

DemoRoom - 製作流程步驟與解說

環境背景:

以泓格科技的階梯教室作為 TouchPAD-280 之實作環境，於其中使用了 19 台

I-7065 與 2 台 I-7018 實作燈光控制、冷氣控制與溫度量測之應用。

製作 DemoRoom 之前置準備工作:

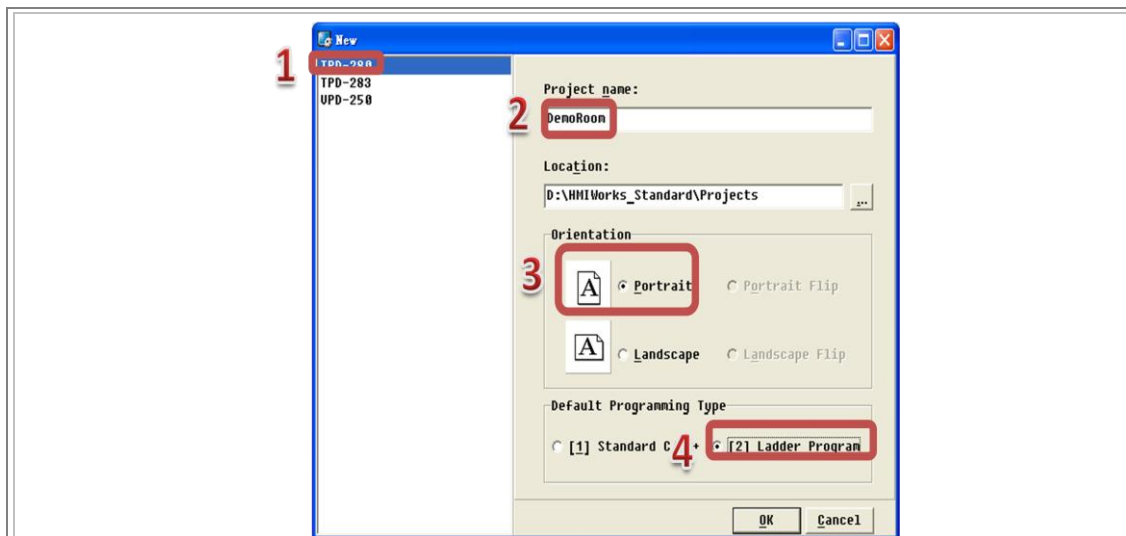
1. 硬體配置圖 (DO、DI、AI 與燈組、冷氣、實體開關之相對關係等)。
2. TouchPAD 背景介面圖。
3. TouchPAD 控制相關按鈕圖(需準備一組兩張，開/關)。

DemoRoom 製作流程與說明 – 首頁

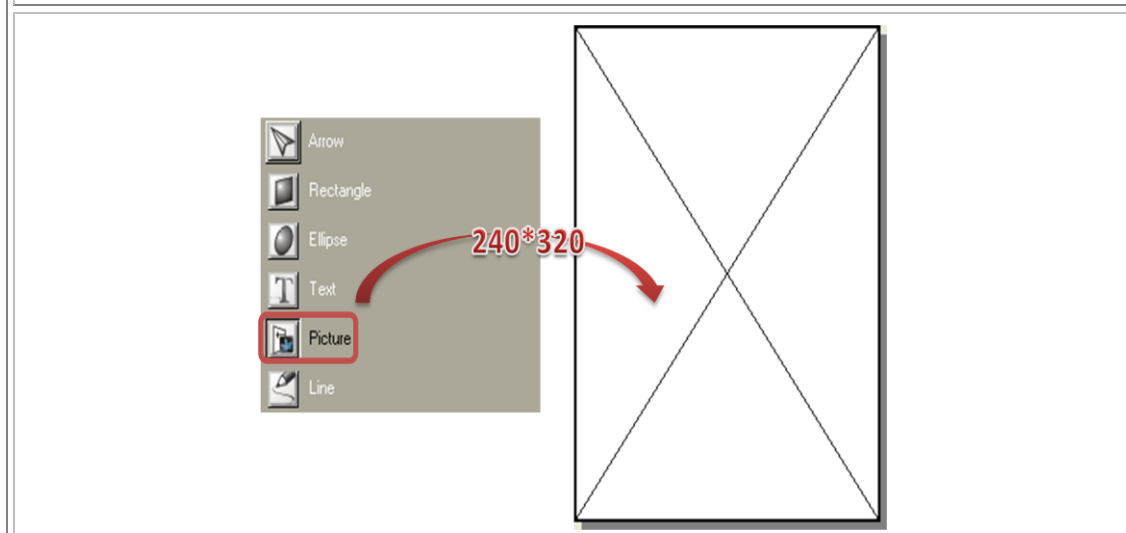
1. 建立一個新的專案。



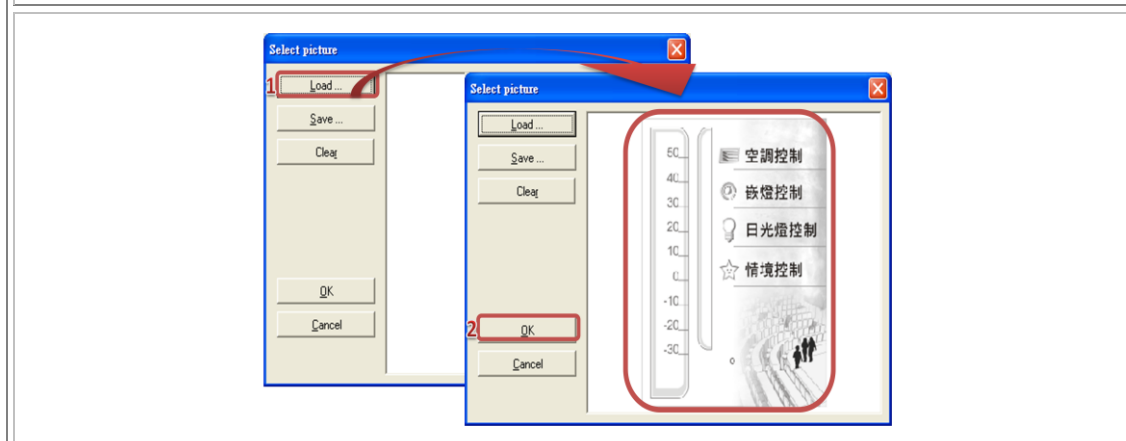
2. 輸入專案名稱: DemoRoom，專案方向: 直立(Portrait)，專案格式: Ladder Program。



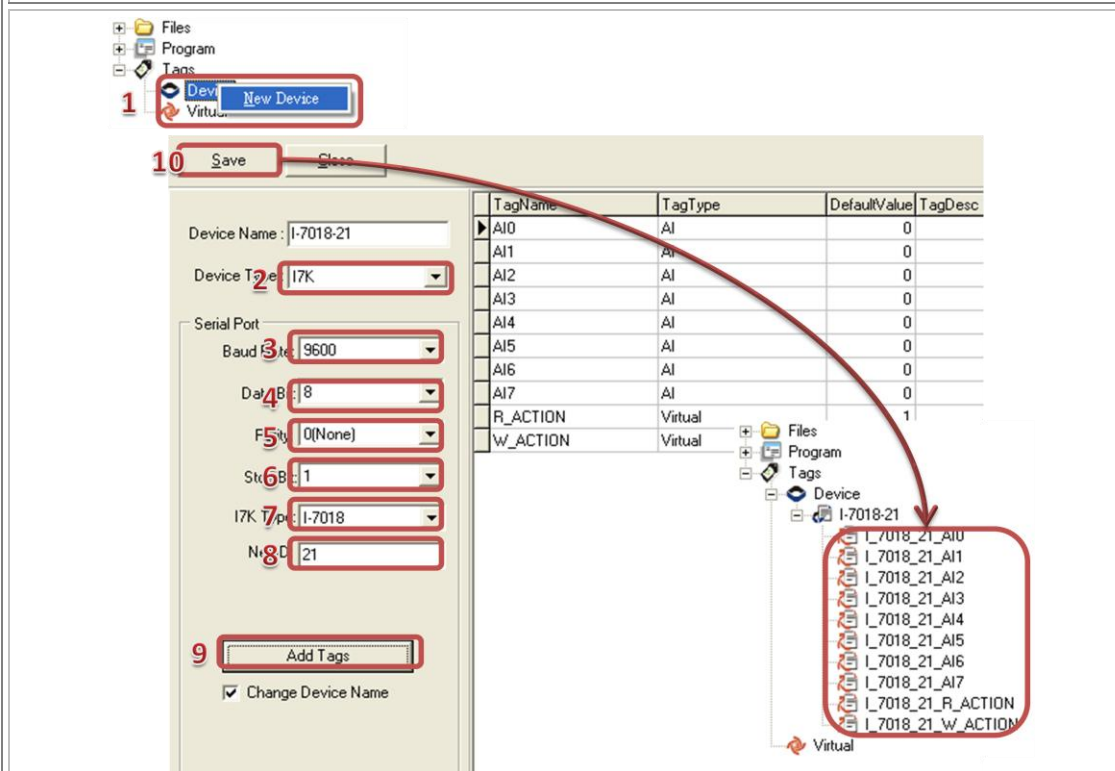
3. 於 Drawing 介面選擇 Pictrue 元件並拖曳至編譯畫面，將 Picture 的 Width 和 Hight 分別設定為 240 與 320。



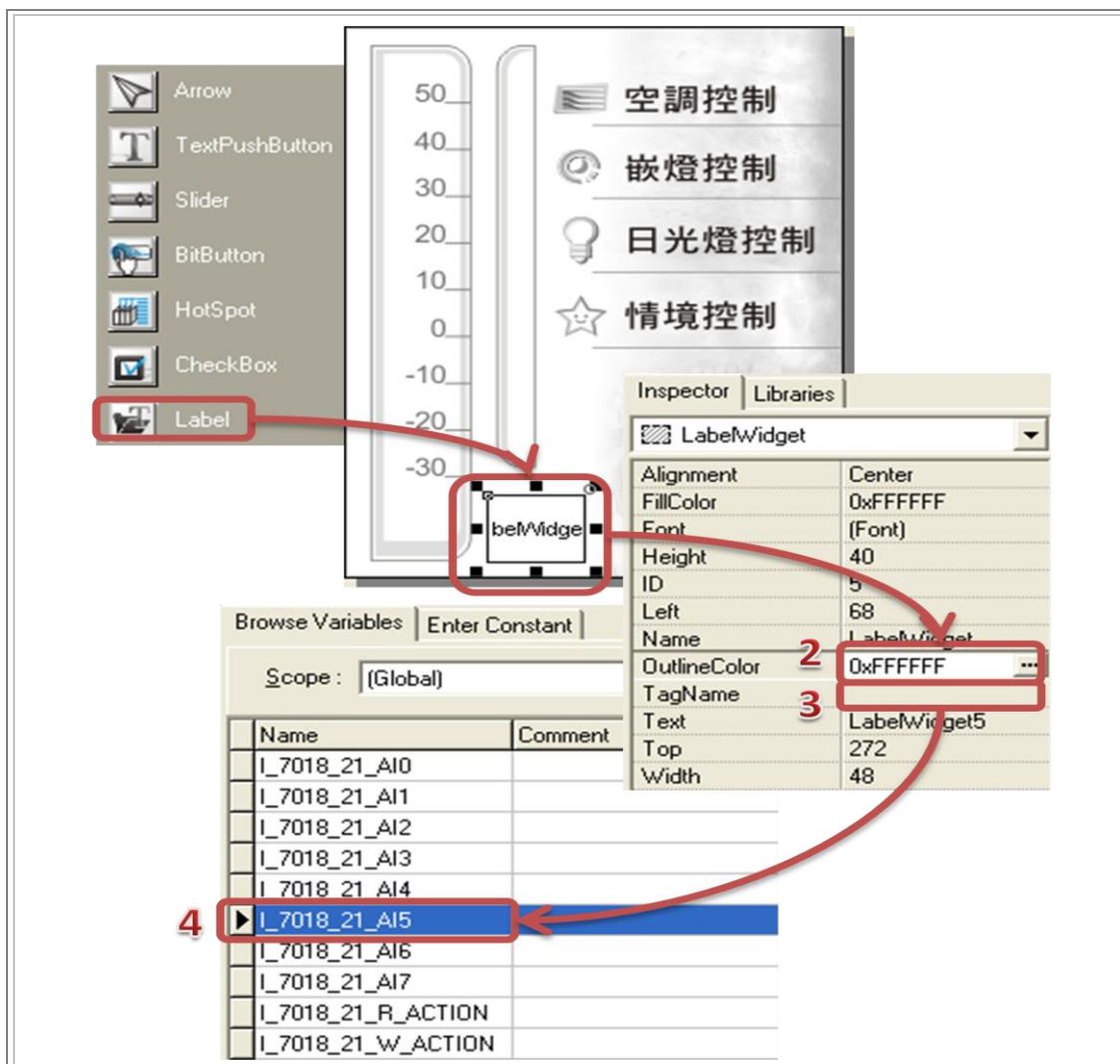
4. 雙擊 Picture 元件，點選“ Load” 並將首頁圖片載入至 TouchPAD 中。



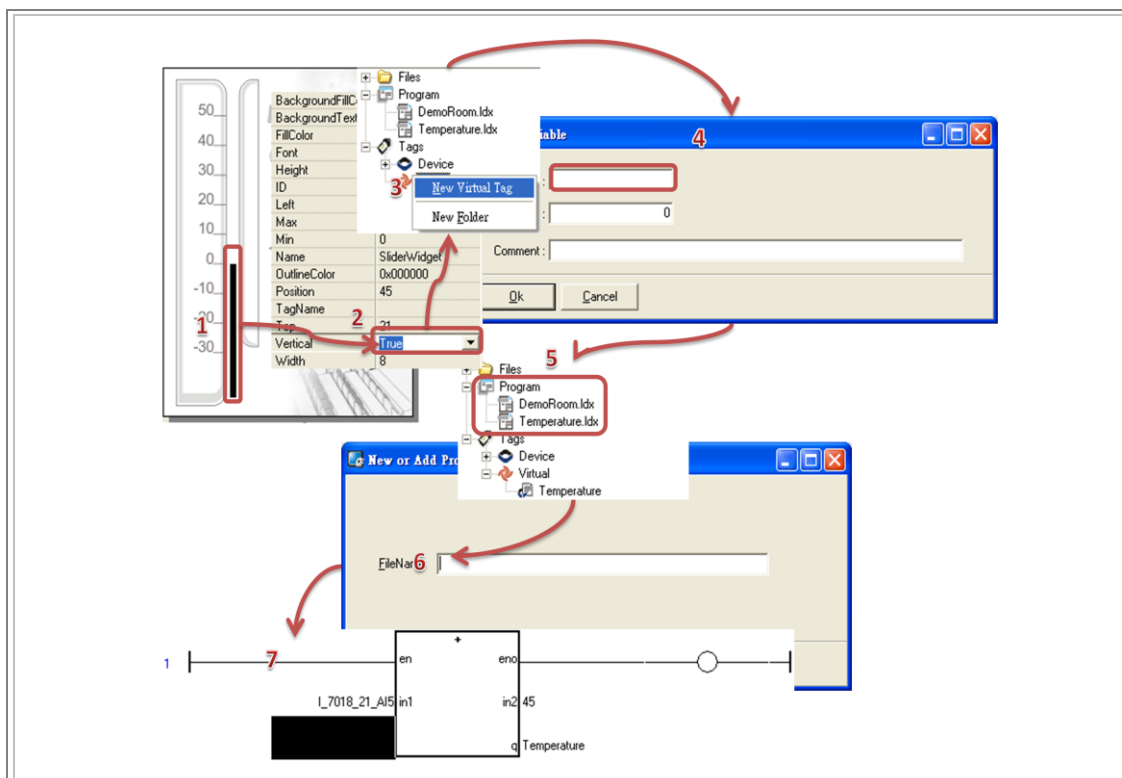
5. 於 WorkSpace 介面之 Device 項目上，按滑鼠右键以新增一個 Device，進入 Device 介面後，首先選擇 Device Type 為：I7K，Baud Rate：9600，Data Bit：8，Parity：0，Stop Bit：1，I7k Type：I-7018，Net ID：21，(以上之設定均需依造現場模組現況進行設定)。設定完成點選“Add Tags”則會出現該模組之 I/O，確認無誤後點選“Save”進行設定儲存。



6. 於 Widget 頁面選擇 Label 元件，並在設計頁面上選擇適當位置放置 Label，建置完成後，在右邊的屬性欄可利用 OutlineColor 改變 Label 的外框顏色，而與 I/O 模組建立關聯只需要在 TagName 屬性上雙擊滑鼠左鍵，點擊完出現的 Select variable 頁面會顯示之前新增的 I/O 模組。因為此 Label 用於顯示溫度，所以選擇 I-7018 其中一個用於偵測溫度的 AI Channel 來建立關聯。



7. 於 Widget 頁面選擇 Slider 元件，在設計頁面選擇適當位置放置 Slider，於右邊屬性欄可利用 Vertical 屬性改變 Slider 為直立式或橫立式。於 WorkSpace 介面之 Virtual 項目上按滑鼠右鍵新增一個 Virtual Value：Temperature，於同一介面之 Program 項目上按滑鼠右鍵新增一個 Program: Temperature。建立完成後，在 Temperature(Program)上雙擊滑鼠左鍵進入 Ladder Diagram 編譯畫面並建立邏輯。



8. 建立完成並將專案 Loading 至 TouchPAD 上，執行結果如右圖所示。



DemoRoom 製作流程與說明 – 空調控制

1. 將空調控制的背景圖 Load 至設計畫面並建立一個新的 Program:

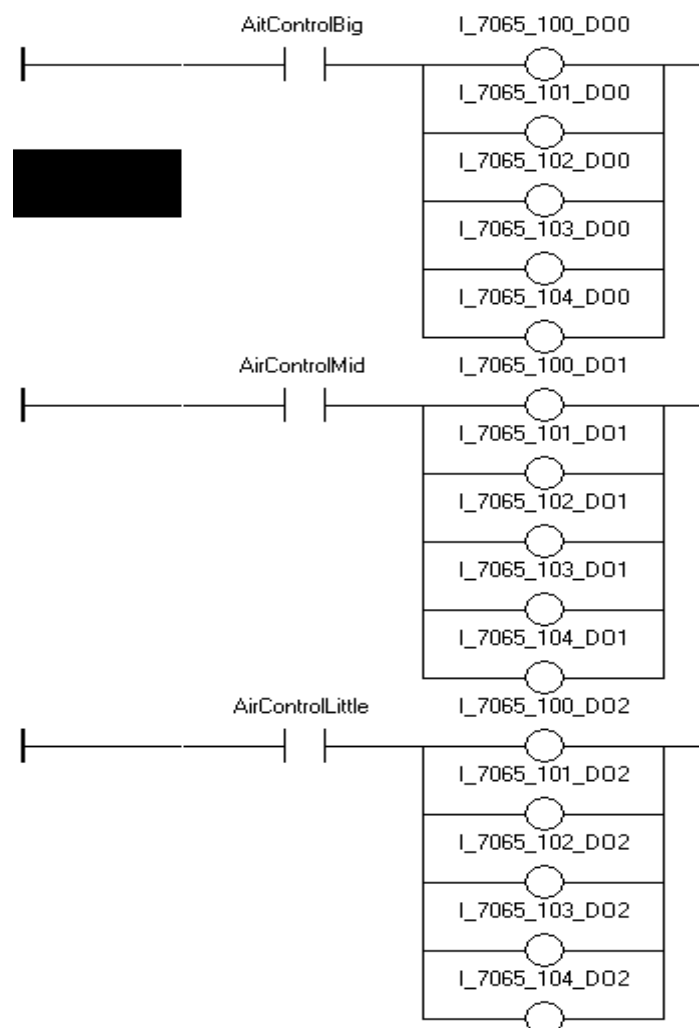
AirCondition °



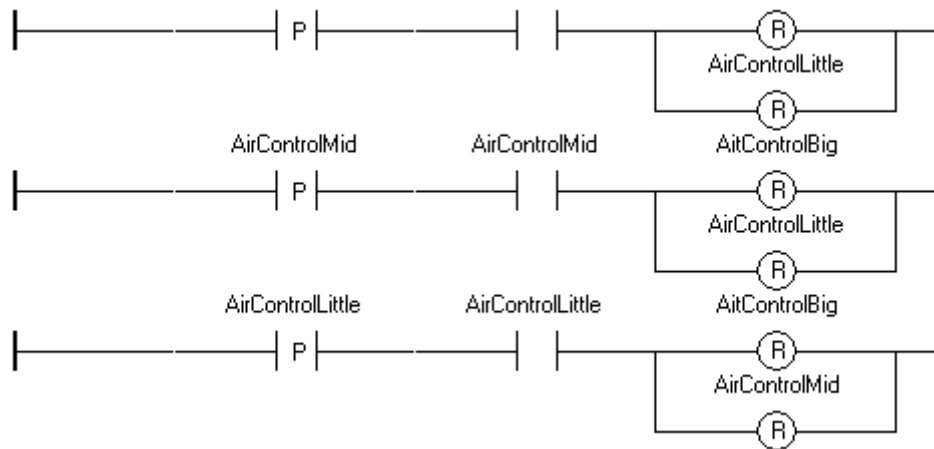
2. 建立五個 Virtual Values : AirControlBig, AirControlMid, AirControlLittle, AirCondition, AirControl · 作為風量大、中、小、冷氣開關與自動溫控使用 · 並將預先製作完成的圖案 Add to Library · 並於 ObjectList 中選擇對應的圖案。



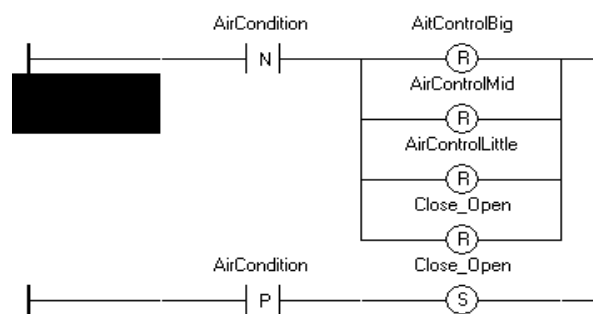
3. 於 AirCondition(Program)中，首先建立風量大中小之 Ladder Diagram。

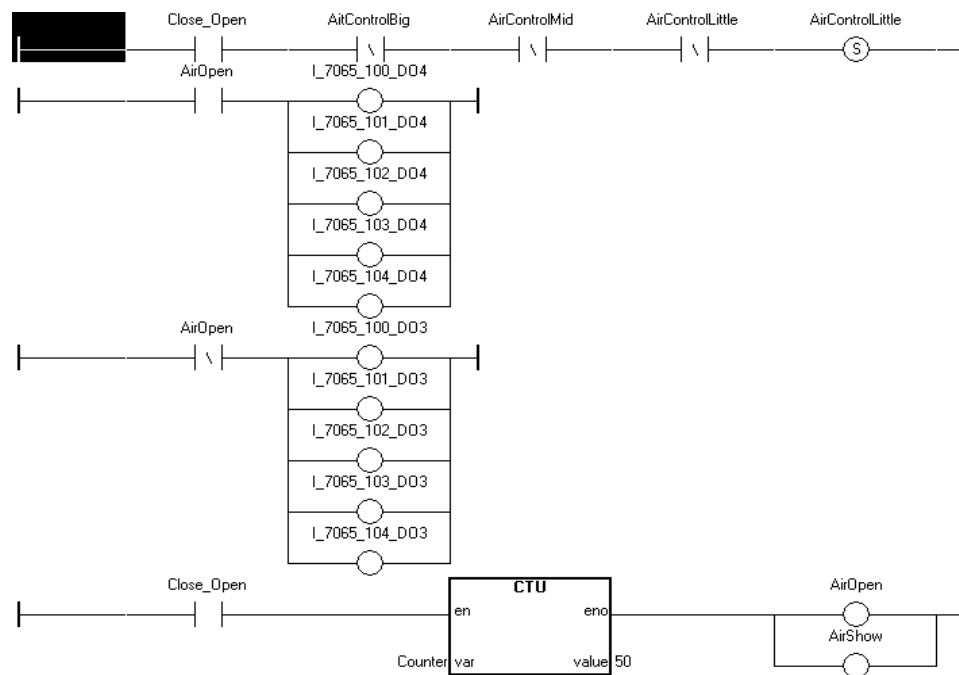


4. 於 AirCondition(Program)中，建立風量大小切換之 Ladder Diagram。

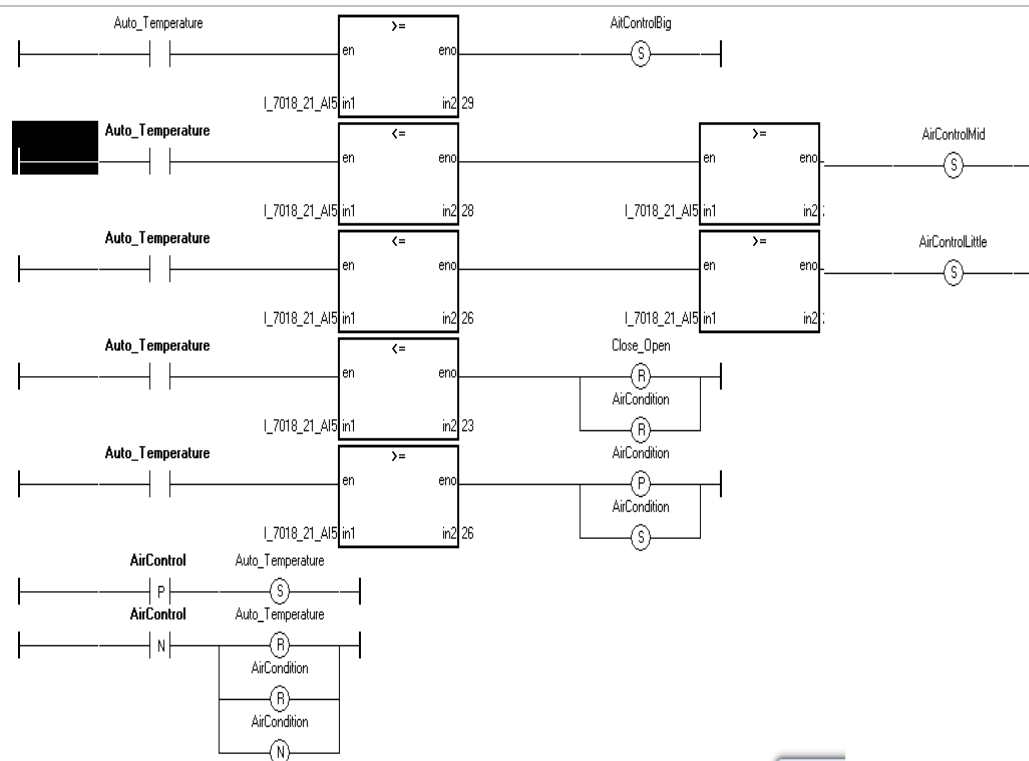


5. 新增兩個 Virtual Value : Close_Open、AirOpen 作為冷氣開關之關聯，並建立冷氣開關之 Ladder Diagram。





6. 於 AirCondition(Program)中，建立自動溫控之 Ladder Diagram。



7. 空調控制的 Ladder Diagram 設計完成，及可進行空調的操控。

DemoRoom 製作流程與說明 – 嵌燈控制

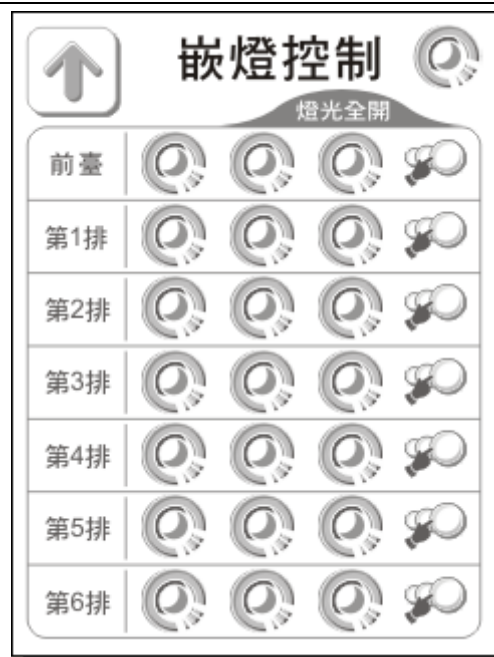
1. 將嵌燈控制的背景圖 Load 至設計畫面並建立一個新的 Program:

DownLight 。

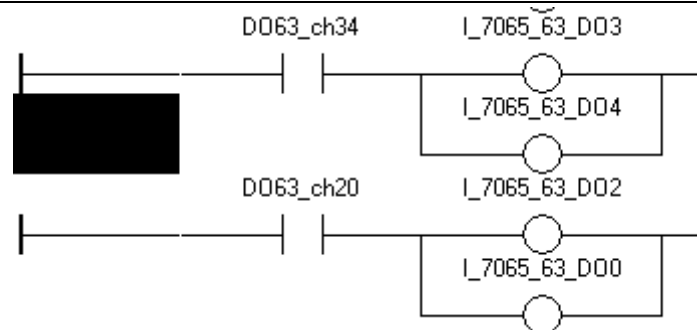


2. 將預先製作完成的圖案 Add to Library，並於 ObjectList 中選擇對應的圖案。

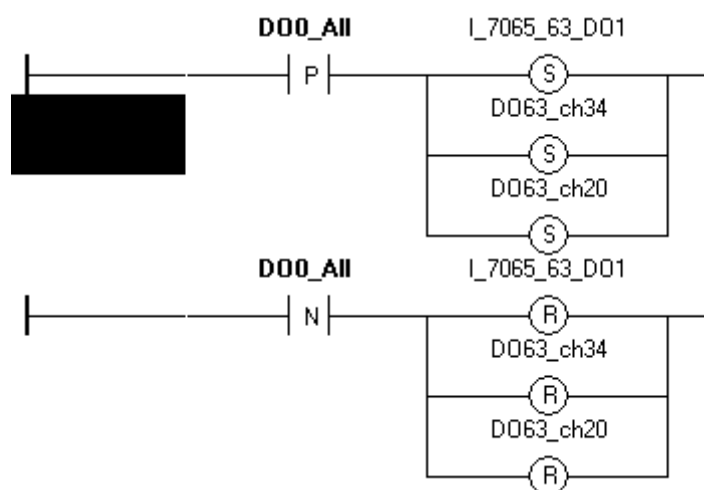
案。將 CheckBox 個別拉至適當的位置。



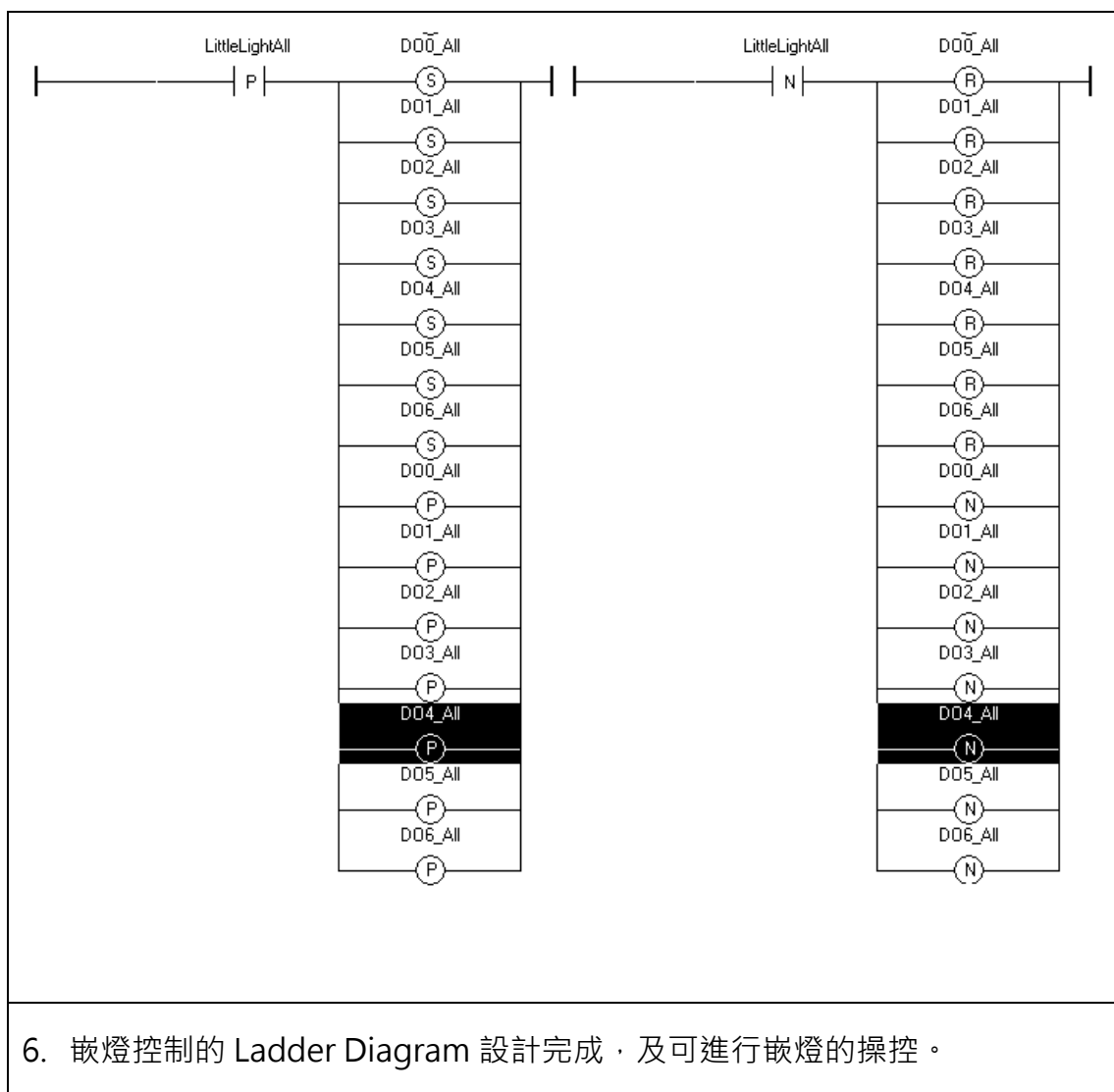
3. 因本範例的嵌燈數目太多，因此建立多個 Virtual Value 作為 DO Channel 的整合，因每一排之控制均相同，因此以前臺嵌燈作為示範。於 DownLight(Program)中建立前臺嵌燈控制之 Ladder Diagram，下圖為對個別嵌燈進行控制，於其中建立了兩個 Virtual Value 並分別整合兩個 DO Channels，分別為前臺燈光中的第一與第二個燈光圖示，而第三個燈光則是利用個別 DO 進行控制。



4. 新增一個 Virtual Value : DO0_All 以作為控制前臺所有嵌燈之應用，並建立 DO0_All 之 Ladder Diagram。



5. 新增一個 Virtual Value : LittleLightAll 以作為控制所有嵌燈之應用，並建立 LittleLightAll 之 Ladder Diagram。

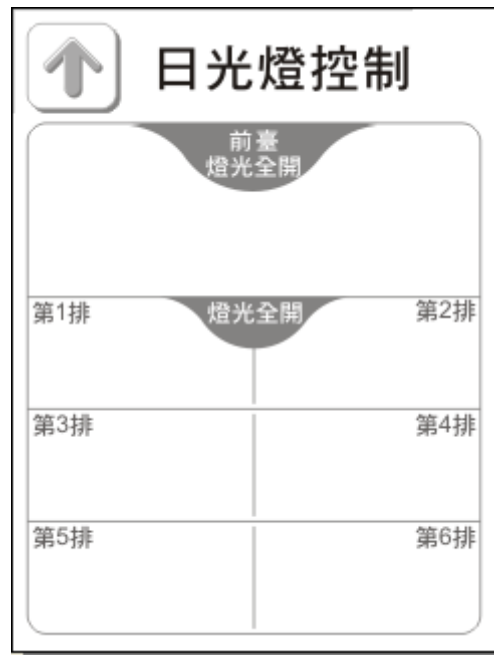


6. 嵌燈控制的 Ladder Diagram 設計完成，及可進行嵌燈的操控。

DemoRoom 製作流程與說明 – 日光燈控制

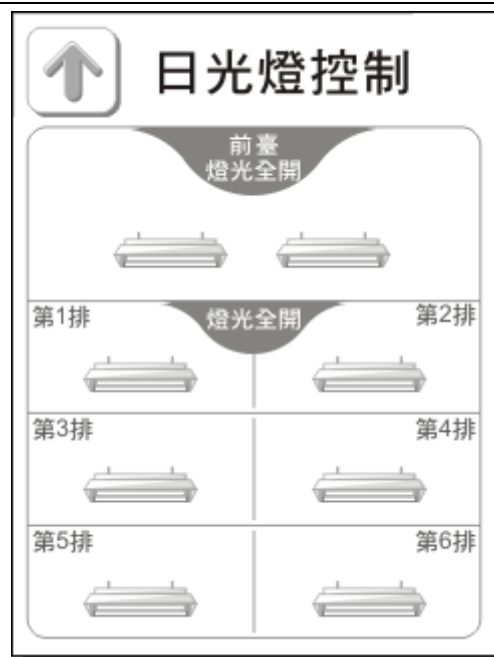
1. 將日光燈控制的背景圖 Load 至設計畫面並建立一個新的 Program:

Fluorescent lamp 。

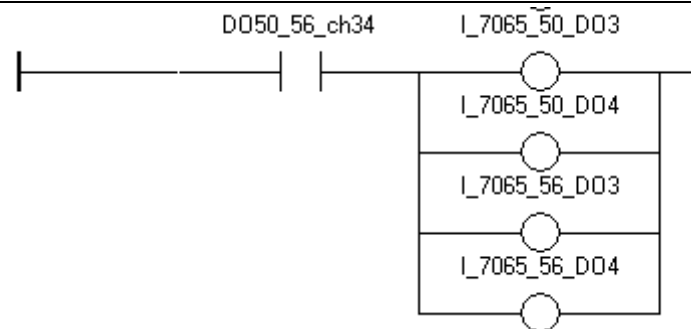


2. 將預先製作完成的圖案 Add to Library，並於 ObjectList 中選擇對應的圖案。

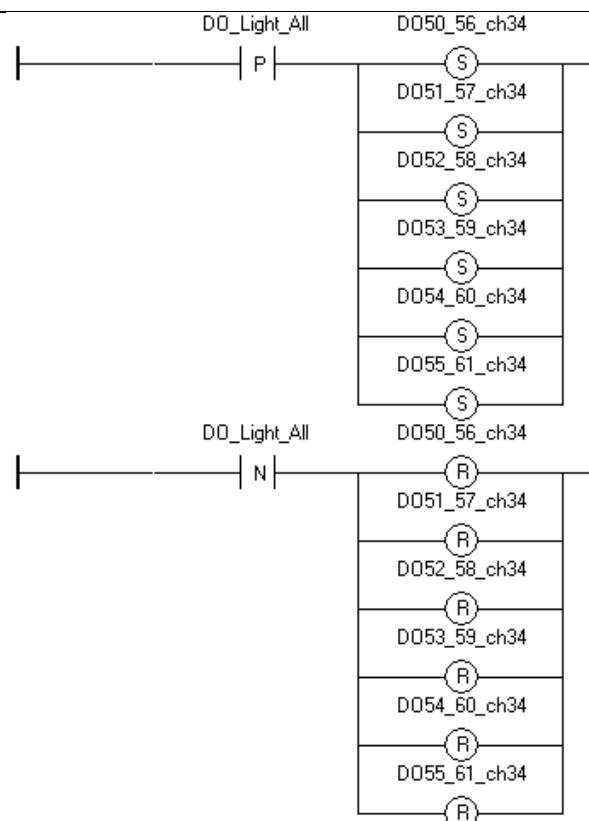
將 CheckBox 個別拉至適當的位置。



3. 因本範例的日光燈數目太多，因此建立多個 Virtual Value 作為 DO Channel 的整合，因每一排之控制均相同，因此以前第一排日光燈作為示範。於 Fluorescent lamp (Program)中建立第一排日光燈控制之 Ladder Diagram。



4. 新增一個 Virtual Value : DO_Light_All 以作為控制所有日光燈之應用，並建立 DO_Light_All 之 Ladder Diagram。



5. 日光燈控制的 Ladder Diagram 設計完成，及可進行日光燈的操控。

DemoRoom 製作流程與說明 – 情境控制

1. 將情境控制的背景圖 Load 至設計畫面並建立一個新的 Program:

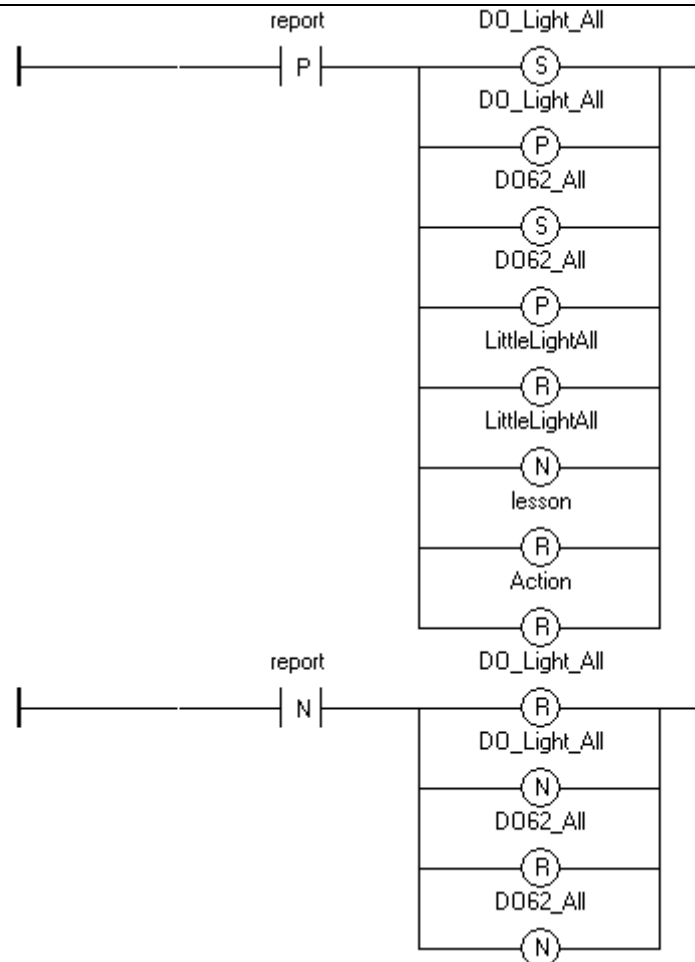
Environment 。



2. 將預先製作完成的圖案 Add to Library，並於 ObjectList 中選擇對應的圖案。將 CheckBox 個別拉至適當的位置。



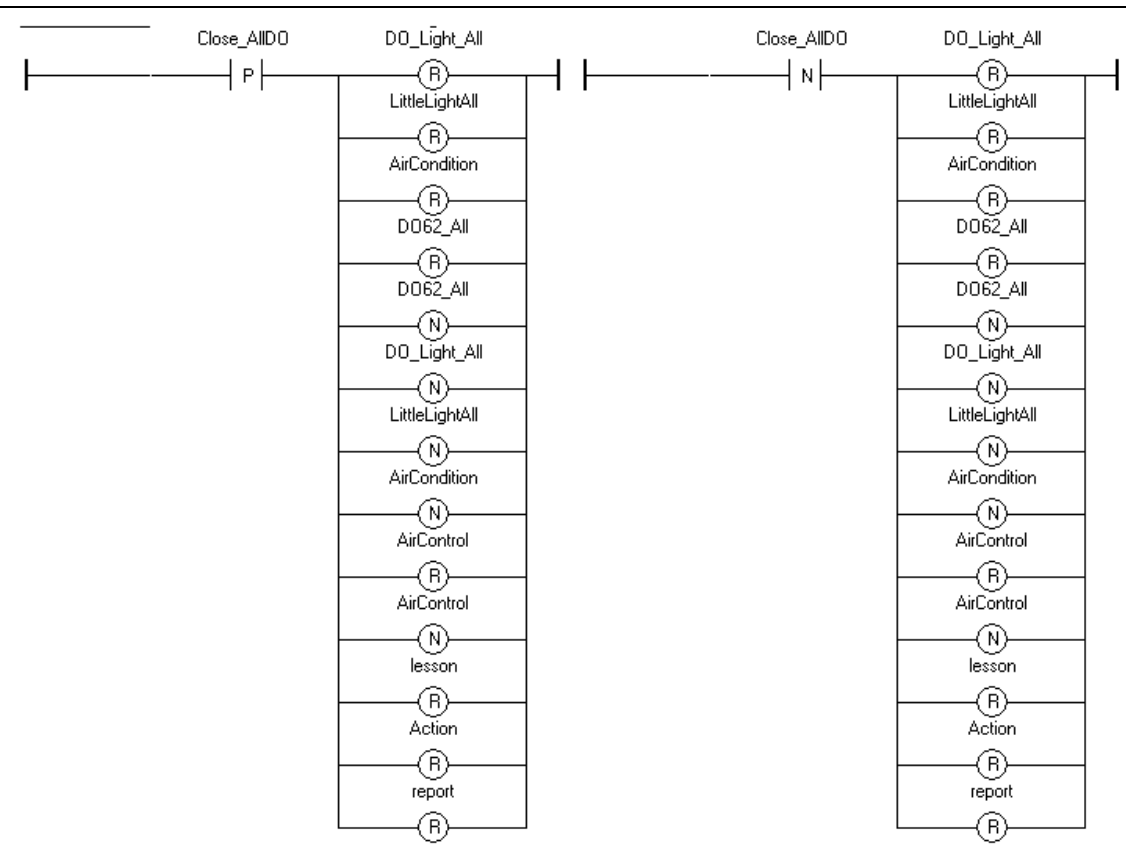
3. 以報告模式為例。新增一個 Virtual Value : report 以作為報告模式之應用，
建立上課模式之 Ladder Diagram。



4. 報告模式的 Ladder Diagram 設計完成，及可進行操控，當按下報告模式時，
則會開啟、關閉 Ladder Diagram 中設定的燈組。

DemoRoom 製作流程與說明 – 全部關閉

1. 本範例的全部關閉建立在首頁上，以便於離開時可以快速的關閉所有燈組與冷氣。當放置全部關閉之按鈕完成後，及建立全部關閉的 Ladder Diagram。



2. 全部關閉的 Ladder Diagram 建立完成後，只要按下此按鈕都是將所有燈組與冷氣關閉。

結論

依照以上步驟進行設計，即可完成與範例中相同的效果，開發者可依需求額外設計所需之控制邏輯來豐富控制效果。