
第五章 硬體整合與時序圖

從第二章封包的偵測、頻率偏移的估計與補償，到第三章封包取樣點的不同步，乃至於第四章通道特性的評估與快速傅立葉的轉換，我們都已經各自介紹過他們的硬體設計與實現，但是這對於我們希望整合整個系統的目標，還有一段的差距；因此本章的重點，便在於介紹每個獨立的硬體單元間，該如何互相的配合與利用，以期能達到有效整合系統的目的。在介紹完系統的整合後，本章將會繼續說明整合後系統的時序圖(Timing Diagram)。

5.1 硬體的整合

下圖(5.1.1)即是整個硬體整合的架構圖，其運作的情形主要是由一個獨立出來的控制單元來決定；控制單元會送出各個控制訊號至每個獨立的硬體單元，掌控整個系統運行的機制，以及整合每個獨立單元間訊號的配合。

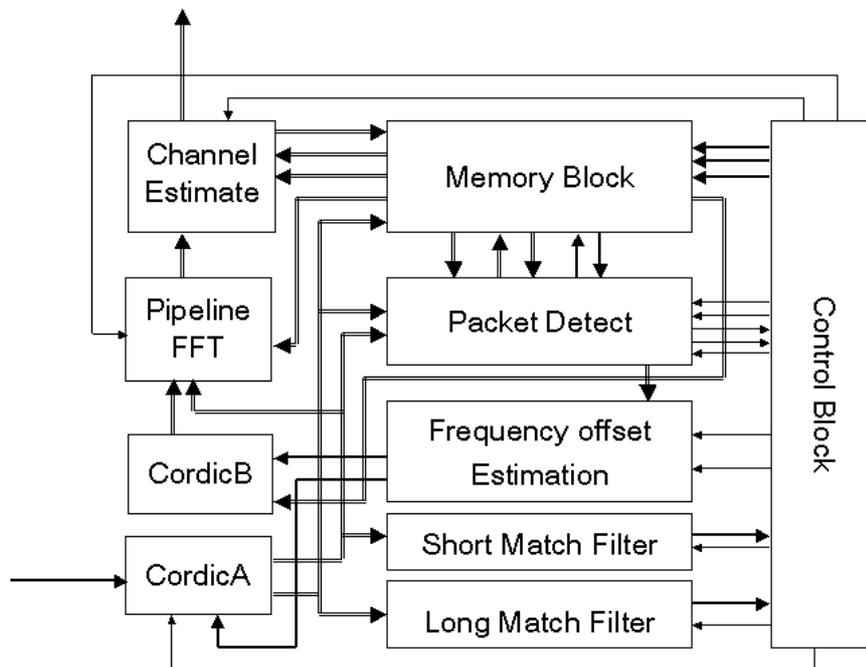


圖 5.1.1 系統整合架構圖

接著我們就依照使用硬體單元的順序，來介紹各個單元間訊號的配合，以及控制訊號的使用。

5.1.1 硬體單元 CORDIC A

本硬體單元主要是針對收到的取樣點，在複數平面上作旋轉的運算。其內部的架構其實就是第二章所介紹過的數位座標旋轉作旋轉角度的硬體，詳細情形可以參照第二章 2.2.4 節。此硬體單元的輸入訊號有二：一是從類比數位轉換器 (Analog to Digital Converter) 而來的資料取樣點；而另外一個輸入訊號是由 Frequency Offset Estimation 硬體單元所計算出來、需要補償的頻率偏移角度。至於他的輸出訊號也有兩個：第一個是經過角度旋轉後的輸出；而另外一個則是將輸入訊號經過延遲兩個時脈 (Clock) 後的結果；延遲兩個時脈的原因，是因為數位座標旋轉作旋轉角度的硬體，需要兩個時脈的時間，才可以將旋轉過後的輸入訊號，正確的輸出；因此為了能將後端硬體接收到的資料取樣點同步，所以我們刻意輸出延遲兩個時脈的輸入訊號，如此對於接續在此硬體單元後端的硬體，將可以同時收到經過頻率偏移補償和未經過頻率偏移補償的兩個輸入訊號，而這兩個訊號是同一個來源的資料取樣點。除了輸入訊號和輸出訊號外，此硬體還有從控制單元 (Control Block) 而來的控制訊號，但是這控制訊號僅只用於控制該硬體的使用與否，其目的在於節省能量的散失，以減小因過熱造成系統運作不正確的可能。

5.1.2 硬體單元 Packet Detection

接著我們來看 Packet Detection 的硬體單元，其實這個硬體單元就跟我們在第二章中，介紹過的封包偵測硬體是一樣；其輸入訊號有五個，其中有三個是來自於 Memory Block 硬體單元，目的是為了實踐正相關函數運算和滑動框架的架構，相關的演算法與硬體操作可以參照第二章的 2.1 節；而另外的兩個輸入訊號則是來自於硬體單元 CORDIC A；這兩個輸入訊號分別是經過頻率偏移補償和未經過頻率偏移補償的資料取樣點，而這兩個輸入訊號都需要傳入 Packet Detection 硬體單元的原因，是因為我們需要用到未經過頻率偏移補償的資料取

樣點，求得其所對應的複數值 C_n 作頻率偏移的粗估；然後再將經過補償粗估頻率偏移後的資料取樣點，所求得的複數值 C_n' ，再作一次細估的動作，如此才可以減少頻率偏移所帶來的干擾。

至於輸出訊號的話，此硬體單元中只有三個輸出訊號；其中兩個跟上述與 Memory Block 硬體單元相連的三個輸入訊號一樣，是為了實現正相關函數的運算和滑動框架的架構，所以這兩個輸出訊號線亦與 Memory Block 硬體單元相連，目的是將此硬體運算出來的結果，寫回到 Memory Block 硬體單元中記憶體內；而最後一個輸出的訊號便是要用來求頻率偏移的 C_n 訊號，因此該訊號線將會與 Frequency Offset Estimation 硬體單元相連，目的是要求得複數值 C_n 的主幅角，以利頻率偏移的補償。

最後，我們來看此硬體單元的控制訊號。細分來說，控制訊號可以分為輸入的控制訊號和輸出的控制訊號，兩者的差別在於：輸入的控制訊號是 Control Block 硬體單元用來控制該單元的運作；而輸出的控制訊號，則是該硬體反應給 Control Block 硬體單元，表示該硬體目前輸出結果的狀態。在 Packet Detection 硬體單元裡有三個輸入控制訊號，這三個輸入控制訊號分別是 Ctrl_A 和 Ctrl_B 以及 Pd_Enable；Ctrl_A 是用來控制正相關函數運算的輸入資料，是經過頻率偏移補償或是未經過頻率偏移補償的取樣點；Ctrl_B 則是用來控制滑動框架的大小為六十四還是一百二十八個取樣點。而 Pd_Enable 則是用來控制 Packet Detection 硬體單元的運行與否。除了三個輸入控制訊號外，還有兩個輸出控制訊號分別是 S_Cond 和 L_Cond；S_Cond 是 Packet Detection 硬體單元告知 Control Block 硬體單元是否接收到一個短調整符元的輸出控制訊號線，其判斷的依據就是 C_n 和 P_n 的比值是否大於我們所訂的 Threshold，若是則表示有偵測到短調整符元，若否則硬體將繼續執行偵測短調整符元的運作；相同的，L_Cond 也是該硬體單元通知 Control Block 硬體單元是否接收到一個長調整符元的輸出控制訊號線，當 L_Cond 為高電位時表示接收到一個長調整符元，若為低電位時則表示未接收到，因此 Control Block 硬體單元可以利用 L_Cond 訊號線的位準來判斷現在運行的階段，以利控制接下來的運作。

5.1.3 硬體單元 Frequency Offset Estimation

接下來我們來看硬體單元 Frequency Offset Estimation。顧名思義，這個硬體單元要處理的便是頻率偏移的估計，因此該硬體單元由兩個部分組成：第一個是數位座標旋轉(CORDIC)作求主幅角用的硬體單元；而另一個部分則是累加器。其運作原理即是接收由 Packet Detection 運算單元而來的複數值 C_n ，接著求得該複數值所對應的主幅角，最後利用累加器將求得的頻率偏移角度累加，然後輸出至數位座標旋轉(CORDIC)作旋轉角度用的硬體單元，作頻率偏移的補償以完成此硬體單元的運作。

此硬體單元只有一個資料的輸入訊號，也就是由 Packet Detection 所輸出的 C_n 訊號；而其資料的輸出訊號有兩個，這兩個分別是經由兩個不同的累加器累加出來的結果，而這兩個結果都是需要補償的頻率偏移角度，所以這兩個輸出的資料訊號會分別接到兩個數位座標旋轉(CORDIC)作旋轉角度用的硬體單元，也就是 CORDIC A 和 CORDIC B；會需要兩個分開的累加器的原因是因為：在第四章中，我們提過我們需要用到長調整符元來做頻率偏移的微調，所以當我們求得需要補償的微調頻率偏移角度時，長調整符元已經存入記憶體中，以利將來通道特性的評估；可是這些存入的資料都是未經過微調頻率偏移補償的，所以若拿這些資料作通道特性的評估，將無法求得正確的通道特性，所以我們勢必需要先將這些長調整符元的取樣點，作過微調的頻率偏移補償後，再作通道特性的評估；可是此時適逢第一個資料符元的取樣點輸入，所以我們需要使用 CORDIC A 對輸入的資料作頻率偏移的補償，而無法共用該硬體單元，用以對記憶體內的長調整符元作微調的頻率偏移補償，所以使用另外一個數位座標旋轉(CORDIC)作旋轉角度用的硬體單元是在所難免的；這也就是需要 CORDIC A 和 CORDIC B 兩個硬體單元的原因。

最後我們來看 Frequency Offset Estimation 的控制訊號；該硬體單元的控制訊號只有兩條而且這兩條都是輸入控制訊號，分別是 TT_Sel1 和 TT_Sel2；這兩個訊號分別控制著兩個累加器的運作，控制著是要重設(Reset)累加器內，暫存器所暫存的累加值，亦或是控制著累加器作累加的動作。

5.1.4 硬體單元 Short Match Filter

接著我們來看 Short Match Filter 硬體單元；該硬體單元就如同在第三章的 3.1.1 節和 3.1.2 節中所述，主要的目的是用來偵測短調整符元和短調整符元間的界線，藉此來達到同步的目的。

受限於我們無法知道第一個接收機所取樣到的取樣點位置，所以我們需要盡可能的早點得知界線所在，因此 Short Match Filter 的資料輸入訊號，必須要選取尚未補償頻率偏移的資料取樣點，就如同圖 5.1.1 所示。

除了有一個資料的輸入訊號外，該硬體單元還有一個輸出的資料訊號、Short_fb，和一個輸入的控制訊號、SMF_Enable；Short_fb 是用來告知 Control Block 硬體單元：取樣到現在這個取樣點後，再經過多少個取樣點便會到達下一個調整符元間的界線。而 SMF_Enable 這個輸入控制訊號線就如同之前一般，用來控制此硬體單元的運作與否，用來節省能量的使用，同時也可以避免因系統過熱而導致運作的失常。

5.1.5 硬體單元 Long Match Filter

接著我們來看 Long Match Filter 硬體單元。該硬體單元的架構和演算法，可以參照第三章的 3.2 節，其目的就是在於估計出循環字首的長度，如果能夠正確的估計出循環字首的長度，再配合 Short Match Filter 硬體單元所求到的調整符元間的界線，如此便可以完整的同步接收到的取樣點。

不同於 Short Match Filter，在時間上來說，此硬體單元有足夠的時間等待接收到的取樣點經過頻率偏移的補償，所以對於 Long Match Filter 的資料輸入訊號，我們可以選取經過頻率偏移補償的取樣點，就如同圖 5.1.1 所示。

除了有一個輸入的資料訊號外，此硬體單元還有一個輸出的資料訊號、CP_Est，和一個輸入的控制訊號 LMF_Enable；CP_Est 輸出的資料訊號，即是將 Long Match Filter 所估計出來的循環字首長度送至 Control Block，用以同步接下來的資料符元；而 LMF_Enable 就如同之前一般，是 Control Block 送來的輸入控制訊號，用來控制 Long Match Filter 的運作與否。

5.1.6 硬體單元 CORDIC B

如同我們在本章的 5.1.3 節中說過：受限於時間上的關係，我們無法只利用一個數位座標旋轉(CORDIC)硬體單元，在對記憶體內的長調整符元作微調的頻率偏移補償之時，同時又對接收機取樣到的資料取樣點，作完整的頻率偏移補償；所以我們勢必需要另外一個數位旋轉(CORDIC)硬體單元，而這正是本硬體單元 CORDIC B 使用的原因。

承前，CORDIC B 硬體單元主要的目的就在於要將存在記憶體內的長調整符元，作頻率偏移補償的微調，因此他的資料輸入訊號有二：一個是來自於儲存在記憶體內的長調整符元取樣點；而另外一個便是來自於本章 5.1.3 節中，累加器所求得、應補償的頻率偏移角度；而經過補償後的結果，需要拿來作通道特性的評估，所以此硬體單元的輸出資料訊號，會將補償後的結果送入 Pipeline FFT 的硬體單元，用以求得接收到的長調整符元、在頻域上所載的資料，如此才可以利用第四章的演算法，求得通道的特性。

5.1.7 硬體單元 Pipeline FFT

接著來看硬體單元 Pipeline FFT。Pipeline FFT 的架構我們已經在第四章的 4.2 節說明過了，其目的就是要將原本在時域上的取樣點轉換到頻域上，以求得載在載波上得資料，達到解調的目的。

在此硬體單元 Pipeline FFT 的資料輸入訊號有兩個，一個是由 CORDIC B 的輸出訊號而來，其使用的時機是當求得微調的頻率偏移時，CORDIC B 對長調整符元作頻率偏移的補償，然後透過此輸入訊號線，用以求得此長調整符元在頻域上的資料，作通道特性評估之用。另外一個輸入的訊號是來自於 CORDIC A，其目的是在求得正交分頻多工資料符元，在頻段上所載有的資料，然後送入 Channel Estimation 的硬體單元作通道特性的補償，以完成資料的解調。

不論是來自於 CORDIC A 的輸入資料訊號，還是來自於 CORDIC B 的輸入資料訊號，其轉換後所求得的结果都是要送入 Channel Estimation 的硬體單元中，所以此硬體的輸出資料訊號線將會與 Channel Estimation 的硬體單元相連；同

時，由於有兩個輸入的資料訊號，所以 Control Block 需要輸出一個對此 Pipeline FFT 硬體單元而言的輸入控制訊號，用來控制該採用何者資料輸入訊號，作快速傅立葉轉換。

5.1.8 硬體單元 Channel Estimation

Channel Estimation 的硬體單元架構，其實我們已經在第四章的 4.1.3 節中介紹過了，那時我們說過此硬體單元有三種模式，因此，會有一個從 Control Block 而來的輸入控制訊號，用來控制此硬體單元運作的模式。

此硬體單元的輸入資料訊號有三個，其中一個是來自於 Pipeline FFT 的輸出資料訊號，將長調整符元在頻域上的資料、或是正交分頻多工資料符元上所載的資料送入此硬體單元中，作通道特性的評估與補償；另外兩個資料輸入訊號來自於記憶體單元，目的在於配合之前由 Pipeline FFT 輸入的長調整符元在頻域上的資料，作通道特性評估、平均和內插的功用，相關的演算法可以參照第四章 4.1.1 節和 4.1.2 節。

至於此硬體單元的資料輸出訊號總共有兩個，一個是與記憶體單元相連的輸出訊號，用來將求得的通道特性寫回記憶體內；而另外一個則是將經過通道特性補償的資料輸出，完成整個解調的動作。

5.2 時序圖 Timing Diagram

下圖(5.2.1)是假設循環字首為三十二個取樣點的封包示意圖，以及其相對本論文設計之系統的時序圖(Timing Diagram)，我們將會在接下來一一介紹每個時序區間(Timing Diagram Interval)，本論文設計之系統所需處理的問題，以及同時硬體的運作。

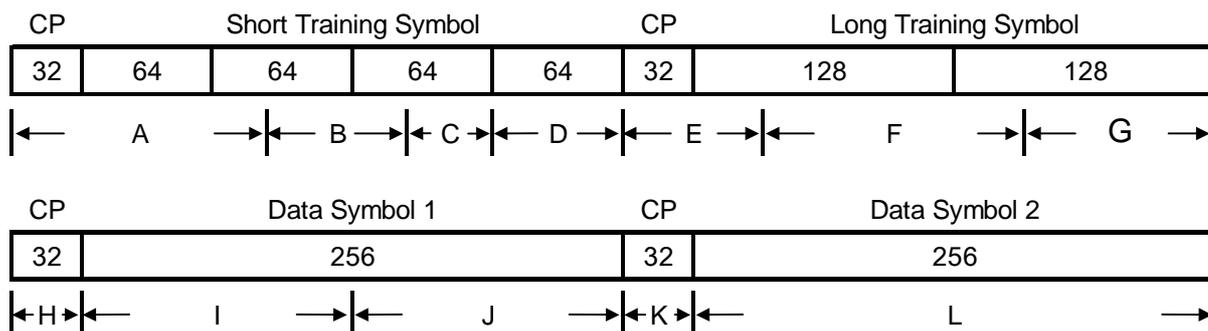


圖 5.2.1 封包時序圖

◆ 時序區間 A：

此區間 A 的開始位置應該要算是從整個硬體打開電源開始，主要的原因是因為對一個還沒有作好同步的硬體而言，我們是無法確定知道一個封包的起始位置，不過以示意圖來說，圖中區間 A 的開始位置就是表示著本硬體所要處理的第一個問題、封包的偵測。在區間 A 中，主要的目的便是要偵測封包的存在與否，此區間的大小是不一定的，端看整個傳輸的通道特性還有雜訊的影響，才會決定出區間的大小；但是我們不需要知道此區間的大小，我們只要去觀測相隔六十四個取樣點，其正相關函數值與訊號強度比是否大於我們所設定的邊界(Threshold)，若是，則此時所取樣到的取樣點便是區間 A 的結束位置；若否，則繼續求取下一個取樣點的正相關函數值與訊號強度比，直到大於我們所設定的邊界為止。

承上所述，雖然區間 A 的大小不是固定的，但是其也有一定的範圍，基本上區間 A 的大小應該介於六十四個取樣點和一百二十八個取樣點之間，其中的原因是因為：若區間 A 小於六十四個取樣點，則根本不可能有正相關的關係存在，故區間 A 應該要大於六十四個取樣點；若區間 A 大於一百二十八個取樣點，則表示此封包所經過的通道特性以及雜訊的影響過於嚴重，使得正相關的特性被瓦解得找不出來，所以理應放棄此封包。當然邊界(Threshold)的設定也要適合於傳輸的環境，如此才能將區間 A 的大小限制在六十四和一百二十八個取樣點之間，這也使得區間 A 的結束位置會落在短調整符元(Short Training Symbol)的第二個循環週期內，如同圖(5.2.1)中所示；而

在找到區間 A 的結束位置的同時，封包偵測的硬體(Packet Detection)也會使得 S_cond 訊號拉起為高電壓準位，通知控制單元(Control Block)作區間 B 的準備。

◆ 時序區間 B：

在區間 A 的結尾時，由於 S_cond 的高電壓準位，所以確立了區間 B 的起始位置。和區間 A 不一樣的，區間 B 的大小是固定為六十四個取樣點，而其存在的主要目的是為了確保估計頻率偏移的準確性；其中的原因是因為：之前我們說過區間 A 的大小不是固定的，可能為六十四到一百二十八個取樣點之間，因此我們無法確立頻率偏移估計所利用到的複數訊號 C_n ，是否是由兩個完整的短調整符元循環區間所求得而來；若否，則求得而來的頻率偏移將會有很大的誤差，這在將來要補償頻率偏移時，由於經過累加器的累加，使得其間的誤差越來越大，將可能會有不補償頻率偏移比補償頻率偏移而來得有較好的效能(Performance)的現象產生，當然這不是我們所樂見的。所以為了避免這樣的問題產生，我們特意增加了區間 B 的存在，而且固定了區間 B 的大小為六十四個取樣點；如此一來，在區間 B 的結束時，我們可以確定由封包偵測硬體所傳出的複數訊號 C_n ，必定是由兩個完整的短調整符元循環區間，作相隔六十四個取樣點的正相關函數所求得而來，如此也才可以確立頻率偏移估計的正確性。

因此，整個硬體在區間 B 的運作，其實是維持著跟區間 A 相同的情形，只是控制單元內的計數器自我累加，累加至區間 B 的結尾時，控制頻率偏移估計硬體，將此結尾所求得的複數訊號 C_n ，找到其相對應的主幅角，達到頻率偏移估計的目的，以供後面的取樣點作頻率偏移補償之用。也就是從下一個區間、區間 C 開始，在圖(5.1.1)中的 CORDIC A 硬體單元便會一直運作；換句話說，我們從區間 C 開始便可以得到經過頻率偏移粗調後的取樣點。

◆ 時序區間 C：

基本上，從整個硬體系統開始運作時，圖(5.1.1)中的 Short Match Filter 硬體單元便是一值在運作著，他隨時在將收到的取樣點送入硬體單元

中，找尋著距離下一次兩個調整符元間的界線(Boundary)，因此當區間 B 結束的時候，表示著現在已經偵測到封包的存在，所以此時 Short Match Filter 硬體單元所找到的界線是可以肯定的，因此控制單元便可以經由該硬體單元的訊號線 Short_fb 得知：需要再經過多少個取樣點便可以到達下一個調整符元間的邊界；這也就是區間 C 的由來。

因此，區間 C 其實是用來對齊兩兩調整符元間的界線，而當區間 C 結束的同時，也就表示著硬體的取樣點和封包的界線取樣點對齊，如此便完成了取樣點同步的第一個步驟。在區間 C 結束的同時，由於已經找到調整符元間的界線了，故 Short Match Filter 硬體單元在此次的封包解調中，不會有再使用的機會。所以控制單元便會將 SMF_Enable 的訊號控制線，從高電壓準位改為低電壓準位，停止 Short Match Filter 硬體單元的運作，以節省能量的散失和可能發生錯誤的機會。

◆ 時序區間 D、時序區間 E：

區間 D 和區間 E 存在的目的主要是為了找尋短調整符元和長調整符元間的界線；其實區間 D 是不一定需要存在的，其原因是因為我們無法預估當硬體啟用時，第一個取樣點的位置；在之前我們提過：區間 A 的大小應該要藉於六十四到一百二十八個取樣點之間。但這是在理想的狀況下，實際的情形，可能因為傳輸環境過差或是雜訊過大，也可能是因為硬體啟用後的第一個取樣點位置，剛好落於短調整符元的第一個循環週期內；這些都有可能造成在區間 C 結束時，所找到的調整符元間的邊界，並非如圖(5.2.1)中所示：為第三個短調整符元循環區間和第四個短調整符元循環區間的邊界，反而可能為第四個調整符元循環區間和長調整符元循環字首間的邊界；在這種情況下，區間 D 便不會存在。

不論是區間 D 還是區間 E，其長度大小皆為六十四個取樣點，原因是因為在區間 C 時，我們已經找出調整符元間的界線，而短調整符元循環區間的週期為六十四個取樣點，所以當我們在區間 C 結束後的第六十四個取樣點，去觀察封包偵測(Packet Detection)硬體的 L_cond 訊號，若該訊號為低電壓

準位，那就表示：在此之前所接收到的六十四個取樣點為短調整符元循環區間；而這正是區間 D 所代表的意義。相反的，若 L_cond 為高電壓準位，那表示：此時間隔六十四點的正相關函數值，與當初偵測到硬體時、間隔六十四點的正相關函數值的比值小於一個我們設定好的邊界(Threshold)，因此我們可以推敲得知：在此之前所接收到的六十四個取樣點為長調整符元的前六十四個取樣點，而這也正是區間 E 所代表的意義。

因此，就硬體的運作而言，當區間 C 結束後，控制單元便會一直自我累加計數器，當累加器內的暫存器值為六十四時，查看此時經由 Short Match Filter 所傳來的 L_cond 訊號，若為低電位，那便進入區間 E 的運作；若為高電位，則進入到區間 F 的運作，藉此找到長調整符元循環字首的起始位置。

◆ 時序區間 F：

由於訊號線 L_cond 的高電壓準位，所以我們可以確立了現在接收到的取樣點，已經進入到長調整符元的階段，但是目前面臨到的問題是：不知道循環字首的長度；所以控制單元無法得知再經過多少個取樣點後，會到達長調整符元的結束位置；因此，區間 F 就是為了求得循環字首長度所準備的。

區間 F 的長度大小固定為一百二十八個取樣點，採用那麼大的取樣點數的原因，是因為在圖(5.1.1)中的 Long Match Filter 硬體單元，至少需要輸入一整個長調整符元的取樣點，如此才可以找出循環字首的長度，相關的演算法可以參照第三章的 3.2 節；但是循環字首的長度最大可能為六十四個取樣點，因此在此最大可能下，我們至少需要輸入一百九十二個取樣點至 Long Match Filter 硬體單元，如此才可以求得正確的循環字首長度；在區間 F 的長度大小固定為一百二十八個取樣點的情況下，配合著區間 E 的六十四個取樣點，恰好可以成為長調整符元的前一百九十二個取樣點，因此當區間 F 結束的同時，也就是求得循環字首長度的時候。

區間 F 除了用來作求得循環字首長度的準備外，由於長調整符元也是我們用來求得微調頻率偏移的一組重要資料，而整個系統硬體也必須要再區間 E 的結束，也就是區間 F 的開始之時，才可以確定目前所處理的取樣點位於

長調整符元內；因此在區間 F 開始的時候，控制單元亦會控制著封包偵測硬體單元內的控制訊號 Ctrl_A 和 Ctrl_B，使得封包偵測硬體會作相隔一百二十八個取樣點的正相關函數值，用以求得兩個長調整符元內循環區間的正相關函數值 C_n ，再送入頻率偏移估計的硬體單元內，求得微調的頻率偏移，以便將來作完整的頻率偏移補償。

◆ 時序區間 G：

當時序區間 F 結束的時候，此時控制單元(Control Block)可以經由 Long Match Filter 硬體單元的輸出訊號線 CP_Est 得知此時該硬體單元估計出來的循環字首長度，在得知循環字首長度後，我們便可以很容易的得知距離長調整符元的結束位置，還需要再經過多少個取樣點，這也就是區間 G 所存在的意義；區間 G 存在的目的是為了將硬體所接收到的取樣點，對齊到長調整符元的結束位置，如此便能完整的同步整個封包的取樣點。

在區間 G 的結束位置時，此時封包偵測的硬體，亦會求得兩長調整符元內循環區間的正相關函數值 C_n ，而這 C_n 值亦會被送到頻率偏移估計的硬體單元內，計算出微調的頻率偏移，因此，在區間 G 之後的取樣點，將可以利用圖(5.1.1)中的 CORDIC A 硬體單元，作出完整的頻率偏移補償。

◆ 時序區間 H：

緊接在長調整符元之後的，便是第一個資料符元的循環字首，由於我們在區間 F 的結束之時，已經求得循環字首的長度，因此時序區間 H 的大小，我們已經可以得知；同時，在這個時序區間內，沒有需要運作的硬體機制，所以整個硬體所要作的事情，便是控制單元累加器自我的累加，直到累加的結果等同於當初估計出來的循環字首長度即可，接著便接續著時序區間 I。

◆ 時序區間 I：

隨著時序區間 H 的結束，這也就表示著接收到的資料取樣點進入到載有資料的取樣點階段，雖然這些資料的取樣點已經經過了頻率偏移補償的調整，但是通道特性的影響我們尚未消弭，因此，本區間的目的就在於：將之前兩

百五十六點的長調整符元取樣點送入 Pipeline FFT 的硬體單元中，求得該取樣點所對應到在頻譜上的資料點，接著回存到記憶體中，以便下一個時序區間 J 來作通道特性的評估。由於我們之前提過，在本硬體的設計上，系統的時脈頻率為資料取樣頻率的兩倍，所以原本需要作兩百五十六點的 Pipeline FFT 硬體單元，僅只需要一百二十八個資料取樣點便可以完成；因此，區間 I 的大小固定為一百二十八個取樣點，如同圖(5.2.1)所示。

◆ 時序區間 J：

在時序區間 I 時，除了將長調整符元的資料取樣點輸入到 Pipeline FFT 的硬體單元外，接收機同時也是在接收資料取樣點的，而這些接收到的資料取樣點會先暫時存放在記憶體內，等到在此區間 J 時，才會從記憶體內讀出，並且送入 Pipeline FFT 硬體單元中，準備求得第一個資料符元在頻率上所載有的資料；同時，Pipeline FFT 硬體單元由於資料的送入，因此會將長調整符元資料取樣點所對應到的頻譜資料送出到圖(5.1.1)中、Channel Estimation 的硬體單元，用以求得通道的特性，並起會回存於記憶體中，以利將來作通道特性的補償；相關的 Pipeline FFT 運作的機制和通道特性的評估，可以參照第四章的 4.1 和 4.2 節。

因此當此區間結束時，兩百五十六個取樣點的第一個資料符元，將會存放於 Pipeline FFT 的硬體單元中處理，直到再區間 L 時的第二個資料符元取樣點輸入，才會將此兩百五十六個取樣點的第一比資料符元，經過快速傅立葉轉換後的結果輸出；而原本存在於 Pipeline FFT 硬體單元內的兩百五十六個長調整符元取樣點，由於第一比資料符元取樣點的輸入，所以其快速傅立葉轉換的結果將彙輸出，並且數入 Channel Estimation 硬體單元內，用以求得通道的特性，而求出的結果將會寫入記憶體中，等待著將來資料符元經過快速傅立葉轉換後，得到其在頻域上所相對應的結果，作通道特性的補償。

◆ 時序區間 K：

緊接著在第一比資料符元之後的，便是第二比資料符元的循環字首；由於我們已經在區間 F 結束之時，評估出循環字首的長度，因此我們可以很容

易的得知此區間 K 的長度大小，此時，不論是第一比的資料取樣點，還是利用長調整符元所求得的通道特性，都已經分別在 Pipeline FFT 的硬體單元和記憶體內準備就緒；因此本區間 K 沒有需要運作的機制，只需要累加控制單元內的累加器，直到接收機所接收的資料取樣點，到達循環字首的結尾，接著準備進入區間 L 的階段。

◆ 時序區間 L：

到了時序區間 L，此時接收機接收著第二比資料符元的資料取樣點，同時將這些資料點輸入 Pipeline FFT 的硬體單元中；對 Pipeline FFT 的硬體單元來說，由於有資料取樣點的輸入，所以會造成此硬體將第一比資料符元經過快速傅立葉轉換後的結果輸出，而這些輸出的結果，將會成為 Channel Estimation 硬體單元的輸入；在此同時，記憶體也會將區間 J 所求出的通道特性輸入到 Channel Estimation 硬體單元中，供其硬體對第一比資料符元上所載有的資料，作通道特性的補償，完成此區間 L 的運作。

因此當此區間 L 結束的時候，第一比資料符元所載有的資料，將會被解調完成並成為 Channel Estimation 硬體單元的輸出；此時，兩百五十六個取樣點的第二比資料符元，將會存放於 Pipeline FFT 硬體單元中處理，等待著第三比資料符元取樣點的輸入，再將第二比資料符元經快速傅立葉轉換後的結果輸出；而通道的特性還是存放於記憶體內，等待下次的使用。

◆ 在區間 L 之後時序區間：

在區間 L 之後的時序區間，其實就是維持著區間 K 到區間 L 再回到區間 K，如此一般的週期；主要的原因是因為在區間 L 之後的資料符元，其實硬體運作的機制都是：在循環字首時，控制單元空數其累加器的值，以求對齊資料符元的起始位置；而在資料符元時，硬體的運行便是要將在此資料符元的前一個資料符元解調，利用將此次資料符元取樣點輸入到 Pipeline FFT 硬體單元，以獲得上次資料符元經過快速傅立葉轉換的結果，同時控制記憶體輸出通道的特性，最後經過 Channel Estimation 的硬體單元解調，完成整個硬體運行的機制。