

## 第十五章 AC97 控制單元

Intel 公司在 1997 年制定出一個聲音編解碼器(Audio Codec)的規格稱為 Audio Codec '97 (AC'97)。簡單來說，編解碼器(Codec)是一個單晶片裝置，將數位訊號透過串列傳輸作解碼的動作轉成類比訊號，傳送到耳機或喇叭輸出；或者將從麥克風進來的類比訊號作編碼的動作，轉成數位訊號，作串列傳輸。現今，AC97 的規格成為一個低價位、高品質的聲音解決方案，即目前的整合音訊標準。AC97 發展至今已經歷經幾次較大的修改，從最初 AC97 v1.x 白皮書中規定固定的 48kHz 取樣率輸出，到後來的 AC97 v2.0 白皮書中擴展了部分音頻特徵，開始支援多種取樣率以及多聲道輸出。因此，要了解整合音訊標準，就從 AC97 控制器著手吧！

### 15.1 總覽

PXA250和PXA210應用程式處理器的AC97控制單元（ACUNIT）支援了AC97 2.0規格校訂版的特性，將列在13.2節的「特性目錄」中。AC97控制單元也支援了音效控制連結（AC-link），音效控制連結是個傳送數位音效、數據機、麥克風輸入、編碼解碼控制器和狀態資訊的序列界面。

AC97編碼解碼器傳送數位化音效取樣讓ACUNIT儲存在記憶體中。對於重放或是合成音效產生，處理器會找出儲存的音效取樣並且將他們經由AC-link傳送給編碼解碼器，接著編碼解碼器裡的外部數位類比轉換器將音效取樣轉換成類比音效波形。

此章節是在描述ACUNIT的程式設計模型。本章節的內容需要對於AC97 2.0規格校訂版有相當的瞭解。

**注意：**ACUNIT和I2S控制器不能在同時間使用。

### 15.2 特性目錄

應用程式處理器ACUNIT支援了AC97以下的特性：

- 立體脈衝編碼調變（PCM）輸入、立體PCM輸出、數據機輸入輸出和單音麥克風輸入都有獨立通道
- 以上所有通道在硬體上只支援16位元的取樣，少於16位元的取樣是透過軟體來支援
- 多取樣率AC97 2.0編碼解碼器（4萬8千赫茲或更低），ACUNIT依靠編

碼解碼器來控制不同的取樣率的變化

- 讀寫/存取到AC97暫存器
- 支援輔助編碼解碼器
- 二個接收FIFO (32位元, 16個通道)
- 三個傳送FIFO (32位元, 16個通道)

應用程式處理器ACUNIT並不支援這些額外的AC97特性：

- 雙倍傳輸率取樣 (n+1取樣PCML, R & C)
- 18和20位元取樣長度

### 15.3 信號描述

從AC-link來的AC97信號是一個點對點同步的序列連結，支援全雙工資料傳輸。所有數位音效流、數據線編碼解碼流和指令/狀態資訊均經由AC-link來溝通。AC-link使用通用輸出輸入 (GPIOs)，軟體必須重新組態GPIOs來把他們當成AC-link使用，AC-link的接腳描述列在表15-41。

表15-1 外部介面到編碼解碼器

名稱	方向	說明
nACRESET	O	低電位觸發編碼解碼器重置, 當nACRESET觸發時編碼解碼器的暫存器重置
GP28/BITCLK	I	12.288百萬赫茲位元率時脈
GP31/SYNC	O	4萬8千赫茲訊框指示器和同步裝置
GP30/SDATA_OUT	O	序列音效輸出資料到編碼解碼器來進行數位類比轉換
GP29/SDATA_IN_0	I	序列音效從主要編碼解碼器輸入資料
GP32/SDATA_IN_1	I	序列音效從輔助編碼解碼器輸入資料

#### 15.3.1 信號規劃步驟

1. 配置SYNC和SDATA\_OUT當作輸出
2. 配置BITCLK、SDATA\_IN\_0和SDATA\_IN\_1當作輸入
3. nACRESET是一個專用的輸出，它在供電時仍然是觸發，完成這些步驟來反

觸發nACRESET：

- a、如同先前的描述來組態其他AC97信號。
- b、整體控制暫存器（GCR）中，設定GCR[COLD\_RST]位元，參考表15-48「整體控制暫存器」在13-20頁有更多的介紹。

**注意：**參考4.1.3節「GPIO暫存器定義」，在4-6頁有設計GPDR和GAFR的程式細節來使用ACUNIT。

### 15.3.2 AC-link範例

圖13-6是一個AC-link的互連範例，ACUNIT支援一種或二種編碼解碼器在AC-link上，如果只有一個主要的編碼解碼器連接著，SDATA\_IN\_1是不需要的。

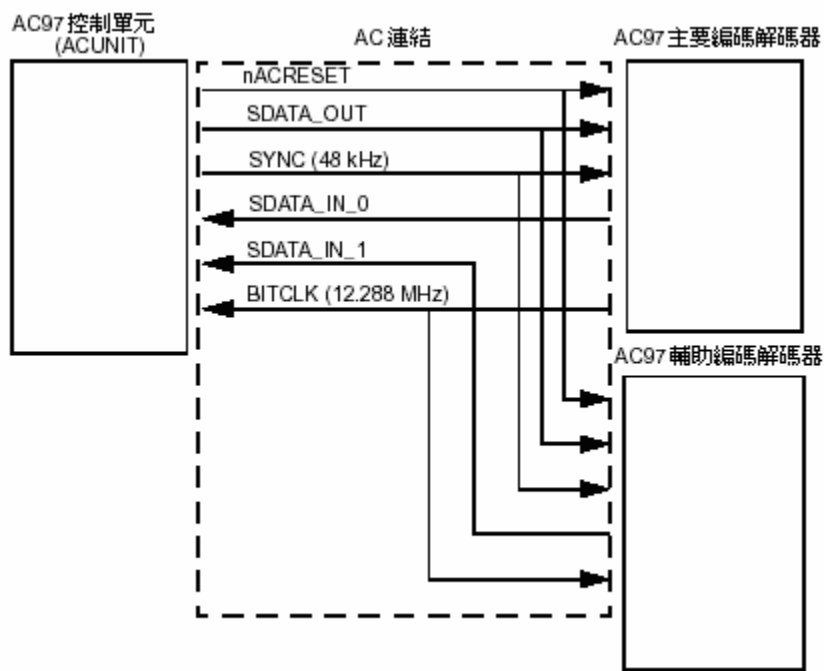


圖15-1 資料透過AC-link傳輸

## 15.4 AC-link數位序列介面協定

每個AC97編碼解碼器包含5個接腳的數位序列介面來連結到ACUNIT，AC-link是一個全雙工、固定脈衝、PCM（脈衝編碼調變）數位流。它使用分時

多工 (TDM) 設計來處理控制暫存器存取和多輸入輸出音效流。AC-link的架構把每個音效訊框分成12個傳送和12個接收的資料流，每個資料流有20個位元的取樣解析度而且需要最少16位元解析度的數位類比轉換器 (DAC) 和類比數位轉換器 (ADC)，ACUNIT所支援的資料流在表15-42。

表15-2 支援資料流格式 (1 of 2)

通道	存儲格	說明
脈衝編碼調變 (PCM) 重放	二個輸出存儲格	二個通道合成PCM輸出流
脈衝編碼調變 (PCM) 記錄資料	二個輸入存儲格	二個通道合成PCM輸入流
編碼解碼器控制	二個輸出存儲格	控制暫存器寫入埠
編碼解碼器狀態	二個輸入存儲格	控制暫存器讀取埠
數據線編碼解碼器輸出	一個輸出存儲格	數據線編碼解碼器數位類比轉換輸出流
數據線編碼解碼器輸入	一個輸入存儲格	數據線編碼解碼器類比數位轉換輸入流

表15-2 支援資料流格式 (2 of 2)

通道	存儲格	說明
專屬麥克風輸入	一個輸入存儲格	專屬麥克風輸入流支援立體音響裝置AEC和其他聲音應用程式
I/O控制	一個輸出存儲格	一個在數據編碼解碼器上GPOs專屬的插槽
I/O狀態	一個輸入存儲格	一個在數據編碼解碼器裡GPIs專屬的插槽，資料從每個訊框回傳

ACUNIT提供所有在AC-link上的資料傳輸同步，資料的傳輸是將256位元的資訊分成13組存儲格組成稱之為一個訊框，存儲格0是16位元長的標籤欄位，其他12個存儲格是資料欄位，標籤欄位包含有一個位元來判別是否為有效的訊框和12個位元來判別資料欄位裡的存儲格是否含有有效資料，每個資料欄位裡的存儲格是20位元長。

一個訊框開始是由SYNC變成高電位時，SYNC是高電位時的總計時間正是標籤欄位，AC97的訊框發生在固定4萬8千赫茲間和12.288百萬赫茲位元率時脈BITCLK同步。

控制器和編碼解碼器何時送出傳送的資料和取樣接收的資料是使用SYNC和BITCLK來決定，傳送端在每個BITCLK高緣時轉換序列資料流而接收端在BITCLK低緣時取樣序列資料流，傳送者必須標記序列資料流中有效的存儲格，有效的存儲格標記在存儲格0。

在AC-link上的序列資料是由最高有效位元到最低有效位元的順序排列，標籤欄位的第一個位元是位元15，而資料欄位中每個存儲格的第一個位元是位元19，任何存儲格的最後一個位元是位元0。

圖15-2為控制器和編碼解碼器的標籤和資料欄位結構，此圖也列出ACUNIT支援的存儲格定義。

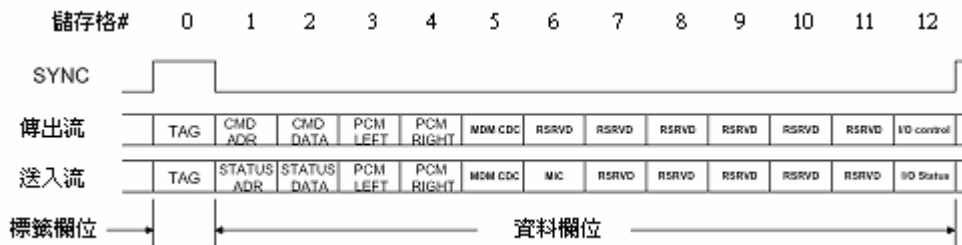


圖15-2 AC97標準雙向音效訊框

### 15.4.1 AC-link音效輸出訊框 (SDATA\_OUT)

音效輸出訊框資料流相當於多工束由數位輸出資料組成到AC97數位至類比轉換器輸入和控制暫存器，每個音效輸出訊框支援十二個20-位元的輸出資料時間存儲格，ACUNIT所產生的取樣不會大於16位元，所以最後四個有效位元補上零。

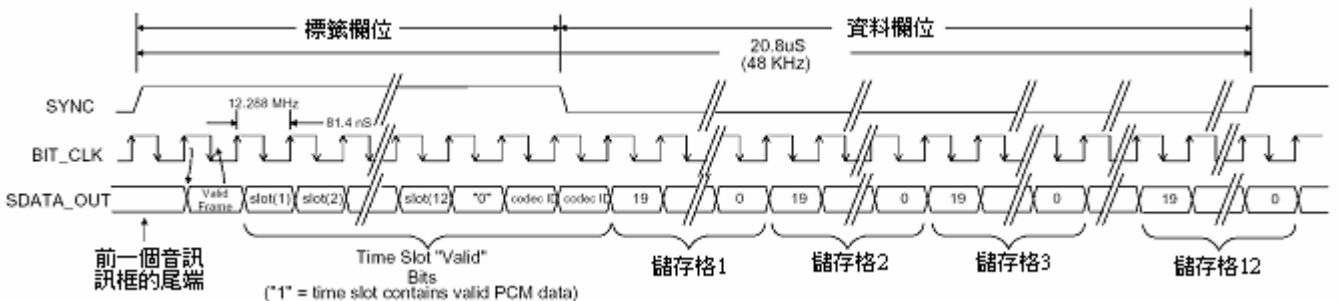


圖15-3 AC-link音效輸出訊框

新的音效輸出訊框開始是由SYNC從低電位轉換到高電位和BITCLK的上升

的邊緣同步時，BITCLK脈衝立即落下而且AC97取樣SYNC的觸發，BITCLK的落下邊緣讓AC-link端知道一個新的音效訊框開始。在BITCLK的下一個上升的邊緣，ACUNIT傳送到SDATA\_OUT存儲格0的第一位元位置（有效訊框位元），每個新的位元位置是由AC-link在BITCLK上升的邊緣然後經由AC97在BITCLK落下邊緣時取樣，這樣的順序保證資料傳送和後來取樣點的送入跟傳出的資料流是時間對準的。

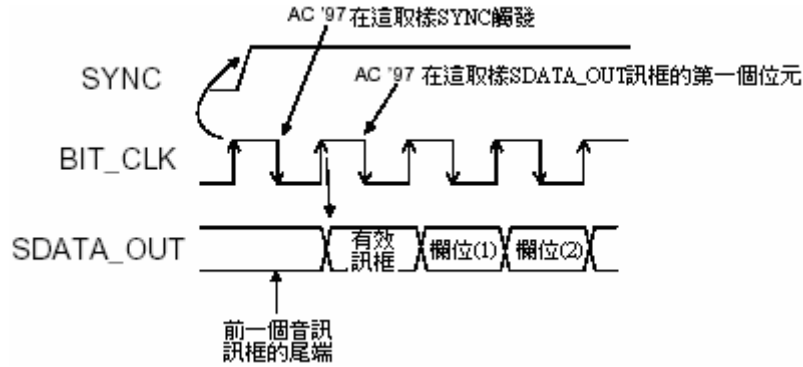


圖15-4 音效輸出訊框的起始

SDATA\_OUT合成流是對齊最高有效位元（最高有效位元優先），ACUNIT填寫零到所有非有效存儲格位元位置，假如少於20個有效位元存在指定的有效存儲格，ACUNIT裝填零到20位元存儲格的帶尾非有效位元位置。

例如：假如16-位元取樣串流被播放到AC97數位類比轉換器，前16位元位置要跟數位類比轉換器的最高有效位元對齊，接下來的四位元位置ACUNIT填寫零，這樣的過程保證最低有效位元不會引起任何DC的偏差，不管數位類比轉換器DAC是使用何種的解析度（16、18或20-位元）。

注意：當ACUNIT傳送單音音效取樣流，軟體必須保證左邊和右邊取樣流時間間隔是填用相同資料。

### 15.4.1.1 存儲格0：標籤欄位

存儲格0中，第一個位元是整體的位元（SDATA\_OUT存儲格0，位元15）標誌檢驗全體音效訊框的正確，假如有效訊框位元是1，則目前的音效音訊框包含至少一個有效資料的存儲格，接下來12個位元位置經由AC97取樣指示出12個存儲格中哪個符合包含有效資料。存儲格0的位元0和1是用來當作讓I/O讀寫到編碼解碼暫存器的編碼解碼器的ID位元，編碼解碼暫存器的編碼解碼器的ID位元在下一節會描述，這個方法，可使不同取樣率的資料流可以在AC-link的固定4萬8千赫音效訊框率交互傳送，編碼解碼器能操控控制器使用SLOTREQ位元的輸出

取樣率，關於SLOTREQ位元的輸出取樣率稍後會介紹（在輸入訊框描述），這下面圖表說明時間存儲格為基礎的AC-link協定。

編碼解碼器就緒（Codec Ready）經由編碼解碼器存儲格0位元15的資料輸出流傳送，在正常操作的期間是不會改變的，AC97 2.0規格校訂版要求編碼解碼器就緒狀態只能因為由控制器轉變發出powerdown（PR）狀態而改變。控制器硬體自己並不監視傳送或收到資料的編碼解碼器就緒，控制器將主要編碼解碼器的編碼解碼器就緒儲存在GSR[PCR]中和輔助編碼解碼器的編碼解碼器就緒儲存在GSR[SCR]中，為了軟體觸發一個DMA或一個程式I/O操作，控制器只會取樣一次有效的編碼解碼器就緒然後忽略它隨後的訊框，編碼解碼器就緒在PR狀態轉變後只會重新取樣。

#### 15.4.1.2 存儲格1：指令地址埠

指令埠控制AC97函數特性和監視狀態，包含有混頻器設定和電源管理（涉及AC97 2.0規格校訂有更多詳細資料），但是不只是限制於這些功能。

控制介面結構支援到六十四個16-位元的讀/寫暫存器，可定址在偶位元組範圍，只有存取到偶暫存器（0x00、0x02等等）是有效的，存取到奇暫存器（0x01、0x03等等）並不是有效的。

音效輸出訊框存儲格1傳達控制暫存器位址和寫/讀指令資訊到ACUNIT。

二個編碼解碼器連接到單一的SDATA\_OUT，要個別的定址主要編碼解碼器和輔助編碼解碼器，隨著下列這些步驟：

存取主要編碼解碼器：

1. 設置有效訊框位元（存儲格0，位元15）
2. 設置存儲格1和2的有效位元（存儲格0，位元14和13）
3. 寫入0b00到編碼解碼器ID欄位（存儲格0，位元1和0）
4. 設定存取的讀寫方向（存儲格1，位元19）。
5. 設定編碼解碼暫存器索引（存儲格1，位元18-12）
6. 假如存取是寫入，寫資料到指令資料埠（存儲格2，位元19-4）

存取輔助編碼解碼器：

1. 設置有效訊框位元（存儲格0，位元15）
2. 清除存儲格1和2的有效位元（存儲格0，位元14和13）
3. 寫入一非零值（0b01，0b10，0b11）到編碼解碼器ID欄位（存儲格0，位元1

- 和0)
4. 設定存取的讀寫方向（存儲格1，位元19）
  5. 設定編碼解碼暫存器索引（存儲格1，位元18-12）
  6. 假如存取是寫入，寫資料到指令資料埠（存儲格2，位元19-4）

**表15-3 存儲格1位元定義**

位元	名稱	說明
位元(19)	RW	1 = 讀取，0 = 寫入
位元(18:12)	IDX	編碼暫存器索引
位元(11:0)	保留	全填寫0

只有一個I/O週期隨時可以透過AC-link決定，ACUNIT 使用寫入和讀取顯示出在連結上I/O的存取，例如，讀出一個編碼解碼暫存器在連結存取前立即回送，為了獲得真實資料，軟體必須監視CAR[CAIP] 位元，軟體必須確認在存取嘗試去確保它是第一次存取，在這之前這個位元沒有被設置，一組CAR[CAIP] 位元指示編碼解碼器存取是待決，在 CAR[CAIP] 位元被清除後，下一個編碼解碼器存取（讀出或寫入）就能通過。

後存取的例外是讀出到編碼解碼器GPIO接腳狀態暫存器（位址0x54），編碼解碼器GPIO 接腳狀態讀出後立即從最後的12個存儲格接收資料傳回，GPIO 接腳狀態暫存器的編碼解碼器必須不斷地發送存儲格12中暫存器的狀態。

為了讀取到編碼解碼器，控制器給編碼解碼器最大四個訊框來回應，之後假如沒收到回應，控制器回傳一個虛擬的讀取完成給中央處理單元（0xFFFF\_FFFF），並且也設置讀取完成狀態位元GSR[RDCS]。

### 15.4.1.3 存儲格2：指令資料埠

命令資料埠傳遞16-位元控制暫存器寫入資料當目前命令埠操作是在一個寫入的週期（由存儲格1，位元19指示出）。

**表15-4 存儲格2位元定義**

位元	名稱	說明
位元(19:4)	控制暫存器寫入資料	如果目前的操作是讀取就填寫0
位元(3:0)	保留	全填寫0



如果目前指令埠操作是讀取ACUNIT填寫零到所有的存儲格時間。

#### 15.4.1.4 存儲格3：PCM重放左聲道

音效輸出訊框存儲格3是合成數位音效左移重放流，假如取樣流傳送的解析度少於20位元，ACUNIT會在存儲格中填寫零到全部帶尾非有效位元位置。

#### 15.4.1.5 存儲格4：PCM重放右聲道

音效輸出訊框存儲格4是合成數位音效右移重放流，ACUNIT會在存儲格中填寫零到全部帶尾非有效位元位置。

#### 15.4.1.6 存儲格5：數據線編碼解碼器

假如有支援線路編碼解碼器，音效輸出訊框存儲格5包含最高有效位元對齊數據機DAC輸入的資料，其他的數據機DAC輸入解析度可以用16、18或20位元執行，ACUNIT的驅動器在啟動時間決定DAC的解析度。在正常運轉時間操作的期間，ACUNIT會在存儲格中填寫零到全部帶尾非有效位元位置輸入零，數據編碼解碼器可能會分隔次要線路上的編碼解碼器或整合音效編碼解碼器。

#### 15.4.1.7 存儲格6-11：保留

這些音效輸出訊框存儲格保留給以後使用而且ACUNIT將他們填寫為零。

#### 15.4.1.8 存儲格12：I/O控制

存儲格12有16個最高有效位元給GPIO 控制(輸出)和狀態(輸入)來使用，這些位元常用於減少改變特殊狀況所造成的存取等待時間，這些位元的值是寫入編碼解碼器狀態暫存器的數據編碼解碼器I/O空間中位址0x54的值，下列規則管理存儲格12的使用：

1. 存儲格12最初的預設是標記為無效的。
2. 寫入編碼解碼器I/O中位址0x54空間時會在下一個訊框傳送存儲格12的資料並且標記存儲格12為有效的，資料也同樣送出存儲格1和2。
3. 在第一次寫入到位址0x54後，隨後全部訊框的存儲格12仍然為有效的，在存儲格12的資料傳送是最後寫入到位址0x54的資料，隨後任何寫入到暫存器會在下一個訊框傳送新的資料輸出。
4. 隨著系統重設或AC97冷開機，存儲格12會是無效的，存儲格12會仍然無效

直到下一次寫入到位址0x54。

### 15.4.2 AC-link 音效輸入訊框 (SDATA\_IN)

ACUNIT有二條SDATA\_IN線路，主要的和次要的，每條線路都有附加的編碼解碼器，附加的編碼解碼器決定哪個存儲格是有效或無效的，二個輸入的資料存儲格是完全正交的（換言之，沒有二個資料存儲格在兩條線路的相同位置是有效的）。

多重輸入資料流在存儲格範圍接收和多路傳輸是被每個流的存儲格有效位元指定，每個AC-link 音效輸入訊框是由十二個20-位元時間存儲格所組成的，存儲格0是保留的而且包含16位元來用於AC-link協定架構。

軟體必須輪詢音效輸入訊框的第一個位元 (SDATA\_IN 存儲格0，位元15) 來得到指示，確認在安排ACUNIT進入操作之前控制器是在編碼解碼器就緒的狀態，當控制器取樣編碼解碼器就緒，接下來12個位元位置取樣指出12個時間存儲格中哪個被分配到輸入資料串流和是否他們包含有效資料，圖13-10，「AC97輸入訊框」說明以時間存儲格為基礎的AC-link協定。

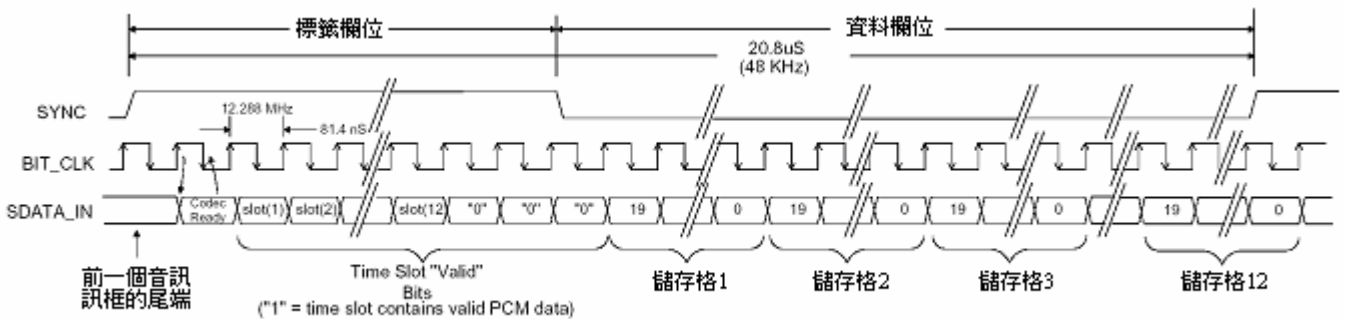


圖 15-5 AC97輸入訊框

當SYNC從低電位轉換到高電位時，新的音效輸入訊框開始，低電位到高電位的轉換和BITCLK 的上升的邊緣是同步的，在BITCLK 的下一個落下邊緣，AC97 取樣SYNC的觸發，落下的邊緣讓AC-link的邊知道一個新的音效訊框開始，在下次BITCLK 上升時，控制器傳送SDATA\_IN到存儲格0的第一個位元位置（編碼解碼器就緒位元），每個新的位元位置是由AC-link在BITCLK 上升的邊緣然後經由ACUNIT在BITCLK 落下邊緣時取樣，這樣的順序保證資料傳送和後來取樣點的送入跟傳出的資料流是時間對準的。

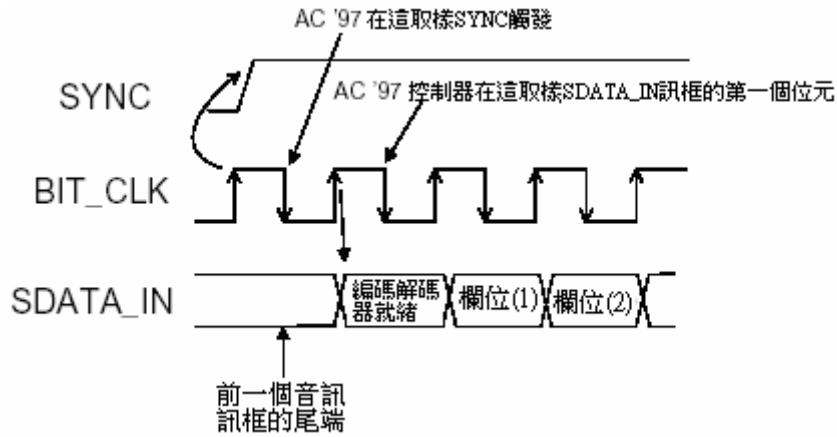


圖15-6 音效輸入訊框的起始

SDATA\_IN 合成流是對齊最高有效位元（最高有效位元優先）而且AC97編碼解碼器填寫零到非有效位元位置，SDATA\_IN 的資料是在BITCLK 的落下邊取樣。

#### 15.4.2.1 存儲格 0：標籤欄位

存儲格0中，第一個位元是整體的位元（SDATA\_IN 存儲格0，位元15）指示ACUNIT是否在編碼解碼器就緒狀態，假如編碼解碼器就緒位元是0，ACUNIT並未準備就緒來正常的運作，在電源反觸發重新開機之後這樣的情況是正常的而且ACUNIT電壓參考是固定的，當AC-link編碼解碼器就緒指示位元為1，AC-link和AC97控制還有狀態暫存器是完全工作的，ACUNIT 必須探查編碼解碼器電源關閉控制/狀態暫存器來決定哪個小部份是就緒的。

#### 15.4.2.2 存儲格1：狀態位址埠/ SLOTRQ位元

狀態埠監視ACUNIT函數的狀態包含了混頻器設定和電源管理，但不是只限制於此功能。

假如控制器在存儲格0期間標示存儲格1和2為有效的，音效輸入訊框1的流回應控制暫存器索引來讓資料在存儲格2被回傳。

控制器只接受狀態資料假如伴隨的狀態位址和在最近的讀取指令期間最後有效命令位址所發出的相符合。

為了多重的取樣率輸出，編碼解碼器在每個音效輸出訊框開始時審查它的取樣率控制暫存器、FIFO狀態，和送入的SDATA\_OUT標籤位元來決定哪個

SLOTREQ位元設定為工作（低電位），在目前音效輸入訊框期間SLOTREQ 位元觸發指示哪個輸出存儲格在下一個音效輸出訊框需要從控制器來的資料。對於固定在4萬8千赫工作，SLOTREQ位元設定為工作（低電位），而且在每個訊框傳送取樣。

為了多重的取樣率輸入，每個輸入存儲格的標籤位元指示是否呈現出有效資料。

存儲格1中，音效輸入訊框的狀態位址埠傳送編碼解碼控制暫存器讀取位址和全部輸出存儲格多重取樣率存儲格要求標籤檢驗，AC97定義十個最低有效位元為輸出存儲格3-12的請求資料要求標籤檢驗，對於二通道音效訊號，編碼解碼器-只有資料-要求標誌檢驗符合存儲格3和4是有意義的。

**表15-5 輸入存儲格1位元定義**

位元	說明
19	保留（填寫零）
18-12	控制暫存器索引（如果AC97標籤它無效則填寫零）
11	存儲格 3 要求: PCM 左通道
10	存儲格 4 要求: PCM 右通道
9	存儲格 5 要求: 數據線 1
8	存儲格 6 要求: NA
7	存儲格 7 要求: NA
6	存儲格 8 要求: NA
5	存儲格 9 要求: NA
4	存儲格 10 要求: NA
3	存儲格 11 要求: NA
2	存儲格 12 要求: NA
0, 1	保留（填寫零）

音效輸入訊框存儲格1標籤位元附屬於狀態位址埠資料，SLOTREQ 位元一直是有效的存儲格不管標籤位元為何。

**注意：**存儲格要求對於存儲格3和4是時常一起設置或一起清除（兩者設置或兩者清除）。

### 15.4.2.3 存儲格 2：狀態資料埠

狀態資料埠傳送16位元控制暫存器讀取資料

**表15-6 輸入存儲格2位元定義**

位元	名稱	說明
位元(19:4)	控制暫存器讀取資料	假如AC97標記它為無效就填寫零
位元 (3:0)	保留	填寫零

**注意：**假如存儲格2標籤為無效，ACUNIT填寫零到所有的存儲格。

#### 15.4.2.4 存儲格 3：PCM 記錄左聲道

音效輸入訊框存儲格3是ACUNIT編碼解碼器左聲道輸出。

ACUNIT傳送它的ADC輸出資料（最高有效位元優先）並填寫零到任何帶尾非有效位元位置來填寫它的20位元時間存儲格。

#### 15.4.2.5 存儲格 4：PCM 記錄右聲道

音效輸入訊框存儲格4是ACUNIT編碼解碼器右聲道輸出。

ACUNIT傳送它的ADC輸出資料（最高有效位元優先）並填寫零到任何帶尾非有效位元位置來填寫它的20位元時間存儲格。

#### 15.4.2.6 存儲格5：選用的數據線編碼解碼器

音效輸入訊框存儲格5包含最高有效位元對齊數據機ADC輸出資料（假如線路編碼解碼器支援）。

應用程式處理器對於選用的數據機支援了16位元的ADC輸出解析度。

#### 15.4.2.7 存儲格 6：選用的專用麥克風記錄資料

音效輸入訊框存儲格6是一個為麥克風專用，選用的第三個PCM系統輸入通道，這個輸入通道增補了準確的立體輸出可以讓喇叭擴音器應用更精確地消除回音的演算法程式。

ACUNIT只支援16位元解析度的麥克風輸入通道。

### 15.4.2.8 存儲格 7-11：保留

音效輸入訊框存儲格7 -11是保留給以後使用而且ACUINT會忽略他們。

### 15.4.2.9 存儲格12：I/O狀態

GPIOs 配置如同輸入返回他們的狀態接通這間隔每一個框，資料送回到最新的訊框是軟體可以經由編碼解碼暫存器在數據編碼解碼器I/O空間的位址0x54上存取的，只有16個最高有效位元使用來回傳GPIO的狀態，最低有效位元的位元0指示GPI 輸入中斷事件，參考AC97 2.0規格校訂版有更多資訊。

從編碼解碼器位址0x54讀取是不透過連結傳送的，存儲格12中的資料接收是儲存在控制器內部而且存儲格12最近的資料是回到位址0x54讀取。

## 15.5 AC-link低電源模式

在進入應用程式處理器的低電源模式之前，軟體必須設置GCR [ ACLINK \_OFF] 位元，這樣保證ACUNIT不會驅動AC-link上的輸出腳位。

### 15.5.1 關閉AC-link

當AC97編碼解碼器電源關閉暫存器（0x26）位元PR4是是設置為1（寫入0x1000），AC-link信號進入低電源模式，然後主要編碼解碼器驅動BITCLK 和SDATA\_IN到邏輯低電位階層，這樣的順序在圖13-12有時序的圖示。

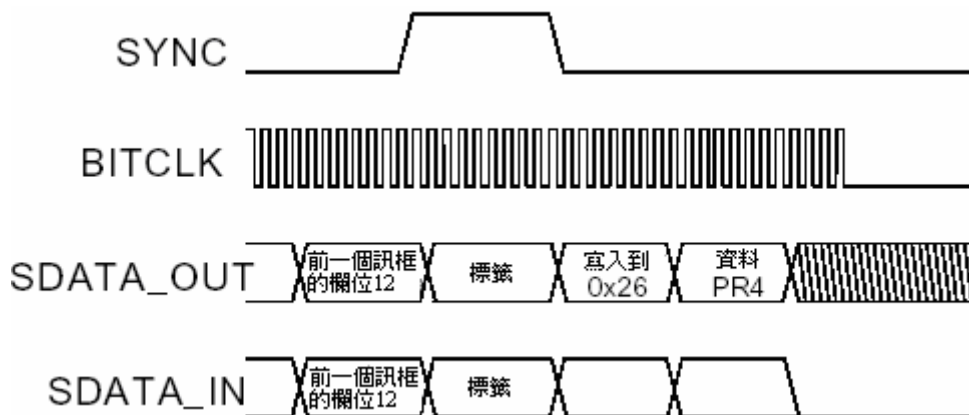


圖15-7 AC-link 電源關閉時序

**注意：**BITCLK 不用放大表示

ACUNIT在AC-link上傳送寫入給電源關閉暫存器（0x26），設定ACUNIT當它寫入到暫存器位元PR4（資料0x1000）時它不傳送資料到存儲格3-12，當它接收到關閉電源的要求，AC97 2.0規格校訂版不需要編碼解碼器來處理其他的資料，當編碼解碼器處理要求時，它立即轉換BITCLK和SDATA\_IN為邏輯低電位階層。

ACUNIT在規劃GCR[ACLINK\_OFF]為1之後驅動SYNC和SDATA\_OUT到邏輯低電位階層，當GCR[ACLINK\_OFF]=1時，ACUNIT維持nACRESET為高電位。

## 15.5.2 喚醒AC-link

### 15.5.2.1 由編碼解碼器喚醒觸發器

為了喚醒AC-link，編碼解碼器驅動它的SDATA\_IN為邏輯高電位階層，上升的邊緣觸發重新啟動中斷假如編碼解碼器重新啟動位元設置為1，中央處理單元接著使用冷或暖重置順序來喚醒編碼解碼器，ACUNIT 使用暖重置來喚醒主要的編碼解碼器，當SYNC驅動1 s的最小值為高電位和BITCLK無作用時，編碼解碼器檢測到暖重置，編碼解碼器必須等到它能啟動BITCLK之前取樣SYNC低電位，編碼解碼器發出喚醒事件信號必須保持SDATA\_IN高電位直到它檢測到暖重置已經完成，編碼解碼器能轉換它的SDATA\_IN成低電位。

圖13 -13為編碼解碼器喚醒觸發的AC-link時序，因為應用程式處理器可能需要喚醒，所以電源管理單位檢測AC97喚醒事件（SDATA\_IN高電位多於1 $\mu$ s）。當ACUNIT就緒，它經由觸發暖或冷重置來回應喚醒事件（參考圖13-13），數據編碼解碼器可能需要處理容量來喚醒AC-link回報事件諸如呼叫者ID和由鈴聲喚醒。

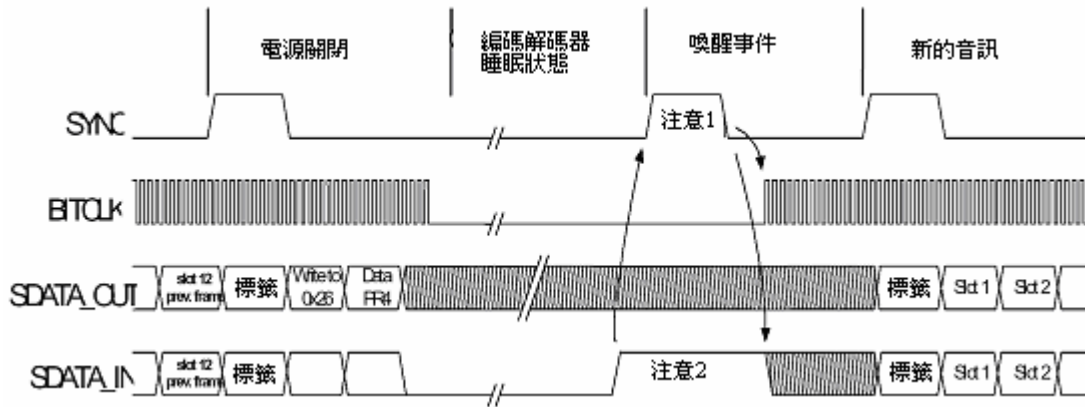


圖15-8 SDATA\_IN喚醒信號

注意：

1. 在SDATA\_IN 升高後，SYNC必須保留最少1 $\mu$ Sec.
2. SDATA\_IN喚醒脈衝頻寬的最小值是1 $\mu$ Sec.
3. BITCLK not to scale

### 15.5.2.2 由ACUNIT喚醒觸發器

AC-link協定提供冷AC97重置和暖AC97重置，目前電源關閉狀態最後會指定哪個重置是使用的，暫存器必須在所有電源關閉模式期間停留在相同的狀態除非冷AC97重置被執行，在冷AC97重置時，AC97暫存器被初始設定到他們的預設值。

在電源關閉後，AC-link必須在訊框於電源關閉發生之後且經由再次觸發SYNC信號而恢復活動之前等待最少四個音效訊框的時間，當AC-link電源啟動時，它經由編碼解碼器就緒位元（輸入存儲隔0，位元15）來指示準備就緒。

#### 15.5.2.2.1 冷AC97重置.

冷重置當nACRESET腳位透過GCR[COLD\_RST]觸發時產生，觸發和反觸發nACRESET會啟動BITCLK（假如提供的編碼解碼器有支援）和SDATA\_OUT，所有AC97 控制暫存器被初始設定到他們預設的電源重置開機值，nACRESET是一個非同步的AC97輸入。

#### 15.5.2.2.2 暖AC97重置

暖AC97重置恢復啟動AC-link沒有變更目前AC97暫存器的值，暖重置當BITCLK無作用和SYNC驅動1 $\mu$ s的最小值為高電位時產生。



在正常的音效訊框中，SYNC是一個同步的AC97 輸入。當BITCLK無作用時，SYNC看作是一個非同步輸入來產生AC97的暖重置。

ACUNIT一定不能啟動BITCLK直到它再次取樣到SYNC為低電位，這可避免檢測錯誤的新音效訊框。

當ACUNIT從編碼解碼器接收到喚醒，它會發出一個中斷（假如這個中斷回復是啟動的），軟體必須隨後發出暖或冷重置到編碼解碼器經由設定GCR中適當的位元。

## 15.6 ACUNIT操作

ACUNIT可以透過處理器或DMA控制器來進行存取，處理器使用程式I/O指令來存取ACUNIT而且能存取四種暫存器類型：

- ACUNIT暫存器:可存取在32位元邊界，他們列在13.8.3節，「暫存器」在13 -18頁。
- 編碼解碼暫存器:音效或數據編碼解碼器能包含多於六十四個16-位元暫存器，編碼解碼器使用16位元位址邊界的暫存器，ACUNIT 提供存取到編碼解碼暫存器經由映射到它的32位元位址領域邊界，13.8.3.18節，「存取編碼解碼暫存器」在13 -32頁描述從32位元映射到16位元邊界，寫入或讀取作業到這些暫存器是經由AC-link傳送。
- 數據編碼解碼GPIO暫存器:假如ACUNIT連接到數據編碼解碼器，編碼解碼GPIO暫存器也能存取，編碼解碼GPIO暫存器使用編碼解碼器領域中的存取位址0x0054，GPIO 寫入作業經由AC-link但讀取並不是，當訊框從編碼解碼器接收時，暫存器內容在控制器領域中是不斷地更新到暫存器，當處理器試著去讀取編碼解碼GPIO暫存器時，非正式的暫存器反而被取代來讀取。
- ACUNIT FIFO資料:ACUNIT對於音效輸出和數據機輸出有二個FIFO傳送，還有音效輸入、數據機輸入和麥克風輸入有三個FIFO接收，FIFO傳送法寫入是藉由寫入到PCM資料暫存器（PCDR）或數據機資料暫存器（MODR）二者之一，接收FIFO通道是透過PCDR，MODR，或麥克風輸入資料暫存器（MCDR）來讀取。

**注意：**在它啟動後，ACUNIT要求DMA馬上填充所傳送的FIFO。

**注意：**ACUNIT 暫存器不儲存DMA要求的狀態或每個FIFO裡資訊重寫的資

料取樣個數，結果是程式I/O不能使用來代替DMA要求資料傳送。

只有DMA可以存取FIFOs，且透過資料暫存器存取，DMA控制器使用8、16或32位元組區塊存取FIFO資料。DMA要求臨界是不可程式化的，當傳送的FIFO少於32位元組時，ACUNIT傳送DMA要求。當接收的FIFO為32位元組或更多時，ACUNIT接收DMA要求。

DMA控制器回應下列ACUNIT DMA要求：

- 當PCM傳送和接收FIFOs達到一半時，PCM FIFO會傳送和接收DMA的要求。
- 當數據機傳送和接收FIFOs達到一半時，數據機FIFO會傳送和接收DMA的要求。
- 當麥克風輸入接收FIFOs達到一半時，麥克風輸入會接收到DMA的要求。

### 15.6.1 初始化

在開啟電源時編碼解碼器和ACUNIT電路系統重置。開啟電源後，nACRESET信號仍然為觸發直到音效或數據機驅動器設置GCR [COLD\_RST]位元。在作業期間，清除GCR[COLD\_RST]位元會重置ACUNIT和編碼解碼器，隨著下列這些步驟來初始設定ACUNIT：

1. 設計GPIO方向暫存器和GPIO交換功能選擇暫存器來指定ACUNIT埠的適當的腳位方向，參考13.3節，「信號描述」有詳細的資料。
2. 設置GCR [COLD\_ RST]位元來觸發nACRESET，直到設置完成，所有其他的暫存器仍然在重置狀態，反觸發nACRESET 有下面的影響：
  - a、 訊框填寫零來傳送因為傳送的FIFO仍舊是空的，這樣的情形不會造成錯誤的狀態。
  - b、 ACUNIT記錄零直到編碼解碼器傳送有效的資料。
  - c、 DMA要求是啟動的。
3. 啟動主要就緒中斷啟動（Primary Ready Interrupt Enable）和/或次要就緒中斷啟動經由分別設定GCR[PRIRDY\_ IEN]和/或GCR[SECRDY\_ IEN]位元。
4. 軟體應主要和次要的就緒中斷而啟動DMA作業。
5. ACUNIT觸發器傳送DMA 要求，DMA等待回應來填充傳送的FIFO。
6. ACUNIT 繼續傳送零直到傳送的FIFO達到一半時。當它達到一半時，有效的FIFO資料經由AC-link傳送。

**注意：**當nACRESET反觸發時，讀取到編碼解碼混頻暫存器回傳硬體的類型是屬於編碼解碼器，假如編碼解碼器無作用或AC不支援的話，ACUNIT就不設置編碼解碼器就緒位元，主要編碼解碼器的GCR[PCRDY]或輔助編碼解碼器的GCR[CRDY]。

## 15.6.2 帶尾位元組

傳送和接收的FIFOs帶尾位元組處理如下：

假如傳送緩衝區沒有32位元組解析度，傳送的FIFO中帶尾位元組就不會被傳送，傳送緩衝區必定填補零，假如他小於32位元組。

在編碼解碼器傳送有效的資料後，ACUNIT記錄零直到nACRESET被反觸發，假如編碼解碼器傳送資料的總計緩衝區小於32位元組，零就會被記錄，當FIFO達到一半時，接收的DMA會發出要求。

## 15.6.3 存取編碼解碼暫存器的作業流程

軟體存取編碼解碼暫存器經由轉換7位元編碼解碼器位址到32位元應用程式處理器實體位址，關於位址轉換的詳細資料，參考13.8.3.18節，「存取編碼解碼暫存器」。

軟體必須讀取編碼解碼器存取暫存器（CAR）來鎖定AC-link，假如CAR[CAIP]位元是0，AC-link是不受控制的，關於CAR的詳細資料，參考表15-54，「編碼解碼器存取暫存器」。

讀出存取到CAR設置CAR[CAIP]位元，ACUNIT清除CAR[CAIP]位元當編碼解碼器寫入或編碼解碼器讀取動作完成，軟體也能經由寫入0來清除CAR[CAIP]位元。

在它鎖定AC-link後，軟體能使用適當的應用程式處理器實體位址寫入或讀取編碼解碼暫存器。

ACUNIT在編碼解碼器寫入動作完成後設置GSR[CDONE]位元，詳細的資料，參考表15-9，「整體狀態暫存器」，軟體藉由寫入1來清除這個位元。

讀取編碼解碼器，軟體必須完成下列步驟：

1. 體發出一個虛擬讀取到編碼解碼暫存器，ACUNIT回應這個讀取作業為無效的資料，ACUNIT然後則透過AC-link初始化讀出存取。
2. 編碼解碼器讀取作業完成，ACUNIT設置GSR[SDONE]為1，詳細的資料，參考表15-49，「整體狀態暫存器」，軟體經由寫入1來清除這個位元。
3. 體重複讀取動作如同步驟1，ACUNIT現在由編碼解碼器回傳資料，第二個讀取動作也是透過AC-link初始化讀出存取。
4. 如編碼解碼器無法在四個SYNC訊框時間內做出回應，則ACUNIT讀取動作逾時，在此情況下，第二個讀取動作就會回傳資料逾時的值0x0000\_FFFF。

## 15.7 時脈和取樣頻率

系統預設，ACUNIT在取樣頻率為4萬8千赫時傳送和接收資料，然而，它能在頻率小於4萬8千赫茲時取樣資料，假如請求的編碼解碼器支援存儲格要求，在此時編碼解碼器執行一個特定的演算法和通知控制器不要在特定的訊框中傳送有效的資料。例如，假如控制器傳送480個訊框，而且編碼解碼器指示控制器不要傳送那些480個訊框中的39個有效資料，編碼解碼器會以4萬4千1百赫茲作有效的資料取樣，當編碼解碼器傳送資料（控制器接收模式），它能使用相同的演算法來傳送有效以及空的相混合的訊框。

所有資料經由AC-link傳送是同步SYNC的上升的邊緣，ACUNIT把BITCLK分為256個來產生SYNC信號，這個計算產生4萬8千赫茲SYNC信號和它的期間定義為一個訊框，資料於每一個BITCLK上升的邊緣在AC-link上傳送而且接著在每個隨後的BITCLK落下邊緣取樣AC-link的接收邊，時序圖表在圖13-8「AC-link音效輸出訊框」。

ACUNIT在二種不同時脈域上使資料同步：BITCLK和一個內部系統時脈，內部系統時脈時常是運行模式頻率的一半，運行模式頻率等於或大於八倍的BITCLK頻率。

## 15.8 功能描述

功能描述這一節適用於所有通道。

## 15.8.1 FIFO

ACUNIT有5個FIFOs：

- PCM使用十六個三十二位元通道來傳送FIFO。
- PCM使用十六個三十二位元通道來接收FIFO。
- 數據機使用十六個三十二位元通道來傳送FIFO（較高的16位元必須都為0）。
- 數據機使用十六個三十二位元通道來接收FIFO（較高的16位元總是為0）。
- 麥克風輸入使用十六個三十二位元通道來接收FIFO（較高的16位元總是為0）。

接收的FIFO會觸發DMA要求，當FIFO有八個或更多的通道，當它保持少於八個通道時，傳送的FIFO會觸發DMA要求，在任何資料經由AC-link傳送之前，傳送的FIFO必須要達到一半滿（填入八個通道）。

### 15.8.1.1 傳送FIFO錯誤

假如啟動的話，特定的通道狀態位元在傳送欠載狀況期間更新而且將觸發中斷，關於狀態位元的詳細資料可參考表15-52「PCM輸出狀態暫存器」和表15-61「數據機輸出狀態暫存器」。在傳送欠載狀況期間，最後的有效取樣是連續透過AC-link來傳送的，傳送執行下會發生下列的情形：

- 有效傳送的資料在記憶體中仍然是可用的但是DMA控制器極需要傳送的FIFO因為他在服務其他較高優先等級的週邊設備。
- DMA 控制器已經傳送所有有效資料從記憶體到傳送的FIFO，這促使最後的有效取樣透過AC-link來回應直到nACRESET被觸發和關閉ACUNIT。

### 15.8.1.2 接收FIFO錯誤

當啟動時，特定的通道狀態位元在接收滿溢狀況期間更新而且將觸發中斷，關於狀態位元的詳細資料可參考表15-53「PCM輸入狀態暫存器」、表15-57「麥克風輸入狀態暫存器」和表15-62「數據機輸入狀態暫存器」，在接收滿溢狀況期間，編碼解碼器傳送的資料並沒有記錄。

## 15.8.2 中斷

當中斷產生時，下列狀態位元會中斷處理器：

- 麥克風輸入FIFO錯誤—麥克風輸入接收FIFO的滿溢或欠載錯誤。
- 數據機輸入FIFO錯誤—數據機接收FIFO的滿溢或欠載錯誤。
- PCM輸入FIFO錯誤—音效接收FIFO的滿溢或欠載錯誤。
- 數據機輸出FIFO錯誤—數據機傳送FIFO的滿溢或欠載錯誤。
- PCM輸出FIFO錯誤—音效傳送FIFO的滿溢或欠載錯誤。
- 數據編碼解碼器GPIO狀態改變中斷—中斷CPU假如存儲格12的位元0設置了，這指出數據編碼解碼器的GPIO暫存器中的一個位元改變了。
- 主要編碼解碼器回復中斷—設置狀態暫存器位元當主要編碼解碼器從低電源模式回復，軟體寫入1到這個位元來清除它。
- 輔助編碼解碼器回復中斷—設置狀態暫存器位元當輔助編碼解碼器從低電源模式回復，軟體寫入1到這個位元來清除它。
- 編碼解碼器指令完成中斷—中斷CPU當編碼解碼暫存器的指令完成時，軟體寫入1到這個位元來清除它。
- 編碼解碼器狀態完成中斷—中斷CPU當編碼解碼暫存器的狀態位址和資料接收完成時，軟體寫入1到這個位元來清除它。
- 主要編碼解碼器就緒中斷—設置狀態暫存器位元當主要編碼解碼器準備就緒，編碼解碼器設置輸入訊框存儲格0的位元0來發信號通知它是就緒的，軟體清除GCR[PRIRDY\_IEN]位元來清除這個中斷。
- 輔助編碼解碼器就緒中斷—設置狀態暫存器位元當輔助編碼解碼器準備就緒，編碼解碼器設置輸入訊框存儲格0的位元0來發信號通知它是就緒的，軟體清除GCR[SECRDY\_IEN]位元來清除這個中斷。

## 15.8.3 暫存器

ACUNIT和編碼解碼暫存器映射位址範圍從0x4050\_0000到0x405F\_FFFF，所有ACUNIT暫存器都是可定址32位元的，雖然編碼解碼器有達過六十四個16位元的暫存器，可定址16位元，他們經由32位元位址映射和轉換到16位元來進行存取。

程式I/O和DMA突發傳輸能存取下列暫存器：

- 整體暫存器：ACUNIT有三種整體暫存器：狀態、控制和編碼解碼器一起共用音效和數據機域來存取暫存器。
- 特定通道音效ACUNIT暫存器：與PCM輸出、PCM輸入和麥克風輸入通道有關的特定通道音效ACUNIT暫存器。

- 特定通道數據機ACUNIT暫存器：與數據機輸出和數據機輸入通道有關的特定通道數據機ACUNIT暫存器。
- 音效編碼解碼暫存器
- 數據編碼解碼暫存器

特定的通道資料暫存器是為了給FIFO存取而且每個PCM、數據機和麥克風輸入FIFOs都有一個暫存器，寫入存取到其中一個暫存器來更新在相同的傳送FIFO中已寫入的資料，讀出存取到其中一個暫存器來清除相同的接收FIFO中的一個通道。

**注意：**暫存器表顯示架構和個別的位元定義，所有保留的位元讀取成未知的值而且必須填寫0，問號表示在重置時值是未知的。

**注意：**一些暫存器位元從編碼解碼器接收狀態，編碼解碼器狀態設置這個位元而軟體清除這個位元（寫入1來清除），狀態隨時能夠進來，即使當位元被設置或是在軟體清除的期間。假如軟體清除這個位元當編碼解碼器狀態更新這個位元時，編碼解碼器狀態事件會有較高的優先等級，這項中斷標誌位元會被這種情況影響。

### 15.8.3.1 暫存器映射摘要

所有ACUNIT暫存器都可以字位址定址（32位元寬）以0x00004的單位增量，編碼解碼器中的暫存器是半字元位址定址（16位元寬），以0x00002的單位增量。這些暫存器設置映射的位址範圍是從0x4050\_000到0x405F\_FFFF。

表15-7 暫存器映射摘要（1 of 2）

位址	名稱	說明
0x4050_0000	POCR	PCM輸出控制暫存器
0x4050_0004	PICR	PCM輸入控制暫存器
0x4050_0008	MCCR	麥克風輸入控制暫存器
0x4050_000C	GCR	整體的控制暫存器
0x4050_0010	POSR	PCM輸出狀態暫存器
0x4050_0014	PISR	PCM輸入狀態暫存器
0x4050_0018	MCSR	麥克風輸入狀態暫存器
0x4050_001C	GSR	整體的狀態暫存器
0x4050_0020	CAR	編碼解碼器存取暫存器
0x4050_0024 - 0x4050_003C	-	保留
0x4050_0040	PCDR	PCM FIFO 資料暫存器

0x4050_0044 - 0x4050_005C	-	保留
0x4050_0060	MCDR	麥克風輸入FIFO資料暫存器
0x4050_0064 - 0x4050_00FC	-	保留
0x4050_0100	MOCR	數據機輸出控制暫存器
0x4050_0104	-	保留
0x4050_0108	MICR	數據機輸入控制暫存器
0x4050_010C	-	保留
0x4050_0110	MOSR	數據機輸出狀態暫存器
0x4050_0114	-	保留

表15-7 暫存器映射摘要 (2 of 2)

位址	名稱	說明
0x4050_0118	MISR	數據機輸入狀態暫存器
0x4050_011C - 0x4050_013C	-	保留
0x4050_0140	MODR	數據機 FIFO 資料暫存器
0x4050_0144 - 0x4050_01FC	-	保留
(0x4050_0200 - 0x4050_02FC) 以0x00004的單位增加	-	主要音效編碼解碼器暫存器
(0x4050_0300 - 0x4050_03FC) 以0x00004的單位增加	-	輔助音效編碼解碼器暫存器
(0x4050_0400 - 0x4050_04FC) 以0x00004的單位增加	-	主要數據編碼解碼暫存器
(0x4050_0500 - 0x4050_05FC) 以0x00004的單位增加	-	輔助數據編碼解碼暫存器

15.8.3.2 整體控制暫存器

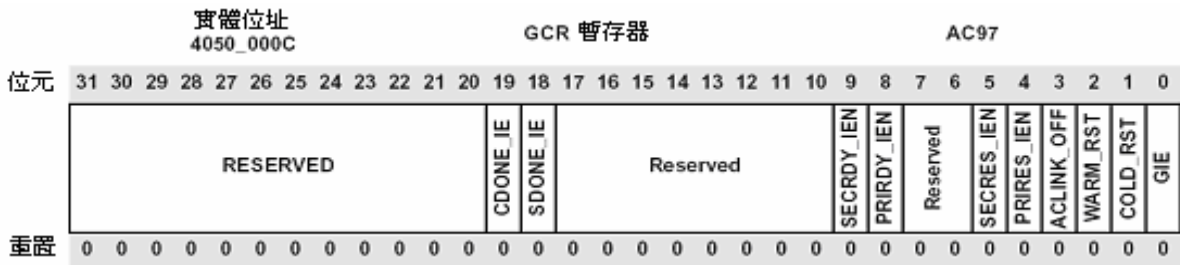
表15-8 整體控制暫存器 (1 of 2)

實體位址 4050_000C	GCR 暫存器																AC97																			
位元	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
	RESERVED												CDONE_IE	SDONE_IE	Reserved								SECRDY_IEN	PRIRDY_IEN	Reserved	SECRS_IEN	PRIRS_IEN	ACLINK_OFF	WARM_RST	COLD_RST	GIE					
重位元	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
位元	名稱		說明																																	
31:20	Reserved		保留																																	



19	CDONE_IE	指令完成中斷啟動 (CDONE_IE) : 0=在傳送指令位址和資料到編碼解碼器後，控制器沒有觸發中斷到CPU 1=在傳送指令位址和資料到編碼解碼器後，控制器觸發中斷到CPU
18	SDONE_IE	狀態完成中斷啟動 (SDONE_IE) : 0=中斷無效 1=從編碼解碼器接收到狀態位址和資料之後，啟動中斷發生
17:10	Reserved	保留
9	SECRDY_IEN	次要就緒中斷啟動 (SECRDY_IEN) : 0=中斷無效 1=當輔助編碼解碼器在SDATA_IN_1腳位上傳送編碼解碼器就緒位元時，啟動中斷發生
8	PRIRDY_IEN	主要就緒中斷啟動 (PRIRDY_IEN) : 0=中斷無效 1=當主要編碼解碼器在SDATA_IN_0腳位上傳送編碼解碼器就緒位元時，啟動中斷發生
7:6	Reserved	保留

表15-8 整體控制暫存器 (2 of 2)



位元	名稱	說明
5	SECRES_IEN	次要回復中斷啟動 : 0 = 中斷無效 1 = 當輔助編碼解碼器在AC-link上引起一個回復事件時，啟動中斷發生
4	PRIRES_IEN	主要回復中斷啟動 : 0 = 中斷無效 1 = 當主要編碼解碼器在AC-link上引起一個回復事件時，啟動中斷發生
3	ACLINK_OFF	AC-link關閉: 0 = 假如AC-link是關閉的，需將它打開，否則這個位元沒

		<p>有作用</p> <p>1 = 造成控制器去驅動SDATA_OUT和SYNC輸出低電位還有關閉輸入緩衝區啟動，重置輸出仍然保持高電位，AC-link不允許存取任何FIFOs</p> <p>設置這個位元並不保證完全關機，在設置這個位元之前軟體必須確保所有的作業是完成的</p>
2	WARM_RST	<p>AC97暖重置：</p> <p>0 = 暖重置不會產生</p> <p>1 = 在AC-link上造成暖重置發生，暖重置將會喚醒暫停的編碼解碼器，不用清除它的內部暫存器</p> <p>假如軟體嘗試去執行暖重置當BITCLK執行時，寫入將會被忽略而且這個位元不會改變。這個位元是自己清除的，換言之，他仍然是設置的直到重置完成而且在他自己清除之後BITCLK顯現在AC-link上</p>
1	COLD_RST	<p>AC'97 冷重置：</p> <p>0 = 造成所有AC97電路系統發生冷重置，在控制器和編碼解碼器中的所有資料都會消失</p> <p>1 = 冷重置不會產生</p> <p>預設為0而且重置之後，驅動器必須設置這個位元為1，這個位元的值在暫停之後仍保留，因此這個位元在暫停前被設置為1的話，冷重置在重新開始時就不會產生</p>
0	GIE	<p>編碼解碼器GPI中斷啟動（GIE）：</p> <p>這個位元控制是否改變任何數據編碼解碼器的狀態會造成中斷</p> <p>0 = 假如這個位元沒有設置，整體狀態暫存器的位元0有設置，但中斷還是不會產生</p> <p>1 = 假如這個位元設置GPI（由存儲格12的位元0指示）的值改變了而造成中斷和設置整體狀態暫存器的位元0</p>

15.8.3.3 整體狀態暫存器（GSR）

表15-9 整體狀態暫存器（1 of 2）

實體位址 4050_001C		GSR 暫存器												AC97																		
位元	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RESERVED												CDONE	SDONE	Reserved	RDCS	BIT3SLT12	BIT2SLT12	BIT1SLT12	SECS	PRRES	SCR	PCR	MINT	POINT	PIINT	Reserved	MOINT	MINT	GSCI		
重置	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位元	名稱	說明
31:20	Reserved	保留
19	CDONE	指令完成 (CDONE) : 0 = ACUNIT 沒有傳送指令位址和資料到編碼解碼器 1 = ACUNIT 有傳送指令位址和資料到編碼解碼器 這個位元經由軟體寫入'1'到這個位置來清除 (可中斷的)
18	SDONE	狀態完成 (SDONE) : 0 = ACUNIT 沒有從編碼解碼器接收到狀態位址和資料 1 = ACUNIT 有從編碼解碼器接收到狀態位址和資料 這個位元經由軟體寫入'1'到這個位置來清除 (可中斷的)
17:16	Reserved	保留
15	RDCS	讀取完成狀態: 這個位元指示編碼解碼器讀取完成的狀態 0 = 編碼解碼器正常讀取完成 1 = 編碼解碼器讀取產生逾時 這個位元仍然是設置直到由軟體清除, 這個位元經由軟體寫入'1'到這個位置來清除
14	BIT3SLT12	存儲格12的位元3 列出最近有效存儲格12的位元3
13	BIT2SLT12	存儲格12的位元2: 列出最近有效存儲格12的位元2
12	BIT1SLT12	存儲格12的位元1 列出最近有效存儲格12的位元1
11	SECRES	次要回復中斷: 0 = 回復事件不會在SDATA_IN_1上發生. 1 = 回復事件在SDATA_IN_1上發生 這個位元經由軟體寫入'1'到這個位置來清除 (可中斷的)
10	PRIRES	主要回復中斷: 0 = 回復事件不會在SDATA_IN_0上發生 1 = 回復事件在SDATA_IN_0上發生 這個位元經由軟體寫入'1'到這個位置來清除 (可中斷的)
9	SCR	輔助編碼解碼器就緒 (SCR) 反應SDATA_IN_1中編碼解碼器就緒位元的狀態 (可中斷的)
8	PCR	主要編碼解碼器就緒 (PCR) 反應SDATA_IN_0中編碼解碼器就緒位元的狀態 (可中斷的)
7	MINT	麥克風輸入中斷 (MINT)

		0 = 麥克風輸入通道沒有中斷發生 1 = 一個麥克風輸入通道中斷發生 當特殊的中斷被清除時，這個位元將會被清除（可中斷的）
--	--	--

表15-9 整體狀態暫存器 (2 of 2)

		實體位址 4050_001C	GSR 暫存器																AC97																		
位元		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
		RESERVED												CDONE	SDONE	Reserved	RDCS	BIT3SLT12	BIT2SLT12	BIT1SLT12	SECREC	PRIRES	SCR	PCR	MINT	POINT	PIINT	Reserved	MOINT	MIINT	GSCI						
重置		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
位元	名稱	說明																																			
6	POINT	PCM 輸出中斷 (POINT) 0 = PCM輸出通道沒有中斷發生 1 = 一個PCM輸出通道中斷發生 當特殊的中斷被清除時，這個位元將會被清除（可中斷的）																																			
5	PIINT	PCM 輸入中斷 (PIINT) 0 = PCM輸入通道沒有中斷發生 1 = 一個PCM輸入通道中斷發生 當特殊的中斷被清除時，這個位元將會被清除（可中斷的）																																			
4:3	Reserved	保留																																			
2	MOINT	數據機輸出中斷 (MOINT) 0 = 數據機輸出通道沒有中斷發生 1 = 一個數據機輸出通道中斷發生 當特殊的中斷被清除時，這個位元將會被清除（可中斷的）																																			
1	MIINT	數據機輸入中斷 (MIINT) 0 = 數據機輸入通道沒有中斷發生 1 = 一個數據機輸入通道中斷發生 當特殊的中斷被清除時，這個位元將會被清除（可中斷的）																																			
0	GSCI	編碼解碼器 GPI 狀態改變中斷 (GSCI) 0 = 清除存儲格12的位元0 1 = 設置存儲格12的位元0，這指示GPI的一個改變狀態和新的值在存儲格12中是可用的 這個位元經由軟體寫入‘1’到這個位置來清除（可中斷的）																																			

15.8.3.4 PCM輸出控制暫存器 (POCR)

表15-10 PCM輸出控制暫存器

實體位址 4050_0000		POCR 暫存器	AC97
位元	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	<div style="border: 1px solid black; padding: 5px;"> <div style="text-align: center; border-bottom: 1px solid black;">RESERVED</div> <div style="display: flex; justify-content: space-between; border-top: 1px solid black;"> <span style="border-right: 1px solid black; padding-right: 5px;">FEIE</span> <span style="border-right: 1px solid black; padding-right: 5px;">Reserved</span> </div> </div>	
重置	0 0		
位元	名稱	說明	
31:4	Reserved	保留	
3	FEIE	FIFO 錯誤中斷啟動 (FEIE) 這個位元控制傳送的FIFO錯誤發生是否會造成中斷 0 = 即使POSR的位元4被設置，還是沒有中斷發生 1 = 假如POSR的位元4被設置，中斷將會發生	
2:0	Reserved	保留	

15.8.3.5 PCM輸入控制暫存器 (PICR)

表15-11 PCM輸入控制暫存器 (PICR)

實體位址 4050_0004		PICR 暫存器	AC97
位元	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	<div style="border: 1px solid black; padding: 5px;"> <div style="text-align: center; border-bottom: 1px solid black;">RESERVED</div> <div style="display: flex; justify-content: space-between; border-top: 1px solid black;"> <span style="border-right: 1px solid black; padding-right: 5px;">FEIE</span> <span style="border-right: 1px solid black; padding-right: 5px;">Reserved</span> </div> </div>	
重置	0 0		
位元	名稱	說明	
31:4	Reserved	保留	
3	FEIE	FIFO 錯誤中斷啟動 (FEIE) 這個位元控制傳送的FIFO錯誤發生是否會造成中斷 0 = 即使PISR的位元4被設置，還是沒有中斷發生 1 = 假如PISR的位元4被設置，中斷將會發生	
2:0	Reserved	保留	

15.8.3.6 PCM輸出狀態暫存器 (POSR)

表15-12 PCM輸出狀態暫存器

實體位址 4050_0010		POSR 暫存器	AC97
位址	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0		
	RESERVED		FIFOE Reserved
重置	0 0		
位元	名稱	說明	
31:5	Reserved	保留	
4	FIFOE	FIFO 錯誤 (FIFOE) 0 = 沒有傳送的FIFO錯誤發生 1 = 傳送的FIFO錯誤發生，假如傳送的FIFO欠載發生，則這個位元被設置，在此情況下，最後有效取樣重複地傳送出去而且指標不會增加，會發生這樣是由於： a 沒有更多有效的緩衝區資料可用來傳送 b 緩衝區資料可用但是DMA控制器有過度頻寬的需求經由寫入1到這個位元位置來將位元清除	
3:0	Reserved	保留	

15.8.3.7 PCM輸入狀態暫存器 (PISR)

表15-13 PCM輸入狀態暫存器

實體位址 4050_0014		PISR 暫存器	AC97
位元	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0		
	RESERVED		FIFOE Reserved
重置	0 0		
位元	名稱	說明	
31:5	Reserved	保留	
4	FIFOE	FIFO 錯誤 (FIFOE) 0 = 沒有接收的FIFO錯誤發生 1 = 接收的FIFO錯誤發生，假如接收的FIFO滿溢發生則這個位元被設置，在此情況下，FIFO 指標不會增加，從AC-link進來的資料不會被寫入FIFO而且會消失，這是由於DMA控制器有過度頻寬的需求，因此無法及時清除接收的FIFO 經由寫入1到這個位元位置來將位元清除	
3:0	Reserved	保留	

15.8.3.8 編碼解碼器存取暫存器 (CAR)

表15-14 編碼解碼器存取暫存器

實體位址 4050_0020		CAR 暫存器	AC97
位元	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	RESERVED	
重置	0 0		CAIP
位元	名稱	說明	
31:1	Reserved	保留	
0	CAIP	依序編碼解碼器存取 (CAIP) 這個位元由軟體讀取來檢查編碼解碼器IO週期是否目前有在進行 0 = 沒有週期在進行而且讀取暫存器的動作設置這個位元為‘1’，這保留了右半給軟體驅動器執行IO週期，一旦週期完成，這個位元將自動清除，軟體能經由寫入‘0’到這個位元位置來清除這個位元假使他決定不要在讀取這個位元後執行編碼解碼器IO週期 1 = 指示其他的驅動器透過AC-link執行編碼解碼器IO週期而目前進行存取的驅動器必須稍後再試一次（這個位元適用於所有編碼解碼器IO週期-GPIO或其他的）。	

15.8.3.9 PCM資料暫存器 (PCDR)

表15-15 PCM資料暫存器

實體位址 4050_0040		PCDR 暫存器	AC97
位元	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	PCM_RDATA   PCM_LDATA	
重置	0 0		
位元	名稱	說明	
31:16	PCM_RDATA	PCM 右通道資料	
15:0	PCM_LDATA	PCM 左通道資料	

寫入32位元取樣到這個暫存器來更新資料到PCM傳送的FIFO，讀取這個暫存器會從PCM接收FIFO得到32位元取樣



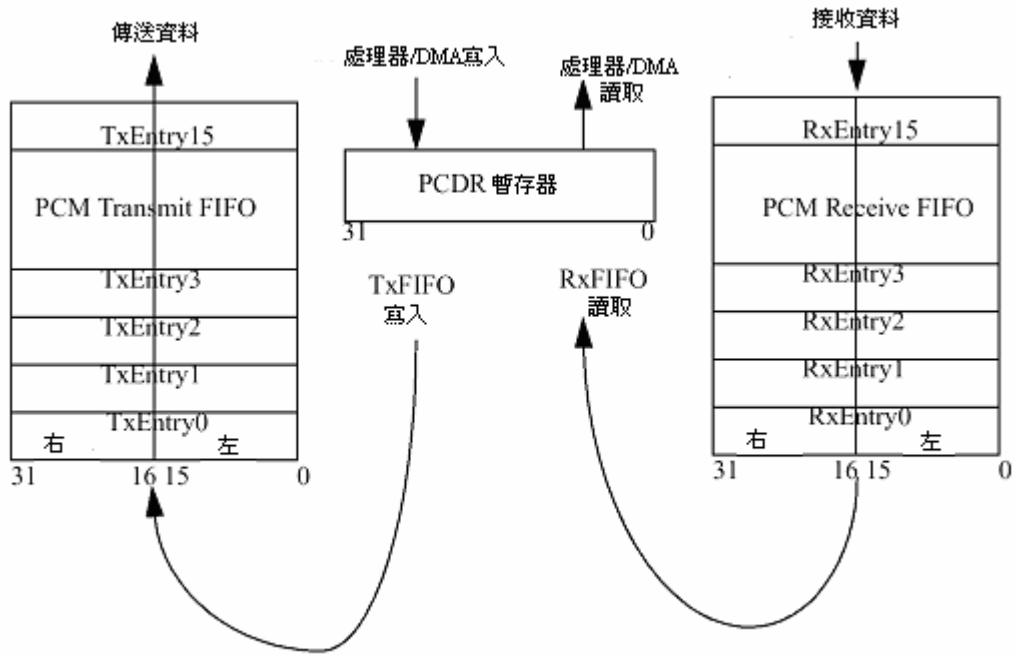


圖15-9 PCM 傳送和接收作業

15.8.3.10 麥克風輸入控制暫存器 (MCCR)

表15-16 麥克風輸入暫存器

實體位址 4050_0008		MCCR 暫存器	AC97
位元	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	RESERVED	
重置	0 0	FEIE Reserved	
位元	名稱	說明	
31:4	Reserved	保留	
3	FEIE	FIFO 錯誤中斷啟動 (FEIE) 這個位元控制接收的FIFO錯誤發生是否會造成中斷 0 = 即使MCSR的位元4被設置，還是沒有中斷發生 1 = 假如MCSR的位元4被設置，中斷將會發生	
2:0	Reserved	保留	

15.8.3.11 麥克風輸入狀態暫存器 (MCSR)



表15-17 麥克風輸入狀態暫存器

實體位址 4050_0018		MCSR 暫存器	AC97
位元	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0		
	RESERVED		FIFOE Reserved
重置	0 0		
位元	名稱	說明	
31:5	Reserved	保留	
4	FIFOE	FIFO 錯誤 (FIFOE) 0 = 沒有接收的FIFO錯誤發生 1 = 接收的FIFO錯誤發生，假如接收的FIFO滿溢發生則這個位元被設置，在此情況下，FIFO 指標不會增加，從AC-link進來的資料不會被寫入FIFO而且會消失，這是由於DMA控制器有過度頻寬的需求，因此無法及時清除接收的FIFO 經由寫入1到這個位元位置來將位元清除	
3:0	Reserved	保留	

15.8.3.12 麥克風輸入資料暫存器 (MCDR)

表15-18 麥克風輸入資料暫存器

實體位址 4050_0060		MCDR 暫存器	AC97
位元	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0		
	Reserved		MIC_IN_DAT
重置	0 0		
位元	名稱	說明	
31:16	Reserved	保留	
15:0	MIC_IN_DAT	麥克風輸入資料	

麥克風輸入資料暫存器是一唯讀暫存器，寫入到這個暫存器是無作用的，讀取這個暫存器會從麥克風輸入接收的FIFO得到32位元的取樣

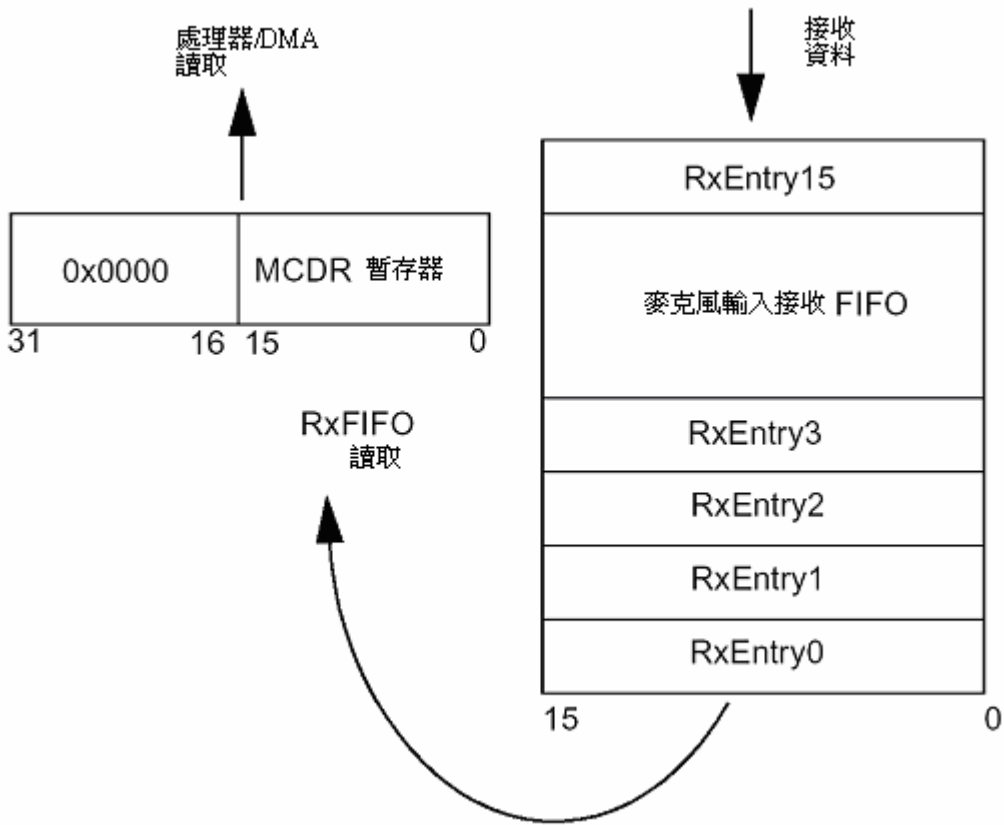


圖15-10 麥克風輸入只接收作業

15.8.3.13 數據機輸出控制暫存器 (MOCR)

表15-19 數據機輸出控制暫存器

實體位址 4050_0100		MOCR 暫存器	AC97
位元	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	RESERVED	FEIE Reserved
重置	0 0		
位元	名稱	說明	
31:4	Reserved	保留	
3	FEIE	FIFO 錯誤中斷啟動 (FEIE) 這個位元控制傳送的FIFO錯誤發生是否會造成中斷 0 = 即使MOSR的位元4被設置，還是沒有中斷發生 1 = 假如MOSR的位元4被設置，中斷將會發生	
2:0	Reserved	保留	

15.8.3.14 數據機輸入控制暫存器 (MICR)

表15-20 數據機輸入控制暫存器

實體位址 4050_0108		MICR 暫存器	AC97
位元	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	<div style="border: 1px solid black; padding: 5px;"> <div style="text-align: center; border-bottom: 1px solid black;">RESERVED</div> <div style="display: flex; justify-content: space-between; border-top: 1px solid black;"> <span style="writing-mode: vertical-rl; transform: rotate(180deg);">FEIE</span> <span style="writing-mode: vertical-rl; transform: rotate(180deg);">Reserved</span> </div> </div>	
重置	0 0		
位元	名稱	說明	
31:4	Reserved	保留	
3	FEIE	FIFO 錯誤中斷啟動 (FEIE) 控制接收的FIFO錯誤是否會造成中斷 0 = 即使MISR的位元4被設置，還是沒有中斷發生 1 = 假如MISR的位元4被設置，中斷將會發生	
2:0	Reserved	保留	

15.8.3.15 數據機輸出狀態暫存器 (MOSR)

表15-21 數據機輸出狀態暫存器

實體位址 4050_0110		MOSR 暫存器	AC97
位元	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	<div style="border: 1px solid black; padding: 5px;"> <div style="text-align: center; border-bottom: 1px solid black;">RESERVED</div> <div style="display: flex; justify-content: space-between; border-top: 1px solid black;"> <span style="writing-mode: vertical-rl; transform: rotate(180deg);">FIFOE</span> <span style="writing-mode: vertical-rl; transform: rotate(180deg);">Reserved</span> </div> </div>	
重置	0 0		
位元	名稱	說明	
31:5	Reserved	保留	
4	FIFOE	FIFO 錯誤 (FIFOE) 0 = 沒有傳送的FIFO錯誤發生 1 = 傳送的FIFO錯誤發生，假如傳送的FIFO欠載發生則這個位元被設置，在此情況下，最後有效取樣重複地傳送出去而且指標不會增加，會發生這樣是由於： c. 沒有更多有效的緩衝區資料可用來傳送 d. 緩衝區資料可用但是DMA控制器有過度頻寬的需求經由寫入1到這個位元位置來將位元清除	
3:0	Reserved	保留	

15.8.3.16 數據機輸入狀態暫存器 (MISR)

表15-22 數據機輸入狀態暫存器

實體位址 4050_0118		MISR 暫存器	AC97
位元	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0		
		RESERVED	FIFOE Reserved
重置	0 0		
位元	名稱	說明	
31:5	Reserved	保留	
4	FIFOE	FIFO 錯誤 (FIFOE) 0 = 沒有接收的FIFO錯誤發生 1 = 接收的FIFO錯誤發生，假如接收的FIFO滿溢發生則這個位元被設置，在此情況下，FIFO 指標不會增加，從AC-link進來的資料不會被寫入FIFO而且會消失，這是由於DMA控制器有過度頻寬的需求，因此無法及時清除接收的FIFO 經由寫入1到這個位元位置來將位元清除	
3:0	Reserved	保留	

15.8.3.17 數據機資料暫存器 (MODR)

表15-23 數據機資料暫存器

實體位址 4050_0140		MODR 暫存器	AC97
位元	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0		
		Reserved	MODEM_DAT
重置	0 0		
位元	名稱	說明	
31:16	Reserved	保留	
15:0	MODEM_DAT	數據機資料	

32位元取樣寫入到這個暫存器來更新資料到數據機傳送的FIFO，讀取這個暫存器會從數據機接收FIFO得到32位元取樣。

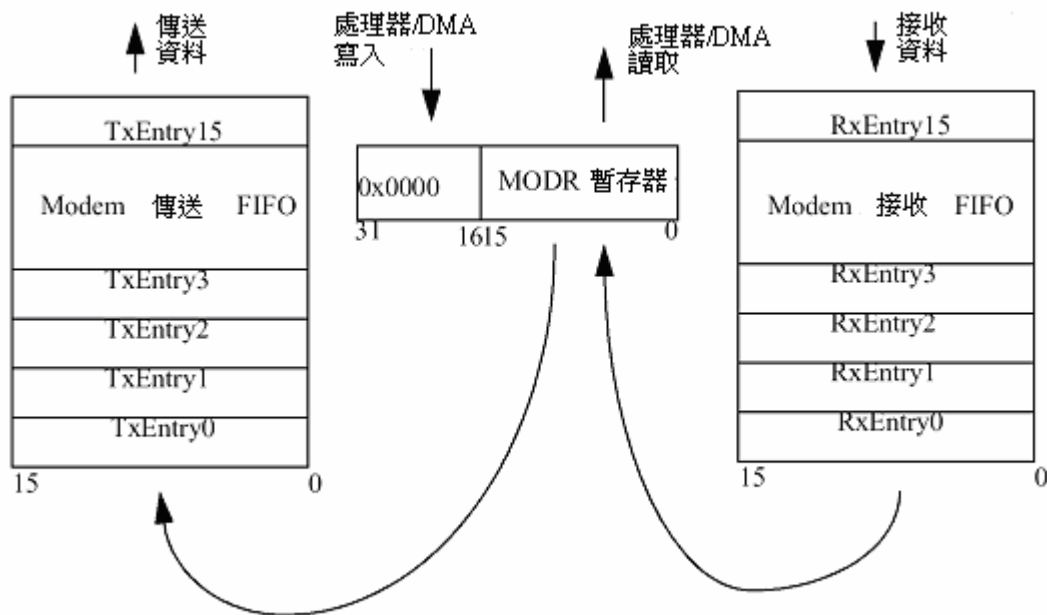


圖15-11 數據機傳送和接收作業

15.8.3.18 存取編碼解碼暫存器

每個編碼解碼器有六十四個16位元暫存器，可以一半字元邊界（16位元邊界）內部定址到編碼解碼器，因為應用程式處理器只支援字元邊界（32位元邊界）的內部暫存器存取，軟體必須選擇下列公式的其中一種來把7位元的編碼解碼器位址轉換成32位元的應用程式處理器位址：

- 主要音效編碼解碼器的應用程式處理器實體位址  
= 0x4050-0200 + 左移一次（內部7位元編碼解碼暫存器位址）
- 輔助音效編碼解碼器的應用程式處理器實體位址  
= 0x4050-0300 + 左移一次（內部7位元編碼解碼暫存器位址）
- 主要數據編碼解碼器的應用程式處理器實體位址  
= 0x4050-0400 + 左移一次（內部7位元編碼解碼暫存器位址）
- 輔助數據編碼解碼器的應用程式處理器實體位址  
= 0x4050-0500 + 左移一次（內部7位元編碼解碼暫存器位址）

在等式中，左移一次（）將7位元編碼解碼暫存器位址向左移動一個位元而且移0到LSB（最低有效位元），位址的轉換在表15-64。

表15-24 編碼解碼暫存器位址映射（1 of 2）

7位元編碼解碼器位址	主要音效編碼解碼器的應用	輔助音效編碼解碼器的應用	主要數據編碼解碼器的應用	輔助數據編碼解碼器的應用
------------	--------------	--------------	--------------	--------------

	程式處理器實體位址	程式處理器實體位址	程式處理器實體位址	程式處理器實體位址
0x00	0x4050_0200	0x4050_0300	0x4050_0400	0x4050_0500
0x02	0x4050_0204	0x4050_0304	0x4050_0404	0x4050_0504
0x04	0x4050_0208	0x4050_0308	0x4050_0408	0x4050_0508
0x06	0x4050_020C	0x4050_030C	0x4050_040C	0x4050_050C
0x08	0x4050_0210	0x4050_0310	0x4050_0410	0x4050_0510
0x0A	0x4050_0214	0x4050_0314	0x4050_0414	0x4050_0514
0x0C	0x4050_0218	0x4050_0318	0x4050_0418	0x4050_0518
0x0E	0x4050_021C	0x4050_031C	0x4050_041C	0x4050_051C
0x10	0x4050_0220	0x4050_0320	0x4050_0420	0x4050_0520
0x12	0x4050_0224	0x4050_0324	0x4050_0424	0x4050_0524
0x14	0x4050_0228	0x4050_0328	0x4050_0428	0x4050_0528
0x16	0x4050_022C	0x4050_032C	0x4050_042C	0x4050_052C
0x18	0x4050_0230	0x4050_0330	0x4050_0430	0x4050_0530
0x1A	0x4050_0234	0x4050_0334	0x4050_0434	0x4050_0534
0x1C	0x4050_0238	0x4050_0338	0x4050_0438	0x4050_0538
0x1E	0x4050_023C	0x4050_033C	0x4050_043C	0x4050_053C
0x20	0x4050_0240	0x4050_0340	0x4050_0440	0x4050_0540
0x22	0x4050_0244	0x4050_0344	0x4050_0444	0x4050_0544
0x24	0x4050_0248	0x4050_0348	0x4050_0448	0x4050_0548
0x26	0x4050_024C	0x4050_034C	0x4050_044C	0x4050_054C
0x28	0x4050_0250	0x4050_0350	0x4050_0450	0x4050_0550
0x2A	0x4050_0254	0x4050_0354	0x4050_0454	0x4050_0554
0x2C	0x4050_0258	0x4050_0358	0x4050_0458	0x4050_0558
0x2E	0x4050_025C	0x4050_035C	0x4050_045C	0x4050_055C
0x30	0x4050_0260	0x4050_0360	0x4050_0460	0x4050_0560
0x32	0x4050_0264	0x4050_0364	0x4050_0464	0x4050_0564
0x34	0x4050_0268	0x4050_0368	0x4050_0468	0x4050_0568
0x36	0x4050_026C	0x4050_036C	0x4050_046C	0x4050_056C
0x38	0x4050_0270	0x4050_0370	0x4050_0470	0x4050_0570
0x3A	0x4050_0274	0x4050_0374	0x4050_0474	0x4050_0574
0x3C	0x4050_0278	0x4050_0378	0x4050_0478	0x4050_0578
0x3E	0x4050_027C	0x4050_037C	0x4050_047C	0x4050_057C
0x40	0x4050_0280	0x4050_0380	0x4050_0480	0x4050_0580
0x42	0x4050_0284	0x4050_0384	0x4050_0484	0x4050_0584

表15-24 編碼解碼暫存器位址映射 (2 of 2)

7位元編碼解碼器位址	主要音效編碼解碼器的應用程式處理器實體位址	輔助音效編碼解碼器的應用程式處理器實體位址	主要數據編碼解碼器的應用程式處理器實體位址	輔助數據編碼解碼器的應用程式處理器實體位址
0x44	0x4050_0288	0x4050_0388	0x4050_0488	0x4050_0588
0x46	0x4050_028C	0x4050_038C	0x4050_048C	0x4050_058C
0x48	0x4050_0290	0x4050_0390	0x4050_0490	0x4050_0590
0x4A	0x4050_0294	0x4050_0394	0x4050_0494	0x4050_0594
0x4C	0x4050_0298	0x4050_0398	0x4050_0498	0x4050_0598
0x4E	0x4050_029C	0x4050_039C	0x4050_049C	0x4050_059C
0x50	0x4050_02A0	0x4050_03A0	0x4050_04A0	0x4050_05A0
0x52	0x4050_02A4	0x4050_03A4	0x4050_04A4	0x4050_05A4
0x54	0x4050_02A8	0x4050_03A8	0x4050_04A8	0x4050_05A8
0x56	0x4050_02AC	0x4050_03AC	0x4050_04AC	0x4050_05AC
0x58	0x4050_02B0	0x4050_03B0	0x4050_04B0	0x4050_05B0
0x5A	0x4050_02B4	0x4050_03B4	0x4050_04B4	0x4050_05B4
0x5C	0x4050_02B8	0x4050_03B8	0x4050_04B8	0x4050_05B8
0x5E	0x4050_02BC	0x4050_03BC	0x4050_04BC	0x4050_05BC
0x60	0x4050_02C0	0x4050_03C0	0x4050_04C0	0x4050_05C0
0x62	0x4050_02C4	0x4050_03C4	0x4050_04C4	0x4050_05C4
0x64	0x4050_02C8	0x4050_03C8	0x4050_04C8	0x4050_05C8
0x66	0x4050_02CC	0x4050_03CC	0x4050_04CC	0x4050_05CC
0x68	0x4050_02D0	0x4050_03D0	0x4050_04D0	0x4050_05D0
0x6A	0x4050_02D4	0x4050_03D4	0x4050_04D4	0x4050_05D4
0x6C	0x4050_02D8	0x4050_03D8	0x4050_04D8	0x4050_05D8
0x6E	0x4050_02DC	0x4050_03DC	0x4050_04DC	0x4050_05DC
0x70	0x4050_02E0	0x4050_03E0	0x4050_04E0	0x4050_05E0
0x72	0x4050_02E4	0x4050_03E4	0x4050_04E4	0x4050_05E4
0x74	0x4050_02E8	0x4050_03E8	0x4050_04E8	0x4050_05E8
0x76	0x4050_02EC	0x4050_03EC	0x4050_04EC	0x4050_05EC
0x78	0x4050_02F0	0x4050_03F0	0x4050_04F0	0x4050_05F0
0x7A	0x4050_02F4	0x4050_03F4	0x4050_04F4	0x4050_05F4
0x7C	0x4050_02F8	0x4050_03F8	0x4050_04F8	0x4050_05F8
0x7E	0x4050_02FC	0x4050_03FC	0x4050_04FC	0x4050_05FC

## 問題：

1. AC97可以接多少個編解碼器？
2. AC97如何分辨不同的編解碼器來源？
3. AC連結有幾隻接腳？他們分別代表著什麼功能？
4. AC97的音效訊框中有多少個存儲格？
5. AC97的音效訊框中的第一個存儲格長度為何？代表著何種意義？
6. 列出AC97的音效訊框中的第一個存儲格每一個位元所代表意義？
7. AC連結在低電源模式下有哪兩種方式去喚醒AC連結？
8. ACUNIT有多少個FIFO？
9. AC97編解碼器的取樣頻率為多少？可否低於此頻率來取樣？
10. 為何ACUNIT有尾數填零的功能？它用於何處？
11. 何謂編解碼器（codec）？
12. AC97控制單元（ACUNIT）的特性為何？