

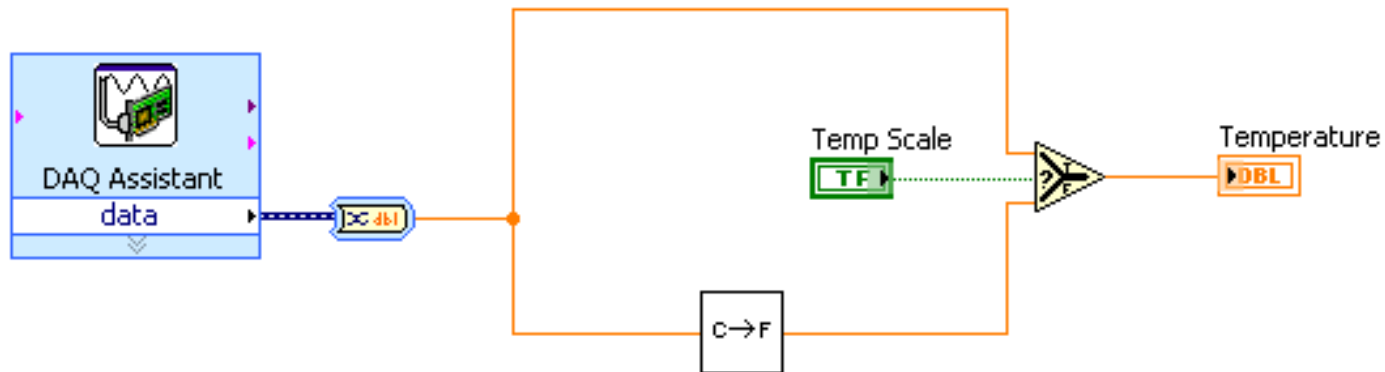
單元7： 在LabVIEW做決策

主題：

- a. 使用Select 函數進行決策
- b. 如何使用Case 結構
- c. 如何使用公式節點(Formula Node)

Select函數

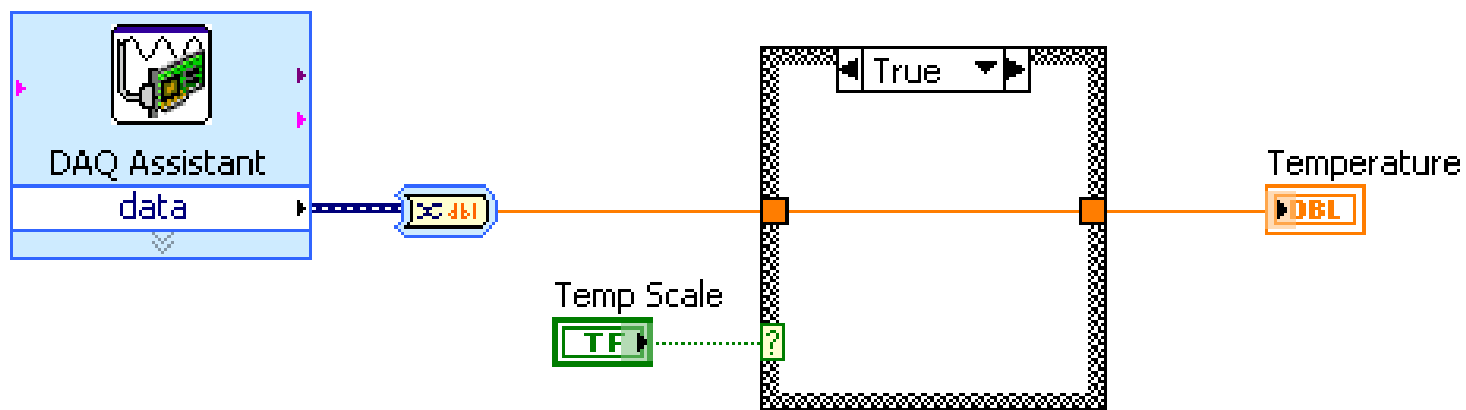
- 如果Temp Scale是TRUE，輸出上方數值；如果temp scale是FALSE，則輸出下方的數值



- 這只適用於簡單的判斷，無果要做更複雜的判斷，則要使用Case結構(Case Structure)

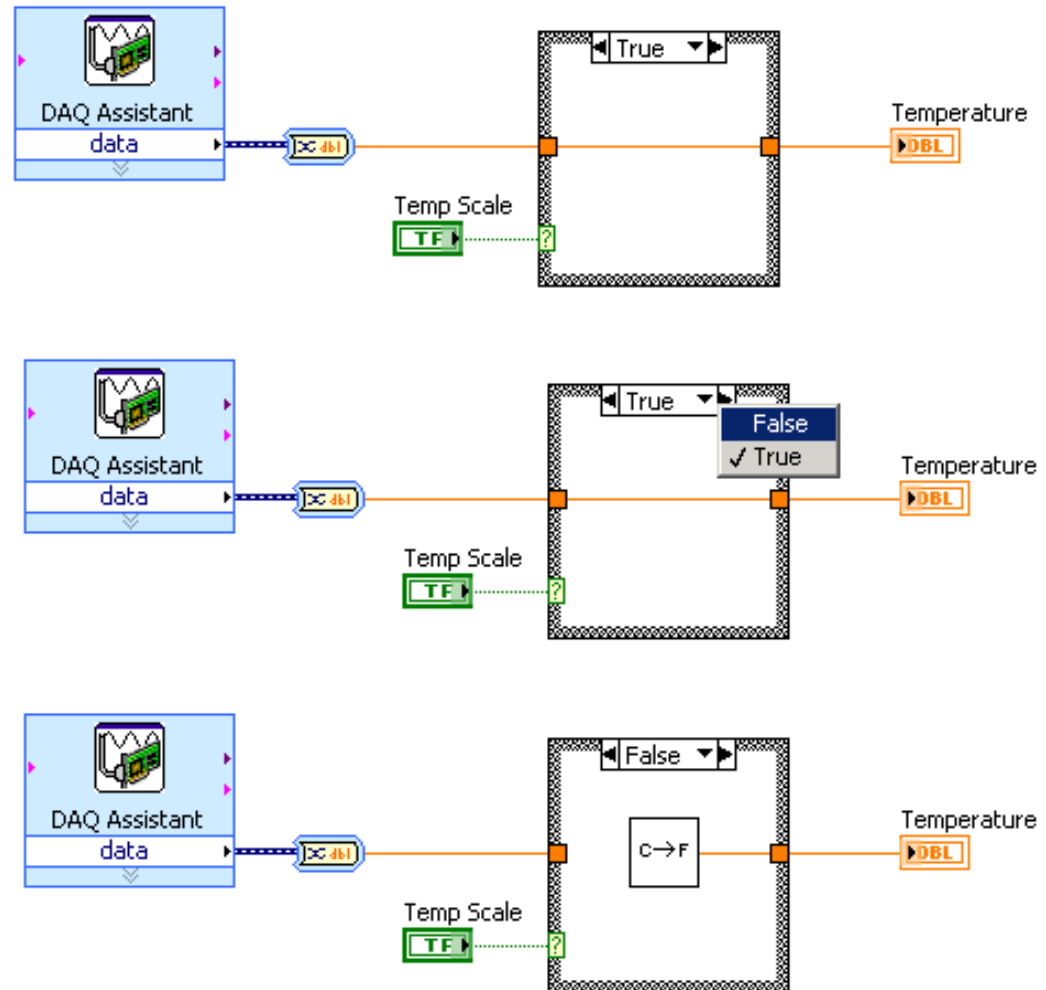
Case結構 (Case Structures)

- 如果布林是True，則資料進入True的結構。
- 如果布林是False，則資料進入False的結構。

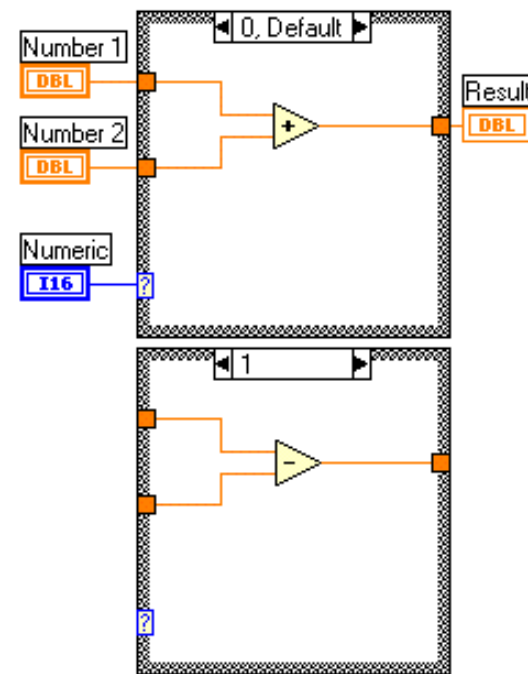
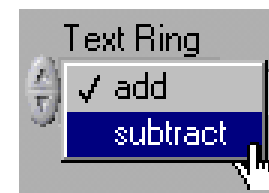
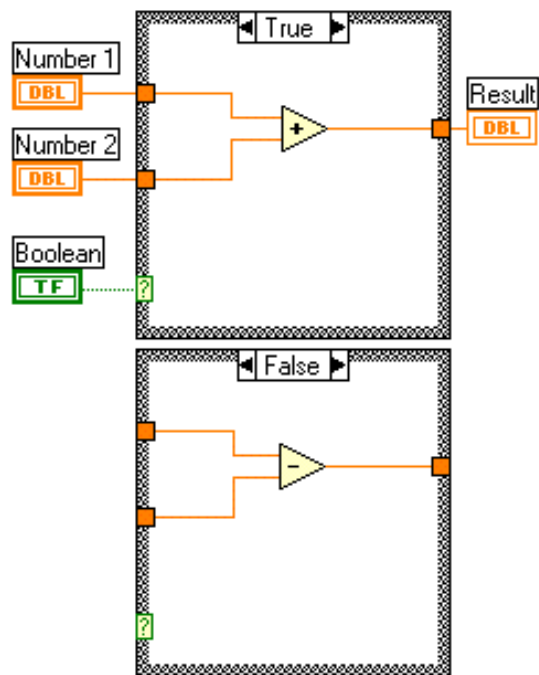


Case Structures

- 使用Case 結構取代 Select 函數。在本圖中，Case 結構中的第一選項是True。
- 要選擇狀況，請在條件選擇辨識元中輸入值，或使用Labeling 工具來編輯其值
- 再同一時間只有一個Case可以通過

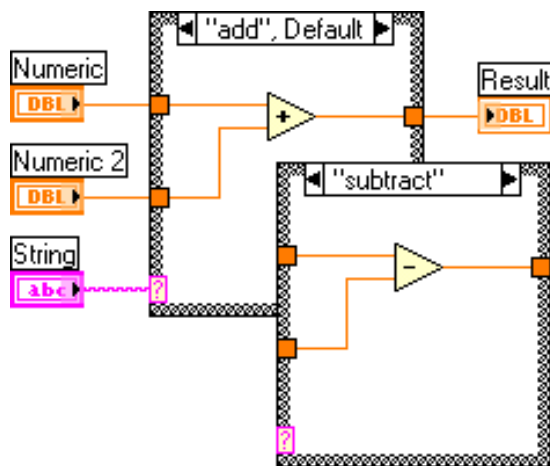


Boolean and Numeric Cases

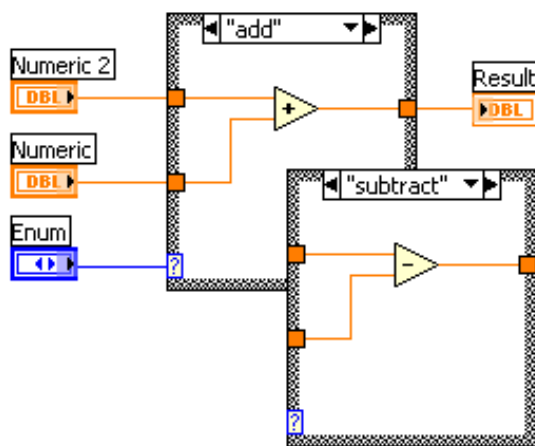


所有對外輸出的通道都要連線。

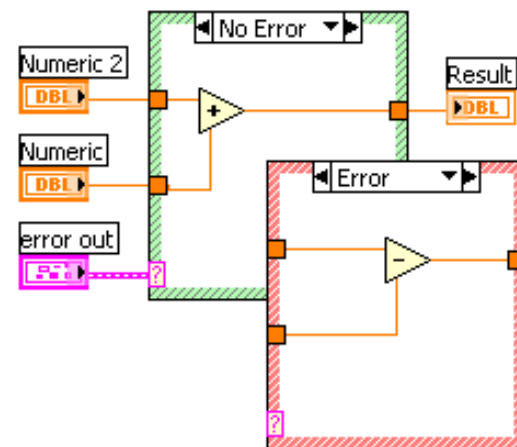
字串(String)、列舉(Enum)、與錯誤(Error)結構



字串結構



列舉結構

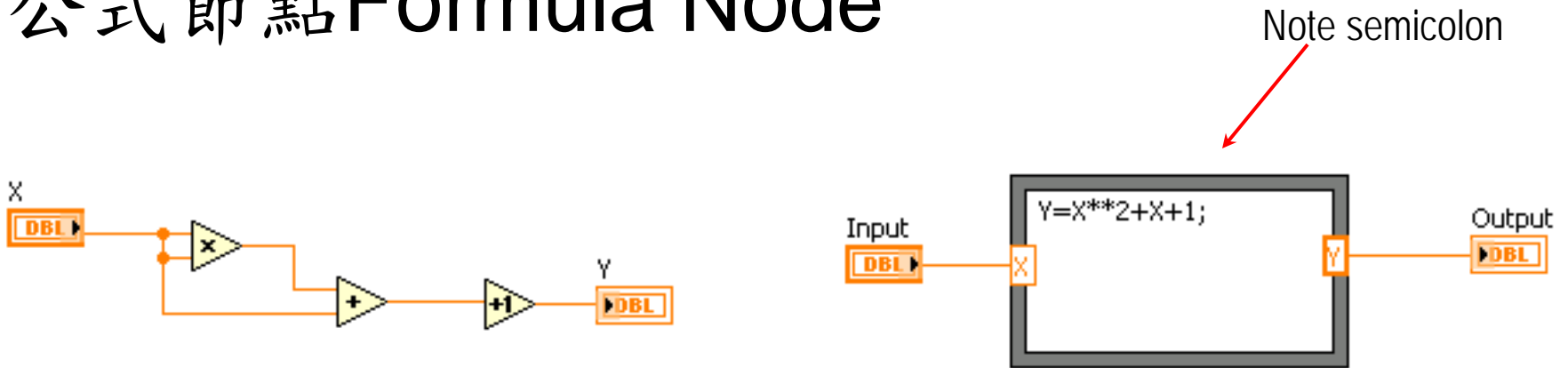


錯誤結構

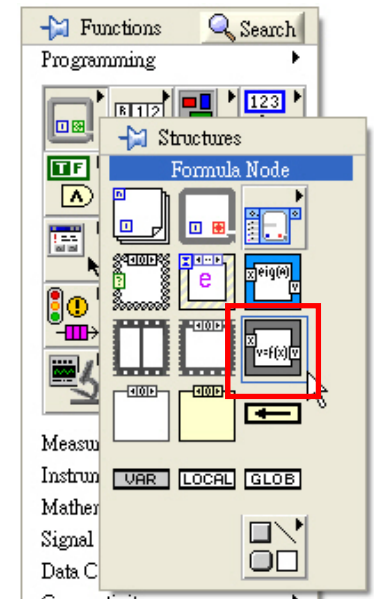
練習7.1 – 計算平方根

- 寫一個可以計算平方根的程式。當輸入值大於0，就計算平方根。當輸入值小於0，就顯示錯誤訊息

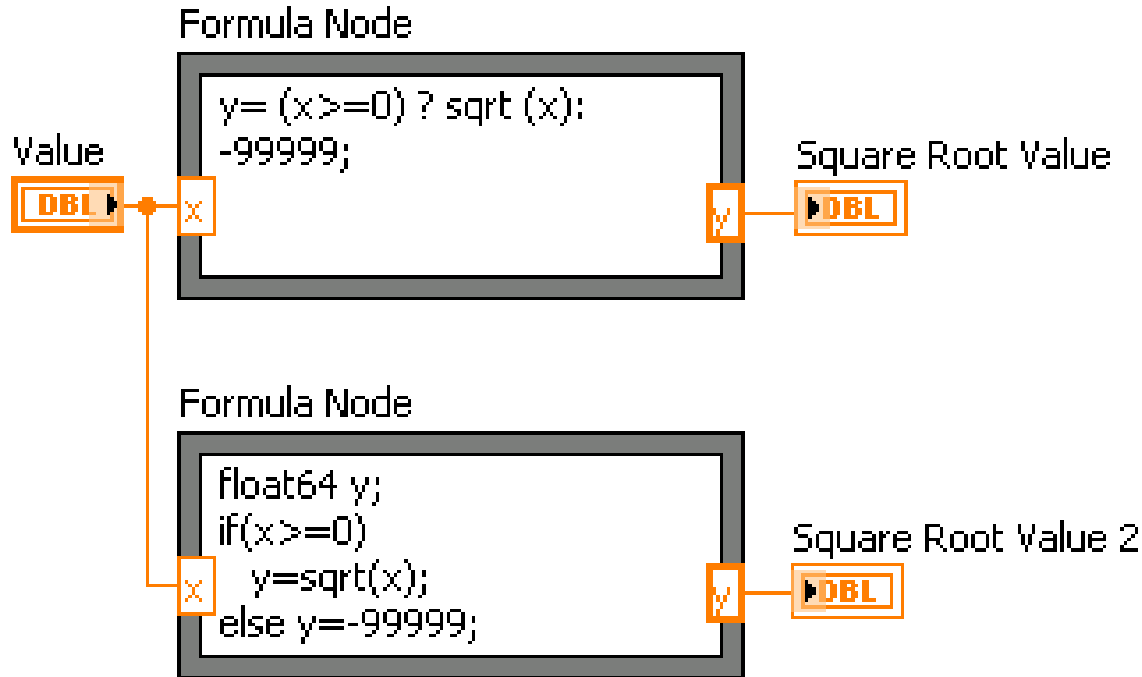
公式節點 Formula Node



- 如果你不習慣使用圖形運算函數，或是你的運算過於複雜，都可以使用公式節點來簡化程式
- 公式節點是一種方便的文字式節點，你可以利用它，在程式區中執行數學運算
- 你可以將現有的文字式程式碼複製到公式節點中，而不需要在程式區中以圖形方式重新撰寫
- 每個方程式敘述必須以分號(;) 做結束



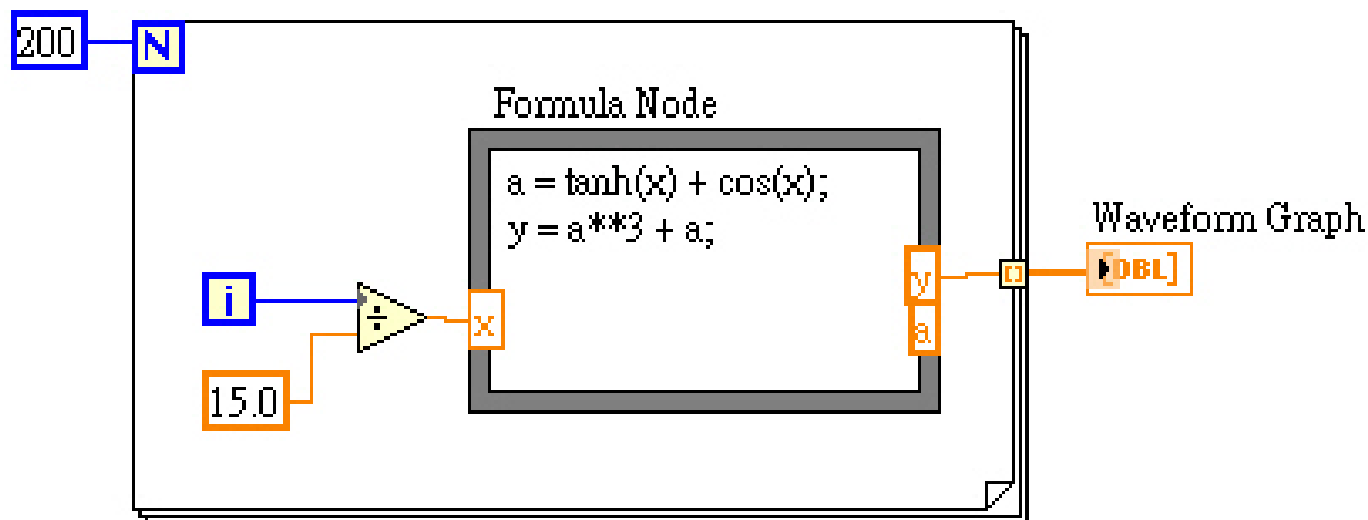
使用 Formula Node 來進行決策



在公式節點裡面使用if-then函數，最後得到的效果跟使用Case結構是相同的

練習7.2 – 使用公式節點進行複雜的數學計算

- 建立以下的程式區，以練習使用Formula Node



本章重點回顧

- **Select** 函數會根據布林輸入值，從兩個輸入值中選擇其一。
 - **Case** 結構擁有兩個或更多的子程序（或稱狀況）。一次只看得見一個狀況，而且結構也只會執行一個狀況。
 - 如果狀況選擇元接頭是布林值，則結構擁有 **TRUE** 狀況及 **FALSE** 狀況。如果選擇元接頭為整數、字串或列舉值，該結構最多可有 $2^{31} - 1$ 種狀況。
 - **Case** 結構中的所有子程序都可以使用其輸入埠，但是子程序不需要使用所有的輸入埠。如果至少有一個輸出埠沒有被定義，則該結構的所有輸出通道都會呈白色方格。
- 在從 **Case** 結構建立 **subVI** 時，要將錯誤輸入埠連接至選擇元接頭，並將所有的 **subVI** 程式碼置於 **No Error** 狀況之中，以便在它發生錯誤時，防止 **subVI** 執行。
- 公式節點適用於擁有許多變數因而過於複雜的方程式，亦適用於使用現有的文字式程式碼。每個方程式敘述必須以分號(;) 做結束。

練習

1. 建立一個VI，使用公式節點來計算下列的方程式
$$y_1 = x^3 + x^2 + 5$$
$$y_2 = mx + b$$

x, m, b 都是輸入值， y 是輸出值
2. 建立一個VI，有兩個輸入值(x, y)，有一個輸出值(z)，並做一個四則運算(加減乘除)的選擇器，用控制器來指定要進行的四則運算
3. 將練習7.1所計算的平方根的功能用公式節點來取代。並再計算立方根。