

行照 OCR 強化與 CarData 衝突防護

日期：2026-06-17 模組：HBSTK/source/TLIB.PRГ、WINFORM.PRГ、c_code/picklist_cursor.c.c

一、訴求與現象

本輪針對「貼圖辨識行車執照 (OCR)」做五項強化，核心是讓使用者能在車輛資料維護中順手以 OCR 建檔。最關鍵也最危險的問題是：當使用者正在 P1111 (車輛資料維護，畫面已開著 CarData) 時，又用 OCR 寫入 CarData，會發生工作區衝突。

項目	訴求／現象
VIN 貼上	GET 內貼行照後跳 VIN 五段拆解，原只能貼 VIS(7碼)，需增加「貼上完整 VIN(17碼)」。
缺牌照	掃描行照若辨識不到牌照號碼，原本直接中止匯入；需自動虛擬一個牌照當主鍵以利建檔。
選單順位	名片／行照兩個 OCR 選單，希望行照排在前面 (較常用)。
辨識後失焦	OCR 跑完後 STK32 視窗失去焦點，要手動點一下才能繼續操作。
P1111 衝突	在車輛資料維護畫面中用 OCR 建檔，可能報錯／崩潰／覆寫當前資料；且新增的車不會出現在清單。

二、根因分析 (P1111 衝突)

P1111 用 WinForm() 以 alias CarData 開檔 (USE ... SHARED NEW)，整個編輯期間保持開啟。OCR 寫檔的 VehicleOcrToDB 卻又呼叫 OpenSingleDbf("CarData")，而 OpenSingleDbf 只檢查 MainCdx，沒有檢查目標 alias 是否已開。Harbour 中 alias 全域唯一，對已開的 CarData 再 USE ... NEW 會造成下列其一：

後果	機制與嚴重性
報錯中斷	USE 同名 alias → 「alias already in use」runtime error → 跳錯誤對話框、OCR 中斷。
P1111 崩潰	若續行到結尾 CarData->(dbCloseArea()) → 把 P1111 正在用的工作區關掉 → 後續 browse/READ 崩潰。
資料覆寫	若 CarData-> 指到 P1111 既有區，REPLACE 可能寫到當前那筆 → 覆寫使用者正看的資料。

觸發路徑釐清：GET 欄位內 Ctrl+V → GETSYS(9002) → OcrPasteMenu(.T.) → PasteHistoryPick（只貼歷史字串，不寫檔，安全）；只有 browse 非 GET 狀態 → WINFORM.PRG → OcrPasteMenu(.F.) → VehicleOcrToDB 才會寫檔並衝突。

三、踩坑與陷阱

坑	說明
OpenSingleDbf 無已開檔防護	只查 MainCdx 對應 cdx，不查目標 alias 是否已被別處開著；名片 OCR(DocOcrToDB) 也同源（已開檔再開→第二實例、瀏覽看不到）。
短路求值不可依賴	原想用 IF !lReuse .AND. !OpenSingleDbf(...) 省事，但若編譯器未短路會多開一次；改用 lOpenOK 旗標 + 巢狀 IF，與短路與否無關。
終端機亂碼誤讀字面	調選單順位時，cp950 終端把 BIG5 顯示成亂碼，害我把「行照」誤讀成「車牌／車輛」；改用 Python 逐字 .decode('big5') + hex 確認後才動。
秒數裝不下 4 碼	虛擬牌照本想用「當天秒數」，但秒數最大 86399 需 5 碼，4 碼裝不下 → 改用「月+流水號」。
新車不顯示	即使寫檔成功，P1111 的 browse 不會自動重讀；必須在 WINFORM 主動 refreshAll 才會出現。

四、解決方法

方案 3=衝突防護 + 新車即時顯示，分兩檔：

修法	做法
A. 已開檔保護 (VehicleOcrToDB)	開頭 CheckDbfOpen("CarData") 若已開 → 重用既有工作區(dbSelectArea + 記住原索引 + lReuse)，不再 OpenSingleDbf；用 lOpenOK 旗標控制（自開才呼叫）。結尾 lReuse 則還原索引、不關區、記錄指標留在新車那筆，否則照舊 dbCloseArea + 還原工作區。虛擬牌照用罄中止路徑同步處理。

修法	做法
B. browse 即時刷新 (WINFORM 9002)	OcrPasteMenu(.F.) 之後套用既有樣板 refreshAll + stabilize + GetList:display + WinFormShow → 新車立即出現在清單並定位。WINFORM 為全域共用，但對其他表單無害（只多一次重繪）。

其餘四項：

項目	做法
VIN 貼上	VinShowMsg 選項加「V. 貼上VIN」，回傳 1=VIS／2=完整VIN／3=取消；回傳值先存 LOCAL 再 DO CASE（不可呼叫兩次否則彈兩次框）。
虛擬牌照	格式 民國年(3)+月(2)+流水號(3)=8碼無分隔（如 11506001）；流水 001 起找當月第一個未用號（不撞既有→走新增）；001~999 用罄才警告中止（不回繞、不覆寫）；完成訊息標「(虛擬)」。
選單順位	OcrPasteMenu／PasteHistoryPick 兩選單把行照排第 1（名稱不變，只調順位＋同步 index 對應）。
辨識後拉回前景	新增 C 函式 STK_SetForeground() (GetActiveWindow + AttachThreadInput 繞過前景鎖 + SetForegroundWindow/BringWindowToTop/SetFocus)；3 處 hb_processRun 後呼叫。

五、使用技術與用法

技術	用法
CheckDbfOpen 已開檔判斷	寫 DBF 前先 CheckDbfOpen(alias,.F.) 判斷是否已被別處開著，已開就重用工作區，結尾不關區——為「OCR 寫檔不衝突」的通用範本。
Python bytes-only 改 BIG5 PRG	中文註解以真中文 str + .encode('big5') 寫入，不對既有檔做 decode/encode 循環；以 ASCII 錨點定位、用 assert count==1 確保唯一匹配。
C 函式併入既有 .c	STK_SetForeground 加在 picklist_cursor.c (已在 hbp, rebuild stk32 即重編，非 merged gtwt lib, 免跑 build_StkGTWVT.bat)。
WinForm 既有刷新樣板	直接沿用 ApplyKey 內 refreshAll/stabilize/GetList:display/WinFormShow，不自創寫法。

技術	用法
hbm2 -comp=bcc + RUN	改 TLIB.PRG/WINFORM.PRG/.c 後 rebuild Stk32.exe 即生效（本批不涉 PrnAsk）。

六、驗證結果

情境	結果
P1111 browse 內 OCR 行照建檔	不再 alias 衝突／崩潰，新車立即出現在清單並定位（含缺牌照的虛擬號）。
重複牌照	走更新覆寫，browse 顯示更新後的值。
不在 P1111（別處清單）建檔	自開自關路徑維持正常。
GET 內 Ctrl+V	VIN MessageBox 可選 P=VIS／V=完整VIN／C=取消，皆正確填入。
各次辨識後	STK32 自動回到前景，不需手動點。