# 📍 Part 2 — 實作設計

Q9. 假設你設計一個新的 I/O 裝置（例如：紅外線感測器），請回答下列問題：

1. 你會選擇 Memory-mapped 或 Port-mapped I/O？請說明選擇原

因。

**假設使用樹梅派的情況，Rpi沒有獨立的I/O指令，因此只能夠使用Memory-mapped。就算是使用其他平台，也會選擇memory-mapped，因為少了I/O指令的切換，可以避免CPU持續中斷，以達到較好的效率 。**

1. 此裝置為 Character-mode 還是 Block-mode？為何？

**會使用Character-mode，因為紅外線感測是比較即時的資料。並且block-mode比較適用有大量資料的傳輸需求的情況，像是網路卡、硬碟等等**

1. 若要整合至系統的 I/O Subsystem 中，你需要完成哪些步驟來支援標準 API？

**應用程式藉由C函式庫的I/O函式，來觸發呼叫systam call讓CPU從user mode轉換成kernel mode，kernel會管理所有的I/O裝置，當system call進來時，會據其裝置檔名來找到相對應的driver，driver會向kernel註冊function pointer，這些pointer處理專門對於此裝置的opan(), read(), write()等操作， 這樣I/O devices就可以透過標準API來達到各個drivers想達成的事。**

Q10. 請 trace 你的 Linux TTY driver 的 source code，並說明他的 I/O 設計特點（請提供相關 source code 為佐證），以及如何透過 I/O Subsystem存取此 driver。

一張含有 文字, 螢幕擷取畫面, 字型 的圖片

AI 產生的內容可能不正確。

**跟tty driver相關的source code存放於drivers/tty內。**

**I/O設計特點:**

1. **tty主要分成三層：**
   1. **Core(提供統一API)**

**Code位於tty\_io.c**

****

**這是TTY Core對外提供的入口**

* 1. **Line Discipline(管理字元處理邏輯)**

**Code位於n\_tty.c**

****

**其中的**

****

**會將字元送到driver**

****

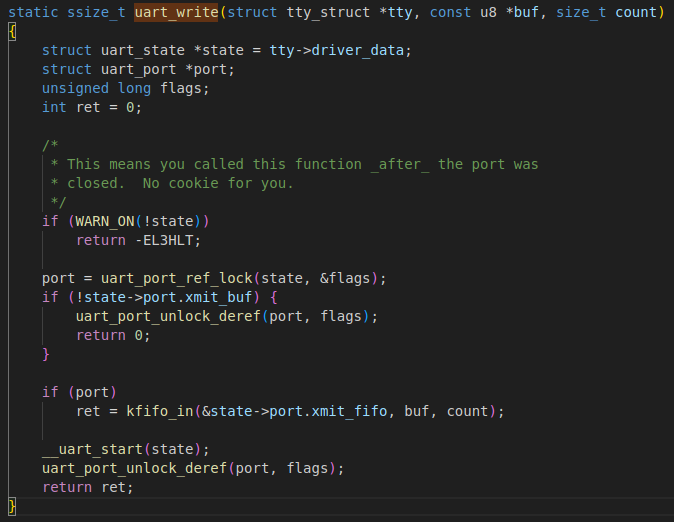
**其中的**

****

**會從buffer裡面將資料copy出來**

* 1. **Serial Core & UART driver(可以直接操作硬體)**

**Source code存放在serial\_core.c**

****

**每個 TTY 裝置都會綁一個uart\_state，裡面放了這個 UART port 的狀態和 FIFO 緩衝區。**

**ret = kfifo\_in(&state->port.xmit\_fifo, buf, count);**

**將資料放進FIFO**

**\_\_uart\_start(state);**

**開始傳輸**

**簡單來說，將資料放進FIFO，再讓driver去送。**

**pl011.c 裡的 pl011\_start\_tx() / pl011\_tx\_chars()才會真正去寫硬體。**

1. **Buffer 機制，source code存放於tty\_buffer.c** **一張含有 文字, 螢幕擷取畫面, 字型, 軟體 的圖片

   AI 產生的內容可能不正確。**

**該函式將insert和push結合，負責將字元放入buffer，以及將buffer的字元送到line discipline**

**Push不會立刻執行，會透過queue\_work先等待**

**一張含有 文字, 螢幕擷取畫面, 字型 的圖片

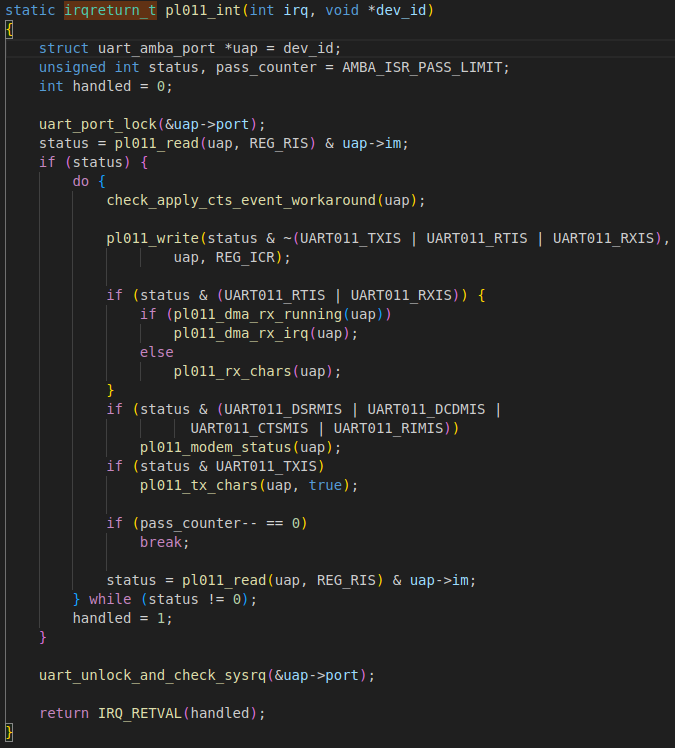
AI 產生的內容可能不正確。**

**當queue\_work被呼叫時，就會執行flush\_to\_ldisc，將字元送入discipline**

1. **Interrupt-driven I/O**

**如果不採用中斷，則驅動會持續poll硬體暫存器，相當浪費CPU資源**

**當硬體 UART 收到新字元時，才會發送中斷信號**

****

**上圖是中斷動作的code**

**pl011\_rx\_chars(uap); 接收中斷：讀取資料**

**pl011\_tx\_chars(uap, true); 傳送中斷：送出資料**

**心得:**這堂課讓我更了解了I/O devices在kernel中是如何運行的，希望老師能在上課時多多講課，覺得只有影片的內容還不夠，然後WS的份量可以像這次一樣，一部分在課堂上寫，一部分回家寫，這樣我覺得可以比較不那麼趕得去寫WS，可以好好吸收上面的題目，覺得這樣的學習模式很舒服