

## 4. 電腦控制系統

智能家居實景分享

「舊屋」老家電也能智能起來！

<https://www.youtube.com/watch?v=gEw6ekrA8IQ>

### 4.1 控制系統

人類使用各種工具和機械時，都要適當地加以控制，否則失控的機械可能會引致意外和損失，例如：失去控制的汽車可能會撞傷行人和損壞財物。控制系統可以用來指引、命令或規律化各種工具和機械。控制系統可以簡單分為兩大類：開環式控制系統和閉環式控制系統。

#### (一) 開環式控制系統



圖4.1 開環式控制系統

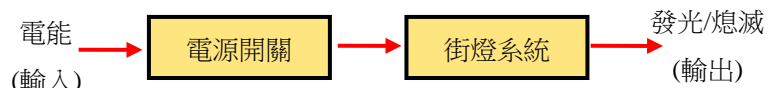


圖4.2 開環式控制系統例子 (街燈系統)

在一個簡單的開環式控制系統中，輸入信號會指示控制元件作出反應，從而獲得輸出的結果。明顯地，開環式系統是一個簡單的系統，它的輸出不會影響輸入。利用開關操作的街燈系統便是一個簡單的典型例子。

#### (二) 閉環式控制系統

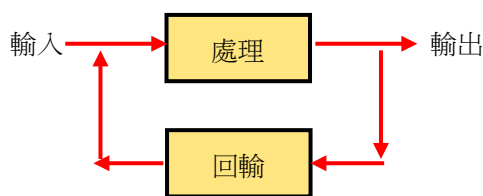


圖4.3 閉環式控制系統

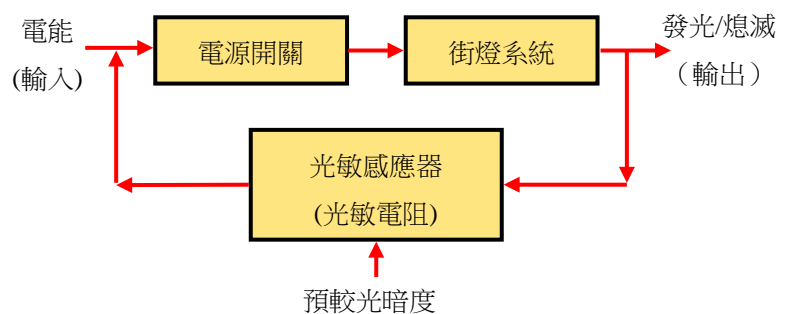


圖4.4 環式控制系統例子 (光敏控制街燈系統)

在閉環式控制系統中，輸出結果會產生信號來調校輸入，這種信號稱為回輸。回輸會比較輸出和預期的效果，然後作出適當的控制行動，以影響輸入信號和新的輸出。適當的回輸可以增加輸出的準確性。

如果在街燈系統中加入光敏感應器 (例如：光敏電阻)來偵察街道的光暗程度，便可以把開環式系統改為閉環式系統。在冬天或陰雨天的上午，當街上光線不足的時候，光敏感應器便可以產生適當的回輸，指示開關開啟街燈來照明。

## 4.2 電腦控制系統

電腦控制系統是利用電腦和程式來處理輸入和產生輸出的控制系統，與個人電腦系統相似，主要由「輸入-處理-輸出」三部分組成。電腦控制系統由各種感應器透過輸入埠來輸入信號，例如：開關、光敏感應器、接觸感應器和熱敏感應器等。輸出信號則透過輸出埠來操控機械或設備，例如：電動機、指示燈和揚聲器等。

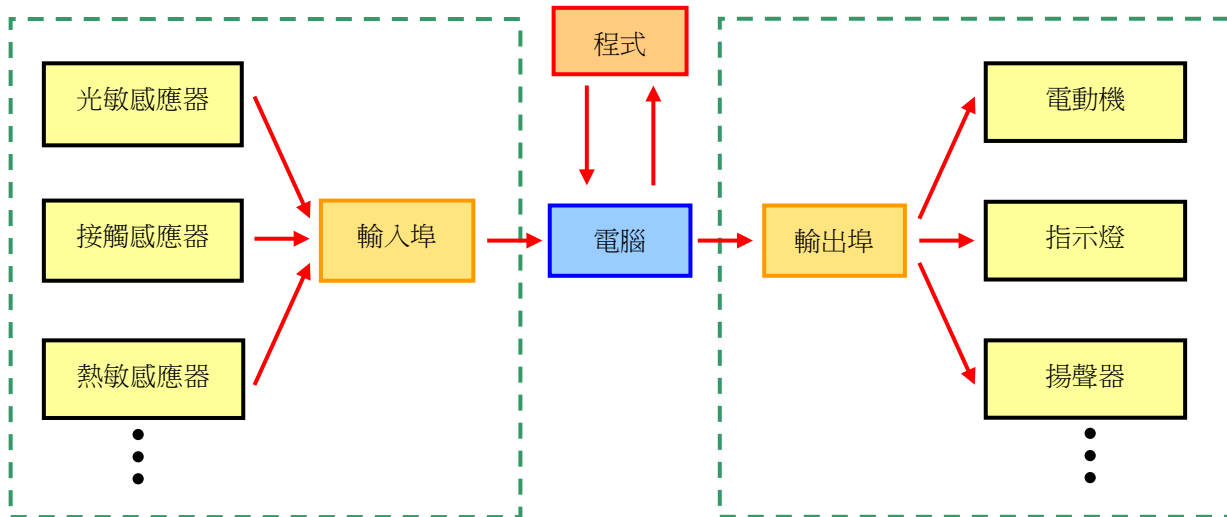


圖4.5 電腦控制系統示意圖

設計電腦控制機械時，一般會按需要而選擇合適的電腦，包括：專用電腦、個人電腦和微型處理器等。一些工業機械會配置特別設計的專用電腦，以提高機械的整體工作效率。個人電腦通常配置顯示器和鍵盤，它可以透過電腦控制介面咭來操控機械。微型處理器就好像是沒有顯示器和鍵盤的小型電腦，它的體積較細小，可以直接安裝在受控制的器具內，稱為嵌入式設計，例如：點矩陣顯示面板、電腦輔助操控的汽車和家庭電器等。



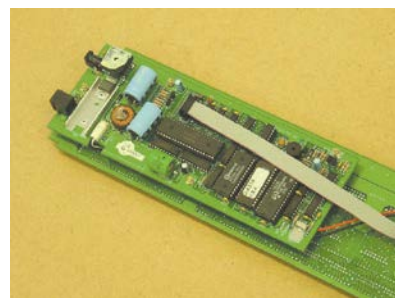
圖4.6 配有專用電腦的電腦控制機床



圖4.7 可安裝在個人電腦上的電腦控制介面咭



圖4.8 (a) 點矩陣顯示面板



(b) 操控面板的微型處理

在開環式電腦控制系統中，電腦程式只需要直接發出指令來驅動機械，例如：指示電動小車向前行駛 3 秒，然後停下。這種系統亦可以稱為電腦驅動系統。

在閉環式電腦控制系統中，電腦程式則需要考慮各種輸入元件的信號，然後才發出適當的指令來驅動機械，例如：指示電動小車向前行駛，直至接觸感應器碰到障礙物時才停下。閉環式電腦控制系統可以用來製造自動化機械，以提高工作效能。

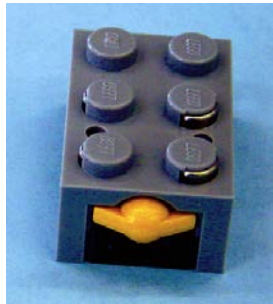


圖4.9 (a) 接觸感應器



(b) 安裝了接觸感應器

### 4.3 控制程式和軟件

由於電腦有多種型號，所以編寫控制程式的語言和方法亦有多種。它們大多利用句子形式的指令來組成控制程式，例如：Java、MSWLogo、PCLogo、QBASIC、Turbo Pascal、Visual BASIC 和 Visual C++等。設計和編寫控制程式後，便可以把它下載，然後執行和測試。

在執行控制程式時，電腦會按照程式中的指令和資料來控制各種輸出元件，例如：電動機轉動 5 秒、燈泡以最大亮度發光等。程式亦可以參考輸入元件所獲得的資料，然後再適當地控制輸出元件，例如：當接觸感應器(輸入元件)的接觸點被推入時，電腦便會控制電動機(輸出元件)向相反方向轉動。

```
public static void main(String args[])
{
    Robot r = new Robot();

    try
    {
        r.openPort("COM1");
    }
    catch (IOException e)
    {
        System.out.println("Error " + e);
    }

    //
    try
    {
        System.out.println("CHECKED " + r.checkM());
    }
    catch (IOException e)
    {
        System.out.println("Error " + e);
    }

    // test the sleep command
    System.out.println("before sleep 0.1, time" + System.currentTimeMillis());
    r.sleepOn(0.1);
    System.out.println("after sleep 0.1, time" + System.currentTimeMillis());

    System.out.println("before sleep 0.2, time" + System.currentTimeMillis());
    r.sleepOn(0.2);
    System.out.println("after sleep 0.2, time" + System.currentTimeMillis());

    System.out.println("before sleep 0.3, time" + System.currentTimeMillis());
    r.sleepOn(0.3);
    System.out.println("after sleep 0.3, time" + System.currentTimeMillis());

    System.out.println("before sleep 0.4, time" + System.currentTimeMillis());
    r.sleepOn(0.4);
    System.out.println("after sleep 0.4, time" + System.currentTimeMillis());

    // test the left command
}
```

圖13.10 電腦控制系統的程式



圖13.11 可用電腦控制的

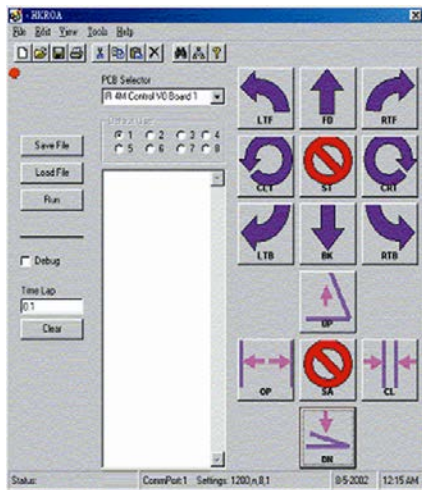


圖4.12 控制步行機械的軟件



圖4.13 可前後左右移動的步行機

不過，設計程式和輸入指令均需要不少時間，而且還可能會出錯。所以，可以先把一些常用的指令，預先組合為較簡單的子程式，例如：轉動不同的電動機，令步行機械向前、向後、左轉和右轉等。然後，利用這些子程式，便可以設計圖象介面的控制軟件。把軟件安裝在電腦上，便可以直接利用滑鼠和鍵盤來操控機械了，例如：香港機械奧委會便編寫了一些控制步行機械的專用軟件。

#### 4.4 設計步驟

電腦控制系統主要分為硬件和軟件兩大部分，硬件包括各種機械和裝置，軟件則包括各種用作操作、控制和通訊等工作的電腦程式。所以，設計電腦控制系統時，不但要分別設計硬件和軟件，而且還要考慮兩部分如何互相配合。下圖顯示電腦控制系統的基本設計步驟：

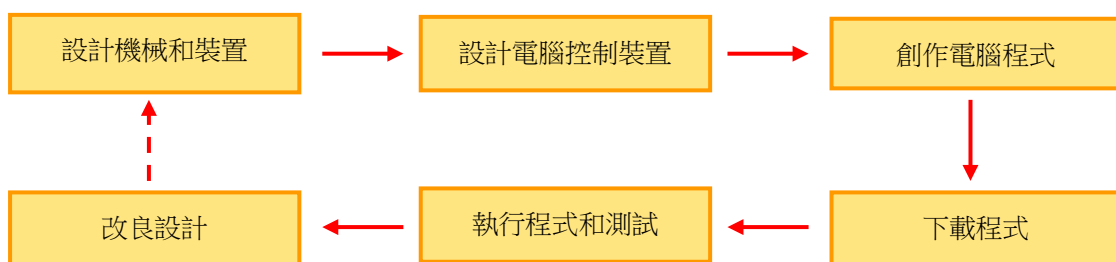


圖4.14 電腦控制系統的設計步驟

首先，按照系統的設計概要，設計和組合各種不同的機械和裝置，以便完成指定的工作。然後，設計該機械的結構，包括：支桿和齒輪組等機械元件。跟著，應考慮如何安裝適合的輸入和輸出元件，包括：電動機、燈泡和蜂鳴器等。你還要考慮採用哪種電腦來操作該系統，利用個人電腦還是嵌入式的微型處理器？

選擇了合適的電腦後，便要一種該電腦可以使用的電腦語言來編寫程式。你可以繪畫流程圖來組織該程式的邏輯。流程圖利用線條和箭咀來連接不同形狀、符號所代表的指令，常用來輔助編寫程式。

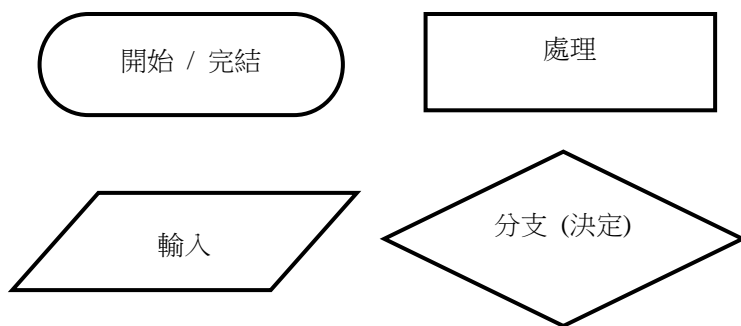


圖4.15 流程圖符號

最後，下載程式到電腦的儲存裝置內，便可以執行程式和測試設計。根據測試的結果，適當地改良設計，直至該系統的硬件和軟件均能完成設計的要求。

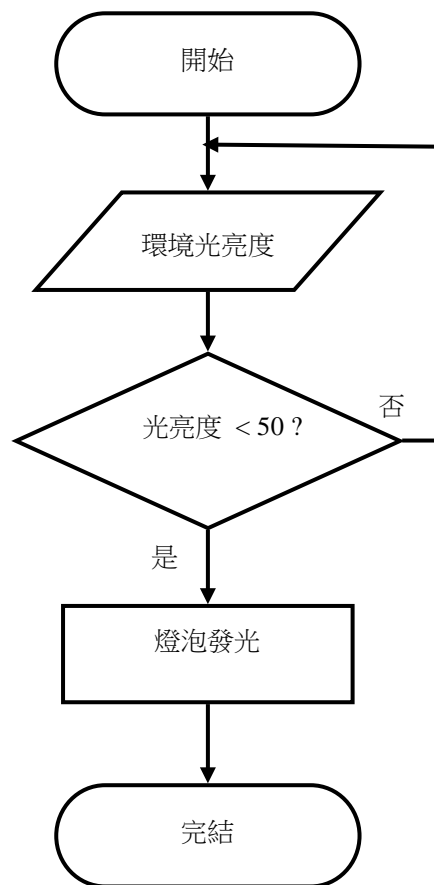


圖4.16 流程圖例子