

# USB2.0信號分析技巧

## 前言

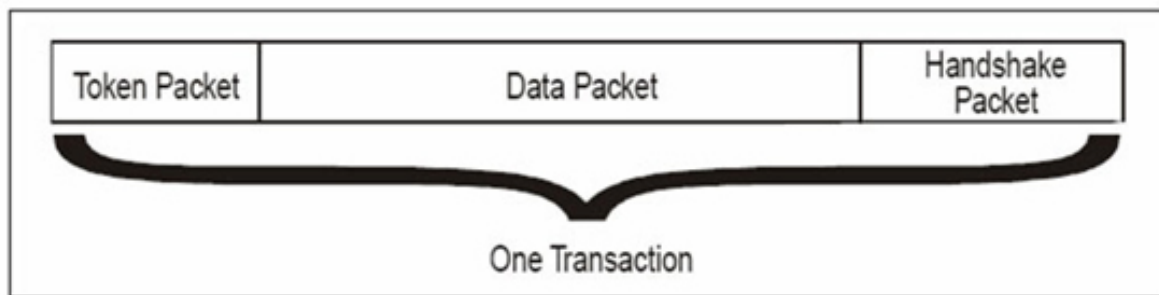
USB是很常見的串列協定。USB有熱插拔的特性，當USB裝置插入主機時，主機會偵測到，並且軟體會對其作規劃，即插即用。USB的連接線供應裝置電源，可提供DC+5V的電壓，電流供應約100mA~500mA。

USB的連接線共有四條，其中兩條是電源與地線，另兩條是訊號線，分別是D+與D-。一般情況兩條線的訊號是相反的，若其中一條的訊號為High，則另一條的訊號就為Low。如此的差動訊號能提升訊號抗雜訊抗干擾的能力，USB就是利用D+ D-兩條訊號線的差動訊號，來達成高速傳輸的目的。

USB主機端在沒有裝置連接時，D+D-上會有一個1.5K歐姆的電阻接地，確保這兩條線是在接地狀態。可支援至127個位址，也就是說，一個USB系統最多支援127個裝置，主機會週期性的探詢每個集線器以便得知裝置的連接狀態。以USB2.0而言，分為低速、全速及高速三個模式。低速傳輸率為1.5M/s，全速為12M/s，高速為480M/s。高速模式時，1秒 / 480M，每個bit的寬度約2.08ns。

## USB封包簡介

封包是組成USB傳輸的最小單位。一個Transaction通常由三個封包組成，但依傳輸型態而定，一個Transaction可能包含一個、兩個、三個封包：



圖一:USB封包架構概述

### Token封包：

每個Transaction以Token封包做起始。Token封包定義裝置、Endpoint數量，傳輸的方向。其中SOF (Start Of Frame) Token包含目前的frame數，而且會廣播 (broadcast) 給所有的full-speed裝置知道。SOF也是唯一一個不指定目標的Token。Token Packet 長度固定為4個Byte。

### Data封包：

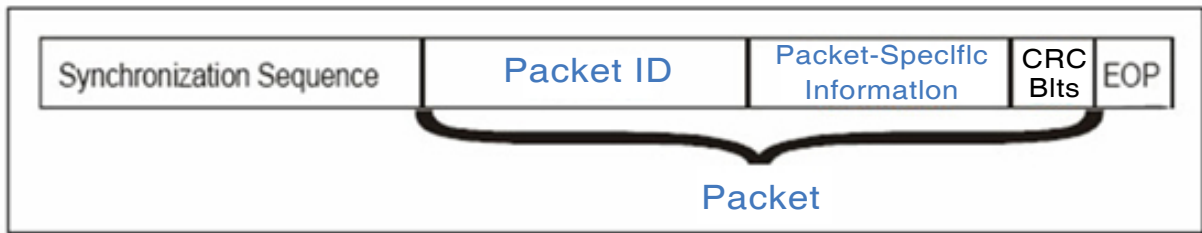
Data封包包含處理此動作的資料。一個Transaction中，Data封包最大的資料量為1023個Bytes，高速模式時可達1024個Bytes以上，其中Data0及Data1是兩個基本的資料封包，這些資料封包都是接續在Address之後，且Data0及Data1採取交互出現方式以達到同步與除錯的效果。另外在USB2.0當中更增加了Data2及MData資料封包，用於執行高速的即時傳輸(Isochronous Transfers)。

### Handshake封包：

除了即時型傳輸(Isochronous)之外，所有的傳輸都保證資料的傳遞正確。Handshake封包回應資料是否正確的被收到。若執行處理動作中發生錯誤，此處理動作將重新執行。

## 封包格式

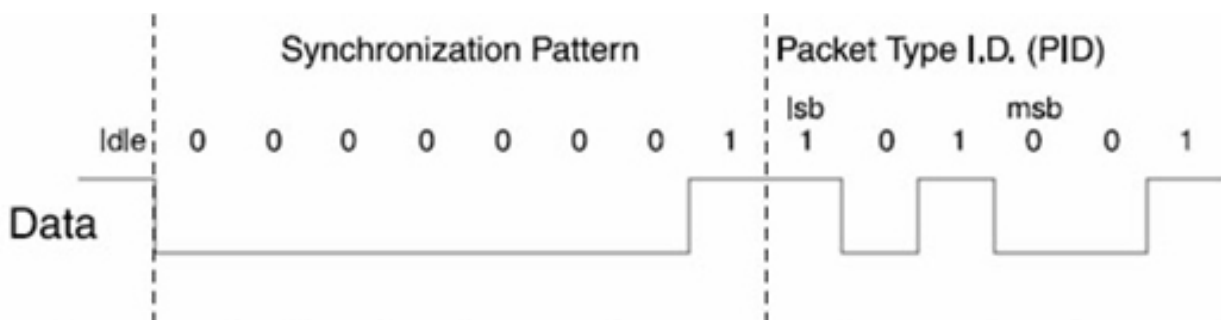
封包是執行所有處理動作的機制。圖二是USB封包的基本格式。封包緊跟在Synchronization sequence (SYNC, 同步列) 之後，Synchronization sequence 使USB裝置能跟封包中的位元速率同步。封包的Type由一個Packet ID來定義。Packet ID之後是隨著封包Type而定義的資訊，如Address或Data。最後，每個封包以一個CRC (Cyclic Redundancy Check) 做結尾。CRC是用來確認資料是否正確的傳輸。每個封包始使用EOP (End Of Paekct) 狀態來做辨識。



圖二:USB封包的基本格式

### Synchronization(同步序列) :

圖三是Synchronization sequence的波形。Synchronization sequence 由八個bit組成，數值是二進位的00000001，依照NRZI編碼原理，資料0代表訊號要轉換，所以前七個0之中的每個Bit都會在相對應的Bit時間中轉換，作用就好像提供了一組可用來同步的Clock。

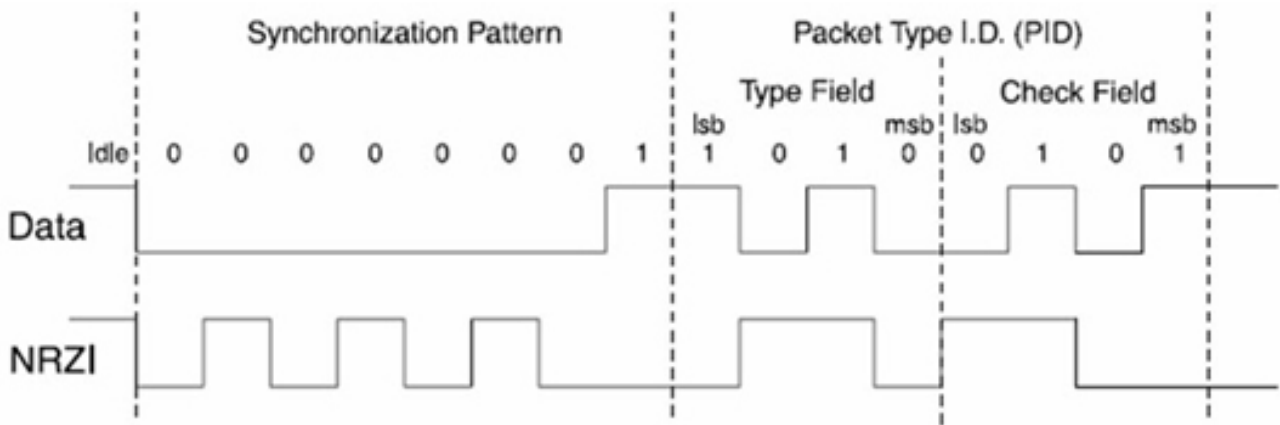


圖三: Synchronization sequence

## Packet ID

Packet ID定義封包的目的與內容。封包主要有四種型態：標誌（Token）、資料（Data）、交握（Hand-shake）及特殊。其中Token封包定義傳輸的型態；Data封包接在Token之後，將傳輸的資料輸出或輸入USB裝置中，Handshake則是提供傳輸成功或失敗的訊息，特殊PID是針對USB2.0特有的PID，包含了PING（檢查端點裝置是否能接受資料）、SPLIT（高頻寬的USB分割傳輸）、PRE（低頻寬的USB前導封包）、ERR（分割傳輸任務錯誤）。

Packet ID的訊號格式如圖四。Packet ID總共有8個Bit，前四個Bit代表ID (Type Field)，後4個Bit是檢查欄位（Check Field），其值固定是ID的反向。如ID為二進位的1001，則檢查欄位就是0110，合起來就是10010110。



圖四：Packet ID 訊號格式



## 封包種類

所有處理動作都是由一個Token封包開始。USB定義四種Token封包：SOF，IN，OUT，SETUP。圖五是四個封包的PID與功能描述。

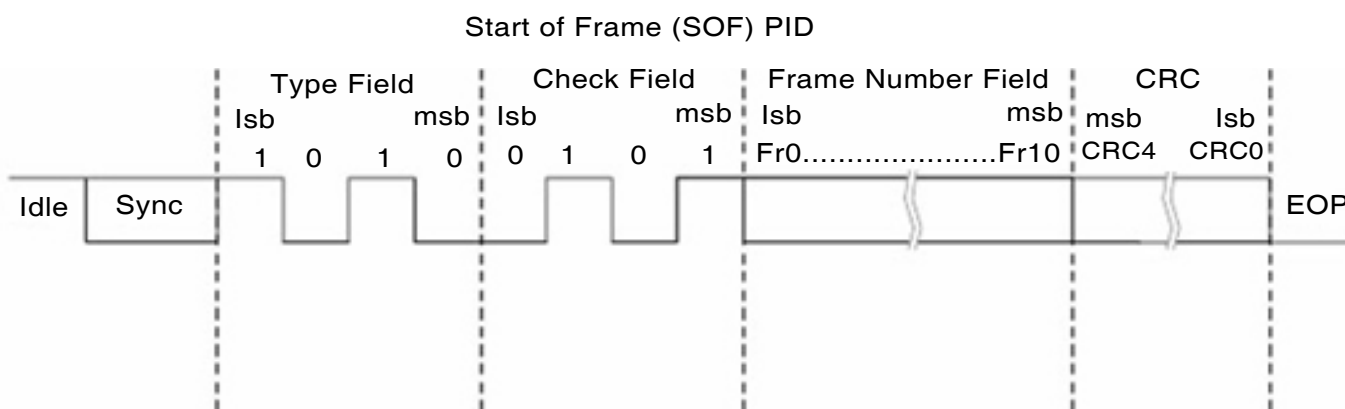
PID Type	PID Name	PID[3:0]	Description of Token Packet
Token	SOF	0101b	Contains start of frame (SOF) marker and frame number. The SOF token is used by isochronous endpoints to synchronize its transfers.
Token	SETUP	1101b	Contains USB device address and endpoint number. Transfer is from host to function for setting up a control endpoint(e.g.,configuration).
Token	OUT	0001b	Contains the USB device address and endpoint number. The transfer is from host to function.
Token	IN	1001b	Contains the USB device address and endpoint number. The transfer is from function to host.

圖五：四種Token封包狀態

**SOF封包：**

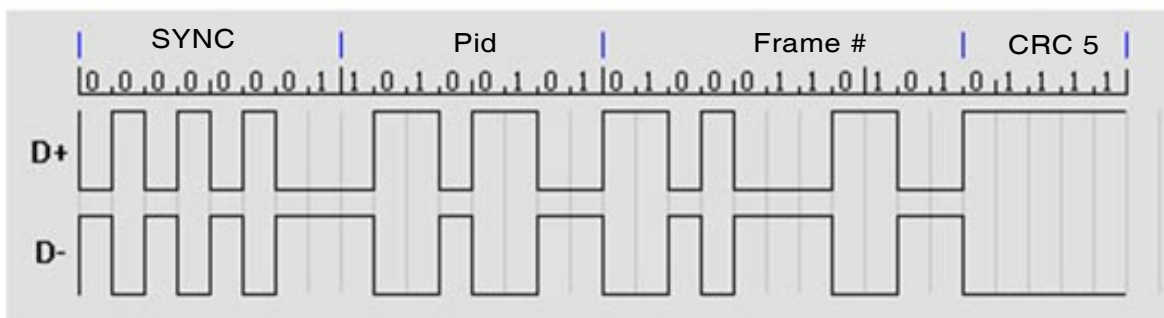
USB裝置用SOF封包來判斷訊框的起點。當每個frame開始的時候，SOF封包被送至每個全速的裝置。即時型傳輸裝置可經由SOF封包來達到同步傳輸的作用。低速裝置不支援即時型傳輸，所以SOF封包不會送至低速裝置。

如圖六所示，SOF封包包含一個長度為11 bit的frame number(訊框號碼)，接收器使用5 bit的CRC位元來確認frame number。SOF封包不含任何資料，其傳輸也不保證一定成功。當USB裝置對此SOF封包PID檢查碼錯誤時，裝置忽略此封包；當CRC檢查錯誤時，忽略此frame number。注意PID與Frame Number都是LSB在前面MSB在後面。



圖六: Start of Frame (SOF)

圖七是一個SOF封包的實際波形。PID前四個 bit “0101” 就是SOF封包。PD之後的是11 bit 的Frame Number，這個例子的Frame Number是10101100010，換算為十進位就是1378，也就是第1378個Frame。

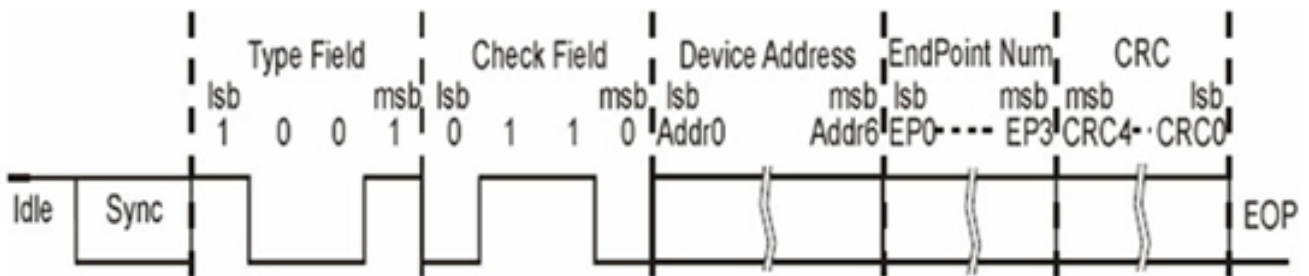


圖七: SOF 封包實際波形

**IN封包：**

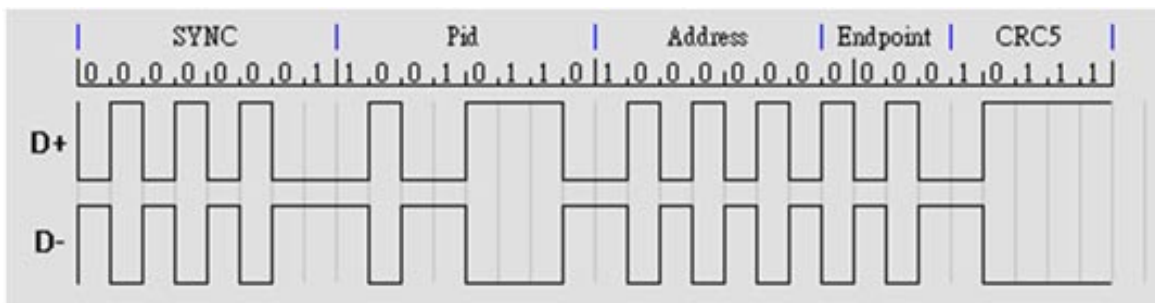
Token還有兩種封包，IN封包與OUT封包。所謂IN與OUT是指對於PC而言。IN封包是PC需要由裝置讀進來資料時使用的；OUT則是PC需要送出資料給裝置時使用的。IN封包可以使用在中斷型傳輸、巨量型傳輸、控制型傳輸的資料時期、即時型傳輸等傳輸。

圖八是IN封包格式示意圖與實際量測圖。如圖所示，除了基本的Type欄位與Check欄位之外，IN封包包含7 bit 的Device Address與4 bit 的End Point Number。執行IN封包動作時，首先PC送出一個IN封包，接著裝置回傳一個資料封包，最後PC送出一個交握封包給Device，表示已收到資料。但在即時型傳輸中，IN處理動作不使用交握封包。IN傳輸所能傳輸的資料量取決於傳輸型態。



圖八：SOFIN封包格式示意圖與實際量測圖

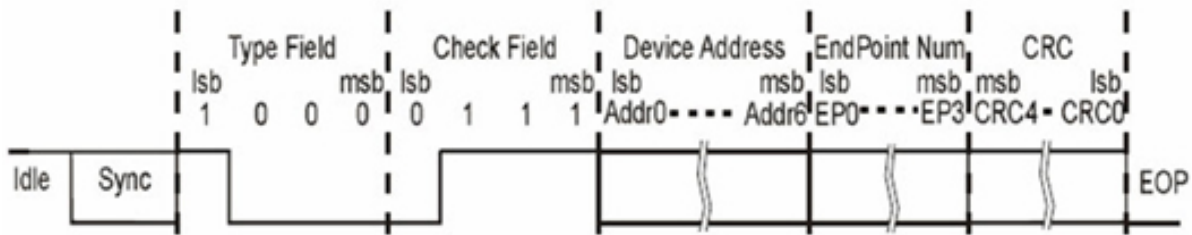
圖九是一個實際的IN封包。PID的前4 bit 為“1001”就是IN封包的ID。再來是7 bit的Address，這個例子的 Address是“1”。再來是4 bit的Endpoint，這個例子的Endpoint是”0”。最後是5 bit 的CRC。



圖九：IN封包實際波形

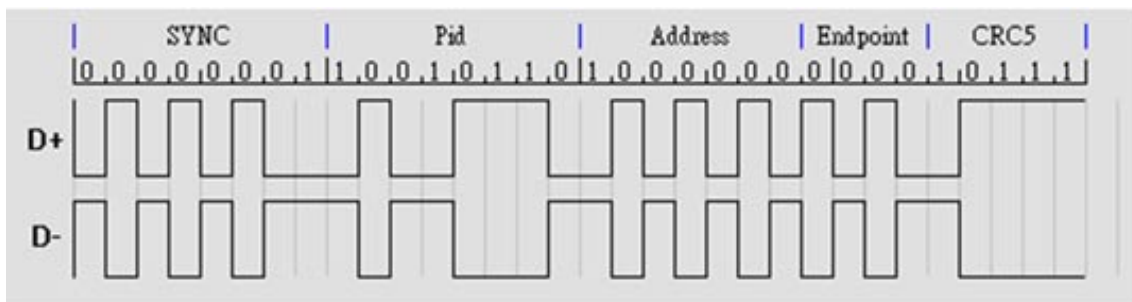
**OUT封包：**

當資料將被送出給某個Device時，系統會進行一個OUT傳輸。有三種傳輸型態用到OUT傳輸：巨量型傳輸、控制型傳輸的資料時期、即時型傳輸。OUT封包除了PID之外，其餘資料格式都與IN封包相同，一樣有7 bit 的Address，4 bit 的End point（如圖十所示）。



圖十:OUT封包格式示意圖與實際量測圖

圖十一是一個實際的OUT封包。PID為“0001”就是OUT封包。這個例子的Address是“1”。再來是4 bit 的Endpoint，這個例子的Endpoint是”0”。最後是5 bit 的CRC。

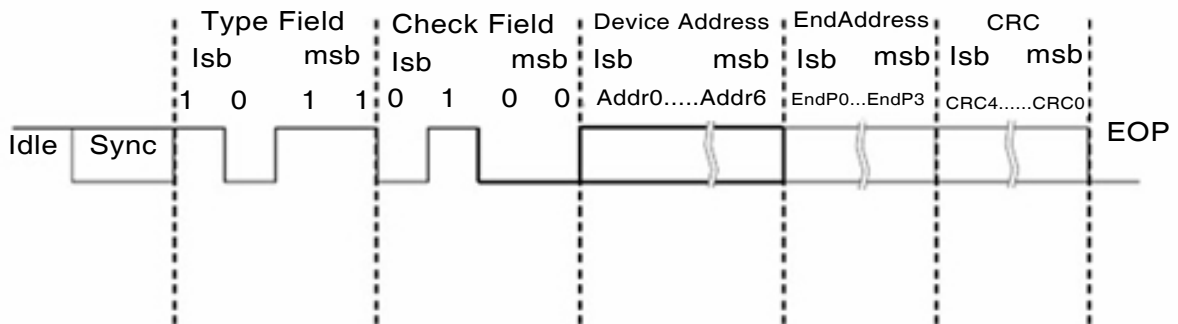


圖十一:OUT封包格式示意圖與實際量測圖



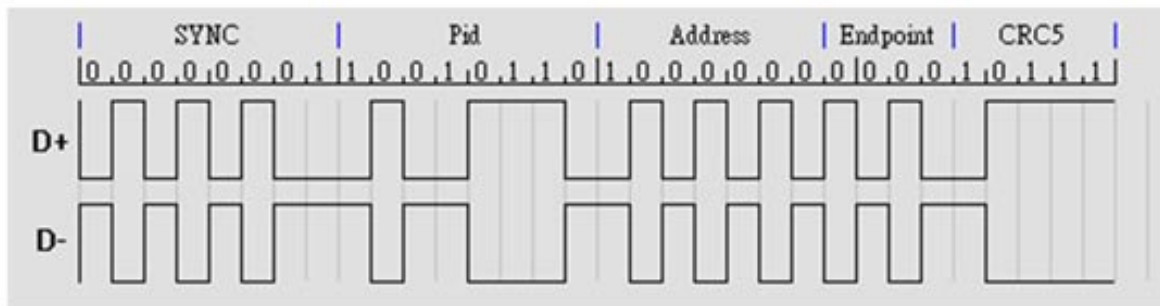
**SETUP封包：**

SETUP封包只有在控制型傳輸的設定階段使用。SETUP處理動作啟動一個控制型傳輸，並且定義為設定階段。SETUP處理動作在形式上類似一個OUT動作：SETUP封包後跟著一個Data0封包與一個回覆封包。SETUP封包的目的是傳送一個要求給Device執行。下圖是SETUP封包的格式。SETUP封包除了PID之外，其餘資料格式都與IN/OUT封包相同，一樣有7 bit 的Address，4 bit的Endpoint（如圖十二所示）。



圖十二：SETUP 封包格式示意圖與實際量測圖

圖十三是一個實際的SETUP封包。PID為“1101”就是SETUP封包。這個例子的Address是“1”。再來是4 bit 的Endpoint，這個例子的Endpoint是“0”。最後是5 bits的CRC。

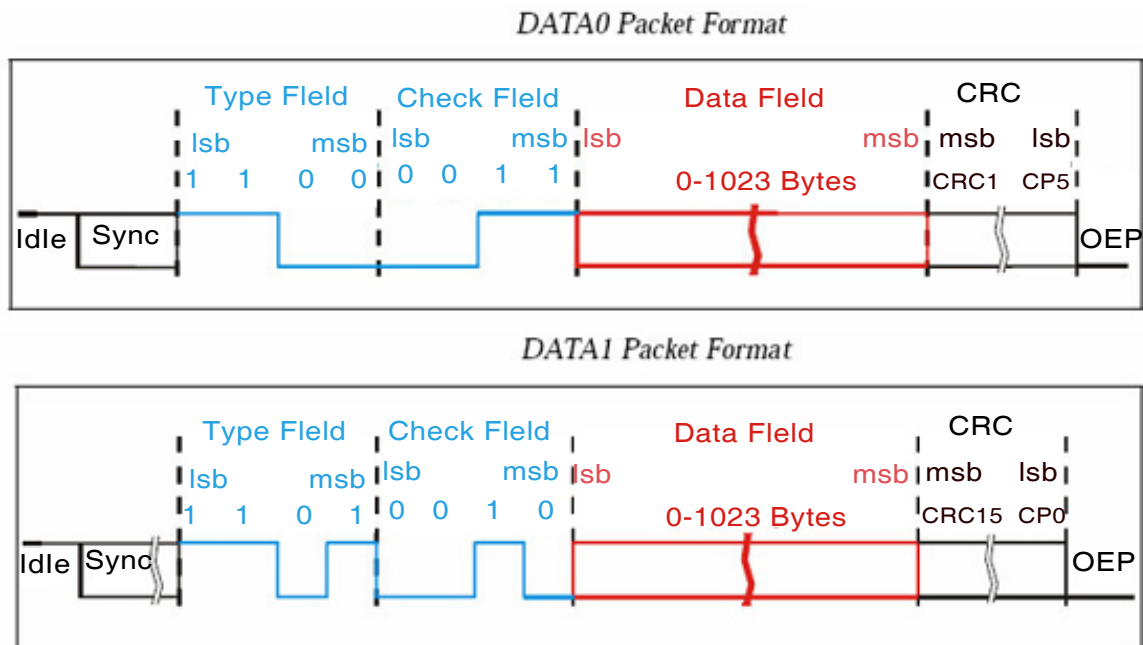


圖十三：SETUP 封包實際波形

**Data0與Data1封包：**

Data封包跟在IN、OUT、SETUP三種封包之後。IN封包之後的Data封包是將資料傳入PC；OUT與SETUP之後的封包是將資料傳出給Device。傳輸資料時，Data0封包與Data1封包是交互的使用，如果此次傳輸資料是用Data0封包，則下次傳輸資料就是使用Data1封包，反之亦然。這是為了用來當作一個同步機制使用。一個很大的資料，會被切成多個小封包來傳輸，此時Data0與Data1封包交互的使用就可以當作一個同步的機制。在高速模式底下還有Data2及MData兩種即時傳輸資料封包。

圖十四是Data0與Data1的封包格式。PID為“1100”就是Data0封包，PID為“1011”就是Data1封包。Data2及MData兩種即時傳輸資料封包的PID分別為0111及1111。



圖十四：Data0 與 Data1 的封包格式

## 交握封包(Handshake Packets)：

交握封包用來回報處理動作的完成狀況。交握封包有三種。

### ACK封包：

確認資料正確無誤的被接收。

### NAK封包：

告知主機現在目標裝置暫時無法接收或傳回資料。

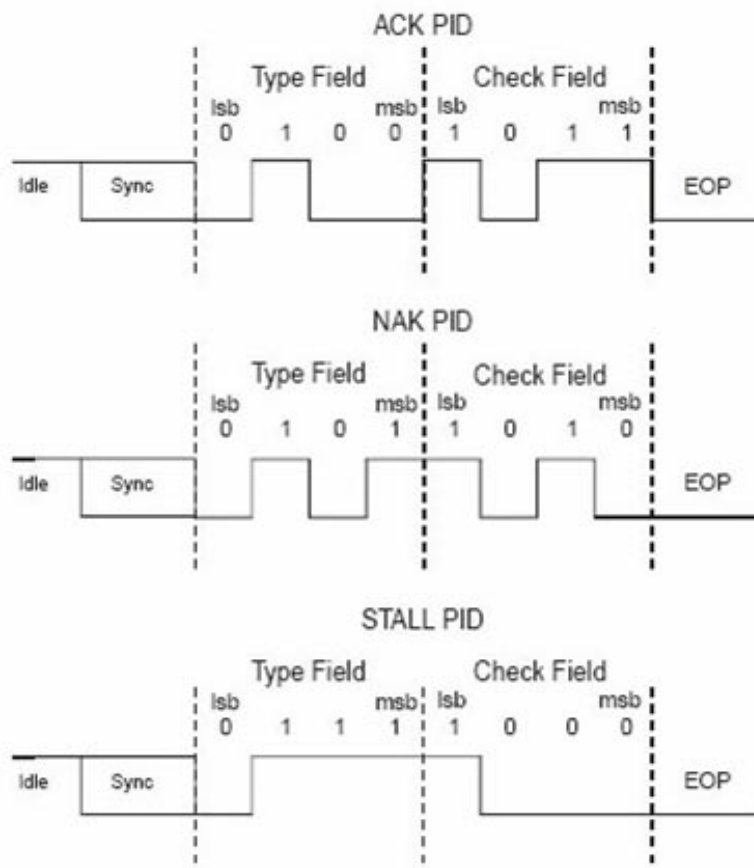
### STALL封包：

無法完成傳輸，且需要軟體介入以便使得裝置能從STALL狀態復原。

在USB2.0中還有一種NYET封包，表示資料尚未備妥無法傳輸。NYET的PID為0110。

圖十五是三種交握封包的格式圖。交握封包沒有資料，也沒有CRC，只有SYNC與PID。

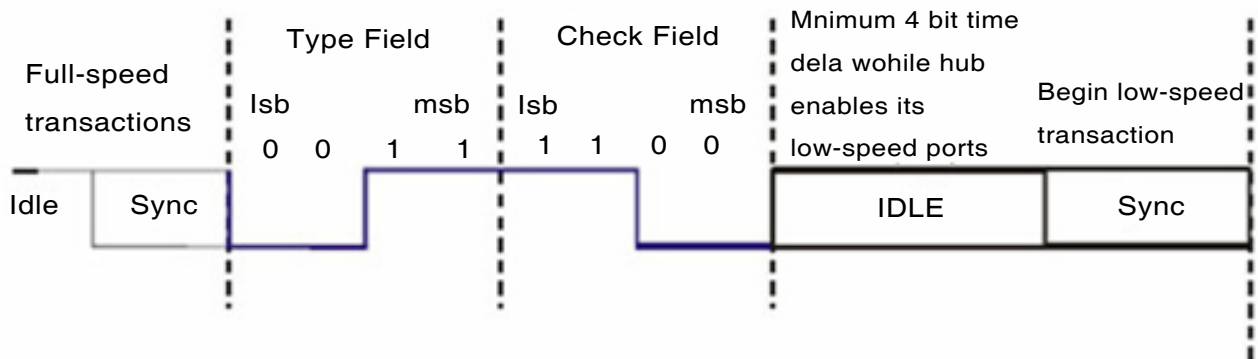
ACK的PID是0010，NAK的PID是1010，STALL的PID是1110。



圖十五：三種交握封包格式圖

**前導封包(Preamble Packet) :**

前導封包用在低速傳輸時。在傳送一個低速封包前，一個前導封包必須被送出，主機保證前導封包之後的封包是低速傳送。圖十六是前導封包示意圖。前導封包由一個SYNC與一個PID組成，PID是1100。有一點必須注意的是，前導封包不以EOP結尾。



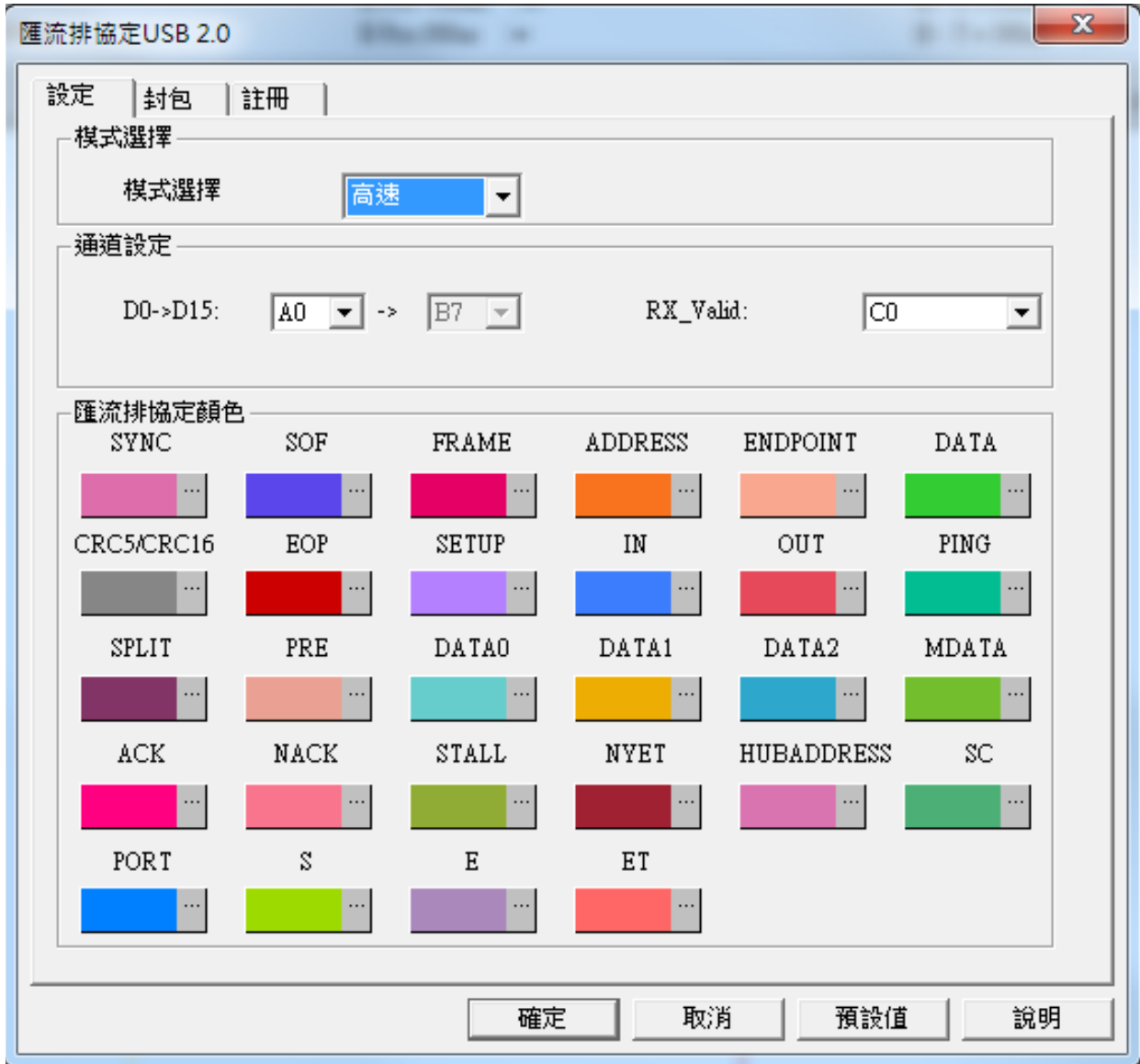
圖十六：前導封包示意圖

## USB2.0匯流排協定模組設定



圖十七: 孕龍科技USB2.0 Protocol Analyzer Module

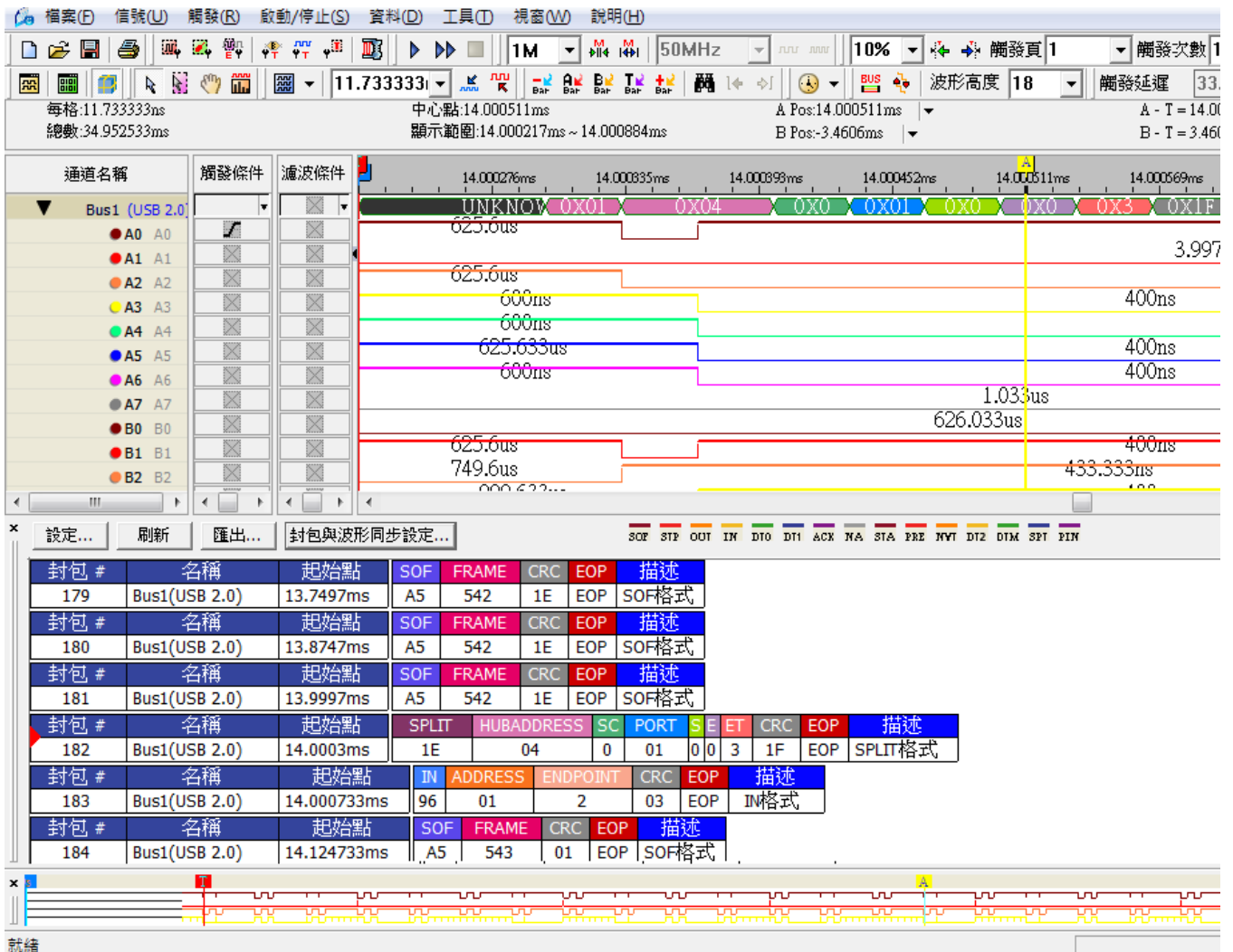
匯流排協定模組設定方便又快速，如圖十八所示使用者只需要依照連接說明將USB2.0硬體模組連接至邏輯分析儀，在USB2.0匯流排協定模組中設定高速模式，點下OK後就可以將USB2.0訊號解碼。



圖十八: USB2.0匯流排協定模組

## 串列協定封包列表

為更加方便分析以及提高效率，孕龍邏輯分析儀更加在新版軟體中加入串列協定封包列表功能，能夠將串列訊號採取封包段落顯示以直列方式把資料內容顯示出來，讓使用者清楚了解各封包先後出現的時序關係。圖十九為USB2.0訊號使用串列協定封包列表顯示的結果



圖十九: USB2.0訊號使用串列協定封包列表顯示結果

在圖十九中的封包列表，除了將各種USB2.0封包使用直列方式顯示外，在封包列表視窗上方更設置了過濾條件，依照USB PID規範可對已擷取封包進行篩選，更可在龐大的資料封包中快速進行分析。

 **總結**

匯流排協定常常被應用在嵌入式系統的設計中，但是若僅使用示波器分析這些訊號是非常花費時間的，孕龍的PC-BASE邏輯分析儀可以提供工程師們更強大的觸發功能、近百種的匯流排協定解碼模組及便利的資料搜尋，讓工程師在開發產品時更能夠得心應手，近期更推出了可與各大廠牌示波器進行堆疊，藉以同時測量數位及類比信號，快速解決電路開發中各種難題。

(孕龍科技網站[www.zeroplus.com.tw](http://www.zeroplus.com.tw))

編者 孕龍科技資深工程師Gary Hsu\_2011/11/15