

國立交通大學  
電機與控制工程研究所

碩士論文

基於 DSP 之主動式噪音消除耳機控制器設計與



DSP-based Controller Design and Implementation for  
Active Noise Cancellation Headsets

研究生：吳俊德

指導教授：周志成 博士

中華民國九十三年七月

基於 DSP 之主動式噪音消除耳機控制器設計與實現

DSP-based Controller Design and Implementation for Active  
Noise Cancellation Headsets

研究生：吳 俊 德

Student : Chun-Te, Wu

指導教授：周 志 成 博士

Advisor : Prof. Chi-Cheng, Jou

國立交通大學  
電機與控制工程學系  
碩 士 論 文



A Thesis  
Submitted to Institute of Electrical and Control Engineering  
College of Electrical Engineering and Computer Science  
National Chiao Tung University  
in partial Fulfillment of the Requirements  
for the Degree of Master  
in

Electrical and Control Engineering

July 2004

Hsinchu, Taiwan, Republic of China

中 華 民 國 九 十 三 年 七 月

# 國立交通大學

## 論文口試委員會審定書

本校 電機與控制工程學系碩士班吳俊德君  
所提論文 基於 DSP 之主動式噪音消除耳機控制器設計與實現

DSP-based Controller Design and Implementation  
for Active Noise Cancellation Headsets

合於碩士論文資格水準、業經本委員會評審認可。

口試委員：

郭俊良

余祥華

周志成

胡竹生

指導教授：

周志成  
胡竹生

教授

系主任：

楊谷津

教授

中華民國九十三年 7 月 23 日

# 基於 DSP 之主動式噪音消除耳機控制器設計與實現

研究生：吳 俊 德

指導教授：周 志 成 博士

國立交通大學電機與控制工程研究所碩士班



傳統上許多 ANC 耳機都以類比電路來實現，但此控制器設計的階數非常高導致難以類比控制器直接實現，亦無法實現較複雜的演算法，因此本論文研究以 DSK5510 來發展主動式消音耳機系統，並探討其實現的效能。

論文大致分為兩大部分，第一部分介紹主動式噪音控制的消音原理及其發展歷史，將消音耳機系統的設計架構及限制加以說明，並以最佳化控制的方法設計控制器且模擬其結果。第二部分利用 DSK5510 實現 ANC 耳機系統，表現其下列優點，一、可程式化。二、可實現複雜的演算法。三、可實現高階的控制器。四、具省電性，可應用在可攜式的產品上。

# DSP-based Controller Design and Implementation for Active Noise Cancellation Headsets

Student : Chun-Te, Wu

Advisor : Prof. Chi-Cheng, Jou

Institute of Electrical and Control Engineering

## ABSTRACT

Implementing the controller in an ANC (Active Noise Cancellation) headset using analog circuit has the drawback of parameter matching difficulties. Secondly, it is also hard to implement a high-order controller in analog circuit. When the performance optimization is necessary, the contribution of this thesis is to implement the controller for ANC headset system digitally using the DSK5510 DSP processor and establish an experimental platform which speeds up the development process.

This thesis can be divided into two parts : the first part is a brief introduction of the ANC theorem and its historical review. It also describes the structure, constraints and simulation results of the ANC system designed by the optimal control methods. The 2nd part of the thesis shows the implementation procedure by DSK5510 and establishment of a platform with the advantages: programmable, achieving complex algorithms, high order and low power consumption.

## 誌謝

對於本論文的完成，首先感謝我的指導教授胡竹生博士及周志成博士。自從大三做專題開始，老師不僅在學識上使我獲益匪淺，教導我做學問需有的態度和方法，及正確的研發觀念，亦耳濡目染中的從他身上學習到待人處事的方法，在此，對於老師多年來的教導，誠摯的致上最真誠的謝意。

此外，也感謝余祥華學長的教導，在論文的研究期間，不厭其煩的陪我解決問題，而他對研究那種孜孜不倦的態度，更是使我感受良多。

另外，感謝在這許多個日子裡，有許多人一起陪伴我：感謝价呈兄教導我許多理論的知識外，在研究陷入低潮時，給我不斷的加油打氣，使我順利度過難關；感謝維瀚兄傳授音響電路經驗，使我瞭解許多音響相關知識；感謝常帶麵包請我們的家瑋在 DSP 程式撰寫的協助；感謝宗敏這好男人；還要感謝實作超強的伯彥學長與立偉學長；當然還要感謝常幫大家訂便當的喵喵、嘉芳和憶如，使我們無後顧之憂，還有想賺大錢的 Angel、愛講冷笑話的 pazz、不管何時都穿藍白拖鞋的鳥哥與康康、超會游泳的鏗元和士奇、認真的晏榮和群棋、和實驗室的一姊岑思和興哥，還有感謝在 Matlab 上提供協助的岳峰，常搞笑的邱雄，及在精神上給我支持的雅芝及婉琪，感謝你們在我研究所的生涯中給我許多的歡樂與幫忙，使我在這兩年間留下了美好的回憶。

最後我要感謝家人對我的關心和照顧，母親使我無經濟上的煩惱，尤其特別謝謝姊姊在這兩年父親生病期間擔起照顧的重擔，讓我能專心研究，因為有你們，我才能順利完成學業，最後，在此感謝每一位關心我的人，謝謝你們。

吳俊德

# 目 錄

中文摘要 .....	i
英文摘要 .....	ii
誌謝 .....	iii
目 錄 .....	iv
表 目 錄 .....	vi
圖 目 錄 .....	vii
<b>一、緒論 .....</b>	<b>1</b>
1.1 主動式噪音控制簡介 .....	2
1.2 研究動機與目的 .....	5
1.3 論文內容概述 .....	6
<b>二、系統簡介 .....</b>	<b>7</b>
2.1 數位式主動控制 .....	8
2.2 迴授控制架構 .....	9
2.3 控制器設計限制 .....	10
<b>三、迴授控制器設計與實現 .....</b>	<b>13</b>
3.1 簡介 .....	13
3.2 原理 .....	14
3.2.1 問題描述 .....	14
3.2.2 利用最佳化方法做 loopshaping .....	14
3.3 解的近似 .....	19
3.4 ANC 耳機的迴授控制器設計 .....	20
3.5 CODEC 時間延遲現象 .....	22
3.6 定點數位控制器係數設計 .....	25
<b>四、實驗平台 .....</b>	<b>28</b>
4.1 平台架構說明 .....	28
4.2 測量平台 .....	29
4.3 設計及模擬平台 .....	30
4.4 硬體平台 .....	31
4.4.1 TMS320C5510 .....	33
4.4.2 多通道串列傳輸埠 (Multichannel Buffered Serial Ports, McBSPs) .....	34
4.4.3 AIC23 CODEC .....	35
4.4.4 麥克風放大電路與耳機擴大電路 .....	36
4.4.5 DSP 程式碼架構與流程 .....	37
4.4.6 DSP5510 耗電測試環境 .....	41

<b>五、模擬與實驗結果</b> .....	<b>43</b>
5.1 實驗步驟及環境簡介 .....	43
5.2 四階控制器模擬與實驗結果 .....	44
5.3 八階控制器模擬與實驗結果 .....	48
5.4 耗電實驗測試 .....	52
5.5 模擬去除 CODEC 時間延遲的估測結果 .....	53
5.6 實驗結果之分析與討論 .....	54
<b>六、結論</b> .....	<b>55</b>
<b>參考文獻</b> .....	<b>56</b>
<b>附錄</b> .....	<b>58</b>
A-1 DSP 主程式程式碼 1 (C LANGUAGE) .....	59
A-2 DSP 主程式程式碼 2 (C LANGUAGE) .....	65
A-3 DSP 組語程式碼 .....	66



# 表 目 錄

表 4.4.1. TMS320C5510 特性表.....	33
表 5.4.1. 耗電測試表.....	52



# 圖目錄

圖 1.1.1	波干涉理論 .....	2
圖 1.1.2	PAUL LUEG 的管路主動式消音構想.....	3
圖 1.1.3	OLSON 和 MAY 的電子吸音器架構.....	4
圖 2.1	ANC 耳機.....	7
圖 2.1.1	數位控制系統架構 .....	8
圖 2.2.1	迴授控制架構圖 .....	10
圖 3.2.1	不同限制在 NYQUIST 圖上造成的區域.....	17
圖 3.4.1	四個具代表性之耳機系統的頻率響應 .....	21
圖 3.5.1	C5510DSK ECHO 的頻率響應結果.....	22
圖 3.5.2	邏輯分析儀訊號圖 .....	23
圖 3.5.3	CODEC DOUT 訊號圖 .....	23
圖 3.5.4	CODEC DIN 訊號圖 .....	24
圖 3.5.5	DSK5510 ECHO 測試的結果.....	24
圖 3.5.6	四個具代表性之耳機系統加入 ECHO 時間延遲後的頻率響應 .....	25
圖 3.6.1	控制器係數量化前與量化後的頻率響應比較圖一 .....	26
圖 3.6.2	控制器係數量化前與量化後的頻率響應比較圖二 .....	27
圖 4.1.1	ANC 實驗平台關係圖.....	28
圖 4.2.1	ANC 測量平台 .....	29
圖 4.3.1	ANC 模擬平台 .....	30
圖 4.4.1	DSK5510 實體圖 .....	31
圖 4.4.2	ANC 耳機之系統架構圖.....	32
圖 4.4.3	ANC 耳機之實體圖 .....	32
圖 4.4.4	McBSP 系統方塊圖 .....	35
圖 4.4.5	麥克風放大電路圖 .....	36
圖 4.4.6	耳機擴大電路圖 .....	36
圖 4.4.7	DSP 程式架構圖 .....	37
圖 4.4.8	DSP 運算狀態示意圖一 .....	38
圖 4.4.9	DSP 程式之基本流程圖 .....	39
圖 4.4.10	DSP 運算狀態示意圖二.....	41
圖 4.4.11	DSP 設定時脈之環境.....	41
圖 4.4.12	測量 DSP 晶片總消耗的電流值.....	42
圖 4.4.13	測試 DSP 耗電的實際環境圖 .....	42
圖 5.2.1	四個具代表性之耳機系統加入 ECHO 時間延遲後的頻率響應 .....	44
圖 5.2.2	四階控制器理論的頻率響應圖 .....	45

圖 5.2.3	四階控制器實際的頻率響應圖 .....	45
圖 5.2.4	使用四階控制器的 NYQUIST 圖 .....	46
圖 5.2.5	使用四階控制器 SENSITIVITY FUNCTION 的頻率響應圖 .....	46
圖 5.2.6	使用四階控制器模擬的 SENSITIVITY FUNCTION 頻率響應圖 .....	47
圖 5.2.7	使用四階控制器實際的 SENSITIVITY FUNCTION 頻率響應圖 .....	47
圖 5.3.1	使用八階控制器的 NYQUIST 圖 .....	48
圖 5.3.2	八階控制器理論的頻率響應圖 .....	49
圖 5.3.3	八階控制器實際的頻率響應圖 .....	49
圖 5.3.4	使用八階控制器 SENSITIVITY FUNCTION 的頻率響應圖 .....	50
圖 5.3.5	使用八階控制器模擬的 SENSITIVITY FUNCTION 頻率響應圖 .....	50
圖 5.3.6	使用八階控制器實際的 SENSITIVITY FUNCTION 頻率響應圖 .....	51
圖 5.4.1	DSP/BIOS 的 STS 工具測得 DSP 處理時間 .....	52
圖 5.5.1	使用八階控制器的 NYQUIST 圖 .....	53
圖 5.5.2	使用八階控制器 SENSITIVITY FUNCTION 的頻率響應圖 .....	53



## 一、緒論

經濟發展至目前的時代，人們漸漸注重生活品質，對電子產品不再一昧的追求運算速度或功能的多少，關心的反而是產品的功能是否符合自身的需求及對健康的幫助。

隨著工商業的進步，大量的機械設備如汽、機車、冷氣機充滿在我們四周，因此在我們的生活環境產生了許多的噪音，這些噪音雖不會立即的危害到我們的身心，但是長久暴露在噪音的環境之下均會使我們的生理與心理受到影響，除了造成暫時的不悅和不便外，長久處於高噪音的環境下，人的聽力會受到永久性的傷害，所以民國 88 年和 89 年政府制訂了噪音管制法與機動車輛管制辦法，也足以顯現政府對噪音問題的重視。

由於噪音問題越來越受到人們的重視，因此噪音控制的研究也陸續的被發展。在噪音的控制上，主要可分為兩種不同的方法：被動式 (Passive) 與主動式 (Active)。被動式噪音控制主要是利用吸音材 (sound absorbent materials) 吸收能力來達成消音的目的。這種方法對高頻噪音有很好的效果;然而對低頻的噪音而言，想達到一定程度的消音所需的吸音材往往是體積龐大且費用高昂。這使得被動式消音對低頻噪音的控制顯得不實際且不符經濟效益。因為此兩種方法具有互補性，工程師可因應不同的噪音環境來決定其消音的方法。

由於目前 IC 技術的突飛猛進，發展了許多電子產品，使得主動式噪音控制越容易實現，其製造的成本也越來越便宜，可見在不久的未來，許多電子產品會陸續加入其功能，將帶給人們更好的生活品質。

## 1.1 主動式噪音控制簡介

主動式噪音控制 (Active Noise Control, 簡稱 ANC) 於 1933 由 Paul Lueg 首先提出。Lueg 提出兩個實現主動式消音的理論：「破壞性干涉消音原理」及「主動式吸音原理」。所謂的破壞性干涉是聲波疊加原理的應用，利用喇叭產生一聲場與原噪音產生破壞性干涉以達到消音的目的。主動式吸音原理是使喇叭震動的动作與空氣一致，將聲壓力量承接吸收下來，造成喇叭附近區域的消音。而主動式的噪音控制對消除低頻的噪音有較佳的效果，剛好與被動式的消音有互補的特性。

但主動式的噪音控制只有在特定區域達成，並非所有區域能以主動式噪音控制達成消音目的。在西元 1690 年，Huygens 對波在三度空間的干涉提出了一個通用性的理論(Huygens Principle)[1]，他的理論是說：

“給定一個主要聲源 (primary source)  $S_P$ ，他會在空間  $V$  中建立一個聲場  $F_P$ ，同時指定一個表面  $Z_0$ ，圍繞 並且將空間  $V$  分割成在  $Z_0$  內部的  $V_1$  以及在  $Z_0$  外部的  $V_2$  兩部分，如圖 1.1.1.(a) 所示；那麼會存在一組第二聲源 (secondary sources)  $S_H$  經適當排置在  $Z_0$  上以後產生聲場  $F_H$ ，使得  $F_H$  在  $V_1$  是 0 而在  $V_2$  等於  $F_P$ ，如圖 1.1.1 (b)。因此這組聲源可取代主要聲源在  $V_2$  的聲場而沒有任何差別。現在我們這組聲源變號且令其為  $S_S$ ，則由於線性的關係，這組聲源會產生聲場  $F_S = -F_H$ ，在  $V_2$  中則為  $-F_P$ 。若讓聲源  $S_P$  和  $S_S$  互相干涉，則所產生的聲場  $F_C = F_P + F_S$  在  $V_1$  是  $F_P + 0$ ，在  $V_2$  則為  $F_P + (-F_P) = 0$ 。

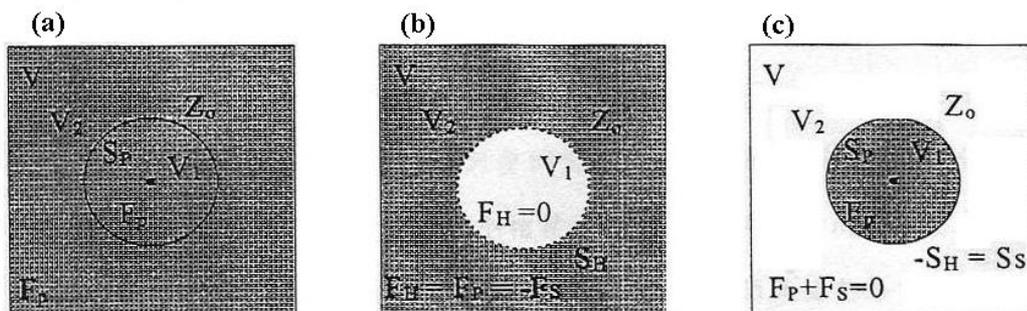
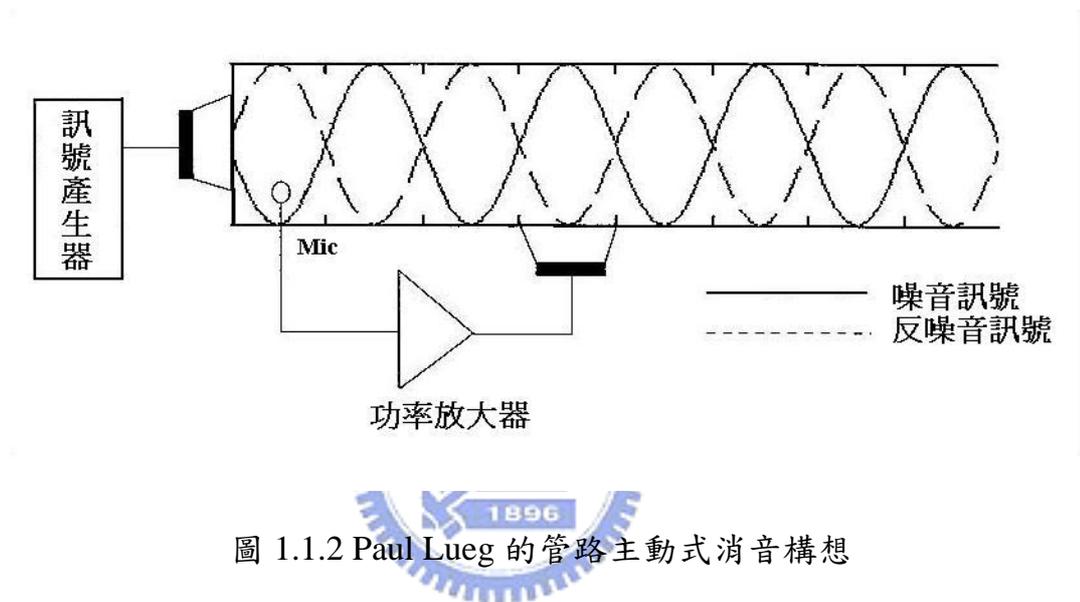


圖 1.1.1 (a)主要聲源聲場 (b)第二聲源聲場 (c)干涉結果

所以基本上主動式噪音控制是藉由引入第二聲源來改變在傳遞介質中聲場的能量分佈，由上述原理可知噪音在某些區域降低，在其他區域則會被加強，這也是為何主動式噪音控制只能是區域消音的原因。

主動式噪音控制的概念最早由 Paul Lueg 提出，並且於 1936 年獲得美國專利 [2]，其系統架構如圖 1.1.2



他所持的理由是：聲波的傳遞速度遠低於電子訊號的速度。所以當聲波從偵測點傳送到消音點時，將有足夠的時間使用一些控制架構對噪音訊號加以控制。但 Lueg 到最後沒有成功，因為他忽略了消音喇叭發出的聲波，會向上游傳遞到偵測點的麥克風處（即聲場迴授（Acoustic feedback）現象），所以麥克風收到的並不是純粹的噪音訊號，這樣會使得喇叭到麥克風形成一個迴授路徑，因此必須謹慎處理系統的穩定問題。而以這種方式的控制架構歸類為前饋控制。

在沈寂了二十年後，ANC 才開始有突破性的發展：Olson 和 May[3][4]在 1953 年引入了”電子吸音器的觀念”，其架構如下

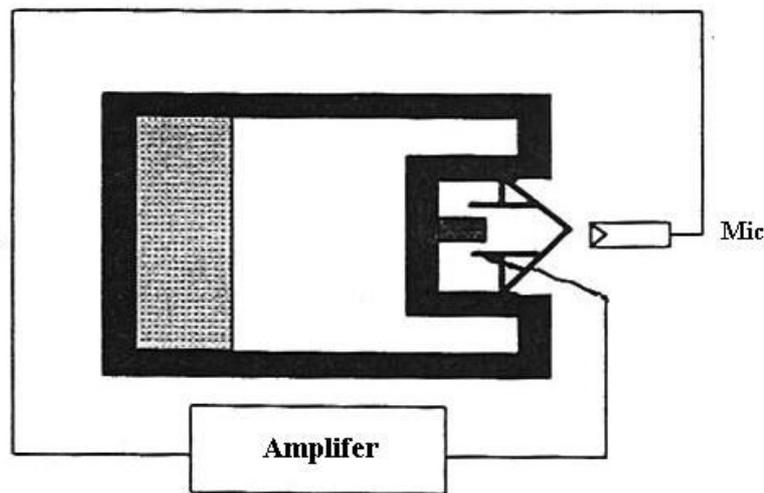


圖 1.1.3 Olson 和 May 的電子吸音器架構

其控制原理為利用一麥克風偵測噪音訊號，經過一控制器後，再由喇叭輸出，明顯的此為一迴授架構的系統。這種架構可達到區域性的寬頻消音效果，也為今日噪音回饋控制奠下了基礎。

主動式噪音消除的觀念雖萌芽於半世紀之前，但因牽扯許多領域的知識及技術，例：噪音傳遞升學響應特性，電子元件特性、寬頻噪音源等問題，都攸關整個系統效能的表現，受限於當時技術，未能有突破性的進展。而近幾十年來，由於相關科技的進步，例如：控制系統技術的改善、適應理論、數位訊號處理理論的快速進步以及固態電子的研究...等。這些技術的快速發展，使得主動式噪音消除的可行性大為增加。因此發展出各種產品，例：消音耳機，房車內的靜音裝置，戰鬥機上的通訊設備等等。

## 1.2 研究動機與目的

經濟發展至今，勞工對工作權力的意識抬高，人們追求更好的生活品質，噪音問題已受到大眾普遍的重視，人人都希望自己擁有一個安靜舒適的生活環境。因此，解決噪音問題已成為一個日益重要的問題。

解決噪音問題一般可由三個方面加以改善

- (一) 噪音源(source)：藉由改變機械結構或材質直接減少震動及減低噪音的產生。
- (二) 噪音的傳遞路徑(path)：改變噪音的傳遞環境或阻絕噪音傳播。
- (三) 噪音接受者(receiver)：改變接受者的位置或配戴聽覺保護系統。

由於噪音源和傳遞路徑其影響原因太多並不容易解決，有時必須由噪音接受者著手，主動式噪音消除耳機即是一種方式，其可在吵雜環境中提供使用者寧靜的環境，並可保護使用者的聽力，例：充滿噪音的工廠環境、直昇機的乘坐環境等等。

而在傳統上，許多 ANC 耳機都以類比電路來實現，不同的耳機需依其規格來設計其控制器，但此控制器設計的階數非常高導致難以類比控制器直接實現，亦無法實現較複雜的演算法。近年來因半導體產業的進步，數位控制器在成本和速度上均進展許多，以致於許多產品都已利用數位電路來實現。

所以本論文主要研究以 DSP (Digital Signal Processing) 晶片發展平台建立一個及時性的主動式噪音消除系統，探討電路數位化的優點，並實際測試其實現的效能。

### 1.3 論文內容概述

本論文內容共分為六章：

第一章：介紹研究動機與研究目標

第二章：闡述系統架構及原理

第三章：說明 ANC 演算法及控制器係數設計方法

第四章：模擬平台及硬體實驗平台介紹

第五章：實驗結果與程式執行效能評估

第六章：結論與未來展望



## 二、系統簡介

本論文所採用的主動式噪音消除耳機裝置類似 Harry Olson 和 Event May 提出的電子消音器 (electronic soundabsorber) [3][4]裝置，如圖 2.1，將一麥克風置於接近耳機處，當外界發生噪音時，由麥克風偵測到噪音訊號後，經一設計的控制器將麥克風的電子訊號處理後迴授到耳機，以控制器輸出訊號，驅動耳機喇叭發出另一聲源，和原噪音形成破壞性干涉而使噪音降低，以達成在麥克風附近形成一消音區域 (zone of quiet) 的目的，這對於區域性的消音十分的有效，但此消音區域是以麥克風為中心，噪音波長  $1/10$  為半徑的區域內，因此消音區域大小和噪音頻率的高低有關，噪音為低頻時因消音區域大而使得耳機配戴者能感受到消音效果，而對高頻噪音而言即使可以使用主動控制方法將噪音降低，但也只能在麥克風附近產生一極小的消音區域，因此當對高頻噪音所造成的消音區域可能其範圍不含蓋耳機配戴著的耳朵，而使配戴耳機者無法感受到消音效果，因此設計時以消除低頻噪音為目標。

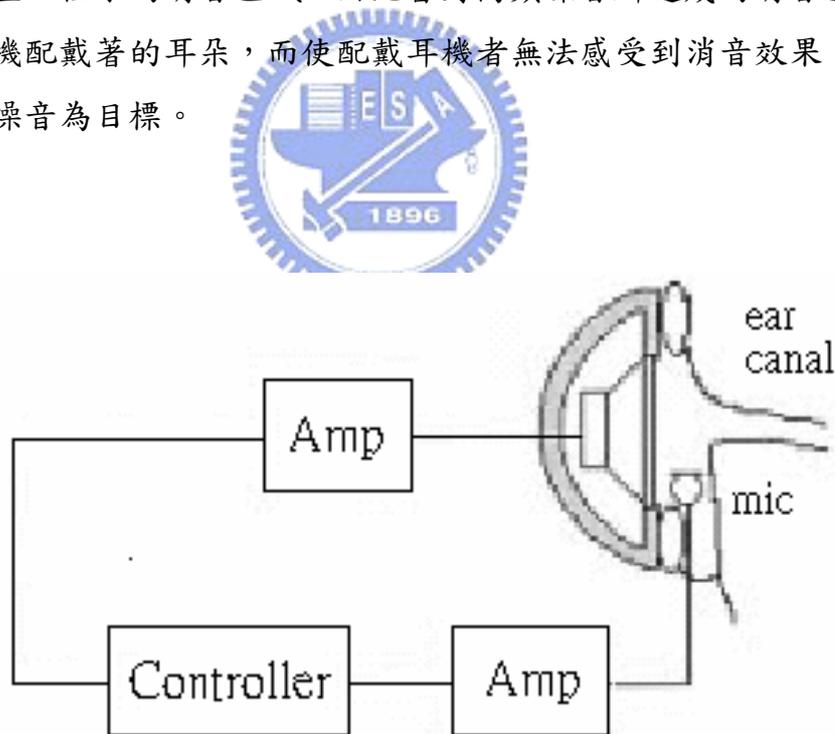


圖 2.1 ANC 耳機

## 2.1 數位式主動控制

本論文採用數位電路方式製作主動式噪音消除耳機之控制器，其架構如圖 2.1.1。

在製作主動式噪音消除耳機控制器時，因採用回授控制的架構，必須考慮到系統穩定度問題，並且時間延遲會造成相位落後（phase lag），相位落後量會隨著頻率增加而增加，當相位落後達  $180^\circ$  時，會使閉迴路系統必須考慮其穩定度問題，因此相位落後會影響到控制的頻寬，進而使噪音控制的效能受到影響。

時間延遲可分成受控體（耳機、麥克風）及控制器兩大部分。而控制器產生時間延遲的產生因素有二：

- 一、 數位控制器本身存在著計算量的時間延遲，其大小與取樣頻率、取樣時間和計算量的多寡有關。
- 二、 製作數位控制系統（如圖 2.1.1）時，數位訊號處理器需在輸入加上一個反失真濾波器（anti-aliasing）和輸出加上一個平滑濾波器（smoothing filter），這兩個濾波器都會使系統產生群組延遲（group delay）。

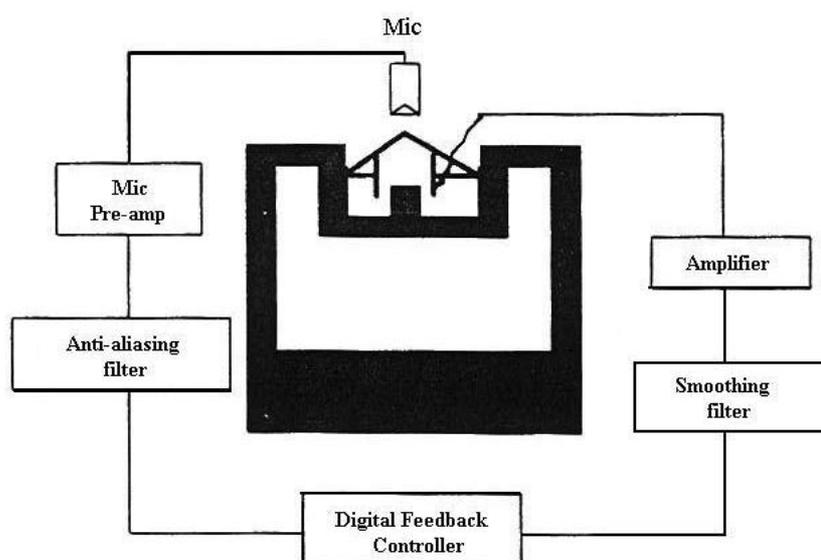


圖 2.1.1 數位控制系統架構

由於以前數位電路運算速率不夠快造成控制器產生時間延遲太大，使得傳統上製作主動式噪音消除耳機大都採用類比電路方式之控制器。但以類比電路實現控制器時卻會產生控制器的準確度考量及控制器參數飄移的問題，因此無法實現高階的控制器。

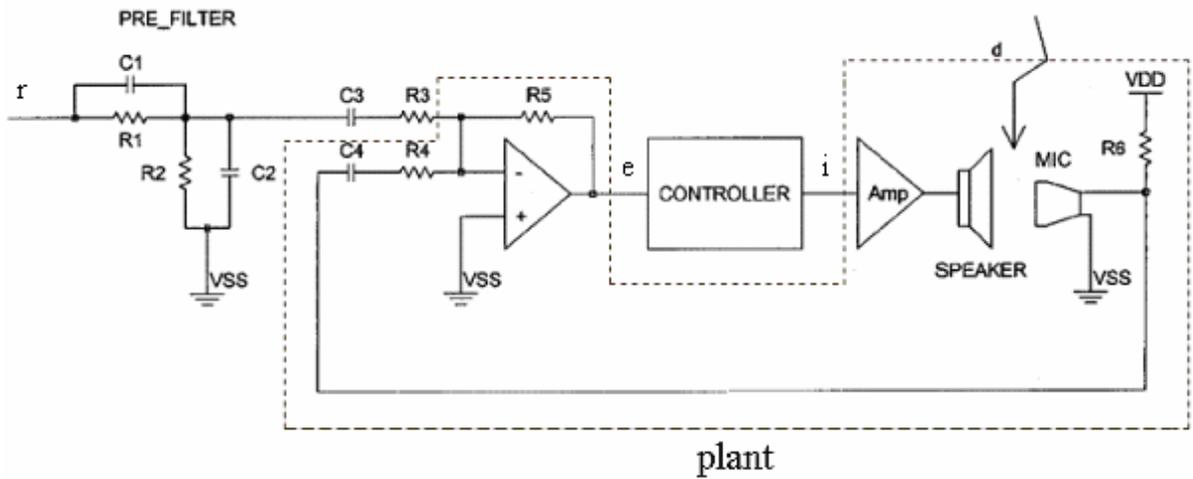
近年來因半導體產業的進步，數位控制器在運算速度上進展許多，隨著取樣頻率的增高，電路造成時間延遲越來越少，且控制器數位化後可具有下列幾項優點。一、可程式化。二、可實現複雜的演算法。三、可實現高階的控制器。四、具省電性。

另外我們在設計數位控制器係數方面，是先求得連續時間控制器的係數，轉換成數位控制器的係數，之後再量化成定點數位訊號處理器（Fixed-point DSP）的係數，所以此論文演算法的控制器大部分都會以連續時間控制器來表示。

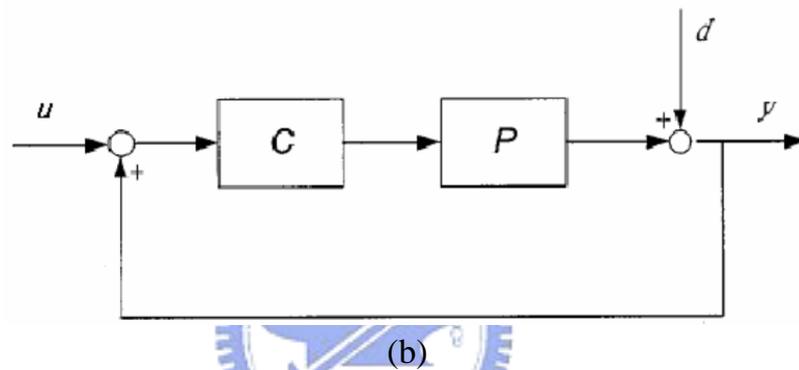
## 2.2 迴授控制架構



本論文採用迴授控制的架構，以麥克風偵測環境噪音，而以耳機喇叭產生另一個聲源，和環境噪音產生破壞性干涉。因此定義受控體為進入喇叭擴大器的訊號到麥克風放大輸出的訊號之間的關係（令其為  $P(s)$ ），其包含了喇叭、麥克風、放大器及環境的聲學阻抗（acoustic impedance）等響應。而迴授控制就是把麥克風收到的訊號經過控制器  $C(s)$  後，和喇叭輸入相加在輸入到喇叭，其結構圖如圖 2.2.1，在此不考慮喇叭的輸入訊號，所以喇叭的輸入就為控制器的輸出。



(a)



(b)

圖 2.2.1 迴授控制架構圖

由圖 2.2.1 迴授控制架構圖中可以看出，從外界環境噪音到麥克風感測訊號經放大器輸出間的轉移函數，由 Mason's Gain Formula 可得

$$\frac{Y(S)}{D(S)} = \frac{1}{1 - P(S)C(S)} \quad (2.2.1)$$

### 2.3 控制器設計限制

如果受控體的頻率響應是平坦而且沒有相位落後的話，那控制器的設計就非常容易，只要一個簡單的 P 控制器 ( $C(s) = K$ ) 即可，而且因為沒有相位延遲，所以 K 任意增加而不必擔心穩定度問題，如此噪音可以任意降低且消音可在所有頻帶都達成。

但不幸的，在真實的世界中不可能找到頻率響應平坦而且沒有相位落後的系統，而相位落後達  $180^0$  時將可能導至系統不穩定。因此控制器的第一個規格即是奈氏穩定守則（Nyquist stable criterion）。

最終的目的是讓噪音  $D(s)$  減小，由式 (2.2.1) 可知控制器  $C(s)$  的設計要使回路增益（loop gain 即  $P(s)C(s)$ ）在穩定的條件下越大越好。

由上面分析可得控制器  $C(s)$  的設計要滿足

$$\begin{aligned} & \underset{C \in RH_{\infty}}{\text{Max}} |1 + P(s)C(s)| \\ & \text{subject to } \frac{1}{1 + P(s)C(s)} \in RH_{\infty} \end{aligned}$$

主動式噪音控制技術對低頻帶噪音消除有較好的效果，而在高頻帶被動式的消音法有較好的表現。而且如前面所述所產生的消音區域大約是噪音波長的十分之一[5]。在 J.F.Abboot 1993 年所提出[6]，在封閉空間中以 1000Hz 的噪音為例，一個喇叭作控制僅能產生一個半徑一英吋的消音區域。而消音區域隨噪音頻率增加而減少，所設計的噪音消除耳機的麥克風到耳朵間的距離約 2 公分，所以耳機的消音效果只能在低頻帶處（約 1000Hz）耳機配戴著所感受到。因此控制器的設計目的希望使耳機能在低頻處消音（約 1000Hz 內），而在高頻不會使噪音增長。但是事實上希望噪音在高頻處都不增長是不可能的，因為受控體有非極小相位零點（non-minimum phase zero），因此如水床效應（waterbed effect）所述 [7][8][9]：

‘The latter situation might be compared to a waterbed : As | S | is push down on one frequency rang , it pops up somewhere else . It turn out that non-minimum phase plants exhibit the waterbed effect ’[7]

所以開迴路系統在低頻處能提高增益以使低頻消音效果佳，但為使系統穩定且不會在高頻處使噪音提高，因此在相位交越附近能使系統增益盡量下降。以保持系統穩定並使噪音不會有增長的現象產生。



### 三、迴授控制器設計與實現

在第二章中，詳細說明了主動式噪音消音耳機系統及原理，在本章中將引用 Shiang-Hwua Yu 及 Jwu-Sheng Hu 提出的最佳化控制方法[10]設計迴授控制器。由於上述提出的方法是用來設計類比控制器，所以必須做些調整來設計數位控制器。

此論文以 DSK5510 實現控制器，因為 DSK 上的 CODEC 產生嚴重的時間延遲，所以設計時需將時間延遲考慮其內。另外在數位控制器係數設計方面，我們先求得連續時間控制器的係數，轉換成數位控制器的係數，之後再量化成定點數位訊號處理器的係數。

#### 3.1 簡介



大部分的 ANC 控制器的設計都是對其受控體做估測成一模型 (Model)，經常以轉移函數 (transform function) 或動態方程式 (dynamical Equation) 表示，再利用各種控制理論對受控體模型設計控制器。因為 ANC 受控體有相當大的頻寬，導致難以估測出與實際系統相當符合的模型。

在利用一些控制理論設計控制器時，像是  $H_{\infty}$  最佳化控制，經常設計出非常高階的控制器而難以實現，所以必須對受控體模型作降階。因此在設計控制器的流程中，必須經過許多次的估測及近似，以致於無法實現當初理想設計的控制。

而 Shiang-Hwua Yu 及 Jwu-Sheng Hu 提出直接利用系統的頻率響應設計出固定階數控制器方法[10]，將所有設計的條件轉換到頻譜，設計 ANC 控制器複雜的步驟變成解一最佳化的問題，再利用電腦計算符合其條件的控制器，如此避免了估測及近似的誤差，也簡化了設計的步驟。

## 3.2 原理

### 3.2.1 問題描述

考慮一典型的 SISO 迴授控制系統，如圖 2.2.1， $C$  代表控制器， $P$  代表受控體， $P \in \mathbf{P}$ ， $\mathbf{P}$  為 LTI 穩定系統的集合，主要目的為在音頻內設計一  $n$  階的控制器  $C$ ，使得在所有的  $P \in \mathbf{P}$  的情形下，其閉迴路系統都為一穩定系統。

由於耳機因配戴的位置不同而會產生不同的受控體  $P$ ，用  $\mathbf{P}$  表示受控體所有可能的情形，在此並不對受控體做模型估測，而是直接使用受控體的頻率響應來設計控制器。

設計步驟一開始先將所有設計的規格轉換成頻譜上的限制條件，在複數平面 (complex plane) 上則可形成一限制的區域，之後利用最佳化的方法找出一  $n$  階控制器使得  $\mathbf{P}$  造成的開迴路的頻譜響應都能落在上述限制的區域內，並盡可能的達到最大的噪音消除效果。

### 3.2.2 利用最佳化方法做 loopshaping

假設已知一受控體所有的頻率響應，設計一控制器使得其系統有強健的效能及穩定度。為了方便符號表示，將受控體所有的頻率響應  $\{P\}$  表示成  $\{P_k, k=1,2,\dots,m\}$  如圖 3.4.1 所示。

$n$  階控制器的參數化：因為要藉由電腦程式計算出控制器，先將控制器參數化，如此所有設計的規格即可以用目標函數及限制函數表示。首先，一  $n$  階控制器表示如下

$$C(s) = \frac{b_0 s^n + b_1 s^{n-1} + \dots + b_n}{s^n + a_1 s^{n-1} + \dots + a_n} \quad (3.2.1)$$

則控制器頻率響應為

$$C(\mathbf{x}, \omega) = \frac{\mathbf{W}^H(\omega)\mathbf{b}}{1 + \mathbf{V}^H(\omega)\mathbf{a}} \quad (3.2.2)$$

其

$$\mathbf{a} = [a_1 \quad a_2 \quad a_3 \cdots a_n]^T$$

$$\mathbf{b} = [b_1 \quad b_2 \quad b_3 \cdots b_n]^T$$

$$\mathbf{W}(\omega) = [1 \quad (j\omega)^{-1} \quad (j\omega)^{-2} \cdots (j\omega)^{-n}]^H$$

$$\mathbf{V}(\omega) = [(j\omega)^{-1} \quad (j\omega)^{-2} \cdots (j\omega)^{-n}]^H$$

符號<sup>H</sup> 代表對矩陣做複數共軛轉置運算。此時，可將一 n 階控制器參數化成一向量  $\mathbf{X} = [\mathbf{a}^T \quad \mathbf{b}^T]^T \in \mathbf{R}^{2n+1}$ 。因此，對一受控體  $P$  而言，其開回路轉移函數的頻率響應為

$$L(\mathbf{x}, \omega) = P(\omega)C(\mathbf{x}, \omega) \quad (3.2.3)$$

如此可在轉換設計規格到頻域限制開回路轉移函數  $L$  決定出理想的參數向量  $\mathbf{x}$ 。

*設計一穩定的控制器*:在此基於下列理由，將設計的控制器形式限制在 n 階實數有理數穩定控制器。一是不穩定的控制器對於雜訊及系統的變化較敏感，在實際應用的成效並不容易滿意。此外，穩定的控制器較也比較容易實現。為了確定控制器的穩定性，控制器極點的實部必須是負數。假設 n 階控制器的分母為  $s^n + a_1s^{n-1} + \cdots + a_n$ 。根據 Lienard-Chipart 準則（參考 Parks and Hahn[11]），當符合

下式時控制器一定穩定

$$(C1) \quad \begin{cases} a_n > 0, a_{n-2} > 0, a_{n-4} > 0, \dots, a_i > 0 \\ D_1 > 0, D_3 > 0, D_5 > 0, \dots, D_k > 0 \end{cases} \quad (3.2.4)$$

在此，當 n 為偶數時， $i=0, k=n-1$ ；當 n 為奇數時， $i=1, k=n$ ；且  $D_1, D_3, \dots, D_k$  為

Hurwitz 行列式，其形式如下

$$D_1 = |a_1|$$

$$D_3 = \begin{vmatrix} a_1 & a_3 & a_5 \\ 1 & a_2 & a_4 \\ 0 & a_1 & a_3 \end{vmatrix}$$

$$D_5 = \begin{vmatrix} a_1 & a_3 & a_5 & a_7 & a_9 \\ 1 & a_2 & a_4 & a_6 & a_8 \\ 0 & a_1 & a_3 & a_5 & a_7 \\ 0 & 1 & a_2 & a_4 & a_6 \\ 0 & 0 & a_1 & a_3 & a_5 \end{vmatrix} \cdots$$

當  $a$  的索引超過  $n$  時應該被設為 0。使用 Lienard-Chipart 準則的優點為在某些  $n$  時不用計算全部的 Hurwitz 行列式，藉此減少了計算的負荷。於是在此採用 Lienard-Chipart 準則來判斷控制器的穩定度。

利用指定邊線 (Margin) 確保穩定度：既然受控體及控制器都是穩定的話，根據奈氏穩定，假如開回路轉移函數  $L$  沒有在  $1+0j$  包圍成一圈的話則此閉回路系統為一穩定系統。然而，在電腦上利用奈氏穩定準則測試穩定度是較困難且沒有效率的 (Polak[12])。因此，提出下列穩定度充分條件來取代奈氏穩定準則。

$$(C2) \quad |1+L(x, w)| - |1-L(x, w)| \leq 2a \quad \forall w \in [0, \infty) \quad (3.2.5)$$

在此  $a \in (0, 1)$ ，與穩定邊限 (stability margin) 有關。上式在複數平面上形成一雙曲面的限制區域，如圖 3.2.1 所示。這個限制區域不但避免了  $L(j\omega)$  在  $1+0j$  包圍成一圈且提供閉回路系統穩定的邊限。在此條件下，增益邊限 (gain Margin GM) 和相位邊限 (phase margin PM) 將滿足下列關係

$$\begin{cases} GM \geq a^{-1} \\ PM \geq \cos^{-1}(a\sqrt{2-a^2}) \end{cases} \quad (3.2.6)$$

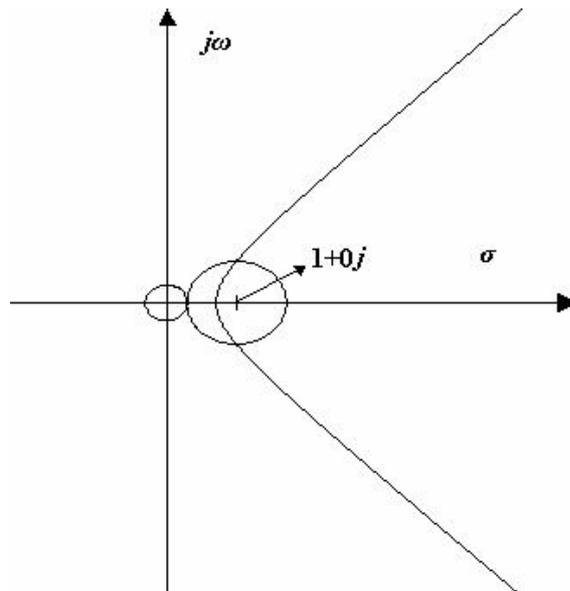


圖 3.2.1 不同限制在 Nyquist 圖上造成的區域

由(3.2.6) 可明顯看出當  $a$  越小時，增益邊限越大，則閉回路系統穩定性越強。此特性可幫助轉換實際需求的穩定邊限到參數  $a$  的選擇。

**噪音消除:**為了得到好的噪音消除效果，必須在噪音的頻寬內使靈敏度函數的增益最小化，既然聲音的音壓等級(Sound Pressure Level)定義為聲壓的方均根值， $H_2$  norm 自然的被利用來測量最小化噪音的平均能量。這個需求可以被表示成下列式最佳化問題的形式

$$\min_{\mathbf{x}} \int_0^{\infty} \left| \frac{1}{1-L(\mathbf{x}, \omega)} \right|^2 W(\omega) d\omega \quad (3.2.7)$$

在此， $W(\omega)$  為噪音的功率頻譜密度 (power spectral density)。這個最小化相等於對圖 2.2.1 中輸出訊號的能量做最小化，假設噪音沒有定義，則可以假設成有頻寬限制的白噪音(Band-limited white noise)。

**水床效應:**因為耳機系統為非極小相位系統，所以某些設計必須考慮。當使得靈敏度函數的增益在某頻率變低時，會造成在其他區段的增益變高 (Francis and Zames [13]; Freudenberg and Looze [14])。因此，避免噪音被過度放大，必須加入一限制避免靈敏度函數的增益變的太大

$$(C3) \quad |1 - L(\mathbf{x}, w)| \geq b \quad \forall \omega \in [0, \infty) \quad (3.2.8)$$

在此  $b \in (0, 1)$ 。上述與頻率相關的限制條件在複數平面上提供了一個以  $1+0j$  為圓心，半徑為  $b$  圓形限制區域，見圖 3.2.1。這圓形限制區域確保了靈敏度函數的增益在  $b^{-1}$  以下。有趣的是，條件(C2)和(C3)創造出一個新的穩定邊限。

$$\begin{cases} \text{GM} \geq \max \{a^{-1}, (1-b)^{-1}\} \\ \text{PM} \geq \max \left\{ \cos^{-1} \left( a\sqrt{2-a^2} \right), \cos^{-1} \left( 1-b^2/2 \right) \right\} \end{cases} \quad (3.2.9)$$

**對高頻不確定性的強健穩定：**一般而言，物理系統在高頻都會有不確定性。舉例來說，系統在延遲上有一些輕微的改變，在高頻的相位部分會產生相當大的變化。於是為了系統的穩定度，開迴路的增益應該要小

$$|L(\mathbf{x}, \omega)| \leq c \quad \forall \omega \in [\varpi, \infty) \quad (3.2.10)$$

在此  $c \in (0, 1)$ 。這個條件使得在高頻  $\omega \in [\varpi, \infty)$  的開迴路的增益小於所指定的值  $c$ ，於是系統對於高頻的不確定性變的更強健。這個條件造成在複數平面上， $L(\mathbf{x}, \omega)$  在高頻的軌跡被吸引到一圓形區域，見圖 3.2.1。

簡而言之，在給定一受控體的頻率響應，可藉由解下列限制性最佳化問題而找出一固定階數的控制器

$$\min_{\mathbf{x}} \int_0^{\infty} \left| \frac{1}{1 - L(\mathbf{x}, \omega)} \right|^2 W(\omega) d\omega \quad (3.2.11)$$

subject to

$$(C1) \quad \begin{cases} a_n > 0, a_{n-2} > 0, a_{n-4} > 0, \dots, a_i > 0 \\ D_1 > 0, D_3 > 0, D_5 > 0, \dots, D_k > 0 \end{cases} \quad (3.2.12)$$

$$(C2) \quad |1 + L(\mathbf{x}, w)| - |1 - L(\mathbf{x}, w)| \leq 2a \quad \forall w \in [0, \infty) \quad (3.2.13)$$

$$(C3) \quad |1 - L(\mathbf{x}, w)| \geq b \quad \forall \omega \in [0, \infty) \quad (3.2.14)$$

$$(C4) \quad |L(\mathbf{x}, \omega)| \leq c \quad \forall \omega \in [\varpi, \infty) \quad (3.2.15)$$

在此設計的參數  $a, b, c$  是在  $(0, 1)$  之間的實數。控制問題也變成一設定不同的限制而創造出不同可達成之結果。

### 3.3 解的近似

上述提到的計算是限制性最佳化的一種 semi-infinite programming，其結果不是變數數目就是限制條件為無窮大，但並非兩者同時成立 (Polaket al.[15])。在此選擇限制條件為無窮大，並藉由將頻率數位化的方法 [見 Boyd et al.[16]; Helton and Sideris[17]; Rafaly and Elliott [18]]，如此，在近似後變成有限的限制條件，並且(3.2.11)藉由黎曼和(Riemann summation)方法近似。由於使用固定階數的控制器，這個近似的限制性最佳化問題並不是 convex。僅能利用 Recurive Quadratic Programming(RQP)的方法找到區域最佳解。在此，為了使 RQP 方法更有效率，加入了有效集合(active set)的方法，反覆的利用原始限制條件的部分集合來尋找解的最佳化方向，這種解的方法已經相當多的發展和討論(見 Lim and Arora[19]; Gabriele and Beltracchi[20]; tseng and Arora[21])，因此，實現的步驟在此就不詳細說明。

對於在整個頻帶的頻率取樣上，一個好的近似可以得到原始的問題，然而，用過多的資料來近似會使得解問題的計算量變的特別大，但資料量太少會使得近似的問題無法代表原始的問題，造成設計出的系統無法保證穩定且實現的效果無法達到設計的規格。因此，一有效的方法為在低頻時頻率響應的資料用較小的取樣間隔，而在高頻時用較大的區樣間隔。因為開迴路在高頻的區間其增益值都很小，系統對高頻的敏感度也較小。因此，在高頻用較大的取樣間隔對結果並不會造成嚴重的影響。

### 3.4 ANC 耳機的迴授控制器設計

這個章節利用提出的方法設計一 ANC 耳機的類比控制器，圖 2.2.1 為 ANC 耳機的電路圖，兩個外部的訊號被引入此系統。一個被視為對系統干擾的噪音源  $d$ ，另一個為電子輸入訊號  $r$ ，例如：從 CD 撥放器或手機產生的音樂（聲音）訊號。在此，ANC 耳機的目標是當減低從外界來的噪音並同時保持訊號  $r$ 。

我們各在耳機兩邊耳朵附近的位置放置一顆電容式麥克風，以用來擷取耳朵附近的聲音，電容  $C1-C2$  和電阻  $R1-R2$  組成一簡單的高通濾波器，來補償整個迴授系統對複製訊號  $r$  的失真影響。 $C3$  和  $C4$  為交偶電容，OP 放大器與  $R3-R5$  組合成一前置放大器（preamplifier）和反相加法器，所以整個電路功用為將放大的麥克風訊號與濾波後的訊號相加，最後，用一個耳機驅動電路用來推動耳機。整個電路主要有兩個濾波器，前置濾波器（prefilter）和迴授控制器，這兩個濾波器對 ANC 耳機都佔有重要的角色，在此主要強調的是迴授控制器的設計，所以實作上沒有加入前置放大器電路。

許多因素都會造成耳機動態響應的不確定性，例如：過大振幅的刺激產生非線性特性，溫度的變化也會造成參數的變化，還有不同的環境造成聲壓的變化（例如：輕微的移動耳機戴的位置或者耳機戴的鬆緊不同）。在這些因素中，不同的環境造成聲壓的變化影響最大。因為耳朵與耳機的喇叭形成一小型封閉區域，耳機戴不同的位置或戴的鬆緊不同造成封閉區域的改變，於是影響了耳機的動態響應。受控體的頻率響應測量方法為移除圖 2.2.1 的前置濾波器和迴授控制器，然後應用一弦波輸入到  $i$  點，同時測量  $e$  點看此頻率的振幅與相位的變化，並將弦波從在音頻範圍內從低頻掃到高频，如此可測得耳機的頻率響應。圖 3.4.1 為耳機戴四種不同位置所測得不同的頻率響應。可發現各個頻率響應間差異都蠻大的，主要就是因為動態響應對於耳朵與耳機喇叭間的小型封閉區域的變化非常敏感所導致。

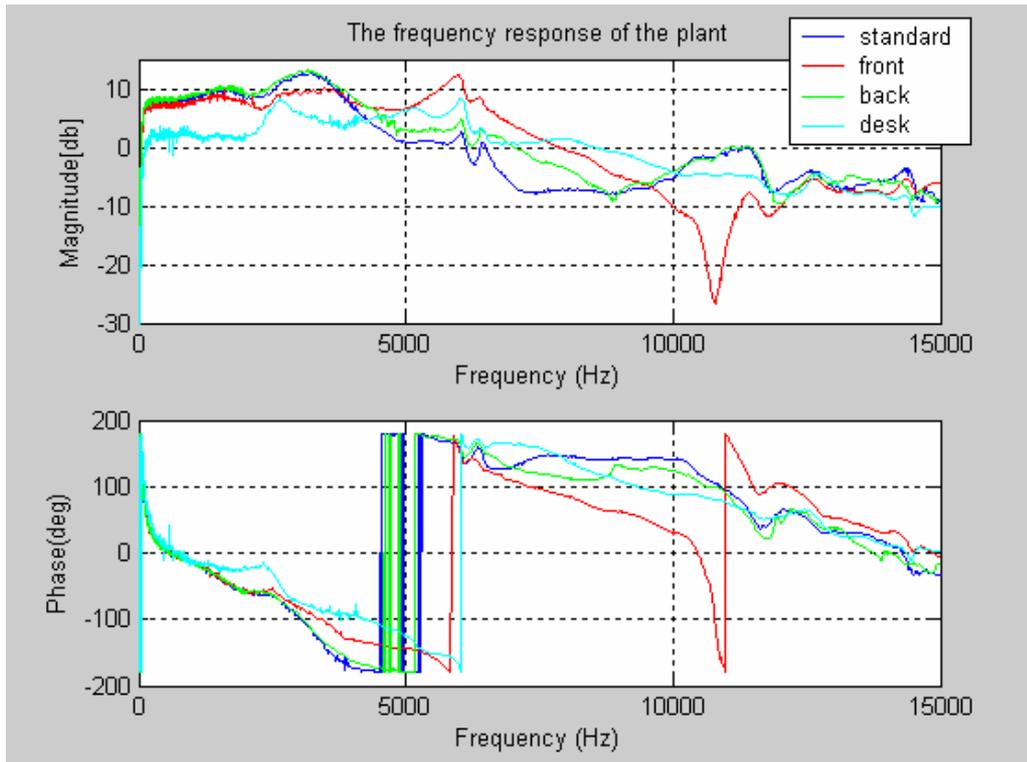


圖 3.4.1 四個具代表性之耳機系統的頻率響應

於是，圖 3.4.1 找出了四個具代表性頻率響應來概括了耳機系統可能的變化，分別為 standard（標準配戴），front（配戴較標準位置往前一點），back（配戴較標準位置往前後一點），desk（放置於桌上），由於耳機系統其他頻率響應大致上與此四個頻率響應差異不大，所以就不再加入考慮。在此，主要的目標是設計一迴授控制器來穩定這些具代表性的受控體，並同時擁有強健的特質。為了達成這些條件，目標函數（11）改成

$$\min_{\mathbf{x}} \sum_{k=1}^4 \int_0^{\infty} \left| \frac{1}{1-L_k(\mathbf{x}, j\omega)} \right|^2 W(\omega) d\omega \quad (3.4.1)$$

$L_k = P_k C$ ，且  $P_k, k \in \{1, 2, 3, 4\}$  為代表的受控體，此外，代表的受控體之開迴路轉移函數都應該滿足(3.2.12)~ (3.2.15)式的限制，而(3.3.1)式代表了被消除後之噪音能量平均值的最小化。對四個具代表性頻率響應在 1-2000Hz 每隔 1Hz 做取樣，2000Hz-15000Hz 每隔 10Hz 做取樣，來對頻率做離散化。此外，(3.3.1)式用黎曼和做近似且(3.2.12)~ (3.2.15)式只在取樣的頻率做考慮。實際設計的例子會在第 5 章有詳細的說明。

### 3.5 CODEC 時間延遲現象

前幾章節對一已知的受控體設計出類比的 ANC 控制器有詳細的描述，而在我們在對 C5510 DSK 做 ECHO 測試時，發現在 CODEC 取樣頻率設為 96k Hz 的情形下，訊號從進入 CODEC 到送出 CODEC，DSP 當收到訊號後馬上送出，我們使用頻譜分析儀發現有一線性相位(Linear phase)現象（參考圖 3.5.5），也就是表示訊號從進入 CODEC 到送出 CODEC 有時間上的延遲，在圖 3.5.1 可看出延遲的時間約 1/6.3k 秒。

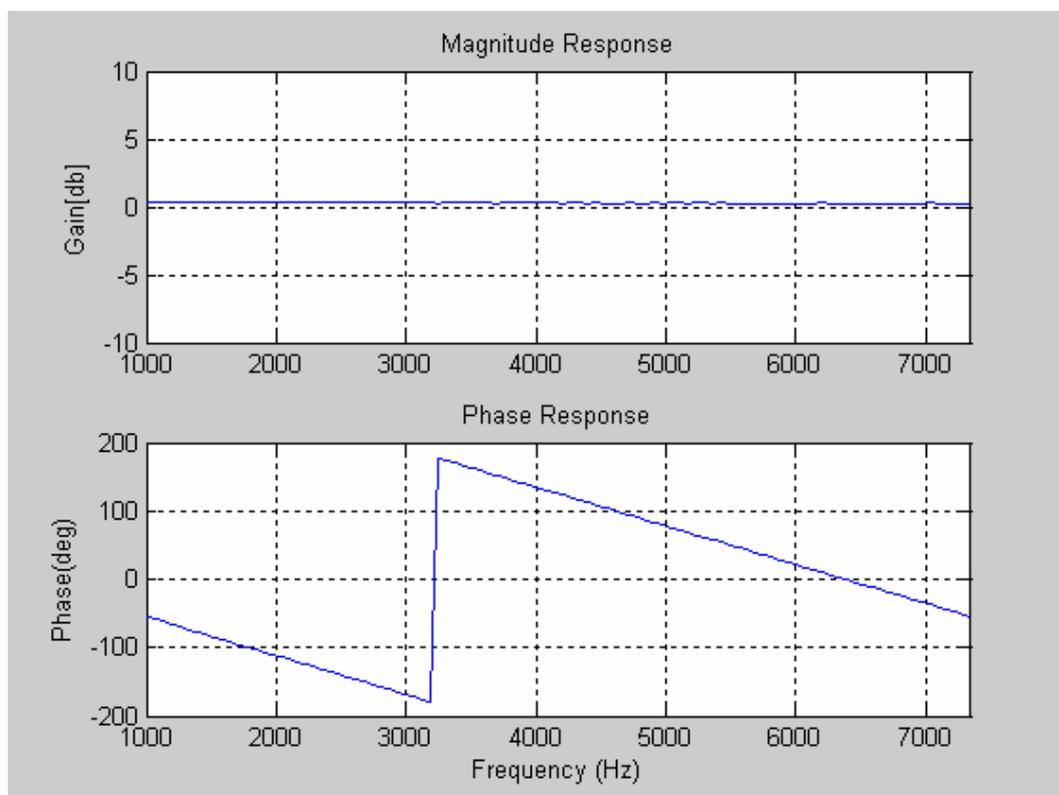


圖 3.5.1 C5510DSK ECHO 的頻率響應結果

為了確定時間的延遲是發生在哪一個部分，利用邏輯分析儀量測訊號 (CODEC 的 DOUT 腳位) 輸入到 mcbasp 到訊號由 mcbasp 輸出 (CODEC 的 DIN 腳位) 耗費多少時間，由圖 3.5.2 可看出在目前時間的 DOUT 訊號會在下個取樣時間在 DIN 腳位出現，圖 3.5.3 與圖 3.5.4 有 DOUT 及 DIN 的訊號放大圖，我們將以上測量結果畫成一關係圖以瞭解整個電路時間延遲的狀況，如圖 3.5.5 所示。

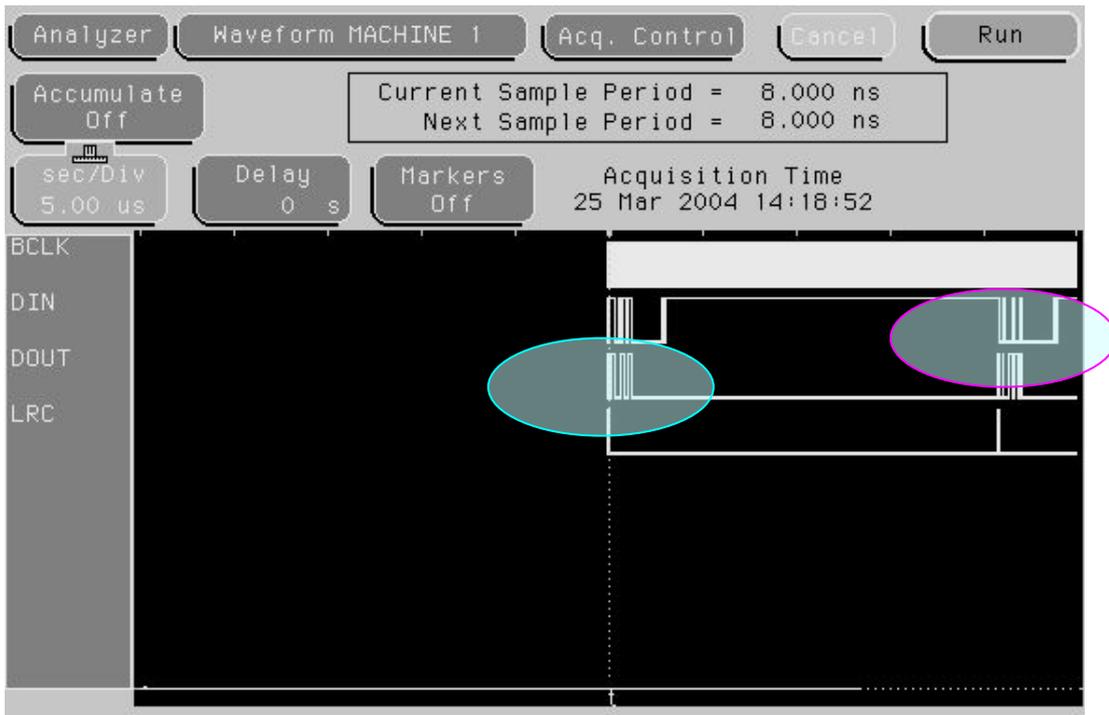


圖 3.5.2 邏輯分析儀訊號圖

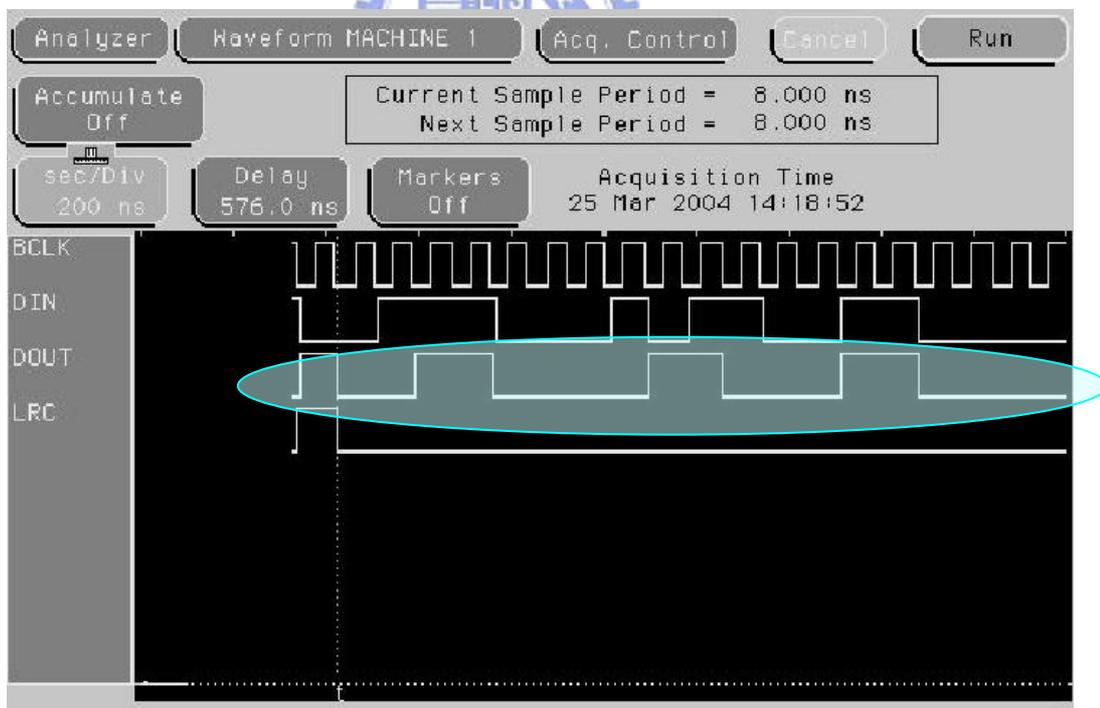


圖 3.5.3 CODEC DOUT 訊號圖

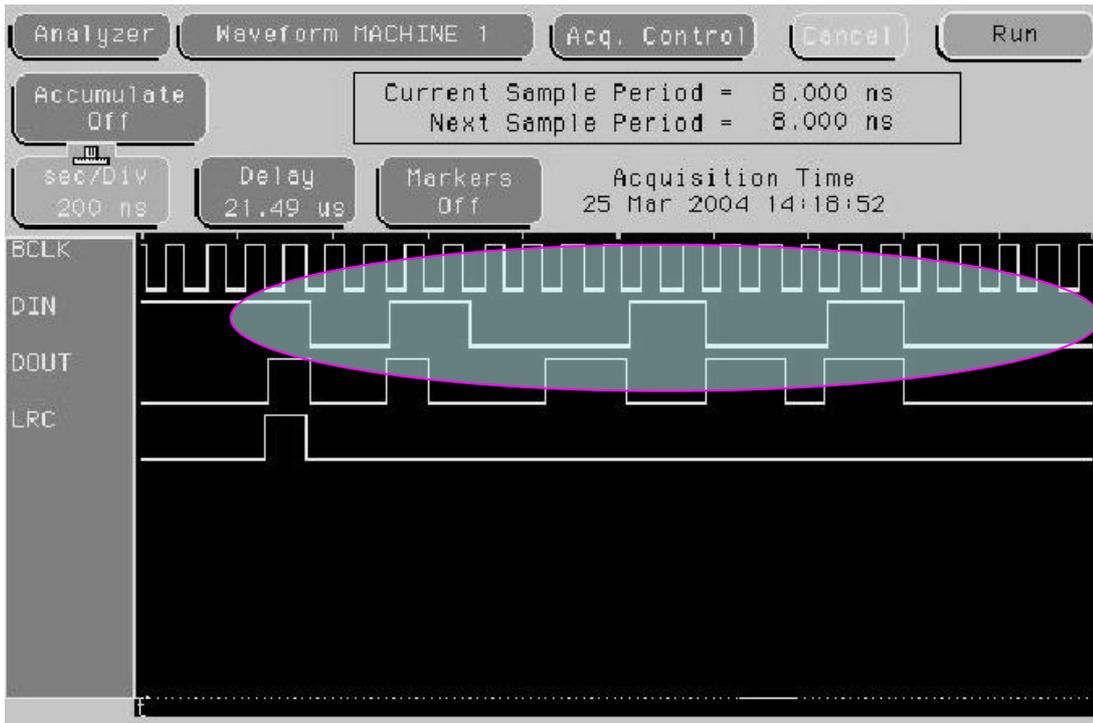


圖 3.5.4 CODEC DIN 訊號圖

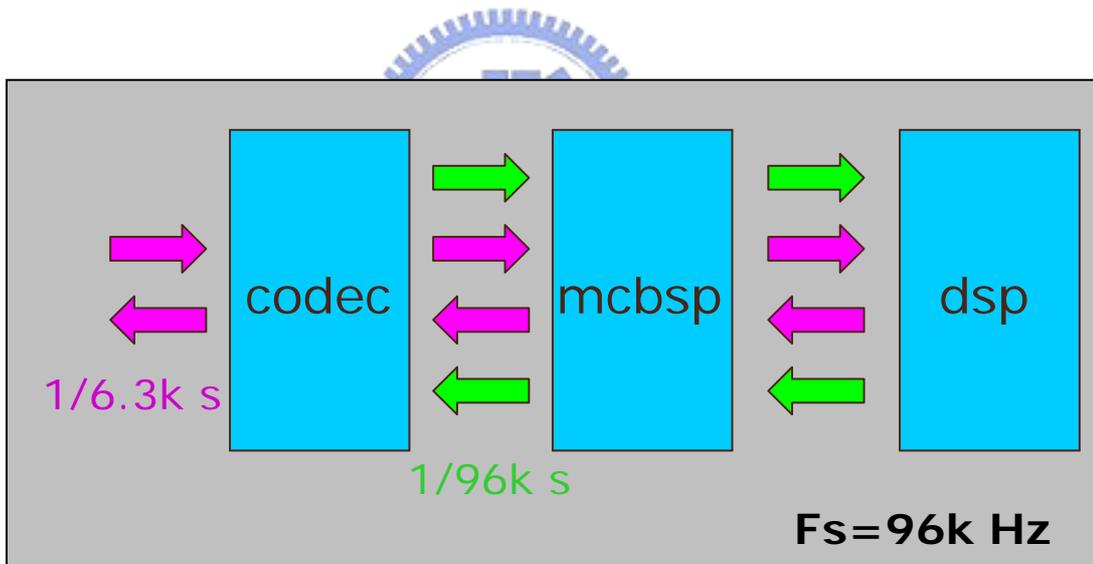


圖 3.5.5 DSK5510 ECHO 測試的結果

圖 3.5.5 可看出，CODEC 所造成的時間延遲為  $(1/6.3k - 1/96k)$  秒，所以 DSK5510 的 ECHO 的時間延遲主要是由 CODEC 造成，但在此我們並不打算更換 CODEC，而是將 ECHO 的時間延遲加入到受控體，當成受控體的一部份，所以受控體的頻率響應從圖 3.4.1 變成圖 3.5.6 所示。另外，雖然此 CODEC 有其他取樣時間可設

定，但取樣頻率設為 96k Hz 時，其造成的時間延遲為最少，所以在實作中 CODEC 的取樣頻率都固定設定為 96k Hz。

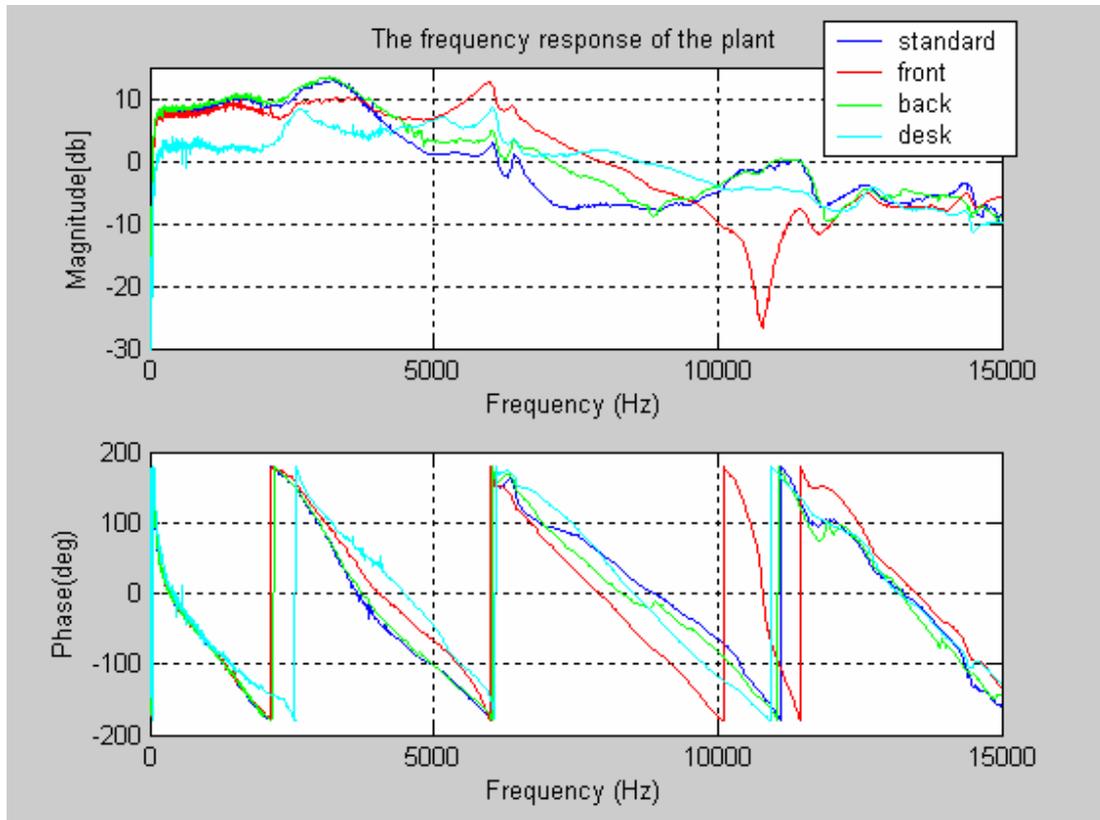


圖 3.5.6 四個具代表性之耳機系統加入 ECHO 時間延遲後的頻率響應

### 3.6 定點數位控制器係數設計

利用前幾章的方法可設計出一類比控制器，其形式如下

$$C(s) = \frac{b_{c0} + b_{c1}s^1 + b_{c2}s^2 + \dots + b_{cn}s^n}{a_{c0} + a_{c1}s^1 + b_{c2}s^2 + \dots + a_{cn}s^n} \quad (3.6.1)$$

而轉成 z-domain 後變成如下

$$C(z) = \frac{b_{d0} + b_{d1}z^{-1} + b_{d2}z^{-2} + \dots + b_{dn}z^{-n}}{a_{d0} + a_{d1}z^{-1} + b_{d2}z^{-2} + \dots + a_{dn}z^{-n}} \quad (3.6.2)$$

由於我們是用定點數位訊號處理器做控制器，所以必須將係數做量化[23]，但如果直接對其係數做量化，其頻譜響應因為量化而產生誤差，而此誤差會隨著

控制器的階數提高而增加，下圖為八階的數位控制器利用上述方式轉換所得的控制器係數量化前與量化後頻率響應比較圖。

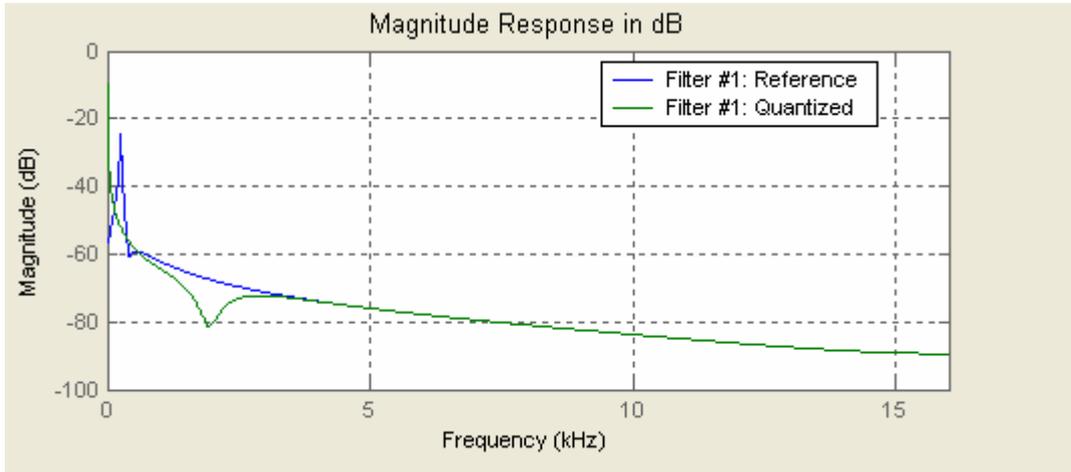


圖 3.6.1 控制器係數量化前與量化後的頻率響應比較圖一

因此我們在轉成 z-domain 之前先做分解的動作，其式子如下

$$\begin{aligned}
 C(s) &= \frac{b_{c0} + b_{c1}s^1 + b_{c2}s^2 + \cdots + b_{cn}s^n}{a_{c0} + a_{c1}s^1 + a_{c2}s^2 + \cdots + a_{cn}s^n} \\
 &= k_c \frac{(s - z_{c1})(s - z_{c2}) \cdots (s - z_{cn})}{(s - p_{c1})(s - p_{c2}) \cdots (s - p_{cn})}
 \end{aligned} \tag{3.6.3}$$

之後再將其轉成 z-domain

$$\begin{aligned}
 C(z) &= k_d \frac{(z - z_{d1})(z - z_{d2}) \cdots (z - z_{dn})}{(z - p_{d1})(z - p_{d2}) \cdots (z - p_{dn})} \\
 &= g \prod_{d=1}^L \frac{b_{d0} + b_{d1}z^{-1} + b_{d2}z^{-2}}{a_{d0} + a_{d1}z^{-1} + a_{d2}z^{-2}}
 \end{aligned} \tag{3.6.4}$$

我們再次對之前的濾波器以新的方法量化，其結果參考圖 3.6.2，可發現新的方法可明顯降低控制器係數在量化前與量化後頻率響應的誤差。

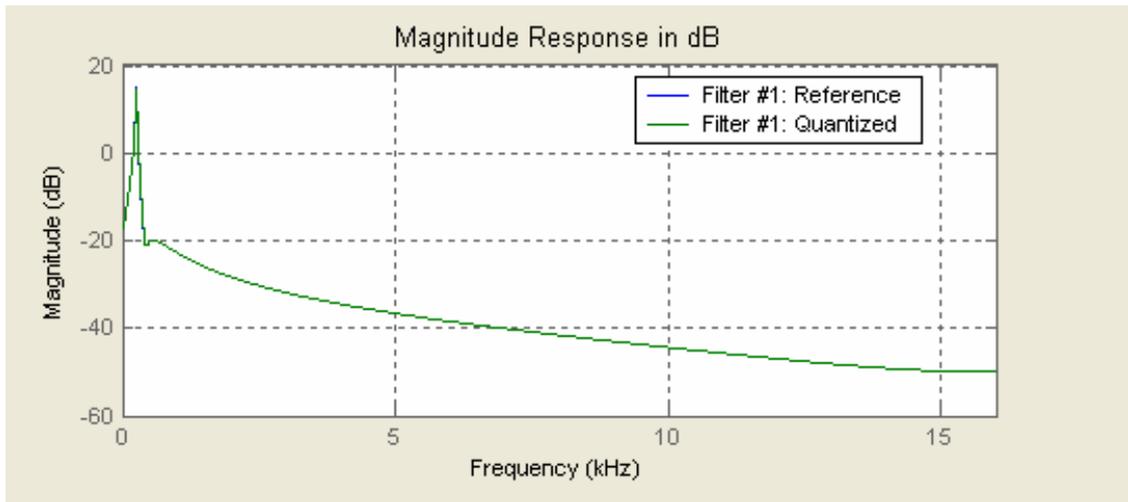


圖 3.6.2 控制器係數量化前與量化後的頻率響應比較圖二



## 四、實驗平台

在前幾章節，已對 ANC 控制器的設計有詳細的說明，在此章節，將對 ANC 耳機的平台及其設計流程做簡單的概述。

### 4.1 平台架構說明

ANC 耳機的平台可大致上分成三大部分，分別為測量平台，設計及模擬平台及硬體平台，其關係可由圖4.1.1看出

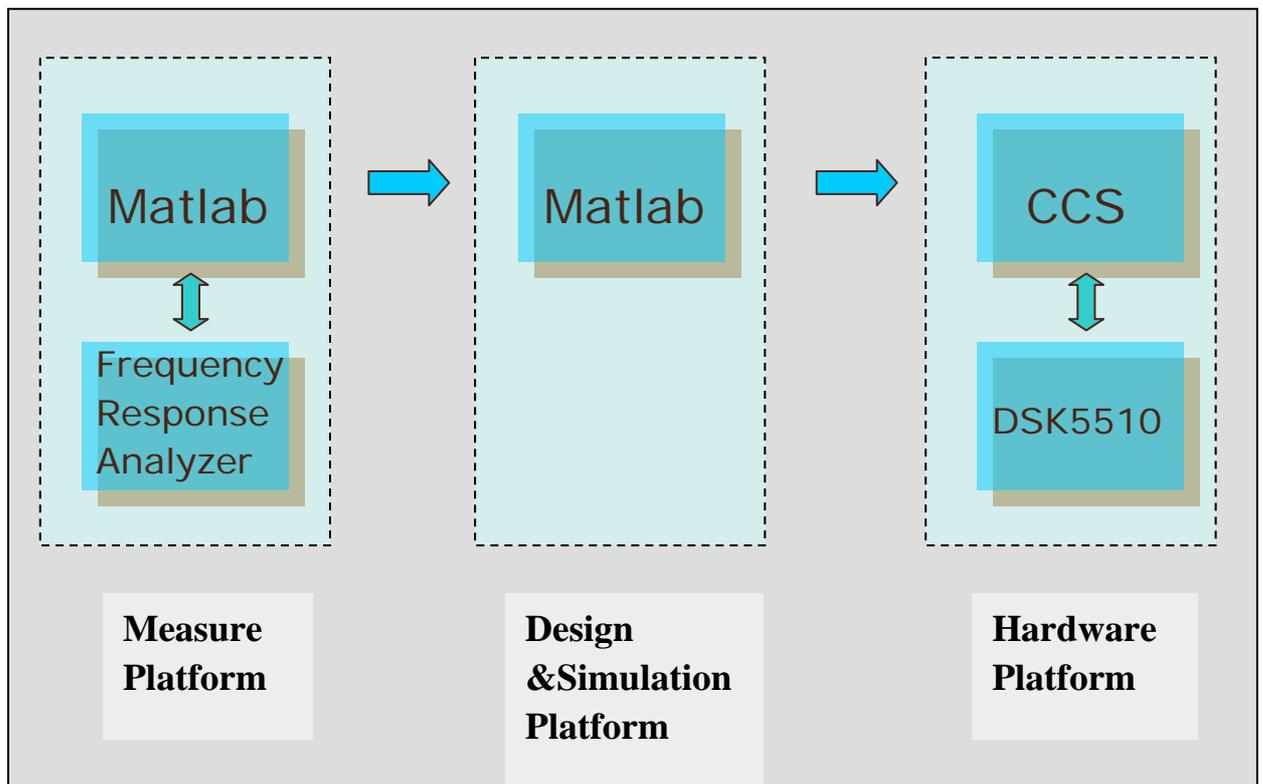


圖4.1.1 ANC實驗平台關係圖

整個設計流程為先由測試平台得到受測體的頻譜響應，再由設計及模擬平台設計出控制器係數，最後再以硬體平台實現，整個平台軟體部分已整合成Matlab及CCS兩大部分，減化設計的環境。以下將對各平台做較詳細的介紹。

## 4.2 測量平台

此平台主要功能為自動測量受控體的頻率響應，整個平台架構如圖 4.2.1 所示。

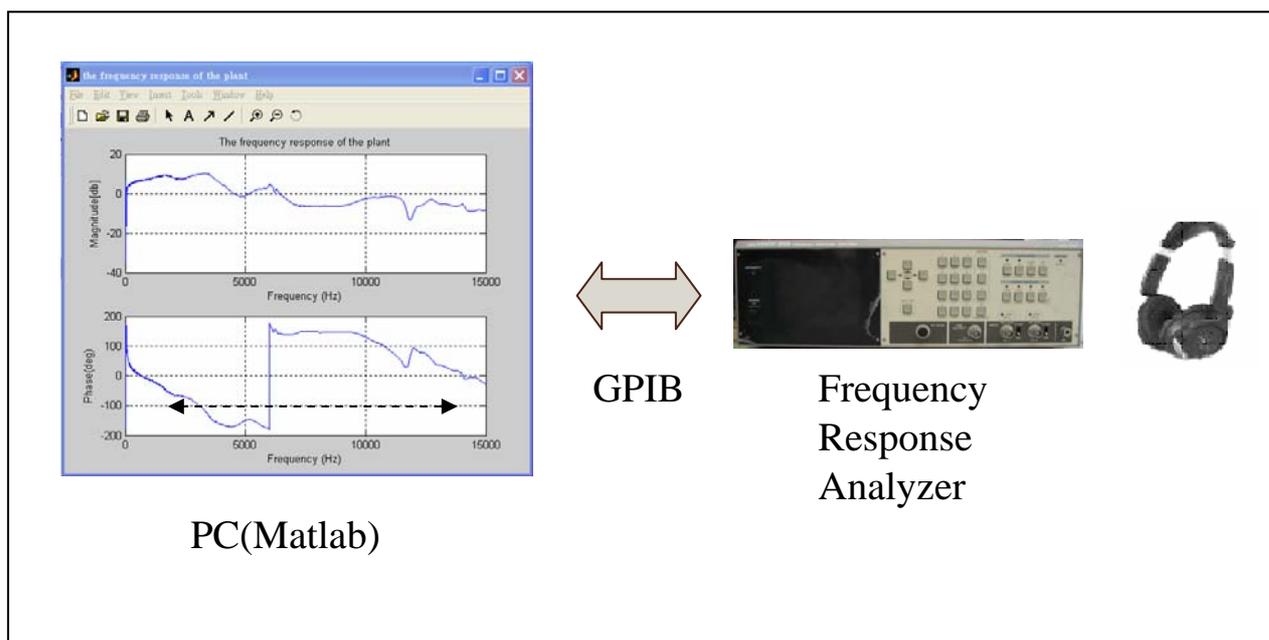


圖 4.2.1 ANC 測量平台

GPIB（通用介面匯流排）為一標準界面，又稱為 IEEE-488(或 HP-IB)，可藉由其介面協定來連結及控制支援 GPIB 介面的儀器。在此使用的頻譜分析儀為 NF 5020，支援 GPIB 介面，而 Matlab6.5 的 Instrument Control toolbox 有支援 GPIB 協定的功能，所以我們利用 GPIB 架構一可程式化的自動測試平台，將所有需測試的步驟寫成 Matlab 的程式，由 Matlab 控制頻譜分析儀及自動的將測試的資料讀回 PC(Matlab)，且將資料存成所需的格式，如此可減少人工操作上的失誤及達到測量工作的自動化。

### 4.3 設計及模擬平台

此平台主要功能為設計 ANC 控制器及模擬其實際的效果，整個平台如圖 4.3.1 所示。

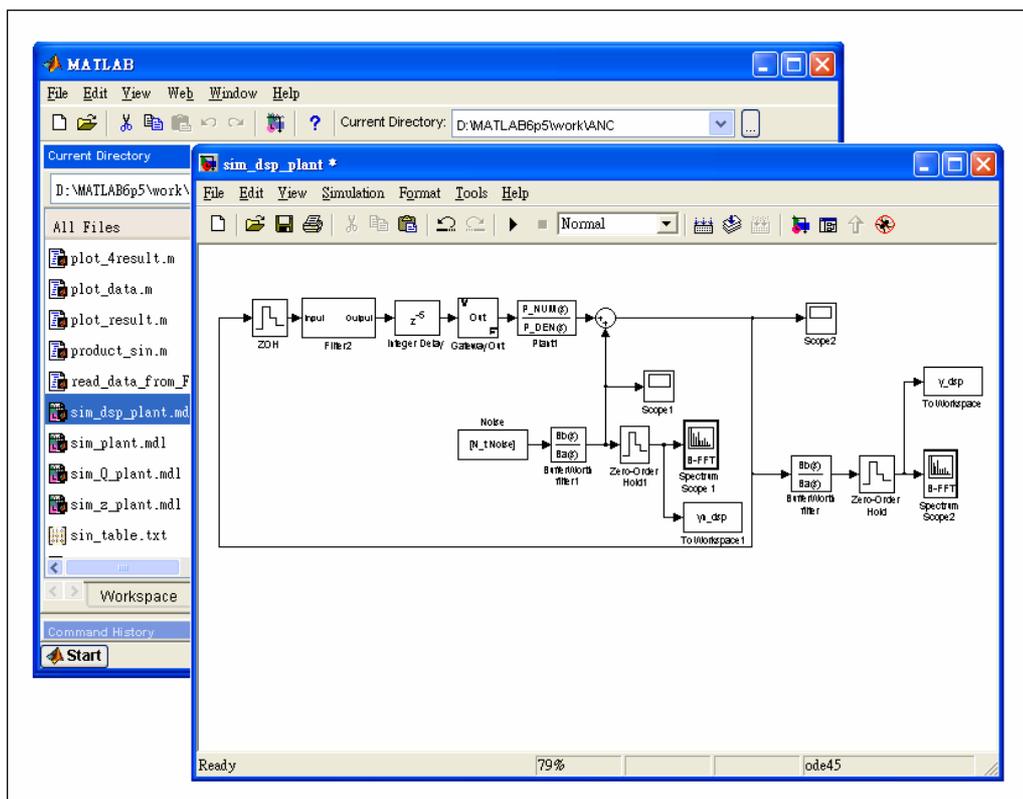


圖 4.3.1 ANC 模擬平台

此平台在 Matlab 環境下設計 ANC 控制器，藉由 Matlab 所提供的 Optimization toolbox、Filter Design toolbox 等強大的工具，使得設計控制器流程簡化許多，並且利用 simulink 架構一定點數位訊號處理器為控制器的 ANC 耳機系統，模擬其量化係數的非線性的特性及 ANC 耳機的環境，如此可將設計的控制器在模擬平台上測試其消音效果，直到模擬結果達到要求，再實際用硬體平台測試，如此可加速設計控制器的速度。

## 4.4 硬體平台

ANC 耳機為一可攜式產品，所以晶片的耗電量為重要考量的因素，因 DSP5510 為 TI (Texas Instruments) 最省電的 DSP，且發展平台支援 C 語言及許多 DSP 相關的函式庫 (library)，因此選擇 DSP5510 數位訊號處理器做為 ANC 耳機的數位控制器。

硬體平台主要是由 DSK5510(DSP Starter Kit)(圖 4.4.1)、麥克風放大電路、耳機驅動電路、電容式麥克風及雙聲道耳機 (AKG K501) 所組成，在此大致簡介 DSK5510 的特性，其中所包含的元件及程式架構在後面小節會詳細解說。

C5510DSK 為一發展 TI C55X DSP 系列的實驗版，其主要特性如下

- 200 MHz TMS320VC5510 DSP
- AIC 23 Stereo Codec
- Four Position User DIP Switch and Four User LEDs
- On-board Flash and SDRAM

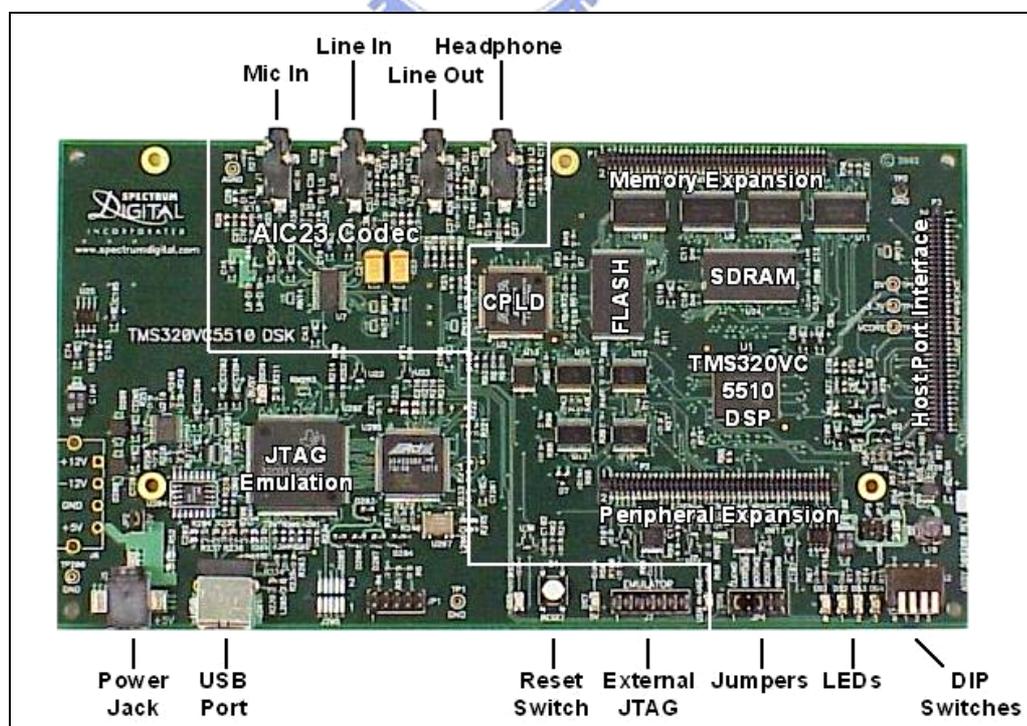


圖 4.4.1 DSK5510 實體圖

由上述說明可知，DSK5510 擁有高速的 DSP，可實現複雜的演算法，AIC23 Codec 提供 AD/DA 轉換，DIP 與 LED 可控制及表示 DSP 晶片的狀態，而 Flash 可燒入已編輯的程式碼，使得 DSK 可脫離 PC 獨自運作。

因為 AIC23 Codec 的麥克風輸入放大電路只支援單聲道及為了量測受控體的頻率響應，所以另外做了雙聲道的麥克風放大電路及耳機驅動電路，再與 AIC23 Codec 的 Line In 及 Line Out 連接。所以整個系統架構圖與實體圖如下所示

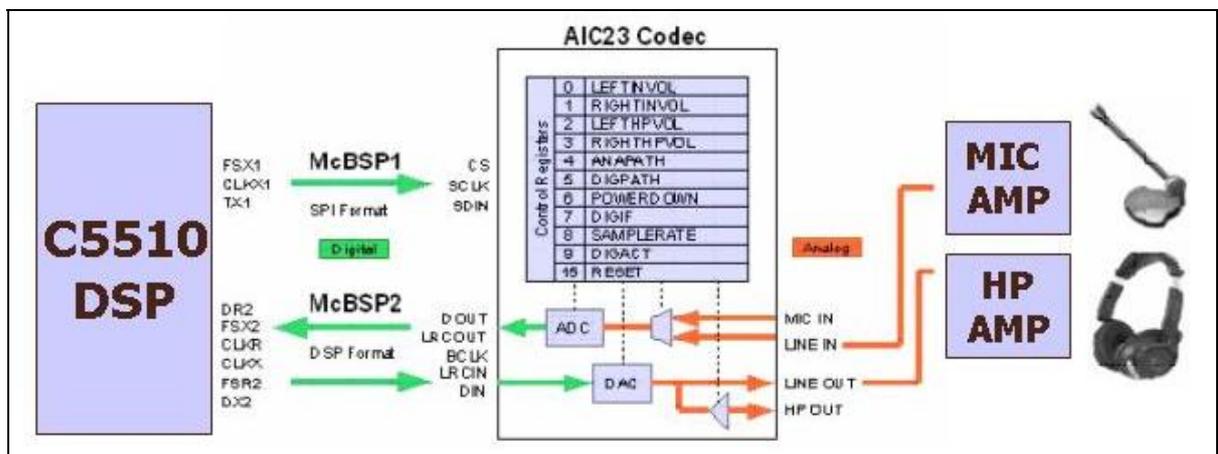


圖 4.4.2 ANC 耳機之系統架構圖



圖 4.4.3 ANC 耳機之實體圖

#### 4.4.1 TMS320C5510

TMS320C5510 DSP 晶片為德州儀器公司新一代 DSP 處理器，為目前工業界最省電的 DSP，相當適合應用在可攜式產品，晶片內部擁有一個程式匯流排(program bus)，三個資料讀取匯流排(data read bus)和兩個資料寫入匯流排(data write bus)增加運算的平行度，藉此達到高效能且省電的目的，並且可自動關閉未使用的周邊，記憶體或核心工作單元來增加電池的壽命。

此 DSP 擁有完整的發展平台及工具，例如 C-Compiler，使用者可用 C 語言發展軟體，容易發展複雜的演算法，提供許多常用的函式庫，例：DSPLIB（包含了 FIR/IIR filters，FFT 等等），加速常用的運算，還有 DSP/BIOS 工具，讓產品能容易的擁有即時多工的功能，上述的這些工具使得此 DSP 能快速的實現複雜的演算法，也兼顧了程式碼執行的效率。

在週邊部分，已內建了多樣化的輸出入通道，包括主機埠界面(Host Port Interface，HPI)、外部記憶體界面(External Memory Interface，EMIF)、高速串聯埠界面(Multichannel Buffered Serial Ports，McBSP)、即時資料交換(Real-Time Data Exchange，RTDX)，充分提供與外部輸出入訊號的傳輸界面，圖 4.4.1 為 TMS320C5510 一些基本的特性。

PARAMETER NAME	TMS320C5510
Clock frequency (MHz)	200
Data/program Memory (words)	160k
DMA	6
External Memory Interface	32
MCBSP	3
IO Supply (Volts)	3.3
Core Supply (Volts)	1.6

表 4.4.1 TMS320C5510 特性表

#### 4.4.2 多通道串列傳輸埠 (Multichannel Buffered Serial Ports, McBSPs)

此 DSP 擁有三個 McBSP，McBSP 可以作全雙工(Full-Duplex)之串聯通信，它們擁有雙緩衝器(Double-Buffered)，可以透過直接記憶體存取(DMA)或處理器作資料雙向存取；McBSP 可以直接連接各種通信介面標準，並可支援最高達 128 通道之多通道傳輸模式出入訊號的傳輸介面。

圖 4.4.4 顯示高速串聯埠界面的系統方塊圖，其中 DR(Received Serial Data) 及 DX(Transmitted Serial Data)為資料輸出入接腳、CLKX(Transmit Clock)、CLKR(Receive Clock)、FSX(Transmit Frame Synchronization)、FSR(Receive Frame Synchronization)為資料信號同步及資料框同步信號的輸出入接腳，CLKS 為使用外部時序信號時之輸入時脈波信號接腳，在傳輸方面可分為下面兩種模式。

接收模式：

在接收端接收到設定的訊號框同步訊號後，啟動接收模式並將資料由 DR 腳位讀入，先送入 RSR，次移入 RBR，然後送入 DRR。當資料從 RBR 送到 DRR 時，McBSP 會設定接收訊號(RRDY)，RRDY 會直接驅動 McBSP receiver event (REVT)到 DMA Controller；同時也會驅動 McBSP receiver interrupt (RINT)到 DSP 告知資料已經 ready，最後由 DMA 或 DSP 透過 32 位元的週邊匯流排將資料讀出。

傳送模式：

在傳送端接收到設定的訊號框同步訊號後，啟動傳送模式，資料會由 DSP 內部或記憶體透過 DMA，透過 32 位元的週邊匯流排將傳輸資料寫入 DXR，次移入 XSR，最後由 DX 腳位輸出。當資料從 DXR 被移入 XSR 時，McBSP 會設定傳送訊號(XRDY)，XRDY 會直接驅動 McBSP transmit event (XEVT)到 DMA Controller；同時也會驅動 McBSP transmit interrupt (XINT)到 DSP 告知可以透過 32 位元的週邊匯流排將新的資料載入到 DXR。

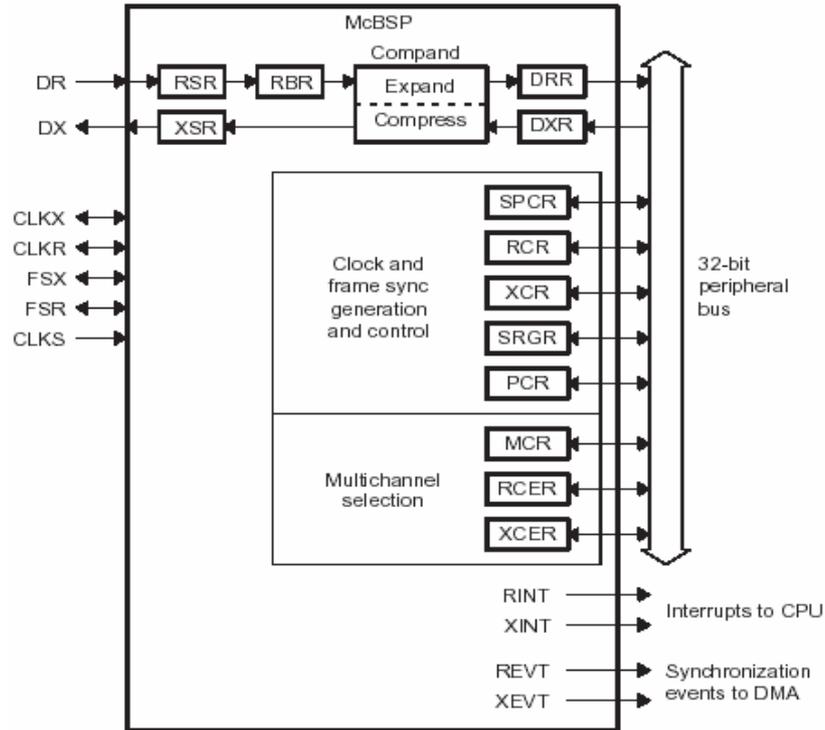


圖 4.4.4 McBSP 系統方塊圖

### 4.4.3 AIC23 CODEC

DSK5510 使用了 AIC23 CODEC 來輸入與輸出聲音訊號。CODEC 將麥克風或 Line In 的類比訊號轉換成數位訊號給 DSP 處理，當 DSP 處理結束後再將資料由 CODEC 轉回成類比訊號輸出到 Line Out 或耳機輸出。

CODEC 具有兩個串聯通道，一個通道用來控制 CODEC 的內部狀態設定，可以用來改變 CODEC 的取樣頻率，資料格式等等，另一個通道用來串送和接收聲音的數位訊號，以下是平台對於 CODEC 設定的詳細描述，可對照圖 4.4.2。

AIC23 設定狀態

控制通道 McBSP1，SPI 模式

資料通道 McBSP2，AIC23 設為主動模式，在每隔一個取樣時間就主動向 McBSP 送資料和接收資料

#### 4.4.4 麥克風放大電路與耳機擴大電路

因為實現 ANC 耳機系統，需要左右聲道各有一組電路，但 DSK 上 CODEC 晶片麥克風輸入只有支援單聲道，所以必須自行製作雙聲道的麥克風放大電路，再將放大訊號輸入到 CODEC 的 Line IN（雙聲道），其電路圖如下

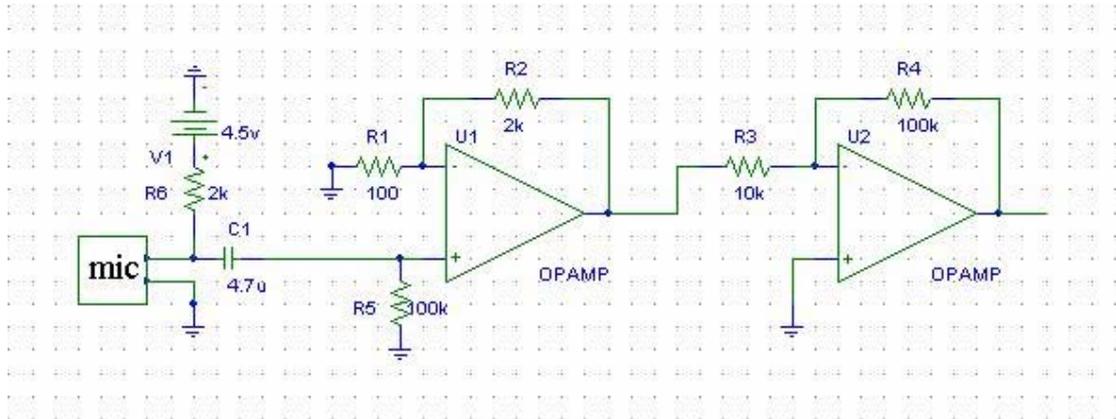


圖 4.4.5 麥克風放大電路圖

而為了測受控體的頻率響應，我們製作一個耳機擴大電路，將 CODEC 的 Line Out 的訊號（雙聲道）輸入到此電路，在此利用德州儀器的耳機擴大晶片 TPA6111A2 來製作，其電路圖如下

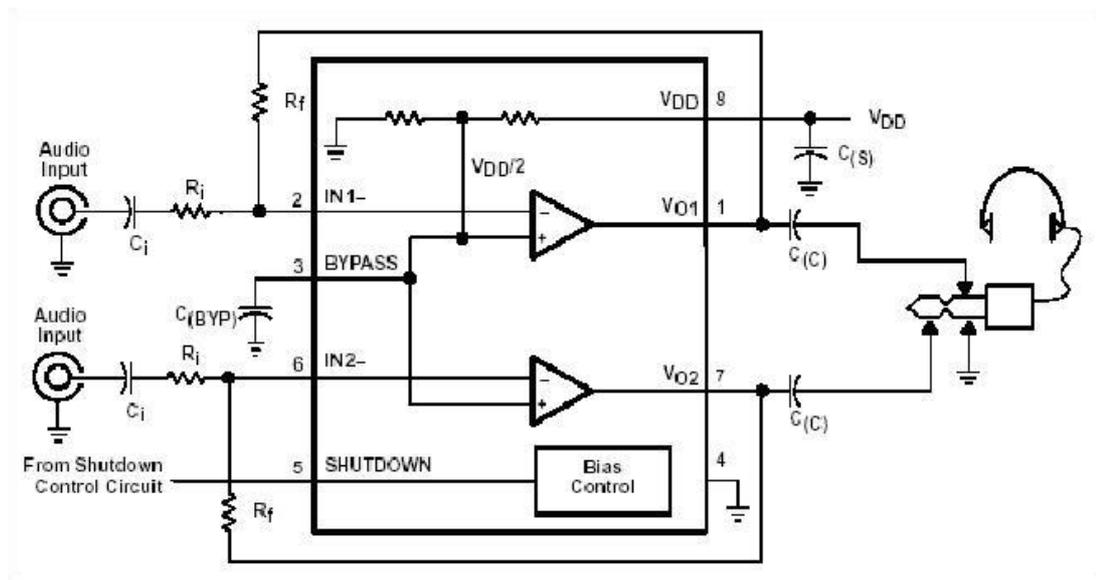


圖 4.4.6 耳機擴大電路圖

#### 4.4.5 DSP 程式碼架構與流程

此章節主要說明硬體平台中 DSP 程式的架構與流程，由於 TMS320C5510 DSP 有支援 C 語言的功能，所以大部分的程式都是以 C 語言來編寫，對於重複使用性較高的程式碼則以組語的方式編寫（例：IIR 濾波器），如此可實現複雜的程式並兼顧程式執行的效率。

##### 程式碼架構

整個 DSP 程式的基本架構如下圖所示，由主程式開始執行，主程式主要分成 main() 及中斷服務程式 read\_isr() 兩部分，程式一開始會執行 main() 函數，主要功能為對周邊和變數的初始化及對中斷的相關設定，執行後則 DSP 進入等待狀態，直到中斷訊號發生才執行中斷服務程式。而 DSP 程式大部分都由 C 語言編寫，只有中斷服務程式內的 IIR 函數(iircas51\_mod)才用組語編寫，以加速程式的執行效率。DSP 程式的主要原始碼附於附錄，可與圖 4.4.7 對照參考。

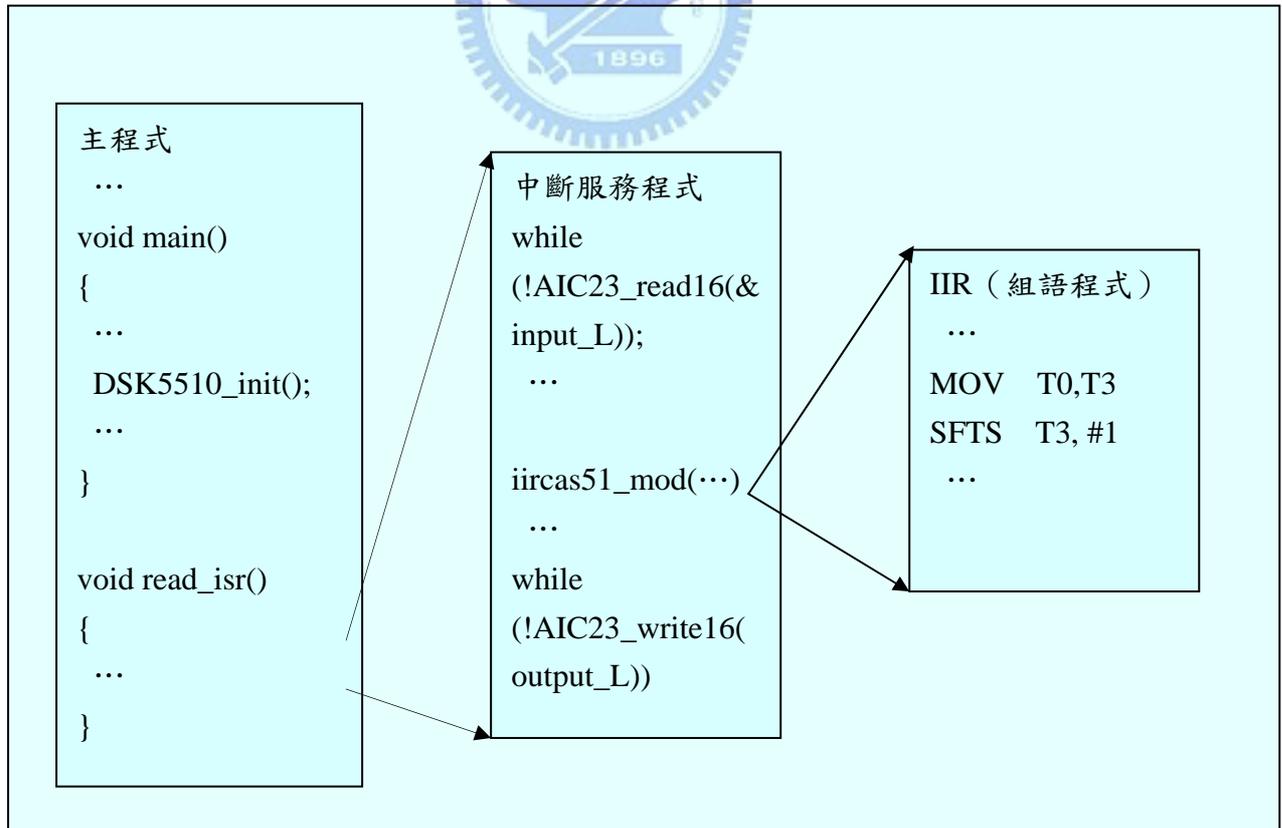


圖 4.4.7 DSP 程式架構圖

## 程式流程

整個硬體平台的基本運作流程為（參考圖 4.4.2）每隔一取樣時間，CODEC 的 A/D 送出雙聲道資料給 MCBSP，MCBSP 在完整收到資料後發出中斷訊號給 DSP，DSP 則執行中斷副程式，將資料處理後送給 MCBSP，在到達下次取樣時間時，MCBSP 則將 DSP 處理過的資料送出給 CODEC。

由之前的章節可知，此平台 CODEC 取樣頻率必須設定成 96KHz（詳情請參考 3.5 節），但在設計數位控制器時發現，如控制器取樣頻率設為 96KHz，則控制器會因為輸入訊號量化的原因而導致錯誤的結果，在測試之後，發現控制器取樣頻率設為 32KHz 可以實現，所以在資料進入數位控制器時降頻取樣（downsampling）為原 1/3，在資料處理完後，再將處理過的資料複製兩次升頻（upsampling）輸出，在此，因為 aliasing 和高頻雜訊現象不嚴重，且顧慮計算量的問題，所以並不加入額外的濾波器處理。

為了分散計算量，以便於未來可降低 DSP 工作時脈，所以我們將左聲道與右聲道分開在不同時間處理，如下圖所示。

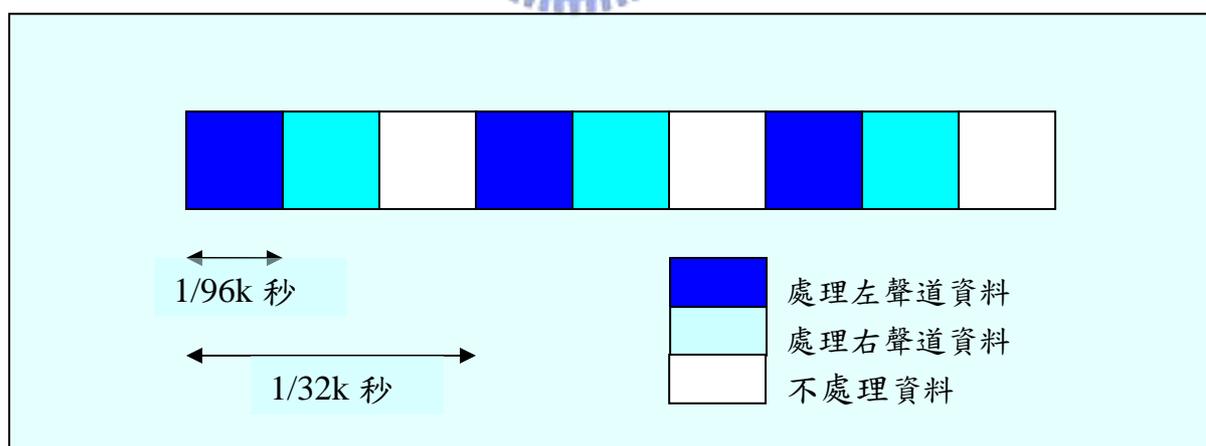


圖 4.4.8 DSP 運算狀態示意圖一

最後，我們在程式碼內利用 DSK 的 BSL，設定當 DIP3=0 (向下的狀態)為開啟噪音消除功能，此時 LED3 為 ON，DIP3=1 (向上的狀態)為關閉噪音消除功能，LED3 為 OFF，詳細的程式流程請參考下圖。

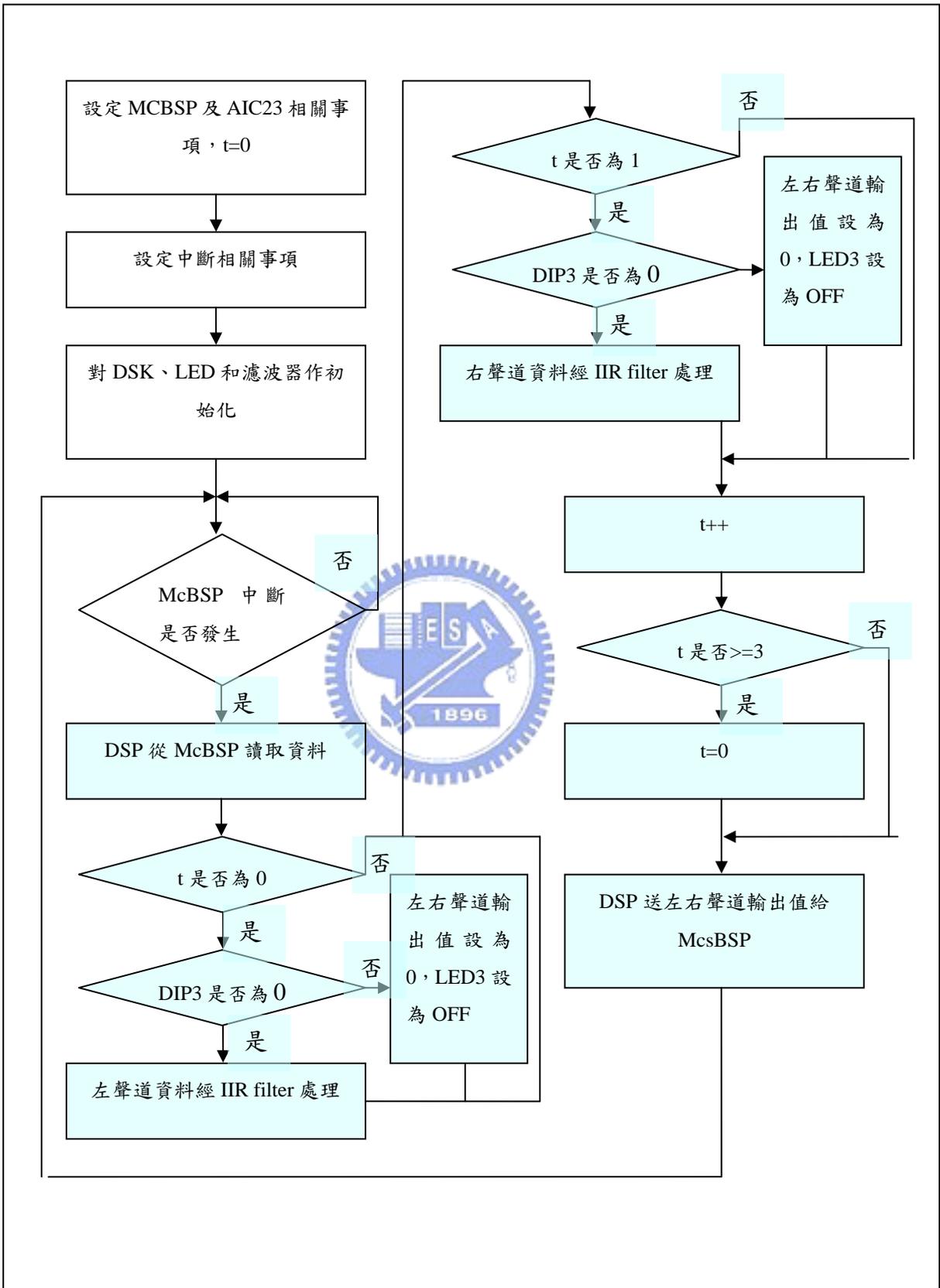


圖 4.4.9 DSP 程式之基本流程圖

## 組語程式

在 DSP 程式中，唯一的組語程式 IIR 函數是從 TI 所提供的 DSP 函式庫 (DSPLIB) 修改而成，我們在此做一簡單的說明。

從觀察 ANC 控制器的係數可發現，絕大部分的極點(Pole)和零點(Zero)在 S-domain 都會集中在低頻及靠近虛軸的地方，在 Z-domain 則分佈在  $1+0j$  的附近，如果 filter 以 second-order section form. 的形式表示 (4.4.1 式)， $-\frac{a_{d1}}{a_{d0}}$  的值會接近

2 且小於 2，

$$C(z) = \frac{Y(z)}{X(z)} = g \prod_{d=1}^L \frac{b_{d0} + b_{d1}z^{-1} + b_{d2}z^{-2}}{a_{d0} + a_{d1}z^{-1} + a_{d2}z^{-2}} \quad (4.4.1)$$

因為我們使用的是定點數位訊號處理器，所有係數必須大於等於 -1 且小於 1，所以在設計控制器係數時，我們將  $a_{d0}$  的值固定為 0.5，如此  $a_{d1}$  的值就會接近 -1 且大於 -1，但 TI DSP 函式庫 (DSPLIB) 所提供的 IIR 函數其  $a_{d0}$  固定為 1，則其  $a_{d1}$  的值，就會小於 -1 而無法利用其函數實現，所以我們對其 IIR 函數做些微的修改，其修改步驟如下。

(4.4.1) 式每一個二階有理式可展成下列形式，當  $a_{d0}=1$  時，則  $y_d(n)$  的值為右

$$y_d(n) = \frac{1}{a_{d0}} [b_{d0}x_d(n) + b_{d1}x_d(n-1) + b_{d2}x_d(n-2) - a_{d1}x_d(n-1) - a_{d2}x_d(n-2)]$$

式中括號內的運算值，將  $a_{d0}$  的值改成 0.5 時，則  $y_d(n)$  的值在算出右式中括號值需再乘上 2，在 DSP 晶片內則是將中括號值向左平移一位即可，實際程式可參考 A-3(p68)。

#### 4.4.6 DSP5510 耗電測試環境

由於此 DSP 省電為其主要特色，所以我們在此說明如何設定讓晶片更省電，並介紹整個測試耗電的環境。

下圖為 DSP 運算狀態的情形，每隔一取樣時間，CODEC 取樣訊號送給 DSP，DSP 經過運算後送回給 CODEC，之後等下一次取樣再做同樣的動作。

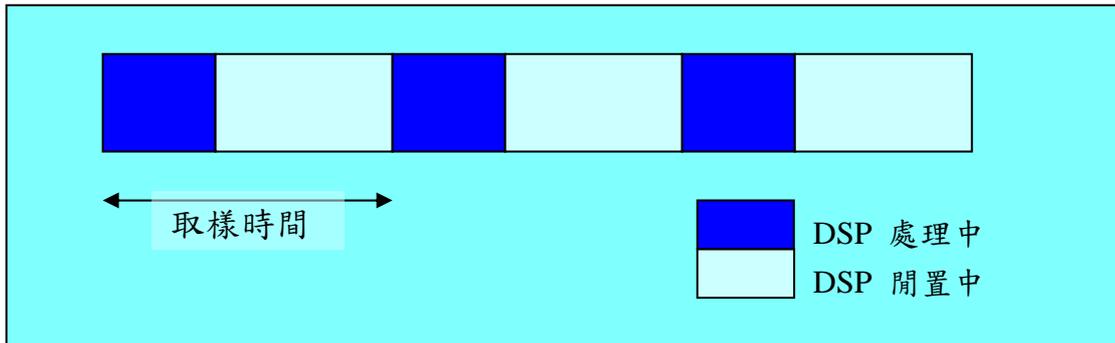


圖 4.4.10 DSP 運算狀態示意圖二

由於我們設定 CODEC 在下次取樣時 CODEC 才把處理過的資料送出去，只要保證 DSP 在下一次取樣前完成所有運算，我們可以降低 DSP 的運算時脈，藉此達到省電的目的，此 DSP 可用軟體設定 PLL 時脈的功能，其設定畫面如下圖。

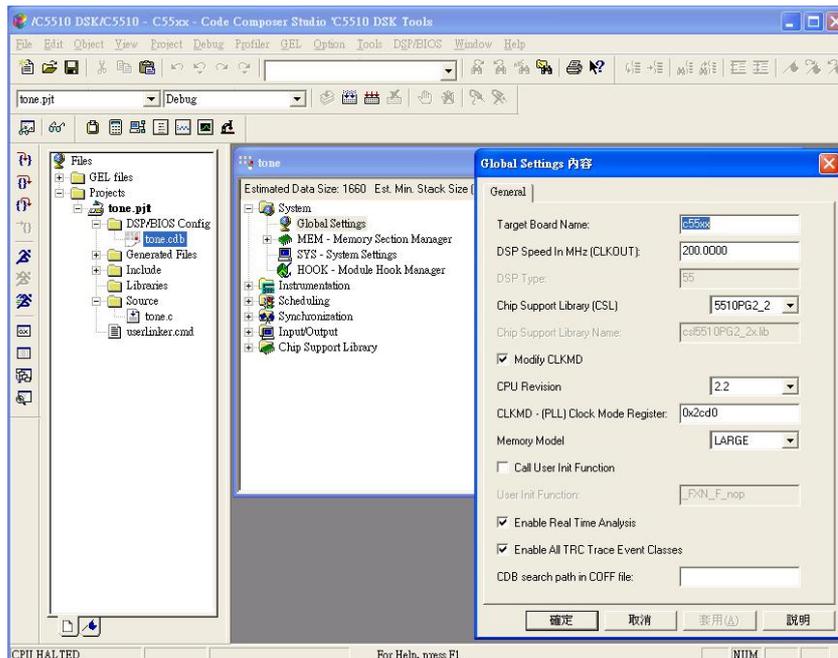


圖 4.4.11 DSP 設定時脈之環境

且當 DSP 的運算時脈下降到某一程度後，DSP 可降低其工作電壓，如此可達到更省電的效果。

我們在供給晶片工作電壓的 pin 腳與電源間串接一低阻抗電阻，藉由測量電阻的電壓算出其流過電流，得知晶片總消耗的電流值，其示意圖如下

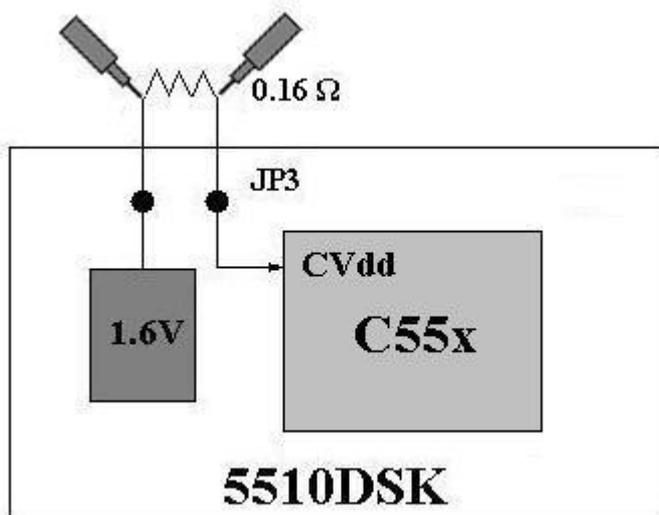


圖 4.4.12 測量 DSP 晶片總消耗的電流值

下圖為測試耗電的實際環境圖

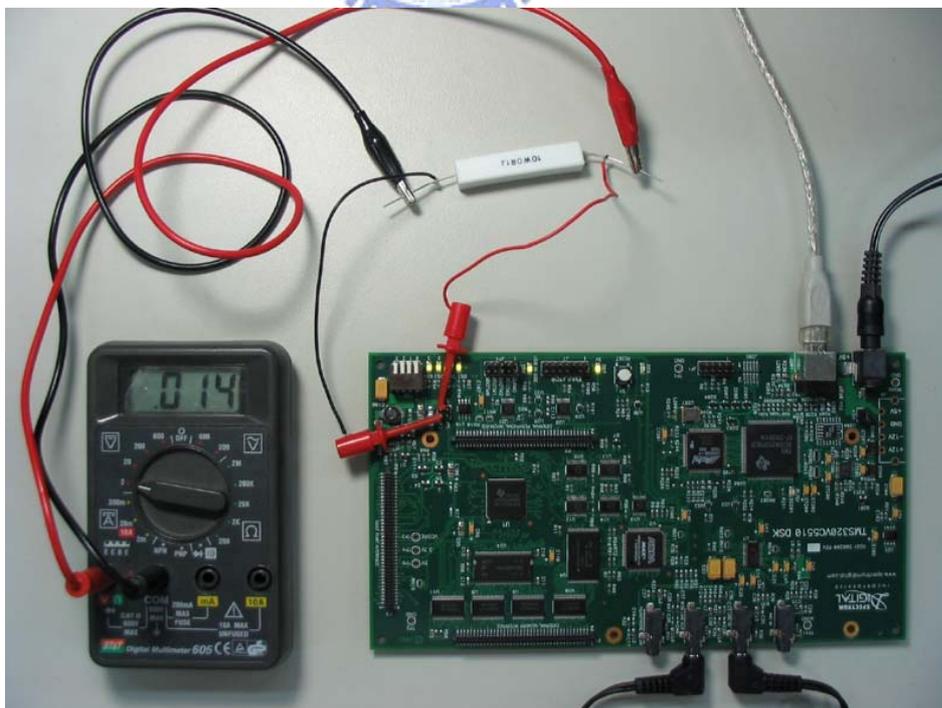


圖 4.4.10 測試 DSP 耗電的實際環境圖

## 五、模擬與實驗結果

在第二章詳細介紹了主動式噪音消除耳機的原理與架構，接著第三章詳細的說明使用的演算法及設計步驟，最後以第四章所提到完整的實驗平台來實現，在此章節，我們會做一些實驗來表現出數位控制器在控制器的階數及耗電性的優點。

### 5.1 實驗步驟及環境簡介

#### 實驗步驟

在本實驗中，我們將利用第三章介紹的演算法設計一四階 IIR 控制器，比較噪音消除的理論值，模擬值(simulink)及實際值的結果，接著設計一八階 IIR 控制器，作相同的實驗，由此看出八階控制器所展現出的優點，且將八階控制器作耗電測試，比較改變時脈和工作電壓後，晶片耗電量的差別，亦表現其省電的特性。最後，將模擬去除八階控制器 CODEC 時間延遲的實驗，估測在移除時間延遲的因素後，系統所能達到的消音效果。

#### 環境簡介

我們利用 Matlab 產生 Band-limited white Noise 和弦波的噪音訊號，整個實驗系統環境設備如下

噪音產生源	:	audiolab 8000B profi(前級擴大機)
		Cambridge audio A70 Power Amplifier(後級擴大機)
		Harbeth monitor loudspeaker(揚聲器)
麥克風訊號量測	:	LeCroy LT354 (示波器)
ANC 耳機平台	:	請參考第四章
		AD/DA 取樣頻率 96kHz
		DSP 運算頻率 32kHz

## 5.2 四階控制器模擬與實驗結果

我們設計控制器參數為  $a=0.7071, b=0.7071, c=0.3, W$  在 50~500Hz 權重值為 0.9，其餘為 0，其計算出連續控制器係數如下

$$C(s) = \frac{-0.032465 - 368.13s^1 - 4.0717 \cdot 10^5 s^2 - 2.3386 \cdot 10^9 s^3 - 9.1477 \cdot 10^{11} s^4}{1.4595 + 2852.6s^1 + 1.0365 \cdot 10^7 s^2 + 8.1675 \cdot 10^9 s^3 + 1.5218 \cdot 10^{13} s^4}$$

其 Fixed-point 控制器係數為

$$C(z) = 0.0222 \frac{-16384 + 27159z^{-1} - 10844z^{-2}}{16384 + -32591z^{-1} + 16308z^{-2}} * \frac{16384 - 31890z^{-1} + 15567z^{-2}}{16384 - 32564z^{-1} + 16222z^{-2}}$$

下圖為加入 ECHO 時間延遲後四個具代表性之耳機系統的頻率響應圖

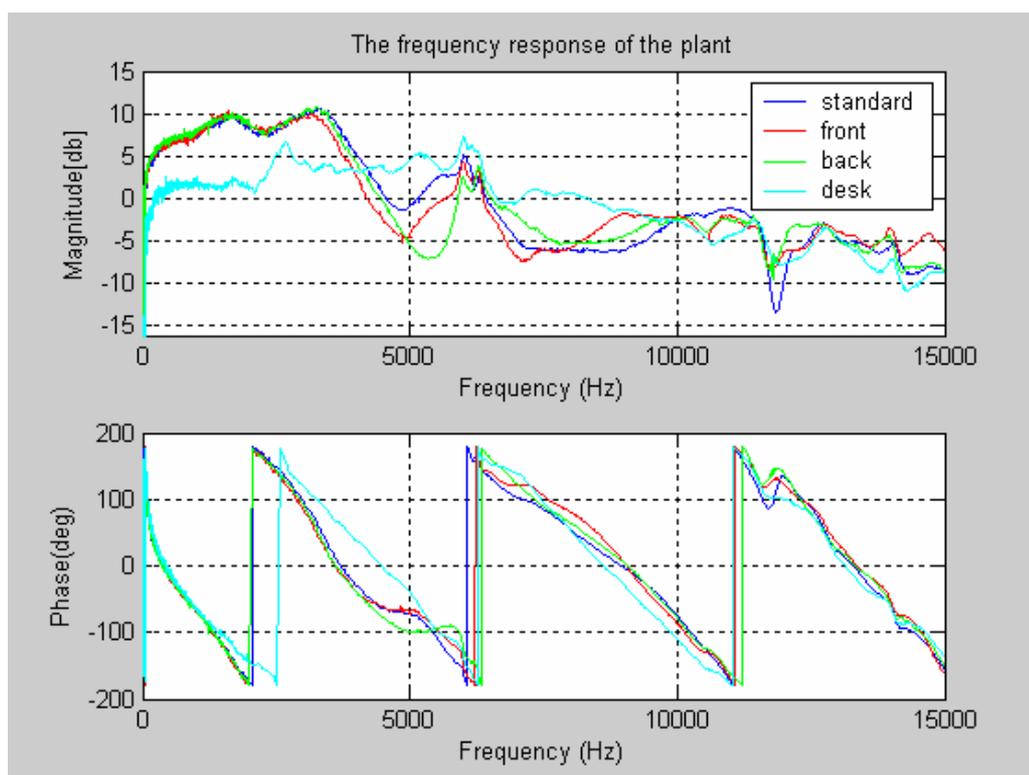


圖 5.2.1 四個具代表性之耳機系統加入 ECHO 時間延遲後的頻率響應

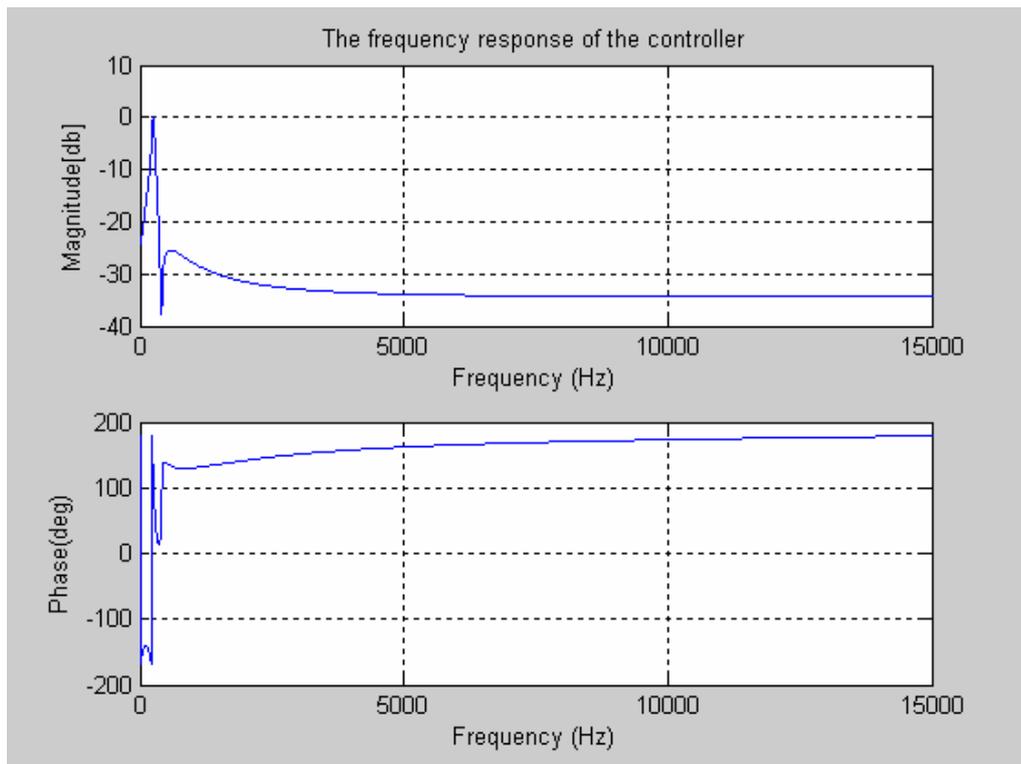


圖 5.2.2 四階控制器理論的頻率響應圖

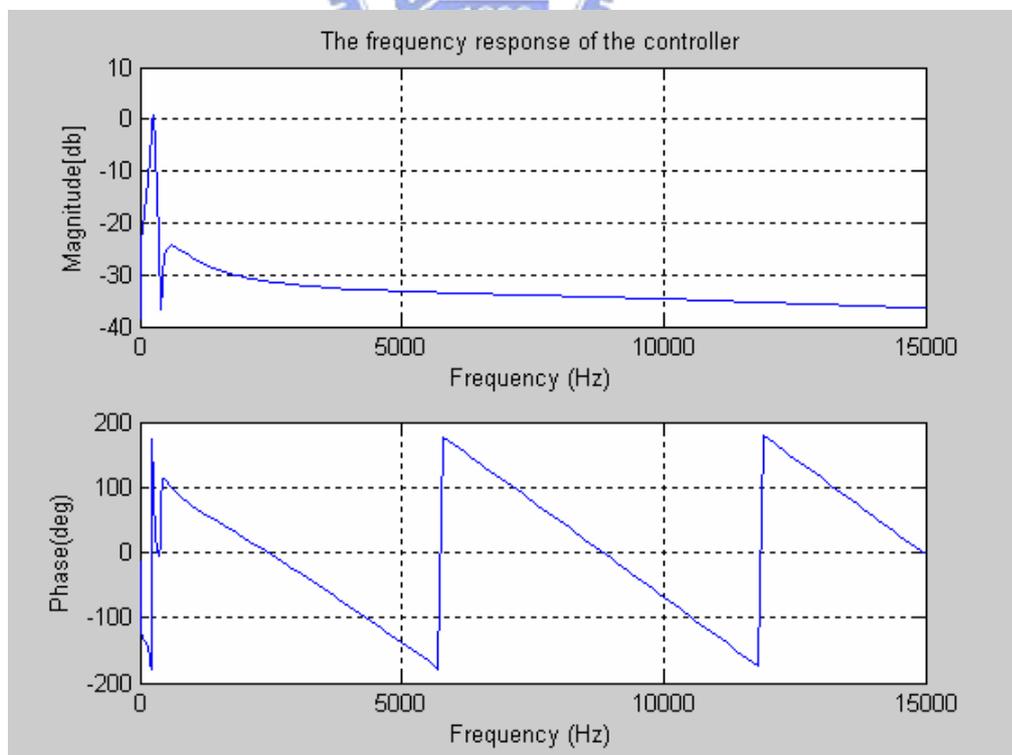


圖 5.2.3 四階控制器實際的頻率響應圖

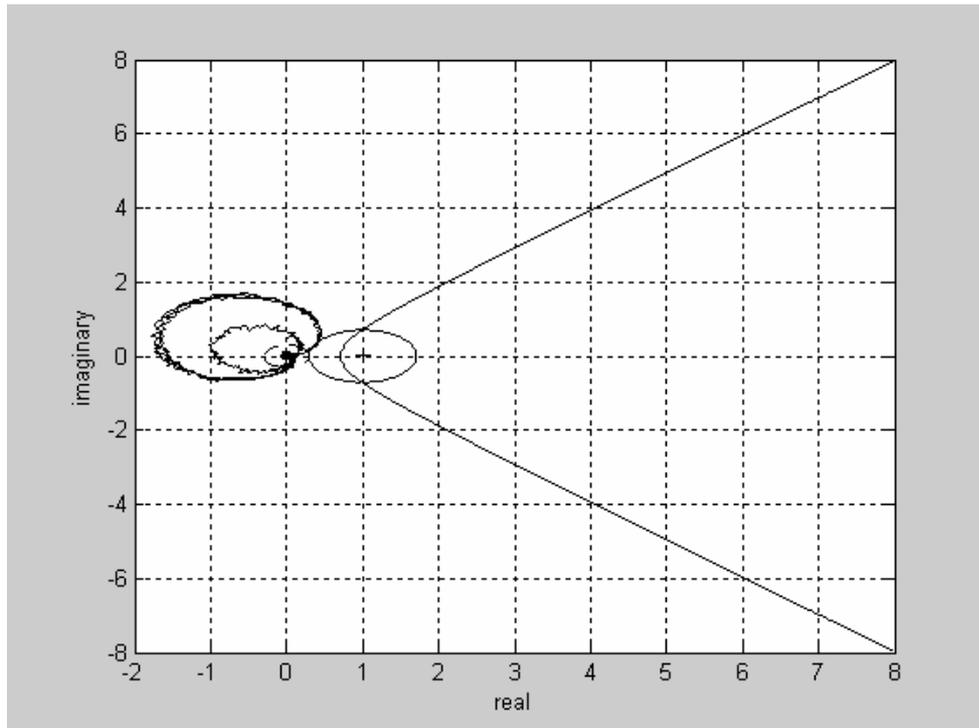


圖 5.2.4 使用四階控制器的 Nyquist 圖

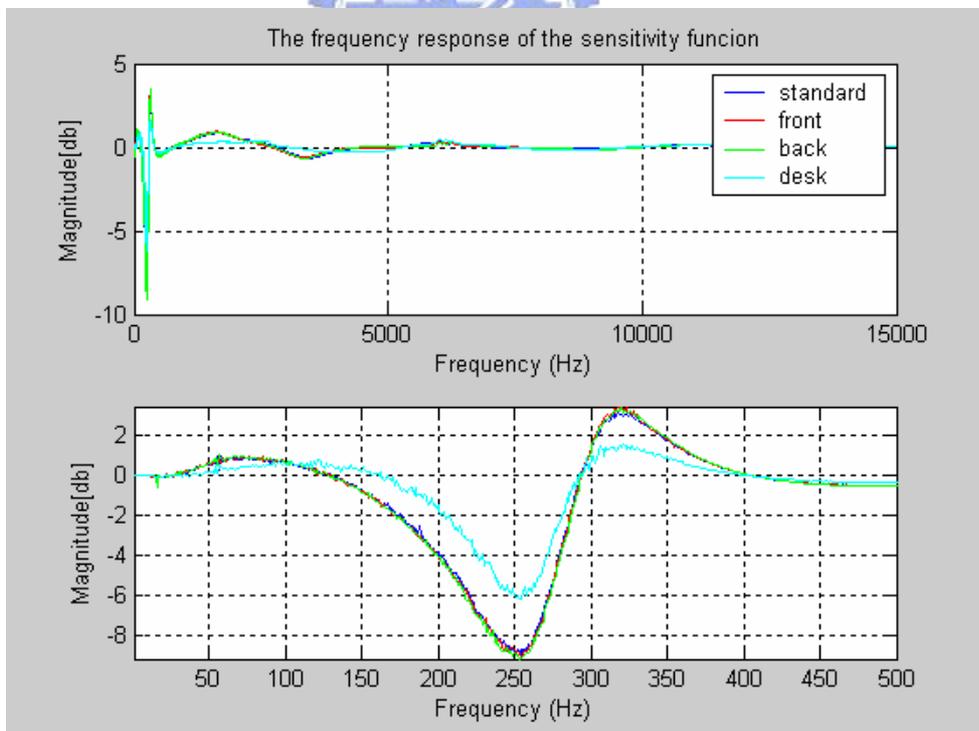


圖 5.2.5 使用四階控制器 sensitivity function 的頻率響應圖

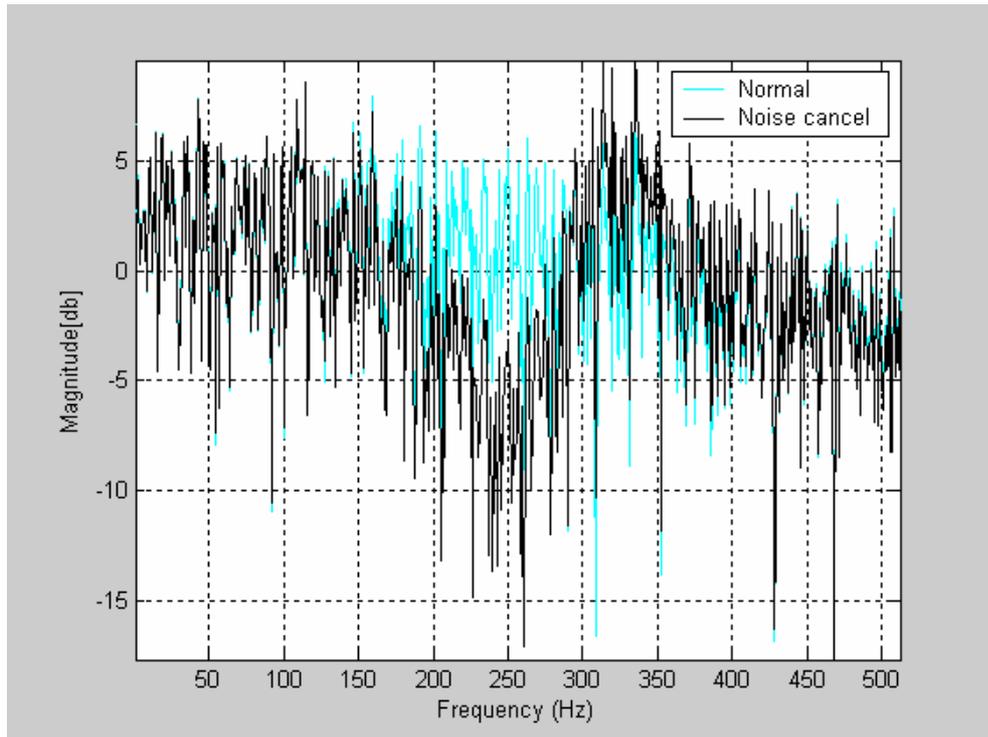


圖 5.2.6 使用四階控制器模擬的 sensitivity function 頻率響應圖

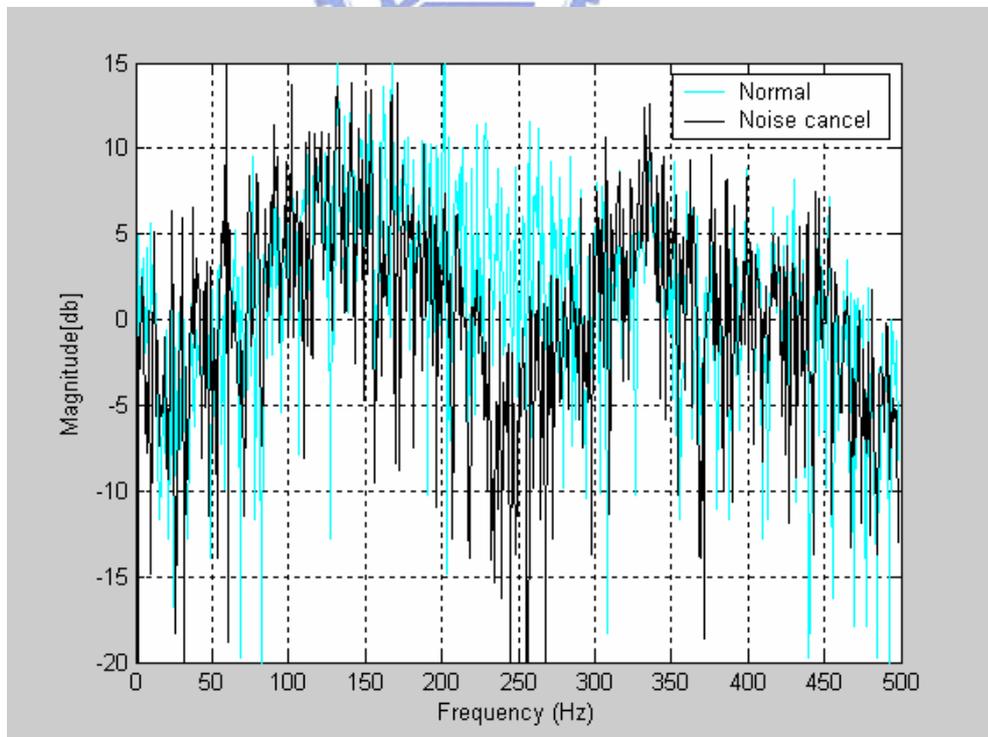


圖 5.2.7 使用四階控制器實際的 sensitivity function 頻率響應圖

### 5.3 八階控制器模擬與實驗結果

我們設計控制器參數為  $a = 0.7071, b = 0.7071, c = 0.3, W$  在 50~500Hz 權重值為 0.9，其餘為 0，其計算出連續控制器係數如下

$$C(s) = \frac{-0.016933 - 833.75s^1 - 2.6511 \cdot 10^6 s^2 - 1.2691 \cdot 10^{10} s^3 - 2.3528 \cdot 10^{13} s^4 - 5.2747 \cdot 10^{16} s^5}{1.577 - 6266.8s^1 - 2.6956 \cdot 10^7 s^2 + 5.7524 \cdot 10^{10} s^3 + 1.3102 \cdot 10^{14} s^4 + 1.6784 \cdot 10^{17} s^5} \\ \frac{-5.1458 \cdot 10^{19} s^6 - 6.3711 \cdot 10^{22} s^7 - 2.0598 \cdot 10^{25} s^8}{+2.4234 \cdot 10^{20} s^6 + 1.5763 \cdot 10^{23} s^7 + 1.4685 \cdot 10^{26} s^8}$$

其 Fixed-point 控制器係數為

$$C(z) = 0.0107 \frac{-16384 + 9255z^{-1} + 6832z^{-2}}{16384 + -31781z^{-1} + 15456z^{-2}} * \frac{16384 - 32331z^{-1} + 16053z^{-2}}{16384 - 31781z^{-1} + 15456z^{-2}} \\ * \frac{16384 - 31781z^{-1} + 15456z^{-2}}{16384 - 32665z^{-1} + 16322z^{-2}} * \frac{16384 - 32665z^{-1} + 16322z^{-2}}{16384 - 32665z^{-1} + 16322z^{-2}}$$

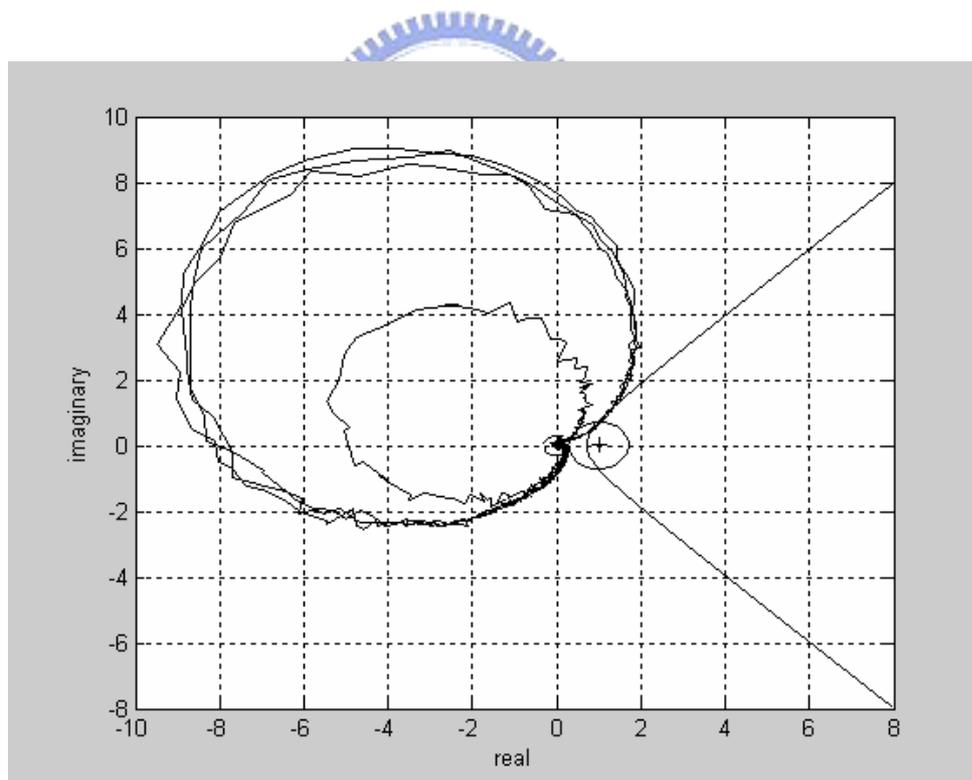


圖 5.3.1 使用八階控制器的 Nyquist 圖

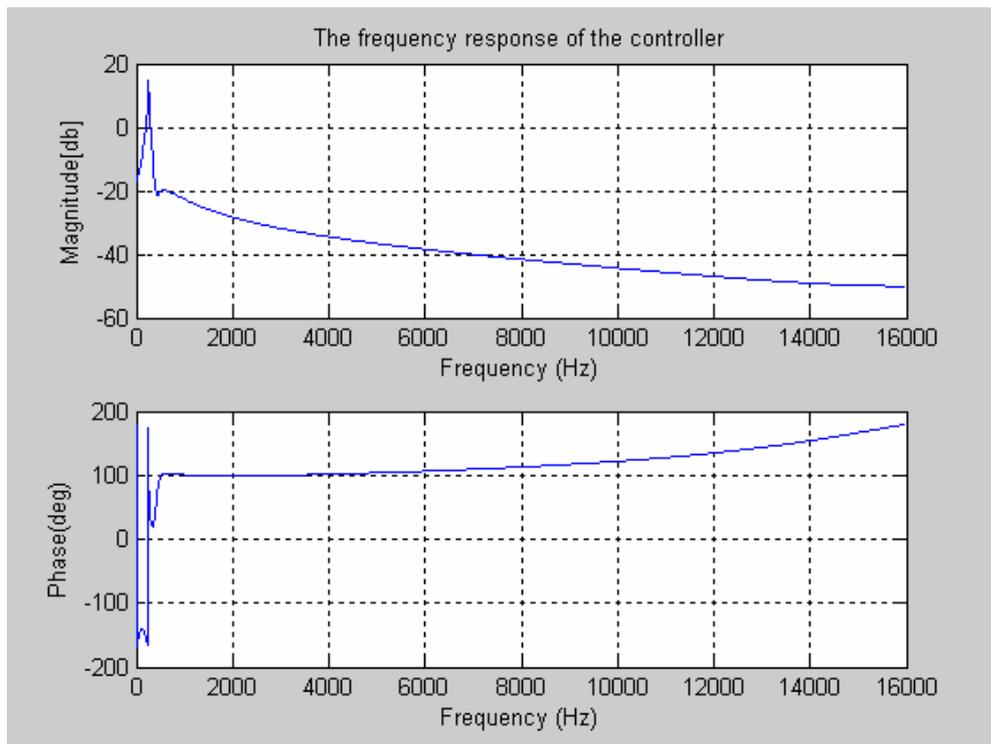


圖 5.3.2 八階控制器理論的頻率響應圖

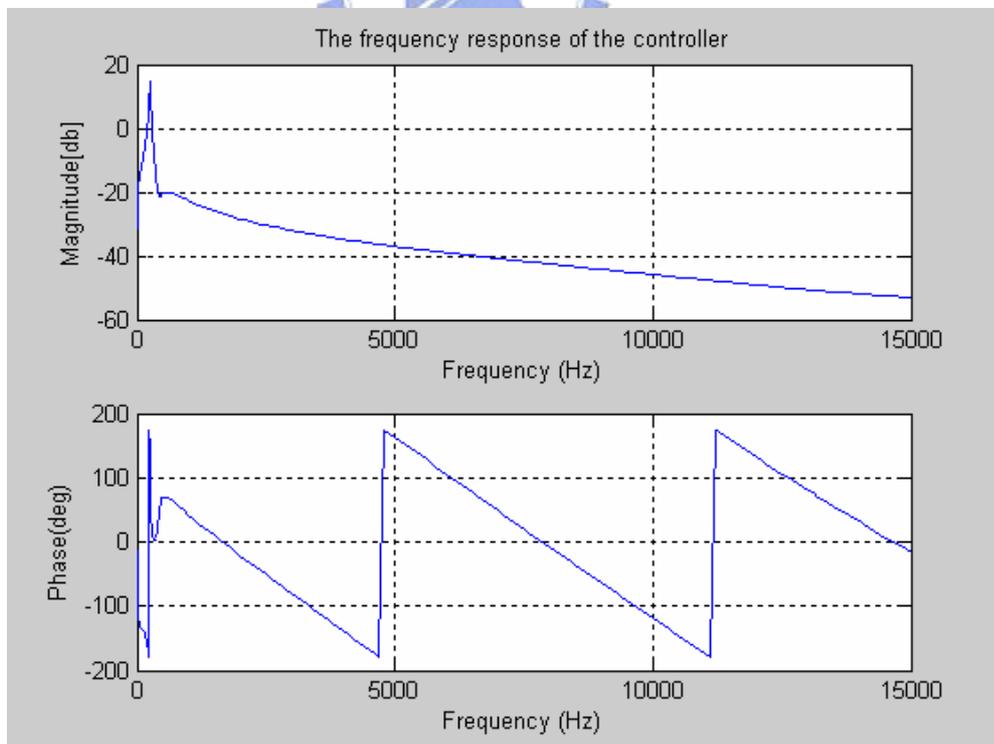


圖 5.3.3 八階控制器實際的頻率響應圖

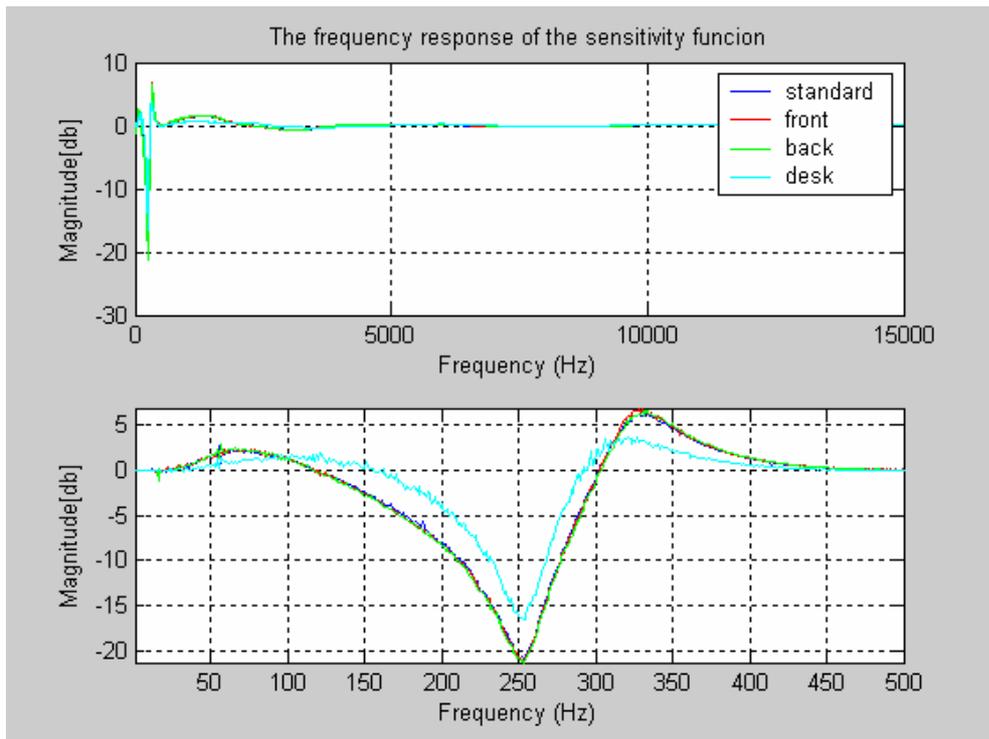


圖 5.3.4 使用八階控制器 sensitivity function 的頻率響應圖

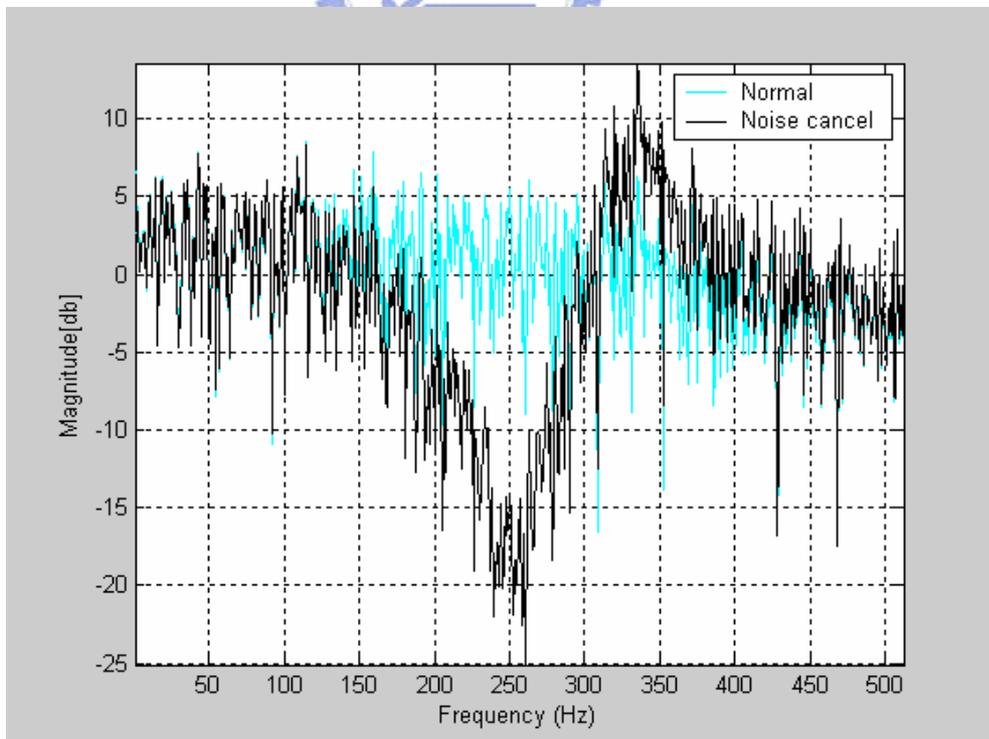


圖 5.3.5 使用八階控制器模擬的 sensitivity function 頻率響應圖

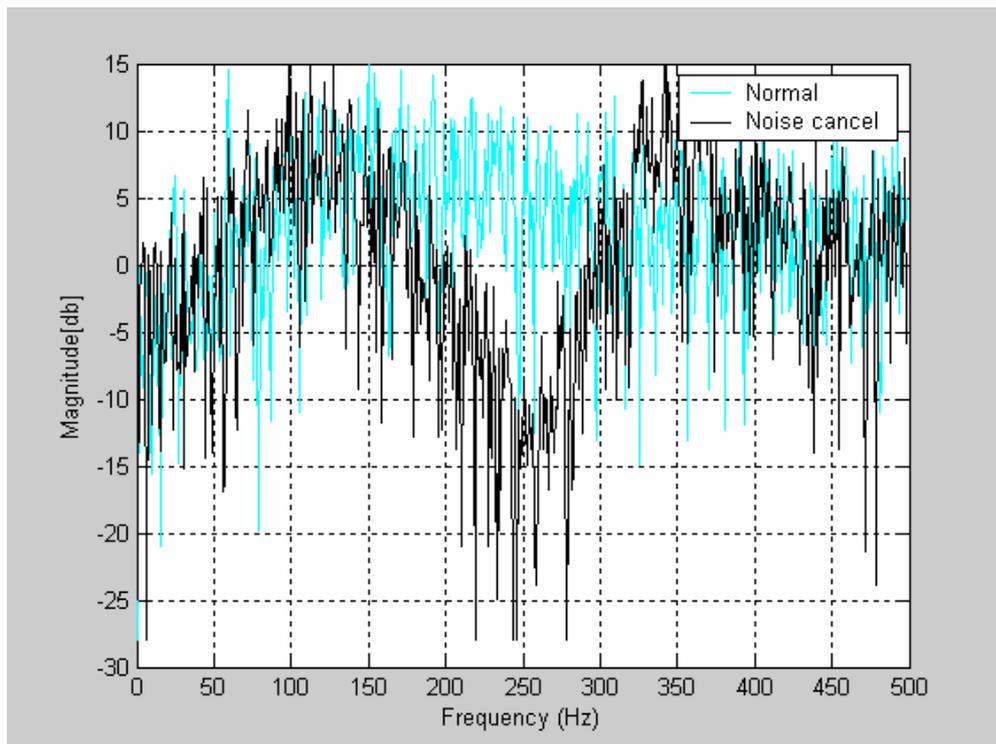


圖 5.3.6 使用八階控制器實際的 sensitivity function 頻率響應圖



## 5.4 耗電實驗測試

在此將用 5.3 節設計的八階IIR控制器做耗電測試，在未做任何調整前，DSP 的工作時脈為 200MHz，工作電壓為 1.6V可從電表量得電壓為 0.014V，而我們藉由DSP/BIOS的STS工具測得DSP處理時間最大值為  $4.2 \times 10^{-6}$ 秒，大約是取樣時間 (1/96k 秒)的 1/2.7 倍，參考圖 5.4.1，所以降低DSP的工作時脈為 100MHz左右並不會影響其正常的運作。

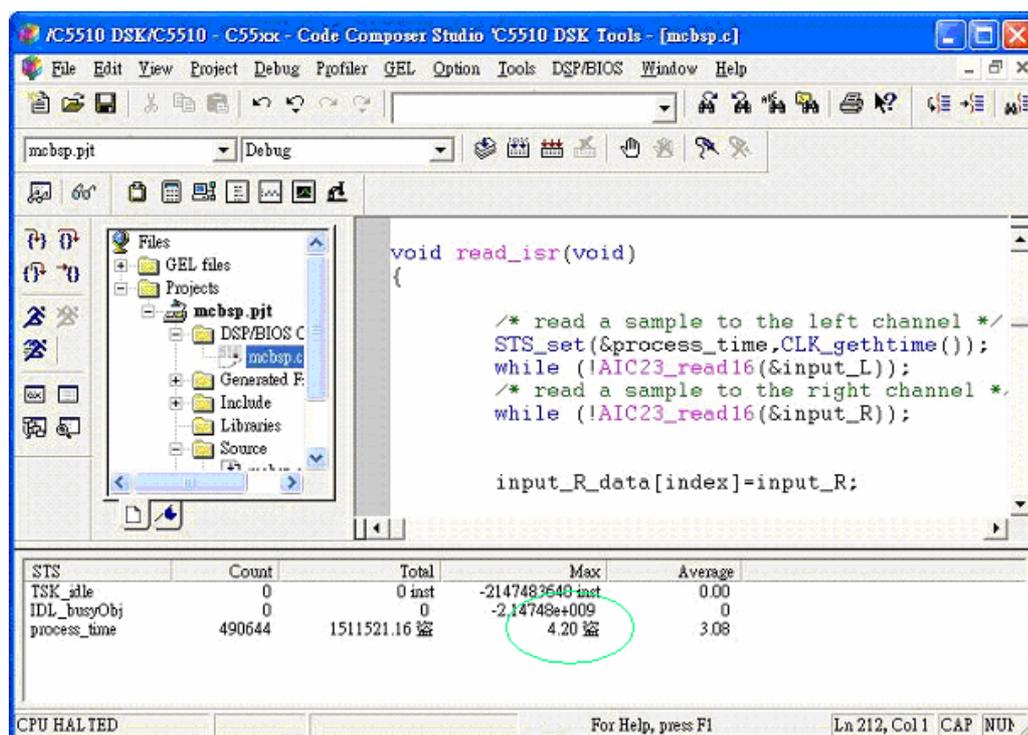


圖 5.4.1 DSP/BIOS 的 STS 工具測得 DSP 處理時間

我們先將 DSP 的工作時脈降成 104MHz，從電表量得電壓為 0.008V，接著再將工作電壓設為 1.1V，可從電表量得電壓為 0.005V，我們將實驗結果整理成以下表格

DSP Clk/Volt	量測電壓值
200MHz,1.6V	0.014V
104MHz,1.6V	0.008V
104MHz,1.1V	0.005V

表 5.4.1 耗電測試表

## 5.5 模擬去除 CODEC 時間延遲的估測結果

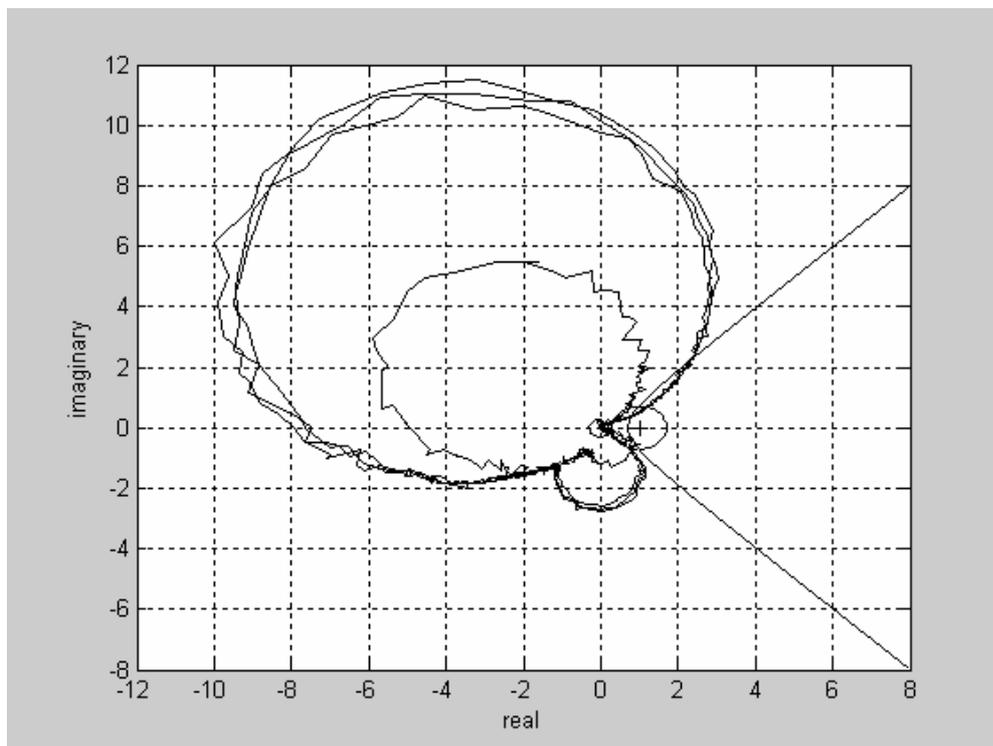


圖 5.5.1 使用八階控制器的 Nyquist 圖

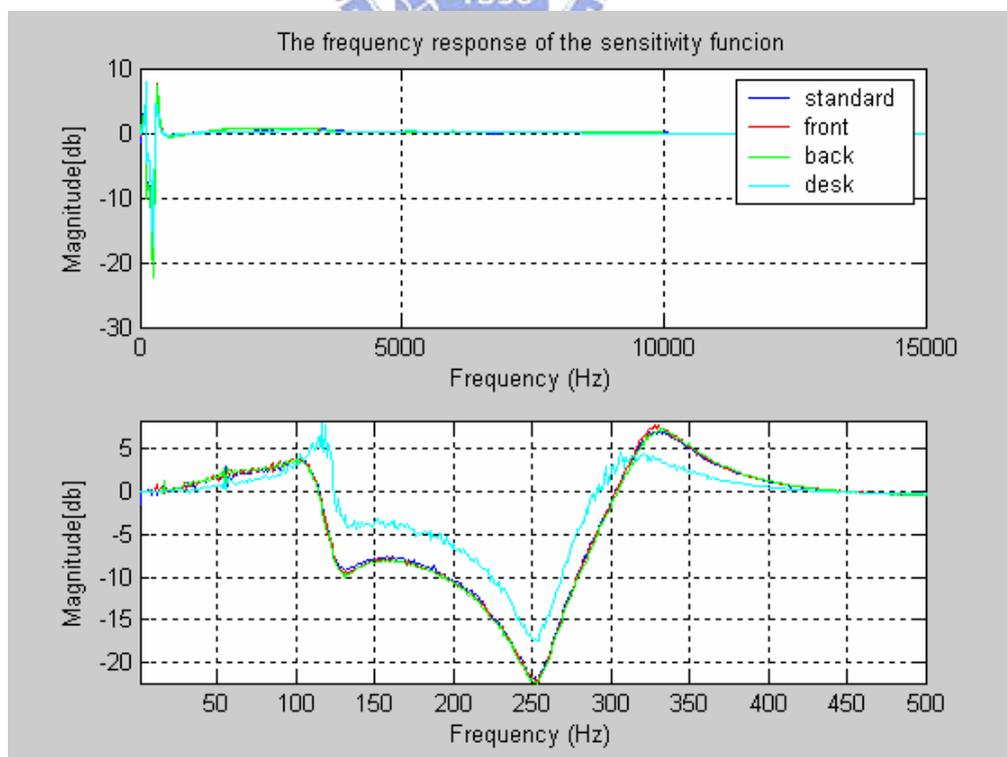


圖 5.5.2 使用八階控制器 sensitivity function 的頻率響應圖

## 5.6 實驗結果之分析與討論

由前面的實驗結果可看出，不論四階或八階的控制器，其模擬及實際測試的結果與理論值都相當接近，可推測模擬平台的環境與實際的硬體平台環境類似，且因為硬體平台的結果與理論值相近，所以相當適合做為實現 ANC 耳機理論的平台。

從比較四階控制器與八階控制器的消音結果，可發現噪音值在頻率 250Hz 的地方，由-8dB 變成-21dB，可知藉由提高階數的方式，增加消音的效果。

由耗電實驗表 5.4.1，在降低 DSP 的工作時脈及工作電壓，使得測量的電壓值由 0.014V 降到 0.005V，降低將近為原來的一半，由於測量電阻值為  $0.16\Omega$ ，可算出 DSP 晶片的總消耗電流為  $0.005/0.16=0.03125\text{A}$ ，因工作電壓為 1.1V，所以 DSP 總消耗功率為  $0.03125*1.1=0.034375\text{W}$ 。

最後，在模擬去除八階控制器 CODEC 時間延遲的實驗，比較圖 5.3.2 和圖 5.5.2 可發現去除 CODEC 時間延遲的因素後，在低頻的地方（約 130Hz），消音值由-3dB 改善到-10dB，增大了消音的頻帶，所以如改善時間延遲的問題後，可達到更好的消音效果。

## 六、結論

本論文的目標分為兩部分，一部份是 ANC 實驗平台的建立，另一部份是研究數位控制器的優點。目前在 ANC 實驗平台方面成果如下

- 一、 測試平台的自動化
  - 減少人工操作的失誤及達到測量工作的自動化
- 二、 利用 simulink 建立模擬環境
  - 可加速設計控制器的速度
- 三、 使用 C5510 DSP 實現硬體平台
  - 其實現結果與理論值相當接近，相當適合做為實現 ANC 耳機理論的平台

在研究數位控制器的優點方面，我們利用前面所提的平台實現了八階的數位控制器，也證明利用高階的控制器可讓噪音消除的效果變好，未來可嘗試別的演算法（例： $H_{\infty}$ 最佳化控制），設計更高階的控制器。

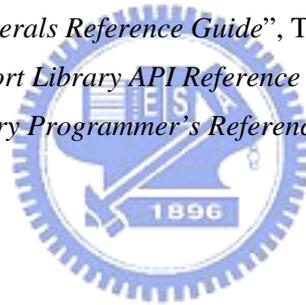
在硬體平台方面，由於我們使用 Fixed-point DSP 且利用組語編寫相關程式碼，達成省電且程式執行效率高的特性，便於未來可繼續增加功能。

而使用 DSK5510 實現硬體平台時，發現其上的 CODEC 晶片造成相當大的時間延遲，雖然可藉由控制器提高階數的方法來改善其效能，但是，如果能去除時間延遲的因素，相信能達更好更穩定的抑制效果。

## 參考文獻

- [1] M.O.Tokhi , R.R. Leitch, Active Noise Control, Oxford Engineering science Series, Oxford Science Publications,1991
- [2] P.Lueg, “Process of Silencing Sound Oscillations,”U.S. Patent No.2043416.
- [3] H.F. Olson, E.G. May, “Electronic sound absorber,”Journal of the Acoustical Society of America,28,pp.1130-1136,1953.
- [4] H.F. Olson, Acoustic Engineering, Van Nostrand Company INC ,1957.
- [5] S.J.Elliott, P.A.Nelson, ”Active noise Control”, IEEE Signal Processing Magazine,pp.12-35, October 1993.
- [6] J.F. Abbott, ”Acoustical Design Criteria For Active Noise Control System”, IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol. 15, 1993,pp. 630-3
- [7] J.C.Doyle, B.A.Francis, A.R.Tannenbaum, Feedback control Theory, Macmillan Publishing Company, New York, 1992.
- [8] V.R.Sule,V.V.Athani,”Directional Sensitivity Tradeoffs in Multivariable Feedback System”,Automatica, Vol 27, No 5, pp. 869-872,1991.
- [9] J.Freudenberg, D.P.Looze, “Right Half Plane Poles and Zeros and Dsign tradeoffs in Feedback System”, IEEE trans. Aut. Control, AC-30, No 6, June, 1985.
- [10] Shiang-Hwua Yu; Jwu-Sheng Hu, ”Controller Design for Active Noise Cancellation Headphones Using Experimental Raw Data”, IEEE/ASME Transactions on Mechatronics , Volume: 6 Issue: 4 , Dec. 2001 page(s): 483 -490
- [11] P. C. Parks and V. Hahn, Stability Theory. Englewood Cliffs, NJ: Prentice-Hall, 1993.
- [12] E. Polak, “A modified Nyquist stability test for use in computer-aided design,” IEEE Trans. Automat. Contr., vol. 29, pp. 91–93, Jan. 1984.
- [13] B. A. Francis and G. Zames, “On  $H^\infty$ -optimal sensitivity theory for SISO feedback systems,” IEEE Trans. Automat. Contr., vol. 29, pp. 9–16, Jan. 1984.
- [14] J. S. Freudenberg and D. P. Looze, “Right half plane poles and zeros and design tradeoffs in feedback systems,” IEEE Trans. Automat. Contr., vol. 30, pp. 555–565, June 1985.
- [15] E. Polak, D. Q. Mayne, and D. M. Stimler, “Control system design viasemi-infinite optimization: A review,” Proc. IEEE, vol. 72, no. 12, pp. 1777–1794, 1984.
- [16] S. P. Boyd et al., “A new CAD method and associated architectures for linear controllers,” IEEE Trans. Automat. Contr., vol. 33, pp. 268–279, Mar. 1988.

- [17] J. W. Helton and A. Sideris, "Frequency response algorithms for  $H_\infty$  optimization with time domain constraints," IEEE Trans. Automat. Contr., vol. 34, pp. 427–434, 1989.
- [18] B. Rafaely and S. J. Elliott, " $H_2=H_\infty$  active control of sound in a headrest: Design and implementation," IEEE Trans. Contr. Syst. Technol., vol. 7, pp. 79–84, Jan. 1999.
- [19] Lim and J. S. Arora, "An active set RQP algorithm for engineering design optimization," Comput. Methods Appl. Mech. Eng., vol. 57, pp. 51–65, 1989.
- [20] G. A. Gabriele and T. J. Beltracchi, "An investigation of Pshenichnyi's recursive quadratic programming method for engineering optimization," J. Mech., Transmissions Automation Design, ASME, vol. 109, no. 6, pp. 248–253, 1987.
- [21] C. H. Tseng and J. S. Arora, "On implementation of computational algorithms for optimal design 2: Extensive numerical investigation," Int. J. Numer. Methods Eng., vol. 26, pp. 1383–1402, 1988.
- [22] 黃怡智,「主動式消音耳機系統之迴授控制器設計與實現」,國立交通大學,碩士論文,1997.
- [23] 胡竹生,賴鴻志,張勝凱, TMS320C54xx晶片原理與應用,全華, 2001
- [24] Edwin K.P.Chong and Stranislav H. Zak, An Introduction to Optimization, WILEY INTER-SCIENCE,2001
- [25] "TMS320C55x DSP Peripherals Reference Guide", Texas Instruments.
- [26] "TMS320C55x Chip Support Library API Reference Guide", Texas Instruments
- [27] "TMS320C55x DSP Library Programmer's Reference 1988", Texas Instruments



# 附 錄



## A-1 DSP 主程式程式碼 1 (C language)

```
#include <std.h>
#include <log.h>
#include "mcbbsp.h"
#include "mcbspcfg.h"
#include "iircas51_mod.h"
#include <csl_irq.h>
#include "dsk5510.h"
#include "dsk5510_dip.h"
#include "dsk5510_led.h"
#include "QP_filter.h"          // filter coefficients

/* Codec configuration settings */
DSK5510_AIC23_Config config = { \
    0x0002, /* 0 DSK5510_AIC23_LEFTINVOL Left line input channel volume \
    0x0002, /* 1 DSK5510_AIC23_RIGHTINVOL Right line input channel volume */ \
    0x01f9, /* 2 DSK5510_AIC23_LEFTHPVOL Left channel headphone volume */ \
    0x01f9, /* 3 DSK5510_AIC23_RIGHTHPVOL Right channel headphone volume */ \
    0x0011, /* 4 DSK5510_AIC23_ANAPATH Analog audio path control */ \
    0x0000, /* 5 DSK5510_AIC23_DIGPATH Digital audio path control */ \
    0x0000, /* 6 DSK5510_AIC23_POWERDOWN Power down control */ \
    0x0043, /* 7 DSK5510_AIC23_DIGIF Digital audio interface format */ \
    0x009D, /* 8 DSK5510_AIC23_SAMPLERATE Sample rate 48KHz control */ \
    0x0001 /* 9 DSK5510_AIC23_DIGACT Digital interface activation */ \
};

short t=0;
Int16 input_L, input_R, zero=0;
Int16 output_L, output_R;
Int16 delayBufferL[4*NBIQ+2]; // Delay Buffers for DSPLIB routine
Int16 delayBufferR[4*NBIQ+2]; // Delay Buffers for DSPLIB routine

void AIC23_openCodec(DSK5510_AIC23_Config *Config);
void AIC23_rset(UInt16 regnum, UInt16 regval);
void AIC23_config(DSK5510_AIC23_Config *Config);
CSLBool AIC23_read16(Int16 *val);
```

```

CSLBool AIC23_writel6(Int16 val);
Uint16 rcvEventId, xmtEventId;
Uint16 old_intm;

void main()
{
    //----- initial all compoment -----
    short i;

    /* Initialize AIC23 and MCBSP */
    AIC23_openCodec(&config);

    /* Initialize the board support library, must be called first */
    DSK5510_init();

    /* Initialize the LED modules of the BSL */
    DSK5510_LED_init();

    /* set the content of delay buffer to zero */
    for (i=0;i<(4*NBIQ+2);i++) delayBufferL[i]=0;
    for (i=0;i<(4*NBIQ+2);i++) delayBufferR[i]=0;
}

void read_isr(void)
{
    /* read a sample to the left channel */
    while (!AIC23_readl6(&input_L));
    /* read a sample to the right channel */
    while (!AIC23_readl6(&input_R));

    if (t==0)
    {
        if (DSK5510_DIP_get(3) == 0)
        {
            /*IIR Filter */
            iircas51_mod(&input_L, COEFFS_L, &output_L, delayBufferL, NBIQ, 1);

```

```

    DSK5510_LED_on(3);
}
else
{
    output_L=zero;
    output_R=zero;

    DSK5510_LED_off(3);
}
}

if (t==1)
{
if (DSK5510_DIP_get(3) == 0)
{
/*IIR Filter */
iircas51_mod(&input_R, COEFFS_R, &output_R, delayBufferR, NBIQ, 1);

    DSK5510_LED_on(3);
}
else
{
    output_L=zero;
    output_R=zero;

    DSK5510_LED_off(3);
}
}

t++;
if (t >= 3)
{
    t=0;
}

/* Send a sample to the left channel */
while (!AIC23_writel6(output_L));
/* Send a sample to the right channel */

```

```

        while (!AIC23_writel6(output_R));
    }
void AIC23_openCodec(DSK5510_AIC23_Config *Config)
{
    Uint32 delay;

    /*
     * Initialize the AIC23 codec
     */

    /* Start McBSP1 as the codec control channel */
    McBSP_start(Mcbbsp1_control, MCBSP_XMIT_START |
    MCBSP_SRGR_START | MCBSP_SRGR_FRAMESYNC, 100);

    /* Reset the AIC23 */
    AIC23_rset(DSK5510_AIC23_RESET_ADDRESS, 0);

    /* Configure the rest of the AIC23 registers */
    AIC23_config(Config);

    /* waiting the Codec become stabile */
    for (delay = 0; delay < 20000000; delay++);

    //set the interrupt for MCBSP2

    /* Clear any garbage from the codec data port */
    if (MCBSP_rrdy(Mcbbsp2_data))
        MCBSP_readl6(Mcbbsp2_data);

    /* Temporarily disable all maskable interrupts */
    old_intm = IRQ_globalDisable();

    /* Get Event Id's associated with MCBSP 2 receive */
    /* The MCBSP_Handle object, Mcbsp2_data has been predefined */
    /* in the code automatically generated by the DSPBIOS/CCSL */
    /* GUI config */
}

```

```

rcvEventId = MCBSP_getRcvEventId(Mcbsp2_data);

/* Clear any pending receive interrupts */
IRQ_clear(rcvEventId);

/* Start McBSP2 as the codec data channel */

MCBSP_start(Mcbsp2_data, MCBSP_XMIT_START | MCBSP_RCV_START |
MCBSP_SRGR_START | MCBSP_SRGR_FRAMESYNC, 220);

/* Enable receive interrupts */
IRQ_enable(rcvEventId);

/* Enable all maskable interrupts */
IRQ_globalRestore(old_intm);
}

void AIC23_rset(Uint16 regnum, Uint16 regval)
{
    /* Mask off lower 9 bits */
    regval &= 0x1fff;

    /* Wait for XRDY signal before writing data to DXR */
    while (!MCBSP_xrdy(Mcbsp1_control));

    /* Write 16 bit data value to DXR */
    MCBSP_writel6(Mcbsp1_control, (regnum << 9) | regval);
}

/*
 * ===== DSK5510_AIC23_config =====
 * Set the default codec register config values
 */
void AIC23_config(DSK5510_AIC23_Config *Config)

```

```

{
    int i;

    /* Assign each register */
    for (i = 0; i < DSK5510_AIC23_REG_NUMREGS; i++)
        AIC23_rset(i, Config -> regs[i]);
}

CSLBool AIC23_read16(Int16 *val)
{
    /* Read the data */
    if (!MCBSP_rrdy(Mcbasp2_data)) {
        return (FALSE);
    }
    *val = MCBSP_read16(Mcbasp2_data);
    return (TRUE);
}

CSLBool AIC23_writel6(Int16 val)
{
    /* Wait for XRDY signal before writing data to DXR */
    if (!MCBSP_xrdy(Mcbasp2_data)) {
        return (FALSE);
    }

    /* Write 16 bit data value to DXR */
    MCBSP_writel6(Mcbasp2_data, val);

    return(TRUE);
}

```

## A-2 DSP 主程式程式碼 2 (C language)

```
#define NBIQ          4    //Number of biquads

#pragma DATA_SECTION (COEFFS_L, "coeff_sect1");
// 0.5*y(n)=b0*x(n)+b1*x(n-1)+b2*x(n-2)-a1*y(n-1)-a2*y(n-2)
Int16 COEFFS_L[5*NBIQ] = {
/* C55x: b0 b1 b2 a1 a2 ... */

//fs=32k Hz
// 4 order filter , AIC23_Config=-13.9060
// -16384, 30129, -13776, -31547, 15250,
// 8192, -16468, 8334, -32583, 16234,

// 8 order filter , AIC23_Config=-26.254523
-16384, 9255, 6832, -31781, 15456,
16384, -32331, 16053, -31781, 15456,
16384, -31781, 15456, -32665, 16322,
16384, -32665, 16322, -32665, 16322
};

#pragma DATA_SECTION (COEFFS_R, "coeff_sect1");
// 0.5*y(n)=b0*x(n)+b1*x(n-1)+b2*x(n-2)-a1*y(n-1)-a2*y(n-2)
Int16 COEFFS_R[5*NBIQ] = {
/* C55x: b0 b1 b2 a1 a2 ... */

// fs=32k Hz
// 4 order filter , AIC23_Config=-22.037167
// -16384, 27159, -10844, -31890, 15567,
// 16384, -32591, 16308, -32564, 16222

// 8 order filter , AIC23_Config=-26.254523
-16384, 9255, 6832, -31781, 15456,
16384, -32331, 16053, -31781, 15456,
16384, -31781, 15456, -32665, 16322,
16384, -32665, 16322, -32665, 16322
};
```

### A-3 DSP 組語程式碼

```
.arms_off
.cpl_off
.mmregs
;*****
;*      Assigns
;*****
    .def_iircas5l_mod

;*****
;*      Constant Defines
;*****

;*****
;*      Variable Declarations
;*****

;*****
; Program section
;*****
    .text
_iircas5l_mod
;Context save

    PSH mmap(ST0_55)
    PSH mmap(ST1_55)
    PSH mmap(ST2_55)
    PSH mmap(ST3_55)

    BCLRCPL          ;DP relative addressing

    PSH mmap(DP)
    MOV #0, DP
    .dp 0

    PSH @T3_L
```

```

PSHBOTH XAR4
PSHBOTH XAR5
PSHBOTH XAR7

```

```

;initialization

```

```

BSETXMD      ;sign extension enable
BCLRSATD     ;D-unit saturate disable
BSETFRCT     ;fractional mode enable
BCLRSATA     ;A-unit saturate disable

```

```

;Save pointer to index at dbuffer[0]

```

```

MOV XAR3, XAR5

```

```

;setup circular addressing

```

```

MOV T0, T3      ;compute 2*nbiq

```

```

SFTS T3, #1

```

```

MOV @T3_L, BK03 ;init AR0-3 circular buf size (2*nbiq)

```

```

MOV @T3_L, BK47 ;init AR4-7 circular buf size (2*nbiq)

```

```

BSETAR3LC     ;init AR3 = circular (dbuffer)

```

```

ADD #1, AR3   ;adjust AR3 to buffer start

```

```

MOV @AR3_L, BSA23 ;init AR2-3 circular start addr: dbuffer(1)

```

```

BSETAR4LC     ;init AR4 = circular (dbuffer)

```

```

MOV XAR3, XAR4 ;adjust AR4 to buffer start

```

```

ADD T3, AR4

```

```

MOV @AR4_L, BSA45 ;init AR45 circ start addr: dbuffer(1+nbiq)

```

```

MOV *AR5, AR3  ;init AR3 offset to x(n) buffer start

```

```

MOV *AR5, AR4  ;init AR4 offset to y(n) buffer start

```

```

SUB #1, T1, T3 ;compute nx-1

```

```

MOV T3, BRC0   ;init outer loop counter (nx-1)

```

```

MOV XAR1, XAR7 ;save original value to reinitialize coeff buffer ptr

```

```

SUB #1, T0, T3 ;init inner loop counter (#bi-quads-1)

```

```

MOV T3, BRC1

```

```

    ADD #1, T0, T1      ;index for buffer reset

; Kernel
; XAR0: x[] input
; XAR1: h[] coefficients
; XAR2: r[] result
; XAR3: dbuffer[x]
; XAR4: dbuffer[y]
; XAR7: reinit XAR1

RPTBLOCAL  loop1-1      ;outer loop: process a new input

MOV *AR0+ << #16, AC1   ; HI(AC1) = x(n)
||RPTBLOCAL loop2-1     ;inner loop: process a bi-quad

MPYM*AR1+, AC1, AC0     ; AC0 = b0*x(n)

MACM*AR1+, *(AR3+T0), AC0 ; AC0 += b1*x(n-1)

MACM*AR1+, *AR3, AC0; AC0 += b2*x(n-2)

MOV HI(AC1), *AR3      ; x(n) replaces x(n-2)
||AADD T1, AR3         ; point to next x(n-1)

MASM*AR1+, *(AR4+T0), AC0 ; AC0 -= a1*y(n-1)

MASM*AR1+, *AR4, AC0; AC0 -= a2*y(n-2)

;*****
; the program change the value of a0 from 1 ot 0.5
SFTS AC0, #1 ;a0=0.5
;*****
MOV rnd(HI(AC0)), *AR4 ; y(n) replaces y(n-2)
||AADD T1, AR4 ;point to next y(n-1)
;
MOV AC0, AC1 ;input to next biquad

```

loop2:

```
MOV XAR7, XAR1      ;reinitialize coeff pointer
```

```
MOV rnd(HI(AC0)), *AR2+ ;store result to output buffer
```

loop1:

```
; Signal overflow
```

```
MOV #0, T0
```

```
XCC check1, overflow(AC0)
```

```
MOV #1, T0
```

check1:

```
;Save last index value
```

```
MOV AR3, *AR5      ;save index to dbuffer[0]
```

```
;Context restore
```

```
POPBOTH XAR7
```

```
POPBOTH XAR5
```

```
POPBOTH XAR4
```

```
POP @T3_L
```

```
POP mmap(DP_L)
```

```
POP mmap(ST3_55)
```

```
POP mmap(ST2_55)
```

```
POP mmap(ST1_55)
```

```
POP mmap(ST0_55)
```

```
RET
```

