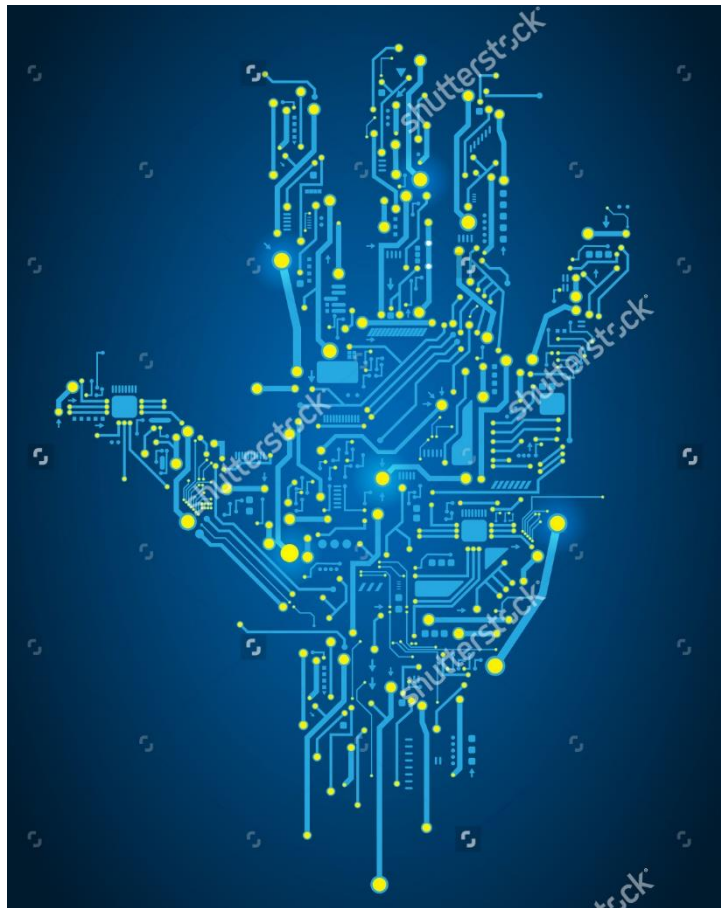


AOOP 期末專題

Final Project

主題：電路模擬系統

Fantasy Circuit



作者

108 級電機 B 班 0410746 林志恩

108 級電機 B 班 0410766 陳胤銓

一、問題描述

使用的 IDE：

Visual Studio 2010, C++, Windows Form Application

製作動機與功能：

動機：

有鑑於我們在電機領域中學習了許多重要的課程，包括電子學、電路學、線性代數、微分方程等等，我們希望能夠結合這些以及 OOP 程式設計的技巧，製作出能夠媲美電子實驗課會用到的 LTSpice 的電路模擬期末專題。

除了能夠活用我們在蔡媽講授的 AOOP 程式課程中所學到的各種知識外，透過實作專題，不僅能夠加深我們對其他相關課程的印象，並且能夠結合程式設計，製作出實用的電路系統，同時也大大提升我們的實力。

功能：

我們所設計的 Fantasy・Circuit 能夠讓使用者繪製出想要分析的電路圖，包括 RL、RC、RLC 電路等等。建立好圖形之後，系統會產生一個 Asc 的檔案，User 可以讓系統讀取 Asc 檔案或是存到硬碟之中，系統再將 Asc 的檔案轉成方便運算的二維陣列，最後將二維陣列轉換成矩陣讓系統能夠運算出各個節點的分壓與電流，並且繪製出像是示波器的圖形，讓使用者一目瞭然並得到自己想要的結果。

二、解法分析

遇到的問題與解決：

1. 使用別人已經做好的線性代數 Library 時 Armadillo+Lapack+blas，在設定環境的部分非常困難，除了要找到正確的 Armadillo 的 Header file 能夠在 VS2010 環境下進行編譯，還要 LINK 兩個.lib 的檔案，一開始 LINK 都沒問題直到 Run-time 的時候會跳出錯誤視窗，後來猜測 dll 檔案有問題，更換新的之後再把編譯的設定改成 Release 就能夠正常執行了。
2. 分析圖上的線段是否為同一個節點，一開始我們並不知道該如何分析，後來想到的分析方法是，先建立一個 `vector<set<pair<int, int>>> v`，然後讀入線段的兩個端點並檢查這條線上的點是否已經存在 vector 的每一個 set 之中，紀錄這些點存在 `v[i]` 的 set 裡面將這些 set 合併刪除重複的 set 只留下 index 最小的 set，把讀入線段的點全部 insert 到這個 set 裡面，把所有線段讀完之後，就完成了，結果是 `v[i]` 的 index `i` 即為節點的編號。
3. 繪圖需要用到 Panel 的 DoubleBuffered 功能，讓 Panel 畫圖時更加順暢。新增 `PanelDoubleBuffered.h`，並在 `Form1.h` 新增下列程式碼：

```
static_cast<MyPanel^>(Circuit_Map)->SetStyle(ControlStyles::AllPaintingInWmPaint, true);  
static_cast<MyPanel^>(Circuit_Map)->SetStyle(ControlStyles::DoubleBuffer, true);
```

4. 在動態產生的 PictureBox 上，我們需要利用以下方式來新增事件：

```
pic->Click += gcnew System::EventHandler(this, &Form1::pic_Click);  
pic->MouseDown += gcnew MouseEventHandler(this, &Form1::pic_MouseDown);
```

值得注意的是，不同事件的控制方式各有不同，需要多加留意。

數學推導：

1. 矩陣建立並求出 node 電壓

先將元件儲存的值，轉換成 `complex<double>` 的複數形式，利用電路學技巧，解出交流電路。

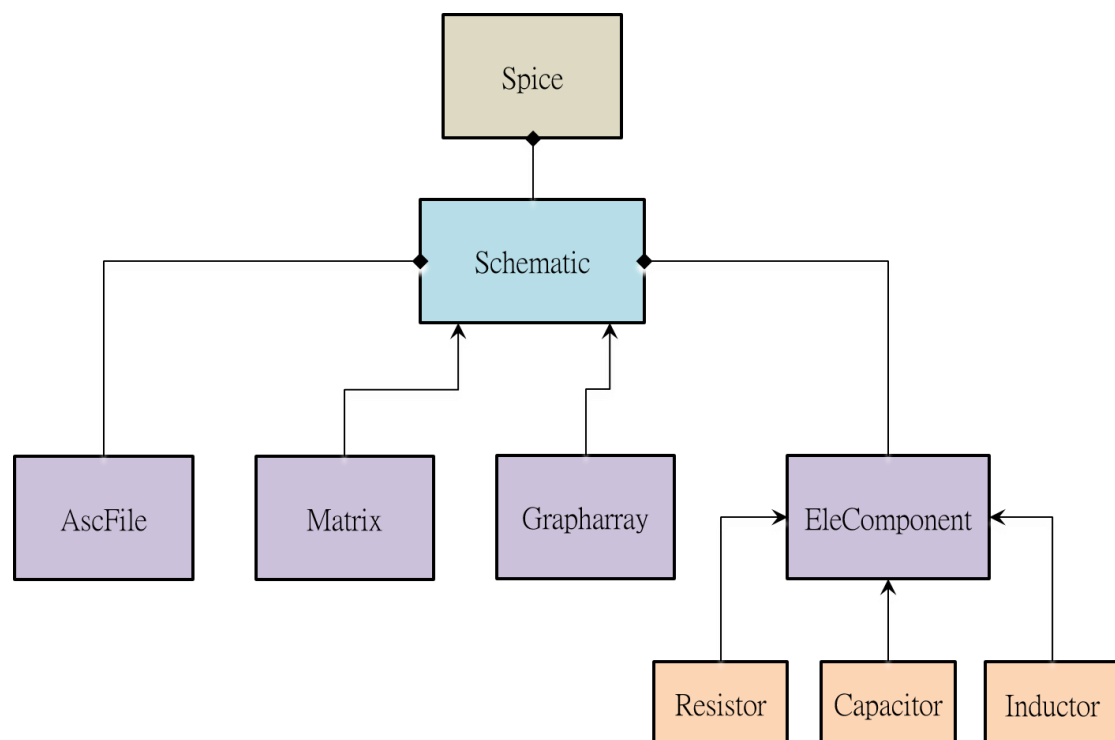
首先透過 Node 的搜尋，可以得到電路中 Node 的數量。在利用迴圈，搜尋每個元件的同時，搭配 Node 的搜尋便可以得到元件兩端各是屬於 Node 幾。

建立好 $n \times n$ 的二維陣列後 (n 為 Node 的數量)，便在搜尋每個元件的同時，在陣列中加上各屬於兩端位置的 equation 係數，利用到的是電路學所教的 node equation。比較麻煩的是，在手算電路時，接地端與電壓源都可以很方便就能看出之間的關係，但是程式在搜尋元件時，兩旁的 node 各自是接地或是接到電壓源難以分辨。因此需要預先搜尋過並另加 flag 的方式，來讓第二次再搜尋時更方便地建立出正確的二維陣列，但是有些例外的連接方向仍需特別作考慮。最後，使用此二維陣列當作矩陣，搭配 Armadillo+Lapack+blas，求出反矩陣後相乘，便能解出各 node 的電壓值矩陣。

2. 畫出波形圖

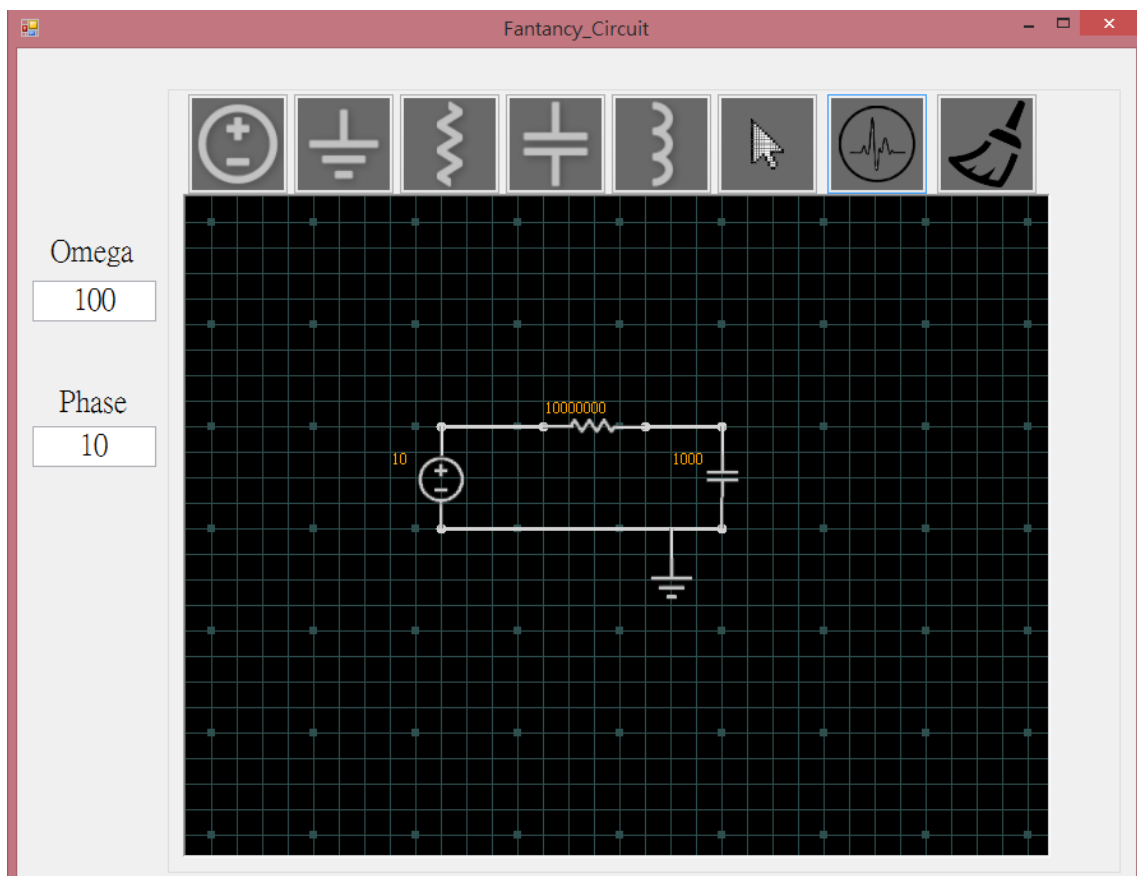
固定在螢幕上畫出兩個周期的波形，並利用已知的 ω 以及算好的 node 矩陣，搭配電路學所學的 phasor，將複數值轉換成實數值，並透過縮放比例的方式畫在 panel 上，彷彿模擬示波器一般的效果。

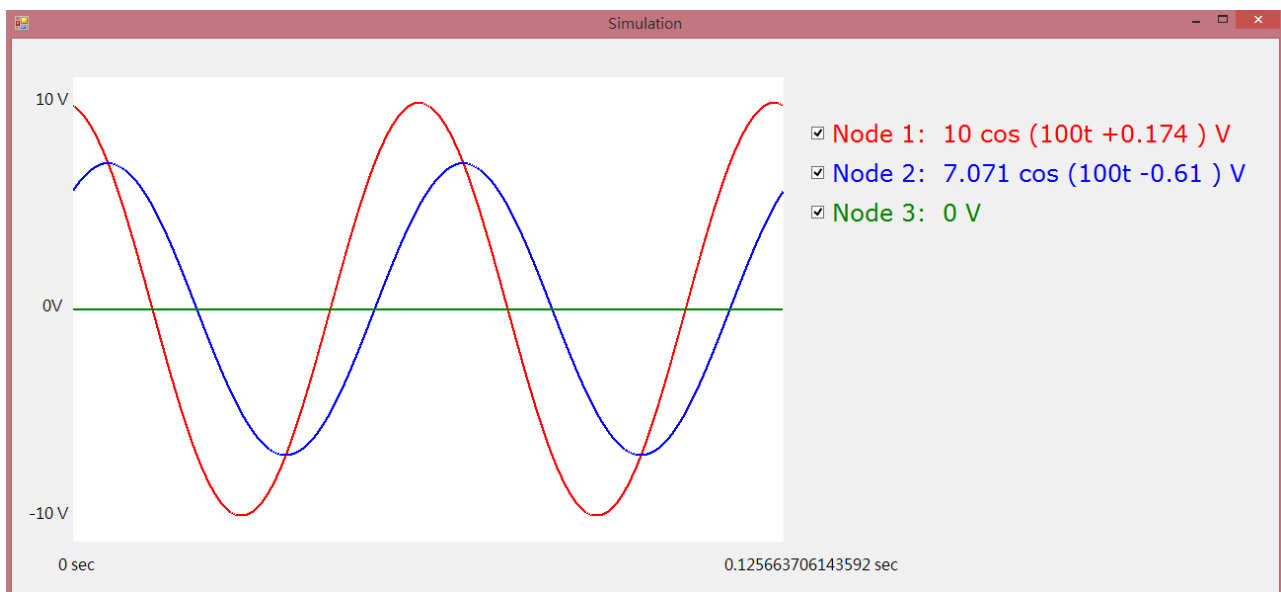
三、架構設計



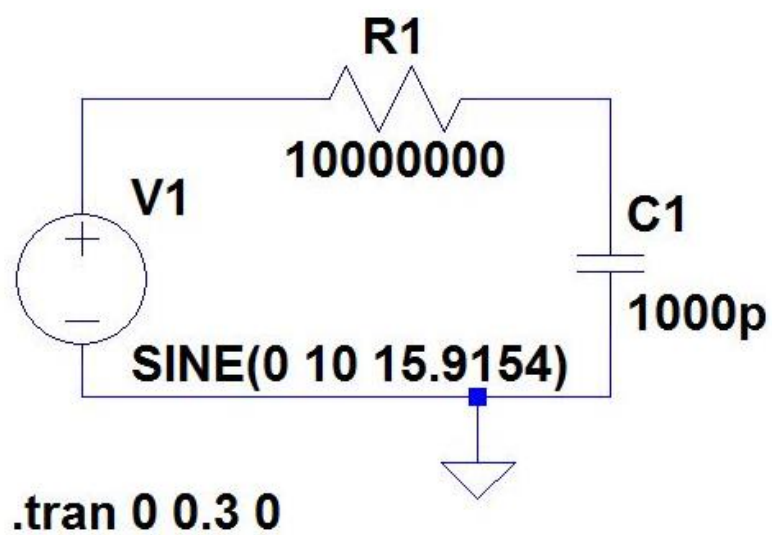
四、結果分析

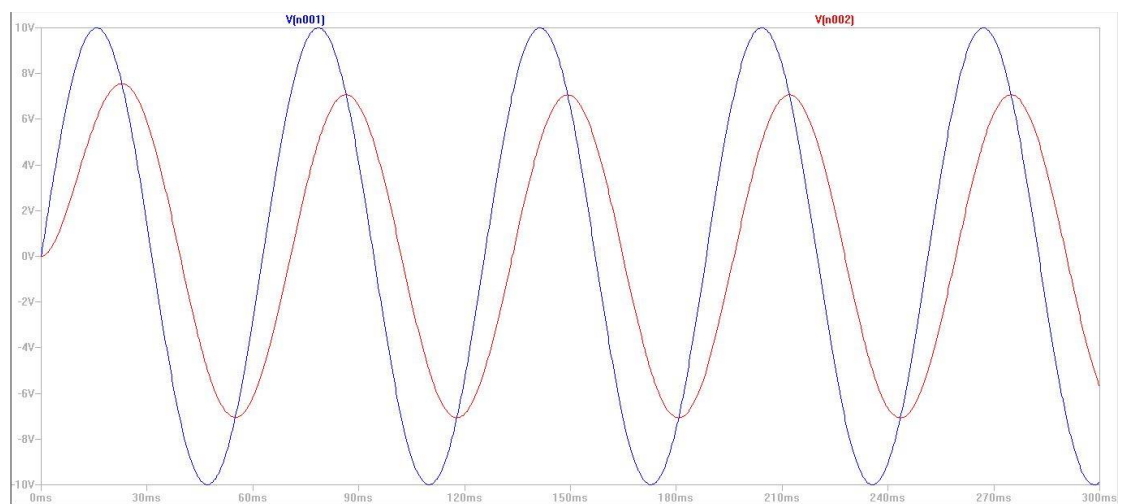
RC 電路：



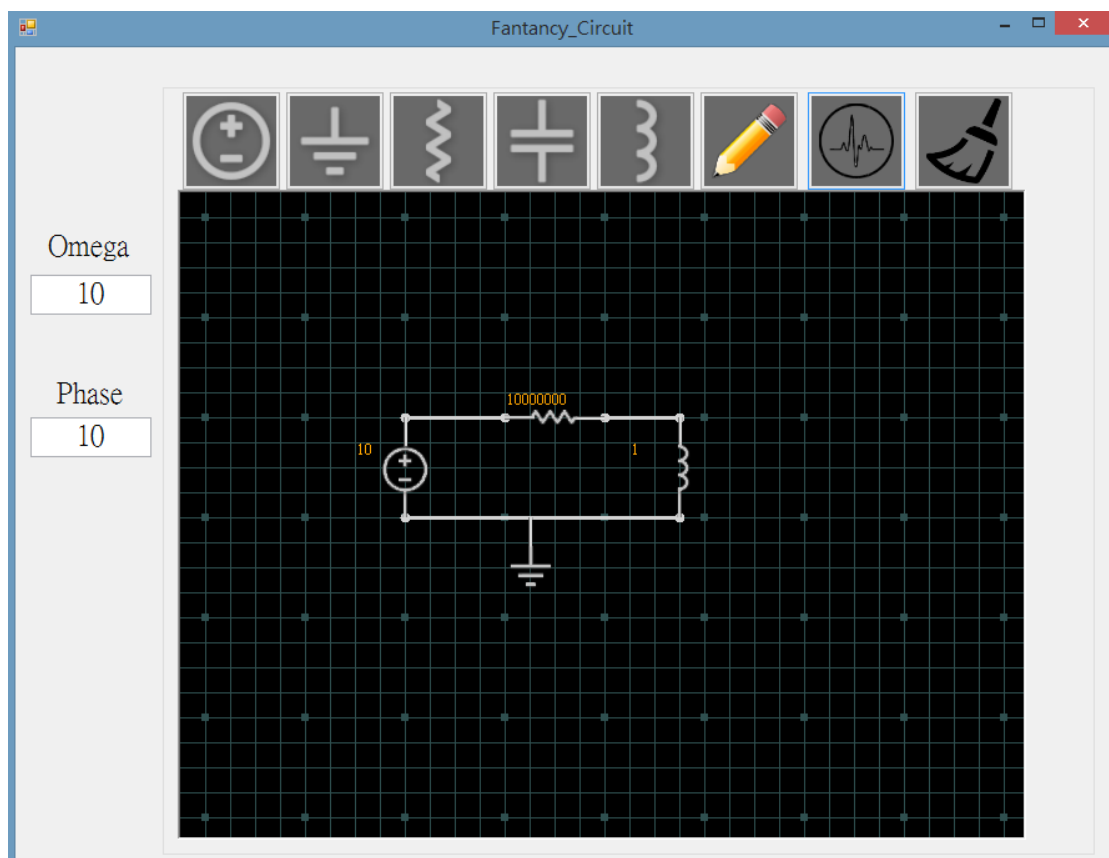


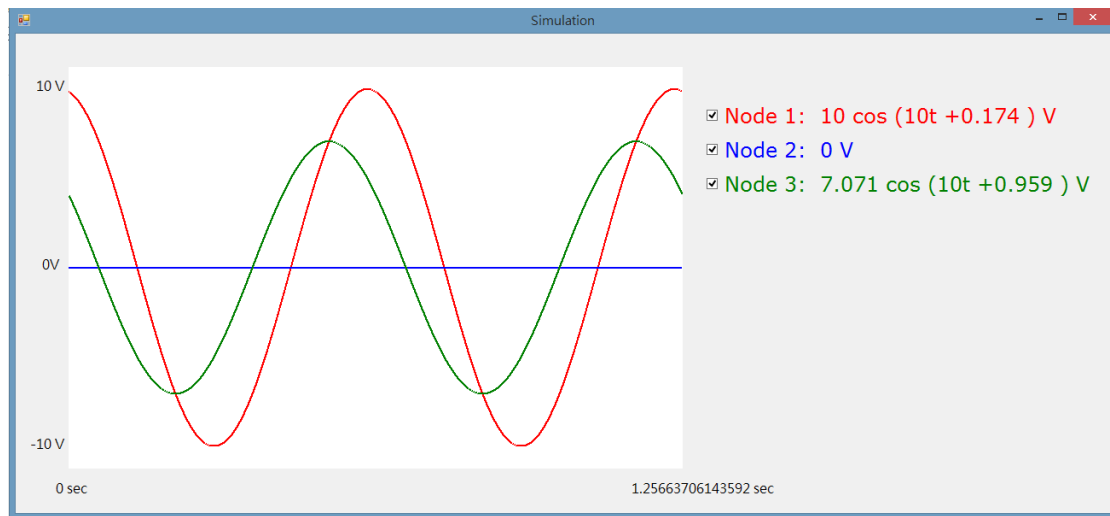
LTSpice 模拟：



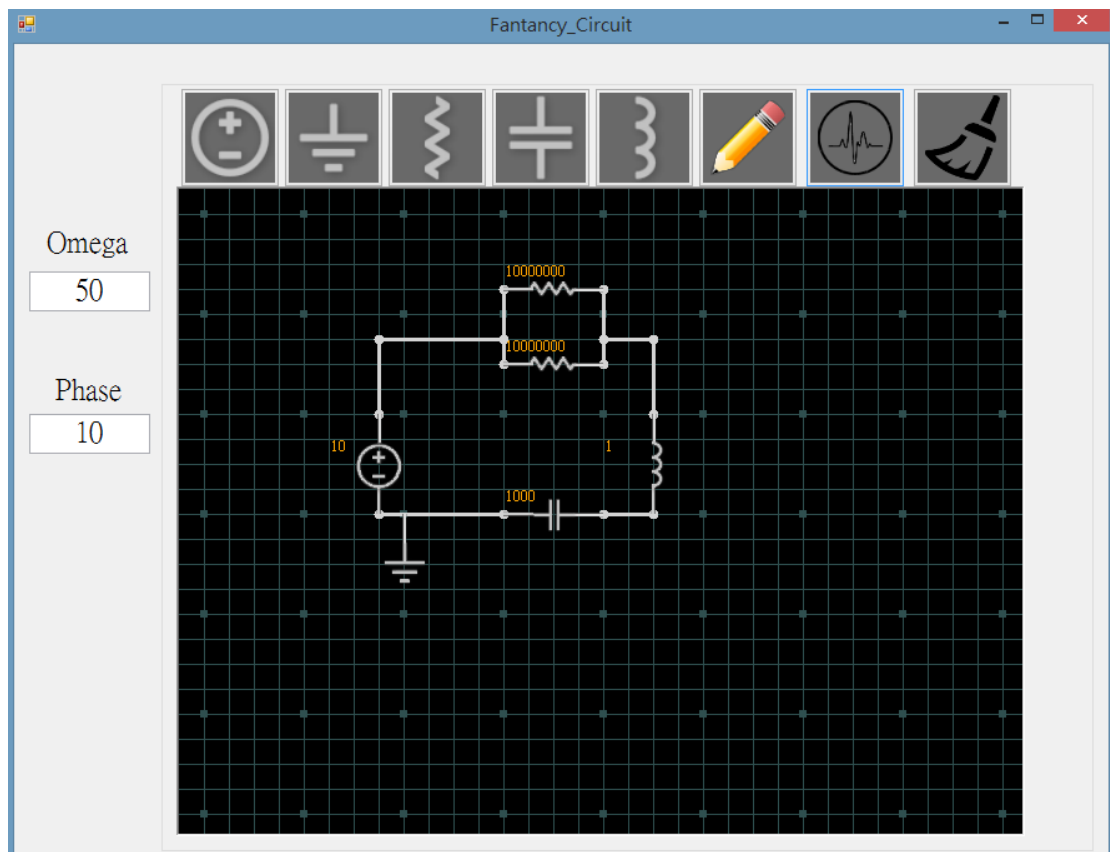


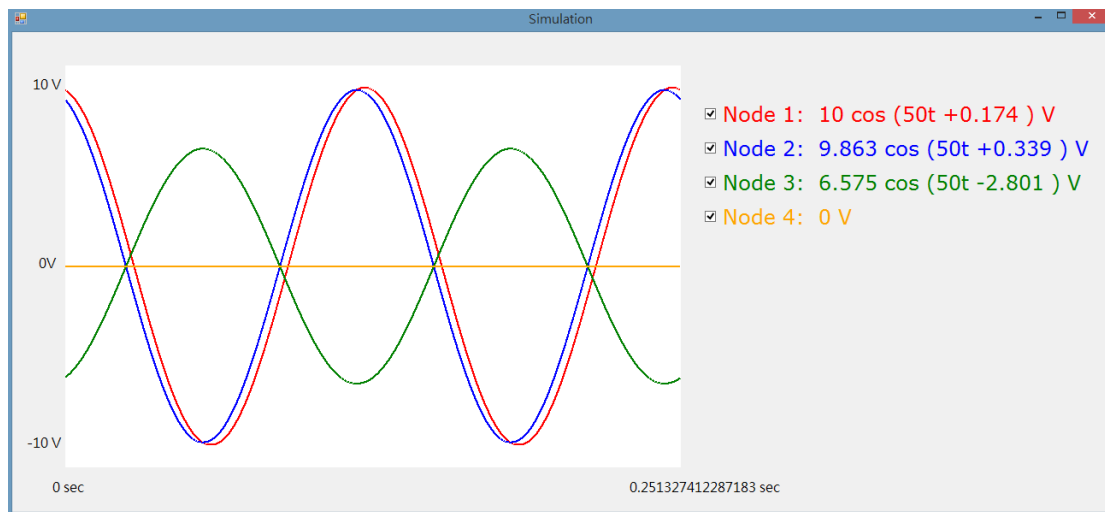
RL 電路：





RLC 電路：





五、心得展望

林志恩：

這次的 AOOP 期末專題經歷了重重考驗，從一開始構思架構圖到開始製作，常常遇到不知道該怎麼繼續的階段，但是我們熬過來了，最後終於將我們的電路模擬系統完成。從一開始設立元件、判斷節點、產生 Ascfile、建立矩陣、解出各 node 的電壓，以及畫出波形圖，每項任務都需要花費很大的工夫去完成。

這段路程上，收穫最大也學習最多的一定是我們自己，除了將上課所學的東西融會貫通，還要想出辦法將 idea 付諸實行，遇到困難要懂得發問並找尋辦法解決，真的有太多事物是平常上課所學不到的。當我們看到最後執行的結果，心裡真的是非常有成就感，一切的努力與辛苦終於化成甜美的果實。

陳胤銓：

從一開始上台發表自己對於電路模擬系統的想法時，我才驚覺原來我理解錯誤了，原本以為電路分析是用判斷並聯和串聯一路解下去就可以了，直到那時候助教點醒了我們，有些接法既不是串聯也不是並聯，因此要用節點分析法，列出方程式，再把係數抽出來，轉換成 $AX=B$ 的線性代數問題，最後將兩邊乘上 A 的反矩陣就能夠解出 X 了，在這過程中遇到了許多困難，不過值得高興的是，我瞭解了如何再編譯的時候 LINK 到 lib 檔案，lib 檔案又會去呼叫 dll 檔案，真的獲益良多，感謝這次的專題中有我的隊友能夠持續督促互相幫忙，感謝助教在繁忙的期末也撥空為我解惑。

六、專題分工說明

林志恩：

元件動態產生

元件動態事件

元件的拖曳與自動貼齊格子點
解出交流電路，將各元件值轉換成複數
判斷元件順序，使用 Node equation 進行列式
將 Node equation 的式子轉換成矩陣
解出矩陣後，將各節點的複數值，轉換成 Magnitude 與 Phase
畫出波形圖
Form1 美化
波形界面美化
Debug
程式優化

Spice – Schematic - Grapharray
Spice – Schematic – Component
Spice – Schematic
Void V_button_Click(System::Object^ sender, System::EventArgs^ e)
Void CreatePictureBox(Control^ parent, String^ boxname, int x, int y, int w, int h)
Void pic_MouseDown(System::Object^ sender, System:: Windows:: Forms:: MouseEventArgs^ e)
Void pic_MouseMove(System::Object^ sender, System:: Windows:: Forms:: MouseEventArgs^ e)
Void pic_MouseUp(System::Object^ sender, System:: Windows:: Forms:: MouseEventArgs^ e)

陳胤銓：

元件的圖片旋轉控制
元件標示元件的值
元件雙擊設定元件值
電路圖上的畫線功能
使用 vector 記錄元件
使用 vector 記錄畫線的端點坐標
判斷線段的連接，自動給定節點編號
判斷元件所連到的節點
解出線性方程解(Armadillo)
畫出波形圖
Form1 美化
波形界面美化

Debug

程式優化

Spice – Schematic – AscFile

Spice – Schematic – CalcNode

Spice – Schematic – Matrix

Spice – Schematic

Void Circuit_Map_Paint()

Void pic_KeyUp(System::Object^ sender, System:: Windows:: Forms::

KeyEventArgs^ e)

Void pic_Click(System:: Object^ sender, System:: EventArgs^ e)

Void pic_DoubleClick(System::Object^ sender, System::EventArgs^ e)

int GetPicIndex(String ^str)

Void Pen_Button_Click(System::Object^ sender, System::EventArgs^ e)

Void Circuit_Map_MouseDown(System::Object^ sender, System:: Windows::

Forms::MouseEventArgs^ e)

Void button1_Click(System::Object^ sender, System::EventArgs^ e)

七、參考資料

Google 等網路資訊

助教在 Lab 所教的內容與其上課的檔案

同學指點並自己融會貫通

諸如：panel 的 Doublebuffered 使用方法、gcnew 物件的使用方法、Armadillo 函式的使用等等。