

# 邏輯系統實驗 實驗三

組別：7

成員：章子嚴、張軒、魏晉成

學號：E24065018、E24066129、E24066226

實驗內容：

## 一、基礎題(一)：簡易電話系統

這個電路由兩個部分組成，分別是畫面右半(或邏輯 IC 的上半部)的二對一多工器，以及畫面左半(或邏輯 IC 的下半部)的一對二解多工器。

多工器的原理，是利用 Input A(最上端黃色線)與 NOT(最上方的邏輯 IC)後的 Sel0(指撥開關最上方的 bit，綠線)，進行 AND(中間的邏輯 IC)，同時 Sel0 與 Input B(黃線下方的藍色線)進行 AND，而這兩個 AND 的輸出，再進行 OR(最下方的 IC)，輸出至解多工器的 Input。

而解多工器的 Input 則會分別與 Sel1(指撥開關最上方的 bit，藍線)以及 NOT(最上方的邏輯 IC)後的 Sel1 進行 AND，分別得到 Output C 與 Output D。

Sel0: 指撥開關的 MSB，綠線。

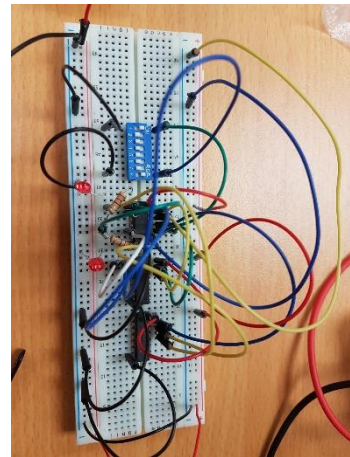
Sel1: 指撥開關的 LSB，藍線。

Input A: 最上端黃線

Input B: 黃線下方的藍線

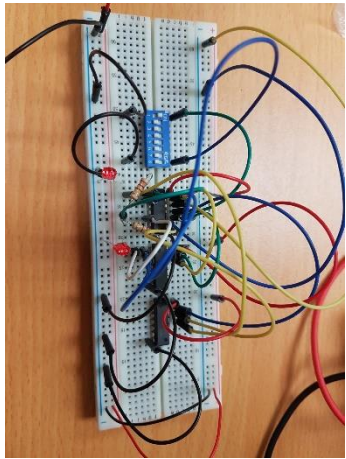
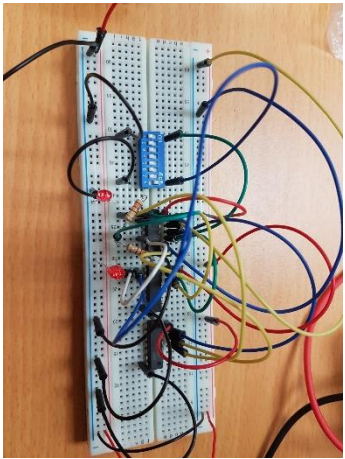
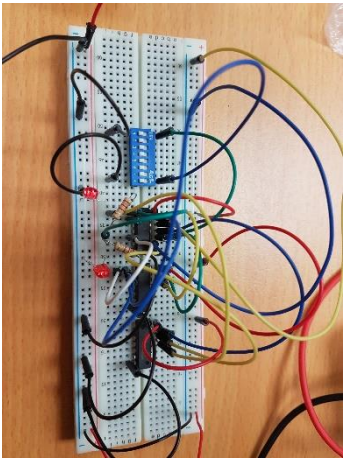
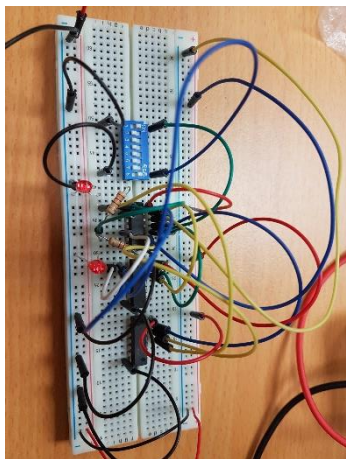
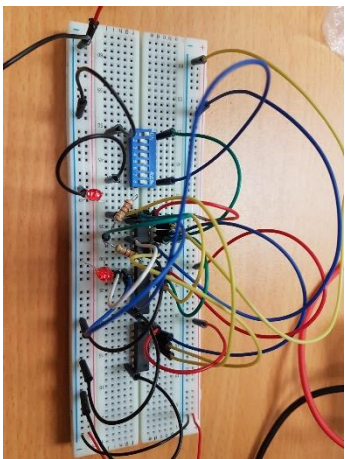
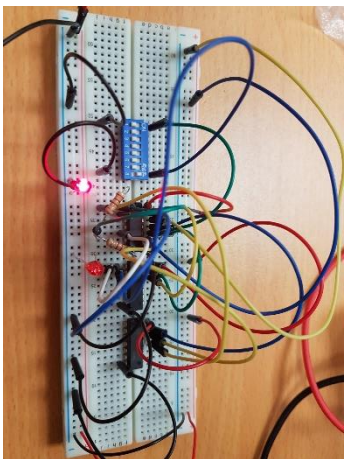
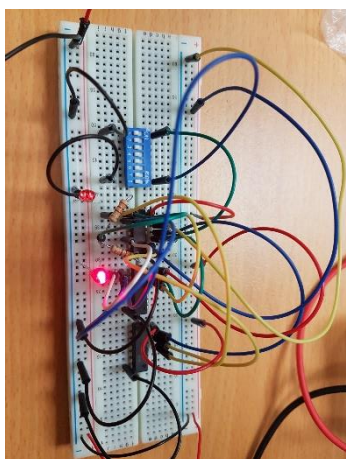
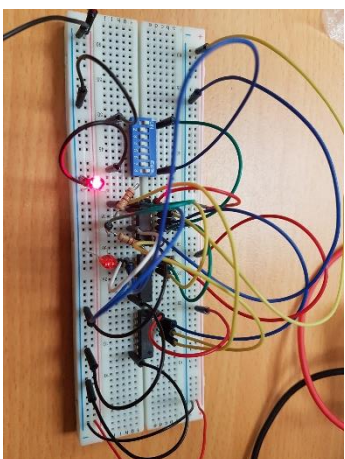
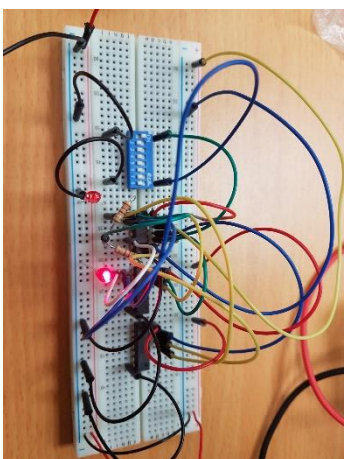
Output C: 較上方 LED 燈

Output D: 較下方 LED 燈

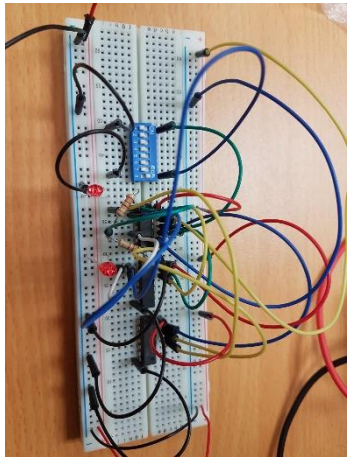
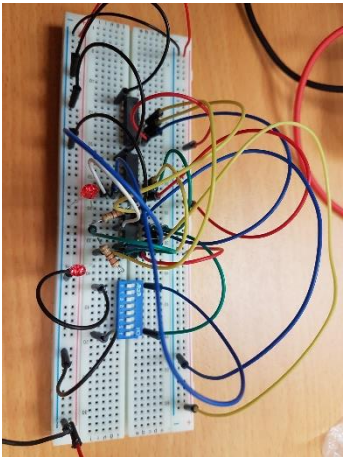
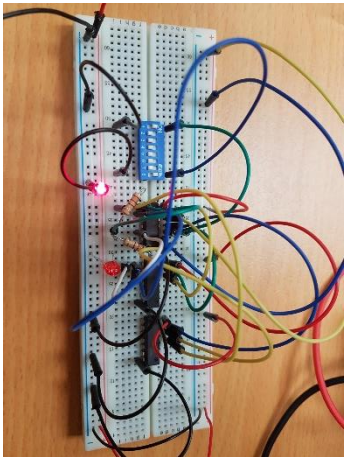
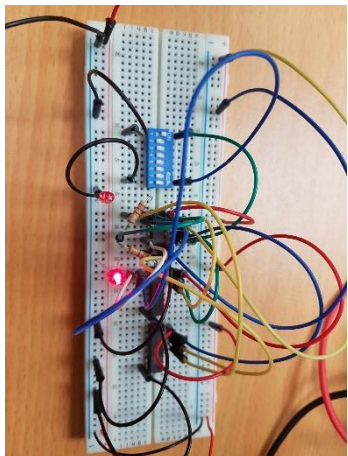
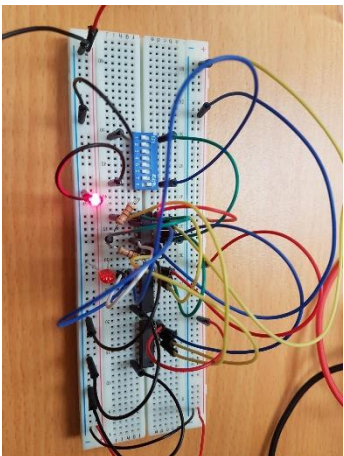
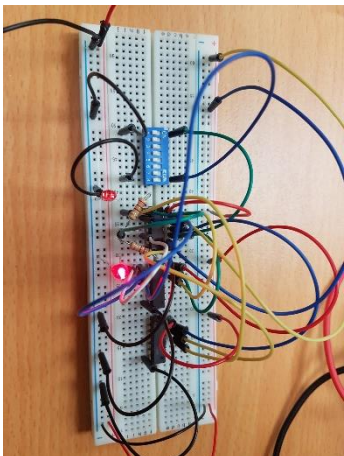


{Sel0, Sel1, Input A, Input B}, {Output C, Output D}

0000, 00

		
0001, 00	0010, 00	0011, 00
		
0100, 00	0101, 00	0110, 10
		
0111, 01	1000, 10	1001, 01



		
1010, 00	1011, 00	1100, 10
		
1101, 01	1110, 10	1111, 01

## 二、 基礎題(二)：四對二優先權編碼器、挑戰題：簡易保全系統

由於兩者電路有相關性，一方編碼一方解碼，所以我直接將其兩份電路進行組合。

下半電路：

由 NOT(左側)、AND(中間)以及 OR(最右側)進行 4 to 2 encoder 的連接，而電路輸入由同塊麵包板左上方由左至右的 Input 0(淡黃色)、Input 1(白色)、Input 2(綠色)以及 Input 3(橘色)作為 Input，經過中間邏輯閘的處理後，輸出至同塊麵包版中上方的 LED 燈做為 Output A，並且由橘色線連接至上半部的解碼器；下方中間的 LED 則為 Output B，並且由白色線連接至上半部的電路；右下方的 LED 燈則是 Enable，其則透過橘色線連接上半部電路。

首先，由於  $\text{Enable} = \text{in0 or in1 or in2 or in3}$ ，所以當所有輸入皆為 0 時，Enable 為 0；而當任意輸入變為 1 時，Enable 就會變成 1。

其次來討論 A，因為  $A = \text{in0}' \text{ and in1}'$ ，所以當 in0 或 in1 為 1 時，A 就會等於零，因此在 1, X, X, X 以及 X, 1, X, X 的組合中，A 都會是 0；反之，只要為 0, 0, X, X 的組合，A 都會為 1。

最後討論 B，因為  $B = (\text{in0}' \text{ and in1}) \text{ or } (\text{in0}' \text{ and in2}') = \text{in0}' \text{ and } (\text{in1 or in2}')$ ，因此是 1, X, X, X 的組合以及 X, 0, 1, X 的組合，B 都會是 0；反之，如 0, 1, X, X，0, X, 0, X 的組合，B 都為 1。

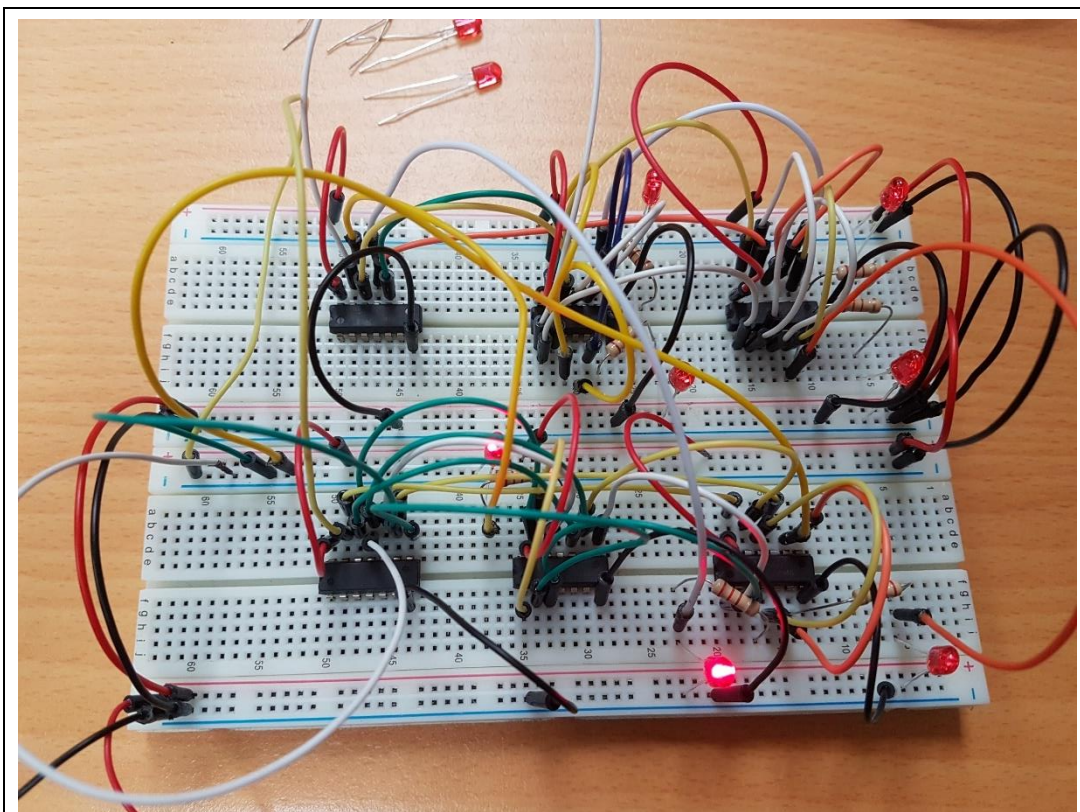
統整下來，在 1, X, X, X 中，A, B=0，Enable=1；0, 1, X, X 的組合中，A=0、B=1，Enable=1；0, 0, 1, X 的組合中，A=1、B=0、Enable=1；0, 0, 0, 1 的組合中，A=1、B=1、Enable=1；最後，0, 0, 0, 0 的組合中，A=1、B=1 但 Enable=0。

上半電路：

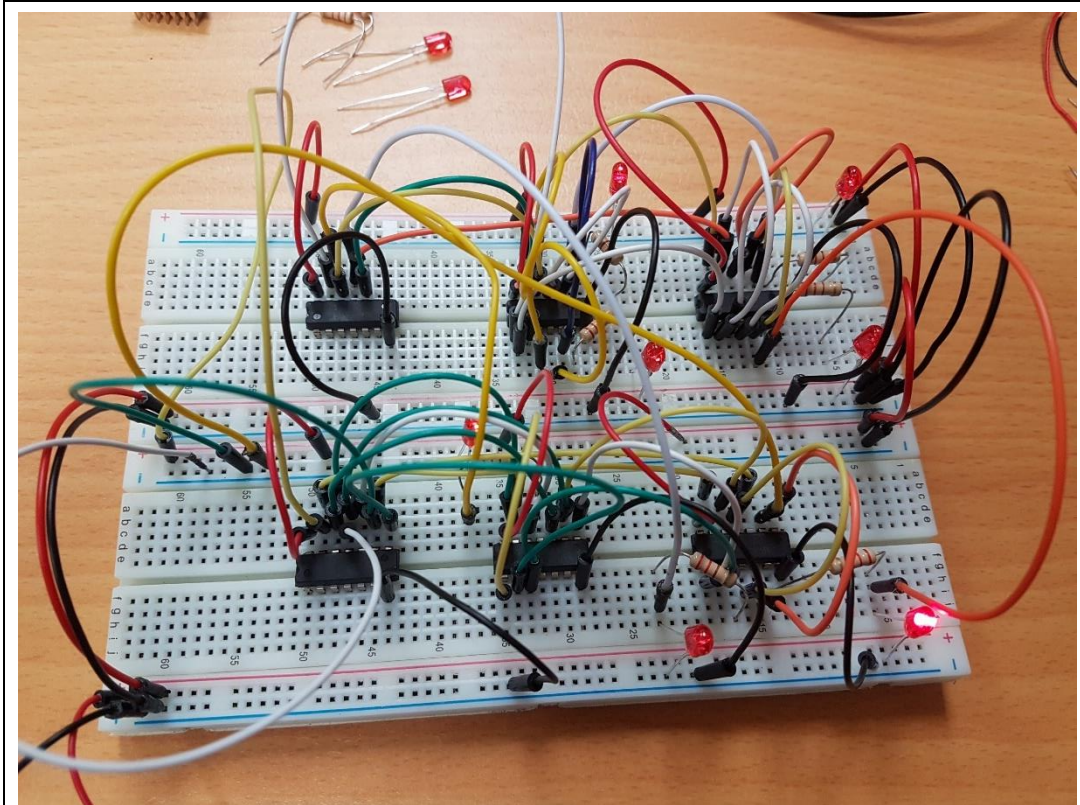
由 NOT(左側)以及兩個 AND(中右)組成的 2 to 4 Encoder，由 Input A、Input B、Enable 以及 Out 0~3(左上、右上、左下、右下)組成。

當 Enable 為 0 時，Out0~3 皆為 0，不論 A、B 的值；而當 Enable 為 1 時，以下 AB 的組合會導致不同的輸出亮起，00 為 Out0，01 為 Out1，10 為 Out2，11 為 Out3，同時對應到優先編碼器的 1XXX、01XX、001X、0001 的四種組合。



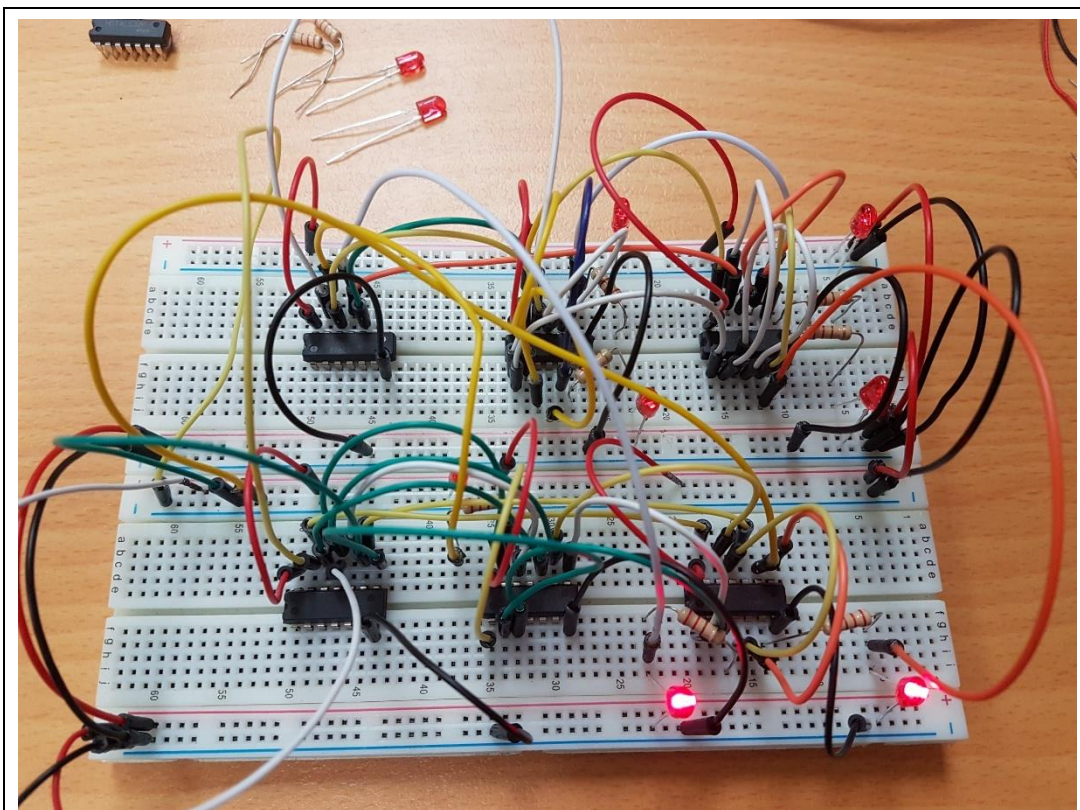


Input=0000, ABE=110, Output=0000

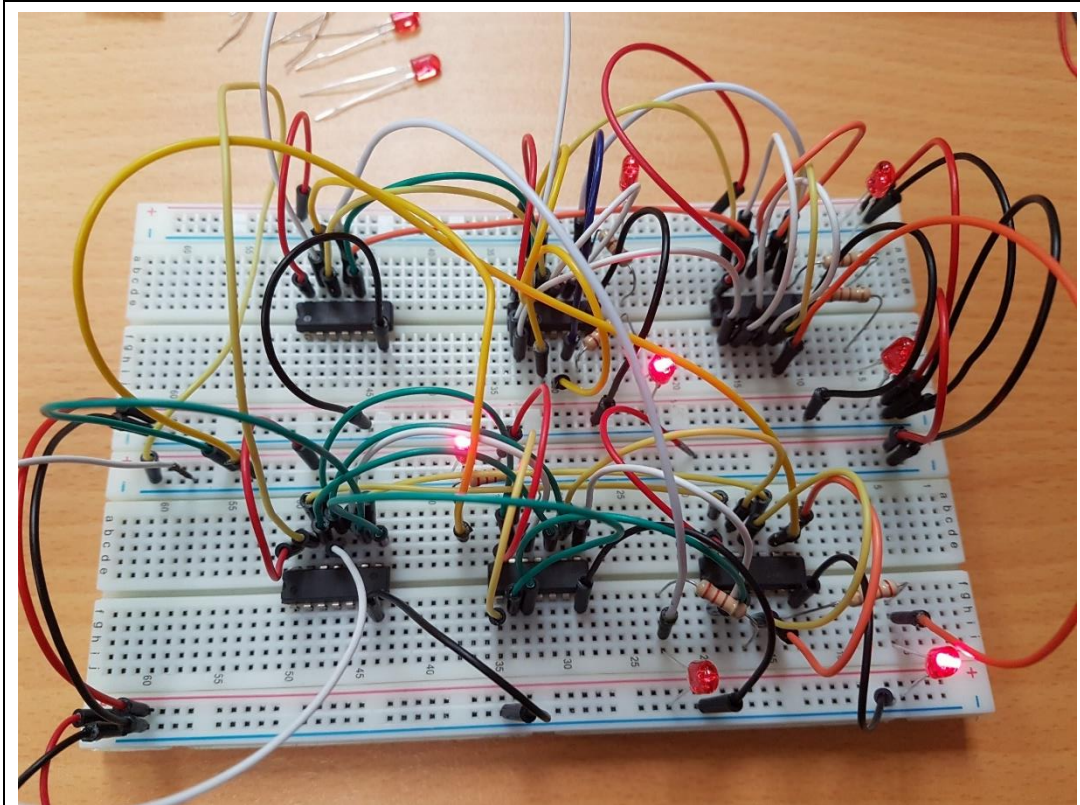


Input=1000 ABE=001, Output=1000(Out0 不明顯)

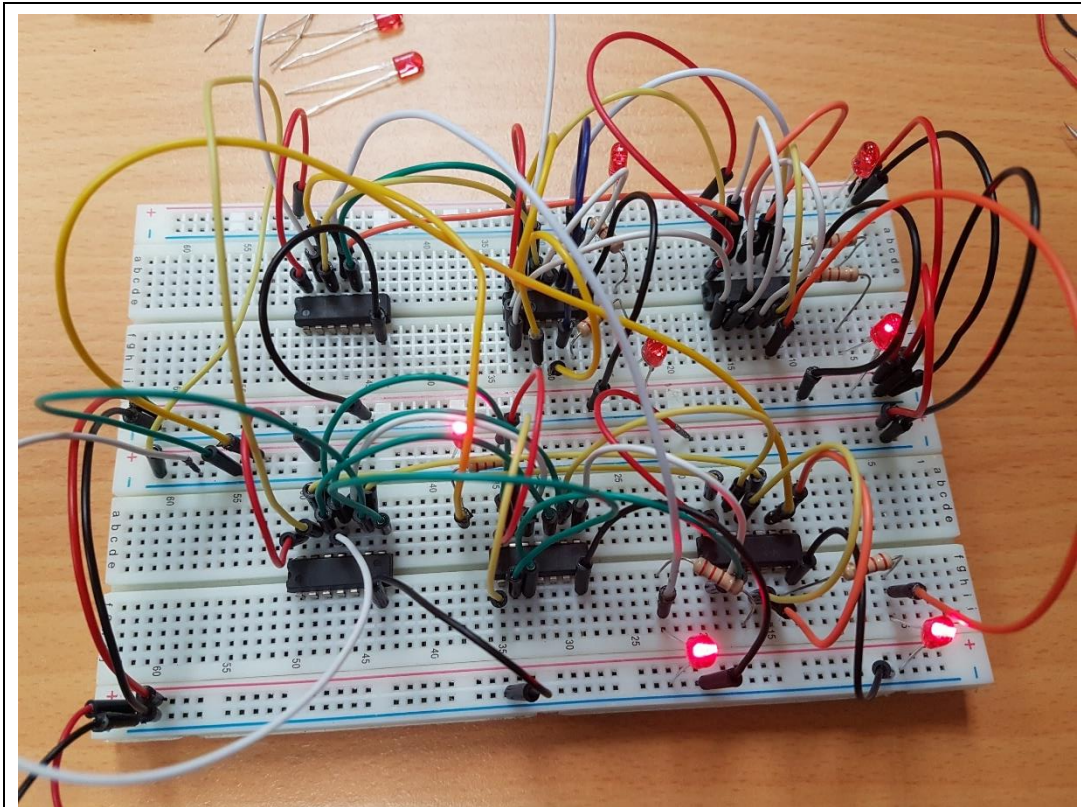




Input=0100 ABE=011, Output=0100(Out1 不明顯)



Input=0010 ABE=101, Output=0010



Input=0001 ABE=111, Output=0001

心得：

張軒：

我覺得多工器很酷，沒想到這麼小的電路就可以做出來，不過實際接起來還是有點複雜，比較容易弄錯線，所以在檢查的時候有點小狀況。不過之後就要上軟體了，感覺就會變很難的樣子，可能沒有機會接麵包板了，說不定到時候會開始懷念接線路的日子？

章子嚴：

這次的實驗讓我了解了多工器和解多工器是這麼在電話系統上運作，也了解保全系統是這麼操作，知道什麼是編碼器和解碼器和它的功能。

魏晉成：

這次的電路也是趁著實驗課前的時間將其接好，但卻用了比較醜而且比較難 Debug 的杜邦線，原因單純是因為在前一晚接線時，我完成線路後卻無法照實執行，因此就等隔天早上再重新接線，屆時我才發現我的 7404NOT 閘 IC 買成 CMOS 的，因為電氣特性不進相同才會無法成功運作；然而在重新接線時，時間稍嫌不足，因此就用杜邦線代替。

這三次的實驗我做的蠻愉快的，以前高中做過 Semi-Adder，因此對 IC 電路有了興趣，希望以後能嘗試用 IC 接出一顆 CPU(?)。

然而在選設正課第一次接觸 Verilog 之後，我對於這種邏輯閘描述語言也有些興趣，因此也很期待未來的實驗課程~