

QEC MySQL Test Tool

使用 86Duino 進行 MySQL 連線、寫入、讀取與自動測試

開發指南

版本 1.0 | 2026 年 6 月

昭營科技股份有限公司

修訂紀錄

版本	日期	變更說明
1.0	2026 年 6 月	內部工程版本。

目錄

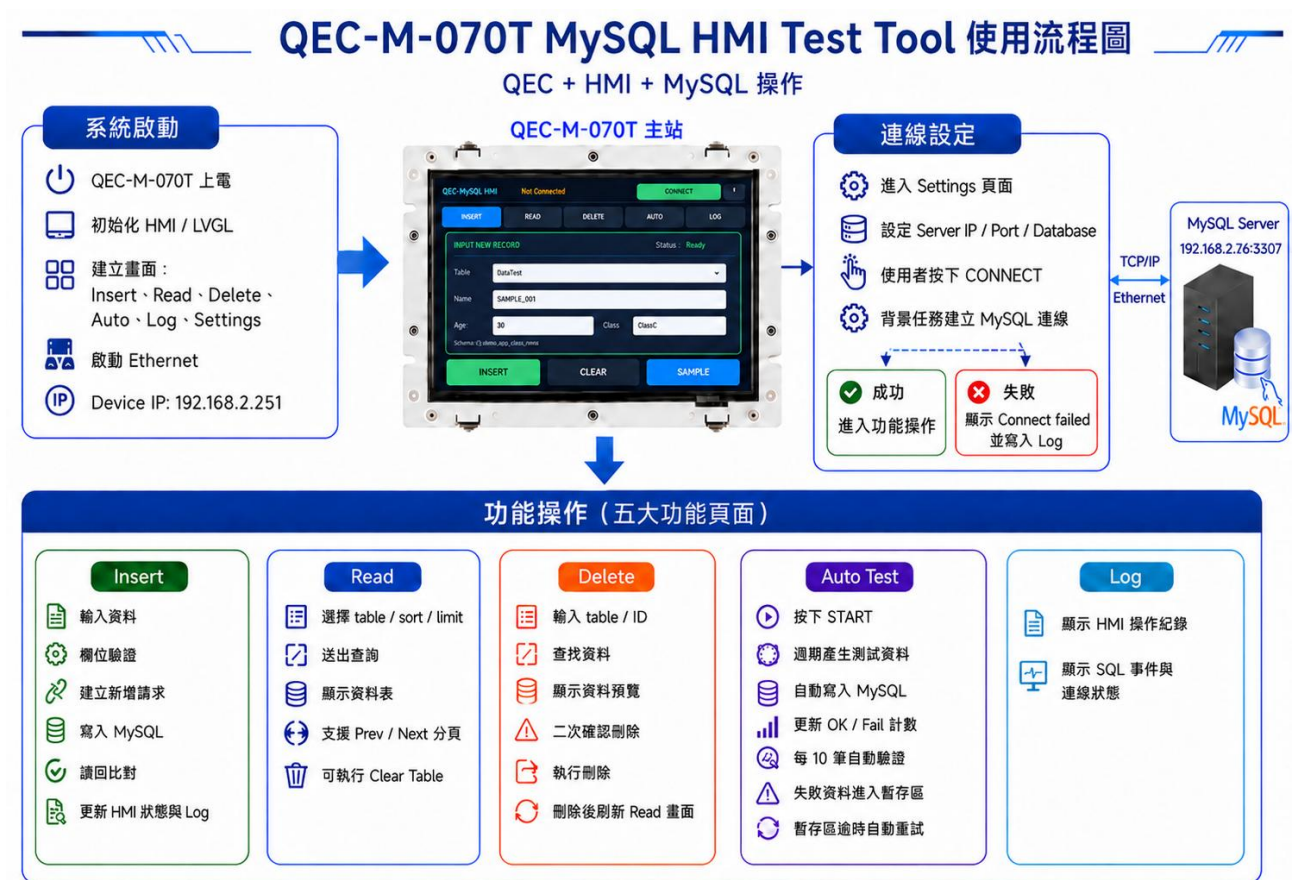
修訂紀錄	2
目錄	3
1. 概覽	5
1.1 系統架構	6
1.1.1 硬體需求	6
1.1.2 通訊架構	6
2. 設定 QEC-M-070T (燒入 .ino)	8
2.1 86Duino 環境設定	8
2.2 程式碼說明	10
2.2.1 網路設定	10
2.2.2 HMI	11
2.2.3 應用邏輯	12
2.2.4 Task / Runtime	13
2.2.5 MySQL 通訊	14
2.3 上傳程式碼	15
3. MySQL Server 與資料表設定	16
3.1 建議資料表格式	16
3.2 連線驗證	16
4. HMI 頁面功能說明	17
4.1 INSERT : 手動新增測試資料	17
4.2 READ : 查詢資料與分頁瀏覽	18
4.3 DELETE : 查找後安全刪除	19
4.4 AUTO : 自動寫入測試	20
4.5 LOG : 操作紀錄與 Retry Buffer	21

- 4.6 SETTINGS：執行期間連線設定.....22
- 5. 操作教學：首次端到端 MySQL 測試.....23
 - 5.1 目的23
 - 5.2 前置條件.....23
 - 5.3 步驟24
 - 5.4 預期結果.....27
 - 5.5 驗收方式.....27
 - 5.6 常見問題.....28
- 6. Log 與 Retry Buffer28
- 7. 常見問題排查29
- 8. 完整程式碼與下載.....29
- 9. 下一步30
- 10. 聯絡我們.....30

1. 概覽

這份指南展示一個初階可用的 MySQL HMI 測試工具架構：使用 QEC 直接透過 Ethernet 連線至 MySQL Server，並在觸控畫面上完成資料寫入、讀取、刪除、自動寫入測試與操作紀錄檢視。由三個主要部分組成：

- QEC_MySQL_Test_Tool.ino：燒入 QEC-M-070T，負責 Ethernet、MySQL 連線、SQL 任務、Auto Test 與應用邏輯。
- hmi_ui.cpp / hmi_ui.h：HMI 顯示層，負責 LVGL 物件、頁面切換、按鈕事件與畫面狀態更新。
- MySQL Server：提供測試資料庫與資料表，儲存 HMI 寫入的測試資料。



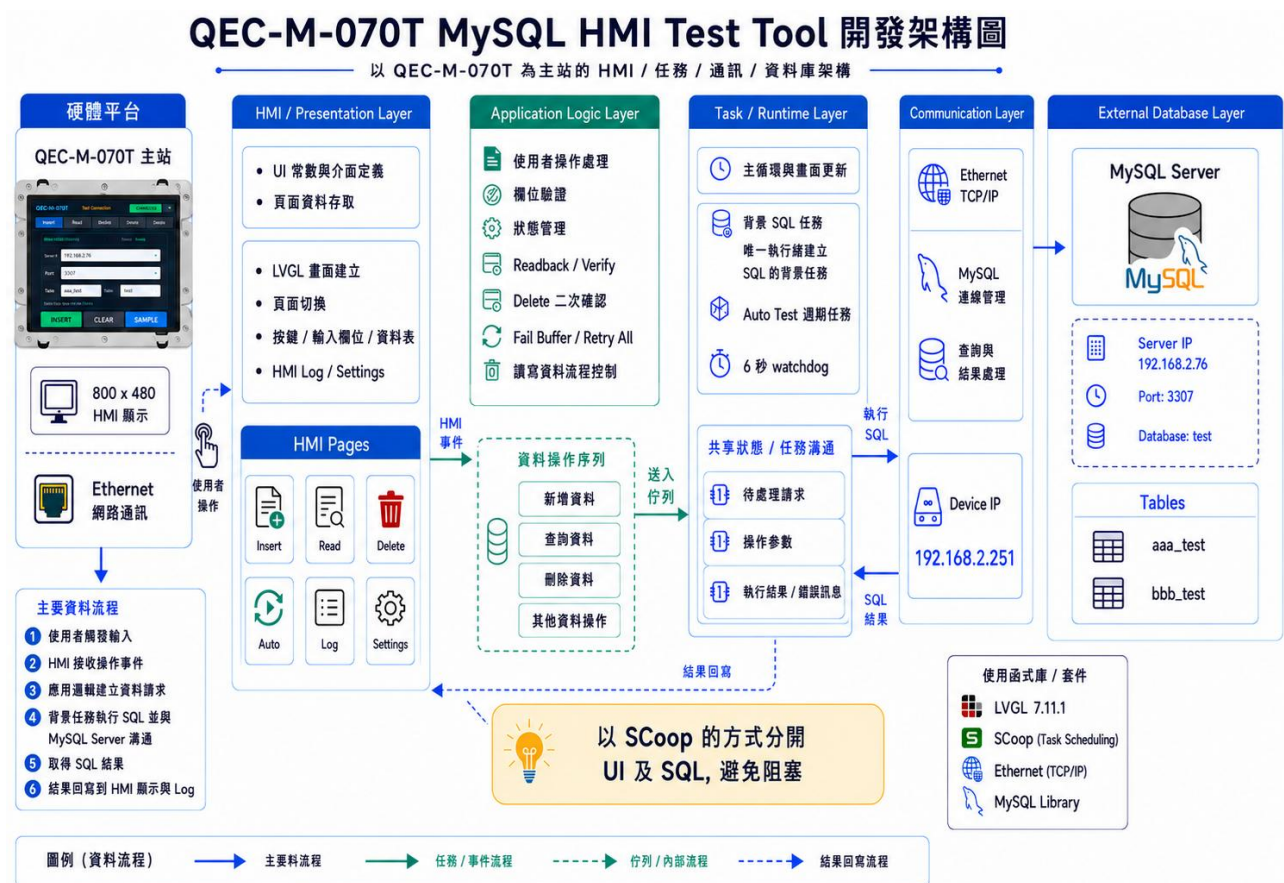
1.1 系統架構

關於此測試工具所需要的硬體及軟體。

1.1.1 硬體需求

元件	建議規格	說明
QEC HMI	QEC-M-070T 或相容 QEC HMI	執行 86Duino 程式及 HMI 操作。
上位 PC	Windows / Linux PC / NAS	執行 MySQL Server，需與 QEC 裝置位於可互通的同網段。
網路	Ethernet	QEC Device IP 與 MySQL Server IP 必須可互相連線。
資料庫	MySQL / MariaDB 相容	建立測試 database 與資料表。

1.1.2 通訊架構



QEC-M-070T 扮演多重角色：對上透過 Ethernet TCP/IP 與 MySQL Server 連線，負責送出 SQL 指令、取得結果並回寫至 HMI；對下則提供觸控式 HMI，讓使用者執行 INSERT、READ、DELETE、AUTO Test、LOG 與 Settings 等操作。

此架構以 SCoop 將 UI 更新與背景 SQL 任務分開，避免資料庫操作阻塞畫面，讓 MySQL 測試流程能在 HMI 上穩定完成。

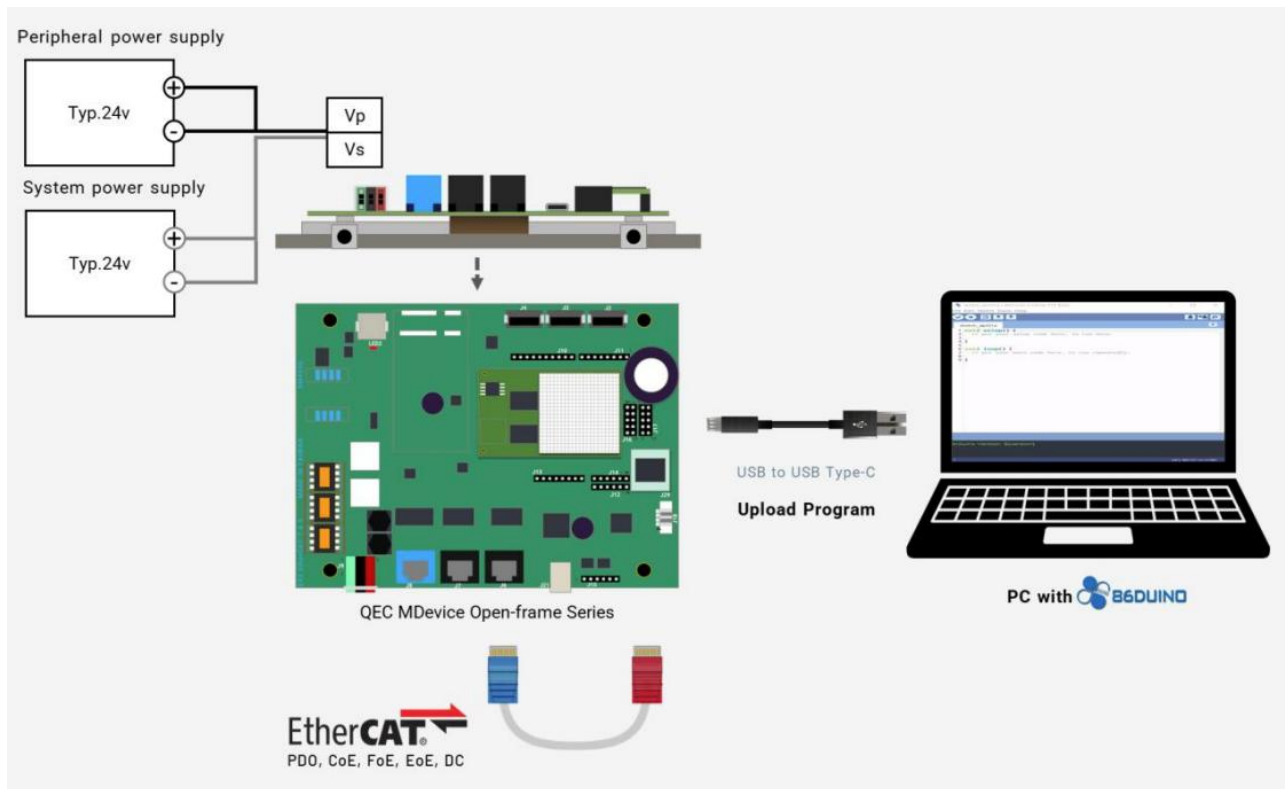
*** 注意**

本範例作為教學與驗證用途，適合在封閉式區域網路或實驗環境中使用。若用於正式設備或跨網段系統，建議改用具備權限控管、封包格式檢查或加密機制的 TCP/Socket 通訊架構。

2. 設定 QEC-M-070T (燒入 .ino)

將 QEC_MySQL_Test_Tool.ino 燒入 QEC-M-070T 中。

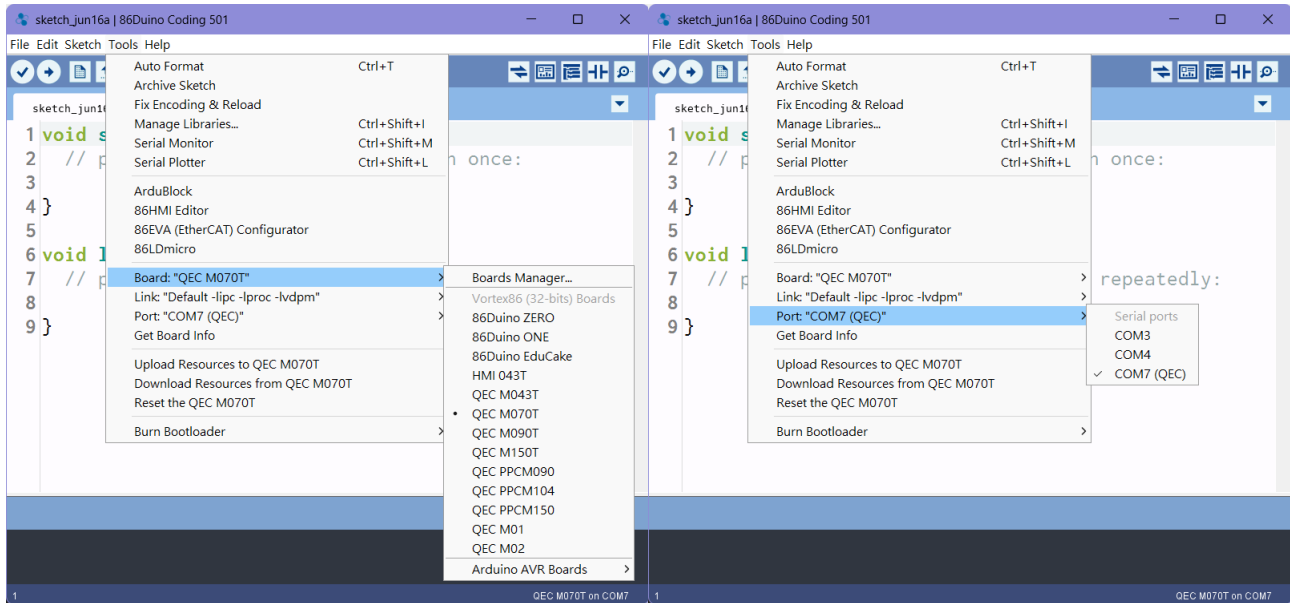
2.1 86Duino 環境設定



請依照以下步驟設定環境：

1. 使用 USB 線將 QEC-M-070T 連接到您的電腦 (需安裝 86Duino IDE) 。
2. 開啟 QEC 電源 。
3. 打開電腦上的“裝置管理員” (按 Win+X 後在選單中選擇) ->“連接埠 (COM 和 LPT)” , 展開連接埠 ; 您應該可以看到“Prolific PL2303GC USB 串列埠 COM 連接埠 (COMx)” ; 如果沒有 , 您需要 安裝所需的驅動程式。(Windows PL2303 驅動程式可在此[下載](#))
4. 開啟 86Duino IDE 。

5. 選擇正確的開發板：在 IDE 選單中，選擇「工具」>「開發板」>「QEC-M-070T」（或您使用的 QEC MDevice 型號）。
6. 選擇連接埠：在 IDE 的選單中，選擇“工具”>“連接埠”，然後選擇要連接到 QEC MDevice 的 USB 連接埠（在本例中為 COM7 (QEC)）。



2.2 程式碼說明

針對 QEC-M-070T 上燒入的程式碼做說明，大致上可分為五個部分：網路設定、HMI、應用邏輯、Task / Runtime 任務，以及 MySQL 通訊。

層級	代表檔案/函式	功能
網路設定	<code>IPAddress ip, IPAddress server_addr, mysql_port, g_database_name</code>	設定 QEC-M-070T 的 Device IP、MySQL Server IP、Port 與 Database。
HMI	<code>build_hmi(), build_page_insert(), build_page_read(), set_*</code>	建立 HMI 頁面、按鈕、輸入欄位與資料表畫面，並更新狀態、Log 與查詢結果。
應用邏輯	<code>app_connect(), app_insert(), app_read(), app_delete()</code>	接收 HMI 操作事件，進行欄位檢查、狀態管理、Delete 二次確認與 SQL 操作排程。
Task / Runtime	<code>sqlTask, scoopTask1, loop()</code>	將 UI 更新與背景 SQL 任務分開執行，避免 MySQL 操作阻塞 HMI 畫面。
MySQL 通訊	<code>exec_sql_operation(), exec_insert_raw(), sql_select_into_result()</code>	負責與 MySQL Server 建立連線、執行 INSERT / SELECT / DELETE，並整理 SQL 結果與 LOG。

2.2.1 網路設定

在 .ino 開頭確認以下設定符合你的網路環境:

```
IPAddress ip(192, 168, 2, 251); // QEC Device IP
IPAddress server_addr(192, 168, 2, 76); // MySQL Server IP
char user[] = "qec";
char password[] = "qec";
int mysql_port = 3307;
char g_database_name[32] = "test";
```

*注意

IP 位址需與 PC / MySQL Server 位於可互通網段，且不可與其他設備衝突。Server IP、Port 與 Database 可在 sketch 中修改後重新燒錄，也可於 HMI 的 SETTINGS 頁暫時調整；重新開機後會回到 sketch 預設值。

2.2.2 HMI

HMI 主要定義使用者看到的操作介面，包含 INSERT、READ、DELETE、AUTO、LOG 與

SETTINGS 等頁面。畫面建立與切換集中在 hmi_ui.cpp 中，主程式則透過 set_*() 函式更新

狀態、查詢結果與 Log。

```
build_hmi();
build_page_insert(scr);
build_page_read(scr);
build_page_delete(scr);
build_page_auto(scr);
build_page_log(scr);
build_page_settings(scr);
set_insert_status("Ready", COLOR_GREEN);
set_read_status("Ready", COLOR_CYAN);
set_mysql_status_all("Not Connected", COLOR_YELLOW);
```

注意

HMI 只負責顯示與接收操作事件，不直接執行 MySQL 指令。

所有 MySQL 操作會由主程式中的 app_*() 函式接手處理，避免畫面程式與資料庫邏輯混在一起。

2.2.3 應用邏輯

應用邏輯負責處理使用者在 HMI 上的操作，例如 CONNECT、INSERT、READ、DELETE 與 SETTINGS APPLY。這一層會檢查輸入欄位、整理 SQL 參數、管理狀態，並將要執行的 SQL 操作送入背景任務。

```
oid app_connect();  
void app_insert();  
void app_read();  
void app_delete();  
void app_apply_settings();
```

例如使用者按下 INSERT 時，程式會先取得 HMI 欄位內容，再檢查 Name、Age、Class 是否有效，最後設定 SQL 操作類型：

```
g_sql_op = SQL_INSERT_HMI;  
g_sql_pending = true;
```

注意

DELETE 採用二次確認流程，必須先 FIND 到資料並顯示 preview，接著按 DELETE 後再次確認，才會真正送出刪除指令。

2.2.4 Task / Runtime

Task / Runtime 負責將 HMI 畫面更新與 MySQL SQL 執行分開處理。主循環 loop() 維持畫面更新、Watchdog reset 與 SQL 結果回收；sqlTask 負責執行可能阻塞的 SQL 操作；scoopTask1 則負責 AUTO Test 的週期寫入。

```
void loop();

defineTaskLoop(sqlTask) {
    if (g_sql_pending) {
        exec_sql_operation();
        g_sql_pending = false;
        g_sql_done_main = true;
    }
}

defineTaskLoop(scoopTask1) {
    auto_insert_test_row();
}
```

注意

MySQL 查詢或寫入可能需要等待網路回應，因此不可直接在 HMI callback 中執行。

本程式使用 SCoop 將 SQL 操作放到背景任務中，讓 HMI 畫面在資料庫操作期間仍能保持回應。

2.2.5 MySQL 通訊

MySQL 通訊負責與 MySQL Server 建立連線、送出 SQL 指令、取得查詢結果，並將結果整理後回傳給 HMI 顯示。主要操作包含 CONNECT、INSERT、READ、FIND、DELETE、VERIFY、RETRY 與 CLEAR TABLE。

```
exec_sql_operation();
exec_insert_raw(table, name, age, cls, result);
sql_select_into_result(table, sort, limit, offset, result);
sql_count_rows(table, result);
```

例如 INSERT 會組成 SQL 指令後送出：

```
INSERT INTO test.aaa_test (name, age, class_name)
VALUES ('SAMPLE_001', 60, 'ClassE');
```



READ 則會依照 Table、Sort、Limit 與 Page 查詢資料：

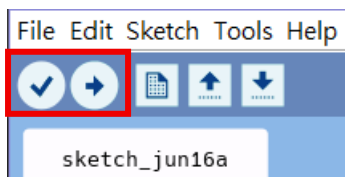
```
SELECT id, name, age, class_name
FROM test.aaa_test
ORDER BY id DESC
LIMIT 10 OFFSET 0;
```

注意

MySQL 通訊只負責 SQL 執行與結果整理，不直接更新 LVGL 畫面。
SQL 執行完成後，結果會交回主循環，再由 HMI 畫面層更新狀態、資料表、Log 與錯誤訊息。

2.3 上傳程式碼

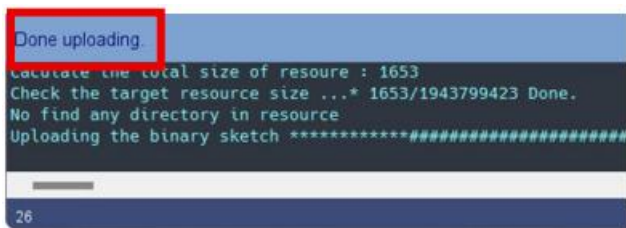
程式碼寫完成後，點選工具列的進行編譯，確認編譯完且無錯誤後，即可點選上傳。



(圖中 sketch 名僅為示意)

注意
上傳期間 IDE 視窗請勿關閉、USB 線勿拔除;一般 sketch 上傳約 10~30 秒。

一般 sketch 上傳約 10~30 秒。視窗最後出現 Done uploading 即代表上傳成功；若出現紅字錯誤，則針對錯誤處去調整。如有任何其他異常，請與我們聯繫。



3. MySQL Server 與資料表設定

本工具預設連線至 `database : test`，並提供兩個測試資料表選項：`aaa_test` 與 `bbb_test`。

兩張表的欄位結構相同，方便在教學時切換不同資料表進行測試。

3.1 建議資料表格式

```
CREATE DATABASE IF NOT EXISTS test;
USE test;

CREATE TABLE IF NOT EXISTS aaa_test (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(32) NOT NULL,
  age INT NOT NULL,
  class_name VARCHAR(32) NOT NULL
);

CREATE TABLE IF NOT EXISTS bbb_test LIKE aaa_test;
```

3.2 連線驗證

- QEC Device IP 與 MySQL Server IP 必須位於可互通的同網段。
- MySQL Server 需允許 qec 使用者從 QEC 裝置所在網段連線。
- 若使用非預設 port，請在 SETTINGS 頁輸入新的 port 後按 APPLY。
- CONNECT 成功後，HMI header 會切換為連線狀態，LOG 頁也會留下連線紀錄。

4. HMI 頁面功能說明

4.1 INSERT : 手動新增測試資料

The screenshot shows the 'INPUT NEW RECORD' form in the QEC MySQL HMI. The form is titled 'INPUT NEW RECORD' and has a status of 'Ready'. It contains the following fields and buttons:

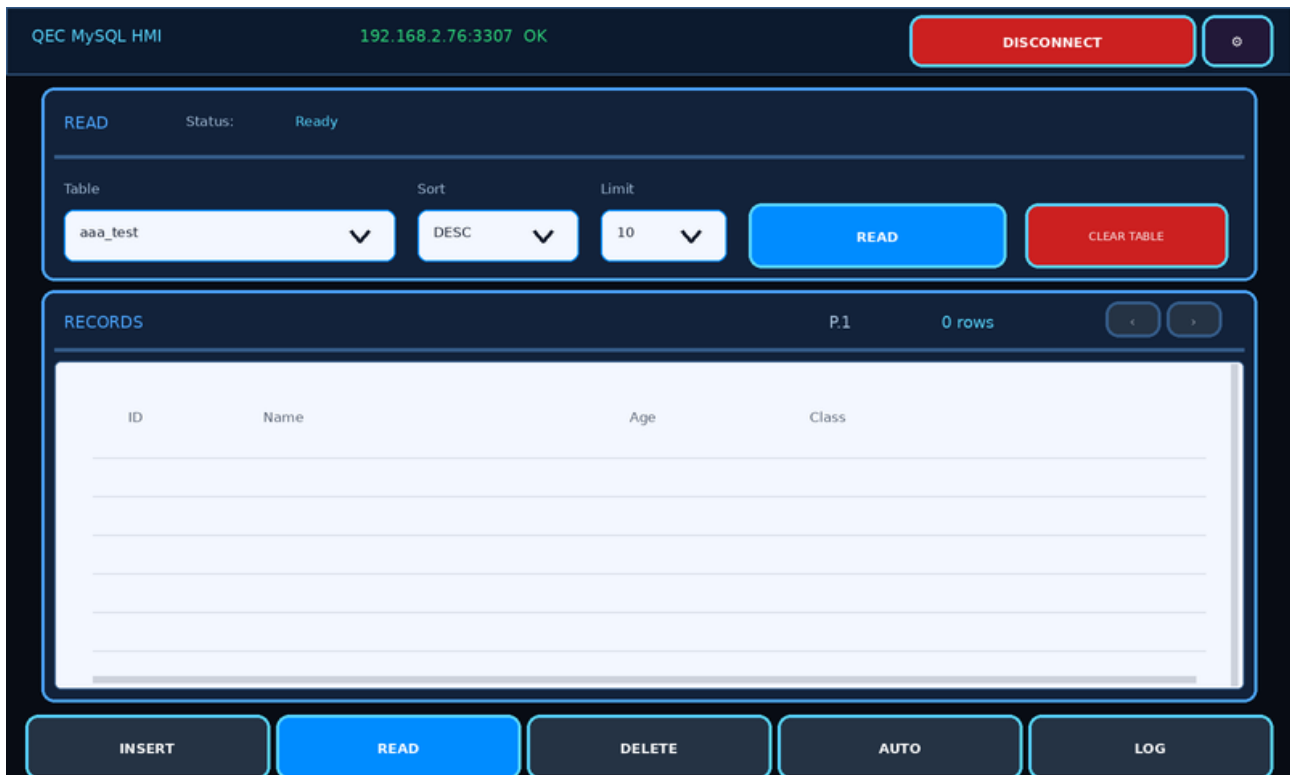
- Table:** A dropdown menu with 'aaa_test' selected.
- Name:** A text input field containing 'D'.
- Age:** A text input field containing '60'.
- Class:** A text input field containing 'ClassE'.
- Schema:** A label indicating the schema is 'id, name, age, class_name'.
- Buttons:** Three large buttons are located below the form: 'INSERT' (green), 'CLEAR' (grey), and 'SAMPLE' (blue).
- Bottom Bar:** A row of five smaller buttons: 'INSERT' (blue), 'READ' (grey), 'DELETE' (grey), 'AUTO' (grey), and 'LOG' (grey).

INSERT 頁用於手動建立一筆測試資料。使用者選擇 table，填入 name、age、class 後按

INSERT，程式會送出 INSERT INTO 指令。

欄位/按鈕	功能
Table	選擇 aaa_test 或 bbb_test。
Name / Age / Class	輸入測試資料，對應資料表欄位 name、age、class_name。
INSERT	寫入一筆資料至 MySQL。
CLEAR	清空輸入欄位。
SAMPLE	自動產生 SAMPLE_### 測試資料，方便快速測試。

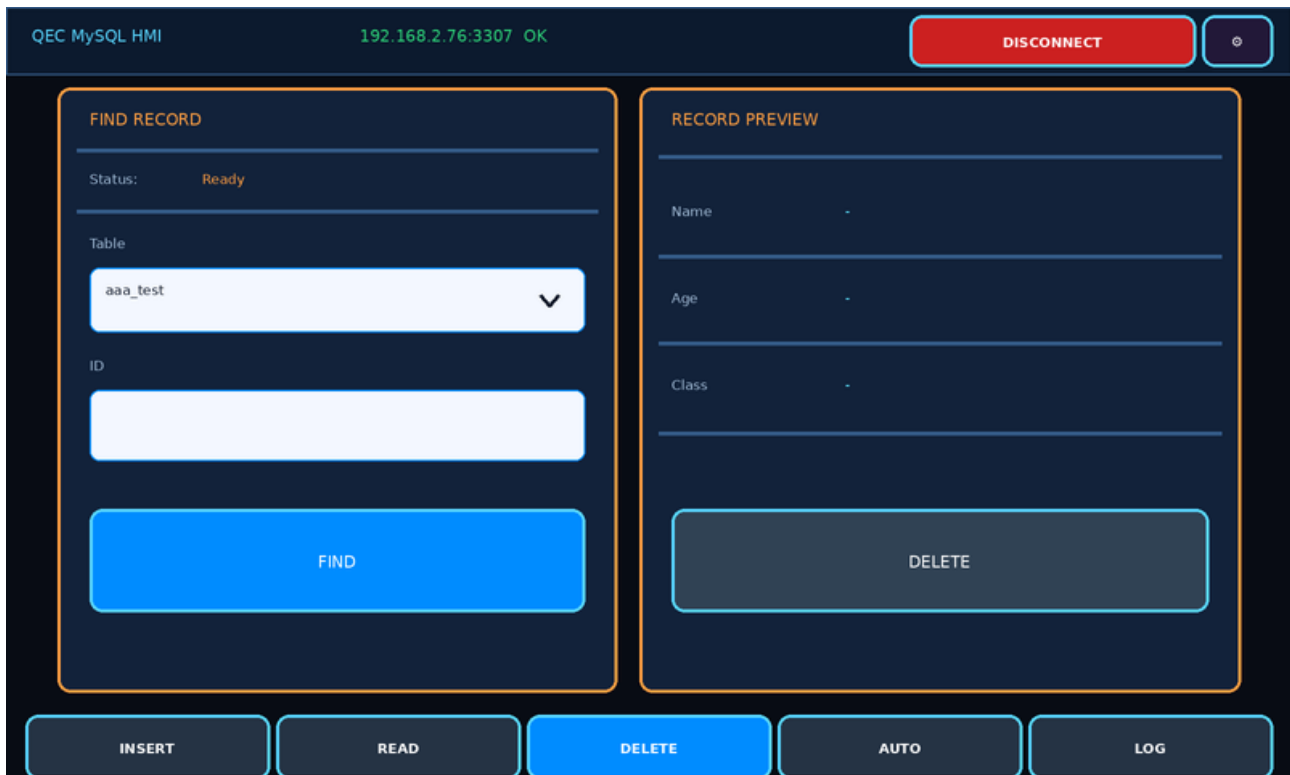
4.2 READ : 查詢資料與分頁瀏覽



READ 頁用於查詢資料表內容。可選擇排序方向、每頁筆數，並透過 PREV / NEXT 進行分頁。畫面會顯示目前範圍與總筆數，例如 1-10 / 35。

- Sort : ASC / DESC，預設 DESC，方便先看最新資料。
- Limit : 10 或 20，每次查詢最多顯示 SQL_MAX_ROWS 筆。
- CLEAR TABLE : 採二次確認設計，用於清空目前選取資料表。

4.3 DELETE : 查找後安全刪除

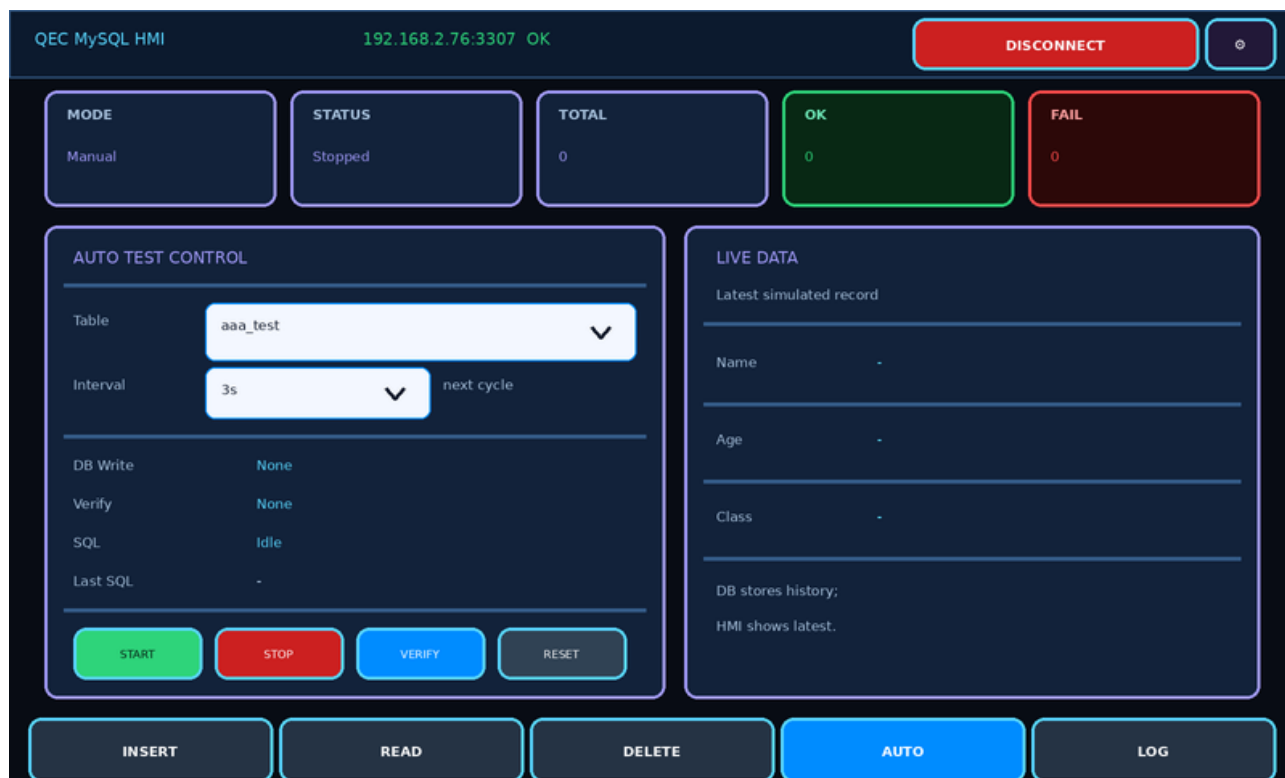


DELETE 頁會先用 ID 查找資料，顯示 name、age、class，preview 後才允許刪除。DELETE 按鈕進入確認狀態後，必須在倒數時間內再次點擊才會真正刪除。

安全設計

DELETE 不直接執行刪除，而是透過 FIND → PREVIEW → DELETE → CONFIRM 的流程，降低教學與測試時誤刪資料的風險。

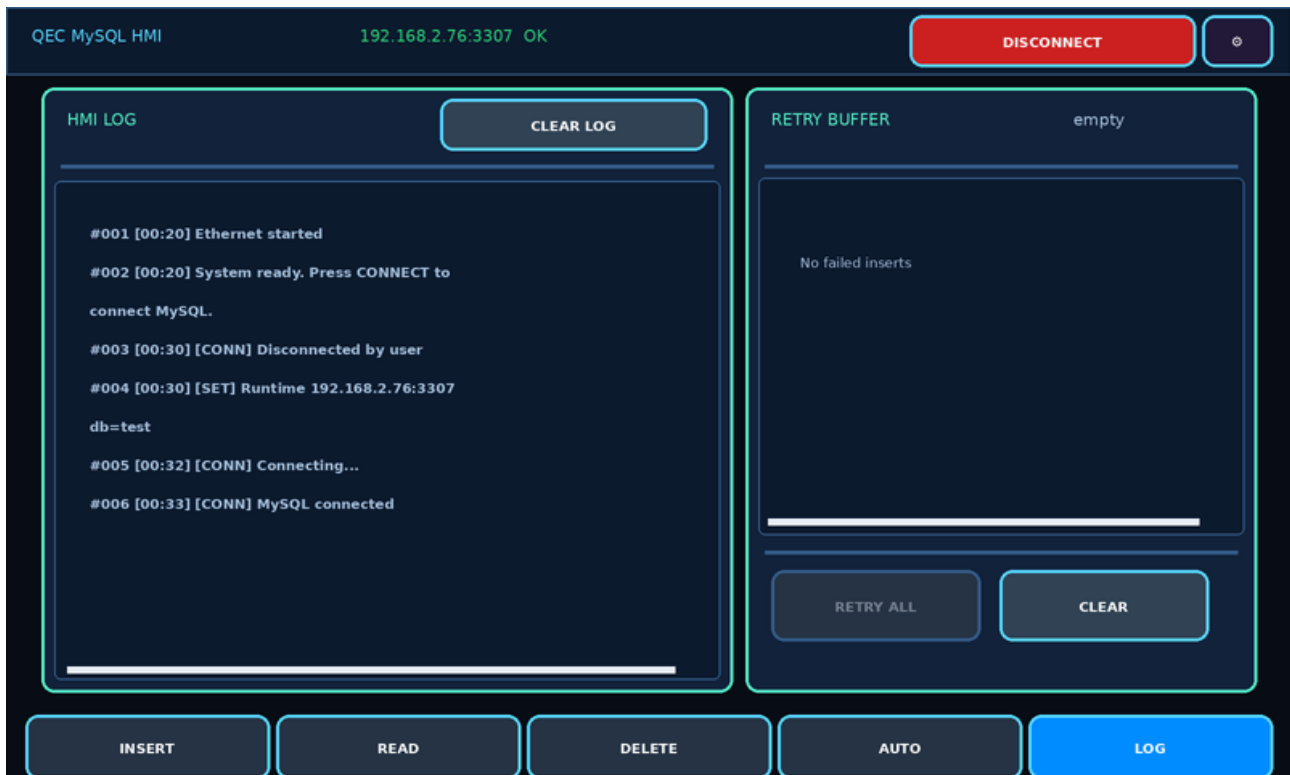
4.4 AUTO : 自動寫入測試



AUTO 頁可定期寫入模擬資料，用於觀察連線穩定性、SQL 耗時、成功/失敗統計與最新寫入資料。

- Interval : 可選擇 3s、5s、10s。
- TOTAL / OK / FAIL : 顯示自動測試累計結果。
- VERIFY : 讀回最新資料，確認資料庫內容與 HMI 顯示一致。
- RESET : 重置 Auto Test 計數。

4.5 LOG : 操作紀錄與 Retry Buffer



LOG 頁左側顯示 HMI 操作紀錄，右側顯示 failed insert retry buffer。當自動寫入失敗時，資料會暫存在 buffer，使用者可按 RETRY ALL 重新送出。

4.6 SETTINGS : 執行期間連線設定

QEC MySQL HMI 192.168.2.76:3307 OK DISCONNECT

MYSQL SERVER **RUNTIME**

Server IP
192.168.2.76

Port
3307

Database
test

User qec

APPLY

DEVICE

Device IP 192.168.2.251

MAC Address DE:AD:BE:EF:FE:ED

Model QEC-M-070T

Display 800x480 / LVGL 7.1.11

INSERT READ DELETE AUTO LOG

SETTINGS 頁可修改 MySQL Server IP、Port 與 Database。按 APPLY 後會更新 runtime 設定並斷開目前連線，下一次 CONNECT 會使用新的設定。

5. 操作教學：首次端到端 MySQL 測試

本章整合前述設定，以 Purpose / Prerequisites / Steps / Expected Results / Verification / Common Issues 的結構，帶使用者完成一次完整 MySQL HMI 測試。

5.1 目的

完成 CONNECT → INSERT → READ → DELETE → AUTO → LOG 的基本流程，確認 QEC HMI 與 MySQL Server 能正常連線、寫入、讀取、刪除與記錄測試狀態。

5.2 前置條件

- QEC_MySQL_Test_Tool.ino 已成功上傳至 QEC HMI。
- QEC Device IP 與 MySQL Server IP 可互相連線。
- MySQL Server 已建立 test database，以及 aaa_test / bbb_test 測試資料表。
- MySQL 使用者 qec 具備 SELECT、INSERT、DELETE 權限。

5.3 步驟

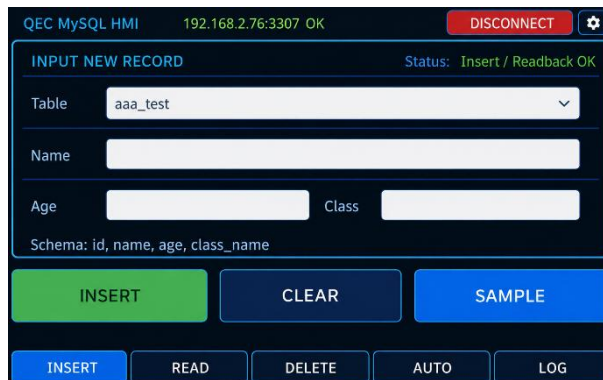
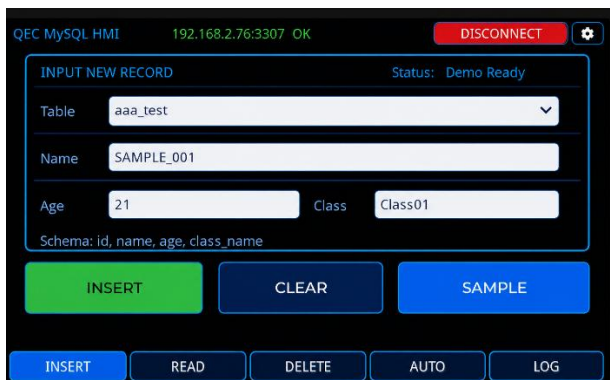
1. 開機後進入 INSERT 頁，確認 header 狀態為 Not Connected。



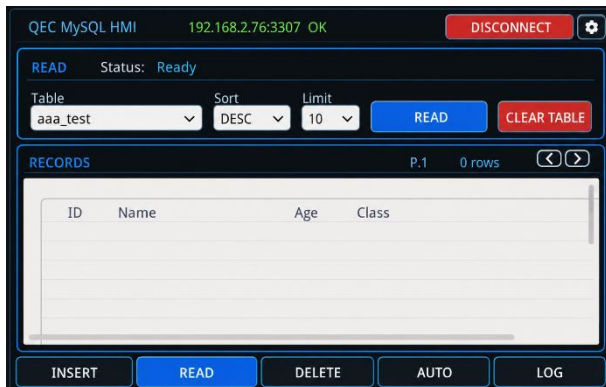
2. 按 CONNECT，確認狀態變為 Connected，LOG 頁出現連線成功紀錄。



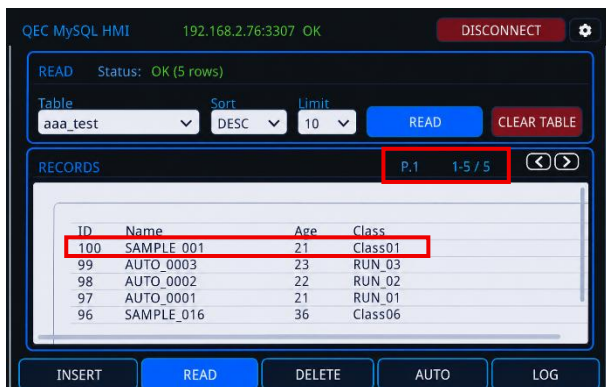
3. 在 INSERT 頁選擇 aaa_test，按 SAMPLE 產生測試資料，再按 INSERT。



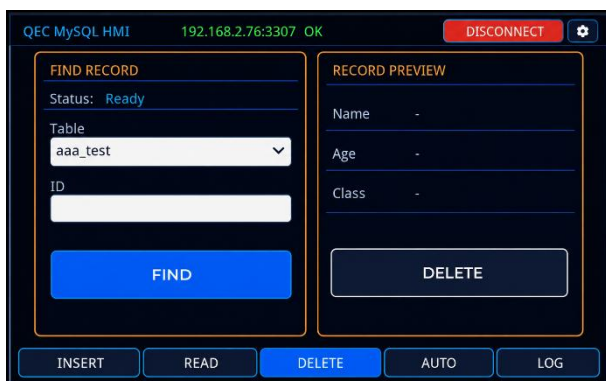
4. 切換至 READ 頁，選擇 aaa_test，Sort 選 DESC，Limit 選 10，按 READ。



5. 確認剛剛寫入的資料出現在 RECORDS 表格中，rows summary 顯示目前範圍與總筆數。



6. 切換至 DELETE 頁，輸入該筆資料 ID，按 FIND，確認 preview 顯示正確資料。



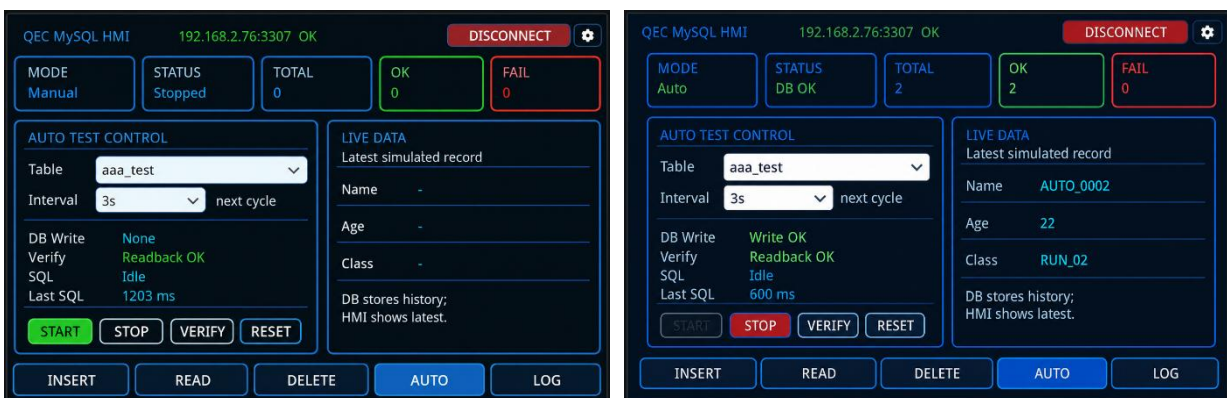
7. 按 DELETE 後再次確認，刪除該筆資料。



8. 回到 READ 頁重新 READ，確認該筆資料已不在列表中。



9. 切換至 AUTO 頁，選擇 aaa_test 與 3s interval，按 START。



10. 觀察 TOTAL / OK / FAIL、Last SQL 與 LOG 頁紀錄，確認自動寫入正常後按 STOP。



5.4 預期結果

- CONNECT 成功後，所有頁面共用的連線狀態會更新。
- INSERT 成功後，READ 頁可查到新增資料。
- DELETE 經二次確認後才會移除資料。
- AUTO Test 期間 OK 數持續增加，FAIL 維持 0 或可由 Retry Buffer 追蹤。

5.5 驗收方式

- MySQL 中 SELECT COUNT(*) 的結果與 READ 頁 total count 一致。
- LOG 頁能看到 CONNECT、INSERT、READ、DELETE、AUTO START/STOP 等紀錄。
- 斷線或 SQL 失敗時，HMI 狀態應顯示錯誤訊息，不應造成畫面卡死。

5.6 常見問題

- CONNECT failed：確認 Server IP、Port、MySQL 使用者權限與防火牆。
- INSERT failed：確認資料表欄位是否為 id, name, age, class_name。
- READ 無資料：確認 Table 選擇、Database 名稱與資料表內容。
- AUTO Test 有 FAIL：切換到 LOG 頁查看錯誤，必要時使用 RETRY ALL。

6. Log 與 Retry Buffer

HMI LOG 使用固定行數緩衝，避免長時間測試後畫面文字無限制增加。Retry Buffer 則保留自動寫入失敗的資料，讓使用者在連線恢復後能手動重試。

項目	用途	操作
HMI LOG	記錄連線、SQL 操作、Auto Test 狀態與錯誤訊息。	按 CLEAR LOG 可清除畫面紀錄。
Retry Buffer	保存自動寫入失敗的資料。	按 RETRY ALL 可重新寫入； CLEAR 可清空 buffer。
Last SQL	顯示最近一次 SQL 操作耗時。	用於粗略觀察 MySQL 回應速度。

7. 常見問題排查

問題	可能原因	處理方式
無法 CONNECT	IP/Port 錯誤、MySQL 未啟動、防火牆阻擋。	確認 SETTINGS、ping server、檢查 MySQL port 與權限。
INSERT 失敗	資料表不存在或欄位名稱不一致。	確認 aaa_test / bbb_test 具有 id, name, age, class_name。
READ 顯示 0 rows	資料表目前無資料，或選錯 database/table。	先 INSERT 一筆 sample，再回 READ 頁查詢。
DELETE 無法按	尚未 FIND 到有效 ID。	先輸入 ID 並按 FIND，preview 有資料後才能 DELETE。
AUTO FAIL 增加	連線中斷或資料庫暫時不可用。	查看 LOG，修復連線後使用 RETRY ALL。
畫面仍可操作但 SQL 慢	SQL 在 sqlTask 中阻塞，主 loop 仍持續更新 LVGL。	觀察 Last SQL，必要時降低 Auto Test 頻率。

8. 完整程式碼與下載

本指南對應的主要檔案如下。

- QEC_MySQL_Test_Tool.ino：主程式與 MySQL 邏輯。
- hmi_ui.cpp / hmi_ui.h：HMI 頁面與 UI API。
- myhmi.cpp / myhmi.h / myeva.cpp / myeva.h：86HMI Editor 產生檔。

9. 下一步

這個範例展示的是最基礎的 MySQL 連線與功能。QEC 平台的擴充方向包括:

- 加入 schema check，讓 HMI 可主動檢查資料表欄位是否符合範例需求。
- 提供 MySQL server 安裝與權限設定補充文件。
- 增加更多實際設備資料格式，例如溫度、狀態碼、計數器或生產線事件。
- 若要進入正式產品，可再加入權限管理、SQL 參數、保持連線與更完整的錯誤分類。

10. 聯絡我們

若您在依本指南實作的過程中遇到任何技術問題,或對 QEC 系列產品有採購或客製化需求,歡

迎透過下列任一管道與我們聯繫:

業務洽詢 / 採購	info@qec.tw (QEC 產品線(EtherCAT MDevice 等))
技術支援 / RD	info@icop.com.tw (ICOP Technology Inc.)