

B0928012 王晟翰

首先安裝老師給的兩個壓縮檔(裡面有 xp iso 後面會用到的程式)

虛擬機安裝 xp

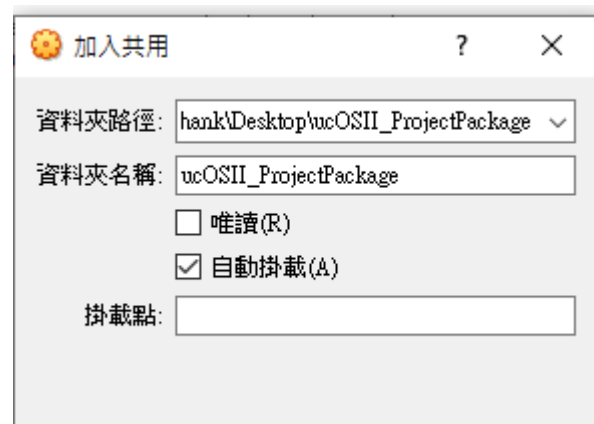
方法跟 ubuntu win11 大致雷同 不用特別改設定

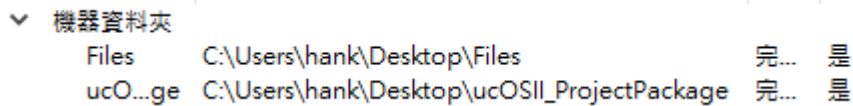


下載 Guest Additions 整合主客端之間的滑鼠游標，且可以分享資料夾等等
啟動 XP 左上角點裝置>插入 Guest Additions CD 映像...>打開資料夾對光碟機內的檔案 VBoxWindowsAdditions 連續按兩下滑鼠左鍵，開始安裝程式。
一步一步點下去，結束後重開機。



共用資料夾方式：



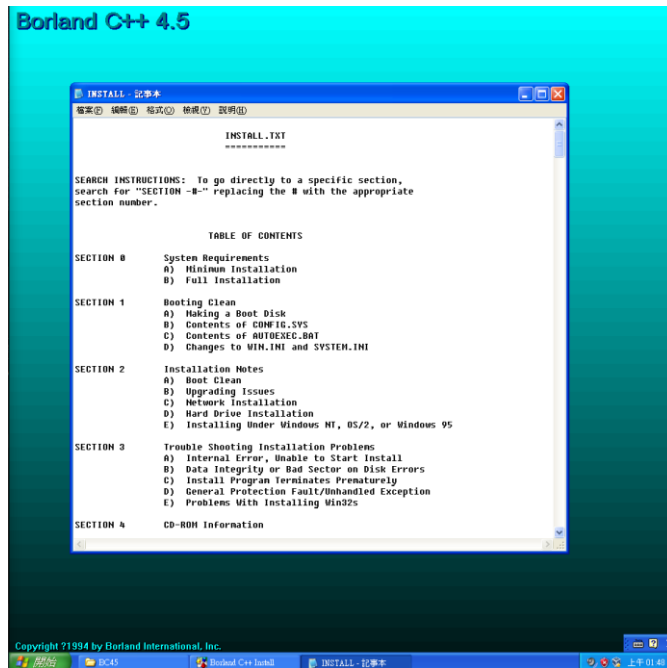


確定後虛擬機打開就會多一個 Z 槽裝剛剛的資料夾

Download Borland C++ :

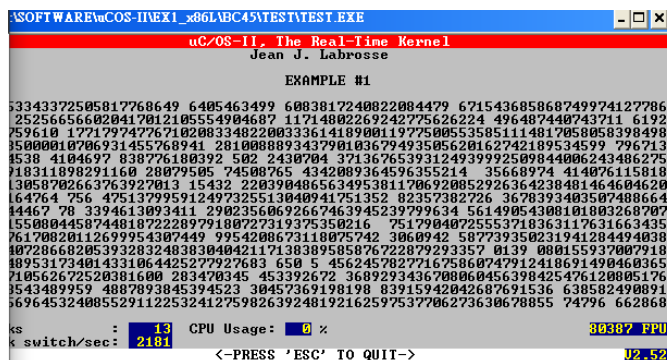
Double click the "INSTALL.EXE" 一直按繼續或者 OK

新增環境變數：控制台>系統>進階>環境變數>path 新增路徑 C:\BC45\BIN



Copy "\tasm\BIN\TASM.EXE" to your "C:\BC45\BIN"

put the source code package "SOFTWARE" directly in C:\
test 執行的樣子

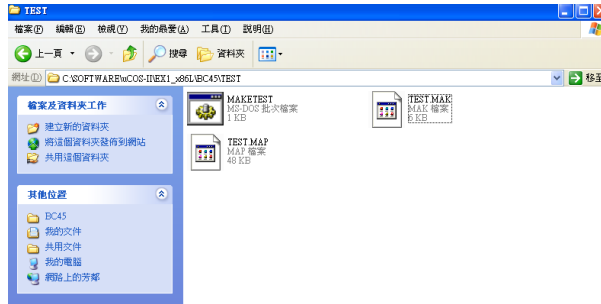


進入 C:\SOFTWARE\uCOS-II\EX1_x86\BC45\TEST

把 test 刪掉

Run MAKETEST.BAT

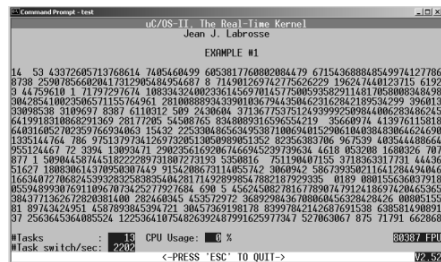
Test 又回來



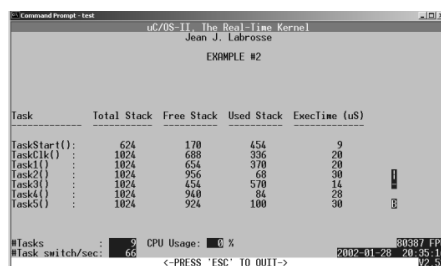
遇到問題：

1. 以為檔案是先開虛擬機再從虛擬機下載。解決方法是載在本機，再透過共用資料夾分享過來。
2. 一開始沒看到 C:\BC45\BIN 的路徑。經過檢查發現是因為 Borland C++ 沒有案後續的繼續，沒測底安裝好。

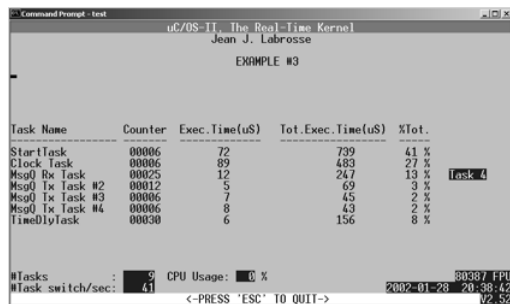
加分題：



Example 1 是演示了 $\mu\text{C}/\text{OS-II}$ 的基本多任務處理能力。十個任務在屏幕上的隨機位置顯示 0 到 9 之間的數字。除此之外，他主要有 13 個任務。其中兩個是空間任務和 CPU 使用率的任務(這兩個都是內部任務)，剩下的任務是隨機定位，並產生出 0 到 9 的數字，無限做下去。



Example2 演示了 $\mu\text{C}/\text{OS-II}$ 的堆棧檢查功能。此外，Example 2 顯示了 stack 檢查函數的執行時間，因為它取決於每個 stack 的大小。課本證明，大量使用的 stack 會有較少的處理時間。Example2 由九個任務組成。在這九個任務中， $\mu\text{C}/\text{OS-II}$ 一樣創建了兩個內部任務：空閒任務和確定 CPU 使用率的任務。另外七個找出每個任務實際使用了多少 stack 空間，以及確定每個任務 stack 的大小需要多少執行時間。

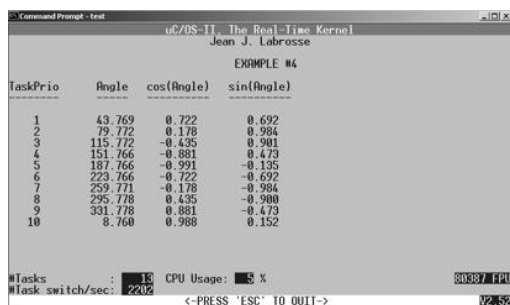


uc/OS-II, The Real-Time Kernel
Jean J. Labrosse
EXAMPLE #3

Task Name	Counter	Exec.Time(uS)	Tot.Exec.Time(uS)	%Tot.
StartTask	00006	72	739	41 %
Clock Task	00006	89	483	27 %
Msg0 Rx Task	00025	12	247	13 %
Msg0 Tx Task #2	00012	5	69	3 %
Msg0 Tx Task #3	00006	7	45	2 %
Msg0 Tx Task #4	00006	8	43	2 %
TimeDlyTask	00030	6	156	8 %

#Tasks : 9 CPU Usage: 0 % 80387 FPU
#Task switch/sec: 41 2002-01-28 20:38:42
<-PRESS 'ESC' TO QUIT-> 02:52

其他任務未就緒時執行的空閒任務，計算 CPU 使用率的統計任務，taskstart，剩下 6 個其他任務。後面任務處理運行次數，執行時間，累積時間，和其他 task 比起來的累積時間百分率。



uc/OS-II, The Real-Time Kernel
Jean J. Labrosse
EXAMPLE #4

TaskPrio	Angle	cos(Angle)	sin(Angle)
1	43.769	0.722	0.692
2	79.772	0.178	0.984
3	115.772	-0.435	0.901
4	151.766	-0.881	0.473
5	187.766	-0.991	-0.135
6	223.766	-0.722	-0.692
7	259.771	-0.178	-0.984
8	295.776	0.435	-0.900
9	331.778	0.881	-0.473
10	8.760	0.988	0.152

#Tasks : 13 CPU Usage: 5 % 80387 FPU
#Task switch/sec: 2202 <-PRESS 'ESC' TO QUIT-> 02:56

其他任務未就緒時執行的空閒任務，計算 CPU 使用率的統計任務，taskstart，剩下 10 個其他任務。去計算每一個角度的 sin/cos 角度。

心得：其實基本題做完後並沒有覺得很難，主要是有些小細節會沒注意到或者是沒做完，整體跟著老師講義做都不會有錯。但是只做基本題，其實真的不太知道在幹嘛，都是一步一步跟著做，但透過做加分題，才比較清楚這些程式是在幹嘛，但又發現其實沒一個東西背後有更多細節要看，其實非常難，我也沒有都看懂。