



LabVIEW 程式結構

Yuan-Hsien Hsu

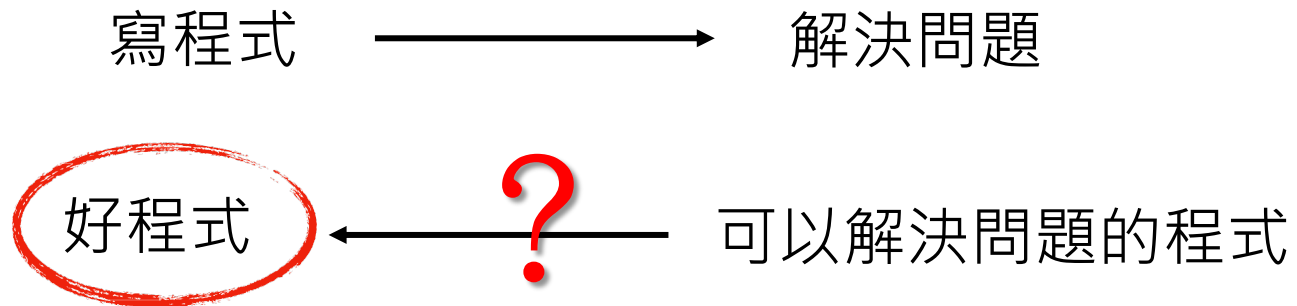
國立中央大學太空科學研究所

2021年4月9日

Outline

- Design Pattern
- State Machine
- SubVI
- Type Define
- Local Variable
- HW 4

Design Pattern



好的程式應具備

- 一致性
- 高閱讀性
- 高擴充性

寫程式常會遇到

- 寫的過程目標改變
- 完成後要求增加功能
- 寫一個跟自己過去寫過的程式功能類似，但直接抓以前的來用卻很難改動

Design Pattern

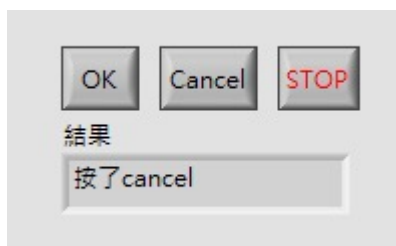
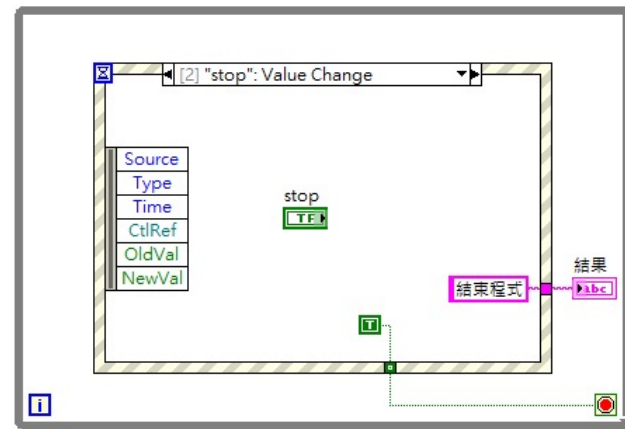
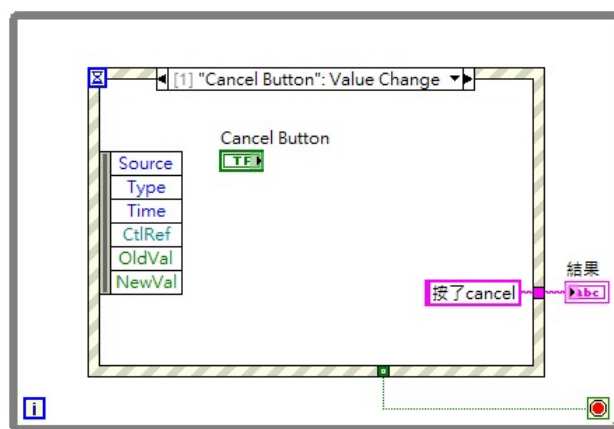
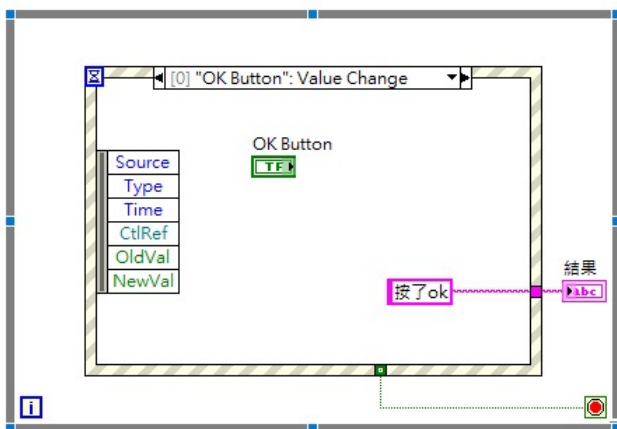
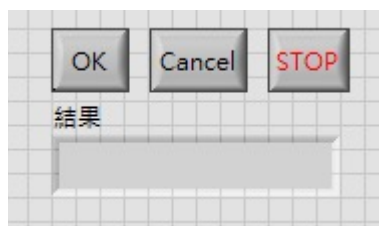
有經驗的專家 v.s. 沒經驗的新手

能在適當的時機，套用適當的公式來解決特定的問題

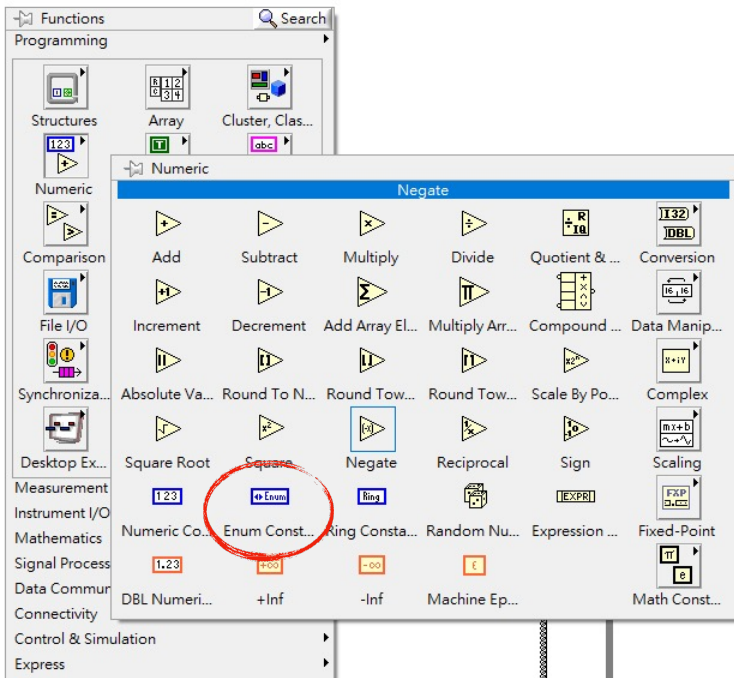
選擇適當的Design Patterns

- State Machine
- Event Structure
- Producer Consumer
- Event Producer/Consumer

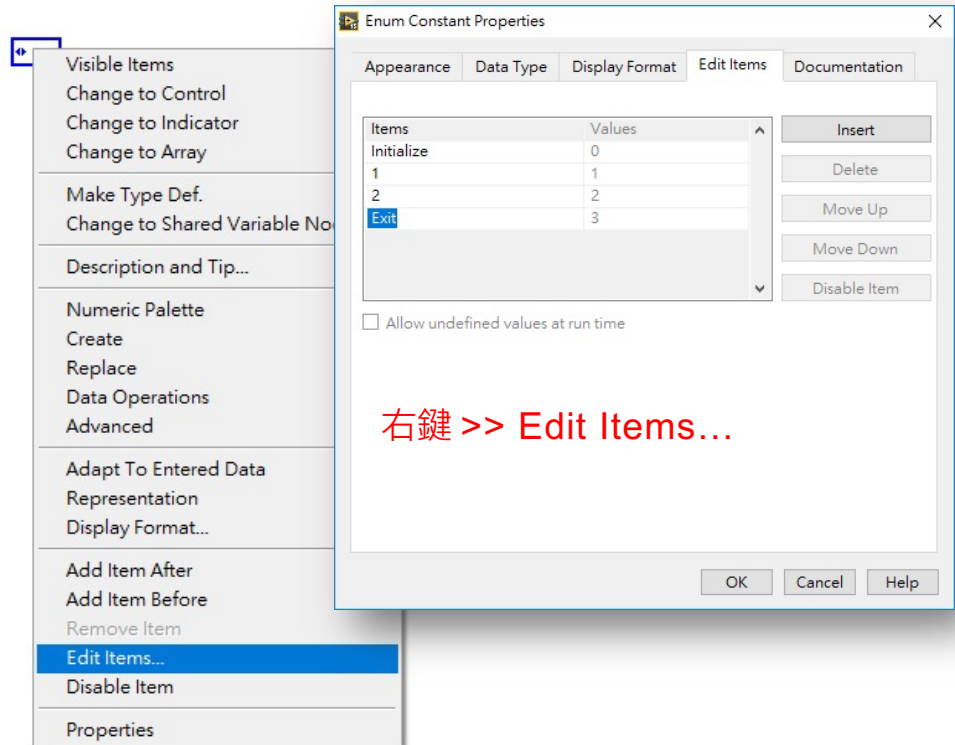
事件結構 Event Structure



Enum



Numeric >> Enum Constant



Use Type Define

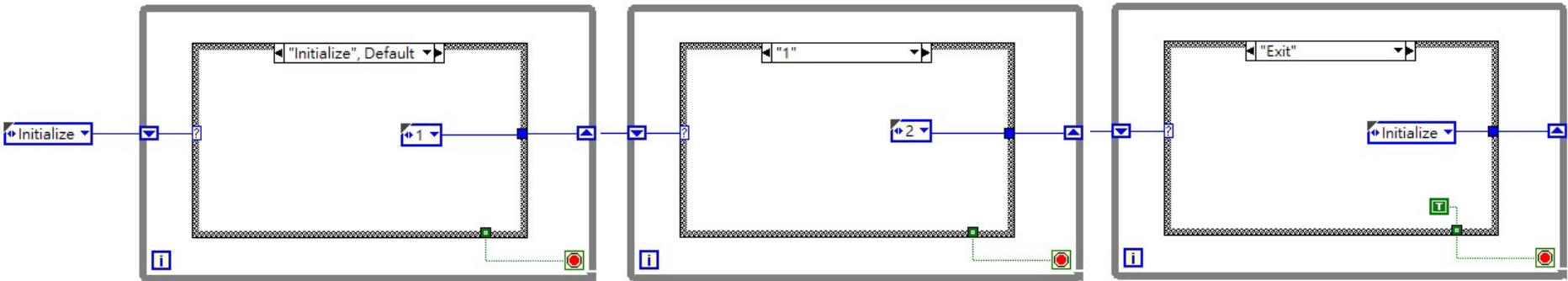


右鍵 >> Make Type Def.

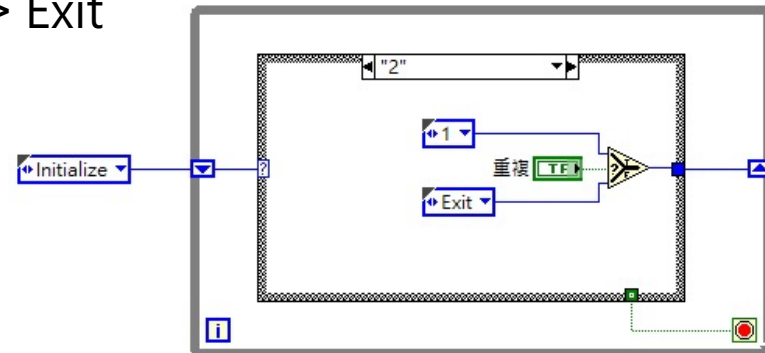
- 自動同步更新最新的狀態
- 其他 VI 也可使用
- 要修改 右鍵 >> Open Type Def. (關掉頁面時，記得儲存)

State machine

Initialize >> case 1 >> case 2 >> Exit



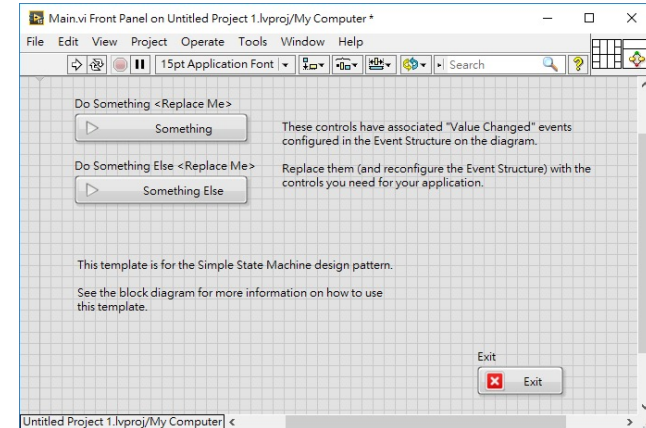
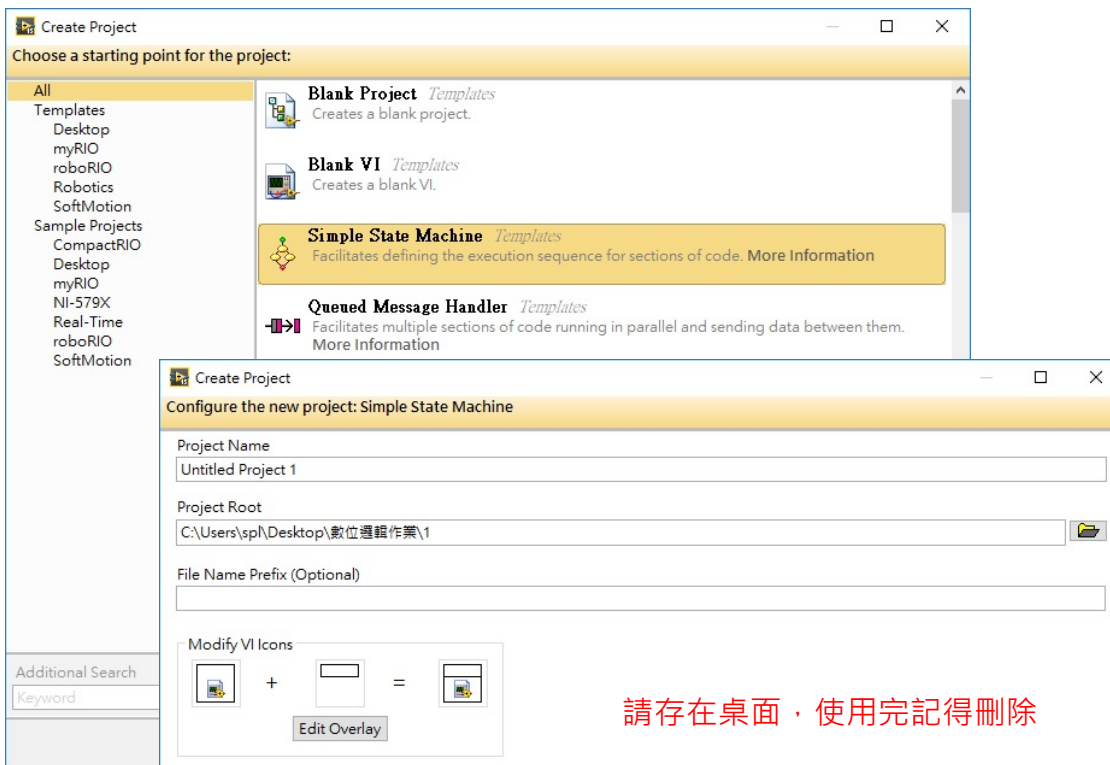
Initialize >> case 1 >> case 2 (重複為True時) >> case 1 >> ... >> Exit
>> case 2 (重複為False時) >> Exit



State machine

亦可加入 Event Structure 共同使用

範例程式參考：Create Project >> Simple State Machine >> Main.vi

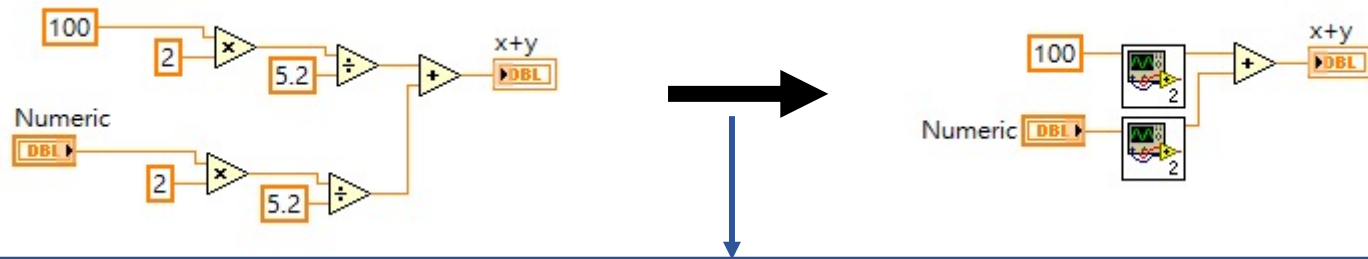


請存在桌面，使用完記得刪除

除了Design Pattern外，必須要活用的東西

- SubVI

只要相同的程式碼，就不要讓他出現兩次



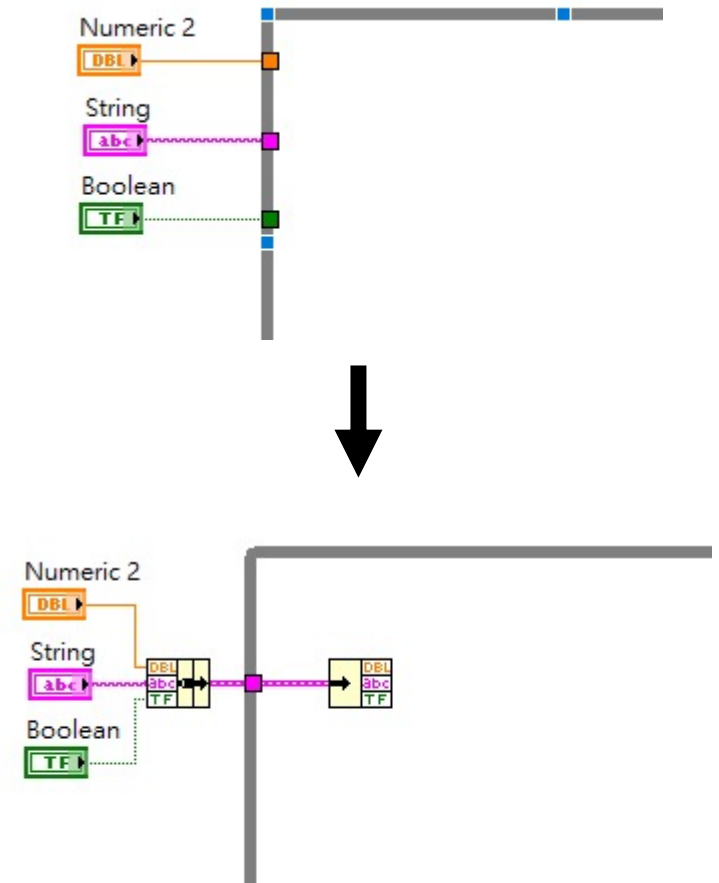
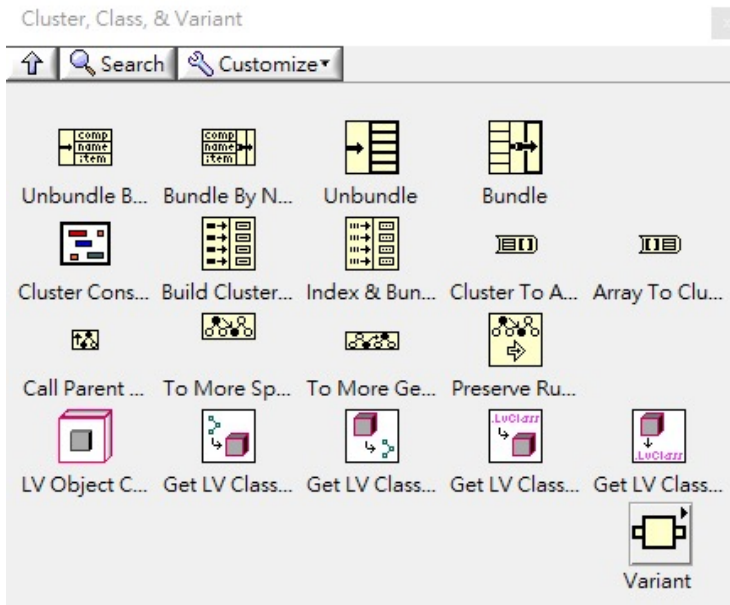
除了Design Pattern外，必須要活用的東西

- **Cluster**

可以將凌亂的雜線有系統的收納

右鍵 >> Programming

>> Cluster, Class, & Variant



除了Design Pattern外，必須要活用的東西

- **Type Define**

可以有效率的去修改相同的元件

Enum >> 右鍵 >> Make Type Def.

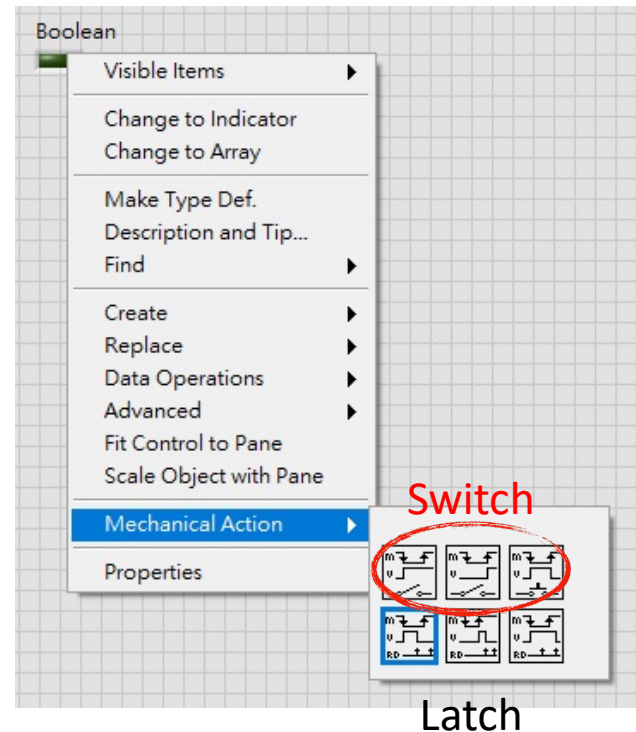
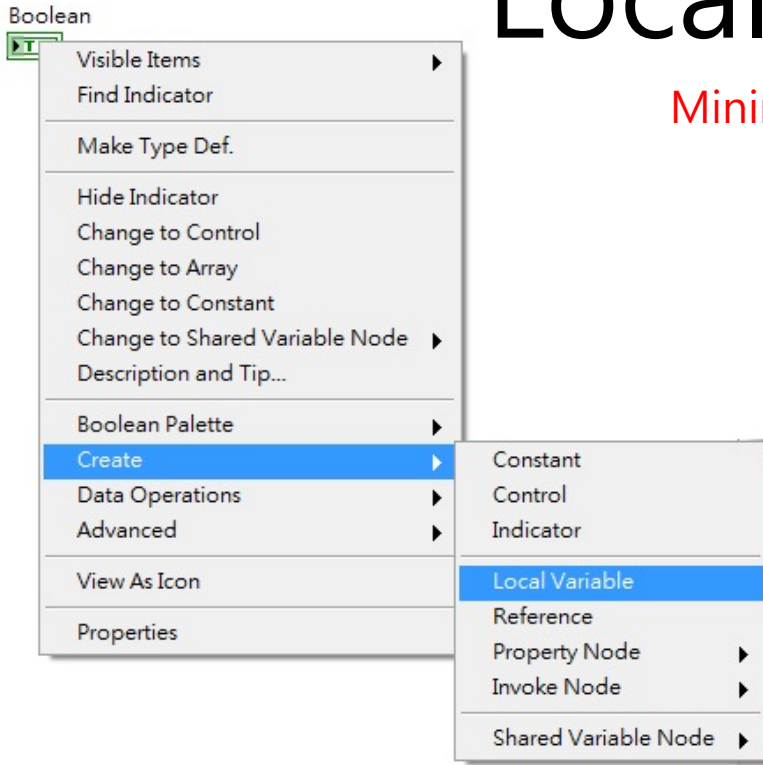


- 自動同步更新最新的狀態
- 其他 VI 也可使用
- 要修改 右鍵 >> Open Type Def. (關掉頁面時，記得儲存)

Local Variable

Minimize the use !!!

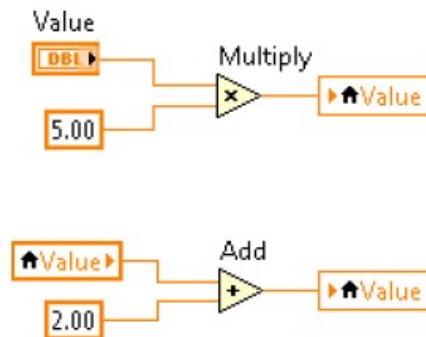
- 點擊 >> 右鍵 >> Create >> Local Variable
- 資料在同個VI中傳遞
- 當布林為 Local Variable 需設為 Switch。



注意 Race conditions

資料流不知從誰先開始。

無法確定資料動向。



HW 4

請參考標準小算盤設計，讓使用者能在 LabVIEW 人機介面上使用計算機。

- 下圖紅框區為基本得分區，若再自行增加其他功能，可加分
- 請讓使用者在可在人機介面上，可同時看到結果顯示與歷程記錄

作業截止時間：

4/30(五) 中午12:00

此次作業為最後一次 LabVIEW 程式作業

請把握加分機會



作業繳交方式

- 繳交時間: 4/30 12:00 助教課
- 繳交方式: 整個Project跟所有的vi都放在資料夾下，資料夾取名為學號。
- 遲交扣分方式: 一天分數*90%、2天80%、3天70%、超過4天0分

- 有任何問題至820詢問或直接Email我:
- Email: f105601043@g.ncu.edu.tw