

SparkNavi

向成電子有限公司 XIANG CHENG Electronic Co., Ltd.



InertiaX core

IMU600-V2

通訊協定說明

Communication Protocol Specification

使用者版

資料格式、指令與系統整合說明

對應韌體	Firmware 0.11.2
發行日期	2026 年 10 月 16 日
文件版號	1.1

XIANG CHENG Electronic Co., Ltd. (向成電子有限公司)

No. 41, Wenxian 1st Rd., North Dist., Tainan City 704, Taiwan (R.O.C.)

Tel +886-6-3586207 Fax +886-6-3581363

ethan@robotlab.com.tw · sparknavi.com

© 2026 XIANG CHENG Electronic Co., Ltd. All rights reserved.

目錄

1. 基本規則	4
2. 資料輸出	4
2.1 \$SPNVIMURAW — 單顆 IMU 資料 (每批 3 行)	5
2.2 \$SPNVIMU — 三顆 IMU 的合併資料 (每批 1 行)	7
2.3 \$SPNVBARO — 氣壓計資料	8
2.4 \$SPNVAHRS — 姿態資料	9
2.5 輸出模式與頻寬	10
3. 狀態與統計資料	10
3.1 \$SPNVSTAT — 每顆 IMU 統計 (每顆一行)	11
3.2 \$SPNVFUS — 合併統計	13
3.3 \$SPNVSYS — 系統統計	13
3.4 \$SPNVHB — 診斷心跳	14
4. 開機資訊與事件訊息	14
5. 操作指令 (主機送給裝置)	16
5.1 校正流程 (fw 0.5.0)	20
6. 典型輸出序列	20
7. 校驗計算範例 (Python)	21
8. Binary 模式 (D6 44)	21
8.1 封包結構	21
8.2 訊息表	22
8.3 二進位輸出的資料量與頻寬	25
8.4 鮑率設定與實際誤差	26
8.5 主機端工具	26
9. 韌體更新	26
10. GPS PPS 時間對齊 (fw 0.8.0)	26
11. UART2 第二輸出埠 (fw 0.8.0)	27
12. EXT_TRIGGER 外部觸發 (fw 0.9.0)	28
13. SPI3 從機輸出 (fw 0.10.0 起)	29
13.1 SPI 封包結構 (128 bytes · little-endian)	29
13.2 飛控讀取建議	30
13.3 實機驗證結果 (Blue2 + ArduPilot 4.7.1 · 2026-10-15)	31
13.4 \$SPNVSPI 欄位	31
14. ArduPilot 4.7.1 驅動 (AP_InertialSensor_SparkNavi)	31

修訂紀錄

版號	日期	內容	編輯者
1.0	2026-10-16	建立使用者版通訊協定文件，對應韌體 0.11.2；新增封面、目錄與修訂紀錄。	Ethan
1.1	2026-10-16	修飾全文說明，統一用語並拆解長句；釐清時間戳、狀態旗標及設定保存方式，更新目錄。	—

SparkNavi-IMU600-V2 \$SPNV* 通訊協定說明 (使用者版，fw 0.11.2，2026-10-16)

本文件說明 IMU600-V2 的資料格式、操作指令，以及與主機或飛控連接時需要注意的事項。對應韌體版本為 0.11.2。

1. 基本規則

第 2–7 節介紹 ASCII 文字格式，第 8 節介紹 D6 44 二進位格式 (binary)。兩種格式的對應欄位代表相同資料，實際排列方式請看各節欄位表。可用 \$SPNVCMD,MODE,binary 切換輸出格式；主機送給裝置的指令一律使用 ASCII 文字。

項目	規格
介面	UART1，預設 460800 bps；8 個資料位元、無同位檢查、1 個停止位元 (8N1)，無流量控制。
行格式	\$<body>*<XX>\r\n
校驗 xx	將 \$ 與 * 之間的所有字元逐位元組做 XOR 運算，再寫成 2 位大寫十六進位數字。計算方式與 NMEA 0183 相同。
分隔	各欄位用逗號，分隔。body 的第一欄是訊息名稱，名稱都以 SPNV 開頭。
數值	數值以十進位文字表示。浮點數保留固定的小數位數 (見各欄位表)，負數前加 -，整數不帶小數點。
單位	角速度 °/s、比力 m/s ² 、溫度 °C、 $\Delta\theta$ rad、 Δv m/s、氣壓 hPa (= mbar，同 ArduPilot)、時間 μ s 或 s
座標	採用機體 FRD 座標：+X 向前、+Y 向右、+Z 向下。裝置水平靜置時，az 約為 -9.79 m/s ² 。
時間戳 ts_us	從開機開始計時，開機時為 0，單位為微秒 (μ s)，以 64-bit 無號整數表示。對 IMU 資料而言，ts_us 是這一筆資料所涵蓋時段的結束時間，不是資料送到主機的時間。同一批三顆 IMU 的 RAW 資料與 FUSED 合併資料，ts_us 完全相同。氣壓計的時間定義另見第 2.3 節。
行長	每行少於 320 字元。傳送緩衝區滿時會丟棄整行，不會只送半行；累計丟棄數可查看 \$SPNVSYS 第 1 欄。
解析建議	先找 \$ 作為一行的起點，再以 * 分開資料與校驗碼。校驗失敗就丟棄整行；校驗通過後，依第一欄的訊息名稱處理。新版韌體可能在行尾新增欄位，解析程式應允許多出的欄位。

2. 資料輸出

連續模式依 RATE 定時輸出 IMU 資料，預設為 50 Hz，也就是每 20 ms 一批。本文的「輸出週期」指每批資料所涵蓋的取樣時段，原文件也稱為時間格或視窗。RAW 是三顆 IMU 各自的資料 (已套用校正)；FUSED 是將可用 IMU 資料合併後的結果。外部觸發模式的第一個 EXT_TRIGGER 上升緣只建立起點，之後每個上升緣產生一批，見第 12 節。

2.1 \$SPNVIMURAW — 單顆 IMU 資料 (每批 3 行)

```
$SPNVIMURAW,<id>,<ts_us>,<gx>,<gy>,<gz>,<ax>,<ay>,<az>,<temp>,<n>,<status>,<dthx>,<dthy>,<dthz>,<dvx>,<dvz>,<dvz>,<dt>*XX
```

#	欄位	型別 / 小數	單位	說明
1	id	整數	–	IMU 編號 1、2、3
2	ts_us	整數 u64	μs	這筆資料所涵蓋時段的結束時間。
3–5	gx gy gz	浮點 4 位	°/s	將本次輸出週期內的原始取樣值 (1475 Hz) 演算處理後，得到三軸角速度。
6–8	ax ay az	浮點 4 位	m/s ²	本次輸出週期內的平均比力，即加速度計量到的值，包含重力的影響。
9	temp	浮點 2 位	°C	這顆 IMU 的內部溫度。
10	n	整數	–	這筆資料包含的原始樣本數。100 Hz 時約 14–15 筆，50 Hz 時約 29–30 筆，5 Hz 時約 295 筆。
11	status	整數 (bit)	–	記錄本次輸出週期內曾出現的狀態。所有樣本的狀態旗標會逐位元合併 (OR)；任一次出現的旗標都會保留。0 表示未偵測到錯誤，也沒有 PPS 標記。見第 2.1.1 節。
12–14	dthx dthy dthz	浮點 7 位	rad	這段時間累積的旋轉向量 $\Delta\theta$ ，採梯形積分，並加入二階圓錐運動 (coning) 補償。
15–17	dvx dvz dvz	浮點 6 位	m/s	這段時間累積的速度變化量 Δv ，包含旋轉與划槳運動 (sculling) 補償。靜置時 $dvz \approx -9.79 \times dt$ 。
18	dt	浮點 6 位	s	這筆資料實際積分的時間長度，約為 1/輸出率。以 dth/dt、dv/dt 可換算出對應的平均角速度與比力。

2.1.1 如何判讀 status

status 記錄的是「這一段取樣時間內，曾經出現哪些狀態」。每個原始樣本要讀取 7 個 SPI 回應封包 (frame)；50 Hz 輸出時，一筆資料約包含 30 個樣本、共約 210 個回應封包。韌體會把它們的狀態逐位元合併 (OR)：只要其中一次出現某個旗標，輸出的對應位元就會是 1。

因此，status 反映整段時間的紀錄，不只代表最後一個樣本。status = 0 表示這段時間內未偵測到 CRC 或位址錯誤，感測器均回報正常狀態 S[1:0] = 00，也沒有 CE、IDS 或 PPS 旗標。

位元	值	名稱	來源	意義	資料可信度	常見原因 / 處置
bit0	1	SPI_CRC	裝置端檢查	48-bit SPI 回應封包的 CRC-8 校驗失敗，或回應的暫存器位址與請求不符。	受影響的暫存器值以 0 代入，會使平均值偏離實際值。	檢查線路雜訊、SCK 速度及 CS 時序；\$SPNVSTAT 的 crc_err 也會增加。正常應為 0。
bit1	2	CE	感測器 frame CE 位	命令錯誤 (Command Error)：感測器收到無效請求，例如寫入唯讀暫存器、使用未定義位址，或在 EOI 後寫入。	資料本身仍有效	正常操作不應出現。若出現，請回報原廠檢查韌體指令。
bit2	4	IDS	感測器 frame IDS 位	內部資料狀態 (Internal Data Status)：感測器回報內部錯誤，與 S=01 表示的錯誤狀態相互對應。	可疑	通常與 bit3 同時出現
bit3	8	S_ERROR	感測器 frame S[1:0] = 01	感測器自我診斷發現異常，可能涉及電源、時脈、溫度或自我測試。	不可信	若持續出現，先用 \$SPNVCM,REINIT,1 重新初始化；仍未恢復時，檢查該顆 IMU 的硬體。
bit4	16	S_SAT	感測器 frame S[1:0] = 10	超出量測範圍 (飽和)：任一軸角速度超過 $\pm 300^\circ/\text{s}$ ，或加速度超過 $\pm 156.9 \text{ m/s}^2$ 。感測器內部濾波前的瞬間峰值也會觸發。	超出量程的部分無法完整量到，可能使 $\Delta\theta/\Delta v$ 偏小。	常見於劇烈運動、撞擊或振動過大。正常靜置或一般飛行時不應出現。
bit5	32	S_INIT	感測器 frame S[1:0] = 11	感測器尚未完成初始化，通常出現在上電或重置後約 250 ms 內。	不可信	只應在開機或 REINIT 後的前幾筆出現
bit7	128	PPS	(fw 0.8.0) 裝置	PPS 時間標記：這筆 ts_us 對應 GPS PPS 的上升緣。對齊期間每秒有一筆，見第 10 節。	此位元本身不代表錯誤。	判斷錯誤時，請先以 status & 0x7F 排除 PPS 位元，再檢查結果是否為 0。

- 在單一 SPI 回應封包中，S[1:0] 一次只能表示一種狀態，因此對應的 bit3、bit4、bit5 最多只有一個為 1。但一筆輸出會合併多個回應封包；若這段時間先後出現不同狀態，輸出的 status 就可能同時包含這幾個位元。
- 感測器同時遇到多種狀況時，回報優先順序為：初始化 11 > 錯誤 01 > 飽和 10 > 正常 00。例如，同時發生錯誤與飽和時，該次回應只回報錯誤。

- 合併資料時，先檢查 (status & 0x7F) 是否為 0。若不為 0，該顆 IMU 的這筆資料會被排除，並在 \$SPNVIMU 的 flags 標示。bit7 的 PPS 標記不影響採用，因此 status = 128 不代表資料有錯。三顆都出現錯誤旗標時的處理方式，見第 3.1.1 節。
- 常見數值：0 = 正常；1 = SPI 校驗或位址問題；16 = 超出量程；12 (8+4) = 感測器錯誤；32 或 44 = 開機初始化期間的狀態；128 = 只有 PPS 標記。

三顆 IMU 的取樣時脈可能略有差異，例如 1470 Hz 與 1461 Hz。因此，即使它們使用相同的輸出時間邊界，每筆包含的樣本數 n 仍可能差 1。原始樣本所涵蓋的時間也可能略有差異，例如一顆以 30 個樣本涵蓋 20.4 ms，另一顆以 29 個樣本涵蓋 19.8 ms。

以下為 50 Hz 輸出、裝置靜置時的格式範例：

```
$SPNVIMURAW,1,12345680,0.0123,-0.0456,0.0089,0.0210,-0.0150,-9.7912,31.25,30,0,0.0000043,-0.0000159,0.000031,0.000420,-0.000300,-0.195824,0.020000*0F
```

2.2 \$SPNVIMU — 三顆 IMU 的合併資料 (每批 1 行)

```
$SPNVIMU,<ts_us>,<gx>,<gy>,<gz>,<ax>,<ay>,<az>,<temp>,<dthx>,<dthy>,<dthz>,<dvx>,<dvy>,<dvz>,<dt>,<n_used>,<flags>*XX
```

#	欄位	型別 / 小數	單位	說明
1	ts_us	整數 u64	μs	與同一批 RAW 資料的 ts_us 相同。
2–4	gx gy gz	浮點 4 位	°/s	合併後角速度
5–7	ax ay az	浮點 4 位	m/s ²	合併後比力
8	temp	浮點 2 位	°C	實際參與合併的 IMU 溫度平均值。
9–11	dthx dthy dthz	浮點 7 位	rad	合併後 Δθ
12–14	dvx dvz dvz	浮點 6 位	m/s	合併後 Δv
15	dt	浮點 6 位	s	實際參與合併的 IMU 之 dt 平均值。
16	n_used	整數	–	這筆合併結果實際使用的 IMU 顆數，範圍為 1–3。
17	flags	整數 (bit)	–	合併狀態，見下表。0 表示三顆都正常且都參與合併，也沒有 PPS 標記。

flags 的各位元代表以下狀態；多種狀態可同時出現：

位元	值	名稱	意義
bit0	1	IMU1_REJECTED	本次未採用 IMU1。原因為 (status & 0x7F) ≠ 0，或量測值持續偏離比較基準、達到排除條件。
bit1	2	IMU2_REJECTED	本次未採用 IMU2，排除條件同 IMU1。
bit2	4	IMU3_REJECTED	本次未採用 IMU3，排除條件同 IMU1。
bit3	8	SPREAD_WARN	某顆 IMU 與比較基準的差距超過有效門檻的一半。這是提醒旗標；也包含已超過完整門檻、但持續時間尚不足以排除的情況。
bit4	16	DEGRADED	可用 IMU 少於 2 顆，這筆結果只使用 1 顆 IMU。
bit5	32	INCOMPLETE	本次未收到所有已就緒 IMU 的資料。

位元	值	名稱	意義
bit6	64	SATURATED	(fw 0.5.0) 至少一顆 IMU 超出量程，出現 S_SAT。超出量程的角速度及旋轉量無法完整量到；接收端的慣性導航系統 (INS) 應降低這段資料的權重，或提高估計誤差。
bit7	128	PPS	(fw 0.8.0) 這筆 ts_us 對應 GPS PPS 上升緣，見第 10 節。此位元是時間標記，不代表錯誤。

2.3 \$SPNVBARO — 氣壓計資料

氣壓計預設每秒輸出 50 筆，可用 BRATE 修改。其量測時機與 IMU 的時間戳定義不同，說明如下。

fw 0.5.0 起，氣壓計預設使用 Forced mode：由裝置按固定週期啟動一次量測，輸出率可設為 10、25、50 或 100 Hz。它與 IMU 使用相同時鐘及開機起點；50 Hz 時，每隔 20000 μ s 啟動一次，ts_us 也落在這些時間點上。

氣壓計的 ts_us 代表「觸發量測的時間」。量測在觸發後才開始，約經過一個轉換時間才完成；可查看 \$SPNVINIT,B 的 conv_us。因此，BARO 與 IMU 即使 ts_us 相同，代表的量測時段也不同。

韌體會依輸出率自動選擇過取樣倍率 (OSR)，讓轉換時間不超過輸出週期減 3 ms：10/25 Hz 使用 16x/2x、50 Hz 使用 8x/2x、100 Hz 使用 4x/1x。Forced mode 仍使用係數為 3 的 IIR 濾波。

```
$SPNVBARO,<ts_us>,<press_hpa>,<temp_c>,<alt_m>,<status>*XX
```

#	欄位	型別 / 小數	單位	說明
1	ts_us	整數 u64	μ s	Forced mode：啟動這次量測的時間。Normal mode：裝置偵測到新資料已準備好的時間。
2	press_hpa	浮點 4 位	hPa (= mbar)	濾波後的氣壓值，IIR 係數為 3。單位與 ArduPilot 的 SCALED_PRESSURE.press_abs 相同，乘以 100 可換成 Pa。輸出解析度為 0.0001 hPa (0.01 Pa)；目前 OSR 可查看 \$SPNVINIT,B 的 osr_p。
3	temp_c	浮點 2 位	°C	氣壓計晶片的內部溫度，較接近板上溫度，不是外界氣溫。
4	alt_m	浮點 2 位	m	依標準大氣公式 $44330.77 \times (1 - (p/1013.25)^{0.190263})$ 換算的高度，適合觀察高度變化或繪圖。數值會受天氣影響，不宜直接當作準確海拔；氣壓較高時也可能出現負值。
5	status	整數 (bit)	—	0 = 正常；bit0 (值為 1) = 取得這筆資料前曾遇到 I2C NACK 或匯流排衝突，但重試後已成功。

範例：

```
$SPNVBARO,9651290,1017.3040,44.14,-33.69,0*XX
```

用 \$SPNVCMD,BARO,0|1 關閉或開啟氣壓資料輸出；用 \$SPNVCMD,BRATE,10|25|50|100 設定輸出率，預設為 50 Hz。要檢查氣壓計狀態，可查看 \$SPNVSYS 的 baro_samples、baro_i2c_err、baro_timeouts、baro_stale；初始化結果則見 \$SPNVINIT,B (第 4 節)。

2.4 \$SPNVAHRS — 姿態資料

此訊息提供姿態角、四元數及陀螺零偏估計，與同一批合併 IMU 資料使用相同的 ts_us。

```
$SPNVAHRS,<ts_us>,<roll>,<pitch>,<yaw>,<q0>,<q1>,<q2>,<q3>,<align>,<flags>,<bgx>,<bgx>,<bgz>,<zb_s>*XX
```

#	欄位	型別 / 小數	單位	說明
1	ts_us	整數 u64	μs	與同一批 \$SPNVIMU 資料的 ts_us 相同。
2	roll	浮點 3 位	deg	滾轉角，向右傾斜為正，範圍 -180..180。使用 FRD/NED 座標與 ZYX 歐拉角順序。
3	pitch	浮點 3 位	deg	俯仰角，抬頭為正，範圍 -90..90。
4	yaw	浮點 3 位	deg	航向角，由上往下看，順時鐘為正，範圍 0..360。align=1 時為相對角度，從 0 起算，會隨時間產生漂移。
5-8	q0 q1 q2 q3	浮點 6 位	—	將機體座標轉到 NED (北、東、下) 座標的四元數，採 Hamilton 定義，q0 為純量部分： $v_n = R(q) \cdot v_b$ 。
9	align	整數	—	對準狀態，見下表
10	flags	整數 (bit)	—	見下表
11-13	bgx bgy bgz	浮點 4 位	°/s	運作中估計的三軸陀螺零偏，使用機體座標，供診斷使用。
14	zb_s	浮點 1 位	s	(fw 0.9.2) 目前已連續靜止的秒數，範圍 0-10。有移動就歸零，滿 10 s 後套用零偏估計，並標示 BIAS_CAL。AZB,0 時固定為 0。GUI 可顯示「校正中 x / 10 s，請保持靜止」。

align 表示姿態與航向的對準進度：

值	名稱	GUI 顯示建議	意義
0	INIT	"Aligning..."	開機或執行 AHRS,reset 後的前 1 s。裝置收集靜置資料以判斷水平姿態，此時 roll、pitch、yaw 尚不可用。
1	LEVEL ED	"Heading: NOT ALIGNED"	roll 與 pitch 已可使用。航向尚無外部參考，yaw 從 0 起算，靠陀螺累積角度，因此會漂移。
2	ALIG NED	"Heading: aligned (manual)"	已設定航向參考，來源由 flags 表示；目前僅支援透過 HDG 指令手動設定。

flags 表示姿態計算的目前狀態：

位元	值	名稱	意義
bit0	1	HIGH_DYN	偵測到較大動態：比力大小 $ f $ 與 g 的差超過 0.5 m/s^2 、 $ \omega $ 超過 $50^\circ/\text{s}$ ，或量測重力方向與估計方向相差超過 5° 。此時 K_p 降為 0.2，減少加速度計對姿態修正的影響。
bit1	2	BIAS_FROZEN	暫停更新陀螺零偏。觸發條件為動態狀態、 $ \omega > 5^\circ/\text{s}$ ，或 $ f -g > 0.2 \text{ m/s}^2$ 。此時姿態仍持續更新，只是暫停學習零偏。
bit2	4	HDG_MANUAL	目前航向由 \$SPNVCMD,HDG 手動設定。

位元	值	名稱	意義
bit3	8	ACC_REJECT	比力大小 $ f $ 與 g 的差超過 3 m/s^2 。這次姿態更新只使用陀螺，不使用加速度計修正。
bit4	16	INIT_MOVING	初始對準的 1 s 內偵測到移動，因此未取得零偏，水平姿態的準確度也會較差。請將裝置靜置後，再執行 <code>AHRS,reset</code> 。
bit5	32	BIAS_CAL	(fw 0.9.1) 已利用至少 10 s 的靜止資料估計三軸陀螺零偏，包含 Z 軸。這是完成狀態，不是警告；航向漂移規格以完成此步驟後為準。
bit6	64	STATIC	(fw 0.9.1) 最近 10 s 判定為靜止，正在更新零偏估計。

靜止零偏估計：fw 0.9.1 起提供此功能，出廠預設開啟，可用 `$SPNVCMD,AZB,0|1` 控制。裝置每 10 s 檢查一次靜止條件：各軸陀螺標準差 $< 0.1^\circ/\text{s}$ 、加速度判定值 $< 0.1 \text{ m/s}^2$ ，且沒有動態旗標。通過判定後，會用這段時間的陀螺平均值估計零偏。第一次估計會直接套用，完成時 flags 出現 `BIAS_CAL`。

姿態計算方式：以 `$SPNVIMU` 的 $\Delta\theta$ 更新四元數，並以 $\Delta v/dt$ 提供重力方向的觀測值。

相關指令為 `AHRS 0|1|reset`、`HDG <deg>` 與 `AKP/AKI`。ASCII 姿態資料每行約 120 B。

2.5 輸出模式與頻寬

用 `$SPNVCMD,OUT,raw|fused|both|none` 選擇輸出的 IMU 資料，預設為 `both`。UART1 設為 460800 bps 時，每秒約可傳送 46 kB；各種設定的資料量如下。

設定	流量	結果
50 Hz both (預設) + BARO 50 Hz + AHRS 50 Hz	$\approx 38 \text{ kB/s}$	OK
100 Hz fused + AHRS	$\approx 26 \text{ kB/s}$	頻寬足夠，建議正式使用。
100 Hz raw	$\approx 45 \text{ kB/s}$	接近傳輸上限，偶爾會丟棄資料行。
100 Hz both	$\approx 59 \text{ kB/s}$	超出傳輸上限，會丟棄資料行。

200、400、500、1000 Hz 的 binary 資料量見第 8.3 節，其中 200/400/500 Hz 自 fw 0.10.3 起支援。

在這些較高輸出率下，460800 bps 無法負荷 ASCII 資料，請改用 binary 並提高鮑率。若只透過 SPI3 連接飛控，可設 `OUT,none` 關閉 UART 的 IMU 資料輸出，見第 13 節。

3. 狀態與統計資料

預設每秒提供一組統計，包括 3 行 `STAT`、1 行 `FUS` 與 1 行 `SYS`。fw 0.4.3 起，裝置先在同一時間記錄全部數值，再於後續主迴圈逐行送出，順序為 `STAT1`、`STAT2`、`STAT3`、`FUS`、`SYS`、`HB`。雖然傳送時間不同，這組統計仍代表同一個時間點。

3.1 \$SPNVSTAT — 每顆 IMU 統計 (每顆一行)

```
$SPNVSTAT,<id>,<dry>,<out>,<crc_err>,<frame_err>,<spi_timeout>,<dropped>,<dcnt_skip>,<dry_timeout>,<dry_kicks>,<dry_pin>,<isr_us_max>,<isr_us_last>,<isr_cpu>,<resync>[,<read_retry>,<read_bad>]*XX
```

#	欄位	說明	正常值
1	id	IMU 編號	1~3
2	dry	從開機起累計的 DRY (資料就緒) 中斷次數，對應原始樣本數。	每秒增加 ≈ 1470
3	out	從開機起累計的資料輸出筆數。	每秒增加 = 輸出率
4	crc_err	SPI 回應 CRC 錯誤累計	0
5	frame_err	SPI 回應中帶有感測器狀態旗標的封包數，旗標包括 S1:S0≠00、CE 或 IDS。此欄統計感測器回報的狀態；通訊問題則看 crc_err、spi_timeout。見第 3.1.1 節。	0 (正常使用下不增加)
6	spi_timeout	SPI 通訊逾時的累計次數。fw 0.10.6 的逾時設定為 100 μs，交換資料期間會暫時遮蔽中斷。	0
7	dropped	因緩衝區或輸出佇列已滿而丟棄的累計數。	0
8	dcnt_skip	感測器樣本計數器不連續的累計次數，表示可能漏讀或重複讀取。	0
9	dry_timeout	目前是否已超過 100 ms 未收到 DRY：0 = 否，1 = 是。	0
10	dry_kicks	為解除 DRY 腳持續維持高電位而進行的額外讀取次數。	開機後固定 1
11	dry_pin	DRY 腳目前的電位：0 = 低，1 = 高。	多為 0
12	isr_us_max	DRY 中斷處理程式 (ISR) 最長執行時間，單位 μs。	約 100 μs 以內
13	isr_us_last	最近一次 DRY 中斷處理的執行時間，單位 μs。	同上
14	isr_cpu	最近 1 s 內，這顆 IMU 的 DRY 中斷處理占用 CPU 的百分比，以整數表示。	約 13
15	resync	重新對齊輸出週期時間邊界的累計次數。	0
16	read_retry	(fw 0.10.6，僅 ASCII) 第一次讀取遇到 CRC 錯誤後，立即重讀的累計次數。	0
17	read_bad	(fw 0.10.6，僅 ASCII) 重讀仍失敗，因而丟棄的原始樣本數。	0

3.1.1 frame_err 增加時如何排查

GUI 將 frame_err 顯示為「感測器旗標」。它記錄 IMU 晶片在 SPI 回應中回報的狀態，不直接表示線路品質。要檢查通訊是否異常，請看 crc_err、spi_timeout，或 GUI 的「SPI 錯誤」。

frame_err 的計數單位是「帶有旗標的 SPI 回應封包」，不是異常事件次數。讀取一個原始樣本需要 7 個封包，分別取得三軸角速度、三軸加速度及溫度。感測器出現異常時，這 7 個封包通常都會帶有旗標，因此一次樣本可能讓計數增加 7。下列實測中，約 5 ms 的劇烈甩動使計數增加 52，不代表發生了 52 次獨立事件。

在未超出量程、也沒有感測器異常的一般運動、振動與溫度變化下，這個計數不應持續增加。

板上測試紀錄 (fw 0.4.3 · 2026-10-08) :

測試	SPI 錯誤 (crc_err/spi_timeout)	感測器旗標 (frame_err)
靜置	0 · 不累加	0 · 不累加
緩和搖晃 (未超過量程)	0 · 不累加	0 · 不累加
劇烈甩動 (角速度峰值 > 300 °/s)	0 · 不累加	三顆同時 +52 (飽和)

若計數增加，可依下表檢查原因：

原因	對應 status 位元	何時發生	三顆同時？	資料可信度	處置
量程飽和	S_SAT (16)	任一軸超出量程：角速度超過 ± 300 °/s，或加速度超過 ± 156.9 m/s ² 。判定包含內部濾波前的訊號，因此即使輸出平均值未超標，瞬間峰值仍可能觸發。	通常會，因為三顆承受相同運動。	超出量程的部分無法完整量到，可能使 $\Delta\theta/\Delta v$ 偏小。	撞擊、掉落或劇烈甩動時，可能因超出量程而觸發，這不一定是硬體故障，但受影響的資料仍須留意。若一般運轉時持續增加，請檢查馬達、螺旋槳或引擎的高頻振動及減振安裝。若角速度經常超過 300 °/s，應評估較大量程的陀螺。
初始化中	S_INIT (32)	上電、\$SPNVCM,REINIT,1 後約 250 ms 內	是	不可信	若只在開機後短暫增加，之後固定不變，通常屬正常初始化。運轉中再次出現時，請檢查是否因電源壓降造成感測器重置。
感測器自我診斷錯誤	S_ERROR (8) · 常伴隨 IDS (4)	感測器內部診斷發現供電電壓、時脈、溫度或自我測試異常。	三顆同時出現：先查共用電源與溫度。只有一顆出現：先查該顆元件。	不可信	檢查 3.3 V / VDDIO 的電源雜訊、壓降及工作溫度。執行 REINIT,1 後若仍持續出現，請檢查該顆 IMU 的焊接與硬體狀況。
命令錯誤	CE (2)	感測器收到無效請求，例如寫入唯讀暫存器、使用未定義位址，或在 EOI 後寫入。	—	資料仍有效	正常操作不應出現；請回報原廠檢查韌體。

排查步驟

1. 先看 GUI 的「SPI 錯誤」。若它也持續增加，請先檢查通訊線路及時序。
2. 再看是三顆一起增加，還是只有一顆增加。三顆一起出現時，先查運動、振動、共用電源與溫度。

3. 設定 OUT,raw，查看 \$SPNVIMURAW 的 status，確認是 16（飽和）、32（初始化）、或 8/12（感測器錯誤）。位元定義見第 2.1.1 節。
4. 合併時，有錯誤旗標的 IMU 會被排除，判斷式為 (status & 0x7F) ≠ 0；PPS 標記不算錯誤。若三顆同時出現錯誤旗標，韌體會退回使用三顆資料產生合併輸出。此時不能只因為仍有合併資料，就認定量測正常；請同時查看 RAW 的 status。

3.2 \$SPNVFUS — 合併統計

```
$SPNVFUS,<outputs>,<rej1>,<rej2>,<rej3>,<incomplete>,<degraded>,<gth>,<ath>*XX
```

#	欄位	說明
1	outputs	合併輸出累計筆數
2–4	rej1 rej2 rej3	IMU1、IMU2、IMU3 各自被排除的累計輸出週期數。原因可能是錯誤旗標，或量測偏差已達持續排除條件；PPS 標記不列入。正常使用，包括一般手持擺動，不應增加。
5	incomplete	尚未收齊三顆 IMU 資料就產生合併結果的累計次數。
6	degraded	可用 IMU 少於 2 顆的累計次數。
7	gth	目前設定的陀螺排除門檻，單位 °/s，保留 3 位小數。
8	ath	目前設定的加速度排除門檻，單位 m/s ² ，保留 3 位小數。

3.3 \$SPNVSYS — 系統統計

```
$SPNVSYS,<uart_drop>,<loops_per_s>,<isr_cpu_total>,<rate_hz>,<out_mode>,<uart_resets>,<pps_cnt>,<baro_samples>,<baro_i2c_err>,<baro_timeouts>,<baro_stale>,<baro_hz>,<proto_mode>,<baud>,<pps_state>,<u2_drop>,<trig_cnt>,<trig_rej>*XX
```

#	欄位	說明
1	uart_drop	傳送緩衝區已滿，因而丟棄整行資料的累計數。持續增加通常表示輸出資料量超過 UART 頻寬。
2	loops_per_s	最近 1 s 內主迴圈的執行次數，約 77000 次。
3	isr_cpu_total	三顆 IMU 中斷處理合計占用 CPU 的百分比，以整數表示，約為 40%。
4	rate_hz	目前輸出率
5	out_mode	none / raw / fused / both
6	uart_resets	UART 傳送停止回應後，自動重置以恢復運作的次數，正常應為 0。
7	pps_cnt	PPS 上升緣的原始擷取次數，包含後續被排除的雜訊。通過條件檢查的次數請看 \$SPNVPPS 的 edges。
8	baro_samples	成功讀到的氣壓樣本累計數；預設 50 Hz 時，每秒約增加 50。
9	baro_i2c_err	I2C 未收到確認（NACK）或匯流排衝突的累計次數，正常應為 0。
10	baro_timeouts	因 I2C 通訊逾時而重新初始化模組的累計次數，正常應為 0。
11	baro_stale	目前是否已超過 200 ms 沒有新氣壓資料：0 = 否，1 = 是。

#	欄位	說明
12	baro_hz	氣壓計目前輸出率
13	proto_mode	ascii / binary
14	baud	目前 UART1 鮑率
15	pps_state	(fw 0.8.0) 0 OFF / 1 NO_SIGNAL / 2 ACQUIRING / 3 LOCKED / 4 HOLDOVER (\$10)
16	u2_drop	(fw 0.8.0) UART2 傳送緩衝區滿，因而丟棄整筆資料的累計數。
17	trig_cnt	(fw 0.9.0) 外部觸發模式下，已接受的 EXT_TRIGGER 上升緣累計數。連續輸出模式不累計。
18	trig_rej	(fw 0.9.0) 被排除的 EXT_TRIGGER 邊緣累計數。原因包括與前一次間隔小於 0.5 ms (視為毛刺或彈跳)，或主迴圈忙碌造成觸發事件緩衝區溢位。

3.4 \$SPNVHB — 診斷心跳

此訊息預設每秒輸出一次，以 key=value 格式提供原廠診斷資訊。一般應用的解析程式可直接略過；不需要時，用 \$SPNVCMD,HB,0 關閉。

4. 開機資訊與事件訊息

出廠預設設定下，裝置開機後直接輸出資料，不主動輸出開機資訊。需要查看資訊時，可送出 \$SPNVCMD,INFO,1 取得精簡內容，或 \$SPNVCMD,DBG,1 取得完整內容。

例外情況會主動回報：裝置驗證失敗時送出 \$SPNVHALT，IMU 或氣壓計初始化失敗時送出 \$SPNVINIT,n,FAIL,...。若已保存不同的 BAUD 或 MODE，開機時也會先送出救援提示，詳見下表 \$SPNVNVM,boot。

訊息	格式	說明
\$SPNVBOOT	\$SPNVBOOT,SparkNavi-IMU600-V2,fw=<ver>,imus=3,frame=body-FRD,proto=ascii-xor8 d644-crc16,baud=<n>,build=rel=<0/1>,bl=<0/1>,auth=<0/1>,halt=<0/1>,bind=<0/1>*XX	產品名稱、韌體版本、座標系、通訊格式與鮑率。fw 0.7.0 起增加建置旗標：rel = 出貨版、bl = 含 bootloader，auth / halt / bind 與裝置驗證有關；出貨版皆為 1。請以 key=value 解析，例如 build=rel=1 的 key 是 build，值是 rel=1。新版可能在尾端新增欄位。
\$SPNVRESET	\$SPNVRESET,rcon=0x<8hex>[,POR][,BOR][,WDTO][,SWR][,EXTR(MCLR)][,CM(config mismatch)]...*XX	上次重置原因，由 RCON 暫存器解碼。正常上電通常為 POR,BOR；EXTR(MCLR) 表示重置按鍵或燒錄器造成的重置。
\$SPNVDBG	\$SPNVDBG,<自由文字>*XX	只在 DBG 模式輸出，供原廠診斷。一般解析程式可略過。
\$SPNVAUTH / \$SPNVBIND / \$SPNVPUB	\$SPNVAUTH,auth=<0/1>,sn=<18hex>*XX ; \$SPNVBIND、\$SPNVPUB 為 key=value 行	裝置驗證資訊，執行 INFO 時也會回報。auth=1、ok=1 表示正常；sn 是 18 位十六進位裝置序號，可作為產品序號。其他欄位供原廠診斷，一般應用可略過。

訊息	格式	說明
\$SPNVINIT	精簡：\$SPNVINIT,<id>,OK*XX；完整 (DBG)： \$SPNVINIT,<id>,OK,sn=<str>,...*XX；失敗 (一律印)： \$SPNVINIT,<id>,FAIL,rc=<rc>,crc_err=<n>*XX	各類 IMU 的初始化結果，sn 是該類 IMU 的序號。 FAIL 時可查看 rc：-2 = 無回應 (MISO 全為 0 或 1)、-3 = CRC 錯誤、-4 = 位址錯誤、-5 = 寫入驗證失敗、-6 = 啟動後狀態異常、-7 = 其他錯誤。 完整格式的其他欄位供原廠診斷。
\$SPNVINIT,B	精簡： \$SPNVINIT,B,OK,odr=<hz>,mode=forced normal*XX；完整： \$SPNVINIT,B,OK,chip=0x..,rev=0x..,odr=<hz>,osr_p=<n>,osr_t=<n>,iir=3,i2c_err=0,bus_recover=0,heals=0,mode=forced,conv_us=<估計>/<上一筆實測>/<實測最大>,overruns=<n>*XX；失敗 (一律印)： \$SPNVINIT,B,FAIL,rc=<rc>,chip=0x..,i2c_err=,timeouts=*XX	氣壓計初始化結果。conv_us 的實測值是從觸發到資料就緒 (DRDY) 的時間，包含 1 ms 輪詢間隔的影響。overruns 表示上次量測尚未完成、下次觸發時間就已到達的次數；該次會跳過量測，正常應為 0。 rc：-1 = I2C NACK、-2 = I2C 逾時、-3 = 晶片識別不符、-4 = NVM 錯誤、-5 = OSR/ODR 組合無效。
\$SPNVNVM	\$SPNVNVM,saved=1,baud=,mode=,rate=,out=,stat=,hb=,baro=,brate=,ahrs=,nvm_err=*XX / \$SPNVNVM,saved=0 (factory defaults),nvm_err=*XX / \$SPNVNVM,backend=none*XX	執行 INFO 或 DBG 時，回報非揮發性記憶體 (NVM) 是否存有設定，以及保存的內容。
\$SPNVCAL	\$SPNVCAL,<imu>,ab=<bx>/<by>/<bz>,ak=<kx>/<ky>/<kz>*XX ×3 + \$SPNVCAL,W,g=<w1>/<w2>/<w3>,a=<w1>/<w2>/<w3>,saved=<0/1>,lever=<0/1>*XX	(fw 0.5.0) 回應 CAL,1。INFO/DBG 也會提供。內容包括各類 IMU 六面加速度校正的零偏 (m/s ² ，5 位小數)、比例係數 (6 位小數)、合併權重，以及校正是否已保存至 NVM。lever=1 表示安裝位置差補償正在作用。
\$SPNVCFG	\$SPNVCFG,baud=<n>,mode=ascii binary,rate=<hz>,out=<0-3>,stat=<0/1>,hb=<0/1>,baro=<0/1>,brate=<hz>,ahrs=<0/1>,gth=<f3>,ath=<f3>,akp=<f3>,aki=<f4>,saved=<0/1>,fw=<ver>*XX	回應 \$SPNVCMDCFG,1，列出目前正在使用的設定。out：0 = none、1 = raw、2 = fused、3 = both。saved=1 只表示 NVM 曾保存設定，不表示所有目前設定都已保存。fw 0.9.0 起附加 trig=0 1 (IMU 輸出模式)；fw 0.9.1 起附加 azb=0 1 (靜止零偏估計)。GUI 參數頁利用此訊息讀回裝置設定 (fw 0.4.2 起)。
\$SPNVNVM,boot	\$SPNVNVM,boot,baud=<n>,mode=<m>,factory_window_ms=500*XX	若保存的 BAUD 或 MODE 與出廠值不同，開機會先以 460800 bps、ASCII 格式送出此訊息，並等待 500 ms。此時可送 \$SPNVCMDFACTORY,1 恢復預設；等待結束後才切到保存的格式與鮑率。未保存設定，或保存值與預設相同時，不會送出此行，也不會等待。
\$SPNVHALT	\$SPNVHALT,<原因>*XX，之後每 2~3 s 重複一行	裝置驗證未通過時，IMU、氣壓計與 AHRS 不啟動，也不輸出量測資料；仍接受 INFO、BOOT (更新韌體) 及 FACTORY。若只是暫時讀不到 IMU，例如飛控 5 V 電源波動，裝置約每 3 s 自動重試，恢復後會自動開始輸出。若持續出現，請洽原廠。

訊息	格式	說明
\$SPNVHEAL	\$SPNVHEAL,imu_reinit,mask=<bit0..2 = IMU1..3>,n=<累計次數>*XX	(fw 0.10.1) 自動恢復紀錄。若任一顆 IMU 的 SPI 錯誤 (CRC 加逾時) 連續 3 s 達到每秒 100 次以上, 或某顆初始化失敗, 裝置會重新初始化三顆 IMU, 作用與 REINIT,1 相同。兩次自動初始化至少間隔 30 s。正常使用不應出現; 飛控重開機或 5 V 電源波動後可能觸發。
\$SPNVSTALL	\$SPNVSTALL,main stuck at where=<w>(<name>) for >300ms*XX	DRY 中斷處理偵測到主迴圈超過 300 ms 未繼續執行。正常運作不應出現。
\$SPNVTRAP	\$SPNVTRAP,code=<n>,<name>,INTCON3=0x...,INTCON4=0x...,INTCON5=0x...,halted*XX	CPU 發生位址、堆疊或數學運算等例外錯誤 (trap); 回報後停止運作。正常運作不應出現。

5. 操作指令 (主機送給裝置)

不論資料輸出使用 ASCII 或 binary, 主機送出的指令都必須是 ASCII 格式的 \$SPNVCMD。binary 模式的指令回覆會包在 0x7F TEXT 封包內。

執行指令後, 設定會先套用在 RAM; 若未保存, 重開機後會載入先前保存的設定, 沒有保存紀錄時才使用出廠預設。要保留目前設定, 請送 SAVE,1; 要清除使用者設定並恢復出廠值, 請送 FACTORY,1。

```
$SPNVCMD,<key>,<value>[*XX]\r\n
```

- 可省略校驗碼 *XX, 方便用終端機手動輸入。若有附上校驗碼, 裝置就會檢查; 錯誤時回覆 \$SPNVNAK,CMD,checksum。
- key 與 value 不分大小寫。行尾可用 \r 或 \n; 單行超過 127 字元時, 整行指令會被丟棄。
- 指令成功時回覆 \$SPNVACK,<key>,<value>*XX, value 是實際套用的值; 失敗時回覆 \$SPNVNAK,<key>,<reason>*XX, reason 說明原因。

key	value	作用	ACK 回覆
RATE	5 10 25 50 100 200 400 500 1000	設定 IMU 輸出率, 單位 Hz; 三顆 RAW 與合併資料會一起切換。200/400/500 Hz 自 fw 0.10.3 起支援。連接 ArduPilot 時請設 1000 Hz, 見第 13、14 節。	\$SPNVACK,RATE,100
OUT	raw fused both none	選擇 IMU 資料: raw = 三顆個別資料; fused = 合併資料; both = 兩者皆輸出; none = 關閉這兩類輸出。	\$SPNVACK,OUT,fused
STAT	0 1 now	0 = 關閉每秒統計; 1 = 開啟; now = 立即送出一組 STAT/FUS/SYS。	\$SPNVACK,STAT,1
HB	0 1	診斷心跳 \$SPNVHB 關/開	\$SPNVACK,HB,0
BARO	0 1	\$SPNVBARO 輸出關/開 (預設開)	\$SPNVACK,BARO,1
AHRS	0 1 reset	0/1 = 關閉/開啟姿態資料輸出; reset = 重新進行初始對準, 請保持靜止 1 s。	\$SPNVACK,AHRS,reset

key	value	作用	ACK 回覆
HDG	浮點 deg (-360..360)	手動設定航向 heading，roll 與 pitch 保持不變。成功後 align=2，並設定 HDG_MANUAL 旗標。	\$SPNVACK,HDG,123.5；尚未調平 \$SPNVNAK,HDG,not leveled yet
AKP / AKI	浮點 0.50	設定 AHRS 演算法的 Kp / Ki，預設分別為 2.0 / 0.05。	\$SPNVACK,AKP,1.5
SAVE	1	將目前使用者設定寫入 NVM，供下次開機載入。包含 RATE/OUT/STAT/HB/BARO/BRATE/AHRS/GT H/ATH/AKP/AKI/MODE/BAUD，以及本表後方列出的 UART2、PPS、觸發與 SPI 設定。寫入期間 IMU 會短暫停止，再重新啟動。校正資料另用 CALSAVE 保存。	\$SPNVACK,SAVE,1； \$SPNVNAK,SAVE,nvm not available write failed
FACTORY	1	清除 NVM 的使用者設定，並立即恢復出廠值，包含 MODE/BAUD。fw 0.5.0 起不會清除 ACAL/FWGT 校正資料。ACK 會先用原本的格式與鮑率送出，主機收到後，請切回 460800 bps、ASCII。裝置在 HALT 狀態或開機救援等待期間也接受此指令。	\$SPNVACK,FACTORY,1
CFG	1	回報目前設定，主要訊息為 \$SPNVCFG，見第 4 節。fw 0.4.2 起支援；附加訊息見表後說明。	\$SPNVACK,CFG,1 後接 \$SPNVCFG,...
INFO	1	索取精簡裝置資訊，包括 BOOT、RESET、AUTH 及 4 行 INIT。出廠預設開機不主動輸出這些資訊。	\$SPNVACK,INFO,1 後接各行
DBG	1	索取完整裝置資訊。在 INFO 的內容之外，增加 \$SPNVDBG 診斷資料，以及 AUTH/INIT 的完整欄位。	\$SPNVACK,DBG,1 後接各行
MODE	ascii binary	切換資料輸出格式。ACK 先用原格式送出，之後的訊息才改用新格式。	\$SPNVACK,MODE,binary
BAUD	460800 921600 1500000 2000000	設定 UART1 鮑率。裝置先用原鮑率送出 ACK，等待傳送完成，再切到新鮑率，間隔約 5 ms；主機收到 ACK 後再調整自己的串列埠。要保留請送 SAVE,1。若保存值不是預設，開機時會先以 460800 bps 提供 500 ms 的 FACTORY 救援等待時間。	\$SPNVACK,BAUD,921600
BRATE	10 25 50 100	設定氣壓計輸出率，單位 Hz。重設感測器最多占用 15 ms，IMU 不受影響。若原過取樣倍率 (OSR) 來不及在新週期內完成，韌體會自動降低倍率，例如 16x、8x、4x。	\$SPNVACK,BRATE,100；失敗 \$SPNVNAK,BRATE,osr/odr invalid i2c error i2c timeout

key	value	作用	ACK 回覆
GTH	浮點 $0 < v \leq 100$	設定合併時判斷陀螺偏差的基本門檻，單位 $^{\circ}/s$ ，預設 0.5；動態時的有效門檻見第 2.2 節。	\$SPNVACK,GTH,0.8
ATH	浮點 $0 < v \leq 100$	設定合併時判斷加速度偏差的基本門檻，單位 m/s^2 ，預設 0.5；動態時的有效門檻見第 2.2 節。	\$SPNVACK,ATH,1.0
REINIT	1	停止取樣，重新初始化三顆 IMU，再恢復取樣；會回報 3 行 \$SPNVINIT。	\$SPNVACK,REINIT,1
PING	任意字串	連線測試	\$SPNVACK,PING,<value>
ACAL	<imu>,<bx>,<by>,<bz>,<kx>,<ky>,<kz>	(fw 0.5.0) 設定指定 IMU 的加速度校正，使用機體座標： $a = (a_raw - b) \cdot k$ 、 $\Delta v = (\Delta v - b \cdot dt) \cdot k$ 。範圍為 $ b \leq 2 m/s^2$ 、 $0.95 \leq k \leq 1.05$ 。設定先套用於 RAM，需送 CALSAVE,1 才保存。RAW 與合併輸出都會套用此校正。	\$SPNVACK,ACAL,<原字串>; \$SPNVNAK,ACAL,range (...)
FWGT	g a,<w1>,<w2>,<w3>	(fw 0.5.0) 設定三顆 IMU 的合併權重：g 用於角速度與 $\Delta\theta$ ，a 用於加速度與 Δv 。權重為相對值，範圍 $0 < w \leq 100$ 。設定先套用於 RAM，需送 CALSAVE,1 才保存。	\$SPNVACK,FWGT,g,1,1,2
CAL	1	以 \$SPNVCAL 回報目前校正值與合併權重。	\$SPNVACK,CAL,1 後接 4 行
CALSAVE	1	將目前 ACAL/FWGT 寫入 NVM。校正資料與使用者設定分開保存，不受 FACTORY 影響。寫入期間 IMU 會短暫停止。	\$SPNVACK,CALSAVE,1; \$SPNVNAK,CALSAVE,nvm not available write failed
CALCLR	1	清除 RAM 與 NVM 中的校正資料，恢復為無校正、三顆等權重。	\$SPNVACK,CALCLR,1
U2OUT	raw fused ahrs baro stat 用 + , 組合; none ; all	(fw 0.8.0) 選擇 UART2 輸出內容，出廠為 raw。stat 包含 STAT×3、FUS、SYS 與 PPS；一般統計每秒一組，PPS 的送出時機見第 10 節。	\$SPNVACK,U2OUT,raw+baro (正規化後的組合)
U2MODE	ascii binary	(fw 0.8.0) UART2 協定，格式與 UART1 相同。出廠 binary	\$SPNVACK,U2MODE,binary
U2BAUD	115200 230400 460800 921600 1500000 2000000	(fw 0.8.0) UART2 鮑率，出廠 460800。UART2 只輸出、不收指令	\$SPNVACK,U2BAUD,921600
SPI	off ardupilot stream (或 0 1 2)	(fw 0.10.0) 選擇 SPI3 功能，見第 13 節。off = 關閉、ardupilot = 飛控 IMU 封包、stream = 資料串流。出廠為 off，此時 SDO 為高阻抗。	\$SPNVACK,SPI,ardupilot
SPIOUT	同 U2OUT	(fw 0.10.0) 選擇 SPI stream 模式的輸出內容，出廠為 none。	\$SPNVACK,SPIOUT,fused+ahrs
SPIMODE	ascii binary	(fw 0.10.0) stream 模式的格式。出廠 binary	\$SPNVACK,SPIMODE,binary

key	value	作用	ACK 回覆
SPICK	3 0	(fw 0.10.0) SPI 時脈模式 (CPOL/CPHA)。出廠 3	\$SPNVACK,SPICK,3
PPS	0 1	(fw 0.8.0) 控制 IMU 輸出週期是否與 GPS PPS 對齊。見第 10 節。出廠為 1；設為 0 時，以開機時間為基準自行定時。	\$SPNVACK,PPS,1
AZB	0 1	(fw 0.9.1) 控制 AHRS 的靜止零偏估計，包含 Z 軸航向零偏。見第 2.4 節。出廠為 1，可用 SAVE,1 保存。	\$SPNVACK,AZB,1
TRIG	0 1 (別名 continuous trigger)	(fw 0.9.0) 0 = 依 RATE 連續輸出，忽略 EXT_TRIGGER；1 = 由外部上升緣觸發輸出。觸發模式的第一個上升緣只建立起點，之後才逐次輸出。見第 12 節。出廠為 0，可用 SAVE,1 保存。	\$SPNVACK,TRIG,1
BOOT	1	(fw 0.6.0) 進入韌體更新程式 bootloader。裝置回覆 ACK 後重啟，在 460800 bps 等待更新工具的 HELLO 封包約 1.5 s；未收到時回到應用韌體。見第 9 節。出貨韌體皆支援，HALT 狀態也可使用。	\$SPNVACK,BOOT,1 ; \$SPNVNAK,BOOT,no bootloader (dev build) use BOOT,1

NAK 原因字串：checksum、not SPNVCMD、missing value、unknown key、use 5|10|25|50|100|200|400|500|1000、use raw|fused|both|none、use 10|25|50|100、range、osr/odr invalid、i2c error、i2c timeout、not leveled yet、use ascii|binary、use 460800|921600|1500000|2000000、use raw+fused+ahrs+baro+stat|none|all、use 115200|230400|460800|921600|1500000|2000000、use 0|1。

SAVE,1 也會保存 U2OUT / U2MODE / U2BAUD / PPS / TRIG / AZB / SPI / SPIOUT / SPIMODE / SPICK；FACTORY,1 會將這些設定恢復為出廠值。

執行 CFG,1 或 INFO,1 時，裝置會額外回報 UART2 狀態：\$SPNVU2,out=<組合>,mode=<ascii|binary>,baud=<n>,drop=<n>,pps=<0|1>,pps_state=<n>,pps_err_ppm=<x>。fw 0.10.0 起也會回報 \$SPNVSPI,mode=...,out=...,fmt=...,ck=...,txn=...,ok=...,short=...,delivered=...,drop=...。詳見第 13.4 節。這些訊息的尾端可能新增診斷欄位，請用 key=value 方式解析。

裝置也支援不需要行尾的單字元指令：1-6 依序對應 5/10/25/50/100/1000 Hz，回覆為 \$SPNVACK,RATE,<hz>；s 表示立即送出統計，r 表示重新初始化。200/400/500 Hz 必須使用完整的 RATE 指令。

5.1 校正流程 (fw 0.5.0)

項目	做法	對應設定
加速度六面校正	將板子分別以 +X、-X、+Y、-Y、+Z、-Z 朝上，各靜置數秒，順序不限。記錄三顆 IMU 的平均比力，再逐顆、逐軸計算： $b = (a_{up} + a_{down}) / 2$ 、 $k = 2g / (a_{up} - a_{down})$ 。GUI「參數設定 → 校正」提供操作精靈，完成後會送出 3 組 ACAL 與 CALSAVE。	ACAL (每顆)
Allan deviation 加權	將板子放在遠離鍵盤、風扇等振動源的位置，靜置至少 30 min。設定 RATE,100、OUT,both 並錄製資料；分析必須有 RAW，both 可同時評估合併結果。原廠 Allan deviation 工具支援 ASCII 與 binary 錄檔，會使用最長的連續資料段，計算各顆各軸的 ARW/VRW ($\tau=1\text{ s}$) 與零偏不穩定性，再依 $w = 1/\sigma^2$ 正規化權重，產生建議的 FWGT 指令。	FWGT,g / FWGT,a
安裝位置差補償 (lever-arm)	三顆 IMU 中心相對參考點的位置 (x, y, z) 是出廠固定值，無法透過指令修改。	出廠設定

六面校正只修正零偏與比例係數，不包含各軸之間的正交誤差。陀螺零偏由 AHRS 在運作中估計，不屬於這項六面校正。

6. 典型輸出序列

使用出廠預設設定時，裝置上電後約 1 s 內開始輸出 50 Hz 資料，不先送出開機資訊。若要查看資訊，可送 \$SPNVCM,INFO,1；主要回覆順序為 \$SPNVACK,INFO,1、\$SPNVBOOT、\$SPNVRESET、\$SPNVAUTH、\$SPNVINIT,1..3、\$SPNVINIT,B。

```

$SPNVIMURAW,1,...*XX
$SPNVIMURAW,2,...*XX
$SPNVIMURAW,3,...*XX
$SPNVIMU,...*XX
$SPNVAHRS,...*XX
$SPNVBARO,...*XX
... ( 預設每 20 ms 輸出 5 行 IMU/AHRS ; BARO 依自己的量測完成時間輸出，每秒 50 行 ) ...
$SPNVSTAT,1,...*XX $SPNVSTAT,2,... $SPNVSTAT,3,... $SPNVFUS,... $SPNVSYS,... $SPNVHB,... (每 1 s)

```

7. 校驗計算範例 (Python)

```
def spnv_checksum(body: str) -> str:
    cs = 0
    for ch in body.encode('ascii'):
        cs ^= ch
    return f"{cs:02X}"

def parse_line(line: str):
    line = line.strip()
    if not line.startswith('$') or '*' not in line:
        return None
    body, cs = line[1:].rsplit('*', 1)
    if spnv_checksum(body) != cs.upper():
        return None          # 校驗錯，整行丟
    return body.split(',')   # f[0] = 'SPNVIMU' 等

# 送指令
cmd = "SPNVCMD,RATE,100"
ser.write(f"${cmd}*{spnv_checksum(cmd)}\r\n".encode())
```

8. Binary 模式 (D6 44)

送出 \$SPNVCMD,MODE,binary 後，裝置傳給主機的訊息會改用二進位封包。主機送出的指令仍使用 ASCII 格式的 \$SPNVCMD。送 MODE,ascii 可切回文字輸出；出廠預設為 ascii。

8.1 封包結構

D6 44 | id(1) | len(1) | seq(1) | payload(len) | crc16(2)

偏移	欄位	說明
0-1	sync	0xD6 0x44
2	id	訊息類別
3	len	資料內容 (payload) 的長度，範圍為 0-255 bytes。
4	seq	各訊息類別 id 分別計數；每送一筆加 1，超過 255 後回到 0。GUI 可用序號判斷是否漏收封包。
5...	payload	多位元組數值採 little-endian (低位元組在前)；f32 為 IEEE-754 32-bit 浮點數。ts_us 為 u64，單位 μs，數值與 ASCII 相同。
末 2	crc16	CRC-16/CCITT-FALSE：poly 0x1021、init 0xFFFF、不反射、無 xorout。CRC 以低位元組在前的順序存放。計算範圍從 byte 2 到 payload 最後一個位元組，包含 id、len、seq、payload。測試字串 "123456789" 的結果為 0x29B1。

8.2 訊息表

id	名稱	對應 ASCII	內容長度	總長度
0x01	IMU_RAW	\$SPNVIMURAW	68	75
0x02	IMU	\$SPNVIMU	68	75
0x03	BARO	\$SPNVBARO	24	31
0x04	AHRS	\$SPNVAHRS	52	59
0x05	PPS	\$SPNVPPS (fw 0.8.0)	24	31
0x10	STAT	\$SPNVSTAT	44	51
0x11	FUS	\$SPNVFUS	32	39
0x12	SYS	\$SPNVSYS	48 (fw 0.8.0 : 40 ; fw ≤ 0.7.0 : 36)	55
0x7F	TEXT	其他所有 (BOOT/RESET/DBG/AUTH/INIT/HB/ACK/NAK/STALL/TRAP)	可變	≤ 262

上表的長度單位為位元組 (bytes) 。各欄位的單位與狀態位元定義，與對應的 ASCII 訊息相同。下方列出欄位偏移、型別與 Python struct 格式；參考程式見 tools/spnv_parser.py 。

0x01 IMU_RAW (68) — <BBHQ3f3ff3f3ff

偏移	欄位	型別
0	imu_id	u8
1	status	u8
2	n	u16
4	ts_us	u64
12	gyro[3]	f32×3 °/s
24	acc[3]	f32×3 m/s ²
36	temp	f32 °C
40	dth[3]	f32×3 rad
52	dv[3]	f32×3 m/s
64	dt	f32 s

0x02 IMU (68) — <Q3f3ff3f3fBBB

偏移	欄位	型別
0	ts_us	u64
8	gyro[3]	f32×3
20	acc[3]	f32×3
32	temp	f32
36	dth[3]	f32×3
48	dv[3]	f32×3
60	dt	f32
64	n_used	u8
65	flags	u8
66	used_mask	u8 ; bit0、bit1、bit2 分別表示是否採用 IMU1、IMU2、IMU3。
67	reserved	u8

0x03 BARO (24) — <QffB3x

偏移	欄位	型別
0	ts_us	u64
8	press_hpa	f32
12	temp_c	f32
16	alt_m	f32
20	status	u8
21	reserved[3]	u8

0x04 AHRS (52) — <Q4f3f3fBBH

偏移	欄位	型別
0	ts_us	u64
8	q[4]	f32×4 (q0 q1 q2 q3, body→NED)
24	roll, pitch, yaw	f32×3 deg
36	gyro_bias[3]	f32×3 °/s
48	align	u8
49	flags	u8
50	zb_ds	u16 ; fw 0.9.2 起表示連續靜止時間，單位為 0.1 s；舊版固定為 0。

0x10 STAT (44) — <BBBB9IHH

偏移	欄位	型別
0	imu_id	u8
1	dry_timeout	u8
2	dry_pin	u8
3	isr_cpu_pct	u8
4	dry	u32
8	out	u32
12	crc_err	u32
16	frame_err	u32
20	spi_timeout	u32
24	dropped	u32
28	dcnt_skip	u32
32	dry_kicks	u32
36	resync	u32
40	isr_us_max	u16
42	isr_us_last	u16

0x11 FUS (32) — <6IfF

偏移	欄位	型別
0	outputs	u32
4	rej[3]	u32×3
16	incomplete	u32
20	degraded	u32
24	gyro_thresh	f32
28	acc_thresh	f32

0x05 PPS (24) — <QIIiBBH (fw 0.8.0)

偏移	欄位	型別
0	ts_us：最近一次通過條件檢查的 PPS 上升緣時間，與 IMU 使用相同時間基準。	u64
8	edges：通過條件檢查的 PPS 邊緣累計數。	u32
12	captures：原始擷取次數，包含雜訊。	u32
16	err_ppb：本地晶振相對 PPS 的頻率偏差；正值表示本地時鐘較快。	i32
20	state (0 OFF / 1 NO_SIGNAL / 2 ACQUIRING / 3 LOCKED / 4 HOLDOVER)	u8
21	align：1 表示輸出週期的時間邊界已與 PPS 對齊。	u8
22	age_ms：距離最近一次合格 PPS 邊緣的時間，上限為 65535 ms。	u16

0x12 SYS (48) — <7IHBBBBBIII (fw 0.8.0 為 40 byte、fw ≤ 0.7.0 為 36 byte、尾端欄位不存在)

偏移	欄位	型別
0	uart_drop	u32
4	loops_per_s	u32
8	uart_resets	u32
12	pps_cnt	u32
16	baro_samples	u32
20	baro_i2c_err	u32
24	baro_timeouts	u32
28	rate_hz	u16
30	isr_cpu_total_pct	u8
31	out_mode	u8 (0 none/1 raw/2 fused/3 both)
32	baro_stale	u8
33	baro_hz	u8
34	proto_mode	u8 (0 ascii/1 binary)
35	pps_state	u8 (fw 0.8.0；舊版為 reserved 0)
36	u2_drop	u32 (fw 0.8.0)
40	trig_cnt	u32 (fw 0.9.0)
44	trig_rej	u32 (fw 0.9.0)

解析 SYS 時，請先依 len 判斷版本：可能為 36、40 或 48 bytes。各版本共用的前段欄位排列相同；只有長度足夠時，才讀取後方新增的欄位。

0x7F TEXT

TEXT 的 payload 是 ASCII 訊息中 \$ 與 *XX 之間的文字，不包含 \$、*XX 或行尾 CRLF。例如，ACK 訊息的內容為 SPNVACK,RATE,100。GUI 可直接顯示，也可用逗號分隔各欄位。

8.3 二進位輸出的資料量與頻寬

設定	流量	460800 (46 kB/s)	921600 (92 kB/s)
50 Hz both + AHRS + BARO	19 kB/s	OK	OK
100 Hz both + AHRS + BARO	37 kB/s	OK	OK
200 Hz fused + AHRS	27 kB/s	OK	OK
400 Hz fused + AHRS	54 kB/s	超過	OK
500 Hz fused + AHRS	67 kB/s	超過	OK
1000 Hz 單顆 raw	75 kB/s	超過	OK
1000 Hz fused + AHRS	134 kB/s	超過	超出上限，請用 1.5M 或 2M。

設定	流量	460800 (46 kB/s)	921600 (92 kB/s)
1000 Hz 三顆 raw	225 kB/s	超過	超出上限，請改用 SPI3。

8.4 鮑率設定與實際誤差

用 \$SPNVCMD,BAUD,<n> 設定 UART1 鮑率。設定值、實際鮑率與誤差如下：

鮑率	實際	誤差
460800	458716	0.45%
921600	925926	0.47%
1500000	1515152	1.0%
2000000	2000000	0%

8.5 主機端工具

原廠參考解析器 tools/spnv_parser.py 可處理 ASCII 與 binary 資料，檢查校驗碼或 CRC，並統計漏收封包。

```
python3 spnv_parser.py COM5 460800 --binary --cmd RATE,100 --cmd OUT,fused
python3 spnv_parser.py --file capture.bin
```

程式介面 SpnvParser.feed(bytes) 會回傳一個列表，每筆結果的格式為 (kind, name, dict)。

9. 韌體更新

- 在 SparkNavi IMU600 GUI 選擇「韌體更新」，並載入原廠發布、適用本產品的 .spfw 檔案。更新檔經過加密與驗證；格式或產品不符的檔案不會被接受。
- 更新流程：GUI 送出 \$SPNVCMD,BOOT,1；裝置回覆 ACK 後重新啟動，進入更新程式 bootloader；GUI 接著傳送並驗證檔案，完成後自動啟動新韌體。bootloader 不輸出文字訊息，會以 460800 bps、8N1 等待更新連線約 1.5 s；若沒有收到更新要求，就回到原有應用韌體。
- 使用者設定與六面校正資料存放在獨立區域，更新韌體不會清除。
- 若更新途中斷電、線材脫落或驗證失敗，裝置會停留在 bootloader 等待重新更新。請在 GUI 使用「救援更新」重新傳送。若開機時發現應用韌體異常，也會進入相同的等待狀態。
- 韌體更新使用的通訊格式不在本文件公開。如需把更新功能整合到自有工具，請洽原廠。

10. GPS PPS 時間對齊 (fw 0.8.0)

將 GPS 或其他準確的 1 Hz 參考訊號接到 GPS_PPS_IN，裝置就能讓 IMU 的輸出週期對齊 PPS 上升緣。這條訊號只提供「整秒邊界」，不包含日期或 GPS 時間資料。

- 鎖定後，相鄰兩個 PPS 上升緣之間會輸出 RATE 筆資料。其中一筆的 ts_us 等於 PPS 上升緣的時間，並在 RAW 的 status bit7、合併資料的 flags bit7 標記為 1。主機可將這筆資料與 GPS 提供的整秒時間配對。

- 裝置每秒量測實際 PPS 間隔，再將這段間隔分成 RATE 等份，作為下一秒的輸出時間點。這樣可補償本地 50 MHz 晶振數十 ppm 的頻率偏差，避免對齊誤差逐秒累積。\$SPNVPPS 的 err_ppm 表示量到的晶振偏差。
- ts_us 仍以裝置的本地 50 MHz 時鐘計算，從開機起算，不會變成 GPS 或 UTC 時間。PPS 的作用是決定每筆 IMU 資料的取樣時段在哪個時間點結束。
- 接線規格：3.3 V CMOS，上升緣有效，輸入端有內部下拉。脈衝寬度至少 1 μ s；一般 GPS 約為 100 ms。輸入電壓不可超過 3.3 V。
- PPS 的相鄰上升緣間隔須在 $1\text{ s} \pm 1000\text{ ppm}$ 內，也就是約 0.999–1.001 s。連續 2 次間隔合格後才進入鎖定。過短的間隔（小於 $1\text{ s} - 1000\text{ ppm}$ ）視為毛刺並忽略；其他不合格間隔會讓裝置回到 ACQUIRING。若只是漏掉 1–9 個脈衝，而下一次間隔仍符合 $k\text{ s} \pm k \times 1000\text{ ppm}$ ，則可沿用原本的對齊基準。擷取解析度為 20 ns。
以 MCU 軟體延遲或一般訊號產生器產生的 1 Hz 訊號，可能達不到精度要求，請用示波器確認。例如標示 1 Hz、實測卻為 993.83 mHz 時，偏差約 6200 ppm；此時 pps_cnt 仍會增加，但無法進入 LOCKED。
- \$SPNVPPS 只在接受合格邊緣，或 PPS 狀態改變時送出。若一直停在 ACQUIRING，且收到的間隔都不合格，不會每秒送一行。此時請查看 \$SPNVSYS 的 pps_cnt（第 7 欄）與 pps_state（第 15 欄）。

PPS 狀態可從 \$SPNVPPS 第 4 欄、\$SPNVSYS 第 15 欄，或 \$SPNVU2 的 pps_state 查看：

值	狀態	說明
0	OFF	PPS 功能已關閉 (PPS,0)。裝置從開機起，每隔 1/RATE 自行決定輸出時間。
1	NO_SIG NAL	尚未收到合格的 PPS，例如未接線；輸出仍按裝置自己的時鐘運作。
2	ACQUIR ING	已收到邊緣，但尚未連續取得 2 個符合 $1\text{ s} \pm 1000\text{ ppm}$ 的間隔。
3	LOCKED	已鎖定 PPS，輸出時間邊界持續與 PPS 對齊。
4	HOLDOV ER	先前已鎖定，但超過 1.5 s 未收到 PPS。裝置暫時沿用最後量到的間隔繼續輸出。PPS 恢復且符合整數秒間隔時，立即回到 LOCKED；若邊緣時間已偏離原本的整秒基準，則回到 ACQUIRING 重新對齊。

\$SPNVPPS 格式如下。每接受一個合格邊緣或每次狀態改變時送出一行；UART1 須開啟 STAT，UART2 則須在 U2OUT 中包含 stat。

```
$SPNVPPS,<ts_us>,<edges>,<captures>,<state>,<err_ppm>,<age_ms>,<align>*XX
$SPNVPPS,1234000000,58,58,3,12.345,0,1*XX
```

- 間隔小於 $1\text{ s} - 1000\text{ ppm}$ 的邊緣視為雜訊或毛刺，直接忽略，不改變目前狀態。captures – edges 可用來查看未被採用的擷取次數。
- 剛進入 LOCKED 時，輸出時間邊界會調整一次，幅度最多為一個輸出週期。因此，該筆 dt 可能較短或較長；完成調整後會持續連續輸出。

11. UART2 第二輸出埠 (fw 0.8.0)

- UART2 使用 U2TX、U2RX 腳位，電氣規格為 3.3 V TTL、8N1。目前只透過 U2TX 輸出資料，U2RX 不接收指令。

- UART2 的設定都由 UART1 下指令：U2OUT 選擇資料內容、U2MODE 選擇格式、U2BAUD 設定鮑率，再用 SAVE,1 保存。出廠值為 binary、raw (三顆 IMU)、460800 bps。
- UART2 的資料格式與 UART1 相同，可選 ASCII 或 D6 44 binary；binary 的 seq 由 UART2 獨立計數。ACK、NAK、INFO 等文字回覆只從 UART1 送出。
- UART2 的輸出選項與 UART1 的 OUT、BARO、AHRS、STAT 互相獨立。例如，可讓 UART1 只送 fused 給 GUI，同時讓 UART2 送 raw 給飛控。
- 頻寬估算：binary 模式每批三顆 RAW 共 225 B。100 Hz 時約 22.5 kB/s，460800 bps 可負荷；1000 Hz 時約 225 kB/s，已超過 2000000 bps 約 200 kB/s 的上限，請改為只送 fused 或降低輸出率。緩衝區滿時會丟棄整筆資料，累計數可查看 \$SPNVSYS 第 16 欄或 \$SPNVU2 的 drop。

12. EXT_TRIGGER 外部觸發 (fw 0.9.0)

送出 \$SPNVCMD,TRIG,1，或在 GUI 將「IMU 輸出模式」改為「外部觸發」後，裝置依下列規則運作：

- 進入此模式後，裝置先等待 EXT_TRIGGER (J2 pin 15) 的上升緣，不主動送出 \$SPNVIMURAW、\$SPNVIMU 或 \$SPNVAHRS。若已保存此設定，開機後也會先等待觸發。
- 第一個上升緣只用來建立起點，不輸出資料。之後每個上升緣結束一段取樣時間，並產生一批資料；實際送出的 RAW、合併及 AHRS 內容，仍依 OUT、AHRS、U2OUT 設定。持續輸入 100 Hz 觸發時，就以每秒 100 批的速度輸出。
- 每筆 ts_us 記錄本次觸發上升緣的時間。EXT_TRIGGER 中斷的優先權高於取樣中斷，只讀取 50 MHz 計時器，以保留觸發時間。 $\Delta\theta$ 與 Δv 累計的是前一次到本次觸發之間的資料，dt 表示這段積分時間。實際切分以原始樣本為單位，因此與精確觸發間隔的差小於 1 個樣本，約 0.68 ms。相鄰輸出的積分時段會接續，不重複也不漏掉樣本；觸發頻率不必是 RATE 的倍數。
- 從觸發到輸出的延遲，約為等待 1 個原始樣本的時間 (≤ 0.7 ms)，再加上處理與傳輸時間。
- 此模式的 IMU 輸出速度由觸發訊號決定，RATE 設定會保留，供切回連續模式時使用。氣壓計仍依 BRATE 獨立輸出，統計資料仍每秒一組。PPS 對齊在此模式不作用，因為輸出時間邊界由 EXT_TRIGGER 決定。
- 建議觸發頻率為 1–1000 Hz。間隔小於 0.5 ms 的邊緣視為毛刺，會被忽略並累計至 trig_rej。若超過約 10 s 沒有觸發，會放棄該段資料，下一個上升緣重新作為起點。若觸發間隔超過 1 s，仍輸出 RAW 與合併資料，但 AHRS 不更新。傳輸頻寬請依實際觸發頻率估算，見第 2.5、8.3 節。
- 接線規格：3.3 V CMOS，上升緣有效，輸入端有內部下拉。觸發源請使用 3.3 V 推挽輸出，脈衝寬度至少 1 μ s，電壓不可超過 3.3 V。
- 送出 \$SPNVCMD,TRIG,0 可回到連續模式，裝置改依 RATE 定時輸出，並忽略 EXT_TRIGGER 的邊緣。
- 目前模式可查看 \$SPNVCFG 的 trig；接受與忽略的觸發次數，分別查看 \$SPNVSYS 的 trig_cnt、trig_rej (第 17、18 欄)。

舊版差異：fw $\leq 0.8.0$ 仍持續輸出，EXT_TRIGGER 只會額外再送出最新一筆資料。觸發頻率與 RATE 不同時，可能重複或漏送同一筆，造成 $\Delta\theta/\Delta v$ 重複計算；當時腳位也沒有內部下拉。這項舊行為已移除。

13. SPI3 從機輸出 (fw 0.10.0 起)

IMU600 作為 SPI 從機，由飛控（主機）按固定週期讀取。這版硬體沒有資料就緒訊號 DRDY，因此飛控須用封包序號與時間戳判斷是否重複讀取或漏收。

本節的通訊行為以 fw 0.10.8 為基準，並於 2026-10-15 與 ArduPilot 4.7.1 完成實機驗證。

J3	訊號
6	FC_SPI_CS，低電位有效。fw 0.10.1 起啟用內部上拉，避免飛控重開機或未上電時，CS 處於不確定電位。
8	SCK3IN
10	SDI3 (MOSI) ；此介面不使用輸入資料，飛控送 0 即可。
12	SDO3 (MISO) ；設為 SPI,off 時為高阻抗。

- SPI 預設為 Mode 3，也可用 SPICK 改為 Mode 0；每個位元組先送最高位元 (MSB first)。建議飛控 SPI 時脈設為 2 MHz，此設定已在 Blue2 + ArduPilot 4.7.1 驗證；8 MHz 尚未驗證。
- 每次將 CS 拉低後，應完整讀取 128 bytes，再將 CS 拉高。讀滿 128 bytes 才算完成這個封包；若未讀滿，下次仍提供同一個封包。若讀取超過 128 bytes，多出的部分會回傳 0。
- CS 拉高後，IMU600 會準備下一個封包。飛控下次將 CS 拉低時，即可開始送出 SPI 時脈並讀取。
- 連接 ArduPilot 時，請設 RATE,1000，原因見第 14 節。SPI3 本身也支援其他 RATE 值 (5/10/25/50/100/200/400/500 Hz)，但 ArduPilot 的主迴圈為 400 Hz 時，起飛前檢查要求陀螺資料更新率至少為 720 Hz。因此，200/400/500 Hz 都無法通過這項檢查。
- 連接 ArduPilot 時，建議設 OUT,none、AHRS,0、U2OUT,none，關閉不需要的串列資料。三顆 IMU 在 1000 Hz 的 ASCII 資料量遠超過 921600 bps；即使在 400 Hz，OUT,both 也會塞滿 UART1，讓 uart_drop 持續增加。這會使 GUI 顯示的接收速率與「漏格」計數失準，但不影響 SPI3 輸出。

13.1 SPI 封包結構 (128 bytes，little-endian)

offset	型別	內容
0	u8×2	0x53 0x4E ("SN")
2	u8	版本 = 1
3	u8	封包類型：0 = 空封包，沒有新資料；1 = IMU 資料；2 = 資料串流。
4	u16	seq：IMU 封包每產生一批輸出就加 1；串流封包則按分段計數。空封包沿用最後完成讀取的 seq。
6	u8	pending：本封包之後還有多少個封包等待讀取。大於 0 時，可立即再讀。
7	u8	IMU 封包：used_mask，bit0–2 分別表示 IMU1–3 是否參與合併。串流封包：payload 的有效長度，範圍 0–118 bytes。
8..125	—	資料內容 (payload)，格式依封包類型而定。
126	u16	CRC-16/CCITT-FALSE，poly 0x1021、init 0xFFFF；計算範圍為 byte 0–125，演算法與 binary 模式相同。

IMU 資料內容 (類型 1 , SPI,ardupilot) : 使用機體 FRD 座標 , 對應同一批 \$SPNVIMU / \$SPNVIMURAW 資料。下表 offset 從整個封包的起點計算。

offset	型別	內容
8	u64	ts_us : 這批資料所涵蓋時段的結束時間 ; PPS 與外部觸發的時間規則和 UART 相同。
16	f32	dt (s)
20	f32×3 / f32×3	合併 $\Delta\theta$ (rad) / Δv (m/s)
44 / 68 / 92	各 f32×6	依序為 IMU1、IMU2、IMU3 的 $\Delta\theta[3]$ 與 $\Delta v[3]$ 。若本次缺少某顆資料 , 該顆數值全為 0 , status 為 0xFF。
116	i16×3	IMU1..3 溫度 (0.01 °C)
122	u8×3	IMU1..3 status (同 \$SPNVIMURAW , bit7 = PPS 標記)
125	u8	合併 flags (同 \$SPNVIMU)

串流資料內容 (類型 2 , SPI,stream) : 從 byte 8 起讀取 len 個有效位元組 , 再將各封包內容依序接起來。得到的格式與 UART2 相同 , 可直接交給既有的 ASCII 或 D6 44 解析程式。內容由 SPIOUT 選擇 , 格式由 SPIMODE 設定。

13.2 飛控讀取建議

- 建議讀取間隔為輸出週期的 0.9 倍。例如 RATE,1000 時 , 每 0.9 ms 讀取一次。若 pending > 0 , 立即繼續讀取 , 但一次最多連讀 5 個封包。
- 每次先檢查固定識別碼 magic、版本與 CRC。類型 0 表示沒有新資料 , 可略過。若 seq 與上次相同 (gap = 0) , 也應略過 , 避免重複使用資料 ; fw 0.10.8 起 , 疑似未正常完成的交易可能重送同一個序號。IMU 封包的 seq 應依序加 1 ; 若 gap > 1 , 表示中間可能因佇列已滿或讀取失敗而漏收。
- 佇列最多保存 16 個封包 , RATE,1000 時約相當於 16 ms 的資料。佇列滿時會丟棄最舊封包 , 累計至 \$SPNVSPI 的 drop。飛控尚未開始讀取前 , drop 增加屬於等待期間的結果 ; 開始正常讀取後 , 就不應持續增加。
- 若 CRC 檢查失敗 , 請丟棄這次讀到的內容 , 並在下次重新讀取。若此次未讀滿 128 bytes , 裝置不會移到下一個封包 , 下次仍會提供同一包資料。

13.3 實機驗證結果 (Blue2 + ArduPilot 4.7.1 · 2026-10-15)

項目	結果
SPI 時脈 / 模式	2 MHz · Mode 3 (SPICK,3) · 飛控 SPI6 + 外部 CS
輸出率	1000 Hz ; 每秒完成讀取約 1000 個資料封包 (delivered) · SPI 交易約 1190 次/s (txn) 。
資料正確性	靜置時 Accel Z 約 -1000 mG · X/Y 約 0 · 與飛控 IMU1/2 一致。轉動方向一致 · 使用 ROTATION_NONE ; EKF3 啟用三顆 IMU (EK3_IMU_MASK=7) 。
起飛前檢查	1000 Hz 通過陀螺資料更新率檢查 · 飛控顯示 Ready to Arm 。
錯誤計數 (fw 0.10.8 · 7.6 分鐘)	short = 4 ; dmaerr = 0 ; drop 在飛控讀取期間不增加
飛控 MP 訊息	不再因偶發錯誤顯示 IMU600: x lost, y bad reads ; 偶發 bad read / lost 記錄於飛控日誌 SNV6 。

13.4 \$SPNVSPI 欄位

送出 \$SPNVCMD,CFG,1 或 INFO,1 後 · 裝置會回報以下 SPI 狀態：

```
$SPNVSPI,mode=ardupilot,out=none,fmt=binary,ck=3,txn=...,ok=...,short=...,delivered=...,drop=...,dma=1,dmaerr=0,...*XX
```

欄位	說明	正常值 / 判讀
mode out fmt ck	SPI / SPIOUT / SPIMODE / SPICK 目前設定	—
txn	SPI 交易的累計次數 · 以 CS 動作計數。	每秒增加約為輸出率的 1.2 倍。
ok	讀滿 128 bytes 的次數	接近 txn
short	未讀滿 128 bytes 的累計次數；下次仍提供同一封包。	測試中偶爾出現個位數。若持續增加 · 請檢查飛控 SCK 速度與 CS 時序。
delivered	已完整交付的資料封包累計數。	每秒增加量應接近輸出率。
drop	佇列已滿 · 因而丟棄最舊封包的累計數。	飛控開始正常讀取後不應持續增加。開機到飛控開始讀取前的累計值 · 可與運作期間的新增值分開判斷。
dmaerr	0/1 · 是否曾發生 DMA 錯誤	0

其他欄位 (dma 、 lf 、 lt 、 ls 、 ks 、 kr 、 sp) 供原廠診斷 · 一般解析程式可略過。新版韌體可能在訊息尾端增加欄位。

14. ArduPilot 4.7.1 驅動 (AP_InertialSensor_SparkNavi)

IMU600 在飛控中登錄為第三組 SPI IMU (IMU3) · 原有兩顆內建 IMU 仍繼續使用。發行包 ArduPilot-4.7.1-SparkNavi-IMU600.zip 包含驅動程式、修補檔 (patch) · Blue2 專用板型 SparkNavi-Blue2-IMU600 · 測試用 .apj 與 bootloader 。

需要完成的設定如下：

位置	設定
IMU600 (用 GUI 或指令設定，完成後送 SAVE,1)	必要設定：SPI,ardupilot；SPICK,3；RATE,1000。另建議設 OUT,none、AHRS,0、U2OUT,none，關閉不需要的串列輸出。
飛控 hwdef	SPIDEV imu600 SPI6 DEVID1 EXTERNAL_CS MODE3 2*MHZ 2*MHZ；IMU SparkNavi SPI:imu600 ROTATION_NONE
飛控參數	EK3_IMU_MASK = 7；新板型的 defaults.parm 已設定。請再執行一次飛控的六面加速度計校正，三組 IMU 一起校正。

RATE 必須設為 1000 Hz 的原因：ArduPilot 的起飛前檢查

AP_InertialSensor::pre_arm_check_gyro_backend_rate_hz 要求非串列介面陀螺的資料更新率，至少為主迴圈的 1.8 倍。Copter 預設主迴圈為 400 Hz，因此最低要求為 720 Hz。若 IMU600 設為 400 Hz，會出現 PreArm: Gyro 2 rate 400Hz < loop rate*1.8 720Hz，無法解鎖。請改用 1000 Hz，以符合檢查條件。

驅動程式的處理方式：

- 開機偵測：讀取 128 bytes，檢查識別碼 SN、版本 1 與 CRC16。探測至少持續 500 ms，最晚等到飛控開機後 5 s；取得 50 筆 IMU 封包後，以 dt 的中位數判定輸出率，並對應到標準速率。運作中每 2 s 再核對一次。成功時 Mission Planner 的 Messages 顯示 IMU600 found, 1000 Hz；失敗時顯示 IMU600 probe: rd ... hdr ... crc ...。未接 IMU600 時會略過，內建兩顆 IMU 不受影響。
- 模式檢查：若 IMU600 設為 SPI stream，會顯示 IMU600: SPI is in stream mode, set SPI ardupilot，並且不將它加入飛控的 IMU 清單。請改設 SPI,ardupilot。
- 資料處理：將合併後的 $\Delta\theta$ 、 Δv (FRD 座標，含 coning/sculling 補償) 交給 _notify_new_delta_angle/_velocity，並使用封包的 dt 換算。若 used_mask = 0 或資料出現 NaN，會累計 gyro/accel 錯誤次數，不交給 INS 使用；SATURATED 會增加 accel clip 計數。溫度取三顆平均，約每秒發布 10 次。
- 讀取方式：間隔為輸出週期的 0.9 倍。pending > 0 時立即再讀，一次最多 5 包。重複序號直接略過；gap > 1 時累計 lost，CRC 錯誤時累計 bad。
- 錯誤紀錄：偶發的 bad read 或 lost 不直接顯示在地面站 (GCS)。若最近 10 s 內有發生，就寫入 SNV6 日誌，欄位為 TimeUS、Lost、Bad、Pkts。只有 10 s 內 lost $\geq$ 20 或 bad $\geq$ 100，才在地面站顯示 IMU600: SPI link errors (...)。驗證時可設 LOG_DISARMED=1，讓飛控在未解鎖時也錄製日誌，再查看 SNV6。

ArduPilot 整合注意事項：

- Mission Planner 顯示的加速度單位為 mG，角速度單位為 mrad/s。SCALED_IMU2 與 SCALED_IMU3 分開傳送，因此運動中看到的兩組時間戳可能不同。
- IMU600 的 ACAL 與 ArduPilot 的 INS_ACCOFFS 是兩套校正。建議選定一套進行校正；若使用 ArduPilot 校正，IMU600 可維持 ACAL 預設值 (零偏 0、比例 1.0)。若之後又透過 GUI 改動 IMU600 校正值，必須重新執行 ArduPilot 校正。

- Blue2 飛控重開機時，IMU600 的 5 V 供電可能波動。Blue2 的 bootloader hwdef 需將 PE2/PE3 設為 OUTPUT HIGH。燒錄 .apj 後，還須在 Mission Planner 執行一次「Flash bootloader」，新的 bootloader 才會實際寫入。