低價位語音辨識－HT48R50 微控制器之應用

雷智偉(1) 、 陳德龍(2) 、鍾啟仁(3)

一、 前言：

在現今的科技發展中，許多電器用品，如：手機、冰箱…等等，已經漸漸加入了語音辨識的功能，所以如果在控制方面上只使用手動輸入，似乎已經不符合時代的潮流！語音辨識控制就成了一個便又好用的方法，也逐漸的變成一種趨勢了！而現在市面上語音辨識系統，不是價格太貴，就是使用及操作複雜，且辨識率不高，對一些只需簡單的控制指令即可動作的系統，似乎有點大才小用，且成本太高。於是我們就興起了利用微控制器來作語音辨識的想法。利用一些簡單的電路及微控制器，如此低成本的組合來達到操控簡單且辨識率高的小系統，也儘可能完全發揮微控制器數學運算的能力。

二、語音辨識簡介：

一般語音辨識在製做功能上可分為:

- 所要辨認字彙的多寡：

(1) 特定字彙：如一般的指令,字詞或單字片語

(2) 少量字彙：如 100 個左右的單字,和詞彙

(3) 大量字彙：沒有特定字彙或範圍很大的指令

在辨認字彙方面，當字彙越多時，辨認時有可能被混淆的機會就越大，相對的困難度也就會提高。

- 所要辨認使用者的限制：

(1) 特定語者(Speaker Dependent)：使用者在辨認前，必須先將語音的參考樣本存入系統中當作辨識時的特徵參數，也就是說使用者在辨識前先要讓系統學習，此即所謂『訓練階段（Training Phase)』。由於它是針對特定使用者所建立的資料庫，

(2) 非特定語者(Speaker Independent)
就是說使用者在辨識前，不需事先訓練系統就能直接進行辨識！若要做到不必學習，而又有很好的辨識率，那是很不容易的！首先必須收集大量且具有代表性的語音特徵參數，然後再建立有效的語音辨識資料庫，這樣的系統較不易在低成本的訴求上實現。

- 使用者說話方式是否連續：

(1) 單字音（Isolated Word）辨識：所要辨識的指令需每一個字都分開。

(2) **連續語音辨識（**Continuous Speech**）：以一般人平常時所說話的方式去辨識語音或指令，但這有牽扯到語音的混音、說話時之速度以及相連音的問題，所以難度也比非連續語音辨識要來的高。**

本辨識系統是針對特定使用者之單字音（Isolated Word**）辨識，因為在設計時的考量是簡單易用、壓低成本，所以用此採用特定語者**(Speaker Dependent)**、單字音（**Isolated Word**）辨識的方式。**

語音辨識的特徵取樣又可分為：

(1) **能量（**Energy**）特徵參數：依照語音輸入時的振幅和頻率來作為語音辨識的特徵，能量特徵參數通常都使用於語音辨識時的起始端和結束端的端點偵測（**End Point Detection**）。**

(2) **頻域（**Frequency Domain**）特徵參數：可以照語音輸入時的頻率分布來作為語音辨識的特徵，頻域特徵參數大多是辨識的主軸，也是語音辨識中極為重要的參數，通常辨識率的高低都取決於頻域參數所使用的適當性。不過由於頻域之特徵參數，通常都需要複雜的數學計算，如快速傅利葉轉換（**Fast Fourier Transform**；FFT**），所以要在小型單晶片上實現實屬不易，有些先運用硬體電路（如帶通濾波器；**Band-Pass Filter**）求出頻域之參數後，再以微控制器加以處理，本專題所採用的就是類似此種技術之作法。**

(3) **時域（**Time Domain**）特徵參數：可利用語音輸入時的時間長度以及時域上之特徵參數，如零交率**(Zero-Crossing Rate)**、能量變化、…等等來作為語音辨識的特徵，或是使用時域搭配頻域特徵參數，在某一段時間內所相對應的頻率取出特徵參數；這也是辨識時不可缺少的參數。**

而本專題在特徵取樣時，利用能量特徵參數來做語音辨識的端點偵測，利用頻域特徵參數來做語音特徵的主要依據，但是由於頻域特徵參數的計算通常都需要複雜的計算過程（如 FFT**），對小型的微控制器而言並非易事，這也是本專題的一項挑戰。**

語音特徵資料的處理：

人的語音頻率範圍大都介於 50HZ~3.4KHz**，而語音基頻範圍是介於** 50HZ~800HZ**，語音訊號是由一連串不同的基頻（**Pitch**）和基頻的二倍頻三倍頻及音與音之間的相連音所組成，我們可以利用特殊的比對演算法，將語音中不需要而又重複的基頻去除，或是用木桶排序法將相同頻率的參數做時間軸的量化，這樣就可以做到有效的參數資料壓縮和處理；而本專題是利用基頻的特性先將一段語音訊號的基頻取出，在利用特殊演算法將重複多餘的基頻去除，做為辨識的重要特徵參數。**

三. 製作理論 ：

本專題是以基週軌跡（Pitch Contour）的變化及分佈做為特徵參數，在硬體設計上，先以低通濾波器（Low Pass Fiter）找出 Pitch 變化值，再利用 HT48R50 歸納其分佈，整理出一組特徵參數做為辨識時之依據。利用能量測量的方式決定擷取語音參數的開始及結束；在濾波整形器之前用硬體電路調整準位電壓，使電壓大於 Level Hold (電壓保持電路)的下降時間，再利用訊號在一定時間所通過的值來作能量測量。因為無語音時所產生的參數遠小於有語音階段時所產生的參數，所以可以用來作語音起使端點和結束端點之偵測（End Point Detection）。

在語音訊號經過低通濾波器之後，所對應的語音頻率從 20Hz~ 800Hz 其中裡面包含了語音基頻當中的二倍頻和三倍頻甚至到四倍頻所以我們使用硬體電路 Level Hold (電壓保持電路)，利用 RC 的充放電原理配合適當的時間常數值，以取出我們所需要的語音基頻參數。

四. 語音處理演算法：

語音輸入經過濾波整型器處理之後，去除了電壓的特性，只留下了頻率特性，而濾波過後只剩下聲音基頻及二次頻，大約在 20Hz 至 800Hz。而我們再利用微控制器－HT48R50 對該語音輸入作頻率分析，得到該語音的頻譜，再經過處理後以 16 個 Bytes 代表一組特徵參數並加以儲存，以作為辨識時之依據。

而在語音特徵參數之處理方面，我們考慮了兩種不同的作法：

(1) 線性頻率分割（Linear Frequency Division）：

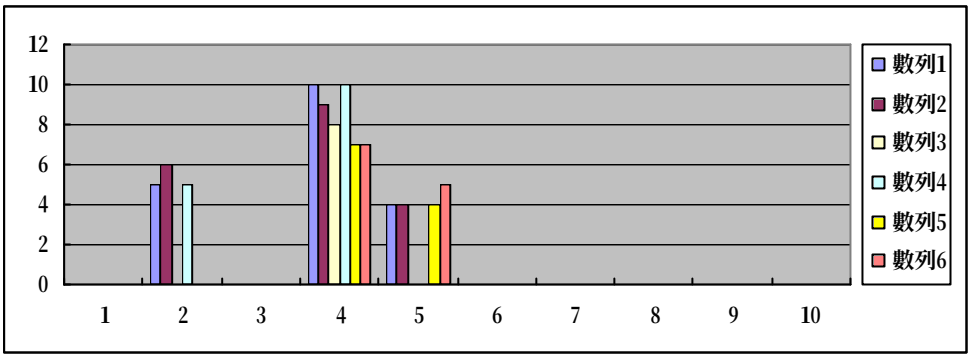
將語音資料以線性分布的方式，平均分佈於 16 個頻率範圍中，

例如： 50hz 100hz 150hz 200hz 250hz 300hz ….
時槽 1 時槽 2 時槽 3 時槽 4 時槽 5 時槽 6 …

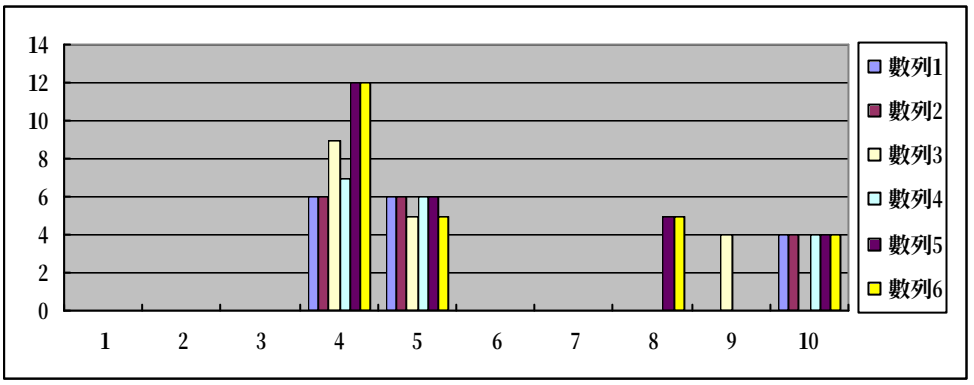
以這樣資料分佈優點是可以很平均的將語音各個頻率成份記錄下來；缺點是資料分佈會變的很廣泛，對於能量集中分佈於低頻率範圍的語音信號，辨識率較不理想。

語音記錄樣本範例：

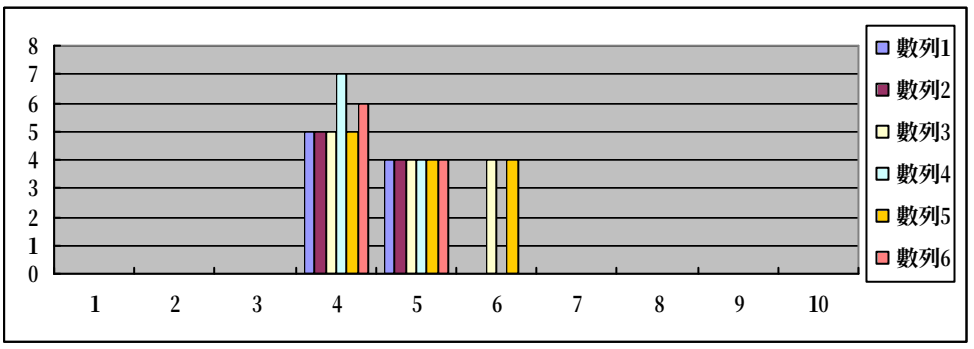
語音內容：『左轉』



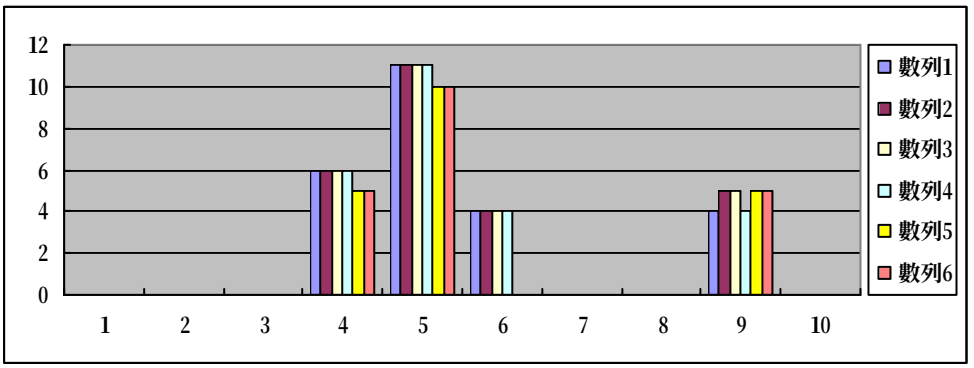
語音內容：『右轉』



語音內容：『冰箱』



語音內容：『電視』



- **非線性頻率分割（**Non-Linear Frequency Division**）：**

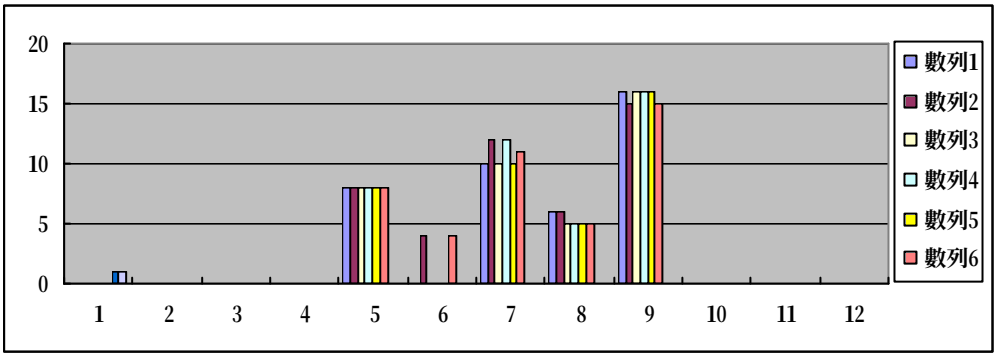
  將語音資料以非線性方式，分佈於 16 **個頻率範圍中，**

  例如: 50hz 150hz 225hz 300hz 450hz 600hz ….
  時槽 1 **時槽** 2 **時槽** 3 **時槽** 4 **時槽** 5 **時槽** 6 …

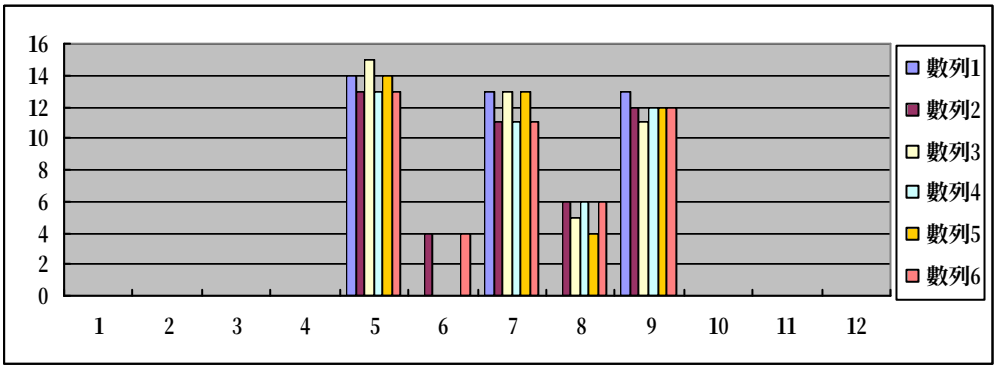
  非線性頻率分割的優點是資料處理容易。由於是針對低頻範圍予以較細之分割方式，因此對於能量集中分佈於低頻率範圍的語音信號，辨識率較佳。缺點是資料不能回覆！

語音記錄樣本範例：

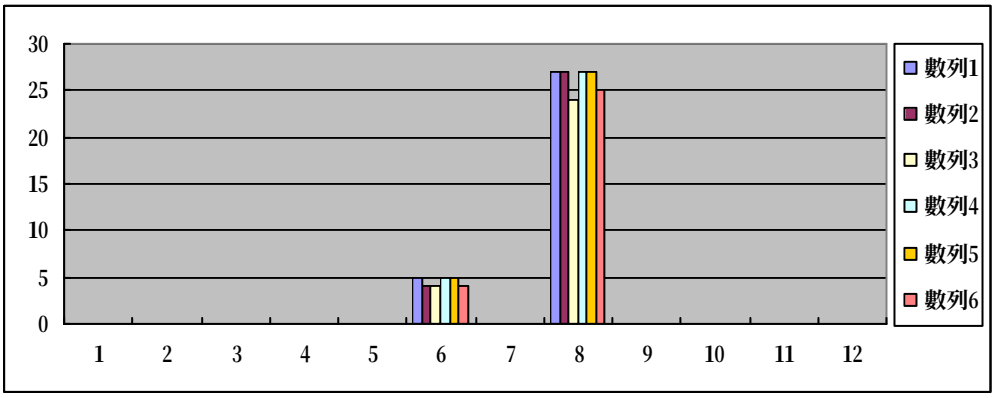
語音內容：『左轉』



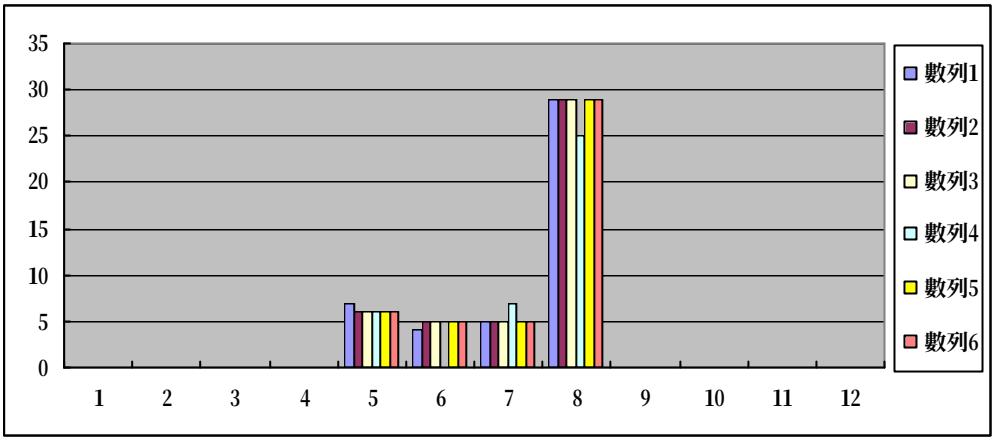
語音內容：『右轉』



語音內容：『冰箱』



語音內容：『電視』



五. 語音參數辨識演算法：

在訓練階段時，輸入之語音指令（Voice Command）經過先前的處理後，即存放到 E^2PROM 內成為參考樣本（Reference Template），因為是 E^2PROM 的結構，所以即使斷電也無須再重複訓練。而辨識過程，則是由 HT48R50 負責將欲辨識之語音輸入參數與樣本進行比對，主要利用語音頻譜特性及分佈作為辨識之依據！本專題以『Eculidean Distance』作為比對時之距離計算，並搭配模糊比對來辨識出為哪一個指令，或是錯誤之輸入。

本辨識系統使用了三個階段的模糊特徵比對：

- 第一階段是根據頻譜分佈位置作比對：在單位時間內輸入不同的語音信號會有不同的頻譜分佈！我們就是用這方式作為第一階段辨識之依據！將所輸入的語音信號轉換成頻譜後，利用模糊比對法，將差異太大的語音輸入信號予以排除（Rejection），以提高辨識率。

- 第二階段是對頻譜面積作積分比對：將語音信號轉換成頻譜後的數值相加，成為一個能量（Energy）特徵參數！不同語音指令（Voice Command）的能量會有不同的變化，所產生的面積也有所不同，所以我們用這個特徵來當作第二階段的辨識依據。

- 第三階段是對主要特徵數值比對：就是針對頻譜中數值最高與次高的值作比對，因為這種比對法對於整個指令的特徵需要有一定的相似度才能辨識成功，所以將此種比對法則放到第三階段，將能更準確的比對出我們所要辨識的指令。

時間的正規化（Time Normalization）：

由於每次的語音輸入長度都會不一樣，為了獲得較高之辨識率，一般必須經過繁雜的正規化程序，如動態時間校準（Dynamic Time Warping；DTW），然而此法之演算過程繁複，不易在記憶體

及運算速度受限的小型微控制器上實現！因此本專題是是直接擷取頻率特性參數，最後都是得到一組（16 Bytes）語音頻譜，不論使用者輸入的指令長短，只要速度差別不大皆可辨識得出來。

六. 功能與特點 ：

本專題—『低價位語音辨識』設計時之走向，是朝向低價位、操作簡單、且語音辨識率高…等重點發展。使用一顆微控制器－HT48R50 來完成所有的語音辨識動作，不僅低成本，且辨識率可高達 80%，可辨識 16 個指令，對於一般的家用控制或是玩具上的應用，已經是綽綽有餘的了。

功能及工作原理：

這套語音辨識系統有一個麥克風作為聲音的輸入以及四個功能按鍵分別是：

功能鍵一 『語音節錄』：

是將使用者欲節錄的語音命令由麥克風輸入，利用頻率擷取的方式將使用者的語音特徵擷取出來，再利用特有的壓縮方式將有效的資料儲存起來，當做這個語音指令的特徵參數，由於我們使用特有的抓取特徵方式將語音頻譜重覆的有效值抓入，再利用演算法將特徵作比較、比對，把不必要的值排除後，將此特徵參數存入 E^2PROM。

功能鍵二 『語音辨識』：

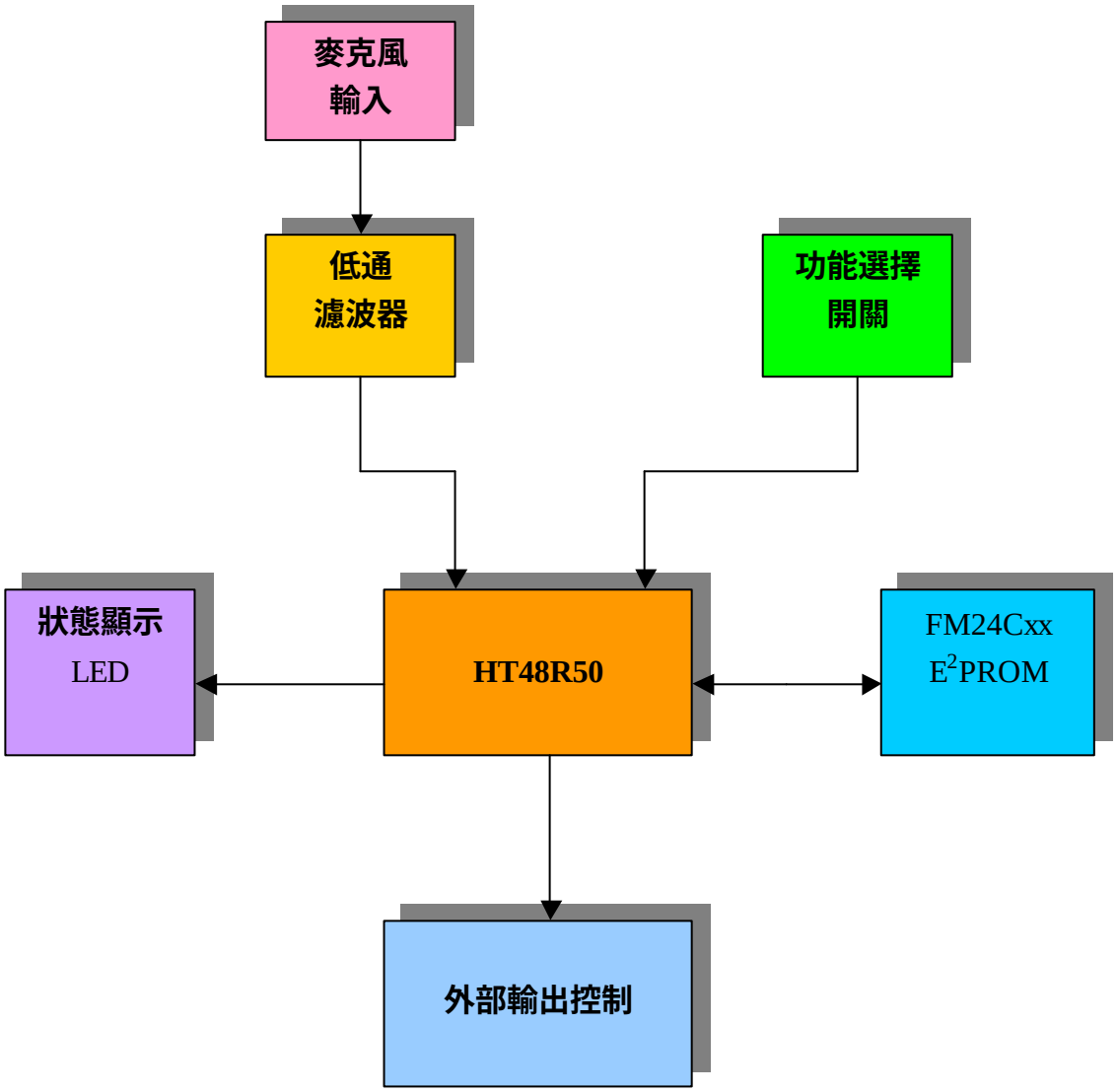
當語音辨識鍵按下時，HT48R50 會將麥克風所輸入的語音語音訊號轉換成特徵參數後和已經存在 E^2PROM 的指令特徵值作比較，以有效範圍的模糊比對方式將指令辨識出來，並以 2 進制的方式顯示在狀態 LED 上，完成語音辨識的動作，等待使用者選擇下一個模式。

功能鍵三 『清除紀錄』：

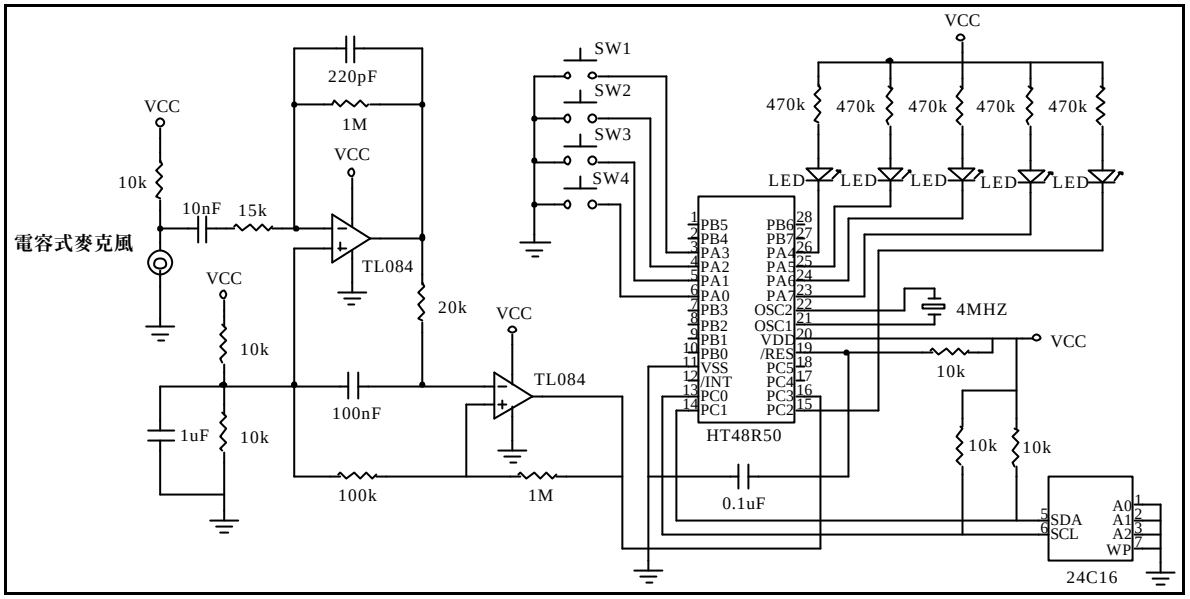
讓使用者清除 E^2PROM 內所存放的的語音指令特性參數，讓下一次辨識時忽略該項紀錄。

功能鍵四 『選擇紀錄』：

選擇欲存入語音指令或清除的記錄。



圖(一) 語音辨識之電路方塊圖



圖(二) 語音辨識之電路圖

七. 電路原理 ：

本專題硬體電路大致上分為幾個部分：

語音取樣電路（Sampling**）：**

首先是麥克風將聲音信號轉成電壓信號，接著輸入到低通濾波器（Low Pass Filter**），將高頻信號及雜訊濾除，再輸入到樞密特觸發電路，最後輸入到** HT48R50**。**

記憶體（Memory**）：**

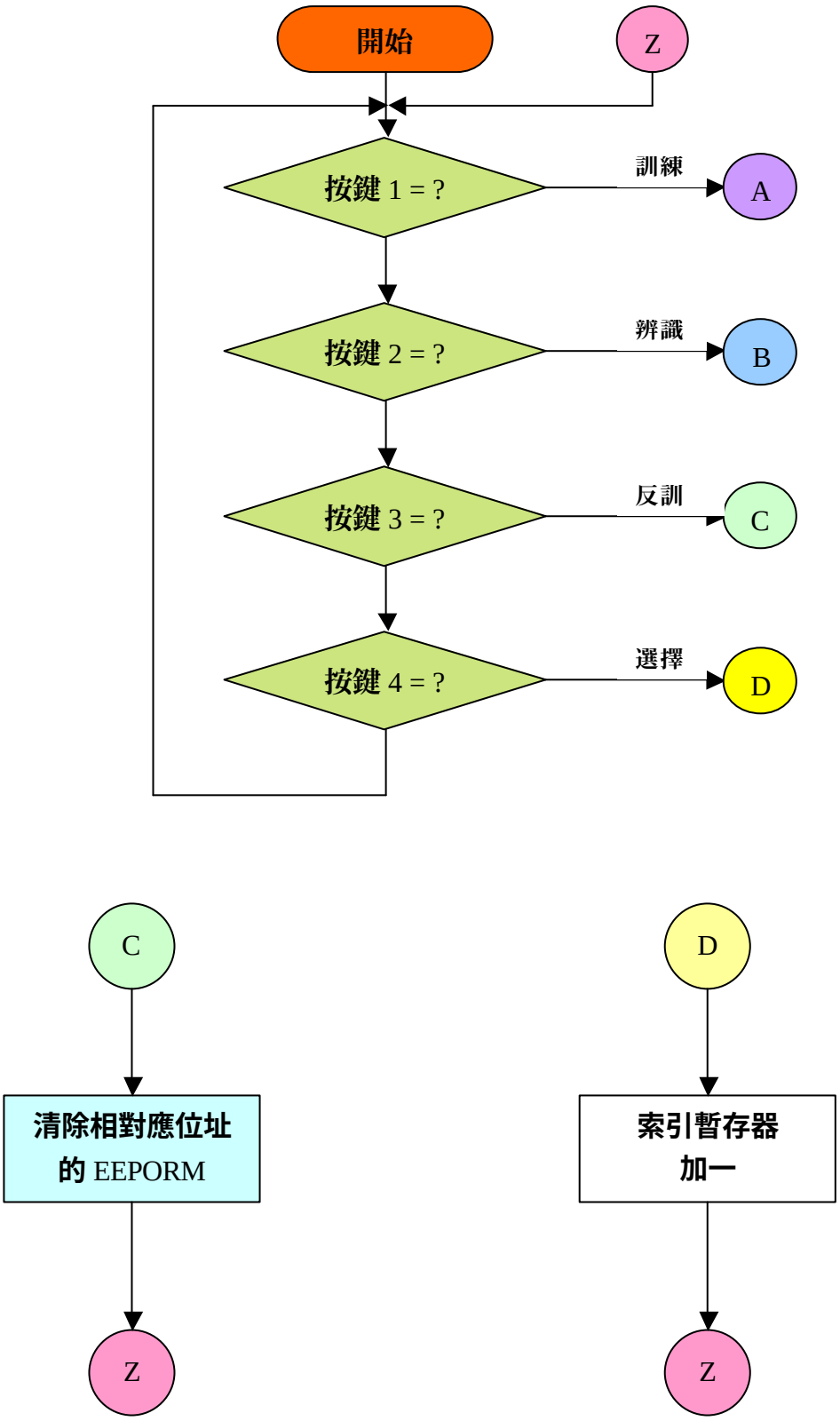
本專題是使用 I^2C **串列** E^2PROM **－**24CXX**，所以只需使用兩隻** I/O **腳，即可作** E^2PROM **的存取動作；由於是** E^2PROM **的結構，所以即使斷電也無須再重複訓練。**

輸入按鍵（Input**）：**

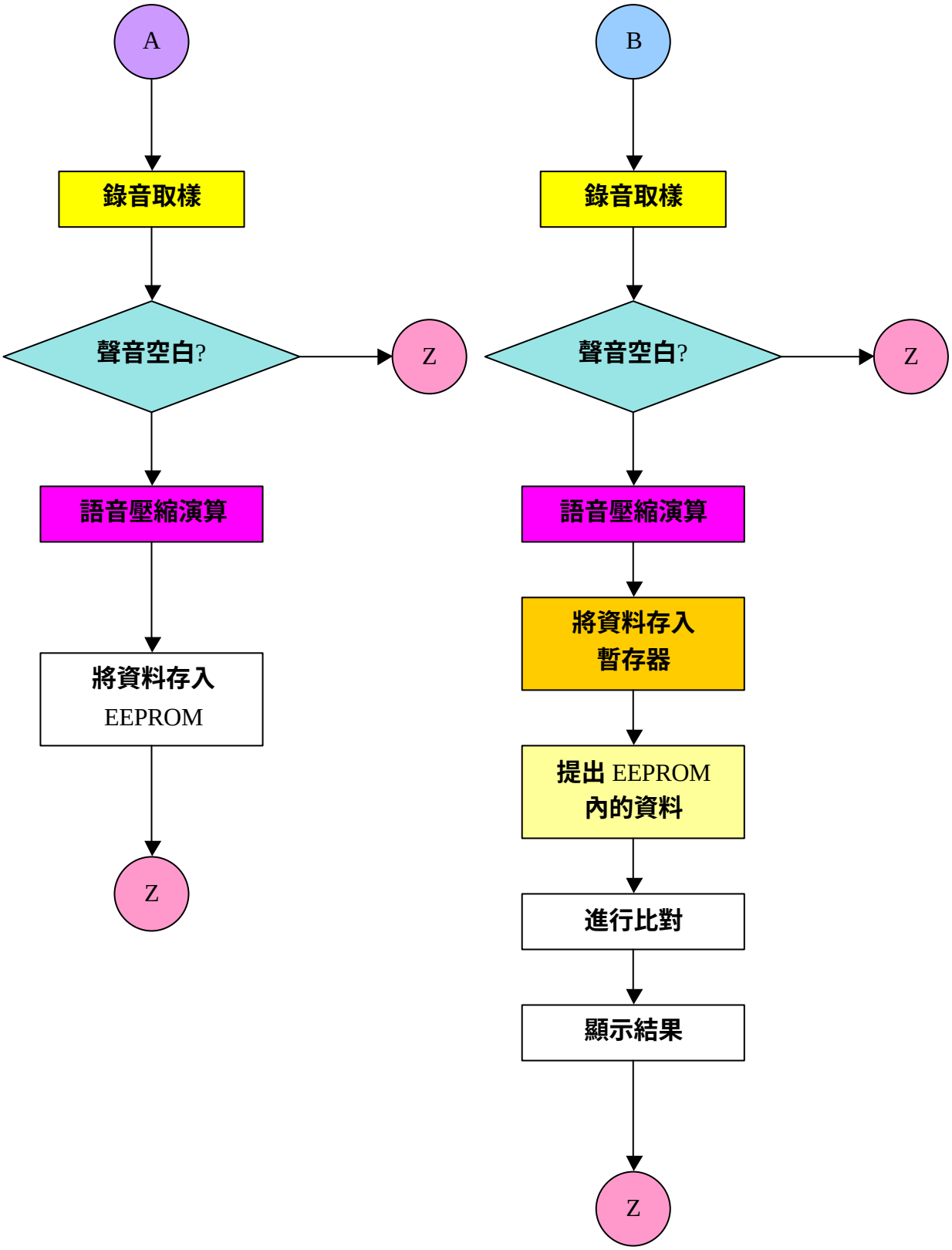
本專題使用四個按鍵來作語音存入、刪除、辨識和選擇的動作，各按鍵之功能請參考前一節之說明。

輸出狀態（Output**）顯示** LED**：**

使用五個 LED **顯示出目前使用的狀態。**


開始
Z
按鍵 1 = ?
訓練
A
按鍵 2 = ?
辨識
B
按鍵 3 = ?
反訓
C
按鍵 4 = ?
選擇
D
C
清除相對應位址
的 EEPORM
Z
D
索引暫存器
加一
Z

圖(三)辨識之程式流程圖


A
錄音取樣
聲音空白?
Z
語音壓縮演算
將資料存入
EEPROM
Z
B
錄音取樣
聲音空白?
Z
語音壓縮演算
將資料存入
暫存器
提出 EEPROM
內的資料
進行比對
顯示結果
Z

八. 未來發展：

- 如需要辨識更多的指令，可以藉由加大記憶體量來擴充。但是隨著辨識指令的增加，辨識率將隨之下降。
- 增加自動啟動辨識功能，完全無須使用按鍵了。
- 如想應用在較高階的地方的話可以加強人機介面的裝置，如增加 LCD 模組……等。
- 使用喉震式麥克風來減少外部的雜音，進而提高辨識成功率。

九. 製作過程與心得 ：

當我們在製作語音辨識時發現，原來語音真的? 一門相當大的學問，在這方面都是我們從未曾碰過的挑戰，雖然在製作的過程中有很多關於語音方面的瓶頸要去克服，而又大多是我們沒學過的領域，所幸有我們的老師以及學長在背後辛勤的指導，替我們解答有關語音的相關問題，也發現我們在製作過程中更了解聲音的領域，很感謝學長和老師在語音方面指導，許多我們不懂的問題，也很感謝「盛群半導體公司」與「e-科技雜誌」，讓我們有這個機會學習和挑戰很多我們不懂的領域！