

國立交通大學

電機與控制工程學系

碩士論文

具備多工排程功能之無線嵌入式腦機介面
系統及其在即時汽車駕駛員疲勞狀態
偵測與提醒之應用



Development of Wireless Brain Computer Interface
with Embedded Multi-task Scheduling and its
Application on Real-time Driver's Drowsiness
Detection and Warning

研究生：謝弘義

指導教授：林進燈 博士

中華民國九十五年七月

具備多工排程功能之無線嵌入式腦機介面系統及其在即時

汽車駕駛員疲勞狀態偵測與提醒之應用

Development of Wireless Brain Computer Interface with
Embedded Multi-task Scheduling and its Application on
Real-time Driver's Drowsiness Detection and Warning

研究生：謝弘義

Student : Hung-Yi Hsieh

指導教授：林進燈 博士

Advisor : Dr. Chin-Teng Lin



A Thesis

Submitted to Department of Electrical and Control Engineering

College of Engineering and Computer Science

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of Master

in

Electrical and Control Engineering

June 2006

Hsinchu, Taiwan, Republic of China

中華民國 九十五年 七月

具備多工排程功能之無線嵌入式腦機介面系統及其在即時

汽車駕駛員疲勞狀態偵測與提醒之應用

學生：謝弘義

指導教授：林進燈 博士

國立交通大學電機與控制工程研究所

中文摘要

近年來，有許多高速公路車禍發生的原因是由於駕駛者精神狀態不佳所造成；而在精神狀態不佳或是疲勞時，腦電位會產生一些特徵訊號，因此可以利用這些生理特徵訊號進行其精神狀態之估測。以往的生理訊號監控系統大多只做到訊號記錄部分，再藉由電腦進行後續離線處理，而缺乏即時分析的功能，因此，本論文提出一套具備即時分析能力的腦機介面系統以改善以往生理訊號監控系統的缺點，並實際應用於汽車駕駛員疲勞狀態偵測與提醒。我們以雙核心處理器，搭配無線傳輸模組、生理訊號放大器，來實現一套即時無線腦機介面系統，包含三大發展主軸，分別為「無線傳輸與腦電位訊號量測」、「嵌入式多工排程」及應用於「即時汽車駕駛員疲勞狀態偵測與提醒」並即時運作於嵌入式系統上。首先，我們開發一套嵌入式多工排程系統與雙核心處理系統來增加系統於接收資料以及訊號處理的即時處理效能，並利用可切換式無線訊號傳輸系統來克服系統接線以及傳輸距離的問題。接著提出一即時的駕駛員疲勞狀態偵測的演算法，最後整合提醒訊號回饋以喚醒精神狀態不佳的駕駛員。實際線上測試顯示，本論文所提出的系統在即時處理效能以及疲勞狀態偵測的準確度上，都有非常好的效果。

關鍵字：無線傳輸，即時雙核心嵌入式系統，嵌入式多工排程，腦機介面，

腦波訊號處理，疲勞狀態偵測與提醒

Development of Wireless Brain Computer Interface with Embedded Multi-task Scheduling and its Application on Real-time Driver's Drowsiness Detection and Warning

Student: Hung-Yi Hsieh

Advisor: Dr. Chin-Teng Lin

Department of Electrical and Control Engineering
National Chiao Tung University

Abstract

Recently, many traffic accidents on the highway are caused by the drivers' drowsiness. If the driver is drowsy, there are some features in EEG signals. We can use these features to estimate the driver's drowsiness. But the past bio-signal monitor system usually can only record the signals and is unable of real-time signal processing. We propose the Brain Computer Interface (BCI) system in this thesis, which can process bio-signal real-time, and we apply it to detect the driver's drowsiness and warn the driver when the driver's drowsiness occurs. The goal of this thesis is to build up a real-time wireless Brain Computer Interface, which include wireless transmission and bio-signal measurement, an embedded multi-task scheduling system, and its application on real-time driver's drowsiness detection and warning. We propose the embedded multi-task scheduling system and the dual-core processing system to improve the ability of real-time processing, including receiving data and processing data, and use a selectable wireless transmission system to overcome the problems of routing and transmission distance. Finally, we propose a real-time driver's drowsiness detection algorithm and integrate the warning signal to warn the drowsiness driver. The practical online test shows that the system we propose is effective with the ability of real-time processing and drowsiness detection.

KEYWORD : wireless transmission, real-time dual-core embedded system, embedded multi-task scheduling, Brain Computer Interface, Brain Signal Processing, drowsiness detection and warning

誌 謝

本論文的完成，學生首先要感謝指導教授林進燈博士在過去兩年研究期間提供豐富的實驗資源、研究環境以及本校生物科技系梁勝富教授從旁指導協助，讓學生受益良多。此外，學生要感謝口試委員台灣大學陳永耀教授、亞洲大學蔡文祥校長以及美國加州大學聖地牙哥分校鍾子平教授在百忙中抽空參與學生的口試，並提供許多寶貴的意見，使得本論文更加完善。

其次，我要感謝腦科學研究中心的全體成員，沒有他們也就沒有我個人的成就。特別感謝的黃騰毅學長和柯立偉學長給予我在各方面的指導，無論是研究上疑難的解答、研究方法、寫作方式以及經驗分享等惠我良多，很慶幸在研究生涯上能遇到如此的學長。另外，要感謝梅庭璋、蕭力碩以及謝宗哲同學在過去兩年研究生活中同甘共苦，相互扶持。此外，也要感謝陳玉潔學姊、趙文鴻學長在研究上的幫助以及生活中的開導，還有感謝林攻學妹過去一年在實務上以及實驗測試中的協助。再者，我要感謝陳明達、林弘章、黃俊傑學弟在過去這一年中的相助，在生活中多一些調劑。另外，我要感謝遠在加拿大的鄭吉泰學長在求學的過程中引導我，並讓我創造許多重要而且美好的回憶。

最後，我要想感謝我的父親謝本松先生、母親張素霞女士以及兩個弟弟，由於他們長久的栽培和照顧，才有今天的我，在此我要說一聲，真的非常地謝謝你們，還有三舅舅王國順先生一家人，謝謝你們的鼓勵與幫助。謹將本論文獻給我的家人以及一路上陪伴我以及幫助過我的所有人，包括大學的社團夥伴、專題戰友，尤其是郭于禎小姐以及孫將鈺同學，常常在我最孤單無助的時候給我協助，在我最徬徨、困擾的時候，願意傾聽我的心聲、開導我，在人生的路程中能認識你們這些好朋友讓我有勇氣面對未來的挑戰繼續前進。

目 錄

誌 謝	iv
目 錄	v
圖目錄	vii
表目錄	x
縮寫對照	xi
第一章 緒論	1
1.1 研究背景.....	1
1.2 研究動機與研究目的.....	2
1.3 腦機介面系統簡介.....	3
1.4 即時可攜式無線腦機介面系統整體架構.....	4
1.5 論文架構.....	6
第二章 腦電位訊號感測與無線傳輸系統	7
2.1 簡介.....	7
2.2 腦電位訊號感測.....	8
2.2.1 腦電位訊號簡介.....	8
2.2.2 腦電位訊號量測電路設計.....	10
2.3 無線傳輸系統.....	17
2.3.1 藍芽無線傳輸簡介.....	17
2.2.2 RF3100/3105 簡介	18
2.2.3 無線通訊系統比較.....	19
2.2.4 CPLD 與 VHDL 硬體描述語言	20
2.2.5 類比/數位訊號轉換與無線傳輸模組控制程式開發	22
2.2.6 以半雙工通訊機制實現雙向資料傳輸.....	27
第三章 嵌入式多工排程系統設計	29
3.1 嵌入式系統開發環境簡介.....	29
3.1.1 嵌入式硬體開發環境.....	30
3.1.2 嵌入式軟體開發環境.....	33
3.1.3 雙核心處理器之內部溝通機制.....	36
3.2 嵌入式軟體系統架構與簡介.....	39
3.3 高效率嵌入式多工排程系統之設計與實現.....	40
3.3.1 工作切割與基本排程架構.....	40
3.3.2 多重處理程序之溝通機制.....	41

3.3.3 基於嵌入式多工排程之系統架構設計.....	47
3.3.4 ARM 處理器端周邊裝置控制程式設計與實現.....	49
第四章 汽車駕駛員疲勞狀態偵測與提醒系統之設計	52
4.1 疲勞狀態偵測與提醒系統架構.....	52
4.2 駕駛員疲勞狀態估測實驗設計.....	53
4.3 疲勞狀態偵測演算法設計.....	54
4.3.1 腦電位訊號處理流程.....	55
4.3.2 訊號之前處理.....	57
4.3.3 主成分分析.....	58
4.3.4 線性回歸分析.....	59
4.4 疲勞狀態偵測與提醒系統實現.....	60
4.4.1 DSP 處理器端運算系統實現.....	60
4.4.2 ARM 處理器端與 DSP 溝通及訊號後處理系統實現.....	63
4.4.3 疲勞提醒系統之實現.....	64
第五章 測試結果與討論	66
5.1 系統測試結果.....	66
5.1.1 生理量測電路驗證測試.....	68
5.1.2 多工排程效能比較.....	69
5.1.3 腦波訊號處理系統正確性測試.....	71
5.2 系統整合與討論.....	75
5.3 相關研究比較與討論.....	78
第六章 結論與未來展望	80
參考文獻	82

圖目錄

圖 1.1、系統方塊圖.....	4
圖 1.2、生理訊號放大電路方塊圖.....	4
圖 1.3、無線訊號傳輸架構圖.....	5
圖 2.1、腦電位訊號感測與無線傳輸系統架構圖.....	7
圖 2.2、10-20 系統電極擺設位置圖.....	9
圖 2.3、腦波量測電路方塊圖.....	10
圖 2.4、前置放大器電路圖.....	10
圖 2.5、前置放大器訊號模擬圖.....	11
圖 2.6、訊號隔離器電路圖.....	11
圖 2.7、高通濾波器電路圖.....	12
圖 2.8、低通濾波器電路圖.....	13
圖 2.9、帶通濾波器頻率響應圖.....	14
圖 2.10、增益放大器電路圖.....	14
圖 2.11、模擬結果.....	15
圖 2.12、各頻率模擬圖.....	15
圖 2.13、帶拒濾波頻率響應圖.....	16
圖 2.14、無線訊號傳輸架構圖.....	17
圖 2.15、藍芽無線傳輸模組實體圖.....	17
圖 2.16、RF3100 無線傳輸模組實體圖.....	19
圖 2.17、RF3105 無線傳輸模組實體圖.....	19
圖 2.18、FLEX 系列發展版.....	21
圖 2.19、Max + plus II 軟體介面.....	21
圖 2.20、無線訊號傳輸架構圖.....	22
圖 2.21、RF3100 控制模組架構圖.....	22
圖 2.22、AD570 內部架構圖.....	23
圖 2.23、AD570 控制時序圖.....	23
圖 2.24、無線訊號接收程式流程圖.....	24
圖 2.25、無線藍芽控制模組架構圖.....	25
圖 2.26、傳輸資料對照表示意圖.....	26
圖 2.27、無線藍芽模組交握溝通流程圖.....	26
圖 2.28、雙向傳輸示意圖.....	28
圖 3.1、OMAP 1510 微處理器架構圖.....	31
圖 3.2、Innovator Development Kit 外觀圖.....	32
圖 3.3、GCC 編譯流程圖.....	34
圖 3.4、CCS 程式發展設計流程.....	35
圖 3.5、CCS 整合發展環境.....	36

圖 3.6、DSP Gateway 系統架構圖	37
圖 3.7、ARM/DSP 內部傳輸說明圖	38
圖 3.8、嵌入式系統架構圖	39
圖 3.9、嵌入式系統排程示意圖	41
圖 3.10、記憶體運用說明圖	43
圖 3.11、共享記憶體連接示意圖	45
圖 3.12、資料處理狀態圖	48
圖 3.13、周邊控制程序流程圖	49
圖 3.14、區域網路共享架構圖	50
圖 3.15、遠端電腦使用者圖形介面	50
圖 3.16、TCP/IP 網路連線流程圖	51
圖 4.1、疲勞狀態偵測與提醒系統架構圖	52
圖 4.2、以虛擬實境產生的快速道路場景	53
圖 4.3、實際實驗場景與動感模擬平台	54
圖 4.4、實驗電極黏貼示意圖	54
圖 4.5、估測模型建立流程圖	55
圖 4.6、腦電位訊號處理流程圖	55
圖 4.7、訊號之前處理	57
圖 4.8、訊號處理流程圖	60
圖 4.9、DSP 訊號處理程式流程圖	60
圖 4.10、訊號處理程式流程圖	62
圖 4.11、腦電波訊號的頻率分布圖	63
圖 4.12、DSP 程式啟動流程	63
圖 4.13、資料後處理程式流程	64
圖 4.14、用於雙向資料傳輸的 RF3100 控制模組架構圖	65
圖 4.15、雙向傳輸資料辨識流程	65
圖 5.1、系統實體圖	66
圖 5.2、即時無線生理訊號分析系統系統細部說明圖	67
圖 5.3、生理訊號放大電路板	68
圖 5.4、 α 腦電波圖	68
圖 5.5、眼動腦電波圖	69
圖 5.6、程序排程示意圖	69
圖 5.7、不用多工排程的執行過程	70
圖 5.8、使用多工排程的執行過程	71
圖 5.9、不含 PCA 的演算法測試	72
圖 5.10、含 PCA 的演算法測試	72
圖 5.11、PCA 矩陣處理驗證流程	73
圖 5.12、PCA 矩陣處理驗證測試	73

圖 5.13、PCA 維度與預測精確度的關係.....	74
圖 5.14、在 OMAP 上做系統測試.....	75
圖 5.15、以方法三在 OMAP 上做預測測試.....	75
圖 5.16、系統整合架構圖.....	76
圖 5.17、顯示介面圖.....	76



表目錄

表 2.1、腦電波分類表.....	9
表 2.2、帶通濾波器輸入輸出表.....	13
表 2.3、帶拒濾波器輸入輸出表.....	16
表 2.4、藍芽無線通訊協定.....	18
表 2.5、無線傳輸格式傳輸功率比較表.....	20
表 2.6、無線傳輸模組規格表.....	20
表 2.7、RF3100 控制模組接腳表.....	22
表 2.8、無線封包.....	24
表 2.9、無線藍芽控制模組接腳表.....	25
表 3.1、旗標對照表.....	44
表 3.2、ARM 和 DSP 工作分配表.....	47
表 4.1、RF3100 控制模組接腳表.....	65
表 5.1、效能比較表.....	70
表 5.2、DSP 處理測試比較表.....	73
表 5.3、演算法於 OMAP 上測試比較表.....	74
表 5.4、相關研究比較表.....	79



縮寫對照(按字母排序)

縮寫	原文	中文翻譯
A/D	Analog to Digital	類比數位訊轉換
API	Application Program Interface	應用程式介面
ARM	Advance RSIC Machines	
BCI	Brain Computer Interface	大腦人機介面
CPLD	Complex Programmable Logic Device	
DMA	Direct Memory Access	直接記憶體存取
DSP	Digital Signal Processor	數位訊號處理器
ECG	electrocardiogram	心電圖
EEG	electroencephalogram	腦電波
EMG	electromyogram	肌電圖
EOG	electro-oculogram	眼電圖
FHSS	Frequency Hopping Spread Spectrum	跳頻展頻
FPGA	Field Programmable Array	
FSK	Frequency Shift Keying	頻率鍵移
GCC	GNU Compiler Collection	
GSM	Global System for Mobile Communication	全球行動通訊系統
GPRS	General Packet Radio Service	通用分組無線服務
HCI	Host Control Interface	主機控制介面
IDE	Integrated Development Environment	整合發展環境



縮寫	原文	中文翻譯
IP	Internet Protocol	網際網路通訊協定
IPC	inter-process communication	處理程序之間的溝通機制
L2CAP	Link Layer Control & Adaptation Protocol	
LCD	Liquid Crystal Display	液晶平面顯示器
LMP	Link Manager Protocol	
LSE	least – square – error	誤差的最小平方
MMU	Memory Management Unit	記憶體管理單元
MPU	Micro Processor Unit	
OMAP	Open Multimedia Architecture Platform	
OS	Operating System	作業系統
PCA	Principle component analysis	主成分分析
PDA	Personal Digital Assistant	個人數位助理
PLD	Programmable Logic Device	可程式邏輯邏輯元件
QoS	Quality of Service	服務品質
RF	Radio Frequency	射頻
RISC	Reduce Instruction Set Computer	精簡指令集電腦
SDP	S ervice Discovery Protocol	服務發現協定
SoC	System on Chip	系統晶片
SRAM	Static Random-Access Memory	靜態隨機存取記憶體

縮寫	原文	中文翻譯
STFFT	Short-Time FFT	短時快速傅立葉轉換
TCP	Transmission Control Protocol	傳輸控制協定
TCS	Telephone Control Service	電話傳送控制協定
UART	Universal Asynchronous Receiver Transmitter	通用異步收發器
USB	Universal Serial Bus	通用串列匯流排
Verilog HDL	Verilog Hardware Description Language	
VHDL	Very High Speed Integrated Circuit Hardware Description Language	



第一章 緒論

1.1 研究背景

人類腦電位訊號分析研究起源於 1920 年代，Berger 等人發現人類的腦波是由大腦皮質下許多神經細胞後端的突觸電流所累積產生。經過數十年的研究結果發現[1][6]，藉由腦波的量測分析可以得知受測者的認知精神狀態。然而進行腦波相關研究通常需要長時間的訊號量測與分析，如此大量且複雜的腦波訊號將耗費許多人力與時間，直到近年來電腦運算速度日益進步，以及各種分析處理方法之改良，才足以即時處理量測到的腦波訊號。傳統上的腦波儀器是利用工業電腦之類的大型主機來做監測，將其腦波記錄後經過人工方式來分析其生理意義。現今軟硬體發展迅速，雖然大型主機腦波訊號即時分析的應用已不是個問題，但是大型主機，空間大且昂貴，在實用上十分不方便。因此，如何發展一個攜帶方便又可以即時分析腦波訊號的系統變成一個很重要的課題。

大腦人機介面(Brain Computer Interface, BCI)是利用腦部訊號，讓使用者可以直接與外界溝通。這項技術的基礎在於，當大腦在進行某一項特定活動的時候，會產生特定的腦波變化，因此透過監測與辨認此特定的腦波變化，來達到利用腦波直接控制儀器或輸出訊號之目的。此種技術最主要的貢獻在於幫助因神經肌肉損傷而行動受到阻礙的人(如肌肉萎縮、中樞神經系統損傷、重度中風的病人等)，使他們可以不需要依靠周邊神經和肌肉，能夠使用腦部的訊號，就能達到與外界溝通、傳達訊息、自主行動，以及自我照顧等目的。在這方面的研究中，大多著重於病患的研究，而且因為腦波量測與分析的儀器非常龐大，所以使用上非常不方便。正如同腦電波量測與即時分析所遇到的問題，在腦機介面的發展中，如何縮小量測、分析系統的體積以及簡化使用上的困難變成為將大腦人機介面時用於日常生活中所需克服的難題。

在相關的研究中，國內一直停留在訊號紀錄的階段[7]-[13]，較少應用於即時的分析與預測受試者的行為。國外則有一些實驗室從事大腦人機介面相關的研究，包括北京清華大學高上凱教授實驗室與奧地利 Graz-BCI 實驗室[14]-[16]。而在可攜式系統的研究上，高上凱教授實驗室研究藉由視覺刺激並透過數位訊號處理器分析腦電波，用以無線控制家電[17]；Graz-BCI 實驗室發展可攜式的人腦與電腦介面系統[18]；美國賓州州立大學發展一套失神性癲癇的腦波偵測系統[19]。另外，在嵌入式系統於大腦人機介面的研究上較為少見，而較多

ECG(electrocardiogram, 心電圖)處理採用嵌入式系統的方式[20]-[22], 處理系統則以單一處理器為主[23-25]。

由於單一處理器的處理能力不足, 因此在傳統中極少有利用嵌入式系統來進行大腦人機介面的研究。因此, 本論文提出以一雙核心處理器為處理核心的嵌入式系統來解決大腦人機介面系統中最複雜的即時運算部份, 因為嵌入式系統強調體積小與省電, 其特色非常適合用於大腦人機介面的開發。透過精簡的腦電位訊號分析演算法搭配強大的嵌入式運算處理器來讓腦機介面有即時紀錄、處理能力, 並結合無線傳輸系統, 強化本系統的實用性與方便性, 以達到即時處理與分析之目的。

1.2 研究動機與研究目的

近年來, 有越來越多高速公路車禍發生的原因, 是由於駕駛者精神狀態不佳所造成。許多研究發現在精神狀態不佳或是疲勞時, 大腦電位會產生一些特徵訊號, 因此可以利用這些生理特徵訊號進行其精神狀態之估測[26][27]。然而, 若能發展一套攜帶方便且可即時估測駕駛者精神狀態之腦波分析與疲勞警示系統, 來降低駕駛者因精神狀態不佳而發生意外的機率, 將具有重要意義的貢獻。因此, 本論文的研究目的為實現一套可以即時偵測駕駛員精神狀態的腦機介面系統, 當駕駛員的精神狀態不佳時, 系統會自動偵測到相對應之特徵腦波, 並即時發出警告訊號來警示駕駛員, 達到降低意外發生之目的。本研究利用生理訊號放大器擷取受測者的電生理訊號, 藉由無線傳輸的方式將資料傳至嵌入式處理器, 並透過我們所發展的雙核心溝通機制以及自動分析演算法來實現這樣一套即時可攜式無線腦機介面系統。

本論文主要研究的目的可分為下列五點, 分別為「多訊號生理訊號量測」、「可切換式無線通訊系統」、「改善式嵌入式訊號處理流程」、「嵌入式多工排程系統」以及「即時分析演算法」, 分別描述如下:

- 一、多訊號生理訊號量測: 本論文所實現的生理訊號量測系統可用於多種不同訊號, 例如: EEG(electroencephalogram, 腦電波)、ECG(electrocardiogram, 心電圖)、EMG (electromyogram, 肌電圖)、EOG (electro-oculogram, 眼電圖) 等電生理訊號。
- 二、可切換式無線通訊系統: 根據不同的距離需求與環境需求, 本系統提供不同的傳輸介面作為切換選擇。此外, 本系統亦提供長距離的資料傳輸與接收。

- 三、改善式嵌入式訊號處理流程：由於硬體空間的限制，當在訊號處理時，便無法完全將電腦上所使用的即時訊號處理流程來實現，而必須根據硬體的限制做一些修改。
- 四、嵌入式多工排程系統：本研究在嵌入式系統中架設了一套多工排程系統，除了可加強其即時運算與顯示的能力外，此多工分離流程，未來亦可方便且快速的加入其他新的處理程序。
- 五、即時分析演算法：在 OMAP(Open Multimedia Architecture Platform)中利用 DSP Gateway 機制將腦波分析演算法交由 ARM(Advance RSIC Machines)和 DSP(Digital Signal Processor，數位訊號處理器)分工處理，建構一駕駛員精神認知狀態系統，以驗證即時駕駛員精神認知狀態估測之能力。

1.3 腦機介面系統簡介

大腦人機介面是利用腦部訊號，讓使用者可以直接與外界溝通。這項技術的基礎在於，當大腦活動的時候，會產生特定的腦波變化，因此我們可以利用腦電波儀對腦波進行監測與辨識，最後用來控制儀器或輸出訊號。此種技術的目的在於幫助因神經肌肉損傷而行動受到阻礙的人(如肌肉萎縮、中樞神經系統損傷、重度中風的病人等)，使他們可以不需要依靠周邊神經和肌肉，能夠使用腦部的訊號，就可以達到與外界溝通、傳達訊息、自主行動，以及自我照顧等目的。腦波人機介面的發展，一方面不但能夠節省社會成本的支出，減輕病人家庭的負擔，另一方面，又能夠讓病人自我行動，建立病人與外界溝通的橋樑，提昇病人的生活品質。所以腦波人機介面在近幾年來受到國外各研究團體的廣泛重視。在過去的十年裡，研究腦波人機介面的研究團隊已經從五個以下增加到二、三十個以上，由此可見世界各國對腦波人機介面的重視。

大腦人機介面的架構可以簡單劃分為兩個部分，前端為腦波訊號的特徵擷取與辨識，後端則為與其他硬體的結合與應用，例如手部義肢，電腦(利用腦波控制滑鼠)等等。而發展成功的大腦人機介面系統決定於兩個重要的部分，包括可辨識之腦波特徵及辨識率高的分類器[28]。本論文針對大腦人機介面系統，從前端的腦波訊號擷取到後端與其他硬體的結合與應用，提出一套完整的系統，並將這個系統應用在日常生活中。

1.4 即時可攜式無線腦機介面系統整體架構

本論文提出的即時可攜式無線腦機介面系統方塊圖如圖 1.1 所示，此系統可分為五大區塊：(1)生理訊號放大、(2)無線訊號傳輸、(3)即時記錄與分析、(4)資訊顯示介面與(5)回授提醒訊號。

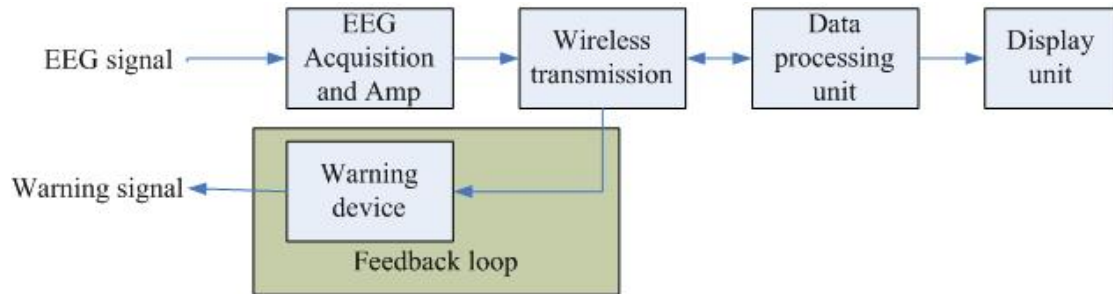


圖 1.1、系統方塊圖

以下針對此即時可攜式無線腦機介面系統所具有的五大區塊分別概述如下：

(1) 生理訊號放大

一般而言，人體電生理訊號的振幅都非常小，例如：腦電位訊號的振幅大約只有幾十到一百五十微伏(μV)，在後端處理上非常不易。因此，要先利用放大電路將訊號放大。如圖 1.2 所示，本論文所完成的放大器電路除了放大的功能外，針對生理訊號的特性，我們另外加入濾波器以及隔離電路等設計。由於生理訊號比較容易受到雜訊的干擾，所以，我們在電路中加入濾波電路來降低雜訊的干擾，以方便擷取的真正的生理訊號。此外，我們除了設計一個帶通濾波器濾掉高頻與低頻的雜訊外，為了降低室內配電的干擾，我們還加了一個帶拒濾波器。由於本系統將實際應用於人體，考量實用時電流藉由頭皮逆流進入人體造成傷害，本論文在設計上加上隔離電路來保護受測者。

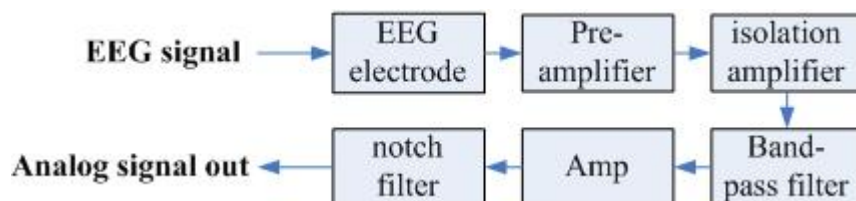


圖 1.2、生理訊號放大電路方塊圖

(2) 無線訊號傳輸功能

在無線訊號傳輸中，數位訊號抗雜訊的能力比較強，所以如圖 1.3 所示，我們先將放大電路输出的類比訊號轉成數位訊號，再以無線傳輸的方式傳至後端。在無線訊號傳輸模組的選擇上，我們考慮到傳輸功率對受測者的影響，我們選擇兩種功率較低的模組，可以根據所需的傳輸距離作選擇。

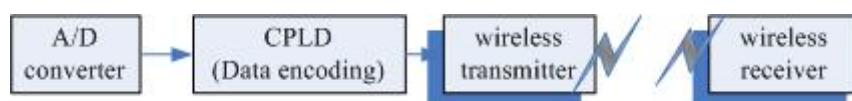


圖 1.3、無線訊號傳輸架構圖

(3) 即時記錄與分析

本論文的重點設計便在於我們所提出的即時可攜式無線腦機介面系統具備個即時訊號分析的功能。由於腦電波是一個較為複雜的訊號，因此，在計算需要比較複雜的運算，所以，我們需要一個擅於比較複雜的數學運算的處理器來負責，例如：DSP。但由於一般的 DSP 周邊溝通的能力較弱，因此，我們使用 ARM 來負責與週邊裝置溝通。因此，我們選擇結合 DSP 與 ARM 的 TI(Texas Instruments, 德州儀器) OMAP1510 來做為處理核心。OMAP 雙核心處理器所扮演的角色便是圖 1.1 中的 Data processing unit，由圖可知，OMAP 除了負責生理訊號分析運算外，還須與無線模組、網路介面以及回授提醒模組做溝通。所以，在 OMAP 上必需執行許多工作，包括：(1)無線模組裝置控制與 EEG 原始資料接收、(2)系統初始狀態配置、(3)EEG 訊號處理與資料分析、(4)數位訊號處理器之使用與管理、(5)與資料顯示介面溝通等工作。為了加強這些工作的管理與增加其即時性，本論文提出一套嵌入式多工排程系統來對這些工作做排程與管理的工作。

(4) 資料顯示介面

為了能及時顯示處理的結果與腦電位訊號，我們利用區域網路資訊共享的方式實現一套圖形化的資料顯示介面。這樣一項功能可以讓遠端的醫生或照護者可以即時掌握受測者現在的狀態。我們以網路通訊的方式來實現這一項功能，因為這種方法較為便利，而遠端不需要加裝一些特製的硬體裝置，只需要使用我們所設計的介面軟體即可，如此一來便可大幅提升其便利性。

(5) 回授提醒訊號

當系統處理完腦電位訊號後，除了偵測出其精神狀態外並藉由網路將結果顯示在遠端的電腦外，為了讓整套系統的應用價值提升，本研究在系統中加入一回授提醒訊號來即時提醒及警告駕駛員，讓系統不僅僅能做即時偵測，亦能提供即時警告的機制。另外，在回授提醒訊號的實現上，我們以聲音及燈號來實現，主要還是以聲音提醒為主。

1.5 論文架構

本論文共分為六個章節，第一章旨在說明本論文的動機、目標、研究背景及系統整體架構的介紹。第二章針對腦電位訊號量測與無線傳輸系統的部份進行說明，第三章著重於本研究在 OMAP 嵌入式系統上所設計的嵌入式排程系統，於第四章將說明本論文所開發腦波分析系統的實驗設計與實現方式。第五章將針對第二、三、四章的設計作測試與驗證，並對我們所實現的系統做討論且跟其他的相關研究做一些比較。第六章中，則是對本論文作一個總結，並對未來的研究方向作一說明。



第二章 腦電位訊號感測與無線傳輸系統

本章節將對本論文所提出的腦機介面中的腦電位訊號感測與無線傳輸系統二方面加以說明。首先，將介紹腦電位訊號的量測。說明腦電位訊號(EEG)是如何產生？有哪些類型？量測方式為何？以及放大腦電位訊號的電路設計等等。另一方面，在無線傳輸系統上，我們依據傳輸距離遠近的需求，我們選擇兩種無線傳輸模組來做實現。在硬體的實現上，我們選擇 CPLD(Complex Programmable Logic Device)及 VHDL(Very High Speed Integrated Circuit Hardware Description Language)硬體描述語言來控制無線傳輸模組。

2.1 簡介

腦電位訊號感測與無線傳輸系統的架構如圖 2.1 所示，包含三大部分：

- (1) **腦電位訊號感測**：主要是以放大器電路及電極所組成，負責腦電位序號的擷取與放大。
- (2) **無線傳輸系統**：由 CPLD 主控，負責控制數位類比轉換(A/D converter)、無線傳輸封包編碼與解碼、訊號格式轉換等與無線傳輸相關的工作。
- (3) **回授訊號提醒裝置**：由 CPLD 主控，當嵌入式運算系統運算出結果後，判斷有發出回授提醒訊號的必要性時，會將提醒訊號利用無線傳輸回傳至 CPLD，CPLD 做完解碼的動作後，會啟動回授提醒裝置。

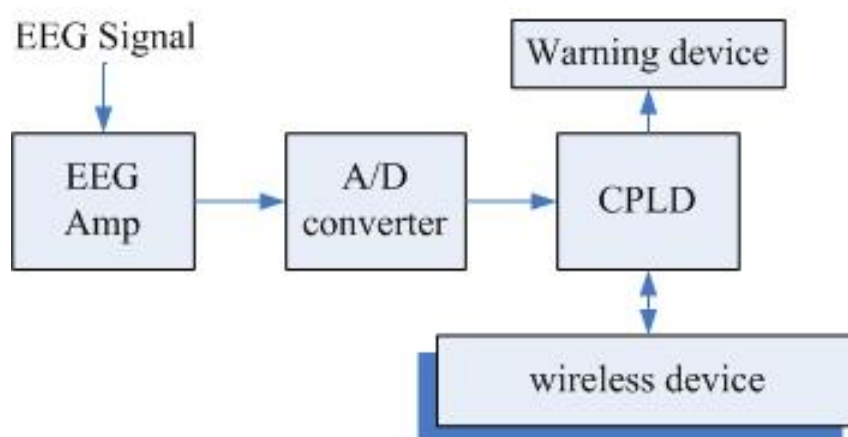


圖2.1、腦電位訊號感測與無線傳輸系統架構圖

2.2 腦電位訊號感測

本節首先簡介腦電位訊號的特徵與量測方式，接著，介紹我們所實現的腦電位訊號量測電路的設計、模擬與實現特色以及一些詳細的電路規格。

2.2.1 腦電位訊號簡介

大腦皮層有大量的神經元，這些神經元的活動有某種程度的規律性，因此置於頭皮上的成對電極可用來記錄自大腦皮層上產生的電位變化，此電位變化是由有節奏的電訊號（Electrical Rhythms）與暫態放電(Transient Discharge)所組成，這些變化波形就是所謂的腦電位訊號，不同的 EEG 測量位置、頻率範圍、振幅大小、訊號波形、週期性和產生方式可以區分不同的 EEG。

EEG 的紀錄有其困難所在，主要在於 EEG 的波形很微小，腦波訊號經過腦膜、腦脊髓液、腦殼到達頭皮時，其振幅峰對峰值大約只有 $1\sim 100\mu\text{V}$ ，頻率大約在 $0.1\sim 100\text{Hz}$ 。其次，不同材質的電極與電極跟頭皮的接觸程度都會影響量測記錄的結果，實驗操作時容易受到雜訊干擾。根據不同頻率範圍，大致上將 EEG 分為五類，如表 2.1 所示，以下針對 Alpha(α)、Beta(β)、Theta(θ)、Delta(δ) 等波形做介紹：

(1) Alpha(α)：

頻率為 $8\sim 12\text{Hz}$ ，一般正常人在處於安靜、身體放鬆、大腦休息的狀態下，會出現這樣的週期波，是「意識與潛意識層面」之間的橋樑，其電位約為 $50\mu\text{V}$ ，在腦枕葉部(occipital region)及頂葉部(parietal region)最明顯。

(2) Beta(β)：

頻率為 12Hz 以上，但一般很少高於 50Hz 。清醒及警覺時尤其明顯，也是邏輯思考、計算、推理時需要的波，屬於「意識層面」的波。在頂葉部及額葉部(frontal region)會較明顯，電位約 $20\mu\text{V}$ 。

(3) Theta(θ)：

頻率介於 $4\sim 8\text{Hz}$ ，主要在兒童的頂葉部及顳葉部(temporal region)會出現，在深睡作夢、深度冥想時特別明顯，屬於「潛意識層面」的波，許多的腦疾病患者，可以找到 θ 波。

(4) Delta(δ) :

頻率低於4Hz 的腦波頻段，在一歲大嬰兒睡眠時，以及有嚴重器官性疾病的患者身上尤其明顯，主要出現在兒童的枕葉部，及成人的額葉部。

表 2.1、腦電波分類表

腦波	頻帶範圍	說明
β 波	12-30Hz	平常活動(緊張及激怒時 25-30 赫)
α 波	8-12 Hz	放鬆的身心
θ 波	4-8 Hz	靜坐進入深度冥想
δ 波	4Hz 以下	熟睡

在臨床實驗上，EEG 訊號會依據電極擺設的位置而影響其振幅、相位與頻率。一般最常用的架構是由 International Federation of EEG Societies 所建立的 10-20 EEG 電極擺設系統[29]。如圖 2.2 所示其位置是以枕骨突出部分和鼻樑為界線將頭部分成左右兩塊，再以雙耳上緣為界將頭部分為前後兩塊。記錄方式基本上可分為單極式、平均式、與雙極式，本論文是採用單極式記錄。

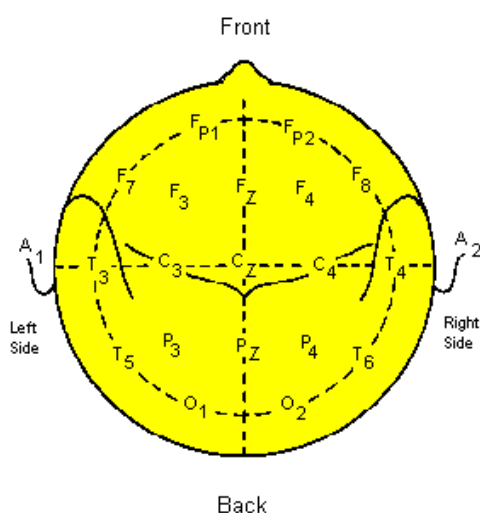


圖2.2、10-20 系統電極擺設位置圖

2.2.2 腦電位訊號量測電路設計

腦波量測電路方塊圖如圖 2.3 所示，先藉由電極將 EEG 訊號量出再做後續放大處理。在生理訊號量測上，我們採用單極式量測，所實現的電路可分為五個部份：(1)前置放大電路、(2)隔離電路、(3)帶通濾波電路、(4)增益放大器、(5)帶拒濾波電路[29][30]。

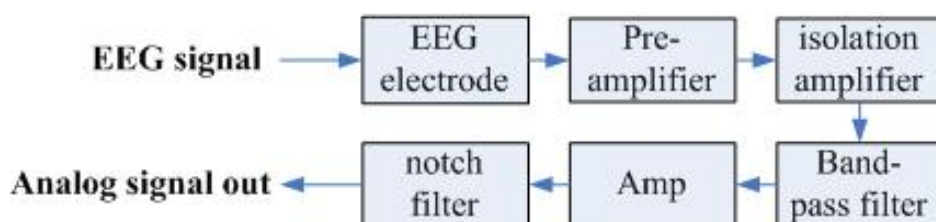


圖2.3、腦波量測電路方塊圖

(1) 前置放大電路

這一級主要為放大作用，由儀表放大器 AD620 為主體所構成，AD620 為一低功率，有高準確度的放大器(Amplifier)，而其增益(Gain)範圍可從 1 到 10000，電路圖如圖 2.4 所示。

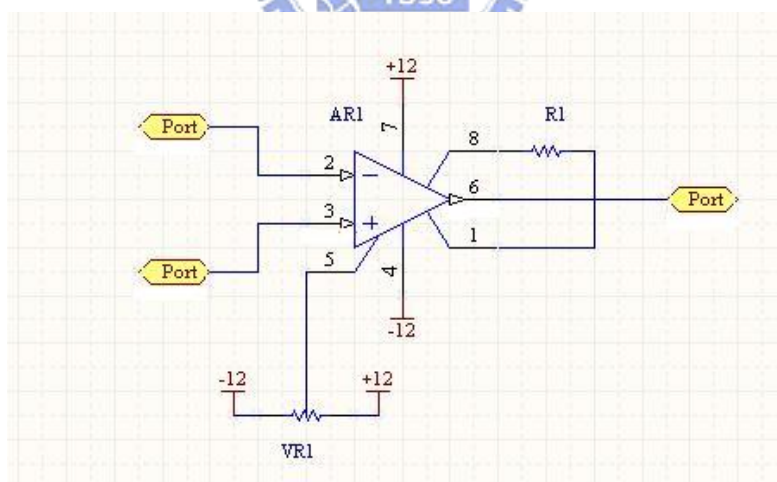


圖2.4、前置放大器電路圖

其增益設計如式子(2.1)所示，

$$A_v = \frac{49.5k\Omega}{R_1} \quad (2.1)$$

由圖 2.4 中，得知 $R_1 = 500\Omega$ ，代入上式中，可得增益 $A_v = 100$ ，藉由模擬，我們輸入一個 1mV 的弦波，看其輸出的狀況：圖 2.5 其中淡藍色為輸入波，其一格為 1mV ；紅色為輸出波，其一格為 100mV 。另外，改變了其輸入源的頻率，結果從低頻到 10kHz 間，皆沒有變化。因此，我們可以得知，這一級僅做放大的動作。在圖中，輸出訊號與輸入訊號有一個反相的關係，這是儀表放大器一般的特性。

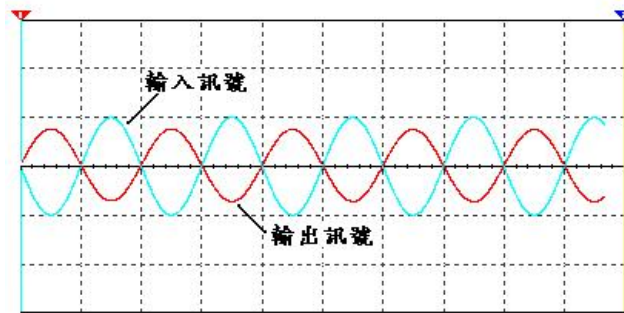


圖2.5、前置放大器訊號模擬圖

(水平軸為時間；垂直軸為電壓大小)

(2) 隔離電路

在前置放大電路與帶通濾波電路之間，有一個訊號隔離器，主要是將一、二級的電源隔離開，以避免受測者因為電源漏電而有所危險。訊號隔離器採用 ISO122JP，為光學式隔離，如圖 2.6 所示。

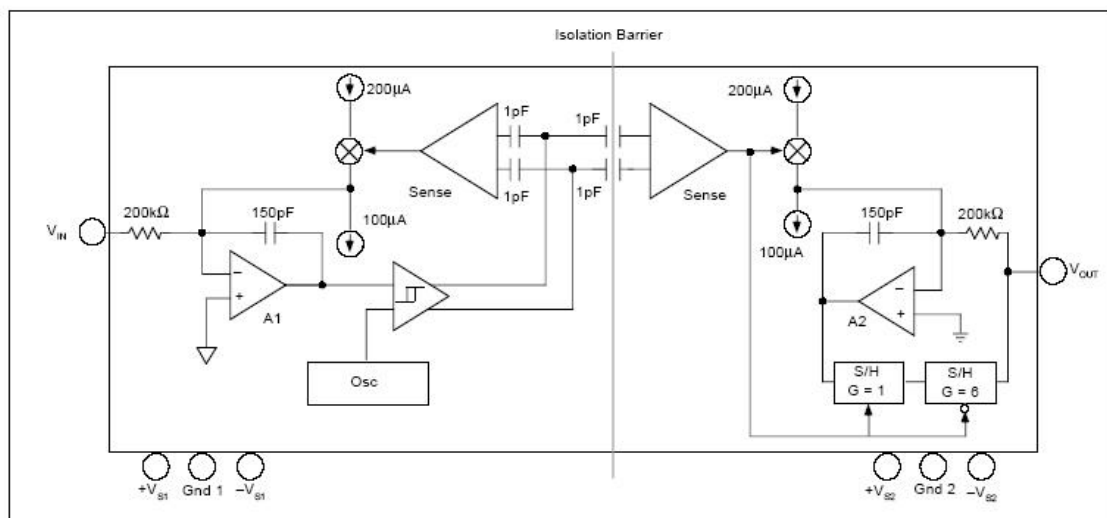


圖2.6、訊號隔離器電路圖

(3) 帶通濾波電路

由於在受測期間，接收到的腦波可能會存在著不可避免的雜訊，因此必須有濾波器的存在。腦波的頻寬大小大約從 Delta 波的 0.5~4Hz 到 Gamma 波的 22~100Hz，因此希望其濾波器的頻帶位於 1 到 100Hz 之間。在帶通濾波電路的設計上，我們使用一組高通濾波器以及一組低通濾波器所組成。如圖 2.7 所示，為主動式二階高通濾波，轉折頻率為 1Hz。其頻率設計由 C1、C2、R2 和 R3 所組成，如式子(2.2)。

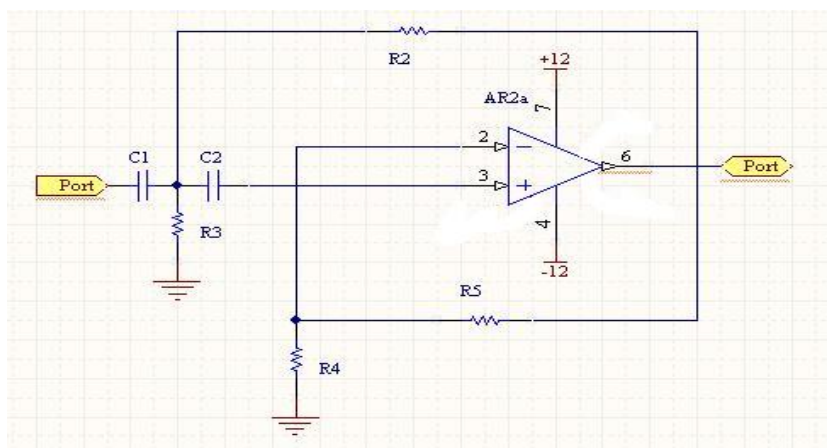


圖2.7、高通濾波器電路圖

$$f_H = \frac{1}{2\pi\sqrt{R_2 R_3 C_1 C_2}} \quad (2.2)$$

其極點設計如式子(2.3)

$$\frac{(R_4 + R_5)}{R_4} = 1.52 \quad (2.3)$$

而低通濾波器的部分是採用主動式二階低通濾波器(如圖 2.8 所示)，轉折頻率為 100Hz。其頻率設計由 R7、R8、C3、C4 決定，如式子(2.4)

$$f_L = \frac{1}{2\pi\sqrt{R_7 R_8 C_3 C_4}} \quad (2.4)$$

其極點設計如公式(2.5)

$$\frac{(R_9 + R_{10})}{R_9} = 1.52 \quad (2.5)$$

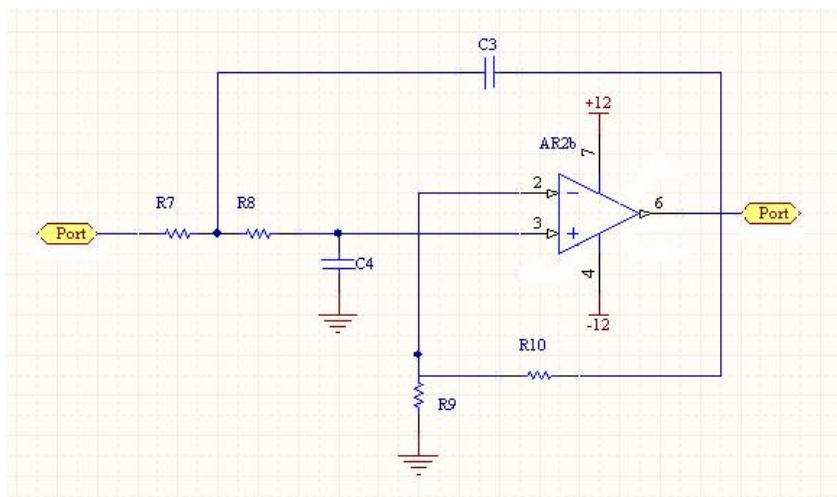


圖2.8、低通濾波器電路圖

依照電路圖，我們利用模擬程式去模擬，記錄下其各個的振幅。表 2.2 即為模擬的結果：

表 2.2、帶通濾波器輸入輸出表

頻率	輸入振幅(V_{PP})	輸出振幅(V_{PP})	dB 值
1Hz	2V	1.997V	-0.01
2Hz	2V	1.999V	-0.004
5Hz	2V	2V	0
10Hz	2V	1.997V	-0.01
20Hz	2V	1.984V	-0.06
50Hz	2V	1.817V	-0.83
100Hz	2V	0.940V	-6.56
200Hz	2V	0.117V	-24.67
500Hz	2V	0.003V	-55.7
1kHz	2V	0	

利用上表，我們用 Excel 畫出其頻率響應圖，如圖 2.9 所示

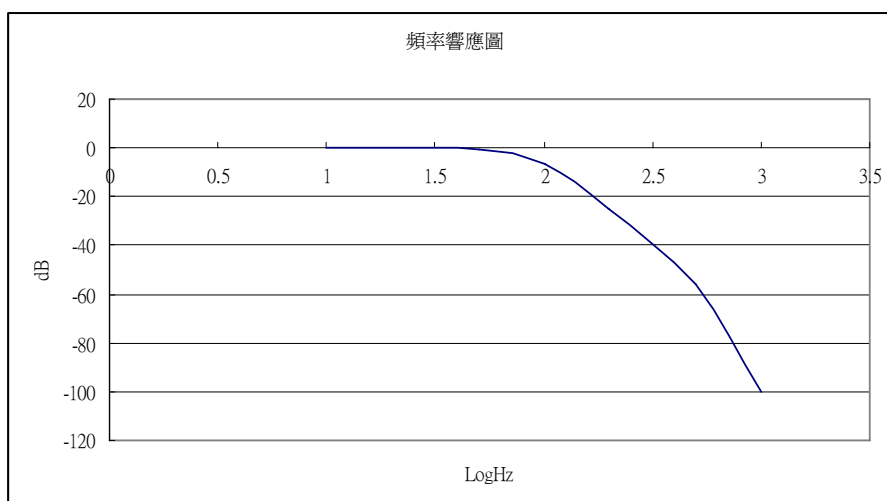


圖2.9、帶通濾波器頻率響應圖

因此，我們得知其 3dB 點大約在 80Hz 左右，然而最重要的腦波為 Alpha 以及 Beta 波，而其頻率約於 8Hz~22Hz，因此這部份可完整地通過濾波器。

(4) 增益放大器

此級主要也為放大的功能。在此級の後端，有一個 switch，其主要功能是切換此放大器的放大倍率。所用的 IC 為 TL082，為基本的非反向放大器，其增益調整如式子(2.6)，電路如圖 2.10 所示。

$$A_v = \frac{(R_6 + VR_2)}{R_6} \quad (2.6)$$

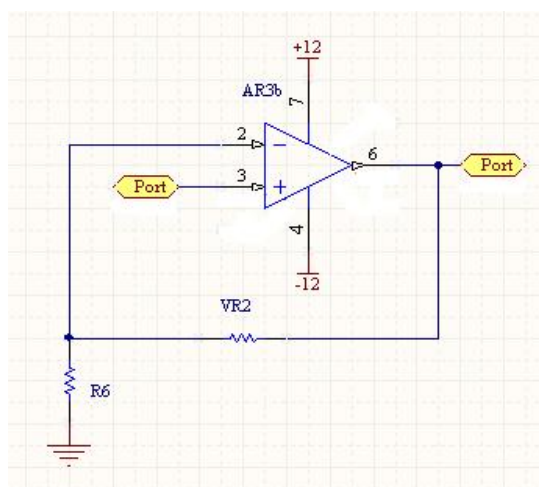


圖2.10、增益放大器電路圖

圖 2.11 即為此級電路的模擬結果，其中綠色為輸入波，其一格為 5mV；紅色為輸出波，其一格為 50mV。改變其輸入波的頻率，我們發現頻率的改變並不會影響輸出的振幅，因此可確保此級的頻率響應不造成任何的問題。

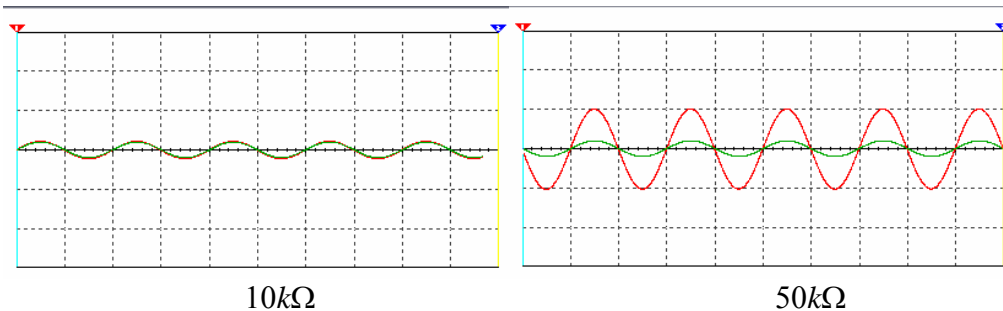


圖2.11、模擬結果(水平軸為時間，垂直軸為電壓大小)

(5) 帶拒濾波器

為避免 60Hz 室內配電干擾可能造成的訊號失真問題，加入帶拒濾波器，讓干擾降到最低。此級的功能為帶拒濾波器，圖 2.12 為模擬結果的波形圖，圖中淺藍色為輸入訊號，而深藍色則為輸出訊號，圖 2.13 則為頻率響應圖。依照電路圖，我們利用模擬程式去模擬，記錄下其各個的振幅，表 2.3 即為模擬的結果。

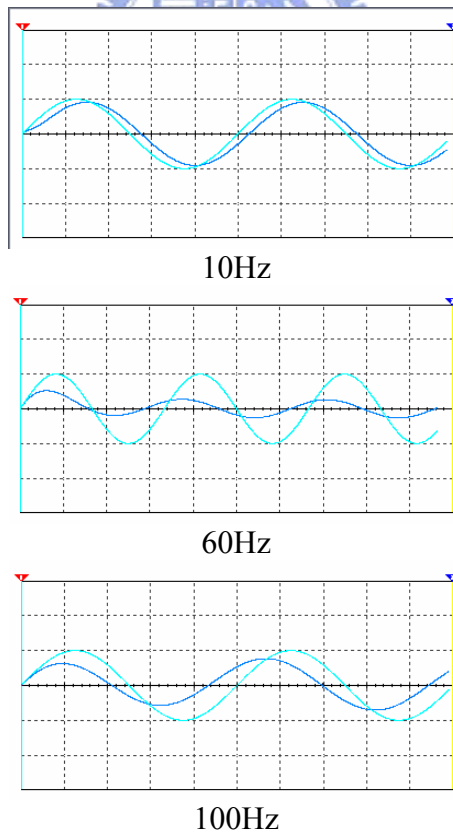


圖2.12、各頻率模擬圖(水平軸為時間，垂直軸為電壓大小)

表 2.3、帶拒濾波器輸入輸出表

頻率	輸入振幅(V_{PP})	輸出振幅(V_{PP})	dB 值
10	20V	18.196V	-0.821
20	20V	14.335V	-2.892
30	20V	10.061V	-5.968
40	20V	5.905V	-10.596
50	20V	3.349V	-15.522
60	20V	5.227V	-11.656
70	20V	8.306V	-7.633
80	20V	10.921V	-5.255
90	20V	12.922V	-3.794
100	20V	14.408V	-2.848
200	20V	18.856V	-0.512
300	20V	19.522V	-0.210
400	20V	19.737V	-0.115
500	20V	19.831V	-0.074
1000	20V	19.958V	-0.018

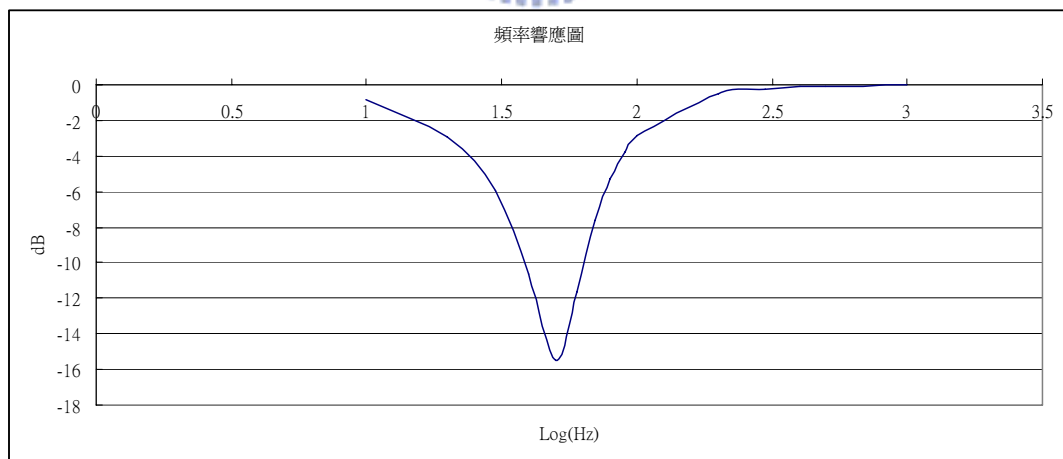


圖2.13、帶拒濾波頻率響應圖

根據圖 2.12、2.13 所示，我們發現 1~100Hz 之間的輸入波只有在低頻以及高頻時，其輸入與輸出相同，只有在 60Hz 會有比較明顯削減的現象。所以，可以證明此級有助於率掉電源交流頻率 60Hz 的雜訊。

2.3 無線傳輸系統

在無線訊號傳輸模組的選擇上，由於考慮到傳輸功率對受測者的影響，本論文選擇藍芽(Bluetooth)無線傳輸以及一種低功率的無線射頻(Radio Frequency,RF)模組來作為無線傳輸的介面，可以根據所需的傳輸距離作選擇。在實作上，本論文以 CPLD 將 A/D(Analog to Digital)轉換器所轉出來的數位訊號包裝成無線傳輸模組所需的封包格式，傳至後端由嵌入式系統做解碼與分析的動作，架構圖如圖 2.14 所示。

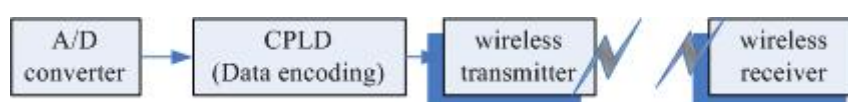


圖2.14、無線訊號傳輸架構圖

2.3.1 藍芽無線傳輸簡介

藍芽是一種保密性高、低功率、低成本和抗雜訊能力高的無線通訊技術，已被廣泛討論與應用[32]。醫療設備無線化需要低干擾、抗雜訊能力高及保密性佳的無線通訊設備，而藍芽恰好符合這些需要。藍芽為短距離的無線區域網路傳輸標準，藍芽技術運作的原理主要是運用跳頻展頻技術(Frequency Hopping Spread Spectrum, FHSS)方式。藍芽技術不但傳輸量每秒鐘可達 1MB，同時可以設定加密保護，每分鐘變換頻率一千六百次，因而很難截收，也不受電磁波干擾。如表 2.4 所示，藍芽系統的基本架構可分為四個層次類別[33]。本論文所採用的藍芽無線傳輸模組如圖 2.15 所示，所使用的為 HCI(Host Controller Interface，主機控制介面)這一層，而控制介面為 RS232。

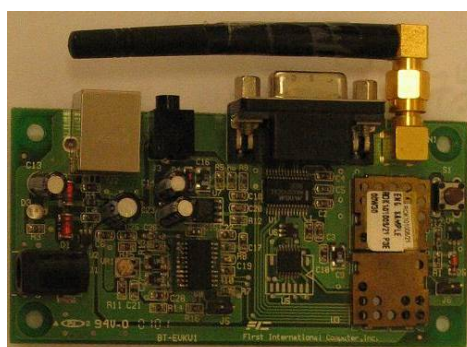


圖2.15、藍芽無線傳輸模組實體圖

表 2.4、藍芽無線通訊協定[33]

通訊協定層別	說明
Link Manager & L2CAP	底層通訊協定的功能，Link Manager 負責 Baseband 連結的設定，釋放及管理；L2CAP(Link Layer Control & Adaptation Protocol)負責包割與組裝、通訊協定多工處理、服務品質(Quality of Service ,QoS)等服務功能。
Radio & Baseband	硬體模組的設計，具有射頻處理和基頻調變的功能。其中，Radio 射頻模組(RF Module)負責頻率合成、資料轉換和雜訊過濾；Baseband(基頻)處理則負責訊息編碼及加解密編碼、跳頻以及封包處理等。
Application Framework & Support	可依據語音、數據、控制等不同之需求，提供應用軟體所需的通訊協定功能與相對應之應用程式介面。在此層面下又包括有 RF Comm、SDP(Service Discovery Protocol)、TCS (Telephone Control Service)等三種架構，分屬不同的技術任務。
Host Controller Interface (HCI)	負責界定藍芽與 Host 設備間之連結界面的控制指令，在 HCI 這部份一般可以分為 USB(Universal Serial Bus，通用串列匯流排)、RS232、UART(Universal Asynchronous Receiver Transmitter，通用異步收發器)等三種介面。

2.2.2 RF3100/3105 簡介

在長距離的無線傳輸模組方面，我們選擇研廣科技無線模組 RF3100/3105 來做為傳輸的介面。我們選擇這個模組的主要原因在於其耗電量低、傳輸距離長以及其模組體積小的優點。其控制模式是以 RS232 的方式來控制，所以在模組切換跟使用上也比較有利於開發與使用。另外，由於藍芽無線傳輸上需要許多協定上的編解碼，因此，在使用上反而消耗一些額外的傳輸，而造成使用上比較麻煩。而在 RF3100/3105 則是用 RS232 即可控制，而在安全性上，RF3100/3105 的模組上有控制器做編解碼的動作。最重要的一點就在於其適用於長距離的傳輸，其傳輸的距離可達 200~600 公尺。RF3100/3105 有三種頻率可供選擇：433~435Hz、868~870Hz、902~928Hz，我們所使用的頻帶是 902~928Hz[34][35]。



圖2.16、RF3100無線傳輸模組實體圖



圖2.17、RF3105無線傳輸模組實體圖

2.2.3 無線通訊系統比較

在藍芽技術中，無線電的發射功率為 0dBm (1mW)，傳輸範圍約為 10 公尺。傳輸功率的範圍為 1mW 到 100mW，100mW 的傳輸功率可到達 100 公尺的傳輸距離，但是要達到 100mW 功率的傳輸時，則須在射頻前再加上一個射頻放大器 (Power Amp) 裝置。所以，藍芽無線傳輸技術並不適用於長距離的傳輸，而 RF3100/3105 的傳輸距離則可達 200~600 公尺，可以解決傳輸距離的問題。根據表 2.5 所示，藍芽無線模組的傳輸功率最低僅有 0dBm，而 RF3100 的傳輸功率為 12dBm 遠比手機 GSM(Global System for Mobile Communication，全球行動通訊系統)1800 的 24dBm 為小。相較之下，RF3100 的傳輸功率小的多。因此，在需要比較遠的傳輸距離(例如：室內)的時候就可以利用 RF3100，而當傳輸距離比較短(例如：汽車內)時，便可利用藍芽無線傳輸模組做為傳輸介面。

表 2.5、無線傳輸格式傳輸功率比較表[32]

傳輸模式	Bluetooth	RF3100	GSM1800
傳輸功率	0dBm(最小值)	12dBm	24dBm(最小)

這兩種無線傳輸模組的詳細比較如表 2.6 所示，藍芽無線的頻帶是無線網路所使用的 2.4GHz；而 RF3100/3105 則是 915MHz。雖然，藍芽無線模組使用的傳輸速率(57600bps)較 RF3100/3105(38400bps)高，但由於其封包較複雜使得其最高取樣頻率反而不如 RF3100/3105。但藍芽無線傳輸模組可以提供全雙工的傳模組，這造成其於雙向傳輸的應用會比半雙工的 RF3100/3105 來的方便。

表 2.6、無線傳輸模組規格表

傳輸模式	Bluetooth	RF 3100/3105
頻帶	2.4GHz	915MHz
傳輸頻率	57600bps	38400bps
傳輸距離	10/100m(1/100mW)	200-600m
取樣頻率(2 channel)	343Hz	914Hz
傳輸方向	全雙工	半雙工
調變方式	FSK**	FSK**
Transmission Power	0/20dBm(1/100mW)	12dBm
連接介面	UART、USB	UART

**：FSK-- Frequency Shift Keying, 頻率鍵移

2.2.4 CPLD 與 VHDL 硬體描述語言

近年來由於大型可程式邏輯閘(如 CPLD、FPGA(Field Programmable Array))的迅速興起，提供使用者可以利用其協助使用者在短時間內開發自己所需的控制器。本論文便是利用 CPLD 與 VHDL 硬體描述語言來開發前端控制無線模組與 A/D converter 的系統。CPLD 的全名是 Complex Programmable Logic Device，是可程式邏輯邏輯元件(Programmable Logic Device, PLD)的一種。可程式邏輯元件是一種可讓使用者組成特定邏輯功能的數位積體電路，並可快速協助邏輯 IC 的設計與驗證工作。CPLD 是由許多個邏輯方塊所組成，各邏輯方塊均類似一個簡

單的 PLD 元件。邏輯方塊間的相互關係則由可編程的連線架構，將整個邏輯電路合成而成[36]。本論文所使用的 CPLD 是由 ALTERA 公司生產的 FLEX 系列的 CPLD，其發展板如圖 2.18 所示。

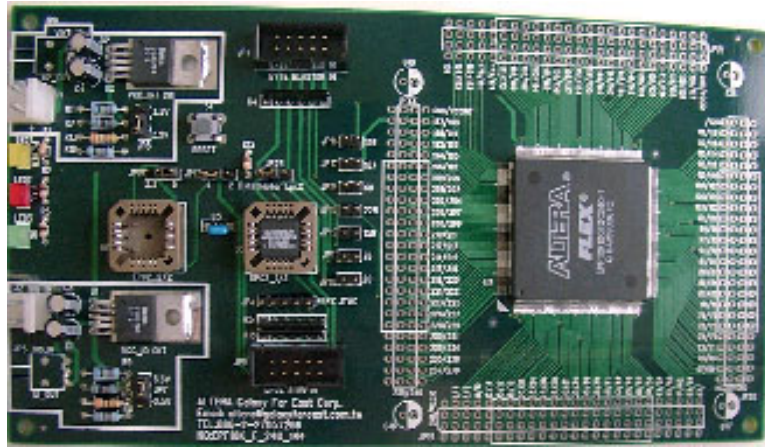


圖2.18、FLEX系列發展版

程式上，我們是以 VHDL 來撰寫我們所設計的系統。VHDL 的全名是 Very High Speed Integrated Circuit Hardware Description Language。VHDL 與 Verilog HDL 一樣是一種描述數位電路的標準語言，是由美國國防部開始發展的，其優點在於功能強大、設計靈活、多種不同的描述風格。而我們所使用的軟體是由 ALTERA 所提供的 Max+plus II，這套軟體提供一套完整的設計工具，包括文字編輯、圖形編輯、模擬、腳位配置、程式燒錄等設計流程一應俱全，其介面如圖 2.19 所示。

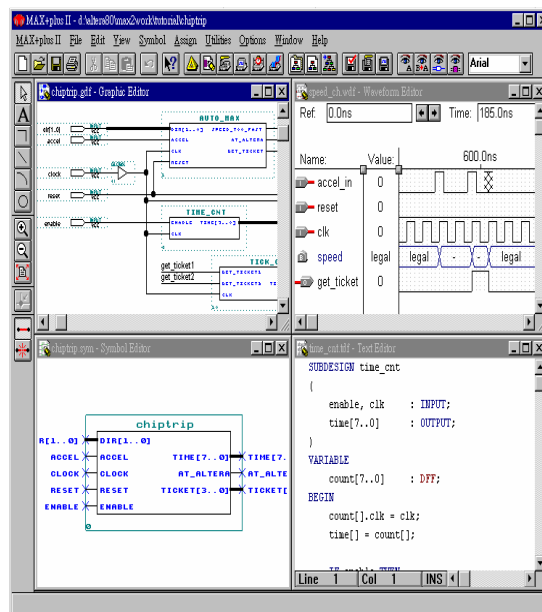


圖2.19、Max+plus II 軟體介面

2.2.5 類比/數位訊號轉換與無線傳輸模組控制程式開發

如圖 2.20 所示，以 CPLD 為中心的無線訊號傳輸部分主要是接收放大電路適當放大後的類比訊號，將其轉換為數位訊號，接者將其轉換為無線傳輸模組所需的訊號格式，再利用無線模組將訊號送出。以下針對使用不同無線模組的實現細節作說明，以下將分別針對如何應用 RF3100/3105 傳輸模組以及藍芽無線傳輸模組來實現無線訊號傳輸的功能作介紹。

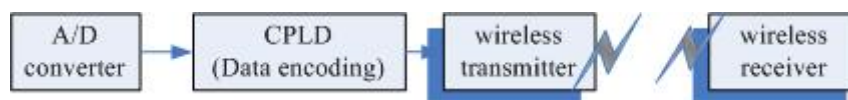


圖 2.20、無線訊號傳輸架構圖

(1) RF3100/3105 傳輸模組

在 RF3100/3105 的控制上，是利用 RS232 界面，所以在 CPLD 上我們必須實現一將訊號轉換成 RS232 封包格式的模組。以兩個頻道為例，我們利用 CPLD 所實現的模組其內部架構如圖 2.21 所示，表 2.7 則為模組接腳表。其中鮑率產生器、UART 傳送模組[37]負責將數位訊號以 RS232 的訊號格式送出，而傳送值產生模組則為本模組的核心部份，其主要負責兩項工作：

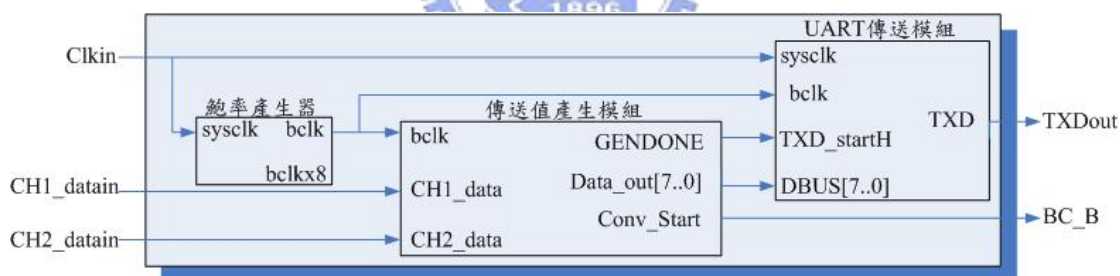


圖 2.21、RF3100 控制模組架構圖

表 2.7、RF3100 控制模組接腳表

信號名稱	屬性	作用極性	功能
Clkin	input		系統之震盪頻率
CH1_datain[7..0]	input		A/D converter 頻道 1 數位資料輸出
CH2_datain[7..0]	input		A/D converter 頻道 2 數位資料輸出
TXDout	output		傳送資料至 RF3100 無線模組
BC_B	output	High	A/D converter 致能訊號

(a) 控制 A/D converter :

控制 A/D converter 以取得其轉換得到的數位訊號。在 A/D converter 方面，我們採用 ANALOG DEVICE 的 AD570，其內部架構圖與控制波形圖分別如圖 2.22 及圖 2.23 所示。

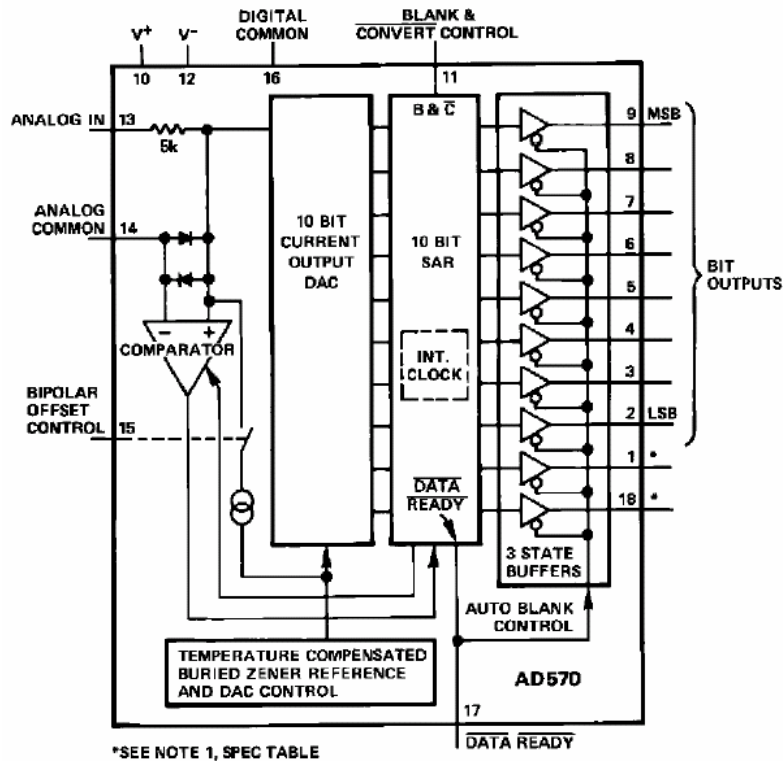


圖2.22、AD570內部架構圖

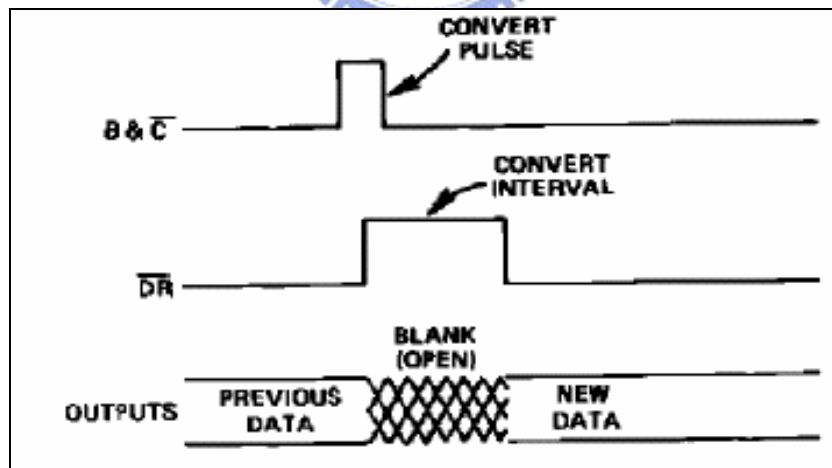


圖2.23、AD570控制時序圖

- (b) 產生送入 RS232 傳送模組的資料：產生送入 RS232 傳送模組的資料並驅動該模組藉以將資料以 RS232 的訊號格式送入無線傳輸模組。此外，本論文所完成的傳輸系統中，同一個傳輸通道必須傳送多個頻道(channel)的資料，所以，除了欲傳送的頻道資訊外，我們

還加上識別碼藉以告知接收端目前收到的訊號是哪一個頻道，封包形式如表 2.8 所示。如圖 2.24 所示，當接收到識別碼後，會隨即讀取五個頻道的資訊，因此，識別碼的數值便不能出現在資料中，而針對這點，我們在傳送值產生模組會對這些值做一些修改，以便傳輸時不會有問題發生。

表 2.8、無線封包

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
識別碼	Data 1	Data 2	Data 3	Data 4	Data 5

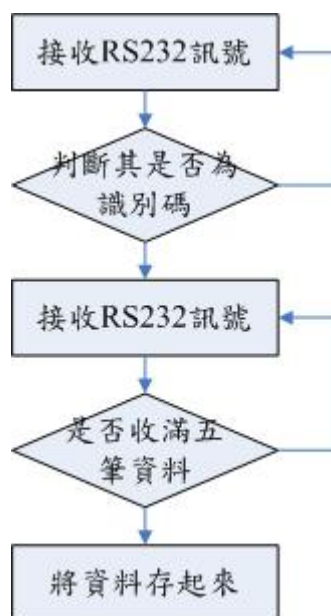


圖2.24、無線訊號接收程式流程圖

為了簡化整個資料傳輸的流程，我們採用單向的資料傳輸，也就是傳送端與接收端不做交握的動作，而是傳送端不斷的傳送資料，而接收端只負責取接收資料。實現上，我們讓傳送值產生模組每隔一段時間去驅動 UART 傳送模組，使其將 RS232 訊號送出，如此一來我們只要控制這一個驅動訊號的頻率便可改變訊號的取樣頻率，例如：使用鮑率為 38400 的 RF3100/3105 模組傳送 5 個頻道的資訊最高可以到達 457Hz 的取樣頻率。為了確保 RS232 的資料順利被送出，所以，驅動訊號的時間間隔必須要大於傳送一筆完整資料所需的時間。

(2) 藍芽無線傳輸

在利用藍芽無線模組做傳輸的動作之前，必須先跟藍芽無線模組建立溝通連線，因此必須先做一連串交握(hand-shake)的動作，將彼此溝通的橋樑建立起來。之後，才開始進行資料傳輸的動作。圖 2.25 為所實現模組的內部結構圖，表 2.9 則為其接腳表。依功能可將此模組區分為三塊：(a)RS232 傳送與接收模組[37]、(b) 傳送資料產生模組以及(c)接收資料判斷模組，以下針對資料傳送模組以及接收資料判斷模組做介紹。

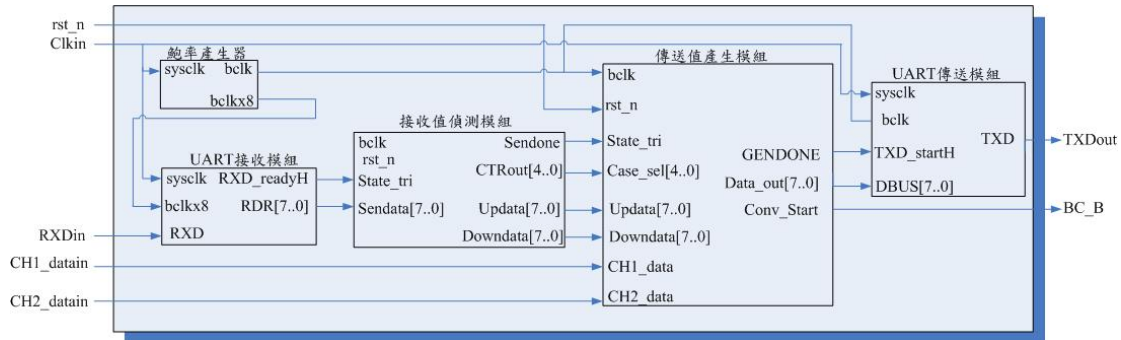


圖2.25、無線藍芽控制模組架構圖

表 2.9、無線藍芽控制模組接腳表

信號名稱	屬性	作用極性	功能
rst_n	Input	Low	重置信號
Clkin	Input		系統之震盪頻率
RXDin	Input		從藍芽模組收取回傳值
CH1_datain[7..0]	Input		A/D converter 頻道 1 數位資料輸出
CH2_datain[7..0]	Input		A/D converter 頻道 2 數位資料輸出
TXDout	Output		傳送資料至藍芽模組
BC_B	Output	High	A/D converter 致能訊號

- (a) **傳送資料產生模組**：由於在交握的過程中會需要多次的資料傳輸，為了節省 FPGA 的硬體空間以及簡化程式的結構，我們建立如圖 2.26 所示的對照表，以查表的方式來決定輸出值，動作流程如圖 2.27 所示。當系統重置時，會啟動這個模組，開始做交握(hand-shake)的動作，當該封包(package)完整送出後，該模組會進入閒置狀態，等待接收資料判斷模組收到正確無誤的回傳值後會再次驅動這個模組，傳送下一個封包。

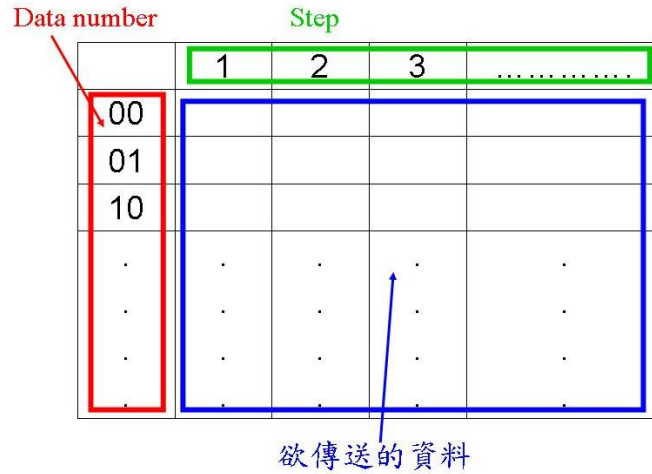


圖2.26、傳輸資料對照表示意圖

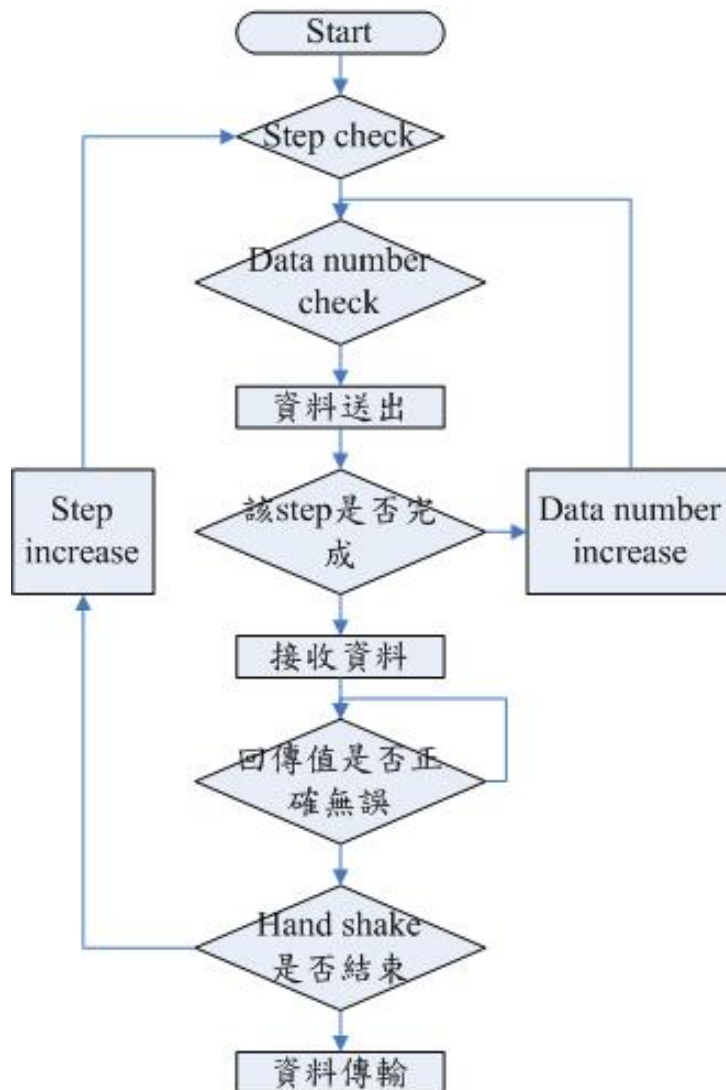


圖2.27、無線藍芽模組交握溝通流程圖

(b) **接收資料判斷模組**：既然是做交握(hand-shake)的動作，自然除了傳送資料給藍芽無線模組外，藍芽無線模組也會有回傳值送出，所以必須要等待其傳出正確的回傳資料後才能再送下一筆值，而這個模組便是在做這樣的工作，必須確定 RS232 接收模組所收到的值正確無誤，才能驅動下一筆資料的傳輸。

在實現上，我們使用查表法的方式來實現傳送資料產生模組以及接收資料判斷模組，由於，利用這種方法在實現上比較快速，而實境使用時速度也比較快，但缺點就在於其所消耗的面積會較大。因此，在嘗試過一些方法後，我們找到面積消耗較小方法以節省 CPLD 的面積。

2.2.6 以半雙工通訊機制實現雙向資料傳輸

訊號傳輸可以分為並列與串列兩種，一般通訊系統都是使用串列資料傳輸。而在串列通訊中，依資料流程的方向，可分為單工、雙工兩種。單工是指其資料傳輸的流向，僅允許單方向傳輸，例如：收音機。而雙工則是允許做資料雙向的傳輸，而雙工又可分為半雙工與全雙工兩種：

- ❖ **全雙工(Full-Duplex)**：當在「全雙工」的資料交換模式下，資料可以在交換資料的雙方以雙向同時傳輸；就像講電話一般，你說話的同時，你的朋友也可以隨時插話而不會影響到你說話的功能。
- ❖ **半雙工(Half-Duplex)**：如果是使用「半雙工」時，則一次僅能有一方傳輸資料，另一方須，等到對方傳送完後才能傳送；就像使用無線電時，必須等對方講完，你才可以講話。

而本系統所使用的 RF3100/3105 模組便是採用半雙工的方式，因此，在雙向傳輸時便需作切換的動作。在不加入回授訊號之前，前端生理訊號量測系統為發射端，將生理訊號即時傳送至後端的嵌入式系統做處理，而後端的無線模組則為接收端，接收無線訊號。而在回授訊號的運用中，傳送與接收的角色正好互換，由嵌入式系統作為傳送端，而前端生理訊號量測系統則為接收端。因此，在角色互換之前，原本的傳送端必須先出傳輸通道等待另一端作傳輸的動作。以下我們以 A 代稱生理訊量測系統，而 B 則為嵌入式系統，如圖 2.28 所示，在單向傳輸時，由於資料僅做單方向的傳輸，所以，在傳輸時可縮短資料與資料間的時間間隔，盡可能的加快傳輸速度。但由於在半雙工中，如要接收資料，則必須先不傳

送資料，所以，我們必須在傳送資料的過程中，保留一些通訊時間，讓另外一端有機會利用這個時間將資料傳送過來。所以，如圖 2.28 所示，我們降低傳送的頻率以獲得一些空檔可讓另一端可利用這個時候送值過來。

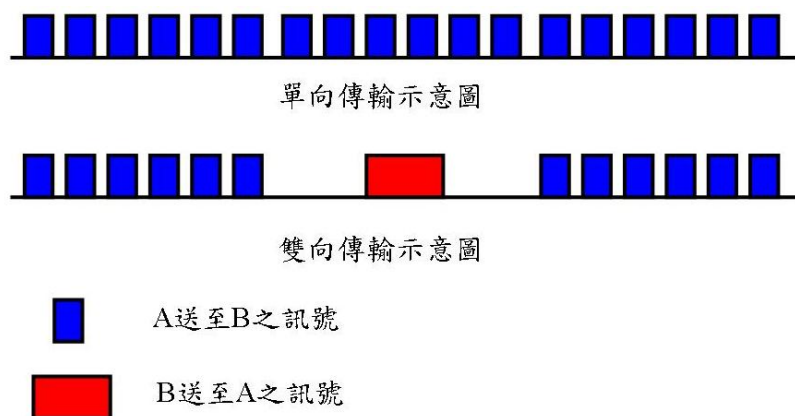


圖2.28、雙向傳輸示意圖

這種作法沒有複雜的握手動作，而其每筆資料所需要的時間也相同，因此，在實現上比較合乎即時(real-time)的要求。在實現上僅對原本的系統做一些修改，並不影響原本的單方向的資料傳輸。由於傳送與接收是在同一個通道上，為了確保回傳值的正確性，我們做了以下幾個工作：

- ❖ 在 A 送至 B 中取適當的傳送頻率，使其有足夠的時間接收資料。
- ❖ 在 A 中加入一些暫存器，以確保所收到資料的合理性。
- ❖ 對傳送的資料做編解碼的動作，以確保其資料的正確性。

第三章 嵌入式多工排程系統設計

雖然，利用嵌入式系統來開發即時無線腦機介面系統可以縮小體積，以及方便攜帶，但為了這些優點將造成在系統開發過程中會遇到許多挑戰：

- ❖ 一般的嵌入式系統所使用的處理器其運算速度勢必比不上電腦，為了加強系統運算的能力，本論文採用具備 ARM 與 DSP 的雙核心處理器 OMAP1510 來增加系統運算能力，但由於這是個雙核心處理器，所以，如何讓這兩個處理器溝通以減少溝通時的時間浪費，便變的非常重要。首先，我們建立一套雙核心處理器的溝通機制以解決溝通的問題；再者，我們建立一套多工排程機制來減少溝通時間的浪費。
- ❖ 在嵌入式系統中所搭載的記憶體往往不像個人電腦中動則幾百 MB(Mega Bytes)甚至上 GB(Giga Bytes)，而儲存的空間也不像電腦的硬碟這麼大，所以，在程式設計的實現上我們會針對這些考量對演算法與程式做適當的修改，以在有限的硬體環境下，能發揮最大的成效。另外，我們利用網路存取的方式來解決資料存取空間不足的問題。
- ❖ 在即時無線腦機介面系統中強調的就是即時，而在處理速度不如電腦這麼快的嵌入式系統中，如何讓系統於運算與周邊溝通上都能保持”即時”，便變的非常困難，因此往往許要花很多時間來做工作的規劃，本論文提出一套嵌入式多工排程系統來減少程式規劃的時間與加強即時訊號接收的能力。

本章分別針對 OMAP1510 的嵌入式系統發展環境、雙核心處理器之內部溝通機制以及嵌入式多工排程系統做一個完整的介紹。

3.1 嵌入式系統開發環境簡介

本論文主要使用 OMAP1510 微處理器為發展嵌入式作業系統的核心，並搭配德州儀器(Texas Instruments, TI)的開發單板 Innovator Development Kit 為實驗平台。本節將對 OMAP1510 微處理器系統做個探討，主要分成二方面加以說明。首先，將介紹 OMAP1510 微處理器的軟硬體架構進行規格介紹。說明硬體的設備有哪些可用，以及如何搭配軟體，進而發展出一套軟硬體相容的系統。另一方

面，在程式開發上，主要有 Linux 平台下的跨平台環境(Cross Compiler)開發與 ARM 微處理器相容的應用程式。

3.1.1 嵌入式硬體開發環境

在硬體環境部分本節將介紹 OMAP 處理器與使用的平台 PSI Innovator; OMAP 為 Open Multimedia Architecture Platform 的縮寫，是一個有雙核心的微處理器，結合 TI-enhanced ARM925 微處理器與 TMS320C55x DSP 核心。PSI Innovator 為 TI OMAP 處理器的手持式可展開的 OMAP 開發平台。OMAP 是德州儀器公司近年來推出一項由 RISC(Reduce Instruction Set Computer，精簡指令集電腦)整合 DSP 的 SoC(System on Chip，系統晶片)嵌入式系統微處理器，其應用定位主要是在於即時的多媒體影音資料處理、語音辨識系統、網際網路通訊及無線通訊等訊號分析應用。OMAP 的最大特點在整合了 TI DSP 核心及 ARM 的 RISC 架構和各種週邊控制器的設計，而這也是本論文利用 OMAP 的原因，我們需要 ARM 對周邊控制以及支援嵌入式作業系統的能力，也需要數位訊號處理器的數學運算能力。

TI 的 OMAP 1510 硬體結構圖，如圖 3.1 所示。從圖 3.1 上可知 OMAP 處理器除了包含 175MHz 的 ARM925 和 200MHz 的 TMS320c55x 外，還整合了多功能的周邊控制元件，例如 LCD(Liquid Crystal Display，液晶平面顯示器)控制器、記憶體擴充介面、紅外線介面、觸控式面版擴充介面及 USB 介面等。由於可以彈性的讓兩個處理器分工，所以它具有高處理效能與較低的耗電量。藉由 ARM 與 DSP 存取相同的記憶體空間(含 SRAM(Static Random-Access Memory，靜態隨機存取記憶體) 與其它記憶媒介)可以讓 OMAP 處理器中的 ARM 與 DSP 資料交互傳遞。

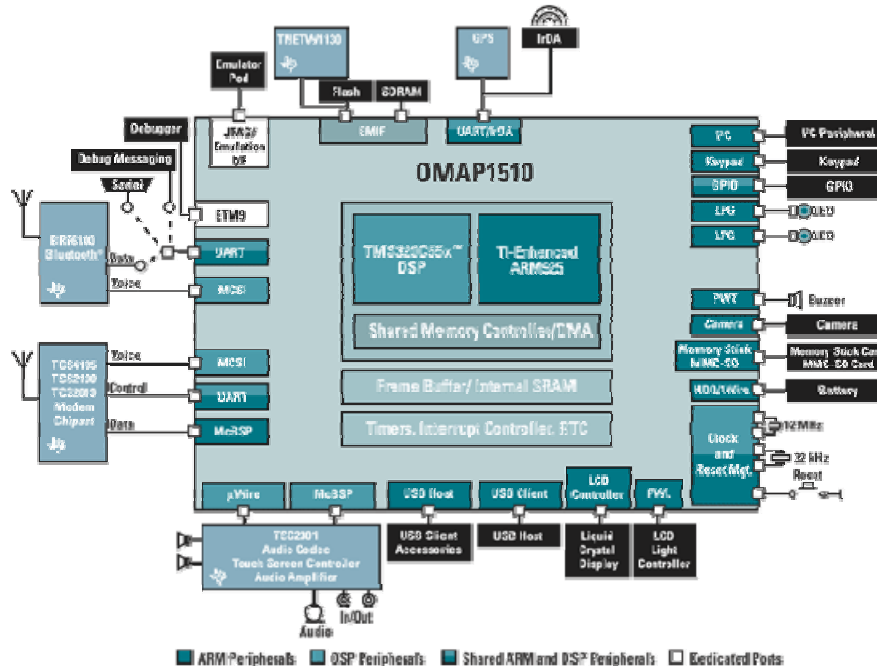


圖3.1、OMAP 1510微處理器架構圖[38]

以下就雙核心架構整合TI TMS320C55x DSP核心和TI925T ARM 核心這兩項不同的技術來作說明。

(1) TMS320C55x DSP Core :

C55x DSP 內含 5 組數據匯流排，在一個周期內允許三次讀取作業和兩次寫入作業。算術邏輯單元有兩組，一為 40-bit 另一為 16-bit。綜上所述，C55x DSP 是高效能，低耗電的處理器，專為多媒體的即時應用而設計。DSP 多用於處理多媒體應用。以下介紹 TMS320C55x 規格：

- ❖ 可達 200MHz(最大處理頻率)，且電壓在 1.5v(一般情況)
- ❖ 一個 cycle 可執行一到二個命令
- ❖ 32K x 16-bit on-chip dual-access RAM (DARAM) (64 KB)
- ❖ 48K x 16-bit on-chip single-access RAM (SARAM) (96 KB)
- ❖ 含有兩個乘累加器(MAC)，每一個 MAC 可以在一個週期內將 17-bit 的乘法完成。
- ❖ 16 KB I-cache, 8 KB D-cache
- ❖ 內建三種影像處理硬體加速器：DCT 、Motion Estimation、 Pixel Interpolation 等硬體加速器

(2) TI925T ARM9TDMI Core :

TI925T 是一顆使用精簡指令集(RISC)的32-bit 處理器，也支援16位元(Thumb模式)指令集，它的核心使用管線(pipeline)的機制讓指令可以連續執行。ARM925用於執行作業系統(Operating System,OS)。以下為ARM925的規格:

- ❖ 可達 175MHz(最大處理頻率)，且電壓在 1.5v(一般情況)
- ❖ 16KB I-cache; 8KB D-cache
- ❖ 192-KB of shared internal SRAM - frame buffer
- ❖ Support for 32-bit and 16-bit (Thumb mode) instruction sets
- ❖ 處理器內部有自己的記憶體管理單元

Innovator Development Kit 是 TI 為 OMAP 推出的可展開之手持式開發平台，如圖 3.2 所示， Innovator Development Kit 是很有彈性的發展與展示平台，不但支援所有主要無線標準，並能協助使用者在常用作業系統下開發各種應用。界面模組包含多項獨特界面功能，可以讓廠商對發展套件進行客製設定；另外還有可選用擴充模組，它們會支援 GSM/GPRS(General Packet Radio Service，通用分組無線服務)、802.11b 和藍芽，一張附加電路板 (breakout board) 則包含其它外部硬體和一組 10Mb 乙太網路連線。OMAP1510 處理器內含 TI 功能強大的 DSP 核心，應用軟體設計工程師只需透過 Innovator Development Kit 電路基板的協助，就能利用高階作業系統發展工具，獲得 DSP 所提供的媒體處理加強功能。



圖3.2、Innovator Development Kit外觀圖

Innovator Delopment Kit 硬體包含三個部分，處理器模組、介面模組和，擴充模組。處理器模組是 OMAP 微處理器的模組部分，除了處理器之外，一些系統記憶體。其中含 32Mbytes 的 SDRAM 供系統程式運作資料運算使用。處理器與外界溝通的 I/O 介面都放在介面模組上，本論文主要用兩個 UART 作為與外界的溝通橋樑。針對 Innovator Development Kit 在發展上的便利性，TI 便開發了 Break Out Board(BOB)的套件，Break Out Board 是可以將 Innvator Development Kit 各個模組攤開平擺在 BOB 上使用，方便於開發時接線的方便性。

Innovator Delopment Kit 主要的特性如下：

- ❖ TI OMAP1510 微處理器
- ❖ 32Mbytes SDRAM、32Mbytes User Flash
- ❖ 4MByte/256KBtyes Boot Flash/RAM
- ❖ 觸控式 LCD 螢幕
- ❖ 四個控制按鈕
- ❖ Dual RS232 的埠



3.1.2 嵌入式軟體開發環境

在前面將 OMAP 微處理器硬體和環境選擇簡單的說明之後，接下來要進入軟體應用的開發環境部分。因為 OMAP 是由 ARM 和 DSP 兩種架構所組成的，所以在軟體開發上，我們分別針對 ARM 和 DSP 的程式開發環境做介紹。

(1) 程式開發環境於ARM架構

自從1991 年Linux 由Linus Torvalds 將第一版Linux 公開在網路上後，這套作業系統已經受到世界普遍的歡迎[39]。現在的Linux 支援相當多的處理器核心，由高階的伺服器、工作站，到常見的個電腦中都有它的蹤跡，甚至是一些手持式裝置例如手機、PDA(Personal Digital Assistant，個人數位助理)等一些現代人常用的嵌入式系統上都看得到Linux 的存在。

所謂的嵌入式系統指的就是為了提高產品的穩定性、可攜性，或者是降低產品成本、增加特定功能等不同的目的，對一般的標準系統進行修改以符合特殊設計上的需求。在硬體方面靠的主要就是被稱為系統單晶片(SoC)的半導體整合技

術，而嵌入式作業系統則是軟體方面的技術核心。Linux 的可靠、高效能、伸縮性、低成本，以及最重要的握有控制權等幾項特點，使得 Linux 在嵌入式作業系統上站在非常有利的地位。

在一般PC的Linux的環境下開發程式，最常使用到的編譯器，莫過於GCC (GNU Compiler Collection)，GCC可以說是 Richard Stallman 所創立的 GNU 計畫中最重要的作品之一，它提供了自由軟體世界高品質的編譯器 (compiler)，實現了我們在自由軟體平台上開發程式的夢想。GCC 一個很大的特色是高度可移植性，目前已知有包括了：x86、arm、ia64、alpha、hppa、m68k、Power PC、mips等超過三十種硬體平台與作業系統可以執行 GCC，此高度可移植性正是GCC 廣為流傳散佈的主要原因[40]。儘管 GCC 是自由軟體計畫開發出來的，但其所編譯出來的程式品質並不輸給商業版的編譯器，甚至所編譯出來程式在特定平台有更好的執行效能。

嵌入式系統開發也需要用到編譯器(Compiler)、連結器(Linker)、組譯器(Assembler)，整合開發環境。以 ARM 而言，在 x86 環境下編譯產生出來的執行檔，並不能於 ARM 環境執行，若要在 x86 環境下開發 ARM 程式則需要建立一個跨平台開發的工具 (cross-platform development tool)。arm-linux-gcc 可將 x86 環境下程式碼針對 ARM 的架構進行交叉編譯 (cross-compiler)，產生出來的執行檔便可以於 ARM 的架構上執行。GCC 編譯流程如圖 3.3 所示。

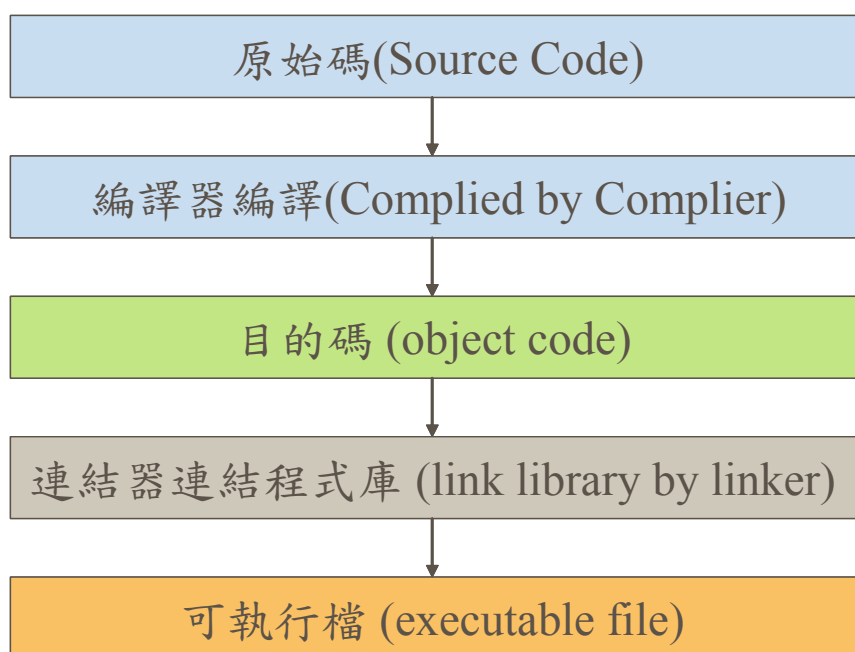


圖3.3、GCC編譯流程圖

(2) 程式開發環境於DSP架構

開發 DSP 程式是採用德州儀器所提供的軟體 CCS (Code Composer Studio)，CCS 提供一個完善的整合發展環境(IDE，Integrated Development Environment)，包含了程式編譯器、程式除錯器以及模擬器。CCS 以一致的環境來整合所有 host 與 target 工具，包括 TI 的 DSP/BIOS kernel、code-generation tools、fast simulators、debugger、與 Real-Time Data Exchange (RTDX) 技術，簡化應用程式的開發。利用這些工具可以簡單地開發出 OMAP 應用程式。

圖 3.4 為 CCS 開發 DSP 程式流程。在設計程式前，先有個設計概念，再開始建立專案來編寫程式碼，編寫過程中如遇到錯誤時需要除錯時，CCS 可以用來對程式編碼做除錯以及即時的分析，甚至整個程式執行過程都可以追蹤。

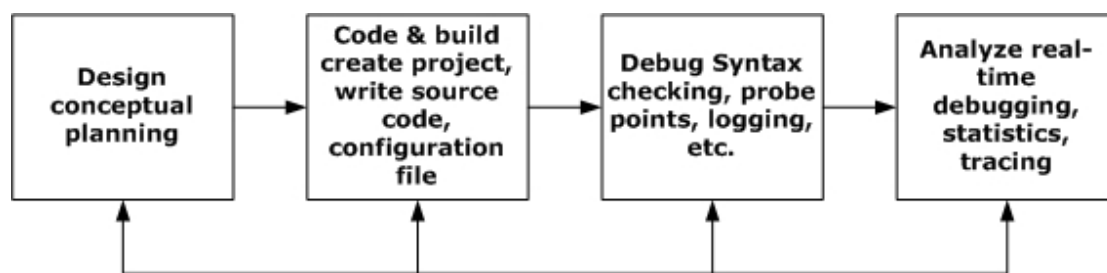


圖3.4、CCS程式發展設計流程[40]

利用 CCS 開發 DSP 程式時，要先做好 ARM 與 DSP 於系統上的配置以及設定 CCS 編程的環境記憶體配置，如此才能跨平台開發程式。而 DSP 的編譯先是由 C/C++的原始碼經過 C/C++的編譯器後，可產生組合語言原始碼。再經過組譯器(Assembler)產生物件(Object)檔，經由連結器(Linker)來產生最後的可執行 COFF(Common Object File Format)檔(以.out 為副檔名)。

DSP/BIOS 核心是一個事件驅動(Event-Driven)的架構，DSP 的應用程式透過 DSP /BIOS Configure 對於 DSP/BIOS 內的機制作設定。DSP/BIOS 系統配置工具是用來幫助開發者在撰寫事件處理函式時，設定事件的特性與參數，如硬體中斷、軟體中斷、背景執行緒、工作 (TASK) 等來使用 DSP/BIOS 的多執行緒能力。

DSP/BIOS 主要被設計出來針對某些應用層面，其中包含：即時排程(Real-Time scheduling)、同步(Synchronization)，以及 HOST 和 TARGET 之間的通訊(如

圖 3.5 所示)。DSP/BIOS 核心提供了(1)強制性的多執行緒(Preemptive Multi-Thread)，也就是在 DSP 中有許多 Task 時，優先權高的工作可以將優先權低的工作中斷，並搶得執行權。(2)硬體抽象化(Hardware Abstraction)，即完全由硬體抽象層控制系統的硬體中斷。使用者可以利用硬體抽象層給予的硬體描述來控制硬體，(3)即時分析(Real-Time Analysis)的功能。除此之外，DSP/BIOS Configuration 工具可將程式的大小作最佳化處理。圖中表示了 DSP/BIOS 在 CCS 上所扮演的角色。

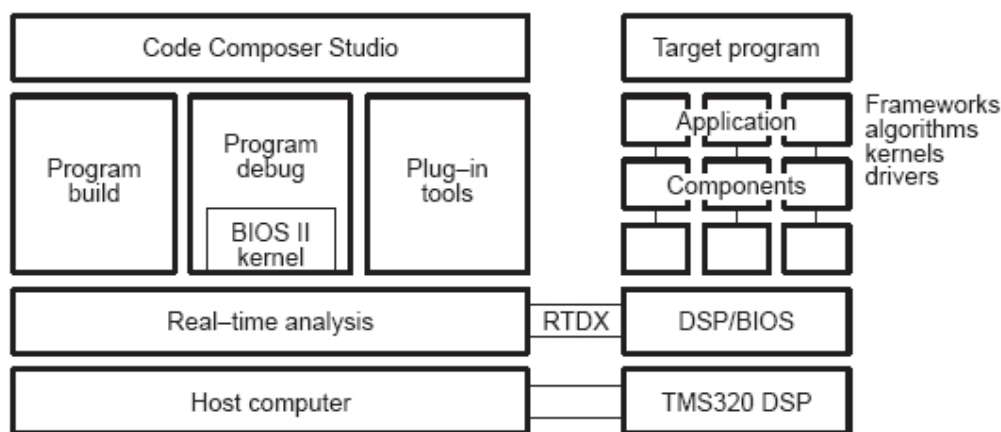


圖3.5、CCS整合發展環境[40]

3.1.3 雙核心處理器之內部溝通機制

OMAP 雖然是一個雙核心元件，但一般 ARM 的開發者有時候並不需要 DSP 核心，他們可能希望 OMAP 開發環境仍然和單核心處理器時相同。我們使用 DSP Gateway 的概念讓 ARM 和 DSP 建立連結，讓元件中 DSP 的存在變得透明，以保持 OMAP 運作時的彈性，讓一般 ARM 的開發者不需要 DSP 核心時，其開發環境仍然和單核心處理器(ARM)時相同。DSP Gateway 可被看作同時存在於兩個核心中的軟體層，其主要用來向 ARM 上執行的多媒體引擎導出一組 API(Application Program Interface，應用程式介面)，以便其存取 DSP 資源。在 DSP 端，DSP Gateway 用於管理 DSP 資源和數據流，並向 DSP 演算法提供介面。在架設 DSP Gateway 之前，由於我們需要可以呼叫與管理 DSP Gateway 的介面，再加上我們在 DSP 與 ARM 上面都有許多工作(task)需要執行，所以，我們分別在 DSP 與 ARM 上面架設即時核心(DSP/BIOS)與嵌入式作業系統(ARM-Linux)

來管理這兩個核心的資源。圖 3.6 表示 DSP Gateway 架構的關係圖。從圖中看出 DSP Gateway 包含兩部份，靠著這兩者合作完成通訊的功能：

- ❖ Linux 裝置驅動程式：提供介面讓 ARM 的程式以系統呼叫(system call)的方式使用 DSP。
- ❖ DSP 的函式庫：提供函式讓 DSP task 可以被 ARM 透過驅動程式使用。

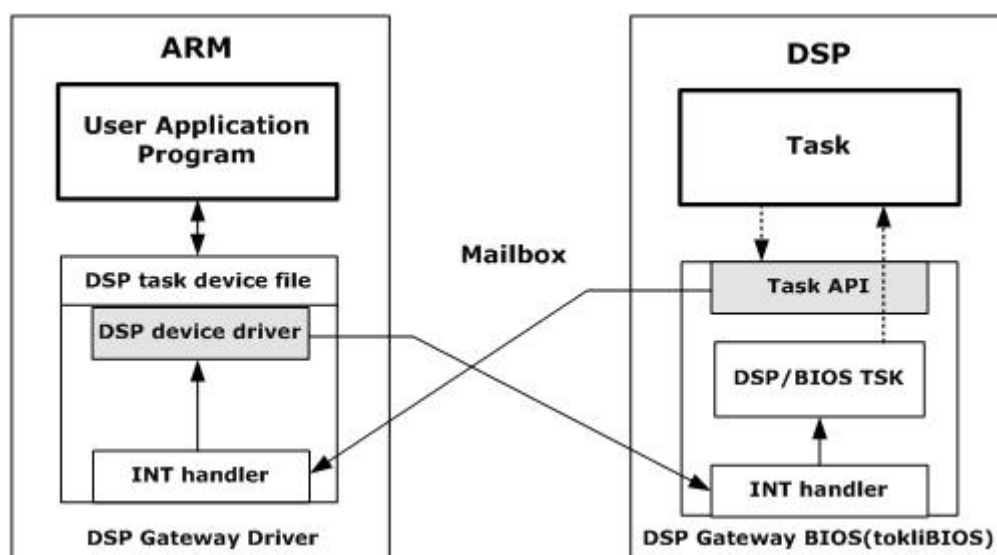


圖3.6、DSP Gateway系統架構圖[41]

在 OMAP 處理器中的 ARM 與 DSP 資料交互傳遞的方法有三種，透過 Mailbox、MPU(Micro Processor Unit) Interface(MPUI)與 Share Memory(共享記憶體)等方式來達成，在 DSP Gateway 中是以 Mailbox 以及 Share Memory 來實現 ARM 與 DSP 資料交互傳遞。以下先對 ARM 與 DSP 資料交互傳遞的三種方法作簡介：

(1) Mailbox 機制：

OMAP1510 中有三組 Mailbox 暫存器，一組是給 ARM 用，當 ARM 有資料寫入 Mailbox 時，會送出中斷(INT5)給 DSP；其他兩組則是當 DSP 有資料寫進時，才會送出中斷(IRQ10/11)給 ARM。每一組 Mailbox 是包含兩個 16 位元暫存器與一個 1 位元旗標暫存器。當中斷發生時，中斷的處理器會用兩個 16 位元的暫存器各自將命令與資料傳到被中斷的處理器上。

(2) MPU Interface(MPUI)機制：

MPUI 可以讓 ARM 和系統的 DMA(Direct Memory Access, 直接記憶體存取)控制器與 DSP 和 DSP 的週邊互相通訊。MPUI 可以存取整個 DSP 的記憶體空

間與 DSP 的週邊匯流排。因此 ARM 與系統的 DMA 可以有 DSP 輸出入空間的完整存取權限。至於透過 Share Memory 這樣的架構則是靠 OMAP 上的 Traffic Controller 達成，藉由 ARM 與 DSP 存取相同的記憶體空間。ARM 可以控制 DSP 的 MMU(Memory Management Unit, 記憶體管理單元)讓 DSP 可以存取這些記憶體媒介。

(3) Share Memory 機制：

由於 mailbox 只能傳送兩個 16 位元暫存器與一個 1 位元旗標暫存器，而無法傳送大量的資料，所以，我們必須借由 Share Memory 的方式來傳送大量的資料。而 Share Memory 這樣的架構則是靠 OMAP 上的 Traffic Controller 達成，藉由 ARM 與 DSP 存取相同的記憶體空間。ARM 可以控制 DSP 的 MMU 讓 DSP 可以存取這些記憶體媒介，在 DSP Gateway 中，我們稱這一塊記憶體為 IPBUF。

圖3.7表示Mailbox和IPBUF在ARM和DSP之間的關係，ARM端的程式要將資料交給DSP task處理時，也要將資料傳給DSP端。資料傳輸的型式，所採取的方式是block send的方式，ARM在傳送資料時的Mailbox command中下達BKSND的命令，並且傳送BID(buffer ID)給DSP。ARM在傳送資料時，會先傳送Mailbox command，這時會有Mailbox中斷(INT5)產生，接收這個命令，經過interrupt handler的處理DSP task收到接收資料的命令，並會到Mailbox command裡面指定的IPBUF內取得所要處理的資料，經過在DSP運算後，將得到結果存在記憶體內。同樣的ARM這時會送接收資料的命令過來，DSP將得到的結果送到IPBUF中，ARM可從IPBUF取得DSP處理後的資料。

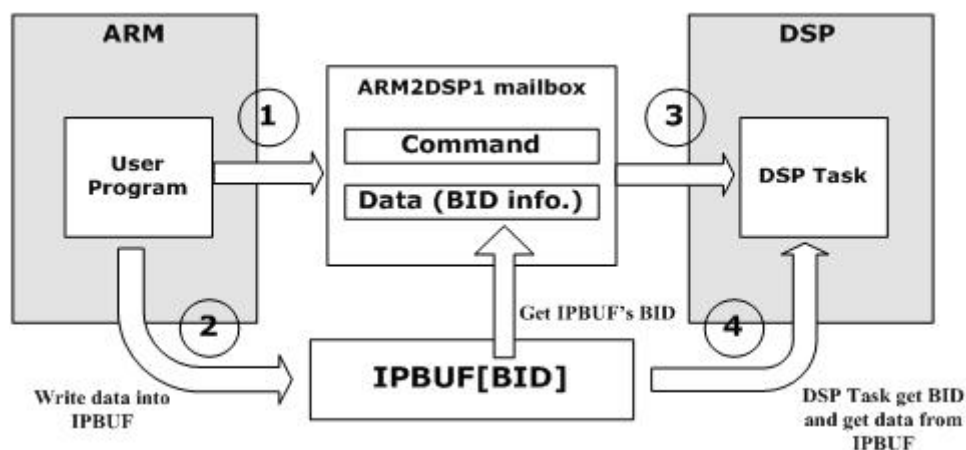


圖3.7、ARM/DSP內部傳輸說明圖

3.2 嵌入式軟體系統架構與簡介

如圖 3.8 所示，OMAP1510 上必須執行許多工作，由於在腦電位訊號的訊號處理上需要比較複雜的計算，所以，本論文將這部份交由 DSP 處理，而 ARM 僅做一般的周邊裝置控制與一些簡單的訊號處理。關於生理訊號的處理運算系統設計與實現，將在第五章有完整的介紹。但除了生理訊號的處理外，在 OMAP 的 ARM 上仍有許多工作需要處理，包括：(1)無線通訊裝置控制、(2) 區域網路資共享以及(3)數位訊號處理器的驅動與管理。因此，我們需要有一個機制管理這些工作，對此，我們發展一套排程管理機制來管理這些工作。

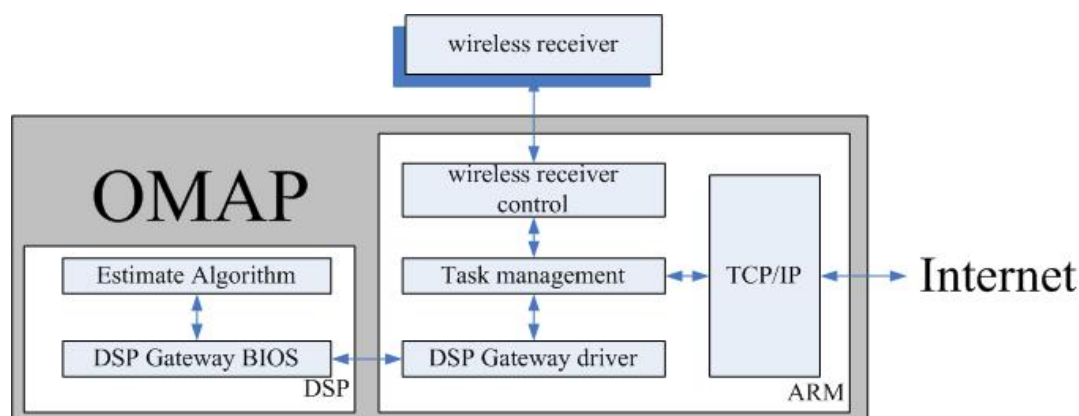


圖3.8、嵌入式系統架構圖

- (1) **無線通訊裝置控制**：本系統中，最重要的就是必須要取得生理訊號，而且必須要即時，因為其必須等時間間隔接收無線訊號，而這也讓我們在軟體實現上面臨一大挑戰，因為我們無法等待數位訊號處理器處理完訊號再繼續接收新的值。也因此，本系統需要多工的機制，我們必須在等待訊號處理器計算的同時，要持續接收無線訊號，以確保資料收集時間間隔的準確性。
- (2) **區域網路資共享**：本系統利用網路讓嵌入式系統上的資訊可以分享出去，並在網路另一端的電腦即時顯示腦電位訊號的資料，以及生理訊號運算後的結果。本論文以 TCP(Transmission Control Protocol, 傳輸控制協定)/IP(Internet Protocol, 網際網路通訊協定)協定以及網路傳輸串流資料的方式來實現，由於這將是將資料分享出去以及與外界溝通的一個重要介面，因此，這項工作在本系統中亦扮演著很重要的角色。
- (3) **數位訊號處理器的驅動與管理**：如同 3.1.3 中所提到的，DSP Gateway 讓數位訊號處理器就如同 ARM 的一個周邊裝置，因此在使用上，其操作方式便如同 UART、USB 等周邊裝置一般，所以，我們可以藉由作業系統來控制與管理數位訊號處理器。

3.3 高效率嵌入式多工排程系統之設計與實現

由於在 ARM 上，有許多工作都必須即時作處理，例如：即時接收無線訊號、即時處理生理訊號。因此，如果要將這些工作由同一個處理程序完成，那麼勢必要在 ARM 會有許多工作要規劃其執行時間，而且也會有一些等待的時間而無法符合即時的要求。因此，本論文提出一套多工排程系統以達到以下目的：

- ❖ 減少時間的浪費，以實現即時運算與資料接收的能力。
- ❖ 將處理續模組化，方便擴充新的處理功能與工作規劃。

在多工排程系統的設計與實現上，可分為幾個步驟：

- (1) **建立工作程序**：首先便是要建立多個工作程序，以便系統可以做排程與切換。
- (2) **建立排程仲裁機制**：在排程系統中，最重要的便是仲裁機制。仲裁機制的存在便是為了決定現在系統必須做哪個工作。由於本論文所實現的嵌入式系統中，便有一個Linux的作業系統，為了善用我們所擁有的資源，一般時候我們讓作業系統去主管排程的動作，而在某些比較特殊的時間點，例如在處理流程中必須強制等待時，我們設計了一個輔助排程的仲裁機制來做仲裁。
- (3) **建立多重處理程序之溝通機制**：這其實是在多工排程中最麻煩的一部分。在我們所實現的系統中，處理程序間會有一定的相關性，例如：需要做資料交換、處理程序同步的工作。本論文針對這項需求提出一套多工排程機制來負責這些工作間同步與通訊的工作。

3.3.1 工作切割與基本排程架構

整體而言，ARM所需要掌管的處理程序一共有四項：(1)無線通訊裝置控制、(2)區域網路資共享、(3)數位訊號處理器的驅動與管理以及(4)訊號後處理等工作。根據工作執行的頻率，我們將所有的工作除了主程式以外區分為三大項：(1)無線模組資料交換與區域網路資共享、(2) 呼叫DSP task運算並傳送資料給IPBUF以及(3)從IPBUF接收資料並做後續的資料處理與分析。由於無線訊號接收以及網路資料分享的頻率相同而且會有很高的相依性，當其分開時會消耗一些不必要的時間與控制，所以將其結合為一個工作。如圖3.9所示，相對於無線訊號接收以及網路資料分享的頻率來說，IPBUF接收資料與傳送資料發生的頻率就小很多，因此，我們讓系統不斷的從無線模組收資料並輸出結果至網路上，而讓作業系統自行判斷將其他工作放入。當無線模組收滿足夠資料時，會啟動仲裁機制來做決定是否可繼續接收無線模組的資料。

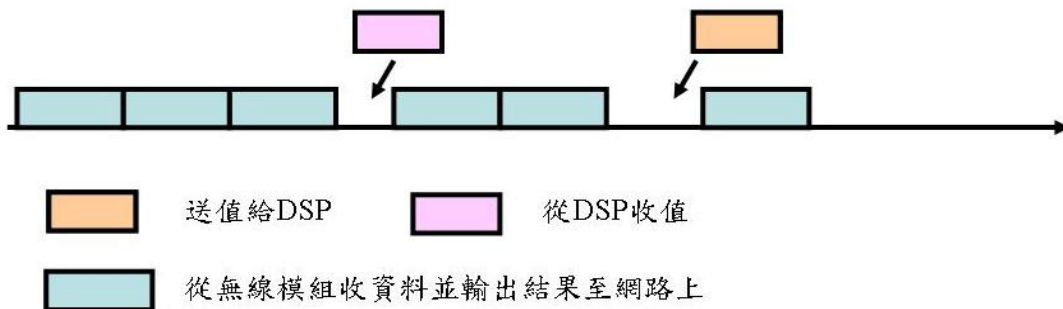


圖3.9、嵌入式系統排程示意圖

如此一來，在 DSP 做訊號處理時，ARM 並不會只是在等待，而同時會持續接收無線訊號以及將資料傳利用送至顯示介面的電腦上。因此，可以滿足即時資料接收與即時訊號處理的需求。

3.3.2 多重處理程序之溝通機制

在多重處理程序的溝通機制中，最重要的目的有二：(1)讓多個處理程序可以做資料交換與資源共享、(2)讓處理程序之間可以同步。在Linux的作業系統中一共提供三種處理程序之間的溝通機制(inter-process communication, IPC) [42]:(1) 訊息佇列(message queue)、(2)信號(semaphore)以及(3)共享記憶體(shared memory)。分別簡述如下：

(1) 訊息佇列(message queue)：

訊息佇列可以視為一些訊息的組合，這些訊息都位於核心程式中，彼此以鏈結串列(linked list)的方式連結，而每個訊息佇列都要透過辨識元來辨識。佇列裡的每筆訊息都有一個型別為長整數(long int)的欄位，一個非負值的訊息長度，和一個真正的資料長度(這個是跟訊息長度一致)，當要傳送訊息到一個佇列時，必須把這些值指定給msgsnd函式。若要從訊息佇列裡讀取訊息則需呼叫msgrcv函式，在讀取時不一定需要使用先進先出的順序來讀取這些資料，也可以根據這些訊息的種類來讀取。在實際上使用發現這個機制並不會比其他的IPC快，所以，這種IPC不夠實用。

(2) 信號(semaphore)：

信號與一般的IPC機制不太一樣，實際上，信號是一個計數器，它用來讓數個處理程序可以共同讀取一個讓大家分享的資料物件。其使用方法如下：

- (a) 先測試控制這個資源的信號。
- (b) 如果這個信號值為正，那處理程序就可以使用這個資源。當程序在使用此資源時，它必須將這個信號的值減一，表示它正在使用這個資源的一個單位。
- (c) 如果測試出的信號值為0，那麼這個處理程序就必須要設為睡眠狀態，直到信號的值大於0為止。當這個處理程序醒來時，它必須再從第一步開始。

如果有一個處理已經用完由信號所控制的資源時，信號的值就會被加1。如果有任何處理程序是因為等待這個信號的值而被設為睡眠，這時他們就會被喚醒。信號其實是一種同步化的機制，而非真正的IPC，其用途有二：

- ❖ 獨立使用時，可用以讓處理程序同步。
- ❖ 與其他的IPC機制搭配使用，可對某些共用的資源進行保護。

(3) 共享記憶體(shared memory)：

共享記憶體可以讓兩個以上的處理程序來分享同一塊記憶體，這是一種速度最快的IPC，因為他不需要伺服器端和用戶端間相互拷貝資料，在使用共享記憶體時要注意一件事：如何在數處理程序使用同一塊共享記憶體時能夠讓它們之間的存取動作同步(synchronize)，通常是以信號來協調各個處理程序讀取共享記憶體的作業。

在實際實現上，考慮到實際上執行的效率、時間以及方便性，本論文採用了三種方式高效率的通訊機制：(a)取代同步機制之溝通機制、(b)仲裁機制、(c)共享記憶體。在本論文所使用的IPC機制中，只有共享記憶體是傳統Linux所提供的。以下針對這三種通訊方式做介紹：

(a) 取代同步機制之溝通機制：

為了保護被共用的資源，所以，不允許同時有兩個以上的處理程序對共用的資源做處理，以避免處理的過程發生錯誤。所以，有些處理程序必須等待另一個處理程序先使用完這塊記憶體，之後才可以使用。而同步機制便可以做到這些功能，Linux所提供的信號就是一種同步機制。在使用信號時，如果需要對同一記憶體做處理時，就必須先拴鎖(lock)住該記憶體藉以保護該記憶體並確保其讀寫資料的正確性，而在拴鎖的同時，其他處理程序便無法作讀寫的動作，如此一來程式就會等待其處理完成，但也就比較無法達到即時處理的效果。信號雖然是一個保護資源的好方法，但是，其使用上會降低程式執行的效率與拖慢時間。因此，

能不用信號這個機制就盡量少用。所以，我們並不使用信號這個同步機制，而是提出兩種高效率的方法來解決這方面的問題：(a)取代同步機制之溝通機制、(b)仲裁機制。

如圖3.10所示，處理程序A會對記憶體做寫入的動作，而處理程序B則會對記憶體做讀取的動作。使用同步機制的方法就如圖4.10中的(a)所示，當處理程序A寫滿記憶體時會告知處理程序B使其開始做讀取的動作，而當處理程序B做完讀取的動作後，再告知處理程序A，使其繼續做寫入的動作，因此在這樣的機制下，當處理程序B在工作時，A是不能動作的。如此一來，這兩個處理程序並無法同時進行，而這也違背了我們要做多工排程的目的。

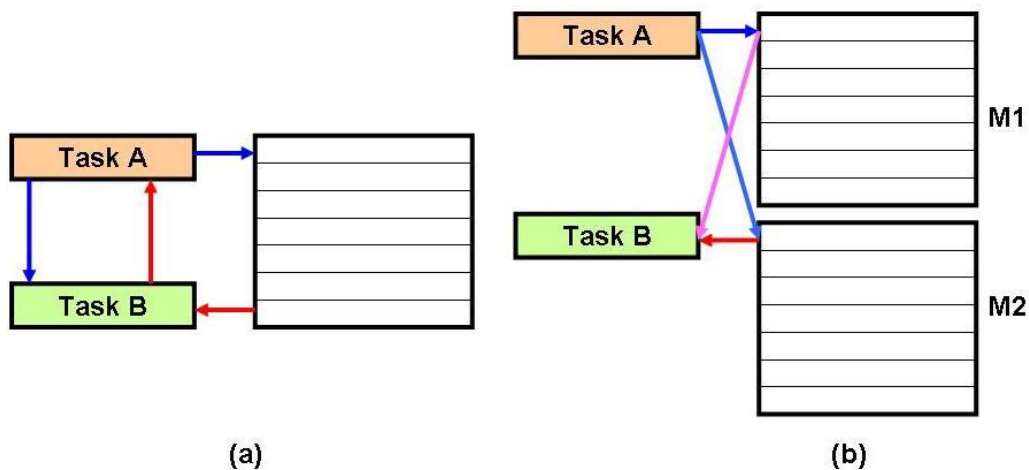


圖3.10、記憶體運用說明圖

因此，我們使用另一種方法解決這個問題。如圖4.10中的(b)所示，我們呼叫了兩塊相同大小的記憶體，當處理程序A寫入資料至記憶體M1時，處理程序B對記憶體M2做讀取的動作。當處理程序都完成工作後再交換，由處理程序A寫入資料至M2時，處理程序B去讀取M1的資料。如此一來，便能讓這兩個程序同時動作並減少一些同步機制的控制與處理程序等待的時間，但缺點是需要以空間換取時間。根據我們對時間上的要求以及ARM上所能支配的記憶體，我們選擇使用這個方法來減少同步機制的使用，以避免在同步時會有一些等待的時間浪費。

(b) 仲裁機制：

在可以避免使用信號的情況中，我們利用上述的方法來解決，但是，還是有許多狀況勢必需要用同步機制的，例如：某些處理程序必須

等待另一個處理程序處理完，才能再繼續執行。此時，就必須使用同步機制，本論文提出以旗標暫存器作為工作排程仲裁機制取代信號。原因如下：

- ❖ 一般在嵌入式作業系統中，是以信號的方式來達成同步的功能；而在一般處理器系統的中，可以以旗標的方式達到類似的效果。
- ❖ 信號使用上的效率很差，而旗標暫存器是以IPC中最快的共享記憶體方式來實現，所以速度會比較快。
- ❖ 在信號的宣告上，一個變數只能代表一個信號，但旗標暫存器則是宣告一個變數可以同時包含許多旗標，所以，使用旗標暫存器可以減少記憶體的使用。

此外，旗標暫存器的用途還不只用於工作排程的仲裁，例如：在取代同步機制之溝通機制中，可以用旗標來讓處理程序知道現在應該對哪塊記憶體作存取的動作。詳細的旗標暫存器的內容如表3.1所示，目前我們只使用這個暫存器中的兩個位元：

- ❖ **bit0**：記憶體區塊切換，在取代同步機制之通訊機制中曾提到我們使用兩塊同樣大小的記憶體來避免同步機制的使用，而bit0就是讓程式知道線再應該讀取哪一塊記憶體。
- ❖ **bit3**：這個bit就是聯結DSP相關程式與一般周邊介面控制程式，這個bit將會驅動DSP的程式，使其開始做運算。

表 3.1、旗標對照表

bit	7	6	5	4	3	2	1	0
功能說明	保留	保留	保留	保留	DSP 致能	保留	保留	記憶體 區塊 切換

(c) 共享記憶體：

在資料傳送至 IPBUF 這個程序中，就必須要先取得從無線模組接收到的資料，此時就必須要能傳遞大量資料的溝通機制。在溝通機制中較常用的有訊息佇列與共享記憶體兩種，而本論文使用共享記憶體來實現溝通機制，主要原因有以下兩點：

- ❖ 由於所需要的通訊的資料是一大塊記憶體，為了避免浪費一些記憶體搬移的動作。
- ❖ 在 IPC 的機制中，共享記憶體是最快的一種。

所以，我們使用共享記憶體的方式，直接取用另外一個程序中的記憶體。由於共享記憶體的使用會需要使用到全域變數的使用，但在 ARM 上全域變數很容易被污染，所以，所以非必要不要使用共享記憶體，而在使用的同時也必須小心使用，避免受到全域變數污染問題的影響。如圖 3.11 所示，我們一共宣告了五個共享記憶體的區塊，以下分別針對這五塊共享記憶體做一簡介：

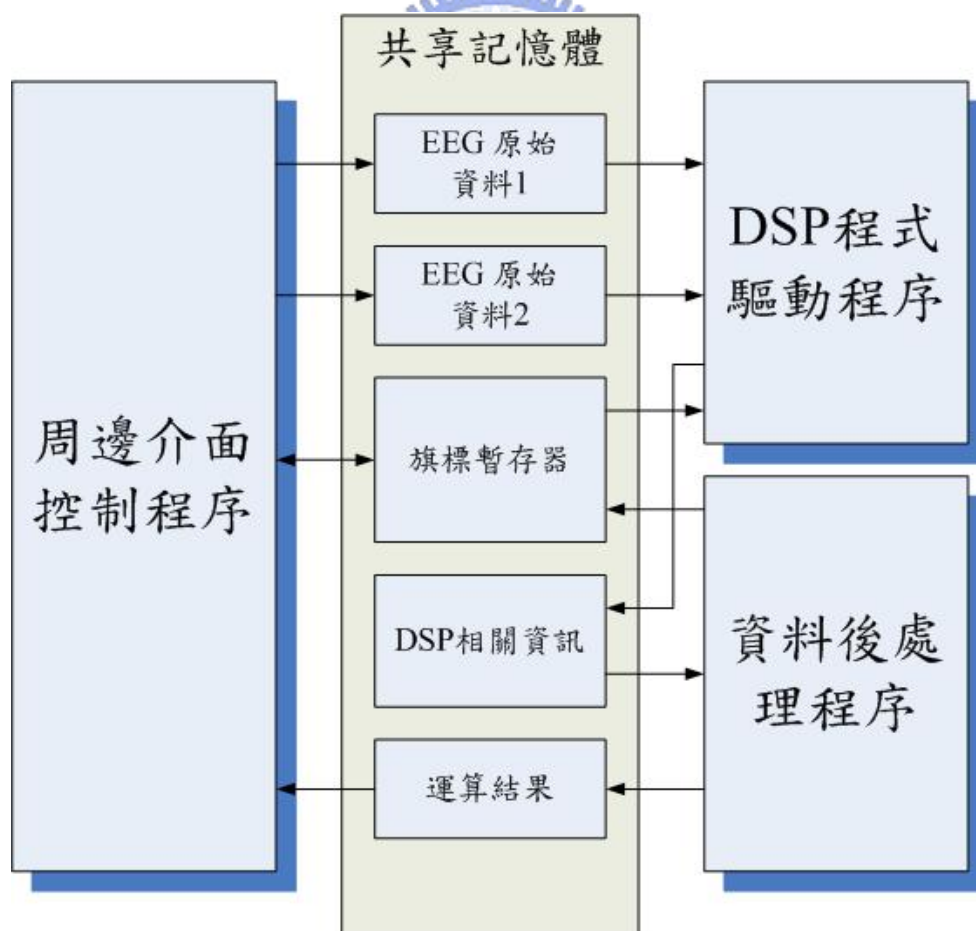


圖3.11、共享記憶體連接示意圖

❖ **EEG 原始資料：**

由於比較複雜的訊號處理是在數位訊號處理器上執行，根據 3.1.3 中的敘述，ARM 利用 DSP Gateway 中的 IPBUF 將資料送至數位訊號處理器做處理，所以在 DSP 驅動程序中邊需要將 EEG 的原始資料載入 IPBUF 中，但 EEG 的資料取得是由無線模組資料交換與區域網路資共享的程序所控制，所以，這兩個處理程序都會共用到這塊資訊。此外，這兩處理程序在讀取與寫入時，為了保護這塊記憶體，避免因信號拴鎖住資料而浪費一些時間，降低執行效率，所以，我們引用之前所提到取代同步機制的方法—另外宣告了一塊共享記憶體來避開這兩個處理程序同時做讀取與寫入的機會。

❖ **旗標暫存器：**

這塊旗標暫存器便是之前工作排程仲裁機制中所提到的旗標暫存器，詳細的旗標內容與使用原因，皆已有詳細的介紹。比較需要注意的是在定義旗標時要避免會有兩個處理程序同時做讀取與寫入的狀況。

❖ **DSP 相關資訊：**

由於跟 DSP 相關的處理程序有兩個，而這兩個處理程序自然都需要所啟動 DSP 裝置的相關資訊，而自然這兩個處理程序也不會有同時執行的機會。

❖ **運算結果：**

運算結果是由 DSP 所產生，但由於這個運算結果將會由網路將其傳送出去，所以，這部份也必須利用共享記憶體來讓這兩個處理程序共享這個資源。

在實現上，還是要避免運用到這些機制，因為只要使用就會增加資料被毀壞以及時間被延遲與浪費的危險，所以，不可以肆無忌憚的濫用。此外要使用之前要審慎評估其可能遭遇的情形，將所有發生錯誤的可能性降到最低。

3.3.3 基於嵌入式多工排程之系統架構設計

在 3.3.1 及 3.3.2 中，我們已詳述本論文所提的排程系統架構以及創新的多重處理程序溝通機制，在本小節將更詳細說明如何應用這些觀念來架設本論文所提出的系統。在 OMAP1510 上，有許多工作需要執行，我們根據其工作的特性將工作分為三類：(1)周邊裝置溝通、(2)複雜的訊號處理、(3)簡單的數學運算。為了發揮 ARM 與 DSP 的專長，以做到有效的平行處理，我們將複雜的訊號處理運算放入 DSP 做運算，而其他工作則交由 ARM 做運算，詳細的工作分配如表 4.1 所示。

表 3.2、ARM 和 DSP 工作分配表

微處理器	工作
ARM	 <ul style="list-style-type: none"> (1). 透過無線裝置接收 EEG 資料 (2). 利用無線裝置傳送回受提醒訊號 (3). Data 位元轉換 (4). 呼叫 DSP task 運算 (5). 傳送資料給 IPBUF (6). 從 IPBUF 接收資料 (7). 將運算結果利用網路封包傳輸出去給其他裝置使用
DSP	<ul style="list-style-type: none"> (1). 從 IPBUF 接收資料，和傳送資料給 IPBUF (2). 執行演算法

而資料的流向則如同圖 3.12 所示，ARM 上面的工作被分為三塊，而資料的連結是用共享記憶體來連接。其資料的流向可分為五個步驟：

- (1) 首先，資料由控制無線模組的處理程序接收進來之後，除了傳至網路上以顯示在遠端的電腦外，同時將資料存入共享記憶體中。
- (2) 資料累積到一定數量後驅動 DSP 的處理程序將 DSP 處理程序啟動，同時將

資料送入 DSP 中。

- (3) 收滿資料後，DSP 開始做運算，我們讓 DSP 做比較複雜的運算，包括 window 運算與頻譜轉換及處理。
- (4) DSP 做完運算後，運算結果回送給 ARM，在 ARM 接收完 DSP 的處理結果後，再做一些後處理程序。運算的結果再存回共享記憶體中，以便將原始資料傳送至網路上時，同時也把運算結果傳送出去。
- (5) 根據運算的結果判斷是否啟動或關閉回授提醒訊號，同時將這個訊號送至回授提醒裝置。

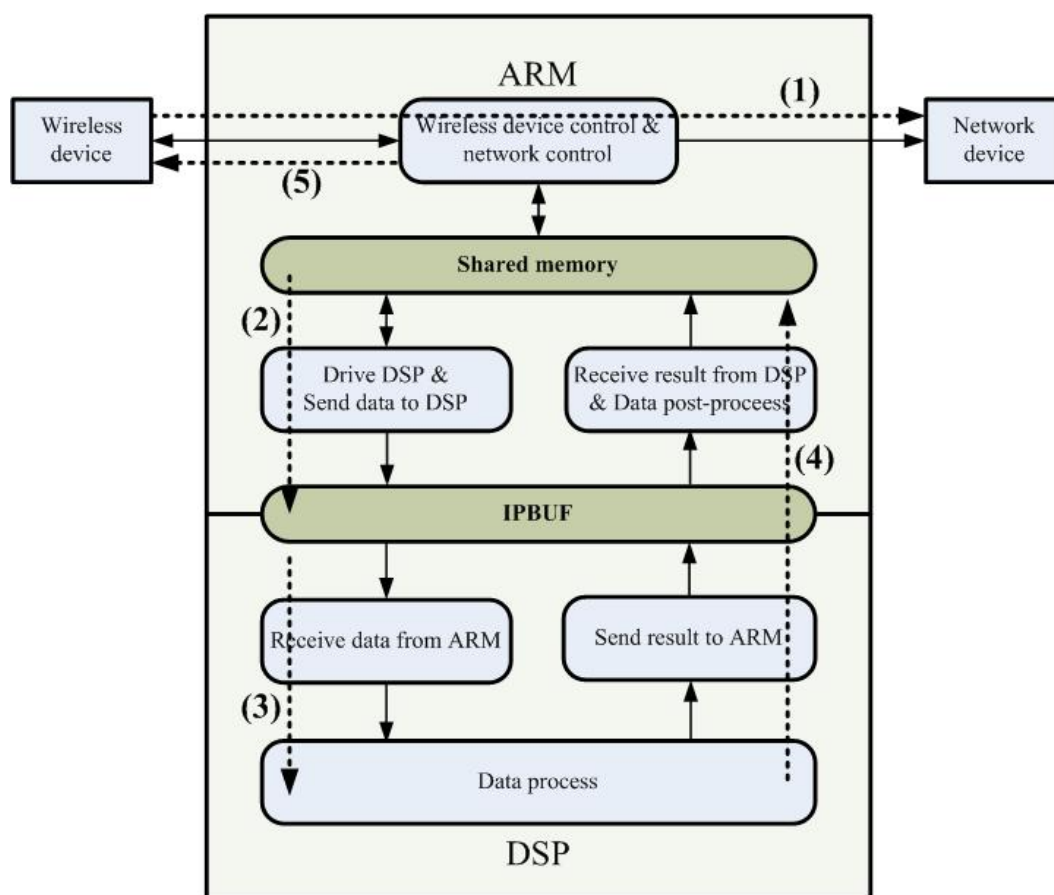


圖3.12、資料處理狀態圖

在圖 3.12 中的每一個處理程序是彼此獨立的，但由上述的流程可以發現，在圖中共享記憶體下方的處理程序是有一定的相依性，所以，會由一定的先後順序。但在下方的幾個處理程序做運算的同時，上述流程中的第一點是不斷的在進行。

3.3.4 ARM 處理器端周邊裝置控制程式設計與實現

如同3.3.1所提到的，我們將ARM上的處理程序分為三大項，其中有兩項與訊號處理有關，我們將在第四章介紹與訊號處理相關的處理程序，而本節將介紹ARM上對周邊裝置做控制的處理程序。此處理程序主要負責與周邊介面的溝通，在本系統中負責(1)從無線通訊模組取得腦電波訊號、(2)將資料以TCP/IP的方式送至網路上，由於本系統必須持續的接收無線訊號，所以，本處理程序會持續進行。其程式流程如圖3.13所示。為了使接收訊號的頻率保持固定，我們必須確保這個處理程序在執行上不能被延遲太多。

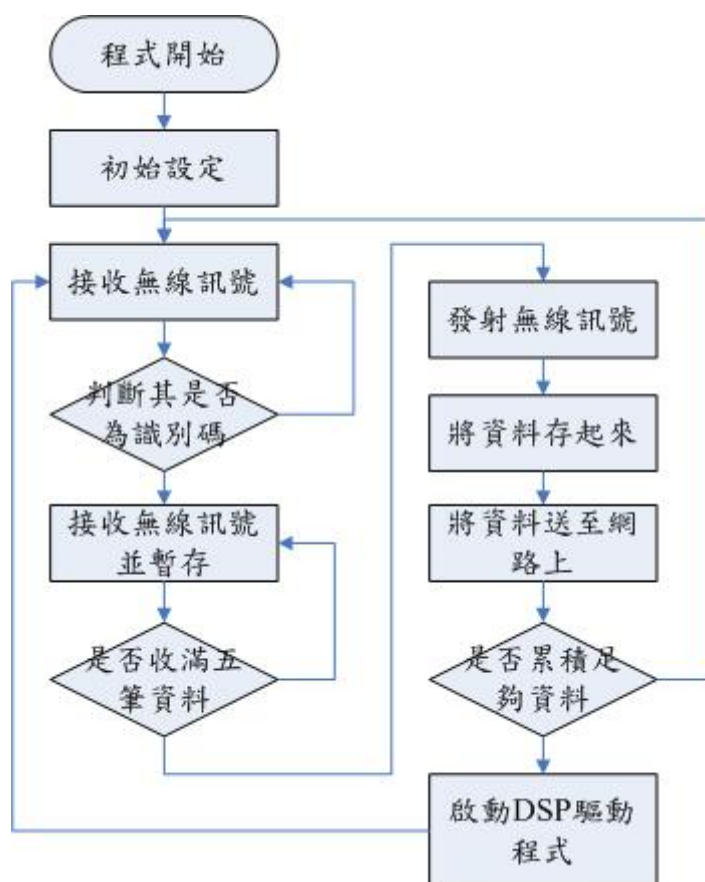


圖3.13、周邊控制程序流程圖

關於無線傳輸的部份，我們在之前已做了非常完整的介紹。接下來我們將介紹這個處理程序所負責的另一項工作——網路控制。如圖 3.14 所示，我們利用遠端的電腦做檔案系統與運算結果顯示。由於生理訊號運算結果的資料量很大，所以，我們以網路檔案系統的方式將資料儲存在遠端的電腦上，而讓嵌入式系統上的記憶體可以運用在數值計算上。此外，我們架設了如圖 3.15 所示的圖形介面，

讓使用者可以即時知道當時的腦波的波形以及其運算的結果，其中 Frame A 與 Frame B 為原始的腦波資料，而 Frame C 與 Frame D 則為即時分析運算的結果。而 PC 與 OMAP 之間的連接，我們是以 TCP/IP 的協定來完成。

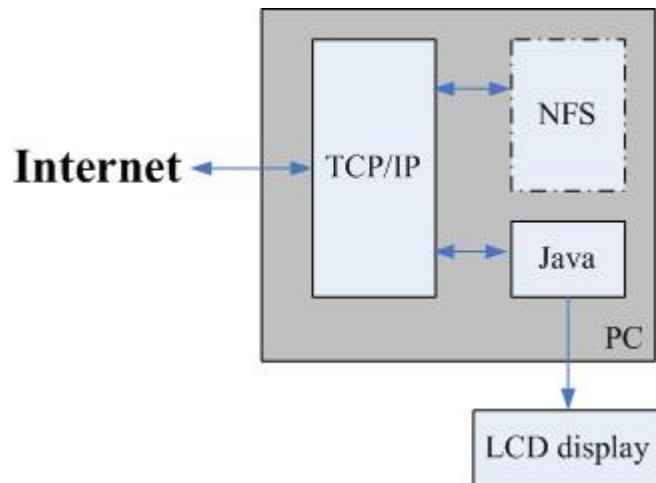


圖3.14、區域網路共享架構圖

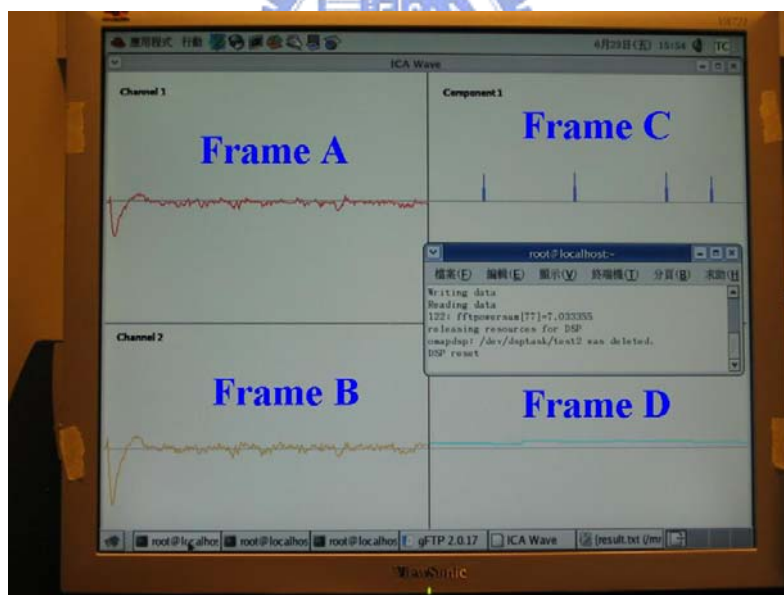


圖3.15、遠端電腦使用者圖形介面

在 TCP/IP 協定下網路中的每一台電腦都有一個主機位置，例如：192.168.0.1。除此之外，如要進一步通訊傳輸的話，還得替主機開啟一個連線插座(socket)。當 A 電腦與 B 電腦之間要通訊時，雙方必須各開一個 socket 來通訊，猶如家中電話或電源插座。實際上的方式為當 A 電腦在主機上的 6633 埠建立一

個連線接口，則 B 電腦就要透過 6633 接口連線傳輸，所謂接口就是 socket 在主機上的明確位置。當 socket 建立完成後，接下來就能透過 socket 這 I/O 管道來傳輸資料。接著 B 電腦可將輸出串流資料寫到指定的連線主機接口，A 電腦可自指定的接口將連線主機的資料讀回，連線流程如圖 3.16 所示。

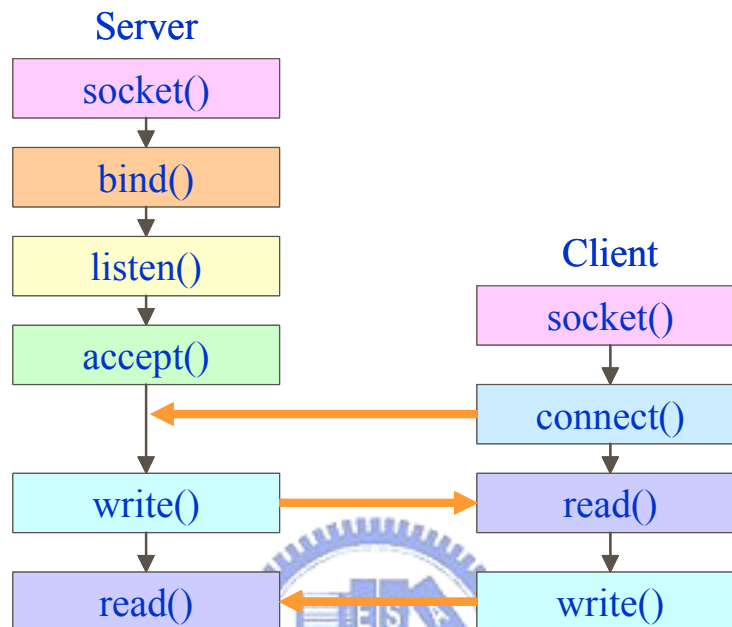


圖3.16、TCP/IP網路連線流程圖

第四章 汽車駕駛員疲勞狀態偵測與提醒系統之設計

本論文的最終目標是利用 OMAP 嵌入式單板建構一套即時可攜式無線腦機介面系統，並將這個系統實際應用於汽車駕駛員疲勞偵測與提醒系統的架設中。在第二章中已經介紹我們如何架設一無線的腦電位訊號感測裝置，而在第三章中，除了介紹我們所使用的嵌入式系統平台外，亦介紹了我們所設計的嵌入式多工排程系統。本章將詳述我們如何整合前面所開發的技術於汽車駕駛員疲勞偵測與提醒系統的應用中。首先，本章將介紹整個疲勞偵測與提醒系統的架構，接著分別針對實驗設計、訊號處理流程、即時訊號處理系統以及疲勞提醒系統的實現細節作說明。

4.1 疲勞狀態偵測與提醒系統架構

整體系統架構如圖 4.1 所示，系統平台的細節已分別在第二章以及第三章中介紹，而本章將著重於資料處理的部份，整個資料的流向可以分為兩條線：

- (1) EEG 訊號藉由電極擷取，再經過放大電路放大、類比/數位轉換，最後以無線傳輸的方式送入 OMAP1510 做處理，同時在 OMAP1510 上，我們利用網路將所接收到的資料傳至遠端電腦上做顯示。
- (2) 運算結果會再利用網路傳輸至顯示介面的電腦上，而同時會針對運算的結果，即時回傳警告訊號給受測者。

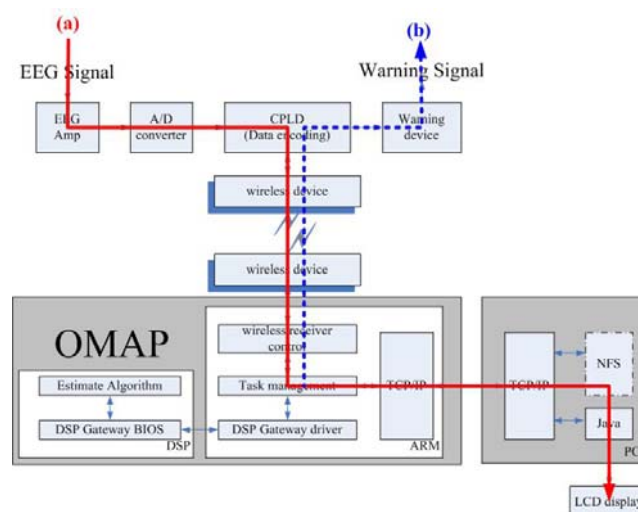


圖4.1、疲勞狀態偵測與提醒系統架構圖

4.2 駕駛員疲勞狀態估測實驗設計

我們設計了一套實驗來量測駕駛員在長途駕車時的腦電位變化，藉以找到駕駛員產生疲倦、昏睡以及失神時的腦電位變化。我們藉由 3D 環繞式虛擬實境場景以及六軸動感平台模擬產生快速道路場景與駕駛者在快速道路上可能遭遇的突發狀況，並且實際量測駕駛者於長途駕駛時的腦電位訊號，我們所設計的長途高速駕車實驗規劃如下：

- (1) 實驗的基本快速道路場景如圖 4.2 所示：為一雙向四線道的快速道路，左邊有行道樹，右邊則為平原。為了減少視覺的干擾，我們簡化場景如圖 4.3 所示，將圖 4.2 中的行道樹與柵欄去除。如圖 4.3 所示，除了虛擬實境的道路顯示外，我們利用一個搭載真實汽車的六軸動感平台來模擬真實駕車時的動態感受。
- (2) 在腦電位訊號的量測上，如圖 4.4 所示，本論文創新以在前額貼上一排電極的方式，來取代傳統的 10-20 系統，因為這樣的設計便於電極的放置以及日常生活的應用，降低對使用者造成的不覺與困擾。
- (3) 駕駛者需要維持車子開在黃線右方內側車道（由左邊數來）的中心點上，車子時速為 100km/hr。
- (4) 車子會隨機的左右漂移，以使駕駛者集中注意力在維持車子在中心車道上。
- (5) 系統的效能(performance)估計以車子是否偏移車道中心線的距離為準則。在實驗的過程中，會不定時施於車子左右偏移之力量。一般而言，當駕駛者的精神狀態清醒時、則車子可以很容易維持在車道的中心線上，反之，當駕駛者精神狀態有出現失神、疲勞或者昏睡時、車子則不容易維持在車道中心線。

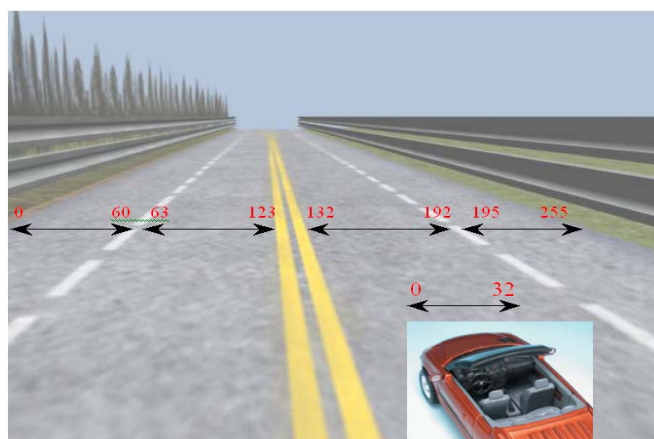


圖4.2、以虛擬實境產生的快速道路場景



圖4.3、實際實驗場景與動感模擬平台



圖4.4、實驗電極黏貼示意圖

量測腦波資料時間選在中午或晚上吃飽飯後容易想睡覺開始。在戴電極帽及打導電膠的時候，一邊告訴受測者要開車於哪一線道上，眼睛要注視車道前方的中間不要亂飄，一邊讓受測者練習駕駛。完成量測前的準備後，開始一個小時的時間，在昏暗的房間裡不受任何人的干擾下開車。而選擇的受測者，年齡在20歲到40歲之間，並有道路駕駛之經驗，每個人至少作兩次的實驗。

4.3 疲勞狀態偵測演算法設計

如圖 4.5 所示，當完成兩天的資料收集後，我們先用第一天的資料放入 OMAP 1510 做基本的訊號處理，接著再將這些處理後的資料放入電腦中做估測模型(Detection model)的建立。為了驗證所建立估測模型的正确性，我們將第二

天的資料同樣放入 OMAP 1510 做基本的訊號處理，之後再將這些資料放回電腦，套入之前訓練出的模型做測試。當測試的結果有一定的準確度，我們才將這個估測模型放入 OMAP 1510 中。本節將介紹相關的訊號處理流程與方法，首先，我們在 4.3.1 中，依據訊號處理的流程概述所有處理的步驟，接著，在 4.3.2 中則將針對訊號前處理的過程做詳細的介紹。在本論文中，我們所實現的估測模型包含主成分分析(Principle component analysis, PCA)以及線性回歸(linear regression)演算法建立模型，而這兩部分將於 4.3.3 以及 4.3.4 中做較為詳細的介紹。

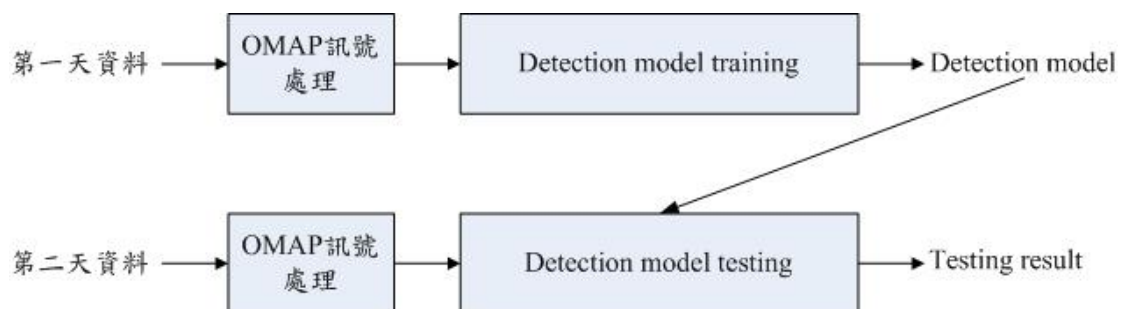


圖4.5、估測模型建立流程圖

4.3.1 腦電位訊號處理流程

OMAP 1510上的訊號處理流程圖如圖4.6所示，當資料取得後，會經過七道處理程序，分別為：(1) Down sampling、(2)Hanning Window、(3)Short-time FFT processing、(4)Normalization、(5)Moving average、(6)PCA model以及(7)linear regression model，分別敘述如下：

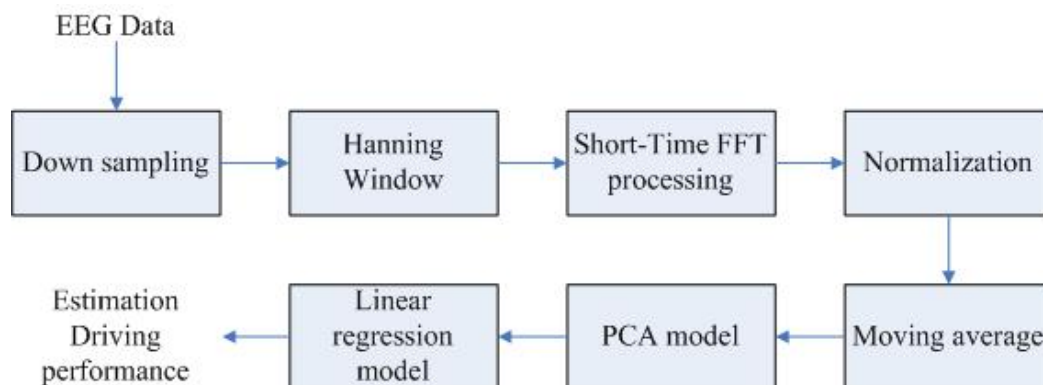


圖4.6、腦電位訊號處理流程圖

(1) Down sampling :

在正式進入資料處理程序前，先將重新取樣成每秒 64 點，由於人體昏睡的時候的波形主要是在於 25Hz 以下的頻率，而之後我們所放進我們所訓練的分析模型也是取 1-25Hz，根據取樣定理，64Hz 的取樣頻率是足夠的。

(2) Hanning Window :

由於如果直接將時間序列的訊號做頻譜轉換，會造成比較嚴重的失真，所以，一般在數位訊號處理上會在頻譜轉換前將時間序列的訊號乘上一個視窗(Window)函數。在我們的系統中，我們使用比較常用的 Hanning Window 來做為我們的視窗函數。

(3) Short-time FFT processing :

由於腦電位訊號是屬於較複雜的訊號，所以必須要轉成頻譜，才可以看出其特徵，在此我們以短時快速傅立葉轉換 (Short-Time FFT, STFFT) 的方式來將時間序列的訊號轉為頻譜訊號。在做完頻譜轉換後，我們僅選擇主要跟人體疲勞度有關的頻帶(1-25Hz)來做相關的分析與運算。關於頻譜轉換與一些訊號前處理的細節將於 4.3.2 做較詳盡的介紹。

(4) Normalization :

經過短時快速傅立葉轉換得出各 EEG 每兩秒鐘的能量狀況後作正規化動作。

(5) Moving average :

由於人體精神狀況的改變，其週期超過 90 秒，因此，在訊號分析上不是只看某個瞬間腦波的值，而是看一段時間的變化。所以，把 EEG 訊號之頻譜作 Moving average(移動平均)可以更進一步減少因為人為因素所引起的 EEG 雜訊而不影響我們所想觀察的腦電位訊號變化。

(6) PCA model :

根據文獻記載，人體疲勞時，腦波主要在後腦勺的附近會有比較強烈的波形變化，而我們所兩側的位置是位於前額，所以距離後腦勺有一段距離，

再加上電極所放置的位置非常接近以及使用電極的雜訊可能很大，所以，我們利用 PCA 來找出真正與我們所要預測的現象相關的成分。在此，我們會先利用之前該受測者的實驗數據，找到一個 PCA 的模型，將其放入 OMAP 中做即時的換算。詳細的主成分分析演算法，將於 4.3.3 中做介紹。

(7) linear regression model :

在預測駕駛者的精神狀態方面，我們是利用線性回歸(linear regression)演算法建立模型，再放入 OMAP 中做即時估測。線性回歸模型是用 50 階的線性多項式，成本函數 (cost function) 用誤差的最小平方 (least - square - error, LSE)，詳細的線性回歸分析方法 (Linear regression)，將於 4.3.4 中做介紹。

4.3.2 訊號之前處理

濾除眼動、心跳及其它雜訊之訊號，在分析之前必須進行前處理。由於人體的精神狀況的改變，其週期超過 90 秒，不是只看某個瞬間腦波的值，所以如圖 4.7 所示把 EEG 訊號之頻譜作 Moving-average (移動平均) 就是取 128 點 (2s) 的 EEG 訊號加上 64 點 (1s) 的重疊為一個 Window。接著把這些 192 點每隔 6 點再分成幾個 32 點的 sub-windows，每個 sub-window 再以補零的方法擴充成 64 點，作 64 點的 FFT。接下來，把每一小段頻譜相對應的頻率上取中位數，得到一段平均之後的頻譜可更進一步減少因為人為因素所引起的 EEG 雜訊，例如：眨眼、搖頭...。在每隔兩秒輸出的 EEG 頻譜序列中，我們僅選擇與昏睡狀態比較相關的頻帶 1-25Hz 的 EEG 功率頻譜取 log 得到 dB 值。

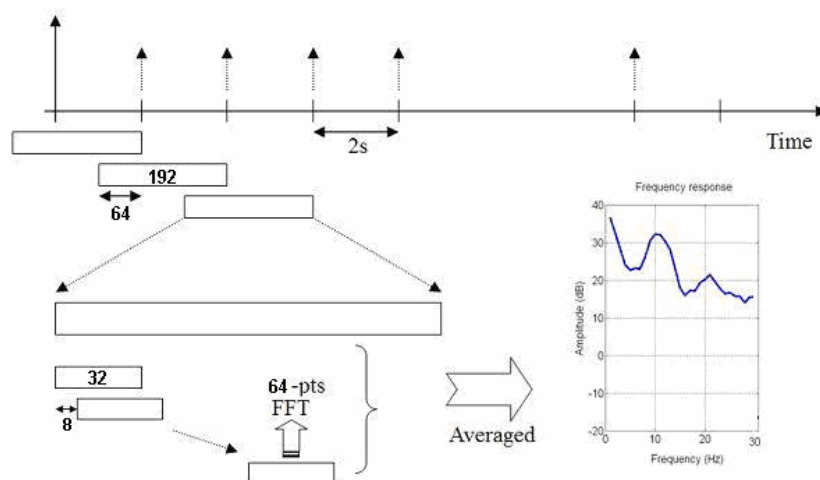


圖4.7、訊號之前處理

4.3.3 主成分分析

在許多研究中，為了全面系統地分析問題，都盡可能充分完整地蒐集資訊。對每個觀測對象往往需測量很多變數，但是由於資訊越多，計算也越複雜，所以在這些數值分析上希望用較少的新變數代替原來較多的舊變數，而這些新變數應盡可能反應舊變數的資訊。

在我們的系統中，我們利用頻譜轉換後的資料作為線性回歸演算法的輸入變數，我們取 5 個頻道、1-25Hz 的頻率一共 125 個變數。但 125 個變數對於嵌入式系統的所能負荷的運算量來說實在太多了，所以，我們希望取較少個變數，而能同樣反映出應有的趨勢。主成分分析正是滿足這一要求的處理多變數問題方法，由於它們能濃縮資訊、使指標維降、簡化指標的結構以及使分析問題簡單、直覺化，故被廣泛地應用於醫學、心理學、經濟學等領域。

主成份分析所著重的在於如何「轉換」原始變項使之成為一些互相獨立的線性組合變數，而且經由線性組合而得的主成分仍保有原變數最多的資訊，其關鍵在「變異數」問題，利用求特徵值(eigenvalue)及特徵向量(eigenvector)之方法，過濾出佔最大變異數的型態，此即為最主要之型態。主成分分析演算法步驟如下：

- (1) 算出共變異數矩陣 S ，或相關矩陣分析 R
- (2) 求 S 或 R 的特徵值及單位特徵向量
- (3) 將特徵值依大小順序排列，設分別為 $\lambda_1 \cdot \lambda_2 \cdot \lambda_3 \dots \lambda_p$
- (4) 求出對應的單位特徵向量 $a_1 \cdot a_2 \cdot a_3 \dots a_p$ ，其中 $a_i' a_i = 1$ 且 $a_i' a_j = 0$ ，則

$$y_1 = a_1' x = a_{11} x_1 + a_{12} x_2 + \dots + a_{1p} x_p \quad \text{為第一主成分}$$

$$y_2 = a_2' x = a_{21} x_1 + a_{22} x_2 + \dots + a_{2p} x_p \quad \text{為第二主成分}$$

$$\vdots \quad \quad \quad \vdots$$

$$y_p = a_p' x = a_{p1} x_1 + a_{p2} x_2 + \dots + a_{pp} x_p \quad \text{為第 } p \text{ 主成分}$$

若只取最大的 q 個主成分代替原有的 p 個變數，則這 q 個主成分解釋的變異比例為

$$\sigma^2 = \frac{\lambda_1 + \lambda_2 + \lambda_3 + \dots + \lambda_q}{\lambda_1 + \lambda_2 + \dots + \lambda_q + \dots + \lambda_p} = \frac{\lambda_1 + \lambda_2 + \lambda_3 + \dots + \lambda_q}{p}$$

一般以能解釋原有變數變異達 70% 以上為原則。一般而言，使用主成分分析的原則是在於變數間的相關性高，所以，可以使用幾個比較具代表性的成分來做化簡，而當變數間相關性低則不適合利用主成分分析作簡化[43]。

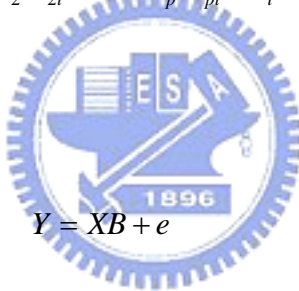
在我們的應用中， x 代表頻譜轉換後的資料， y 則是經過主成分分析後所取得的新變數； p 為頻譜轉換後的資料共125筆，至於 q 則是轉換後的變數個數， a_{ij} 便是在偵測模型中我們所需要的PCA矩陣。

4.3.4 線性回歸分析

使用線性回歸或回歸分析方法作預測時，並不是數據點的散布圖看起來像直線的才適合，只要數據點的散布圖看起來像某一種函數的圖形，都可以用迴歸分析的方法，其中如果迴歸模式的待求、未知參數呈線性的話，就是所謂的線性迴歸分析模式。本研究使用的迴歸模型，如式子(4.1)所示，其中線性迴歸模型的輸入參數 X 代表經過 PCA 轉換後的新變數，而輸出參數 Y 則代表預測駕駛員的駕駛軌跡與車道中心線的距離，藉以作為駕駛員精神狀態評估的依據。

$$Y_i = a + b_1 X_{1i} + b_2 X_{2i} + \dots + b_p X_{pi} + \varepsilon_i \quad (p \geq 2) \quad (4.1)$$

可以表示成



$$Y = XB + e \quad (4.2)$$

其中 Y 是 $n \times 1$ 的矩陣， X 是 $n \times (p+1)$ 的矩陣， B 是 $(p+1) \times 1$ 的矩陣， e 是 $n \times 1$ 的矩陣， B 與 e 便是之後要放入嵌入式系統中的線性迴歸模型參數。如式子(4.3)所示：

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix}, \quad X = \begin{bmatrix} 1 & x_{11} & x_{21} & \cdots & x_{p1} \\ 1 & x_{12} & x_{22} & \cdots & x_{p2} \\ 1 & x_{13} & x_{23} & \cdots & x_{p3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{1n} & x_{2n} & \cdots & x_{pn} \end{bmatrix}, \quad B = \begin{bmatrix} \alpha \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix}, \quad \varepsilon = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_n \end{bmatrix} \quad (4.3)$$

同理若使用誤差的最小平方 (least-square-error, LSE) 估計參數，則可以表示成式子(4.4)：

$$\hat{B} = (X^T X)^{-1} X^T Y \quad (4.4)$$

4.4 疲勞狀態偵測與提醒系統實現

在實際的系統實現上，我們在 OMAP1510 上實現一套即時分析演算法，同時即時根據處理結果，若有昏睡、失神的情形發生時，會即時傳送回訊號至會受提醒裝置已發出提醒訊號警告使用者。本節將針對訊號處理系統以及回授提醒系統實現的細節作解說。

4.4.1 DSP 處理器端運算系統實現

如圖4.8所示，我們將最消耗運算量與運算時間的window運算、頻譜轉換及處理交由DSP來做處理。在DSP上實際實現的程式流程如圖4.9所示，當DSP由IPBUF收滿資料後，會針對每一小段資料乘上Hanning window，接著做short-time FFT，當所有的資料都做過頻譜轉換後，選擇我們所想要的頻段(1~25Hz)，做取平均的動作，最後就得到我們所需要的五個頻道、1~25Hz的頻譜響應。

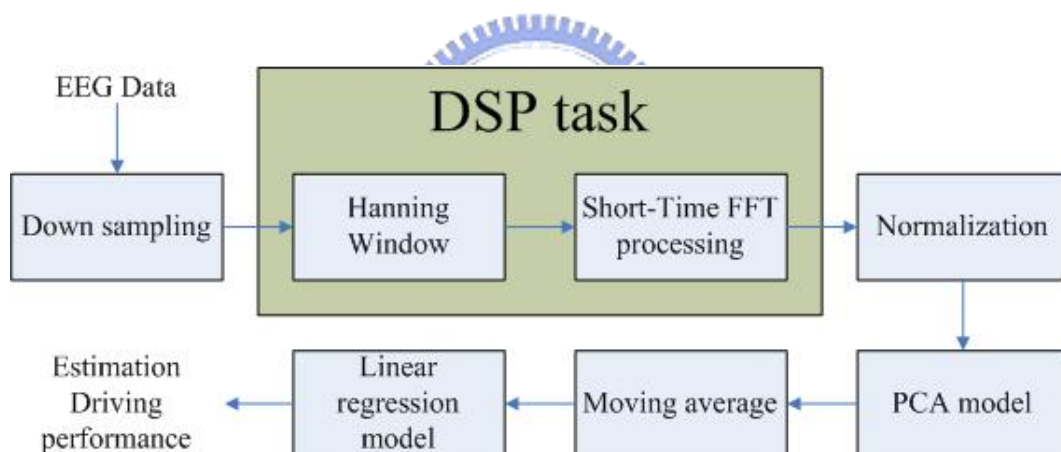


圖4.8、訊號處理流程圖

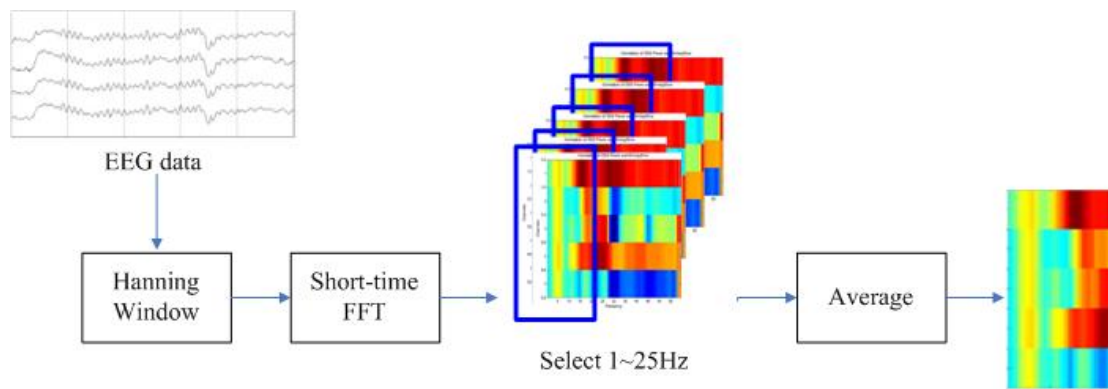


圖4.9、DSP訊號處理程式流程圖

對DSP來說，由於其記憶體有限，但我們的運算上又必須需要大量的資料來做運算，所以必須減少變數的使用並大量重複使用記憶體。舉例來說，如果以電腦來實現前處理的運算程式，如圖4.10中(A)所示，可以依據時間同時放入五個頻道的資料，等到時間累積到一定時即同時針對這五個頻道做運算並輸出其結果，而這也比較符合即時運算的概念，但這種方法會消耗較多的記憶體，主要是其所被宣告的變數無法被重複使用，假設一個頻道做相關的運算需要100個單位的記憶體，做五個頻道的運算則需要500個單位的記憶體。但由於在實現上，我們是一次傳送一大筆資料進DSP做運算，所以其實我們會有一段時間的資料，因此，我們可以視為這一段時間的資料做離線分析，所以，一次算一個頻道的資料，算完之後再算下一個頻道的資料，其流程如圖4.10中(B)所示，如此一來，在運算期間許多記憶體都可以被重複使用，如此一來就可以節省許多記憶體空間。

另外，在電腦上做處理的時候，在Average的運算中，我們是以取中位數的方法來取得其運算值。舉例來說，192筆資料可以得到21次運算的結果，在電腦上做分析運算時，我們利用取中位數的方式來代表該頻道該頻率的運算結果，但這樣的運算必須要做排序的動作，所以，其運算量就會很大，但其特色就在於能比直接取平均的方式降低一些例外的值造成的影響。例如，大部分的值都落在5~6之間，而只有一個13的值，如此一來，取平均的值就會比較大，也比較不能代表這群資料的特性。但由於運算空間以及運算時間的考量，如果在嵌入式系統上實現取中位數的作法，其投資報酬率非常的低，因為，花費大量的記憶體暫存空間以及排序所需的時間，只為了取到較足以代表該資料特性的值，因此，我們不方便在DSP中實現這種取中位數的演算法。在此，根據資料的特性以及我們所要實現的目的，我們實現了三種方式來做測試，以及供使用者選擇：

- (1) **方法一**：我們曾在電腦上試著以取平均的方式來取代取中位數的方法，雖然說效果會比較差，但以實現方法以及運算時間來說取平均的方法較為省時間與省空間。此外，由於不同受測者的腦電波的特徵也不盡相同，所以，我們就保留了取平均的方法。
- (2) **方法二**：為了消去某些特殊例外的值，又希望可以取到比較能代表這組數值特性的值。所以，我們將所得到的資料先做排序，取得中位數後，將與中位數整數相同的所有值相加後取平均，藉以作為輸出。然而在取中位數的計算上，為了加快時間，我們僅取整數做排序。
- (3) **方法三**：如圖4.11所示，由於在腦電波的頻譜上來說相鄰的幾個頻率其頻譜數值大小差距不大，所以，我們可以利用一個數值來代表這幾個頻

率的特徵值，再拿這些特徵值的整數值來做排序，取得中位數後，將與中位數整數相同的所有值相加後取平均，藉以作為輸出。我們針對這三種方法做了一些分析與比較，將在成果與討論中介紹。

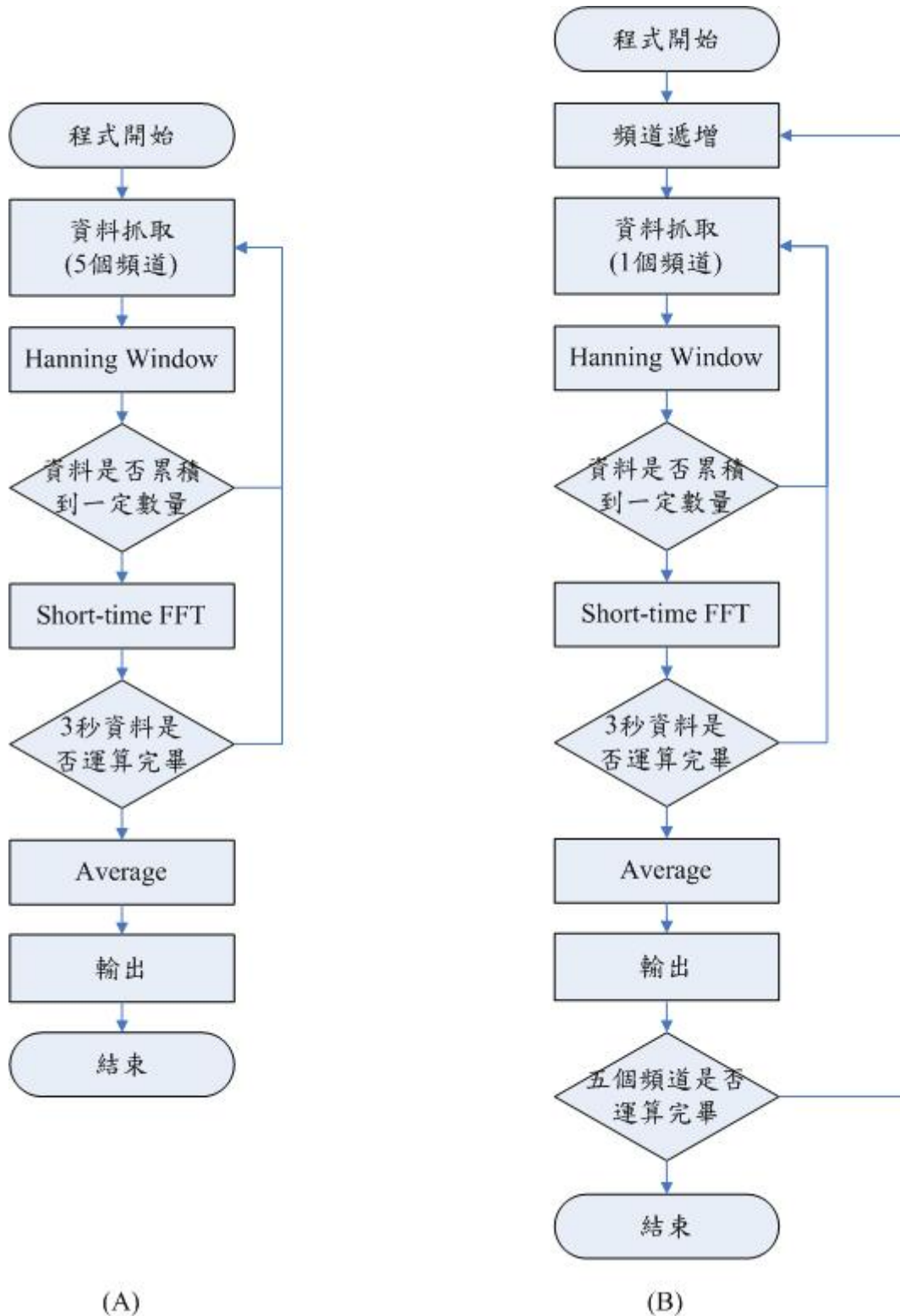


圖4.10、訊號處理程式流程圖

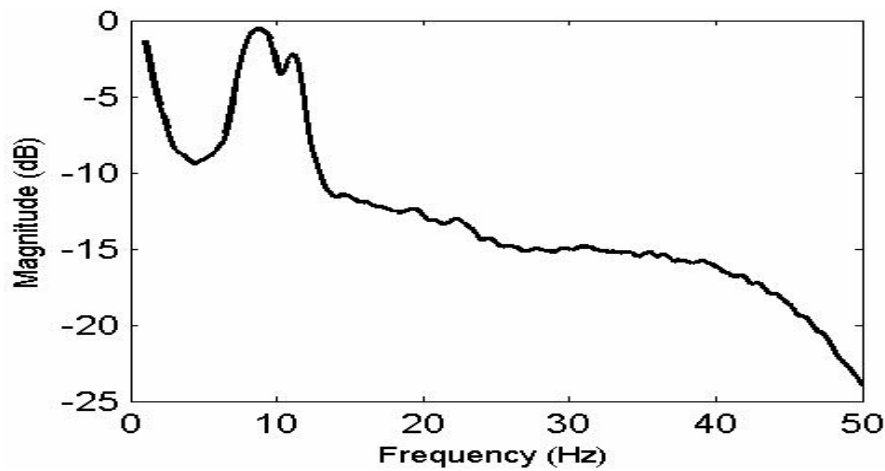


圖4.11、腦電波訊號的頻率分布圖

4.4.2 ARM 處理器端與 DSP 溝通及訊號後處理系統實現

在ARM處理器上跟DSP相關的處理程序一共有兩個，包括DSP處理程式驅動以及資料後處理等兩項工作，以下分別針對這兩個處理程序作說明：

(1) DSP處理程式驅動處理程序

本程序主要的目的是開啟DSP與喚醒DSP上的訊號處理程式，其處理流程如圖4.12所示，在傳送資料至DSP端之前，由於ARM跟DSP上的資料格式不一樣，所以在將資料傳輸過去之前要先做資料格式的轉換。接著，將轉換後的資料傳送至IPBUF，同時告知資料後處理程序可以開始等待運算結果。



圖4.12、DSP程式啟動流程

(2) 資料後處理程序

如圖4.13所示，這個處理程序除了接收DSP运算完的結果並做資料格式轉換外，在這個處理程序中亦做正規化、移動平均、乘上PCA權重矩陣以及乘上線性迴歸模型的參數等运算。這些运算放在ARM上做處理有三個原因：

- ❖ 這些运算過程較簡單。
- ❖ 減輕DSP的負擔，加強平行處理的效果。
- ❖ 將DSP上的處理程式模組化，可以被重覆使用，在使用不同的分析模型時，只要更改資料後處理的程式即可。

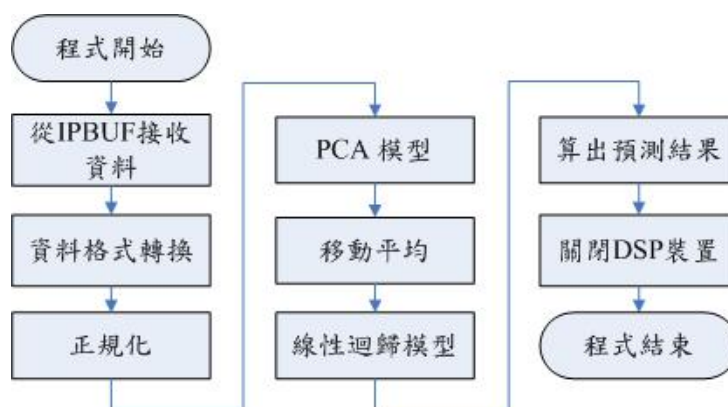


圖4.13、資料後處理程式流程

4.4.3 疲勞提醒系統之實現

在實現疲勞提醒系統之前，我們先測試一下哪一種回授提醒訊號的提醒效果比較好？在視覺與聽覺的回授訊號中，聽覺訊號使用上比較方便也比較合理。而在聲音訊號中，我們嘗試使用不同頻率的訊號來看哪一種訊號的提醒效果較好，我們使用 500Hz、1750Hz、3000Hz 三種頻率來作測試與比較。測試結果發現聲音的提醒訊號的確有助於增加開車的績效，而在這三種頻率中，我們發現 1750Hz 的提醒效果最好，所以，在實現上我們便以 1750Hz 作為參考。

在回授訊號的迴路上，根據运算結果，會將精神狀態分為清醒與昏睡兩種，回傳至回授提醒裝置，如果精神狀態為昏睡，系統會發出警告音來提醒駕駛員。修改過後的 RF3100 控制模組如圖 4.14 所示，表 4.2 則為模組接腳表，除了原本的傳送模組外，再加入接收模組以及回授訊號辨識模組，圖 4.15 為回授訊號的辨識流程，當接收到無線訊號後，會先比對該訊號，確認連續收到的訊號是相同的，才會進入資料解碼的動作，如此一來便有兩層關卡可以保護資料，以確保其接收到的資料是正確的。

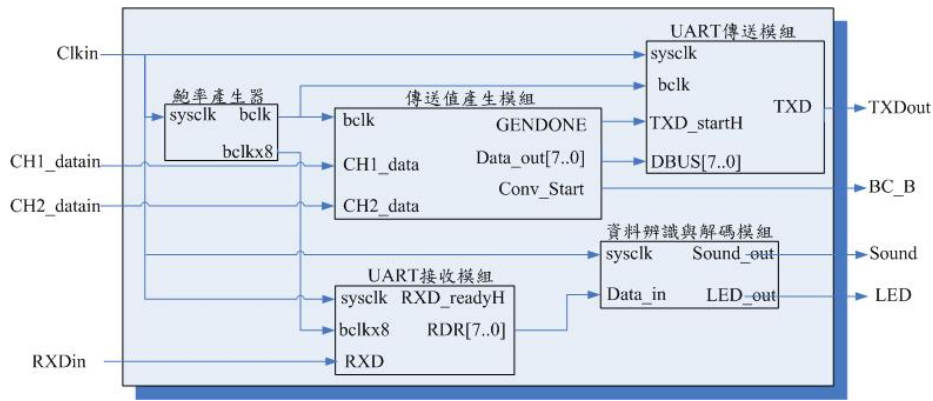


圖4.14、用於雙向資料傳輸的RF3100控制模組架構圖

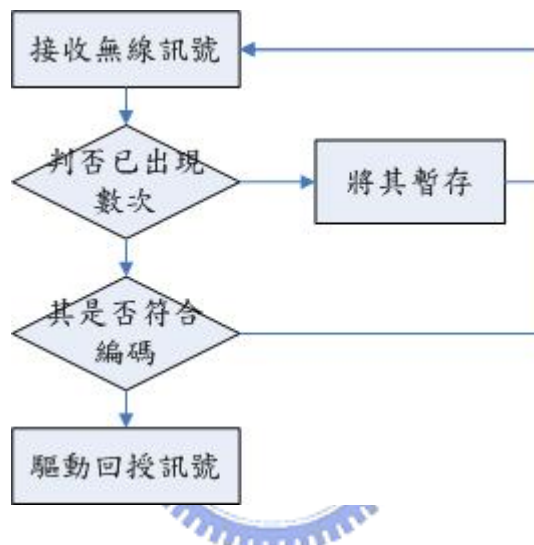


圖4.15、雙向傳輸資料辨識流程

表 4.1、RF3100 控制模組接腳表

信號名稱	屬性	作用極性	功能
Clkin	Input		系統之震盪頻率
CH1_datain[7..0]	Input		A/D converter 頻道 1 數位資料輸出
CH2_datain[7..0]	Input		A/D converter 頻道 2 數位資料輸出
RXDin	Input		從無線模組接收資料
TXDout	Output		傳送資料至無線模組
BC_B	Output	High	A/D converter 致能訊號
Sound	Output		警告聲音輸出
LED[7..0]	Output		警告燈號輸出

第五章 測試結果與討論

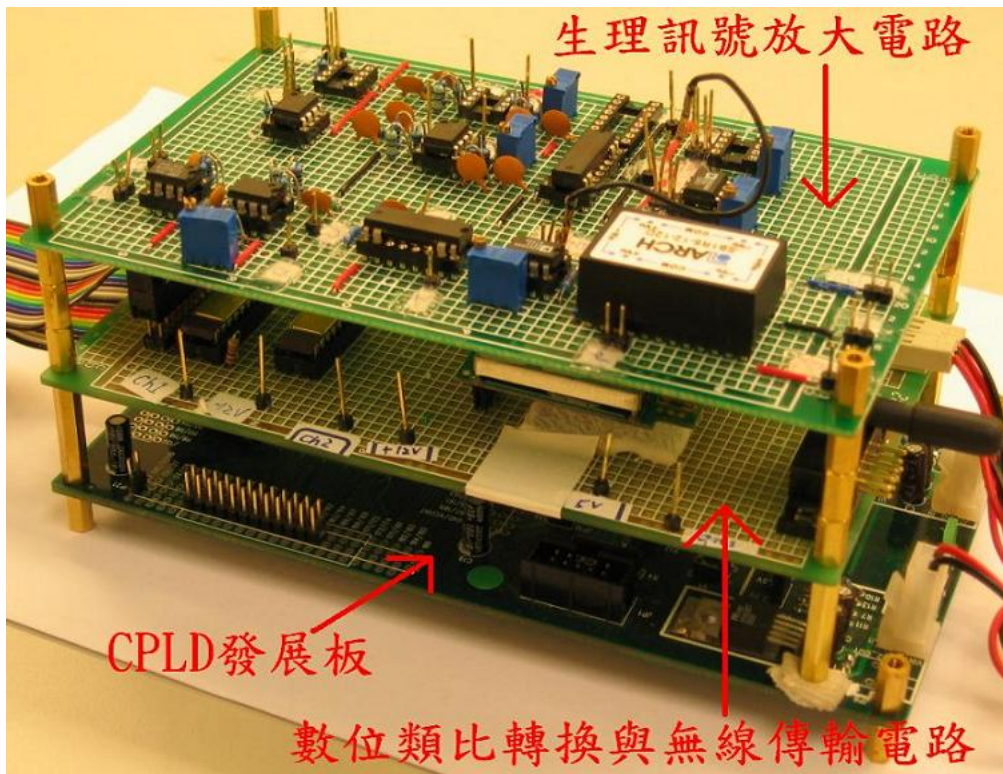
本論文已分別於第二章介紹腦電位訊號感測與無線傳輸系統，第三章介紹嵌入式多工排程系統的設計與開發，並於第四章說明整個汽車駕駛員疲勞狀態偵測與提醒系統實現方法。本章將針對整個系統的實現做一些測試、比較與討論，包括電路測試、腦電位訊號分析方法驗證與比較以及多工排程系統的比較。以下分別就「測試結果」、「系統整合」以及「相關研究比較與討論」進行說明。

5.1 系統測試結果

本節將針對前面幾章所提部份的測試結果做說明與比較，以下分別就「生理量測電路驗證測試」、「腦波訊號處理系統正確性測試」以及「多工排程效能比較」進行說明，並於 5.2 針對系統整合做一說明及討論。本論文所完成的系統如圖 5.1 所示，包括無線電生理訊號量測模組、嵌入式生理訊號分析系統以及顯示介面，細部說明圖則如圖 5.2 所示。



圖 5.1、系統實體圖



(a)



(b)

圖5.2、即時無線生理訊號分析系統系統細部說明圖：

(a)無線生理訊號量測模組、(b)嵌入式系統與區域網路及遠端電腦

5.1.1 生理量測電路驗證測試

本論文所完成的生理訊號量測電路如圖 5.3 所示，本節將針對生理量測放大電路作了一些測試，藉以證明此量測電路可用性。第一項測試為看 α 波，以確保所量得的訊號為腦電波。 α 波在人閉起眼睛休息時，於腦波 8~12Hz 的頻帶會連續出現相同波形，如圖 5.4 圈起來的部分所示，其中(a)為針對腦波頻帶所設計的電路，而(b)則為涵蓋所有電生理訊號的放大電路所測得的波形，而(b)中的訊號比較複雜，但在後處理的時候我們並不會選用到這些頻帶的訊號，因此，在使用上不會有任何影響。另外，也可以從眼動訊號來判別，當眨眼時，肌肉訊號會干擾到腦波訊號，形成很大振幅的雜訊，稱為眼動訊號，如圖 5.5 所示。

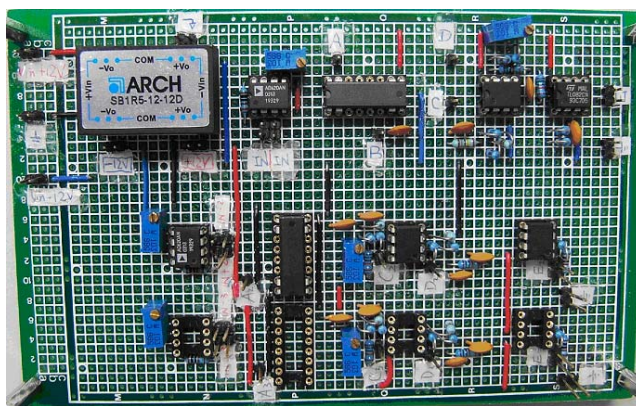


圖5.3、生理訊號放大電路板

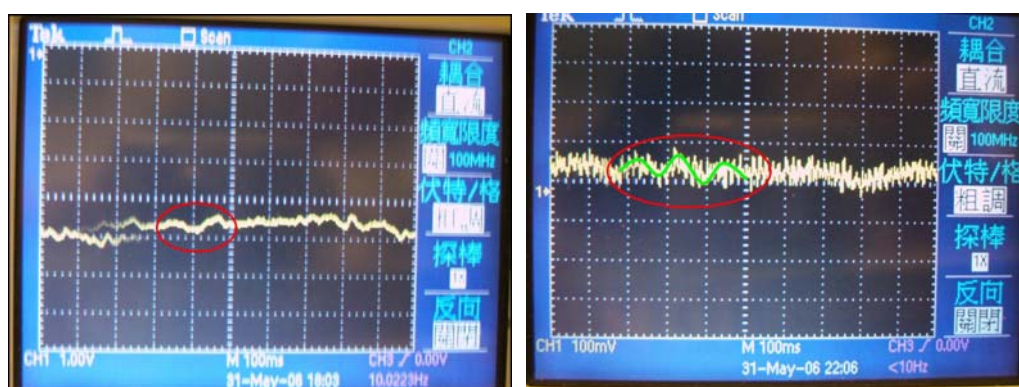


圖5.4、 α 腦電波圖

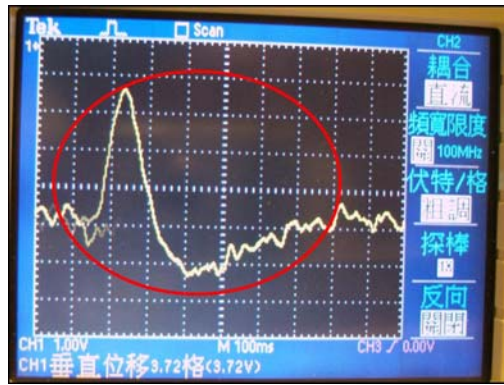


圖5.5、眼動腦電波圖

5.1.2 多工排程效能比較

如果不使用多工排程，如圖 5.6 中(A)所示，其最大的問題就在於如果程序 A 必須每隔一段時間就必須執行一次，換言之，就是執行的時間限制非常嚴格，但由於必須要執行程序 B，所以會造成程序 A 會有延遲的現象產生。解決這個問題的方法有兩個：一是自行規劃與切割，使得程序 B 的執行時間不會干擾到程序 A 的執行頻率，另一個方法則是交由系統做多工排程。如果使用多工排程則可以將原本比較大塊的處理程序分解成許多小塊的處理程序，並將其插入空閒的位置，如圖 5.6 中(B)所示。如此一來，將可以準確的確保程序 A 可以符合其執行的頻率，而這也是我們使用多工排程的原因。因為，我們必須讓接收無線模組訊號的處理程序一定頻率便執行一次，以便確保其取樣頻率的準確度。

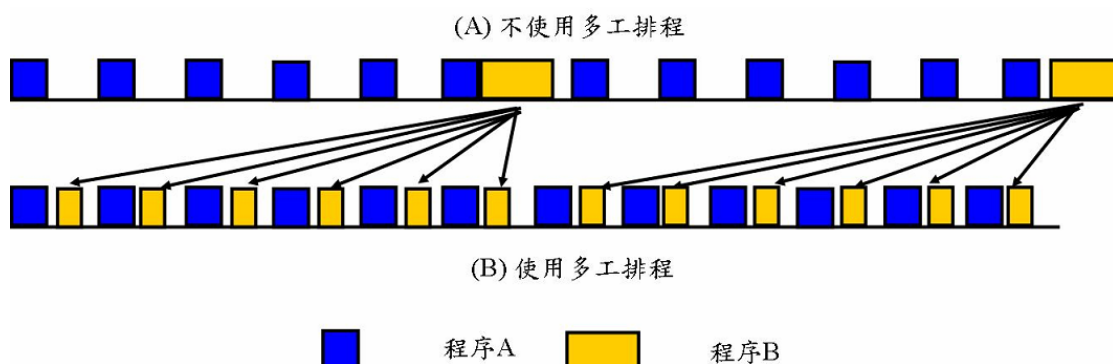


圖5.6、程序排程示意圖

在多工排程的效能評比上，我們以兩個頻道的資料，接收資料的頻率為每個頻道約 65Hz，以接收足夠資料與資料處理為一次完整的流程，執行這個流程 1000

次，分別比較使用多工排程與不使用多工排程兩種方法所需的時間。如表 5.1 所示，不使用多工排程的程式做完 1000 次的資料處理循環需要 40 分 25 秒，而使用多工排程的情形下只需 32 分 13 秒。平均而言，不使用多工排程的程式每個循環需要 2.425 秒，而使用多工排程僅需 1.933 秒。如果扣除資料接收的時間，不使用多工排程的情形下，每做一次循環將會延遲近 0.6 秒，而使用多工排程的話，僅延遲不到 0.1 秒。所以，把時間拉長來看，多工排程的效果會越明顯。

表 5.1、效能比較表

	純資料接收	不使用多工排程	使用多工排程
執行 1000 次 時間	30 分 38 秒 (1838 秒)	40 分 25 秒 (2425 秒)	32 分 13 秒 (1933 秒)
平均一次時間	1.838 秒	2.425 秒	1.933 秒
平均延遲時間	0 秒	0.587 秒	0.095 秒

接著我們來看實際執行的情形，我們擷取使用多工排程與不使用多工排程個一段執行影片，並每三個 frame 擷取一張圖像來觀察。首先，我們觀察剛開始的一段時間，由圖 5.7 以及圖 5.8 可以發現，加入多工排程後執行起來會比較平順。由圖 5.7 看來，後三張圖片幾乎是相同的，而在圖 5.8 則沒有這個現象發生。然而，這也是我們使用多工排程最大的原因，我們希望讓資料接收的頻率固定，而圖看來則是希望看起來較平順，而沒有時間上斷斷續續的問題。

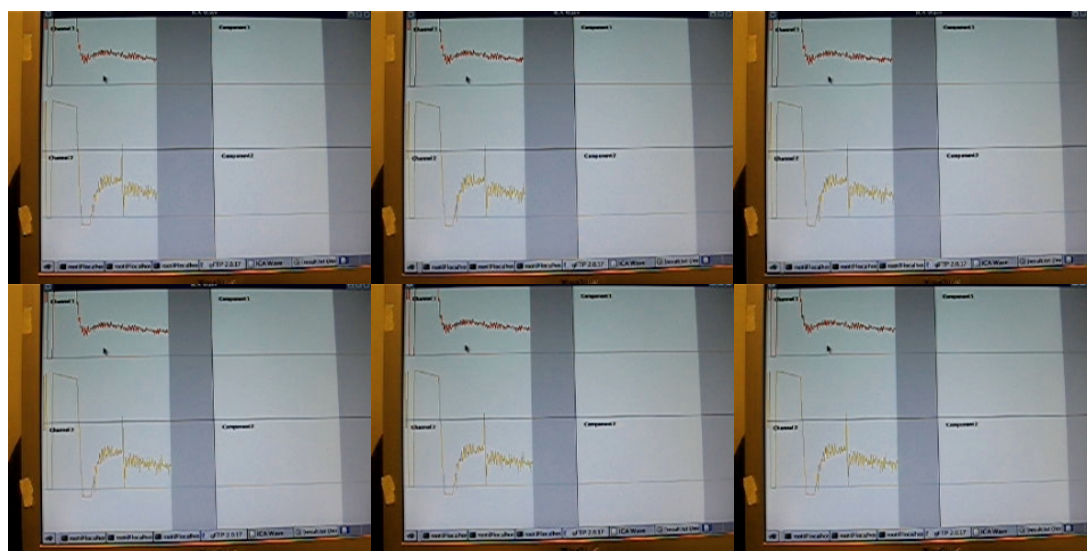


圖5.7、不用多工排程的執行過程

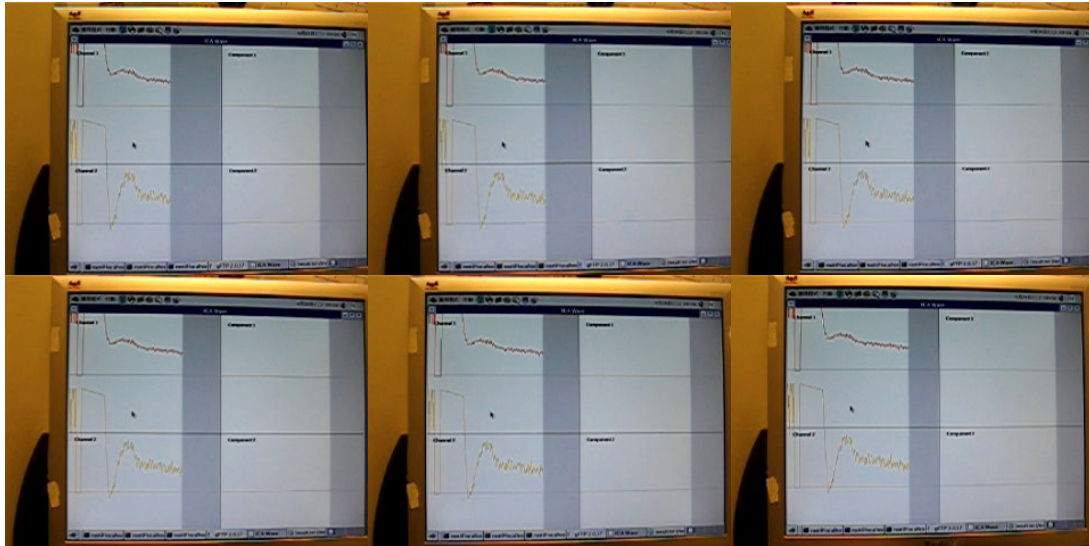


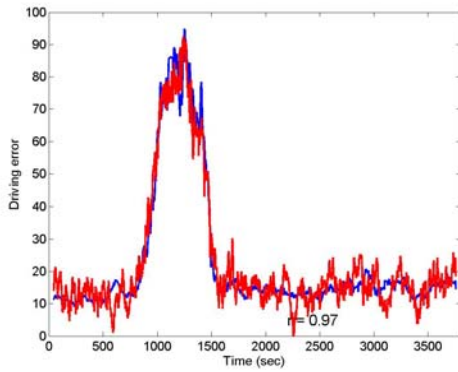
圖5.8、使用多工排程的執行過程

5.1.3 腦波訊號處理系統正確性測試

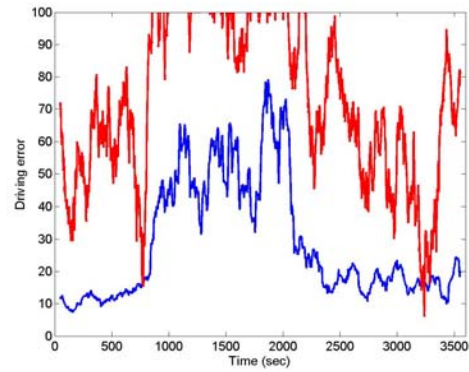
在嵌入式單板實現演算法之前，我們先利用電腦開發及測試我們的演算法。為了測試主成分分析(PCA)的必要性，我們分別針對含 PCA 以及不含 PCA 的演算法來做測試，圖 5.9、5.10 分別是我們針對含、不含 PCA 演算法所做的測試，其中以紅線(利用腦電波所預測的駕駛軌跡)與藍線(實際上駕駛者的駕駛軌跡)的相關程度代表其偵測的準確率。圖 5.9(a)為該受測者在一次實驗中訓練出模型，再用同一筆資料作測試，估測出各時間之駕駛軌跡，而圖 5.9(b)則為將另一天的資料套入圖 5.9(a)所訓練的模型做測試。在效能評比上，我們計算實際駕駛的軌跡誤差與利用腦電波所預測出的駕駛軌跡誤差的相關係數，藉以代表我們所預測駕駛軌跡的準確率，其算式如(6.1)所示：

$$r_{xy} = \frac{s_{xy}}{s_x s_y} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2} \sqrt{\sum (y_i - \bar{y})^2}} \quad (6.1)$$

相關係數 r_{xy} 便代表我們所預測駕駛軌跡的準確率，在這個式子中， x 為實際駕駛軌跡的時間序列， y 則為我們預測駕駛軌跡的時間序列，而 \bar{x} 、 \bar{y} 為 x 與 y 的期望值。如圖 5.9 所示，在這個不含 PCA 的分析與預測演算法中，雖然訓練資料的準確度可以到達 0.97，但其測試的結果(圖 5.9(b))卻不太好。而在加入 PCA 演算法後，我們選擇 PCA 萃取維度為 20 時，其測試的圖形如圖 5.10 所示，雖然訓練的資料僅能到達 0.94 的準確率，但其測試的資料卻可以到達 0.82 的準確率，而由圖形也可看出其改善，其預測的結果比不加入 PCA 的結果好很多，而其曲線也更加近似真實的駕車路線。

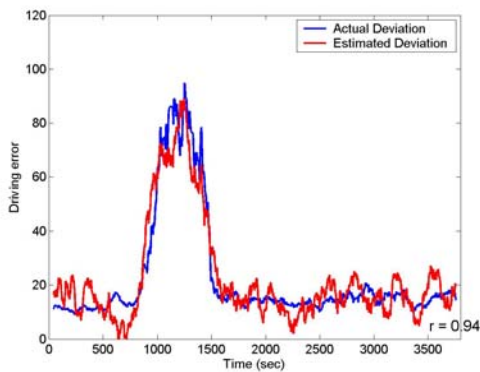


(a)訓練資料

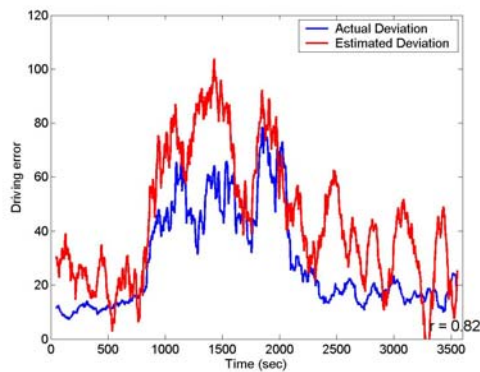


(b)測試資料

圖5.9、不含PCA的演算法測試



(a)訓練資料



(b)測試資料

圖5.10、含PCA的演算法測試

經由在 PC 上的測試，我們證明第四章所提出的即時分析演算法之準確度與可行性。接著，我們將在 OMAP 上作實際驗證，但由於 OMAP 上的分析演算法分布在 DSP 與 ARM 上，所以在驗證上我們分成幾個步驟來做驗證：

首先，我們測試 DSP 上的運算程式以及做 PCA 模型之前的訊號後處理程式，將其運算的結果放入電腦中作為模型訓練的輸入特徵，皆以其訓練出的模型的準確度與電腦上處理的結果做比較。同時，我們針對不同 Average 的方法作測試與比較，其結果比較如表 5.2 所示。整體來說，我們在 OMAP1510 上所實現的訊號處理方法的測試效果都還不錯，而三種方法中又以方法一的效果最好，而其消耗時間也最短。由表中的結果發現，由我們在 OMAP1510 上實現的三種方法所訓練出的模型，再由訓練資料套回去的測試結果似乎都比電腦運算好，但由於這只是訓練的資料，並無法真正代表其成效，要測試資料亦能到達很好的效果才可以下定論。

表 5.2、DSP 處理測試比較表

		資料一	資料二	資料三	平均
電腦運算準確度		0.87	0.89	0.95	0.9033
方法一	準確度	0.96	0.93	0.93	0.94
	消耗時間	39:01	39:05	38:53	39:00
方法二	準確度	0.96	0.94	0.89	0.93
	消耗時間	41:35	41:27	41:09	41:24
方法三	準確度	0.95	0.91	0.91	0.9233
	消耗時間	41:19	41:18	40:53	41:10

接著，我們測試 ARM 上面後處理程式，測試流程如圖 5.11 所示。我們將 DSP 運算以及 ARM 訊號後處理完的資料分別在 ARM 以及 PC 上乘上 PCA 的矩陣，最後在 PC 上做整體的效能評估與驗證，其結果如圖 5.12 所示，綠色的虛線為 ARM 做運算的結果，而紅色的實線則是在 PC 上做運算的結果，而根據圖上顯示，這兩條線是完全重合的。

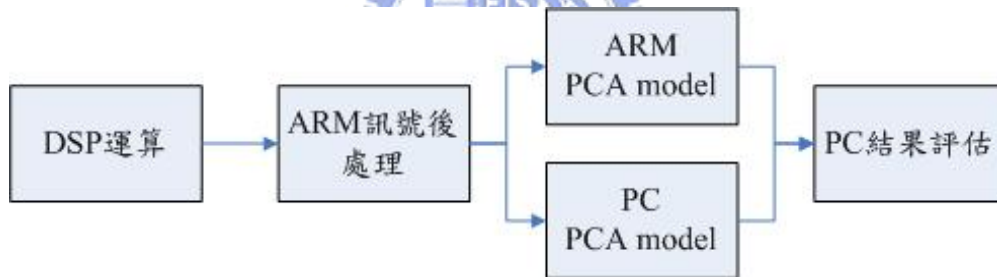


圖5.11、PCA矩陣處理驗證流程

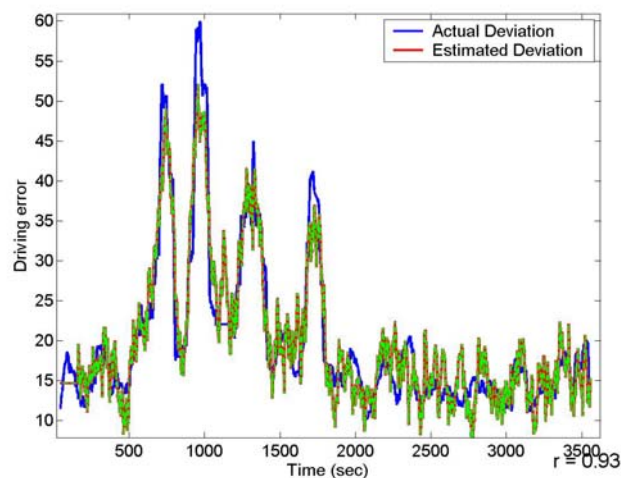
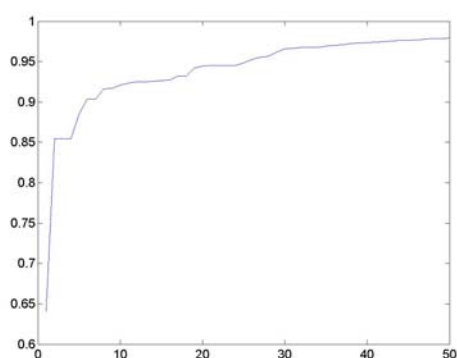


圖5.12、PCA矩陣處理驗證測試

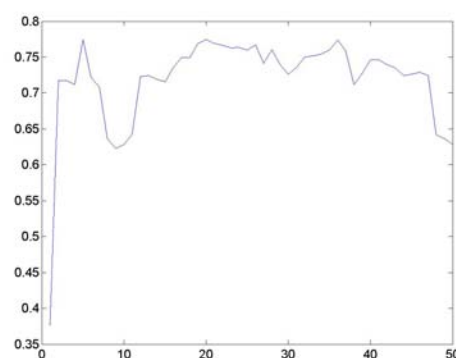
最後，我們使用已經完成量測的兩筆資料放入嵌入式單板作處理，做完處理之後再回到 PC 上做預測演算法的評估。如表 5.3 所示，當 Aveage 使用方法一時，即可達到不錯的效果。其訓練的成果可達 0.8、0.94，而測試成果也可到達 0.85 與 0.77。此外，如圖 5.13 所示我們發現在訓練時，所使用主成分分析的維度高，其訓練的成果也越好，但測試時卻發現，其測試的精準度並不與其維度成正比，而是當其到達最佳值之後就開始往下降。利用方法一的實際訓練與測試圖如圖 5.14 所示，由圖可以看到除了在訓練的時候可以到達 0.97 的準確度外，測試的時候也有 0.77 的準確度，而且其整體來說都相當符合駕車軌跡的趨勢。我們另外針對方法二、方法三做同樣的測試，測試準確度亦可到達六、七成。以方法三在 OMAP 上做預測的測試結果如圖 5.15 所示，雖然圖形並沒有很漂亮，預測的軌跡也沒有很符合實際的駕駛誤差，但是，其於疲勞狀態(藍色駕駛誤差軌跡的峰值)的時候，卻有不錯的預測效果，亦能符合我們系統的需求。

表 5.3、演算法於 OMAP 上測試比較表

	Section 1 預測 Section 2			Section 2 預測 Section 1		
	訓練 樣本	測試 樣本	PCA 萃取維度	訓練 樣本	測試 樣本	PCA 萃取維度
電腦運算	0.93	0.85	12	0.94	0.82	20
方法一	0.8	0.85	6	0.97	0.77	36
方法二	0.85	0.62	21	0.97	0.75	31
方法三	0.83	0.61	18	0.97	0.71	38



(a)訓練資料



(b)測試資料

圖5.13、PCA維度與預測精確度的關係

(水平軸為 PCA 的萃取維度，垂直軸為辨識準確率)

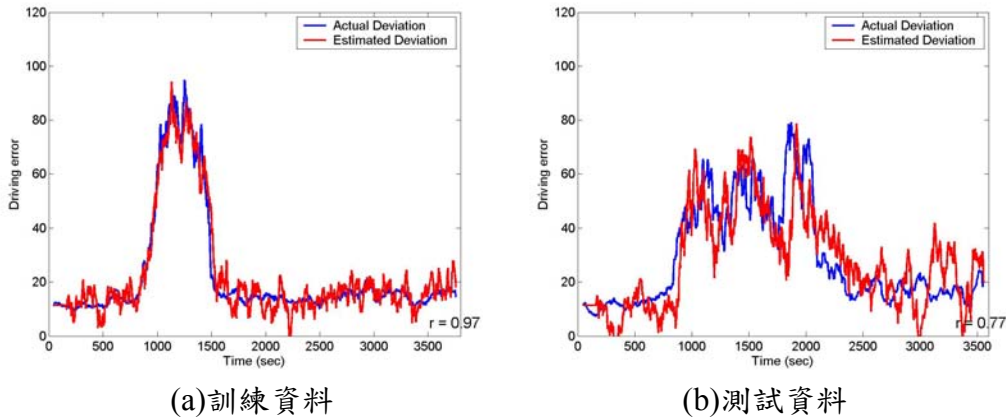


圖 5.14、在OMAP上做系統測試

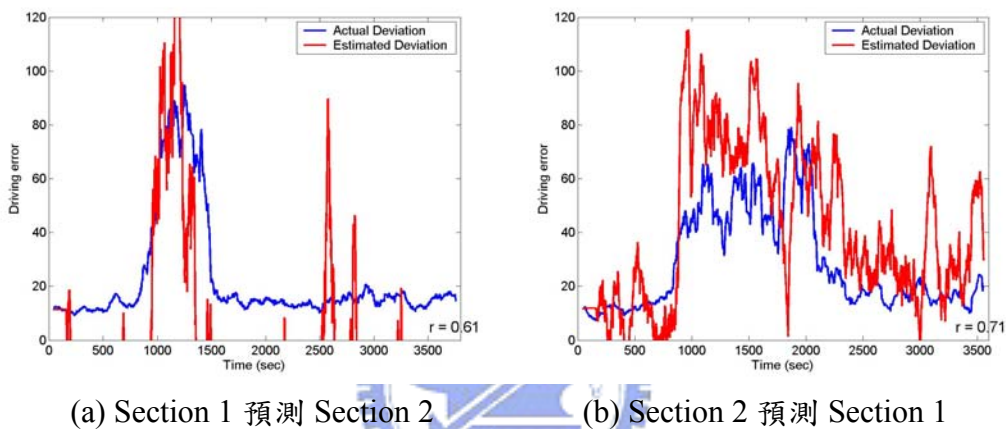


圖 5.15、以方法三在OMAP上做預測測試

5.2 系統整合與討論

系統整合架構圖如圖 5.16 所示，以 OMAP1510 為核心的嵌入式生理訊號分析系統會藉由無線傳輸系統接收即時量測到的電生理訊號做運算與分析，分析的結果會顯示在遠端的電腦上，如有需要提醒駕駛員時，則會有警告聲音發出。顯示介面如圖 5.17 所示，圖中 Frame A 與 Frame B 為原始腦電位訊號的圖形，而 Frame C 以及 Frame D 則是預測的結果。如圖中(a)所示，如果駕駛員的精神狀態很好，右半面的 Frame C 以及 Frame D 是空白的，然而，如果駕駛員的精神狀態不佳，則將在 Frame C 中出現藍色的峰值(peak)，這個峰值代表計算出的駕駛員的駕駛軌跡與車道中心線的誤差，根據第四章的說明，誤差越大則駕駛員的精神狀態越差，其峰值也就越高，如圖中(b)所示。

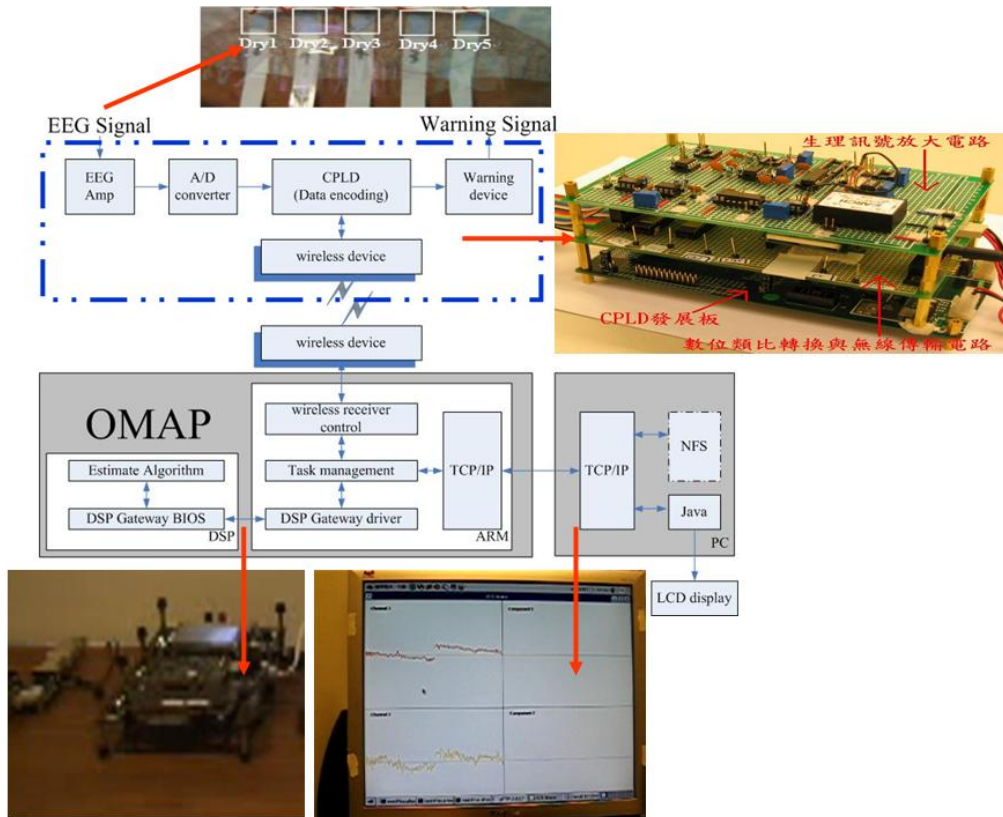


圖5.16、系統整合架構圖

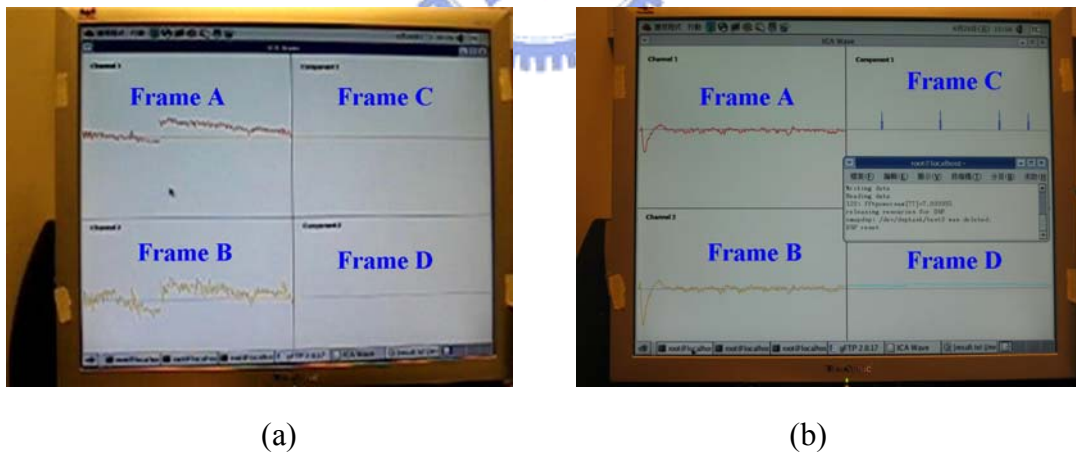


圖5.17、顯示介面圖

因為，疲勞提醒裝置是根據的線性迴歸的輸出值來判斷其受否啟動，在 5.1.3 中以驗證過我們所提的汽車駕駛員疲勞狀態偵測演算法有不錯的準確率，只要設定適當的門檻值，將可使得提醒系統的效能大為提升。

根據無線模組的傳輸速率以及我們所要求的取樣頻率，我們可以求出我們系統所能容忍的最多頻道數。首先，我們以 RF3100/3105 的速率 38400 來作計算，每秒傳送 38400bits 而依照我們每十四個速率傳送一個 byte 的實現方式，每秒約可以傳送 2743bytes，如果系統的取樣頻率訂在 256Hz，系統可以承受 10 個頻道，128Hz 的取樣頻率則是 21 個頻道，64Hz 的取樣頻率將讓系統可以到達 42 個頻道。

我們已於 5.1.2 中比較過使用多工排程與不使用多工排程在執行時間上的差異，由測試結果可以發現在時間效能上，使用多工排程的執行時間與延遲時間都縮短許多。除了縮短時間外，使用多工排程還有以下幾點優點：

1. 使程式模組化：

將程式分離可以縮短程式規劃的時間。此外，由於程式模組化，所以，在開發時間較短，也有助於多種不同功能程式的整合。模組化的程式設計亦有助於程式的再次使用。

2. 讓系統在週邊裝置的控制上更即時：

由於整套資料接收與處理的流程都必須即時，而在資料處理的即時運算上，我們採用 DSP 來做運算克服這個問題。至於在資料接收的即時上，我們以多工排程的方法來克服，以讓整個系統能準確符合資料接收所需要的頻率。

3. 讓系統更有效利用系統資源：

在許多排程的研究上，就是為了能善用系統的資源，本論文所發展的系統亦有這個優點，在實現上可以減少系統等待時間時的浪費。

因此，引入多工排程系統有助於執行時間的控制、開發時間的縮短以系統資源的利用。本論文所實現的無線即時腦機介面系統，已經於 2006 年六月二十四日「榮台聯大（榮總與台灣聯合大學）合作計畫」的研究成果發表會中亮相，亦於 2006 年六月三十日由德州儀器所舉辦的亞洲區 DSP 應用競賽『台灣區分區競賽』中獲得佳作的殊榮。不論從數據、圖片甚至影片都可以觀察到本系統可完整實現我們所預期的功能，也顯示本研究應用於日常生活當中的可行性。

5.3 相關研究比較與討論

我們曾在第一章提過國外也有少數實驗室在做一些可攜式即時生理訊號分析系統的研究，以下針對這些系統的規格與實現與本研究做一些比較，分析比較如表5.4所示，本系統與其他系統的比較分為五點：

1. 生理電訊號的訊號量測電路

如表所示，這幾個關於可攜式的即時生理訊號分析系統所適用的訊號大多是針對單一種訊號設計(如：EEG)，而本論文所實現的生理訊號量測系統可應用於多種不同訊號，例如：EEG、ECG、EMG、EOG等電生理訊號。

2. 無線傳輸系統

在無線傳輸系統方面，由表可知只有C是使用Bluetooth來作為無線傳輸系統的介面，而其他兩種是使用有線的方式來做傳輸。在醫院中，頻帶在2.4GHz的bluetooth是被允許的，但在我們的研究中，我們希望所發展的腦機介面系統可以被廣泛應用在各個環境，所以，我們選用兩種無線傳輸系統來實現，而這也意味著我們所發展的系統將可適用於許多不同規格的無線裝置，這也增加系統的擴充性與適應性。

3. 訊號處理

在系統B中，其重點在強調其所提出的硬體系統，並為針對其分析系統做說明；而在系統C中，則僅做硬體的部份，而並未做到處理系統的實際實現。有實際詳述分析處理過程的只有系統A與本系統，然而在系統A中，除了做頻譜轉換外，僅做簡單的特徵萃取，但在本系統中須做七道手續的訊號處理，較為複雜。

4. 即時性

由上一點可發現，本系統所提的訊號處理系統較為複雜，也因此我們考慮到系統必須要同時接收資料與訊號處理的即時性。所以，本論文提除一套在其他系統中所沒有的多工排程系統來保證資料接收的即時性。

5. 模組化

在本論文的設計中，無論軟體或硬體，我們都考慮到了模組化的設計，盡量使得在系統中的每個模組都是獨立的，在測試與應用上將更加方便，在硬體中，放大器、類比/數位轉換、無線傳輸模組、嵌入式單板以及顯示介面都是可以彼此獨立的。而在軟體中，我們在第四章有提過應用我們所提出的多工排程系統可以讓程式模組化。

相較於其他研究而言，本研究從嵌入式系統以及硬體的觀點出發，提出許多在傳統研究中極少見到的設計方式，以期能加快系統的處理速度、縮短系統的開發時間以及加強系統與其他模組的整合能力。本系統最大的特色除了速度外，就是其模組化的設計將加強其整合更多功能與不同設計的能力。

表 5.4、相關研究比較表

系統	適用訊號	頻道數	無線傳輸	A/D bits	取樣頻率 (Hz)	Gain	Filter	處理單元	軟體處理
A [17]	EEG	2	有線	12	200	--	(軟體) 4~35Hz帶通 50 Hz帶拒	DSP	1. FFT 2. 特徵擷取
B [18]	EEG ECG EOG	8	有線	16	256	--	--	PDA	
C [19]	EEG	6	藍芽	12	10000	10000	60 Hz低通	筆記型電腦	
本系統	EEG EMG ECG EOG	5	藍芽 RF	8	457	5000	1~100Hz帶通 60 Hz帶拒	OMAP 1510	七大訊號處理步驟

第六章 結論與未來展望

本論文提出一套加強嵌入式系統使用效率的多工排程系統來實現無線嵌入式腦機介面系統，並將其應用於即時汽車駕駛員疲勞狀態偵測與提醒。此外，在本論文中我們提出一套即時分析與處理腦電位訊號的演算法，將其針對嵌入式單板的特性，做適當的修改並在嵌入式單板上實現。在傳統的嵌入式系統中都是以 ARM 與 Strong ARM 為處理單元，但由於腦電位訊號的處理上較為複雜，所以，如果僅以 ARM 或 Strong ARM 為處理單元，將會有運算能力不足的問題。因此，我們以具備 ARM 以及 DSP 的雙核心處理器 OMAP1510 來作為處理核心，而實驗結果也發現 OMAP1510 的運算能力足夠滿足我們的需求。而在訊號接收上，為了加強訊號接收的即時性，我們引入了多工排程來加強訊號接收的即時性，而在第六章的測試中，我們也發現訊號接收更為即時。而這兩點也證明了我們所提出與實現的無線嵌入式腦機介面系統具即時訊號接收與處理的能力。

在訊號處理的技術上，我們已在第五章中作測試與驗證。在測試後，我們發現在 OMAP1510 上測試的效果亦能達到七、八成的預測成功率。因此，本論文所實現的系統皆符合我們所要的需求。在實現上，我們亦克服了系統記憶體的問題，在有限的記憶體中做最有效的利用。

雖然，在本論文的許多測試與評比中都顯示我們所完成的即時無線腦機介面系統在各方面都有相當高的水準，有許多的功能都是在相關研究中所沒有的，但是，我們仍有著進步的空間，主要的改善方向為：

1. 系統體積：

本論文實現了一套無線嵌入式腦機介面系統的雛型，但整體的體積還是太過於龐大，不適用於使用著攜帶。無論是生理訊號放大電路或著是嵌入式處理單板都還有縮小的空間，例如：生理訊號放大電路可以使用類比積體電路的方式來實現。而在嵌入式單板上，可以朝縮小體積與降低功率消耗的目標進行改善，以提升其實用價值。

2. 多工排程系統：

在多工排程系統上，本論文提出一套非常好的程式發展平台，有助於程式

的開發，在我們系統中也非常的有效。但是，未來可以在我們所發展的平台
上開發一些新的排程演算法，使其在工作量塞滿處理器時亦能負荷。

3. 周邊控制介面：

在週邊控制上，目前我們主要使用的介面是 UART、網路介面，未來可以使
用 USB 介面，以提升其傳輸速率。

隨著這一套系統的誕生，我們看到了『未來』，也希望有朝一日可以將這套系
統應用於日常生活中，使得未來的行車環境更加安全。本論文讓我們對『科技始
終來自於人性』這句話有更深的體會，因為，科技的進步會讓我們的
生活更加舒適與安全。未來我們將讓這套系統更加人性化，為人類帶來更多
進步與舒適的生活。



參考文獻

- [1] M. Falkenstein, J. Hohnsbein, J. Hoormann, and L. Blanke, "Effects of crossmodal divided attention on late ERP components. II. Error processing in choice reaction tasks," *Electroencephalography and Clinical Neurophysiology*, vol. 78, pp. 447-55, 1991.
- [2] J. Klopp, K. Marinkovic, P. Chauvel, V. Nenov, and E. Halgren, "Early widespread cortical distribution of coherent fusiform face selective activity," *Human Brain Mapping*, vol. 11, pp. 286-293, 2000.
- [3] S. Makeig, and M. Inlow, "Lapses in alertness: Coherence of fluctuations in performance and EEG spectrum," *Electroencephalography and Clinical Neurophysiology*, vol. 86, pp. 23-35, 1993.
- [4] S. Makeig, A. J. Bell, T. P. Jung, T. J. Sejnowski, "Independent Component Analysis of Electroencephalographic Data," *Advances in Neural Information Processing Systems*, vol. 8, pp. 145-51, 1996.
- [5] S. Makeig, S. Enghoff, T. P. Jung, and T. J. Sejnowski, "An efficient basis for brain-actuated control," *IEEE Trans. Rehab. Eng.*, vol. 8, pp. 208-211, 2000.
- [6] S. Makeig, M. Westerfield, T.P. Jung, S. Enghoff, J. Townsend, E. Courchesne, and T. J. Sejnowski, "Electroencephalographic Sources of Visual Evoked Responses," *Science*, vol. 295, pp. 690-94, Jan. 25, 2002.
- [7] 連怡仲, "數位腦波機系統之設計與研製", 國立台灣大學, 碩士論文, 民國八十六年。
- [8] 杜益昌, "多通道腦波機系統之設計與裝置", 國立台灣大學, 碩士論文, 民國八十七年。
- [9] 林伯星, "數位腦波機系統之人機介面設計與研發", 國立台灣大學, 碩士論文, 民國八十七年。
- [10] 蔡進寶, "腦波紀錄儀設計與製作", 國立交通大學, 碩士論文, 民國八十九年。
- [11] 楊勝文, "可攜式智慧型腦波記錄器", 國立台灣大學, 碩士論文, 民國九十年。
- [12] 黃名斌, "USB 介面之模組化腦波記錄儀", 國立中原大學, 碩士論文, 民國九十一年。

- [13] 吳炎法, “研製無線化多通道腦波系統並運 NAB 方法進行腦波分類之研究”, 國立台北科技大學, 碩士論文, 民國九十二年。
- [14] Ming Cheng, Xiaorong Gao, Shangkai Gao, and Dingfeng Xu, “Design and Implementation of a Brain-Computer Interface With High Transfer Rates,” *IEEE Trans. biomedical Eng.*, vol. 49, no. 10, 2002.
- [15] Obermaier B, “Design and implementation of an EEC based virtual keyboard using hidden Markov models”, Ph.D. dissertation, Technical University-Graz, Graz, Austria, 2001.
- [16] Obermaier B, Neuper C, and Pfurtscheller G, “Information transfer rate in a 5 classes brain-computer interface,” *IEEE Trans. Rehab. Eng.*, vol. 9, pp. 283-288, 2001.
- [17] Xiaorong Gao, Dingfeng Xu, Ming Cheng, and Shangkai Gao, “A BCI-Based Environmental Controller for the Motion-Disabled,” *IEEE Trans. Neural Syst. and Rehab. Eng.*, vol. 11, no. 2, pp. 137-140, June 2003.
- [18] G. Edlinger, G. Krausz, F. Laundl, I. Niedermayer, C. Guger, “Architectures of Laboratory-PC and Mobile Pocket PC Brain-Computer Interfaces,” *Proceedings of the 2 International IEEE EMBS Conference on Neural Engineering*, Arlington, Virginia, March 16 - 19, 2005.
- [19] Ashwin K. Whitchurch, B. Han Ashok, R. Vinod Kumar and K. Sarukesi, and Vijay K. Varadan, “Wireless system for long term EEG monitoring of Absence Epilepsy,” *Biomedical Applications of Micro- and Nanoengineering*, Proceedings of SPIE Vol. 4937, pp. 343-349, November 2002.
- [20] Young Han Nam, Zeehun Halm, Young Joon Chee, Kwang Suk Park, “Development of Remote Diagnosis System Integrating Digital Telemetry for Medicine,” *Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 20, no. 3, pp. 1170-1173, 1998.
- [21] W. C. Kao, W. H. Chen, C. K. Yu, C. M. Hong, S. Y. Lin, “A real-time system for portable homecare applications,” *Consumer Electronics, 2005*, pp. 369- 374, June 14-16, 2005.

- [22] F. Lamberti, C. Demartini, "Low-Cost Home Monitoring Using a Java-Based Embedded Computer," *Proceeding of the 4th Annual IEEE Conf. on Information Technology Applications in Biomedicine, UK*, pp 342-345.
- [23] Luigi Bianchi, Fabio Babiloni, Febo Cincotti, Marco Arrivas, Patrizio Bollero, and Maria Grazia Marciani, "Developing Wearable Bio-Feedback Systems: A General-Purpose Platform," *IEEE Trans. Neural Syst. and Rehab. Eng.*, vol. 11, no. 2, pp.117-119, JUNE 2003.
- [24] Piccini, L. Arnone, L. Beverina, F. Cucchi, A. Petrelli, L. Andreoni, G. Bioeng. Dept., Politecnico di Milano, "Wireless DSP architecture for biosignals recording," *Signal Processing and Information Technology, 2004. Proceedings of the Fourth IEEE International Symposium on*, pp. 487- 490, Dec. 18-21, 2004.
- [25] Kondra S, Yew C, Ahmed F, Hofmann UG, "Prototype of a Patient Monitoring Device based on an embedded RISC/DSP system," *ICMP 2005/BMT 2005 conference*, Nuremberg, Germany, September, 2005.
- [26] C. T. Lin, R. C. Wu, T. P. Jung, S. F. Liang, and T. Y. Huang, "Estimating driving performance based on EEG spectrum analysis," *EURASIP Journal on Applied Signal Processing*, vol. 2005, no. 19, pp. 3165-3174, Mar. 2005.
- [27] C. T. Lin, R. C. Wu, S. F. Liang, W. H. Chao, Y. J. Chen, and T. P. Jung, "EEG-based drowsiness estimation for safety driving using independent component analysis," *IEEE Transactions on Circuits and Systems I*, vol. 52, no. 12, pp. 2726-2738, Dec. 2005.
- [28] <http://ibru.vghtpe.gov.tw/chinese/bci.htm>
- [29] <http://faculty.washington.edu/chudler/1020.html>
- [30] 高士政, "以雙核心處理器為基礎之即時嵌入式無線腦波訊號感測與處理系統及其在駕駛員瞌睡偵測之應用", 國立交通大學, 碩士論文, 民國九十四年。
- [31] 謝弘義、張巍瀚、洪精佑、張百毅、徐國凱, "可攜式藍芽無線生理訊號傳輸系統之實現", 國立交通大學, 電機與控制工程學系專題報告, 民國九十四年。
- [32] 禹帆, "無線藍芽技術的深入探討", 文魁資訊股份有限公司, 民國九十一年五月。
- [33] <http://www.getgoal.com.tw/tech/tech-13.htm>

- [34] <http://www.ancher.com.tw/rf3100.html>
- [35] <http://www.ancher.com.tw/rf3105.html>
- [36] 淡江大學智慧型控制實驗室，”An Introduction Altera CPLD/FPGA & VHDL”，民國九十一年。
- [37] 陳慶逸、林柏辰，”VHDL 數位電路實習與專題設計”，文魁資訊股份有限公司，民國九十二年。
- [38] <http://www.slat.org/project/software-map/b1.01/map.html>
- [39] GNU Compiler Collection, <http://gcc.gnu.org>
- [40] Code Composer Studio Getting Started Guide Rev C(SPRU509C), Texas Instruments.
- [41] Toshihiro Kobayashi, Kiyotaka Takahashi, Linux DSP Gateway Specification Rev2.0, Nokia Corporation, November 13 2003.
- [42] W. Richard Stevens, *Advanced Programming in the UNIX Environment*, Addison-Wesley, June 1992.
- [43] <http://web.thu.edu.tw/wenwei/www/cgi/stat/pca/>

