

# GF AI Box — 測試人員操作指南

文件版本：v2.0 適用韌體：gf\_ai\_box\_v4 更新日期：2026-06-14

---

## 一、測試環境準備

### 所需工具

- 藍芽串口工具（iOS：LightBlue / Serial Bluetooth Terminal；Android：Serial Bluetooth Terminal）
- CAN bus 分析儀（選配，用於驗證 CAN 訊號）
- 油門踏板（CAN 測試時需要）

### 連線方式

1. 開啟藍芽工具，掃描並連線到 AI Box 的 BLE 裝置
2. 完成握手後即可送出指令
3. 指令為純文字，大小寫不分，直接輸入後送出

### 重要說明

- 硬體 **Reset** 後預設為正常模式，必須先送 test\_enter 才能執行測試指令
  - 進入測試模式後，AI Box 的 AI 推論結果暫停處理（NPU 繼續運算但不派發事件）
  - 測試模式下如果 **60** 秒無任何指令，系統會自動退出測試模式並恢復正常
  - 若 TEST\_ALL 系列正在執行，請勿重複送出測試指令
- 

## 二、基本控制指令

### 指令

test\_enter

test\_exit

## 三、大項一：BLE JSON 測試

測試目的：驗證各警示事件觸發時，藍芽送出的 JSON 格式與內容正確。注意：只送出 BLE JSON，不觸發 CAN bus 指令與 Buzzer。

### 3.1 單項測試指令

#### Violation (草地違規)

指令	說明	預期 BLE 回傳 JSON
test_ble violation 1	觸發 violation level=1	{"response_type":"violation","content": {"id":"...","date":"...","type":"lane","level":1}}

test_ble violation 0	解除 violation	{ "response_type": "violation", "content": { "id": "...", "date": "...", "type": "lane", "level": 0 } }
----------------------	--------------	---

### Collision (碰撞警示)

指令	說明	預期 BLE 回傳 JSON
test_ble collision 1	觸發 collision level=1 ( type=vehicle )	{ "response_type": "collision_warning", "content": { "level": 1, "type": "vehicle" } }
test_ble collision 0	解除 collision	{ "response_type": "collision_warning", "content": { "level": 0, "type": null } }

### Alert (方向警示)

指令	說明	預期 BLE 回傳 JSON
test_ble alert 1 tree 0 null	左側 tree 警示，右側無	{ "response_type": "alert", "content": { "left": { "level": 1, "type": "tree" }, "right": { "level": 0, "type": null } } }
test_ble alert 0 null 1 person	左側無，右側 person	{ "response_type": "alert", "content": { "left": { "level": 0, "type": null }, "right": { "level": 1, "type": "person" } } }
test_ble alert 1 tree 1 vehicle	左右都有警示	{ "response_type": "alert", "content": { "left": { "level": 1, "type": "tree" }, "right": { "level": 1, "type": "vehicle" } } }
test_ble alert 0 null 0 null	解除所有 alert	{ "response_type": "alert", "content": { "left": { "level": 0, "type": null }, "right": { "level": 0, "type": null } } }

**Alert 指令格式：** test\_ble alert <左 level> <左 type> <右 level> <右 type>  
 type 可以是：tree / person / vehicle / null

## 3.2 BLE 全項自動測試

### 指令

test\_ble\_all

**TEST\_BLE\_ALL 執行順序：**

[00s] violation level=1 → 等 3 秒

[03s] violation level=0 → 等 2 秒

[05s] collision level=1 → 等 3 秒  
[08s] collision level=0 → 等 2 秒  
[10s] alert left=tree, right=無 → 等 3 秒  
[13s] alert clear → 等 2 秒  
[15s] alert left=tree, right=vehicle → 等 3 秒  
[18s] alert clear → 等 2 秒  
[20s] TEST\_BLE\_ALL\_DONE , 自動退出測試模式

---

## 四、大項二：CAN bus 測試

測試目的：驗證各警示事件觸發時，CAN bus 送出正確速度指令。注意：只觸發 CAN bus 速度指令，不送出 BLE JSON，不啟動 Buzzer。

### 速度對照表 事件

---

正常行駛

Violation / Alert

Collision

#### 4.1 單項測試指令

##### *Violation CAN 測試* 指令

---

test\_can violation 1

test\_can violation 0

##### *Collision CAN 測試* 指令

---

test\_can collision 1

test\_can collision 0

##### *Alert CAN 測試* 指令

---

test\_can alert 1

test\_can alert 0

#### 4.2 CAN 全項自動測試 指令

---

test\_can\_all

test\_can\_all 15

**TEST\_CAN\_ALL 執行順序 (預設觀察 10 秒) :**

[00s] violation level=1 → speed=35

↓ 觀察 10 秒 (踩油門, 確認轉速被限制)

[10s] violation level=0 → speed=240

↓ 等 3 秒 (確認恢復正常)

[13s] collision level=1 → speed=10

↓ 觀察 10 秒 (踩油門, 確認轉速被限制到最低)

[23s] collision level=0 → speed=240

↓ 等 3 秒

[26s] alert level=1 → speed=35

↓ 觀察 10 秒

[36s] alert level=0 → speed=240

↓ 等 3 秒

[39s] TEST\_CAN\_ALL\_DONE, 自動退出測試模式

---

## 五、大項三 : Buzzer 測試

測試目的：驗證各警示事件觸發時，Buzzer 發出正確嗶聲節奏。注意：只觸發 Buzzer，不送出 BLE JSON，不發 CAN bus 指令。

### Buzzer Pattern 對照表

Pattern

---

OFF (靜音)

GRASS (Violation 草地)

ALERT (Alert 方向警示)

COLLISION (碰撞)

### 5.1 單項測試指令

指令

---

test\_buzzer grass

test\_buzzer alert

test\_buzzer collision

test\_buzzer off

Buzzer 指令送出後立即生效，持續到下一個指令為止。

## 5.2 Buzzer 全項自動測試

### 指令

---

test\_buzzer\_all

**TEST\_BUZZER\_ALL** 執行順序：

[00s] GRASS 慢速嗶 - 等 4 秒

[04s] ALERT 中速嗶 - 等 4 秒

[08s] COLLISION 急促嗶 - 等 4 秒

[12s] OFF 靜音

[12s] TEST\_BUZZER\_ALL\_DONE，自動退出測試模式

---

## 六、全功能自動測試

### 指令

---

test\_all

test\_all 15

執行順序（預設觀察 10 秒）：

1. **BLE** 群組（約 20 秒）— 驗證 BLE JSON
2. 休息 3 秒
3. **CAN** 群組（約 39 秒）— 驗證 CAN bus 速度指令
4. 休息 3 秒
5. **Buzzer** 群組（約 12 秒）— 驗證 Buzzer 嗶聲節奏
6. 送出 TEST\_ALL\_DONE，自動退出測試模式

進度訊息：

回傳訊息

---

TEST\_ALL\_START

TEST\_ALL\_BLE\_GROUP\_START

TEST\_ALL\_BLE\_GROUP\_DONE

TEST\_ALL\_CAN\_GROUP\_START

TEST\_ALL\_CAN\_GROUP\_DONE

TEST\_ALL\_BUZZER\_GROUP\_START

TEST\_ALL\_BUZZER\_GROUP\_DONE

TEST\_ALL\_DONE

## 七、錯誤回覆說明

回傳訊息

---

TEST\_ERR NOT\_IN\_TEST\_MODE

TEST\_ERR ALL\_ALREADY\_RUNNING

TEST\_ERR UNKNOWN\_CMD

TEST\_ACK ALREADY\_IN\_TEST

TEST\_TIMEOUT\_EXIT

## 八、快速測試流程範例

### 範例 A：只測試藍芽 JSON

test\_enter

test\_ble collision 1

( 確認收到 collision\_warning JSON )

test\_ble collision 0

test\_exit

### 範例 B：只測試 CAN 限速

test\_enter

test\_can collision 1

( 踩油門 - 確認轉速被限制 + 確認 ACK speed=10 )

test\_can collision 0

( 確認轉速恢復正常 )

test\_exit

### 範例 C：只測試 Buzzer

test\_enter

test\_buzzer collision

( 聽到急促嗶聲 )

test\_buzzer off

( 靜音 )

test\_exit

## 範例 D : 全自動測試 ( 觀察 15 秒 )

test\_enter

test\_all 15

( 等待約 92 秒 , 觀察各測項 )

( 收到 TEST\_ALL\_DONE 後自動退出測試模式 )

---

## 附錄 : 指令速查表

`` test\_enter test\_exit

test\_ble violation level = 0 或 1 test\_ble collision level = 0 或 1 test\_ble alert test\_ble\_all

test\_can violation level = 0 或 1 test\_can collision level = 0 或 1 test\_can alert level = 0 或 1

test\_can\_all [觀察秒數] 預設 10 秒

test\_buzzer grass test\_buzzer alert test\_buzzer collision test\_buzzer off test\_buzzer\_all

te