

八十九年度【可攜式電腦週邊之控制系統設計及在多媒體應用之研究】

執行期間：88年8月1日 --- 89年7月31日

主持人：謝景棠

執行機構及單位：私立淡江大學電機工程學系

執行計畫編號：89-2213-E-032-023-

摘要：

SSFDC (智慧型媒體磁碟機) 規格標準是依據 PCMCIA 協會所提出的 PCMCIA-ATA 及 ATA-TrueIDE 介面標準而設計出來的 PC 卡, 是一種高容量的靜態儲存裝置。SSFDC 和其他的 PC 卡不同之處, 是它把儲存的主體 Flash Memory 和控制器分開。我們計畫發展的 SSFDC 控制器是針對 Toshiba 和 Samsung 的 Flash Memory (SmartMedia Card) 而開發的控制器。基本上其分成兩部份: 硬體 (Hardware) 和韌體 (Firmware) 兩部份。針對硬體的部份, 主要是由一顆單晶片的微處理控制器, ATA 介面電路, 和 ECC 控制電路所構成。針對韌體的部份, 主要為記憶管理程式。和其他的 PC 卡不相同。PC 要讀寫 SmartMedia Card 內部的資料, 中間必要透過 SSFDC 控制器來運作, 而這顆控制器可以說是整個 SSFDC 規格標準中最重要的部份。

計畫緣由與目的：

智慧型媒體磁碟機, 也稱智慧型媒體卡, 不僅可以當作電腦的另一附加硬碟, 但不含有讀寫頭讀寫與碟片旋轉的機械驅動動作, 故其具有省電、體積小與抗震的優點。此特點特別適合於可攜式的電腦設備使用, 諸如: 筆記型電腦、次筆記型電腦、掌上型電腦、個人數位助理 (Personal

Digital Assistant, PDA)、數位式電子照相機 (Digital Still Camera, DSC) 等。

本計畫的提出目標在於結合學術單位的人力資源, 研究經驗。針對 SSFDC 系統作關鍵性技術之分析研究。期望能藉由此計畫的執行, 能對所缺乏之關鍵技術有進一步的掌握, 並期望藉著研發過程對於 SSFDC 系統設計人才的培養及研發成果的技術移轉能對我國消費性電子工業, 電腦週邊及多媒體系統產業在關鍵系統上研發之需求有所助益。

結果與討論：

微電腦控制器與外界溝通共有兩處, 其中一個是以 PCMCIA-ATA 或 ATA-True IDE 介面來跟主機做溝, 另外一個則是與 Flash memory 做溝通, 所以在設計上, 這兩部份的邏輯電路必須符合標準規格, 而其他的邏輯功則視本身的設計所需而自行設計

在 SSFDC Controller 的邏輯電路設計方面, 初期是採用 ALTERA 公司的 Max+Plus II 硬體設計軟體; 在這套軟體上, 是以 VHDL 語法去做硬體的設計。在這之後, 再用 ALTERA 編號 EPF10K50GC403-3 這顆 FPGA 晶片在實驗測試板上做實際電路上的邏輯功能驗證, 這時要與韌體 (Firmware, 是用 C51 的語法來完成這韌體, 再來是透過 DEEMAX 公司的 P-ICE 8051 來做實

際電路上的驗證)配合在一起來做驗證的工作,在驗證的當中如果有發現錯誤,再回到 Max+Plus II 上做修改的工作,修改過後再回來實驗板上,如此重復的不斷驗證修改,一直到得到真確的結果。

初期的工作完成後,則進入實際 IC 電路的時序模擬驗證,這時所用的工具是 View Logic,驗證以高階硬體描述語言所設計的出來硬體,在轉成實際的 IC 電路後,其時序是否符合規格所要求,如有不符合,則再回到 Max+Plus II 上做修改的工作,直到符合要為止。

控制器的設計方式是採用 Top Down Modeling 的方式,也就是先把整個設計分成若干個模組方塊,對於每個區塊都給於所要的定義,然後驗證它的功能及時序的正確性,最後把全部的區塊整合在一起成為一個大的系統,再對這個大系統功能(Function operation)和時序(Timing operation)來做驗證。

SSFDC 是一個公開的規格,為了我們所展出來的控制器要能與其他產品相容,因此這個控制器的功能也一樣是有一定的標準,但除了符合標準規格所定下來的功能外,我們還可以配合我們的設計要求多加一些其他的功能下去的。以下就是我們控制器功能的要求:

1. PCMCIA-ATA 和 ATA-True IDE Interface,SSFDC 內部這顆控制器,主機必須透過這顆控制器來對內部的 SmartMedia(Flash Memory)來做讀、寫等動作。控制器的是以 PCMCIA-ATA 或 ATA-True IDE 介面標準和主機做溝通,所以這顆控制必須要能夠正確的接受主機傳遞過來的控制訊號,並且能產生符合 PCMCIA-ATA 以及 ATA-True IDE 介面標準的控制訊

號。

2. 8032uP Interface,負責接受 8032uP 傳遞過來的訊號和送出訊號給 8032uP。在控制器的內部裡須要一個處理器(在這裡所用的微處理器是 8032uP),用來對主機傳送過來的訊號(指令)做處理以及送出相對應的訊號(指令)給主機,這個微處理器就好像是控制器的心臟一樣,控制控制器內的一切的動作;對於這個微處理器(8032uP),另外多加 XRAM (2k+256 bytes)給 8032uP 用,而 8032uP 所用的 ROM 是 64Kbytes。
3. Sector Buffer Manager,這裡是用來管理控制內部的 Buffer。實際上,當要從主機寫一筆資料到 SmartMedia 時,由於 SmartMedia 的特性問題(寫入資料後,如果再要對它做寫入的動作,則必須先要把資料抹除後,才能再寫入),也為了設計者有更大設計空間(發展出自己一套完整的 algorithm、控制器對內部資料的 mapping 等),一般上,控制器內部必須要設定一個 Buffer,主機先把資料先寫入到預先設定的 Buffer (容量為 1Kword + 256bytes)裡,然後由控制器內部的處理器(8032uP)配合邏輯電路再寫入到 SmartMedia 中,相同的要從 SmartMedia 中讀出一筆資料,也是先把資料先寫到 Buffer 中,把後再由主機去讀取。
4. ECC(Error Correct Code) Circuit,用來檢查資料是否錯誤且給予更正。在把資料寫到 SmartMedia 時,為了要避免資料寫到 SmartMedia 的過程中,資料在資料匯流排上會有遺失(在控制器內部預設的 Buffer 和 SmartMedia 之間的資料傳輸)導

致寫入的資料是錯的，或是 SmartMedia 本身就有壞掉的部份，造成資料寫不進去 SmartMedia，而讀出來的資料是錯誤的；所以在資料寫入 SmartMedia 時先做編碼產生錯誤更正碼 (Error Correction Code)，以便將資料從 SmartMedia 讀出時用來檢查是否有錯誤並且給予更正；然後由韌體來做這個工作也是可以，但是太費時且暫 ROM 的空間，所以這個產生錯誤更正碼的工作交由硬體來做是最適當。

5. Flash Sequence Control Logic，用來產生對 SmartMedia 做讀、寫等動作所需要的控制訊號和自動時序。由 SmartMedia 規格裡，可以知道 SmartMedia 只有 8 個 bit I/O BUS 和一些其他重要的控制訊號線，data、address 或是指令，都是透過這 8 個 bit 的 I/O BUS 來完成傳送，所以它有它特定時序的控制訊號。這部份的邏輯電路對整個控制器來說，它不但分擔了 8032uP 的工作負擔，也加速了控制器對 SmartMedia 做讀、寫等動作。

由依照上面所說的，可以把整個控制器劃分成以下 7 個模組來設計：

1. RST_PROC 模組：這部份是負責對控制器內部的 Register 做歸零的動作，也就是對控制器做初始化的動作。
2. FLH_TOP 模組：這部份是負責產生必要的控制訊號和寫入、讀出資料的時序，把資料寫到 SmartMedia 或是把資料從 SmartMedia 中讀出來，在這裡所產生的控制訊號的時序必須是要滿足 SmartMedia 的規格。另外也設計了一些邏輯電路，當下一個指令時就能自動產生一組控制訊號來對 SmartMedia 做處理。在這
3. HOST 模組：這部份的主要目的是用來與主機做溝通，所設計的邏輯電路必需要滿足 PCMCIA 協會所規定的介面規格，這樣主機才能透過這個 PCMCIA 介面規格來與這個控制器做溝通。這模組主要是負責接收主機傳來的控制訊號，經過內部邏輯電路做硬體解碼，再傳送到 8032uP 來做處理，產生相對應的動作。同時這模組也負責把 8032uP 所產生的控制訊號傳回給主機，告訴主機控制器現在內部的情況，讓主機去執行下一部所要執行的動作。當然在這 Host 模組的底下還是把它分成幾個小模組來設計，不同的模組都有其不同的功用。
4. BUF_TOP 模組：這部份只用來對 Buffer 的控制，當主機送來一筆資料時，先透過 HOST 模組來接收，然後透過這裡 (BUF_TOP 模組) 才寫到 Buffer 去，再由 FLH_TOP 模組寫把資料寫到 SmartMedia。如果主機要讀取 SmartMedia 的資料時，也必須由 FLH_TOP 模組從 SmartMedia 把資料搬出的同時也透過這裡 (BUF_TOP 模組) 把資料先寫到 Buffer 去，再由 HOST 模組那裡把資料傳送給主機。
5. ECC_TOP 模組：這部份就是負責產生錯誤更正碼，當把資料寫到 SmartMedia 時，這裡會做編碼的工作，產生一組錯誤更正碼，然後這組錯誤更正碼會被寫入到 SmartMedia 的 space region 去，當把資料讀回出來時，這時這裡負責做解碼的工作，去判斷資料是否有錯誤，如有錯誤還要去資料更正的動作，所以由這裡可以確保的資料儲存到 SmartMedia，再讀出時資料錯誤

的機率變的非常的小。

6. WIZARD 模組:其實前 5 個模組都可以獨立運作,而之間也有相關的訊號線,當前五個組都各別設計好之後,我們必把它們之間的相訊號連接在一起,有必要時也要做一些解碼的電路來連接之的各個模組。WIZARD 模組就是在做這份工作,這裡是接受外部進來的訊號,直接或是經由這裡解碼然後送到各模組去,而把各模組之間相關的訊號互相接通。總的來說,WIZARD 模組把前面 5 大模組整合成一個大系統,也就是功能健全的 SSFDC 控制器。
 7. WIZ 模組:這裡也是設計的最後工作了,把整個設計分成 7 大模組來個別設計,而 WIZ 模組則用來做最 I/O 腳的 mapping 工作,因為我們的設計上會出現錯誤,常常需要更改設計,更改過後 FPGA 就要重新繞線,但是 FPGA 的 I/O 腳是要定死那才有辦法在測試版上做測試,所以我們把 I/O 腳的 mapping 留到最後與整個設計分開出來,以後雖然設計有需要改變,但其對應到外部的 PIN 腳位址是不改的。把前面的 7 個模組都完成後,這顆控制器的邏輯電路就算是初部在電腦軟體上設計完成了。
- 5、CompactFlash Specification, CFA, 1996
 - 6、PC Card Standard Specification, PCMCIA, 1995
 - 7、Inside PC Card-CardBus And PCMCIA Design, Faisal Imdad-Haque, 1996
 - 8、IEEE Standard Signaling Method for a Bidirectional Parallel Peripheral Interface For Personal Computer
 - 9、MultiMediaCard(MMC) Specification
 - 10、esign with Flash Memory Brian Dipert & Markus Levy
 - 11、The Programmable Logic Data Book
 - 12、VHDL Compiler Reference
 - 13、CIC 訓練課程 Altera(PC) 國科會晶片設計製作中心 著
 - 14、微處理系統的實用界面技術 王晉良 譯
 - 15、VHDL second edition Douglas L.Perry
 - 16、Device Drivers 設計巧 邱文達 譯

本計畫完成 PCMCIA - ATA 標準介面的設計 ATA - True - IDE 標準介面的設計、Sector Buffer Manager、Error Correction Circuit、Flash Sequence Control Logic 之設計

參考文獻:

- 1、PCMCIA 卡研究報告,經濟部技術處,1995
- 2、MIND SHARE,PCMCIA System Architecture,1995
- 3、王生耀、張凱鈞,PCMCIA 應用大全,1996
- 4、Perry ,VHDL second edition,1994