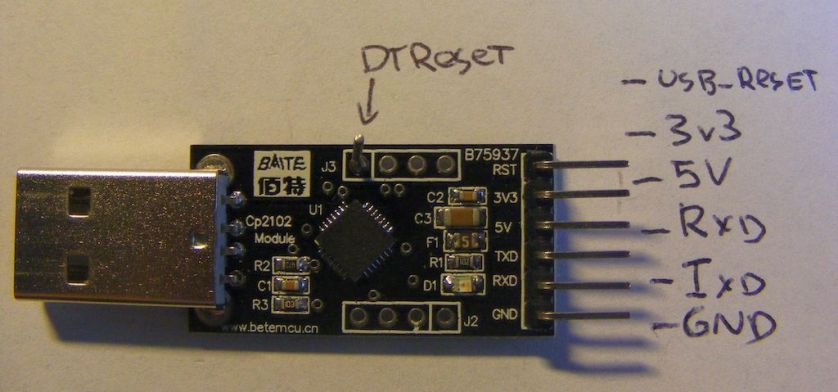
**【Maker電子學】認識UART通用非同步收發器**



USB to UART connector

在創客、自造者領域經常會聽到一個詞叫USB-to-UART（USB轉UART），USB就是大家熟知的USB埠、USB介面，但UART是什麼呢？這其實是有淵源歷史的。

UART全稱是Universal Asynchronous Receiver/Transmitter，這個詞中的第一個關鍵是Asynchronous（非同步），相對於這個詞的是Synchronous（同步）。

所謂同步，其實是介面中的資料傳輸線路，都依據同一條時脈線路的信號來動作，例如I2C介面、SPI介面，而所謂的非同步，就是每條資料傳輸線路有自己的傳輸步調，不倚賴獨立的時脈線路，或根本沒有獨立的時脈線路，通常是將時脈信號埋藏在自己的傳輸封包中，規律性的每隔一段時間發出。

同步傳輸有I2C、SPI等，而非同步傳輸的介面也有很多，例如RS-232、RS-422、RS-485等，不同的非同步傳輸介面有不同的取向，例如RS-232追求實現方式簡單低廉，RS-422追求較長距離的傳輸，RS-485追求較多的裝置連接數目等。

由於RS-232/422/485等的傳輸本質相同，因此微控制器晶片商想出一個方式，在微控制器上只提供多種非同步傳輸的共通本質收發能力，而不同介面的差異則由不同的外部電路與軟體程式來實現。

如此，一個收發器電路可以規劃成RS-232，可以規劃成RS-422，也可以規劃成RS-485等，完全由取得微控器晶片的電子工程師來決定，不過同時間一個收發器電路只能扮演三者其一，無法三者兼備，也因為可以規劃成多種非同步傳輸介面，所以稱為Universal。



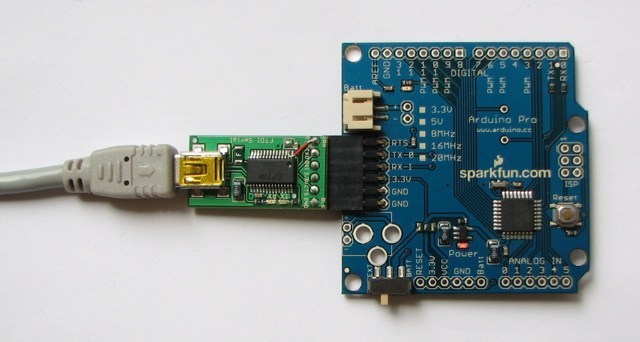
RS232連接埠

在上述的多種非同步串列傳輸介面中，其實個人電腦也取用其一，即取用了RS-232，在USB介面還沒出現前，個人電腦是用RS-232介面連接滑鼠或數據機的。RS-232的好處是簡單便宜，但傳輸率不高（早期最快為115.2kbps），且一個RS-232介面只能連接一個裝置，不像RS-422、RS-485可以同時連接多個裝置。

所以，一部個人電腦通常有兩個RS-232介面（也稱為COM埠，即通訊Communication；或稱串列埠Serial Port，因為當時的印表機埠為並列傳輸埠，串列成為一個相對詞），一個拿來接滑鼠，另一個空著不用，或空著的那個用來接數據機。

但是，USB介面發創後，個人電腦逐漸只提供USB埠，而廢除COM埠，但Arduino上的微控制器晶片沒有USB介面，只有UART介面（規劃成RS-232的一對一連接），因此個人電腦理論上無法連接Arduino。

為了讓PC能跟Arduino通訊傳輸，因此Arduino開發板上設置一個USB-to-UART晶片，該晶片可以把USB傳輸翻譯成UART傳輸，同時也能反向翻譯，使Arduino晶片能跟PC連通。



Arduino Pro透過USB to UART介面來對外連結或供電。

有了這個介面轉譯晶片後，再搭配上驅動程式，即可讓Windows將外接裝置認定成一個串列埠裝置，而Windows上原本支援COM埠的軟體（例如PuTTY），能夠用原本對COM埠的傳輸方式傳輸，但實際上卻轉成USB協定方式傳輸，一路傳到USB-to-UART晶片上時，才轉成原本的UART傳輸，並送入微控制器晶片內，反之亦然。

所以，有時候USB-to-UART也稱為USB-to-COM，Windows中的裝置管理員也是顯示COM埠，例如COM5、COM6等（用軟體模擬出的串列埠，硬體本質是USB），另外也有人稱為USB-to-TTL，其實也是類似意思，但TTL是指Transistor–Transistor Logic，是一種0、1邏輯信號準位的規範，嚴格而論在此用這個詞不太妥當，但也已約定成俗而慣用。

最後一提的是，有的微控制器晶片也允許規劃成同步與非同步收發電路，如此就稱為USART，S即同步，A則為前述的非同步，但同樣的，同一時間只能選擇同步或非同步其一，不能同時啟用。當然，有的晶片同時有2個或2個以上的UART，就可以同時各別規劃、同時各別運用。