

開放電腦計畫

— 從一顆最簡單的 MCU 談起

金門大學 資訊工程系 陳鍾誠
2014 年 8 月 2 日於成功大學

話說

- 我在金門大學資訊工程系教書

教甚麼呢？

很多、像是

- 視窗程式
- 系統程式
- 網路程式
- 動畫設計
- 計算機結構
- 機率統計
- 人工智慧

疑

- 好像哪裡怪怪的？

哪裡怪？

感覺

- 領域好像太分散了一點！

對阿、沒錯！

- 因為我們是小學校，老師比較少！

所以

- 一個老師教的課程範圍就比較廣

但是

- 通常也沒有這麼廣啦！

一個老師

- 同時教「硬體、韌體、軟體、網路、多媒體」

說真的

- 是有點不太正常啦！

會這樣教

- 其實是我的一個心願

那就是

- 想要了解整台電腦
- 從最底層到最上層
- 是怎麼設計出來的

對我來說

- 這有點困難

特別是

- 有關硬體的這部分

話說

- 我大學念的是交大資訊科學系

雖然有硬體方面的課

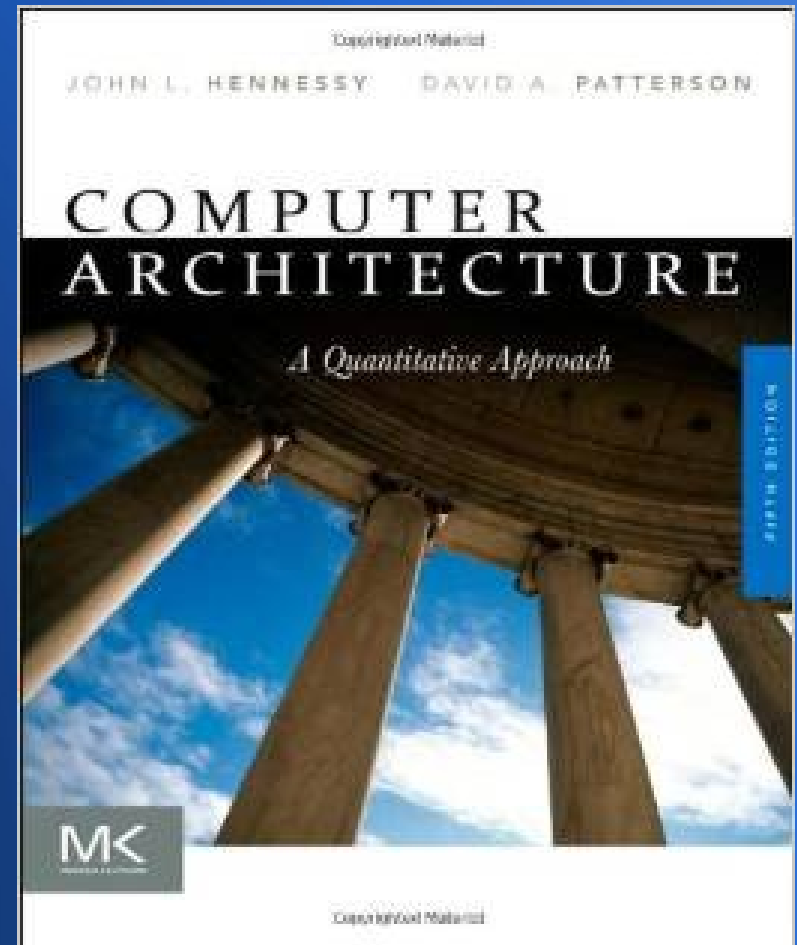
- 但我基本上都是低空飛過
- 或者是似懂非懂

舉例而言

- 電子學我覺得自己從來沒懂過
- 特別跑去電子系修的電子實驗也只有 60 分

而對計算機結構

- 我在大學時根本就沒選
- 研究所與博士用的好像都是右邊這本
- 我只記得，一直在計算速度、加快取、....
- 而且、一開始就教流水線 pipeline



但是

- 我連一顆最簡單的 MCU 都沒設計過啊
- 又怎麼會對這些「快取、流水線、MMU、平行」機制與公式有任何感覺呢？

這種感覺

- 就好像我們在還不瞭解「萊特兄弟」的那種飛機構造之前
- 就被迫要研究設計「超音速渦輪噴射機」時會遇到甚麼問題一樣。

於是

- 我們也只能夠「鴨子聽雷」，然後在考試的時候假裝自己已經懂了。
- 而且、假裝到最後，我們都覺得自己真的懂了！

然後

- 在很久很久以後的某一天

我變成老師了

更幸運的是

- 我還教了「計算機結構」

這時候

- 我也就把那套「自以為已經懂了」的課本
- 拿出來教給學生

不過

- 我或許騙得了學生
- 但是卻騙不了自己

對於

- 一個從來都沒有設計過任何處理器的老師
- 在上計算機結構課程的時候
 - 也就只能講講理論罷了

所以我決定要學習

- 如何設計一顆處理器
- 還有如何設計整台電腦

但是

- 這真的很困難

因為要設計一顆處理器

- 我得先問問張忠謀
- 看他願不願意幫我代工

而且

- 代工費用是零
- 連材料與工本費都要幫我出

當然

- 這件事情不需要問
- 我們都知道結果會怎麼樣！

所以

- 我沒有去問張忠謀！

還好

- 那時候系上有人申請國科會計畫
- 買了一堆 Altera DE2-70 的 FPGA 板子

雖然

- 我沒有申請國科會計畫
- 但是、我知道一件事情

那就是

- 國科會計畫買的東西
- 通常買來之後沒有多少人會去用

所以

- 既然申請人沒在用
- 那一切都好商量

於是

- 我就借了一塊 Altera DE2-70 來玩
- 想要用這個玩意來設計 CPU

可惜的是

- 那個學期我失敗了！

我可以設計出

- 算術邏輯單元 (ALU)
- 暫存器單元
- 記憶體單元

但是

- 就是設計不出「控制單元」

於是

- 那學期我又只能講講理論
- 然後就結束了！

對不起

- 我又誤人子弟了！

於是後來

- 我上網找了好久
- 終於發現華盛頓大學有個叫 Richard 的老師
 - 他設計了一顆簡單的 CPU
 - 姑且讓我們稱之為理察一號

而且我發現

- 他寫的 VHDL 程式我看得懂ㄟ！

所以後來我就仿照他的做法

- 把 CPU0 簡化成只有六個指令，然後放上了 FPGA 。

這應該算是

- 我所設計的第一顆 CPU
- 雖然、只有六個指令

去年

- 我又教了「計算機結構」這門課

於是

- 我在教完「數位電路、加法器、ALU、暫存器、記憶體」之後
- 就把 CPU0 拿出來教

結果

- 沒想到很多同學還是聽不懂

所以

- 後來我決定把程式再度簡化
- 於是就弄出了更簡單的 MCU0

然後、在期末的時候

- 我要求每位同學都要設計一顆自己的 CPU

當然、難度不拘

- 但是、必須是全世界獨一無二的

結果

- 當然還是有人聽不懂
- 也有人聽懂了卻寫不出來
- 也有人寫出來了卻是「修改」別人的
- 當然也有人真的設計出簡易的 CPU 了。

今年

- 我又要教計算機結構了！

但是我知道

- 在這條無止境的道路上
- 我還有很多東西要學

舉例而言

- 我們所設計的 CPU ，其實最多只能說是 MCU 等級的而已
- 而且、沒有快取、沒有流水線 (pipeline)、沒有多核心、也沒有 MMU

所以

- 這也是我今天為何來這裡的原因

因為我知道

- 在場的所有人
- 都是我的老師

我來這裡偷學

- 然後再回去教我們金大的學生

因為、我知道

- 最好的學習方式、就是：

現學現賣

- 邊學邊賣

如果

- 您有甚麼東西學不會

那麼

- 請您先找一個學生
- 然後對他說
 - 下禮拜上課，我教你！

只要您這樣做

- 相信在下個禮拜之前

您應該就可以

- 學會那個東西了！

如果

- 我們希望學生學會某個東西

我認為

- 需要做的事情

不是幫他們找老師

而是

- 幫他們找幾個學生

然後、讓他主動告訴學生們

下禮拜、我教你！

參考文獻

- 開放電腦計畫 - 計算機硬體結構 (使用 Verilog 實作)
 - <http://ccckmit.github.io/co/htm/book.html>
- MCU0 的迷你版 – mcu0m
 - <http://ccckmit.github.io/co/htm/mcu0m.html>
- MCU0 的區塊式設計 – MCU0bm.v
 - <http://ccckmit.github.io/co/htm/mcu0bm.html>

參考文獻（續）

- MCU0 完整版
 - <http://ccckmit.github.io/co/htm/mcu0s.html>
- 將 MCU0 放上 FPGA 執行
 - <https://dl.dropboxusercontent.com/u/101584453/pmag/201408/htm/article0.html>