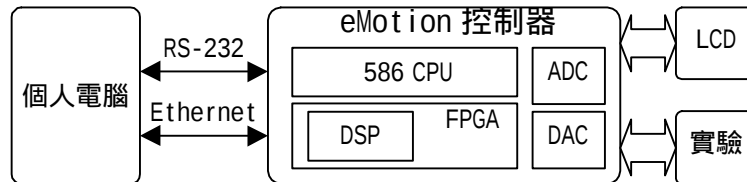


## 第一章

# 系統規劃

eMotion控制器(簡稱e控器)是一個可以獨立工作的PC級控制器。其CPU部分採用標準的PC架構，包括586CPU、RAM和單晶硬碟(Disk-On-Chip)，使用開放式Linux的作業系統標準。



e控器中具備了類比和數位界面。在類比界面部分，包含8頻道12位元的ADC以及4頻道12位元的DAC，可執行高達32KHz的抽樣頻率。在數位界面部分，採用可程式的大型邏輯閘(FPGA)，其內部電路可以依需要隨時做設定和更新，可彈性應用於各種不同的系統設計。

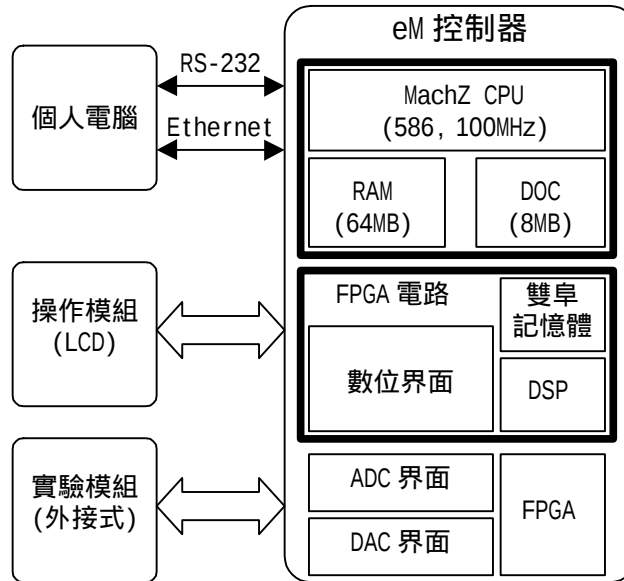
不但如此，在FPGA內部更規劃了一個16位元的DSP控制器，負責執行所有的即時控制軟體，控制程式的抽樣頻率可高達32KHz，且不受任何作業系統或通訊環境所干擾。DSP控制器和586CPU間採用雙埠記憶體直接連線，所有的DSP中的資料和程式，都可由586CPU讀寫、設定和管理。

e控器可透過網路界面和個人電腦連線，在TCP/IP的通訊協定下執行遠端監控與管理工作。應用程式可以透過C程式、Java程式、Matlab和Simulink做不同的系統應用和操作連線。在實驗搭配方面，可透過類比和數位界面連接不同的實驗模組，同時提供LCD面板和紅外線遙控器，做多頻道示波器的現場測試工作。

本章中針對e控器的硬體設計、模組配置、軟體規劃和應用規劃等各部分，做詳細的說明。

## 硬體設計

e控制器中包括了586CPU和64MB的記憶體，執行開放式的Linux作業系統，整個作業系統和應用程式都壓縮在一顆8MB的單晶硬碟(DOC)上。由於Linux系統的通用性，大幅降低了研發成本、時限和風險。

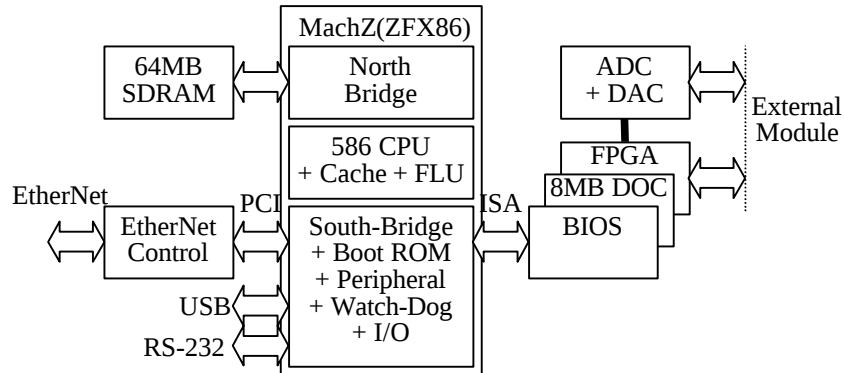


由於Linux作業環境並不是真正的即時作業系統，無法提供高速的即時處理軟體。所以我們在FPGA自行規劃了一個內藏式DSP控制器，專門負責即時控制軟體的執行，並以雙埠記憶體(Dual-Port Memory)直接和586CPU溝通。DSP即時控制的抽樣頻率可高達32KHz，且不受任何作業系統干擾。

在界面部分，e控制器提供了相關的類比和數位界面。類比界面包括了8頻道的類比輸入(ADC)和4頻道的類比輸出(DAC)，數位界面則由大型可程式邏輯閘(FPGA)所執行，提供高密度和彈性規劃的設計。所有界面都由I/O控制器統一處理，並利用分時控制將資料傳到雙埠記憶體中，不論是DSP或是586CPU都能很方便的運用。

## 內藏式PC控制器

我們選用ZF-Linux公司出品的MachZ做內藏式PC控制器，方塊圖如下：



設計重點包括：

1. MachZ本身為100MHz的586CPU，還包括了南橋、北橋、相關周邊和浮點運算器，耗電只有0.5W，不需額外散熱片。
2. MachZ提供現成的BIOS碼和其參數設定工具，可彈性設計PC主機板而不需要重新設計BIOS。
3. 外加元件極少，只需RAM、BIOS、DOC(Disk-On-Chip)和EtherNet，我們再加上FPGA和AD/DA，即可構成完整的PC控制器。
4. 研發過程中可利用RS-232、USB及Ethernet界面連接其他周邊和研發工具，在產品真正應用時，只要留下Ethernet界面即可進行連線控制。
5. 在模組中併入FPGA和AD/DA，使得易受雜訊干擾的匯流排完全內藏，可以大幅提高系統的抗雜訊能力。
6. 全部電路板尺寸約10cm\*20cm，採用子母板的堆疊(Piggyback)方式對外連接。
7. 程式和資料儲存在可讀寫的單晶硬碟(Disk-On-Chip，簡稱DOC)中，一顆ROM大小的晶片即可支援8MB到256MB的硬碟空間。

## FPGA電路

主要的FPGA我們選用Xilinx公司Virtex-II系列的XC2V250，其中的重要規格包括：

1. 閘數(Gate Count)為250K，內部電源為1.5V，可允許420MHz的脈波頻率。
2. 邏輯閘的單位為CLB，每個CLB包含四組Slice，而每個Slice中都有一組D型正反器和4位元的可程式邏輯。XC2V250中有24\*16個CLB，相當於1536個Slice。
3. 每個Slice都可規劃成16-bit的記憶體，也可另行規劃成16-bit的移位暫存器。
4. 擁有獨立的雙埠RAM，容量為18Kbit或是2Kbyte，可自行規劃成不同的匯流排寬度，包括1/2/4/8/16/32多種。XC2V250中共有24個雙埠RAM，容量共48KB。
5. 擁有硬體乘法器，為18\*18=36的整數乘法器，同時具有正負號的處理功能。  
XC2V250中共有24個硬體乘法器。
6. XC2V250中最多可同時擁有16組的公用Clock，由於具備硬體連線，delay差異極小，可忽略不計。
7. 就繞線能力而言，上下左右的CLB之間可允許  
24 long lines,  
120 hex lines,  
40 double lines, 和  
16 direct connect lines。
8. FPGA電路以RAM型式儲存，所以開機後必須執行下載工作。
9. XC2V250最多可擁有200個I/O點，每個I/O點都可規劃成不同型式的邏輯準位，我們採用3.3V的LVTTTL邏輯準位。

eM卡中還有另一顆較小的FPGA，是Xilinx公司Spartan系列的XCS05XL，重要的規格包括：

1. 閘數(Gate Count)為5K，內部電源為3.3V，可允許80MHz的脈波頻率。
2. 邏輯閘的單位為CLB，每個CLB包含兩組D型暫存器和4位元的可程式邏輯。  
XCS05中有10\*10個CLB，相當於200組暫存器。
3. 每個CLB都可規劃成16-bit的記憶體。
4. 最多可同時擁有4組的公用Clock，由於具備硬體連線，delay可忽略不計。
5. 最多可擁有72個I/O點，每個I/O點都可規劃成3.3V的LVTTTL邏輯準位。

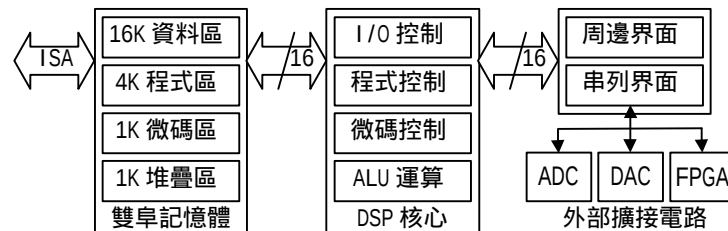
XC2V250中規劃了DSP控制器和其相關周邊，設計完整並測試完畢，實驗者可以直接拿來使用。而XCS05XL則完全開放，沒有任何預設電路。實驗者可以根據各個實驗的不同需求，自行設計使用。由於XCS05XL的閘數較小，可以大幅降低實驗者的設計門檻。

## 內藏式DSP控制器

在FPGA電路中，我們建立了一個16位元的DSP控制器，以執行高速的即時控制程式。規劃如下：

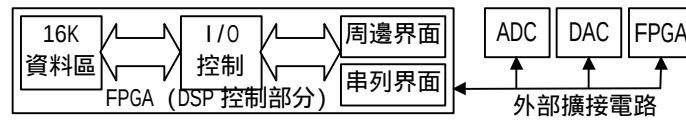
1. 為了避免智慧財產權上的爭議，這個DSP是在不參考任何現成設計的條件下，完全獨立設計的。
2. 有DSP就必須搭配組合語言。為了避免學習上的障礙，組合語言的規劃是以簡單、易讀為目標。一般的精簡指令集(RISC)是以最少的組合語言個數為目標，撰寫的程式自然較長。我們卻是以程式最短的組合語言為目標，所以並不採用精簡指令集。
3. 由於DSP必須搭配586CPU工作，所以DSP本身不必包括完整指令集，只要能夠應付即時控制的工作即可。其他非即時控制的軟體工作可由586CPU處理。
4. DSP和586CPU之間是以雙埠記憶體連接，其中包括了所有的程式和資料。
5. DSP的設計是以快速的執行能力為主，目標是能執行32KHz的即時控制程式。

就功能而言，DSP控制器的電路方塊圖如下：



1. DSP利用雙埠記憶體和586CPU連線。
2. 就586CPU而言，雙埠記憶體佔用了16位元ISA匯流排上的4KB位址。總共48KB的雙埠記憶體規劃成24頁，每頁佔用2KB。586CPU可以先選頁再讀寫其中2KB記憶體中的任意位址。
3. 就DSP而言，雙埠記憶體規劃成資料區、程式區、微碼區和堆疊區四個部分，除了微碼區為32-bit外，其他部分都是16-bit寬度。
4. 不論是DSP或586CPU都不准直接控制I/O界面，所有I/O界面都由I/O控制器透過分時(cycle-stealing)方式，在DSP的執行空檔中放置到對應的資料區中。對於DSP和586CPU而言，所謂的I/O只是某個位址的資料而已。
5. 外部擴接電路(包括ADC、DAC和另一顆FPGA)都是由I/O控制器透過串列界面連接。由於格式統一並且直接對應到雙埠記憶體中，程式設計者可以不用處理任何I/O界面，直接使用即可。

## DSP周邊界面



DSP的外部界面包括了類比和數位界面。在類比界面部分，包含8頻道的ADC以及4頻道的DAC，都是12位元的精度。在數位界面部分，都是由FPGA執行的可程式設計，可依需要隨時做設定和更新。

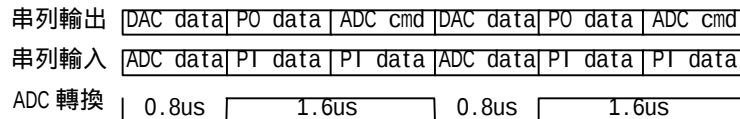
類比輸入界面採用Analog Device公司的AD7891，規格如下：

1. 12位元精度的ADC，轉換時間1.6us。
2. 內含8頻道多工器，可選擇8頻道的電壓數入訊號。多工器的穩定時間約0.6us。
3. 每個頻道是 $\pm 10V$ 的輸入範圍。
4. 單電源(5V)低耗電結構。
5. 採用串列傳輸界面，最高傳輸頻率為20MHz。訊框(frame)的單位為16-bit，所以每筆資料約需0.8us的時間傳遞。

類比輸出界面採用兩顆Analog Device公司的AD5322，規格如下：

1. 12位元精度的DAC，輸出穩定時間10us。
2. 內含2頻道的獨立DAC，所以由兩顆AD5322組成4頻道的DAC。
3. DAC輸出經由放大後，產生 $\pm 10V$ 的輸出範圍。
4. 單電源(5V)低耗電(300uA)結構。
5. 採用串列傳輸界面，最高傳輸頻率為30MHz。訊框(frame)的單位為16-bit。

為了搭配ADC和DAC，所以外接FPGA也採用同樣的串接界面，以20MHz的頻率傳遞。而最有效率的傳輸分配方式如下：



1. ADC、DAC和FPGA採用一樣的傳輸界面，以20MHz的頻率傳輸，每個訊框為16-bit，所以共需0.8us來傳遞一筆資料。
2. 三者分享同一界面，以串列輸出而言，必須由DAC資料、數位輸出以及ADC命令三者分享。以串列輸入而言，必須由ADC資料和數位輸入兩者分享。
3. 如果各項資料以上圖的順序做安排，則 $(2.4us \times 8 = 20us)$ 的週期內可以做完所有的掃描工作。換句話說，抽樣頻率可高達50KHz。

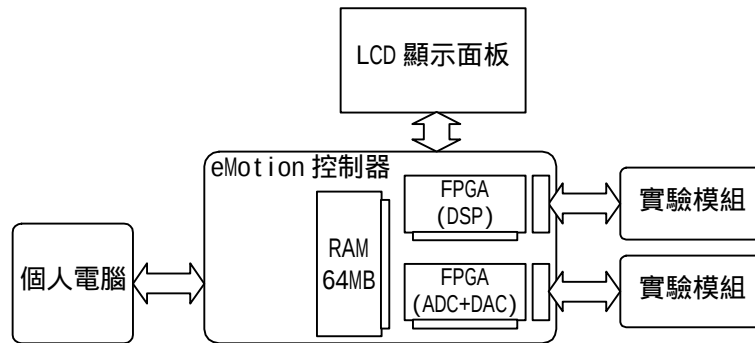
4. 而在這50KHz的抽樣頻率下，可以同時傳輸  
8頻道的12位元ADC資料，  
4頻道的12位元DAC資料，  
16頻道的12位元數位輸入資料，以及  
12頻道的12位元數位輸出資料。
5. 就ADC訊號而言，剛好包括了0.8us的多工器穩定時間和1.6us的ADC轉換時間，完全符合AD7891的規格需求。
6. 就DSP端而言，所有的串列控制訊號由I/O控制器處理，轉換資料會直接傳到雙阜記憶體中，DSP程式不需再做處理即可直接使用。
7. 就另一頭的FPGA設計而言，界面規格非常清楚，只要將容量限制在  
16頻道的12位元數位輸入資料，以及  
12頻道的12位元數位輸出資料即可。  
實際上，這規格已經遠超過XCS05XL的邏輯閘容量，應用上絕無問題。
8. 50KHz的系統抽樣頻率是工作上限而非唯一選擇。事實上，我們將選擇32KHz的系統抽樣頻率，以配合語音界面的規格需求(8KHz\*N，而N=4)。

由於XC2V250的邏輯容量相當大，除了可以包含DSP電路核心外，還有許多I/O界面的設計空間。由於是FPGA型式的可程式設計，實際上我們可以任意更換排列組合。當CNC控制器的應用時，我們選擇下面的組合以搭配四軸的AC伺服系統規格：

4頻道的三相PWM產生電路(共計12組獨立PWM產生器)，  
8頻道的光電盤解碼電路，為12位元精度，  
4頻道的雙向脈波產生器，為12位元精度，  
LCD掃描控制，以搭配LCD顯示器，  
紅外線遙控器解碼界面，以搭配紅外線遙控器的操作。

同樣的，所有的周邊界面都由I/O控制器將資料直接轉換到雙阜記憶體中，DSP程式不需再做處理即可直接使用。

## 模組配置

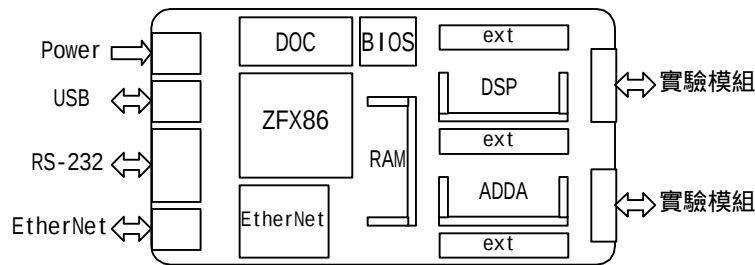


整個實驗模組的系統配置圖如上，包括：

- eMotion控制器： 包括內藏式CPU的整個電路(e控制器)。
- RAM卡： e控制器中以SO-DIMM包裝的RAM卡(64MB)。
- DSP卡： e控制器中以SO-DIMM包裝的FPGA卡，內含DSP控制器。
- ADDA卡： e控制器中以SO-DIMM包裝的FPGA卡，內含ADC/DAC元件。
- LCD顯示面板： 240\*128精度的LCD操作面板，內含麥克風、喇叭及IR遙控器。
- 實驗模組： 搭配實驗的多種模組，以標準的26P接線連接eM卡。
- 個人電腦： 研發操作作用，以Ethernet和e控制器連線。

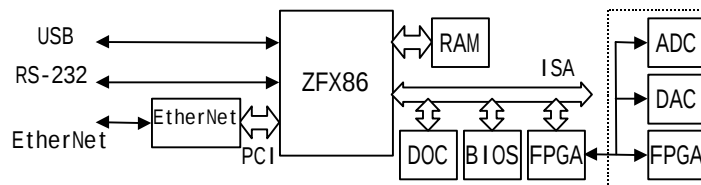
其中的三組SO-DIMM包裝的控制卡都是e控制器的必備元件，插卡方式的設計是為了便於裝配維修，以及將來的更改和擴張用。

## eMotion控制器



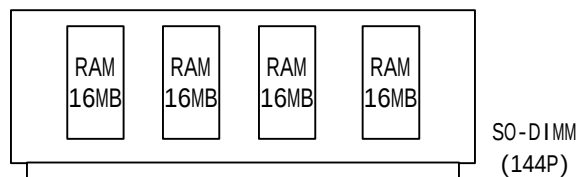
eMotion控制器的電路板配置圖如上，其中包括：

- ZFX86：586CPU，也是整個控制器的核心。
- EtherNet：網路界面IC，以PCI匯流排和586CPU連線。
- DOC：單晶硬碟(Disk-On-Chip)，8MB的容量。
- BIOS：PC的I/O控制器，為256KB的容量。
- RAM插槽：SO-DIMM插槽，準備插上64MB的RAM卡使用。
- DSP插槽：SO-DIMM插槽，準備插上內含DSP功能的FPGA卡使用。
- ADDA插槽：SO-DIMM插槽，準備插上包括ADC/DAC的FPGA卡使用。
- ext擴張座：3組50P的插座，可作為彈性擴張用，目前用以接LCD操作面板。
- 實驗接頭：2組26P的插座，可連接兩組的實驗模組。
- Power：外接直流電源，連接5V和+/-12V的三組直流電源的供應器。
- USB：586CPU的標準USB界面接頭，可外接兩組USB裝置。
- RS-232：586CPU的COM1串列接頭，可外接個人電腦。
- EtherNet：網路接頭，用以連接個人電腦或其他網路裝置。



整個電路的系統方塊圖如上。由於ZFX86包含了586CPU和所需的周邊控制，各組件都可直接連上匯流排使用，使得結構大為簡化。整個系統都安裝在一片10cm\*20cm大小的電路板上，其中除了586CPU和EtherNet外，大部分零組件都以插槽連接，使得系統安裝和維修都更為容易。

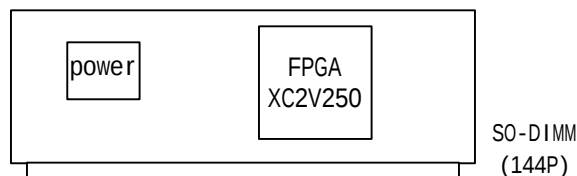
## RAM卡



RAM卡的電路板配置圖如上，其中包括：

1. 4顆16MB的RAM，共計64MB的記憶體容量。
2. 標準的SODIMM-144界面板尺寸(3cm\*7cm)，可平插在eM卡的插槽中。
3. 原則上可用市售SODIMM型式的RAM卡來取代，可是因為ZFX86的RAM-bus只有32-bit寬，並不符合SODIMM的64-bit標準，所以插上後只剩一半容量。例如64MB的RAM卡插上後只剩32MB，而128MB的RAM卡插上後只剩64MB。
4. 本RAM卡針對ZFX86的規格設計，所以可以完整使用所有64MB的RAM容量。

## DSP卡

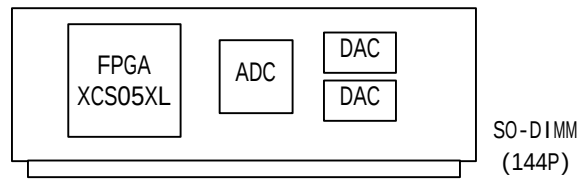


DSP卡的電路板配置圖如上，其中包括：

1. Xilinx出品的FPGA(XC2V250)，執行DSP核心和周邊界面控制。
2. XC2V250需要1.5V的內部電源，所以需要額外的1.5V電源供應器。
3. 標準的SODIMM-144界面板尺寸(3cm\*7cm)，可平插在eM卡的插槽中。

必須特別提醒：同樣的SODIMM尺寸的電路板，但是接腳定義卻和RAM卡完全不相容。所以一定要插在指定位置，不得交換使用。

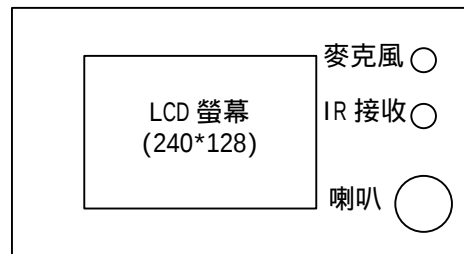
### ADDA卡



ADDA卡的電路板配置圖如上，其中包括：

1. Xilinx出品的FPGA(XCS05XL)，執行實驗所需的FPGA電路，完全開放使用。
2. 8頻道12位元的ADC(AD7891)，執行類比電壓輸入界面。
3. 兩組2頻道12位元的DAC(AD5322)，執行類比電壓輸出界面。
4. 標準的SODIMM-144界面板尺寸(3cm\*7cm)，可平插在eM卡的插槽中。

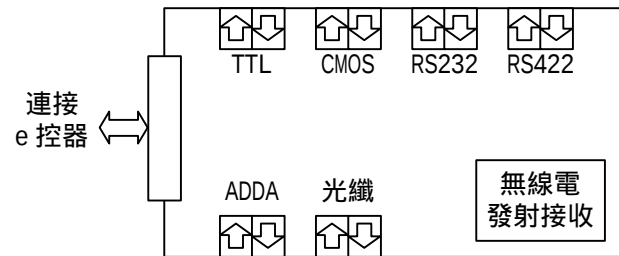
### LCD操作界面



LCD操作界面的電路板配置圖如上，其中包括：

1. 240\*128解析度的單色LCD，顯示範圍約為11cm\*6cm，但是PCB的板面稍大，佔用了14cm\*10cm的空間。
2. 安裝了麥克風和喇叭，可以接收和傳遞語音訊號。要注意，本系統雖然具備語音辨認所需的硬體，但是並未提供相關軟體。使用者若需語音辨識的功能，必須自行開發軟體。
3. 安裝了IR接收器，可接受市售紅外線遙控器的控制。
4. 遙控器搭配LCD螢幕，就成為控器的人機操作界面。我們會提供示波器軟體以監測實驗結果。
5. 電路板尺寸約10cm\*18cm，可平放在e控器上，透過ext擴張座和e控器連接。
6. LCD操作界面完全由FPGA控制，LCD控制程式已完成，但是使用者仍可依需要自行變更。

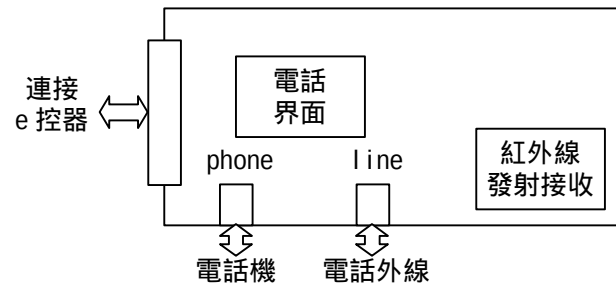
## 實驗模組配置



實驗模組的電路板配置圖如上，其中包括：

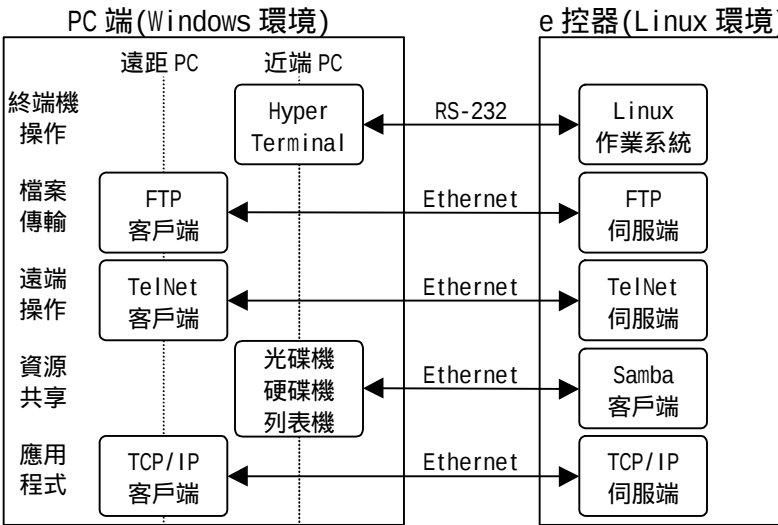
1. 實驗模組是獨立單板，尺寸為7cm\*15cm。由26P接頭(實際上只有25P的連接)對eM卡連接。
2. 26P的接頭訊號中包括了
  - 1-ch的類比輸出訊號，
  - 1-ch的類比輸入訊號，
  - 8-bit的數位輸出訊號，
  - 8-bit的數位輸入訊號，
  - +12V和+5V電源。
3. 實驗模組提供數位通訊和類比通訊的實驗界面，包括：
  - A. 直接數位連線(TTL/CMOS)。
  - B. 直接類比連線(+/-10V的AD/DA訊號)。
  - C. 串列界面(RS232/RS422)。
  - D. 無線電發射與接收界面。
  - E. 紅外線發射與接收界面。
  - F. 光纖通訊界面。
  - G. 電話通訊界面。
4. 實驗時只要將對應的送訊和收訊接頭直接短路即可。

由於板面限制，紅外線和電話界面獨立放在另一塊單板上，電路配置如下：



# 軟體規劃

e控器雖然工作在Linux作業環境下，但是e控器本身並不提供終端機，所有的操作界面都在PC端的視窗環境下完成。操作系統的方塊圖如下：



操作系統包括了e控器、近端PC和遠端PC三部分，其中

- e控器：提供所有硬體控制和軟體管理等黑箱作業的控制功能。
- 近端PC：提供軟硬體開發過程中的開發工具和作業環境。
- 遠端PC：提供實際應用時的遠端監控功能。

以下針對各部分的規劃做詳細說明。

## Linux作業系統

Linux是標準的開放式作業系統，我們採用內藏式(embedded Linux)系統，結構說明如下：

| 項 目        | 說 明                                                                                                                                                                                                                                                                                            |
|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Linux版本    | 1. 基於Slackware 7.0版本建成壓縮檔<br>2. Kernal為v2.2.19                                                                                                                                                                                                                                                 |
| 記憶體需求      | 1. 壓縮後為6.5MB，佔用DOC(8MB)的88%容量<br>2. 解壓縮後為14MB，佔用RAM(64MB)的22%容量                                                                                                                                                                                                                                |
| C編譯器版本     | 1. GNU的C編譯器為egcs-1.1.2<br>2. GNU的C工具庫為binutils-2.9.1.0.25<br>3. GNU的C程式庫為glibc-2.1.2                                                                                                                                                                                                           |
| 主機         | 1. 固定為COM1，參數為9600,N,8,1<br>2. 文字操作模式(Text Mode)                                                                                                                                                                                                                                               |
| 檔案結構       | /boot 開機執行區<br>/bin Linux執行程式區<br>/sbin Linux系統工具區<br>/usr/bin 使用者的執行程式區<br>/usr/sbin 使用者的系統工具區<br>/etc 系統參數設定區<br>/dev 驅動程式區<br>/proc 程序控制檔案區<br>/mnt 新裝置裝設區域<br>/lib 動態程式庫區<br>/usr/lib 使用者的動態程式庫<br>/usr/src 使用者的程式碼<br>/smbcd 對應到PC端的光碟機<br>/smb 對應到PC端的硬碟目錄<br>/smb/usr/src 對應到PC端的使用者程式碼 |
| FTP伺服器端    | 執行FTP伺服器端程式，可接受遠端FTP操作                                                                                                                                                                                                                                                                         |
| Telnet伺服器端 | 執行Telnet伺服器端程式，可接受遠端Telnet操作                                                                                                                                                                                                                                                                   |
| Samba客戶端   | 執行Samba客戶端程式，分享PC端的光碟機、硬碟機和印表機                                                                                                                                                                                                                                                                 |

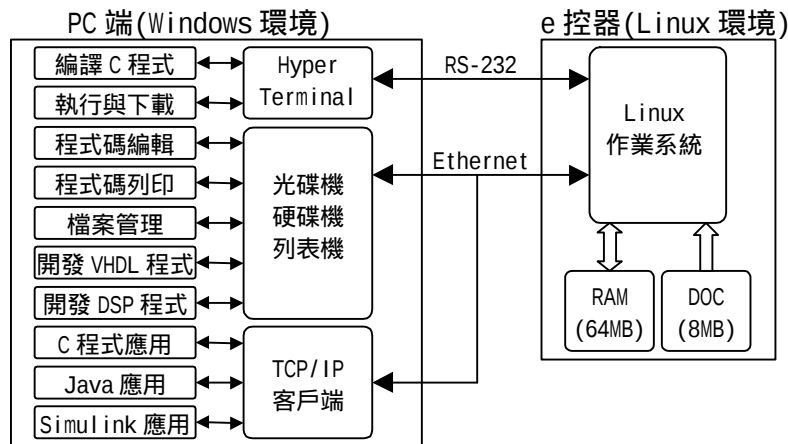
從操作的觀點而言，這個Linux系統不止是包括e控制器，還包括PC端的軟硬體裝置，例如終端機、光碟機、硬碟機和印表機等等。這些經由網路連線後達成的資源共享功能，使得小小的一台e控制器能發揮整台電腦和所有周邊設備的完整功能。

## 視窗操作環境

但是內藏式Linux也有相當多先天和後天上的缺點，例如說：

1. 不論如何，這是一個文字模式的操作系統。對於用慣視窗環境的設計者而言，好像時光又倒流到古早的年代。
2. 微軟視窗的作業環境是市場主流，這是無法忽視的事實。對大部分的操作者而言，Linux作業環境簡直是天書，完全不知道如何操作。

因此，我們在PC端的操作上放棄Linux系統，仍然採用微軟的視窗作業環境，以方便絕大多數熟悉視窗環境的客戶群。



在e控制器的Linux作業環境中，我們設定：

1. DOC中的壓縮檔在開機時解壓縮並下載到RAM中執行。
2. 檔案的讀寫或管理都在RAM中，一旦關機，使用者所儲存在RAM中的資料都會喪失，不會儲存到DOC中。
3. 換句話說，DOC所儲存的檔案都是唯讀檔案，一般的使用者不准改變。在病毒橫行的網路世界中，提供了一個最佳的免疫環境。
4. 只有終端機控制的系統管理者准許重設DOC檔案內容，以應付專案設計和應用的特殊狀況。
5. 所有的使用者程式實際上都儲存在PC端共享的硬碟檔案中。因為共享，所以PC端也可以在視窗環境下直接處理檔案，不用透過Linux系統處理。
6. 同樣的，光碟和印表機也可在視窗環境下直接處理，不用透過Linux系統處理。

如此一來，必須在Linux中執行的程序就會大幅減少，最後需要的只是C程式的編譯和下載而已。而這些都是極簡單的指令，還可做成批次檔再度簡化。而最麻煩的程式編輯和檔案管理工作，都變成視窗環境中的內部處理工作，和Linux作業系統已經完全無關。

我們將e控制器中所需的所有開發程序一一列舉於下，其中只有兩三行和Linux相關。對於不懂Linux的使用者而言，也可輕易執行這些開發程序。

| 項 目                  | 說 明                                                 |
|----------------------|-----------------------------------------------------|
| C程式開發<br>(e控制器部分)    | 1. 直接在視窗中編輯程式碼<br>2. 在終端機中下達編譯指令<br>3. 在終端機中執行下載指令  |
| DSP程式開發<br>(e控制器部分)  | 1. 直接在視窗中編輯程式碼<br>2. 直接在視窗中下達編譯指令<br>3. 在應用程式中下載並執行 |
| VHDL程式開發<br>(e控制器部分) | 1. 直接在視窗中編輯程式碼<br>2. 直接在視窗中下達編譯指令<br>3. 在應用程式中下載並執行 |
| C程式開發<br>(PC端部分)     | 所有程序都在視窗環境處理                                        |
| JAVA程式開發             | 所有程序都在視窗環境處理                                        |
| Matlab開發             | 所有程序都在視窗環境處理                                        |
| Simulink開發           | 所有程序都在視窗環境處理                                        |
| 應用程式執行               | 所有程序都在視窗環境處理                                        |

對於PC端的應用程式而言，e控制器只是透過TCP/IP操作的遠端裝置，完全沒有作業系統的差異，對於e控制器是否使用Linux已經完全無關。因此，不論是視窗環境中的C程式、Java程式、Matlab程式或是Simulink程式，都可以輕易在TCP/IP的規格下對e控制器做遠距遙控。

## 開機啟動程序

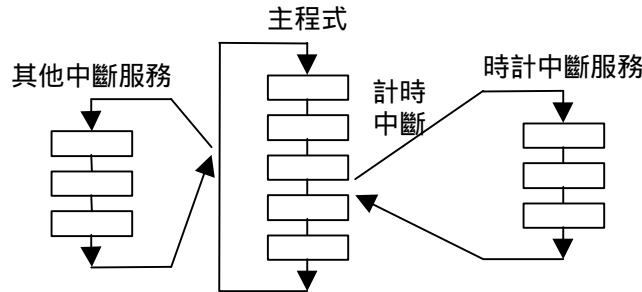
e控器本身像是一個黑箱作業的控制器，從外部很難瞭解其中的動作和意義。下面我們透過其開機啟動程序，詳細說明其中的細部控制動作：

| 開機程序    | 搭配程序                         | 說 明                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|---------|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| BIOS準備  | 所有參數事先設定，直接燒入BIOS的flash ROM中 | <ol style="list-style-type: none"> <li>1. 設定COM1為主機，參數為(9600,N,8,1)。</li> <li>2. 跳過RAM測試。</li> <li>3. 設定DOC的位址在c8000~c9fff的8K空間。</li> <li>4. 設定FPGA的位址在d0000~d0fff的4K空間。</li> </ol>                                                                                                                                                                                                                                                  |
| PC端準備   |                              | <ol style="list-style-type: none"> <li>1. 設定網路的識別資料，預設為"WORKGROUP"和"PCClient"。</li> <li>2. 確認PC和e控器的IP位址屬於同一族群。</li> <li>3. 將光碟機設為共享，名稱為"smbcd"</li> <li>4. 將硬碟的c:\smb目錄設為共享，名稱為"smb"</li> <li>5. 將印表機設為共享，名稱為"smbprt"</li> </ol>                                                                                                                                                                                                    |
| BIOS啟動  | PC端執行HyperTerminal模擬終端機      | <ol style="list-style-type: none"> <li>1. 檢查I/O，但不檢查鍵盤和軟碟。</li> <li>2. 以COM1為主機，顯示BIOS資訊</li> <li>3. 檢查並載入DOC的驅動程式，連接DOC。</li> </ol>                                                                                                                                                                                                                                                                                                 |
| Linux啟動 | PC端從HyperTerminal監視訊息        | <ol style="list-style-type: none"> <li>1. 從DOC中載入Linux壓縮檔案並解壓縮。</li> <li>2. 解壓縮後將整個Linux系統檔案置入RAM中。<br/>執行動作依序為<br/>DOC-lilo<br/>Lilo<br/>Linux核心碼<br/>預設I/O初值<br/>預設TCP/IP初值<br/>安裝FTP伺服器<br/>安裝Telnet伺服器</li> <li>3. 其中DOC-lilo和Lilo的參數檔為<br/>/etc/lilo.conf。</li> <li>4. 其中Linux核心碼的參數檔為<br/>/etc/inittab</li> <li>5. 其中PC端的預設名稱定義在下述檔案中<br/>/etc/HOSTNAME</li> <li>6. 設定COM1為主機，參數為(9600,N,8,1)</li> <li>7. 執行login動作</li> </ol> |
| 操作開始    | PC端從HyperTerminal下達指令        | <ol style="list-style-type: none"> <li>1. 系統操作者的login動作為<br/>user: root<br/>password: (無)</li> <li>2. 一般操作者的login動作為<br/>user: guest<br/>password: guest</li> </ol>                                                                                                                                                                                                                                                                  |

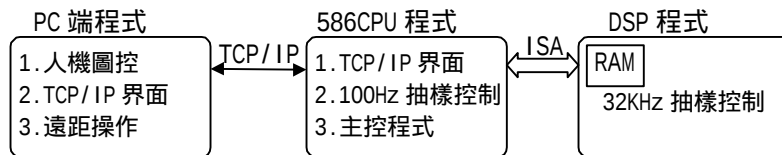
|      |  |                                                                                                                                                                                                                                                        |
|------|--|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 硬碟安裝 |  | <ol style="list-style-type: none"><li>1. 將"e控制器"CD置入光碟機中。</li><li>2. 執行安裝程序。</li><li>3. 安裝後所有e控制器的相關檔案都置於c:\smb的目錄中。</li><li>4. 將c:\smb設為共享，供Linux透過網路使用。</li></ol>                                                                                    |
| 應用啟動 |  | <ol style="list-style-type: none"><li>1. 下載FPGA電路(內含DSP電路)</li><li>2. 下載DSP中的微控碼</li><li>3. 下載DSP中的程式碼</li><li>4. 下載DSP中的資料區預設值</li><li>5. 執行TCP/IP通訊程式</li><li>6. 執行PC端應用程式</li><li>7. 啟動DSP控制</li><li>8. 上述程序可用批次檔執行，或是啟動PC端應用程式後在應用程式中執行。</li></ol> |

## 即時控制環境

即時控制在軟體設計上就是週期性的抽樣控制，每隔一個固定的抽樣週期就產生中斷訊號，而執行時計中斷服務的程式。如下圖所示：



如果微電腦中其他的中斷服務過多，或是負載過重，都會影響時計中斷服務的精度，直接或間接影響到整體的控制精度。事實上，Linux作業系統並不是完全理想的即時作業環境，在高密度的網路通訊環境下也的確造成不小的工作負載。因此，我們在FPGA電路中設計了DSP控制器，將週期性的抽樣控制工作完全移轉到DSP中執行，確保即時控制環境的效率及精度。方塊圖如下：



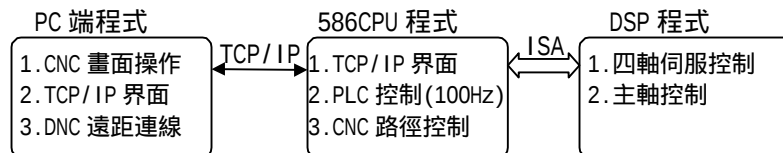
其中

1. DSP部分只負責即時控制程式，所以抽樣控制頻率可高達32KHz，完全不受Linux作業系統干擾。
2. 而在Linux作業系統下執行的586CPU，則需要掌管主控程式和TCP/IP通訊界面程式，因此最多只能執行100Hz左右的抽樣控制。
3. 所謂100Hz左右，是指基本頻率是100Hz，但是多少會受到其他程式負載所影響，不能完全保證。
4. 586CPU和DSP之間可透過雙埠記憶體共享所有的參數和變數。透過ISA匯流排傳送，可以視為沒有傳訊速度限制。
5. PC端應用程式則可執行人機圖控和TCP/IP通訊界面，執行遠距操作的功能。
6. PC端和586CPU之間透過網路做連線。若是近距離的Ethernet，可以執行100Hz左右的即時資料傳輸(200-byte左右)。
7. 若是透過網際網路做遠距連線，則只能傳遞資料而不能保證傳訊速度了。

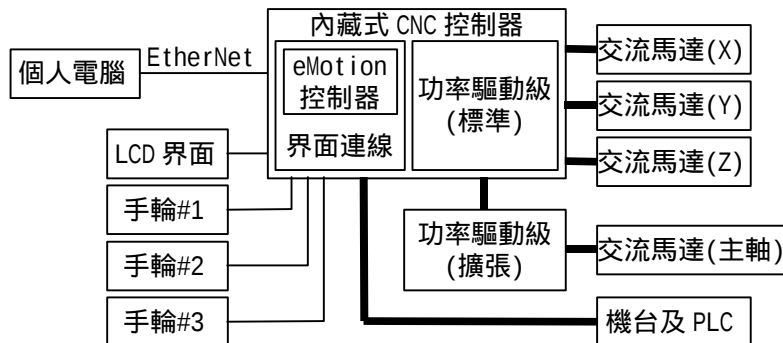
## 應用規劃

e控制器本身是獨立模組，也可以透過網路搭配PC工作。可以執行黑箱作業，也可以透過網路監控所有內部參數，執行完全透明的教學實驗。下面列出幾個重要的應用構想。

### 四軸CNC控制器

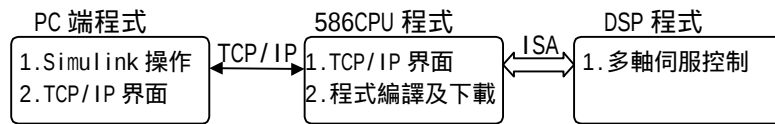


四軸CNC控制器的軟體方塊圖如上，其中PC端負責操作界面和遠端監控，e控制器負責CNC的整個控制動作，而將控制負載由586CPU和DSP分別承擔。

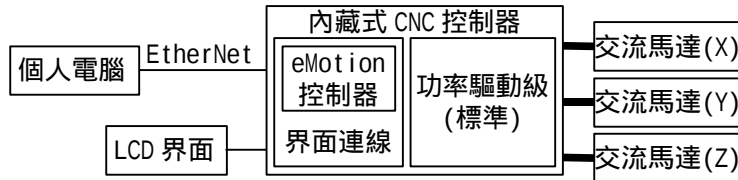


在硬體部分，則將e控制器直接整合在功率驅動級中，形成單一模組的內藏式CNC控制器。不需外加伺服驅動器即可控制多軸的伺服馬達。整個系統可連接工業級PC做標準型的CNC控制器，也可只連接LCD面板做簡易型的CNC控制器。

## 電動機控制實驗

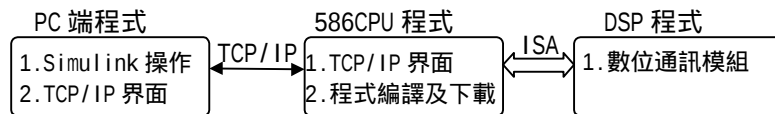


電動機控制實驗的軟體方塊圖如上，其中PC端負責Simulink的操作和監控，而e控制器負責整個多軸伺服控制動作。e控制器中的586CPU只負責網路資料的傳遞，所有的控制都以模組的方式架構在DSP程式中。



硬體部分的架構則和CNC控制器雷同，將e控制器直接整合在功率驅動級中，以控制多軸的伺服馬達。整個系統在PC的操作下執行細部模組的教學實驗，也可連接LCD面板做即時顯示的示波器功能。

## 數位通訊實驗



數位通訊實驗和電動機控制實驗一樣，只是實驗目標換成通訊原理而已。但是硬體更單純了，直接用e控制器連接各實驗模組即可。



不只是教學實驗。因為e控制器體積小可以直接置於實驗系統中，軟硬體工具齊全、系統開放又有測試透明化的優點，是專題實驗和專題研究的最佳選擇。