Syscall.h -> Start.s(C->mips) 軟體結束，C function裡的參數放進Register4.5.6…….

進到硬體

Exception.cc

1. 讀Register4.5.6…
2. 再呼叫ksyscall.h的東西

Ksyscall.h

1. SysOpen
2. 確認是否有該檔名的檔案
3. 判斷是否重複開檔(寫在fileSystem的Open function裡，位於filesys.h)
4. 在fileDescriptorTable尋找空位=>entry

(判斷是否已經達到開檔案的個數上限:20)

1. 回傳entry = fileID
2. SysClose
3. 依照丟入的id，呼叫fileSystem的Close function
4. SysWrite
5. Buffer = 想要寫入的內容
6. Size = 寫入的字數

(ex. Buffer=100 Size = 5 => 只寫入前5個字)

1. Id = 目標檔案
2. SysRead
   1. Buffer = 讀入的內容
   2. Size = 要從file讀出多少字
   3. Id = 目標檔案

指令順序

* 1. Make clean
  2. Make fileIO\_test1 ileIO\_test2
  3. ../build.linux/nachos –e fileIO\_test1
  4. ../build.linux/nachos –e fileIO\_test2

執行fileIO\_test1.c->fileIO\_test2.c