

# 何整合 N1MM+WSJT-X(或 JTDX)及 Gridtracker 於 Log4OM 中

Rev1 Bruce / BV2KI (2023/10/25)

## 目錄

A、前言:	5
B、方法說明:	7
C、基本設定	8
C_1、N1MM+中的設定	8
C_1_1、開啟 N1MM+ :	8
C_1_2、開始進入 Config 配置作業;	8
C_1_3、進入 Configurer 配置總畫面	9
C_1_4、HardWare 硬件設定--單機版	12
C_1_5、Mode Control 模式控制設定	13
C_1_6、N1MM+跟 WSJT-X(或 JTDX)有關的設定	14
C_1_7、在 N1MM+環境下、如何打開 JT 模式的動作	20
C_1_8、N1MM+的音效設定	28
C_1_9、N1MM+環境下、如何關閉 JT 模式的作業?	29
C_2、N1MM+環境下、WSJT-X(或 JTDX)中的設定	31
C_2_1、WSJT-X(或 JTDX)中「一般 General」的設定:	31
C_2_2、WSJT-X(或 JTDX)中 Radio_1 的「機組」設定	33
C_2_3、WSJT-X(或 JTDX)中 Radio_2 的「機組」設定	35
C_2_4、F/H 模式的考量:	36
C_2_5、WSJT-X(或 JTDX)之「音效 Audio」的設定	43
C_2_6、WSJT-X(或 JTDX)中「進階 Advanced」的設定	44
C_2_7、WSJT-X(或 JTDX)中「推播 Reporting」的設定	46
C_3、四個程式[Gridtracker/Log4OM/WSJT-X(或 JTDX)/N1MM+]的關係	58
C_4、配合 N1MM+的基本設定版本	60
C_4_1、測試「N1MM+的基本設定版本」:	62
C_4_2、N1MM+的兩個機組實際作業紀錄	66
C_5、N1MM+環境下、配合 Gridtracker 版本	71
C_5_1、Gridtracker 的相關設定:	73
C_5_2、Gridtracker 的三個 UDP 參數設定	75
C_5_2、測試 Gridtracker 版本實作案例	79
C_5_3、我們再做一個測試	84
C_6、Log4OM 的配合設定	88
C_6_1、Log4OM 的 Connection 設定	88
C_6_2、Log4OM 的 ADIF Function 功能的設定:	91
C_6_3、如何確定 Log4OM 的設定是否成功?	94
C_6_4、Log4OM 的診斷視窗	97
C_6_4、KI Format 的解說:	98
C_6_5、Log4OM 的 Schedule(User Configuration)的設定	103

C_6_6、CAT interface 之 OmniRig 的設定 .....	111
C_7、電腦的音效設定及測試: .....	115
C_7_1、電腦本身內建音效設定 .....	115
C_7_2、音效測試 .....	120
C_8、操作常規 WSJT-X(或 JTDX)的 Radio 設定 .....	122
C_9、操作常規 WSJT-X(或 JTDX)的 Reporting 設定 .....	123
C_10、操作常規 N1MM+時的 ADIF Function 檔案移轉 .....	125
C_11、Log4OM 的單一欄位更新功能(KI Format 的套用) .....	128
D、配置檔與批次檔 .....	132
D_1、綜合設定(四程式之間的 UDP 及 ADIF 的設定) .....	132
D_2、多重配置 Multiple Configuration 之考量 .....	139
D_2_1、利用批次檔來做初級的運用 .....	139
D_2_2、N1MM+的多重配置 Multiple Configuration .....	144
D_3、批次檔的應用 .....	148
D_3_1、簡化的組合版批次檔 .....	148
D_3_2、綜合版的批次檔 .....	150
D_3_3、完整的組合版批次檔 .....	155
D_3_4、改變機組數的方法 .....	158
E、好用的參考細節 .....	159
E_1、善用 Log4OM 及 N1MM+內建之 Telnet .....	159
E_2、退出 WSJT-X(或 JTDX)時，順序錯誤關機的處理方法: .....	164
E_3、強制關閉應用程式的方法: .....	164
E_4、解決驅動程式失效的小秘訣: .....	165
E_5、單一機組的設定 .....	168
F、參考: .....	169
如何在文章內任意跳至各章節處(開啟「Document Map」)或 (ALT+<--)	169
如何在單一電腦上同時操作兩個以上的 WSJT 以及與其他軟體的整合設定 .....	169
N1MM+的解碼清單視窗 .....	169
批次檔文件內容(「3APP_WSJTDX_2R.BAT」不在 N1MM+之環境下運行) .....	170

如果你以 Word 閱讀本文、有使用到鍵盤 CTR+CLK 去看 Hyperlink 的超連結、且在閱畢該超連結之章節後，想退回超連結之段落原出處時，請在鍵盤上按「Alt+左箭頭」即可返回你先前按 CRL+CLK 之處，繼續閱讀；

如果以 Acrobat 閱讀器閱讀本文時，有使用到超連結功能、且在閱畢該超連結之章節後，想退回超連結之段落原出處時，請在鍵盤上按「Alt+左箭頭」，也可返回你先前之超連結原始之處。

Word 版本請注意：返回前。滑鼠千萬不要去放在任何「圖片」上，這會使該圖片做旋轉動作、而不是「回家」。

這種「自動返家」的功能只限於「連續的四次」。Why？不知道耶！可能要去找 ChatGpt AI 問一問!!

如果是連結至外部文章或 URL 處、且有另開啟其他應用程式時，Word 或 PDF 閱讀器中、這招可不管用；所以要做這類的超連結去閱讀時，請先將該超連結處的頁碼先記下來，以免「回不了家」！。

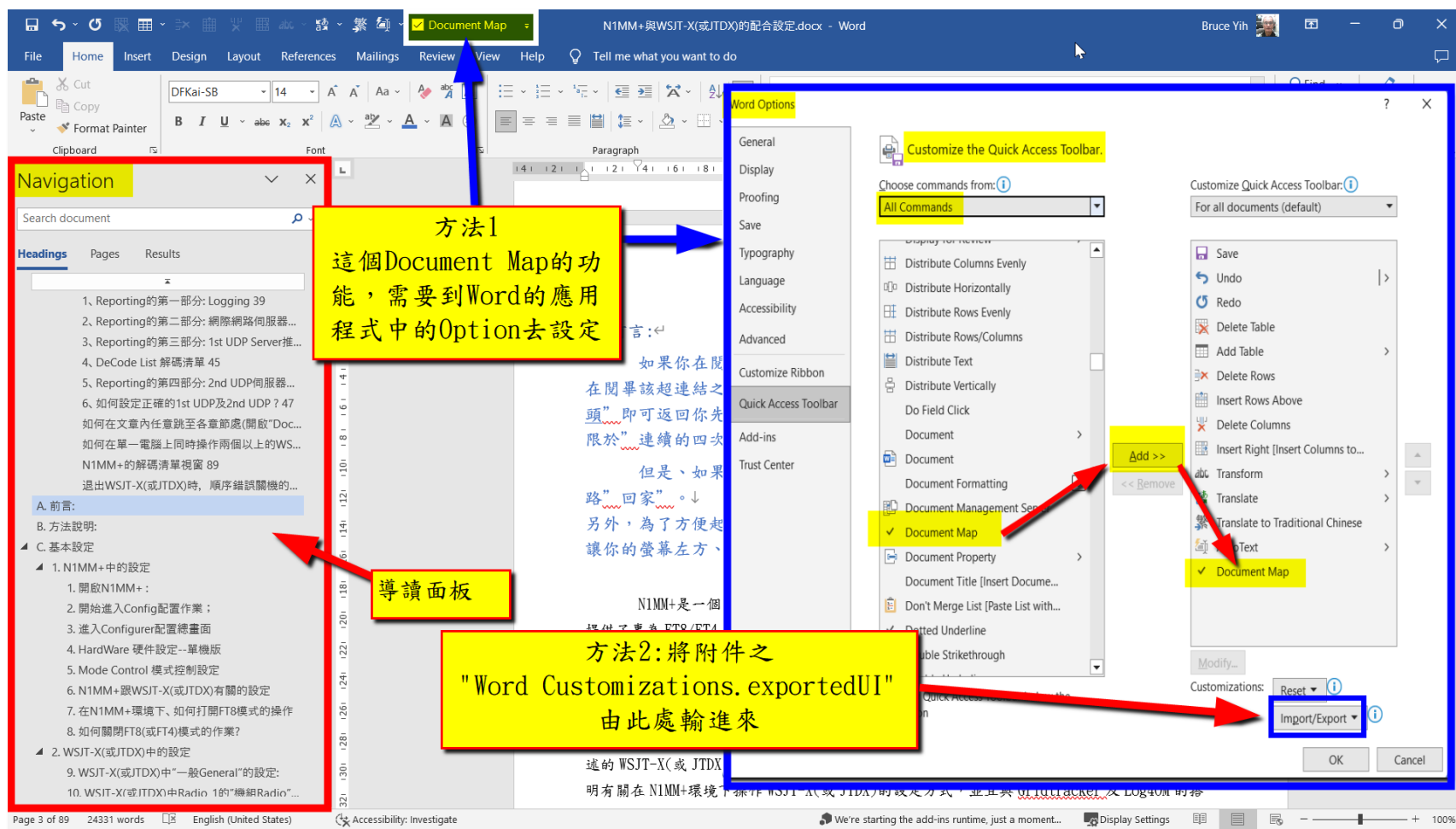
另外，為了方便起見，建議您將 Word 的 Document Map 啟動起來(見圖\_0)，這個啟動會讓你的螢幕左側方，出現 Navigation 導航面板。便於查詢及導讀。

或者你可以將附件的「Word Customizations.exportedUI」檔案、可直接在

Word→File→Options→Quick Accesses Toolbar

的「Import/Export」之選項中，調用此「Word Customizations.exportedUI」進來使用。

如果你更改了本文且須另存新檔，有關超連結的地址更新時，請參考這篇文章  
[2 Methods to Batch Change Multiple Hyperlinks' Addresses in Your Outlook Email](#)



圖\_0 WORD之導讀面板的樣子及設定



## A、前言：

N1MM+是一個以比賽為主的應用軟體，平常時候，你可以用 N1MM+來操作報務(CW)、話務(SSB)、數據(RTTY、PSK31)等模式。自 2017 年以來，微弱信號通訊模式(以下將此微弱信號通訊模式簡稱為「**JT** 模式」代表、它包含了比賽用的 FT4、對月通訊 EME 的 JT65、JT4、流星雨通訊的 MSK144 以及微弱信號回報 WSPR 等模式)、這個「**JT** 模式」的軟體開發與蓬勃應用，得到相當多的同好跟進與好評。目前的「**JT** 模式」發展出的軟體、據我所知的，有 WSJT-X、JTDX 及 MSHV 等三套系統。N1MM+也提供了專為 FT8/FT4 比賽的介面設計；在 N1MM+的環境下、採用 WSJT-X 或 JTDX(以下簡稱為 **JT**)兩個應用程式來當作主要的「子物件應用程式」；這類 **JT** 程式，於 QSO 完畢後會產生 ADIF 檔案、並有其自身的儲存之位置(不在 N1MM+環境下執行時、亦復會產生 ADIF 檔案、只是儲存於不同之位置而已)外，同時也會將 **JT** 程式產生的 ADIF 檔案儲存於在 N1MM+的 Database 中。上述的 MSHV 也是有同樣的有其儲存 ADIF 的位置。因此、就以 ADIF 資料儲存位置而言，操作 N1MM+或者 **JT** 或 MSHV 之 ADIF 時的儲存，都屬於「散置」狀態；對於處理 Log 時非常麻煩。我們如何將以上的所有程式產生 QSO 的 ADIF 檔案集中到一個「總舵主」的 Database 中，這就撰寫是本文的目的。

本文假定你已經曉得如何使用 **JT** 以及 N1MM+ Logger 的基本操作、且上述的 **JT** 及 N1MM+這兩款軟體都已經安裝在你的電腦上。本文最主要目的就是說明有關在 N1MM+環境下操作 **JT** 的設定方式，並且與 **Gridtracker** 及 **Log40M** 這兩個應用程式的搭配設定做一整體配合說明(其他的詳細配合說明，請參閱如何在單一電腦上同時操作兩個以上的 WSJT 以及與其他軟體的整合設定)。至於 N1MM+的基本設定，或許會跟本文所述、有些重複，但更進一步的其他模式的設定、如 RTTY 及 PSK 等，則不在此文討論範圍。有關 N1MM+環境下如何使用 **JT** 及其一般的基本訊息，請參閱這個有關 N1MM+對於 Decode Lists 的翻譯本。

由於 N1MM+所仰賴的「Decode List 解碼清單」，其功能是專為比賽目的之產物；該解碼清單只針對「喊 CQ」的電台做顯示，並集中於此解碼清單中；對於一般的常規作業做 QSO 的電台「們」的解碼信息、是不予顯示的。

換句話說:如果有一位您的「稀有電台」、正在 On Air 工作，這些信息是可以在 **JT** 的主視窗左半部分的 Band Activity 視窗中完全顯示出來，但在 N1MM+的「解碼清單」中，則是否定的!。如果你很想得到這位「稀有電台」的青睞，則可以直接在 **JT** 視窗中，按照一般 **JT** 的常規習慣動作，直接在該「稀有電台」的呼號處、左鍵點擊兩下(DBL\_CLK)，就可以去跟該「稀有電台」做 QSO；換句話說、這個解碼清單的功能看起來像似一個半吊子的設計。

N1MM+的這個「解碼清單」的設計還有一個功能:已經做過 QSO 的電台(指此次比賽的同一波段時),也不會在「解碼清單」中出現;以避免比賽時,會有「Dupe 重複」出現。所以,我認為 N1MM+的「解碼清單」的侷限目標之設計,對於比賽時,會有幫助,但是,對於平常的 QSO 作業,似乎頗多受限。

目前備受歡迎的 Gridtracker 應用軟體中的 Call Roster(呼號名冊),有著與「解碼清單」一樣的功能,而且 **Call Roster** 之其他功能更勝於「解碼清單」。所以,如何將 Gridtracker 的 Call Roster(呼號名冊)取代 N1MM+中的「解碼清單」,而且互不相影響本文所述的四個應用軟體(Gridtracker、Log4OM、WSJT-X(或 JTDX 或 MSV)及 N1MM+)之間的基本功能,這就是撰寫本文的另一個目的。

## B、方法說明：

N1MM+，可以提供 S02R(單人雙機)的操作，本文就以 S02R 操作模式作為佈局。當然，你如果並無操作 S02R 的計畫，也可以將本文所述的設定中，將不想使用的機組，「關閉不使用」即可(或者你可以參見：[單一機組在 N1MM+WSJT-X\(或 JTDX\)及 Gridtracker 於 Log40M 中的設定](#))。因此，為了配合操作 S02R 畫面的設定，本文將以圖片方式做說明，讓你跟著每一個步驟去做，應該可以輕鬆達成。

本文係採用 IC-7300(Radio\_1)及 IC-9700(Radio\_2)兩部機組為主；且以 WSJT-X(或 JTDX)作為操作「**JT** 模式」為說明依據(如果你是 JTDX 愛用者，請依照本文敘及 WSJT\_X 的有關名詞、功能等，自行調整相關之設定。以下均同)。最後將會以[批次檔方式](#)來進行開啟 N1MM+以及其他應用軟體的配合應用。

應用本文的方式時，由於畫面的操作務實起見，我是採用兩個監視器來做操作；這樣才有第二個螢幕供給 Gridtracker 以及 Log40M 這兩款應用軟體來使用(該第二螢幕，或許還可以有剩餘的空間，來讓其他的應用軟體做重疊的操作)。如果你暫時想先將 N1MM+如何操作 **JT** 模式的方法弄熟後，再來追加操作 Gridtracker 及 Log40M 這兩款應用軟體，也是可以的。

有關本文的撰寫大綱如次：

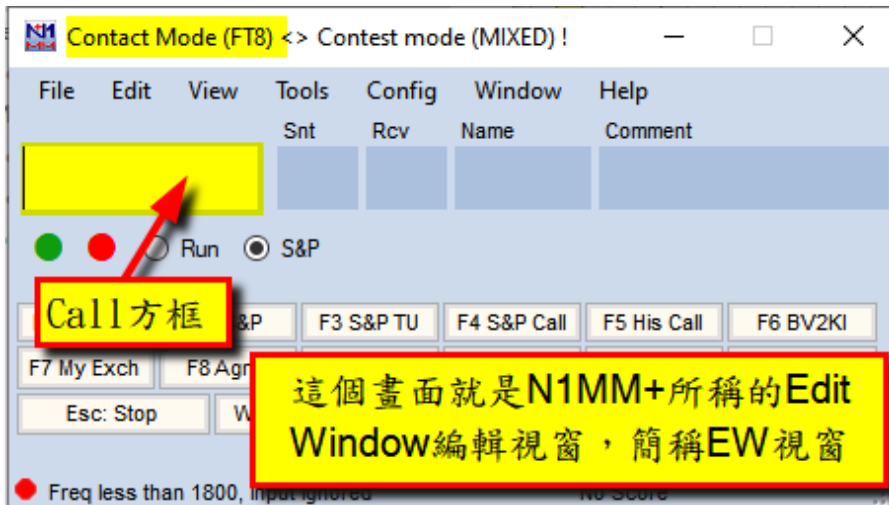
- 1、介紹 N1MM+的設定(僅與 **JT** 模式有關者，)；
- 2、介紹 **JT** 的所有設定(JTDX 者，請比照辦理)；
- 3、N1MM+的原始版及配合 Gridtracker 的改良版；
- 4、Log40M 及 Gridtracker 的相關設定(ADIF 檔案儲存及 Cluster 的相互參考)
- 5、測試結果。
- 6、結論:分別將 WSJT-X 及 JTDX 以圖解流程方式去做設定。
- 7、快速啟動：如果你看完本文後，想要馬上行動，請[直接到此動手](#)(九個批次檔菜單)。
- 8、有關「MSHV」者請另參考：[單一機組在 N1MM+WSJT-X\(或 JTDX\)及 Gridtracker 於 Log40M 中的設定](#)

## C、基本設定

### C\_1、N1MM+中的設定

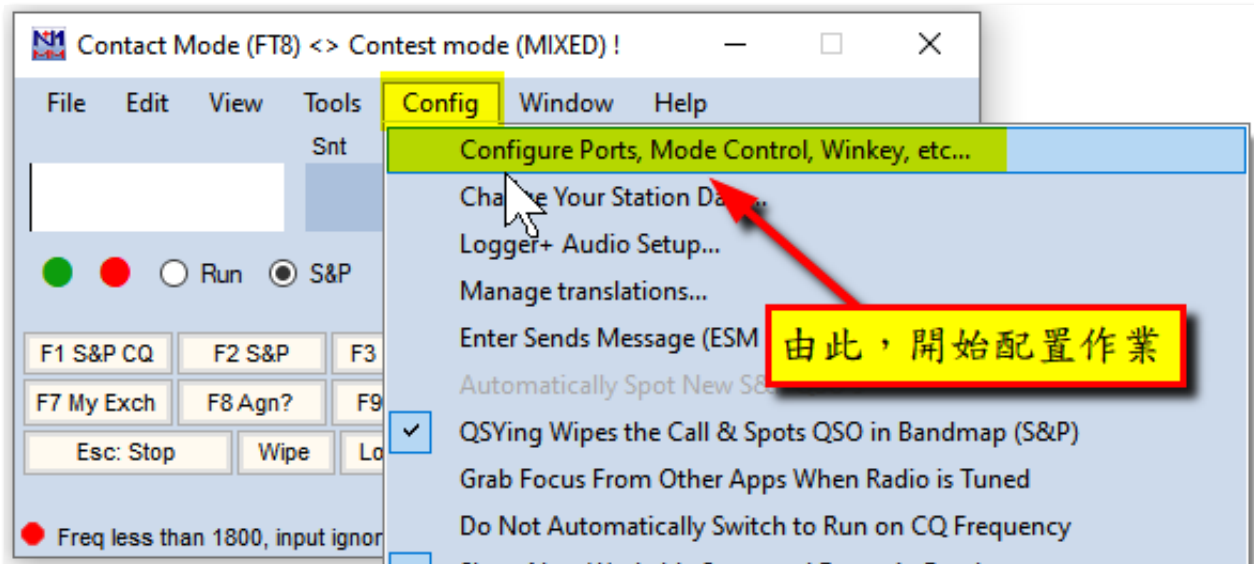
#### C\_1\_1、開啟 N1MM+：

請將主視窗 Edit Windows(簡稱 **EW**)尺寸大小，調成如圖\_1(在 EW 主視窗上空白處、按右鍵可以去多種選項的調整)；



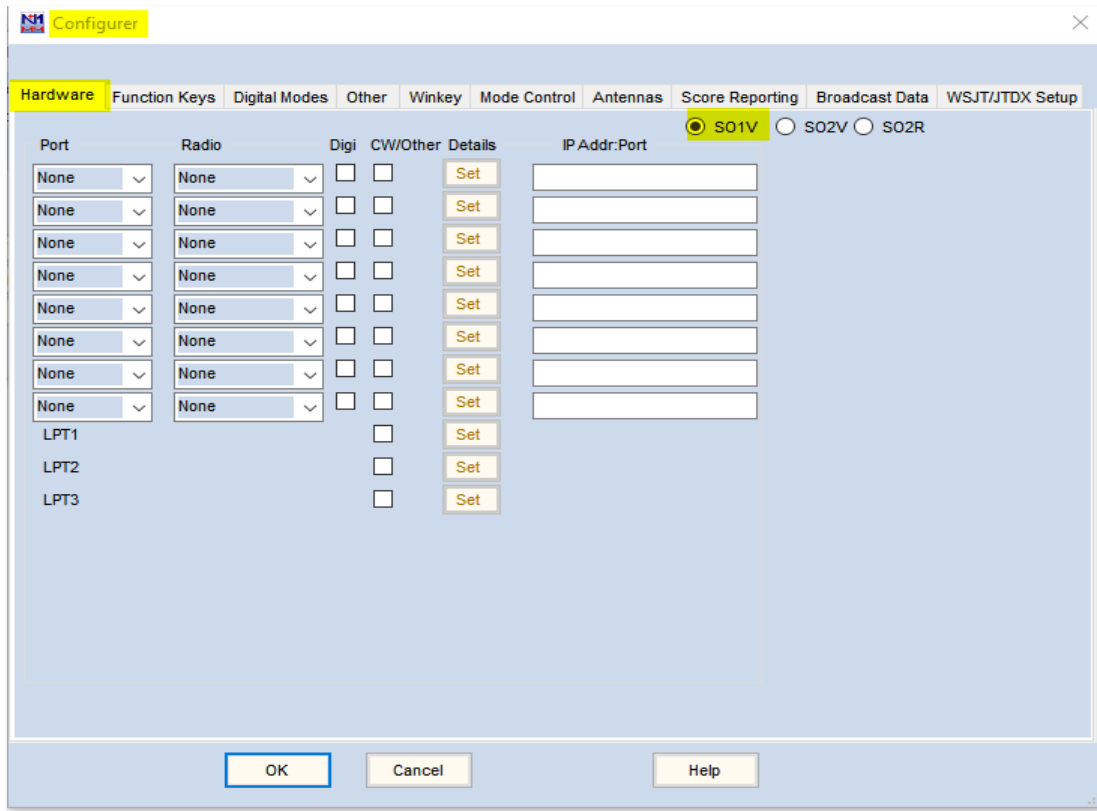
圖\_1 N1MM+主視窗介面

#### C\_1\_2、開始進入 Config 配置作業；



圖\_2 開始進入 N1MM+\_Configurer 作業

### C\_1\_3、進入 Configurer 配置總畫面



圖\_3 N1MM+的 Configurer 總畫面

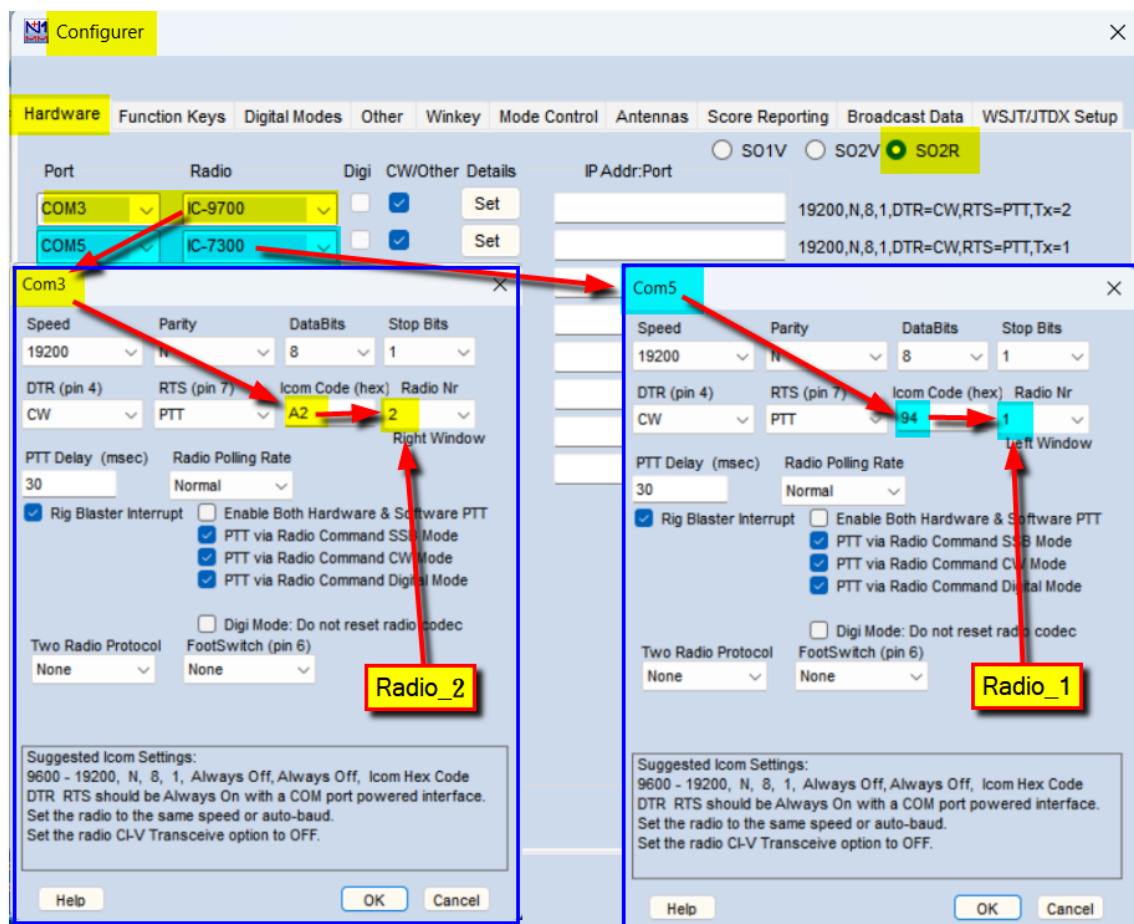
3-1、圖\_3，這個畫面是「選項卡 Configurer→硬件 Hardware」的 S01V(單人單機組)的設定初始畫面；圖\_4 則是先將圖\_3 墊在底部當成「底圖」，然後分別在圖\_3 中的表格填寫資料、並且到 Detail 欄位下的「SET」處、左鍵按下，就會跳出各相關的細部設定之視窗；再將這兩個細部設定視窗以貼圖方式，與圖\_3 重疊後就產生如圖\_4 的樣子。

3-2、機組使用的 COM port 來源，請先在電腦的「裝置管理員 Device Manager」(圖\_126)中查詢。目前、我這裡的 IC-9700 的 COM Port=COM3；而 IC-7300 者=COM5。各該機種的細部資料，也請查詢該機組技術資料後，一一填入如圖\_4 中的細部設定視窗中。

3-3、由於 N1MM+的兩個機組安排順序是由左至右排列成 Radio\_1 及 Radio\_2，是不能改變的順序；但是、是可以按照個人習慣去組合。我則是以常用的 IC-7300(HF)當作主要機組 Radio\_1，放在螢幕之右側；不常使用到或傳播較少的 IC-9700(VHF)放在螢幕左邊 Radio\_2。而我的第二個監視器是放在我電腦主監視器的左邊，如此之安排，可以讓 Gridtracker 應用軟體中經常用得到的呼號名冊(Call Roster)、將之放在 Radio\_2 的螢幕重疊位置(請參見圖\_57)；這樣子，就可以比較方便處理螢幕空間較為壅擠的困

擾。

你也可以將本設定的 Radio\_1 設為 IC-9700(右半部)，Radio\_2(左半部)設為 IC-7300 的配置。如果你有第二個監視器時，那就將第二監視器放在主監視器的右側，呼號名冊 (Call Roster)也是放在 Radio\_2 的螢幕重疊位置。這個放在右側的選項，你就可以作為 S01R 的模式(只有 Radio\_1=IC7300)，如此、呼號名冊(Call Roster)就完全佔用 Radio\_2 的(右半部)螢幕空間。這樣子的安排，你可以將圖\_4 中的 Radio\_1 及 Radio\_2 對調即可。



圖\_4 N1MM+的雙機板 S02R 硬體的設定

3-4、在圖\_4 上，勾選 S02R(右端，著黃色者)

3-5、在 Port 欄位裡，填入 IC-9700 的 COM Port=COM3 並在 Radio 欄裡選出 IC-9700

3-6、在 Port 欄位裡，填入 IC-7300 的 COM Port=COM5 並在 Radio 欄裡選出 IC-7300

3-7、CW/other 都勾選；Dig 欄位沒顯示，就不去理會它。

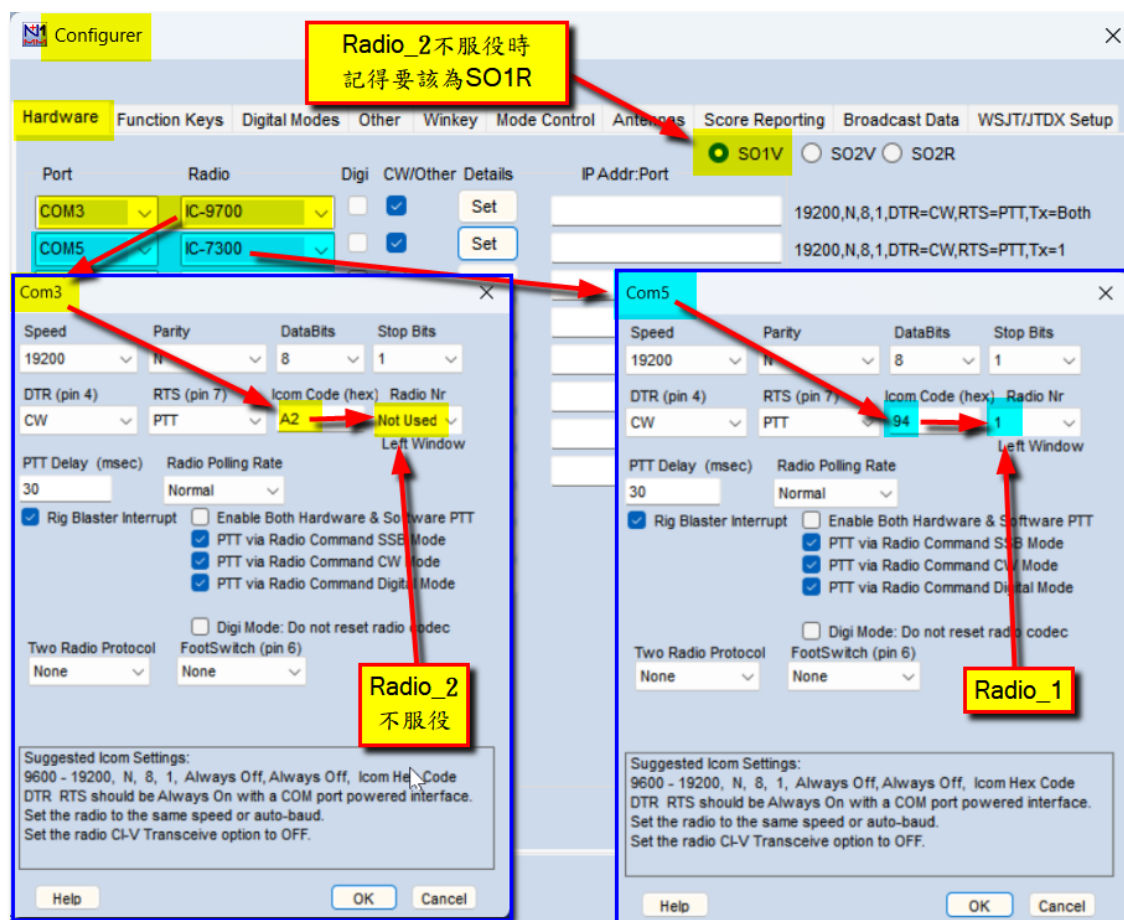
3-8、分別去按 Detail 欄位下的「SET」鍵，會出現相對機組的細部設定視窗；如圖\_4 的下半部顯示。

**後記:**有關 COM Port 的 Speed 的數值：圖\_4 中、之前 Speed 填寫為 9600；這是跟隨著十年前的習慣使然；由於每個機種的 CIV 與當時電腦 CPU 之速度有關、例如：IC-275 的老機器匹配 20 年前的 CPU，非用 4800 不可。由於近年來 CPU 之速度提升神速，取用各機種的默認最高值已不成問題，請參閱這篇[有關建立 CIV 連結的文章](#)。因此、依據此篇文章的建議我就採用 19200 bps 來做設定。

其餘部分，請就各個機組的出廠設定資料分別填入該相對的視窗中；並請依照去做「勾選」動作(請注意著色部分：**黃色**及**青色**，不要漏掉！)



## C\_1\_4、HardWare 硬體設定--單機版



圖\_5 N1MM+的單機板 SO1R 硬體的設定

圖\_5、是將 Radio\_2 設定為「NOT Used」，作為單機使用的設定範例。

如果你不需要使用 Radio\_2 時，那就去將 Radio\_2 設定成「Not Used」即可。如果，得不到預期結果，或「關」錯了 Radio，那就試著去將另外一個 Radio 完全不做設定，或許會比較單純。當然，SO2R 也要改為 SO1V，否則會出現不可預知之結果。

接下來 N1MM+跟 JT 模式有關的配置設定，就是圖\_3 中的選項卡 Configurer→Mode Control；如圖\_6：

## C\_1\_5、Mode Control 模式控制設定



圖\_6 N1MM+的 Configurer 之 Mode Control

圖\_6 中，Mode Control 的選項牽涉到各種廠牌的設計以及採用不同的模式來驅動數據模式。圖\_6 黃色的註解歸納，是遵從 N1MM+手冊上提供的一些可能的設定。是否有效用？尚待各方大德去試著選用看看！

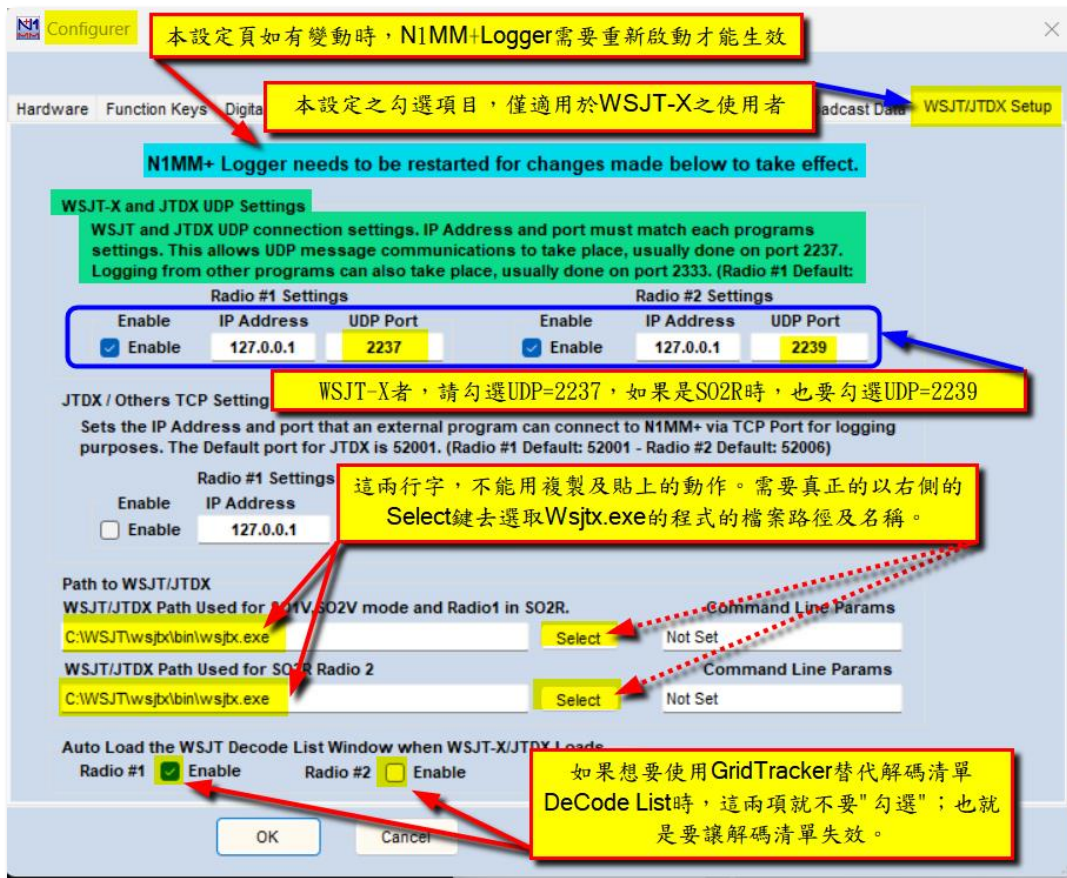
據我的試作，IC-9700 與 IC-7300 設定成「AFSK-R」是可行的。

IC-7100 及 IC-7410 設定成「USB」（屬於舊機種）。

KX3 設定成 PSK。

圖\_6 中，Mode sent to radio 之右側的 Radio 2/VFOB 設定也是選用與 Radio 1/VFOA 一樣者即可。

## C\_1\_6、N1MM+跟 WSJT-X(或 JTDX)有關的設定



圖\_7 N1MM+有關 WSJT-X 的 IP 設定說明(本圖僅適用於 WSJT-X)

有關圖\_7 內著 **綠色** 的意思如下：

### WSJT-X及JTDX的UDP設定

WSJT與JTDX的UDP連結設定，在各個應用程式之間都需要相互以同一個IP位址做搭配。如此才能觸發彼此之間的UDP信息傳遞的功能；一般UDP推播的埠位址設定為2237。而、為了讓其他的應用程式得以在N1MM+做Log登錄動作，一般是以2333作為UDP埠位址。

從這段話以及該說明文字下的選項勾選列看來，Radio\_1 的推播 UDP 默認埠=「2237」；Radio\_2 的推播 UDP 默認埠=「2239」；這兩個默認值是作為 N1MM+的連結通道(Connecting)。換句話說，UDP=2237 是推播給**解碼清單\_1**(Decode List 1)來使用的。而 UDP=2239 是推播給**解碼清單\_2**(Decode List 2)來使用的。這兩組號碼是 N1MM+特別設計給 WSJT-X 的兩把鑰匙；如果改變了這兩組號碼，解碼清單\_1 及解碼清單\_2 就無法顯示 **JT** 的所有解碼資訊於此二個解碼清單裡，這兩個解碼清單的所有功能也就無法運作，此種情形，正如同你正在操作單機版的 **JT** 一樣、沒有了 N1MM+提供的特殊功能。除此之外，由於你改變了

UDP=2237 或 UDP=2239 時，就等於斬斷了「解碼清單」的這個橋樑，解碼清單的另一個「Logging 登錄」的功能也跟著失效；換句話說，**JT** 產生的 ADIF 檔案，無法透過該「橋接」功能，讓 N1MM+ 無法達成「Logging 登錄目的」。

圖\_7 中的所有著黃色者，為了簡化起見、我建議不要去變動更改。N1MM+ 設定之 UDP Port 2237 及 2239 會另外產生 UDP=2240 及 UDP=2241 的虛擬 UDP Port(這是執行 **Cports.exe** 時發現的現象)。在其他的應用軟體，如 Gridtracker APP 或 Log4OM APP 等，最好避開採用到 2237, 2239, 2240 及 2241 這四個屬於 N1MM+ 之內部默認 IP 位址，以免往後要做修改時會增加的困擾。

因此、無論如何，圖\_7 的 UDP=2237 或 UDP=2239(或者是使用 JTDX 者，UDP=52001 及 UDP=52006)這兩組選項都給予打勾，作為基本上可以讓你用來操作 N1MM+ 所提供的解碼清單之功能(含 Logging 登錄)。

2333 是一個由 N1MM+ 接收其他應用軟體產生 QSO 之 ADIF 檔案時的默認推播 UDP 位址；我們可以在 Log4OM APP(或其他類似於 Log4OM 的 APP)的 Connection 的 UDP OutBound 設定項目中、設定一個專門用來讓 N1MM+ 作為接收之 UDP=2333 的專用位址的設定指令，讓其他的 APP 產生的每一個完成後的 QSO 資料 ADIF 檔案、藉由 UDP=2333 的通知(對等窗口)來讓 N1MM+ 接收 ADIF 檔案。

**後記：**「MSHV APP」的 UDP 設為「2333」時，「MSHV APP」的每個 QSO 產生之 ADIF 是可以儲存到 N1MM+ 的 Database 中的。

請注意，前一句講的完成後的 QSO 資料 ADIF 檔案，也是從 **JT** 所產生的，但是儲存至 N1MM+ 的 Database 後，這兩個檔案內容可能有差異的；因為 N1MM+ 還會在 **JT** 經由「解碼清單」傳送過來的資料中、會去添加或減少一些 N1MM+ 程式所需要的其他計算或控制內容，這個增、減動作，根據 Log4OM APP 之術語，稱之為資料傳畢更新「DB\_Updated」。

**這個 UDP=2333 是 N1MM+ 默認的接收埠**，市面上許多 Logger APP 的設計者，都曉得 N1MM+ 這位老大哥已經規劃過此一 UDP=2333 值，幾乎是一個公認的 ADIF 推播 UDP 默認值；他們在各 APP 之間設計 UDP 時都會予以尊重的不去重複採用此一數值。在 **JT** APP 也不例外、有關此項 **Second UDP** 的設定(如圖\_31 中的「**第四部分**」)，就明白的告訴你不建議你去勾選 UDP=2333 的選項；這是 **JT** 設計時，係以單機版為主的考量；雖然 WSJT-X 自有其本身 LOGGING 登錄的地方；如果是想要讓其他的 APP 去取得 **JT** 產生的 QSO 資訊時，你就在圖\_31 的第四部分去指定一個「推播 UDP」，這個 UDP 值，隨你高興，你寫「2033」也是可行的，



如果我們在 **Setting**→**Reporting**→**Second** UDP=2333 改為 UDP=2033，時，其他的接收 APP(例如 Log4OM)的設定中，去設一個 UDP 也等於「2033」，這樣子你就可以直接去接收來自於 **JT** 傳過來的 ADIF 檔案。

為了配合 N1MM+默認的 Radio\_1 的 2237 及 Radio\_2 的 2239 值，上述的 Second UDP、我就可以將 **JT** 設定成 2237 及 2239(見圖\_32 的青色數值)，用以作為解碼清單與 N1MM+ Logging 時的「敲門磚」。一旦有了此兩把鑰匙，N1MM+就可由 **JT** 之通告，去做其相對應的解碼、完成 QSO 後的通知、再藉由 UDP=2237 的虛擬 UDP=2240 或 UDP=2239 的虛擬 UDP=2241 的通告，讓 N1MM+去做 ADIF 的檔案儲存動作。

換句話說：**JT** 與 N1MM+之間、要透過 UDP=2237 或 UDP=2239 的建立起 DeCode List 來做解碼分析動作，而 DeCode List 與 N1MM+之間則係透過 UDP=2240 或 UDP=2241 來做 ADIF 檔案的儲存動作。

至於你是否有其他的功能替代選項，則不會受到圖\_7 之設定限制；功能替代選項(例如欲採用 Gridtracker 之 Call Roster 來取代 N1MM+的解碼清單)之設定，是在另一個地方去做指定的、與圖\_7 毫無關係。

請注意：圖\_7 最底部 Auto Load WSJT Decode List when WSJT-X/JTDX Loads 的勾選選項是指：當 **JT** 被叫進到 N1MM+環境下工作時，N1MM+會各自產生一個解碼清單(Decode List)給 Radio\_1 或 Radio\_2 使用。這個解碼清單的功能就等同於 Gridtracker APP 的 Call Roster(呼號名冊)；這個呼號名冊的功能遠超過 N1MM+所產生的解碼清單(請參考 DeCode List 解碼清單)；因此，本文會就講解 N1MM+原始設計採用的解碼清單方式作為基本設定版本；往後則再以 Gridtracker 如何以 Call Roster(呼號名冊)來替代前者；後者我則稱之為 Gridtracker 版本。所以，在圖\_7 中的最底下的解說意思就是指：

勾選時，表示採用 Decode List 解碼清單之功能；

不勾選時，表示採用 Call Roster 呼號名冊之功能(或不需要 Decode List 功能)。

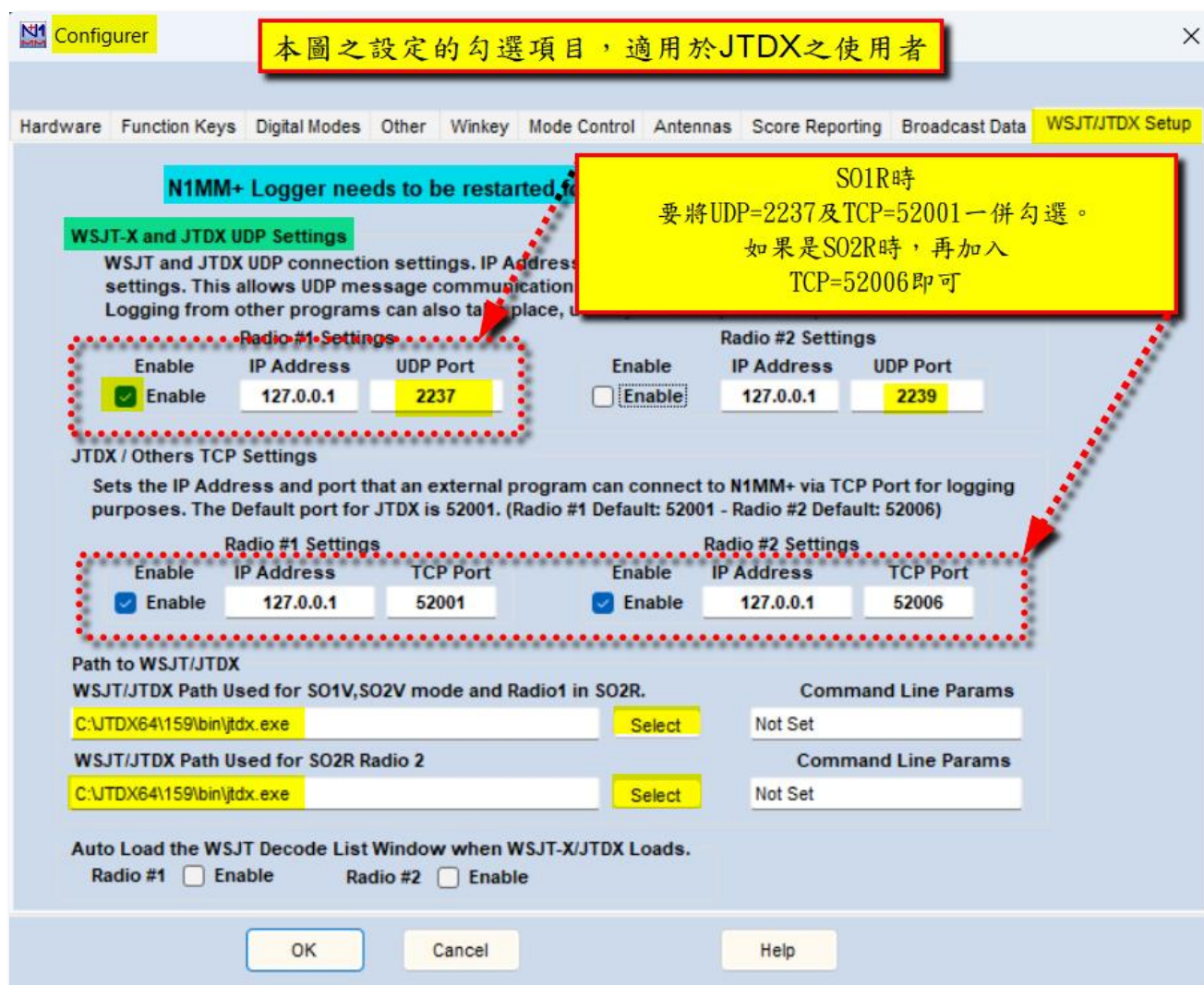
**後記**:如果你執意要去圖\_7 之最後「複選框」，做勾選時，將會是甚麼狀況？

答案是：「解碼清單」與「呼號清單」會同時出現。

你可以去對兩個解碼的表現成果逐一做個比較，你就可發現「解碼清單」的功力、實在是不怎樣！

如果你的電腦是屬於老舊或較慢的 CPU 時，最好不要讓「解碼清單」出現，以免讓你的玩性大減。

如果你是使用到 JTDX 時，你就去勾選圖\_7 中有關 JTDX 的選項。請比照圖\_7 的設定方法去選用 JTDX 的主程式路徑，其結果如圖\_8 所示：



圖\_8 N1MM+有關 JTDX 的 IP 設定說明(本圖僅適用於 JTDX，S01R 或 S02R 者)

如果你使用 JTDX、且需要用到「解碼清單」時，UDP=2237 也要勾選。否則會無法顯示「解碼清單」。如果你對於「解碼清單」不感興趣時，就無所謂了

至於 Log40M APP 的目標我將之定義為:終極儲存 LOG 登錄位置「總舵主 database」。UDP 設定係取決於 JT 以及 N1MM+產生 Logging 登錄時，視該二者設定 Second UDP 之值後、再去作相對應的設定即可。簡言之，請看圖\_62 至圖\_64 的詳細說明。

後記:根據 N1MM+有關本段落的手冊上提到:

如果你操作 WSJT-X APP 或 JTDX APP 之外的類似之應用程式時，該程式所產生的「Logging」資料，如果要丟回給 N1MM+的 database 時，也可以用「UDP=2333」的「封包」送出；其設定方法:

「請將此視窗頂部的無線電 #1 的 UDP 連接埠設置為 2333，並忽略此視窗中的其他設置」  
(請參考這裡「[N1MM+引用外部應用程式時的 Logging 設定](#)」)。



圖\_9 在 N1MM+之檔案推播及 Cluster 引用之設定

圖\_9 這是 N1MM+的推播的 UDP 設定選項。其中包含了 N1MM+本身作為多機電台 (Multi\_Stations)比賽時，讓同伴的電台彼此之間的信息交換資料，都能夠讓各電台的電腦之間享有「共享」之功能。

為了讓 Log40 應用程式能夠成為一個「終極儲存」之功能，在圖\_9 中所謂的「Contacts」就是指 N1MM+儲存每一筆 QSO 資料後，對外推播的一個管道 UDP=12060。這個值



是 N1MM+ 的默認值，是不能變動的。我們只要在 Log4OM 的 Connection 設定項中，給「UDP INBOUND」如圖\_62、設一個 UDP=12060 的「ADIF Message」，就可以讓 Log4OM 去取得 N1MM+ 提供的「共享」資料。

圖\_9 底端有一個「External Callsign Lookup」的選項，這個「外部呼號搜尋」功能係以 N1MM+ 內建的 Telnet 作為搜尋呼號之資料源。UDP=12060 係 N1MM+ 之默認值，也是不可以變動的。

這個功能可以讓你在 Telnet 視窗上的任一個 DX 呼號上、按左鍵兩次時，可將該呼號顯示於 EW 視窗中 CALL 框之上方外、在 JT 的「DX Call」框內也會出現該 DX 呼號；該雙擊動作會將目前之波段及模式改為該 DX 電台的工作頻率及「Mode」。如此，你就很容易的 QSY 到該 DX 電台頻率及 Mode 上去、準備跟他通聯。

這個「External Callsign Lookup」選項，建議你要去做勾選，否則在 N1MM+ 環境下、你就如同失去了一套優質的「雷達系統」；盲人摸象總不是咱們所願！

請參考這裡(「善用 Log4OM 及 N1MM+ 內建之 Telnet」)

## C\_1\_7、在 N1MM+ 環境下、如何打開 JT 模式的動作

假定你已經根據以上述的說明做了一系列的 N1MM+ 的基本設定版本(請按照圖\_2 至圖\_9 去設定)，接下來我們就要實際操作來開啟 N1MM+ 的 S02R 模式，並教你如何開啟 JT 的應用軟體，接著才會有下列的有 JT 的基本設定版本說明。

原本 N1MM+ 的主視窗會因依據本文的前述設定，且將操作模式設定為 S02R；當你重新開啟 N1MM+ 後，N1MM+ 就會出現兩個編輯視窗(Edit Windows)，名之為 EW1 及 EW2。同時，也會有一個 N1MM+ 的 LOGGER 視窗出現。如果沒有出現兩個 EW 視窗時，請回頭再去檢查 N1MM+ 所有的設定是否正確？。如圖\_10 所示。

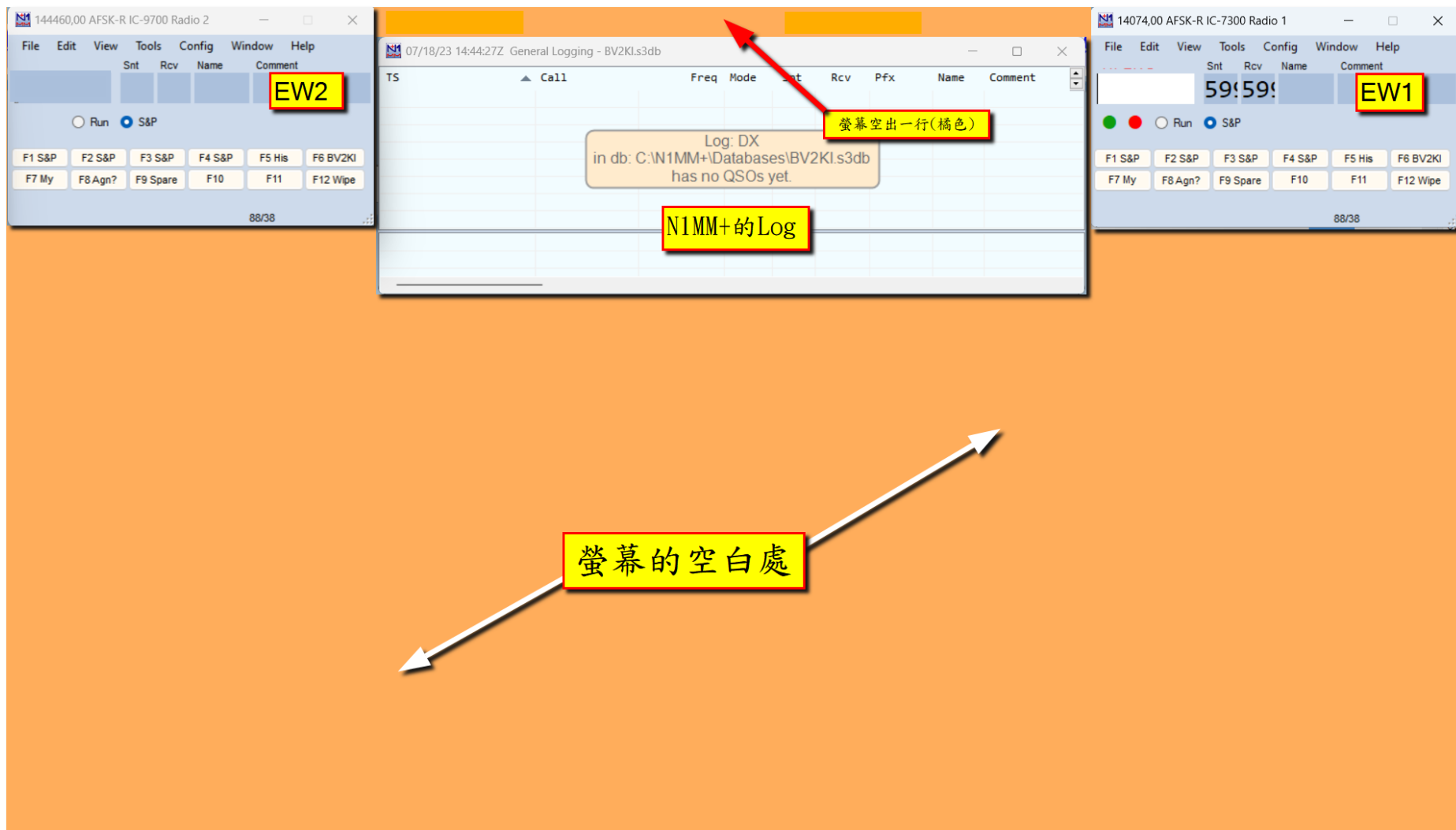
如果 EW1 或 EW2 的樣子跟圖\_10 不太一樣時，就請你在 EW1 或 EW2 視窗的下方任一空白處，以右鍵去取消一些不必要的選項勾選，以求畫面「精簡一點！」。請將上述三個視窗，盡量按照圖\_10 的布置、將之安排成為對稱狀態；

我是將兩個 EW 視窗分別放在螢幕頂層之左、右兩側。Radio\_1(IC-7300)是放在主螢幕之右側部、而 Radio\_2(IC-9700)則是放在主螢幕之左側。之所以這樣子的安排，是我考慮到以後要統合 Gridtracker 的 Call Roster 在有限的螢幕空間，能夠將之與其他較少作為的視窗畫面重疊運用(雖然已運用了兩個螢幕之現況)。

由於左側的 IC-9700 機組，我僅僅使用於 2 米波，QSO 使用機會較少，Radio\_1 的 WSJT-X 的主視窗部分畫面、就可以允許 Gridtracker 的 Call Roster 來共同重疊使用(請參見圖\_57)。

首先、開啟 Radio\_1：在編輯視窗 EW1 中的 Call 方框裡，填寫「FT8」、按 Return，就可以觸發 Radio\_1、叫出 N1MM+ 之 WSJT-X 的主程式視窗；此時，除了 WSJT-X 的主視窗畫面會呈現出來外，後續地也會有兩個視窗相繼出現；一個是叫：「N1MM-WSJT Radio Connection-1」(簡稱:搭線開關\_1)以及一個叫：「WSJT-X Decode List 1」(簡稱:解碼清單\_2)視窗。如圖\_10 及圖\_11。

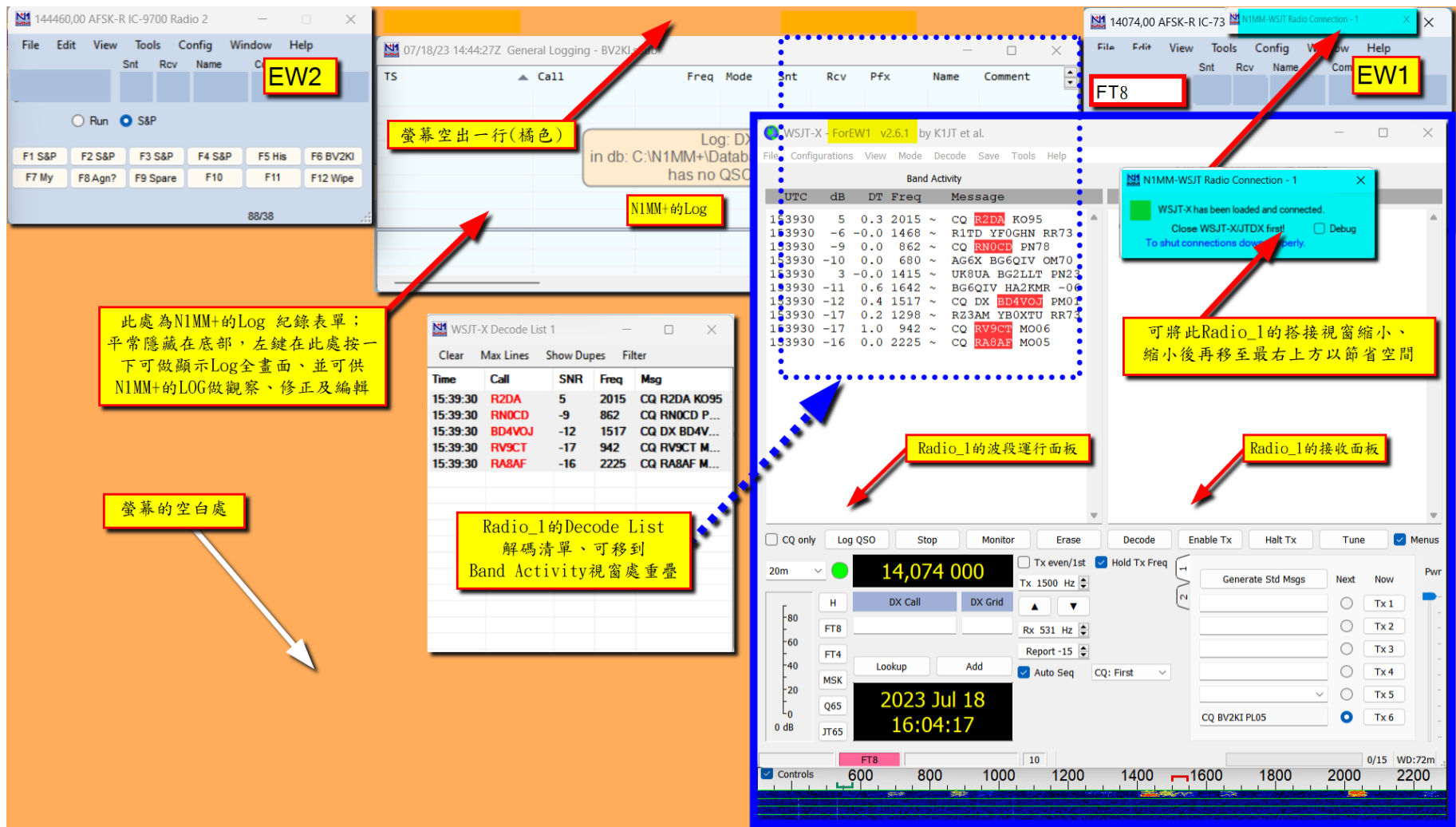
備註 1:在編輯視窗 EW1 中的 Call 方框裡，只能填寫「FT8」或「FT4」；如果想要在 N1MM+ 的環境下操作其他的工作模式，如「JT9」、「JT65」等等，當你在「Call」內填寫「JT9」或「JT65」時，是無法驅動 JT 的程式。因為 N1MM+ 不認為「JT9」或「JT65」是用來退出 JT 的關鍵詞、N1MM+ 會將「JT9」或「JT65」視之為「呼號」的一種、換句話說:N1MM+ 將「JT9」或「JT65」當作一個 QSO 的呼號來處理(想當然耳，是無法處理的!)。如果要執行「JT9」或「JT65」等模式，事實上、要去啟動 JT 以後，才能夠在 JT 的程式內去做切換的。



圖\_10 N1MM+的兩個編輯視窗(Edit Windows) EW1、EW2 及 N1MM+的 LOG 視窗之排列

在圖\_10 中、在 LOGGER 視窗(中央處)之頂端留一行字的高度，以便於跟 N1MM+之 Telnet 視窗重疊之用。

我是先在 EW1 中的 CALL 方框內，填寫「FT8」，按 Return 後，就會顯示如圖\_11 的畫面



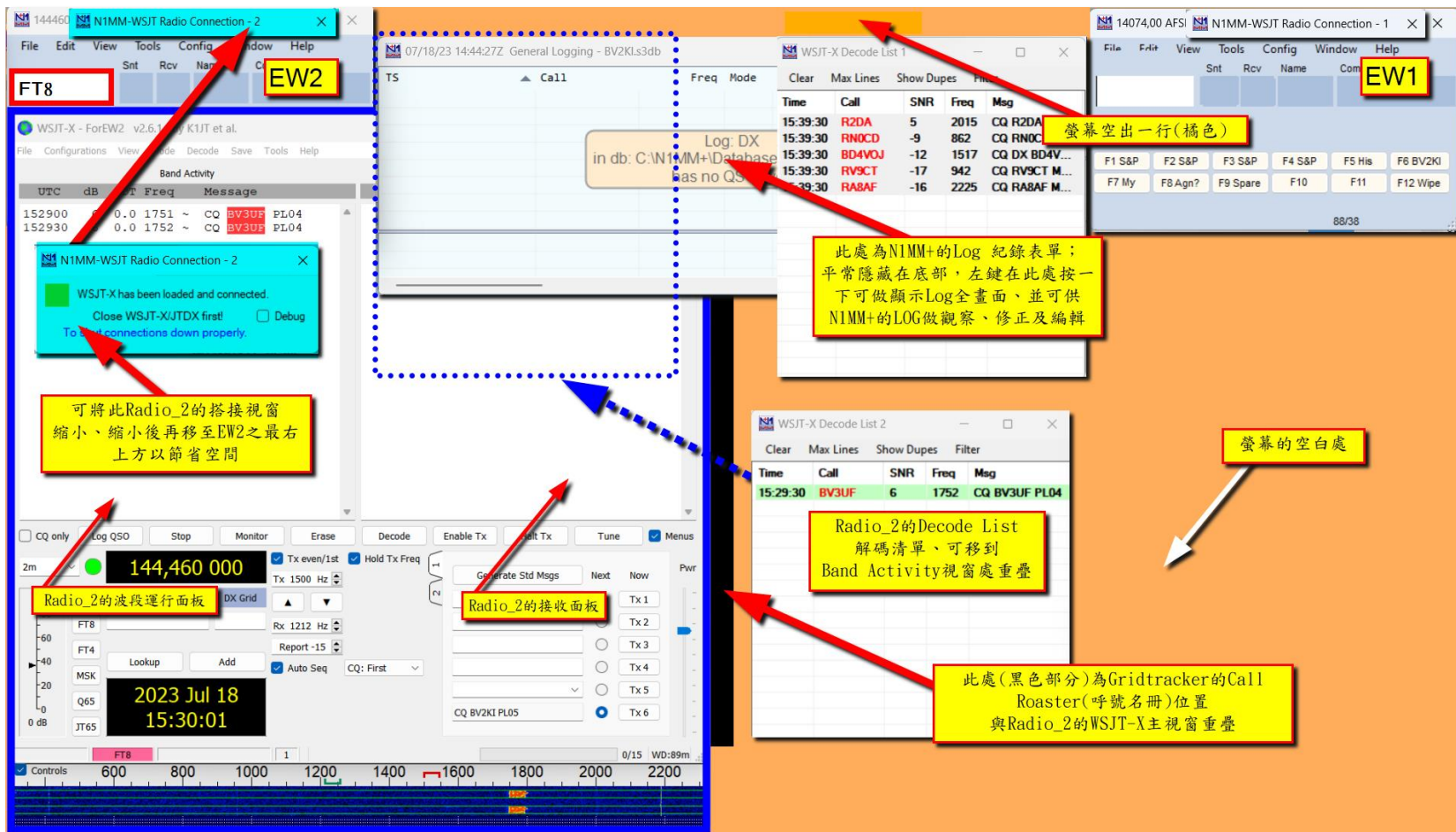
圖\_11 在 EW1 中開啟 WSJT-X 主視窗

- 1、將 WSJT-X 的主程式視窗(包含「瀑布視窗」)、Radio\_1 的搭建視窗以及 Radio\_1 的解碼清單，請依照圖\_11 的比例大小，緊密接合布置。
- 2、N1MM+的 Logger 視窗雖與 Radio\_1 的主視窗部分重疊，必要時、仍然可以隨時以點擊方式叫「她」出來「獻唱一番」！

3、在使用 JT 模式時，EW1 的視窗畫面，僅僅 CALL 的方框有功能外，其餘的畫面資訊均無意義；因此可以將 Radio\_1 的搭接視窗蓋於 EW1 上，一點都不礙事；甚至於可以將之縮小並移至 EW1 的最右上端、與 EW1 視窗做重疊(如圖\_11 之 EW1 的兩個「青色」的圖塊)。

4、Radio\_1 的解碼清單、為方便說明起見，我將其尺寸縮小並與 Logger 重疊；實務使用時，你可以在自行調整至實用之尺寸及適當之位置即可。

現在、讓我們以同樣的方式來處理 EW2；在圖\_11 之左上側、EW2 編輯視窗中的 Call 方框裡，填寫「FT8」、按 Return，就可以觸發 Radio\_2，叫出 N1MM+之 WSJT-X 的主程式視窗；此時，除了 WSJT-X 的主視窗畫面會呈現出來外，後續地也會有兩個視窗相繼出現；一個是叫：「N1MM-WSJT Radio Connection-2」(簡稱:搭線開關\_2)以及一個叫：「WSJT-X Decode List2」(簡稱:解碼清單\_2)視窗。如圖\_12 所示。

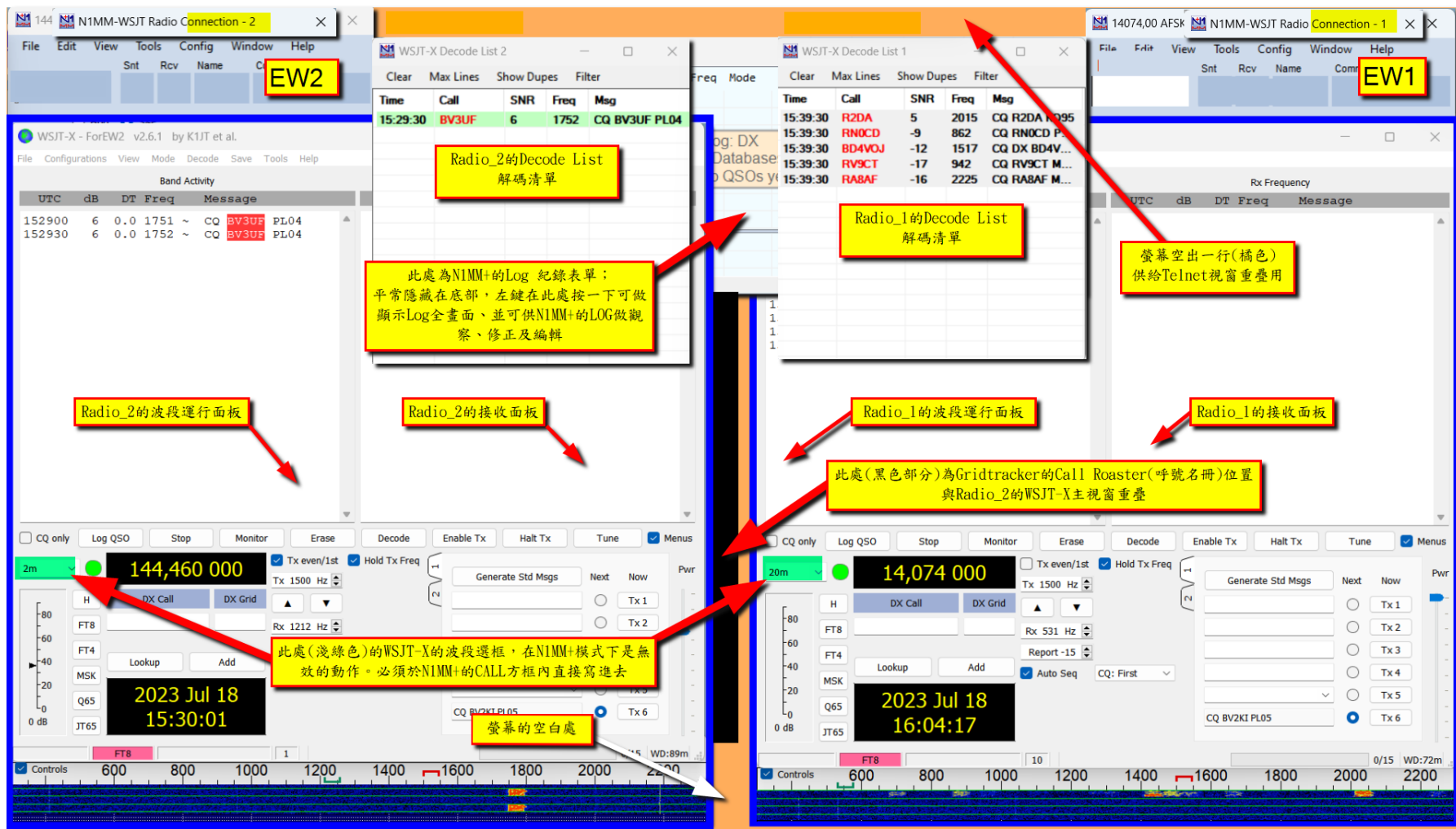


圖\_12 在 EW2 中開啟 WSJT-X 主視窗

目前的圖\_12 畫面，是為解說及清晰起見，我暫時將圖\_11 之右側的已出現過的有關 EW1 之 WSJT-X 畫面予以隱藏。在圖\_12 開啟的 EW2 或是在圖\_11 開啟的 EW1，都是採同樣的設定步驟，差異之處，僅在於前、後順序分別啟動而已；

這個屬於 EW2 的 WSJT-X 的出現，請依照與圖\_11 布置成與 EW1 者、做成對稱樣；其綜合結果如圖\_13 所示：



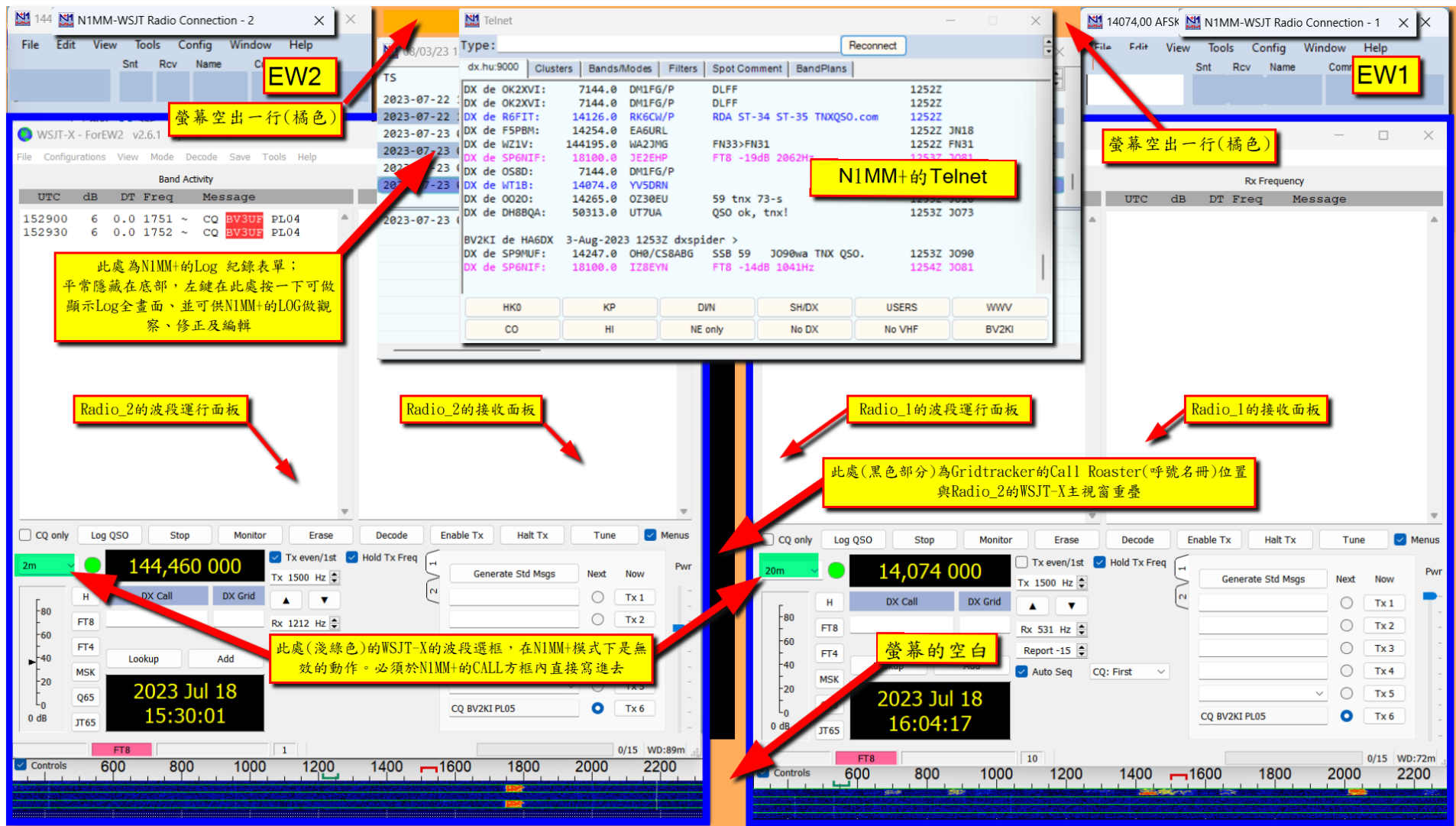


圖\_13 N1MM+中的雙 WSJT-X 主程式排列的樣子，這就是 N1MM+的基本設定版本操作模式。

請注意:由於圖\_7 裡的最底端設定是有做勾選的；因此，Radio\_1 及 Radio\_2 的解碼清單是有所作用的。此時圖\_13 表示 N1MM+的基本設定版本操作模式就是這副模樣。

後記:若採用 Gridtracker 版本設定時，圖\_7 之最底端不做勾選，兩個解碼清單已無作用就不會顯示出來、以至於 N1MM+的 Logger 視窗之頂端會保留一行的空白狀，此時就將該 Logger 的視窗頂端擴充至最頂端即可。或者、我們引用 N1MM+的 Telnet 功能，如圖\_13\_1 所示。有關 Telnet 的說明請參考「善用 Log40M 及 N1MM+內建之 Telnet」及參見圖\_122。





圖\_13\_1 N1MM+中的雙 WSJT-X 主程式排列的最後版本(採用 Gridtracker)

N1MM+的在設定 S02R 時，將 EW1 設為螢幕之左側、EW2 則設定為螢幕之右側。經過測試後，我認為將 EW1 及 EW2 的位置對調成為目前圖\_13 或圖\_13\_1，都不影響所有之操作及其結果。N1MM+之原意可能是在做 S02R 比賽的習慣使然吧！如果你仍願意堅守 N1MM+的規則時，那就將 EW1 及 EW2 對調位置即可。

前述的 EW1 及 EW2 的顯現的各四個視窗，你先暫時不要去管他的尺寸、大小及位置，只要將之挪至大概跟圖\_11 與圖\_12 的樣子、且可供操作即可；俟所有有關後續在 WSJT-X 裡面做的設定完成且處於正常操作後，再來做一次桌面整理即可。完整的桌面就像圖\_13(或圖\_13\_1)一樣就好。

桌面配置畫面原則上、很清楚的看出、係以左、右區隔開、做為兩部機組的獨立操作；各半個畫面中、上方部位，隱藏著 N1MM+的 Logger 視窗；這個視窗視需要時，去左鍵單「點」之，可以按照常規工作、做為 Logger 的檢視、修改及編輯的動作。

圖\_13(或圖\_13\_1)的中、下半部，有個黑色的區域，這是保留給部分 Call Roster 呼號名冊視窗的重疊後顯露一小部份之區塊。這個 Call Roster(呼號名冊)的開啟，我會在講解有關 **JT** 的基本設定後，再於講解如何去設定 Gridtracker 版本時、會再詳加說明的。

以上的說明，已經是可以逕行於 N1MM+的環境下直接去操作 **JT** 的工作所有相關功能。接著下來的工作就是如何將 **JT** 裡面的基本設定做說明，然後才能讓 N1MM+來駕馭操作 **JT** 模式。

## C\_1\_8、N1MM+的音效設定

在 N1MM+環境下採用 **JT** 時，除了電腦本身內建音效設定以及 **JT** 應用程式之內部的音效應依照各該設備要做適當設定外，在 N1MM+中仍然要注意去做「Logger+音效設定」。請依照下面之圖\_14 及圖\_15 處理。

在 EW1 或 EW2 視窗、點選「選項卡 Config→ Logger+ Audio Setup...」後會出現如下所示：



圖\_14 N1MM+中，有關的音效設定\_1(音效播放)

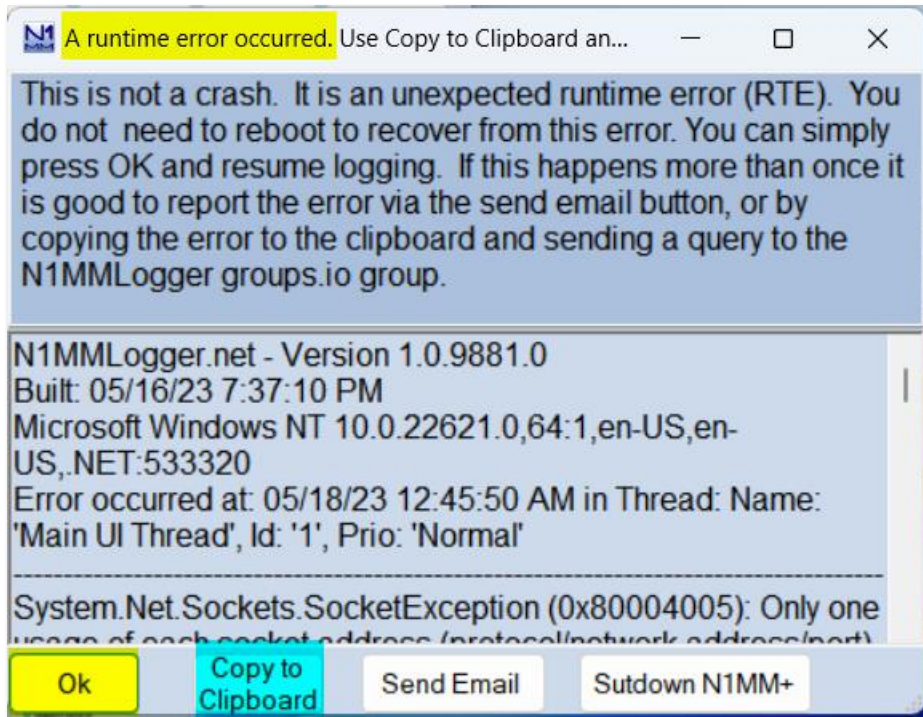


圖\_15 N1MM+中，有關的音效設定\_2(音效錄音)

行文至此，可以說已經完成了所有有關 N1MM+這套軟體的相關設定，也可以說:完成了「N1MM+的原版設定」所需之基本要求。

### C\_1\_9、N1MM+環境下、如何關閉 JT 模式的作業？

N1MM+環境下、操作 JT 途中，如需切換至 CW、SSB 或 RTTY 等其他模式時，需先關閉或退出 JT，然後再切換至其他模式；如果要關閉 N1MM+，也是要先將 JT 關閉退出後、才能繼續作業；否則會有卡頓、甚至於有可能會遇到 N1MM+ 迸出「不恰當關掉 N1MM+ 時的警告信息」（如圖\_16）



圖\_16 不恰當關掉 N1MM+ 時的警告信息

這個警告信息，並不意味著有「崩盤、當機」之意味。不需要重新將電腦重開機來恢復此一錯誤；你只要按「OK」鍵後，就可以繼續去執行 N1MM+。

如果不只發生一次時，最好將此錯誤信息以 E Mail 方式寄給 N1MMLogger 主事群組、讓他們作為改進之依據。

關掉 JT 時如果不小心，甚至當機或無法關閉時，此時請依照本文附錄中去執行一段 KillAll.BAT 程式(將此 BAT 檔案以記事本存檔並放在桌面上後，直接去左擊兩次)去執行[清除動作]的批次檔。

參見:(退出 WSJT-X(或 JTDX)時，順序錯誤關機的處理方法)。

所以、切換至其他作業或關機的程序是這樣子的：

#### 1、關掉 JT:

正規跳出方式:File →Exit (建議你採用此方式) 或者是:

左鍵單擊 EW1 或 EW2 之右上方的「X」符號(這方法、有時候會出問題，建議少用)，

接著程式會自動將「搭線開關」關掉，關掉的時間或許會長一點(可能有 5 至 10 秒，稍安勿躁)、等待 N1MM+程式關閉 JT 完畢為止！

(後記：如果久久看不到「搭線開關視窗」有結束的跡象時，八成是當機了；尤其是 JTDX 採用 TCP 的網內 IP 位址之作業方式，最易發現此現象。此時，唯一的解決方法就是「強制關掉」N1MM+。方法是：看看這裡「強制關閉應用程式的方法」。

## 2、接著、切換至其他模式之操作。

方法：在 EW1 或 EW2 中的呼號(或頻率)方框內填寫你想要區執行的模式即可！(如：CW、RTTY、PSK 或 SSB)；

或：不想玩了！

## 3、將 N1MM+關掉

左鍵單擊 EW1(或 EW2)之右上方的「X」符號(這個 X 號，還算是很穩定的；放心使用)。

至此，你已經可以開啟 S02R 模式，並且有兩組 JT 模式可以直接操作；當然你也可以只做一個機組的 JT 模式而另一個機組就可以自行去做 CW、RTTY 等其他模式。

如果你沒有打算玩「S02R」，只想在 N1MM+操作單機時，你將 Radio\_2(目前為 IC-9700)在 Configurer→Hardware→Set、設定成「Not use」(如圖\_5)時，可以當作 S01R 單機使用了。

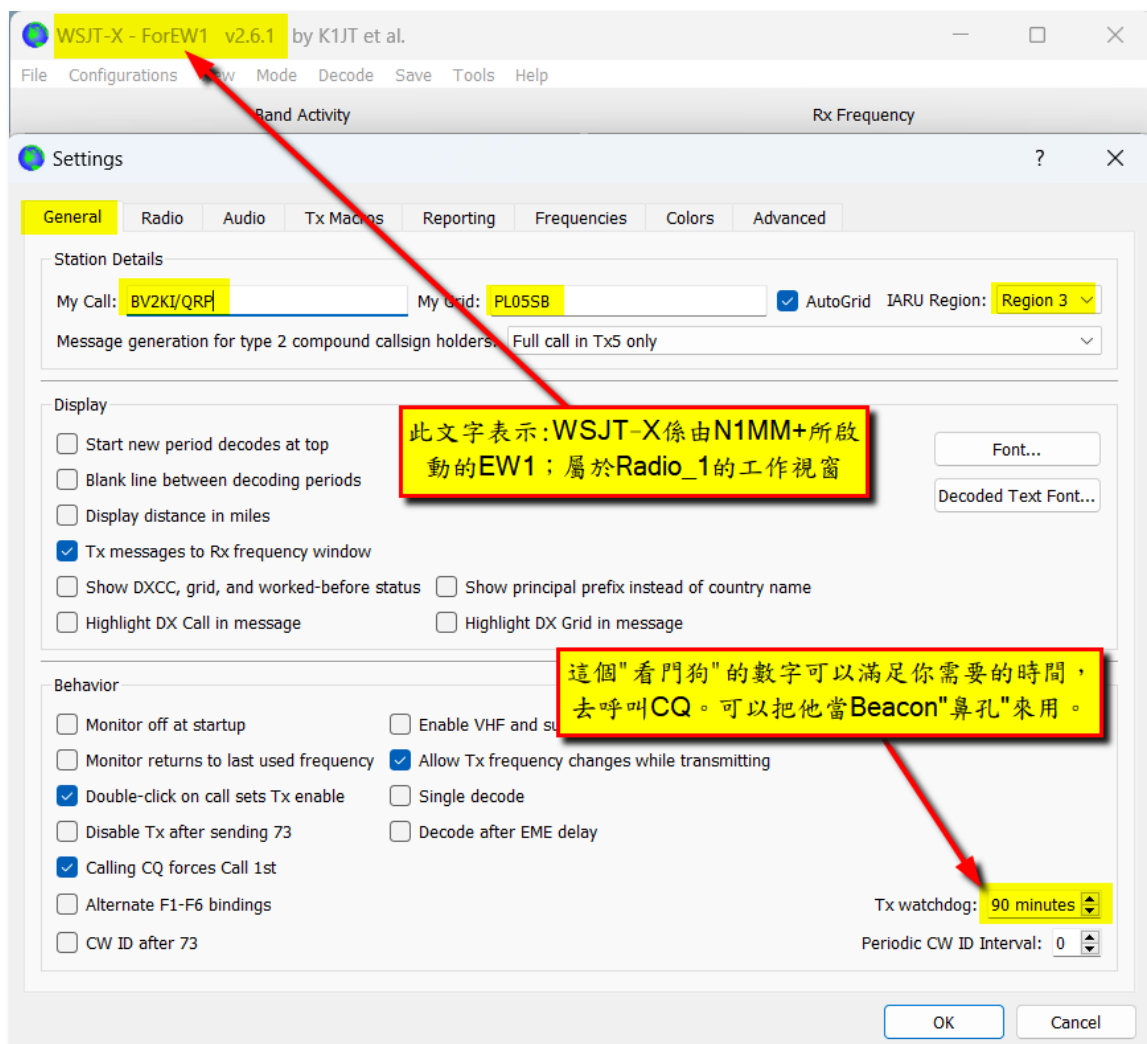
接下來請依照以下的章節來做 JT 的相關設定。



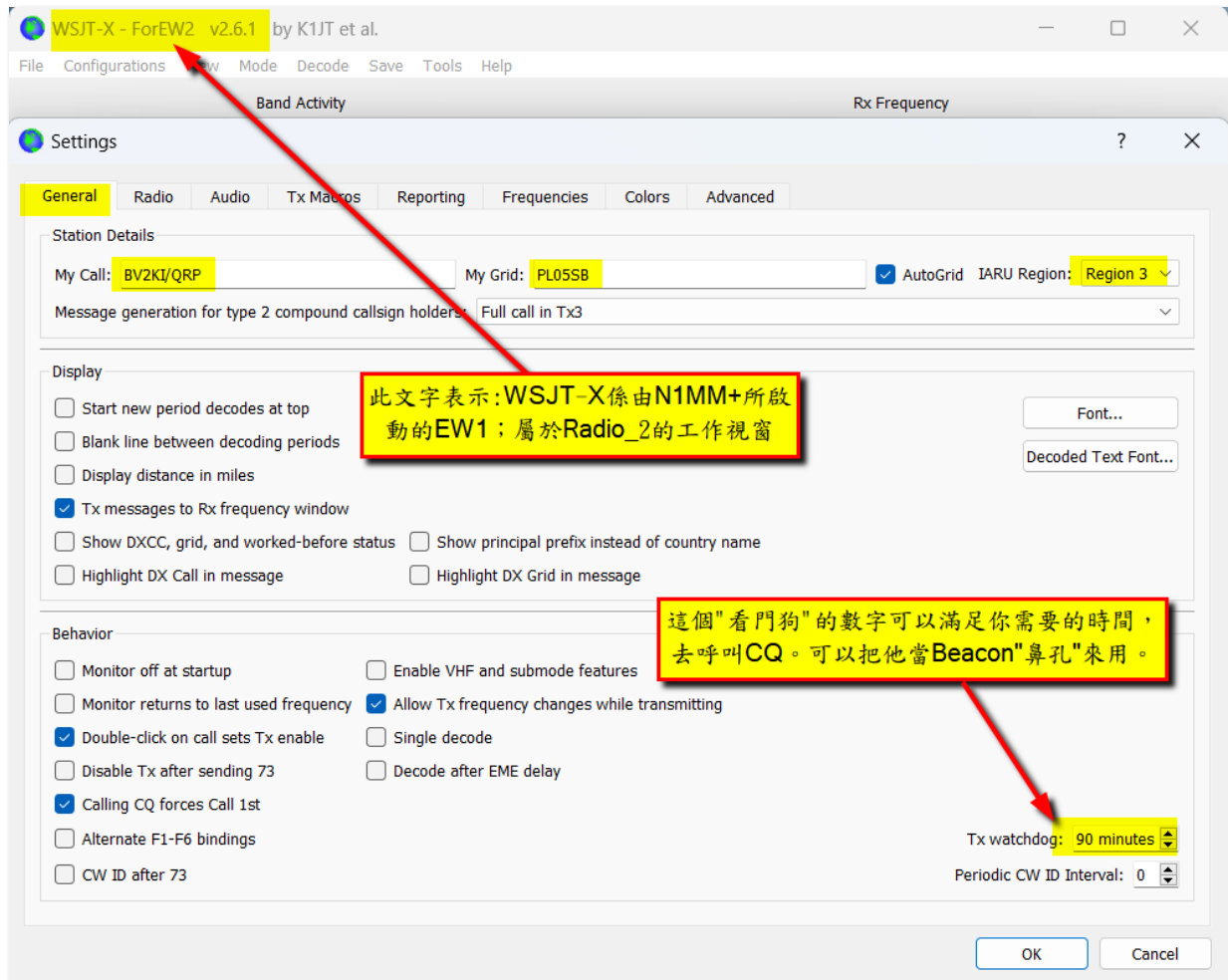
## C\_2、N1MM+環境下、WSJT-X(或 JTDX)中的設定

### C\_2\_1、WSJT-X(或 JTDX)中「一般 General」的設定：

同原單機版的設定；唯一不同點是:如圖\_17 左上角黃色顯示「ForEW1」是屬於 Edit Window 「EW1」，也就是指 Radio\_1 的工作視窗；同理，ForEW2 的字眼（見圖\_18），是屬於 Edit Window(EW2)，也就是指 Radio\_2 的工作視窗。



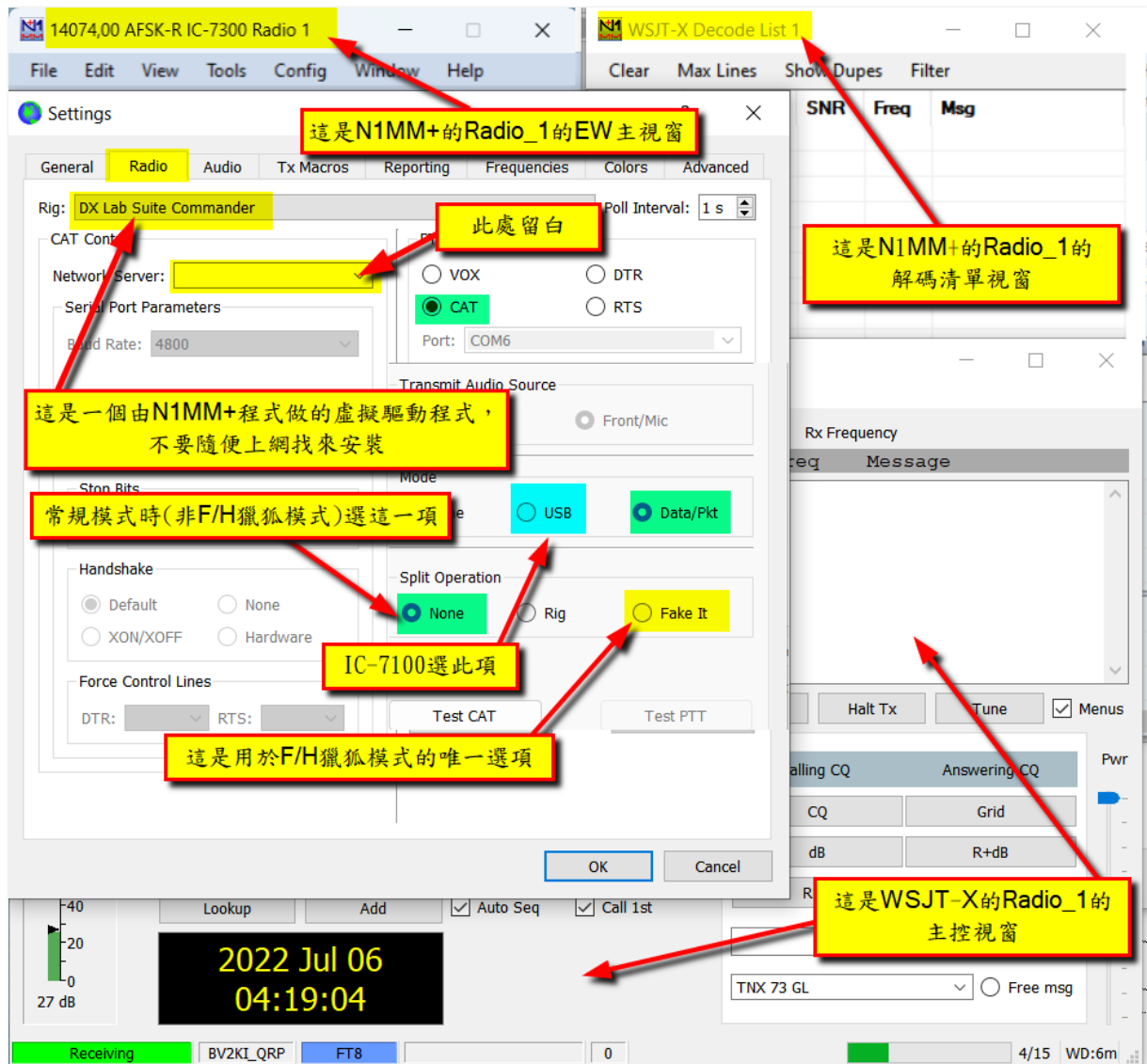
圖\_17 WSJT-X 之 Radio\_1 的一般設定(N1MM+環境下)



圖\_18 WSJT-X 之 Radio\_2 的一般設定(N1MM+環境下)



## C\_2\_2、WSJT-X(或 JTDX)中 Radio\_1 的「機組」設定



圖\_19 WSJT-X 之 Radio\_1 的 Radio 一般設定(N1MM+環境下)

請注意:這個屬於 Radio\_1 的 Rig 選項文字，必須是「DX Lab Suit Command」；這是 N1MM+內部設計的一個虛擬的驅動程式，跟「Rig 是甚麼名字」完全無關；不要傻呼呼的上網去搜尋該「名字」去配合安裝。這個「DX Lab Suit Command」是 N1MM+之設計者依照 [HAMlib](#) 的 Commander 設計成專為 N1MM+的 **JT** 模式所設定 Rig 的選項、與圖\_4 的硬體 Hardware 設定，完全無關，不要彼此混淆。

IC-7300 及 IC-9700 一般用於數據模式時，在 MODE 的選項欄裡、請選用 DataPkt(圖\_19 著綠色者)；如果你是用 IC-7100 時，這機型屬於舊機種、請選用其左側的「USB」選項(參考圖\_6 Mode Control 模式控制設定)。

順便一提：你稍加留意，當你完成 Radio 之設定並按 OK 後，可以察覺收發機上，有可能產生 Filter 變化(如圖\_20)所示。設定前的 Filter 原為「FIL1」、在設定並按 OK 後 Filter 會跳到 FIL2 的情形。

我以 IC-7300 或 IC-9700，選擇此兩項，在切換波段時，Filter 會跳到 Filter2 而 IC-7100 則仍然維持為 Filter1。追查許久後，這是由於這三種機型之原廠的默認值所致、你無法改變他，除非你以 Omni-Rig 來操控收發機時，且配合修改 Omni-Rig 的「.ini」檔案(參看再舉個例子及因此採用 FIL1 是最佳選擇)，才能將 JT 配合去操作這些必須用到的較寬的 Filter 1 的功能。可惜的是：N1MM+與 Omni-Rig 不互相支援、且無法整合運作，只好請你隨時注意，當你轉換波段時，看看 Filter 是否跟著變動成不是「Fil1」的情形；如有此情形時，僅能勞駕你以手動方式去收發機之面板上、在「Fil2」(或「Fil3」)指尖循環點按幾次，就可找回「Fil1」即可。



圖\_20 IC-7300 及 IC-9700 機組的 Filter 1 會跳到 Filter 2

如果你不採用 N1MM+、且使用 OmniRig 時，你就無此 Filter 變換之顧慮，因為去修改「機組的 ini 檔案」就可以改善(參看再舉個例子)。

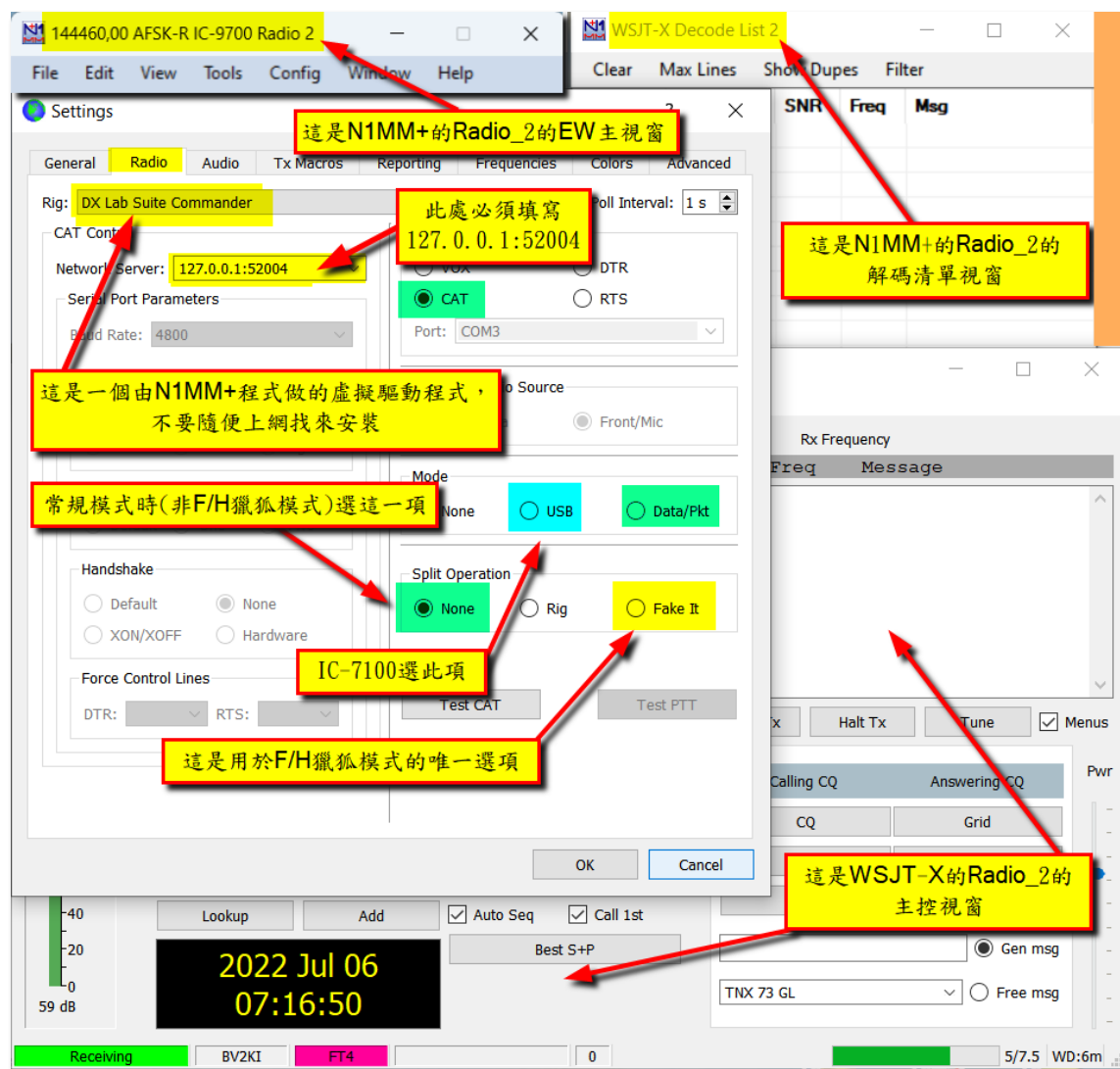
### 為何一定要用 Fil1 的選項？

這裡另有一篇文章，談到 ICOM 收發機之 Filter 及 TBW 的相關說明；

[Best Icom 7300 Transmit Bandwidth TBW Settings by K0PIR](#)

(最佳 Icom 7300 發射帶寬 TBW 設定 譯者:BV2KI2023/04/28)

### C\_2\_3、WSJT-X(或 JTDX)中 Radio\_2 的「機組」設定



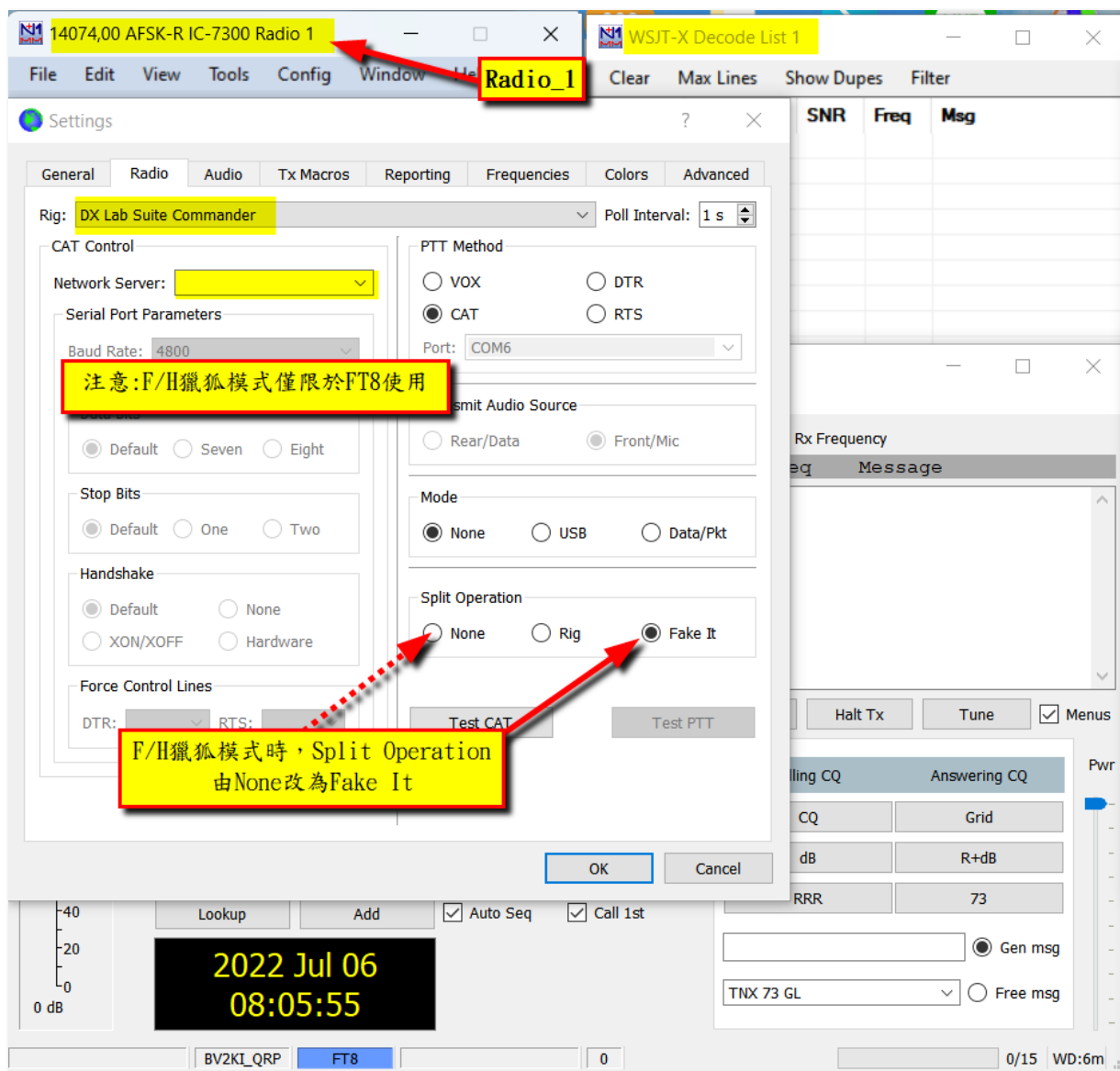
圖\_21 WSJT-X 之 Radio\_2 的 Radio 設定(N1MM+環境下)

Radio\_1 及 Radio\_2 的 Rig 填寫「DX Lab Suit Command」，是制式的選項；且 Radio\_2 的 Network Server 必須填寫「127.0.0.1:52004」；這也是不能隨便變動的。

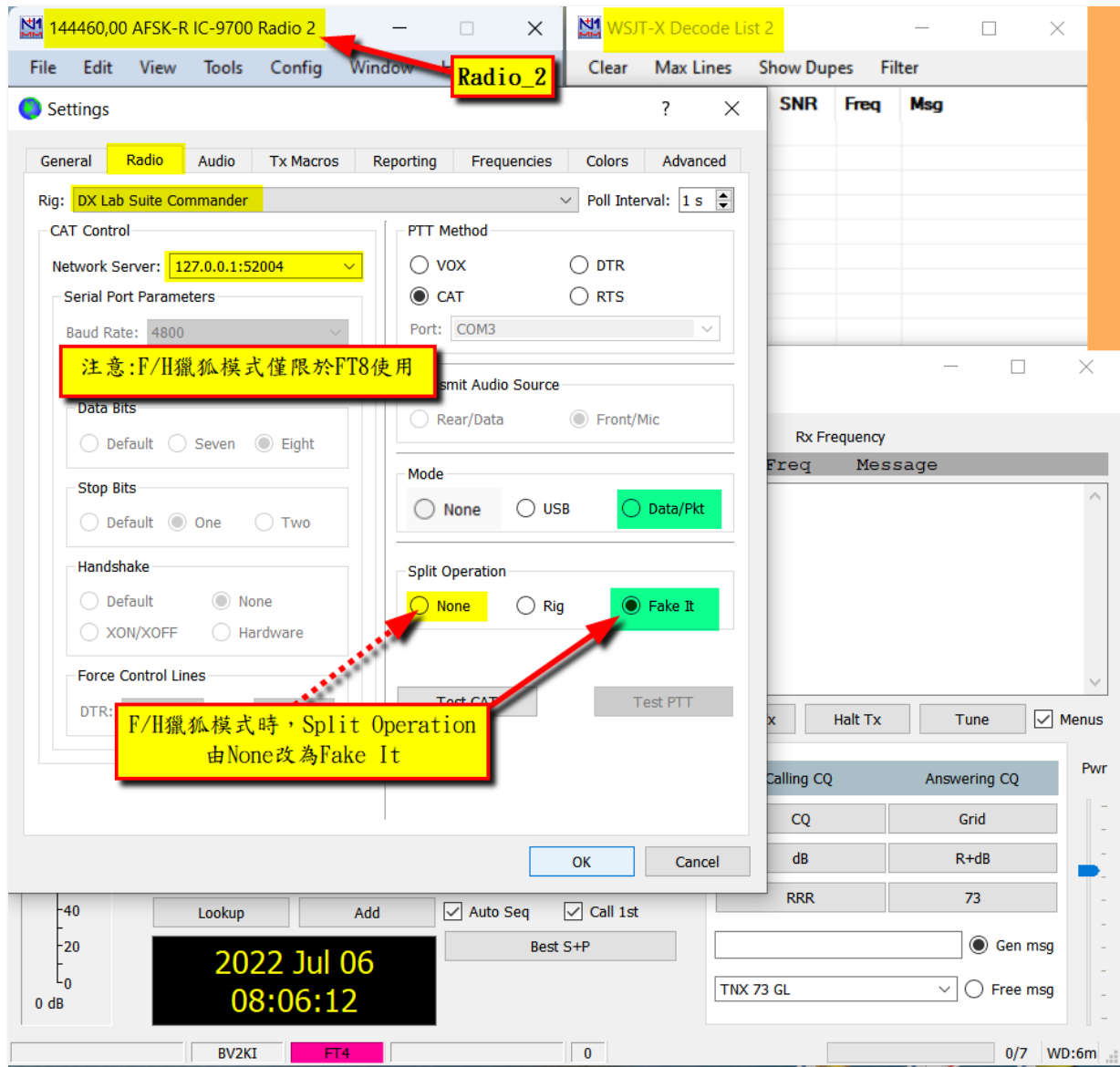
#### C\_2\_4、F/H 模式的考量：

F/H 的功能在 N1MM+ 的環境下使用 JT 時，僅能使用於 WSJT-X 應用程式。在單機組模式時，FT4 是適用於 F/H 獵人模式的。

如果採用 F/H 獵人模式時，除了在 JT 的選項卡 Advanced 需要另行勾選外，在 Radio 也需要做一小修改，這就是像圖\_22 的 Radio\_1 及圖\_23 的 Radio\_2 的樣子：



圖\_22 WSJT-X 中採用 F/H 獵人模式時的 Radio\_1 的設定(N1MM+環境下)



圖\_23 WSJT-X 中採用 F/H 獵人模式時的 Radio\_2 的設定(N1MM+環境下)

所謂「F/H 獵人模式」，是這樣看待的：

我們是「獵人」，而 DX 遠方電台稱則稱為「狐狸」。

「獵人」的 TX 頻率除了不能低於 1000 hz 外，圖\_22 或圖\_23 的 Split Operation 就要從「None」改為「Fake It」；如此、才能奏效；這是 JT 的原始設計規則。

不過，我另有妙招，下列這個方法，供您參考：

1、圖\_22(或圖\_23)中的 Split Operation，仍然勾選「None」

- 2、**JT** 應用程式之選項卡 Advanced 的設定中，不去勾選「獵人」模式。除非、你是參加像其他特定的比賽、例如:ARRL Field Day 之類者，你才要去 **JT** 的 Setting 項目下找出該選項卡 Advanced 去做該特別類的勾選。
- 3、將 TX 發射頻率，固定放在 1500 hz 處；「Hold TX Frequency」也要勾選。  
(請參考這篇 KV5R--有關數據通訊的設定---[1500 hz 的重要性](#))。
- 4、咱們的「TX」發射時隙(Time Slot)、要放在秒數為「5 或 15」處(亦即 Tx event/1<sup>st</sup> 不做勾選，這個「Tx event/1」是獵人作 TX 用的)。看們狗 Watch Dog(**JT** 之設定選項卡 **Setting->General** 之最右下方處，如圖\_17 或圖\_18 所示)；可以設定為長一點的時間，比如說:30 分鐘，你就可以不受原設計在 F/H(獵人模式)下、只能有連續四次的呼叫狐狸 DX 的限制(四次以後，你必須再次去啟動呼叫按鈕)。如設定成 30 分鐘、你可以努力不間斷的呼叫、直到 DX(狐狸)電台回答你或者可以叫滿 30 分鐘再說！
- 5、WSJT-X 主視窗之中央底下部位，[Auto Seq](#) 及 [Call 1<sup>st</sup>](#) (見圖\_26) 都要做勾選。
- 6、DX(狐狸)電台採用 F/H 模式操作時，這些增強的模式設計目的，無非是想要「清淨跑道、讓出空間」，避免在 DX((狐狸)的接收頻率上受到干擾，這就是另一種 Split 的作業方式，也可以說:讓這些遠征電台有多一點特權、好讓 Pile Up 得以掌握。  
DX(狐狸)會在 200 hz 至 800 hz 之間遊走、守聽，且以每 60 hz 之間隔、它會去建立一個類似專門的「談話房間」，來邀請它想要的呼叫電台去做 QSO。它可以一次邀請好幾位同時呼叫它的獵人電台、分別去到電腦指定的地方「開個房間、促膝而談」。  
(如果你在 1000 hz 以下發射、縱使拼了老命的呼叫時，DX(狐狸)的電腦會拒絕這一個段頻率段落的解碼、會讓你白費力氣、切記!。)
- 7、有很多高手、常常使用「多機組 Multi-Stations」的 [MSHV 程式](#) 來做類似獵狐的模式操作；我曉得這種程式設計似乎有些優點，他好像不受到 200 hz 至 800 hz 的範圍限制，可以隨意遊走四方、且隨時啟用或停用 F/H 模式，讓你模不著頭緒:我們是要開啟(或者關掉)F/H 模式來配合這個捉摸不定的 MSHV 電台呢?! 還是仍然矇著頭做無謂的「狂喊」?  
如果這傢伙又是跟遠征電台在同一個頻率同時出現時，那更是讓我們這些「獵人」搞得暈頭轉向。



甚至於，有些歐洲人到亞洲出差、順便玩玩無線電，無可厚非，可是往往對於他附近強大信號的亞洲電台呼叫總是充耳不聞。9N7AA 就是一個其中一個例子。聽說這款 MSHV 程式是有限制某些前綴呼號的功能、可以設定不去解這些亞洲的電台前綴碼、可真狠毒啊！

後記 1: Gridtracker 的 Call Roster 也可以使用 [REGEX](#) 的功能選項做此「狠毒」操作。此一選項可以達到如上述類似的「專門去解碼有關某一地區系列的前綴呼號之能力」。這個選項的寫法就像這樣：

`^(j8|w|a|k|n|vu|CO|Hi|9y):`



圖\_24 在 Gridtracker APP 中的 Call RegEX 的毒招

第一個開頭之符號是「`^`」(鍵盤按 Sht+6)；接著是以小括弧做限制範圍；小括弧裡面的前綴碼 `j8|w|a|k|n|vu|CO|Hi|9y` 等字母就是你想要去做 QSO 的電台，字母之間以「`|`」符號分開來，這就是可以限定哪一些呼號的前綴碼號可以顯示出來，其餘者皆不予顯示。你說「獨不獨」!?

後記 2: JTDX APP 也有將 Filters 的「隱藏某一地區」的設定選項；選項中是可以依照「洲別」來做限制單獨某個洲或幾個洲的呼號，不予以解碼、顯示。

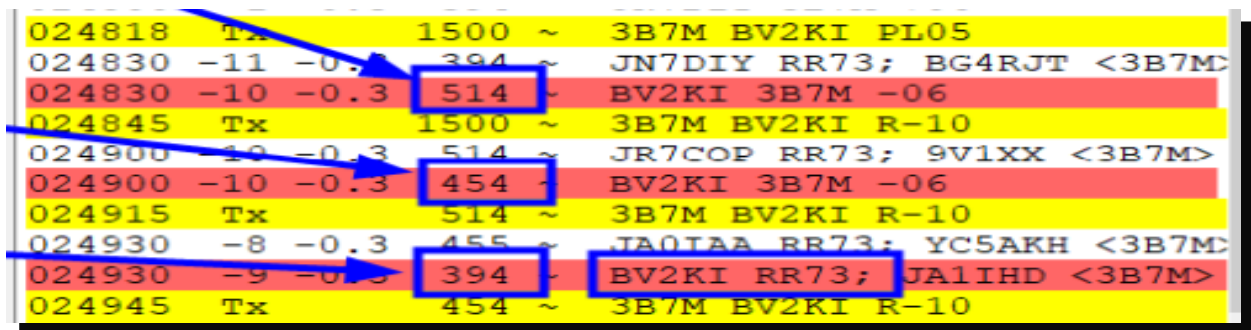
8、在前述之 1 到 3 的步驟完成後就可以盡情「不斷」的去呼叫 DX(狐狸)；只要將你的接收頻率放在 200 hz 處；這 200 hz 處(保證是沒有人會在此呼叫的)，因此你的「解碼接收視窗 RX Frequency」(就是指 JT 主視窗的右半部)是經常保持空白的(當然、開始操作前、你必須先將之清除呈空白狀才行)。

如果你有收到 DX(狐狸)的解碼信息，就可以將你的接收頻率移到他的發射位置上(以滑鼠左鍵在他的呼號上輕按即可)。如此，你的「解碼接收視窗」，就只有這個 DX(狐狸)的信息，會源源不斷的出現在「解碼接收視窗 RX Frequency」裡。其他獵人的信號不會顯示出來，用以減少「眼球」之干擾。

9、DX(狐狸)邀請你去「開個房間」時，會是怎麼樣子呢？

不要太在意，他如果開始邀請你時，在你的「解碼接收視窗」上一定會出現你的呼號，且是帶有紅字的呼號(至於是紅色還是其他顏色，端視你在 JT 應用程式之 Setting 的 color 的設定了)。

例如：請看看下面這張截圖：



圖\_25 F/H 模式下的「解碼接收視窗」狀況

這是 3B7M 遠征台正在以 F/H 模式、我跟他做 QSO 時的一小段截圖。

當我得到他的青睞時，我的呼號會以紅字的 BV2KI 出現(第一行紅底黑字部分、他給我回報-06 的強度)；

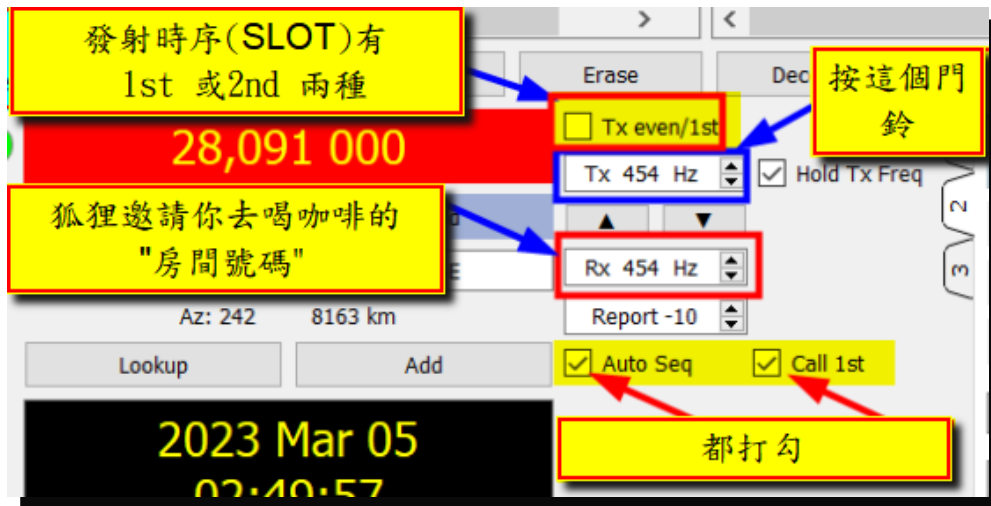
請注意他邀請我的房間號碼是「514」(第一行紅底黑字部分的藍色框框處)，但是、當時我回給他的頻率卻仍然在「1500」(514 行的下一行、黃色)，這 1500 不是他守候的房間，他當然無法跟我「促談」；

他再度以 454 的房間號碼再邀請我(見第二行紅底黑字部分的藍色框框)。

所以我馬上將 TX 頻率 454 寫至 TX 框框內(見圖\_26 藍色框框內)

或者你可以直接在 RX(紅色框之上方的▲符號輕按一下左鍵，

就可將 454 轉移上 TX 框內)並發射之。



圖\_26 迅速換個房間號碼「454」去約會

賓果！我總算是敲對了房間門，跟他完成了交易！（見圖\_25 第三行紅底黑字 **BV2KI RR73**；）。

他又換了一個房間號碼 **394**、然後去邀請下一位 **JA1IHD** 去喝咖啡（見圖\_25 第三行紅底黑字）。

請注意圖\_26 的最底端，Auto Seq 及 Call 1<sup>st</sup> 都要做勾選、如此才會有「自動應答」的功能。

這個妙招的**好處**是：

- 1、不必去理會「Split Frequency」一定要改成「Fake it」的規定；
- 2、不會受到連續「4 次」呼叫狐狸的限制；
- 3、依照常規的 QSO 模式操作，具一致性，免傷腦筋去配合做有關 F/H 設定之變動。

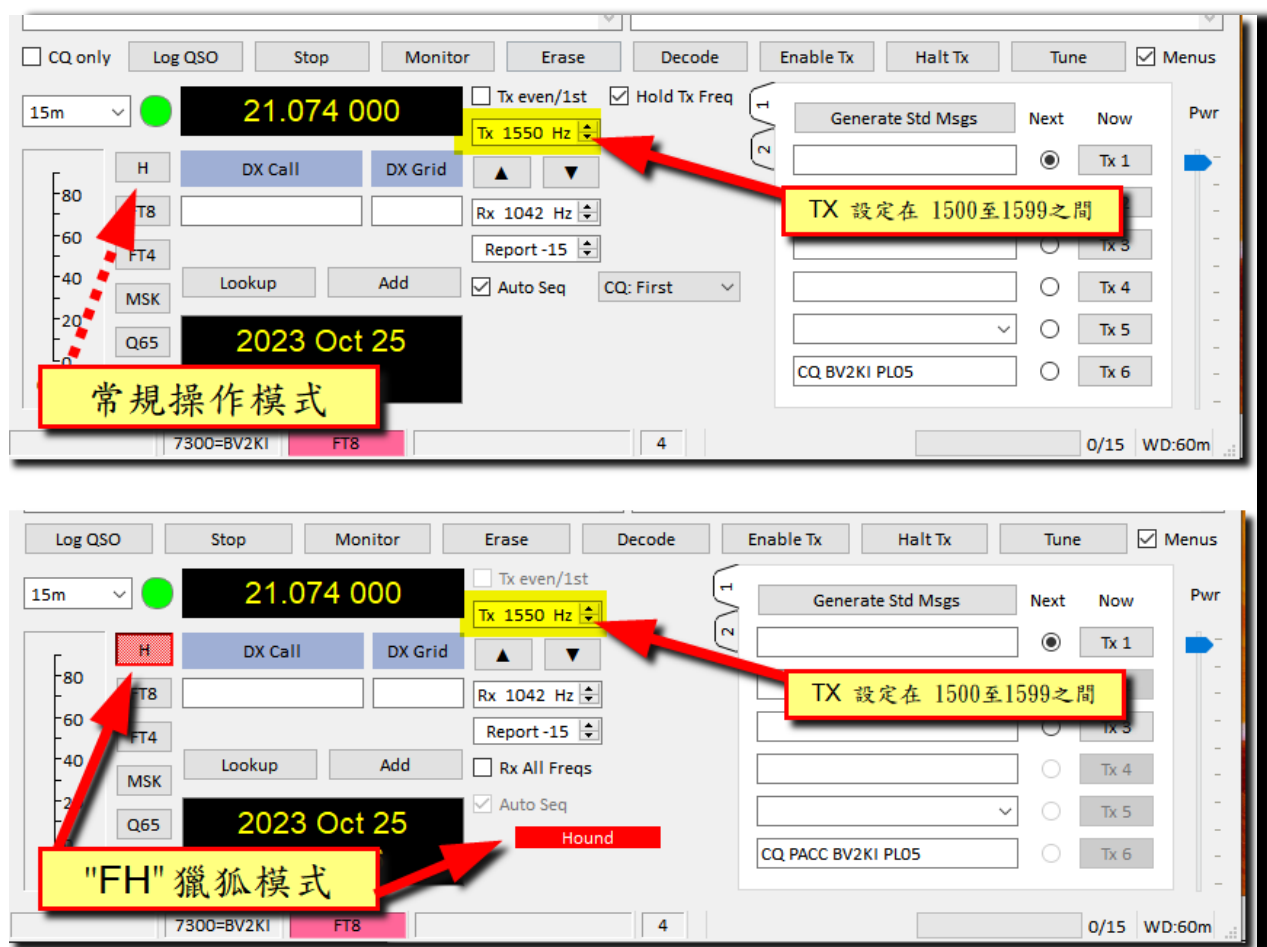
**不便之處**：

要去記得狐狸邀請你去「開房間」時的「房間號碼」，免得找不到門路；且狐狸邀請你的次數只有 3 次；錯過 3 次時，他會將你「棄養」，你得重新再呼叫它；所以要腦筋要清晰、手腳要俐落一點才行！

後記(2023/10/25)：

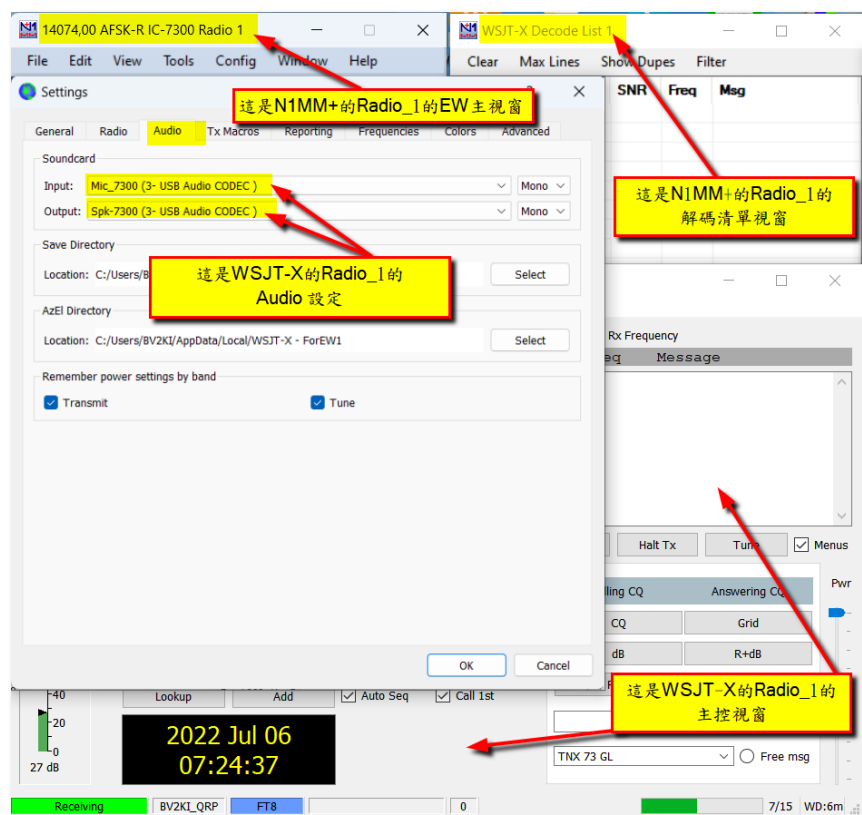
我後來發現將 Radio 之 Mode 設定為「Fake it」且將 TX 的音頻位置放在「1500 至 1599」間，無論是否採用「F/H 獵狐」模式，都是可以的運作。因此建

議您將圖\_19、圖\_21 至圖\_23 的「Mode」都改成「Fake it」。這樣的設定，只要你選擇「H」(如圖\_26\_1)與否就可安心的操作，不必再擔心使用「F/H」時的設定變更程序的惱人問題。上述的「妙招」就當作一般常識咀嚼，必要時也可以拿出來「炫技一番」、也不失為「冒險犯難，勇於實驗」之「業餘精神」。

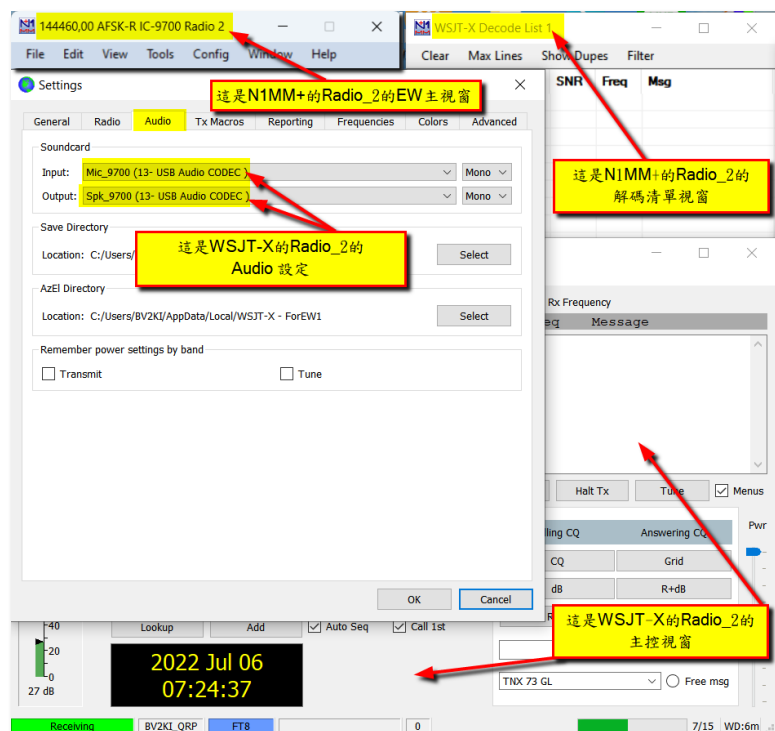


圖\_26\_1 JT 之 Radio 設定成 Fake\_it

## C\_2\_5、WSJT-X(或 JTDX)之「音效 Audio」 的設定

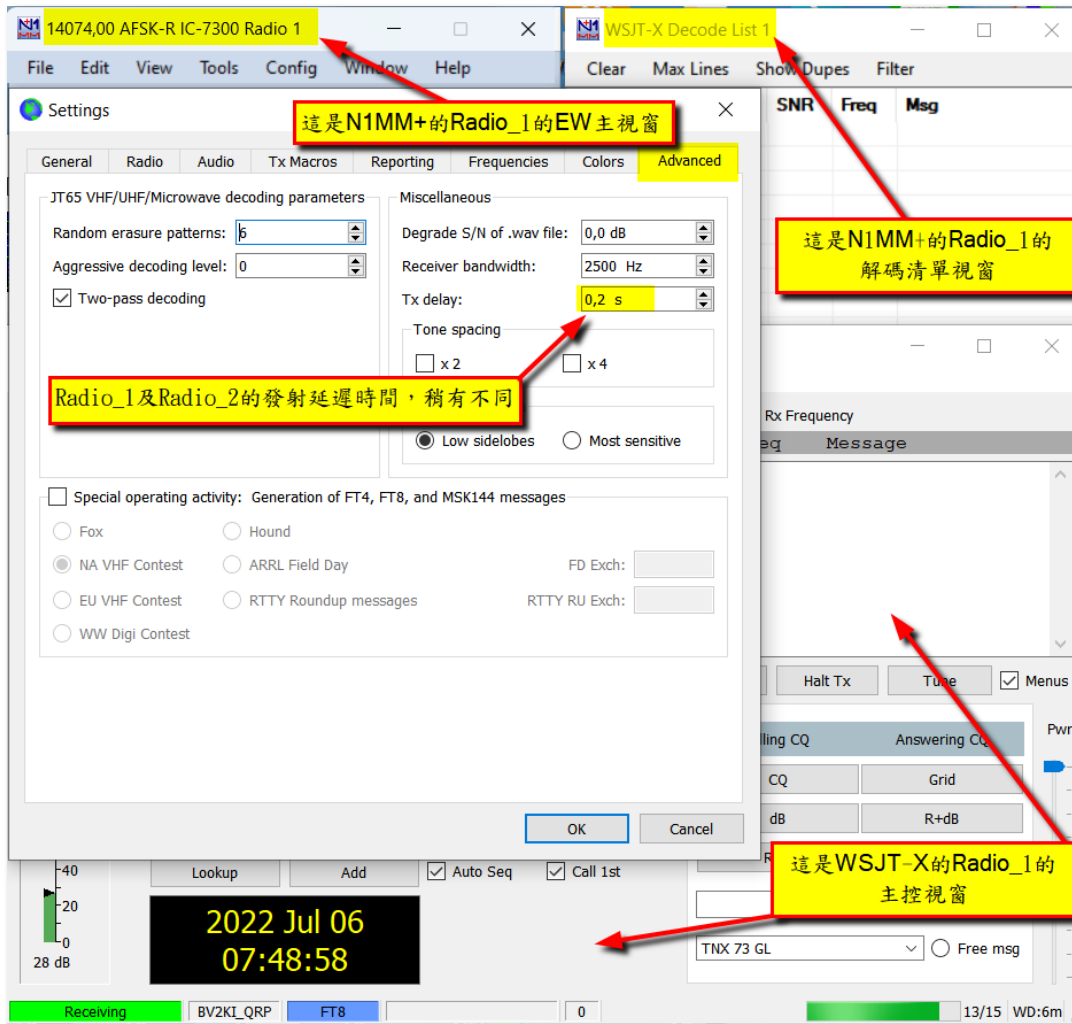


圖\_27 WSJT-X(或 JTDX)之 Radio\_1 的 Audio 設定



圖\_28 WSJT-X(或 JTDX)之 Radio\_2 的 Audio 設定

## C\_2\_6、WSJT-X(或 JTDX)中「進階 Advanced」的設定



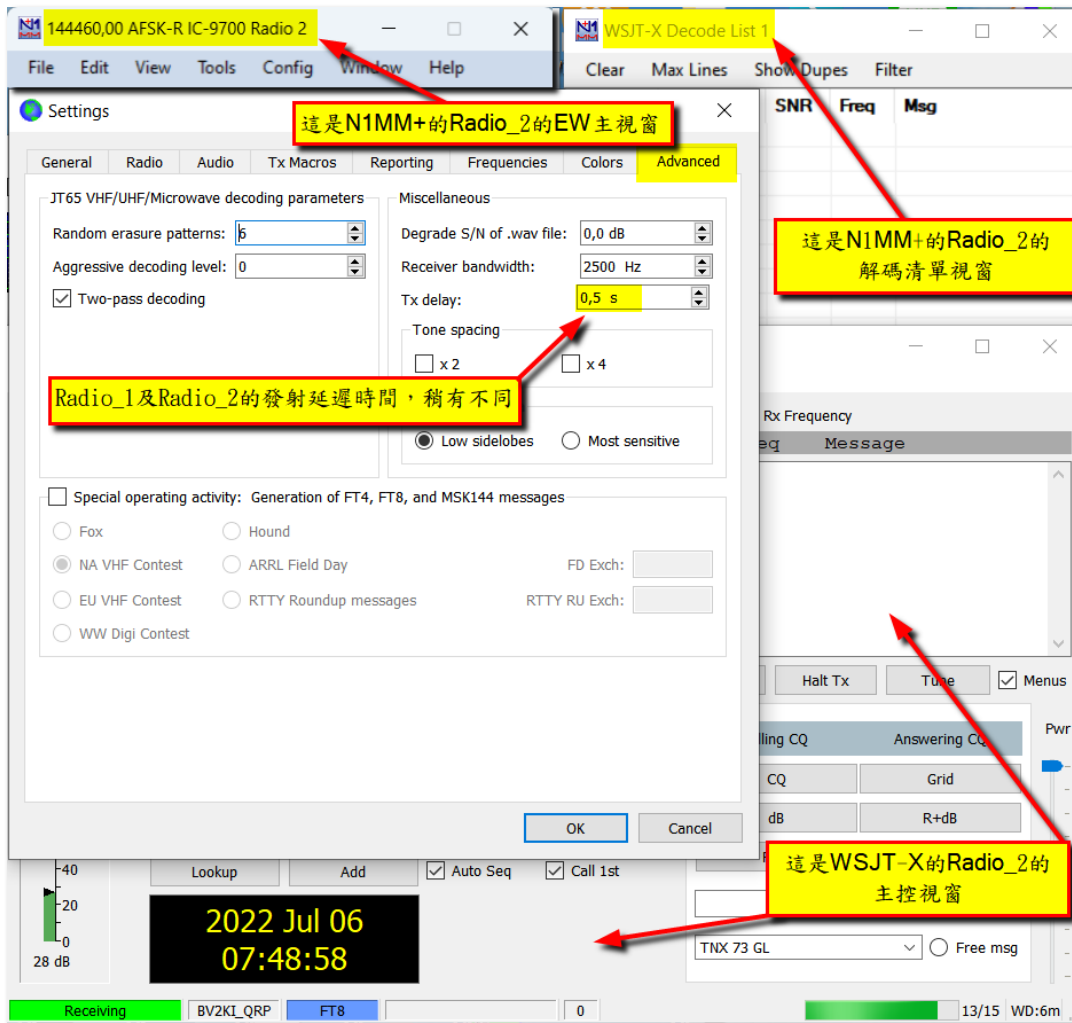
圖\_29 WSJT-X(或 JTDX) 中 Radio\_1 的 Advanced 設定

Advanced 的設定，主要是提供給 F/H 及一些特定的比賽的「進階設定」的有關 F/H 的設定。

原則上，這個選項卡之選項，還是以 F/H 或特殊比賽而建立的，你可以不必特別設定，大都維持其既有的默認值就可。

圖\_29 及圖\_30、只是顯示 Radio\_1 及 Radio\_2 的發射延遲時間，設為錯開狀態，以避開同一時間之機組的同時發射，減少一些 CAT 或 OmniRig 的衝突。





圖\_30 WSJT-X(或 JTDX) 中 Radio\_2 的 Advanced 設定

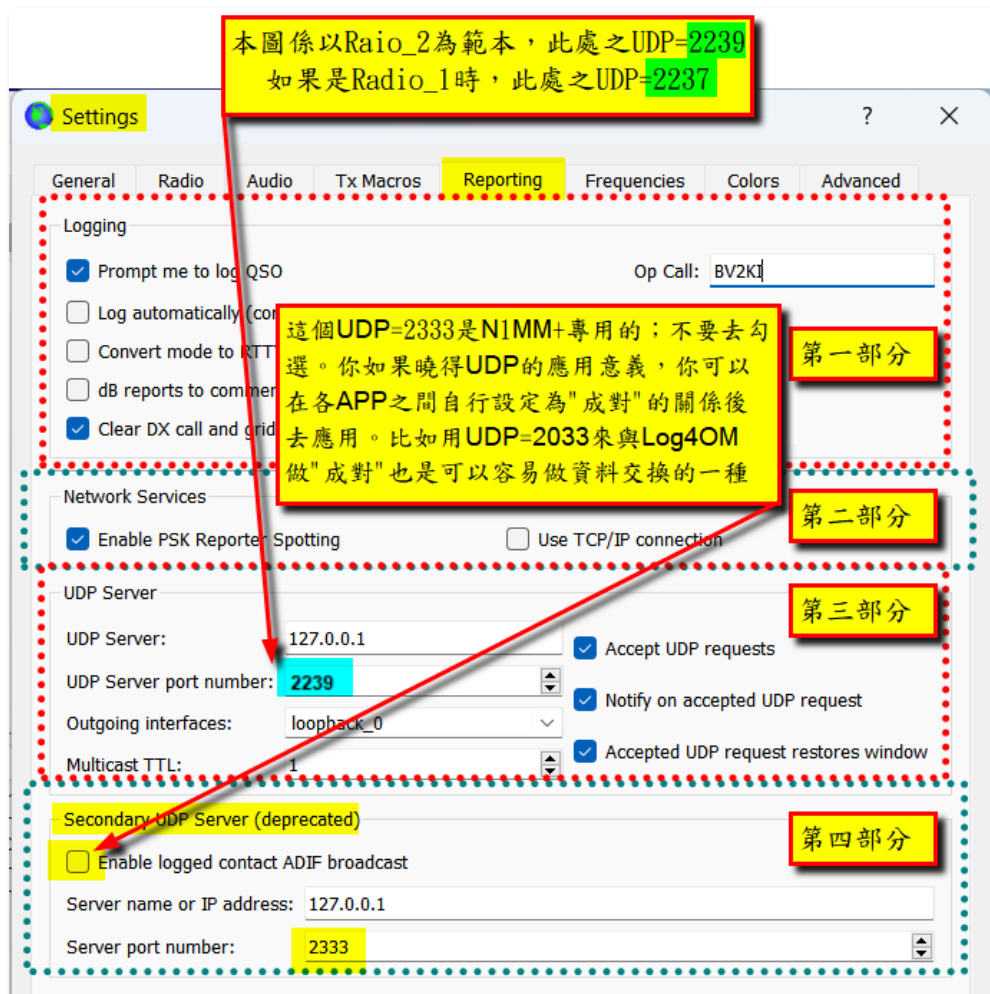
## C\_2\_7、WSJT-X(或 JTDX)中「推播 Reporting」的設定

這個 Reporting 設定，應該是本文的重點。透過以下說明，理解運作相關設定途徑後，才能夠去將 **JT**(或「**MSHV APP**」)產生的 QSO 之 log 引用於 N1MM+資料庫中；甚至於讓 Gridtracker 或 Log4OM 等應用軟體去引用、並進一步作為各該應用軟體之相關運作。

這裡的「推播 Reporting」功能，分為 4 個部分；

### C\_2\_7\_1、Reporting 的第一部分：Logging

Logging 部分是針對 **JT** 內部的日誌登錄 Logging 作業的幾個選項：這些選項，視您的需要去勾選。請參考圖\_31 的「第一部分」。



圖\_31 WST-X(或 JTDX)的 Reporting 設定說明(默認版本)

## C\_2\_7\_2、Reporting 的第二部分：網際網路伺服器

這個屬於對外部的推播，可以讓有關的外部網站或其他使用到內部網路系統 UDP 類的伺服器應用程式、所做的廣播(或通報)工作之用。

外部的接收推播網站:是指這個網站 **Digimode Automatic Propagation Reporter**([簡稱 DAPR](#))。目前已採用此網站這項服務的應用軟體計有 WSJT-X, Digital Master780, fldgi3.1, Airlink Express, JT65-HF, ROS, UR5EQF Logger 等七個應用軟體。如果你勾選了圖\_31 中的 Enable PSK Reporter Spotting 時，你電腦上的所有從 **JT** 產生的解碼信息都會透過網際網路、不定時的傳送到這個 DAPR 網站。這些集合了全球送來的解碼資訊都可以免費的提供給各方有興趣者，前往該站取得並加以分析、應用。

「DAPR」本身就有一個[地圖形式](#)的標記網頁可以讓人參考。你可以試著 將你自己的呼號寫進頁面最頂端、左起第五個方框內(記得要按 Return 鍵)，並以滑鼠「游」到你自家地址去看看、去了解一下:該網頁會是提供了甚麼有關你的資訊!(前提是:你要在 WSJT-X 之選項卡 **Setting->Report** 下去勾選「Enable PSK Reporter Spotting」以及發送 WSJT-X 的動作才行)。

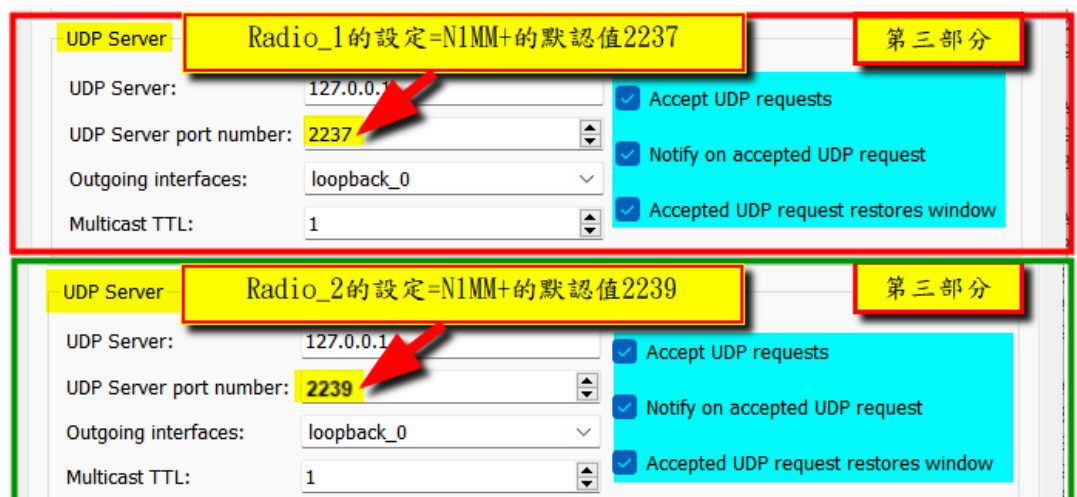
### C\_2\_7\_3、Reporting 的第三部分:1st UDP Server 推播的功能:

此一功能(可以稱為 1st UDP 通告)、是要讓我們所處內部網路的電腦(包括家裡的其他電腦)，在指定的 UDP 埠位址，去當作廣播、通告的腳色；好讓網內(或自身的電腦)使用該指定的 UDP 埠位址的應用軟體時，在接收到 Radio\_1(或 Radio\_2、或其他電腦上的 Radio\_3 等)的通知後，可以去讓該應用軟體去做其後續應做的動作；這個所謂的後續動作、對於 **JT** 而言，指的是：

「解碼部分的顯示」及

「選取 QSO 對象後的回饋給 WSJT-X(或 JTDX)去繼續執行 QSO」

的動作。



圖\_32 Reporting 的第三部分截圖(N1MM+環境下)

請看圖\_32，著青色的這三個選項的意思::

- 1、同意 UDP 之請求；
- 2、將已同意 UDP 之請求者的事宜通告出去；
- 3、同意 UDP 請求並重新儲存回原視窗。

白話來講就是:如果有某些應用程式對於 **JT** 在「2337(Radio\_1)」或「2239(Radio\_2)」的埠位址上，有所「觸發」時，**JT** 就會:

- 1、接受「通告」這檔事；
- 2、把「通告」這件事情，也張揚出去，給其他的應用程式知道說: 有這麼一回事；
- 3、同意「通告」者之要求，再次地將視窗資訊以更新方式做顯示。

註：此「更新方式」也可以說是「資料傳畢更新 DB\_Updated」的一種：各 APP 之間對於其所知道的資訊內容，可以相互印證、補充或更正；例如執行 **JT** 作業時、你可能沒有抄寫到對方電台的「網格 Grid」資料，到了與 Gridtracker 相互以 UDP 溝通後、Gridtracker 早已發揮其「搜索功能」並已取得對方電台的「Grid 網格」資料在握，此時 Gridtracker 就會將該 Grid 資料回饋給 **JT**、讓 **JT** 之 QSO 後的 ADIF 檔案得以補足、獲取全貌。當然、這種「更新」的動作，端視各該 APP 程式開發者的設計目的而定；例如：Log40M 的接收通告動作中，他也針對 WSJT-X、JTDX 及 N1MM+ 這三個 APP 設計了相對應的「更新」套件。然而、Log40M 的這三套指定的 UDP 設定值，均指定為 UDP=2234，不知道其用意為何？(請參見圖\_62 至圖\_64 之右半部的設定視窗)

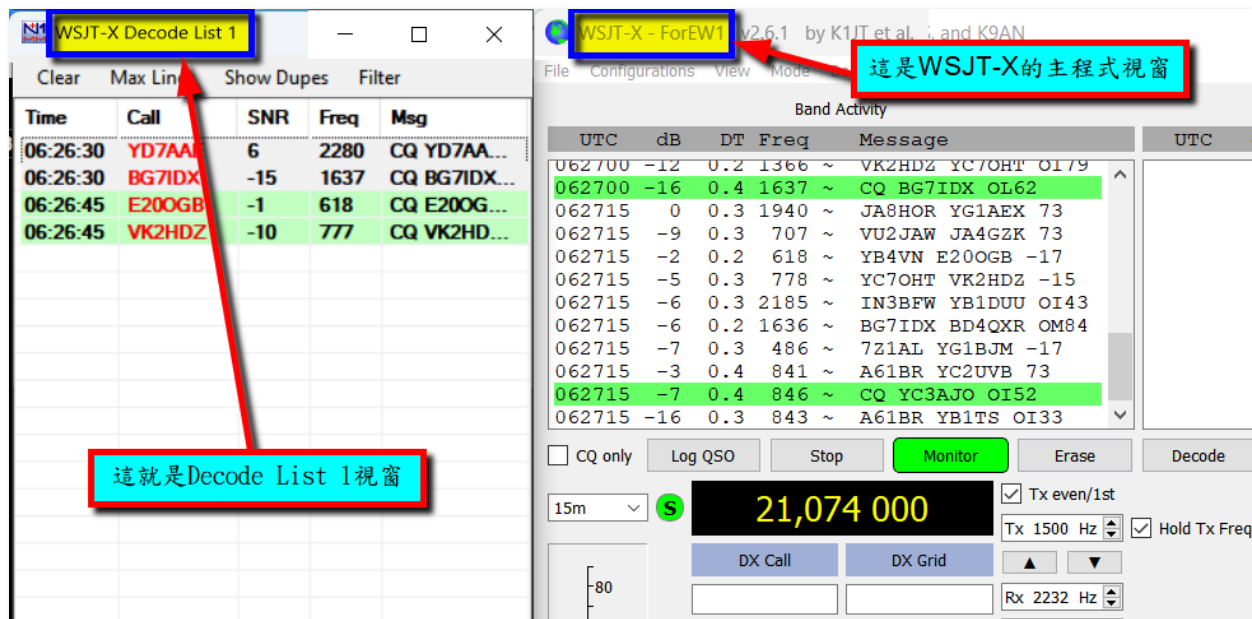
如果對於圖\_32 之右側的三個選項、不做勾選時，本文所做的所有的應用程式間之互為「溝通」的門被鎖住一般，就如同不給其他應用程式鑰匙一樣，就無法開門、往來；資訊也就無法交換了！所以、這三個勾選項目的位置都要將之「打勾」。

當然，你如果有實驗精神，不妨試著一個個不去打勾，看看會有甚麼樣的結果？別問我！我還真沒時間去嘗試呢！？

圖\_32 中的上半部分，「UDP Server port number」處，如果填的是「2237」，這是 N1MM+針對 Radio\_1 的默認值，可以將 **JT** 產生的解碼資訊在 N1MM+之 DeCodeList\_1 的視窗呈現。而、圖\_32 中的下半部分，「UDP Server port number」處，填寫的是「2239」，這個值是 N1MM+針對 Radio\_2 的默認值；如果你忘了為何有 2237 及 2239 這檔事，請回去看看圖\_7 的說明，就可明白「2237」及「2239」的原由出處了。

本小節與前述有關圖\_7 的內容似有重複之處，這是為了強調 N1MM+在實施操作 **JT** 模式時做了一個 Decode List 之「轉接口」，用 UDP 方式來承接 N1MM+本身具有的許多功能以及 **JT** 之長處間，發揮了整合作用。

圖\_33 是以 N1MM+的 Radio\_1 之解碼清單做示範的說明。這個 Radio\_1 的 Decode List 1，它長的就像圖\_33 左邊的樣子！



圖\_33 WSJT-X(或 JTDX)的 Decode List 的樣子(N1MM+環境下)

至於 Decode List 視窗的如何操作？請自行試試或者先看一看下一節說明，本文不做深入解說。



## C\_2\_7\_4、DeCode List 解碼清單

([這裡有一篇 N1MM+手冊原作的翻譯本，有興趣者可以前往看一看](#))

在 N1MM+應用程式之環境中，**JT** 在各 Radio 啟動下、有一種 Decode List 解碼清單視窗也會提供服務，Radio\_1 及 Radio\_2 各自擁有一個。**JT** 本身運行的解碼資訊會顯示於自己的解碼視窗外、一旦接收 1st UDP 的通報後、N1MM+就將解碼信息去顯示於「Decode List 解碼清單」上；這些顯示在解碼清單上的資訊，你可以在解碼清單上、有呼號的那一行地方、點擊該視窗上任一行解碼資料(簡稱「**點擊回車**」)後、就可逕行來回於 **JT** 與解碼清單之間做交換、更新信息的顯示，並持續做呼叫其他電台、直至做完 QSO 的動作為止。

在 Gridtracker APP 裡，有個 Call Roster(呼號名冊)，也是等同於 N1MM+的 Decode List 解碼清單的功能；只是前者功能比後者強了許多！如果你採用 Call Roster 時，一旦接收 1st UDP 通報後、Gridtracker 將會將解碼信息去顯示於「Call Roster」上。

Call Roster 呼號名冊與 Decode List 解碼清單雖然是屬於同一性質的一個「轉化介面」，但是需要透過 1st UDP 值之設定，才會建立起「轉化介面」的溝通管道。

Decode List 之 1st UDP 之默認值為 2237 及 2239

Call Roster 之 1st UDP 值，則是隨意給、只要設定值不會引起誤解即可。

Gridtracker 中、選項卡 **Setting→General** 中，有關的 1st UDP 是指「Received UDP」，其默認值是 2237；這個 2237 像似要故意與 N1MM+雷同。

根據前述 UDP 的準則、為了避免遭到誤解，我採用 1st UDP=2031；這個 2031、跟 N1MM+的 2333 似乎雷同、無妨!! 就當成拍馬屁吧! Hi ! Hi !。

至於 2nd UDP 之設定值、在 Gridtracker 中的「推播值」、我則設定為 2nd UDP=2032。這個 2nd UDP 之設定、會留到下一節、去做更詳細的運用說明。

## C\_2\_7\_5、Reporting 的第四部分：2nd UDP 伺服器的設定

Reporting 之 2nd UDP 的作用是用來做 QSO 資料登錄 Logging 指定用的埠位址。通常、**JT** 一旦完成 QSO 後，就會有兩個動作產生：

**動作 1**、**JT** 所產生的 QSO 資料、自己先儲存起來(儲存的位置請看圖\_35)

以圖\_35 中、舉例來說：

WST-X 應用程式言

Radio\_1 : C:\Users\BV2KI\AppData\Local\WSJT-X -IC7300

Radio\_2 : C:\Users\BV2KI\AppData\Local\WSJT-X -IC9700

JTDX 應用程式的位置

Radio\_1 : C:\Users\BV2KI\AppData\Local\JTD-JT\_7300

Radio\_2 : C:\Users\BV2KI\AppData\Local\JTDX-JT\_9700

WSJT-X-IC7300 及 WSJT-X - IC9700 是運行單機版之 WSJT-X 時，複製出來的機組名稱之資料夾。

JTDX-JT\_7300 及 JTDX-JT\_9700 也是運行單機版 JTDX 時，複製出來的機組名稱之資料夾

**動作 2**、視 **JT** 之 Second UDP 之設定值而定，去做儲存 ADIF 的通告。

Case A :如果 **JT** 模式是在 N1MM+的環境下作業時，N1MM+係透過默認值

UDP=2237、UDP=2239 埠位址，將完成的 QSO 資料、依據 N1MM+的虛擬 2040、2041 埠途徑去做存檔動作；所以，在 **JT** 的 Reporting 的第四部分之 2nd UDP 必須設為 2237 或 2239(見圖\_34)。

Case B :如果 **JT** 模式是在一般常規環境下作業時、**JT** 之 2nd UDP 的設定值設定成任何數字都是可以的。本文是以 2nd UDP=2033 設定之(這個值是我故意設定成與 N1MM+的默認值 2333 相仿)；其他的像 Log40M 的應用程式中、其接收 UDP 也設為 2033 時(見圖\_64)，此一 QSO 的 ADIF 資料也會被 Log40M 去做儲存。

Case C :在 Gridtracker APP 中，我在前一節已提到 1st UDP=2031 及 2nd UDP=2032 的設定方式。這種設定是藉著 **JT** 的 1st UDP=2031 的「共享」作用，將 **JT** 完成 QSO 後的儲存資料、會無條件的傳給 Gridtracker。然後再由 Gridtracker 透過其 2nd UDP=2032 以「轉播」方式、「推播」出去。此時、如果在 Log40M APP 有關 Connection 的設定中，事前已經設定好以 UDP=2032 作為「接收」橋

樑時(見圖\_64)，這個由 Gridtracker 「借花獻佛」方式，就會將 QSO 資料傳給 Log4OM 並做儲存 ADIF 檔案的動作。

The figure consists of three vertically stacked screenshots of the 'Secondary UDP Server (deprecated)' dialog box in the N1MM+ software. Each screenshot shows the 'Enable logged contact ADIF broadcast' checkbox, the 'Server name or IP address' field (set to 127.0.0.1), and the 'Server port number' field. Red arrows point to the port number field in each screenshot, and yellow text boxes provide instructions.

- Top Screenshot:** The 'Enable logged contact ADIF broadcast' checkbox is unchecked. The 'Server port number' is 2333. A red arrow points to the checkbox. A yellow text box says: '這個2nd UDP=2333是WSJT-X的默認值；不建議你去做勾選' (This 2nd UDP=2333 is the default value for WSJT-X; it is not recommended that you check it). A red box labeled '默認' (Default) is next to the checkbox.
- Middle Screenshot:** The 'Enable logged contact ADIF broadcast' checkbox is checked. The 'Server port number' is 2337. A red arrow points to the port number field. A yellow text box says: '這個2nd UDP=2337會將Radio\_1完成QSO的資料(.adi)存入N1MM+的Database中' (This 2nd UDP=2337 will save the QSO data (.adi) for Radio\_1 into the N1MM+ database). A red box labeled 'Radio\_1' is next to the checkbox.
- Bottom Screenshot:** The 'Enable logged contact ADIF broadcast' checkbox is checked. The 'Server port number' is 2239. A red arrow points to the port number field. A yellow text box says: '這個2nd UDP= 2239 會將Radio\_2完成QSO的資料(.adi)存入N1MM+的Database中' (This 2nd UDP= 2239 will save the QSO data (.adi) for Radio\_2 into the N1MM+ database). A red box labeled 'Radio\_2' is next to the checkbox.

圖\_34 在 N1MM+環境下之作業 WSJT-X(或 JTDX)之 Reporting 的第四部分

## C\_2\_7\_6、如何設定正確的 1st UDP 及 2nd UDP ？

本小節的說明是為了讓各位深入了解起見，而做的補充說明。如果你懶得再詳細了解，就跳過去，直接跳到「N1MM+的基本設定版本」。

為了單純起見，我將這個「第三部分:1st UDP Server 以及第四部分的 2nd UDP 埠位址的設定，分兩小節來說明。

第一小節的是針對未曾使用過 Gridtracker，或目前不想去學習 Gridtracker 的同好們的說明。第一小節的設定方式，我暫時稱之為「N1MM+的基本設定版本」。只要完成這個原版設定，你就可以在 N1MM+的環境下完全去操作 JT 的常規作業。一旦你完成設定、並實際操作一陣子、了解整個 logging 的來龍去脈後，如果想要進一步享用 Gridtracker(或其他類似者)的好處時，你也可以回頭來看看第二小節的設定，也是一種循序漸進的方法。

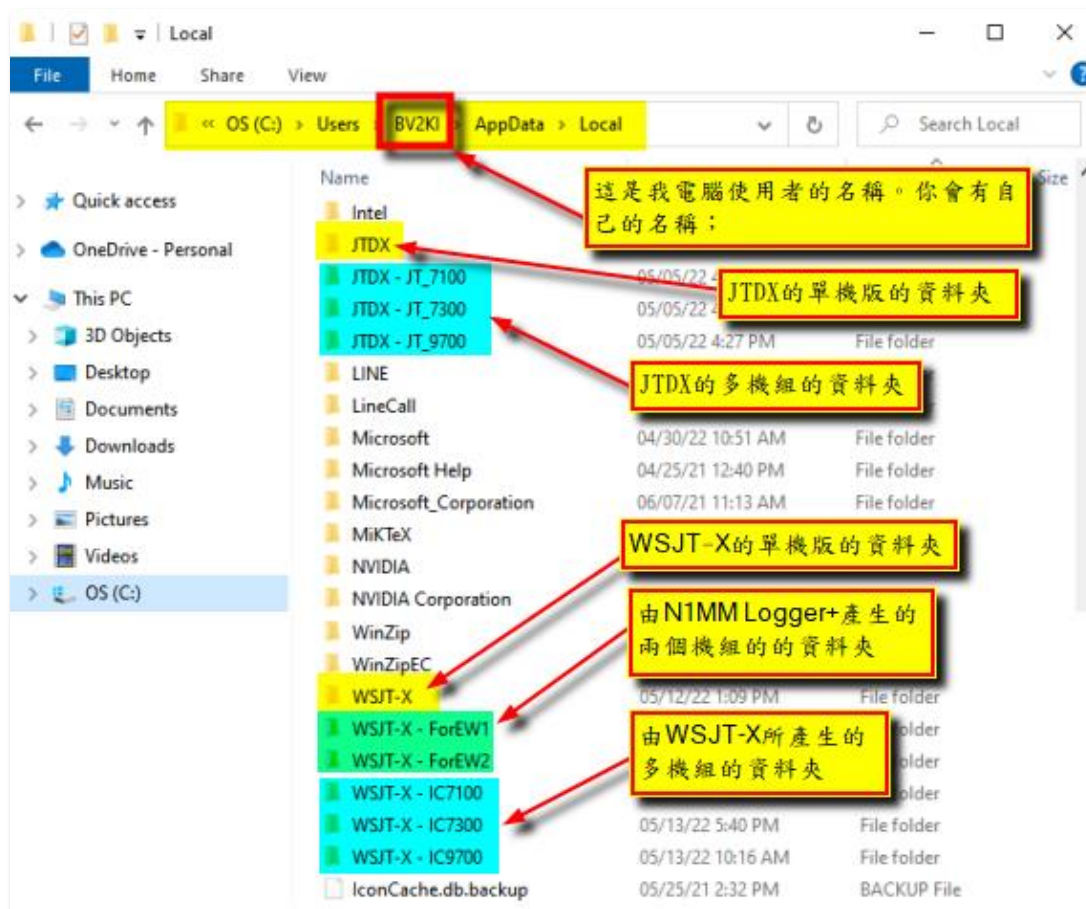
第二小節則是針對有使用過 Gridtracker 這套軟體的使用者來做設定的說明。因此、如果你是 Gridtracker 的愛好者，就請你直接跳到第二小節去做相關的設定，這個設定名稱我就稱之為「Gridtracker 版本」。

如你也有興趣了解「N1MM+的基本設定版本」，那就繼續看下去！

由於本文係以四個應用程式作為整合的對象；這四個 APP 如果在操作過程中，如果有一個 APP 因為不知原因造成當機或卡頓，可能會導致寶貴的 QSO 資料無法儲存至 Log4OM 之中。因此，我設計了一套備份方法，避免發生無法完整的儲存現象。這個方法其實也是非常簡單，就是去利用 Log4OM 中，有一個「ADIF Function 功能」。這個功能就是將所有可能用得到的 JT 所產生的 QSO 之 ADIF 檔案找出來、再與目前 Log4OM 已經儲存的 QSO 資料，做一比對、補充或更新。如此就不會漏掉任何一筆 QSO 資料了。

## C\_2\_7\_7、儲存 ADI 檔案的位置

**JT** 在操作過程中會將每一筆成功的 QSO 以 ADIF 的檔案形式儲存起來；在 **JT** 的環境下，各個 Radio(例如本文的 Radio\_1 或 Radio\_2)產生的檔案名稱都叫做「wsjtx\_log.adi」；這些「.adi 檔」、分別儲存在不同的資料夾裡面、請參看圖\_35 所示。



圖\_35 由 WSJT-X(或 JTDX)程式所產生的 ADIF 檔案儲存的資料夾位置

例如:本文舉例的兩個 Radios，其儲存的位置是分別放在圖\_35 的 WSJT-X-ForEW1 及 WSJT-X-ForEW2 的資料夾裡(記得嗎? 本文目前為止、所述的都是在講:N1MM+環境下操作)。其他如淺藍色者，則是各個機組自行產生的資料夾；這裡也包含了以 JTDX 的應用軟體所產生的資料在內。

請注意，所有淺藍色部分資料夾的名稱，其後綴名字皆以機組名稱、主要目的是易於識別。而 N1MM+的資料夾(如淺綠色者)，係以「ForEW1」及「ForEW2」當默認值。至於著黃色者，則是 **JT** 以執行「單機模式」時的儲存資料夾。



所謂「單機模式」、是指你第一次開始安裝 JT 後，直接去執行的當時的運行狀態；或者是指你目前只有一部機組在電腦上執行 JT 的狀態；就稱為「單機模式」。如果你有需要將兩部以上的機組同時在一部電腦上去執行 JT 時，這就是「多機種模式」，在 N1MM+應用程式中、因為只能同時運行兩部機組，故稱為「雙機」模式。

如果你有三個機組以上如何組合？請參閱這篇文章：

「[如何在單一電腦上同時操作兩個以上的 WSJT-X\(或 JTDx\)以及與其他軟體的整合設定](#)」(稱為「參考文件 1」)。甚至於你也可將 N1MM+的模式再加上另外一個「單機模式」做組合應用、也是可行的。

如何讓 N1MM+去取得這些由 JT 所產生的 log 資料？

這裡先以 JT 默認的 Reporting 設定做說明(如圖\_36)。

N1MM+的方法就是藉 2nd UDP 埠位址，來做為儲存 logger 的溝通管道。Radio\_1 接收到「2237」推播通知後，將 2nd UDP 埠位址、去取回 ADIF 資料(UDP=2240)，並將該筆 ADIF 檔案、放入 N1MM+的資料庫中，成為 N1MM+的最終 ADIF 檔案之一。同理，「2239」的通告觸發時，程式只會去「2239」的通告者處(Radio\_2)的資料庫去將 ADIF 檔案(UDP=2241)，決不至於跑到 Radio\_1 的資料庫去找 ADIF 檔案。

這說明了 2nd UDP 埠位址是可以任意給的。為了配合其他多機種的 2nd UDP 埠位址之相關設定，有益於辨識起見，我將此「便於辨識」的原則，援用在 N1MM+環境下的 JT 中的 2nd UDP 埠位址；例如：

將 Radio\_1 的「第二 IP 位址」設定為:2297；(望文生義：就是給 IC-9700 用的)

將 Radio\_2 的「第二 IP 位址」設定為:2273。(望文生義：就是給 IC-7300 用的)

這個「2297」及「2273」的運用原則，請參見「[參考文件 1](#)」所述的有關之命名詳細說明。換句話說：由「2333」改成「2297」或「2273」、或其他易於辨識的位址，甚至於兩個 Radios 都給予 UDP=2033，都是可行的。簡單起見，甚至於不管是哪一個機組，全部都以 2nd UDP=2033 去做設定，也是可行的；因此，從此時開始，如需要以 UDP 之送出或接收者，皆以 UDP=2033 為主，以期單純化。

有關 NMM+最後收集到 QSO 的 Log，在比賽結束、並且完成 N1MM+內部之成績的檢核及計算後，你當然可以遵循比賽規則、送出以 ADIF(或 Carblillo)的檔案給賽事舉辦單位作為參賽成績。如果你不是處於比賽模式時，而且是在 N1MM+環境下仍然操作 JT 時，N1MM+的資料庫仍然可以以其本身的 ADIF 的格式、Export 輸出到你喜用的 logger 軟體去。

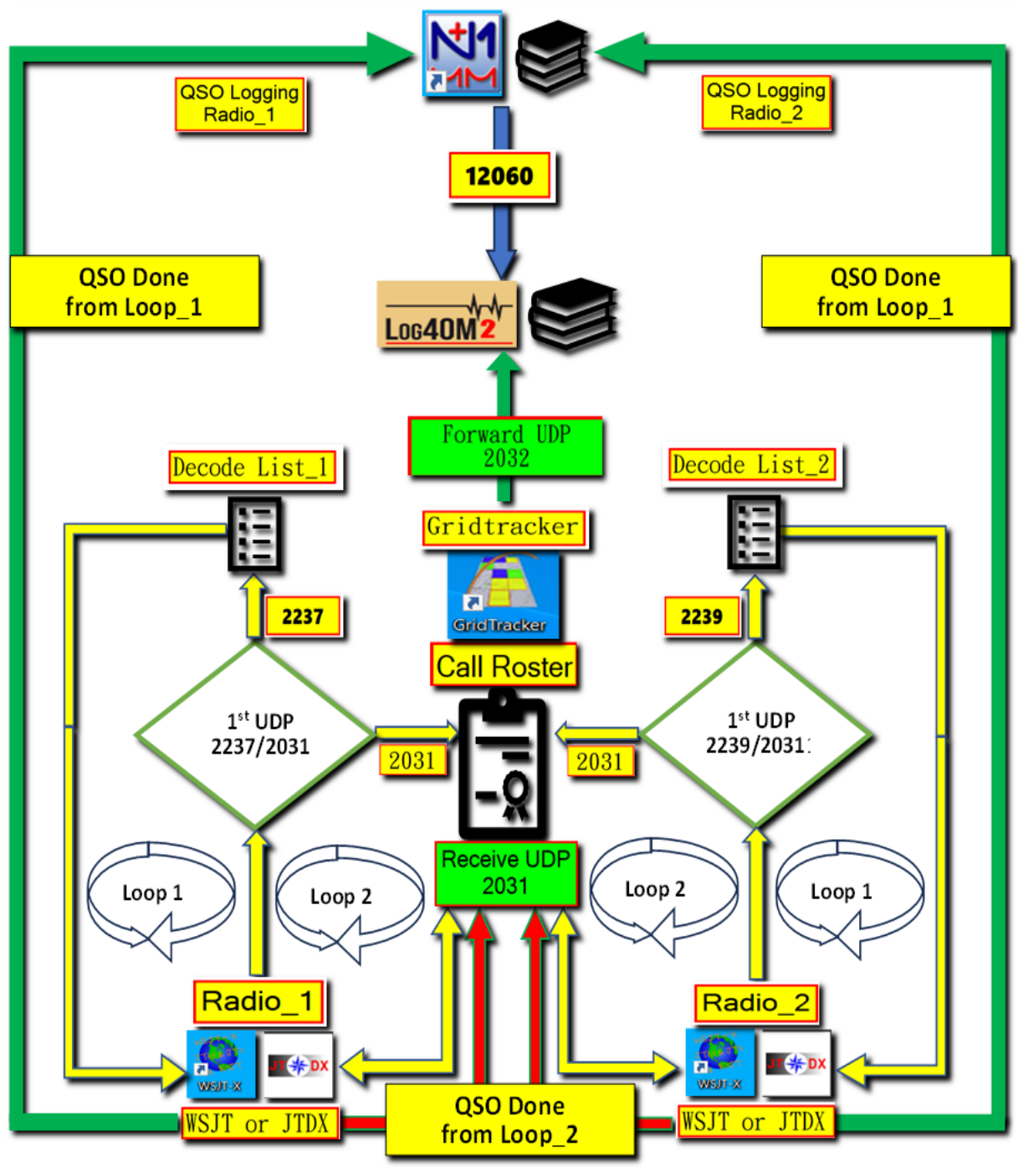


這一小節所述的檔案位置，目的是要給咱們一個概念，知道檔案儲存位置以及如何處理 2nd UDP 埠位址後，你就可以在其他的應用軟體(如 Log4OM、HamRadioDelux 等)去做出對應的設定。

有關 Log4OM 的設定中，涉及本節內容者，請先到這裡「Log4OM的ADIF Function功能的設定」看看。

### C\_3、四個程式[Gridtracker/Log40M/WSJT-X(或 JTDX)/N1MM+]的關係

我畫了一張有關四個程式[Gridtracker/Log40M/WSJT-X(或 JTDX)/N1MM+]的作業流程圖，請看圖\_36。



圖\_36 四個程式(N1MM/Gridtracker/WSJT/L40M)之間通告及資料傳輸的流程圖

圖\_36 中有四個迴圈(Loop 1 及 Loop 2 各有兩個)，Loop 1 是在 1st UDP 設成 2237(Radio\_1)或 2239(Radio\_2)後，採用了 N1MM+的 Decode List 功能。Loop 2 則是將 1st UDP 設定成 2031 後，採用 Gridtracker 之 Call Roster 功能。

以 Radio\_1 之 Loop 1 來說(圖\_36 之左半部)：**JT** 配合 N1MM+時，是利用 N1MM+ 默認的 Decode List 功能，一旦完成 QSO，Radio\_1 會透過 UDP=2237 去儲存 ADIF 檔案至 N1MM+的 DataBase 之中。同樣的、Radio\_2(圖\_36 之右半部)也是採相同方式以 UDP=2239 去儲存 ADIF 檔案至 N1MM+的 DataBase 之中。

N1MM+本身有個對外「推播」功能，就如同圖\_9 所示，該圖中第三選項的「Contacts」就是 ADIF 檔案藉著 UDP=12060 推送出去；在 Log4OM 設定一個「InBound」的 UDP=12060 時、Log4OM 就可由 N1MM+取得同樣的一筆 ADIF 資料。

至於圖\_36 之中央部位，屬於 Loop 2 的範圍，此一迴圈的動作，是將 1st UDP 設定成 2031，這是採用了 Gridtracker 的功能，一旦 **JT** 完成 QSO 後，Gridtracker 接收到 **JT** 之存檔通知、同時會藉著 Gridtracker 的 2nd UDP=2032 之推播出去；此時 Log4OM APP(已預先設定好 UDP=2032 為存檔的管道)就會去儲存 ADIF 檔案至 Log4OM 的 DataBase 之中。

因此、由 **JT** 應用程式完成的 QSO，如果是在 N1MM+環境下作業時，N1MM+及 Log4OM 的資料庫都會取得一筆 ADIF 檔案；如果不處於 N1MM+環境下的 **JT** 應用程式運行時，ADIF 的產生流程、僅限於 Loop 2 者，此時之運行狀況就稱之為「常規操作」，就是本文之 C\_8、操作常規 WSJT-X(或 JTDX)的 Radio 設定至 C\_9、操作常規 WSJT-X(或 JTDX)的 Reporting 設定節所述。

根據前一節所述，N1MM+的解碼清單功能，如果你認為很給力、想繼續延用，那你就將 1st UDP 設為「2237」給 Radio\_1 用、Radio\_2 則設為「2239」這兩個默認值。

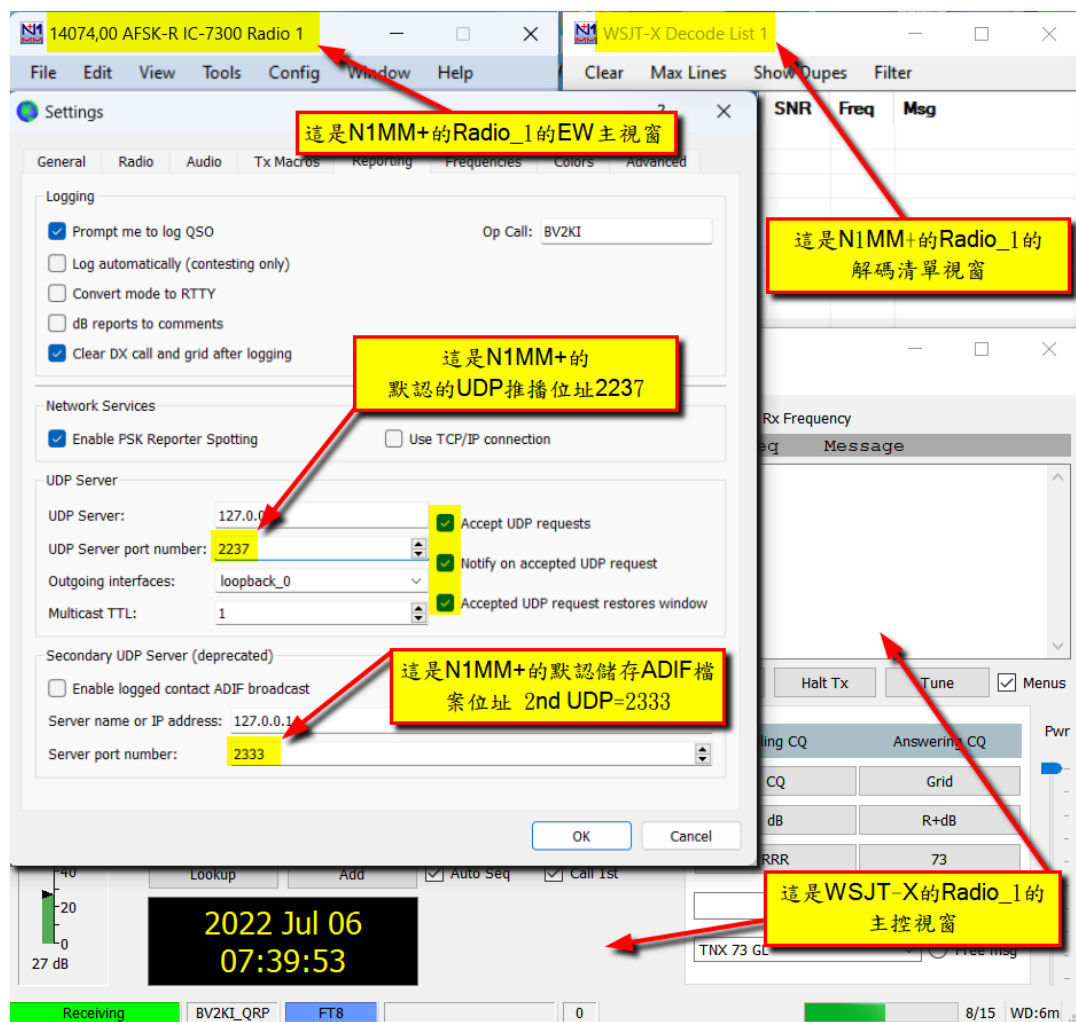
如果你認為有其他功能較強的 APP 者，例如 Gridtracker 或 [JT\\_Alerts](#) 之類的應用程式，那就將 **JT** 中的 1st UDP 設為 2031；此類之 APP 的「推播 UDP」的埠位址，UDP 就去設為 2032。

行文至此，我已經將所有在 N1MM+環境下的 **JT** 設定方式，說明完畢、如果照著本文「C-2 WSJT-X(或 JTDX)中的設定」這個章節裡的編號 1 至編號 6 的六個小節、再加上第 7 小節 Reporting 的第一及第二部分(總共 7 小節)，逐步地去設定(照著圖\_2 至圖\_9 去做即可)，應屬無疑的。也就是說：這 7 小節的設定參數，就是所謂的：**N1MM+的基本設定版本**。

## C\_4、配合 N1MM+ 的基本設定版本

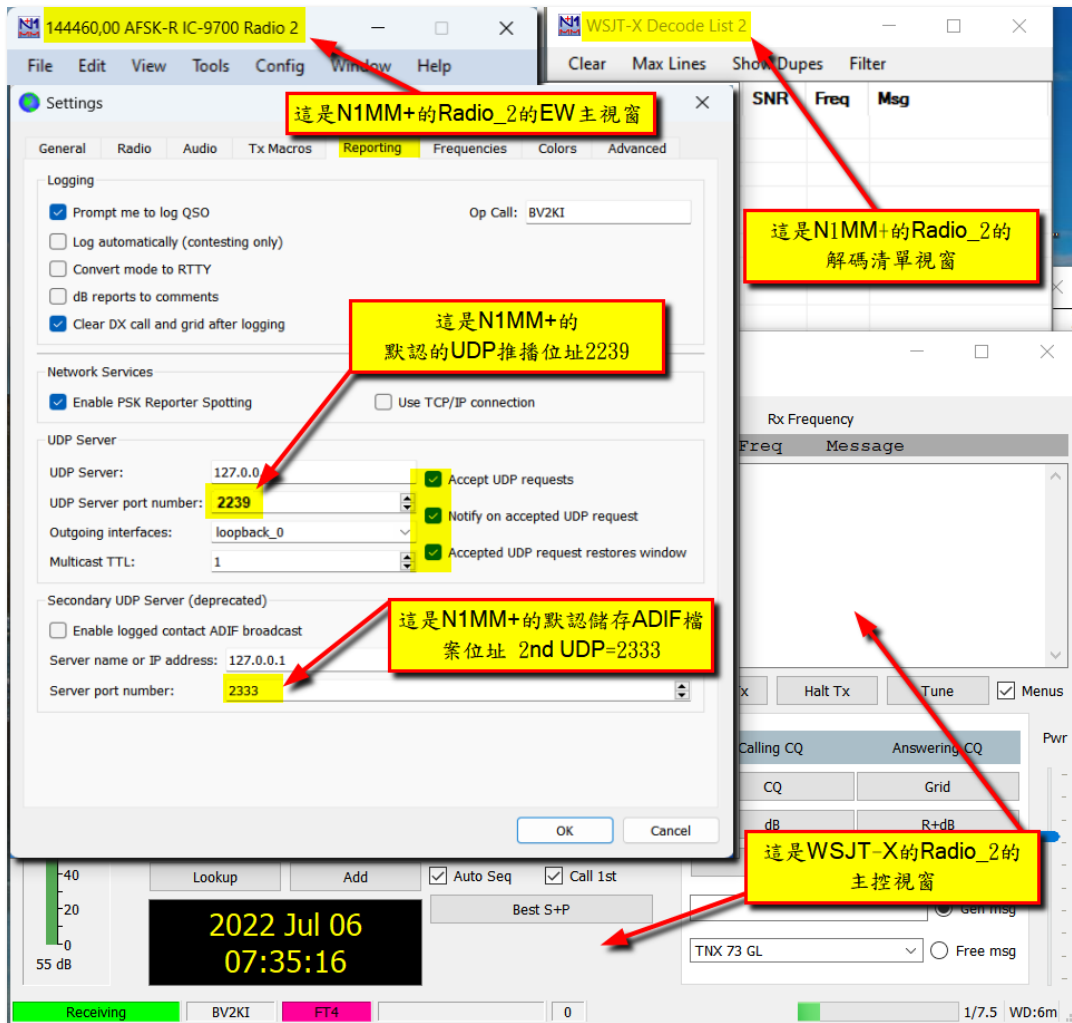
請依照前文所述:從「C-2 WSJT-X(或 JTDX)中的設定」章節起,所說的7個小節的資料、逐步的去做「WSJT-X 的基本設定」。(按照圖\_2 至圖\_9 去設定即可)

接著,最後一小節的動作,就按照如圖\_37 及圖\_38 所示的設定即可。:



圖\_37 WSJT-X(或 JTDX) 中 Radio\_1 的 Reporting 設定(N1MM+基本設定版本)

UDP=2337 係 N1MM+之 Radoio\_1 的 Code List 功能默認值;這個默認值可以透過 N1MM+之虛擬 UDP=2240、將 ADIF 送進 N1MM+的 Log 裡;所以這個基本版就仍以原設計之設定為準。



圖\_38 WSJT-X(或 JTDX) 中 Radio\_2 的 Reporting 設定(N1MM+基本設定版本)

圖\_38 係針對 Radio\_2 做設定的；你只要將 Second UDP 改為 2239 即可。其餘者仍然依照圖\_37 所述之設定即可。

以上、就是「N1MM+的基本設定版本」的全部過程。

#### C\_4\_1、測試「N1MM+的基本設定版本」：

經過以上的九個小節的設定以及 N1MM+ 的實際布置(S02R)，此時的 N1MM+ 接著就可以提供後續整個 **JT** 的完整操作。此刻、你可以將 Radio\_1 去到一個傳播比較好的波段（或者打個電話，邀請友人前來試作 QS0），靜觀片刻後、應該可以看到解碼信息會出現在 Decode List 1 中。同樣的，將 Radio\_2 播到某一有人工作的波段，這也會將 Radio\_2 的解碼信息呈現在 Decode List 2 視窗中。

這樣的情況表示：在 N1MM+ 環境下、**JT** 已經是完整的啟動了，接下來，咱們就試著做 Log 的登錄動作，看看是否成功？

你可以假裝剛剛做完一個 QS0，在 Radio\_1 及 Radio\_2 的 **JT** 模式下，以人工方式去各做一個「假的 QS0 Log」登錄；當然，DX Call 的方框裡[這裡講的 DX Call 方框是指 **JT** 的，不是指 EW1(或 EW2)的方框唷!] 隨便填一個呼號就好，例如「TE1ST」及「TE2ST」，其餘的資料都跳過、故意不填、讓他們都是空白資料呈現、按「OK」鍵(不要理會跳出來的警語視窗(如圖\_68)，繼續按「OK」鍵就可往下進行)，看看這兩筆資料是否會放進 N1MM+ 的 Log 裡？

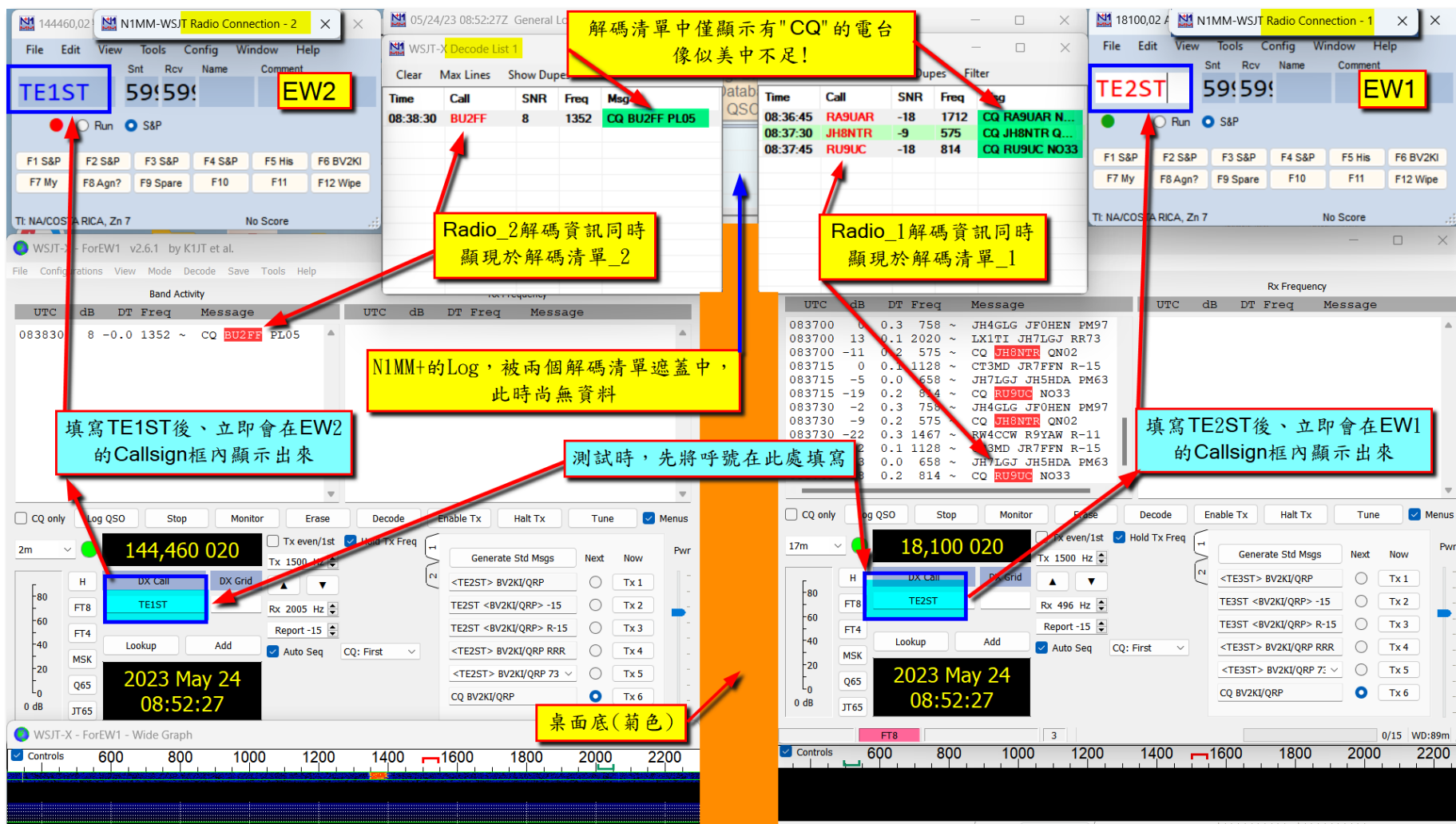
如果你的 N1MM+ 的 LOG 裡、新增了「TE1ST」及「TE2ST」，這兩筆「假的紀錄」，表示完全成功啦！恭喜您了！老爺！

這個初步測試，見下面的圖\_39 至圖\_41，很容易理解的。

此時，如果你的 Log40M 已經有了適當的設定，在做 **JT** 模式之 QS0 且登錄至 N1MM+ 後、約 3 秒之後，你的 Log40M 的 Database 中，應該會增添一筆 QS0 的紀錄。

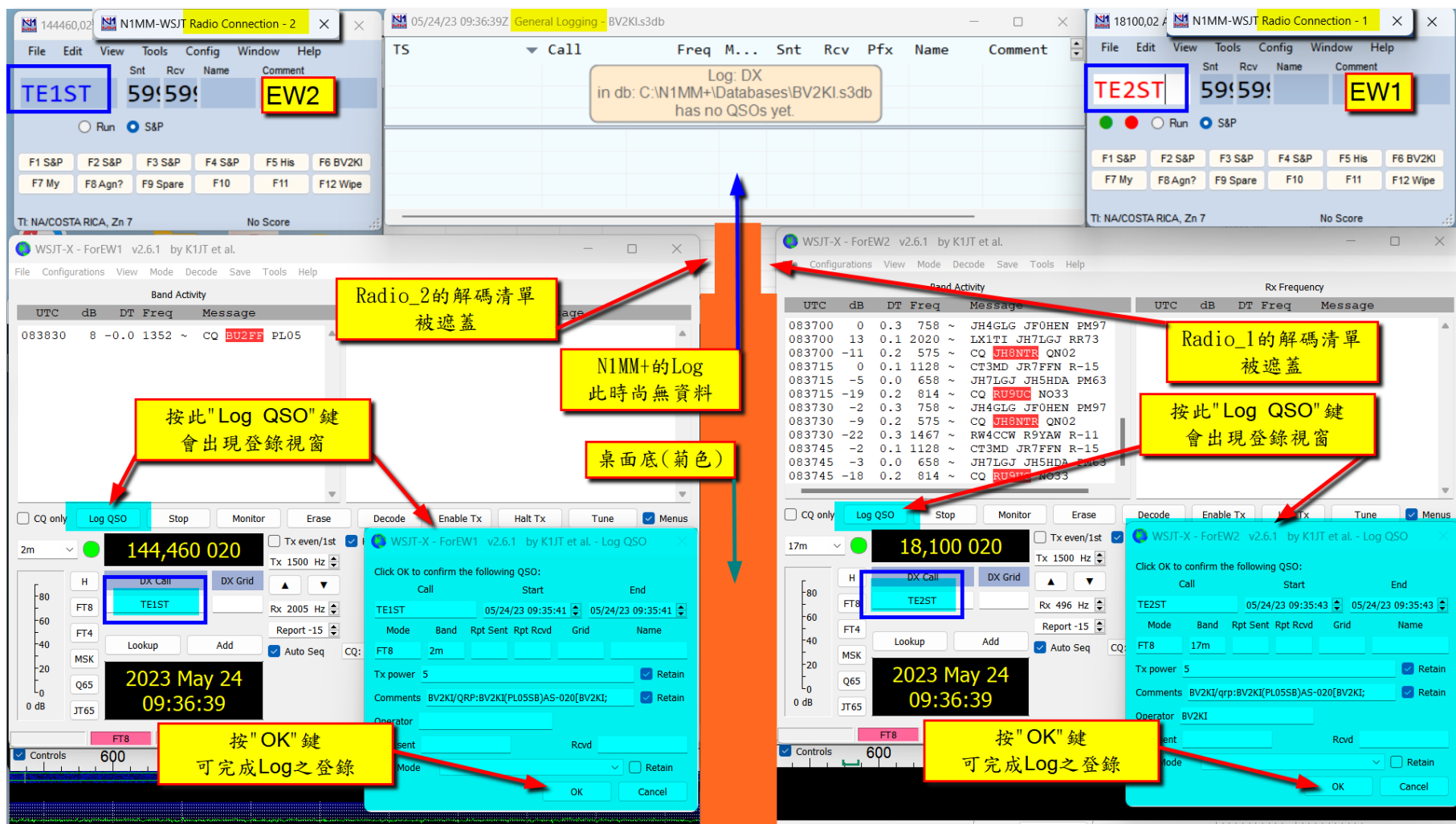
如果以上兩個動作並無所願，請回頭仔仔細細檢查一下、前面的設定是否正確？





圖\_39 基本設定版本測試\_1

分別在 Radio\_2 之主視窗的 DX Call 填寫「TE1ST」以及 Radio\_1 之主視窗的 DX Call 填寫「TE2ST」。  
然後分別去 Radio\_1 及 Radio\_2 主視窗中的「Log QSO」按鈕，去做儲存 QSO 的動作(如圖\_40 所示)。



圖\_40 基本設定版本測試\_2

按 Log QSO 視窗之「OK」鍵，儲存兩個 QSO；其結果請見圖\_41。

The screenshot displays the N1MM-WSJT software interface with two radio connections, Radio\_1 and Radio\_2, and their associated logs and decode lists.

**Radio\_2 (Left):** The main window shows a frequency of 144,460,020. The log window displays two QSO test records:

TS	Call	Freq	M...
2023-05-24 09:35	TE1ST	144461,52	FT8
2023-05-24 09:35	TE2ST	18101,52	FT8

A yellow box highlights the text: "N1MM+的Log, 已有2個QSO測試紀錄放了進來" (N1MM+'s Log, 2 QSO test records have been entered).

The decode window for Radio\_2 shows a list of decoded messages, including:

UTC	dB	DT	Freq	Message
083700	0	0.3	758	JH4GLG JF0HEN PM97
083700	13	0.1	2020	LX1TI JH7LGJ RR73
083700	-11	0.2	575	CQ JH8NTR QN02
083715	0	0.1	1128	CT3MD JR7FFN R-15
083715	-5	0.0	658	JH7LGJ JH5HDA PM63
083715	-19	0.2	814	CQ RU9UC NO33
083730	-2	0.3	758	JH4GLG JF0HEN PM97
083730	-9	0.2	575	CQ JH8NTR QN02
083730	-22	0.3	1467	RW4CCW R9YAW R-11
083745	-2	0.1	1128	CT3MD JR7FFN R-15
083745	-3	0.0	658	JH7LGJ JH5HDA PM63
083745	-18	0.2	814	CQ RU9UC NO33

A yellow box highlights the text: "Radio\_2的解碼清單被遮蓋" (Radio\_2's decode list is obscured).

**Radio\_1 (Right):** The main window shows a frequency of 18,100,020. The log window displays two QSO test records:

TS	Call	Freq	M...
2023-05-24 09:35	TE1ST	144461,52	FT8
2023-05-24 09:35	TE2ST	18101,52	FT8

A yellow box highlights the text: "Radio\_1的解碼清單" (Radio\_1's decode list).

The decode window for Radio\_1 shows a list of decoded messages, including:

UTC	dB	DT	Freq	Message
083700	0	0.3	758	JH4GLG JF0HEN PM97
083700	13	0.1	2020	LX1TI JH7LGJ RR73
083700	-11	0.2	575	CQ JH8NTR QN02
083715	0	0.1	1128	CT3MD JR7FFN R-15
083715	-5	0.0	658	JH7LGJ JH5HDA PM63
083715	-19	0.2	814	CQ RU9UC NO33
083730	-2	0.3	758	JH4GLG JF0HEN PM97
083730	-9	0.2	575	CQ JH8NTR QN02
083730	-22	0.3	1467	RW4CCW R9YAW R-11
083745	-2	0.1	1128	CT3MD JR7FFN R-15
083745	-3	0.0	658	JH7LGJ JH5HDA PM63
083745	-18	0.2	814	CQ RU9UC NO33

圖\_41 基本設定版本測試\_3

N1MM+的Log裡, 已經有兩個測試檔案了。

表示設定完成, 測試成功!

## C-4\_2、N1MM+的兩個機組實際作業紀錄

我這裡有兩個機組的實際作業，以圖片說明如下，供作參考：這張圖\_42 是個「N1MM+基本設定版本」的標準桌面。

The screenshot displays the N1MM+ software interface with two radio setups, Radio 1 and Radio 2, side-by-side. The interface includes various windows and controls, with several annotations in yellow boxes and red arrows pointing to specific elements.

**Radio 1 (Right):**

- Radio\_1的解碼清單 (Radio 1 Decode List):** A window showing a list of decoded signals. The first entry is highlighted: 095600 -8 0.1 1254 ~ CQ 9M2DA OJ12.
- Radio\_1的搭接視窗 (Radio 1 Connection Window):** A window showing the connection status for Radio 1. It includes a "Close WSJT-XJTDX first!" button and a "Debug" checkbox.
- Radio\_1正在解碼中 (Radio 1 is decoding):** A yellow box annotation pointing to the highlighted entry in the Radio 1 Decode List.

**Radio 2 (Left):**

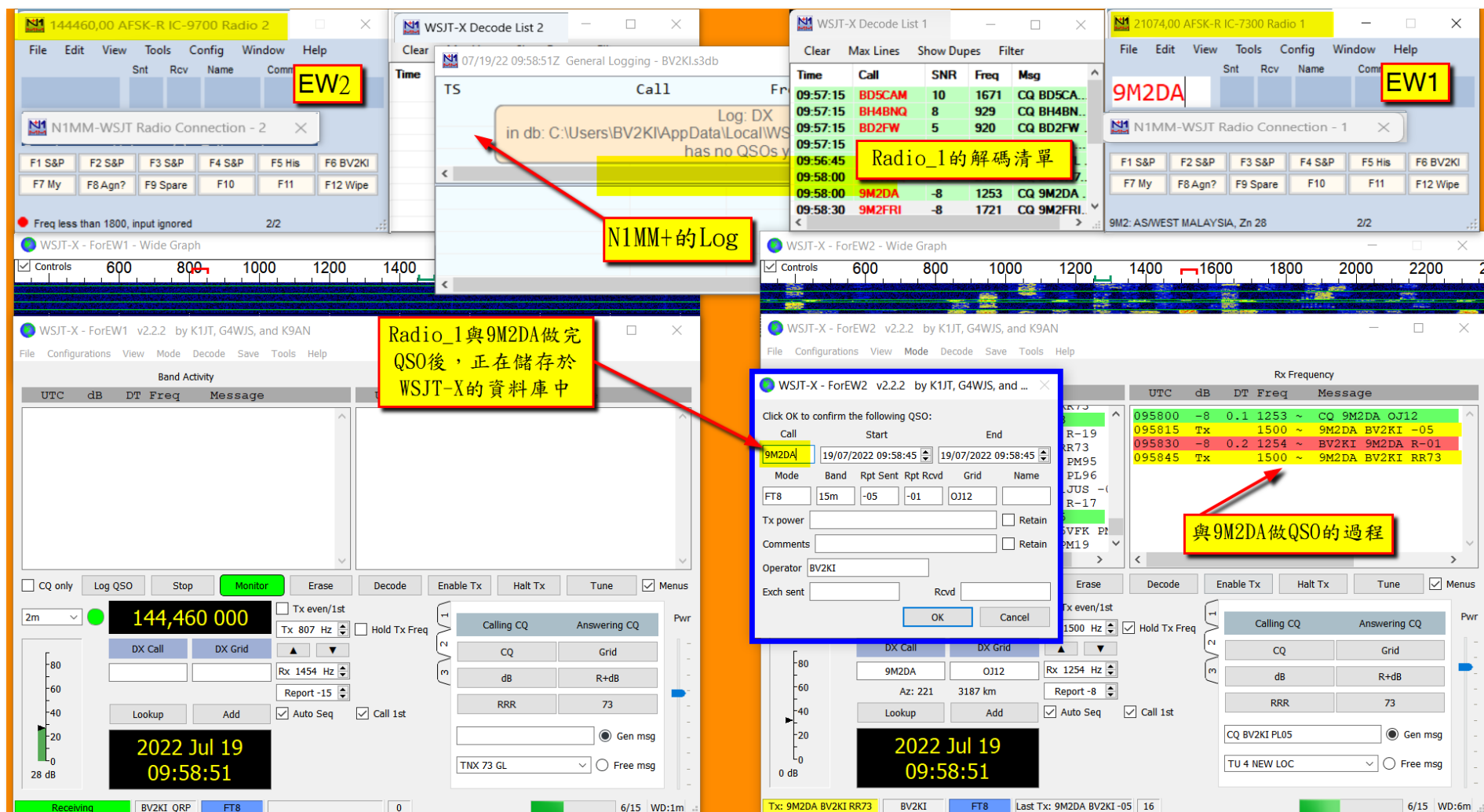
- Radio\_2的解碼清單 (Radio 2 Decode List):** A window showing a list of decoded signals. The first entry is highlighted: 095600 -8 0.1 1254 ~ CQ 9M2DA OJ12.
- Radio\_2的搭接視窗 (Radio 2 Connection Window):** A window showing the connection status for Radio 2. It includes a "Close WSJT-XJTDX first!" button and a "Debug" checkbox.
- Radio\_2正在解碼中 (Radio 2 is decoding):** A yellow box annotation pointing to the highlighted entry in the Radio 2 Decode List.

**Other Annotations:**

- N1MM+的Log (N1MM+ Log):** A yellow box annotation pointing to the "Log" button in the bottom right corner.
- Radio\_2的解碼清單被N1MM+的LOG視窗遮住 (Radio 2 Decode List is obscured by N1MM+ Log window):** A yellow box annotation pointing to the "Log" button.
- Radio\_1的搭接視窗可以捲成小一點不占空間 (Radio 1 Connection Window can be scrolled to be smaller and not take up space):** A yellow box annotation pointing to the "Close WSJT-XJTDX first!" button.

圖\_42 Radio\_1\_正在執行解碼中(N1MM+基本設定版本)

JT 模式的解碼資訊會出現在 Radio\_1 的解碼清單裡(只有喊 CQ 的，才能「上榜」)；當我在 Radio\_2 的解碼清單\_1 上的 9M2DA 那一行、輕輕左鍵按它一下，9M2DA 的呼號就會出現在 EW1 的 Call 方框裡。接著 WSJT-X 就自動地去呼叫 9M2DA。圖\_43 可以看得到該 QSO 的過程記錄(請看圖\_42 的 WSJT-X 主視窗的右方部位)。



圖\_43 Radio\_2 與 9M2DA 做完 QSO 後，將 QSO 資料儲存起來(N1MM+基本設定版本)

剛剛跟 9M2DA 做完 QSO 並正在做儲存的動作。請注意 Radio\_1 的解碼清單中，又多出了好幾行資料，而且看到的解碼、好像都是只有「喊 CQ 的」。這個屬於 WSJT-X 的 Logging 儲存的視窗，是短暫性的。置放在哪裡，都無所謂。但是、使用 Gridtracker 時，就有講究的必要。

注意:此時，N1MM+的 LOG 目前還是沒有東西儲存的跡象。



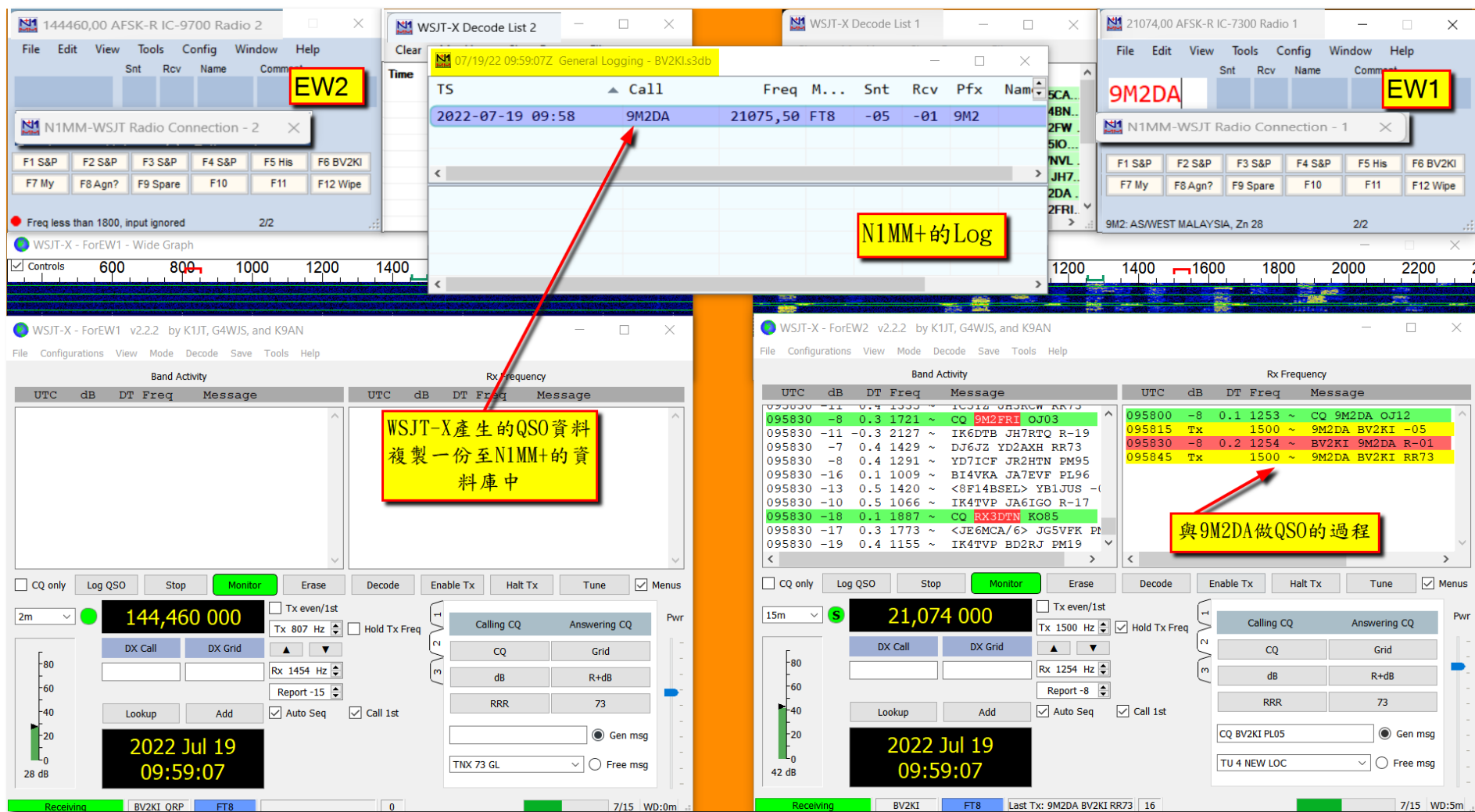


圖 44 Radio\_1 與 9M2DA 做完 QSO 並將 QSO 資料儲存後，複製一份存進 N1MM+ 的 LOG 中 (N1MM+ 基本設定版本)

這張圖\_44，是讓你看，完成 QSO 後、且 WSJT-X 自身的存檔動作完成；如果 1st UDP 設定為「2237」時(為何是 2237 ?)

提示:此時係在 Radio\_1 運作中，N1MM+ 就會自動的將 QSO 的 ADIF 檔案複製一份、裝進自己口袋裡。



打了一個電話給 BX2AB，邀請他到 2 米波來測試一下。圖\_45 顯示了跟 BX2AB 做了 QSO 過程的紀錄以及 WSJT-X 自身的存檔截圖。

The screenshot displays the WSJT-X interface with two radio connections, EW1 and EW2. The main window shows the 'Radio\_1' (EW1) interface, which includes a 'Band Activity' table and a 'QSO Log' window. The 'Band Activity' table lists various QSOs, including one with BX2AB. The 'QSO Log' window shows a QSO with BX2AB on 2m FT8, dated 19/07/2022 10:08:15 to 10:09:15. The 'Radio\_2' (EW2) interface is also visible, showing a 'Band Activity' table and a 'QSO Log' window. The 'Band Activity' table for Radio\_2 shows a QSO with BX2AB on 2m FT8, dated 19/07/2022 10:08:15 to 10:09:15. The 'QSO Log' window for Radio\_2 shows a QSO with BX2AB on 2m FT8, dated 19/07/2022 10:08:15 to 10:09:15. The 'QSO Log' window for Radio\_2 also shows a QSO with BX2AB on 2m FT8, dated 19/07/2022 10:08:15 to 10:09:15. The 'QSO Log' window for Radio\_2 also shows a QSO with BX2AB on 2m FT8, dated 19/07/2022 10:08:15 to 10:09:15.

EW2

Radio\_2的解碼清單

N1MM+的Log

EW1

Radio\_1的主視窗

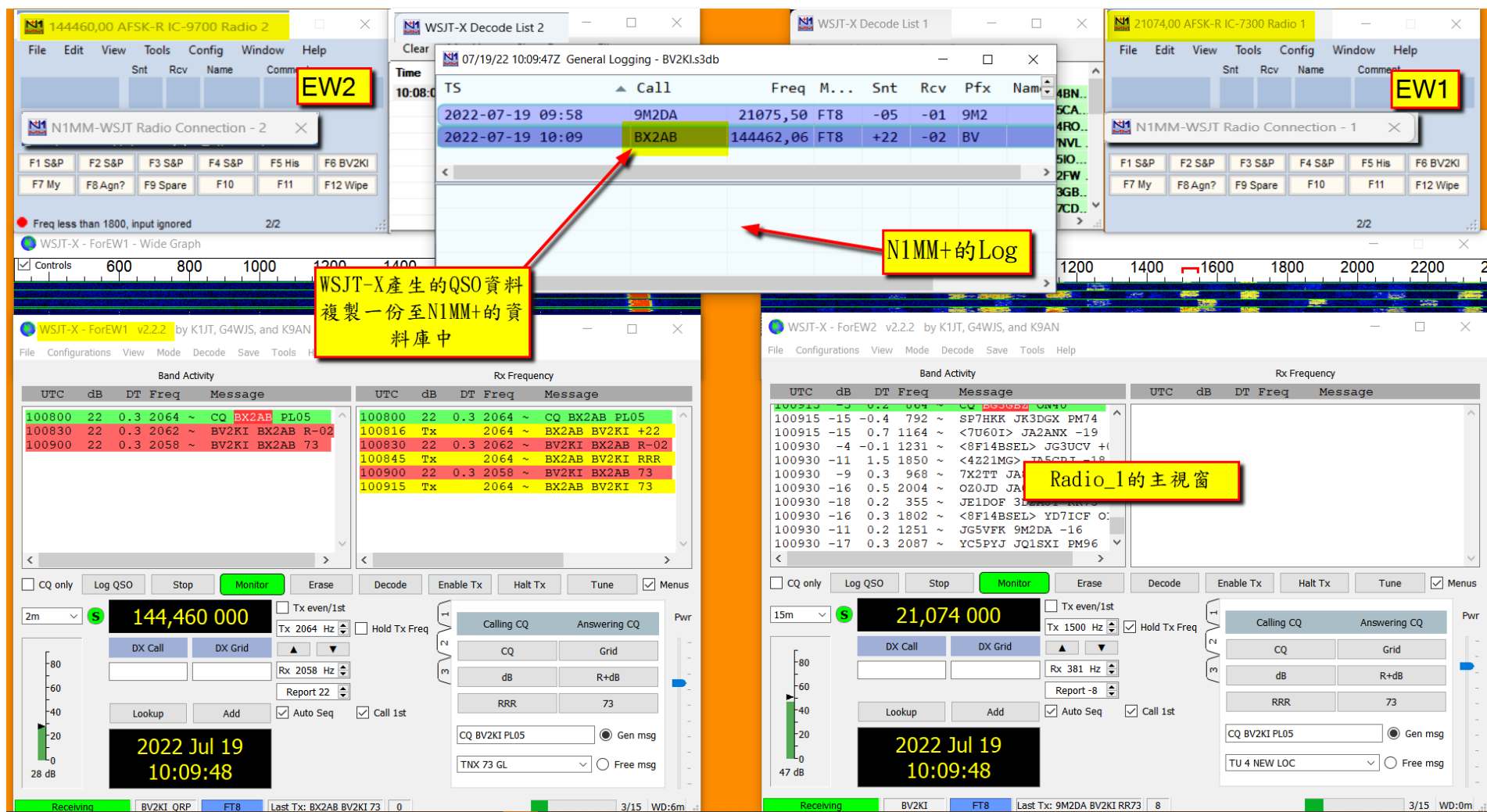
與BX2AB做QSO的過程

Radio\_2與BX2AB做完QSO後，正在儲存於WSJT-X的資料庫中

圖\_45 Radio\_2 剛剛跟 BX2AB 做完 QSO，正在將 QSO 的資料儲存到 Radio\_1 的 WSJT-X 的資料庫去(N1MM+基本設定版本)。

目前，進程只到 WSJT-X 的儲存作業；N1MM+的資料庫也還是維持 EW1 的歷史資料。

繼續看下去！



圖\_46 Radio\_2 與 BX2AB 做完 QSO 並將 QSO 資料儲存後，複製一份存進 N1MM+ 的 LOG 中(N1MM+基本設定版本)。

這裡，你可以看到 BX2AB 的 QSO 資料已經被拉到 N1MM+ 的資料庫裡！

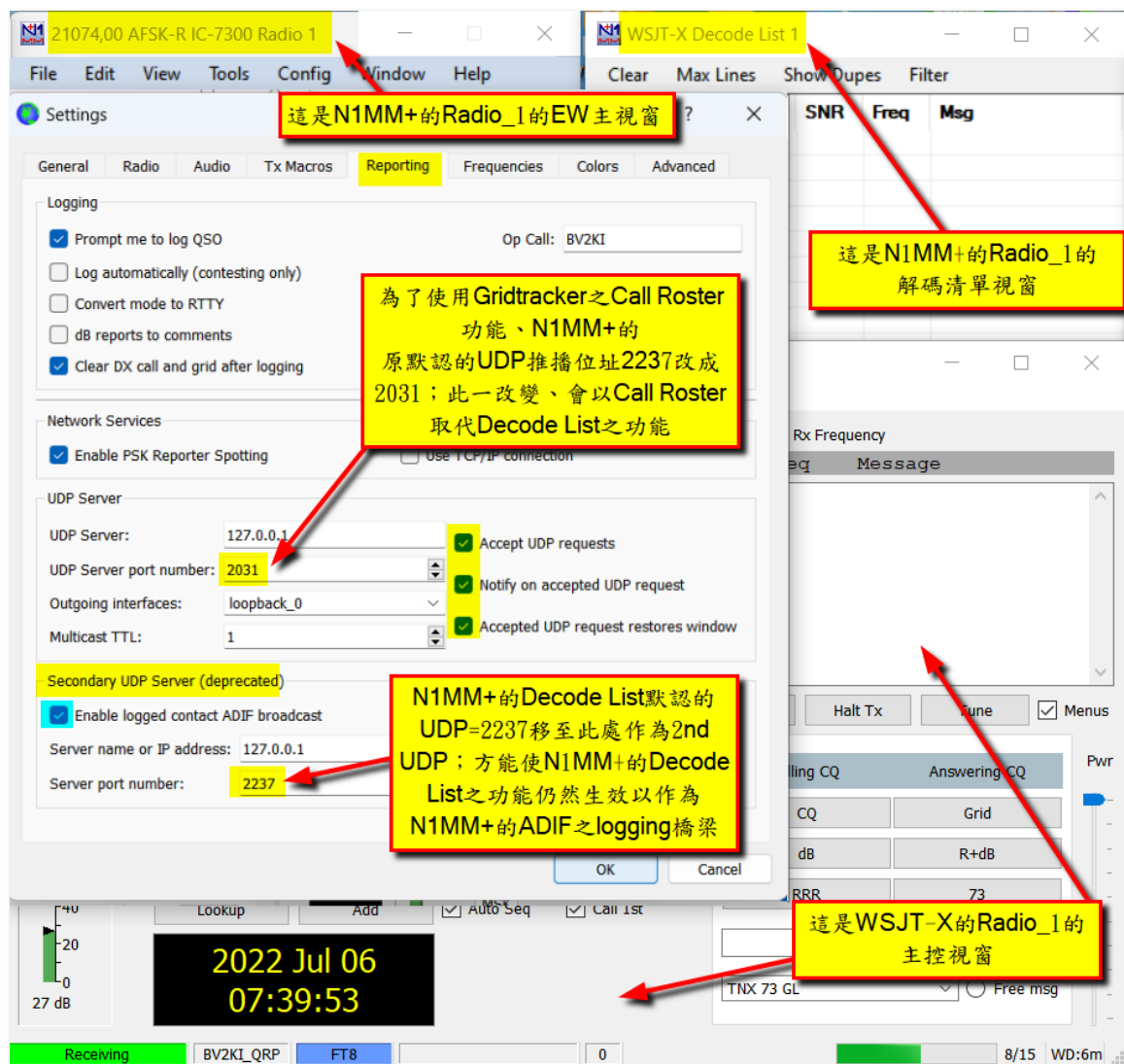
至此、第一小節的「N1MM+的原版設定」，已經證實 OK！

放鞭炮、慶祝一下！

## C\_5、N1MM+環境下、配合 Gridtracker 版本

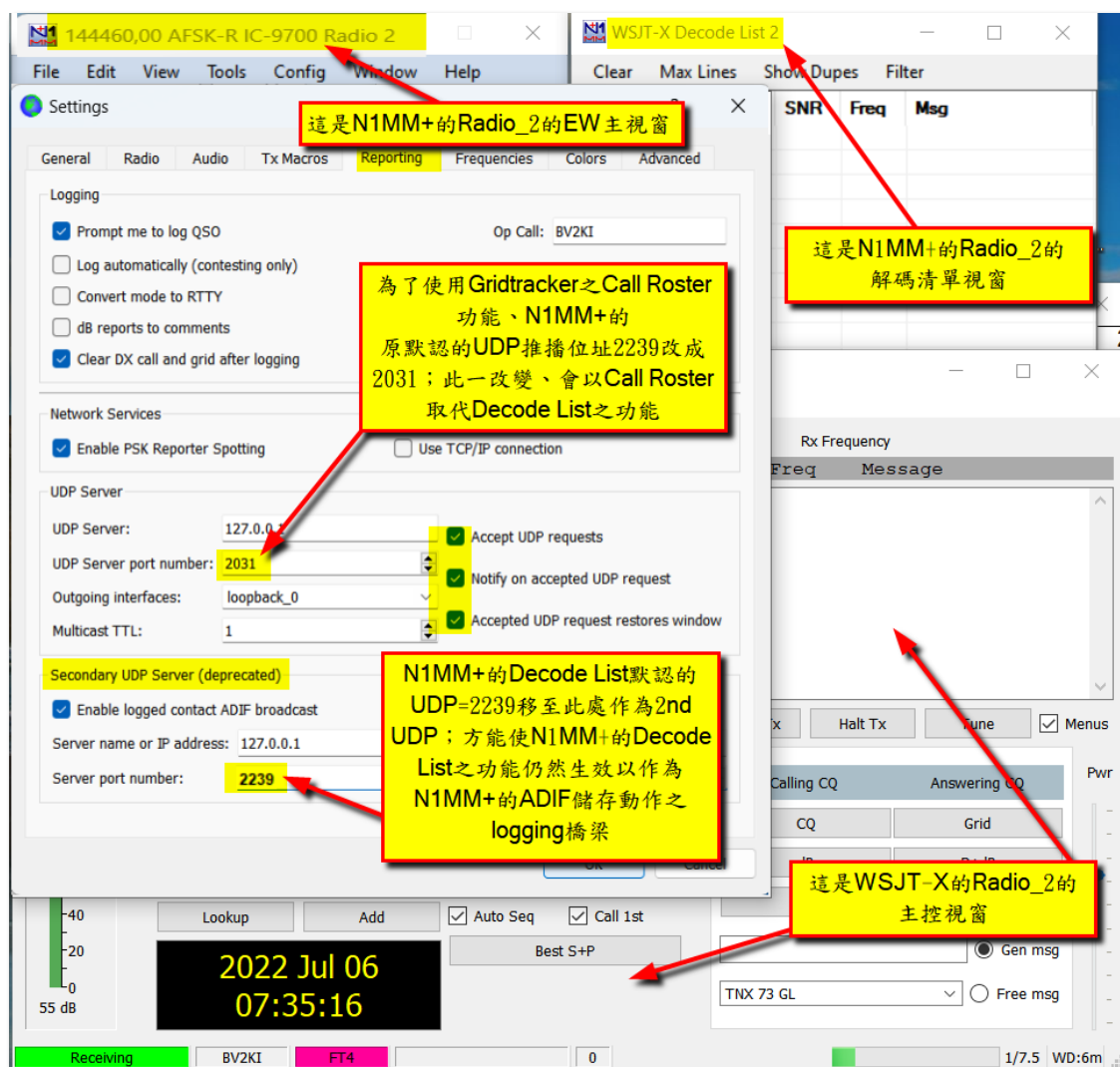
請依照前文所述:從第「C-2 WSJT-X(或 JTDX)中的設定」章節起，所說的九個小節的資料、逐步的去做「WSJT-X 的基本設定」。(按照圖\_2 至圖\_9 去設定即可)

接著，最後一小節的動作，就按照如圖\_47 及圖\_48 所示的設定即可。:



圖\_47 WSJT-X(或 JTDX) 中 Radio\_1 的 Reporting 設定(Gridtracker 版本)

在 1ST UDP server port number 填寫「2031」，將原有 N1MM+的默認值「2237」，移至 2nd UDP Server 處；記得要勾選 Enable logged contact ADIF broadcast；才會有起了「通告」給 N1MM+的作用。



圖\_48 WSJT-X(或 JTDX)中 Radio\_2 的 Reporting 設定(Gridtracker 版本)

同理，在 1ST UDP server port number 填寫「2031」，將原有 N1MM+的默認值「2239」，移至 2nd UDP 處；

至此、「Gridtracker 版本設定」，全部完成。

## C\_5\_1、Gridtracker 的相關設定：

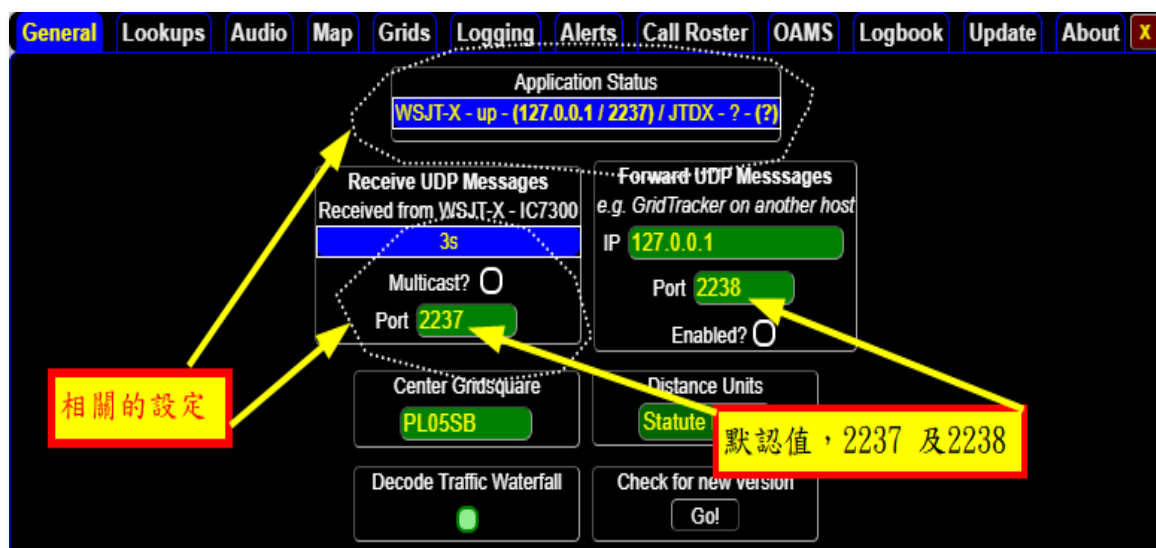
依據這個「Gridtracker 版本」的文意，我們需要將 Gridtracker 的應用軟體拉進來，讓 Call Roster 的功能取代 NIMM+中之「笨拙的 Decode List」；所以我就將 Gridtracker 開啟。接著我們就去 Gridtracker 的 Setting 設定工具按鈕組(如圖\_49)



圖\_49 Gridtracker 的「設定工具按鈕組」

在圖\_49 之「設定工具按鈕」，以左鍵按一下，進入：

Settings → General 就可看到如下圖\_50 的內容。這是原始的 Gridtracker 之 General 默認畫面：

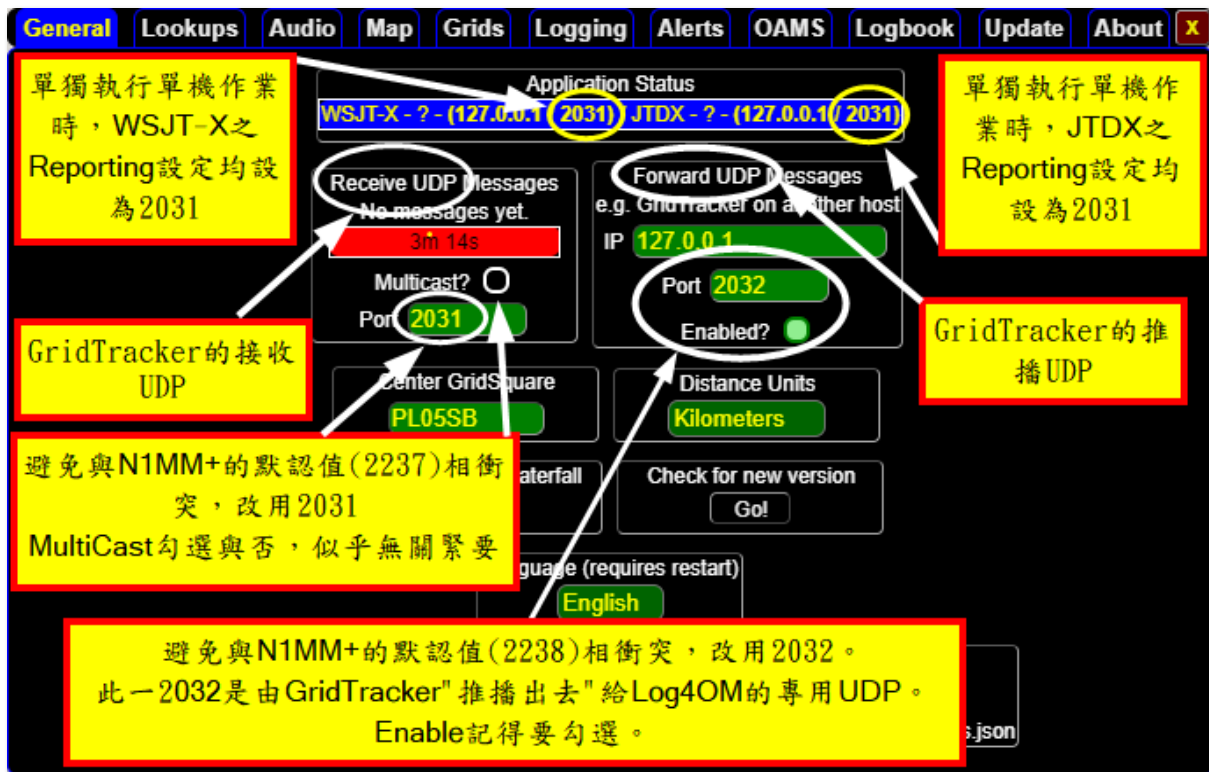


圖\_50 Gridtracker UDP 默認設定內容

在此畫面內容處頂端，Application Status 顯示出：

WSJT-X - ? - (127.0.0.1 / 2237) / JTDX - ? - (?) 的這行字；表示

Gridtracker 與 WSJT-X 的關係，已經互通款曲，並顯示在「Application Status 應用程式狀態」上。其後半段表示：如果你的應用程式是 JTDX 者，也會顯示出來；但，目前的電腦是沒有執行過 JTDX 程式，所以只顯示兩個「？」問號。請注意：Gridtracker 的接收 Received 的 UDP 默認值為 2237、推播 Forward 的 UDP 默認值為 2238。



圖\_51 將「接收 UDP 訊息」之默認位址埠改為「2031」；Forward UDP 改為 2032

圖\_50 原 Gridtracker 的接收 UDP 訊息之默認位址埠為「2237」。我們將其設定改為如圖\_51 的「2031」。

圖\_50 原 Gridtracker 的推播 UDP 訊息之默認位址埠為「2238」。我們將其設定改為如圖\_51 的「2032」、且記得要勾選 2032 底下的啟動 Enable 選鈕。

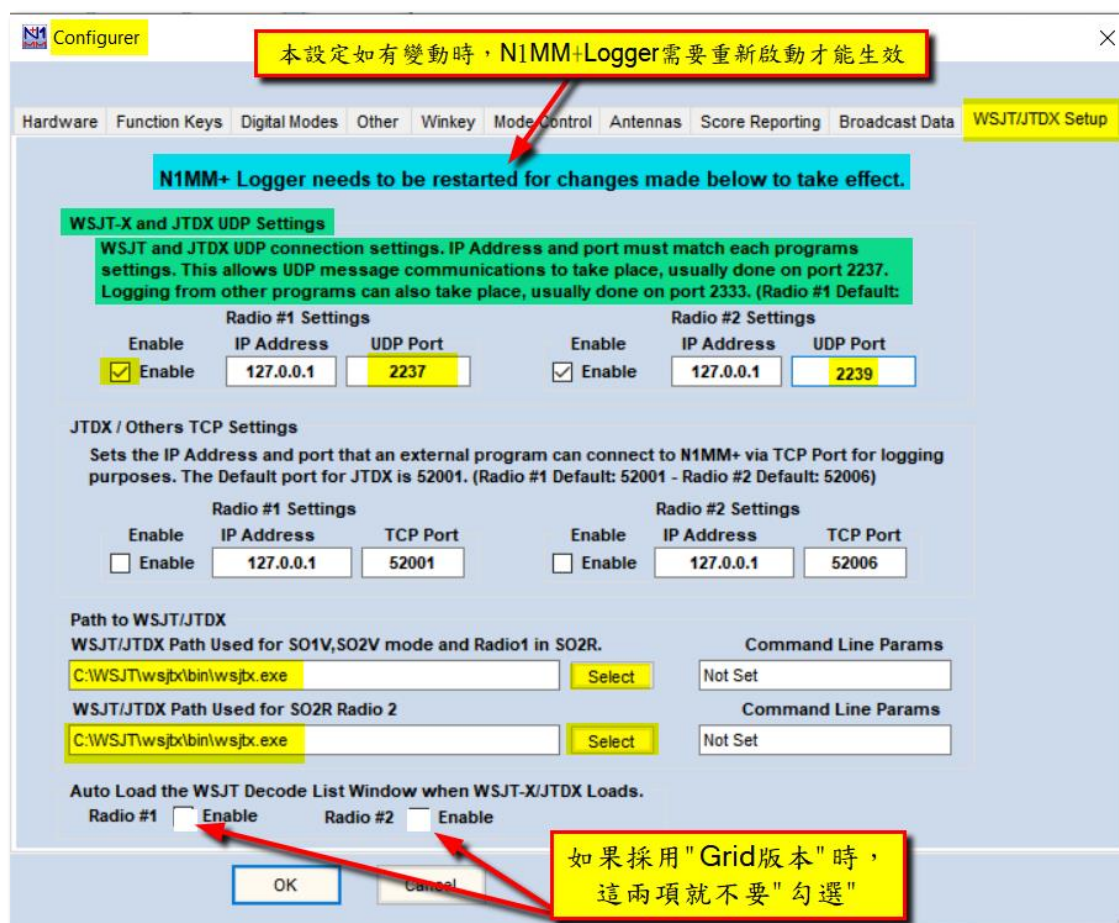


## N1MM+的相應修改：

在這個「Gridtracker 版本中，唯一在 N1MM+中要修改的事項：請到圖\_7 中，將最下面的兩個」勾選項目，「不要打勾」（見下圖\_52）

如此作為的用意就是：

不想去使用到 N1MM+環境下的 Decode List 1(或 Decode List 2)這項「解碼清單」功能；而是想要完全地將 **JT** 的解碼功能轉給 Gridtracker 的 Call Roster 去發揮其更佳出色之功能表現；



圖\_52 N1MM+有關 WST-X(或 JTDX)的 IP 設定(Gridtracker 版本)說明

## C-5\_2、Gridtracker 的三個 UDP 參數設定

在 Gridtracker 的 General 之設定裡，有關 Forward UDP 由原默認之「2238」、改為「2032」之目的，就是為了讓 Gridtracker 在 1st UDP=2031 處、接到 **JT** 的通告後  
[註：此一通告，有兩種可能：一是想要獲得 Gridtracker 提供相關呼號之進一步的詳細

資料；另一種可能是 **JT** 完成 QSO 後，想要 Gridtracker 來儲存 ADIF 檔案的意圖。]；一旦 Gridtracker 被 UDP=2031 通知時，馬上經由 UDP=2032 去通知 Log4OM 來做 Log4OM 的後續動作。當然、這個後續動作是取決於 Log4OM 對於 **JT** 的既有設計裡，已經有其相對應於 UDP=2032 的默認動作在內。

例如：

**JT** 完成「每一筆 QSO」後，透過 UDP=2031 去讓 Gridtracker 的推播 UDP=2032 再去通知 Log4OM 說：有一筆 ADIF 得資料請求 Log4OM 去處理。Log4OM 接收到 UDP=2032 的通知後，馬上去將 Gridtracker「轉過來」的 ADIF 檔案在 Log4OM 設定完成的「UDP=2032 接收通知」、Log4OM 就會以其設計好的儲存動作、將該 ADIF 檔案儲存起來。

同時，由於 UDP=2031 在 **JT** 都是設定成「同意 UDP 請求並重新儲存回原視窗」之選項（見圖\_32 青色部分），因此，**JT** 也會被告知：Gridtracker 已經達成 UDP=2031 的「回報使命」。

換句話說：各應用程式之間(含該應用程式的多項實例 Instances)，都會設立一個相對應的 UDP 來做為溝通的窗口，例如 UDP=2031 這是 Gridtracker 與 Radio\_1 之間的窗口，同時也是 Gridtracker 與 Radio\_2 的窗口。

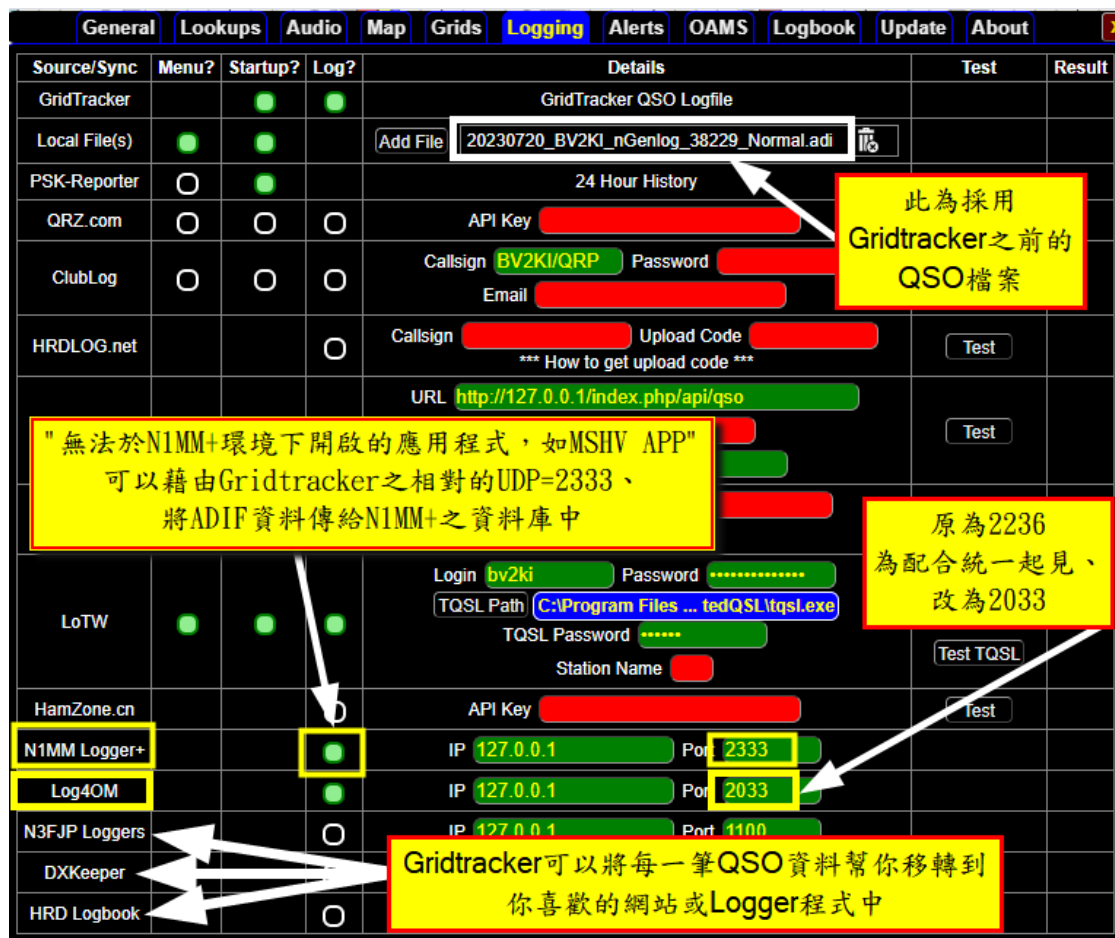
Gridtracker 與 Log4OM 之間、係由 Gridtracker 推播的 Forward UDP=2032 與 Log4OM 做窗口(Log4OM 方，則是以 Gridtracker Received UDP 做接收)。

記得 N1MM+與 **JT** 是以 UDP=2237 做為 N1MM+與 Radio\_1 的窗口，同樣的，UDP=2239 也是作為 N1MM+與 Radio\_2 的窗口。

在 Log4OM 如何設定 UDP=2033 及 UDP=2032、甚至於在 Log4OM 也可以設定 UDP=2237 或 UDP=2239 去取得 **JT** 所產生的 ADIF 檔案（你可以自行試試!）

在 Gridtracker 的手冊裡談到「Gridtracker 並不是一般性的 Logging 程式」，不具有一般市面上的 Logger 的強大功能，如 Log4OM、N3FJP、DXKeeper、HRD Logbook 等那些強的功能。而且 Gridtracker 本身的資料來源主要是針對 **JT** 之應用軟體而做的設計，其主要的目的就是就是將你以前完成的 QSO 資料加上目前採用 Gridtracker 後的 QSO 資料統合起來並做有關的分析。例如顯示你目前完成的 DXCC、CQ Zones、ITU Zones、WAC/WAS、WPX 等獎狀蒐集成果；最主要的還是讓追求「Grid」的同好們，在做 QSO 時，能夠提供詳細的網格 Grid 參考。

Gridtracker 應用程式中有個「Logging」的設定選項，請看圖\_53



圖\_53 Gridtracker 的檔案轉移功能設定

這個選項卡的功能就是讓每一筆 QSO 資料，都可以及時地、幫你「移轉」至你喜愛的 Logger 網站，例如、最受歡迎的 QRZ.COM 以及最具權威性的 DXCC 所主持的 LOTW 等（當然、HRD Logbook 也是功能頗多的一個軟體，它也具有即時「移轉」以及一些獎狀數據分析等功能，不過，這個 HRD Logbook 的大堆頭軟體、每年要「繳」錢維護，而且、如果你的電腦不夠強的話，經常會造成資源不足而卡頓、甚至於當機！）

這裡還有讓你將 Gridtracker 取得的 QSO 資料(由前述的包封方式，UDP=2031 取得的)，即時的「移轉」給你喜愛的 Logger 之中。本文採用 Log4OM 這個免費、短小精幹、超能力的軟體，作為儲存檔案的主角(也稱之為「總舵主 Database」)也是本文強調的重點之一。

Gridtracker APP 原設計將 Log4OM 的「第三對口埠」訂為「2236」。不過、經過幾番考量、為了不與其他軟體設定值衝突，以及簡化起見，我將此 UDP=2236 改為 UDP=2033(這個 2033 就是前述的 2nd UDP Server 的值)。相對的在 Log4OM 的 Connection

設定也是要將 2033 列入作為「窗口」之一，如此才能接收到來自於 Gridtracker 的「移轉」資料。請務必去看看圖\_62 至圖\_64，看完後請馬上會到這裡(按 ALT 左←鍵)。

你是否懷疑？Gridtracker 既然已採用 UDP=2031 及 UDP=2032 的「包封移轉」手段、將 QSO 資料送進 Log4OM 之 Database 中，你又讓 Gridtracker 在這個選項卡的指示下，同時讓「QSO 移轉」至同一個地方、是否會有兩筆一模一樣的 QSO 資料、以先、後方式儲存進去？

好問題、答案是：**有可能**。除非你將 UDP=2237 及 UDP=2239 也放進 Log4OM 的 Connection 當中(方法:如圖\_62 至圖\_64)，如此、就有重複的 QSO 紀錄送進 Log4OM Database 中。

解決的另一方法，請看這裡(如果有這種疑慮時)

為了避免「重複移轉」的現象，我會將 UDP=12060、UDP=2032 及 UDP=2033 這三個屬於 Log4OM 的 UDP Inbound 的接收狀況，做一區分。

區分的方法是：UDP=12060 為單獨的一組(Schedule\_2)，

另一組(Schedule\_1)則是包含 UDP=2032 及 UDP=2033。

這兩組的 UDP INBOUND 事實上是以「採用 N1MM+之運行之+應用程式與否」、做為界定。

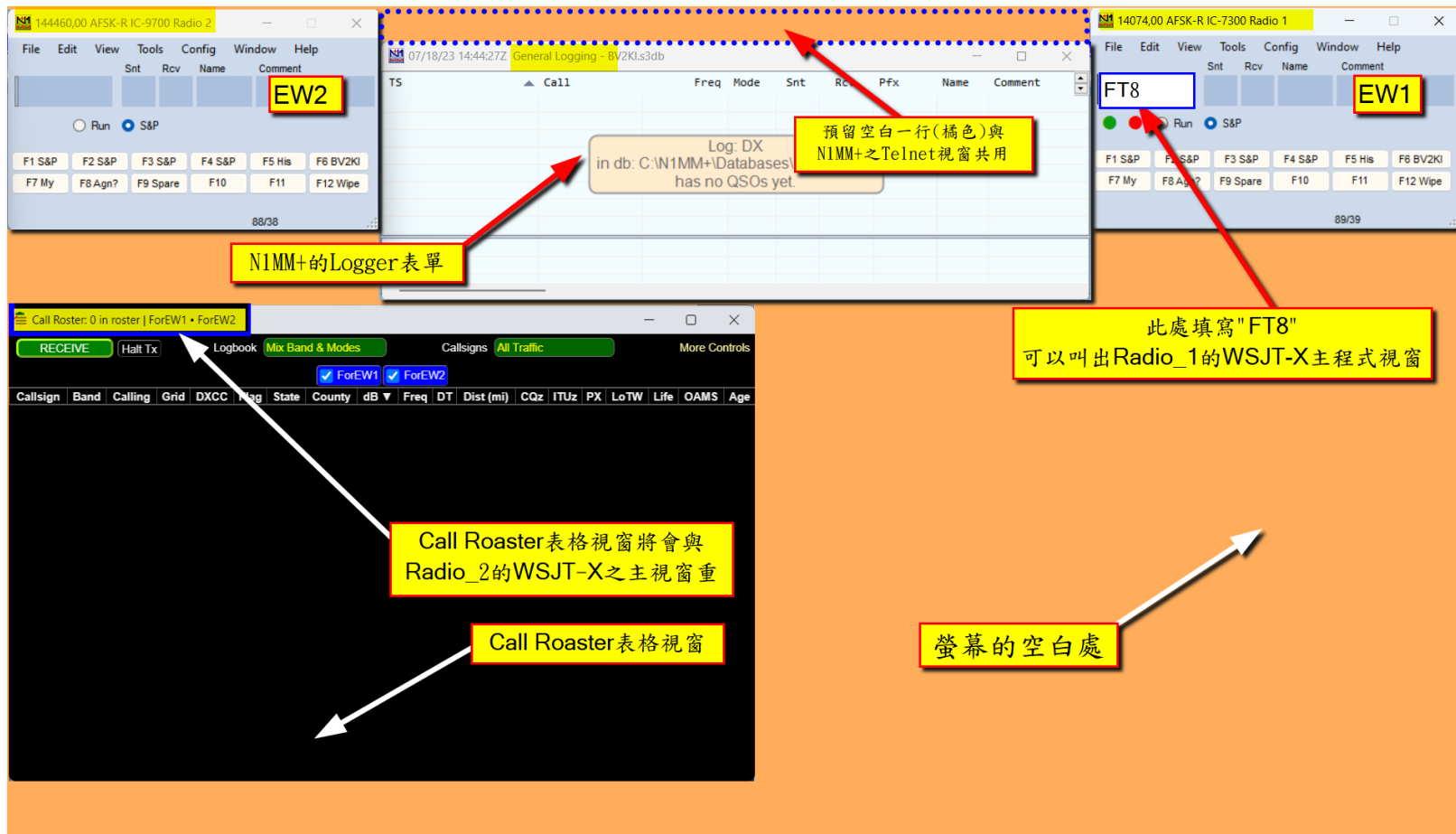
請看看這裡有關 Log4OM 的 Schedule(User Configuration)的設定以及圖\_84(Schedule 的機碼)。

這兩個參考點，目前對你來說、可能太過於匆忙，難以消化；在此先做一伏筆。等你讀到圖\_107 至圖\_108 圖\_107 圖\_107 圖\_107 圖\_108 時，再回到此(避免「重複移轉」的現象)一節，應該可以「豁然開朗」！。

以上這一節所述的三個 UDP 設定(如圖\_107 或圖\_108，Received、Forward 及自身移轉的 ADIF 之 UDP)，就是本文做整合時，在 Gridtracker 中扮演的三個角色。

## C\_5\_2、測試 Gridtracker 版本實作案例

接下來，我們去到圖\_49 的「設定工具按鈕組」、在「Call Roster 按鈕 」，以左鍵按兩下，叫出 Call Roster 的呼號名冊視窗，並將之拖到如圖\_54 的大概位置。

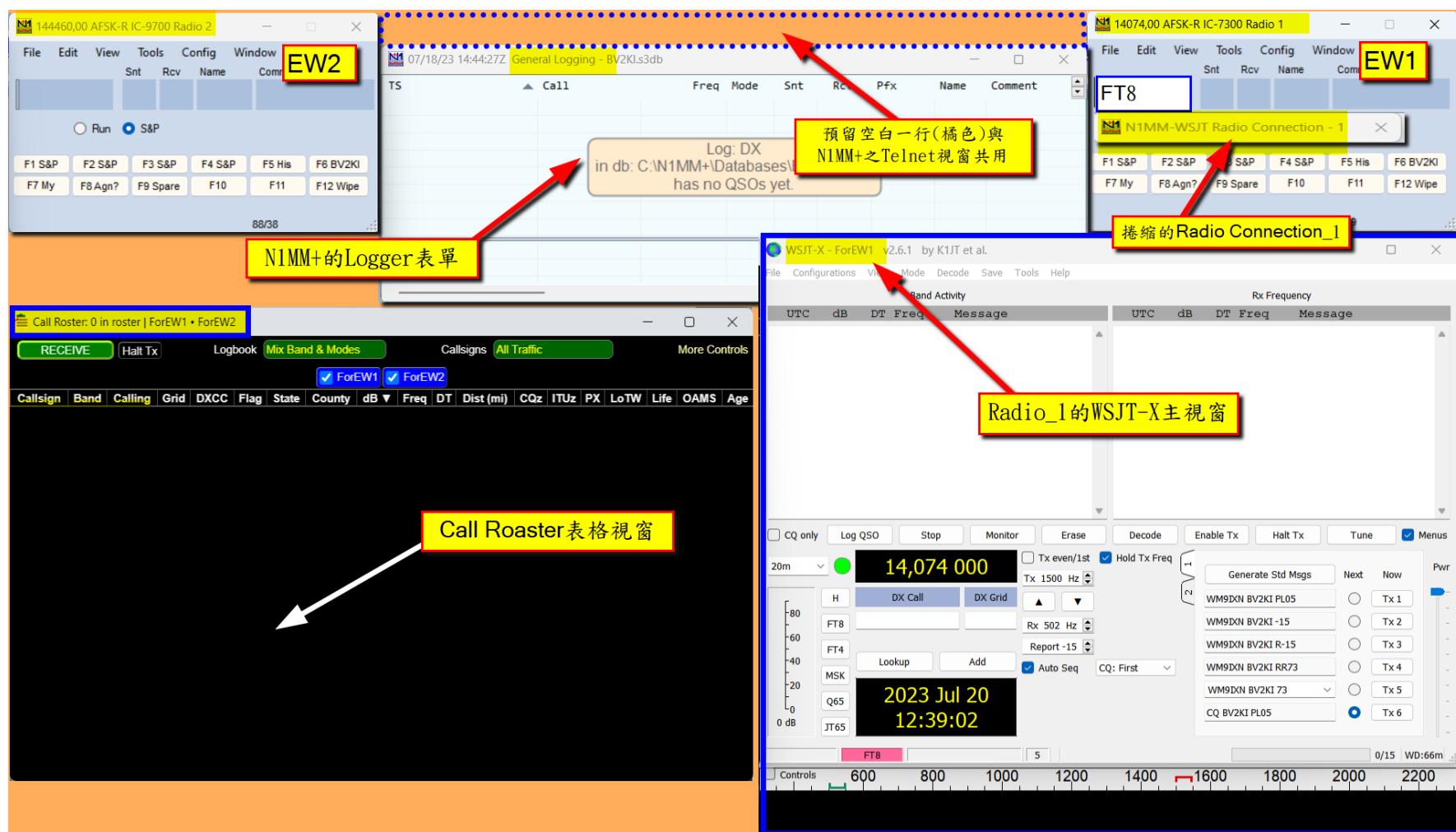


圖\_54 N1MM+的兩個 EW 視窗+N1MM+Log 視窗以及+Gridtracker 的 Call Roster 視窗

Call Roster 的位置，初略的如同圖\_54、大該安排即可。俟以下的所有動作做完後、再回頭將所有的視窗做一次完整、精確地佈置。



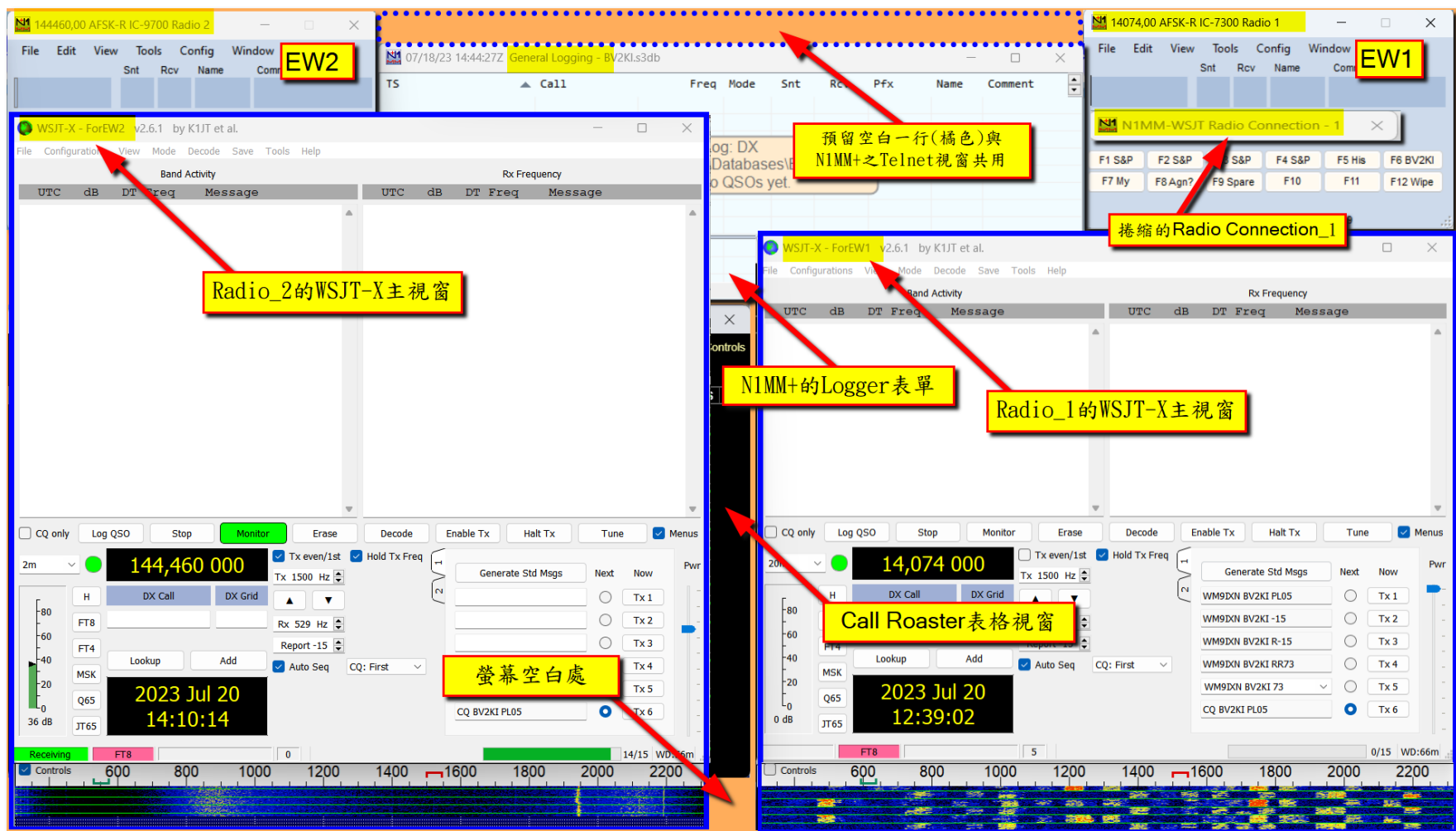
在圖\_54 中、EW1 的 CALL 方框中、填寫「FT8」，就可叫出 Radion\_1 之 WSJT-X 的主程式視窗，如圖\_55。



圖\_55 N1MM+的兩個 EW 視窗+Call Roaster 視窗以及 Radio\_1 的 WSJT-X 主程式視窗)

同樣的，在圖\_55 中、EW2 的 CALL 方框中、填寫「FT8」，就可叫出 Radion\_2 之 WSJT-X 的主程式視窗，如圖\_56。

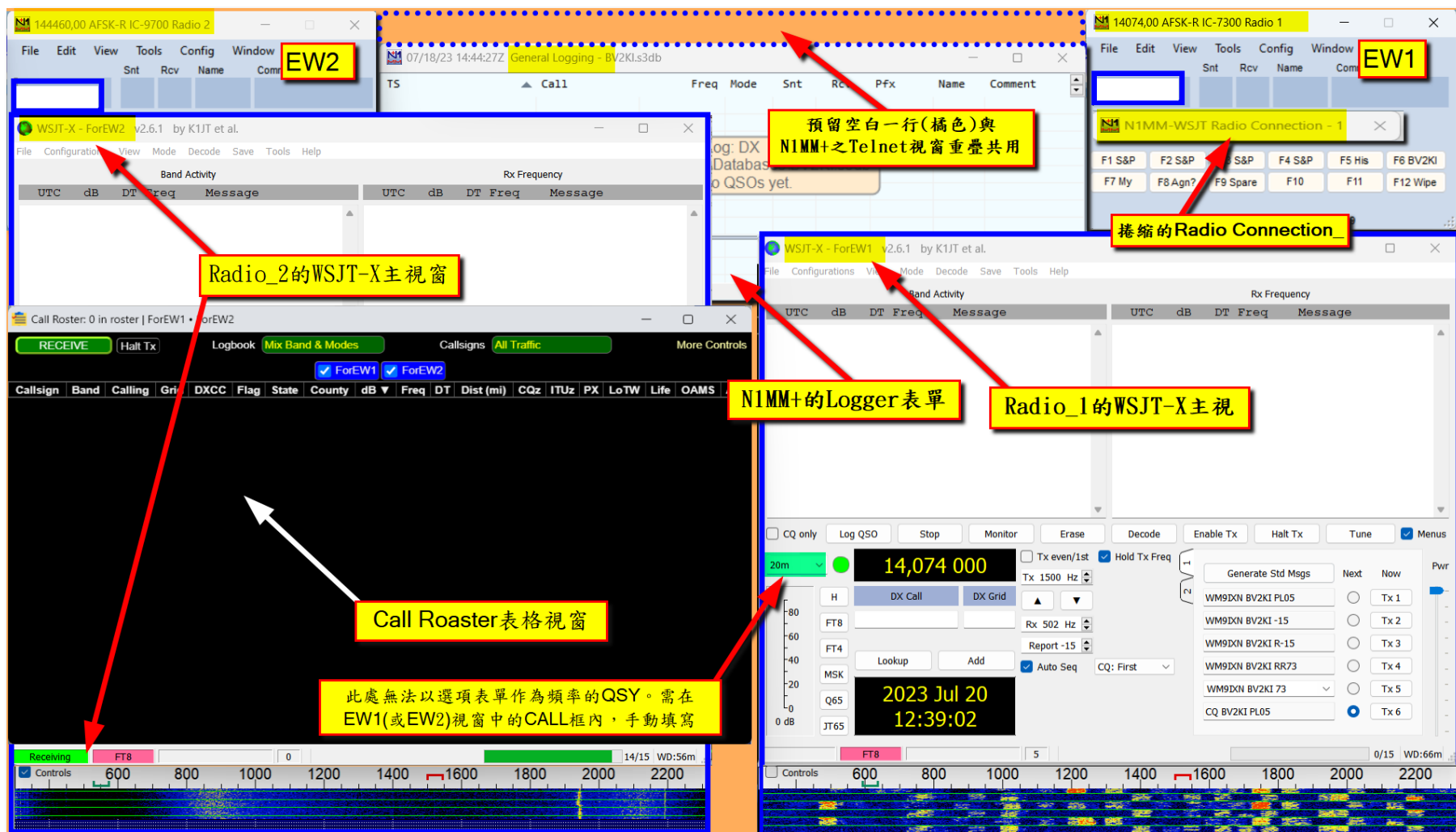




圖\_56 N1MM+的兩個 EW 視窗+N1MM+Log 視窗+Call Roster 視窗以及 Radio\_2 的 WSJT-X 主程式視窗

圖\_56，Call Roster 被 Radio\_2 的 WSJT-X 主視窗遮住，但不影響操作；WSJT-X 隨時有解碼時都會在 Call Roster 顯示、一旦有 Radio\_2 的解碼顯現於 Call Roster 時、再以左鍵回到 Radio\_2 的 WSJT-X 主視窗按一下，就可以呈現出 Radio\_2 完整的主視窗。請注意:如果設定無誤時(Gridtracker 版本)，Radio\_1 的解碼清單是無作用的，這時候你也看不到 Radio\_2 的解碼清單。

當你將在 Call Roster(圖\_56 之中樣部位的)黑色重疊處以滑鼠點擊時，就如圖\_57 一樣。



圖\_57 N1MM+的所有視窗以及Call Roster 視窗的完整組合畫面

這張圖\_57 是一個完整的「Gridtracker 版本」的完整組合畫面。操作重點：

- 1、WSJT-X 的波段選擇(著綠色的)，在 N1MM+的環境下是無效的。如要切換到其他波段時，需要到 N1MM+的 EW1(或EW2)的 CALL 方框內(著白底藍色框)、去填寫正確的頻率，且要以 HZ 為單位；例如:18100 或 14074 等；

- 2、在 Call Roster 中，可以點擊該表格清單中的任何一行，就可以與該行的電台做 QSO。此時 Gridtracker 會根據 Call sign 去到 QRZ.COM 這類的網站上去搜尋該 Call Sign 的其餘細部資料並提供至 **JT** 的資料欄內，例如:Grid 等。
- 3、在 Call Roster 視窗之任一 Call sign 上，右鍵選取「Get MSG」時，該 Call Sign 的細部資料會直接傳給 **JT** 之相關欄位裡；同時也會直接做出 TX 動作；甚為方便，比前述點擊兩下更具威力！
- 4、上述的點擊動作，同時也會經由 UDP=2031 及 UDP=2032 去觸發 Log40M 去做 DB\_Updated 動作。此時你可以到 Log40M 的主視窗，看到前一項的電台細部資料；此一細部資料有可能與 Gridtracker 送出來的，會有些差異，這是因為 Log40M 本身也具有「搜尋」功能，因此不同之應用程式，搜尋後的資料就有不同的結果。
- 5、Call Roster 完全取代 Decode List 之功能，可將兩個 Radio 的解碼資訊同時顯示在一個 Call Roster 中；所以 Radio\_2 如果其使用率較少時，Radio\_2 的 WSJT-X 視窗的大部分面積，可以用與 Call Roster 共用、讓 Call Roster「露臉時間」較多。

### C\_5\_3、我們再做一個測試

首先請看圖\_58：在 Call Roster 呼號名冊視窗裡找到 E200GB 來做 QSO；在 E200GB 的呼號上，左鍵點擊兩下後，會出現該呼號的「Call sign Lookup」視窗；這個視窗提供了 E200GB 在 QRZ.COM 裡面登錄的一些基本資料。你可以按照圖\_58 的動作編號去仔細看看，直至 WSJT-X 完成之 Logging 為止。

**EW2** Radio\_2的 WSJT-x 之視窗

**E200GB** Mr. Pattarachai Phumchaiya  
Chanthaburi, 22000  
Thailand  
e200gb@gmail.com

**EW1** 14074.00 AFSK-R IC-7300 Radio 1

**E200GB**

如果 E200GB 在呼叫 CQ 時，沒有標示出其 Grid 時，這裡就是參考的依據

動作\_3：E200GB 的呼號會自動的送到此

動作\_4：自動做 QSO 的紀錄

Logging 時，如果沒有標示出 Grid，就將 E200GB 的 Grid 資訊 Copy 到 Logging 中。

動作\_1：在此白色方框內任一處、左鍵輕按後、就可由 Radio\_1 接手並繼續自動去做 QSO

動作\_2：此處的 E200GB 的自傳會由 Gridtracker 程式從 QRZ.COM 抓回來、貼到此處讓你參考

Logging 視窗不要遮住此處

WSJT-X - ForEW2 v2.2.2 by K1JT, G4WJS and ...

Click OK to confirm the following QSO:

Call: E200GB Start: 26/07/2022 07:00:45 End: 26/07/2022 07:01:30

Mode: FT8 Band: 15m Rpt Sent: -09 Rpt Rcvd: -14 Name: OK12

Tx power: Comments: Operator: BV2KI Exch sent: RRR

2022 Jul 26 07:02:15

動作\_5：做完 QSO 後，WSJT-X 的 Logging 視窗自動跳出來

此處為 Call Roster 表格視窗與 Radio\_2 的主視窗重疊；Radio\_1 及 Radio\_2 的 WSJT-X 解碼資訊會移轉至此 Call Roster 中

Call sign	Band	Mode	Grid	Country	Frequency	Power	Time
E200GB	15m	FT8	OK12	Thailand	14.168	2m 16s	
VK3XAL	15m	FT8	QF24	Australia	14.168	2m 1s	
JN1EJH	15m	FT8	PM96	Japan	14.168	2m 1s	
RW40	15m	FT8	LO48	European Russia	14.168	2m 1s	
RA0SAT	15m	FT8	HZ1GPGF	Asiatic Russia	14.168	1m 46s	
IR44FF	15m	FT8	AP2FI	European Russia	14.168	1m 59s	

圖\_58 Radio\_2 跟 E200GB 做 QSO 的過程



請從圖\_58 之左下角的：

動作\_1 開始看起。

動作\_2，Call Roster 的一項非常有意思且有助益的「Callsign Lookup 呼號搜尋」功能。它會將 E200GB 的自傳資訊，從 QRZ.COM 抓回來並顯示在畫面上。如果 E200GB 在呼叫時忘了標示其 Grid Square(Location)，且動作\_5 做 Logging 時，Logging 視窗中的 Grid 方框會是空白的，你就可以從 E200GB 的自傳中的 Grid 的值「OK17UM」上、輕輕左鍵按一下，表示將「OK17UM」自動的 Copy 至電腦系統的剪貼簿中。如此，你就可以在 WSJT-X 的 Logging 視窗裡的 Grid 的方框處，按右鍵、選擇「貼上」動作；這個「OK17UM」就可以放在 Grid 的方框裡，完成了一個完整的 Logging 作業。這個 QSO 的案例，E200GB 正好在喊 CQ 時就已標示其 Grid 為「OK12」，所以在動作\_5 的 Logging 視窗中，Grid 方框會自動地將「OK12」加入 logging 視窗中。(Gridtracker 需要有可連上網際網路才可應用>)

動作\_3 至動作\_5，圖\_58 中的解釋已經夠清楚了，不再贅述，請自行細心體會。

**後記**：Log4OM 對於將 QSO 完成後的 ADIF 檔案，在做儲存時，有一個很有用的選項，他可以將你漏掉的應有項目，比如前述 Grid 如果以「空白」資料儲存時，在 JT 內部檔案內也是「空白」的；但是經由 UDP(2237 或 2239 或 2032 或 2033)轉存至 Log4OM 時，Log4OM 會自動地去將遺漏的資料(如 Grid 等)補上、做一個「更新的動作」

**咱們來個小考**：你知道為什麼 E200GB 自傳中的 Grid「OK17UM」跟他在喊 CQ 時的標示「OK12」為何不一致？

1\_可能是搬家關係，他老兄忘了去 QRZ.COM 更正？

2\_是不是 QRZ.COM 中的資料、亂抓的？

3\_也可能是 E200GB 在 WSJT-X 中的設定，弄錯了?!

**答案是**：不要亂猜。寫個 E-mail 問問他就得了！

### WSJT-X 之 logging 視窗位置要擺在哪兒？

WSJT-X 的 Logging 視窗位置，雖然是短暫的，如果這時候沒有跟 Callsign Lookup 呼號搜尋視窗、做一個適當的安排，且需要複製該視窗裡的 Grid 資料時，這兩個視窗擺放的位置就很重要了，你總不希望將呼號搜尋視窗的 Grid 資訊被 Logging 視窗遮住，不是嗎？

所以，WSJT-X 的 Logging 視窗置放位置：只要不去遮住 Grid 的資料那一行，就可以了！

The screenshot displays the N1MM+ software interface with several windows and annotations:

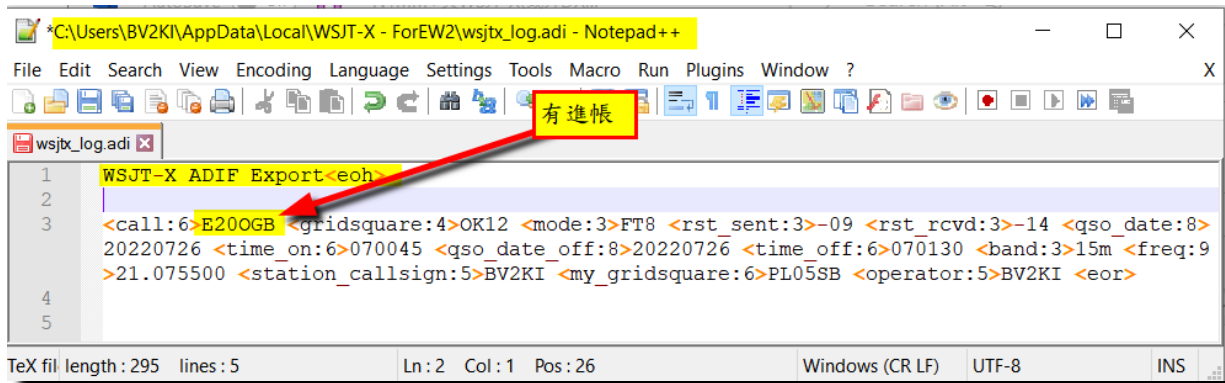
- Top Left:** Window titled "144460.00 AFSK-R IC-9700 Radio 2" with a yellow box labeled "EW2".
- Top Center:** "General Logging" window showing a log entry for "2022-07-26 07:01" on frequency "21075.50 FT8" with call "E200GB". A red arrow points to a blank line above it, with a yellow box labeled "預留空白一行(橘色)與 N1MM+之Telnet視窗重疊共用".
- Top Right:** Window titled "14074.00 AFSK-R IC-7300 Radio 1" with a yellow box labeled "EW1".
- Bottom Left:** "Call Roster" window showing a list of stations. A yellow box labeled "動作\_6: N1MM+將WSJT-X產生的QSO資料, 複製一份並送至N1MM+的資料庫中" points to the roster.
- Bottom Center:** "WSJT-X - ForEW2" window showing a QSO confirmation dialog. A yellow box labeled "動作\_4: 自動做QSO的紀錄" points to the "Log" button. Another yellow box labeled "動作\_5: 做完QSO後, WSJT-X的 Logger視窗自動跳出來" points to the "OK" button.
- Bottom Right:** "Band Activity" window showing a list of stations and their frequencies. A yellow box labeled "Radio\_2的波段運行面板" points to the "Band Activity" window. Another yellow box labeled "Radio\_2的接收面" points to the "Band Activity" window.
- Bottom Center:** A yellow box labeled "此處為Call Roster表格視窗與Radio\_2的主視窗重疊; Radio\_1及Radio\_2的WSJT-X解碼資訊會轉移至此Call Roster中" points to the "Call Roster" window.

圖\_59 跟 E200GB 做完 QSO 後的資料儲存

這是 N1MM+將 WSJT-X 做完 QSO 後所產生的資料、複製一份，放在 N1MM+的資料庫中。此圖\_59 中，動作\_2 已省略掉。

成果檢驗：請根據儲存 ADI 檔案的位置(圖\_35)的樣子在你電腦裡指定的「WSJT-X\_ForEW1」的資料夾中、以文字編輯器開啟「wsjtx\_log.adi」，去檢查看看有無剛剛 QSO 過的這筆 ADIF 檔案的資料進帳？(請看圖\_60)





圖\_60 跟 E200GB 做完 QSO 後的資料驗證

圖\_60，我是以免費的文書編輯軟體 [Notepad++](#) (台灣設計的) 來開啟這個 ADIF 檔案。請注意圖\_60 視窗最頂端的「抬頭」

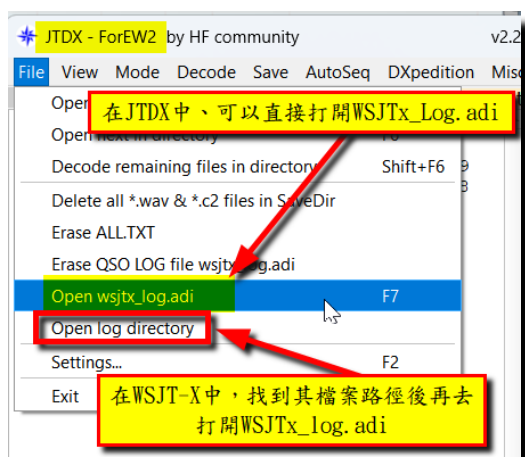
(C:\Users\BV2KI\AppData\Local\WSJT-X - ForEW2)，

這就是檔案的路徑。你的 `wsjtx_log adi` 檔案路徑，應該是將「BV2KI」換成是你在電腦上使用的「使用者名稱」即可。

以上的案例中，JT 的自身儲存 ADIF 的位置在哪裡可以找得到？

在 WSJT-X 的主視窗中你可以去 **File**→開啟上述檔案的資料夾、找到 `WSJTx_Log adi` 後，將之開啟即可查看。

在 JTDX 中，你則可以在 **File**→中、直接開啟前述檔案即可查看。如圖\_61 所示。

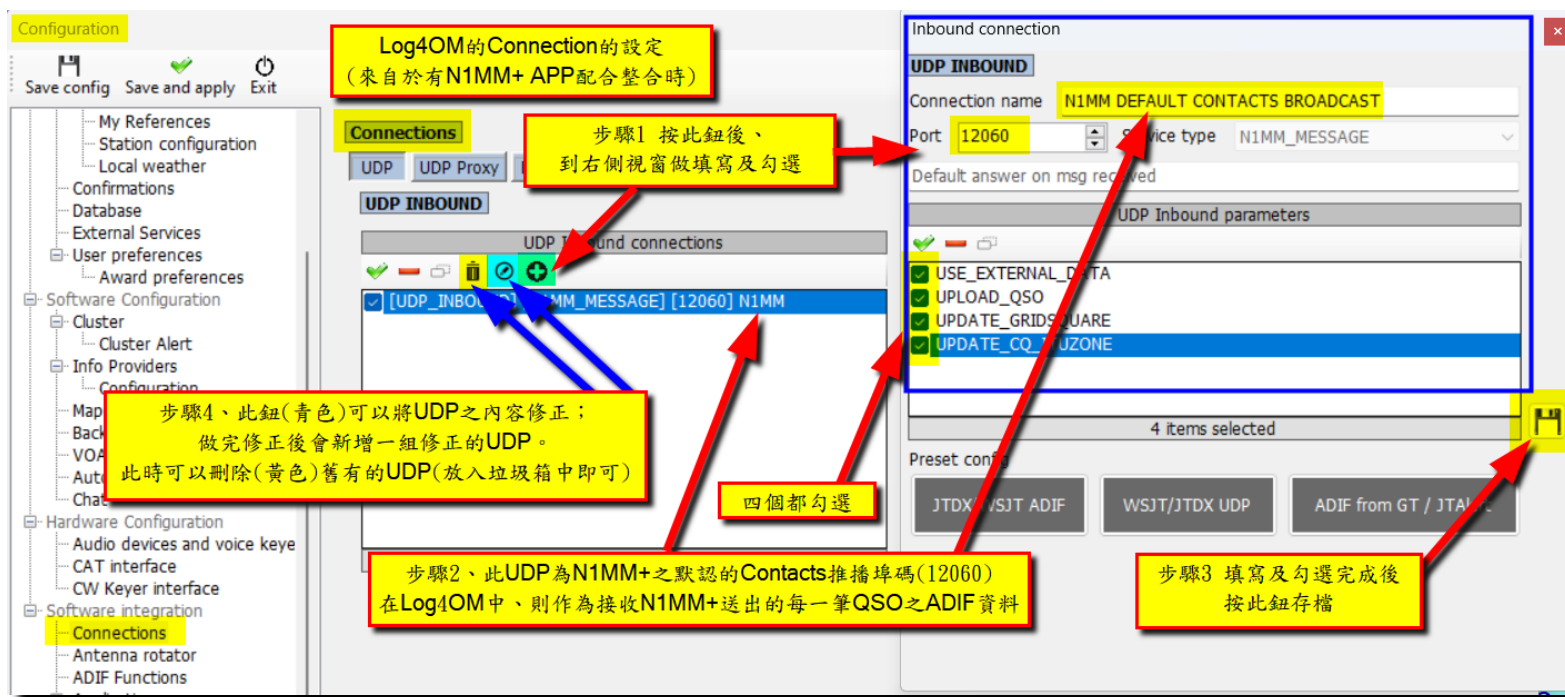


圖\_61 開啟 WSJTX\_LOG 的地方

## C\_6、Log4OM 的配合設定

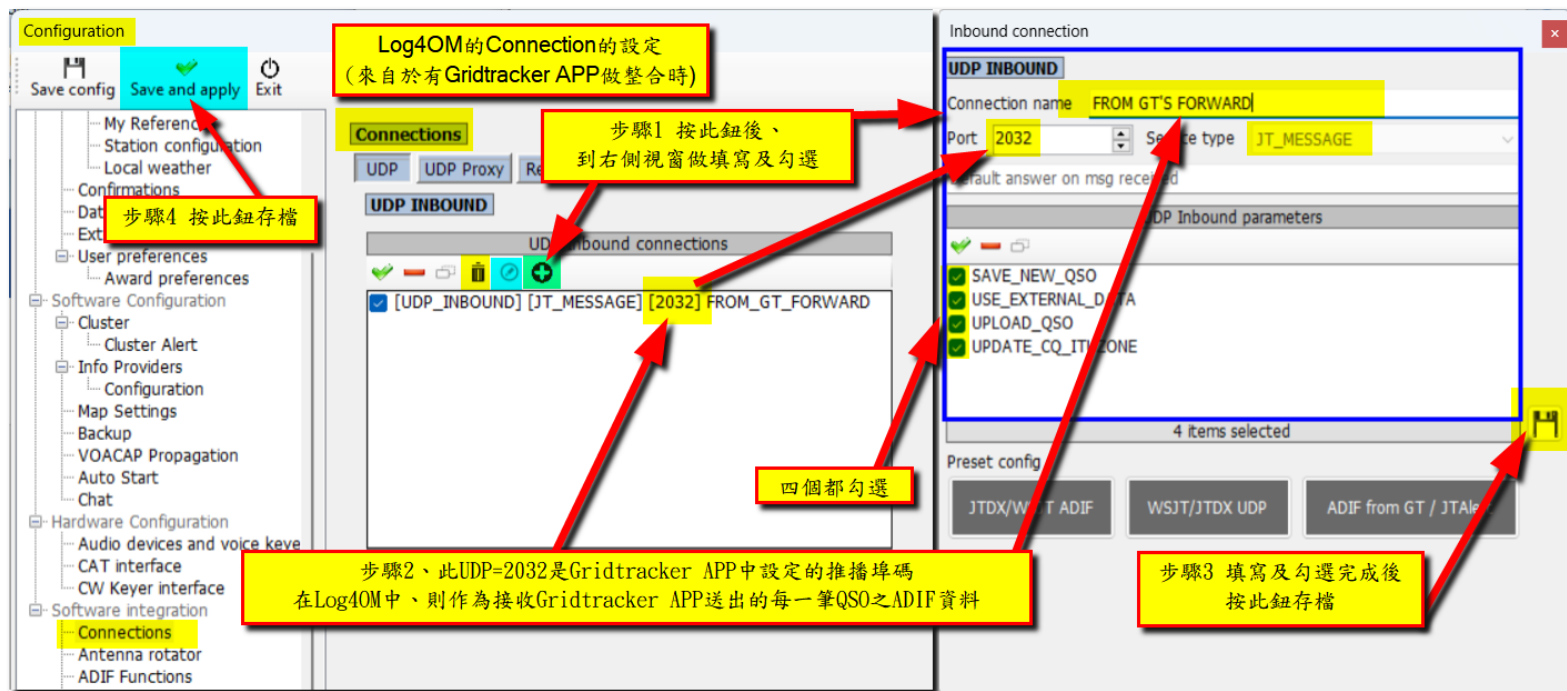
### C\_6\_1、Log4OM 的 Connection 設定

Log4OM 之 Connection 設定目的:用以設定與其他應用程式之間的資料交換或共享。本文的整合對象，以 Log4OM 而言，N1MM+的 QSO 完成後的所有資料(含 SSB、CW、RTTY、PSK 及 FT8 等)、會透過 **UDP=12060** 之管道(UDP=12060 是 N1MM+內建的默認值，請參見圖\_9)儲存進到 Log4OM 的 DataBase 中；請看圖\_62 中的設定步驟；



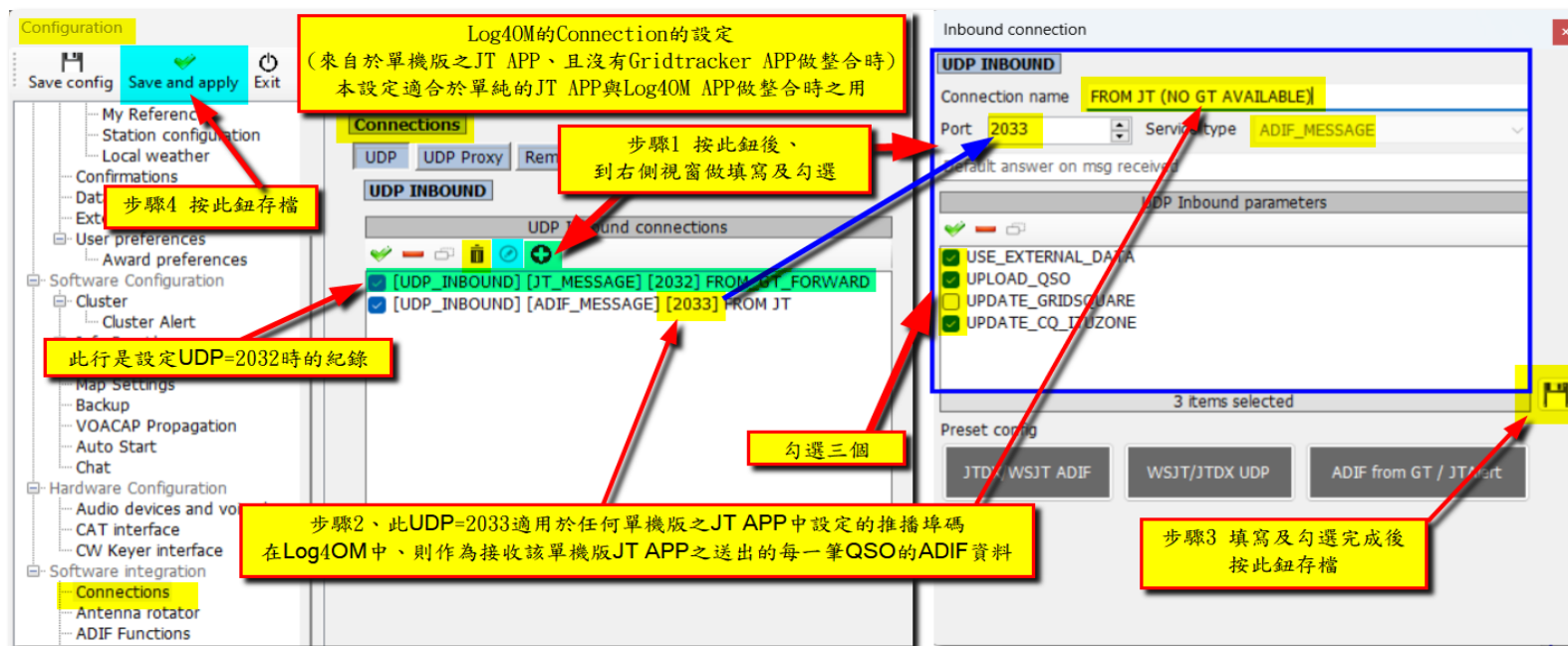
圖\_62 Log4OM 的 Connection 設定\_UDP=12060。此一 12060 純係用於 Schedule\_2(N1MM+環境下之 JT 操作)

**JT** 單獨產生的資料透過 UDP=2031 先傳至 Gridtracker 後、再由 Gridtracker 以 UDP=2032 將 **JT** 的 QSO 資料「移轉」進 Log4OM 之 database 中。請參看圖\_63 的設定步驟。



圖\_63 Log4OM 的 Connection 設定\_UDP=2032(此 UDP=2032 不能與 UDP=12060 者，放在一起；屬 Schedule\_1)

另外、在前述 Gridtracker 之三個參數設定一節提到的第三對口埠，用到 UDP=2033 時、請看看圖\_64 的設定步驟。



圖\_64 Log4OM 的 Connection 設定\_UDP=2033(此 UDP=2032 不能與 UDP=12060 者，放在一起；屬 Schedule\_1)

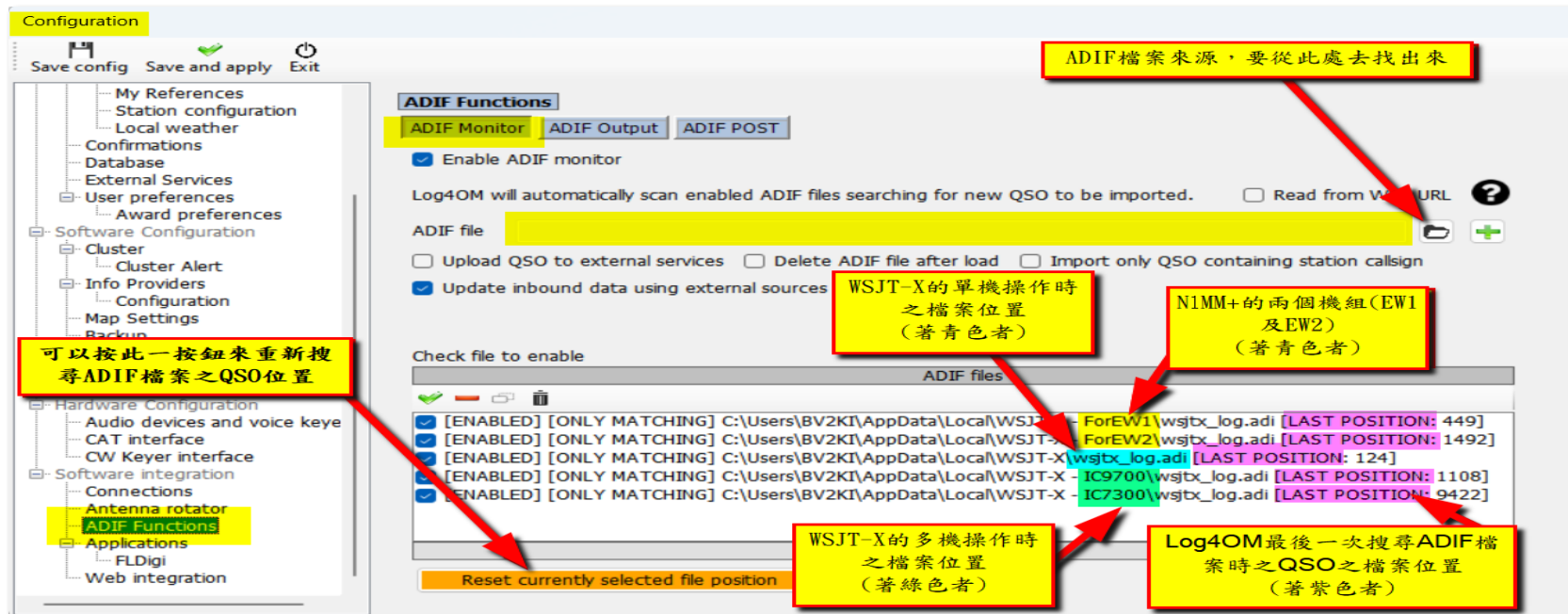
以上所述之前兩種方式係以 N1MM+ APP、Gridtracker APP 及 JT APP 做整合時之考量所設計的方式。如果考慮到僅僅使用單機版的 JT APP 而不考慮採用 Gridtracker APP 作為時、我們也可以將單機版的 QSO 資料以 2nd UDP=2033 直接傳給 Log4OM 之 database 中(如圖\_64)步驟 2；此時所有的單機版之 JT APP 的 2nd UDP Server 設成「2033」即可。

註：圖\_62 的設定方法，是專給 N1MM+環境下使用的(Schedule\_2, 2nd UDP=12060)

圖\_63 及圖\_64 的設定方法。是給不在 N1MM+環境下使用的(Schedule\_1 包含 UDP=2032 及 2nd UDP=2033)。

請參看圖\_107 或圖\_108 的最下方，有兩組不同之 Schedule 截圖，請依照該區分方式去操作。


## C\_6\_2、Log4OM 的 ADIF Function 功能的設定：



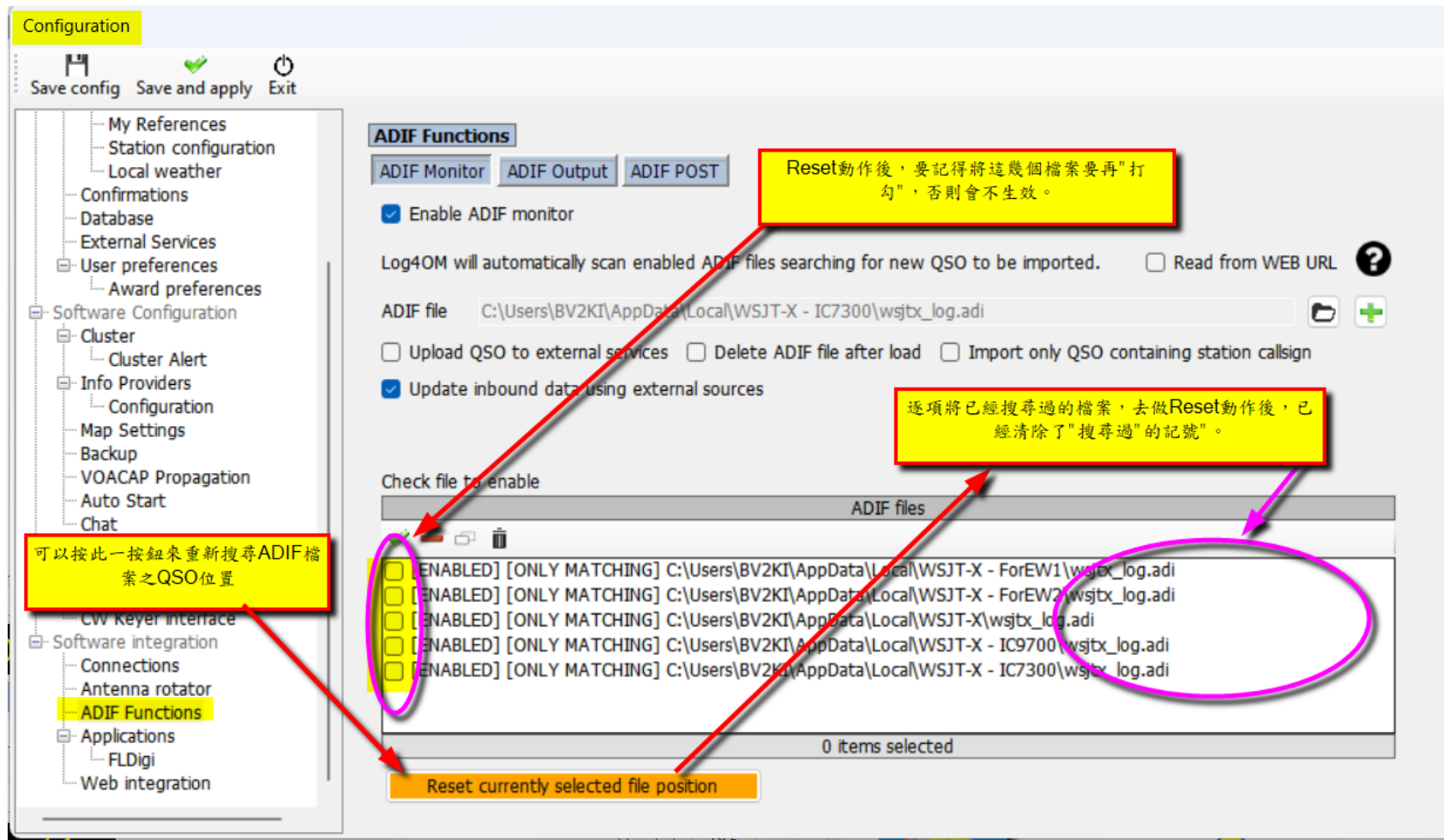
圖\_65 Log4OM 的 ADIF Function 的設定

圖\_65 是設定 Log4OM 的 ADIF Function。這個功能可以讓 Log4OM 去到 JT 的 ADIF 原產地(圖\_35)去取得各應用程式所生產的 ADIF 檔案；Log4OM 取得這些 QSO 資料後，會去 Log4OM 的 DataBase 去做比對、補充及更新動作；如此，就可以避免在各應用程式之間可能的閃失、引起 QSO 儲存不全之顧慮。操作方法：

以 IC-7300 機組為例，將其機組 ADIF 的檔案路徑及檔名指示給上圖\_65 的土黃色空格內：

C:\Users\BV2KI\AppData\Local\WSJT-X - IC7300\wsjtx\_log.adi、然後按  鍵，將 IC-7300 的路徑(含檔名)加入到儲框清單裡。

至於在圖\_65 中的「ADIF files」視窗裡各行字的最後有(Last POSITION: xxxxxxxx)的字眼，這是表示 Log4OM 曾經在上述的幾個 wsjtx\_log.adi 檔案去採用該檔案的資料後，留下一個記號，說他最後拿的資料是停在該資料檔的哪一行位置的意思，作為下次索取資料時的參考、以避免重複索取。



圖\_66 LOG4OM 之 ADIF function 之重新設定

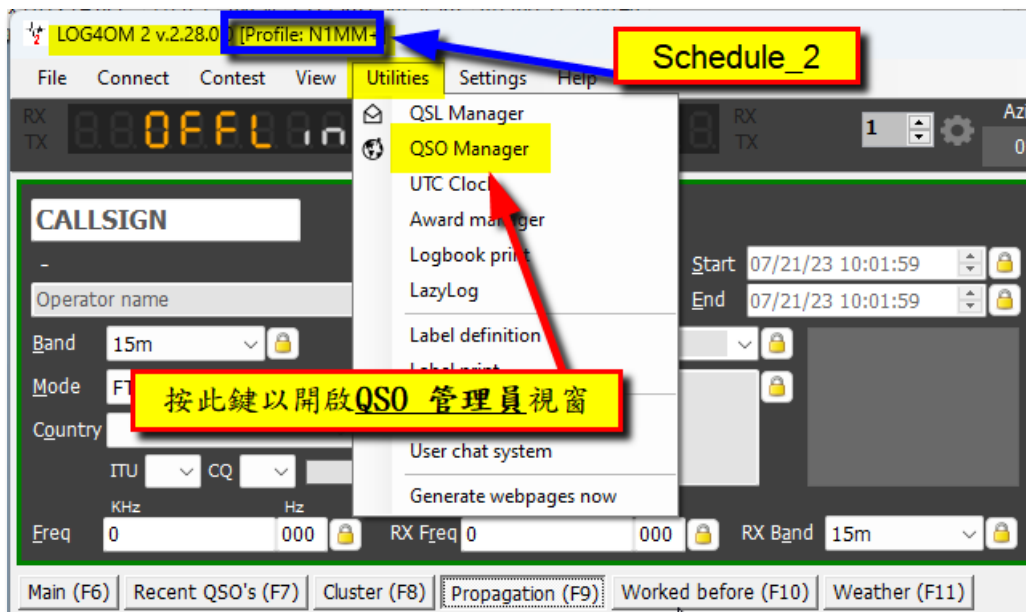


如果在操作本文所述的整合作業時，發現有不正常的現象，譬如各應用程式運行之間、有「不動作」或已退出「共同作業」時、為了避免 QSO 資料未能及時儲存之漏失，就可以利用此一重新設定「搜尋、補充及更新」的功能，做為「事後搜、補事宜」的一種補救措施。換句話說:Log40M 的 Database 如果未能及時 On\_Line 的接收到的資料，透過此 Reset 動作，表列的五個原始檔中的資料，可以去再去做「搜尋及補足」動作；Log40M 有漏失情形時，Log40M 會將漏失資料補充進 Database 內。唯、**請注意**，這個「備份動作」只能用於有異樣時，才去啟動，如果你運行正常之 QSO 也開啟時，你將會發現有「重複登錄之現象」；所以、上圖中的「check file to enable」都沒有打勾，就是暫時不去執行這項「搜、補事宜」。

### C\_6\_3、如何確定 Log4OM 的設定是否成功？

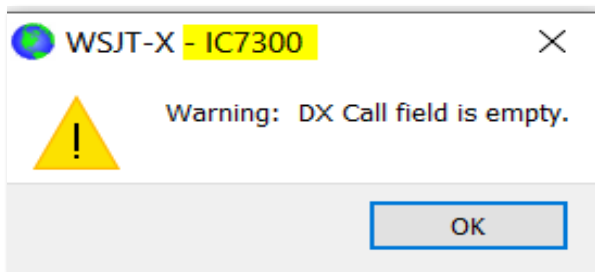
以上在 Log4OM 的 Connection 及 ADIF Function 的設定是否成功？我們來姑且測試一下。

按照本文的方法，以**批次檔方式**將 Log4OM、WSJT-7300、WSJT-7100、WSJT-9700 或 Gridtracker 等應用程式開啟(你也可以將以上的各個 APP 分別開啟)，將各相關的應用程式，妥善安排在桌面上:首先，你可至 Log4OM 的 QSO Manager 視窗如下圖\_67，去啟動 QSO Manager 視窗後、靜候、稍待!將會出現如圖\_70 的一個 QSO Manager 視窗。



圖\_67 在 Log4OM 主視窗中開啟 QSO Manager

在圖\_57， WSJT-X-IC7300 機組的主視窗(Radio\_1 為例)，在 Log QSO 方塊標籤框框上、左鍵單擊，假裝去輸入一個 QSO 記錄；如果 WSJT-X 主視窗中的 DX CALL 方框內是空白的話、會有如下圖的警語出現



圖\_68 如果 DX CALL 是空白時的警語

無所謂、不理會！直接按「OK」鍵、就會顯示下圖的空白的 Logging 表單，以供填寫。

WSJT-X - IC7300 v2.2.2 by K1JT, G4WJS, and K9AN - L...

Click OK to confirm the following QSO:

Call: **TE1ST** Start: 29/05/2022 02:30:56 End: 29/05/2022 02:30:56

Mode: FT4 Band: 15m Rpt Sent: Rpt Rcvd: Grid: Name:

Tx power: 5 ☒ Retain

Comments: **Test from 7300** ☐ Retain

Operator: bv2ki

Exch sent: Rcvd:

OK Cancel

圖\_69 測試 IC-7300 時的填寫內容

請按照上圖示範填寫兩段文字(如黃色者)；填寫完成後按「OK」鍵時，電腦響起一陣歡呼、並恭喜你(這是 **JT** 的默認功能)、表示已經完成 Logging 動作。此時，你可以在 QSO Manager 視窗底端的右下側，可以看到一紅色字串「**REFERENCING...**」顯示約 1 至 2 秒後，剛剛 Logging 後的測試 QSO 紀錄會出現在清單中的最上一行。如圖\_70

QSO Manager

Update info Export to ADIF Export to CSV Backup

Search QSO Single field update Date/Time update "My" update QSO Checks Direct SQL

CALLSIGN QSO date range From: 05/29/22 To: 05/29/22

多了一筆資料 擷取資料成功!

Qso Date	Callsign	Name	Freq	Band	Mode	Rst	Rst	Station	Gridsquare	Comment
05/29/22 02:30:56	<b>TE1ST</b>		21141,5	15m	FT4			BV2KI/QRP	EK80	<b>Test from 7300</b>
05/29/22 02:00:45	BV5OQ	Ke Yi-Tian	144461,5	2m	FT8	-11	-26	BV2KI/QRP	PL03	
05/29/22 01:42:15	BA6BK		50314,5	6m	FT8	-10	-15	BV2KI/QRP	OM81	
05/29/22 01:35:30	VR2ZXP							BV2KI/QRP	OL72dh	
05/28/22 16:02:30	9K2GR							BV2KI	LL49	BV2KI:bv2ki(PL05SB)AS-020[BV2KI;
05/28/22 15:40:30	UF4S	Stepan Kudry...	10137,5	30m	FT8	-12		BV2KI	LO36	BV2KI:bv2ki(PL05SB)AS-020[BV2KI;
05/28/22 15:38:45	RA3VX	Vyacheslav Sil...	10137,5	30m	FT8	-15	-13	BV2KI	LO06fd	BV2KI:bv2ki(PL05SB)AS-020[BV2KI;
05/28/22 14:53:45	VR2IL	Li Chi Sing	18101,5	17m	FT8	+17	-01	BV2KI	OL72cg	BV2KI:bv2ki(PL05SB)AS-020[BV2KI;
05/28/22 14:50:00	HA5CPN	Laszlo Czebe	18101,5	17m	FT8	-15		BV2KI	JN87	BV2KI:bv2ki(PL05SB)AS-020[BV2KI;
05/28/22 14:42:45	BH7EUE	Zeng Jianke	21075,5	15m	FT8	-07	-21	BV2KI	OL68	BV2KI:bv2ki(PL05SB)AS-020[BV2KI;

擷取資料時，會有此一通知信號出現

REFERENCING... Selected 1 of 5000 max: 5000

圖\_70 Log4OM 的 QSSO Manager 視窗，可以看到測試結果

同樣地，在 WSJT-X-IC9700 機組(Radio\_2)的主視窗中的 Log QSO 方塊標籤、左鍵單擊，假裝去輸入一個 QSO 記錄；此時、WSJT-X 主視窗中的 DX CALL 方框內、因為我保留了上次做過 QSO 的呼號「BX9AA」當我左擊 Log QSO 就不會出現如圖\_68 的警語，而是出現了如圖\_71 的填單；同時 BX9AA 的「Grid=OL93」也會自動地填進去。

Click OK to confirm the following QSO:

Call	Start	End
BX9AA	29/05/2022 02:32:59	29/05/2022 02:32:59

Mode	Band	Rpt Sent	Rpt Rcvd	Grid	Name
FT8	2m	-20		OL93	

Tx power: 5 ☒ Retain

Comments: Test from 7100 ☐ Retain

Operator: bv2ki

Exch sent:  Rcvd:

OK Cancel

**BX9AA喊CQ時的Grid資訊會自動的填進來**

圖\_71 測試 IC-7100 時的填寫內容

1、填寫完成後按「OK」鍵，去看看結果吧！

Update info ADIF Export to ADIF CSV Export to CSV Backup

Search QSO Single field update Date/Time update "My" update QSO Checks Direct SQL

CALLSIGN

QSO date range From 05/29/22 To 05/29/22

**又多了一筆資料(黃色者)**

Qso Date	Callsign	Name	Freq	Band	Mode	Rst	Rst	Station	Grid	Comment
05/29/22 02:32:59	BX9AA		144461,5	2m	FT8	-20		BV2KI/QRP	OL93	Test from 7100
05/29/22 02:30:56	TE1ST		21141,5	15m	FT4			BV2KI/QRP	EK80	Test from 7300
05/29/22 02:00:45	BV5OQ	Ke Yi-Tian	144461,5	2m	FT8	-11	-26	BV2KI/QRP	PL03	
05/29/22 01:42:15	BA6BK		50314,5	6m	FT8	-10	-15	BV2KI/QRP	OM81	
05/29/22 01:35:30	VR2ZXP	Alfred Leung ...	50314,459	6m	FT8	-12	-16	BV2KI/QRP	OL72dh	
05/28/22 16:02:30	9K2GR	Khaled Alhashel	10137,5	30m	FT8	-06	-15	BV2KI	LL49	BV2KI:bv2ki(PL05SB)AS-020[BV2KI;
05/28/22 15:40:30	UF4S	Stepan Kudry...	10137,5	30m	FT8		-12	BV2KI	LO36	BV2KI:bv2ki(PL05SB)AS-020[BV2KI;
05/28/22 15:38:45	RA3VX	Vyacheslav Sil...	10137,5	30m	FT8	-15	-13	BV2KI	LO06fd	BV2KI:bv2ki(PL05SB)AS-020[BV2KI;
05/28/22 14:53:45	VR2IL	Li Chi Sing	18101,5	17m	FT8	+17	-01	BV2KI	OL72cg	BV2KI:bv2ki(PL05SB)AS-020[BV2KI;
05/28/22 14:50:00	HA5CPN	Laszlo Czebe	18101,5	17m	FT8	-15		BV2KI	JN87	BV2KI:bv2ki(PL05SB)AS-020[BV2KI;

Select/deselect Filters QSO Map Selected 0 of 5000 max: 5000

圖\_72 WSJT-X 的兩筆 QSO 資料已經連接進入 Log40M 的資料庫

2、完全成功！恭喜！

## C\_6\_4、Log4OM 的診斷視窗



圖\_73 Log4OM 的診斷視窗，可以作為檢視 UDP 之設定是否發揮作用

圖\_73 是 Log4OM 的一個小程式；用來檢視 Log4OM 所設定的 2nd UDP 是否已發揮作用。可以將此視窗放在桌面上，可隨時監看咱們的 QSO 資料是否如預期地送進 Log4OM 的資料庫中。

該程式也同時監控如：Telnet Cluster 之接收狀態、ADIF Function 是否正常處於正常運作？送進來的每筆 QSO 資料是否做更新？「稀有電台 On Air」之設定提醒等等資訊，隨時提供 Log4OM 的內建功能等資訊，讓你參考，還算是蠻有親和力的軟體。

#### C\_6\_4、**KI\_Format** 的解說:

一般的 Logger 大都僅提供單一呼號做為操作對象，這是考慮到每一個呼號有不同的屬性，例如 BV2KI/QRP，BV2KI/8 或 9M2/BV2KI 都可能有不同的 Location(Grid)等資訊。為了方便起見大都是以各別的呼號讓你做運用。因此、如上所舉例的三個呼號，就有三個不同的 Logger(或稱之為不同的 Instance 實作例)產生；如果要處理 QSL 卡片時，我必須在這三個 Logger 之間遊走及出進的；皆屬麻煩之至；尤其、當我要處理 20 個以上不同呼號的 QSL 卡時，更是不可想像的瑣事。

如果可以讓所有各電台的 LOGGER 放在一個位置時，每當有個呼號(例如:JA1AA)與我手中的某幾個電台(例如:BV0WPX、BOOM 及 BV2KI)有做過通聯的紀錄，但是 JA1AA 僅有一張 QSL 卡是向 BV0WPX 請求回覆時，我就可以將其餘的 BOOM 及 BV2KI 的 QSL 卡、集中在一個 LOGGER 上一併做處理。如此，不需要各別去翻找、找出該單一電台的 LOGGER，省掉很多功夫。

因此，我在 2001 年採買了當時最新的一款叫 nGenLog 的 Logger 軟體；號稱是下一世代的日誌(Logger)軟體。該軟體具有當時具有較新的 DataBase 操作功能，速度快，短小精悍。可惜的是，三年後因為該軟體開發者與幫忙他們寫軟體的工程師意見不和而拆夥以致斷炊，無法繼續更新至今。但是、它可以讓我的所有電台的 LOG 存放在同一個 Logger 裡，我可以同時處理所有的 QSLing。至今已經有超過 30 萬筆 QSO 放在一起，速度一點都不受影響。雖然、我也購置了豪華的 HAM RADIO DELUXE 軟體，該軟體是目前最火紅且據稱是最進步的佼佼者，這點從它們不斷的版本更新、可以看出其用心，對於個人使用者而言，很值得購買！

但是、對於身為一個 QSL Manager 來說，還是有點使不上力。我曾經將 30 萬筆的紀錄放在該軟體上，發現速度上有極大之差異。該軟體，所有你想要的功能幾乎無所不包，而且由於其介面頗多、各介面間之銜接有許多「卡頓」，其他豪華的附屬功能也不常用；因此，我就暫時將其「擱置」，看看日後能否可有增進之處，再考慮回頭使用它。

其實、將所有的 log 放在一個軟體當中是很容易解決的。我將每一筆 QSO 紀錄，給予一個容易識別的「索引紀載」，讓我們自己一目了然。以這個「索引」的內容，就可以容易去辨識此一 QSO 是屬於哪一個電台的 log，不會混淆。

為了達成「將所有的 QSO 做統合性的儲存於單一 Logger 裡」、且易於辨識每筆 QSO 紀錄之屬性」的目的，我則是遵照 **ADIF 標準規格**(此規格是目前所有設計 Logger 或網站存取 QSO 紀錄的統一標準)、所規定的欄位名稱做一組合後並將此一組合的文字字



串，存放在「COMMENT」欄位中，讓此「COMMENT」欄位裡的訊息，做為辨識每筆 QSO 之索引資料。

每一筆紀錄這個標記的標準欄位名稱及組合方式、就像這樣的：

CallsignOnAir:OwnerCall(My\_Grid)MyIOTA[Operator]SIG\_Info(SWL' s Heard);Real comment

請注意上一行文字中，各相關欄位名稱(以底線標註者)之間，逐續以

「:」、「(」、「)」、「[」、「]」及「;」

等 6 個符號做隔開符號(Delimeter)，這些間隔符號的最後一個分號「;」不能漏掉。

說明：

CallsignOnAir: 喊 CQ 者的電台呼號

OwnerCall: 電台台主(有可能是自己家裡的電台，也可能是去到別家電台去作業的電台)

My\_Grid: 電台住在地的網格編碼(4 碼或 6 碼)

My\_IOTA: 電台的 IOTA 編碼(6 碼)

Operator: 值機時的操作員呼號

SIG\_Info: SWL 聽到的他方電台(與 CallsignOnAir 通聯時的電台)的呼號；

此處的「SIG\_Info」是一個借用的欄位；ADIF 規範中的意思：「在填寫 QSO 時、對方電台的特殊資訊」；例如：對方電台可能是為該某些組織之「50 周年紀念台台慶」而做的慶祝活動等等；這些資訊有可能是由本地之操作員、在執行 QSO 時，已登錄在 Comment 或 Note 之欄名內者；也有可能是後來在 QRZ.COM 網頁上 copy 下來後、填進此 SIG\_Info 欄位裡。

我發現此規格之原意使用機會不大，因此，才援用來做為 SWL 者提供資訊的填寫之處。

此一欄位的填寫內容，係針對 SWL 而言，就是填寫當時正在與本電台做 QSO 時、他方電台呼號，這個他方電台呼號是 SWL 來信(或 QSL 卡)時所記錄下來的呼號，我就將此一「他方電台呼號」填寫在這個「SIG\_Info」地方。

**SWLer's Heard**: 這是延續前述「SIG\_Info」的選項。

一般而言，SWL 可能會寫一些他收聽時的天氣狀況、傳播條件、甚至於會敘述他的收聽裝備等等；SWL 如果僅僅收到本電台的信號時(他可能無法收到他方電台呼號)、就沒提到「他方電台」的呼號。例如:他只聽到本電台正在喊「CQ」、而非給予前述所說的「他方電台呼號」時。我們只好記下「CQing」這幾個文字用以替代「他方電台呼號」了。

這個「**SIG\_Info(SWLer's Heard)**」的出現，僅僅是針對 SWL 來信申請 QSL 卡片交換時才會有一筆新增紀錄。全球 SWL 的呼號千奇百怪，有的是該國政府有某些規定的，有的是自行設定的。例如:JA1-1234 或者 ON-5678。像這種非 HAM 的標準呼號、許多 Logger、在填寫進「CallSign」欄位時、是不允許做登錄的；而 nGenLog 就可以允許 SWL 的呼號做登錄，這也是我喜歡 nGenLog 的原因之一(Log40M 也有這樣子的功能)。

**Real Comment**: 實際的 Comment；

這裡指的是本地電台操作員，在實際運行 QSO 時所記錄下來的文字。比如：本地電台操作員抄下對方的城市名，或是對方是個 YL 或年齡等。如果有這項「Real Comment」的文字時，就把這個實際的「Comment」，附加在最後、形成 KI Format 之「索引資料」之結尾。如果沒有「Real Comment」時，最後的一個「分隔符號；」也要「寫進去」。

KI Format 實例一(Comment 欄位):

**BV0WPX:BV8BC(PL02ED)AS-020[BV2NT;**

這是最基本的資料：BV0WPX 在 BV8BC 的電台作 QSO。接下來表示 BV8BC 電台的網格位置及 IOTA 編碼，接著表示：這一筆 QSO 是 BV2NT 操刀的

再來第二例，有個「SWL」叫「ON-7890」來信說：有聽到「BOOM」跟「JA3AAA」做 QSO：

我們在 Log 中就將「Callsign」填寫成「ON-7890」；「Comment」欄位則填寫成：

BOOM:BOOM(OL96XD)AS-113[BX2AB]**JA3AAA**;**Sasaki 85 才**

解讀:

BOOM 在馬祖作業，「OL96XD」是馬祖的 Grid 網格信息；操作員 BX2AB，「SWL」同時聽到「BOOM」與「JA3AAA」在做 QSO。BX2AB 操作 QSO 時，當場記下對方電台(JA3AAA)的名字叫「殺沙去」，今年 85 歲。

第三個例子：

請看下圖，這些都是新增的 SWL 來信時所做的紀錄檔案截圖

Time on	Callsign	Name	Mode	Band	RST	RST R	Comment
1992-07-14 08:32	NL-8794/DL	SOR	SSB	10m	57		BV2KI:BV2KI(PL05SB)AS-020[BV2KI]CQING;
1994-04-03 10:00	NL-8272-R2		SSB	20m	33		BOOM:BOOM(OL96XD)AS-102[BOOM]JA3AAW;
1994-07-21 14:48	NL-8992-R2	Anthoniell	SSB	20m	54		BOOM:BOOM(OL96XD)AS-113[BOOM]JV3OWC;
1994-07-30 13:35	NL-8992-R2	Anthoniell	SSB	20m	55		BOOM:BOOM(OL96XD)AS-113[BOOM]OH2NAG;
1996-04-07 18:36	NL-8746						[BV2KI]LZ2QGS;
2001-03-26 11:42	NL-8992-R2						[BN0W]DJ4SP;
2005-07-29 22:18	NL-8951		SSB	20m	55		BV9G:BV9G(PL02RQ)AS-155[BV9G]EA8AK;
2005-07-31 07:11	NL-8992-R2	Anthoniell	SSB	15m	52		BV9G:BV9G(PL02RQ)AS-155[BV9G]G0BLB;
2006-04-08 17:22	NL-8951		SSB	20m	57		BV50CRA:BV50CRA(PL04JI)AS-020[BV2KI]EA1BT
2007-07-27 15:08	NL-8951		SSB	20m	55		BOOM:BOOM(OL96XD)AS-102[BV2KI]K2GHW;
2011-04-24 12:03	NL-8071		SSB	20m	59		BV100:BX5AA(PL04GB)AS-020[BV2KI]VE3AXW;
2017-10-24 13:47	NL-8992-R2	Anthoniell	MFSK16	30m	-15		BV2KI:BV2KI(PL05SB)AS-020[BV2KI]UA9AFA;

圖\_74 有關 SWL 的在 Comment 欄位裡的 KI Format

圖\_74 顯示的 QSO 紀錄，只顯示出有關 SWL 的部分內容，請注意：RST RCVD 這個欄位全部是空白的；因為 SWL 無法發送信號、這是正常的。因為這種「非正常的 Logging 登錄」，在許多 Logger 是不允許的，包括 nGenLog 有時也不例外。如何達到如圖\_74 的空白呢？答案也很簡單，因為處理 SWL 來卡時，應該都是個別於不同時間送到手裡的，這都是需要以人工去 Key\_In 的。先假裝填上 599(或 59，視 mode 而定)，騙過程式對這個 RST RCVD 的檢查作業、完成 Key\_in 並予以儲存後，再去做一次人工修正，就可以將 RST RCVD 大方的變成空白(此時程式已無暇去檢查作業)；這可能是設計者 nGenLog 當初沒想到的小小 Bug 吧！

以上所述，就是如此的設計一個「索引記載」掛在每一筆 QSO 的紀錄上，這個「索引記載」究竟是放在 ADIF 標準規定的哪一個欄位最為適當？我仔細研究了 ADIF 規劃者對於 Comment 及 Note 這兩個欄位的解釋，發現似乎有些模稜兩可；且從幾個著名的 Logger 的檔案格式去探討，發現使用 Comment 或 Note 者，也莫衷一是。不過統

計起來，引用 Comment 欄位者居多，因此我就選用「Comment」欄位作為此「索引記載」標記的地方了。

另外有一個影響檔案大小以及將來發展空間的因素，也要加以考慮。

「Comment」只能記錄單行文字，字數有限(255 字？視系統 OS 而定)；而「Note」欄位可以有多行的記載功能，可以日後從事對該筆 QSO 紀錄做重複、多行的記載。所以，選用「Comment」欄位作為這種小小的「索引記載」也不會去阻擋日後需要另外加註的空間，也是蠻正確選擇。

這就是 KI\_Format 的運作方法。

你可以運用在你目前使用的 LOGGER APP 中。可以將以前分別儲存於不同的呼號的 QSO 紀錄，全部放在一個儲存 DATABASE 中。例如 BV2KI/QRP，BV2KI/8 或 9M2/BV2KI 等等多個 log database 集中至你的「總舵主 BV2KI 之 Database 中」；一旦完成後，你今後的任何 portable station 的紀錄等都可都在一個 Logge 一覽無遺。如果你是某人的 QSL Manager 你也可以將該某人(客戶)的 Log 資料一併放進來，都可以輕鬆以對。

以下的設定所產生的 DataBase 中，將會把 KI Format 引用進去。

## C\_6\_5、Log40M 的 Schedule(User Configuration)的設定

透過上一節的「KI Format」方式、將 Log40M 作為我們的最終儲存 QSO 的地方，就是本文的目的。

為了將 Log40M 能夠負擔起「總舵主」的任務，有關 Log40M 的開始安裝時的 DataBase 設定、就起了很重要的關鍵。這個起始的設定，提供了基礎 DATABASE。

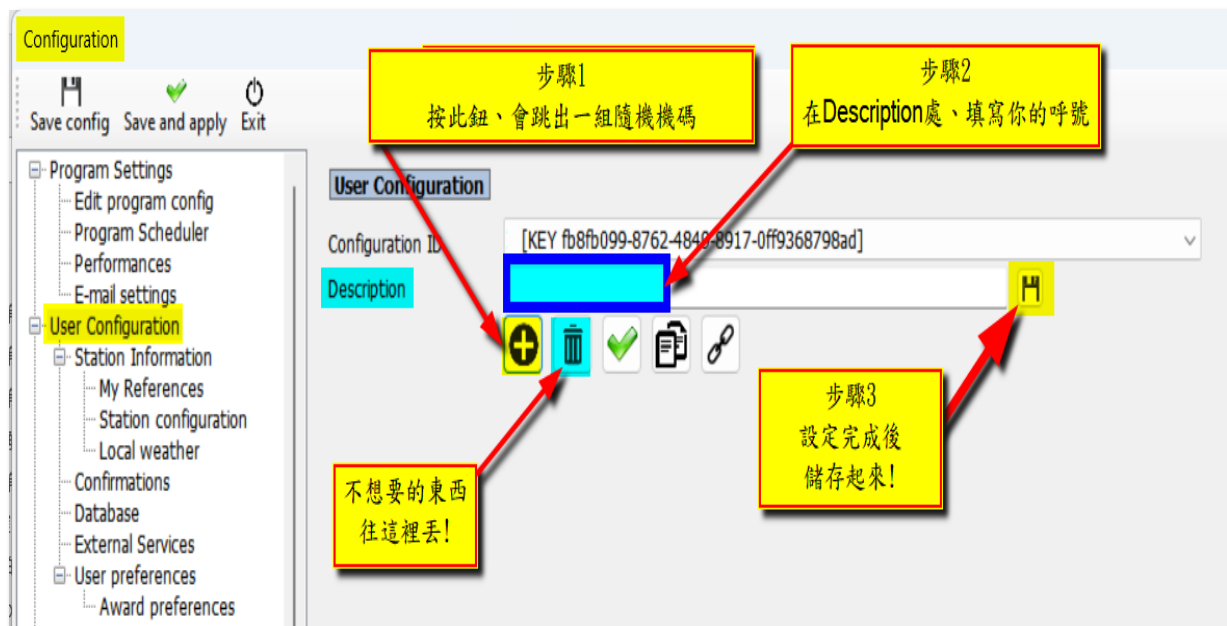
例如：你必定是以你自己的呼號去做「總舵主」之初始設定。一旦以自己的呼號做成「總舵主」時，此時、你的基準 DATABASE 就會以此為準、去做「分靈動作」。

如何設定此一「總舵主資料庫」？

請繼續看下去……

### C\_6\_5\_1、總舵主(Schedule\_1)的資料庫之設定

首先到 Log40M 主視窗->Settings-> User Configuration(如下圖所示)



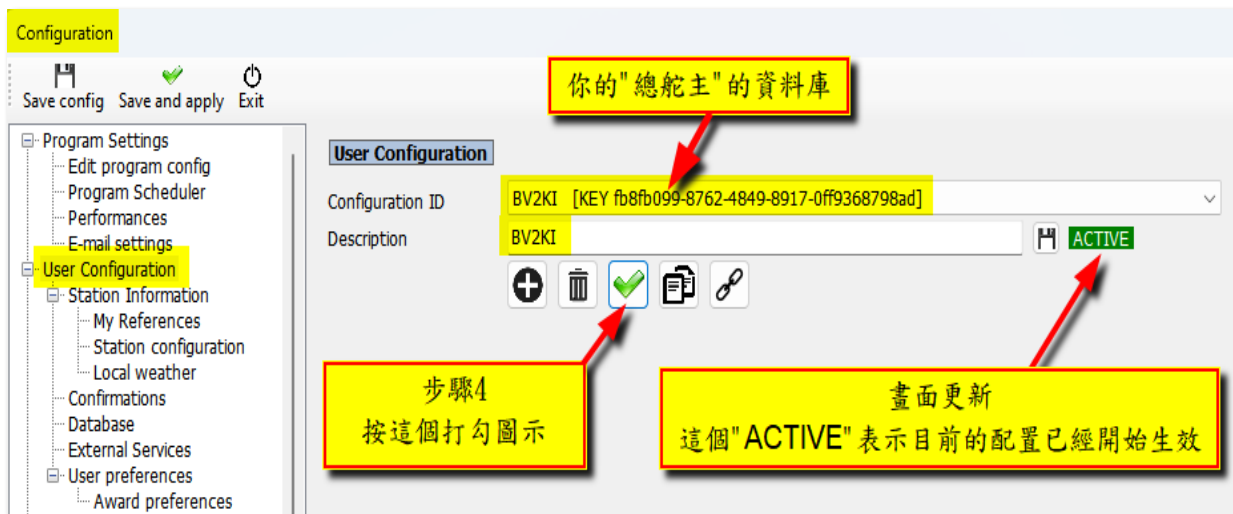
圖\_75 建立使用者基本配置

填寫你的總舵主呼號後，按右側之儲存鈕，將此「首個配置」儲存起來。



圖\_76 總舵主基準資料庫建立完成

步驟 1 至步驟 3 完成，畫面更新後、出現的就是你的「總舵主」資料庫已經完備。



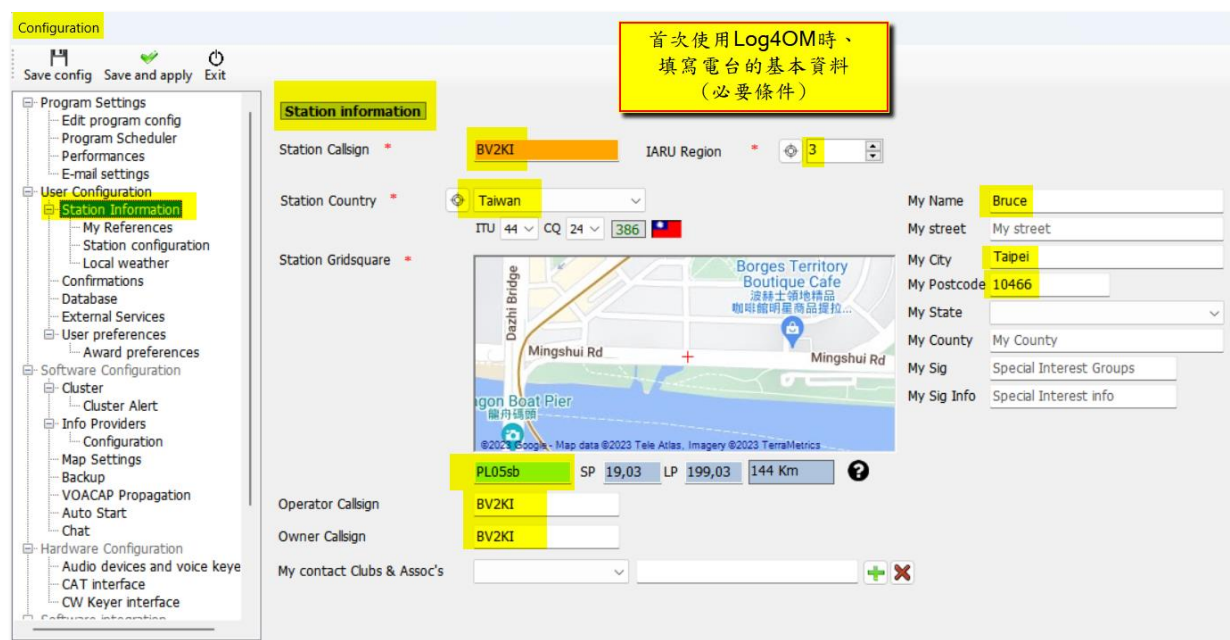
圖\_77 總舵主資料庫已經生效

做完步驟 4 後，可以讓你的總資料庫成為「總舵主資料庫」。同時，在圖\_77 的「Configuration ID」欄位那一行，於末端的向下符號「V」處、按下時，你就可以看到剛剛生效、且唯一的 BV2KI fb8fb0998762484989170ff9368798ad 機碼顯示出來，我們暫稱之為 Schedule\_1。(如圖\_78)。





接著，我們必須指定總舵主的身分資料，請看圖\_80



圖\_80 總舵主 BV2KI 的電台資料登錄

完成總舵主之電台登錄後，至此階段才算完成 Log4OM 的第一個「配置檔」的設定(我們暫時稱之為 Schedule\_1)。

建立一個「配置檔」後，接著 Log4OM 之 Configuration 中、還需要有一些進一步的設定後，才能夠符合整合的需求。這些設定(如圖\_80 之左側的部分)的作業細節，除了本文前面已經敘述過的 [Connection](#)、[ADIF Function](#) 外，還須考慮到 [CAT Interface](#)(這個設定與 Schedule 種類有關)，其餘不相關於整體配合的設定，就不在本文中討論，請您自行前往試試看看 Log4OM 還有哪些功能。

在 Log4OM 應用程式裡有個 CAT Interface 功能，這個介面有四個選項用以讓控制收發機組(如頻率、模式等)或控制相關設備(如天線旋轉器)等資訊與應用程式之間取得共享或同步。這四個選項是：

- 1、Nothing: 無
- 2、[OmniRig](#): 由不同作者用不同語言編寫的多個程式，同時掌控收發機之操作；
- 3、[HamLib](#): 用於業餘無線電整合收發機或天線旋轉器與各應用程式之程式庫介面的應用程式。
- 4、[TCI](#): 一種平台的程式工具命令語言，用以作為控制收發機與應用程式之間的控制介面之開發或實作設計之工具。

本文的四個應用程式，需要用到共享目的者，僅有 Log4OM 以及 N1MM+這兩個應用程式。OmniRig 可以擔任此一任務，唯、只限於 Log4OM 才用得上；在 N1MM+應用程式時，係以利用 Hamlib 程式庫介面去做資訊分享，用不著去採用 OmniRig 介面(N1MM+之手冊中也提到其他運用方法；由於涉及範圍頗廣，N1MM+也不做硬性規定)。因此，在做整合時，我們必須考慮到兩種不同的配置設定，簡言之，以：是否採用以 OmniRig 作為 CAT Interface 作為分野；也可以說「**是否採用 N1MM+之運行、作為分界**」。

**設定方針如下：**

首先、我們先將總舵主的「配置 Schedule\_1」設定成採用 OmniRig，使 Schedule\_1 成為配置主角。然後複製此主角配置的所有設定給予「不採用 OmniRig 的配置」(如 N1MM+應用程式)。我們稱這個「不使用 OmniRig 的配置」為 Schedule\_2。

接著將 Schedule\_2 的 CAT Interface 選項內容、修改成以 Hamlib 的選項。


換句話說：**Schedule\_2 之配置，就是專給使用到 N1MM+時的「專用配置檔」**。其餘者，都是以採用 Schedule\_1 配置檔。

好了！讓我們開始動手開始做設定。

## C\_6\_5\_2、N1MM+(Schedule\_2)的資料庫之設定

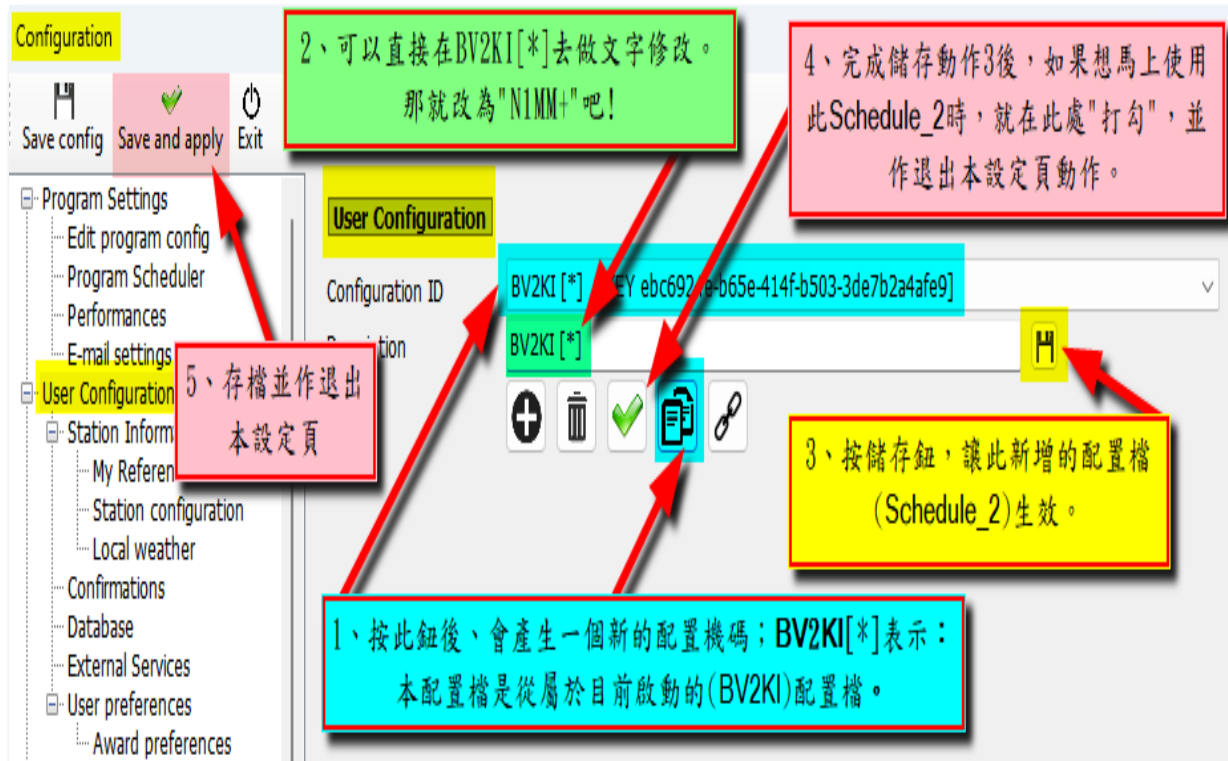
按照前述之設定方針 2，我們需要將目前生效的 Schedule\_1 的設定值，作一「複製或分靈」動作；請先看看圖\_79，回味一下：

- 1、如果仍然使用總舵主的同一個資料庫，那我們就需要以「複製」方式」處理；
  - 2、如果想另起爐灶，用一個「新的」資料庫時，就用「新創」方式」處理。
- (圖\_79 中的註解所示的青色或綠色按鈕)

我們就採用「複製目前啟用的配置」，請在「複製鈕扣」、按一下。

接下來的步驟，就請按照圖\_81 逐步去做。

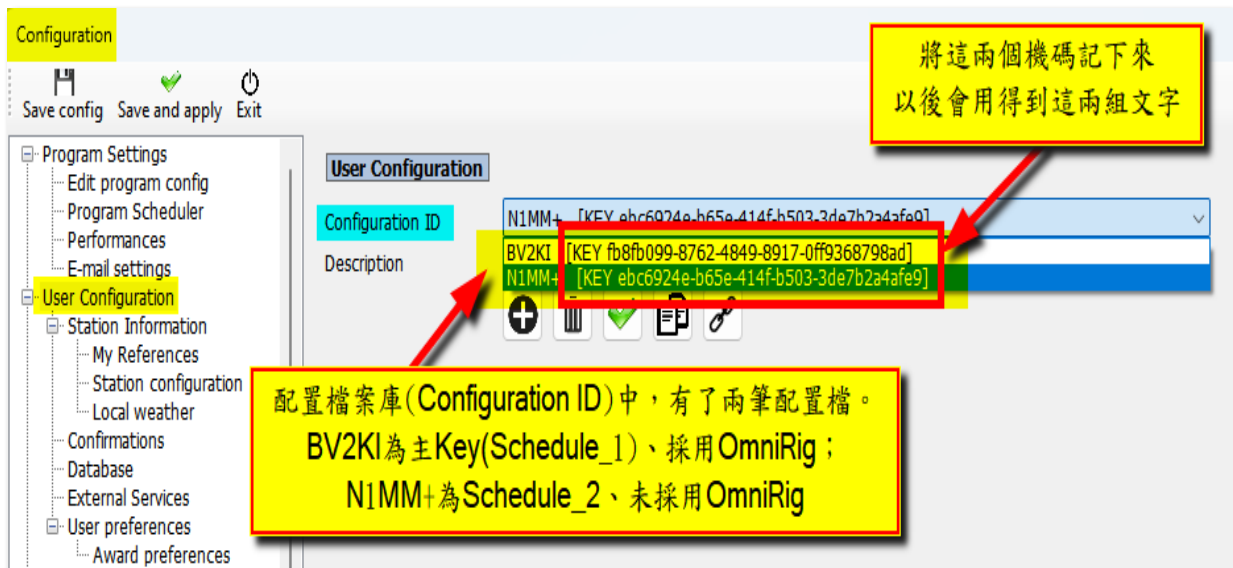
當你將 BV2KI[\*] 完成修改為「N1MM+」(步驟 2)後，就會看到如圖\_82 的成果。



圖\_81 N1MM+(Schedule\_2)之配置檔案複製步驟



圖\_82 將 BV2KI 改為 N1MM+的成果



圖\_83 配置檔案庫裡有兩筆檔案

請將此兩筆機碼暫時以筆抄下來(或照相存檔)，當我們進行至「**使用批次檔操作**」時，就會用得到。

如果想知道這兩組機碼的儲存位置，請看圖\_84。

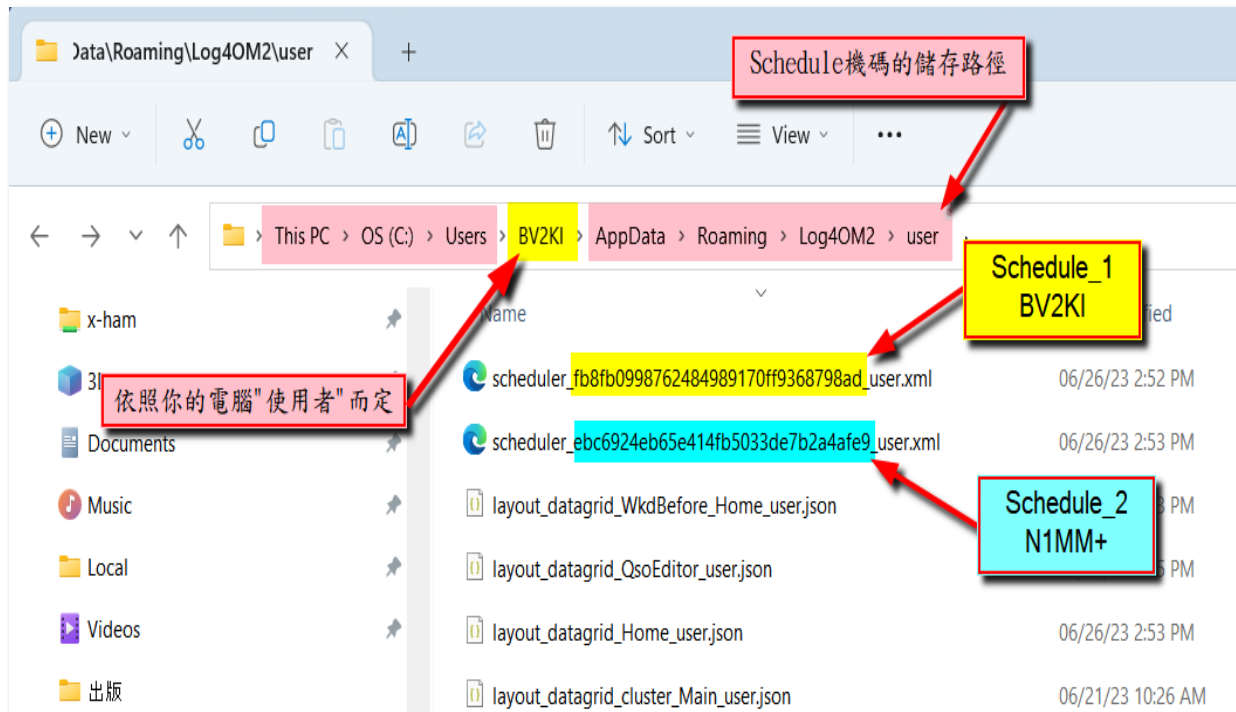


圖 84 儲存機碼的資料夾

既然都已經知道機碼的藏身之處、那就可以直接去 copy 該檔名、不必費心去抄寫機碼了！Hi Hi ！

招頭去尾，就是這兩組機碼了。你不必死記，因為你的設定機碼不會跟此時此刻所看到的一樣；只要記住前述之操作步驟即可。

Schedule\_1 : fb8fb0998762484989170ff9368798ad

Schedule\_2 : ebc6924eb65e414fb5033de7b2a4afe9

有何用處？這是要被「批次檔」用的。請看這裡：

「[如何在單一電腦上同時操作兩個以上的 WSJT 以及與其他軟體的整合設定.docx#以批次檔 BatchFile 方式啟動](#)」。

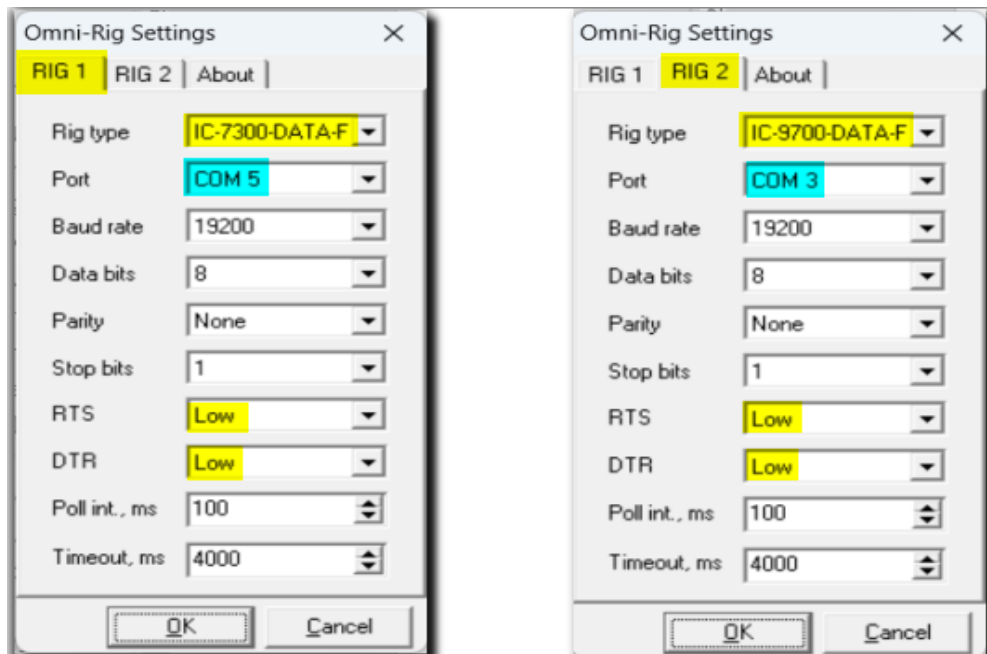


## C\_6\_6、CAT interface 之 OmniRig 的設定

OmniRig 是一個程式與程式之間的一個共用介面。我在處理本文的時候，想把收發機的信息與 Log4OM 共享；比如：頻率、模式甚至於往後會試者去發揮 Log4OM 的其他數據通訊等功能。因此，我將 IC-9700(VHF 用)與 IC-7300(HF 用)的資料藉由 OmniRig 來傳遞給 Log4OM 用。當然，利用 TCP/UDP 也是 log4OM 可以支援的。可是我覺得使用 OmniRig 是比較容易理解的

OmniRig 像似一個水庫，他有兩個槽([V2.0](#) 版則有四個槽)，我把 IC-9700 及 IC-7300 的資料動態的分別儲存到 OmniRig 的桶槽裡，利用 COM 埠的傳輸功能讓其他的應用程式來取用或向著收發機來發號指令(這裡的應用程式當然是指向 Log4OM 而言)。如此，我就可以使用 Log4OM 的許多功能，來做各個機組之 **JT** 所建立的 QSO 紀錄檔案，做儲存、分類、分析等工作。同時也仍然可以享用 Log4OM 的其他功能，例如 Web Cluster 的最新電台動向 PO 文、傳播狀況檢視、Alert 功能(一種出現了想要的稀有的通報機制)等等。

因此，先將 IC-7300 及 IC-9700 的相關資料填入 OmniRig 後，可以儲存起來並關掉 OmniRig。這個關掉的動作並不會影響其他的應用程式引用；因為，只要有啟用該應用程式(如 Log4OM)、且該應用程式已經有設定採用 OmniRig 的需求時，該應用程式自然地會去呼叫 OmniRig 放到電腦的系統中，供給該應用程式使用。



圖\_85 Rig\_1 設定為 IC-7300；而 IC-9700 設定給 Rig\_2 用

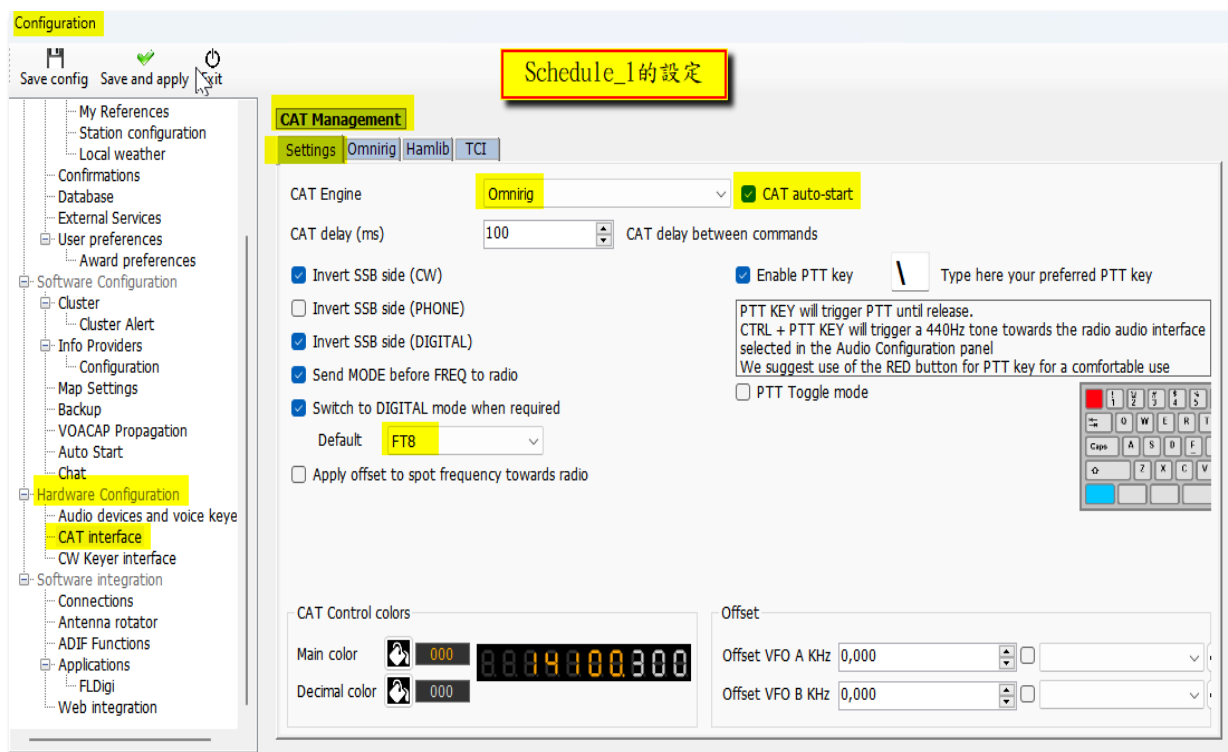
## C\_6\_6\_1、總舵主(Schedule\_1)的 CAT interface 設定

為了設定 Schedule\_1 的 CAT interface，我們先要確定目前啟用的「配置」是否在 Schedule\_1 之啟用？

請到 Log40M 的主設定頁面 Configuration-->Hardware Configuration，看看「User Configuration」的「Configuration ID」之內容是否以「BV2KI」為前導的機碼，如果看起來就像圖\_77 一樣(注意：「Active」有出現)，這就表示，我們確實是處於 Schedule\_1 的「配置」。

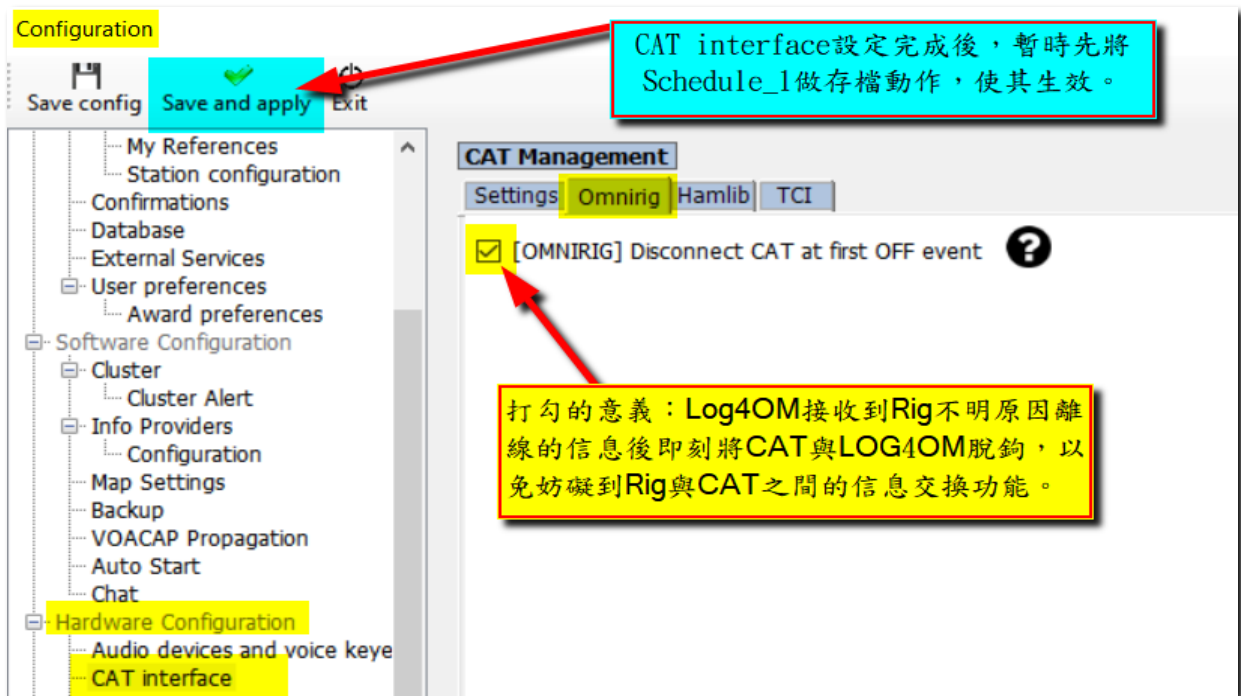
我們就開始來為 Schedule\_1 做設定；請到 Log40M 的主設定頁面

Configuration-->Hardware Configuration-->CAT Interface-->Settings (如圖\_86)



圖\_86 Schedule\_1 的 CAT 的設定

按照圖\_86 逐步去設定 Settings 的選項。接著，左鍵去啟動 Settings 之右側的選項卡 OmniRig 後，可以看到如圖\_87 的唯一選項、打勾！。



圖\_87 CAT\_OmniRig 的防呆裝置勾選

勾選了 OmniRig 的防呆裝置後，此刻已經完成了總舵主的「專用配置檔 Schedule\_1」的必要項目設定(其他非必要的項目，就請您再去斟酌做設定)。完成此階段設定後，為了確認起見，請依照圖\_87 中的青色指示動作先去做一次存檔、以免忘記、做白工！。

### C\_6\_6\_2、N1MM+(Schedule\_2)的 CAT Interface 設定

現在讓我們來做 Schedule\_2 的 CAT interface 設定。

前面已經說過，Schedule\_2 事實上是為了不採用 OmniRig 的應用程式而做的設定。N1MM+就是屬於 Schedule\_2 配置的一種。

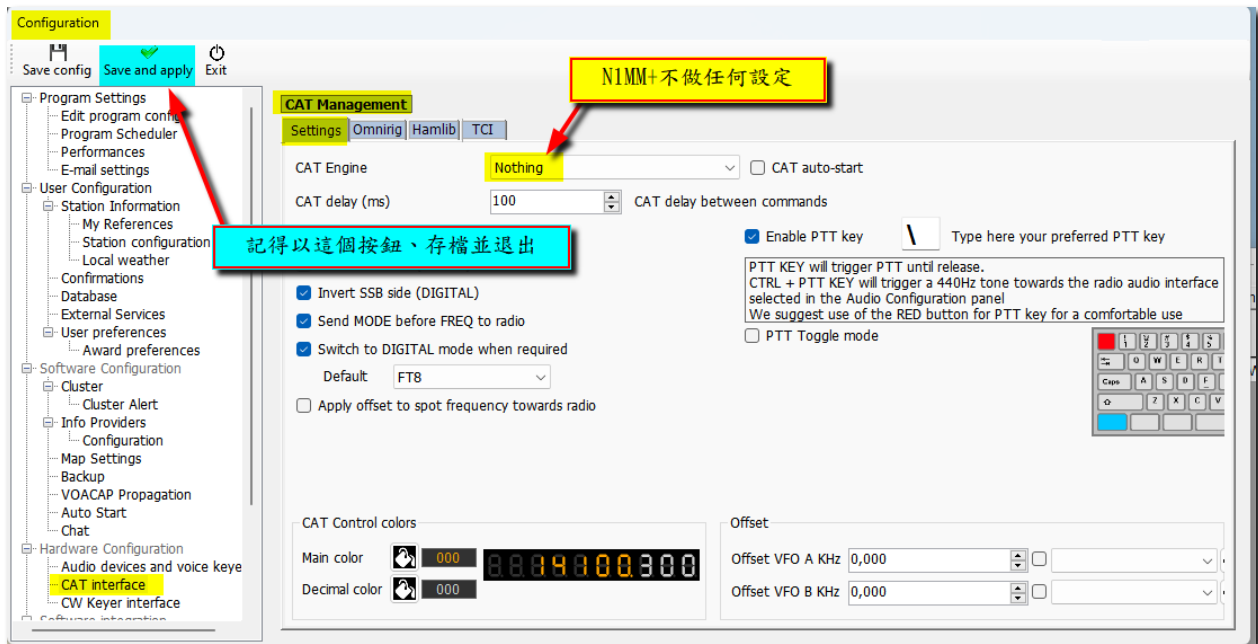
首先、我們要將啟用 Schedule\_2 配置，讓他成為 Active 狀態。要這樣做：

我們回到圖\_77 看看，在「Configuration ID」的選項框裡將 N1MM+的那組機碼「點選」出來，然後去按綠色的鉤鉤「」，讓這個 Schedule\_2 呈啟動狀態。

接著、請到 Log4OM 的主設定頁面

Configuration-->Hardware Configuration-->CAT Interface-->Settings (如圖\_88)

我們選用「Nothing」，然後按「Save and apply」存檔並退去。



圖\_88 N1MM+(Schedule\_2)的 CAT Interface 設定(N1MM+不做任何設定)

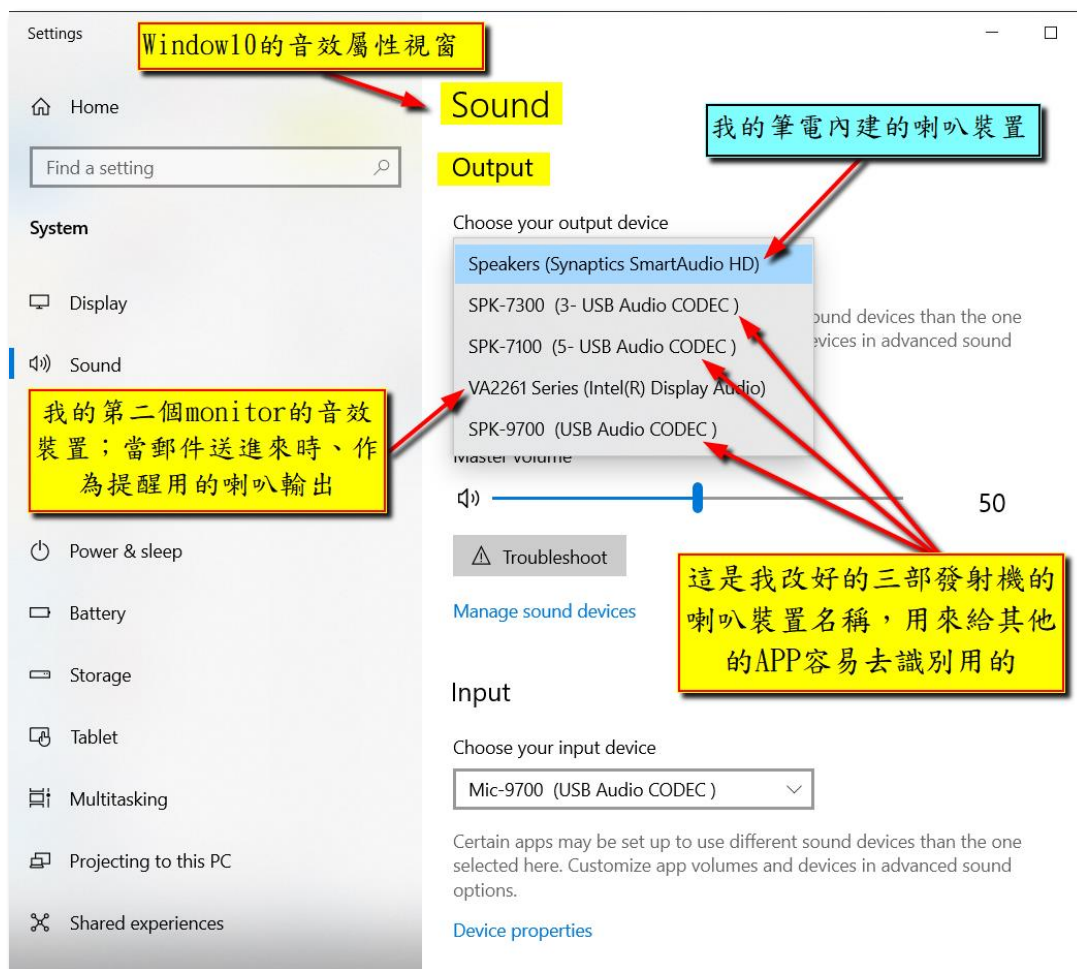
做完這個 CAT interface 動作後，為了保險起見，我們一定要將 Log4OM 結束後再將它啟動一次後、才能繼續「玩」下去。

## C\_7、電腦的音效設定及測試：

### C\_7\_1、電腦本身內建音效設定

WSJT-X 程式的主要裝置，以音效卡產生的聲頻為主角，因此發射機的音效卡（有默認的、如 ICOM 最近的機型。也有外接式的，如 [DigRig](#) 或 [RigBlaster](#) 之類的介面）與電腦的音效卡需要明確的作區別；不能混用。

在開啟單機或多重機組作業之前，先扯掉所有的發射機的 USB 連接纜線（或其外接的音效介面），讓電腦的音效成為預設狀態（電腦上的左下角 **Start**→**Setting**→**Sound** 可以看得到的）。此圖\_89 係以 Windows 10 做說明的。Windows 11 之用戶，請比照圖\_90 至圖\_92 逐步處理。



圖\_89 電腦本身專用的喇叭及麥克風的設定(Windows 10)



圖\_90 電腦本身專用的喇叭及麥克風的設定(Windows 11)





圖\_91 音效-喇叭設定



圖\_92 音效-麥克風設定

圖\_92 是調節麥克風接收度的指標、以 30db 左右為最佳。



圖\_93 音效-麥克風調制設定，使 WSJT-X 的接收程度約為 30db 左右

圖\_93 是在 JT 之主視窗的左下角的截圖。Mic 音量滑桿一詞、是指圖\_92 中所述者。

## C\_7\_2、音效測試

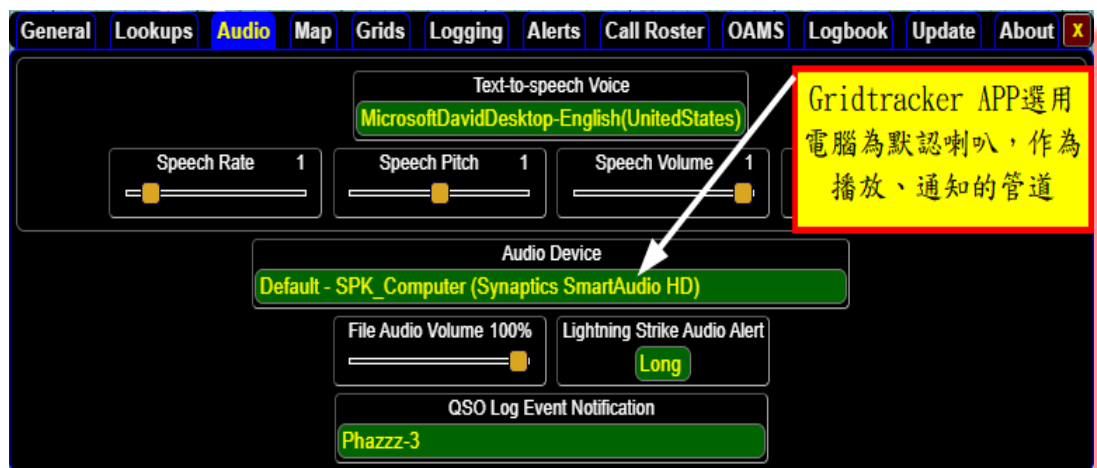
測試電腦喇叭輸出是否正常；此一動作是要確保電腦的喇叭輸出沒有被誤植為其他收發機的 USB CODEC 或被誤植到其他的收發機相關的喇叭輸出裝置(如使用到 [RigBlaster](#) 介面等)。

例如；Gridtracker 及 Log40M 程式可能會使用到 Alert(警告)或通知(Notify)功能時，當然要去選用電腦內建的或電腦本身的外接喇叭；以避免該類之警告或通知的音源輸出至你的收發機的喇叭裝置裡。如果真有此事、一旦處於發射設狀態時，該警告或通知的內容就會隨著電波發送出去，讓你成為「廣播電台」、而你還不自知。

在開始使用 **JT** 之前，最好先去測試一下各個機組單獨使用到正確的喇叭與否；也就是需要確定各個收發機相關的喇叭及麥克風的音量大小的調整，都是要處於正常狀態的。

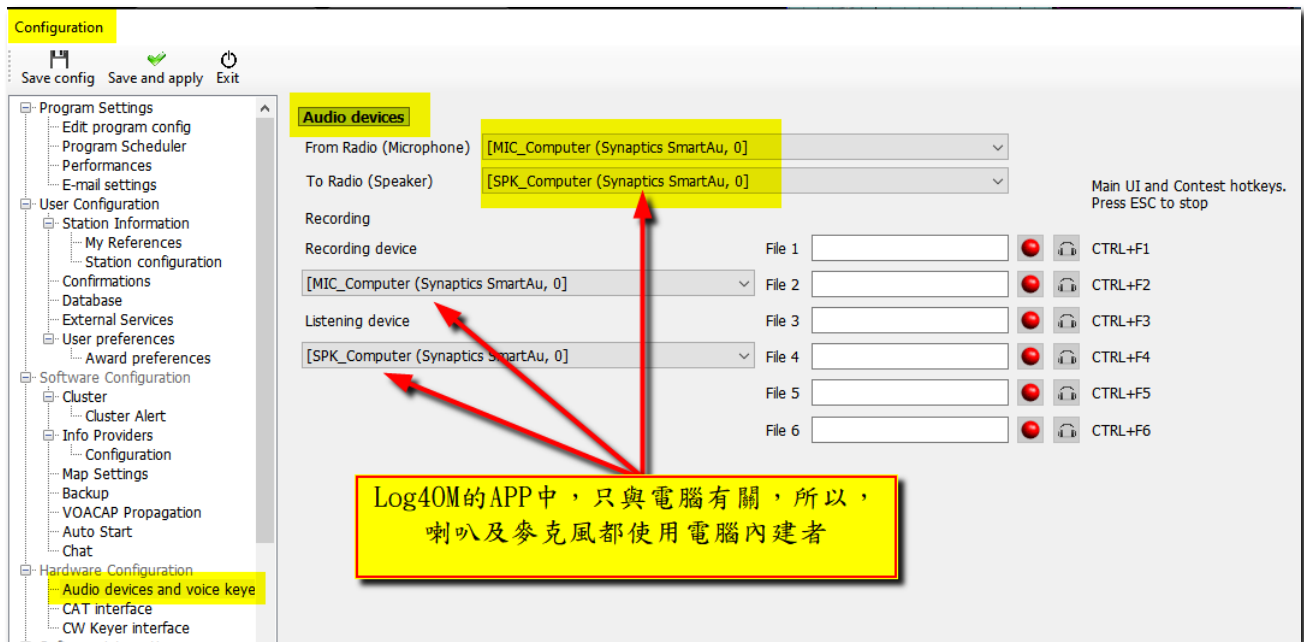
一旦開啟多重機組作業時，各機組將會各自去選擇該機組前已設定完成選用其應有的音源設置(喇叭及麥克風)，其他的周邊程式也會依照前述的喇叭選擇至電腦的默認者。

圖\_94 是 Gridtracker 之相關的「通告用喇叭輸出的選項」；Audio Device 顯示的喇叭名稱「Synaptics SmartAudio HD」就如同圖\_89 中「我的筆電的喇叭裝置」所示的英文名字一樣。不能選錯喔！



圖\_94 Gridtracker 的喇叭選項

同理，我們來檢查一下在 Log40M 應用程式中的「通告喇叭」之設定，是否指向「我的筆電的喇叭裝置」「Synaptics SmartAudio HD」。請看看圖\_95。



圖\_95 Log4OM 的喇叭設定

沒錯，圖\_95 的喇叭及麥克風選項都是指向 Computer 指定的名稱「Synaptics SmartAudio HD」！OK 的！

接下來，我們需要分別利用各 **JT** 主畫面的「TUNE」按鈕，先測試發射機是否被啟動且有功率輸出；若發現 PTT 有動作而發射機無功率輸出，這表示輸出的音頻訊號沒有輸出至該收發機機組的喇叭中；或者、你可能聽到確有喇叭發出單一頻率的聲頻信號，請追溯此聲音是否為電腦默認喇叭的輸出？如果是的話，那就是該收發機機組有選錯了喇叭輸出音源的結果。

解決方式：從 **JT** 主視窗畫面 **SETTINGS→AUDIO** 去檢查 INPUT 及 OUTPUT 是否選對(看看是否選對了你已經更名過的喇叭或麥克風)？若調整選項後，仍無法正常發射時。則退出 **JT** 程式，去到電腦的 **START→ Device Manager** 裝置管理員(圖\_126)，檢查音效卡是否存在或者有不正常之警訊出現；大多數的情形，應該是 CODEC 的「驅動程式」出了問題；尤其是 Windows10 升級至 Windows11 時，「驅動程式」過於「太新」所致。必要時、可能需要找回原廠出品者、「重灌」該收發機的 CODEC 驅動程式。

如果是屬於「Silicon Labs」的產品時，請到這個網頁去下載「CP210x Universal Windows Driver」。解壓縮後，找到「silabser.inf」檔案，以右鍵輕點之，找出「Install」後，任其自行完成安裝作業。

這裡有一個「解決驅動程式失效的小秘訣」。試試看，可能有幫助。

有關音效裝置的詳細設定，請參考下面這篇文章

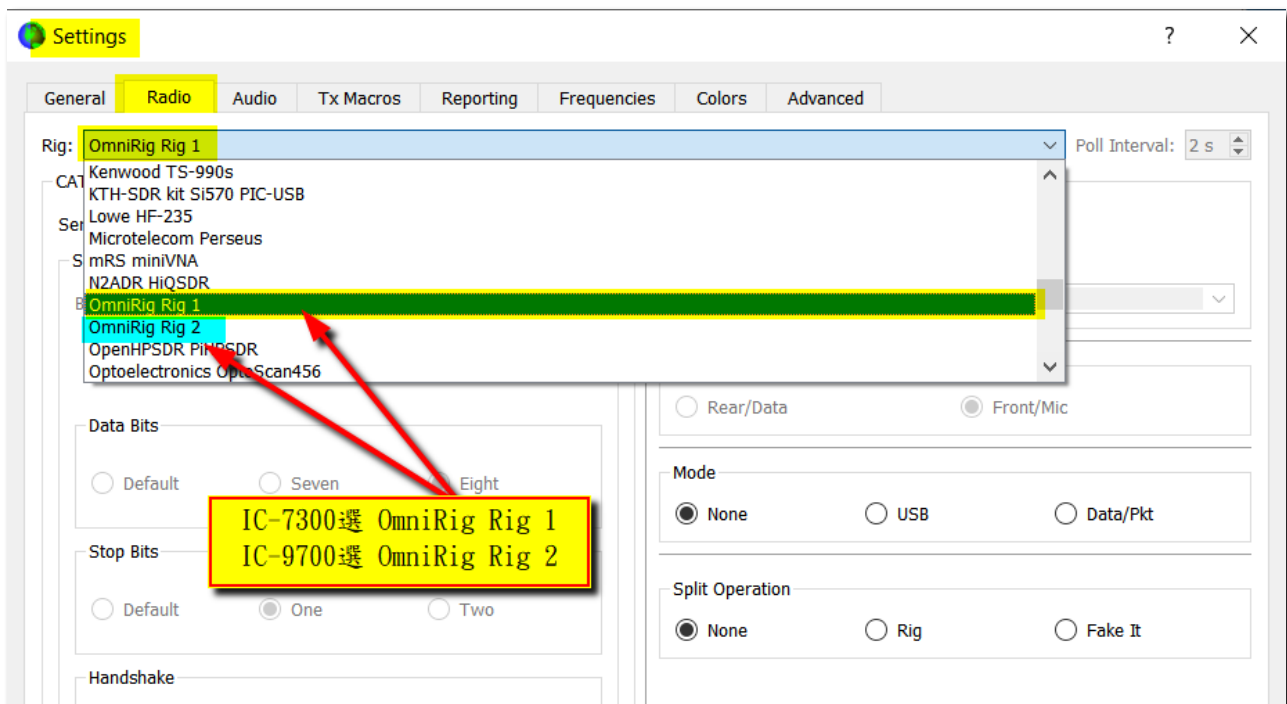
「在 WSJT-X 中如何調校出純淨的音質輸出及其相關的一些小技巧」

## C\_8、操作常規 WSJT-X(或 JTDX)的 Radio 設定

前述的 C\_2\_2 或 C\_2\_3 章節所提到的如何做 **JT** 的 Radio 設定，那是指「在 N1MM+的環境下操作 WSJT-X」所做的設定動作。

本節所謂針對常規 WSJT-X(或 JTDX)的 Radio 設定，是指不在 N1MM+的環境下的一般作業。因為前述的 Schedule\_2 就是針對 C\_2 章節所提到的做設計。至於常規作業採用 OmniRig 與 Log4OM 共享「資源」的前提下，此一 Schedule\_1 的設定就應該如圖\_96 所示。

開啟 WSJT-X 程式後 Setting→Radio，設定如下圖

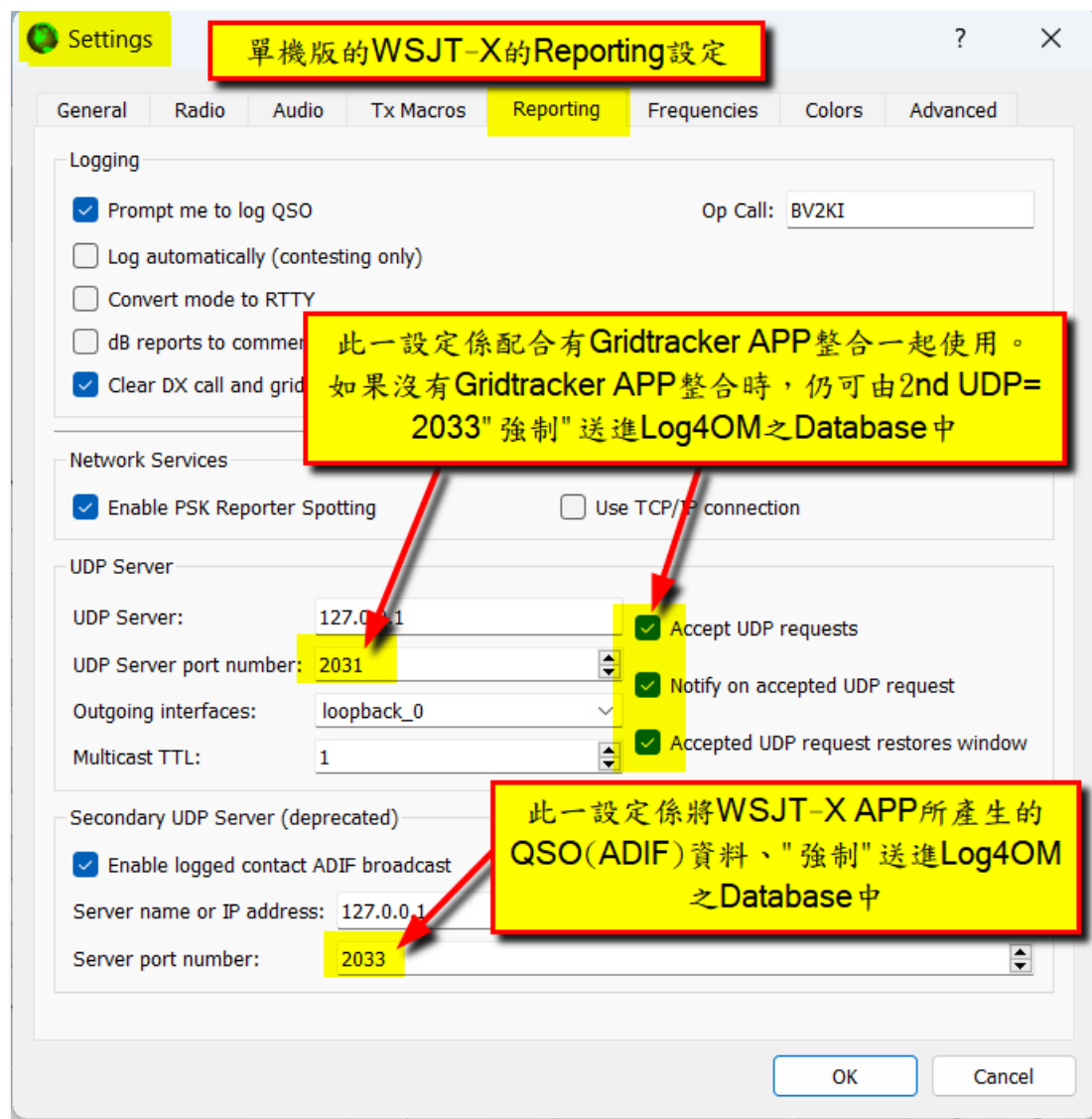


圖\_96 WSJT-X(或 JTDX)的 Radio 設定值與 OmniRig 要相配(給 Schedule\_1 用的)



## C\_9、操作常規 WSJT-X(或 JTDX)的 Reporting 設定

前述第 C\_2\_7 節中所敘述的 Reporting 設定係針對 **JT** 配合 N1MM+的環境下而做的設定。如果不在 N1MM+環境下的常規使用 WSJT-X 應用程式時，哪就按照圖\_97 設定即可。



圖\_97 WSJT-X\_單機版之 Reporting 設定(常規運行)

如果運行 JTDX 應用程式時，請按照圖\_98 設定。

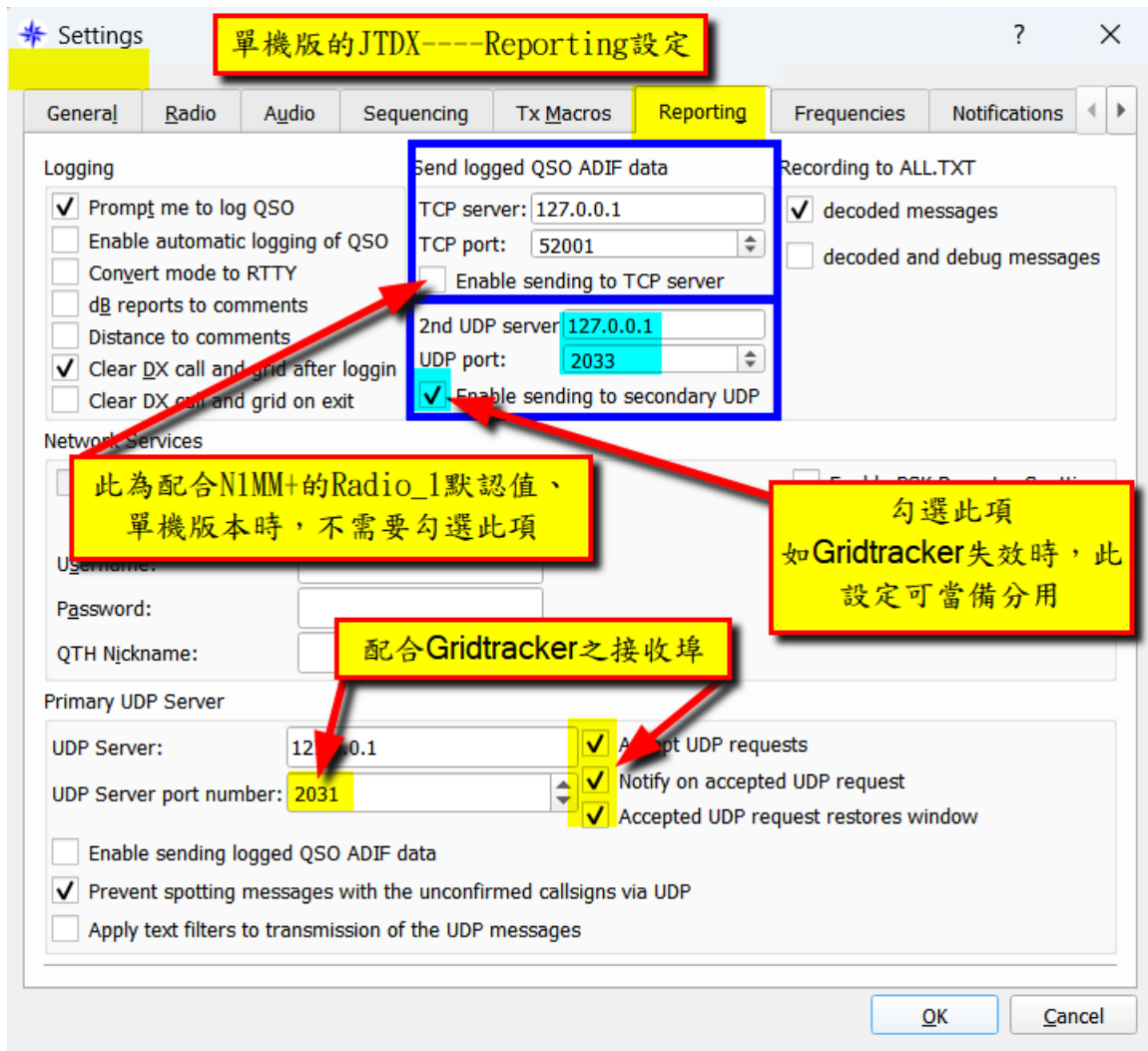


圖 98 單機版 JTDX 的 Reporting 設定(常規運行)

## C\_10、操作常規 N1MM+時的 ADIF Function 檔案移轉

本文前面所述的整合 N1MM+四個程式運行時，皆以 JT 模式為主，所產之 QSO 之 ADIF 檔案，都可以用 Log4OM 的 Connection 送進總舵主的 DataBase 中，加上利用 Log4OM 的「ADIF Function」做「事後補充」，看起來完美。

但是、如果我們仍然要以 N1MM+操作其他模式(如 CW、SBB、RTTY 或 PSK31 等)比賽或一般常規所產生 QSO 的 ADIF 檔案時、要如何將該作業所產生的 QSO 資料、轉進總舵主的 DataBase 中？

其實，我們在 Log4OM 的 Connection 設定時已經略有說明，或許你匆匆閱讀過、好像沒有印象??!!

那就請按(儲存進到 Log4OM 的 DataBase 中)去回味一下，然後按「ALT+←」鍵回到此處。

換句話說，前述 C\_6\_1 節第一段設定的 UDP=12060，就可以達到 N1MM+的所有 QSO 資料，都可以藉此 Connection 的 UDP=12060，會自動的「移轉」至總舵主的 DataBase 中。

所以本節標題「操作常規 N1MM+時的 ADIF Function 檔案移轉」，只是做為提醒你注意而已。

如果 UDP=12060 設定無誤時，你可以在 N1MM+的 EW1(或 EW2)之 Call 框框中、隨便填一個呼號，然後按「Return 回車」；去到 Log4OM 之 QSO 視窗檢查一下，這個測試的呼號應該已經傳到總舵主的 DataBase 中。

除此之外，我們也可以用人工方式處理：


首先、將 N1MM+比賽結束(或平常 QSO)後的檔案匯出，給個固定之路徑及檔名，例如：

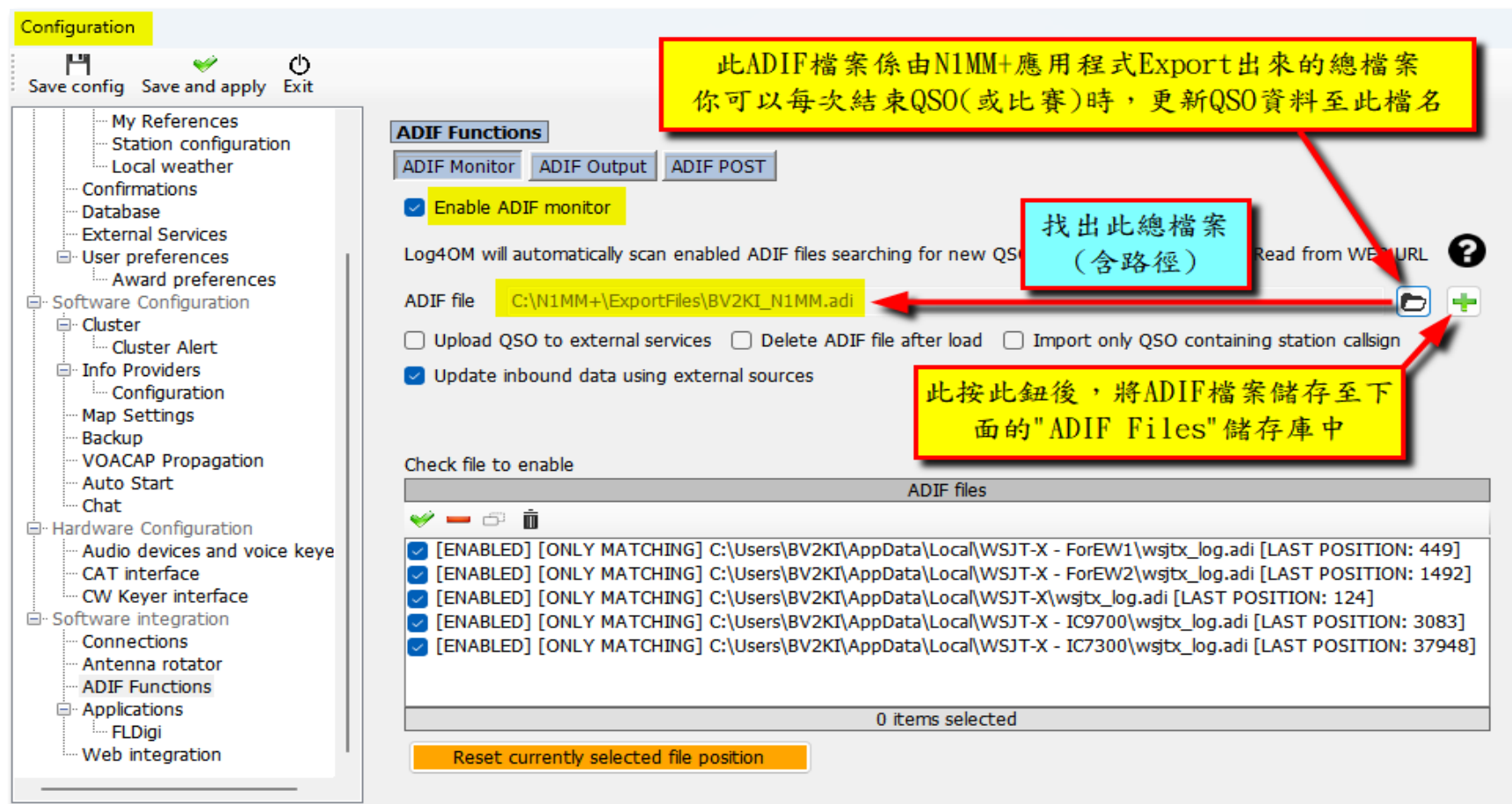
「C:\N1MM+\ExportFiles\BV2KI\_N1MM.adi」(這個 N1MM+匯出去的路徑 C:\N1MM+\ExportFiles 是不會變的，省事不少)。

有兩種手工方式處理：

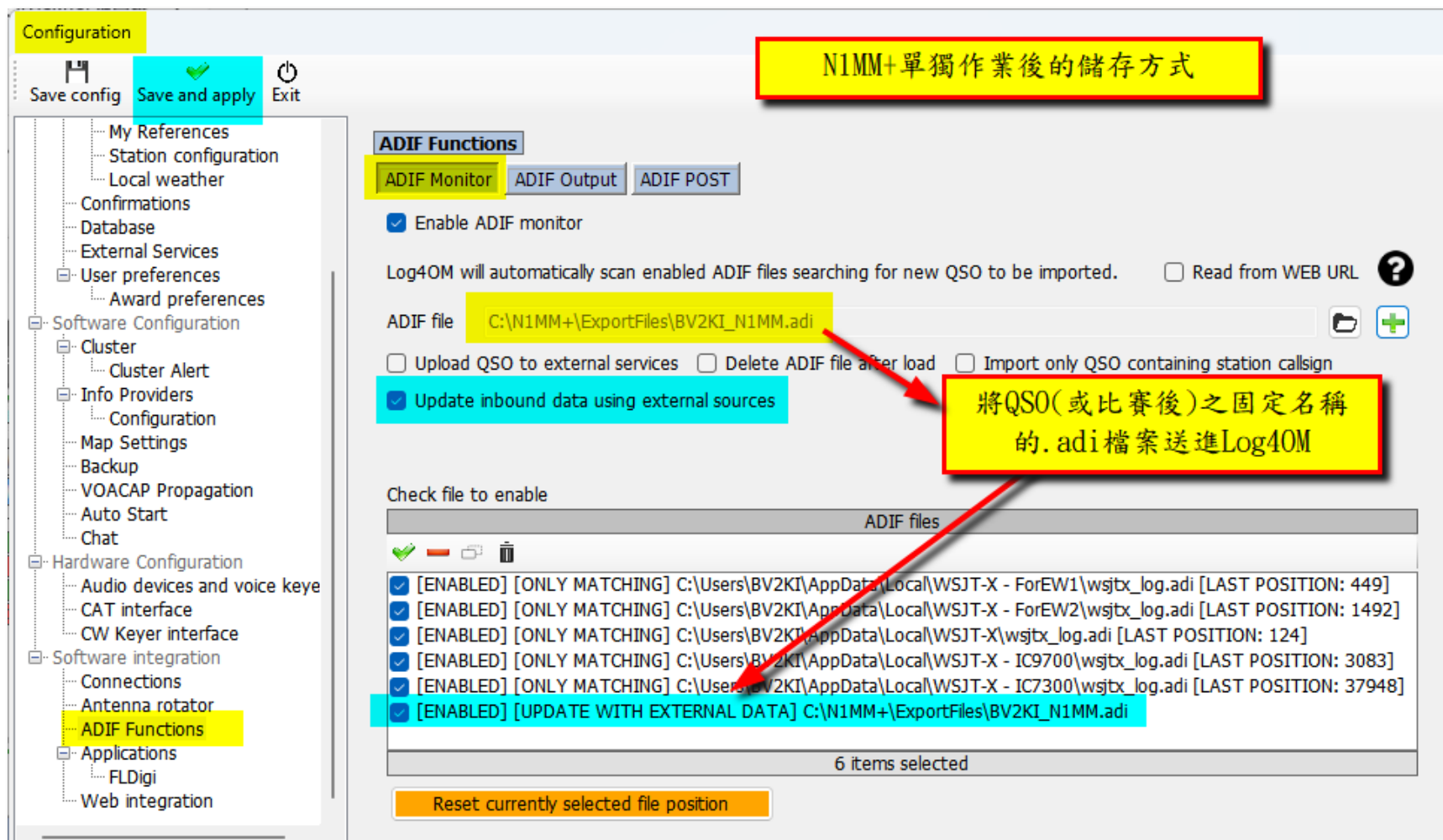
- 1、手工方式將該檔案 import 至 Log4OM 之 DataBase 中、或
- 2、引用 Log4OM 的 ADIF Function 功能將 N1MM+所產生的 QSO 資料自動移轉過來。

此處就以第 2 種方式說明如下:如同前述 C\_6\_2 節所述的設定「ADIF Function」方式，如圖\_99 所示：

找出 N1MM+ 的 .adi 檔案「C:\N1MM+\ExportFiles\BV2KI\_N1MM.adi」填寫至黃色欄位內、按「」號，將檔名儲存至 ADIF Files 庫中。



圖\_99 找出並填寫 N1MM+ 的匯出檔



圖\_100 N1MM+的總檔案已經入庫

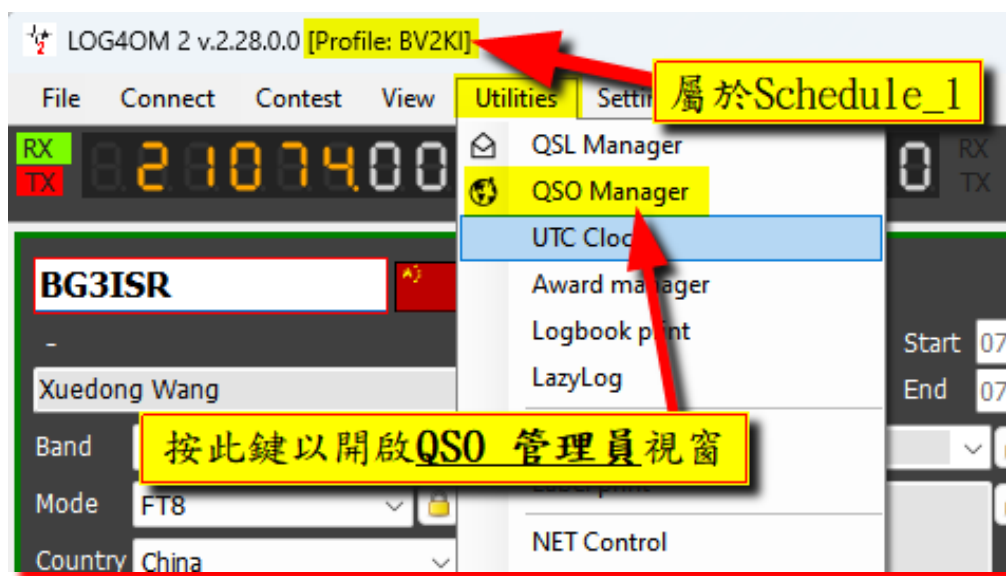
最後、不要忘了去按「Save and apply」鈕，做存檔動作

## C\_11、Log4OM 的單一欄位更新功能(KI Format 的套用)

這兩種方式在移轉至 Log4OM 後，再以手工方式去到 Log4OM 的一個非常有用的一個叫做「Single field update 單一欄位更新」功能，將 Comment 欄位改為「KI Format」的更新，以便於在總舵主資料庫中易於辨識該 Logging 屬於誰的。

方法是：

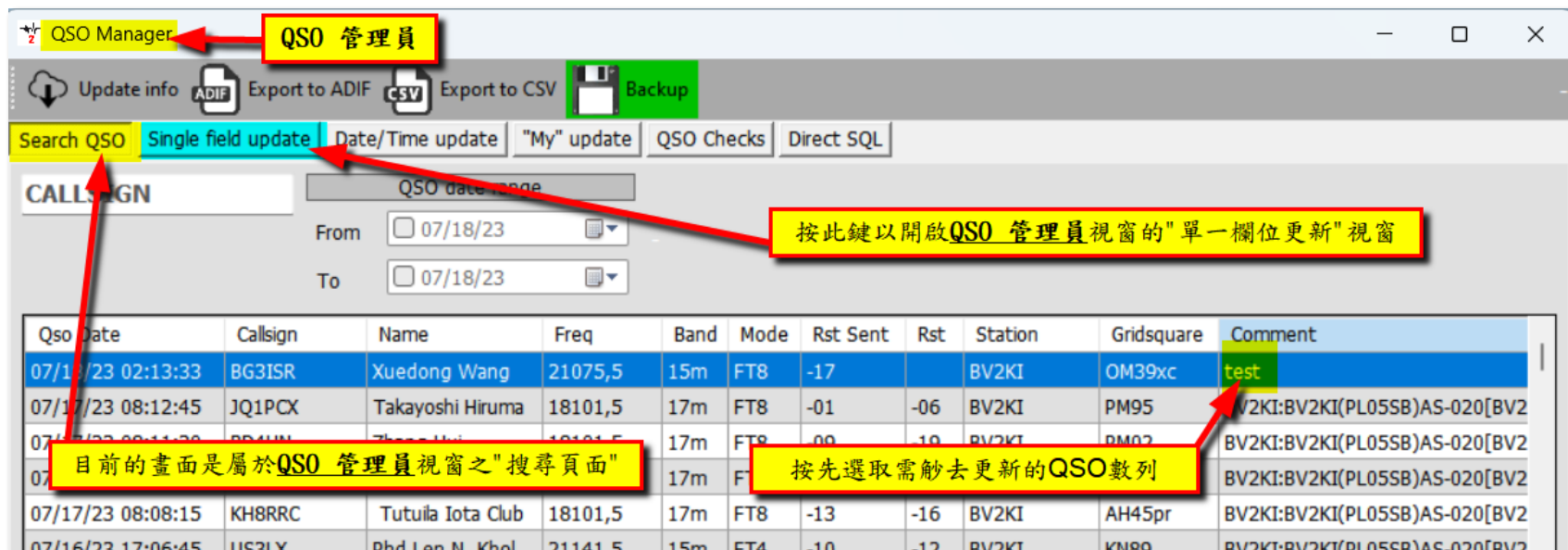
請先到 Log4OM 的主視窗 **Utilities**→**QSO Manager** 開啟 QSO 管理員(圖\_101)



圖\_101 開啟 Log4OM 之 QSO 管理員視窗

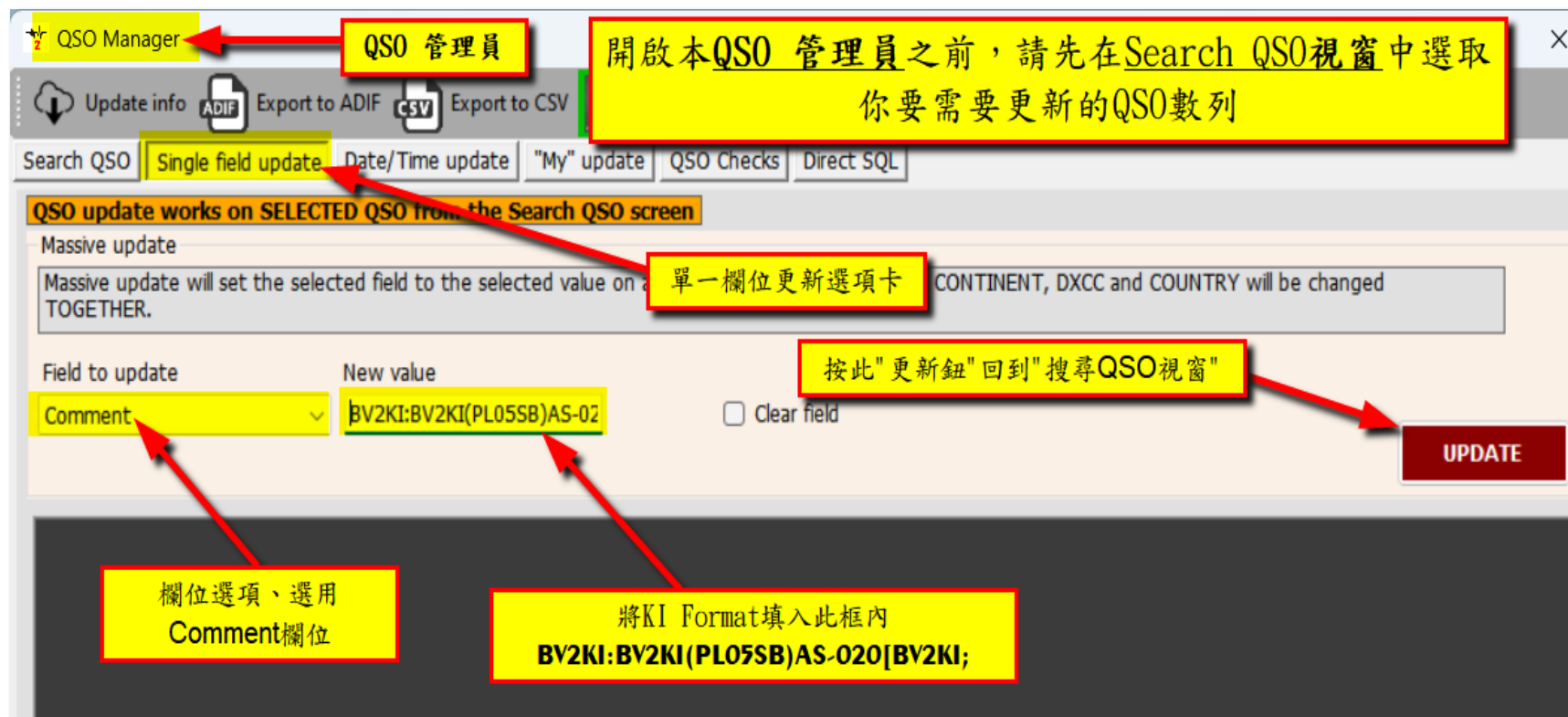
開啟如圖\_102 的 QSO 管理員後、選擇需要去更新的 QSO 數列；本例選了 BG3ISR 的 Comment=「test」的這筆 QSO 紀錄。





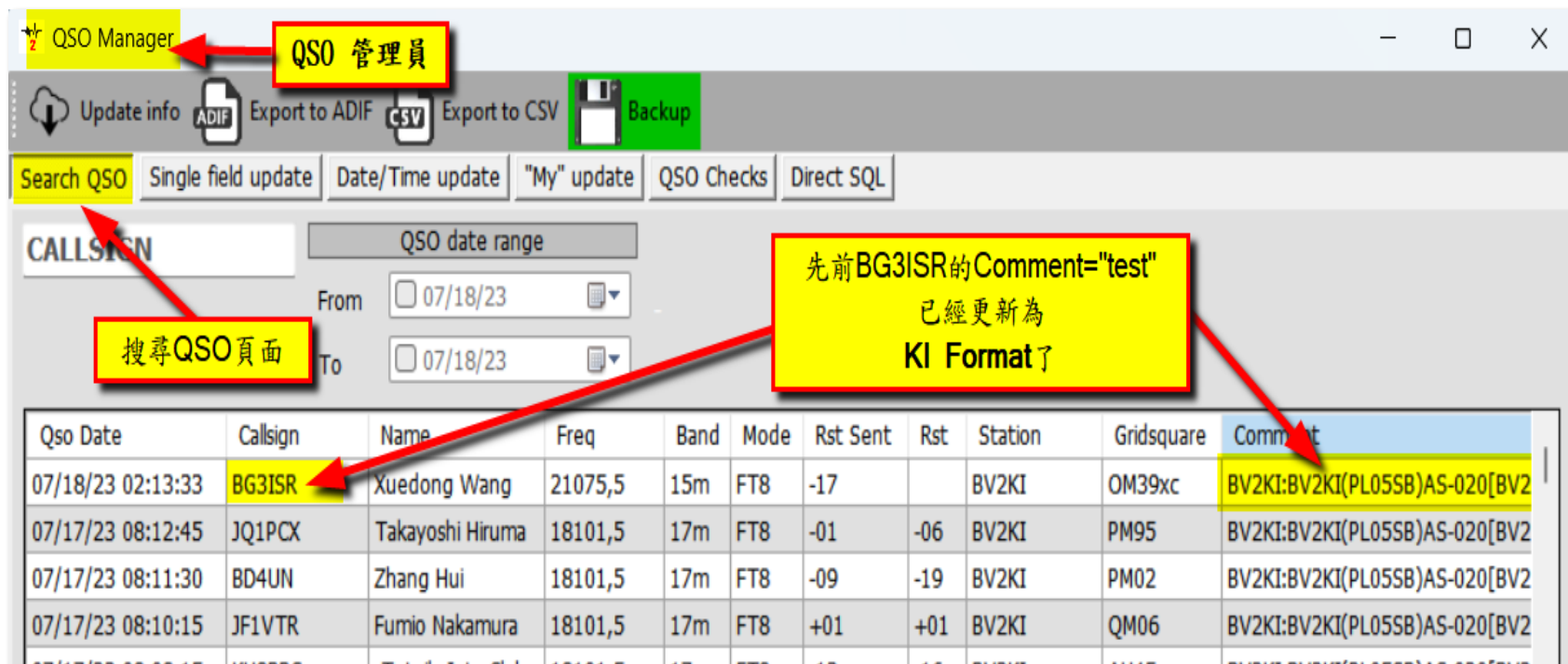
圖\_102 Log4OM 之 Search QSO 視窗

開啟了如圖\_103 的「單一欄位更新視窗」，接著請按照圖\_103 的步驟執行。



圖\_103 Log4OM 之單一欄位之更新

至此圖\_104 後，你已經完成 QSO 的「單一欄位更新」。恭喜你！



圖\_104 Log4OM 的單一欄位更新完成

上述的「BV2KI\_N1MM.adi」檔案內容，你都可以一直累加進去；因為 Log4OM 的「ADIF Function」會記得上次他已經選用過的資料位置。累加進去的資料不會重複。如果不怕麻煩時，你也可以將「BV2KI\_N1MM.adi」的檔案、僅以當次比賽(或當天的 QSO 資料做一次性 Reset 更新；然後進一步以圖\_65 之 Log4OM 的 ADIF Function 的設定)方式去做「Reset」動作，作為「今日事、今日畢」之標竿原則，也是可行的。

## D、配置檔與批次檔

### D\_1、綜合設定(四程式之間的 UDP 及 ADIF 的設定)

本文到此，已經分別將 N1MM+配合 WST-X(或 JTDX)的設定以及配合 Gridtracker 與 Log40M 四個應用程式之整合，完成了詳細的解說。

這四個應用程式的各自擁有其基礎設定，或具有其他特色的功能設定，由於篇幅關係、本文目前或許沒有對於這些功能、多加著墨，不過以後如有心得時，會以「後記」的方式增列於相關章節之中。

有關需要整合的部分，請確實按照本文所述有關 N1MM+的各項「基本項目(圖\_1至圖\_9)」做設定外、請您按照下列所述之關係圖\_107(及圖\_108)，去做各程式之間的 UDP 籍 TCP 等細部資料的設定。

下列的描述係來自於圖\_36之總體考量後所做的一個綜合設定。

#### 四程式之間的 UDP 及 ADIF 的設定

A, Gridtracker :接收 UDP=2031, 推播 UDP=2032(需要勾選「啟動」按鈕)。圖\_51

B, Log40M :UDP=12060(ADIF 檔案來自 N1MM+, 屬 Schedule\_2); UDP=2032(來自 Gridtracker 之推播); UDP=2033(若 Gridtracker 失效時、作為來自 WSJT-X/JTDX 或 MSHV 之備份)(UDP=2032 及 UDP=2033, 屬 Schedule\_1)。  
見圖\_62、圖\_63及圖\_64

C, WSJT-X(N1MM+環境下啟動者):兩機組之 1st UDP=2031; 機組\_1 之 2nd UDP=2037; 如有機組\_2 時, 其 2nd UDP=2239。見圖\_47及圖\_48

D, WSJT-X(常規啟動時者):兩機組之 1st UDP=2031; 2nd UDP=2033。見圖\_97

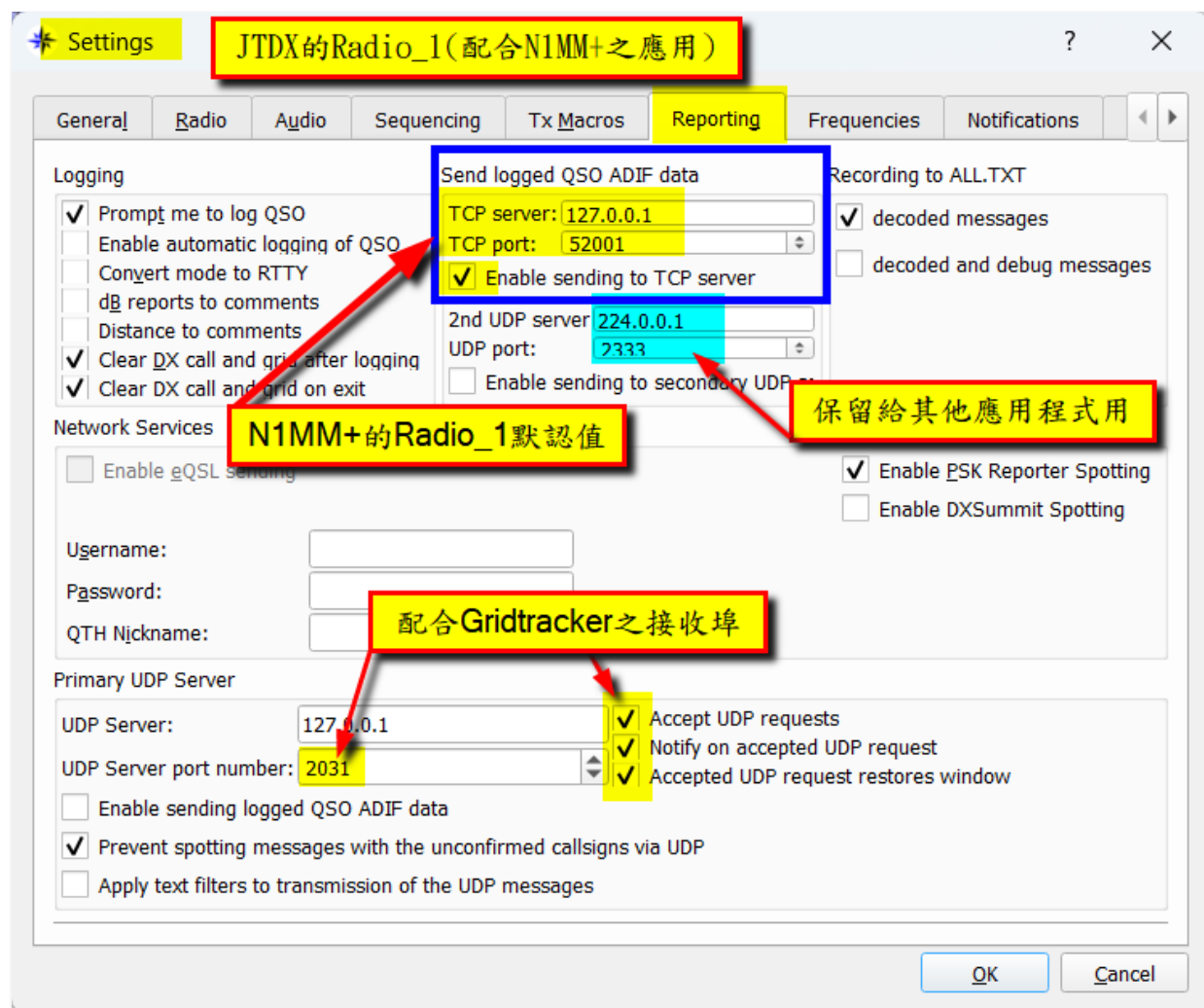
E, JTDX(N1MM+環境下啟動者):兩機組之 1st UDP=2031、2nd UDP=「無」  
機組\_1 之 TCP=52001; 如有機組\_2 時之 TCP=52006。見圖\_105及圖\_106

F, JTDX(常規啟動時者): 每個機組之 1st UDP=2031、2nd UDP=2033。見圖\_98

G, N1MM+ :運行 WSJT-X 時: 機組\_1 之 1st UDP=2337; 如有機組\_2 時之 UDP=2339。

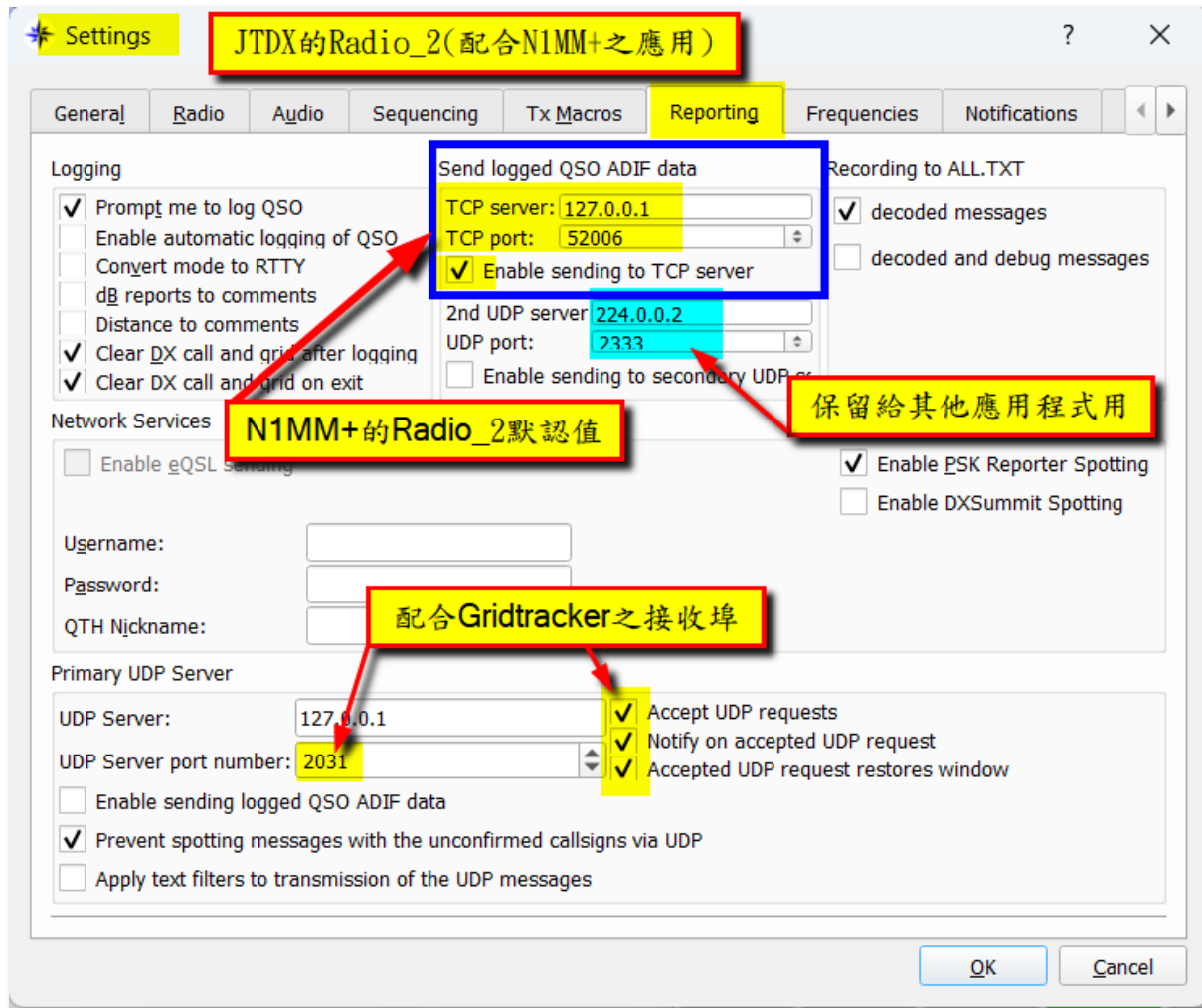
運行 JTDX 時: 機組\_1 之 1st TCP=52001; 如有機組\_2 時之 TCP=52006;  
如果要使 N1MM+之 Telnet 生效時, 需要再將 WSJT-X 的 1st UDP=2237 者、也要勾選。(見圖\_8)

H, Schedule 的設定(以採用 N1MM+與否作為區分)。



圖\_105 JTDX\_Setting\_Reporting\_Radio1(配合 N1MM+)

圖\_105 及圖\_106 是配合 N1MM+，用於 JTDX 的設定。



圖\_106 JTDX\_Setting\_Reporting\_Radio2(配合 N1MM+)

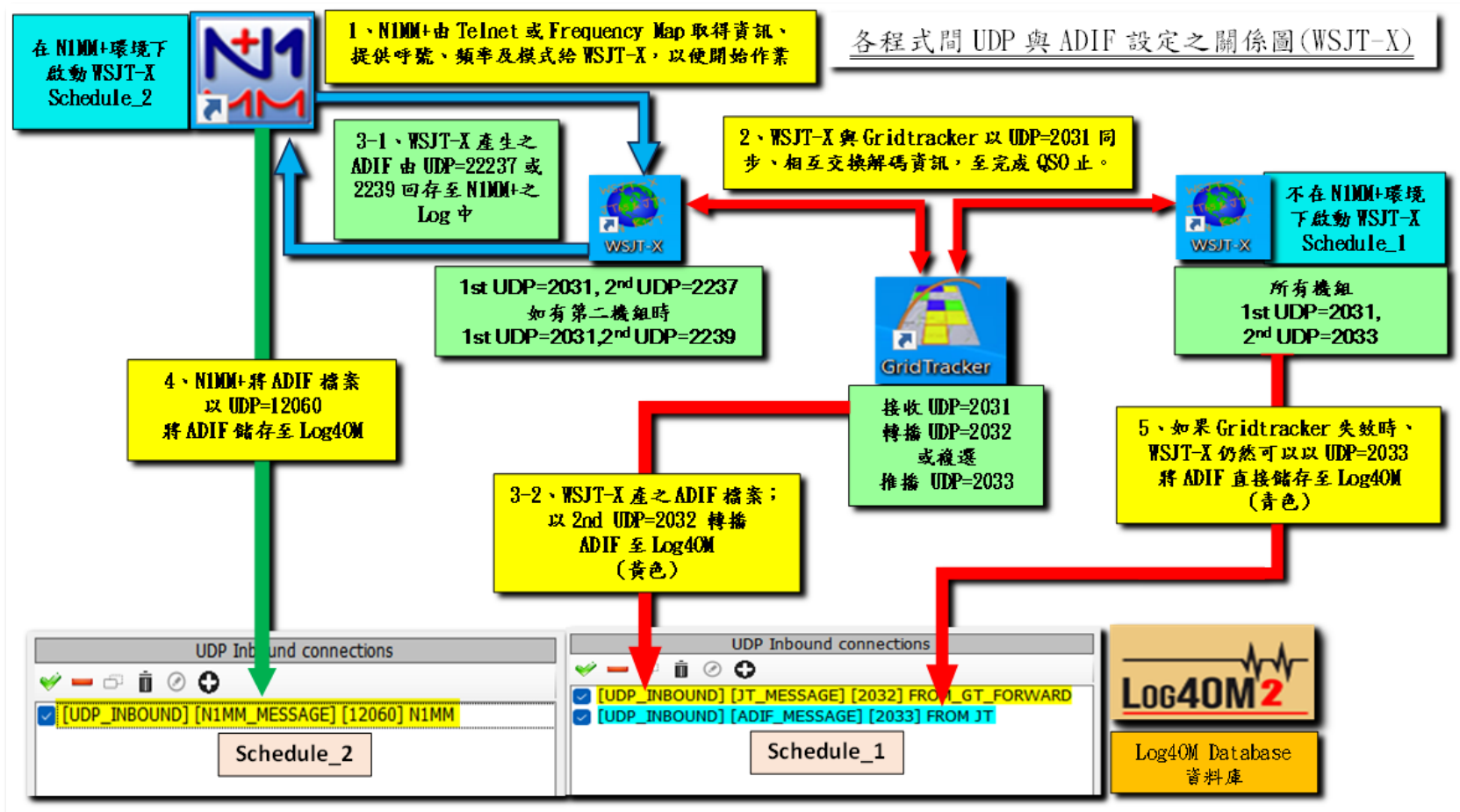
本節以上的八項文字設定(A 項至 H 項)，加上 11 個圖示(著黃色者)，請你一一的按步就班地去設定，應該是很容易執行的。

設定完成後，請依照圖\_107(針對 WSJT-X 使用者)或圖\_108(針對 JTDX 使用者)一一地將 UDP 之數值與圖中「淺綠色」的說明方塊做比對，看看是否完全符合。

如果無誤時，請繼續以下節之批次檔方法執行。



茲將四個程式之相關的 UDP 及 ADIF 關係繪圖於下，你只要針對四個程式分別做設定 WSJT-X 時、依照圖\_107 之 UDP 數字填寫即可。如設定 JTDX 時、依照圖\_108 的 TCP 及 UDP 數字填寫即可。



圖\_107 四程式相關之 UDP 與 ADIF 之相對關係圖\_WSJT-X

圖\_107 之左側是不能採用 OmniRig 的(即 N1MM+環境下的 Schedule\_2)，而右側就是常規的設定(採用 OmniRig 的 Schedule\_1)。

操作 WSJT-X 說明如下：

步驟 1、左側之 WSJT-X 之應用程式是由 N1MM+啟動；N1MM+主視窗之 CALL 框，負責讓 WSJT-X 做頻率之輸入或直接經由 Telnet(或 Band Map)視窗中按鍵取出之 Callsign 及頻率、模式等資料供給 WSJT-X 後，使得開始作業。

步驟 2、各種 Schedule 之 WSJT-X 以 UDP=2031 與 Gridtracker 做 Call Roster 的「解碼資訊共享」，直至完成 QSO。

步驟 3-1、完成 QSO 後，各種 Schedule 之 WSJT-X 自身會儲存 QSO 之 ADIF 檔案外，左側之 WSJT-X 會以 UDP=2237 或 UDP=2239 傳給 N1MM+之 Logger 作為登錄之用。

步驟 4、在 Log40M 的 Schedule\_2 的 UDP INBOUND 設定(如圖\_62)為 12060 即可接收 ADIF 檔案。

步驟 3-2、右側之 WSJT-X 同時與 Gridtracker、會透過 UDP=2031 同步接收該 QSO 之 ADIF 檔案，並經 2nd UDP=2032「轉播」出去。

(註：Gridtracker 也會透過「轉傳」之設定，將 ADIF 檔案傳給其他網站或其他之 Logger(見圖\_53)，此「轉傳」對象包括 LOW、QRZ.COM 等網站及其他的 Logger 等知名軟體、如本文採用的 Log40M 應用軟體)。

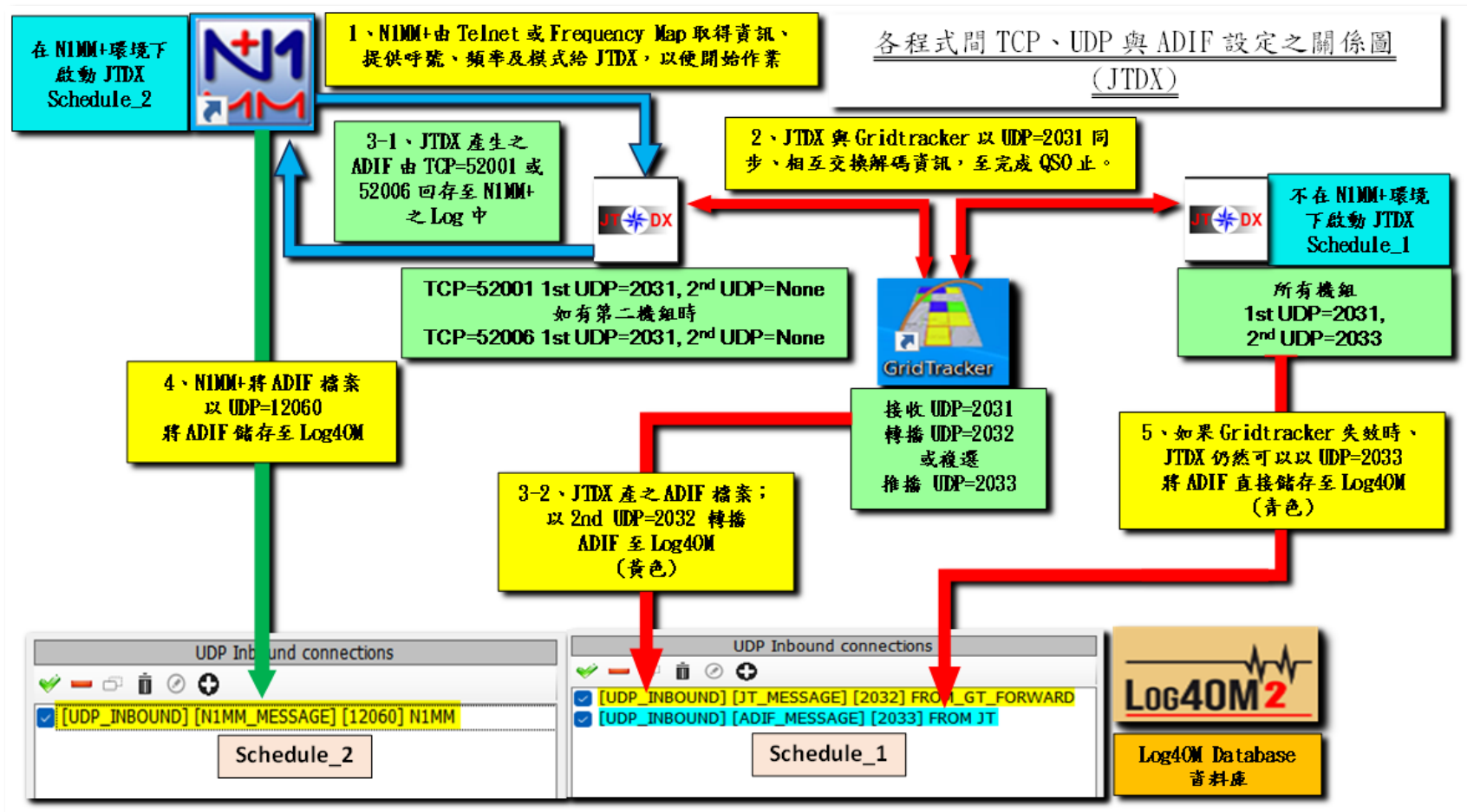
步驟 5、右側之 WSJT-X 完成 QSO 後，首先會以 Gridtracker 的 UDP=2032 將 ADIF 檔案，轉出去給 Log40M 儲存(如步驟 3-2 所述)；然而、如果 Gridtracker 因某種原因，比如：網路不穩定致使搜尋時間過長而使 UDP=2032 的包封傳送失敗，此時，QSO 完成後之 ADIF 檔案，無法傳給 Log40M。所以 UDP=2033 的功能、此時可以發揮作用、可直接送到 Log40M 之 Log 中。

你是否會有疑慮：

如果 Gridtracker 並無失效、是否會有 UDP=2032 及 UDP=2033 重複地送進 Log40M ？

目前為止、我試了好幾種狀況都沒有發現有「重複移轉」現象。

如果有這種疑慮時，你就將 UDP=2033(見圖\_53)，將 Gridtracker 之 Logging 選項卡中的 Log40M 的「Log」的打勾取消掉即可。



圖\_108 四程式相關之 UDP 與 ADIF 之相對關係圖\_JTDX

圖\_107 係為 WSJT-X 而作、圖\_108 則係為 JTDX 而作，兩者之流程均是相同的；為方便起見，茲將操作 WSJT-X 之內容複製至下面一頁，並以「JTDX」逐句替換，方便閱讀去執行。

操作 JTDX 說明如下：

步驟 1、左側之 JTDX 之應用程式是由 N1MM+啟動；N1MM+主視窗之 CALL 框，負責讓 JTDX 做波段之輸入或直接經由 Telnet(或 Band Map)視窗中按鍵取出之 Callsign 及頻率、模式等資料供給 JTDX 後，使得開始作業。

步驟 2、各種 Schedule 之 JTDX 以 UDP=2031 與 Gridtracker 做 Call Roster 的「解碼資訊共享」，直至完成 QSO。

步驟 3-1、完成 QSO 後，各種 Schedule 之 JTDX 自身會儲存 QSO 之 ADIF 檔案外，左側之 JTDX 會以 TCP=52001 或 TCP=52006 傳給 N1MM+之 Log 作為登錄之用。

步驟 4、在 Log4OM 的 Schedule\_2 的 UDP INBOUND 設定(如圖\_62)為 12060 即可接收 ADIF 檔案。

步驟 3-2、右側之 JTDX 同時與 Gridtracker、會透過 UDP=2031 同步接收該 QSO 之 ADIF 檔案，並經 2nd UDP=2032「轉播」出去。

(註：Gridtracker 也會透過「轉傳」之設定，將 ADIF 檔案傳給其他網站或其他之 Logger(見圖\_53)，此「轉傳」對象包括 LOW、QRZ.COM 等網站及其他的 Logger 等知名軟體、如本文採用的 Log4OM 應用軟體)。

步驟 5、右側之 JTDX 完成 QSO 後，首先會以 Gridtracker 的 UDP=2032 將 ADIF 檔案，轉出去給 Log4OM 儲存(如步驟 3-2 所述)；然而、如果 Gridtracker 因某種原因，比如：網路不穩定致使搜尋時間過長而使 UDP=2032 的包封傳送失敗，此時，QSO 完成後之 ADIF 檔案，無法傳給 Log4OM。所以 UDP=2033 的功能、此時可以發揮作用、可直接送到 Log4OM 之 Log 中。

至此、本文所述之各程式整合設定已經完成。

## D\_2、多重配置 Multiple Configuration 之考量

### D\_2\_1、利用批次檔來做初級的運用

所謂的「Configuration 配置」，在許多的軟體中，可以有因人、事、地、物之不同需求、而去設定其所需要的「配置」；這些設定，就如同本文前述有關 N1MM+ APP、**JT** APP(或 MSHV APP)、Log4OM APP 以及 Gridtracker APP 等四種程式的設定參數，會有因採用 N1MM+與否時的不同考量，就會有 Schedule\_1 或 Schedule\_2 之區別。

例如:N1MM+ APP 的單機版是一種「配置」；雙機版的設定內容當然會跟單機版者、有所不同，也是另一種形式的「配置」。就以 **JT**(包含 WSJT-X、JTDX 及 MSHV) APP 的單機版設定參數，也會以使用不同的機組時，也就會有不同的「配置」設定。

這些林林總總的「配置」，就好比餐廳裡的各式「餐單」、供客人挑選一樣。各餐廳雖然各有其特色，都具有包容不同的「選項」能力，提供服務，但是廚房的作業能力，還是做為一個「餐廳」不可或缺的主角。

各種 APP 軟體的設計者，也是同樣的會提供不同的「選項」、會給予客戶(或稱使用者)有選擇使用哪些功能的權力。各軟體的「廚房」，也是不例外，總是要推出一個「主引擎」出來。這個「廚房」或「主引擎」總是有其各自一套功夫；例如：「客人甲」的菜單可以很簡單的修改幾項食材後、就可以換個「名稱」賣給「客人乙」。

軟體的「主引擎」配置一些「附件」功能，可以立即成為「共用」的訂單；如果增、減幾個「附件」項目，就成為另一個專門款式的「訂單」，提供給「尊貴的客人」使用。

所以，上述的各種 APP 的「配置」也就將視之為不同的「菜單」或「訂單」時、也是蠻符合情境的。

各軟體的「訂單」名稱，大都會將「.ini」作為「訂單」的檔案型態(File Extension)。(Gridtracker 者除外)。例如：

「阿明的 101 忠狗.ini」是賣給「阿明」有關的「小狗交易」的「訂單」。

「N1MM Logger.ini」是 N1MM+ APP 的一個「配置檔案名稱」。

「WSJT-X.ini」是 WSJT-X APP 的一個「配置檔案名稱」。

「JTDX.ini」是 JTDX APP 的一個「配置檔案名稱」

Log4OM APP 更是單純，它就用「Config.ini」命名。

以此類推：

「WSJT-X\_IC7300.ini」是 WSJT-X APP 環境下的有關使用到「IC7300」的一個配置檔案名稱。

「JTDX-ForEW1.ini」是一個在 N1MM+環境下啟動的 JTDX APP 的一個配置檔案名稱」。

這些「配置檔案名稱」是如何產生的？

就以餐廳例子而言：

A、一是餐廳的服務生當場記下客人之點菜動作後、轉到廚房去「料理」。

B、另一個是透過電話或網站登錄(菜單可做挑選)後、轉到廚房去「料理」。

以應用程式而言：

C、一是使用者親自按照應用程式的指引，去做設定，成為一個「主配置檔案」。

D、另一種方式：將上述的「主配置檔案」(或稱為訂單)，請應用程式幫我們去修改成使用者專屬的「配置檔案」。

餐廳的 A 或 B 的「下單」方式，你我都懂，很容易了解。

至於 C 者，看到本文的設定過程，你也應該懂的。因為當你在該 APP 完成任何的設定後，任何應用程式都會自動的給你一個「配置檔檔名」；

在 N1MM+ APP 中，他就叫「N1MMLogger.ini」

在 Log40M APP 中，它就叫做「Config.ini」

在 WSJT-X APP 中，它叫做「WSJT-X.ini」

在 JTDX APP 中，它叫做「JTDX.ini」

至於 D 者，字面看起來，還是很模糊，不知如何去請程式幫忙？其實一點也不難，只要仿照套餐 B 的方式處理即可。

方法是這樣的：拿到 C 項的「主配置檔案」後，將之改個名字(隨便你喜歡或有意義的名稱即可)；然後丟給「主引擎」去處理。



首先，我們看一下這裡的一節文章(安裝 WSJT 的基本認識)至這一節(多重實例(Multi-Instances)啟動方式)及圖\_9的位置止。

這三節的敘述是告訴我們可以用電腦的 CMD 命令提示視窗的功能來完成 IC7300 機組的一個叫「WSJT-X-IC7300.ini」的配置檔案稱(這個.ini 檔案是位於本文圖\_35中的WSJT-X-IC7300的資料夾中)。

這是以：

```
「C:\WSJT\wsjtx\bin\wsjtx.exe --rig-name=IC7300」
```

這個短短 DOS 命令列去完成的。

當然，那個「WSJT-X-IC7300.ini」的「配置檔」的產生，是在上述命令列開始執行時，就已產生的一個標準的內建的「.ini」檔案。同時、WSJT-X 的主程式也就開始運行。運行時，你可以去做相關的參數設定(如本文之圖\_17至圖\_31)或直接關閉該 WSJT-X 之主程式。如果你選擇相關的參數設定步驟後，此時所有的參數設定內容就會記載到 WSJT-X-IC7300.ini 中。如果你先選擇直接關閉 WSJT-X 之主程式時，該「WSJT-X-IC7300.ini」的內容會是一片白紙或是內建的。

請注意命令列的前半部分(C:\WSJT\wsjtx\bin\wsjtx.exe)，這個黃色的應用程式，就是前述說的「主引擎」。而命令列的後段(--rig-name=IC7300)是所謂的「命令參數」，也就是前述的「主配置檔案」。

因此，在 Windows 系統中啟動一個 APP 的方式，一般都是以下面的格式處理：

### 路徑\主引擎(空一格)命令參數

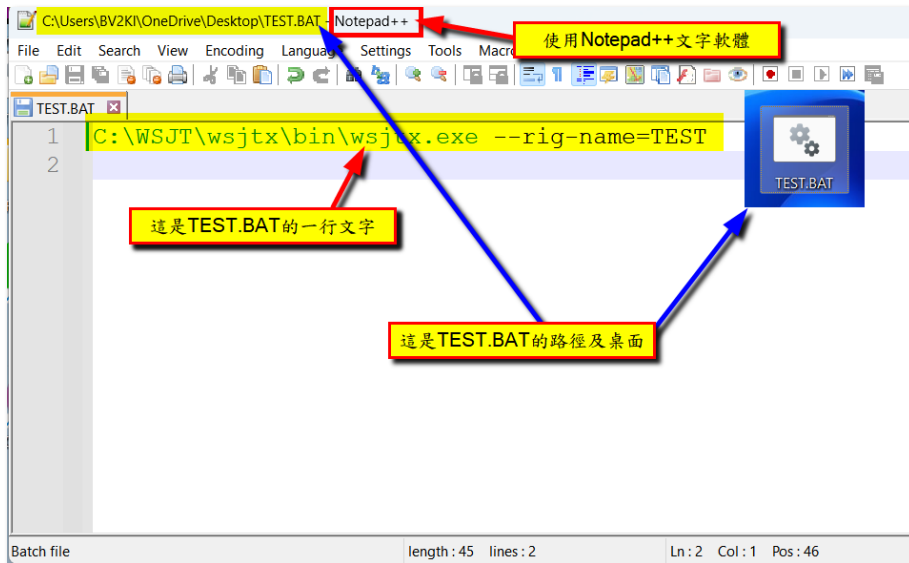
例如，我們將「C:\WSJT\wsjtx\bin\wsjtx.exe --rig-name=IC7300」

這行命令句子寫成一個文字型態的「批次檔」，並將之命名為：

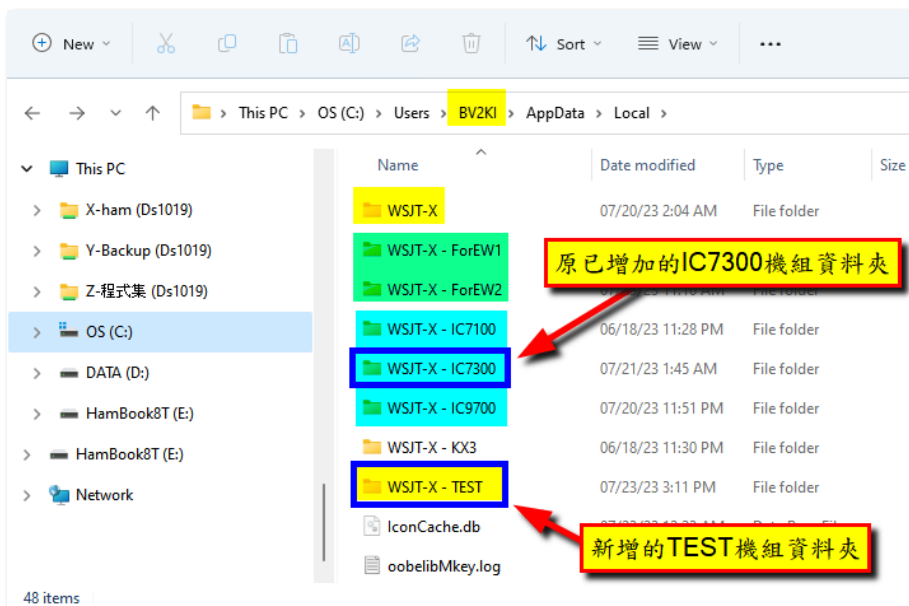
「IC7300.BAT」；把這個批次檔放在桌面上。然後左鍵雙擊此「批次檔」，接著你就會啟動 WSJT-X 的 APP。注意：這個「.ini」檔案是儲存在圖\_35中的WSJT-X-IC7300的資料夾中的。在此資料夾中會有一個「WSJT-X-IC7300.ini」的配置檔案產生。

如果你將前述的「IC7300.BAT」的檔名改為「Test.BAT」，並將其內容中，命令參數「--rig-name=IC7300」、改成「--rig-name=TEST」(如圖\_109)後、儲存到桌面，左鍵雙擊此「Test.BAT」；你就啟動了另外一組「全新的」WSJT-X 的 APP。在圖\_35，你將發現有一個叫「WSJT-X-TEST」的資料夾會出現在圖\_35 的底端處；請參看(如

圖\_110)所示。在「WSJT-X-TEST」的資料夾內、會有一個內建的「WSJT-X-TEST.ini」檔案產生。



圖\_109 TEST\_BAT 的內容



圖\_110 新增的 TEST 資料夾

至此，由於這兩個新增出來之「配置檔案」的格式，我們可以看出來、WSJT-X APP 的「基本配置檔案」的名稱是：「WSJT-X.ini」，只要將命令列參數的「—Rig

name=」後面加上一個收發機名稱如「IC7300」或「TEST」則這類的配置檔案名稱，就成為前述所見到的「WSJT-X-IC7300.ini」或「WSJT-X-TEST.ini」。

當然，這個「TEST」不是一個正常的發射機名稱；你可以在圖\_109裡的「TEST」改為「FT1000」後，在桌面上儲存成「FT1000.BAT」；雙擊此一「FT1000.BAT」後、去開啟另一個新的WSJT-X之主程式。哪麼、你將在相關「WSJT-X-FT100」的資料夾裡，會有一個「WSJT-X-FT1000.ini」的「配置檔案」出現。

這套轉換個「命令參數」的一種方法，也就是前述「配置檔案名稱產生來源」所說的「D」項方法（請應用程式幫我們去修改成使用者專屬的「配置檔案」）。換句話說：我們也是將「甲客人」的「菜單」，稍作修改後，可以做為「乙客人」的「專屬菜單」。然後去找「主引擎」出面幫忙開啟其後面的「配置檔案」。

開啟之後、在 JT 的主程式中的所有設定(如本文之圖\_17 至圖\_31)，就是所謂的「專屬菜單」。所以，你可以有 IC7300、IC9700、IC7100 或 FT1000 的「專用配置」。當你執行過各「專屬配置」的 QSO 的 ADIF 檔案，就會分別儲存至該專屬配置檔的同一個資料夾裡。因此採用不同收發機運行 JT 時，所產生的 ADIF 的檔案，就不會混淆。

這個方法，也可以引用至 JTDX APP。方法是：

將本節的「WSJT-X」的字眼，換成「JTDX」後，再加上找出 JTDX APP 之路徑（「C:\JTDX64\159\bin」），放到每一個「.BAT」之批次檔即可。

這裡所講的「批次檔」，其內容雖然只有一行即可達到「如何去開啟 Windows 的某一個 APP」。其實利用「批次檔」的特性，可以讓你去做成不同的需求。

接著，我們先看看有關 N1MM+ APP 跟這一節所講的「多重配置」檔案有何不同？請繼續看下去。。。。。

## D\_2\_2、N1MM+的多重配置 Multiple Configuration

上一節所講的 WSJT-X 或 JTDX 批次檔，作為「廚房」及「菜單」的比喻，在 N1MM+ 手冊中有一篇專章討論此一有關之話題(請參見「[Multiple INI Files](#)」一文)。

這篇文章對照於上一節所述的「命令列句子」時，以

N1MM+的「[N1MM Logger.net.exe](#)」相當於上一節所述的「主引擎」。

N1MM+的 [Ini=Radio1.ini](#) 或 [Ini=Radio2.ini](#) 視為上一節所述的「命令參數」。

如此、你就可以很容易的找到這篇文章與上一節的共通點。

這篇文章，係將 [N1MM Logger.net.exe](#)(主引擎)建立成兩個桌面上的「捷徑圖示」，分別取名為「[Radio1](#)」及「[Radio2](#)」。然後在該兩個「捷徑圖示」中、比照上一節

「路徑\主引擎(空格)命令參數」之格式去修改成「圖示命令」；

請注意:Windows 所謂的「捷徑 ShotCut」，放在桌面上，就是可直接以左鍵雙擊時予以執行的。上一節我們所做的「TEST.BAT」及「FT1000.BAT」等等之「批次檔」也是具有同等功能。

無論是以「批次檔」或「捷徑圖示」方式，去開啟各自的 WSJT-X APP 或是 N1MM+ APP，進而運行各種不同的「配置」設定(如本文之圖\_17 至圖\_31)後、關閉各該 APP 時，當下之 APP 的設定參數，都會如實的紀錄至各自的「.ini」檔案中。

在 WSJT-X 或 JTDX 的「WSJT-X-???????.ini」檔案是儲存至以收發機名稱???????下的資料夾裡(如上一節的「WSJT-X-IC7300」資料夾等)。

在 N1MM+的「Radio1.ini」或「Radio2.ini」檔案，則「都是」儲存在同一個資料夾中。這個資料夾的位置、視你安裝 N1MM+主程式時的指定而異(請參見導讀者註)，為方便說明起見，我暫時以我目前的「[C:\N1MM+](#)」作為儲存 Radio1.ini 或 Radio2.ini 之資料夾，稱為「[INI 資料夾](#)」(事實上，除非已經按照「[Multiple INI Files](#)」一文的講解並演練過，在這個「[INI 資料夾](#)」裡，你可能還看不到 Radio1.ini 或 Radio2.ini 這兩個檔案。但是在這個「[INI 資料夾](#)」裡，一定可以找得這個名叫「N1MMLogger.ini」的檔案；因為這是 N1M+的內定的默認檔案)。

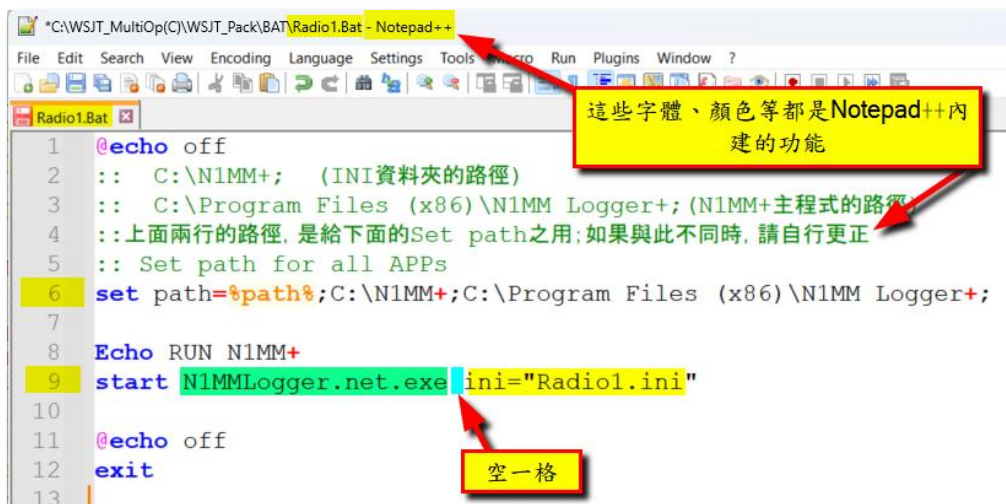
這些的「配置檔案」，會隨著你運行各該 APP 時，如果做了某些參數的設定變更，各該「.ini」的內容也跟著變動。WSJT-X 或 JTDX 的「WSJT-X-???????.ini」檔案名稱不會改變，只是存檔案時間會採用當時存檔的時間。而 N1MM+的「Radio1.INI」或「Radio2.ini」檔案名稱也不會改變，也是以新的時間註記，但是 N1MM+會將舊有者，以另外不同的改變時間做一個「.bak」的「配置檔

案」的備用名稱。如果你在 N1MM+ 的資料中看到 10 個註有「.bak」的檔案時，就表示你已經將該同一個「配置檔案」修改過 10 次了。其實這些變動資訊只用來做識別以供參考而已；為了解省檔案資源起見，可以將所有之「.bak」檔案刪除掉。

N1MM+ 的多重配置的方法與「批次檔」者，在意義上，實在沒有甚麼差異。N1MM+ 是以手工去做一個相對於「批次檔」的「捷徑」，而且每個「捷徑」只能針對一個 APP 為對象。而「批次檔」的做法可以在一個「.BAT」的文字檔中，給予多種 APP 的開啟動作 1

因此，我就將「[Multiple INI Files](#)」一文所述的：如何做出「Radiol.ini」的過程以「批次檔」方式完成。做法是這樣的：

- 1、找到 N1MM+ 主程式「N1MMRotor.Net.exe」的路徑，如：  
「C:\Program Files (x86)\N1MM Logger+\」，抄錄下來；
- 2、到「INI 資料夾」，找到「N1MMLogger.ini」檔案；右鍵將其複製，並放在同一個資料夾中，改名為「Radiol.ini」
- 3、新創一個叫「Multi\_Op」的資料夾，放在桌面上；以後所有的「.BAT」都儲存在此資料夾裡，以求方便運作；
- 4、在文字編輯軟體(Notepad)中，新創一個叫「Radiol.BAT」的文字檔；
- 5、在「Radiol.BAT」中，將這幾行字寫到裡面(第 6 及第 9 行這兩行就足夠，其餘文字皆視為解說)



```
*C:\WSJT_MultiOp(C)\WSJT_Pack\BAT\Radio1.Bat - Notepad++
File Edit Search View Encoding Language Settings Tools Window Run Plugins Window ?

Radio1.Bat
1 @echo off
2 :: C:\N1MM+; (INI資料夾的路徑)
3 :: C:\Program Files (x86)\N1MM Logger+; (N1MM+主程式的路徑)
4 :: 上面兩行的路徑, 是給下面的Set path之用; 如果與此不同時, 請自行更正
5 :: Set path for all APPs
6 set path=%path%;C:\N1MM+;C:\Program Files (x86)\N1MM Logger+;
7
8 Echo RUN N1MM+
9 start N1MMLogger.net.exe ini="Radiol.ini"
10
11 @echo off
12 exit
13
```

圖\_111 Radiol.BAT 的內容

- 6、儲存「Radiol.BAT」。

- 7、到「Multi\_Op」資料夾中，在「Radiol.BAT」上，左鍵雙擊去啟動 N1MM+。
- 8、此時，你會開啟 N1MM+的主程式；
- 9、在 N1MM+中，去做你該做的「配置設定」設定(以一個機組為主；如圖\_17至圖\_31)
- 10、關掉 N1MM+主程式。

此時，你已經自然地完成了 Radiol.ini 的「配置檔案」。

在第 9 的步驟中，假定你是以一個機組作為運行 N1MM+時，這個「Radiol.ini」的名字，尚稱適當。

如果，你是以本文開始為例，以兩個機組作為設定時(如圖\_17至圖\_31)，為了字義上免生困擾，就要將「Radiol.ini」改名了。例如我以 IC7300 及 IC9700 作為雙機的設定，那就改成「IC7397.ini」會比較適當(當然批次檔名就跟著改為「IC7397.BAT」)。這個改名的字眼就到圖\_111 中的第 9 行黃色的部分更改(不要漏掉雙括弧)。將像這樣：

```
start N1MMLogger.net.exe ini="IC7397.ini"
```

(注意：N1MMLogger.net.exe 後面一定要以 1 個「空格」插在 ini="IC7397.ini" 之間)

於圖\_111 中，第二行字是「INI 資料夾」的路徑，你一定要確定此一路徑的正確性；第三行是 N1MM+的主程式路徑，也要確認。如果與此範例不同時，請在 Set Path 中去更正為正確的路徑。「Set Pth=%path%」以後的每個路徑名稱係以「；」(分號)符號將所有路徑串接起來。

好了！以上所述的方式，就是以簡短的「.BAT」批次檔、去完成開啟 N1MM+ APP 的動作；事實上，如果你將改名時的再次作業(IC7397.BAT)，並按照如圖\_17至圖\_31之設定方式去做雙機組時，你已經有兩個 N1MM+的「配置檔案」(一個是單機組的 Radiol.ini 加上另外一個雙機組的 IC7397.ini)。

在 N1MM+ APP 環境下，可以開啟 WSJT-X APP 及 JTDX APP 這兩個應用程式(請參見圖\_7)。如果我是以雙機組於 WSJT-X APP 運行時，如何寫「N1MM\_WSJT.BAT」？如果換成雙機組於 JTDX 運行時，又是如何做出「N1MM\_JTDX.BAT」？

事實上，前述的「IC7397.ini」的產生，如果是照著圖\_17至圖\_31去設定的話，都是以 WSJT-X 為運行對象，記得嗎？圖\_7最底端的 APP 是以 WSJT-X 做了勾選的，因此，IC7397.ini 其實就是「N1MM\_WSJT.BAT」的同一個批次檔。只要將 IC7397.BAT 換個名字成為「N1MM\_WSJT.BAT」外，再將「N1MM\_WSJT.BAT」的內容中的第 9 行，改為：



start N1MMLogger.net.exe ini="N1MM\_WSJT.ini" 就可以了。

至於、換成雙機組於 JTDX 運行時，又是如何做出「N1MM\_JTDX.BAT」

這樣做：

將 IC7397.BAT 換個名字成為「N1MM\_JTDX.BAT」外，再將「N1MM\_JTDX.BAT」的內容中的第 9 行，改為：

start N1MMLogger.net.exe ini="N1MM\_JTDX.ini" 就可以了。

但是，當你以左鍵雙擊「N1MM\_JTDX.BAT」後，進入 N1MM+ 之主程式，你就去圖\_7 的最底端處，將 FT8 的 APP 選項，改為 JTDX 主程式路徑(如圖\_8)

「C:\JTDX64\159\bin\jtdx.exe」。

看出來沒？以上這兩個在 N1MM+ 環境下執行 FT8 或 FT4 的「配置檔」，不是外在的「.BAT」去設定的。這兩個 APP 是要先開啟 N1MM+ 後，才在如圖\_7 或圖\_8 中經過手工設定的結果才會產生配置檔的。

所以，批次檔檔名跟配置檔檔名設定為相同時，是易於辨識及方便使用。

當你完成上述在 N1MM+ 環境下執行 WSJT-X APP 或 JTDX APP 時的設定動作後，你下一次要執行哪一個 APP 時，就可以於「Multi\_Op」資料夾(最好將此資料夾放到桌面)中，去選擇「Radio1.BAT」或「Radio2.BAT」或「IC7397.BAT」或「N1MM\_WSJT.BAT」或「N1MM\_JTDX.BAT」後、以左鍵雙擊之，就 OK 了！

這一節以批次檔方式替代「多重配置捷徑」的方法，是以 N1MM+ APP 作範例，是否在其他 APP 上、也可以適用？

可以的，只要符合「命令列指令」原則：

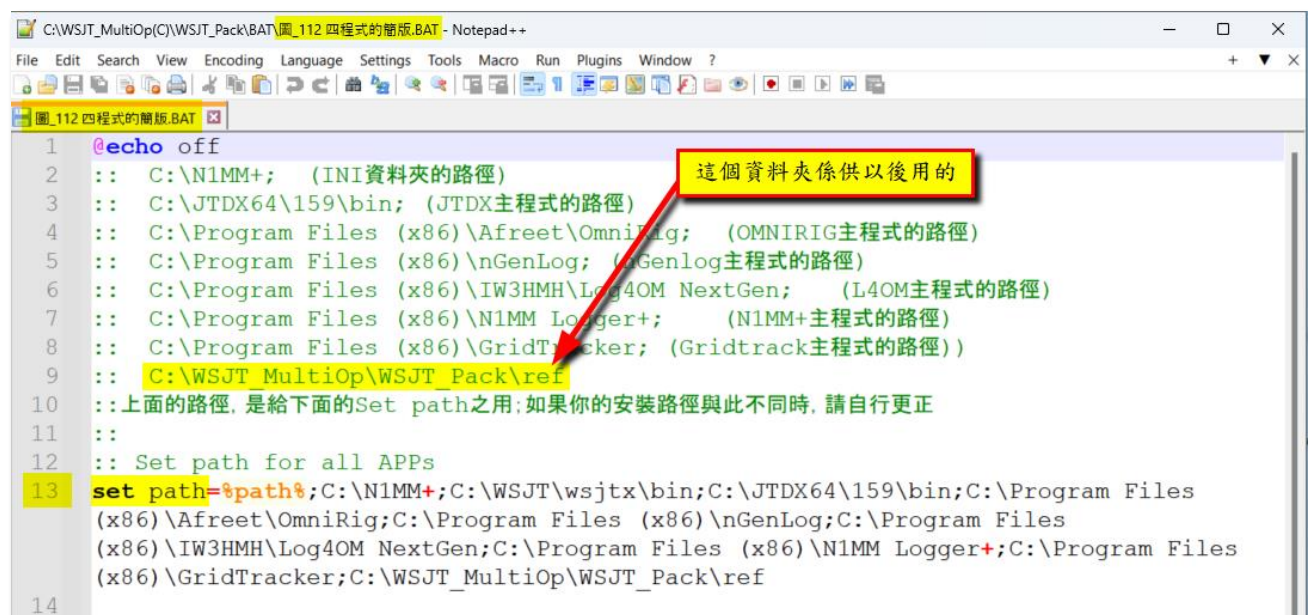
路徑\主引擎(空一格)命令參數 即可。

我們看看下一節的解說：

## D\_3、批次檔的應用

### D\_3\_1、簡化的組合版批次檔

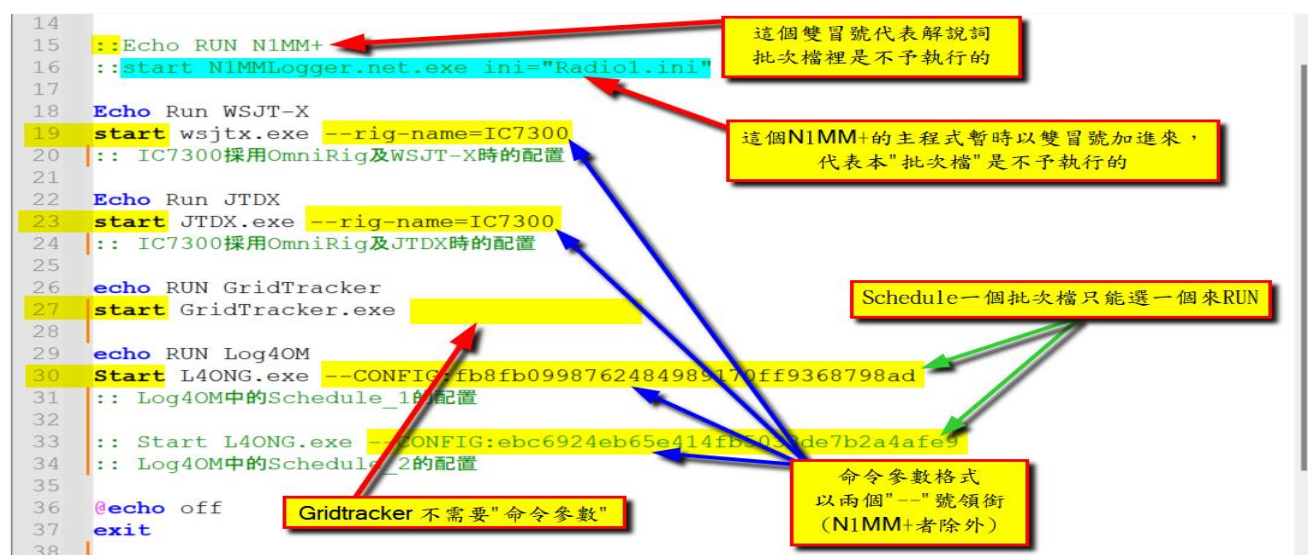
上一節係以 N1MM+ 的「多重配置」為主，本節則將此原則延伸至其他三個 APP 的應用。首先、讓我們先找出所有主程式的路徑、並予以抄錄下來，再將所有路徑串接至 Set path 中(如圖\_112 之第 13 行):



```
1 @echo off
2 :: C:\N1MM+; (INI資料夾的路徑)
3 :: C:\JTDX64\159\bin; (JTDX主程式的路徑)
4 :: C:\Program Files (x86)\Afreet\OmniRig; (OMNIRIG主程式的路徑)
5 :: C:\Program Files (x86)\nGenLog; (nGenlog主程式的路徑)
6 :: C:\Program Files (x86)\IW3HMH\Log4OM NextGen; (L4OM主程式的路徑)
7 :: C:\Program Files (x86)\N1MM Logger+; (N1MM+主程式的路徑)
8 :: C:\Program Files (x86)\GridTracker; (Gridtrack主程式的路徑)
9 :: C:\WSJT_MultiOp\WSJT_Pack\ref
10 :: 上面的路徑, 是給下面的Set path之用; 如果你的安裝路徑與此不同時, 請自行更正
11 ::
12 :: Set path for all APPs
13 set path=%path%;C:\N1MM+;C:\WSJT\wsjtx\bin;C:\JTDX64\159\bin;C:\Program Files
  (x86)\Afreet\OmniRig;C:\Program Files (x86)\nGenLog;C:\Program Files
  (x86)\IW3HMH\Log4OM NextGen;C:\Program Files (x86)\N1MM Logger+;C:\Program Files
  (x86)\GridTracker;C:\WSJT_MultiOp\WSJT_Pack\ref
14
```

圖\_112 主程式的路徑(四個程式的簡版. BAT)

我將其他的三個 APP 加進來，如圖\_113



```
14
15 ::Echo RUN N1MM+
16 ::start N1MMLogger.net.exe ini="Radio1.ini"
17
18 Echo Run WSJT-X
19 start wsjtx.exe --rig-name=IC7300
20 :: IC7300採用OmniRig及WSJT-X時的配置
21
22 Echo Run JTDX
23 start JTDX.exe --rig-name=IC7300
24 :: IC7300採用OmniRig及JTDX時的配置
25
26 echo RUN GridTracker
27 start GridTracker.exe
28
29 echo RUN Log4OM
30 start L4ONG.exe --CONFIG:fb8fb0998762484988170ff9368798ad
31 :: Log4OM中的Schedule_1的配置
32
33 :: Start L4ONG.exe --CONFIG:ebc6924eb65e414fb501de7b2a4afe9
34 :: Log4OM中的Schedule_2的配置
35
36 @echo off
37 exit
38
```

圖\_113 將其他三個 APP 加進來(四個程式的簡版. BAT)

這兩張圖合併起來是「四個程式的簡版.BAT」之批次檔文件。

如果在此批次檔上左鍵雙擊時會去執行第 19 行、第 23 行、第 27 行及第 30 行的四個程式(前提是:要先在你的電腦上、安裝好這四個程式才行)。

如果你有興趣想知道有關這個 BAT 的內容，是怎麼來的，請看看[這裡](#)。

在圖\_113 中，第 30 行的 Log40M 的程式是以：

```
--CONFIG:fb8fb0998762484989170ff9368798ad
```

這個 Schedule\_1 的配置檔，這組號碼請回頭看一下 [Schedule\\_1](#)。

在第 33 行的 Log40M 的程式是以：

```
--CONFIG:ebc6924eb65e414fb5033de7b2a4afe9
```

這個 Schedule\_2 的配置檔，這組號碼請回頭看一下 [Schedule\\_2](#)。

這兩組號碼出處，請再看看 [圖\\_84](#)，回味一下。

提醒你一下，如果你執行此一批次檔時，因為第 19 行及第 23 行中，雖然是分別去執行 WSJT-X APP 及 JTDX APP，但是這兩個的「命令參數」都採用同一個機組「IC7300」；這樣子會使得後面才啟動的 JTDX APP 因為找不到「IC7300」而發出警告。這個 BAT 的內容，是我故意寫成這樣的，製造一些小樂趣嘛！

如果你將第 23 行的「命令參數」改為「--rig name=IC9700」的話，你就可以有 IC9700 的機組處於 JTDX APP 的環境下作業的畫面出現。你就會曉得，原來我們是可以同時去以 WSJT-X APP 以及 JTDX APP 在同一部電腦上運行的([混血搭配](#))。不過、我是不建議這樣子的安排，因為這兩個的「引擎」雖同，但操作介面仍有許多不同，除非你有天賦異稟可以隨時「手、腦並用」應付；建議你還是採用同一種介面為宜。

### D\_3\_2、綜合板的批次檔

本文整合的四個應用程式的設定方法已如前面幾個章節詳述過；如何以批次檔去做開啟各相關的應用程式也在上面的一節中略為介紹。

由於這四個程式[N1MM+、Log40M、WSJT-X(或 JTDX)及 Gridtracker]選用時機是如何來做？本節就此做一分析及建議，供給你來參考。

我將針對這四個程式以操作模式以及 Logging 的方式，分別列表說明，做為選擇適當的批次檔組合前的全形印象。

	SSB	CW	RTTY/PSK	FT8/FT4 etc	Logging
N1MM+	OK	OK	OK	須與 JT+GT 配合使用	有獨立存檔功能
Log40M	OK	須配合使用 <a href="#">FLDigi</a> APP 功能		須與 JT+GT 配合使用	可儲存所有 Log
JT+GT	NG	NG	NG	OK	有獨立存檔功能

表\_1 操作模式、Log 登錄及四程式之間的功能列表

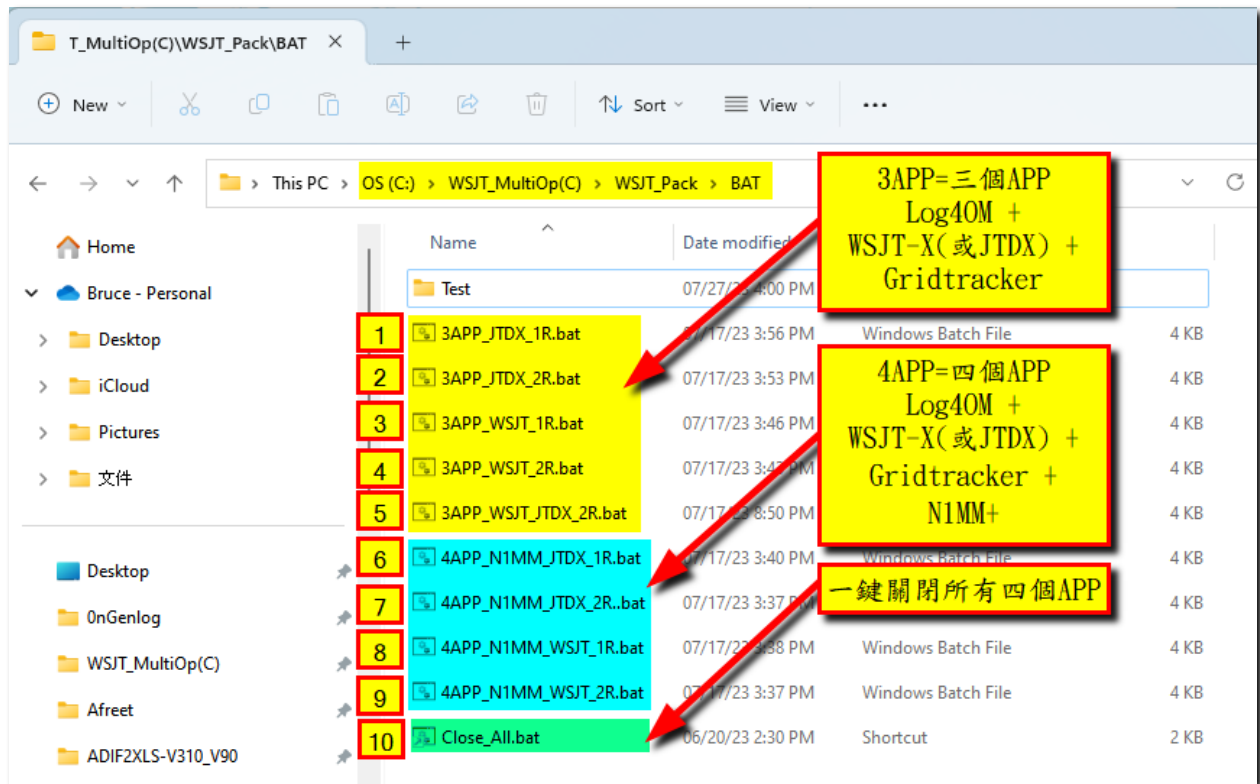
表\_1 中，**JT** 係指 WSJT-X 或 JTDX。GT 是指 Gridtracker。

[FLDigi](#)，是一個跨平臺的數據機程式，支援業餘無線電頻段上使用的大多數點對點（即時鍵盤）數位模式。

本文的前言提到如何將所有的 ADIF 以一個「總舵主」的位置來儲存，是我們最終目標；因此選擇一個免費又實用的多功能 Logging 軟體，就很重要。

在表\_1 中，我們可以依照操作模式的左邊三項，操作 SSB、CW 及 RTTY/PSK 等模式看來，N1MM+及 Log40M 都是可以作為平常時候的一個實用的軟體；喜歡比賽的同好們、還是會以 N1MM+為主。當然、你如果熟悉於 FLDigi 者，你可能會去選擇 FLDigi 的介面在 Log40M APP 操作三個模式(CW、SSB、RTTY/PSK)來替代 N1MM+ APP 者。前提是：你對於 FLDigi 介面來配合操作 **JT** 模式時，你可能要去找跟 Gridtracker 如何設定之配置的方法，才能配合本文採用之批次檔的應用。

因此、就整體之操作模式(含 FT8、FT4 等)之操作方便起見，我將前述的批次檔組合模式訂出了以下九個樣式，你也可以將些樣式視為「九個套餐」。這「九個套餐」名稱很容易懂；請參看圖\_114 的九個批次檔菜單：



圖\_114 九個批次檔菜單

以圖\_114 所示，所謂的 3APP 是指 Log4OM、WSJT-X(或 JTDX)以及 Gridtracker 三個 APP 的整合。4APP 則是再加上一個 N1MM+ APP 的意思。也可以說、4APP 者，是本文的所有敘述的設定都包含在內。

這個包含 N1MM+的「套餐群」，用於 SSB、CW、RTTY/PSK 以及 FT8(或 FT4)等模式，都可適用，尤其是你需要作為「比賽」時，就選 4APP 者沒錯。

更簡單得分野如下：

A、如果你只玩「FT8/FT4」(不做「比賽」)時，就選 3APP 者。

B、如果你只玩「SSB 或 CW 或 RTTY/PSK」時，就選 4APP 者(N1MM+包場了!)。

這就是本文的批次檔的「初步終結指引」；這兩種選項都可以達到「終極儲存」之目的。

圖\_114 中，各批次檔名稱中、有「JTDX」或「WSJT」的，表示該批次檔是採用「JTDX」或「WSJT-X」的意思。檔名中有「1R」者，是表示操作單一機組；「2R」則表示操作兩個機組。

圖\_114 中的第 10 個檔案「Close\_All.BAT」是用來做 3APP 或 4APP 的「一鍵關機」的批次檔。



可有三個以上的機組「菜單」嗎？有的，請繼續看下去，你自然學會自行修改你需求「菜單」的技巧（「三機的設定方法」）。

好了，你可依照操作模式去選擇「A」或「B」的「初步指引」後，採用「WSJT-X」或「JTDX」做「第二個選項」考量、最後再依「一機」、「二機」（甚至於「三機」）之判斷後，你就可以如圖\_114 去選擇你要的「批次檔」、以左鍵雙擊之，就可以進行 QSO 了。

再次提醒：當你採用任何之「批次檔」進行 QSO（尤其是你第一次去執行）時，你先要核對各個程式 APP 的「配置」設定，是否按照本文針對各該程式的說明去做設定（回味一下、到這裡看一看：看出來沒?）。做完一次的完整檢查後，你的這個「批次檔」裏頭的 3APP 或 4APP 的「配置」設定，就已定型；此後就可放心去做 QSO，不會再頻頻處於各種「情境」變更時，作一些經常性且無盡的摸索手工設定。

如果你想採用「N1MM+ APP」，例如：採用「4APP\_N1MM\_WSJ\_T\_2R.BAT」做批次檔時，當你雙擊此批次檔後，可能會出現「找不到 4APP\_N1MM\_WSJ\_T\_2R.ini」的警語。這是因為你電腦中的「INI 資料夾」中可能還沒有這個「4APP\_N1MM\_WSJ\_T\_2R.ini」的配置檔名。

為什麼要有這個配置檔才能啟動？

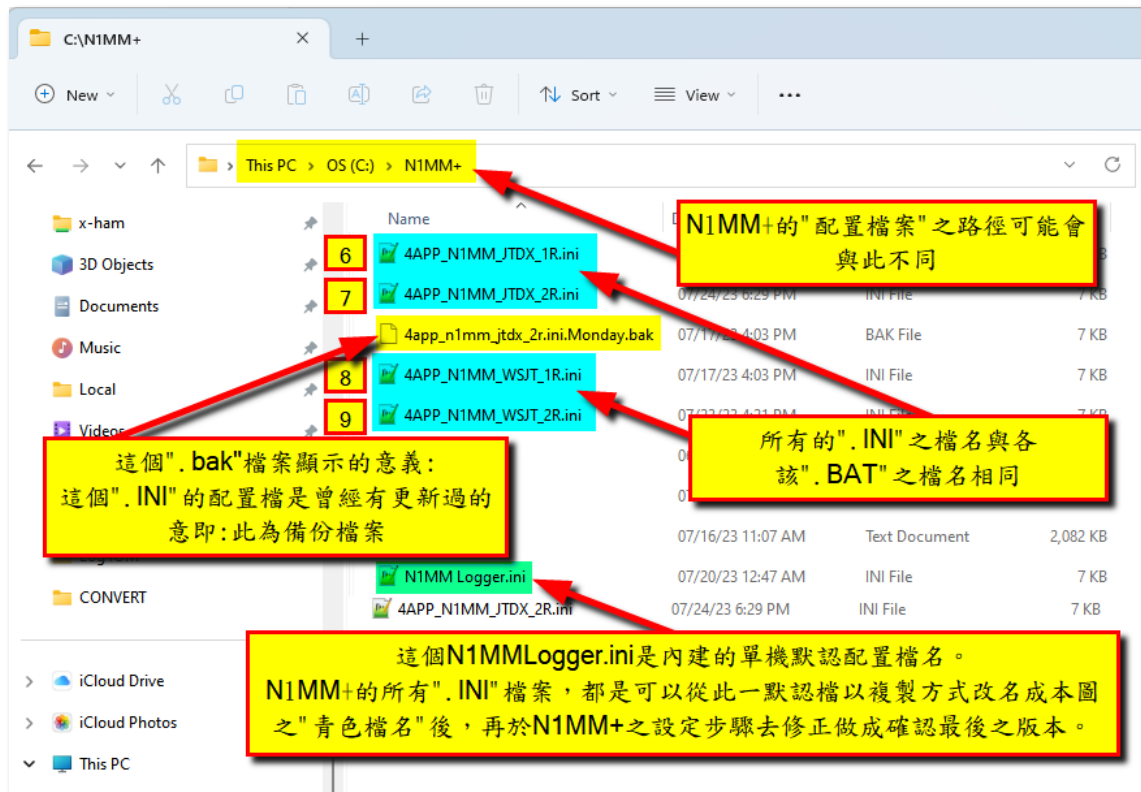
請你回顧一下「如何做出「Radiol.ini」的過程」的這段敘述。

解決方式：按照「做出「Radiol.ini」的方法」，去到「INI 資料夾」中，將「N1MMLogger.ini」複製並改名成「4APP\_N1MM\_WSJ\_T\_2R.ini」就可以了！

注意到了嗎？批次檔的檔名是跟配置檔的檔名是有一致性的；就是含有這樣的字眼「4APP\_N1MM\_WSJ\_T\_2R」。

如果你的「INI 資料夾」，如圖\_115 中，有經存在四個著「青色」的配置檔，你去執行如圖\_114 的四個著「青色」的批次檔時，就不會出現「找不到 XXXXXXXX.INI」的警語了。



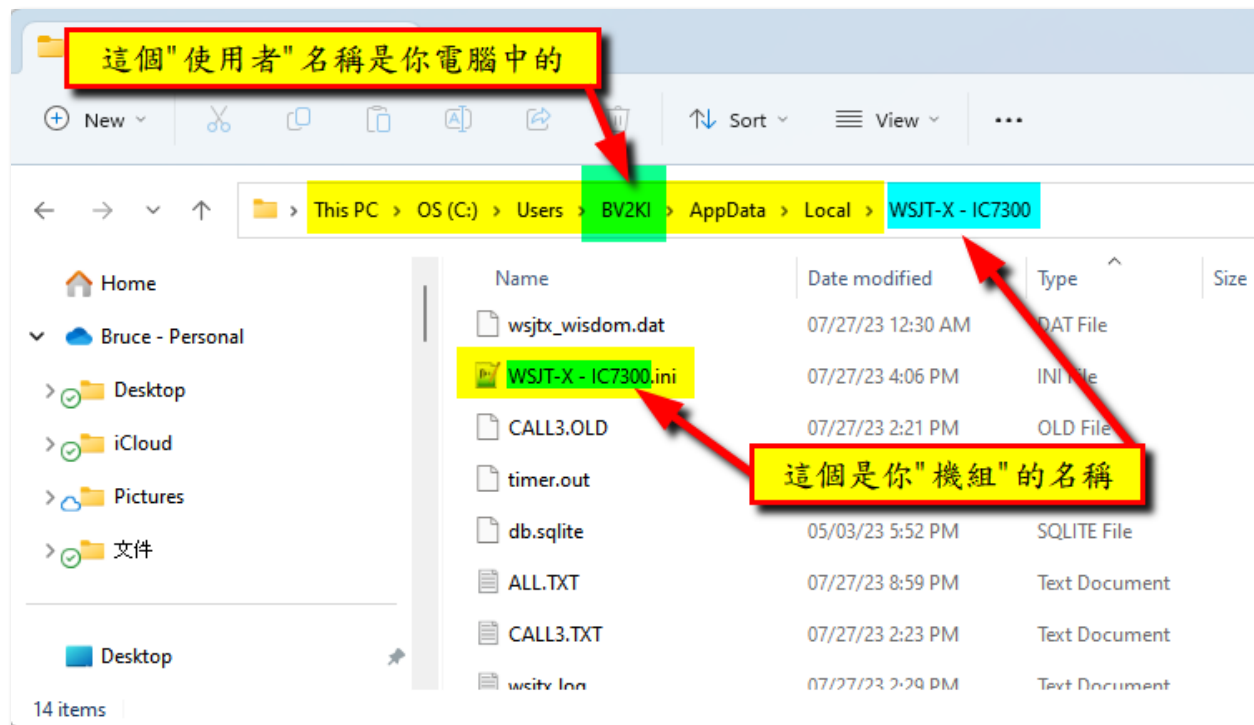


圖\_115 N1MM+配置檔案的路徑，

從圖\_115 中，你可以看到有一個「4app\_n1mm\_jtdx\_2r. ini.Monday.bak」（著黃色的.bak）檔案顯示；這是因為這個批次檔「4app\_n1mm\_jtdx\_2r.BAT」開啟 N1MM+主程式後，即使你只修改過相關的 Configurer 中的任一個小設定，程式就會將上次的「配置檔檔名」改成「4app\_n1mm\_jtdx\_2r. ini.Monday.bak」，以作為備份之參考。如果想恢復之前的設定時，先要將目前的「4app\_n1mm\_jtdx\_2r. ini」刪除，並就將該備份檔案中的「Monday.bak」這幾個字就可以了。

至於 WSJT-X(或 JTDX)的「配置檔」的路徑是放在各該「機組」所屬的路徑。我舉一個 IC7300 機組為例子，請看圖\_35 中的 WSJT-7300 資料夾去找；

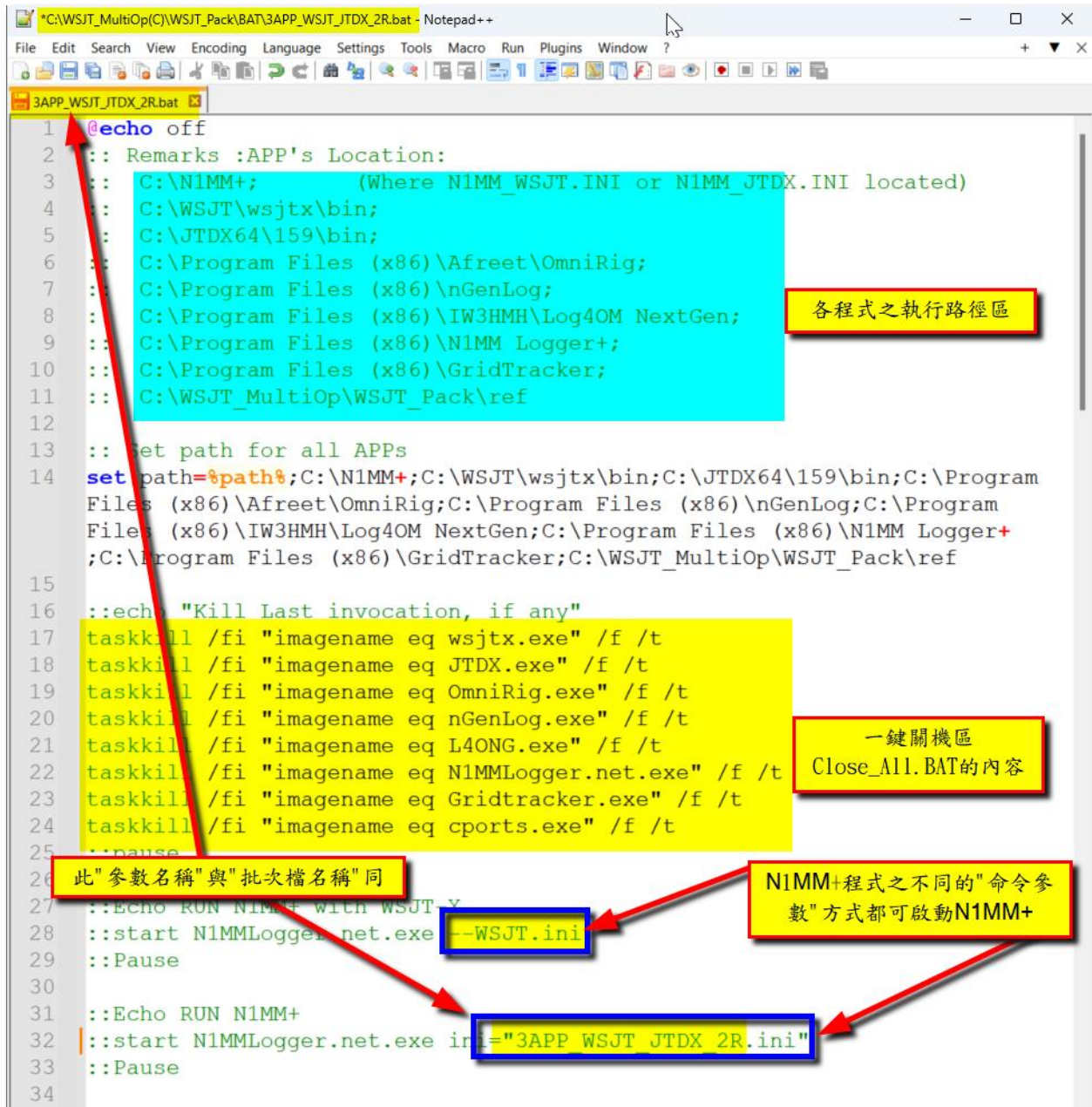
圖\_116 是 IC7300 在圖\_35 裡的「配置檔案」路徑。你也可以仿照「USER」的路徑找到一個「WSJT-X-IC7300. ini」的配置檔案。這個「USER」係指你登錄在你電腦時的名字。



圖\_116 機組的配置檔案路徑

### D\_3\_3、完整的組合版批次檔

根據以上的九個批次檔「菜單」中，我選擇一個叫「3APP\_WSJT\_JTDX\_2R.bat」者，以 Notepad++完整呈現內容，如圖\_117 至圖\_119



```
1 @echo off
2 :: Remarks :APP's Location:
3 :: C:\N1MM+; (Where N1MM_WSJT.INI or N1MM_JTDX.INI located)
4 :: C:\WSJT\wsjtx\bin;
5 :: C:\JTDX64\159\bin;
6 :: C:\Program Files (x86)\Afreet\OmniRig;
7 :: C:\Program Files (x86)\nGenLog;
8 :: C:\Program Files (x86)\IW3HMH\Log4OM NextGen;
9 :: C:\Program Files (x86)\N1MM Logger+;
10 :: C:\Program Files (x86)\GridTracker;
11 :: C:\WSJT_MultiOp\WSJT_Pack\ref
12
13 :: Set path for all APPs
14 set path=%path%;C:\N1MM+;C:\WSJT\wsjtx\bin;C:\JTDX64\159\bin;C:\Program
Files (x86)\Afreet\OmniRig;C:\Program Files (x86)\nGenLog;C:\Program
Files (x86)\IW3HMH\Log4OM NextGen;C:\Program Files (x86)\N1MM Logger+
;C:\Program Files (x86)\GridTracker;C:\WSJT_MultiOp\WSJT_Pack\ref
15
16 ::echo "Kill Last invocation, if any"
17 taskkill /fi "imagename eq wsjtx.exe" /f /t
18 taskkill /fi "imagename eq JTDX.exe" /f /t
19 taskkill /fi "imagename eq OmniRig.exe" /f /t
20 taskkill /fi "imagename eq nGenLog.exe" /f /t
21 taskkill /fi "imagename eq L4ONG.exe" /f /t
22 taskkill /fi "imagename eq N1MMLogger.net.exe" /f /t
23 taskkill /fi "imagename eq Gridtracker.exe" /f /t
24 taskkill /fi "imagename eq cports.exe" /f /t
25
26
27 ::Echo RUN N1MM+ with WSJT-Y
28 ::start N1MMLogger.net.exe --WSJT.ini
29 ::Pause
30
31 ::Echo RUN N1MM+
32 ::start N1MMLogger.net.exe inl="3APP_WSJT_JTDX_2R.ini"
33 ::Pause
34
```

各程式之執行路徑區

一鍵關機區  
Close\_All.BAT的內容

此"參數名稱"與"批次檔名稱"同

N1MM+程式之不同的"命令參數"方式都可啟動N1MM+

圖\_117 完整的批次檔\_1

各程式之執行路徑必須先行找出來，放在上圖得第一個區塊中，然後用「；」作為區隔符號形成圖中第14行「Set path」的文字串，以便於電腦的快速執行。

```
34
35 ::Echo Run JTDX
36 ::start JTDX.exe --rig-name=IC7300
37 ::Set Rig_1 for IC7300 if using OmniRig
38
39 start JTDX.exe --rig-name=IC9700
40 ::Set Rig_2 for IC9700 if using OmniRig
41
42 ::start JTDX.exe --rig-name=IC7100
43 ::Set Rig_1 for IC7100 if using OmniRig
44
45 ::start JTDX.exe --rig-name=Kx3
46 ::Set Rig_1 for KX3 if using OmniRig
47
48 ::Echo Run WSJT-X
49 start wsjtx.exe --rig-name=IC7300
50 ::Set Rig_1 for IC7300 if using OmniRig
51
52 ::start wsjtx.exe --rig-name=IC9700
53 ::Set Rig_2 for IC9700 if using OmniRig
54
55 ::start wsjtx.exe --rig-name=IC7100
56 ::Set Rig_1 for IC7100 if using OmniRig
57
58 ::start wsjtx.exe --rig-name=Kx3
59 ::Set Rig_1 for KX3 if using OmniRig
60
61 echo RUN GridTracker
62 start GridTracker.exe
63 ::Echo wait 5 seconds for Gridtracker main Windows completely Shown
64 ::pause
65
66 echo RUN Log4OM
67     :: 2 Case Configuration,
68
69     :: Case A
70     :: [2 Apps] : Lo4OM+WSJT           :: (Short as: L4OM+JT)
71     :: [3 Apps] : Lo4OM+GridTracker+WSJT :: (Short as: L4OM+GT+JT)
72 Start L4ONG.exe --CONFIG:fb8fb0998762484989170ff9368798ad
73     :: the above schedule index(fb8fb0998762484989170ff9368798ad) is
```

一部電腦上可以跑"JTDX"  
也可以同時跑"WSJT-X"

第三部機組跑"WSJT-X"

Schedule\_1的Log4OM  
非採用N1MM+之應用程式

圖\_118 完整的批次檔\_2

這張圖片裡面，你可以用雙冒號作為選擇 APP 之「執行」與否的依據；這個批次檔中，我故意選擇 Radio\_1 以 IC7300 的機組在 WSJT-X APP 去跑；採用 JTDX APP 玩玩 IC9700。由於是 3APP 的選項(不去執行 N1MM+)、Log4OM APP 則是選擇了 Schedule\_1(此項包含徵用 OmniRig、用於 Log4OM 與機組之間的資訊共享)。

```
71      :: [3 Apps] : Lo4OM+GridTracker+WSJT      :: (Short as: L4OM+GT+JT)
72  Start L4ONG.exe --CONFIG:fb8fb0998762484989170ff9368798ad
73      :: the above schedule index(fb8fb0998762484989170ff9368798ad) is
74      :: quoted From
75      C:\Users\BV2KI\AppData\Roaming\Log4OM2\user\scheduler_fb8fb099876
76      2484989170ff9368798ad_user.xml
77      :: Both of "2 Apps" and "3 Apps" are use the same schedule,
78      :: Where Setting on Lo4OMM using the "OmnRig" utility for
79      :: Case B
80  ::Start L4ONG.exe --CONFIG:ebc6924eb65e414fb5033de7b2a4afe9
81      :: the above schedule index(ebc6924eb65e414fb5033de7b2a4afe9) is
82      :: quoted From
83      C:\Users\BV2KI\AppData\Roaming\Log4OM2\user\scheduler_ebc6924eb65
84      e414fb5033de7b2a4afe9_user.xml
85      :: Where Setting on Lo4OMM using "TCI" instead of "OmnRig" for
86      Cat-Connection.
87  ::pause
88  ::echo "Open CurrPorts...."
89  ::open CurrPorts.exe to check if any last invocation programme exist.
90  ::start cports.exe
91  ::pause
92  @echo off
93  exit
94
95
```

Schedule的儲存路徑

Schedule\_1的Log4OM  
非採用N1MM+之應用程式

Schedule\_2的Log4OM  
採用N1MM+之應用程式

Schedule\_1與Schedule\_2  
只能擇一應用

圖\_119 完整的批次檔\_3

以一個「完整」的「菜單」讓你依照前一節之

「初步指引」、

「選用 APP」、

「選擇機數」

等三個步驟，來完成你喜歡的「菜單」去下單、這就是本節設定的一個「完整菜單」的構想，你可以在這個「完整的 BAT」之基礎上去做增減、修改或做不同的「組合」，以滿足你的需要。

茲將一個完整的「批次檔文件內容」「3APP\_WSJT\_JTDX\_2R.bat」內容，放在本文之參考文件。你可以複製後使用。



## D\_3\_4、改變機組數的方法

### 「三機」的設定方法：

- 1、將你的第「三」部機組「連上」電腦；例如：選「KX3」
- 2、將上述的批次檔名稱、改為易於辨識的檔名、如  
「3APP\_WSJT\_JTDX\_3R.bat」；（注意：是「3R」喔！）
- 3、將圖\_118 中的第 58 行的「雙冒號」刪除後存檔。

如此，你就有了一個專屬三機組的批次檔「菜單」。當你去執行這個批次檔後，你還是要去檢查 WSJT-X-KX3 的 Setting 配置，看看是否正確。

當然、在 Log4OM APP 中，僅能容許 2 個機組的位置與 OmniRig「共享」；這個「小三」的可供「共享」之資訊可能會被屏棄於門外、無法與 Log4OM 連線。但是這個「小三」產生的 ADIF，仍然是可以送到 Log4OM 之 Database 中的。

### 「一機」的設定方法：

請自行在「菜單」中、將不要的「機組」之「Start」前，以「雙冒號」掛上去後，將「菜單」改成一個你喜歡的檔案名稱即可。

例如，圖\_114 的第 3 個的「3APP\_WSJT\_1R.bat」批次檔及  
第 8 個的「4APP\_N1MM\_WSJT\_1R.bat」批次檔。

涉及「4APP」者，亦即：會採用到 N1MM+時：

還需要先去「INI 資料夾」、以複製後更名的方式去預設一個「INI」檔，然後到該「.INI」檔中的「INI 檔名」改成與「批次檔」檔名一致即可。



## E、好用的參考細節

### E\_1、善用 Log40M 及 N1MM+內建之 Telnet

Log40M 應用程式中，有一個 Telnet Cluster 遠端登錄叢集 的功能視窗；它是用來報導當前全球所有電台上線的活動資訊的一種以單行文字列表方式的視窗顯示。在該視窗中、你可以做各種欄位的排序(如工作頻率)以利閱讀。你可以在你有興趣的電台(你的稀有電台、遠征電台或 New DXCC 等)呼號上，以左鍵單擊方式去將該電台的細部資訊顯示於 Log40M 的主視窗中以供查閱；你也可以左鍵雙擊方式去將該電台目前的呼號及工作頻率及時地「轉移」至 **JT** 之「DX Call」中；此時、如果你已在 **JT** 中、設定好「AUTO TX」選項時，**JT** 就會自動的去向該電台做呼叫的動作(JTDX 則需視其他的 MISC 的設定而異)；非常方便。

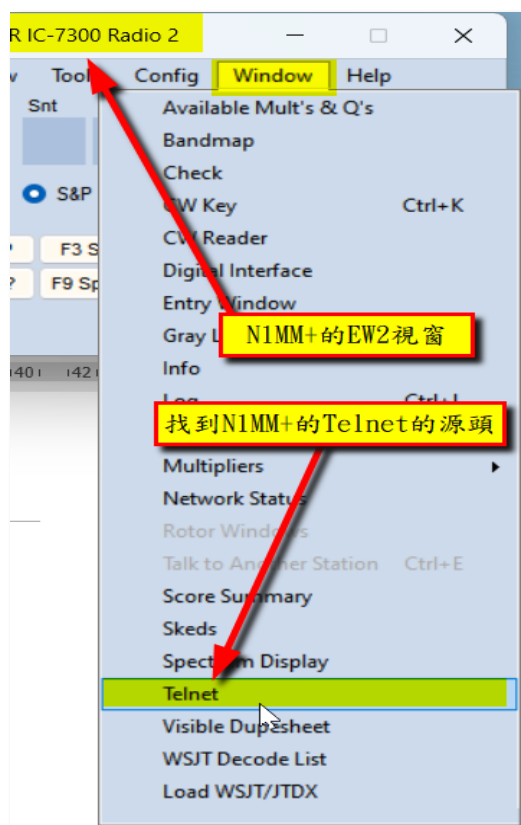


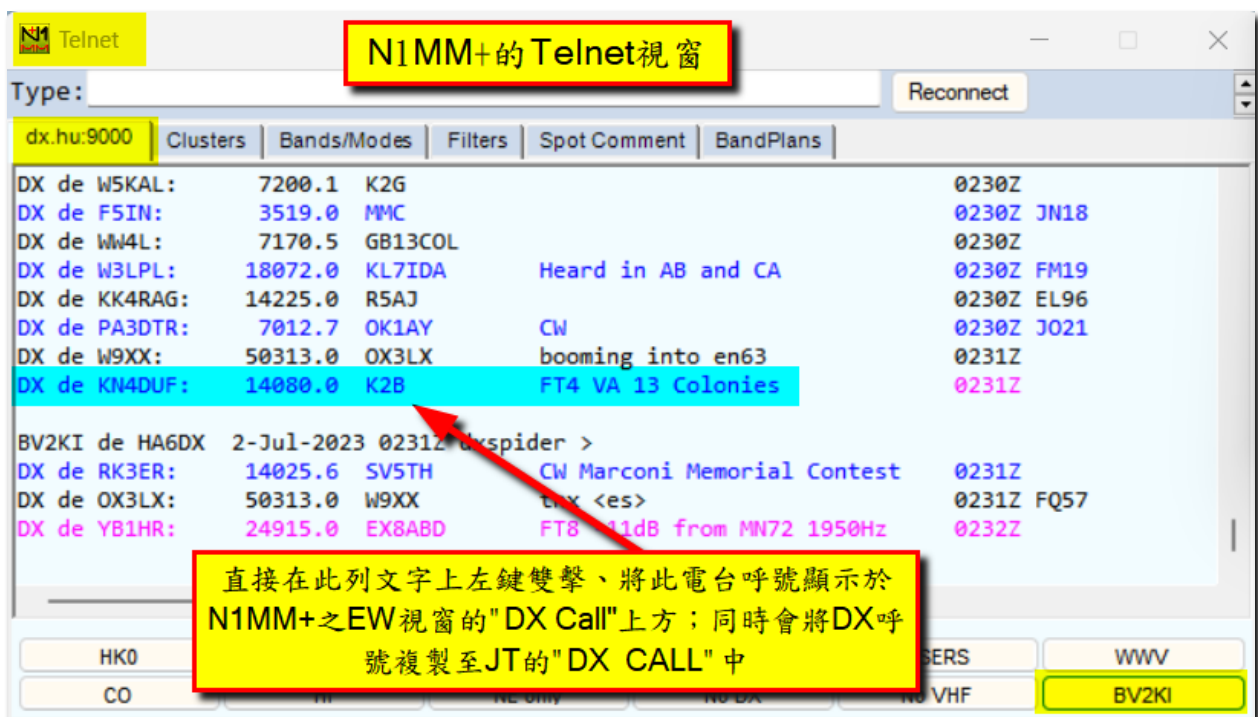
圖 120 N1MM+之 Telnet 的源頭

在 N1MM+ 應用程式裡，也具有一個 Telnet 的功能視窗(在 EW1 或 EW2 之視窗上方的「Window」選項卡中可以找的到；請參看圖\_120)。各種應用程式採用的 Telnet 功能皆各具特色、呈現方式不一；N1MM+ 的 Telnet 也不例外，它也可以左鍵雙擊於 Telnet

的任一呼號或頻率上、此一動作會將該電台呼號及時地顯示於 EW1(或 EW2)視窗的「CALL」上端處及 **JT** 的「DX Call」框框內、同時會將 Telnet 所「擷取」的頻率及模式等轉給 **JT** 應用程式使用。但是、做雙擊動作之前，請確認你當前的 EW 視窗是何者？以免「移轉」至錯誤的 EW 機組。

Log4OM 與 N1MM+的 Telnet 雖然都是具有相同的功能，我們到底要採用何者？

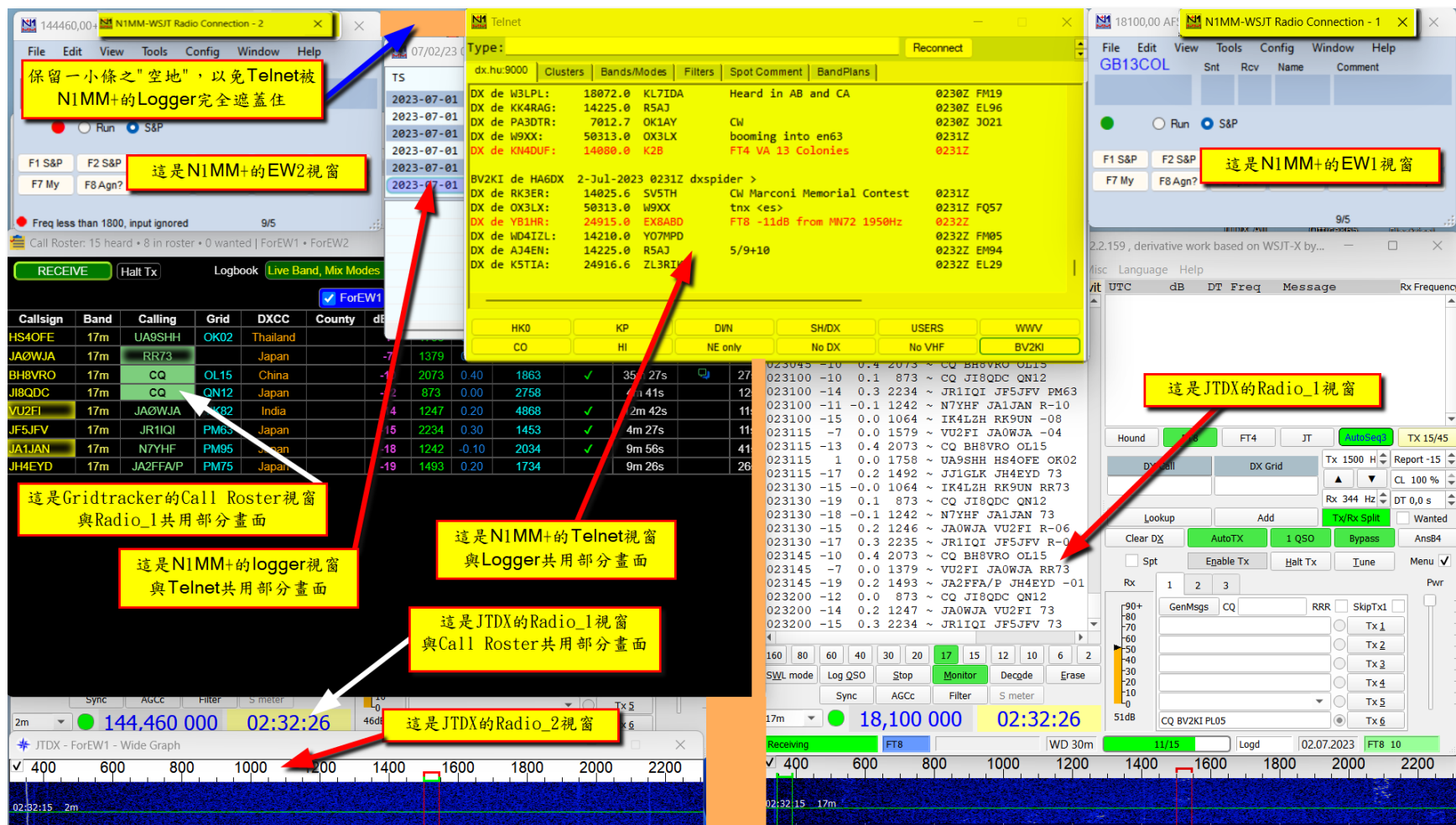
Log4OM APP 的 Telnet Cluster 中之「雙擊移轉功能」僅能適用於「常規環境下」之 **JT**、且在 N1MM+環境下的 **JT** 則無法使用。換句話說：在 N1MM+環境下的任何模式(含 FT8 等)，則須採用 N1MM+的 Telnet 功能。不能混淆。



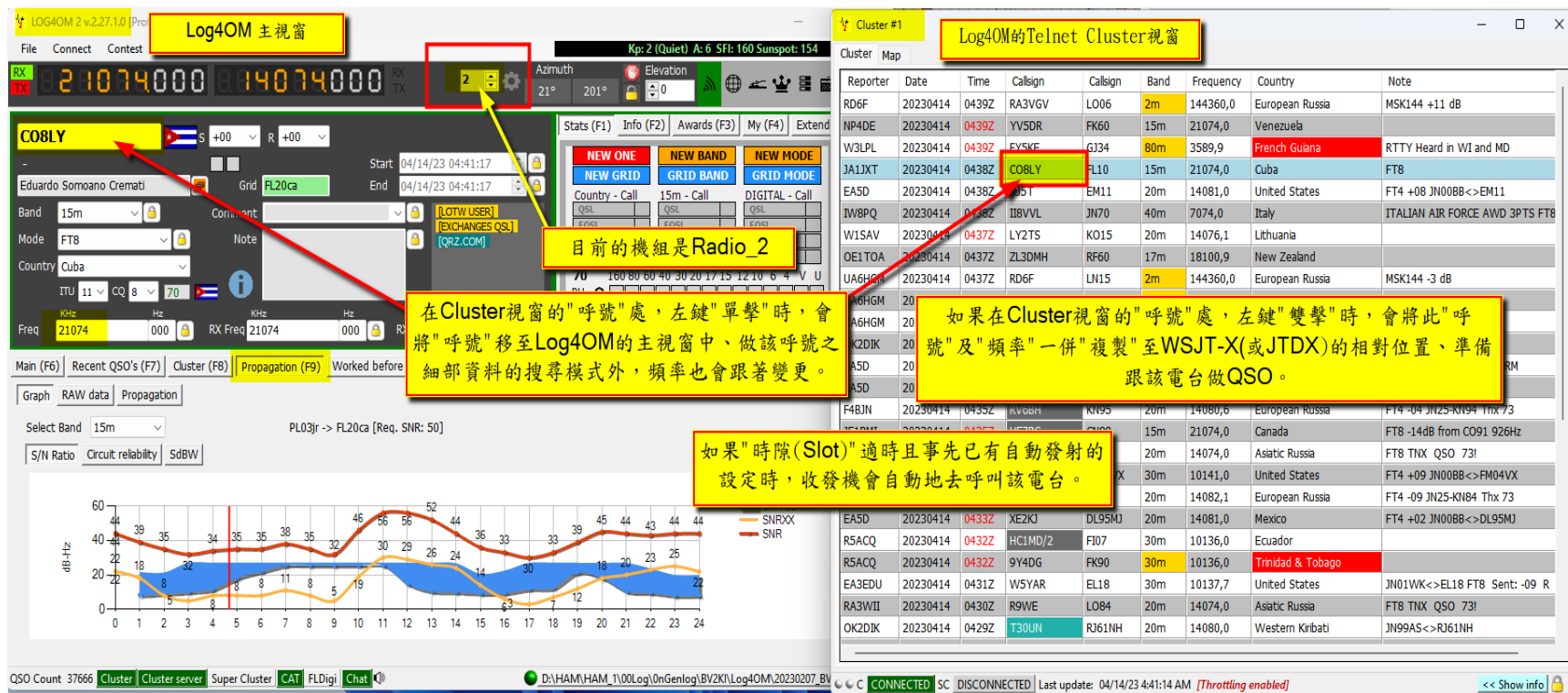
圖\_121 N1MM+的 Telnet 視窗。

上述的 Telnet 或 Telnet Cluster 都需要去做「選擇結點站 NODE」的引用設定；每一個「結點站」的運行系統不一，設定方式也就不一致。但是在 Log4OM 及 N1MM+這兩個軟體設計者的精心策畫下，已經簡化不少，兩者都很容易去做設定及操作；請自行摸索，本文就不再詳述。

這個 N1MM+的 Telnet 視窗我把它放到跟 N1MM+的 Logger 視窗做部分重疊，保留一些空白地方，可供我們隨時以滑鼠去到 Logger 或 Telnet 之視窗處，打開來察看。請參見圖\_122 的排列方式。



圖\_122 運用 N1MM+的 Telnet 時，排列的樣子



圖\_123 Log4OM 的 Cluster 運用解說

圖\_123 是 Log4OM 的 Cluster 視窗(右半部)及 Log4OM 的主視窗(左半部)畫面。Cluster 的最上一行的每個欄位上，點擊兩下，可做為資料排序的主 Key 鍵，譬如在 Frequency 頻率上做降冪式排序，可讓你容易查看哪一個可能 open 的波段可供參考。

圖\_123 中、在 Country 這欄裡，出現了「紅底白字」者，都屬於你的「稀有電台」；專注於此「稀有電台」的去向及波段是否 open、這個 Cluster 就成為「獵物」能夠手到擒來與否的一項有利工具。

有關之操作方式，請看看圖 123 所寫的步驟，很簡單、是不 ?!

請注意：如果你使用 2 組收發機時，一定要選對機組號碼，例如圖\_123 中、左上方紅色框框所示機組號碼為 Radio\_2。這個機組的號碼是要用手動方式去撥動的。如果選錯了機組，Cluster 的「移轉功能」可能無法為你服務。同時也提醒你：此時的「移轉功能」是以 OmniRig 作為 Log4OM 與 JT 之間的同步介面；如果你沒有選擇 OmniRig 做介面，本節之「移轉功能」也是無能為力的。

## 要善用你的 Telnet Cluster ！



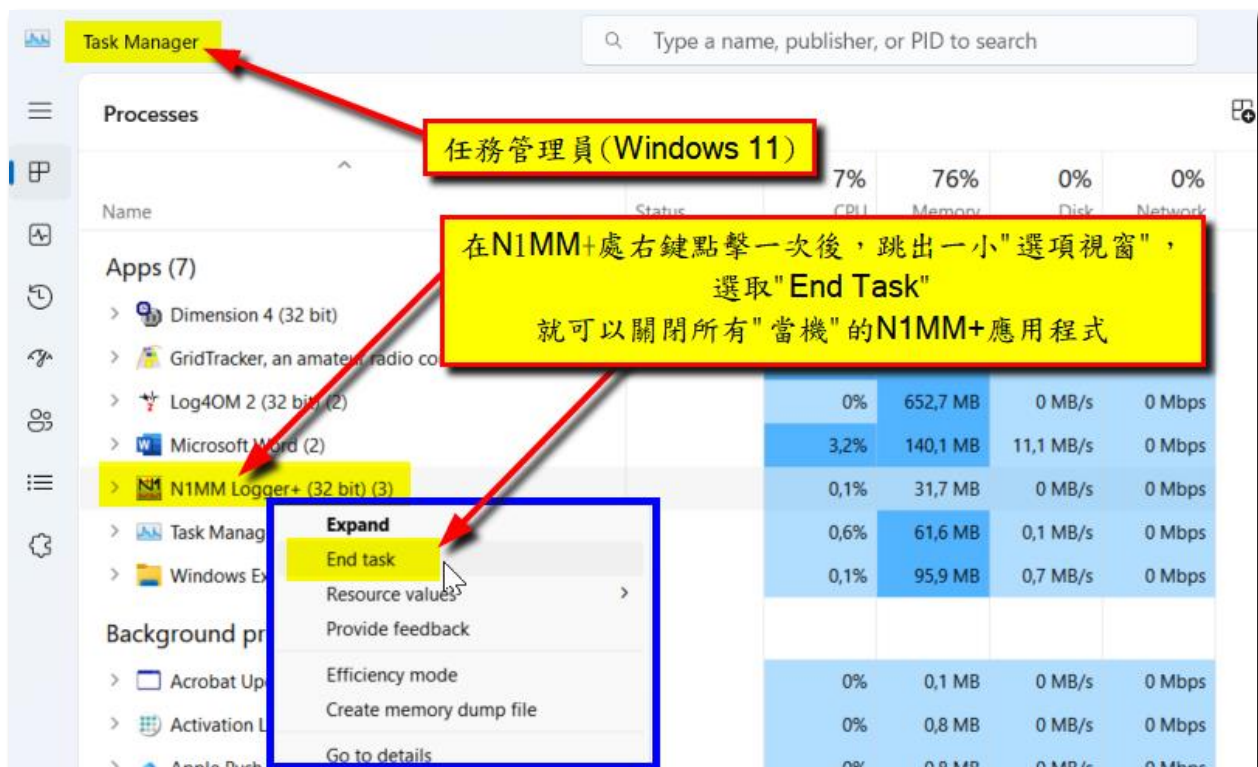
## E\_2、退出 WSJT-X(或 JTDX)時，順序錯誤關機的處理方法：

將以下的批次檔案(儲存之檔名:CloseAll.bat)放在桌面上，左鍵按兩下、去執行完整的關機動作：

```
::echo "Kill Last invocation, if any"
taskkill /fi "imagename eq wsjtx.exe" /f /t
taskkill /fi "imagename eq JTDX.exe" /f /t
taskkill /fi "imagename eq OmniRig.exe" /f /t
taskkill /fi "imagename eq L4ONG.exe" /f /t
taskkill /fi "imagename eq N1MMLogger.net.exe" /f /t
taskkill /fi "imagename eq Gridtracker.exe" /f /t
exit
```

## E\_3、強制關閉應用程式的方法：

滑鼠移至螢幕最下方、右鍵選用「Task Manager 任務管理員」如下圖\_124:



圖\_124 強制關閉應用程式



#### E 4、解決驅動程式失效的小秘訣：

ICOM 的 CODEC 之驅動程式經常會受到鄰近電台(比如說:同一城市的同好們)強大的音頻干擾，尤其是在 **JT** APP 的情況下，更為明顯。下面有一段取自於 IC7300 之操作手冊 P 4-4 的記載：

NOTE: When you are receiving weak signals, and a strong signal is momentarily received, the AGC function quickly reduces the receiver gain. When that signal disappears, the transceiver may not receive the weak signal because of the AGC action. In that case, select FAST, or touch [AGC] for 1 second to open the AGC screen, and then select OFF the time constant setting.

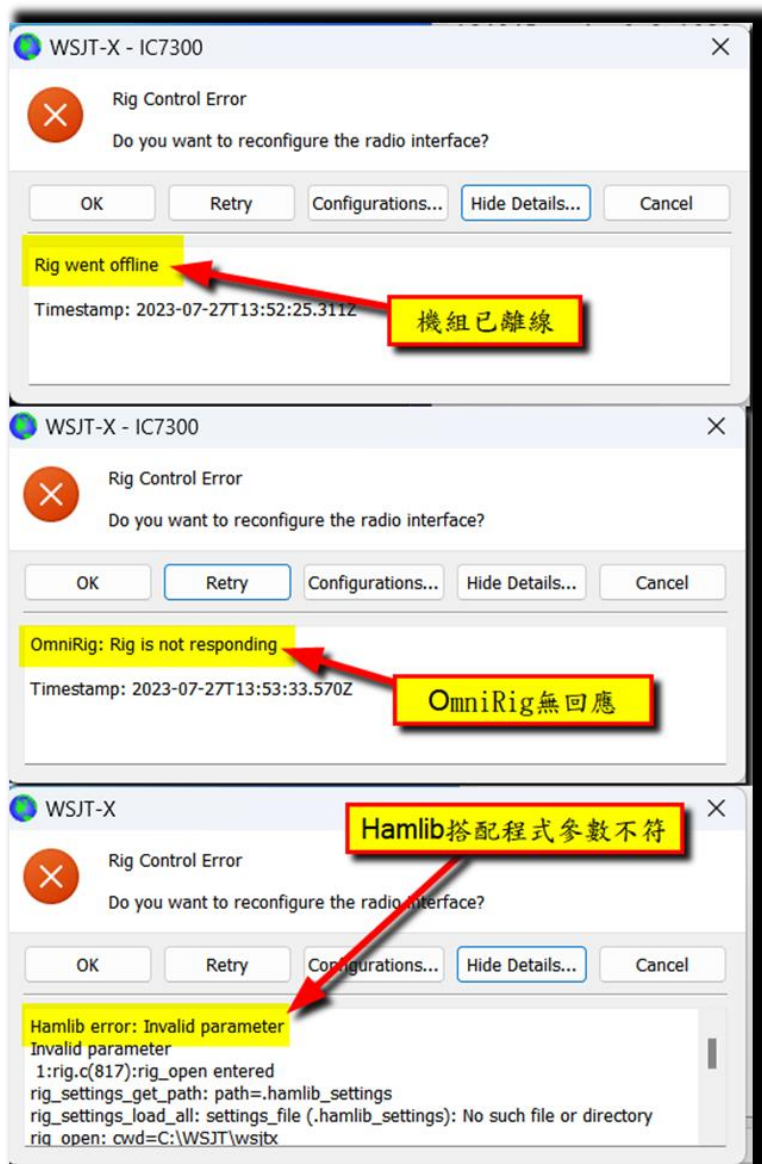
當你正操作接收一微弱信號時，突然間收到一股強烈信號產生時，收發機的 AGC 則很快的啟動並降低接收強度。當該強烈信號消失時，由於 AGC 之作動、而使得想要接收的微弱信號、就可能無發接收到。此種情況，您可選用 AGC=Fast，或者是將 AGC 關閉掉 OFF。

(筆者的補充說明:IC7300 或 IC9700 之機組面板上，找出 AGC 的畫面上，在 DEF 按鈕上、長按約 1 秒鐘後，旋轉收發機上的頻率的大旋鈕，將 SLOW 的值設無 0 即可(設為 OFF 狀態，以供後續選用)。

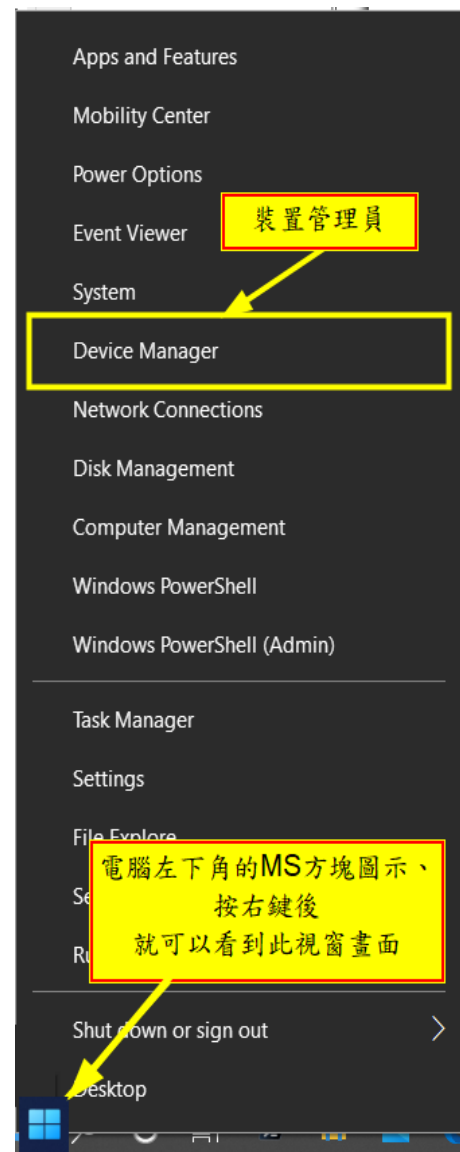
因此、在同一城市的同好們(方圓約 20 公里直徑之內者)，都有默契的選擇在同一時序(SLOT，如圖\_26 之左上方的文字說明，event/1st 做勾選與否)去做發射動作，以免「相互傷害」

**JT** APP 如果發生了 CODEC 之驅動程式失效時，會有產生一個「噹」的聲響(有些時候「停機」時間過長、也會產生驅動程式失效，你又不在此、就可能聽不到「噹」聲!)，同時會有一個停止作業的訊息視窗出現在螢幕(如圖\_125 中的一個警語)；此一視窗的出現，只得逼得你去關閉 **JT** 應用程式。在此信息視窗中的 Detail 上，點進去看一下它到底講些甚麼！你如果發現說是：「你的音效裝置失效」之類的文字或是有關 Hamlib 無法繼續提供服務時，就是犯了這種 CODEC 驅動程式已成為一個「不回家的人」了！

有關「過度驅動音頻晶元並產生失真」也會產生驅動程式的失效」。請看這篇文章「KV5R—有關數據通訊的設定」。



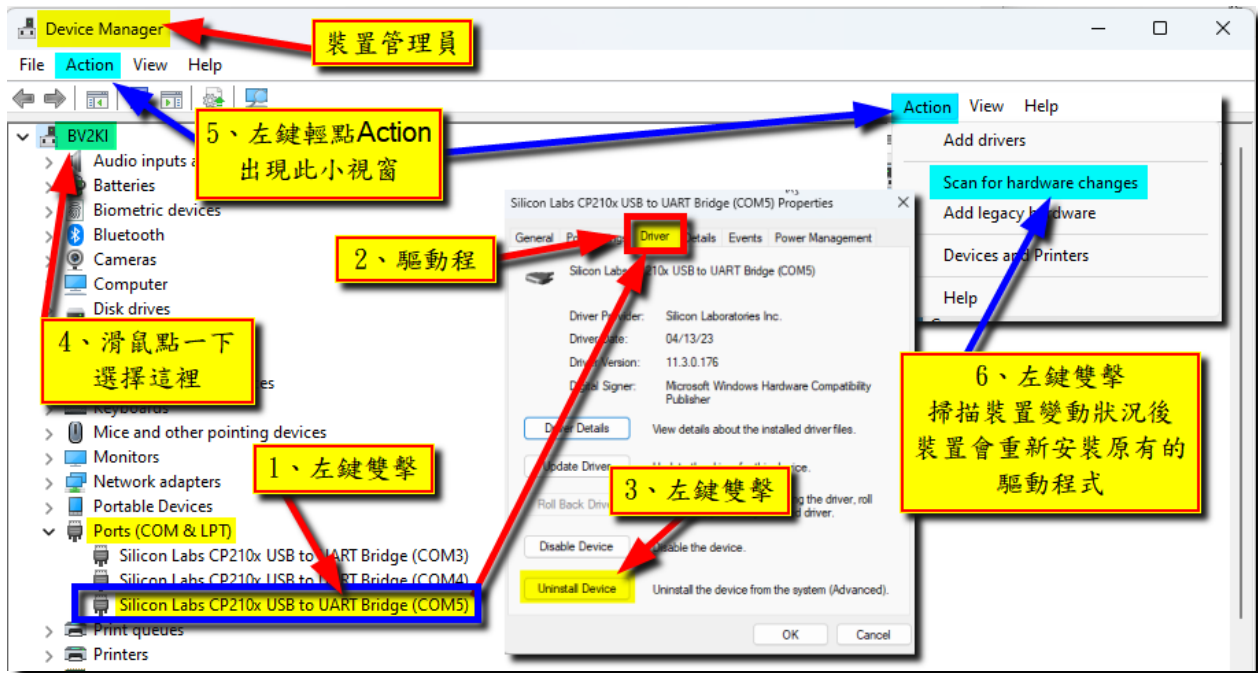
圖\_125 機組當機之警告信息\_驅動程式失效的主因



圖\_126 電腦管理工具視窗

圖\_125 是幾種可能產的警告視窗之顯現。解決方法:以「COM5」為例

- 1、到裝置管理員、如圖\_126 電腦管理工具視窗
- 2、依照圖\_127 之步驟執行。
- 3、執行當中，如果電腦告訴你:「需要重新開機」時，表示:「病的不輕」、那就去關機重啟電腦吧！
- 4、執行完成後，原先保存在電腦的驅動程式會重新安裝回來。



圖\_127 重新安裝原有的驅動程式

後記:如果你的機組持續的呈開機狀態，且久久並無讓 Omni-Rig 作動時，有可能讓 CAT 的功能暫停工作；如果突然有「稀有電台出現」、急著去想跟他做 QSO，當發射幾次後，結果造成機組的 USB 之「驅動程式」失效。此時，發射機會持續呈「發射狀」；這時候的第一個動作，就是將發射機關閉、再開機。

再依圖\_127 去改善，然而走到第 3 步驟時，電腦要求重新開機。

重新啟動電腦後，「驅動程式」恢復正常；省了圖\_127 之地 4、5 及第 6 的步驟。

有趣的發現！

## E\_5、單一機組的設定

本文前述所設定的機組係以「雙機」(以上的)為主。如何就「單一機組」做整合?

其實、本文已經說過，稍微變通一下就可以了

(答案:在 N1MM+中將 Radio\_2 設定成「Not Used」即可；請看圖\_5)。

為了讓你免傷腦筋起見，我另外依據本文，予以簡化，分別就 WSJT-X 及 JTDX 兩種應用程式各去做了「單一機組」的設定。請看下面這篇文章。

[「單一機組在 N1MM+WSJT-X\(或 JTDX\)及 Gridtracker 於 Log4OM 的設定」](#)

後記:上面的文章中、我另將「MAHV APP」的設定方法也一併加了進去；

如果你已經開啟了上面的文章，請到這一節去閱讀：

[「C\\_3\\_C\\_1、MSHV 之「一般 General」設定」](#)。

如果尚未開啟上述文章時，請按[這裡](#)

可直接參考「MSHV APP」的設定。

F、參考：

[如何在文章內任意跳至各章節處\(開啟「Document Map」\)或 \(ALT+<--\)](#)

[如何在單一電腦上同時操作兩個以上的 WSJT 以及與其他軟體的整合設定](#)

[N1MM+的解碼清單視窗](#)

## 批次檔文件內容(「3APP\_WSJT\_JTDX\_2R.BAT」不在 N1MM+之環境下運行)

```
@echo off

::=====

Echo Remarks :APP' s Location:

::      C:\N1MM+;
::      C:\WSJT\wsjtx\bin;
::      C:\JTDX64\159\bin;
::      C:\MSHV;
::      C:\Program Files (x86)\N1MM Logger+;
::      C:\Program Files (x86)\GridTracker;
::      C:\Program Files (x86)\IW3HMH\Log4OM NextGen;
::      C:\Program Files (x86)\Afreet\OmniRig;

::=====

Echo Set path for all

set path=%path%;C:\N1MM+;C:\WSJT\wsjtx\bin;C:\JTDX64\159\bin;C:\MSHV;C:\Program Files
(x86)\N1MM Logger+;C:\Program Files (x86)\GridTracker;C:\Program Files (x86)\IW3HMH\Log4OM
NextGen;C:\Program Files (x86)\Afreet\OmniRig

::=====

Echo Kill Last invocation, if any

taskkill /fi "imagename eq wsjtx.exe" /f /t
taskkill /fi "imagename eq JTDX.exe" /f /t
taskkill /fi "imagename eq MSHV_WIN64.exe" /f /t
taskkill /fi "imagename eq N1MMLogger.net.exe" /f /t
taskkill /fi "imagename eq Gridtracker.exe" /f /t
taskkill /fi "imagename eq L4ONG.exe" /f /t
taskkill /fi "imagename eq OmniRig.exe" /f /t

::=====

::Echo RUN N1MM+ with WSJT-X 、 JTDX or MSHV(No OmniRig)

::start N1MMLogger.net.exe ini="4APP_N1MM_WSJT_1R.ini"
::start N1MMLogger.net.exe ini="4APP_N1MM_WSJT_2R.ini"
::start N1MMLogger.net.exe ini="4APP_N1MM_JTDX_1R.ini"
::start N1MMLogger.net.exe ini="4APP_N1MM_JTDX_2R.ini"
::start N1MMLogger.net.exe ini="4APP_N1MM_MSHV_1R.ini"
::start N1MMLogger.net.exe ini="4APP_N1MM_MSHV_2R.ini"
```



```

::=====
Echo Run MSHV (Using Schedule_1 or Schedule_2 as well)
start C:\MSHV\Rig1\MSHV_WIN64.exe
::Set Rig_1 for IC-7300 if using OmniRig

::start C:\MSHV\Rig2\MSHV_WIN64.exe
::Set Rig_2 for IC-9700 if using OmniRig
::=====

::Echo Run JTDX
::start JTDX.exe --rig-name=IC7300
::Set Rig_1 for IC7300 if using OmniRig

start JTDX.exe --rig-name=IC9700
::Set Rig_2 for IC9700 if using OmniRig
::=====

::Echo Run WSJT-X
::start wsjtx.exe --rig-name=IC7300
::Set Rig_1 for IC7300 if using OmniRig

::start wsjtx.exe --rig-name=IC9700
::Set Rig_2 for IC9700 if using OmniRig
::=====

Echo RUN GridTracker
start GridTracker.exe
::=====

Echo RUN Log4OM(有兩種選擇:Case A 或 Case B)
:: Case A---[3 Apps] : Lo4OM+GridTracker+WSJT-X (簡稱: L4OM+GT+JT)
Start L4ONG.exe --CONFIG:fb8fb0998762484989170ff9368798ad
::上面這一行的尾端「青色」的字體是屬於「BV2KI」的 Schedule_1。
::來源取自這個路徑 C:\Users\BV2KI\AppData\Roaming\Log4OM2\user\
::檔名為:「scheduler_fb8fb0998762484989170ff9368798ad_user.xml」
::採用 Schedule_1

::-----
:: Case B---[4 Apps] : Lo4OM+GridTracker+WSJT-X+N1MM (簡稱: L4OM+GT+JT+N1MM)

```

```
::Start L4ONG.exe --CONFIG:ebc6924eb65e414fb5033de7b2a4afe9
::上面這一行的尾端「青色」的字體是屬於「BV2KI」的 Schedule_2
::來源取自這個路徑 C:\Users\BV2KI\AppData\Roaming\Log4OM2\user\
::檔名為: 「scheduler_ebc6924eb65e414fb5033de7b2a4afe9_user.xml」
:「4 Apps」係專為配合 N1MM+而設
(因為 N1MM+無法採用 OmniRig 的原因而設成 Schedule_2)
::=====
@echo off
exit
```