



LabVIEW程式設計與實習

陳怡姝

國立中央大學太空科學研究所

2017年7月18日

大綱

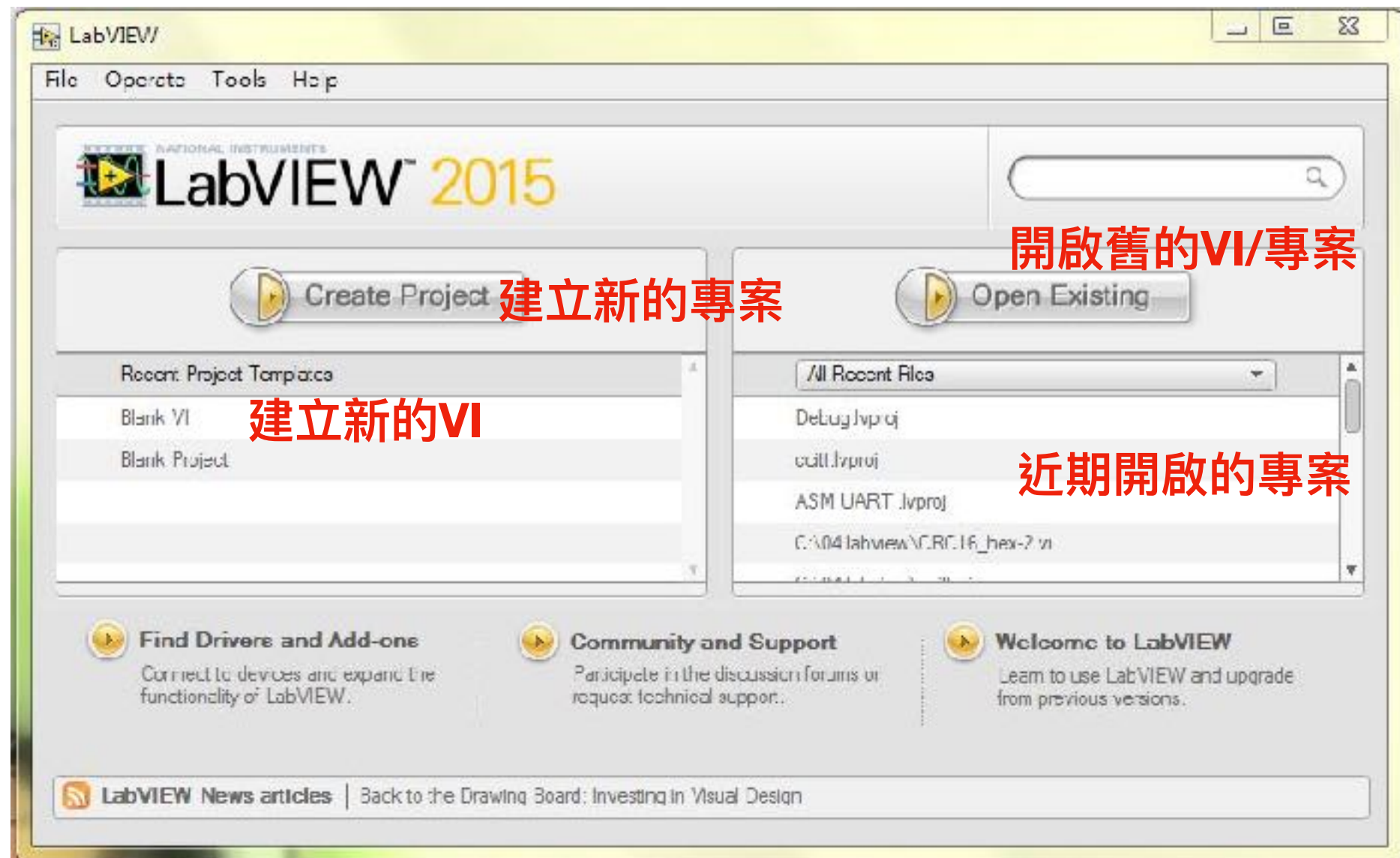
- LabVIEW簡介
- 操作環境與基本操作
 - 工具面板
 - Front Panel 人機介面
 - Block Diagram 程式區
 - 資料型態、線
 - 資料流
- Structure 結構
 - While Loop
 - For Loop
 - Case Structure
- Timing
- subVI
- 作業練習

LabVIEW 簡介

- **Laboratory Virtual Instrumentation Engineering Workbench**，簡稱LabVIEW，是由美國國家儀器公司所開發的圖型化程式編譯平台。
- NI LabVIEW 軟體為圖形化(G)的程式設計環境，是透過圖示、端點、接線而設計出工程師心中所要的物件。
- LabVIEW的特色在於軟體與硬體的整合，可以讓我們透過虛擬的方式模擬硬體，也可以直接和硬體整合進行資料擷取與輸出等等。










操作環境

- 開始 >> 搜尋”LabVIEW 2015”



Choose a starting point for the project:

- All
- Templates
 - Desktop
 - myRIO
 - roboRIO
 - Robotics
 - SoftMotion
- Sample Projects
 - CompactRIO
 - Desktop
 - myRIO
 - NI-579X
 - Real-Time
 - roboRIO
 - SoftMotion

-  **Blank Project** *Templates*
Creates a blank project.
-  **Blank VI** *Templates*
Creates a blank VI.
-  **Simple State Machine** *Templates*
Facilitates defining the execution sequence for sections of code. [More Information](#)
-  **Queued Message Handler** *Templates*
Facilitates multiple sections of code running in parallel and sending data between them. [More Information](#)
-  **Actor Framework** *Templates*
Creates an application that consists of multiple, independent tasks that communicate with each other. This template makes extensive use of LabVIEW classes. [More Information](#)
-  **myRIO Project** *Templates*
Creates a new project for controlling your myRIO. This template uses code written with the LabVIEW myRIO Toolkit. [More Information](#)
-  **roboRIO Project** *Templates*
Creates a new project for controlling your roboRIO. This template uses code written with the LabVIEW roboRIO Toolkit. [More Information](#)
-  **myRIO Custom FPGA Project** *Templates*
Facilitates customizing the myRIO FPGA personality. This template uses code written with the LabVIEW FPGA Module. [More Information](#)
-  **roboRIO Custom FPGA Project** *Templates*
Facilitates customizing the roboRIO FPGA personality. This template uses code written with the LabVIEW FPGA Module. [More Information](#)

Additional Search

Keyword

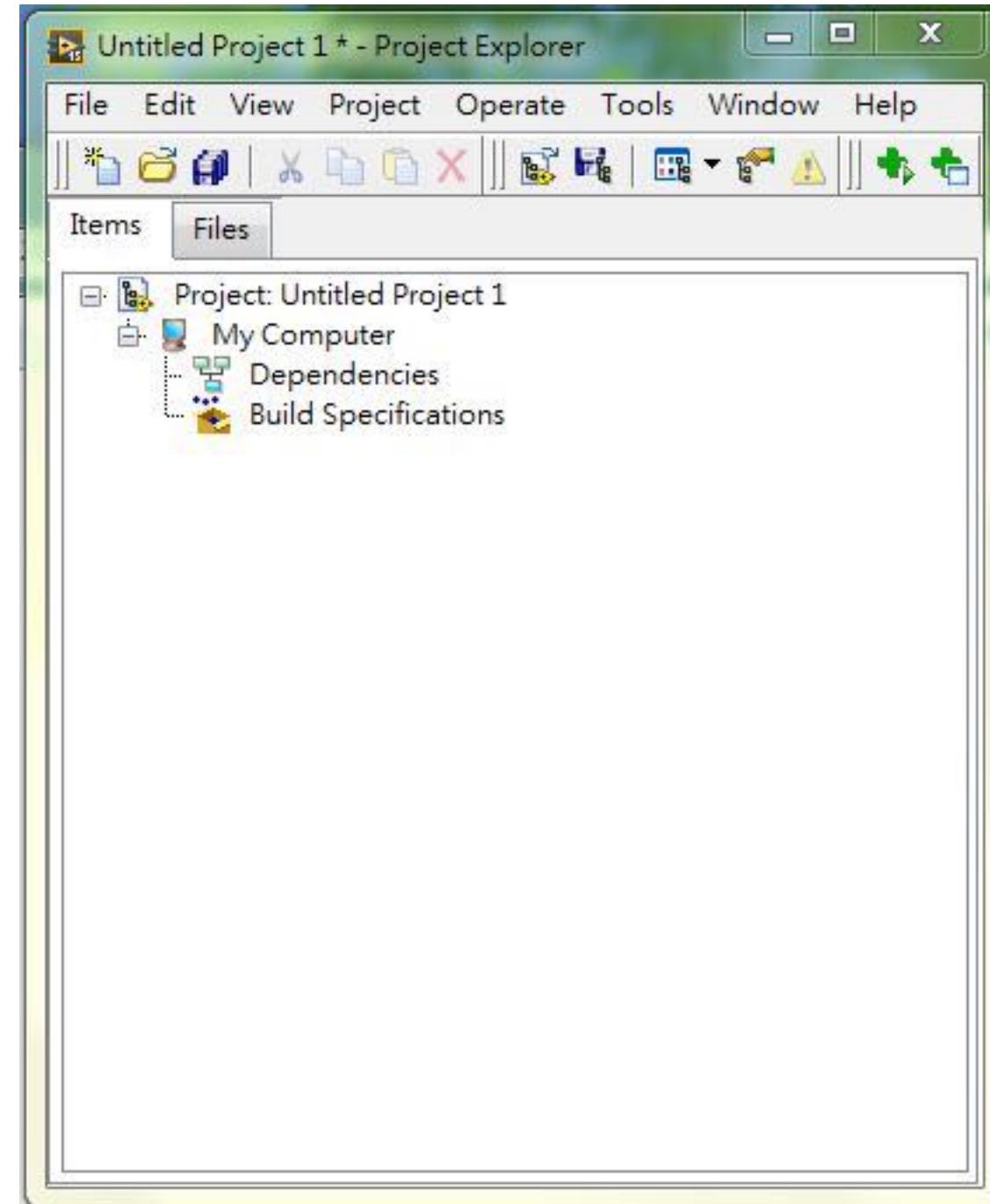
Finish

Cancel

Help

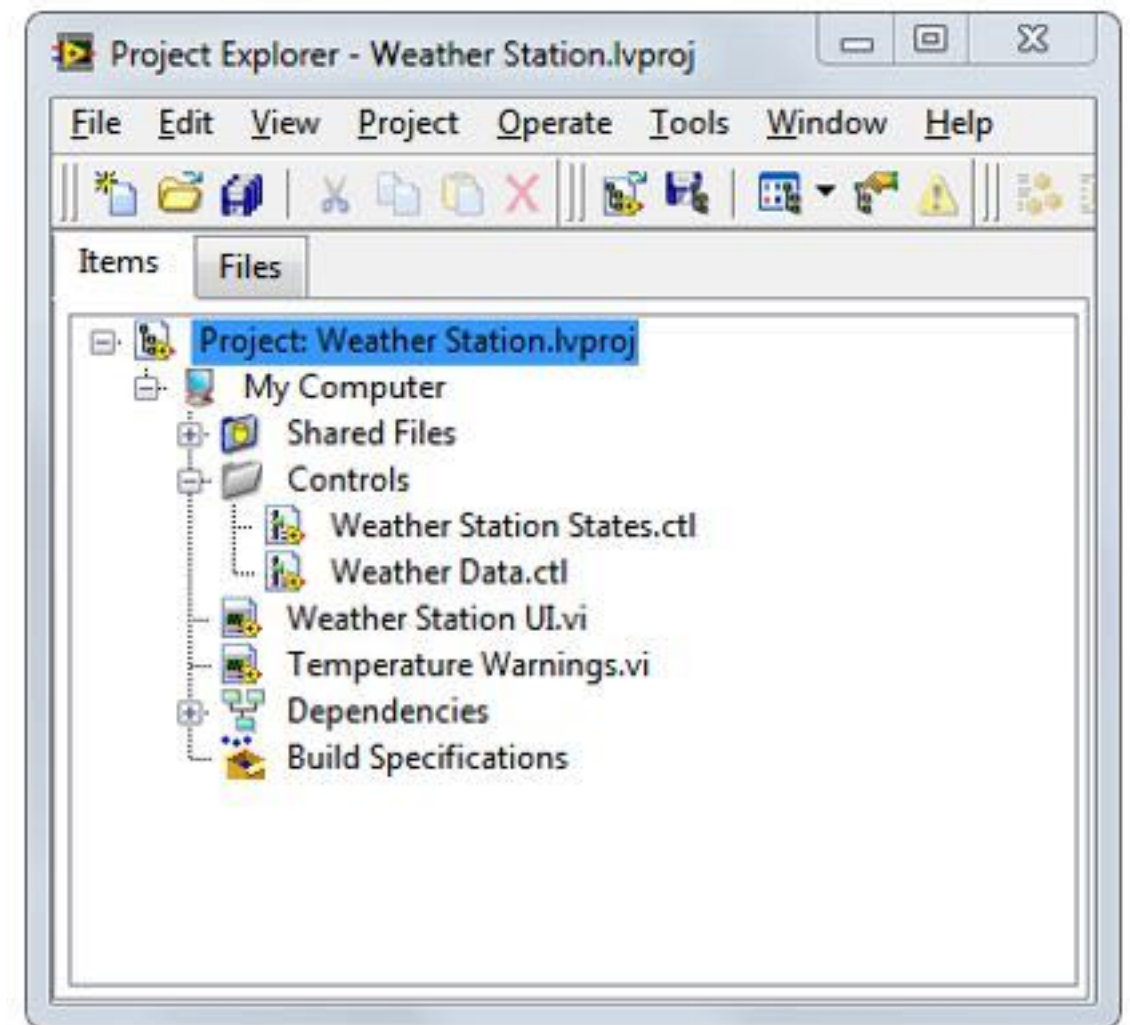
Project 專案

- 開發大型程式或專案
 - 管理多個VI及subVI，將LabVIEW與non-LabVIEW檔案包在一起
 - 程式較大或需要有規劃的整理，就會需要用到專案來管理。



LabVIEW Files

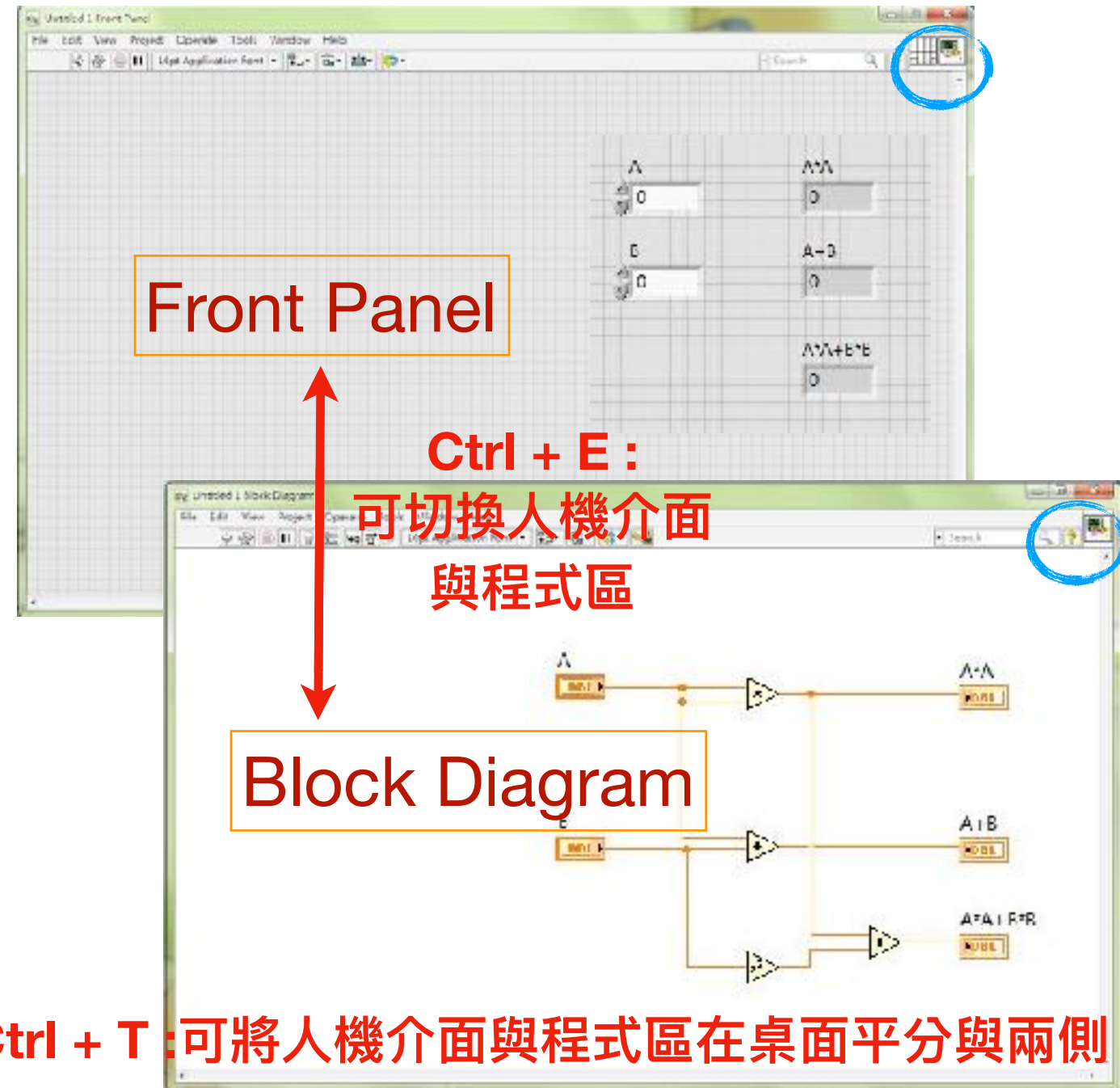
- Common LabVIEW file extensions:
 - LabVIEW project
 - .lvproj
 - Virtual instrument(VI)
 - .vi
 - Custom control
 - .ctl



myComputer右鍵 >> New >> VI

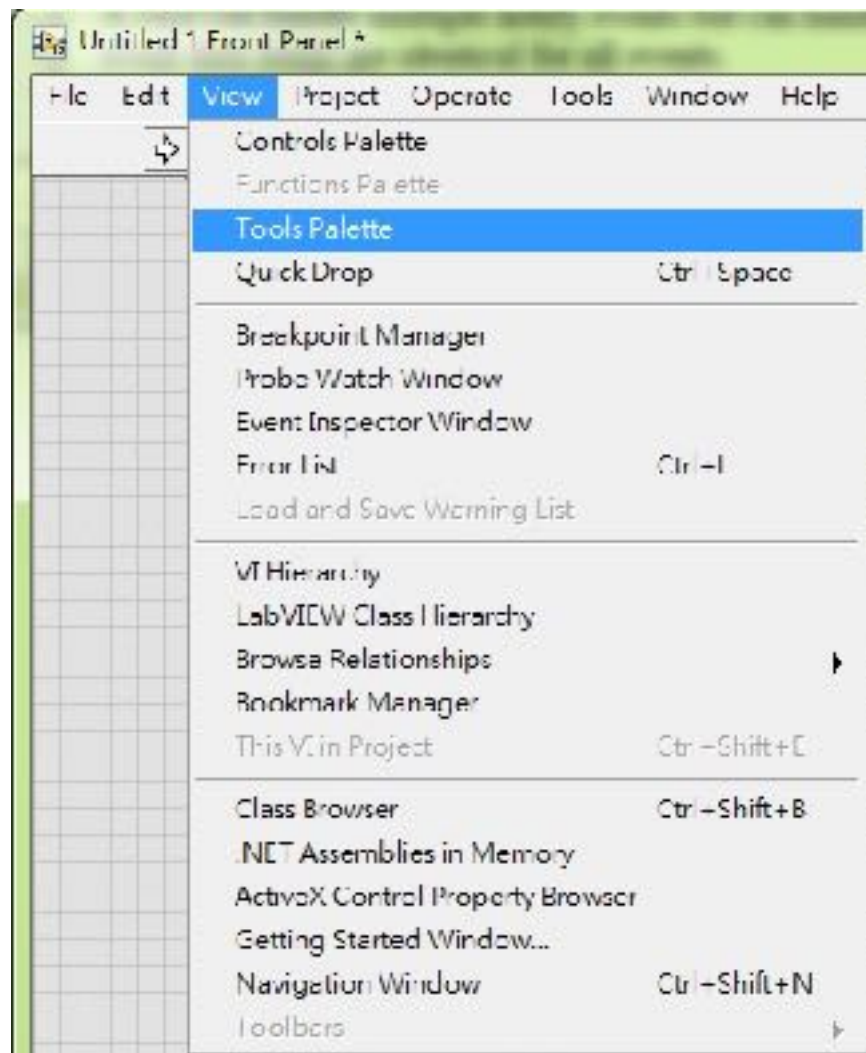
VI






- 人機介面(Front Panel)
Control : 輸入端
Indicators : 輸出端
- 程式區(Block Diagram)
將Front Panel端的物件
程式化，組織功能。
- 圖示/連接器面板
(Icon/Connect Pane)



工具面板

- 在front panel操作：Shift + 右鍵、View >> Tools Palette

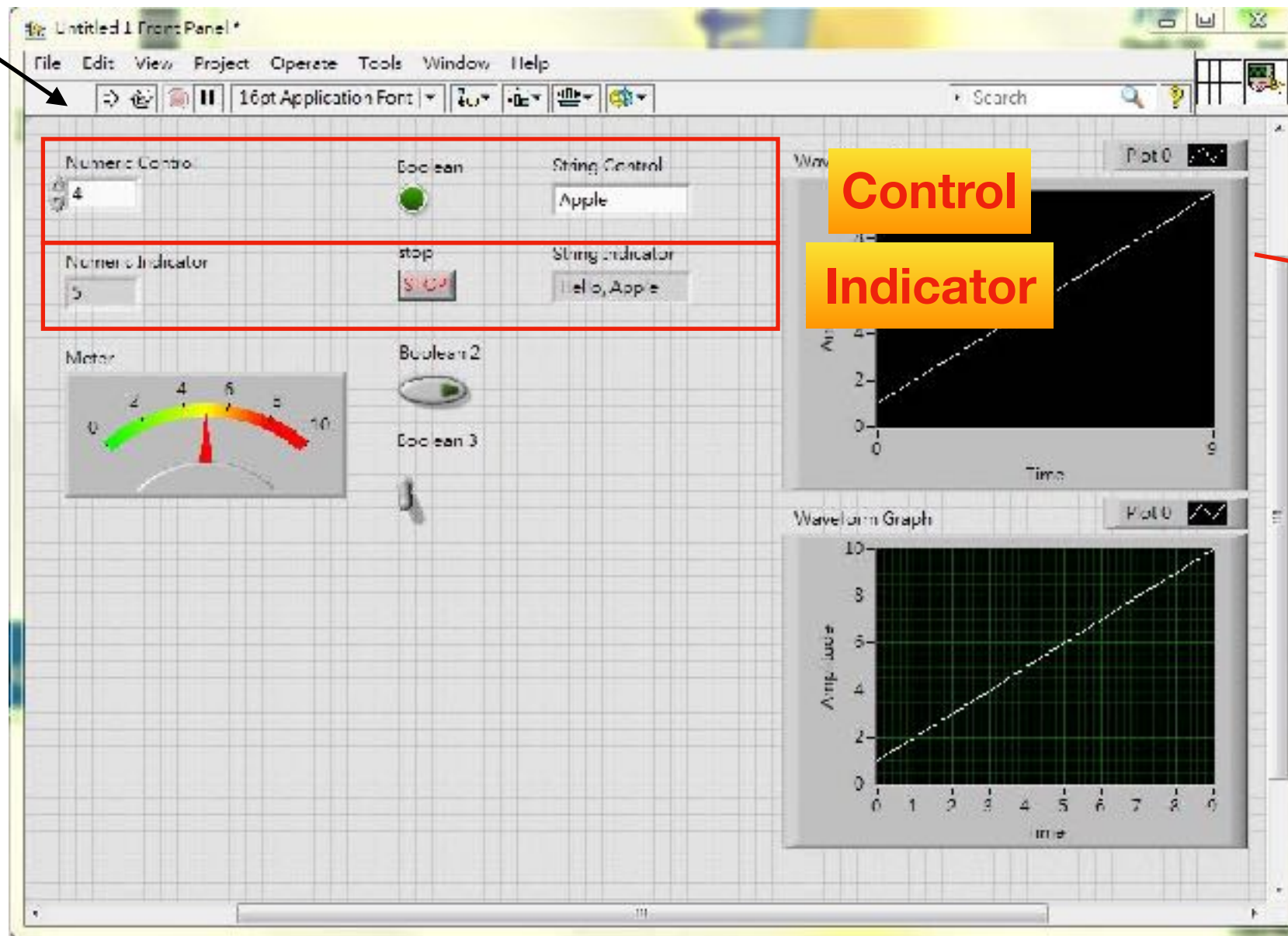


	自動選擇工具
	操作工具
	定位/改變大小工具
	標籤工具
	連線工具



Front Panel 人機介面 (使用者介面)

Tool bar

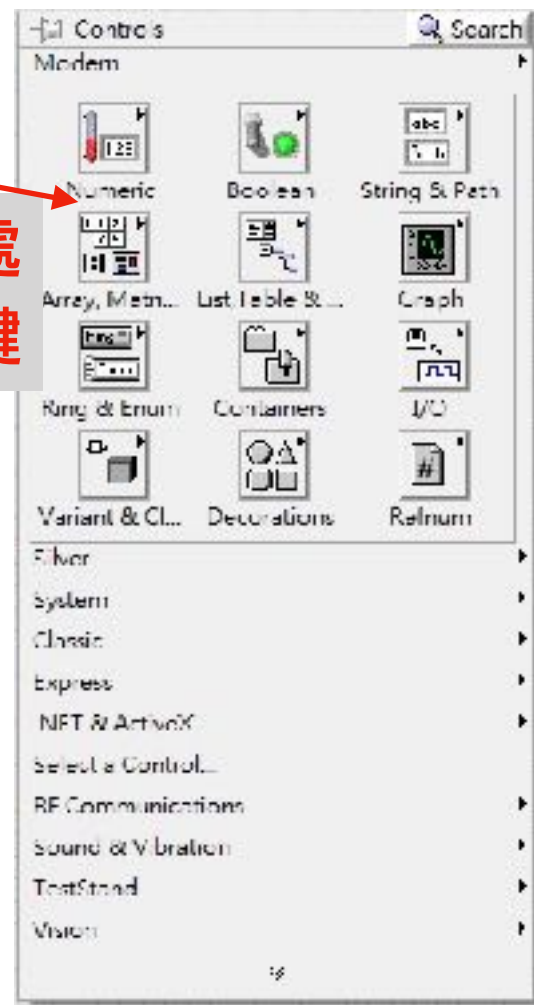


Control

Indicator

Icon

Control 控制面板



在空白處
滑鼠右鍵

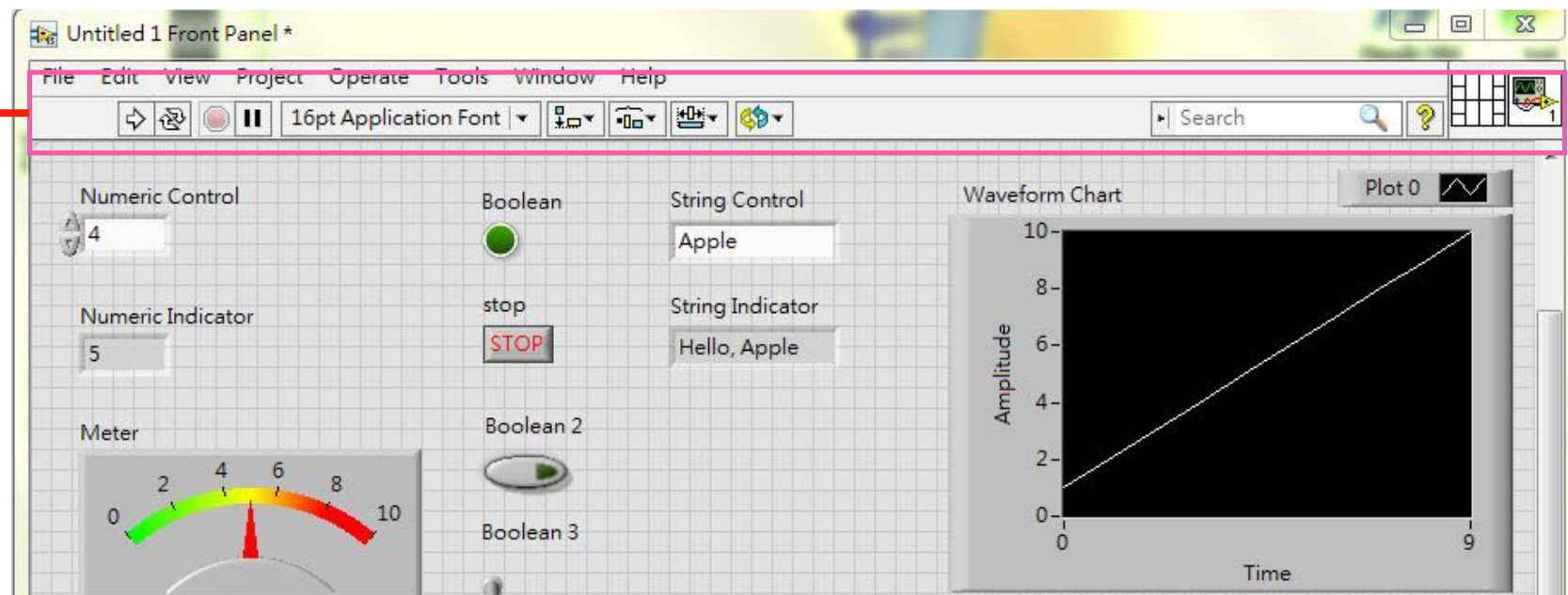
Front Panel

- 人機介面的工具欄



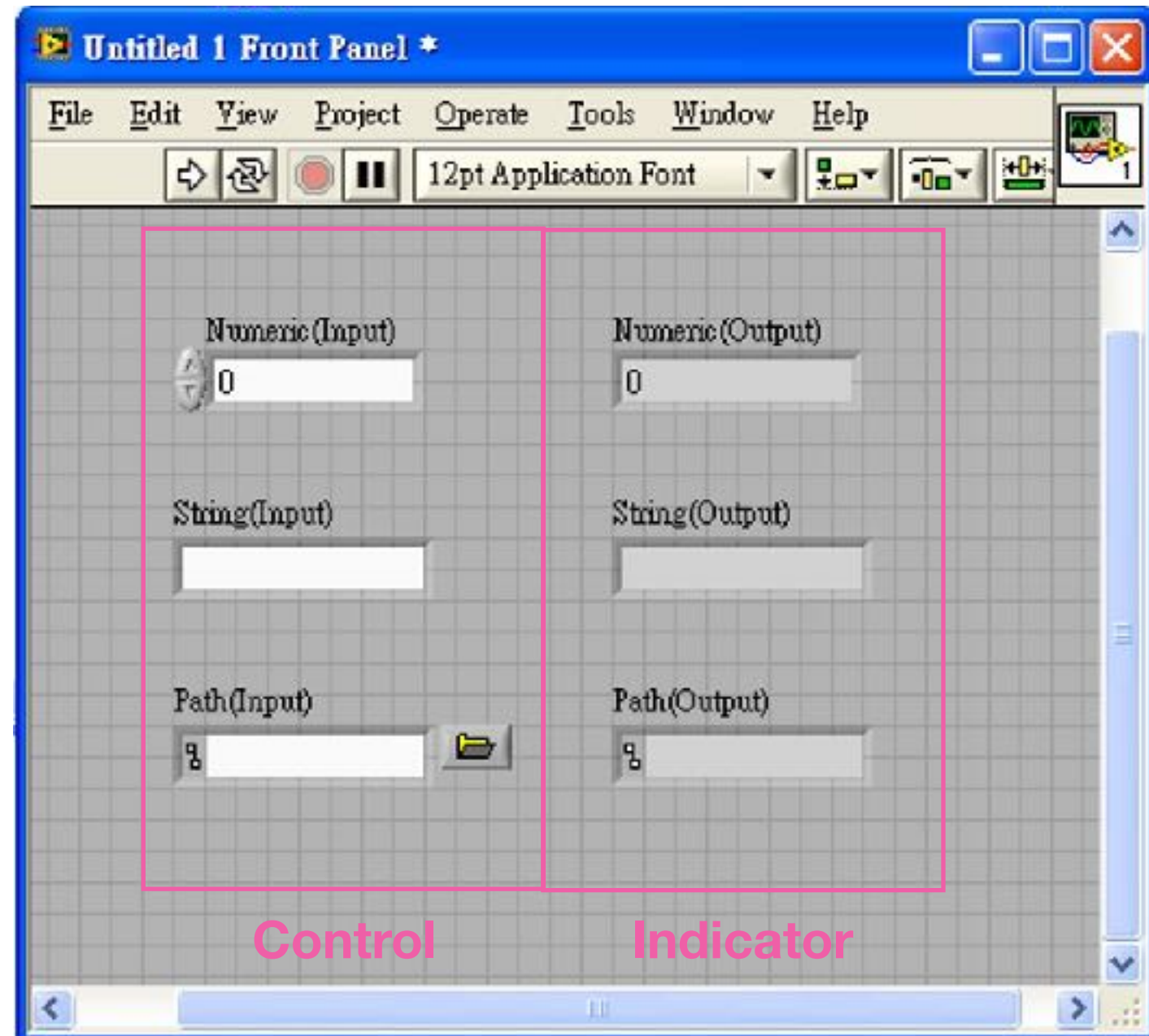
執行與停止 字型設定

↓
↓
↓
Align Objects : 對齊物件
Distribute Objects : 對齊物件
Resize Objects : 調整物件尺寸



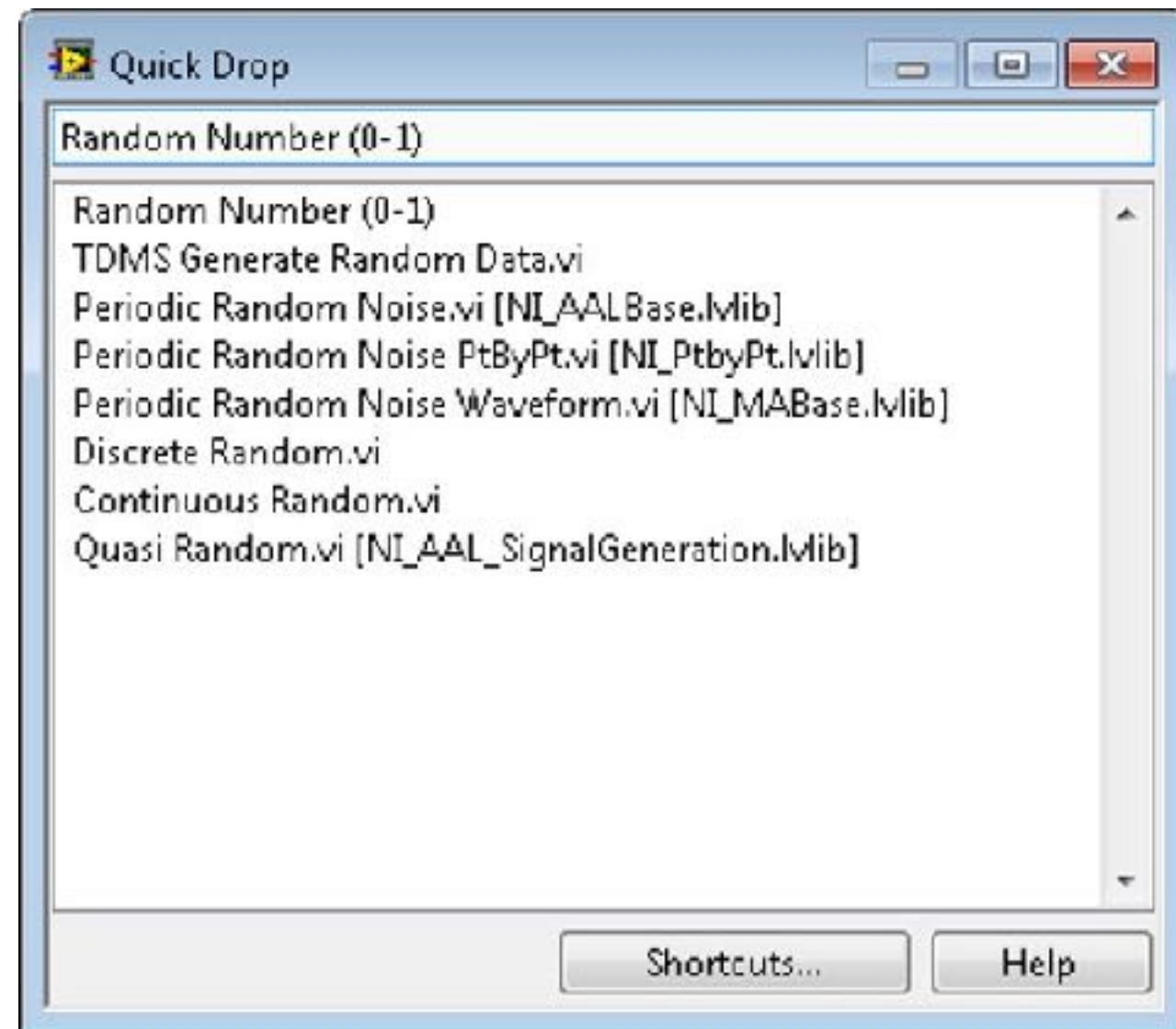
Controls & Indicators

- Controls
 - Input devices
 - Knobs, buttons, slides
 - Supply data to the block diagram
- Indicators
 - Output devices
 - Graphs, LEDs
 - Display data the block diagram acquires

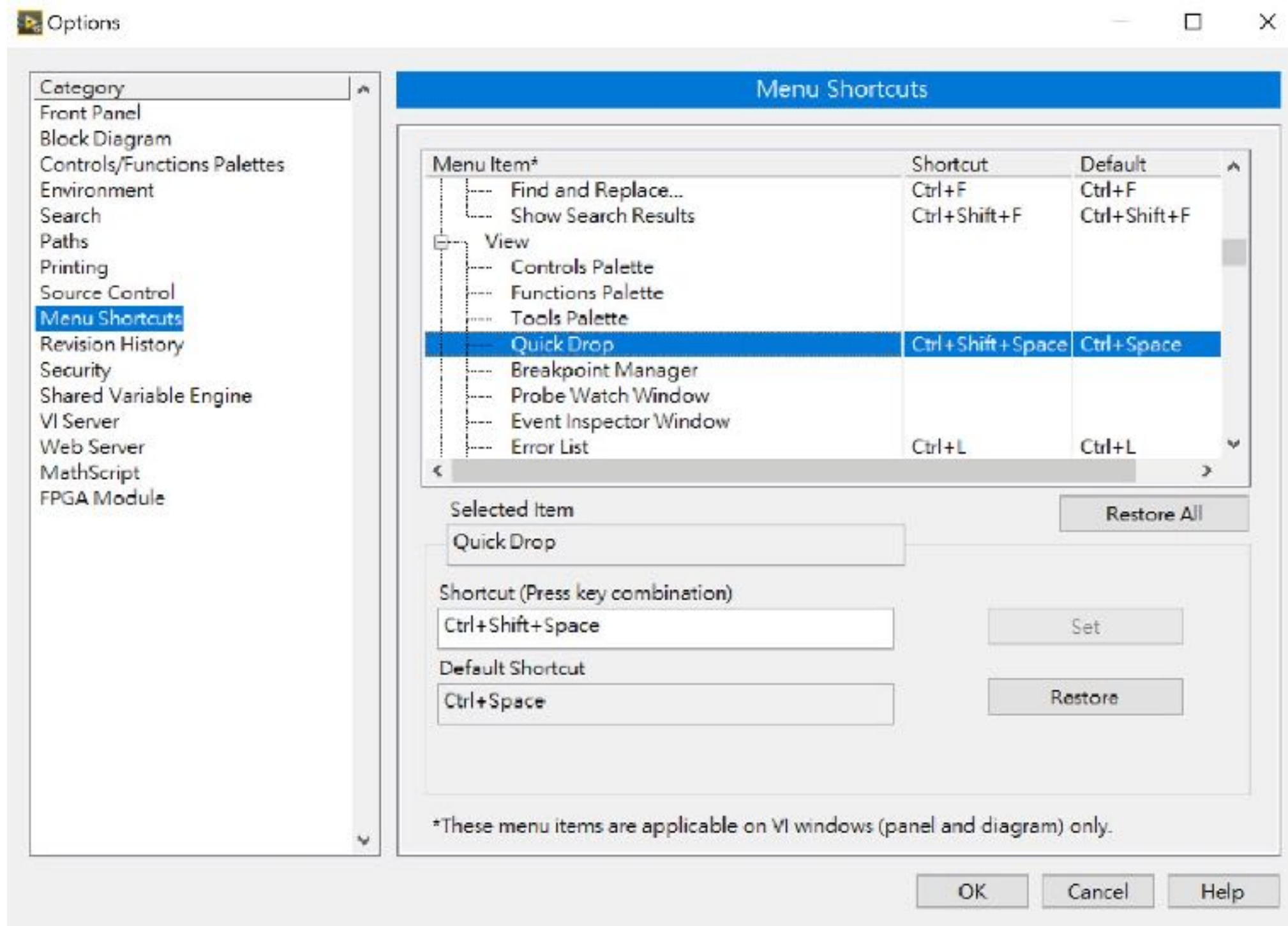


Quick Drop

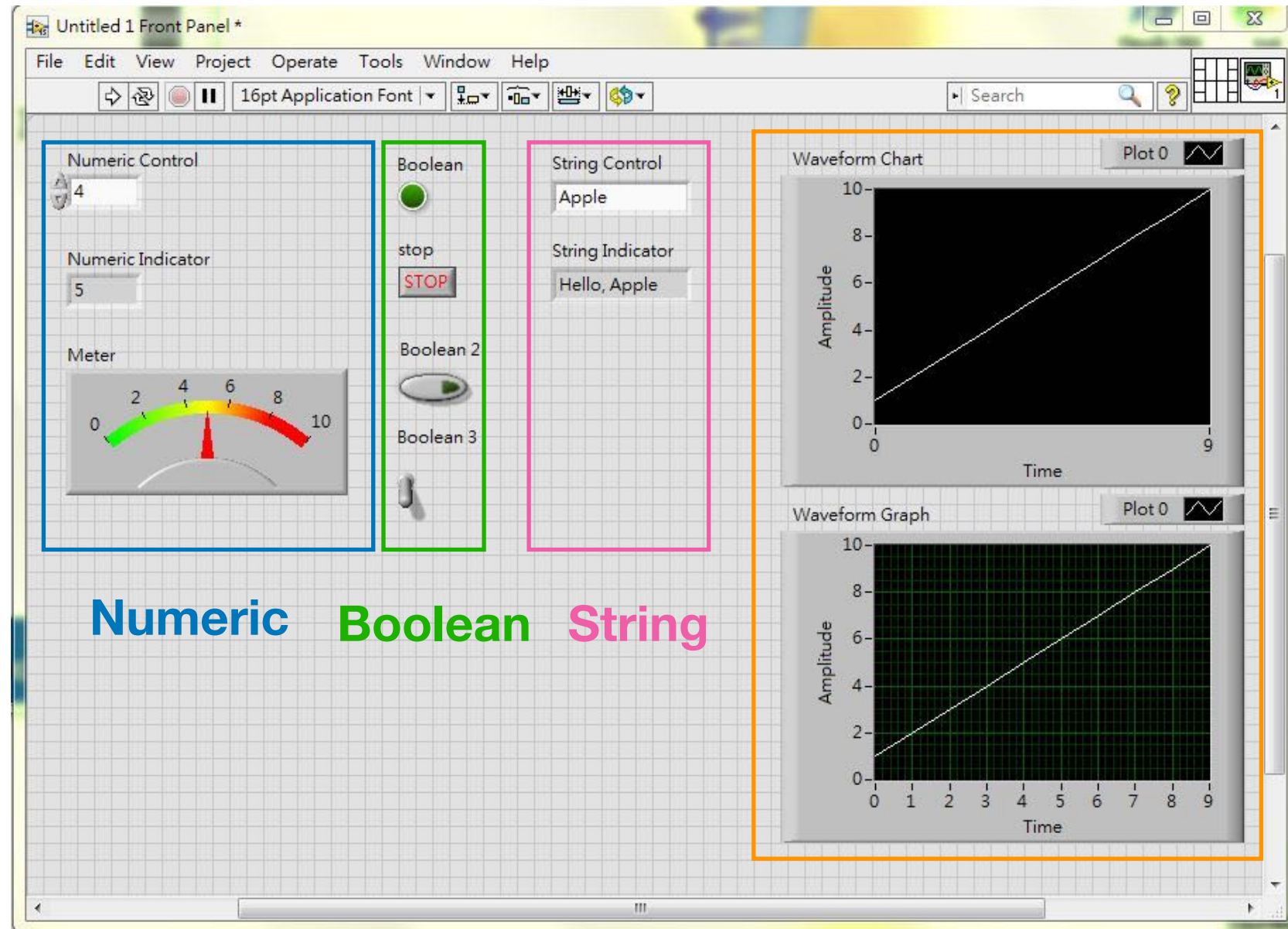
- 可以快速的可藉由名稱找到 controls, functions...
- 快速鍵 <Ctrl-space>



Quick Drop

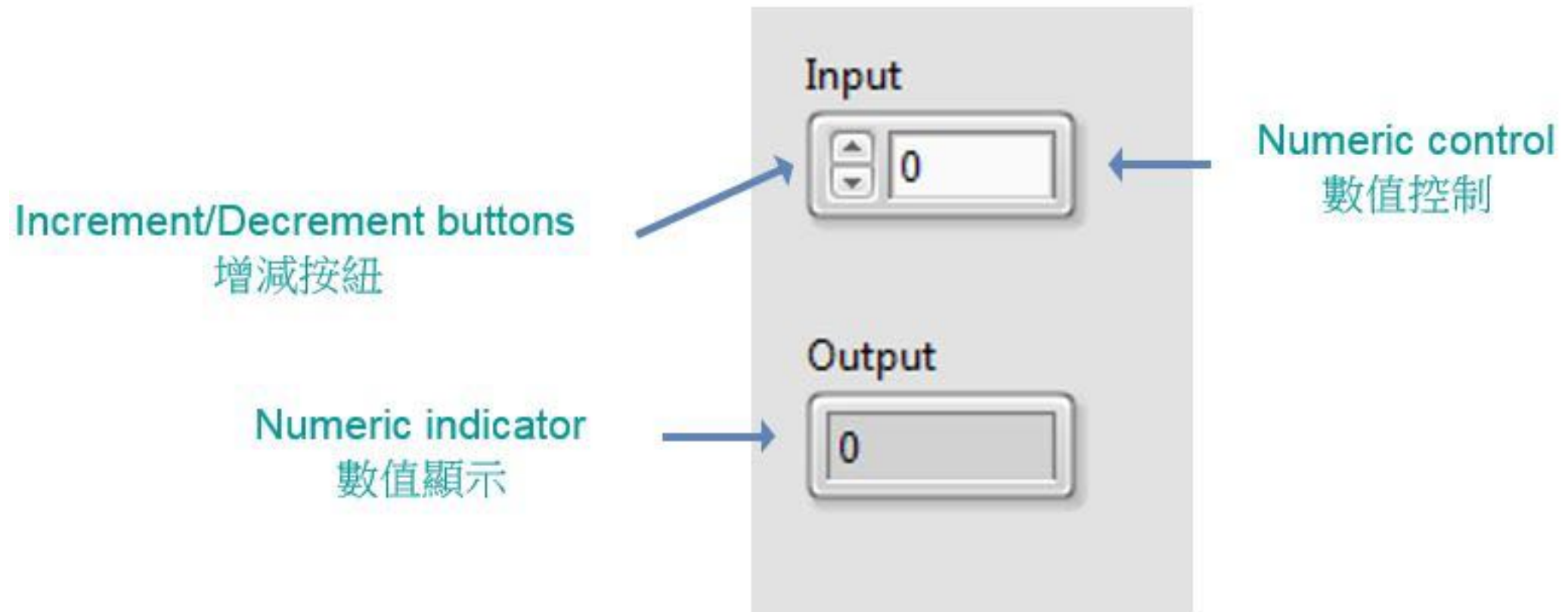


Front Panel



Numeric

- 數值可為整數或浮點數



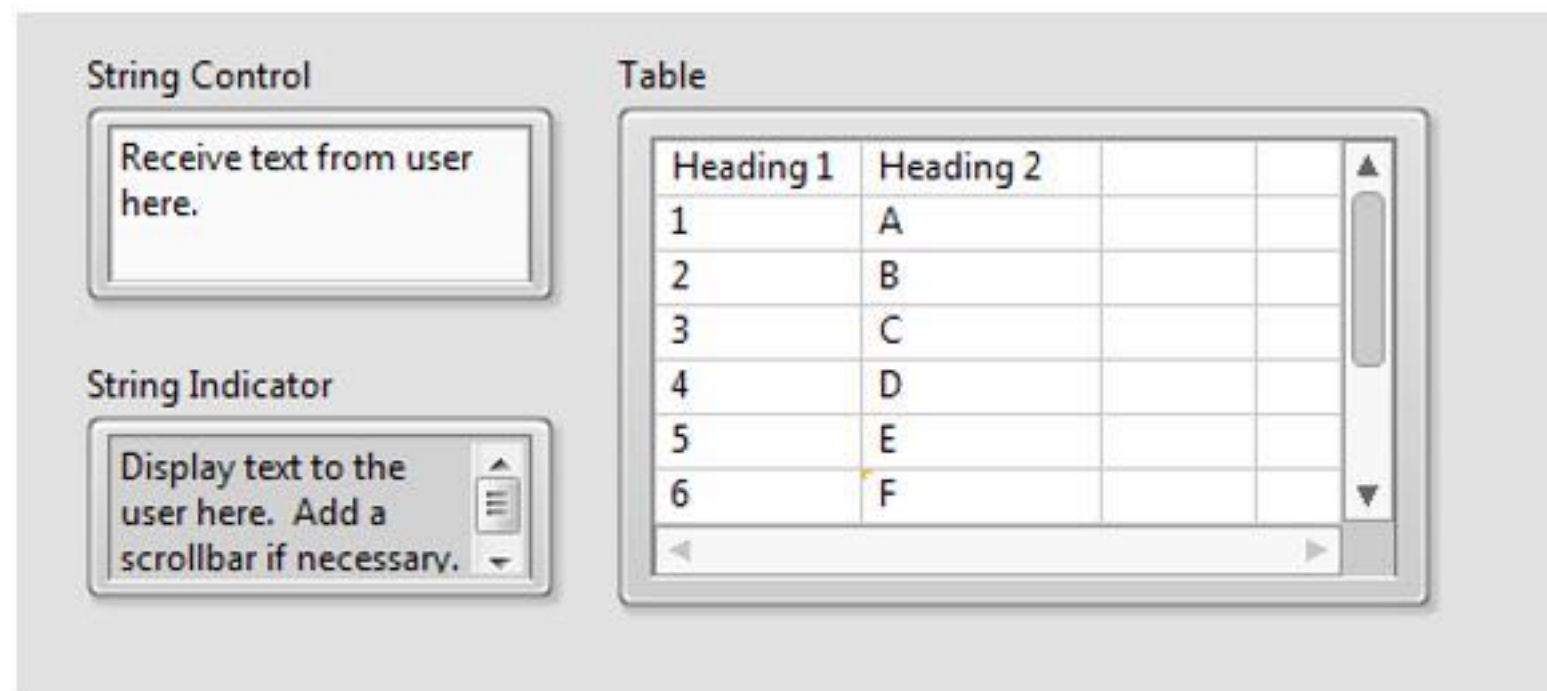
Boolean

- 布林的資料型態只有兩種，如True/False或On/Off。
- 使用布林元件輸入或顯示布林資料(TRUE/FALSE)。
- 布林元件就像開關(switches)，按鈕(push buttons)，燈號(LEDs)。

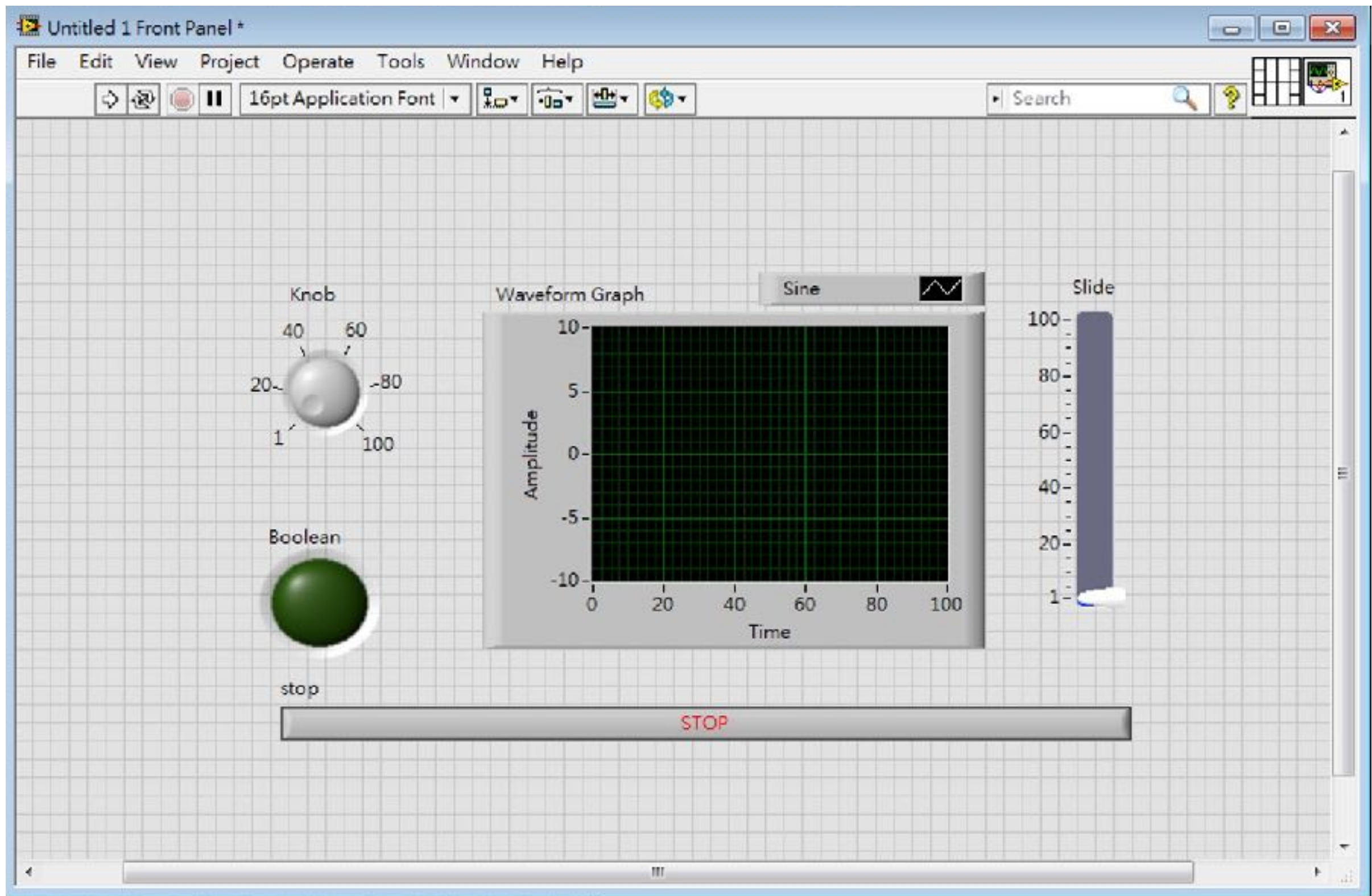


String (字串)

- 字串資料類型為一串ASCII字元
- 使用字串控制元件接收使用者輸入文字，如名字或密碼。
- 使用字串顯示元件和文字給使用者

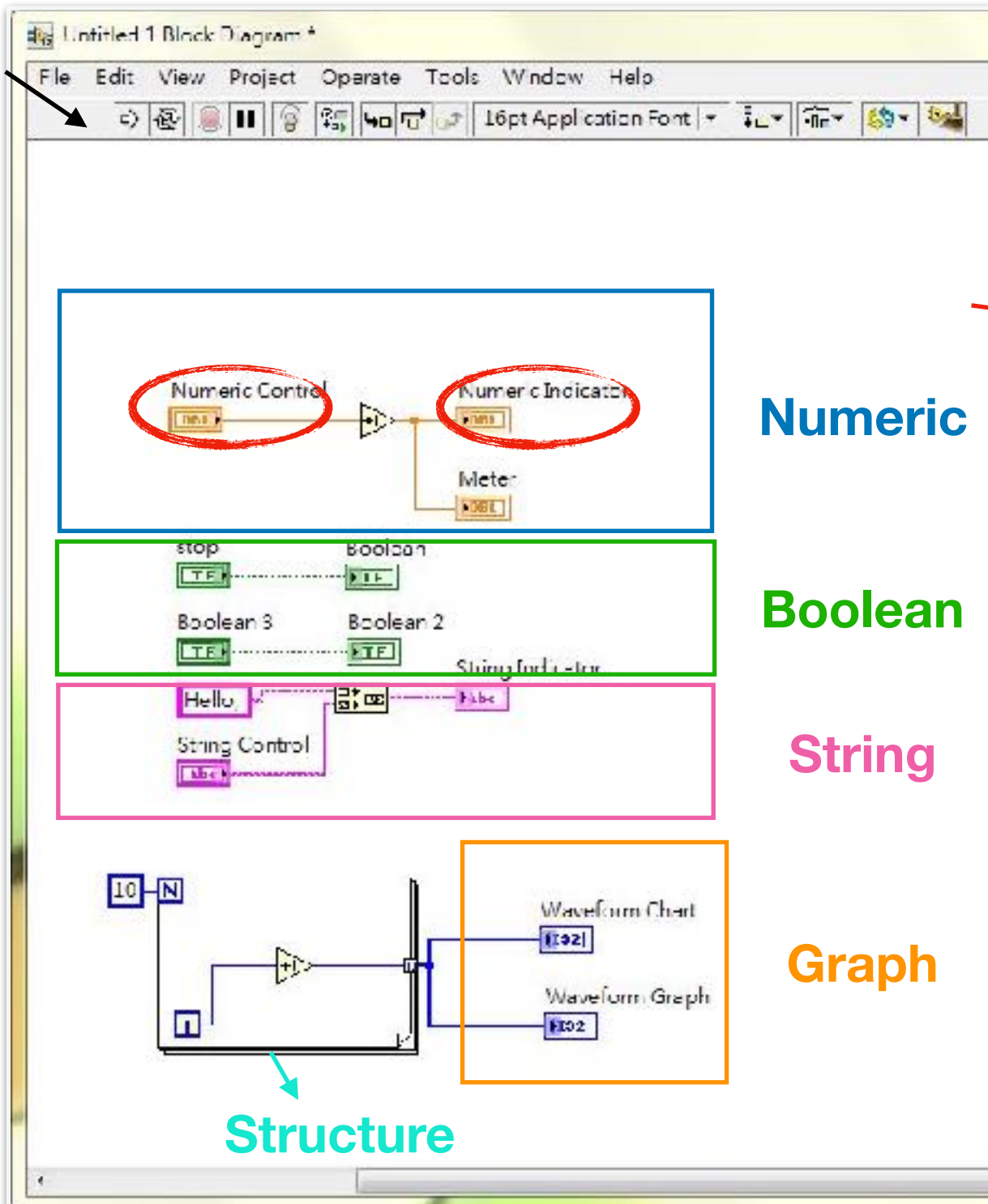


練習一-建立Front Panel

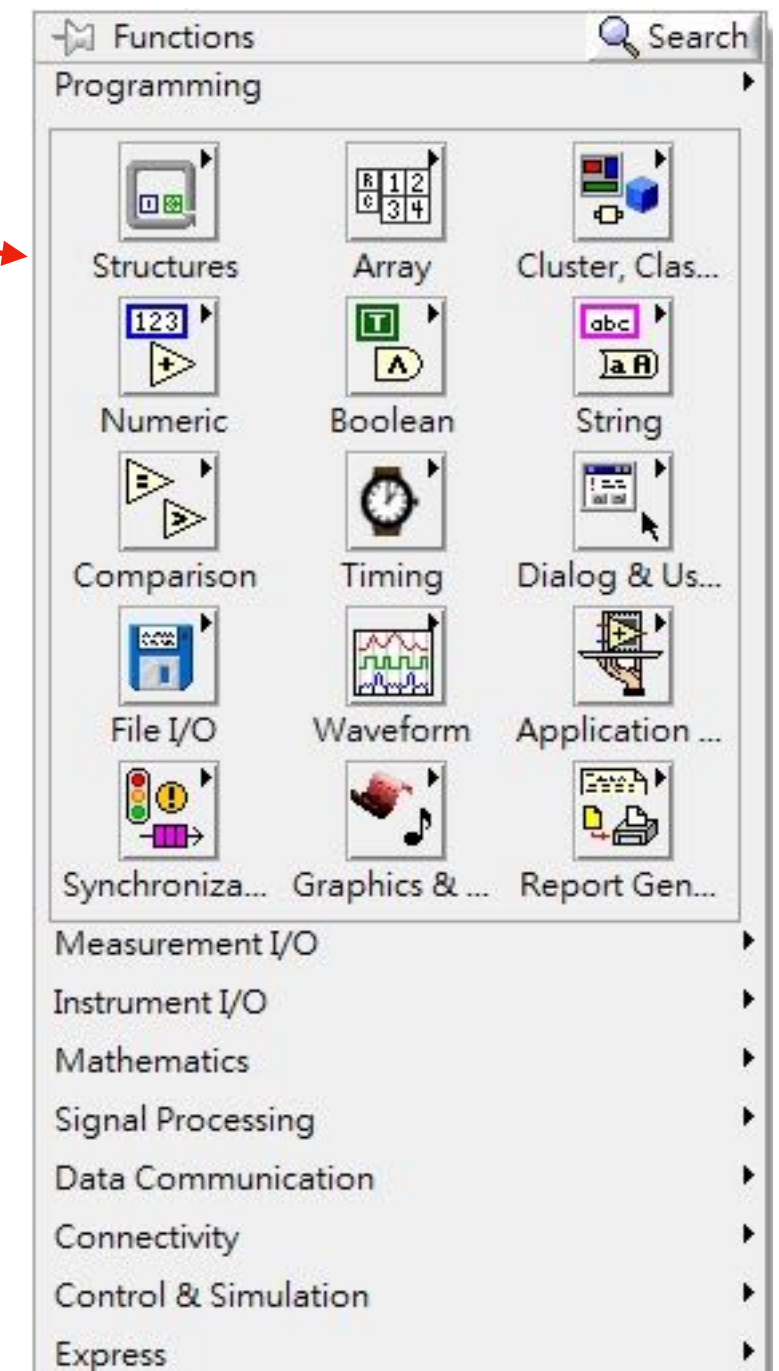


Block Diagram 程式區

Tool bar



Function 函數面板

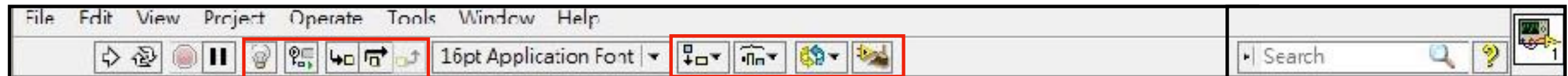


在空白處
滑鼠右鍵

工具欄

- 程式區的工具欄

整線工具



- Run : 執行
- Run Continuously : 連續執行
- Abort Execution : 停止
- Pause : 停止

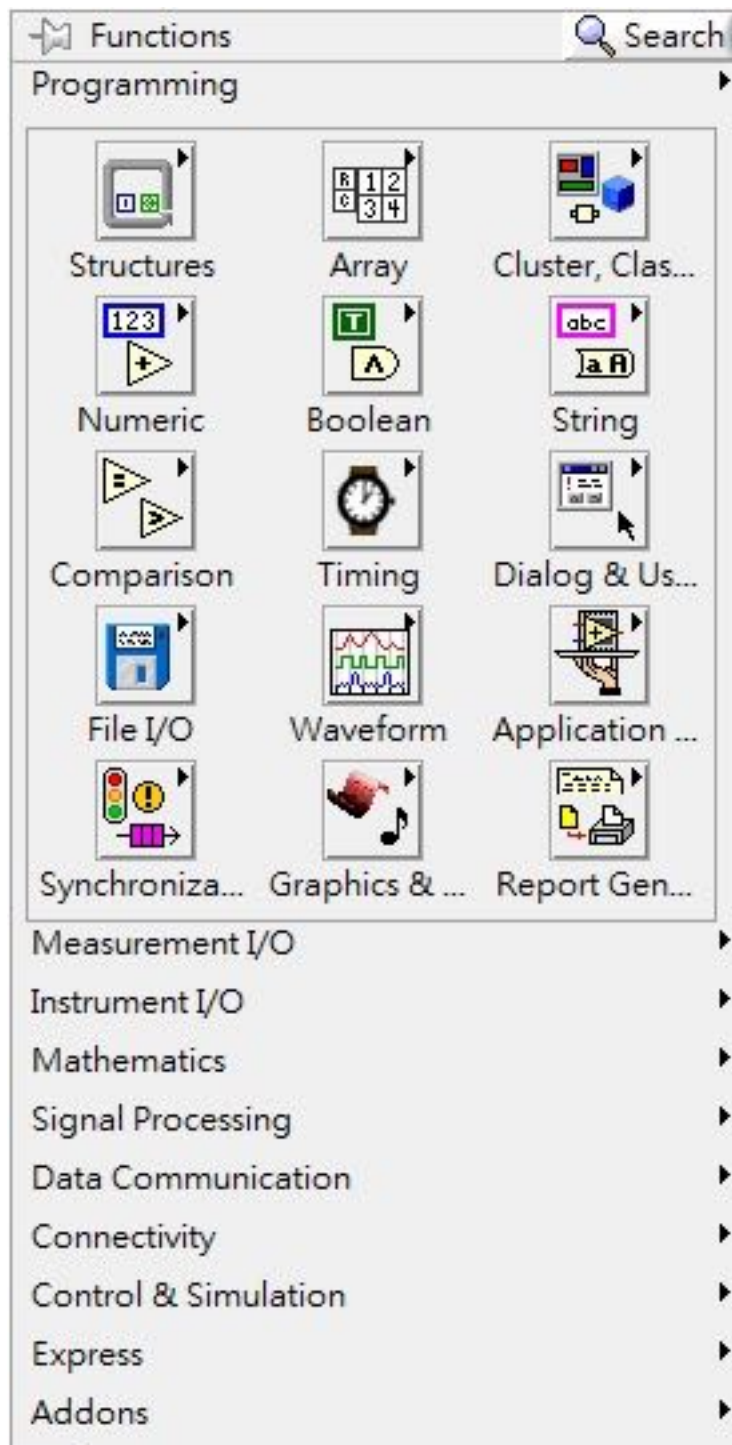
除錯工具



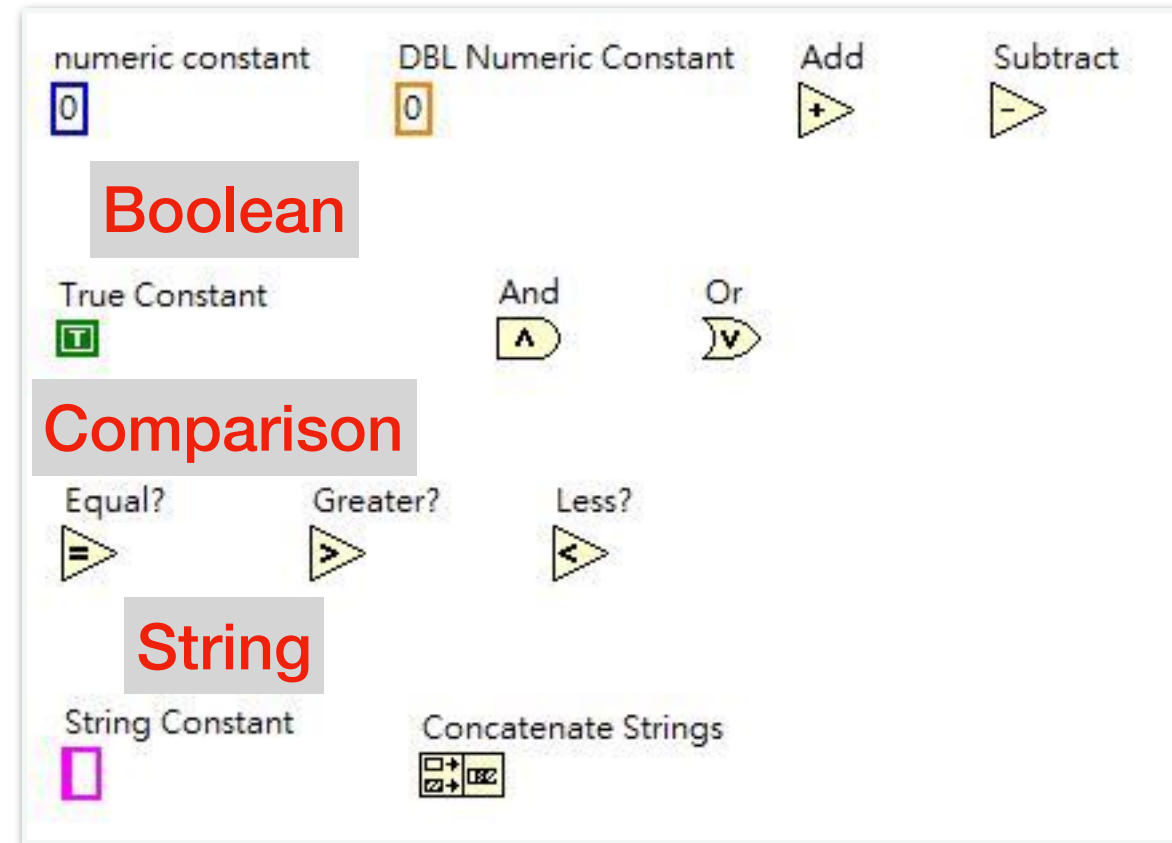
- Execution Highlighting button : 執行資料流動畫
- Retain Wire Values : 留住資料流的數值
- Step Into button : 可開啟一個節點(node)並暫停
- Step Over button : 執行一個節點，在下一個節點暫停
- Step Out button : 完成執行目前的節點並暫停

Functions Palette

Function 函數面板



Numeric



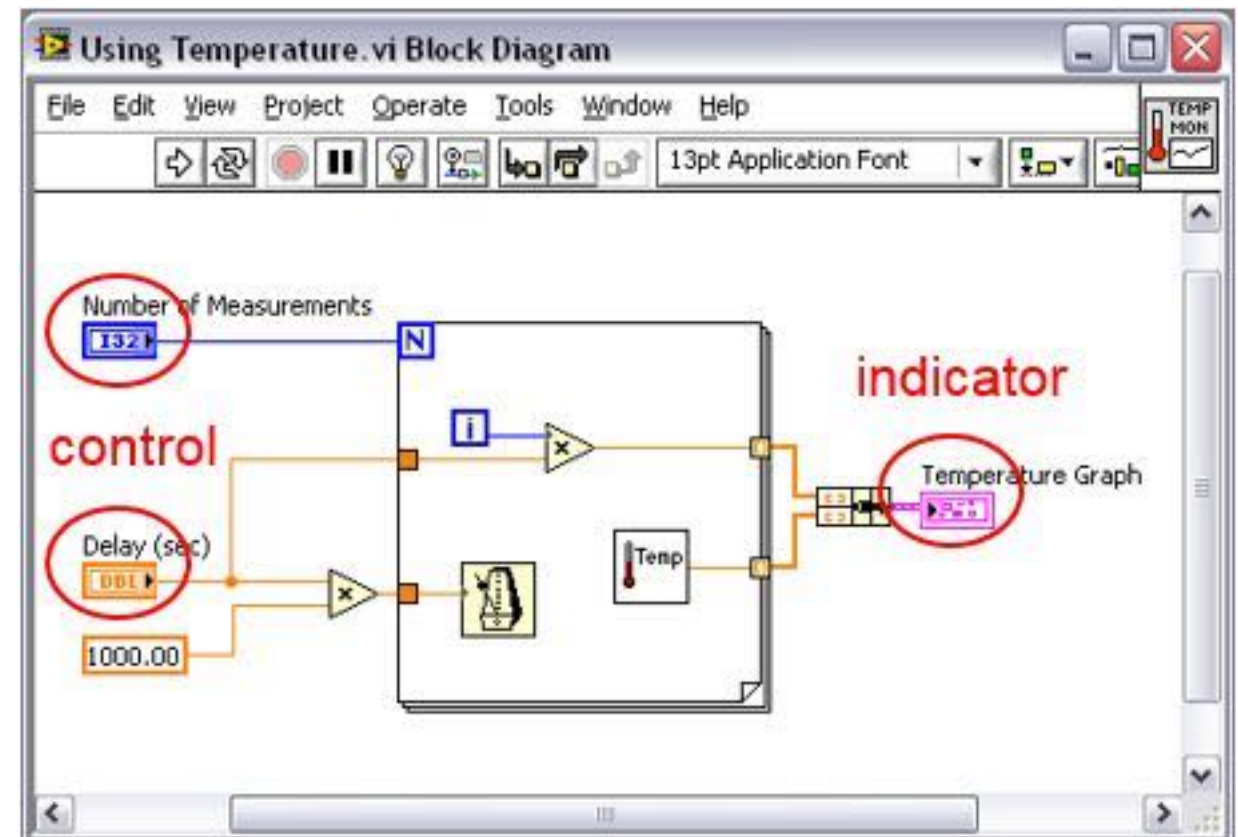
Boolean

Comparison

String

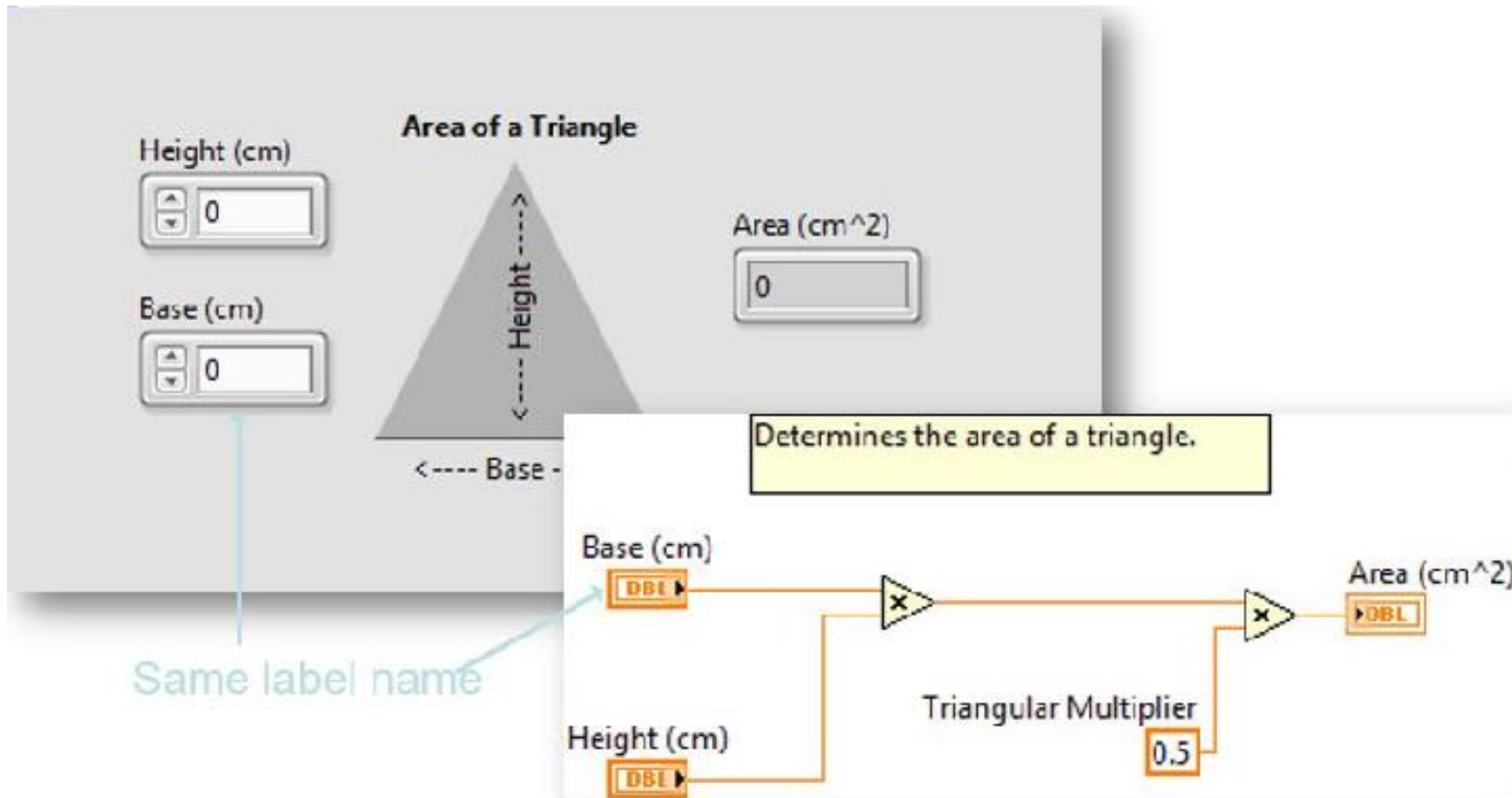
Block Diagram

- Terminal(接點)
 - Control & Indicator
- Constant(常數)
- Node(節點)
 - SubVI(副程式)
 - Function(函數)
 - Structure(結構)
- Wires線 (斷線、顏色、Coercion dot)



Terminals

View As Icon



Numeric icon

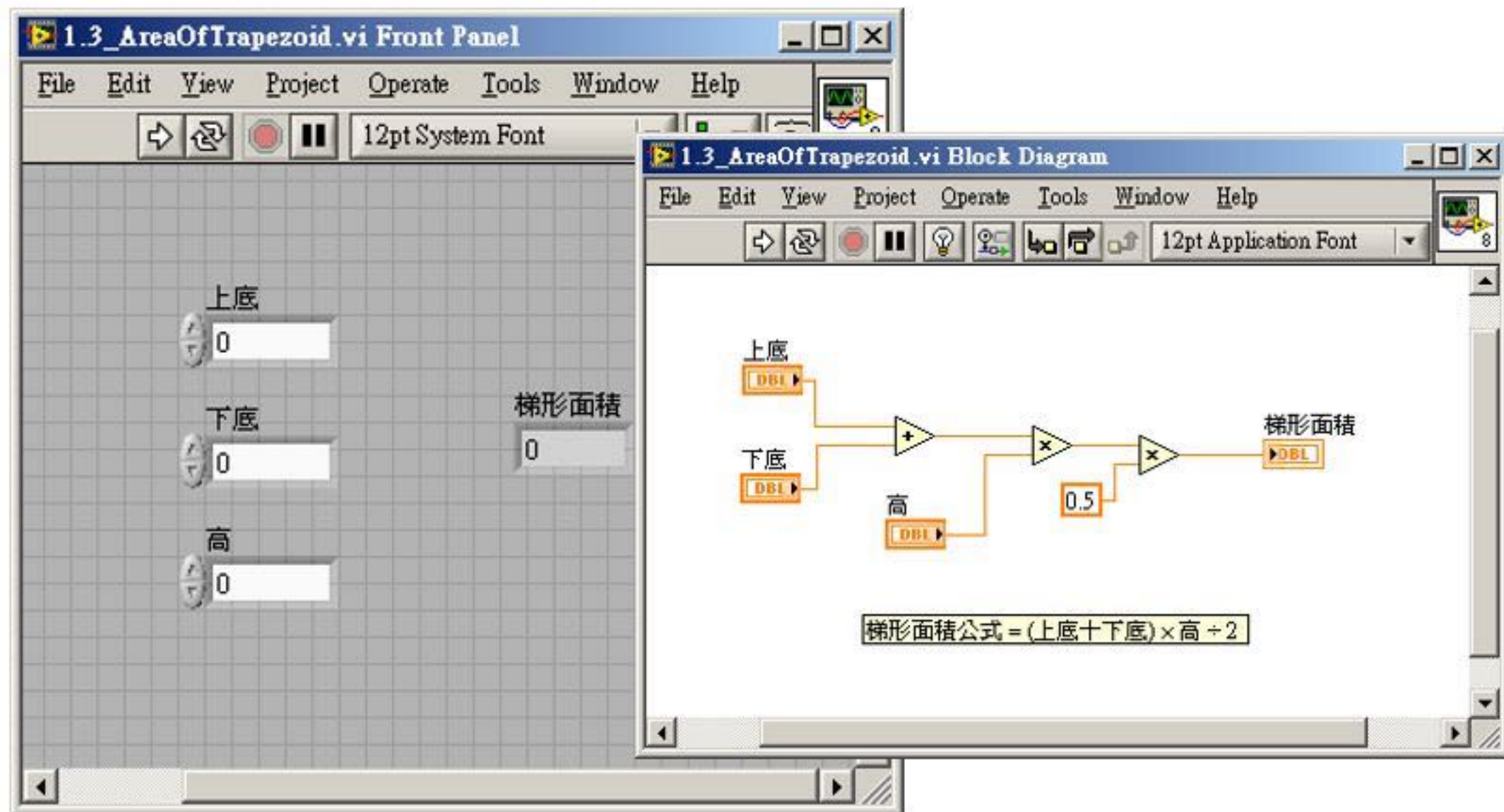
- Visible Items
- Find Control
- Make Type Def.
- Hide Control
- Change to Indicator
- Change to Array
- Change to Cursor L
- Description and Tip...
- Numeric Palette
- Create
- Data Operations
- Advanced
- View As Icon
- Representation
- Properties

Control / Indicator



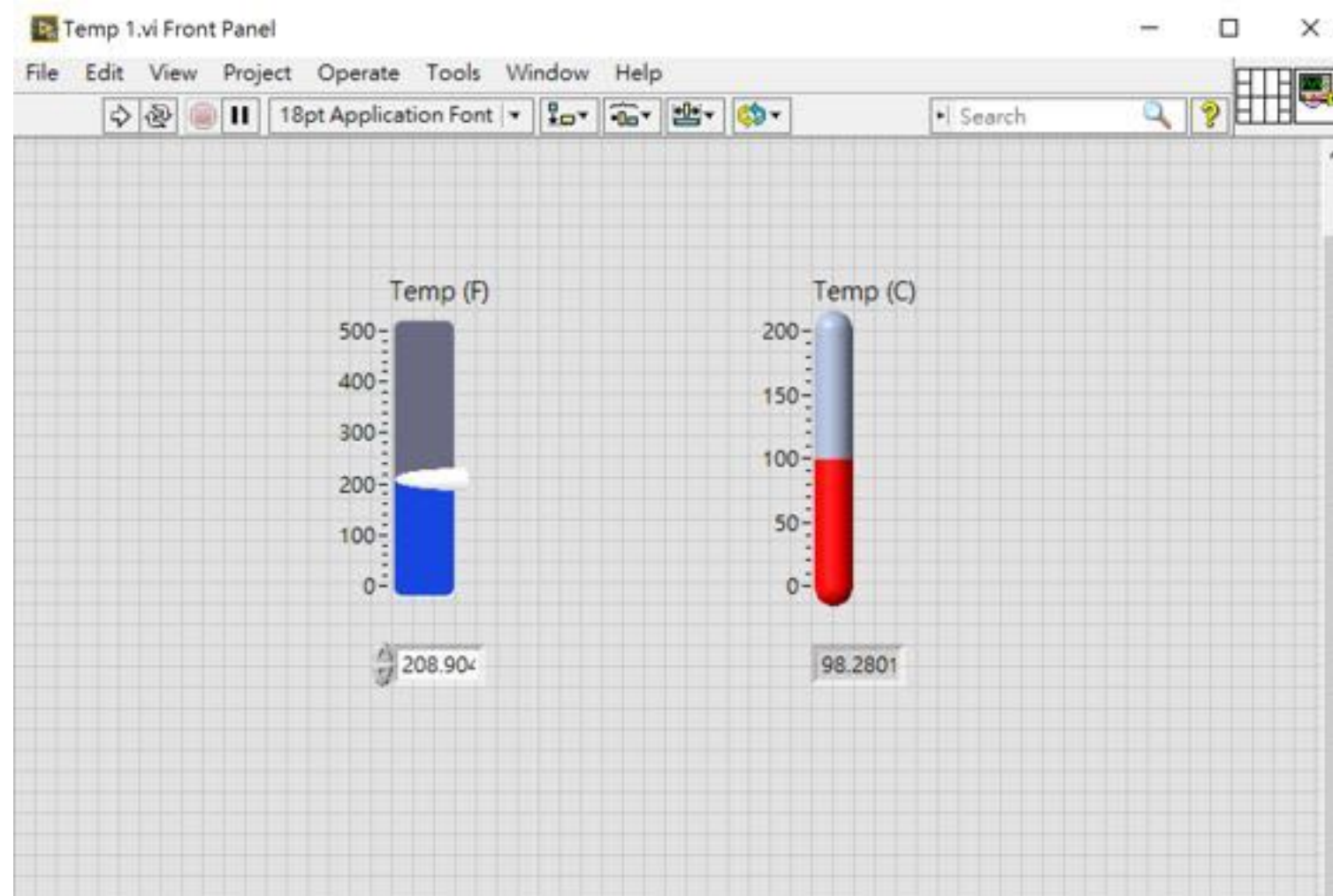
練習二-梯形面積計算

- 公式：(上底 + 下底) × 高 ÷ 2 = 梯形面積



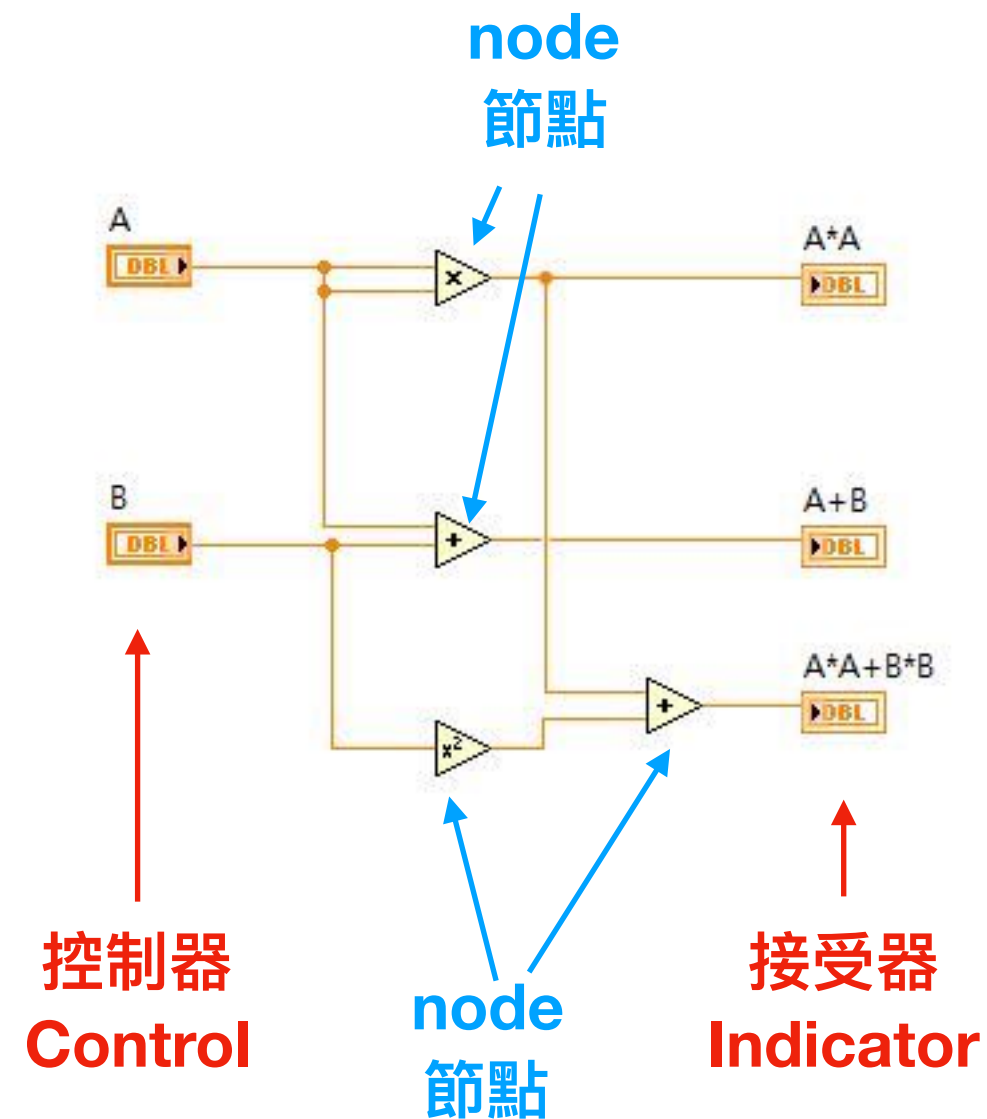
練習三-溫度計算

- 華氏 = 攝氏 \times (9/5) + 32
- 攝氏 = (華氏 - 32) \times 5/9



Node 節點

- Nodes是一種物件，他有Input和Output，當VI執行時，有運算的功能。
- Function Nodes 呈現淡黃色，為基本LabVIEW內基本要件，連點兩下無法像VI或subVI一樣被打開，看到程式碼，為最底層程式。



Wires 資料線

- 資料線種類：不同的資料型態有不同的顏色與外觀

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

- 連線技巧

>>		點滑鼠左鍵將兩接點連接
>>		按空白鍵
>>		點一至三下不同效果

所有資料型態

整數

Signed integers
(0)

- I8** 8-bit
- I16** 16-bit
- I32** 32-bit
- I64** 64-bit

Unsigned integers
(0)

- U8** 8-bit
- U16** 16-bit
- U32** 32-bit
- U64** 64-bit

Floating-point
(0.0)

- SGL** Single
- DBL** Double
- EXT** Extended

Complex floating-point
(0.0 + i0.0)

- CSG** Single
- CDB** Double
- CXT** Extended

浮點數

布林

Boolean
(FALSE)

String
(empty string)

字串

Path
(<Not A Path>)

Reference number

Enumerated type

Cluster

- Numeric
- Mixed data

Array

- 1D
- 2D

陣列

Waveform

Digital waveform

Digital

I/O name

Variant

Dynamic

Picture

Time stamp
(12:00 a.m.,
1/1/1904,
Universal Time)

標量

一維陣列

二維陣列

整數



藍

浮點數



橙

布林



綠

字串



粉

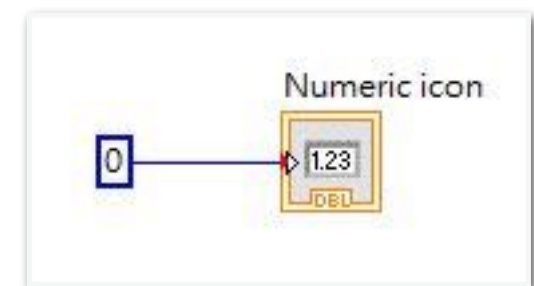
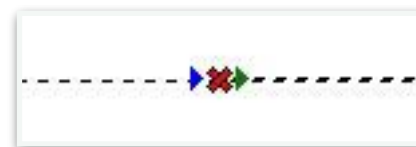
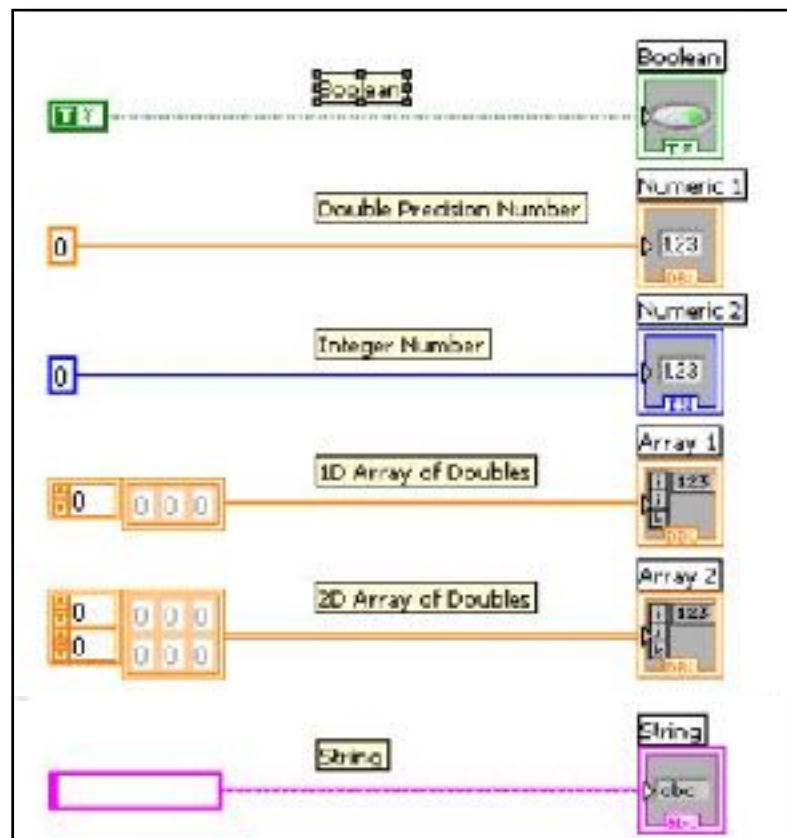
檔案路徑



青

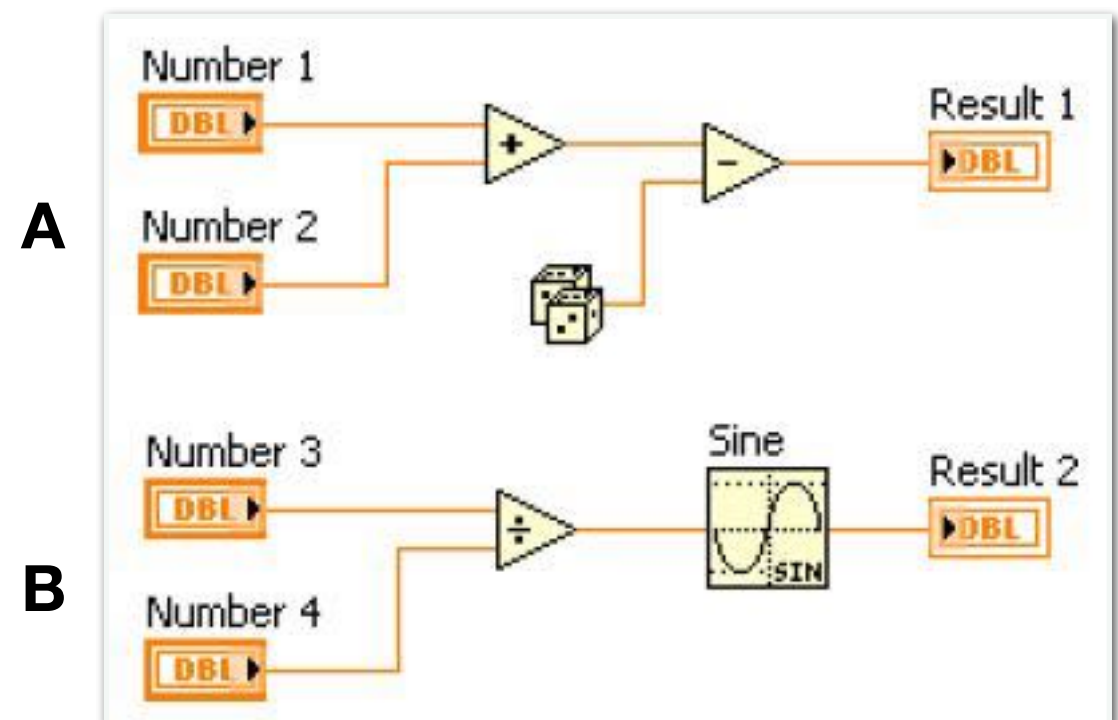
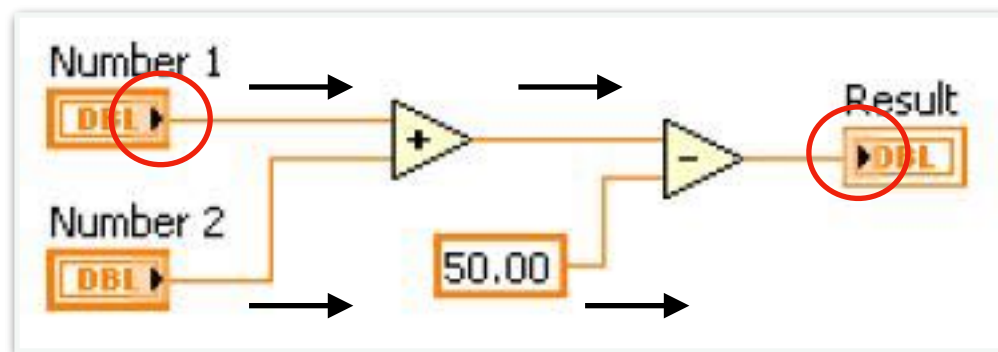
Wires 資料線

- 正常連線：接線傳送物件之間的資料。
- 斷線：資料型態不同可能會造成斷線。
Ctrl + B : 可清除斷線。
- Coercion dot : 不一樣的整數型態會造成紅點的出現。



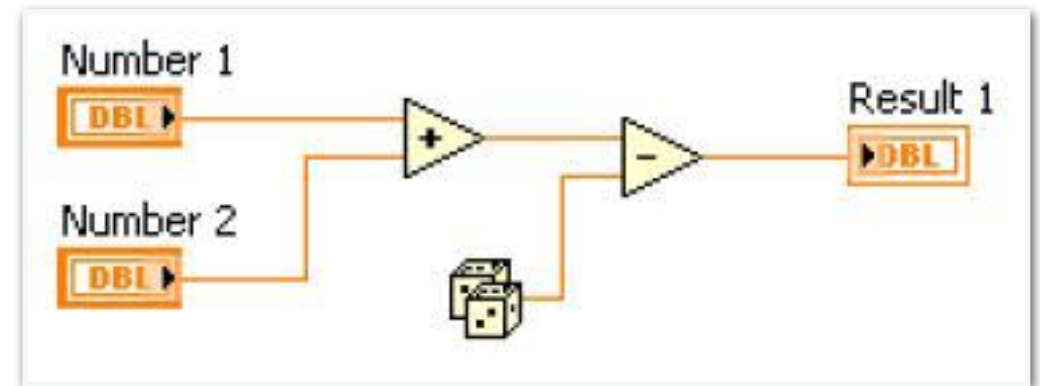
Data flow 資料流

- LabVIEW是資料導向的程式，資料跑到哪就執行到哪裡。
- 任何節點必須等到所有輸入進來後才會執行。
- 可執行平行處理。（誰先執行不知道。）



Quiz1

- Which function executes first: Add or Subtract?
 - A. Add
 - B. Subtract
 - C. Unknown



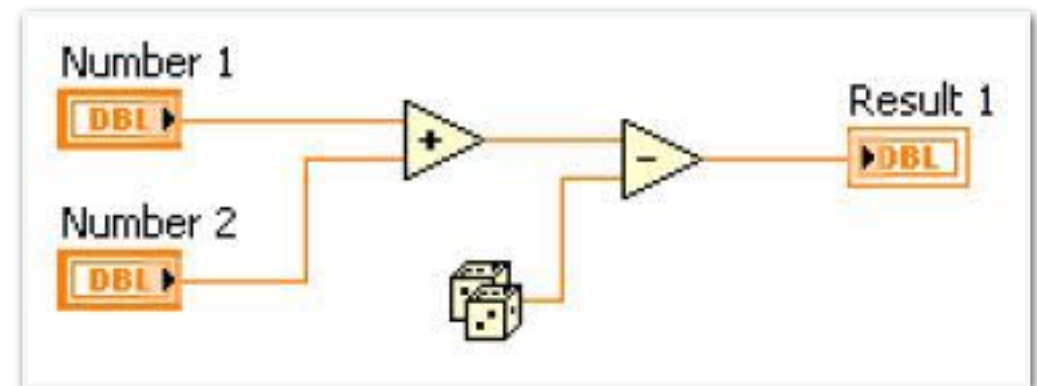
Quiz1-Ans

- Which function executes first: Add or Subtract?

A. Add

B. Subtract

C. Unknown



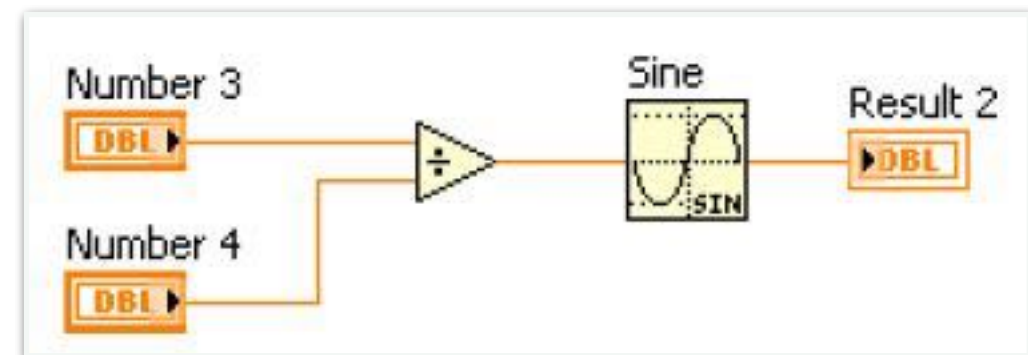
Quiz2

- Which function executes first: Sine or Divide?

A. Sine

B. Divide

C. Unknown



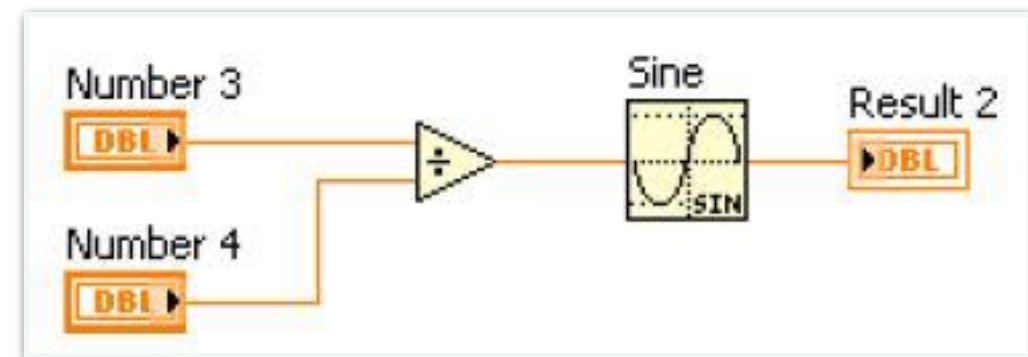
Quiz2-Ans

- Which function executes first: Sine or Divide?

A. Sine

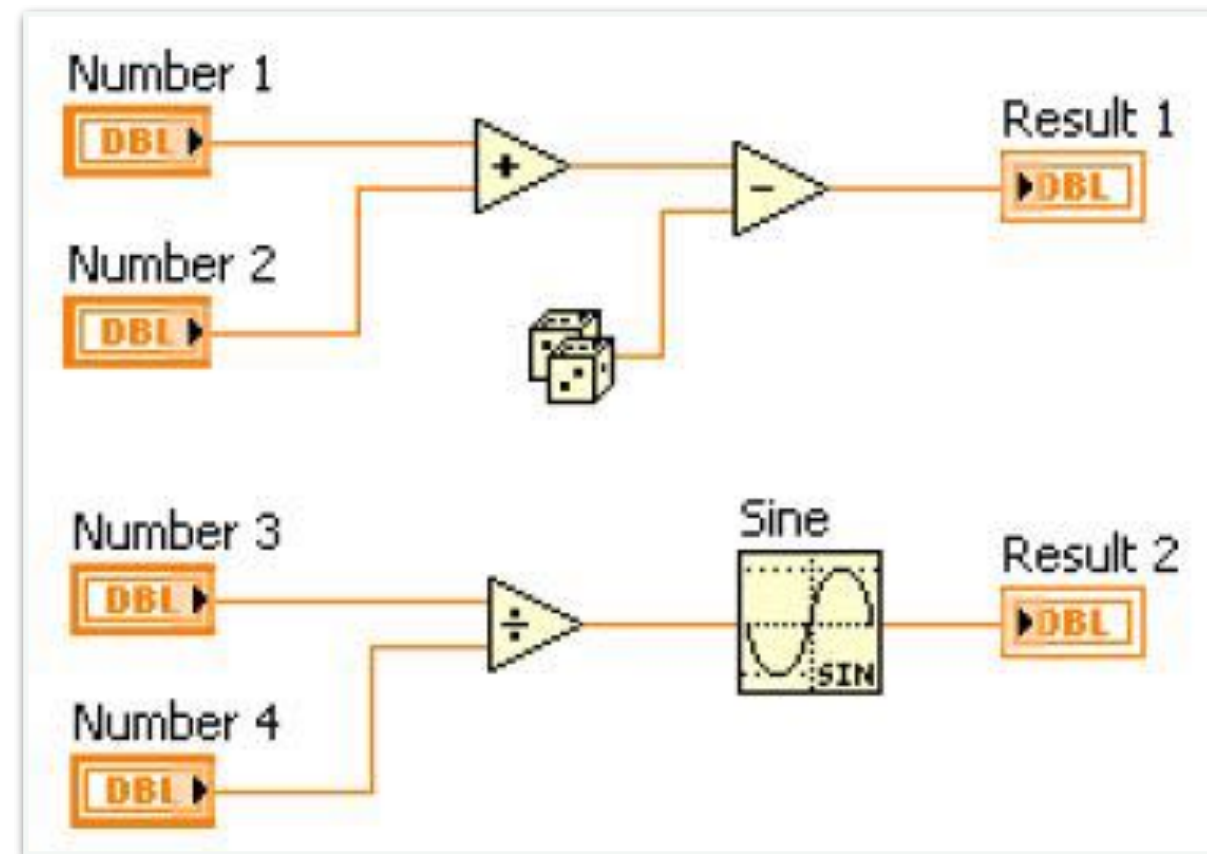
B. Divide

C. Unknown



Quiz3

- Which node executes first?
 - A. Add
 - B. Subtract
 - C. Random Number
 - D. Divide
 - E. Sine



Quiz3-Ans

- Which node executes first? **(No single correct answer)**

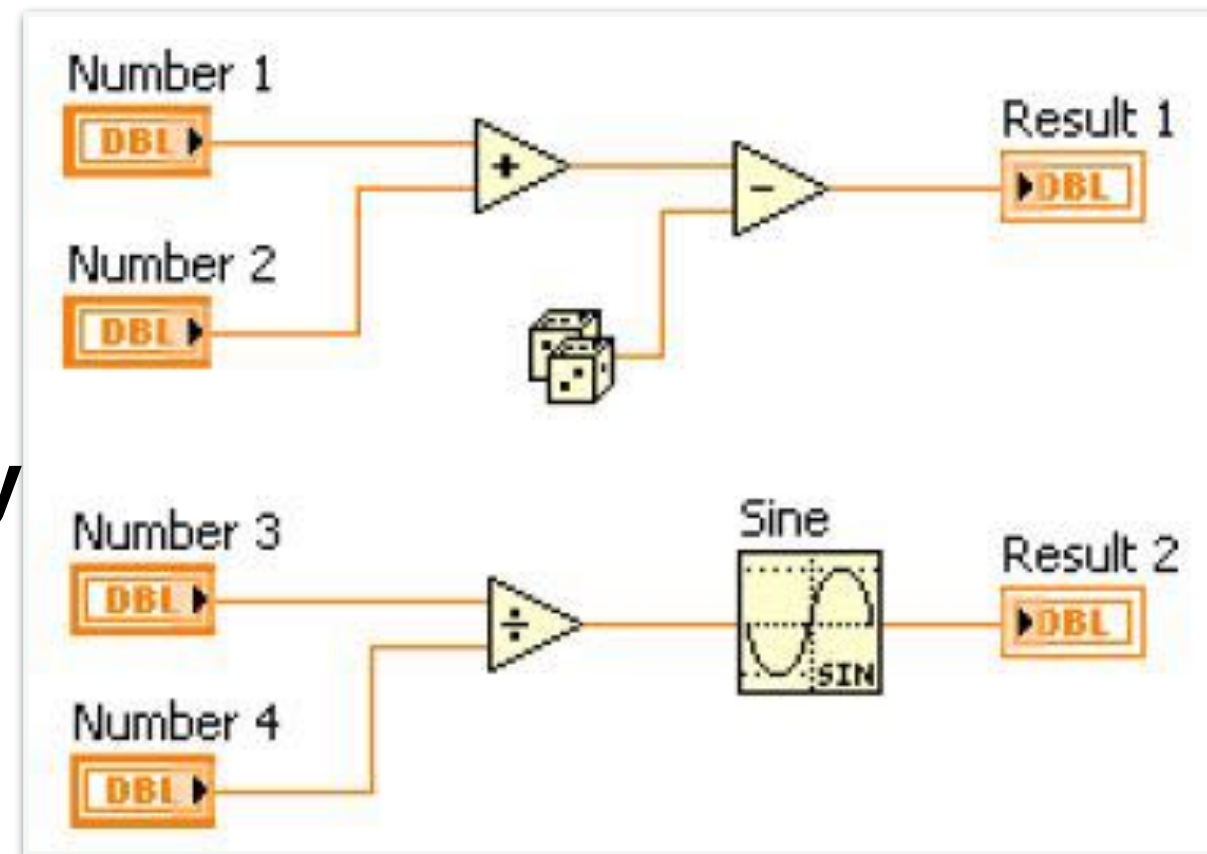
A. Add -**Possibly**

B. Subtract -**Definitely not**

C. Random Number -**Possibly**

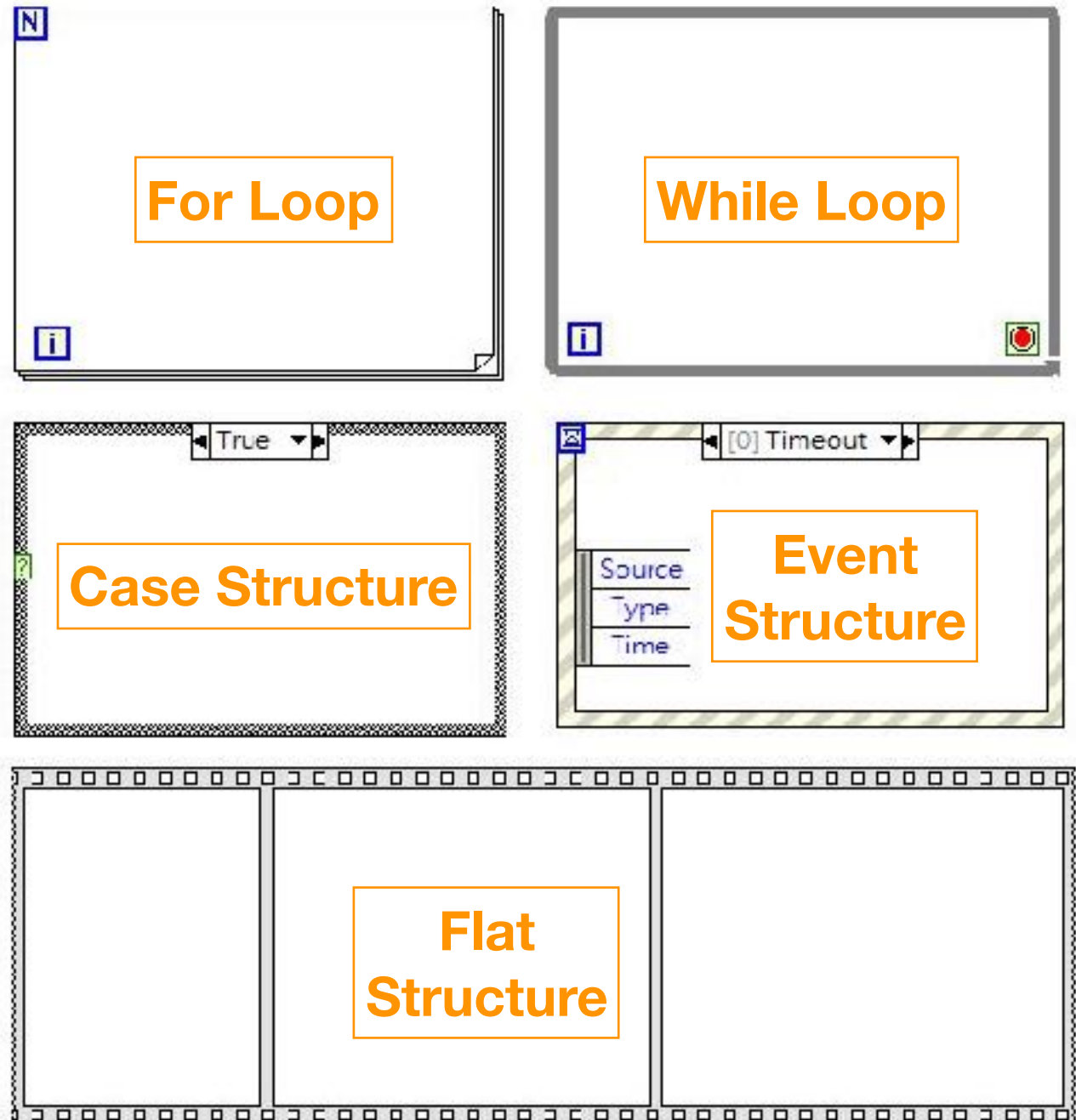
D. Divide -**Possibly**

E. Sine -**Definitely not**



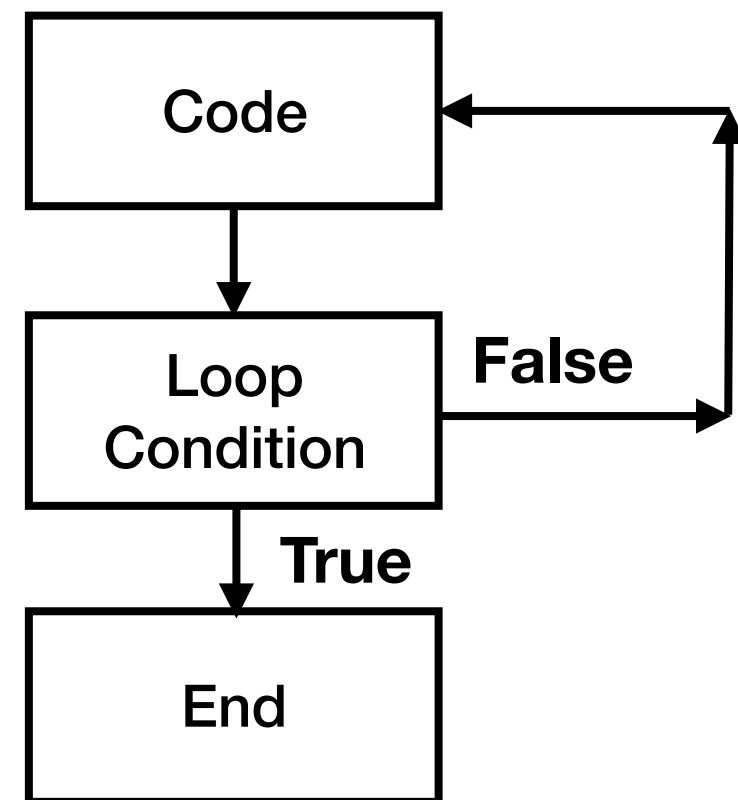
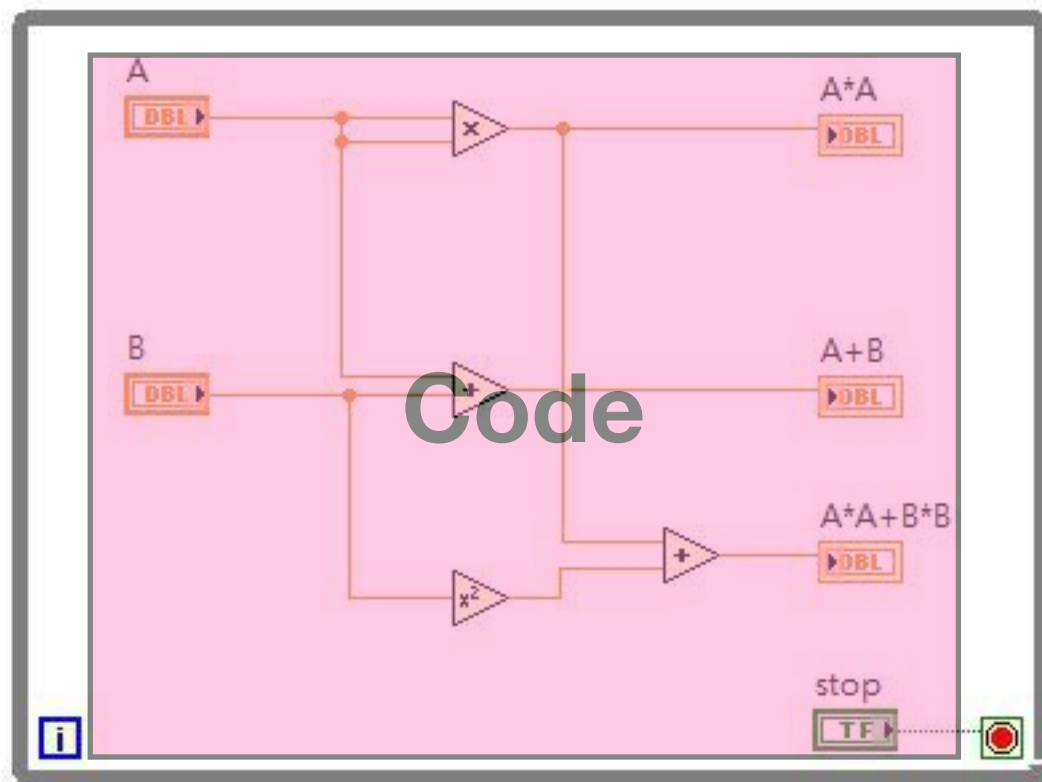
Structure 結構

- For Loop
- While Loop
- Case Structure



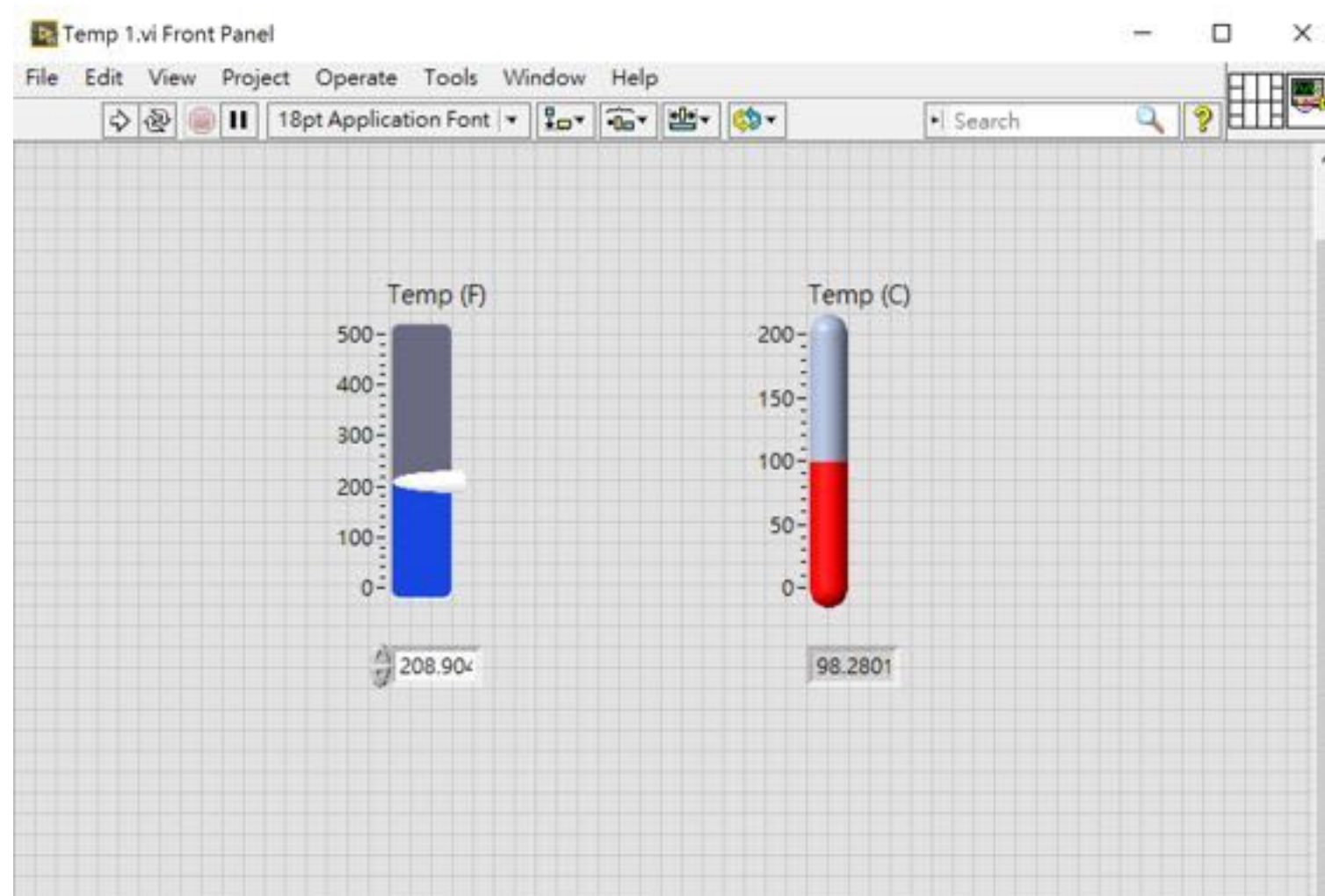
While Loop

- Structure >> While Loop



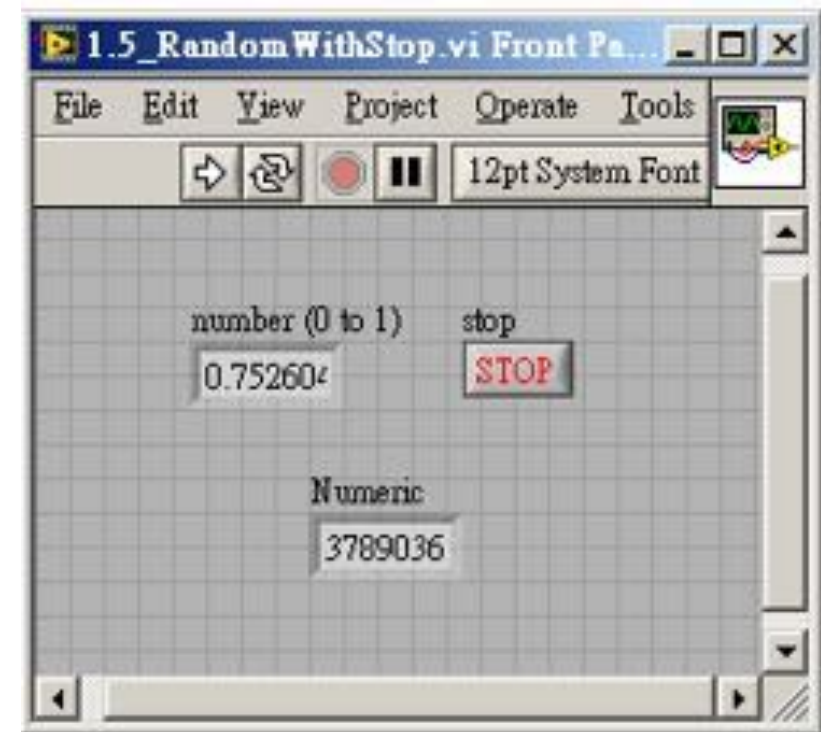
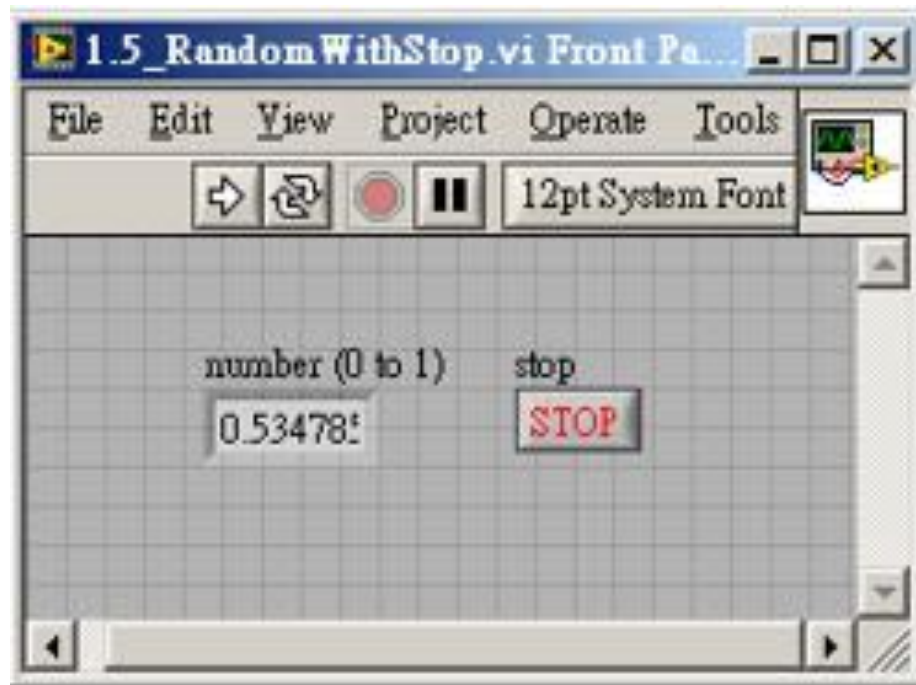
練習四-溫度計算 (即時更新)

- 華氏 = 攝氏 $\times (9/5) + 32$
- 攝氏 = (華氏 - 32) $\times 5/9$



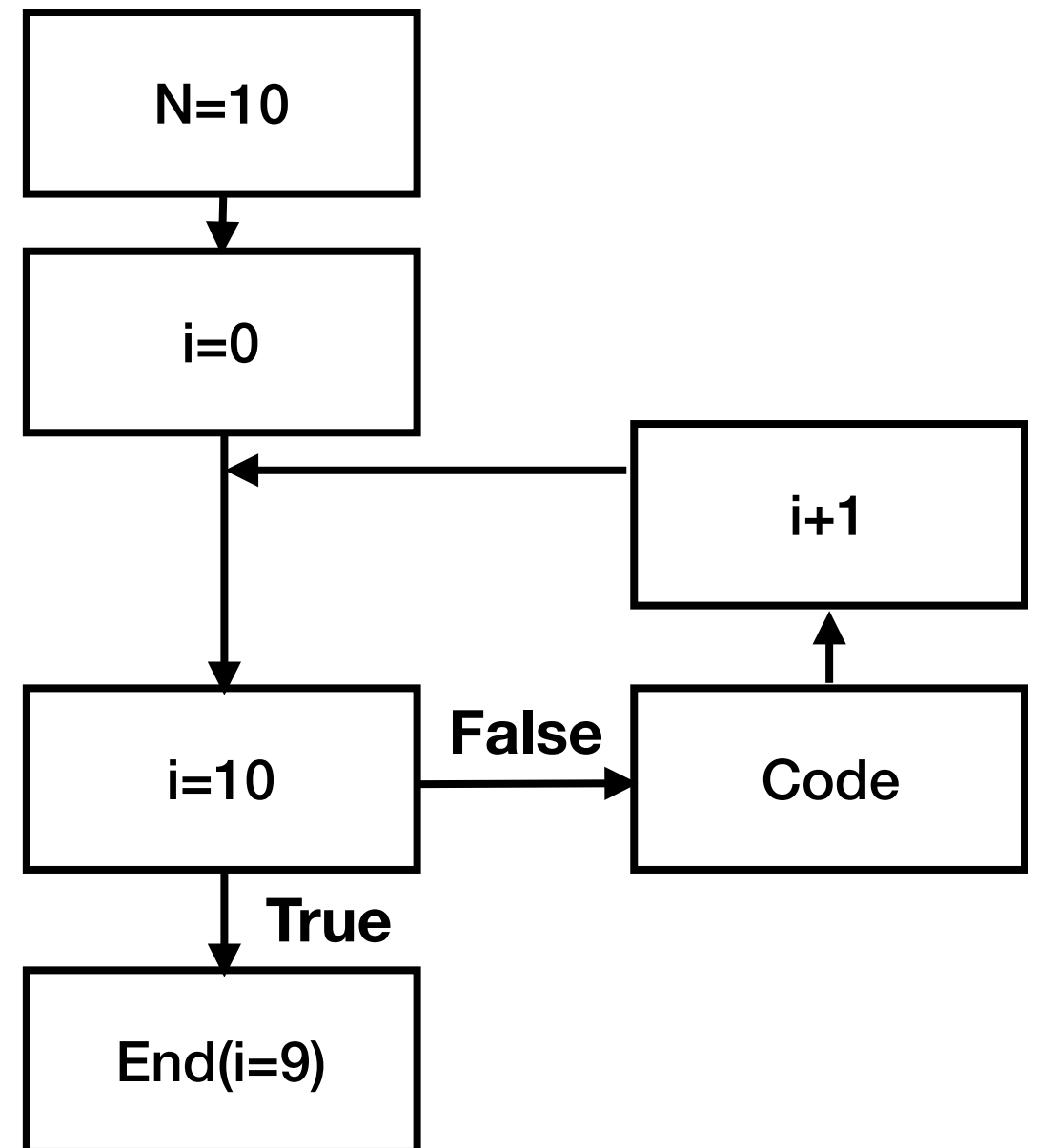
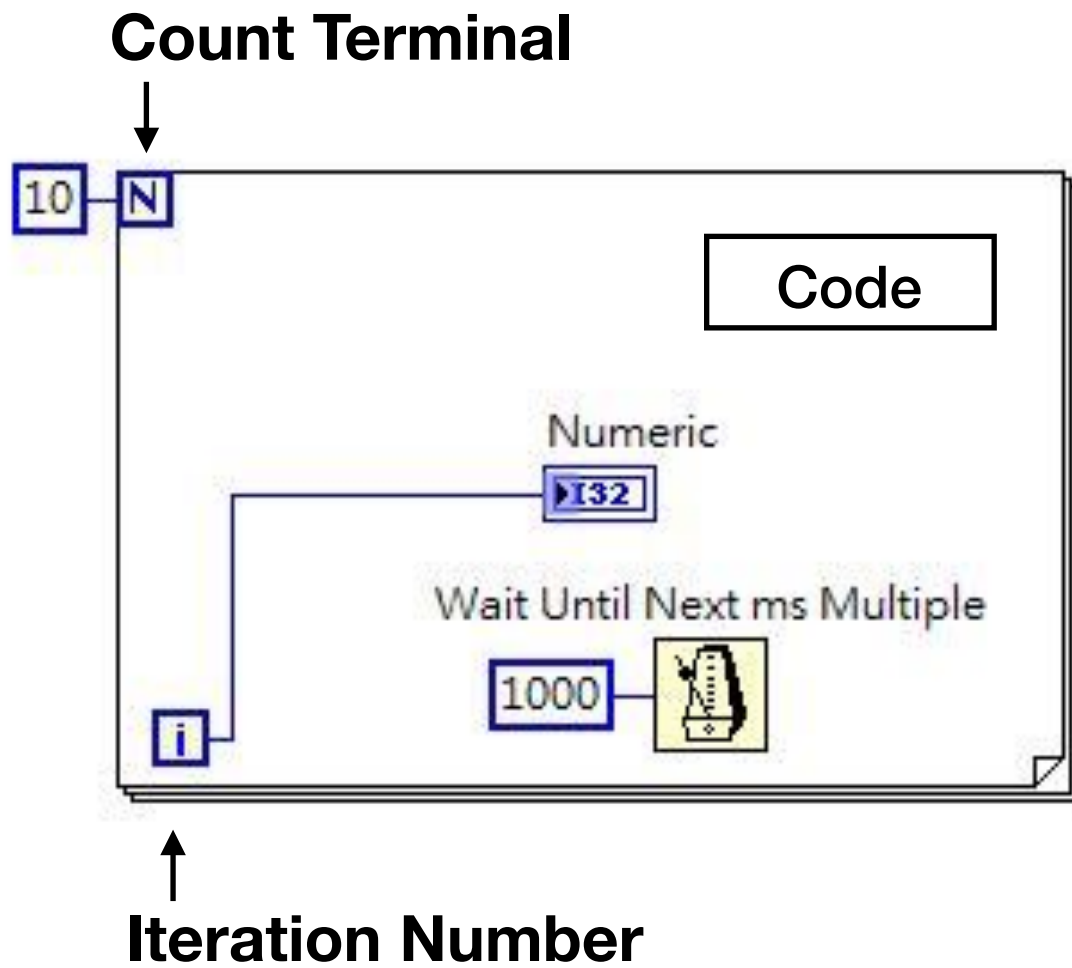
練習五-亂數產生(While-Loop)

- 程式不斷產生出0~1之間的亂數，當使用者按下停止鍵停止產生了多少個亂數？



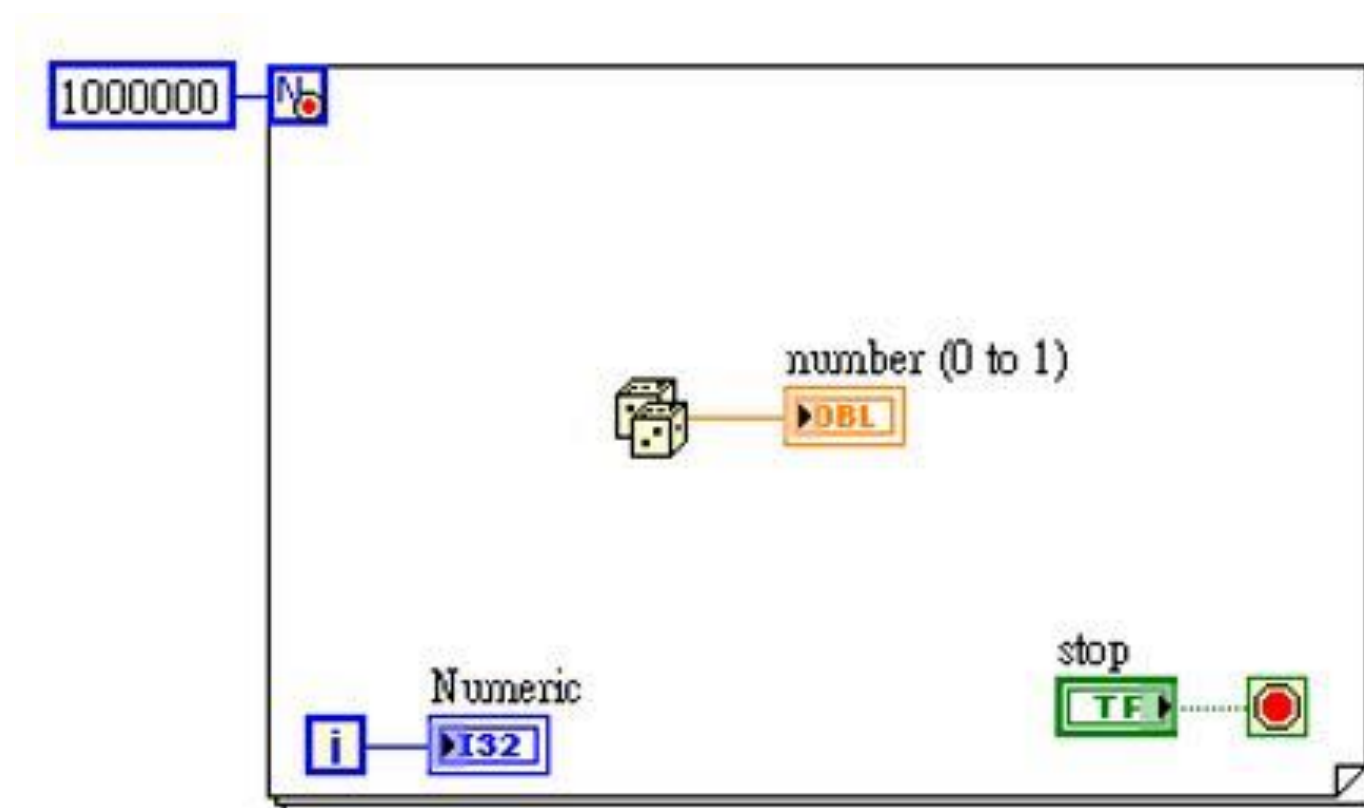
For Loop

- Structure >> For Loop

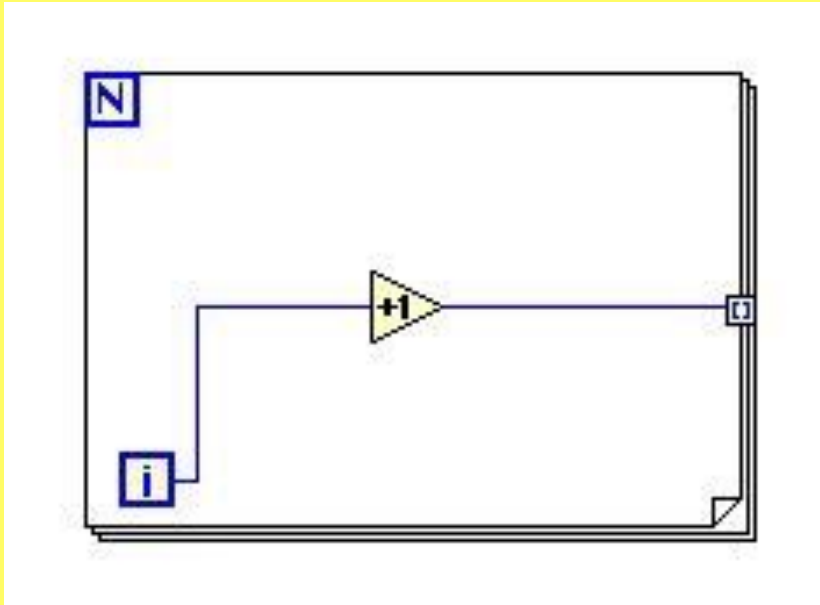
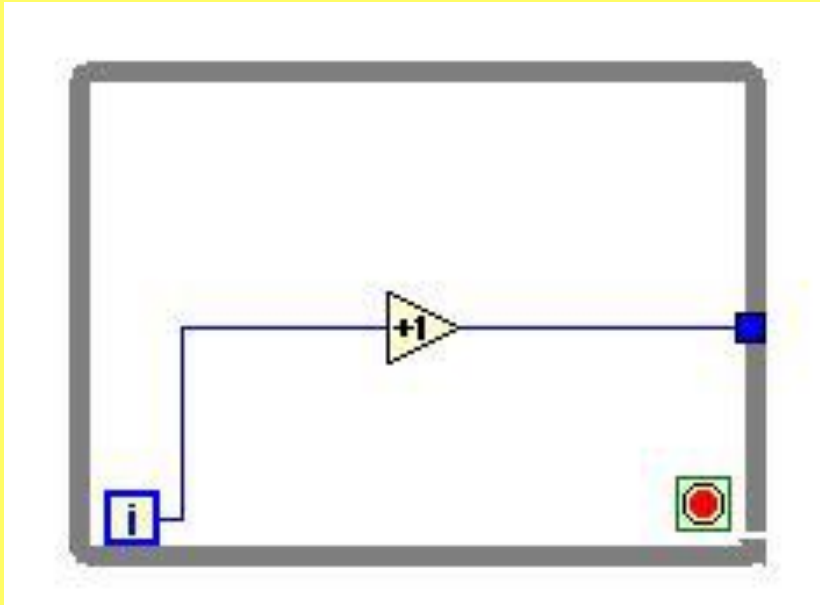


練習六-亂數產生(For-Loop)

- 將上一個練習改為用For做，得到第1,000,000個亂數時停止。



For Loop & While Loop

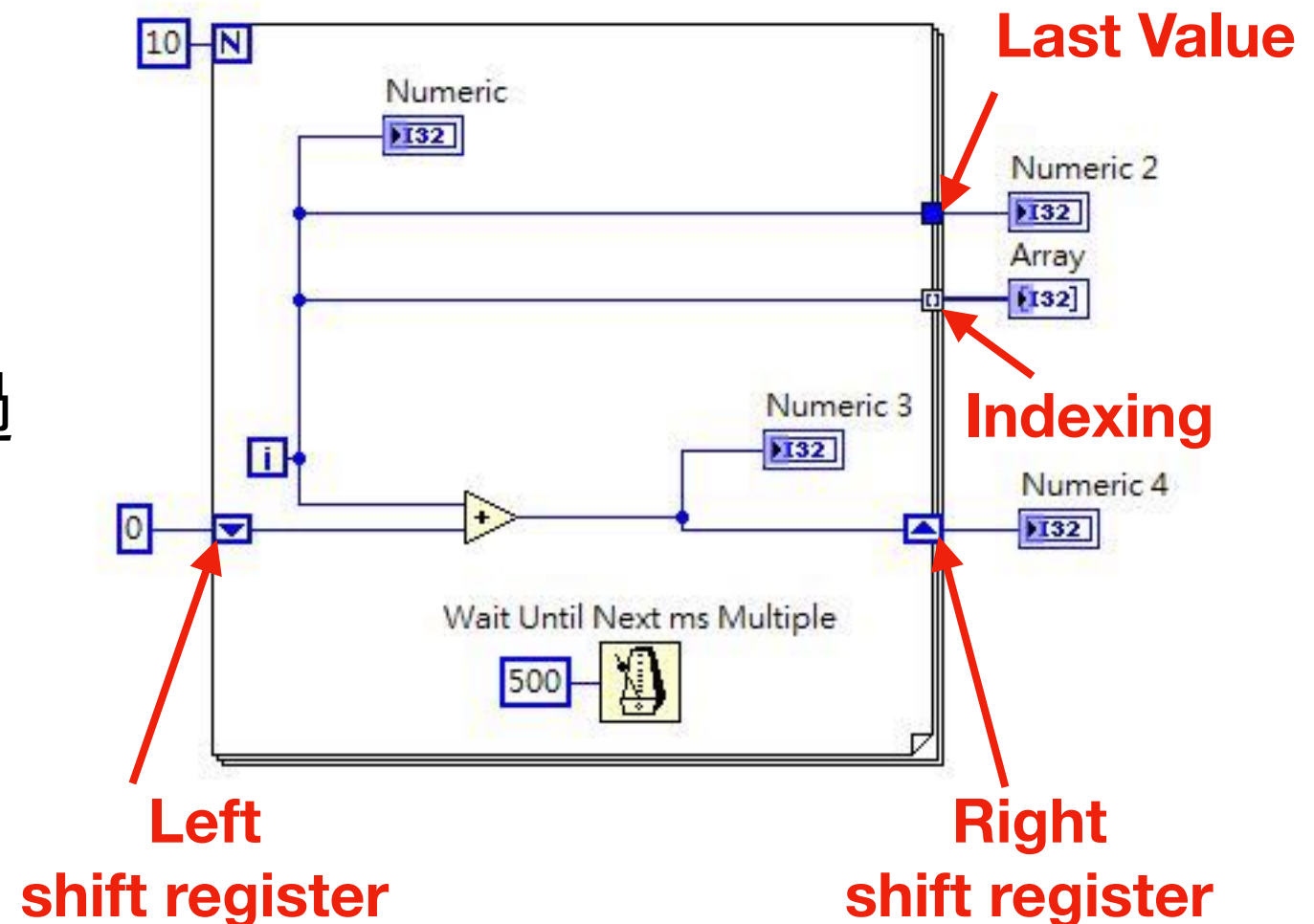
	
For Loop	While Loop
內部程式最少執行0次	內部程式最少執行1次
已知執行次數	執行次數未知，但可設定條件停止
通道的輸出自動設為陣列資料 Tunnel Mode > Indexing	通道的輸出自動為最後一個值 Tunnel Mode > Last Value

Tunnel Mode

- Shift Registers : 同一回圈內，左邊的shift register為上一圈計算的值，右邊的shift register為此圈計算後的值。

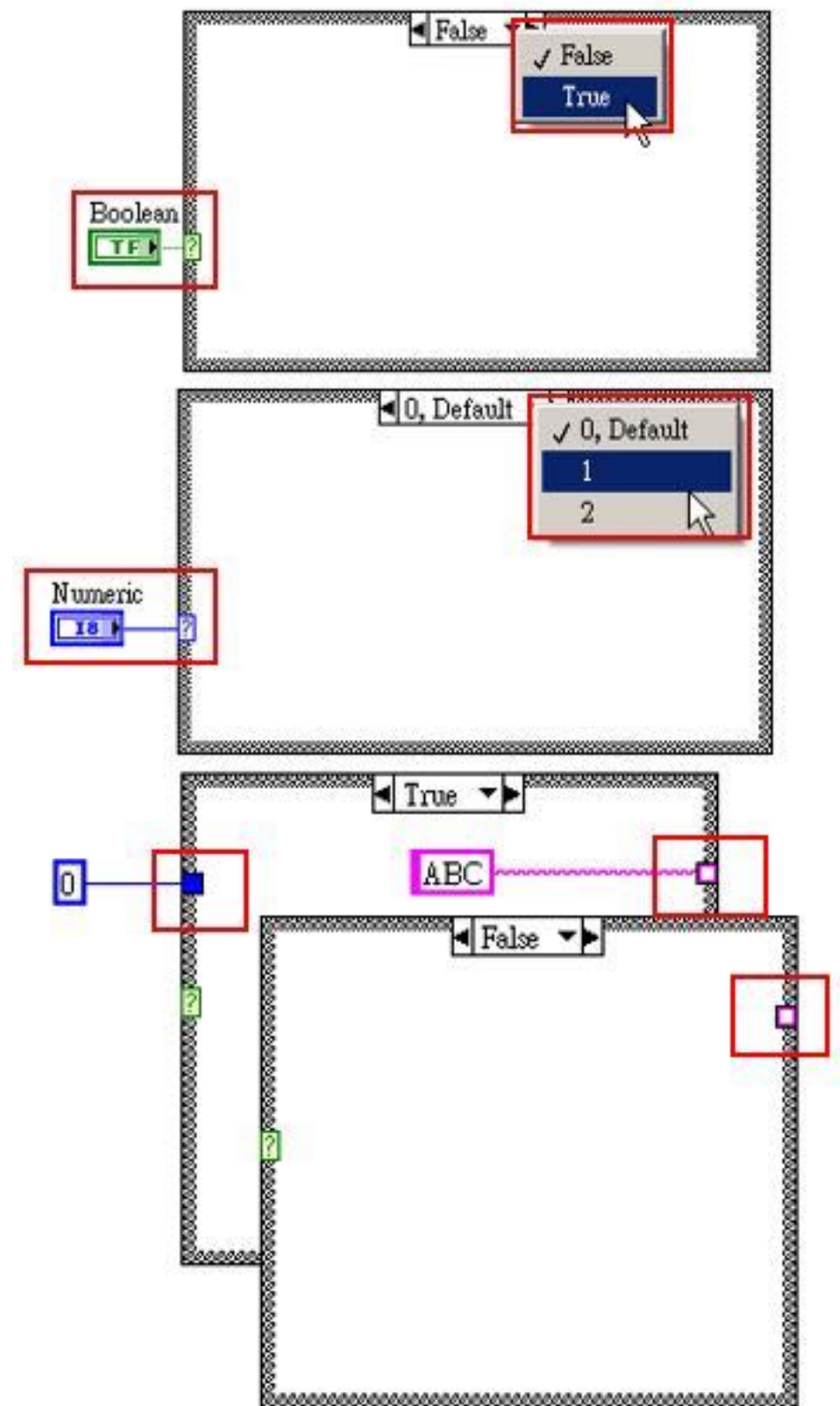
i	0	1	2	3	4	5	6	7	8	9
Left	0	0	1	3	6	10	15	21	28	36
Right	0	1	3	6	10	15	21	28	36	45

- Last Value : 迴圈執行後的最後一個值。
- Indexing : 迴圈執行間，經過通道時的值，形成陣列輸出。



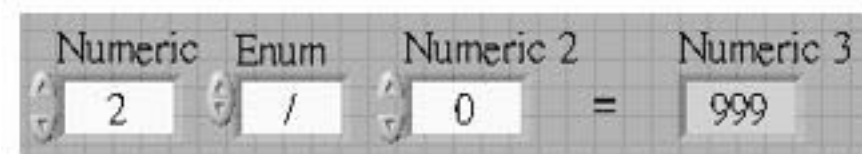
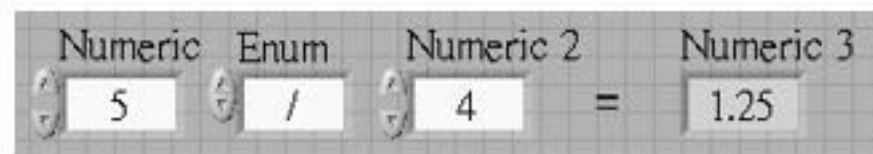
Case Structure

- Structure >> Case Structure
- 面對不同的情況，要讓程式做不同的動作。
- 一次只有一種可能（只能進入其中一種case）
- 不一定每個case都要使用輸入資料，但若有輸出，則必須為每個case定義各自的輸出。
- “？”可接受的格式Integer, Boolean, String, Enum



練習七-計算機

- 使用者可以選擇「加」、「減」、「乘」、「除」
- 若發生「除以零」的情形，則答案顯示出「999」



Timing

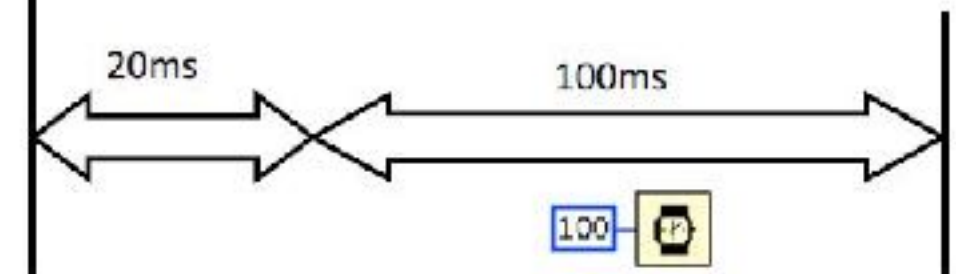
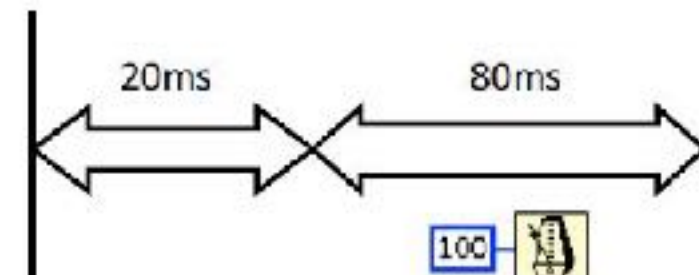
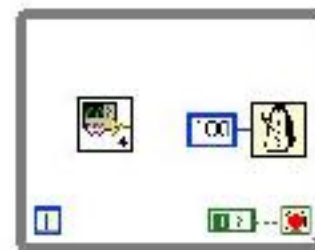
- Timing > Wait(ms)
- Timing > Wait Until Next ms Multiple



Wait (ms)

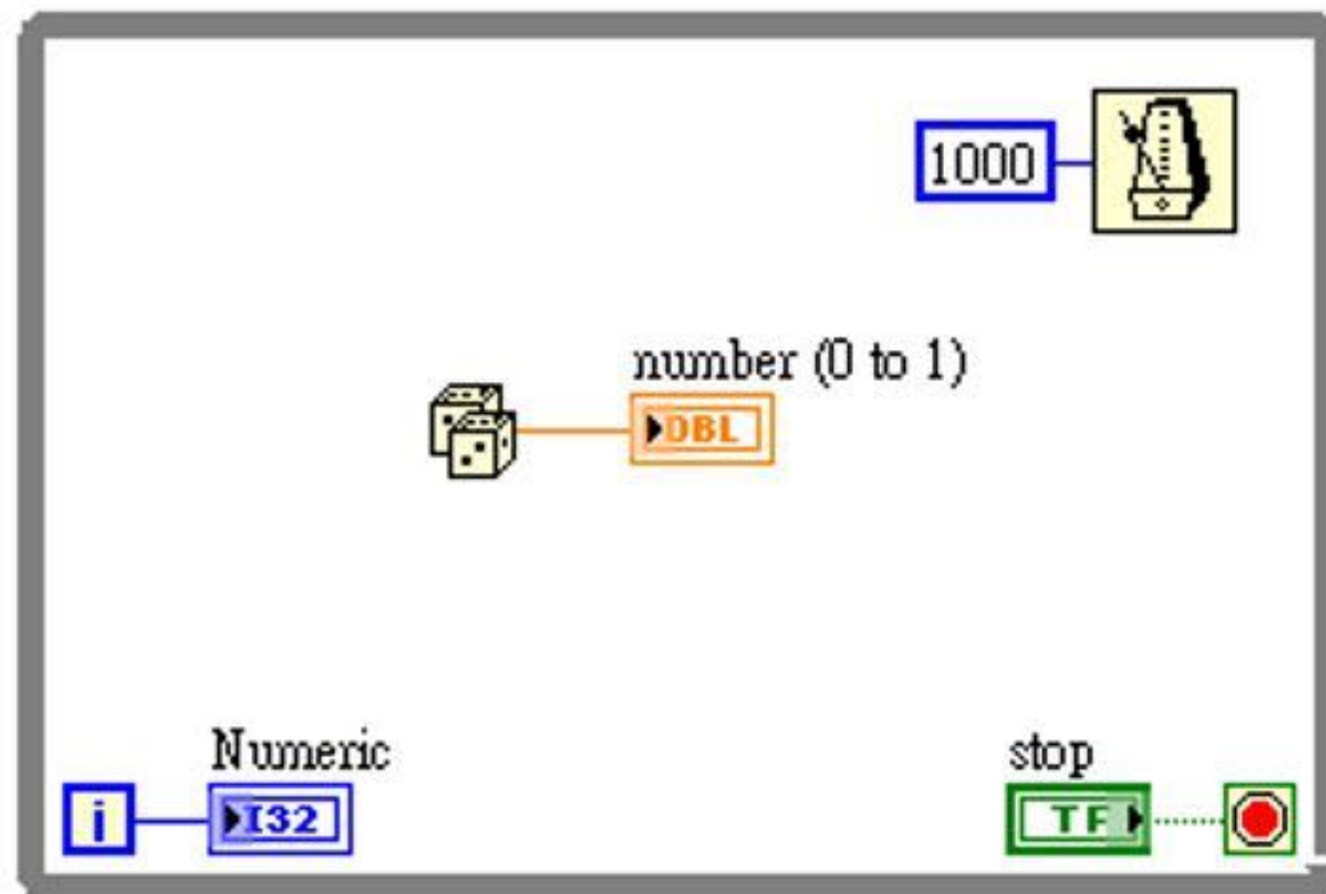


Wait Until Next ms Multiple





練習七-Timing

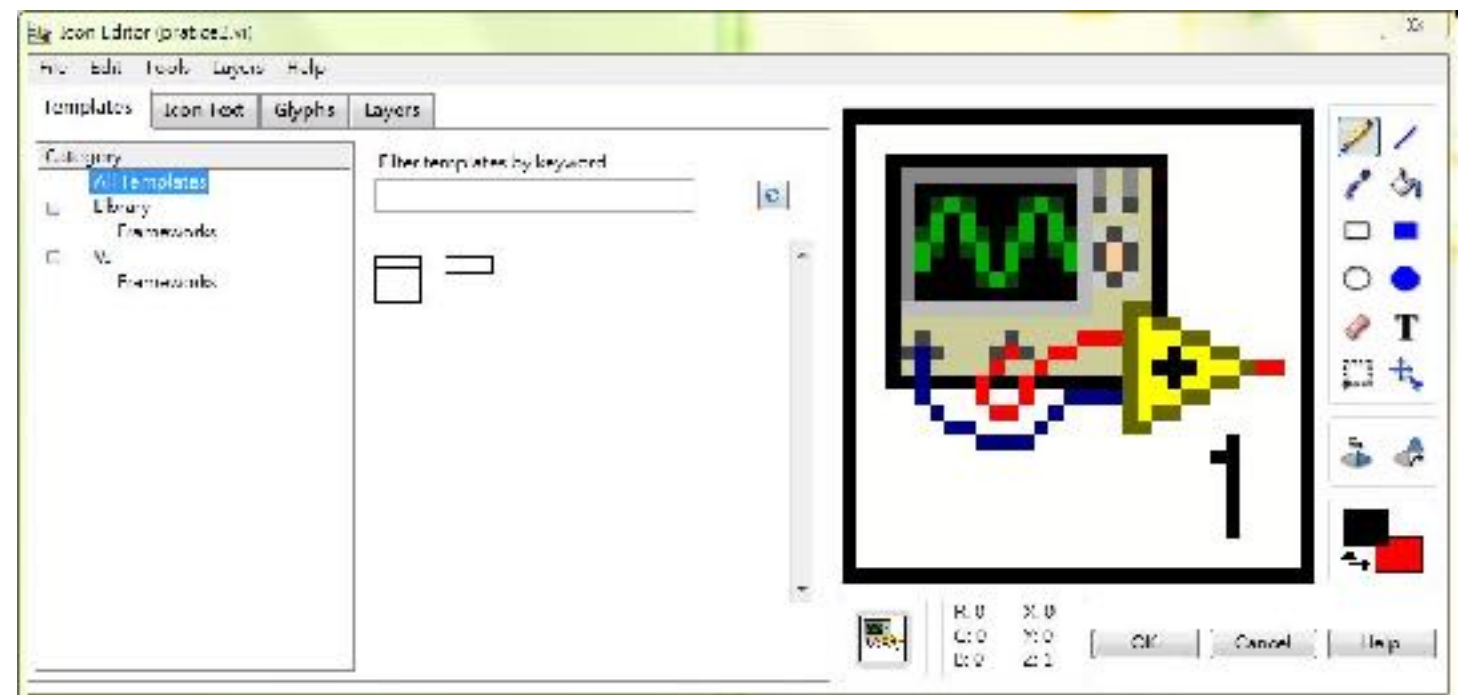
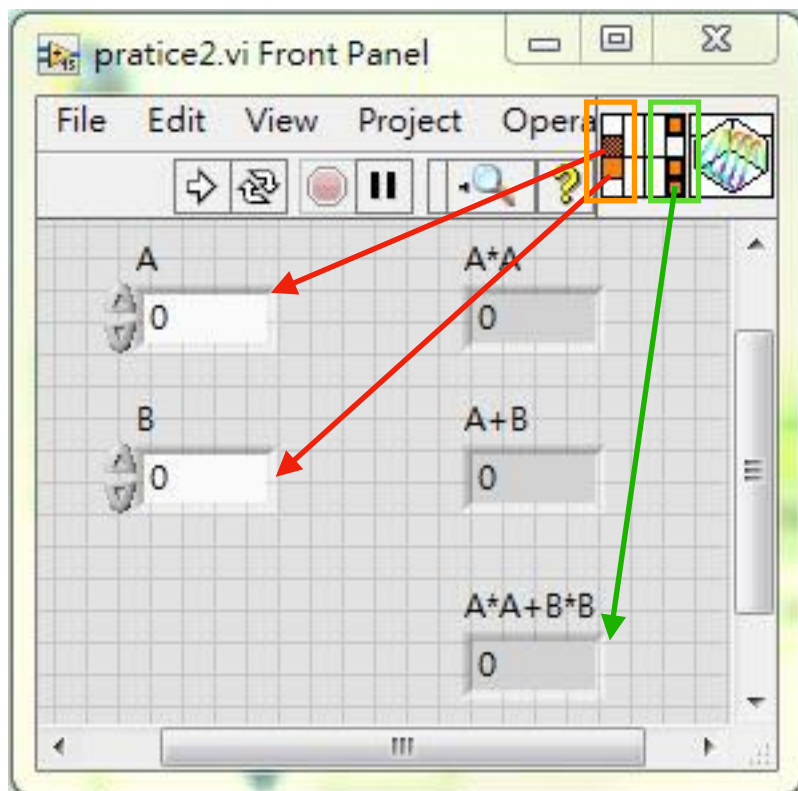
- 每秒產生一個0~1之間的亂數，並畫在Chart上。




subVI

- 在front panel  設定subVI的輸入與輸出
- 對 front panel 上的  點選右鍵，選擇Edit Icon，可編輯圖示。 
- File >> VI Properties >> Category >> Documentation，在VI description寫入敘述，使用此subVI時Help內會顯示描述。
- 如何使用subVI？開啟新的VI，從資料夾內拖曳進Block Diagram內。

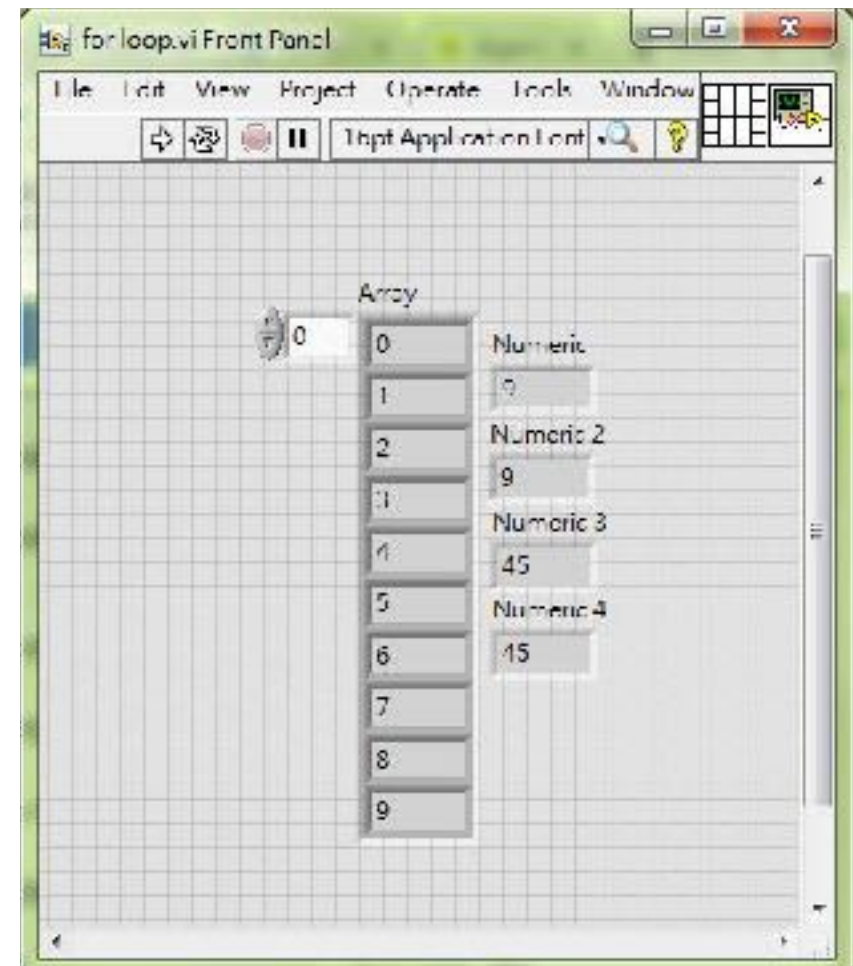
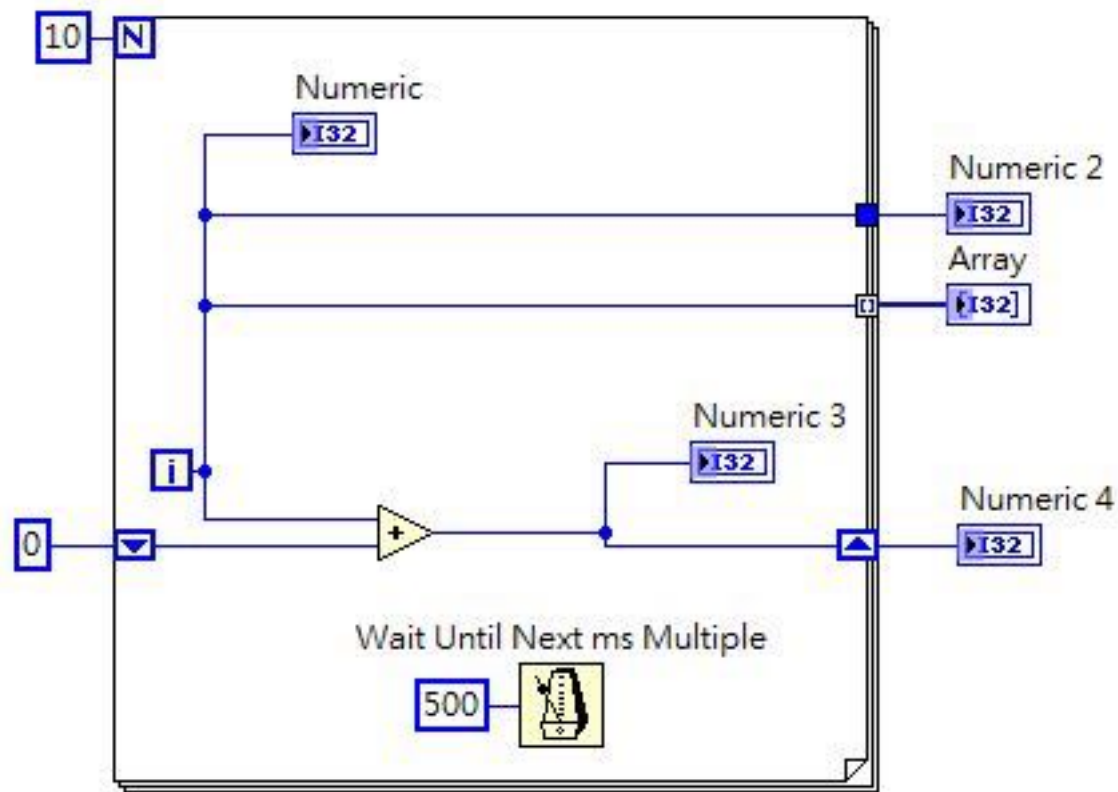
Input **Output**



練習八

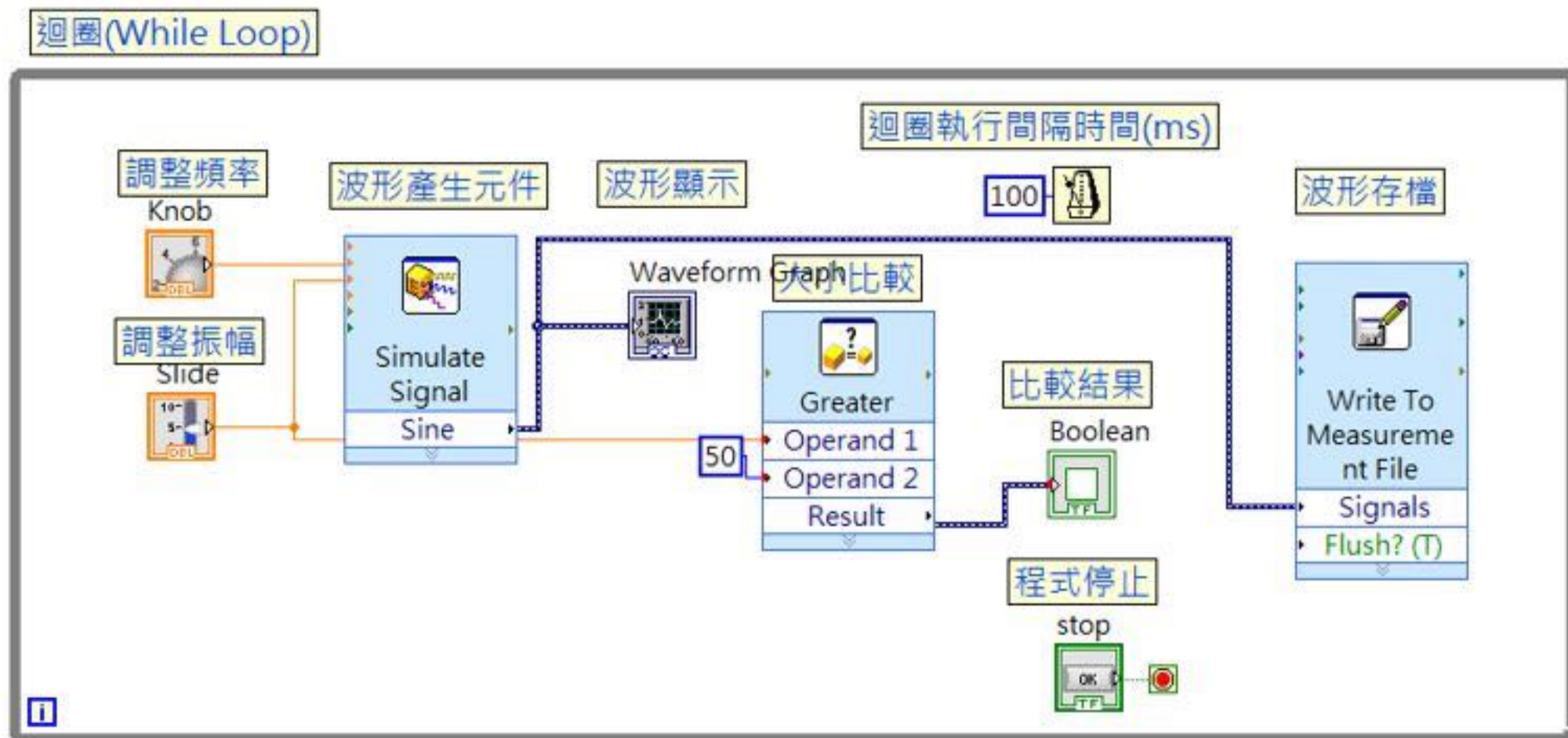
- 運用練習一找到的東西完成一個VI，並存成praticice2.vi
- 請打開.  Execution Highlighting button，執行資料流動畫，查看資料流的動向。
- 小技巧：找不到的東西可以直接Search

執行結果



練習九-建立程式區

- 利用練習一已建立的LabVIEW基本物件
- 模擬訊號和展現



第一、二題截圖放進word

第三題直接寄vi檔

信件標題： [G10] LabVIEW

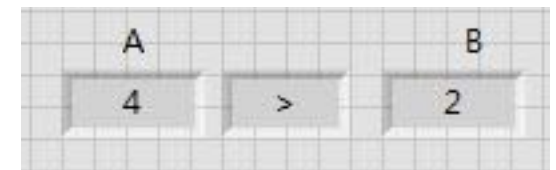
請寄至chen.wendy@g.ncu.edu.tw

期限**2017/07/21 20:00**

-陳怡玟

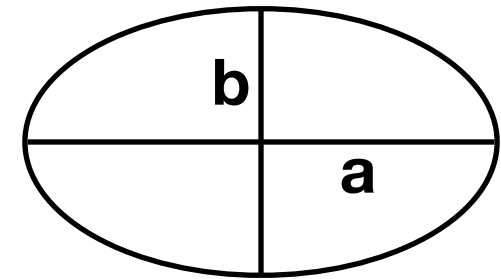
作業一

- hw3-數字大小判斷
- 隨機產生1~10之間的整數，並判斷兩者之間的關係，並顯示餘人機介面。
- Functions >> Programming >> Numeric >> Random Number (0-1)
- 將程式碼與結果截圖放到word，並標記作業題號。



作業二

- hw4-橢圓繪製
- 按照下列給定算式畫出橢圓形，ab大小自訂。
- Hint : Graph >> XY graph, sine & cosine, bundle
- 將程式碼與結果截圖放到word，並標記作業題號。



中心位於點 (h, k) 的主軸平行於 x 軸的橢圓由如下方程指定

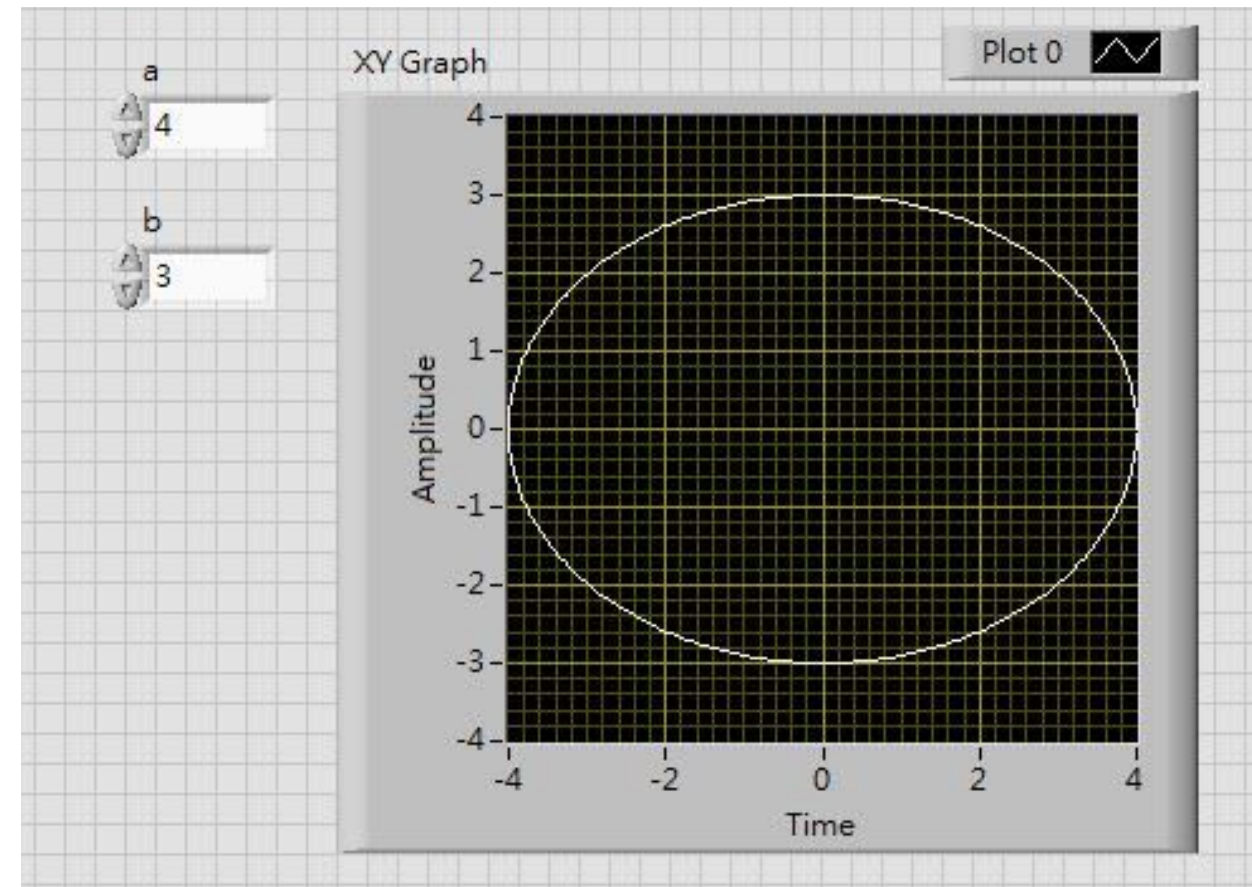
$$\frac{(x-h)^2}{a^2} + \frac{(y-k)^2}{b^2} = 1$$

這個橢圓可以參數化表達為

$$x = h + a \cos t,$$

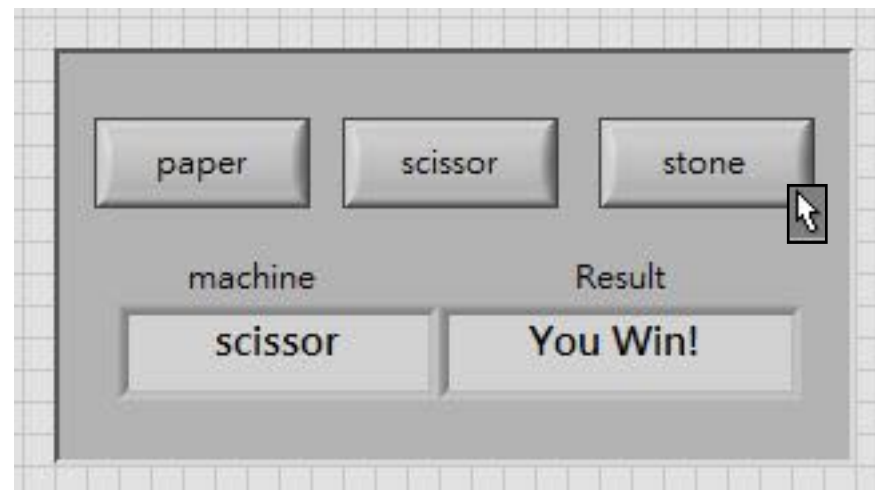
$$y = k + b \sin t$$

這裡的 t 可以限制於區間 $-\pi \leq t \leq \pi$ 。



進階-作業三

- 請做出一個與機器猜拳的程式，可連續使用，並將結果顯示於人機介面。
- 此題請直接寄vi檔案。



第一、二題截圖放進word

第三題直接寄vi檔

請寄至chen.wendy@g.ncu.edu.tw

期限**2017/07/21 20:00**

-陳怡姘