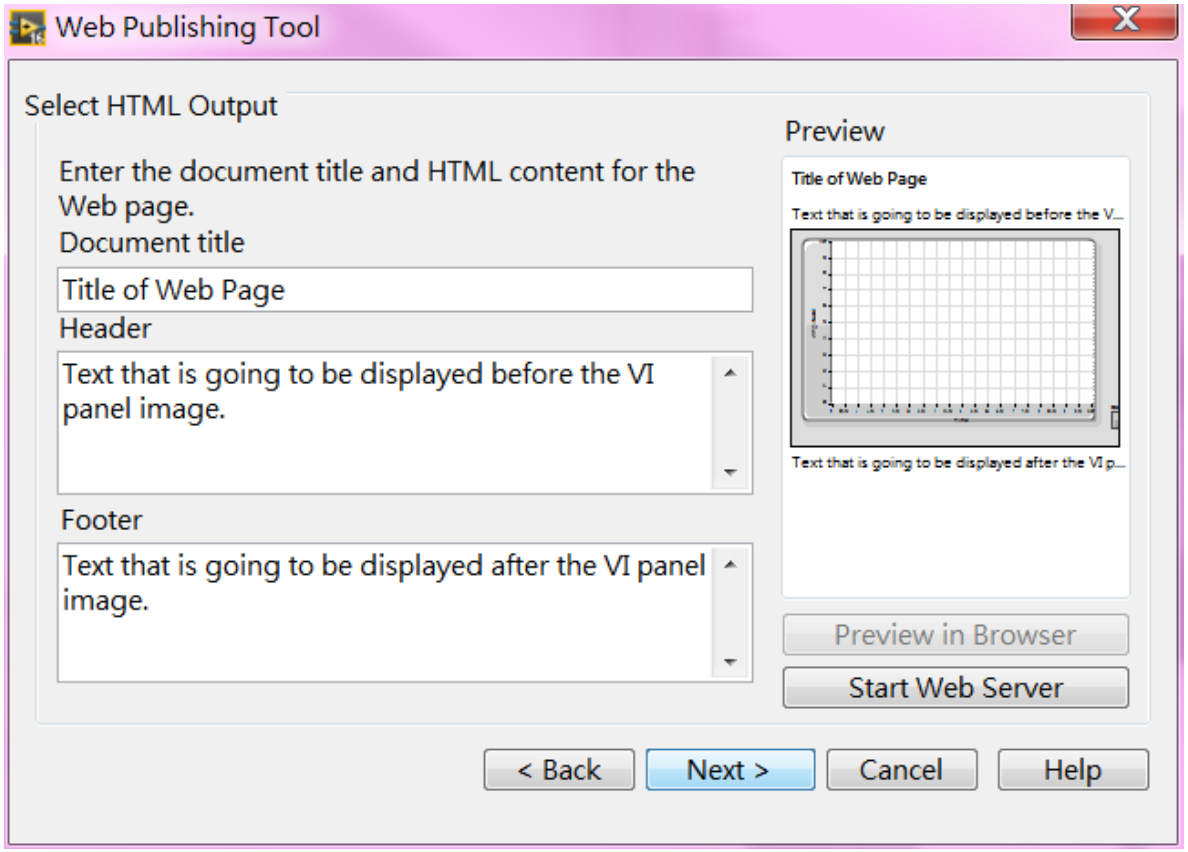
- 那如果今天要想控制Front Panel能夠置中，此時你要使用的就是VI Server Reference。然後再建立一個Invoke Node，因為讓視窗置中是屬於方法的一種，所以需要使用Invoke Node來設計，當連結好後，請選擇[Front Panel]>>[Center]，所以當程式執行到此方法節點時，視窗就會自動置中。



作者；曾智勇,chihyungtseng@hotmail.com

Web Server

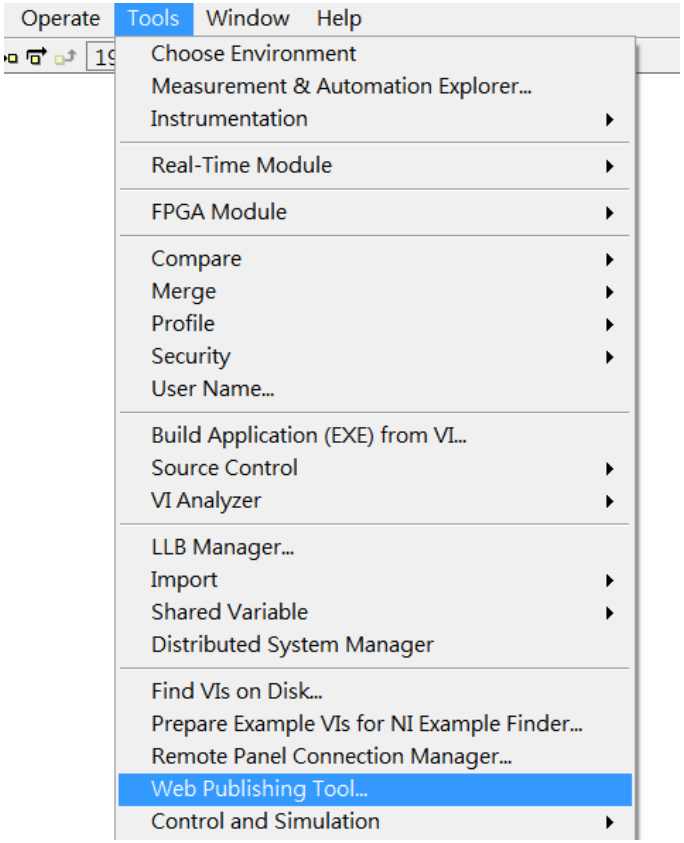
- Web Publishing Tool，可簡單透過網頁監控LabVIEW程式

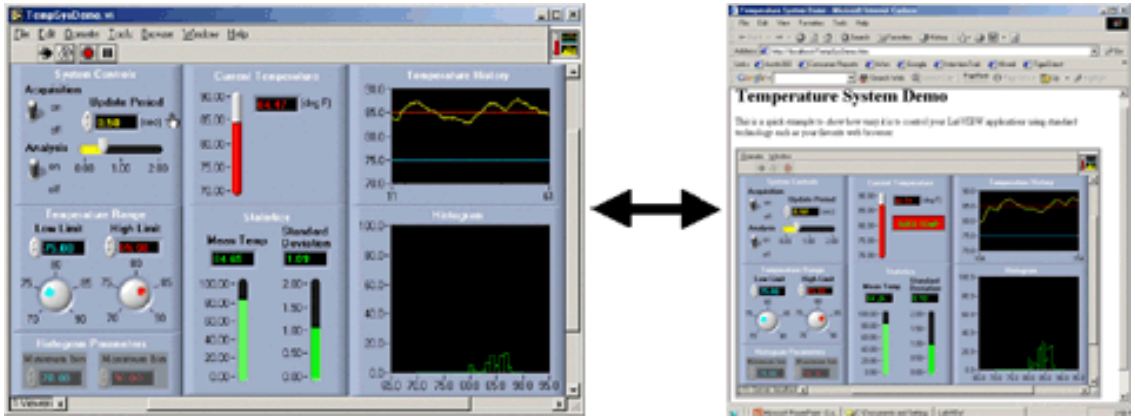


作者；曾智勇, chihyungtseng@hotmail.com

Web Server

- Web Publishing Tool，可簡單透過網頁監控LabVIEW程式

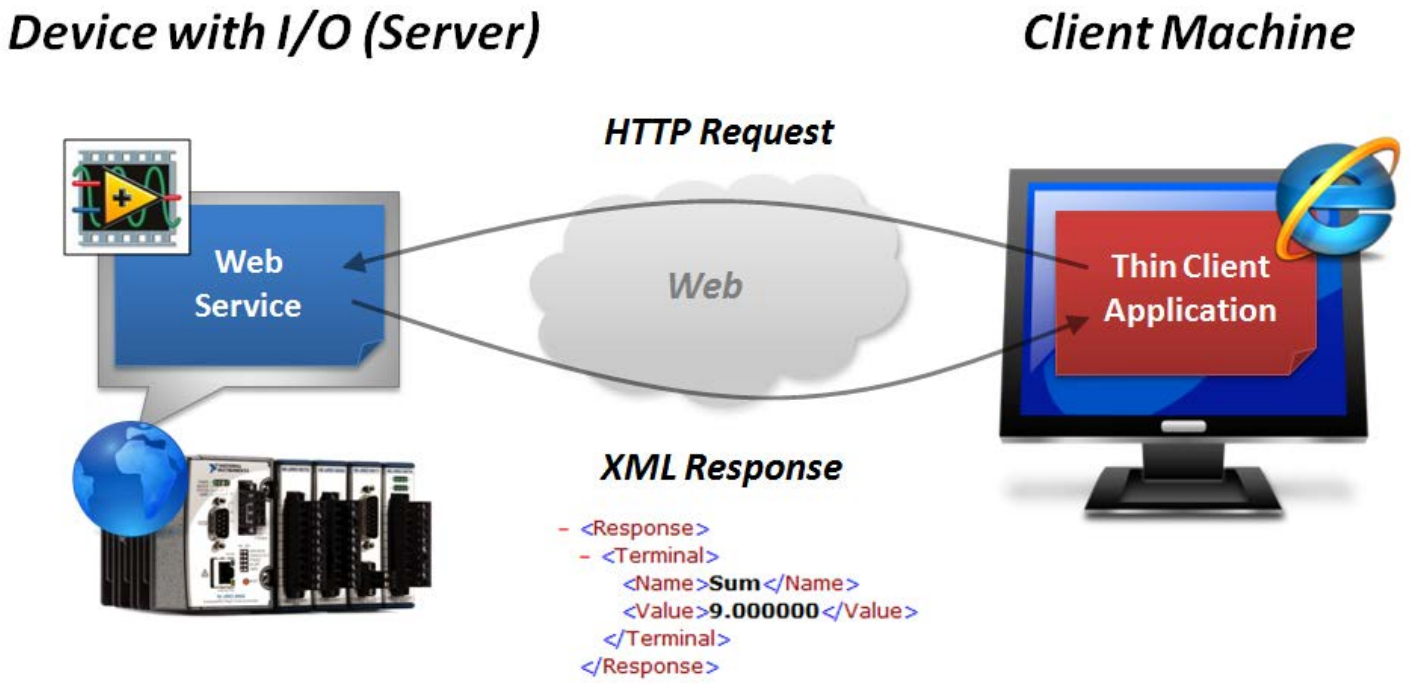




作者；曾智勇, chihyungtseng@hotmail.com

Web Server

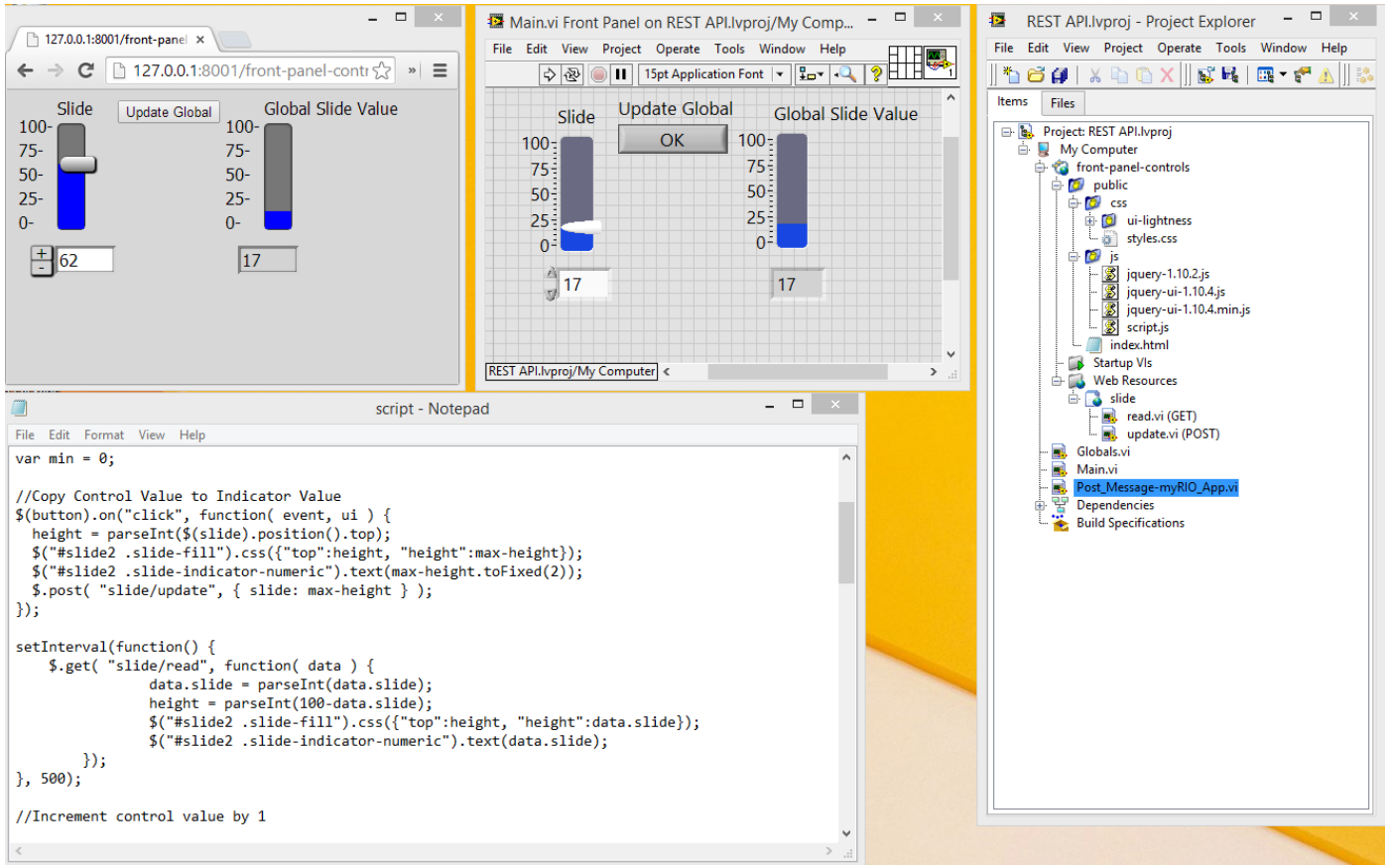
- Web Service，可簡單掛載伺服器程式，可開機後自動執行
- Web Publishing Tool=>　　電腦端需要啟動LabVIEW主程式
- Web Service =>　　電腦端不需要啟動程式，只需要開機



作者；曾智勇,chihyungtseng@hotmail.com

Web Server

- Web Service，可簡單掛載伺服器程式，可開機後自動執行



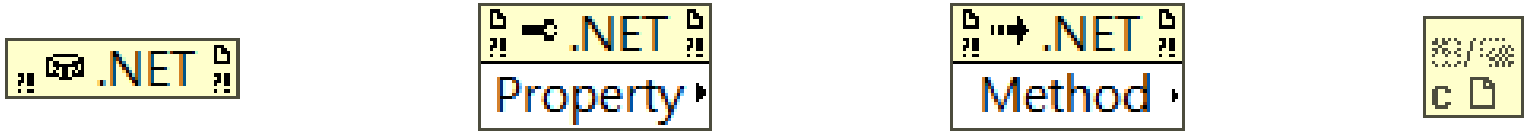
作者；曾智勇,chihyungtseng@hotmail.com

網路通訊

- TCP/IP

- ActiveX

- .NET

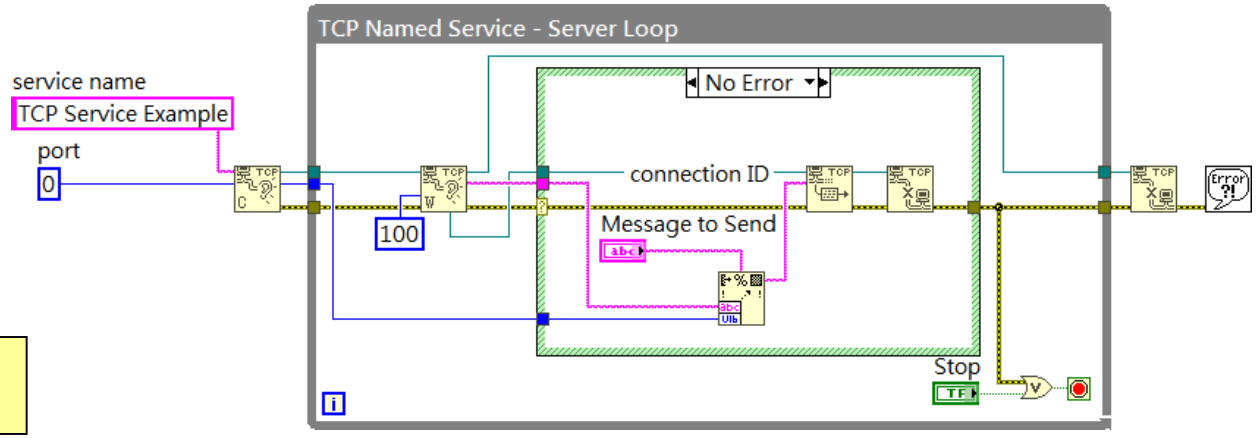


作者；曾智勇, chihyungtseng@hotmail.com

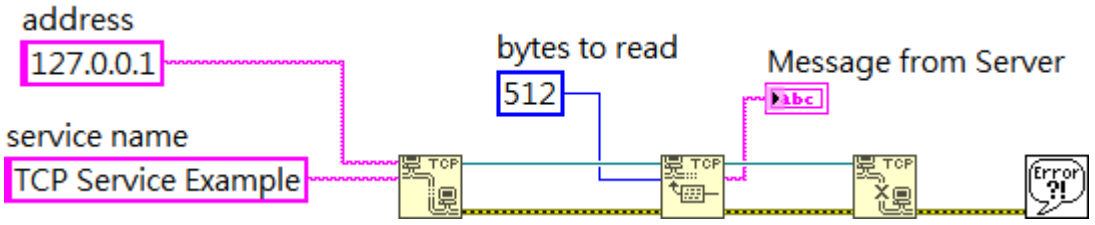
網路通訊

- TCP/IP

Server



Client



作者；曾智勇, chihyungtseng@hotmail.com

網路通訊

- ActiveX & .NET

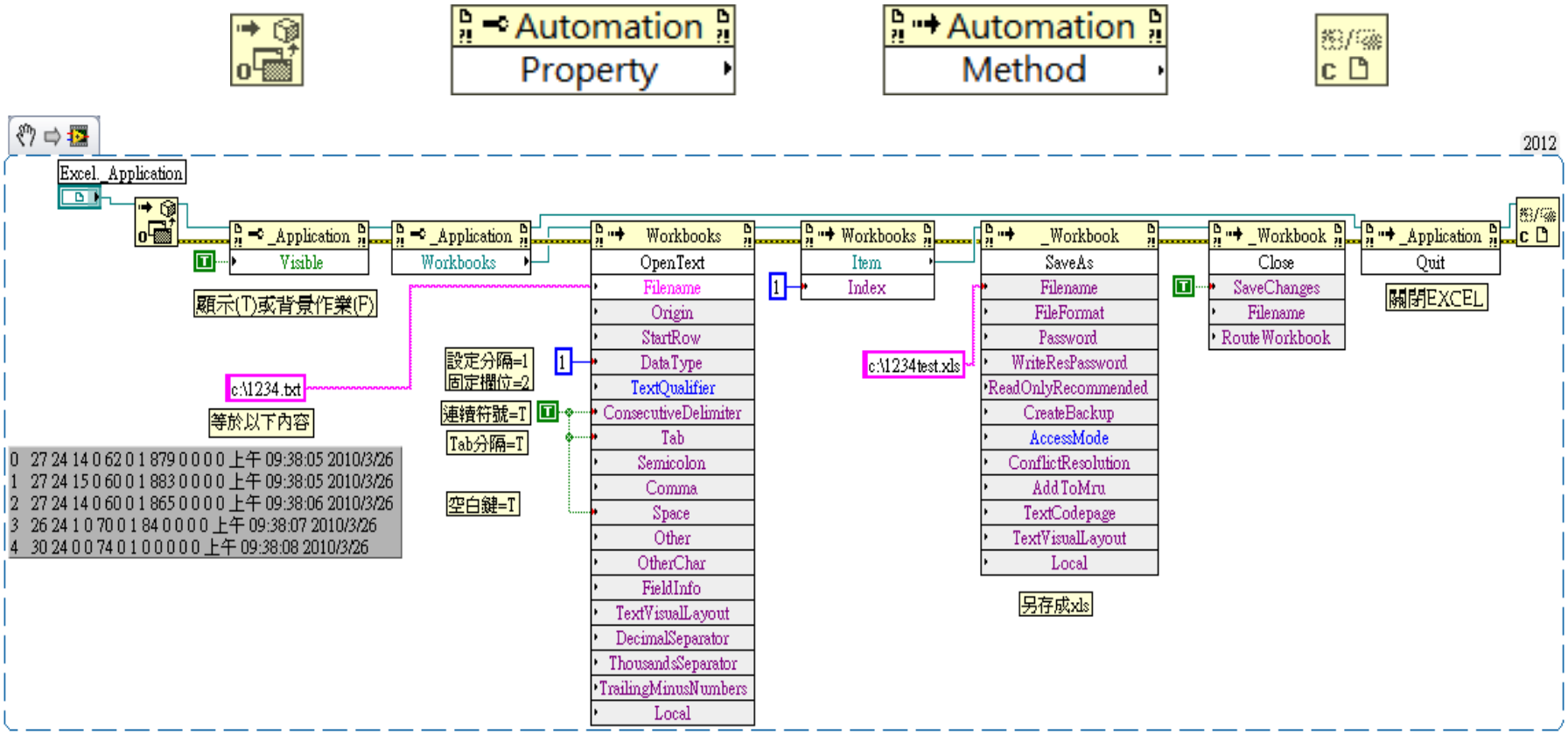
ActiveX是Microsoft在1996年推出的一個軟體應用程序開發框架。ActiveX框架通過組合和改編的早期COM（組件物件模型）與物件鏈接和嵌入技術（OLE）的組合和廣泛用於網站的下載內容。 雖然理論上， ActiveX是為了在所有平台上工作而創建的，但它主要用於Windows平台，並且由於編譯碼關係，它主要用於Intel x86架構。

Microsoft.NET是由Microsoft開發的高級軟體應用程序開發框架。 .NET定位是一種跨平台技術，從Web、桌面到移動裝置等。.NET支持多種程式語言（C#、F#、VB.NET、C++、Python等） ，可以多語言協作開發。除了Windows平台，Linux和OS X也可使用。

作者；曾智勇,chihyungtseng@hotmail.com

網路通訊

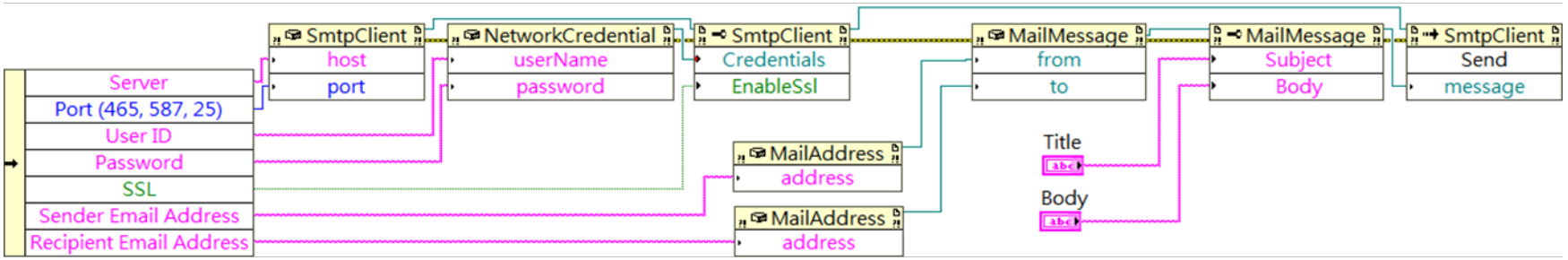
- ActiveX (Excel)



作者；曾智勇,chihyungtseng@hotmail.com

網路通訊

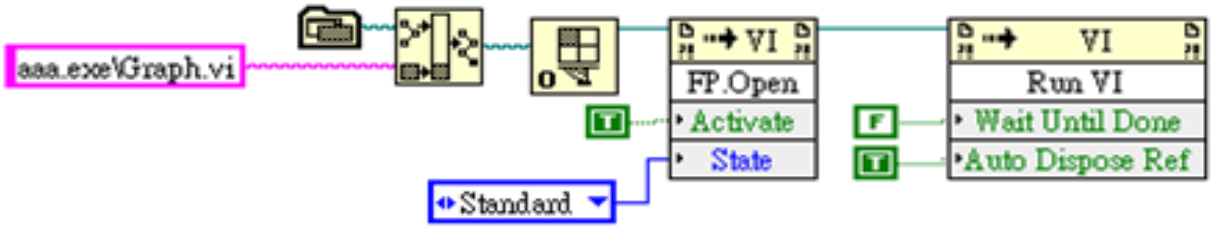
- .NET (Email)



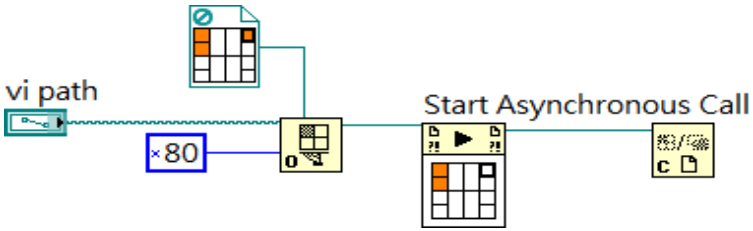
作者；曾智勇,chihyungtseng@hotmail.com

VI Server

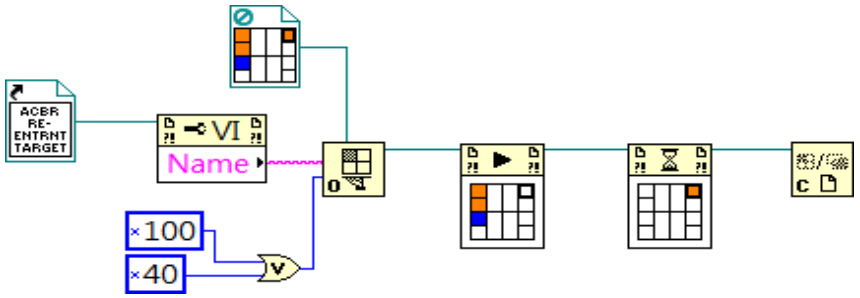
- 動態呼叫



無接腳，無回傳



有接腳，無回傳

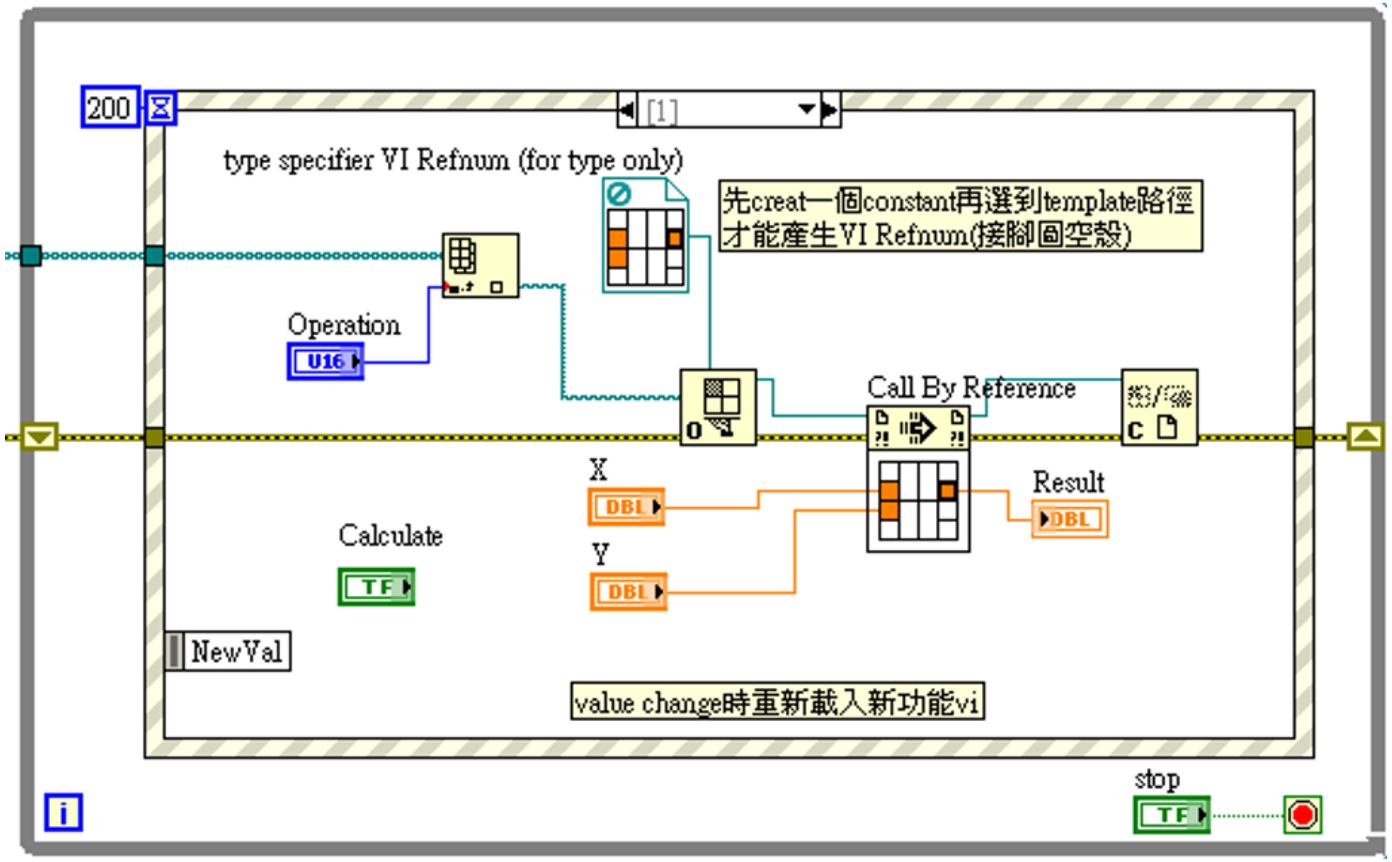


有接腳，有回傳

作者；曾智勇,chihyungtseng@hotmail.com

VI Server

- 動態呼叫

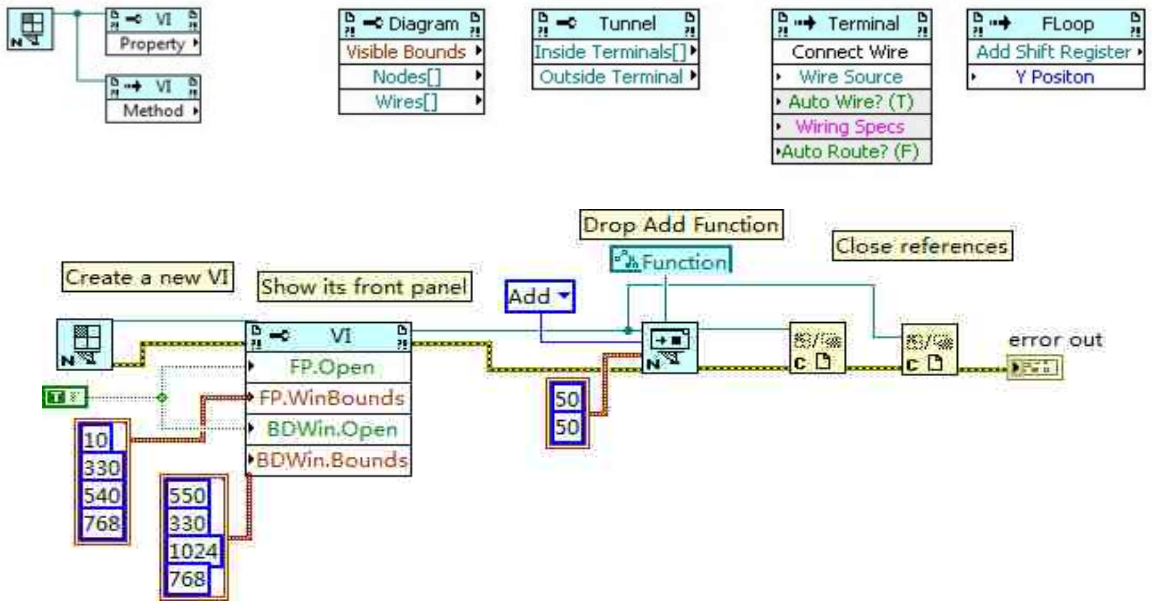


作者；曾智勇, chihyungtseng@hotmail.com

VI Scripting

- VI Scripting：可透過程式寫程式，一種LabVIEW 腳本，藉由程式設計、創建、修改、執行 LabVIEW程式碼。

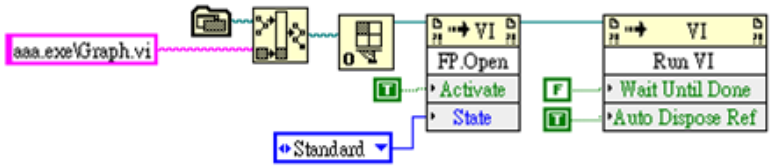
- 一般函式為淺黃色，VI Scripting函式為淺藍色，只有在開發階段可以使用，佈署成執行檔階段則無法使用。

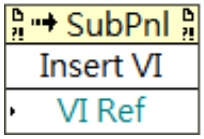


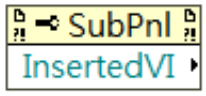
作者；曾智勇,chihyungtseng@hotmail.com

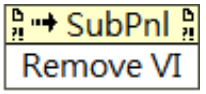
Subpanel

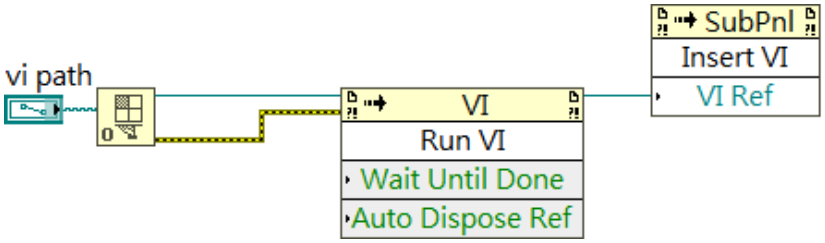
- 動態呼叫+Subpanel

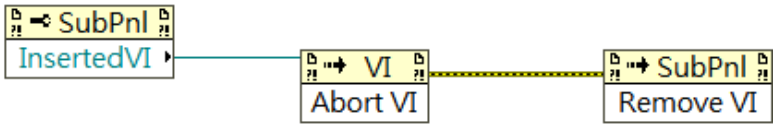












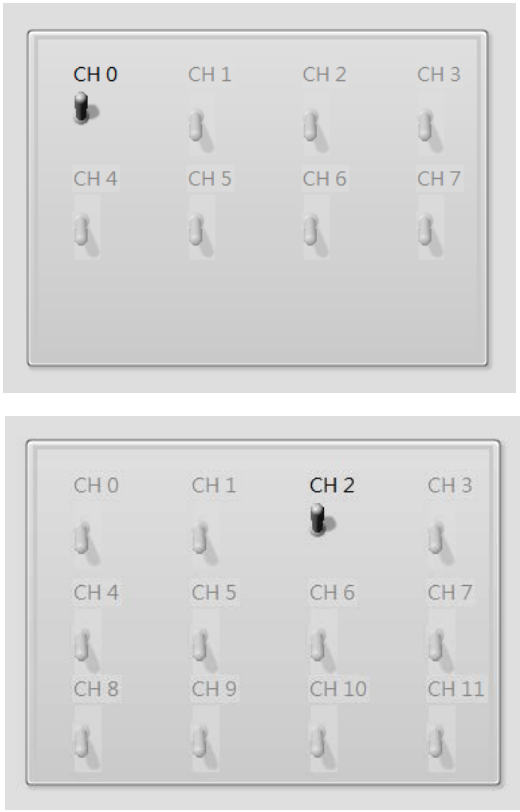
載入Subpanel

卸除Subpanel

作者；曾智勇, chihyungtseng@hotmail.com

程式效能

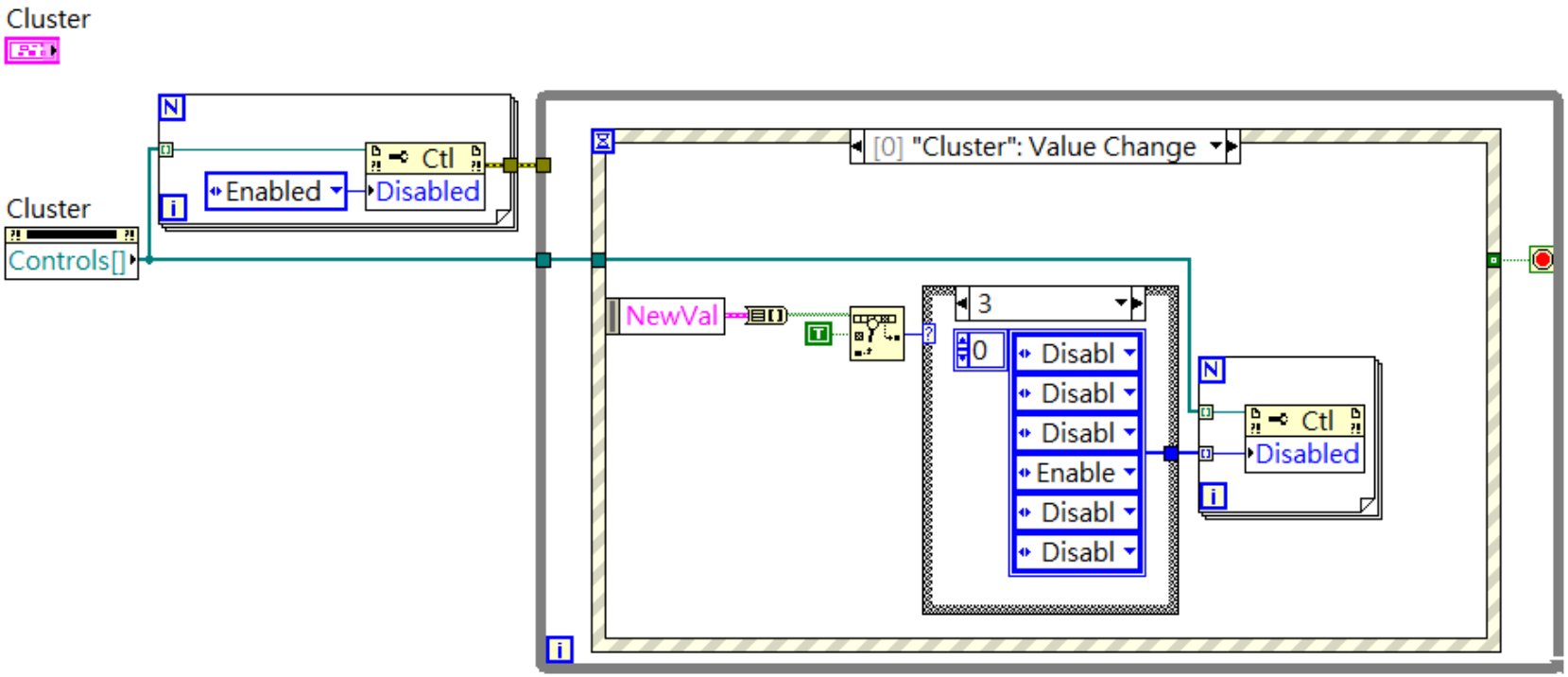
- 目標：具有防呆機制的開關組
- 需求：取消原本選擇的之後, 才能進入下一步



作者；曾智勇, chihyungtseng@hotmail.com

程式效能

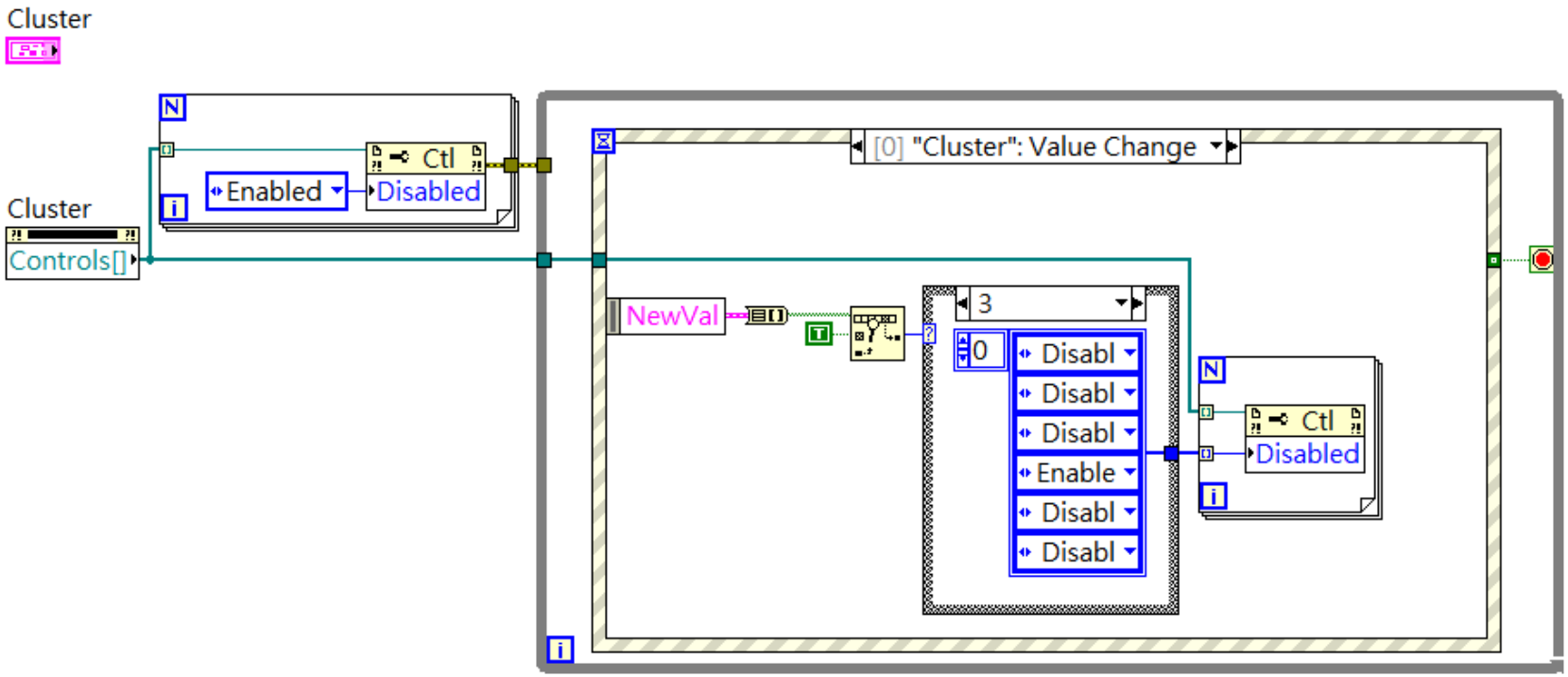
- 目標：具有防呆機制的開關組
- 需求：取消原本選擇的之後, 才能進入下一步



作者；曾智勇, chihyungtseng@hotmail.com

程式效能

解（1/5）：

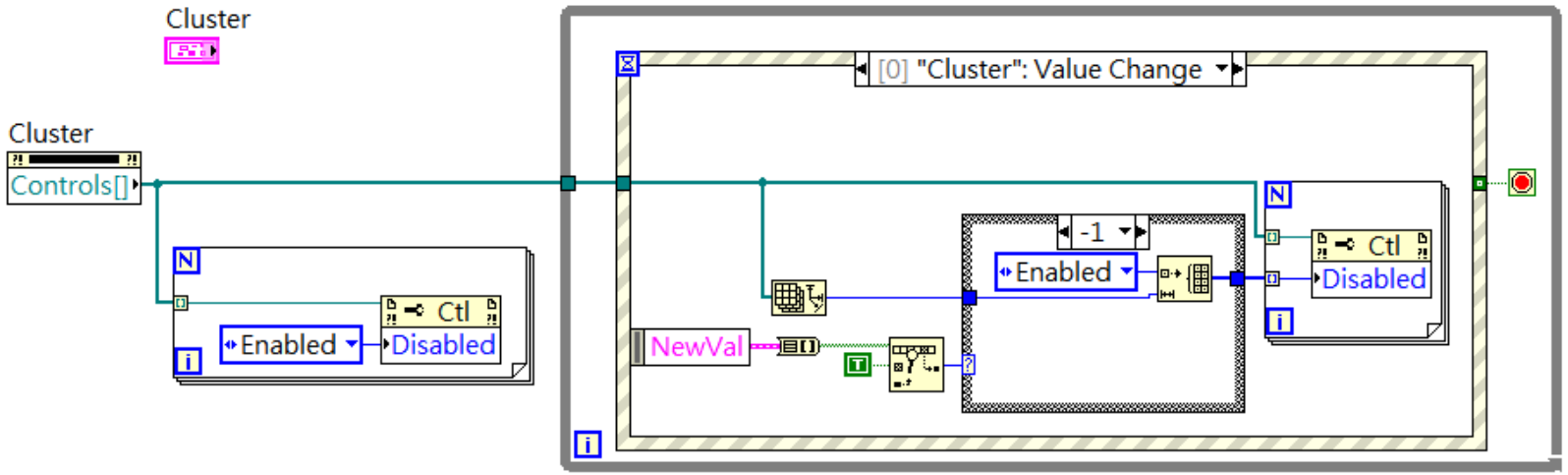


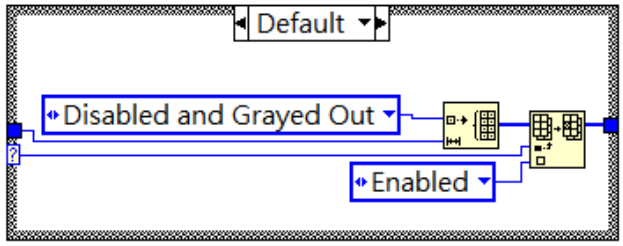
優點：淺顯易懂

作者；曾智勇, chihyungtseng@hotmail.com

程式效能

解（2/5）：



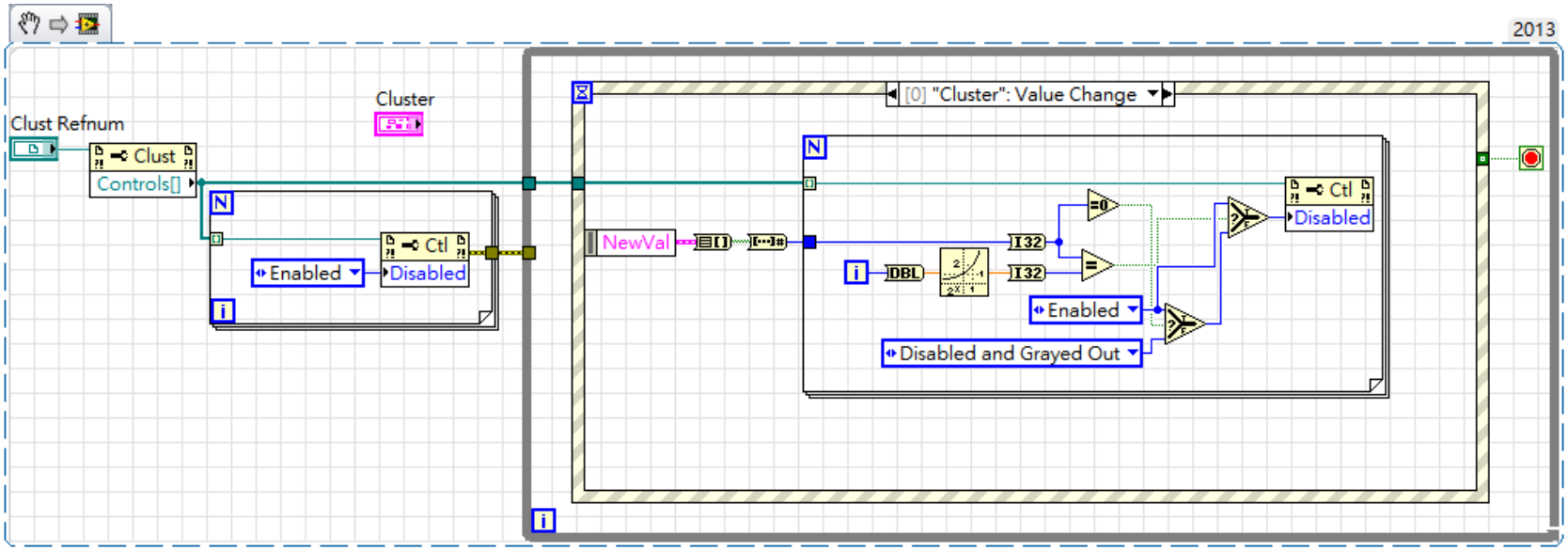


優點：多個case→2個case

作者；曾智勇,chihyungtseng@hotmail.com

程式效能

解（3/5）：



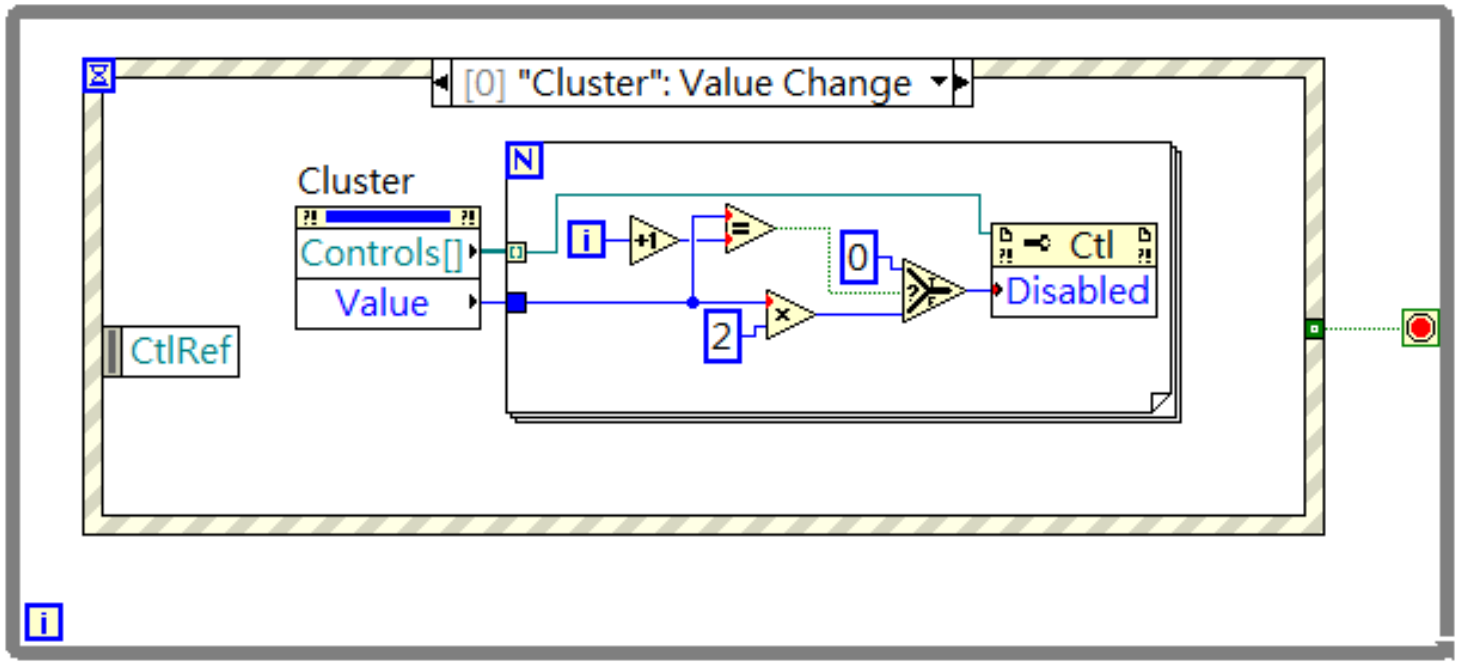
優點：2個case→2個select

Boolean Array To Number

Power Of 2

程式效能

解（4/5）：



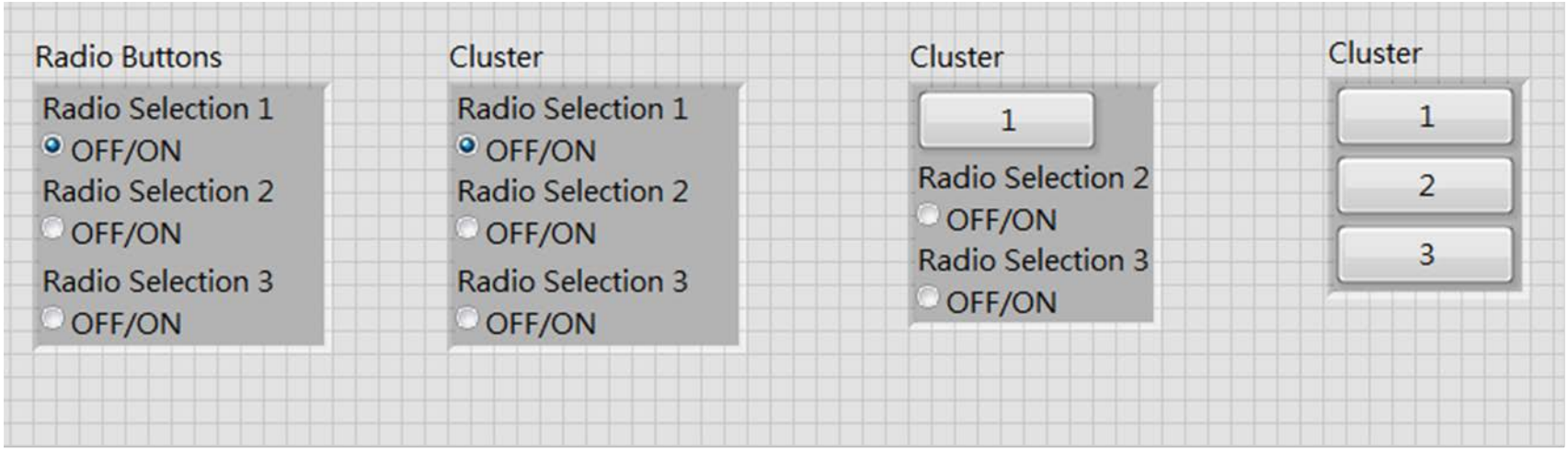
優點：簡潔，但不易懂

(提示：它是假的Cluster)

作者；曾智勇, chihyungtseng@hotmail.com

程式效能

解（4/5）：

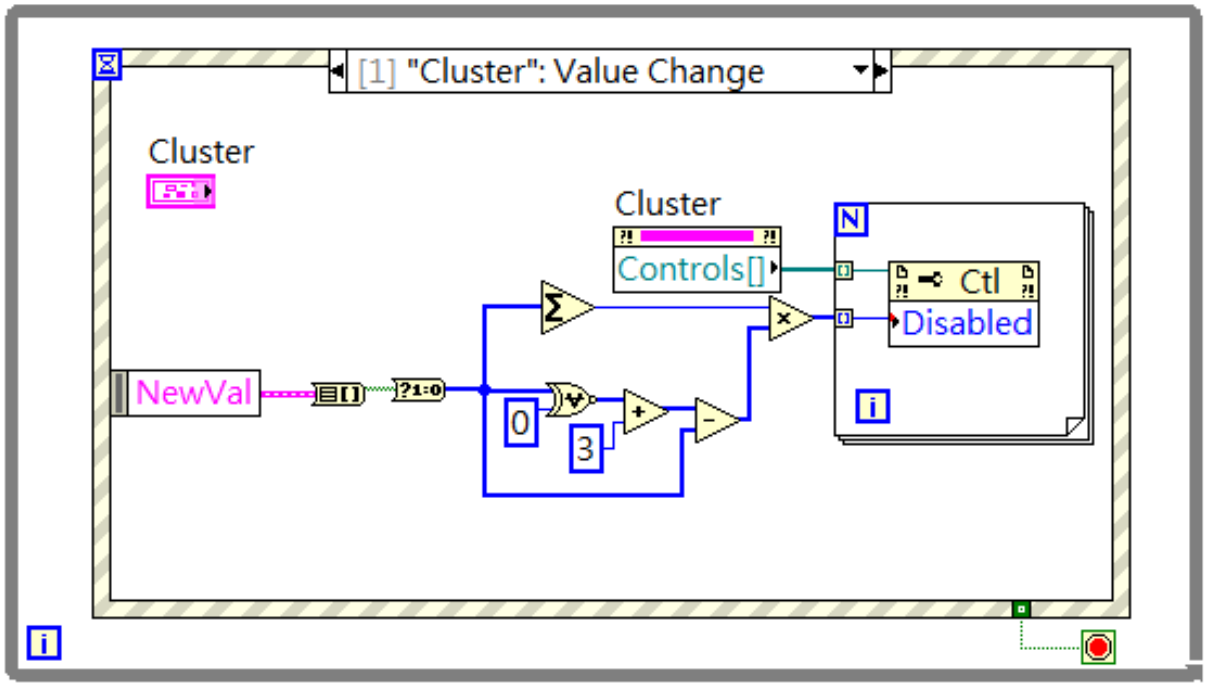


Create → Rename → Replace → Allow No Selection

作者；曾智勇, chihyungtseng@hotmail.com

程式效能

解（5/5）：先思考“結果背後的結果” 再開始動作



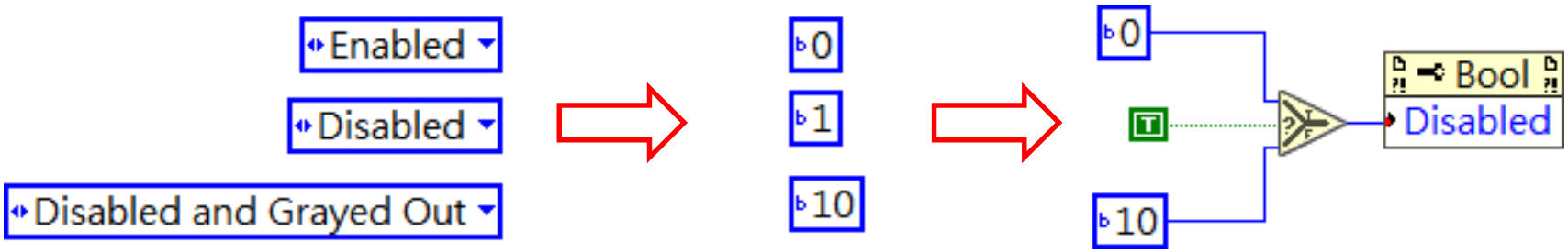
優點：簡潔到難以理解

(提示： 2進制補數，將正值進行位元反相，再加1)

作者；曾智勇,chihyungtseng@hotmail.com

程式效能

解（5/5）：先思考“結果背後的結果”再開始動作



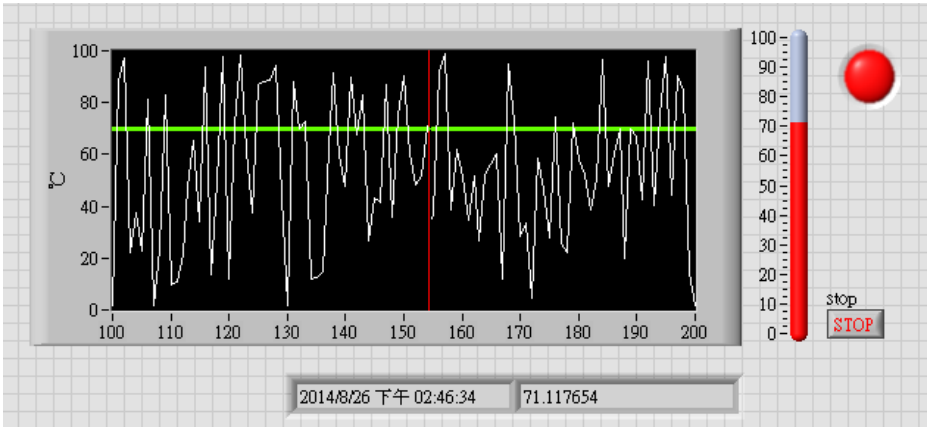
想的結果

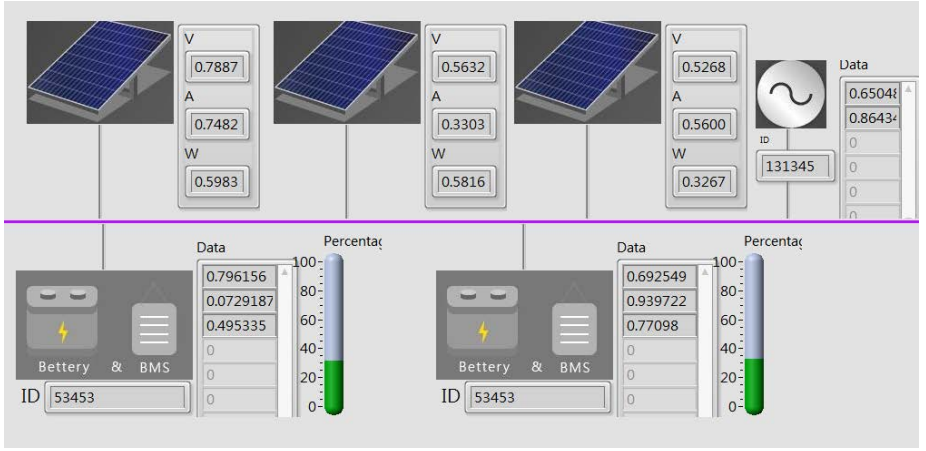
結果背後的結果

作者；曾智勇,chihyungtseng@hotmail.com

XControl

- User Control：
  - 單一物件
  - 資料+外觀
  - 開發容易
  - 功能單一
- XControl：
  - 可封裝多個物件
  - 資料+外觀+功能/行為
  - 開發繁雜
  - 功能多樣
  - 可將其看做 **迷你版程式**

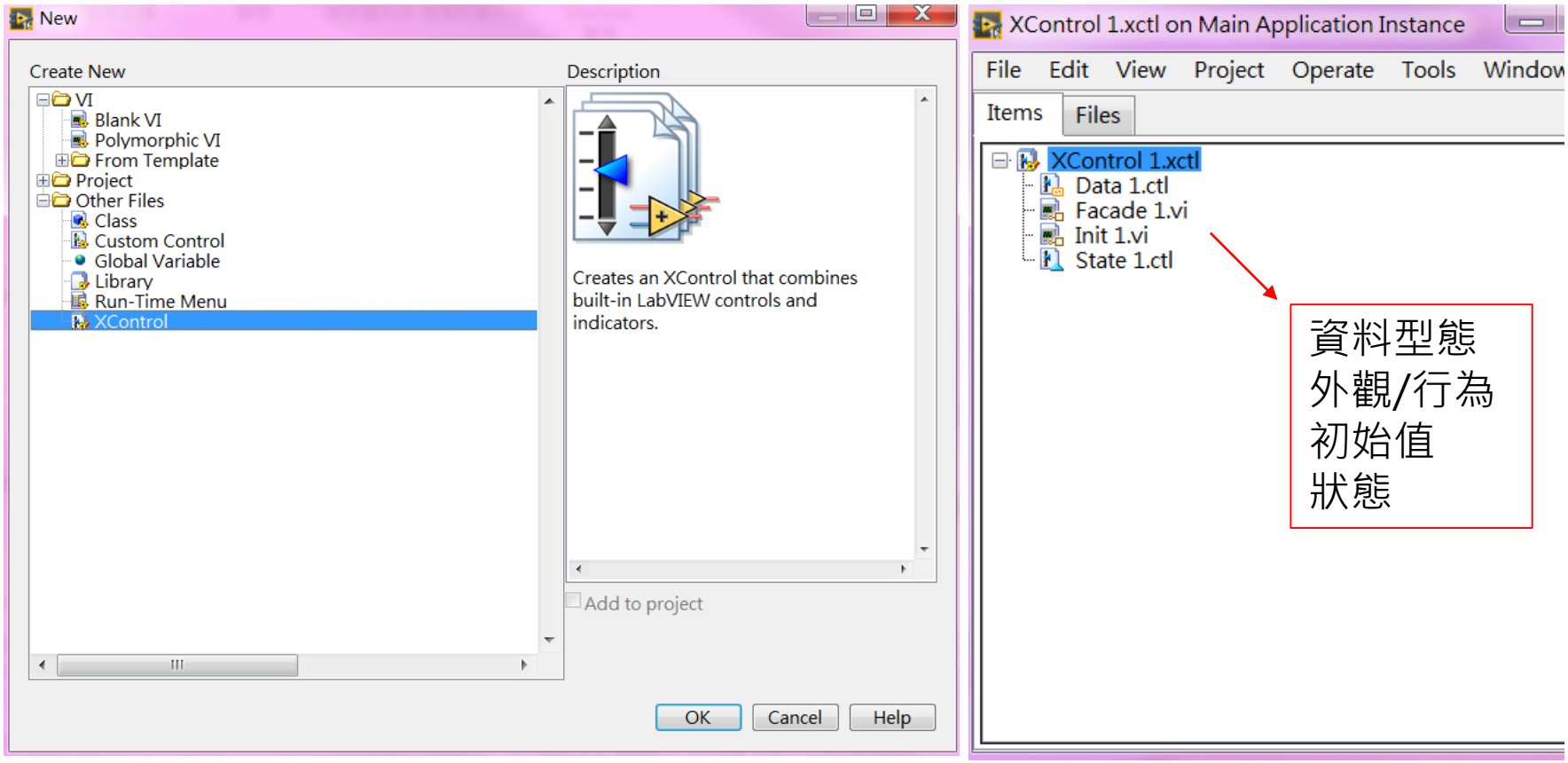




作者；曾智勇,chihyungtseng@hotmail.com

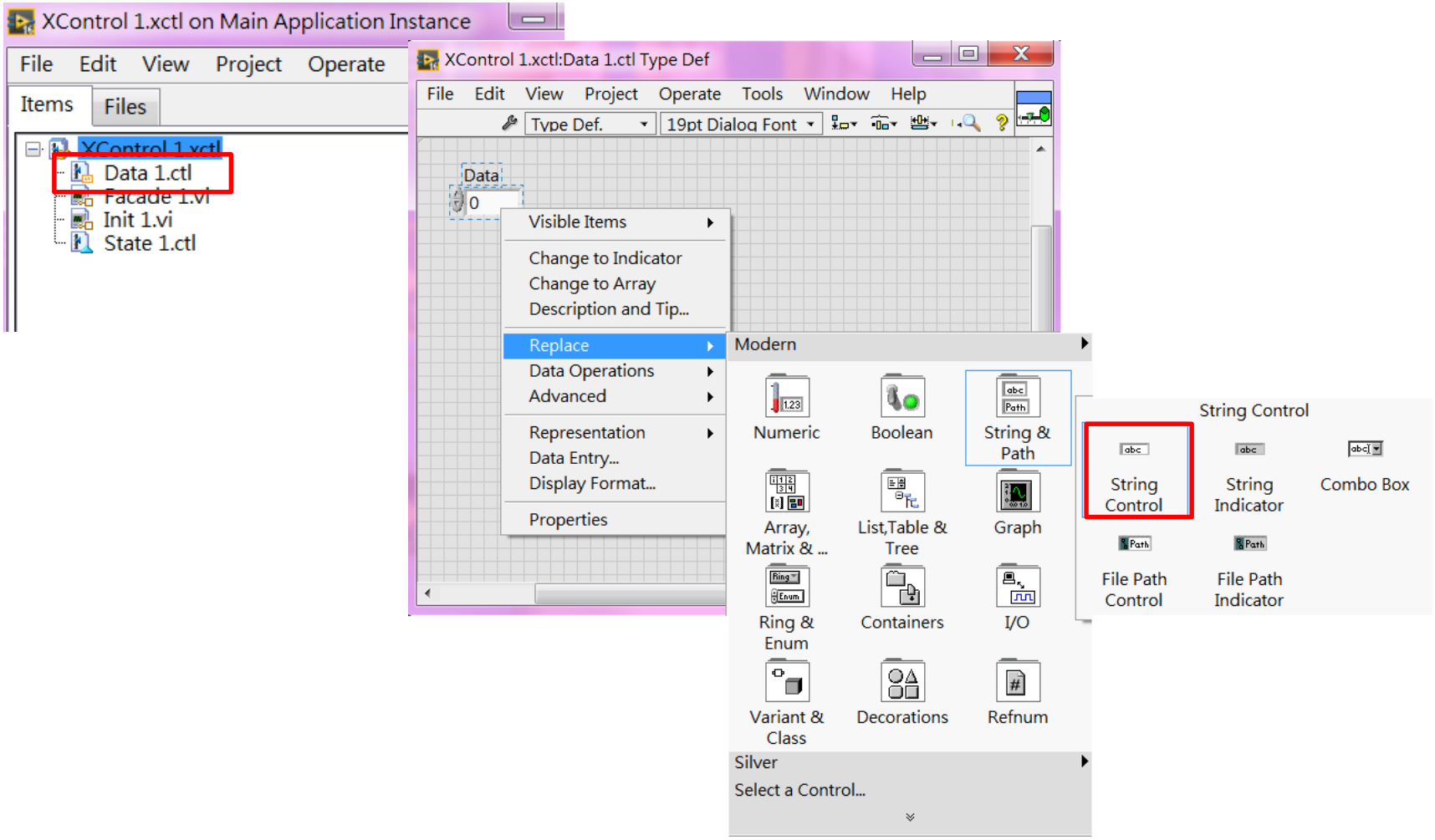
XControl

- 輸入文字顯示數字，開發流程：



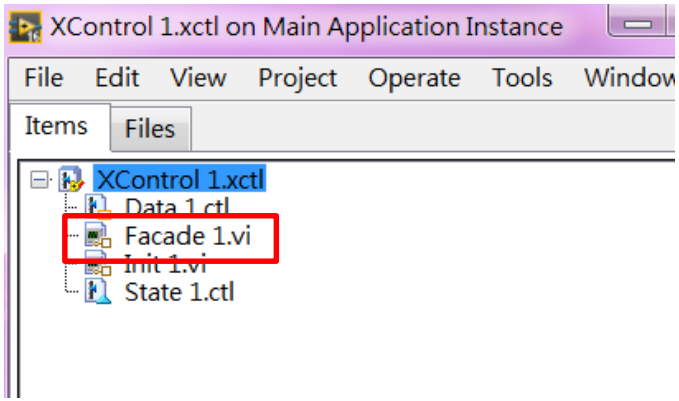
作者；曾智勇, chihyungtseng@hotmail.com

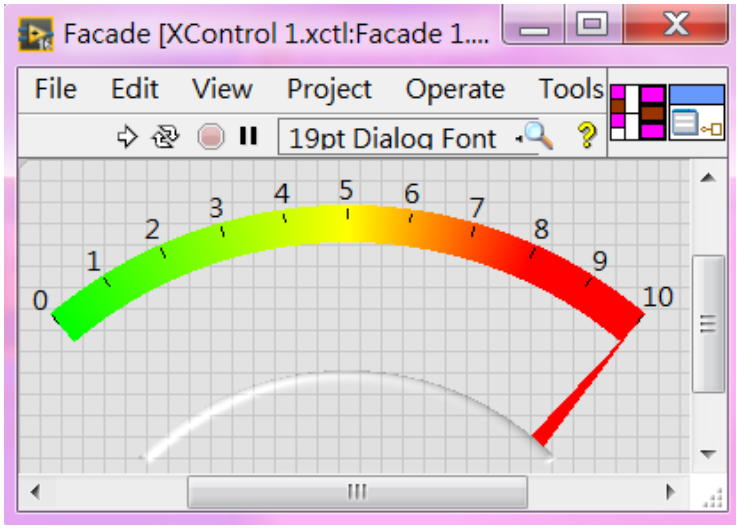
XControl

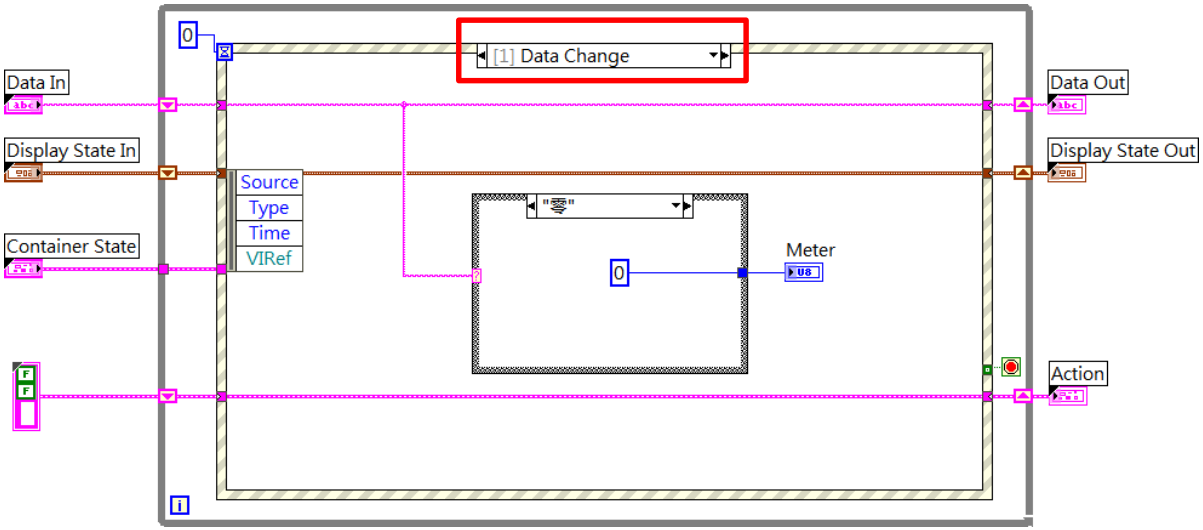


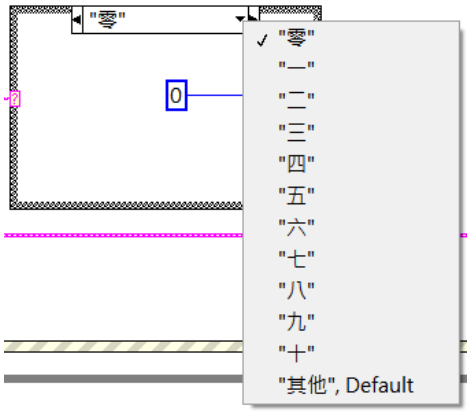
作者；曾智勇,chihyungtseng@hotmail.com

XControl





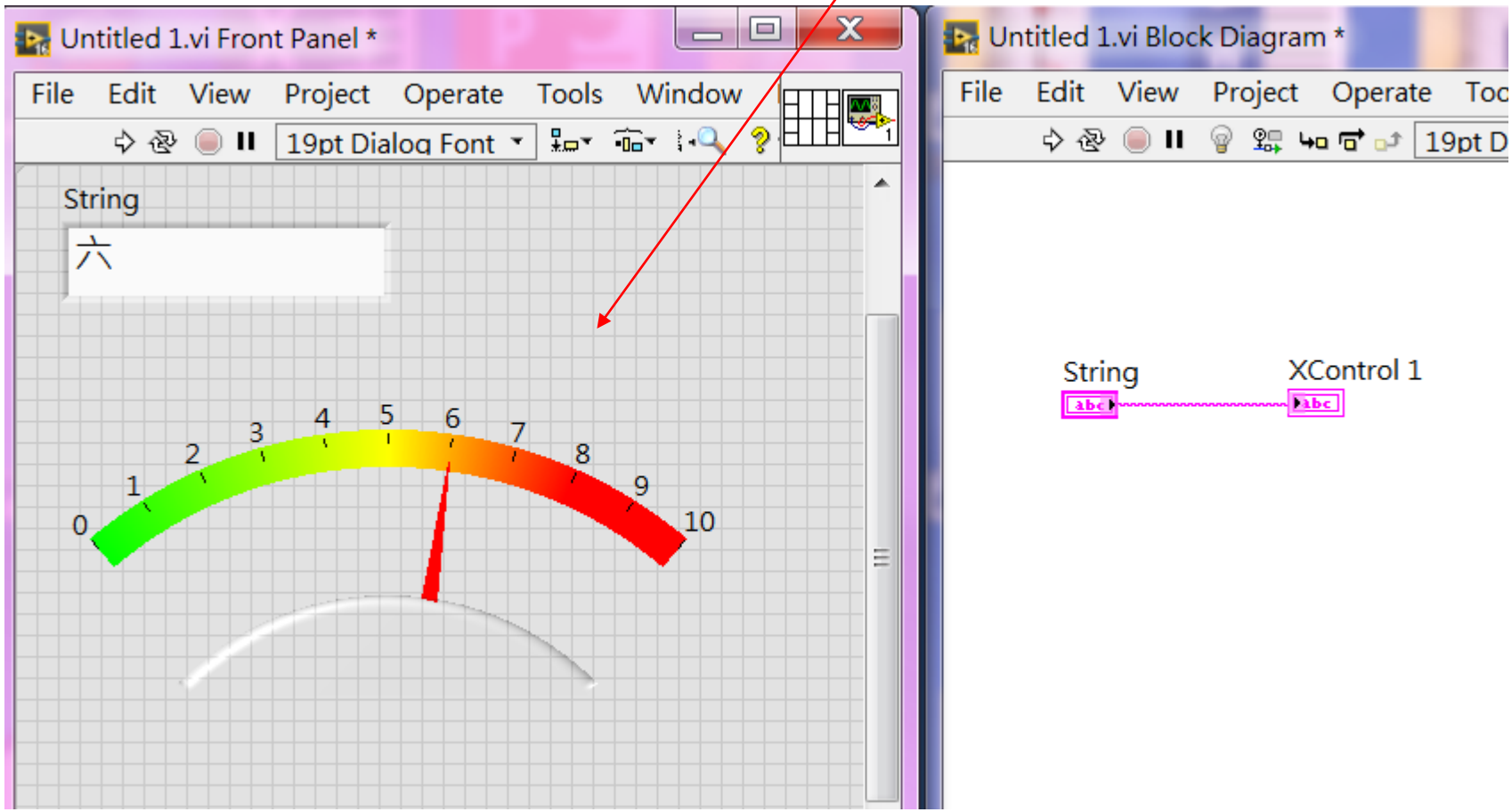




作者；曾智勇, chihyungtseng@hotmail.com

XControl

將XControl拖曳到前面板即可使用

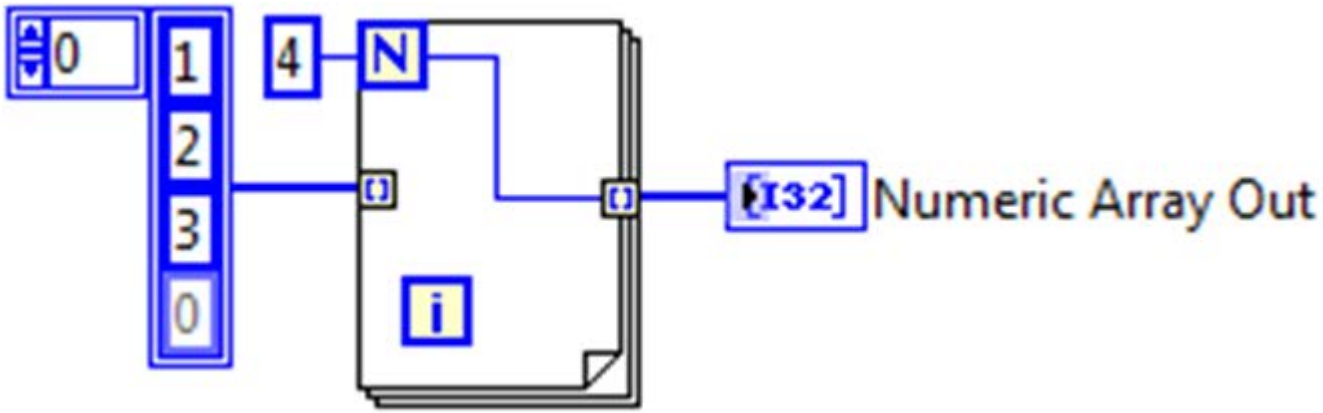


作者；曾智勇,chihyungtseng@hotmail.com

熱身練習

作者；曾智勇, chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



a) {0,1,2}

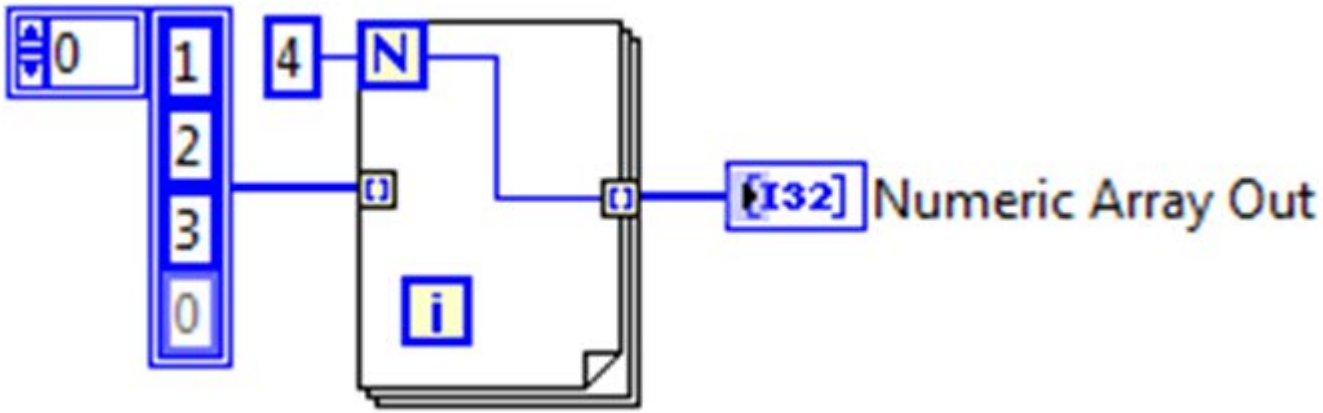
b) {0,1,2,3}

c) {3,3,3}

d) {4,4,4}

作者；曾智勇, chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



a) {0,1,2}

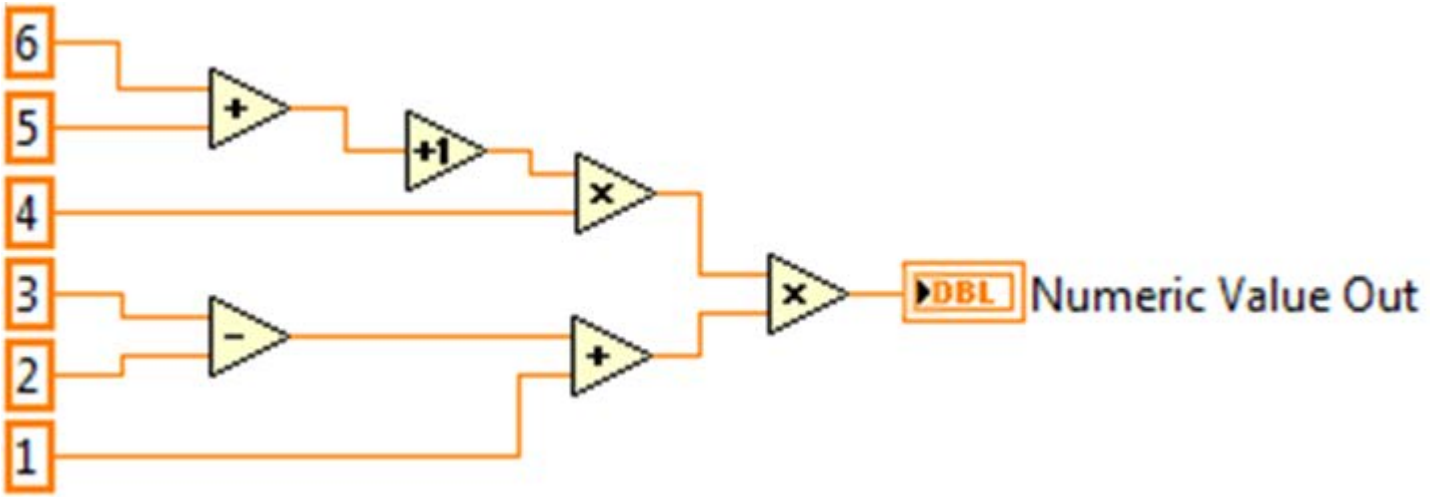
b) {0,1,2,3}

c) {3,3,3}

d) {4,4,4}

作者；曾智勇, chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



a) 94

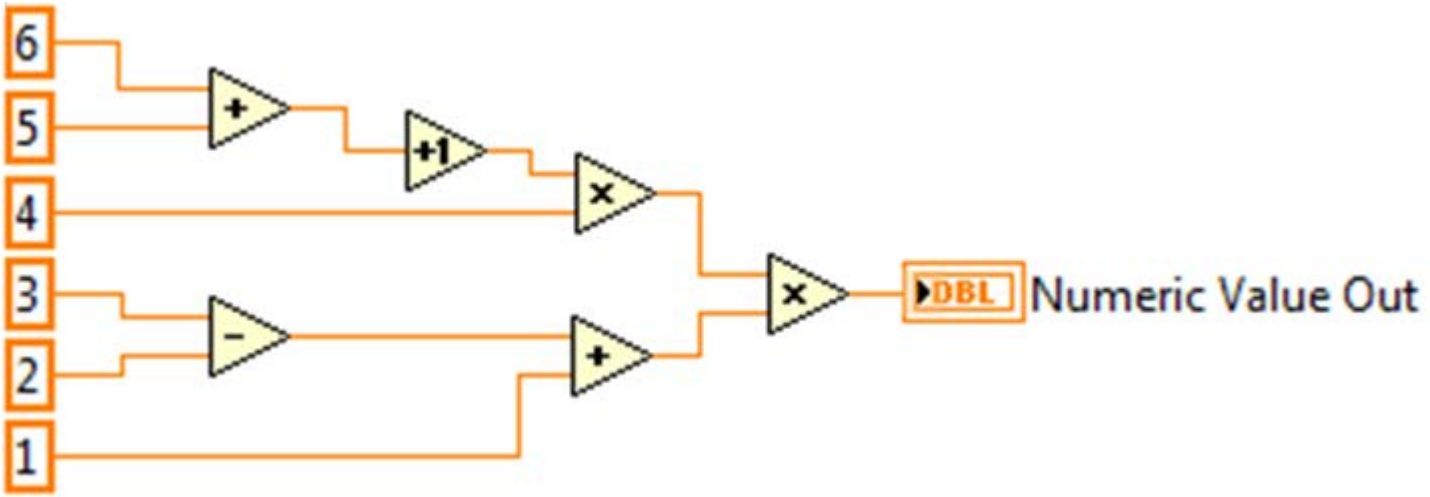
b) 95

c) 96

d) 97

作者；曾智勇, chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



a) 94

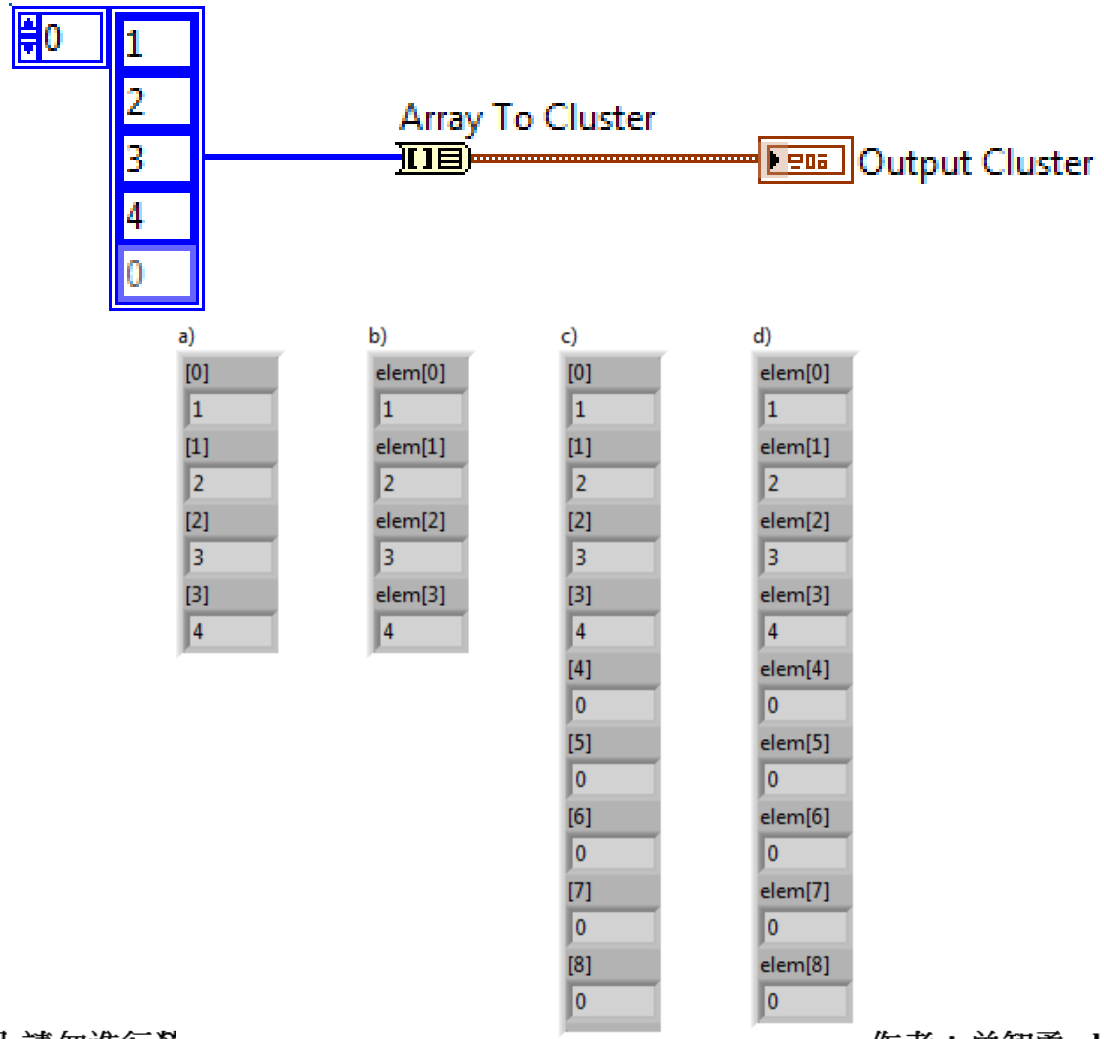
b) 95

c) 96

d) 97

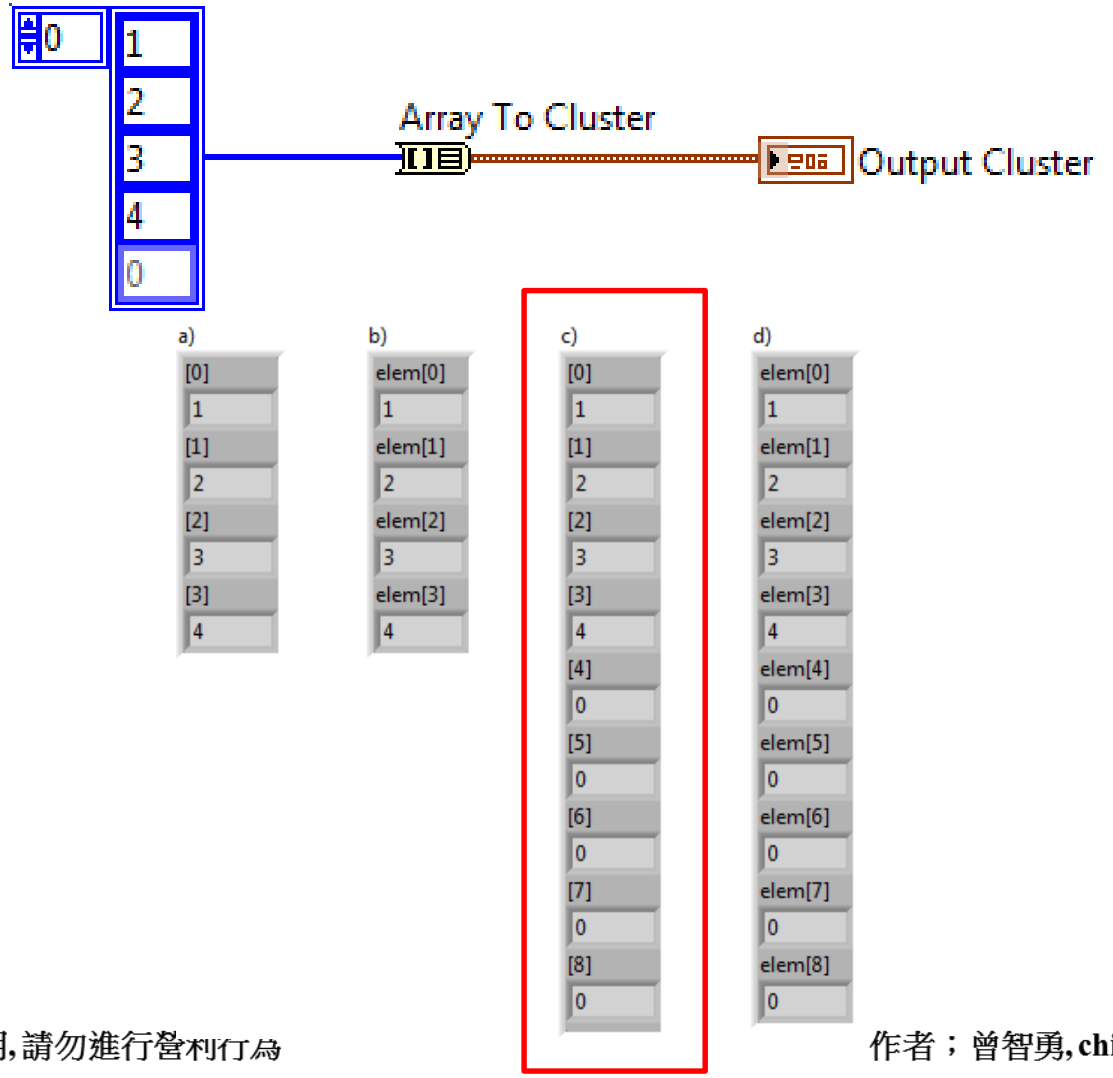
作者；曾智勇, chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



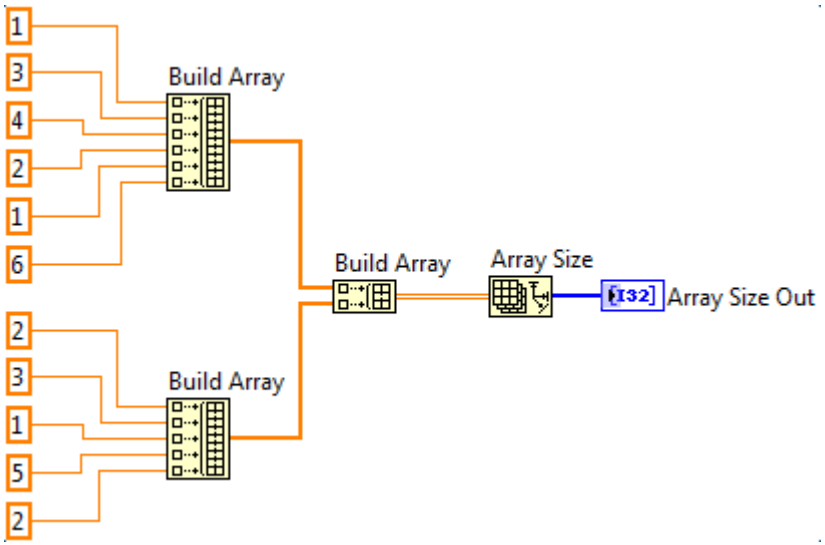
作者；曾智勇, chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



作者；曾智勇, chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



a) 1D Array 包含 {2,5}

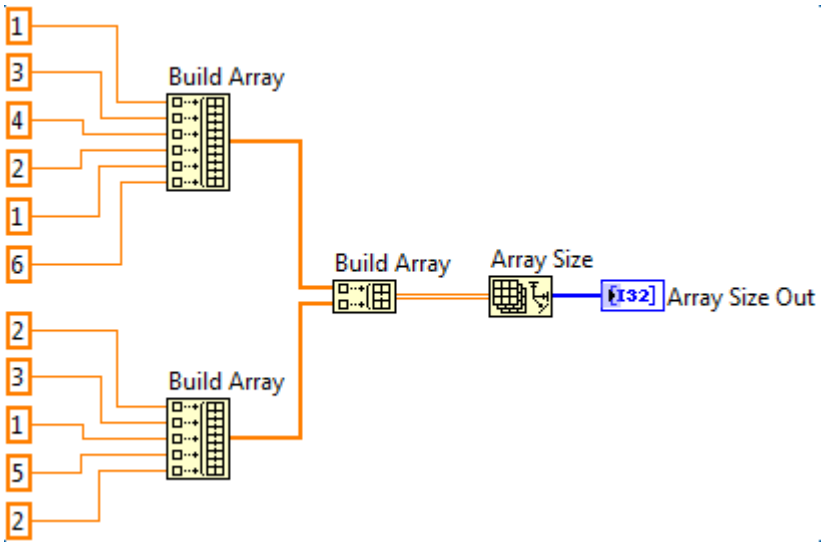
b) 1D Array 包含 {2,6}

c) 1D Array 包含 {5,5}

d) 1D Array 包含 {5,6}

作者；曾智勇, chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



a) 1D Array 包含 {2,5}

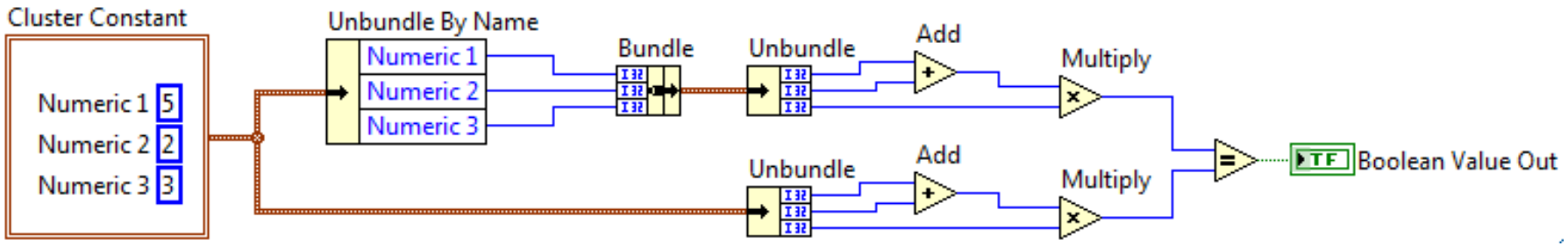
b) 1D Array 包含 {2,6}

c) 1D Array 包含 {5,5}

d) 1D Array 包含 {5,6}

作者；曾智勇,chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



a) TRUE

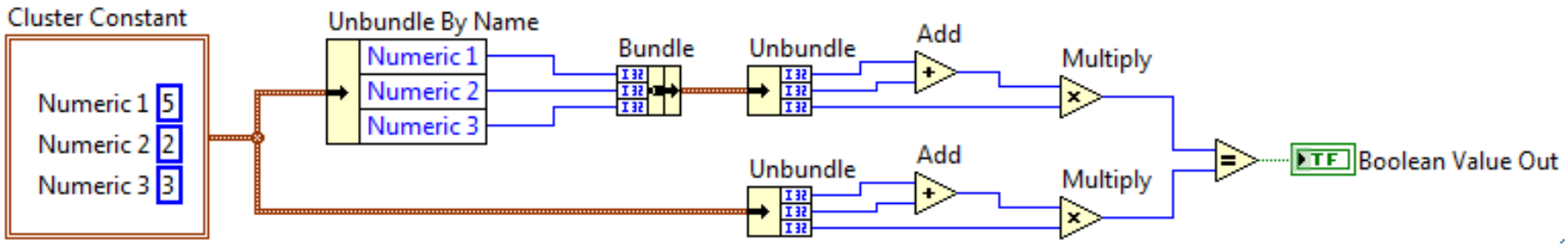
b) FALSE

c) 可能 TRUE 或 FALSE

d) TRUE 或 FALSE 皆錯誤

作者；曾智勇, chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



a) TRUE

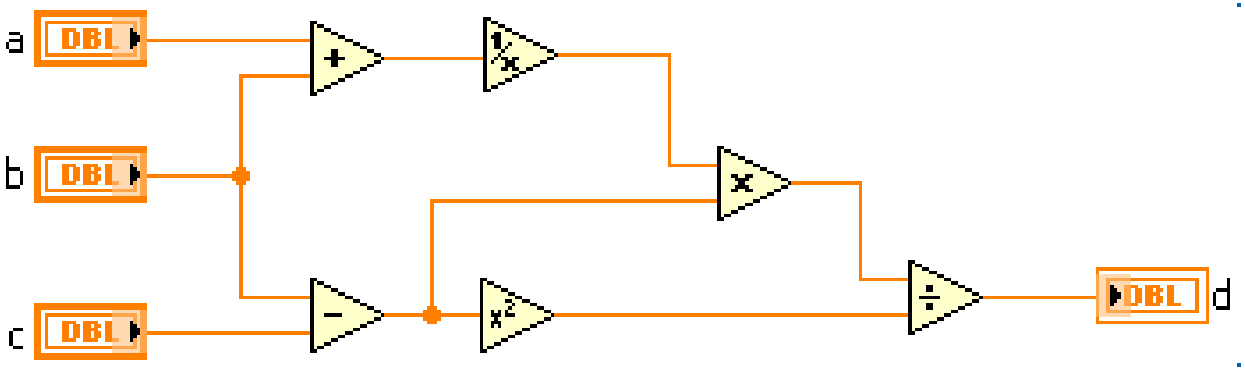
b) FALSE

c) 可能 TRUE 或 FALSE

d) TRUE 或 FALSE 皆錯誤

作者；曾智勇, chihyungtseng@hotmail.com

- 以下程式碼等效於合者？



a) $d = (a+b)^{-1} \times ((b-c) \div (b-c))^2$

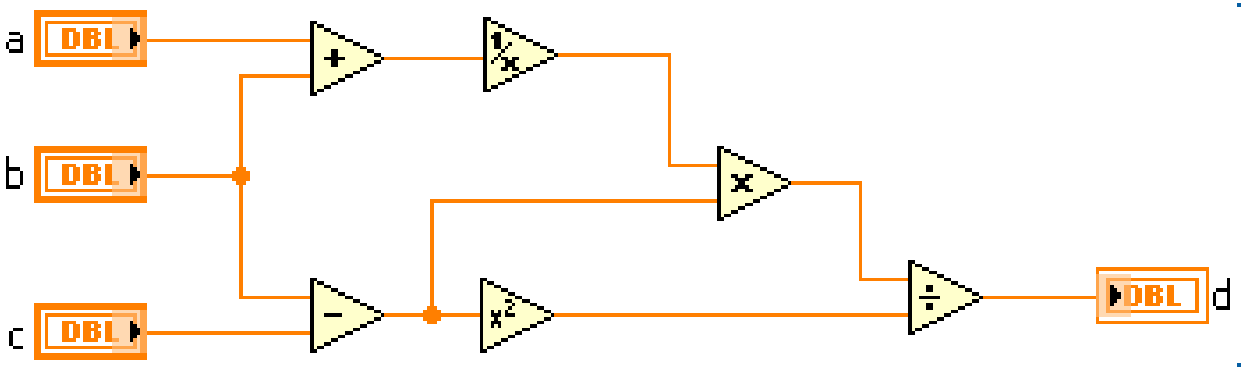
b) $d = ((a+b)^{-1} \times (b-c)) \div (b-c)^2$

c) $d = ((a+b)^{-1} \times (b-c)) \div (b-c)$

d) $d = (b-c) \div ((a+b)^{-1} \times (b-c))^2$

作者；曾智勇, chihyungtseng@hotmail.com

- 以下程式碼等效於合者？



a) $d = (a+b)^{-1} \times ((b-c) \div (b-c))^2$

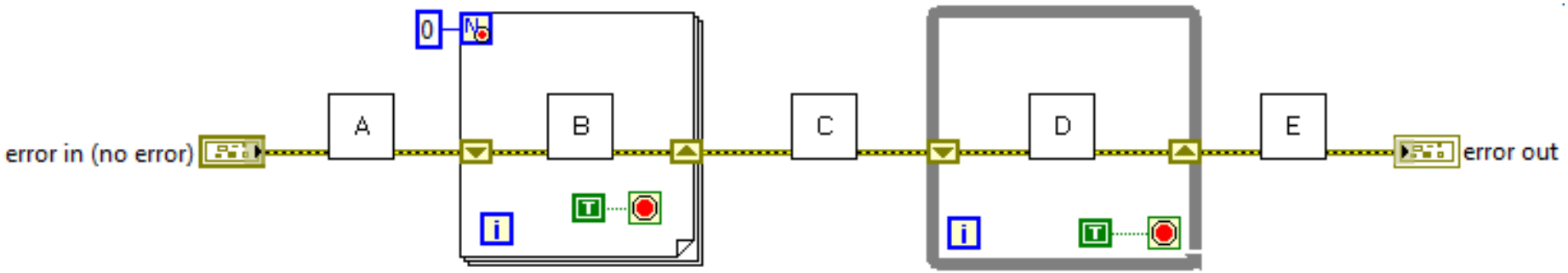
b) $d = ((a+b)^{-1} \times (b-c)) \div (b-c)^2$

c) $d = ((a+b)^{-1} \times (b-c)) \div (b-c)$

d) $d = (b-c) \div ((a+b)^{-1} \times (b-c))^2$

作者；曾智勇, chihyungtseng@hotmail.com

- 執行順序為何？



a) A，B，C，D，接著 E。

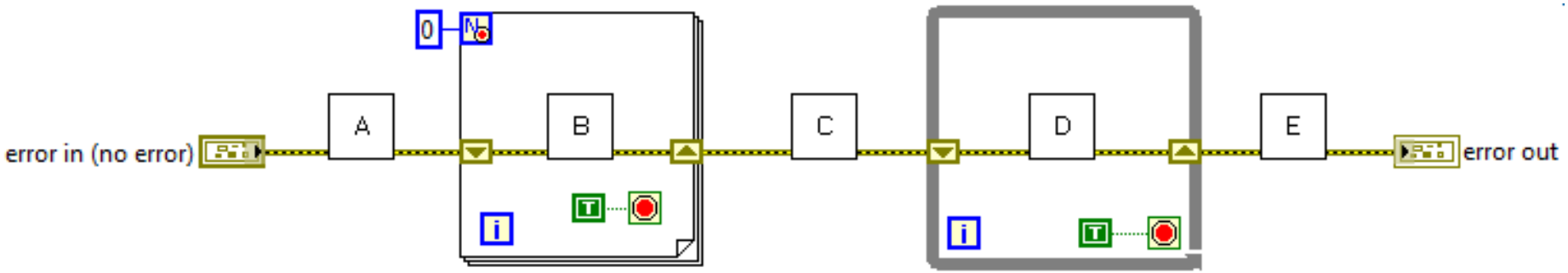
b) A，C，接著 E。

c) A，C，D，接著 E。

d) A，B，C，接著 E。

作者；曾智勇,chihyungtseng@hotmail.com

- 執行順序為何？



a) A，B，C，D，接著 E。

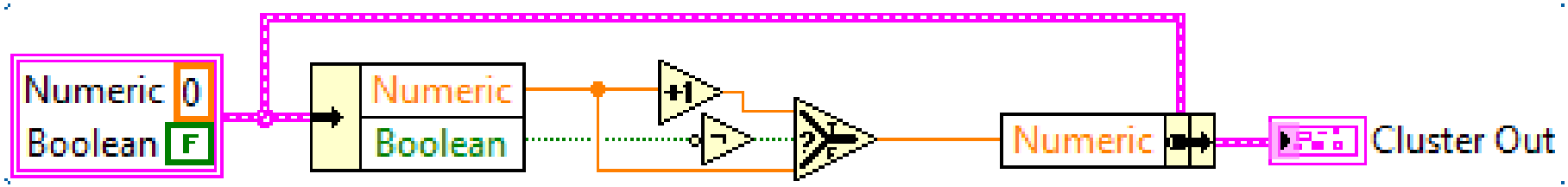
b) A，C，接著 E。

c) A，C，D，接著 E。

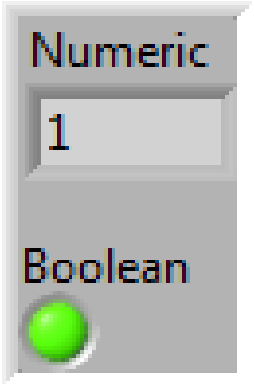
d) A，B，C，接著 E。

作者；曾智勇,chihyungtseng@hotmail.com

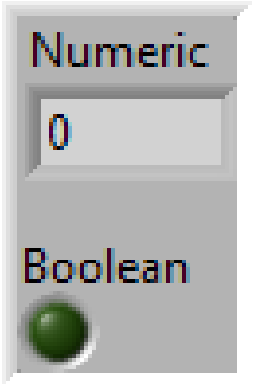
- 執行 VI 之後的結果為何？



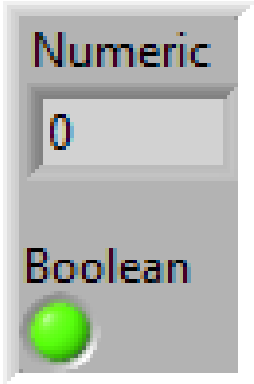
a)



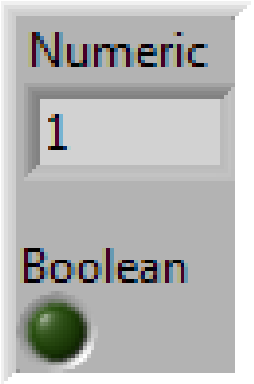
b)



c)

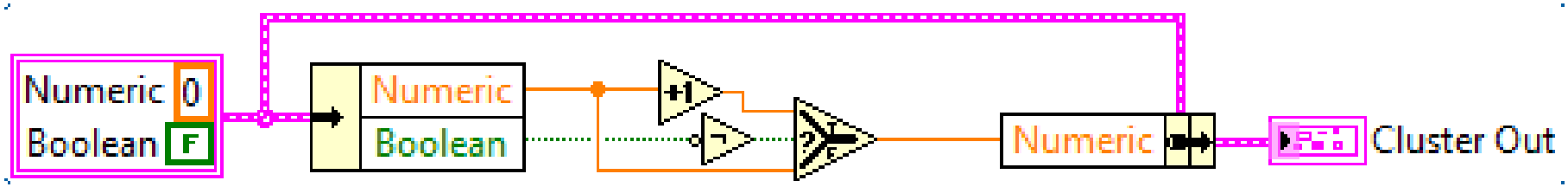


d)

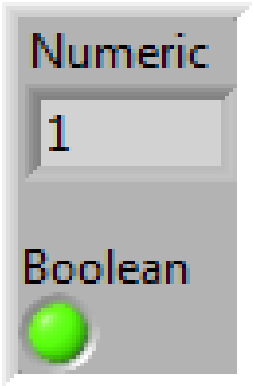


作者；曾智勇, chihyungtseng@hotmail.com

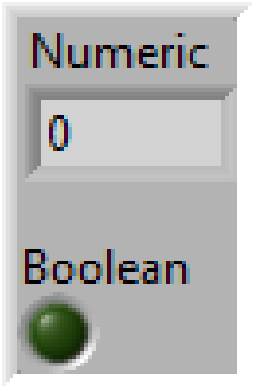
- 執行 VI 之後的結果為何？



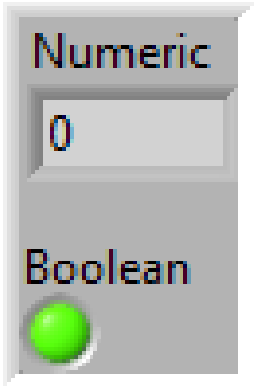
a)



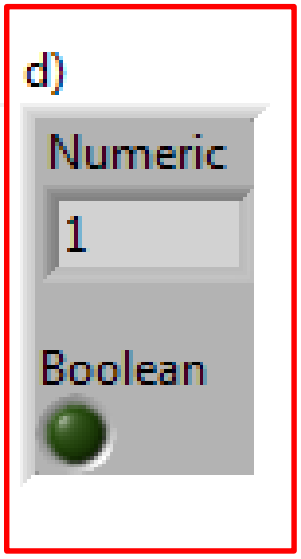
b)



c)

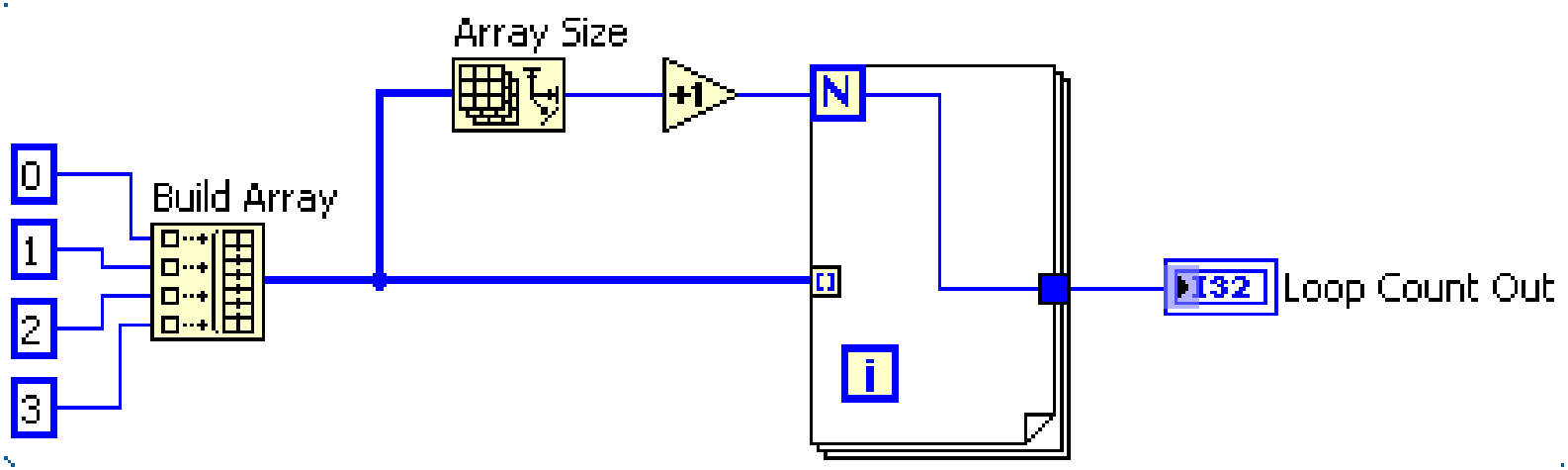


d)



作者；曾智勇, chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



a) 5

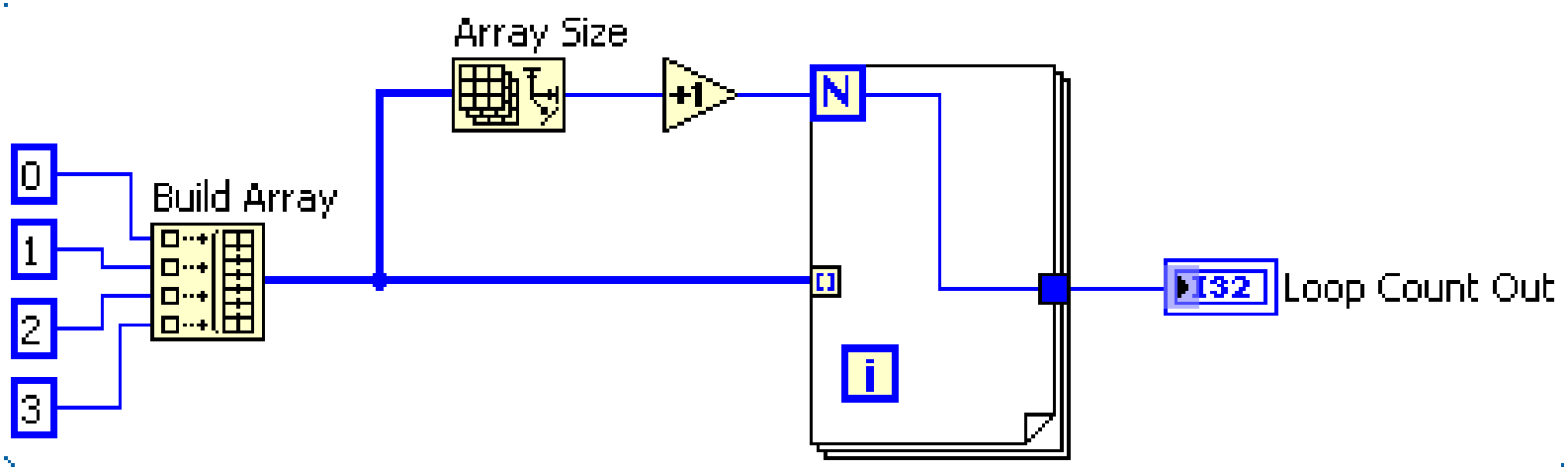
b) 4

c) 3

d) 1

作者；曾智勇,chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



a) 5

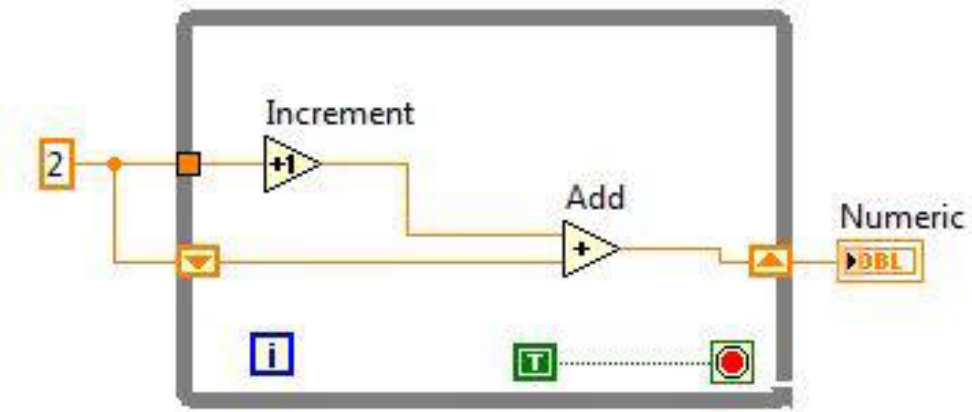
b) 4

c) 3

d) 1

作者；曾智勇,chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



a) 0

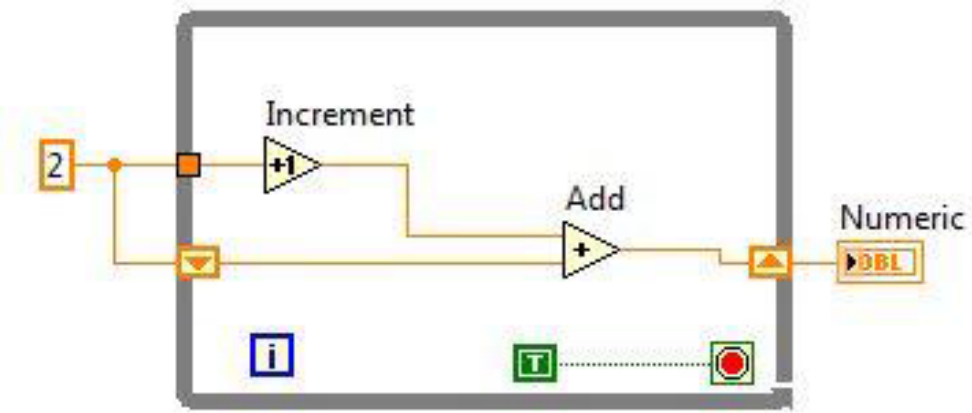
b) 4

c) 5

d) 無限迭代

作者；曾智勇, chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



a) 0

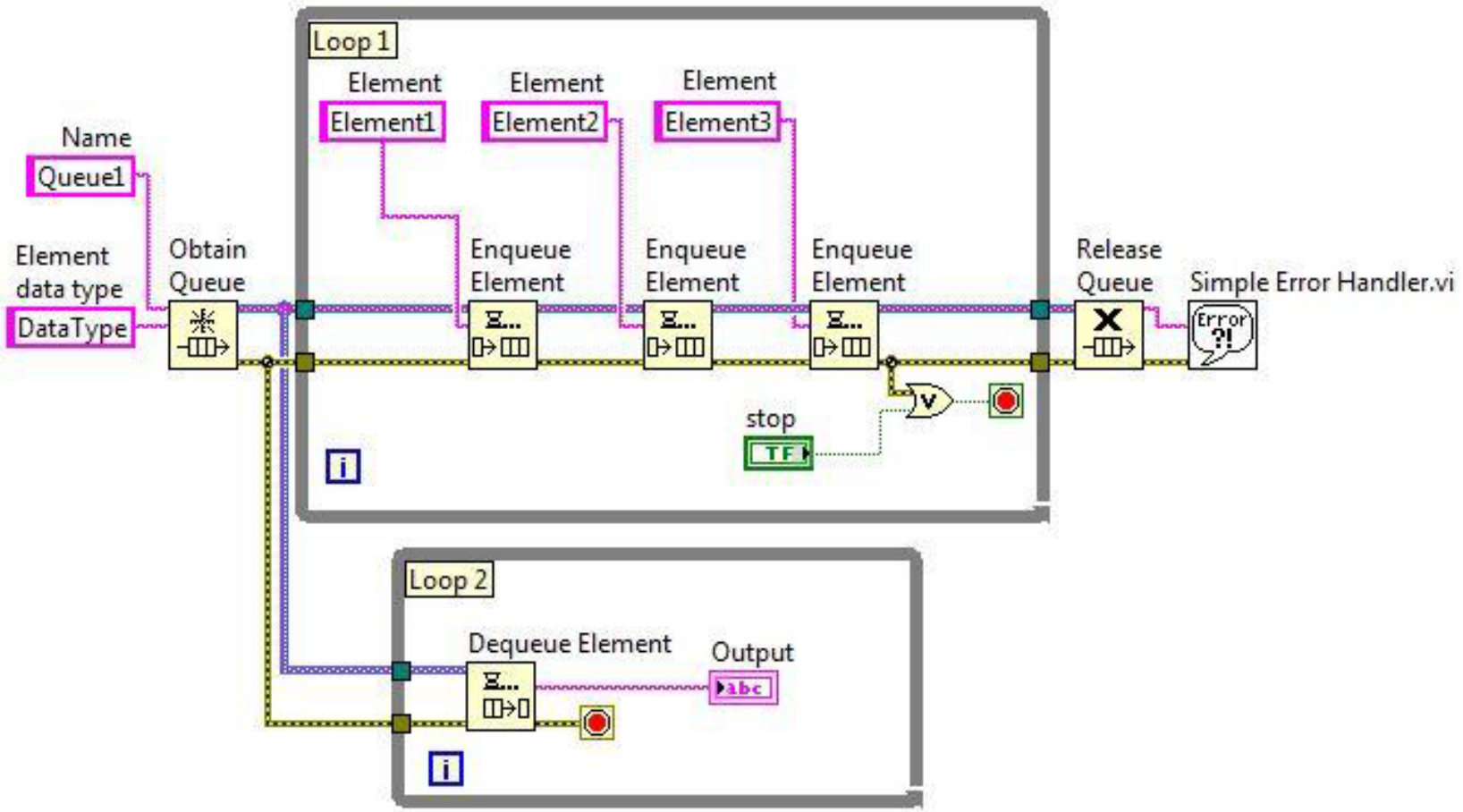
b) 4

c) 5

d) 無限迭代

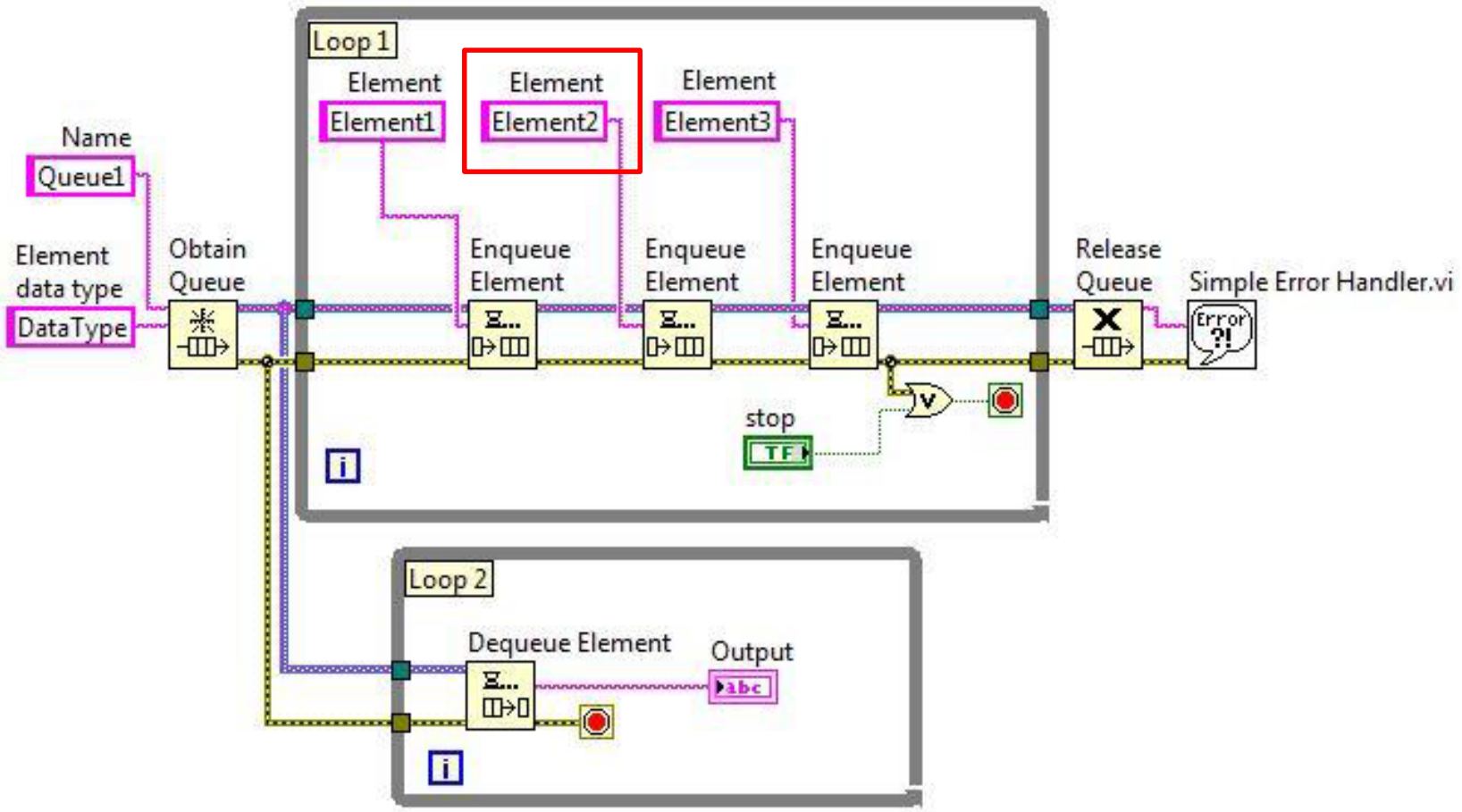
作者；曾智勇,chihyungtseng@hotmail.com

- Loop 2，i=4時，Output 顯示為何？



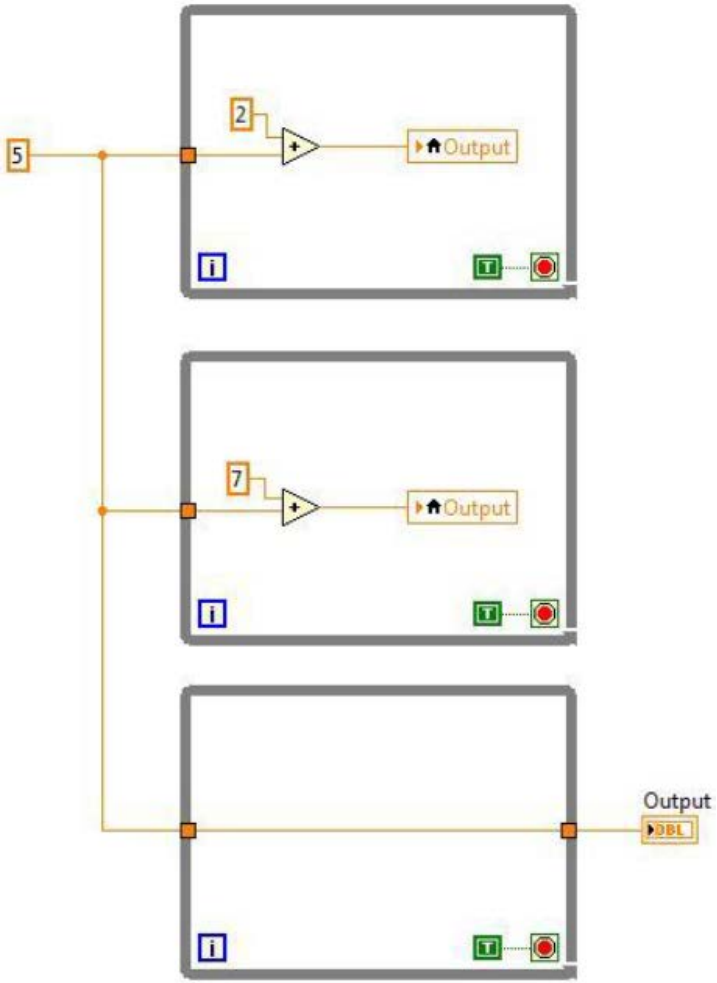
作者；曾智勇, chihyungtseng@hotmail.com

- Loop 2，i=4時，Output 顯示為何？**(Element2)**



作者；曾智勇,chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



a) 5

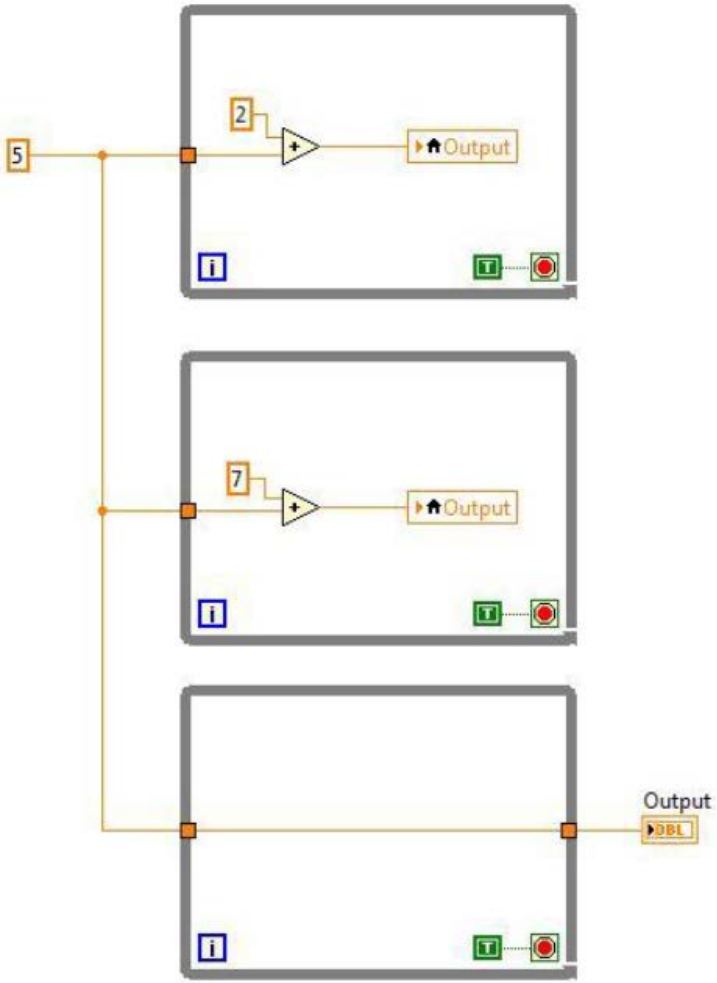
b) 7

c) 無法確認

d) 12

作者；曾智勇, chihyungtseng@hotmail.com

- 執行 VI 之後的結果為何？



a) 5

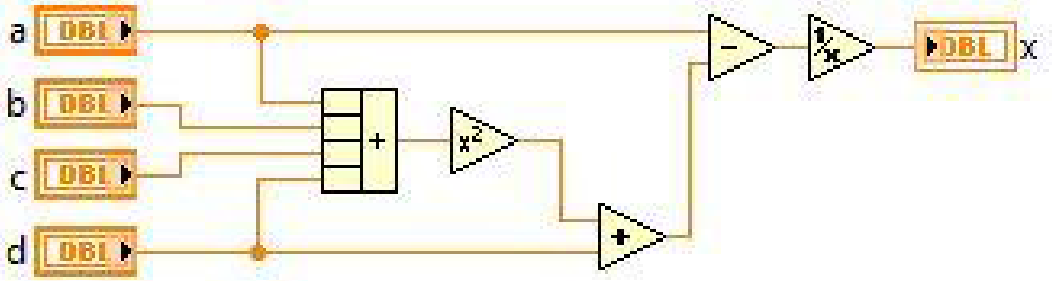
b) 7

c) 無法確認

d) 12

作者；曾智勇,chihyungtseng@hotmail.com

- 以下程式碼等效於合者？



A $$x = \frac{1}{((a+b+c+d)^2+d)-a}$$

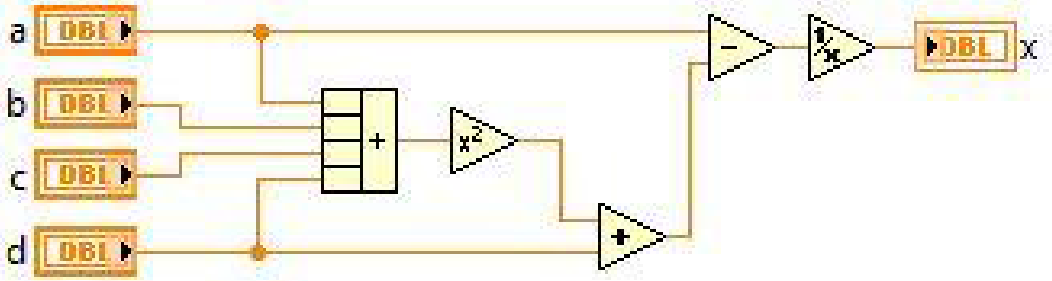
B $$x = \frac{1}{d-(a+b+c+d)^2+a}$$

C $$x = \frac{1}{a-((a+b+c+d)^2+d)}$$

D $$x = \frac{1}{a-((a^2+b^2+c^2+d^2)+d)}$$

作者；曾智勇, chihyungtseng@hotmail.com

- 以下程式碼等效於合者？



A $$x = \frac{1}{((a+b+c+d)^2+d)-a}$$

B $$x = \frac{1}{d-(a+b+c+d)^2+a}$$

C $$x = \frac{1}{a-((a+b+c+d)^2+d)}$$

D $$x = \frac{1}{a-((a^2+b^2+c^2+d^2)+d)}$$

作者；曾智勇, chihyungtseng@hotmail.com

基礎練習

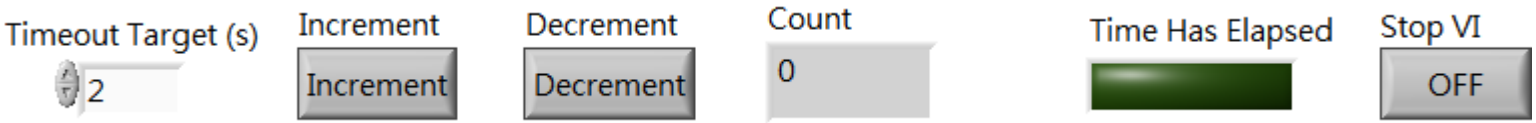
作者；曾智勇, chihyungtseng@hotmail.com

- 練習1. 計時器

計時器
1. 設定秒數
2. 手動重置
3. 自動重置
4. 經過時間
5. 計時完畢
6. 停止

作者；曾智勇, chihyungtseng@hotmail.com

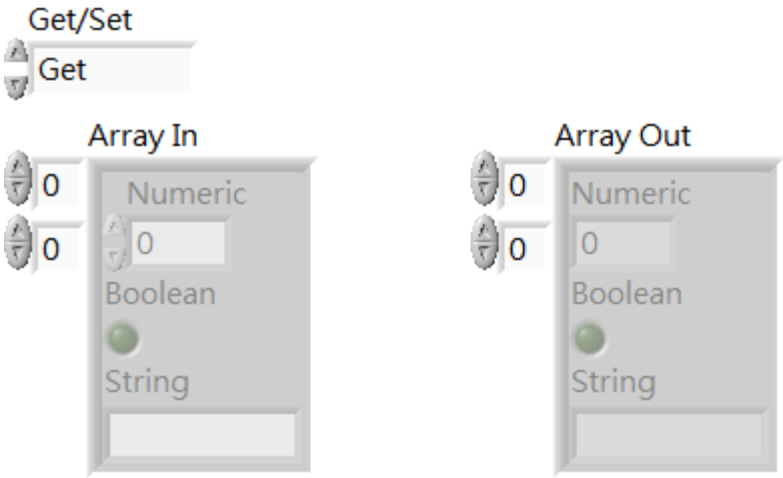
- 練習2. 計時器(無動作計時器)



超過時間沒動作, 則亮燈

作者；曾智勇, chihyungtseng@hotmail.com

- 練習3. 存取ini檔

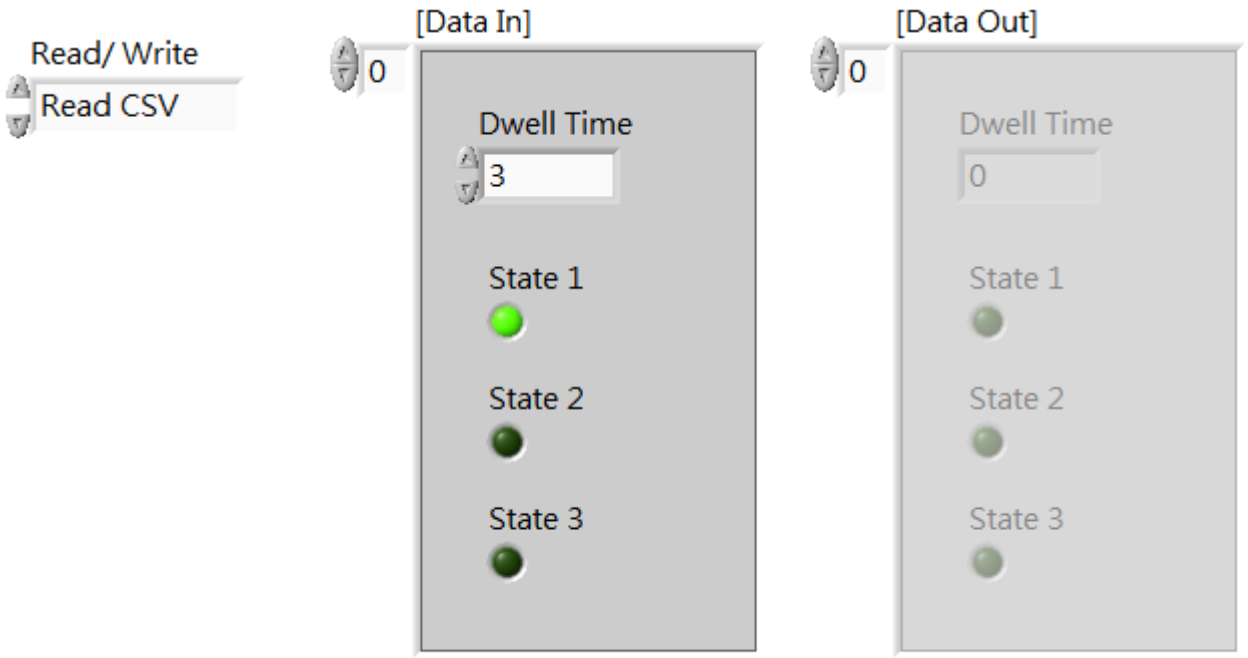


[Item One]
Numeric = 2.250000
String = "One"
Boolean = FALSE
Index1 = 1
Index2 = 1

[Item Two]
Numeric = 1.350000
String = "Two"
Boolean = TRUE
Index1 = 1
Index2 = 2

作者；曾智勇, chihyungtseng@hotmail.com

- 練習4. 存取csv檔



作者；曾智勇, chihyungtseng@hotmail.com

- 練習5. 取得系統時間(年/月/日/時/分/秒)

Code String Out

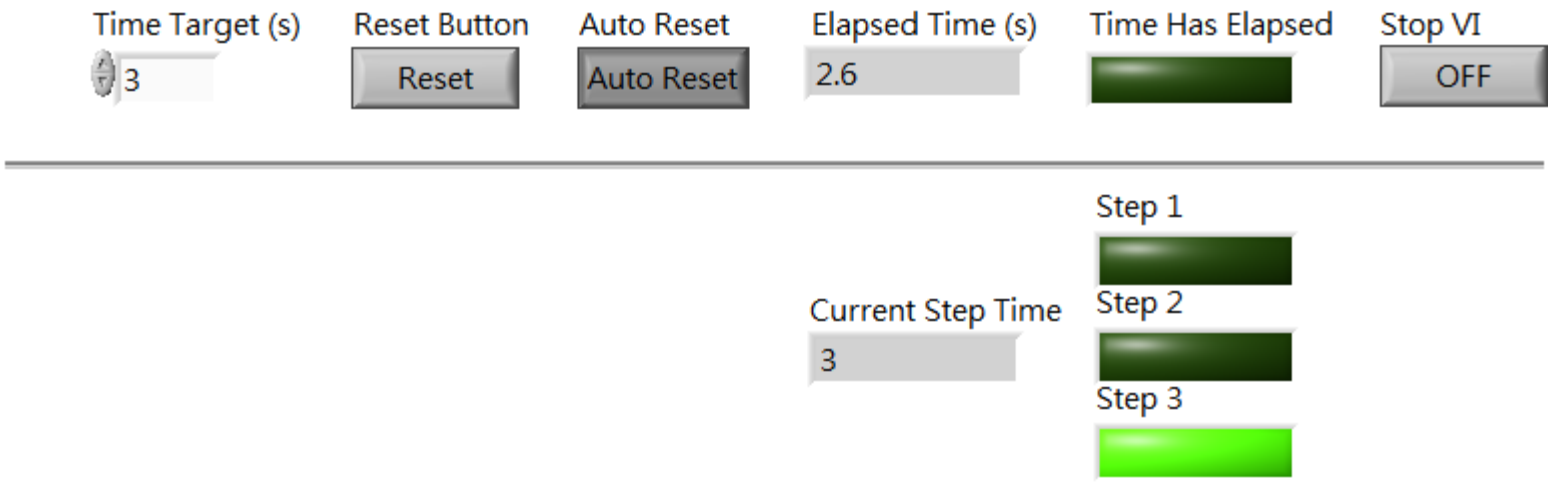
```
2017年2月24日  下午 04:45:11
```

current time

```
下午 04:45:11.454
2017/2/24
```

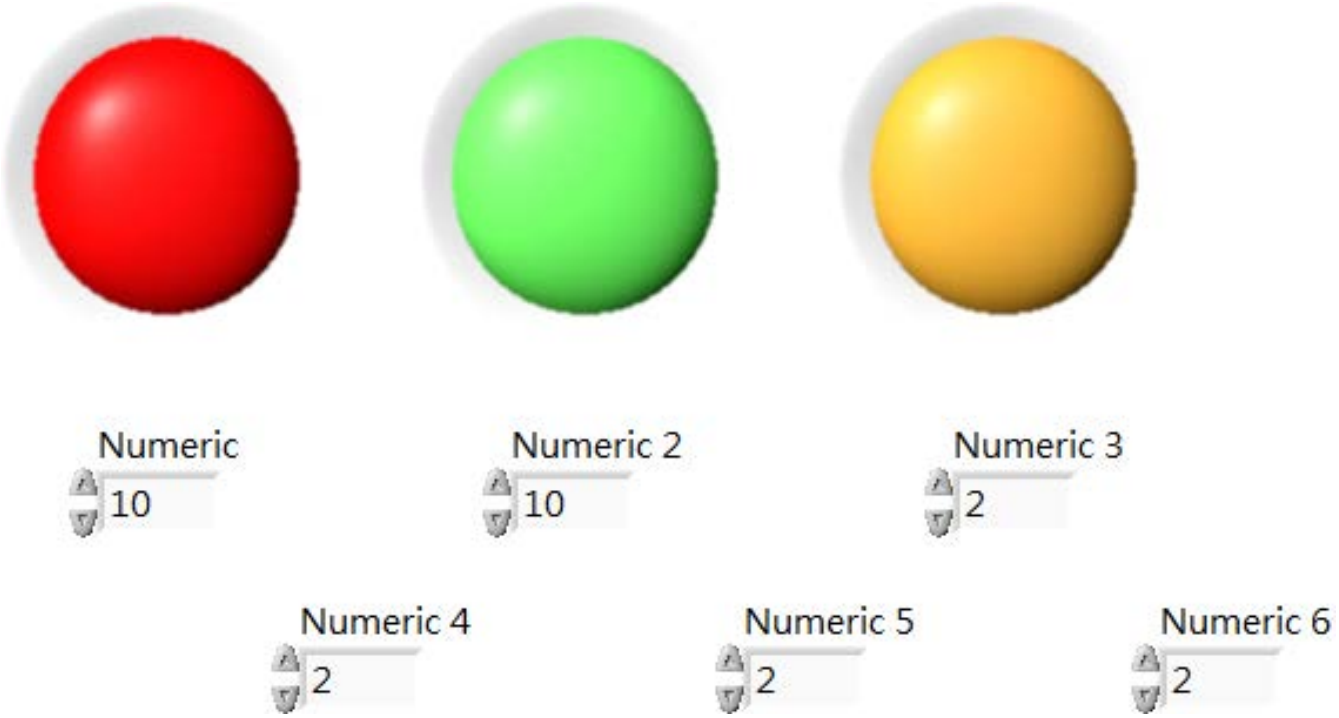

作者；曾智勇, chihyungtseng@hotmail.com

- 練習6. 計時器, 時間到則切換燈號



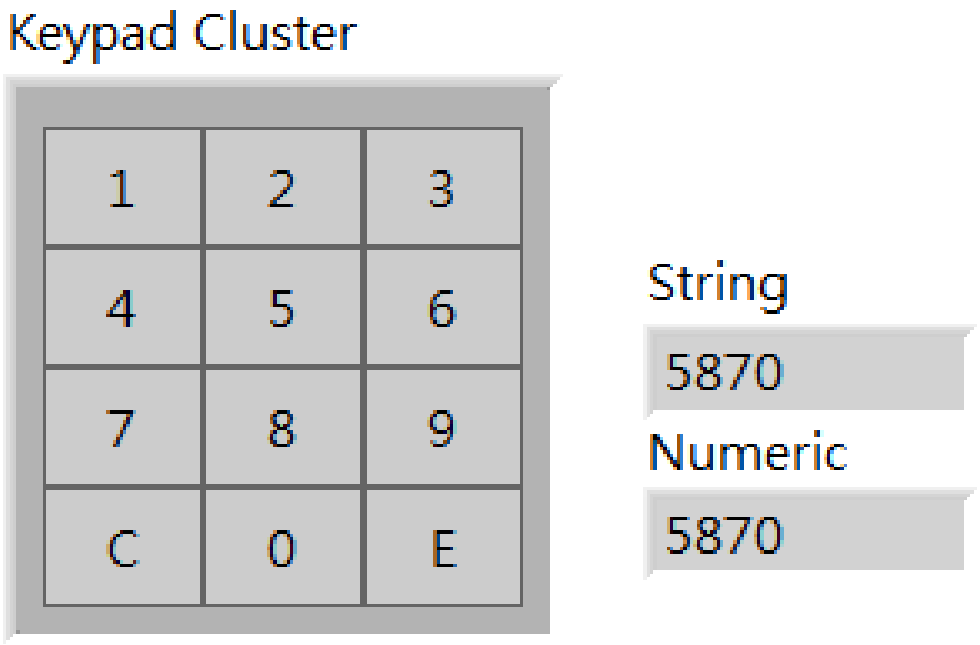
作者；曾智勇, chihyungtseng@hotmail.com

- 練習7. 可設定秒數的紅綠燈



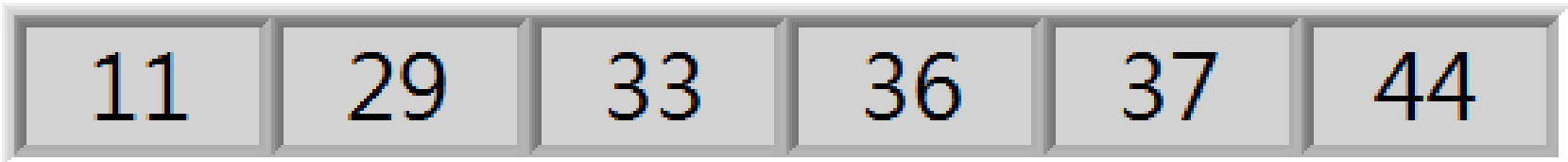
作者；曾智勇, chihyungtseng@hotmail.com

- 練習8.按鍵式數字密碼輸入



作者；曾智勇,chihyungtseng@hotmail.com

- 練習9.大樂透產生器, 49選6, 不重複, 由小到大排列



作者；曾智勇,chihyungtseng@hotmail.com

專案講解

作者；曾智勇, chihyungtseng@hotmail.com

專案實做

作者；曾智勇, chihyungtseng@hotmail.com