

# IMU配置手冊

適用產品：CH0x0/CH10x/HI14系列 韌體v1.5.2

官方網站：<https://sealandtech.com.tw>

屬性：公開

採購、支援：[sltech@ms28.hinet.net](mailto:sltech@ms28.hinet.net)

## IMU配置手冊

### 1. 模組配置概述

#### 1.1. 地磁輔助場景(AHRS/9軸模式)

#### 1.2. 同步輸入與同步輸出

##### 1.2.1. 同步輸入(SIN)

##### 1.2.2. 同步輸出(SOUT)

### 2. 模組配置命令

#### 2.1. 配置命令總覽

#### 2.2. 指令詳解

##### 2.2.1. REBOOT

##### 2.2.2. SAVECONFIG

##### 2.2.3. SERIALCONFIG

##### 2.2.4. CONFIG

工作模式配置: 6軸 或者9軸(地磁輔助)模式

水平校準

手動設定航向角

座標系旋轉(改變安裝方式)

多功能IO複用功能配置

設定IO報警觸發參數(PMUX5: ALARM)

設定SOUT\_DIV分頻係數(PMUX4: SOUT\_DIV)

##### 2.2.5. LOG

ENABLE/DISABLE: 全域性打開/關閉數據輸出

VERSION: 顯示模組版本資訊

USRCONFIG: 顯示使用者配置資訊

COMCONFIG: 顯示串列埠配置資訊

配置數據幀輸出及設定輸出頻率

配置為同步輸出模式

##### 2.2.6. UNLOGALL

##### 2.2.7. FRESET

#### 2.3. 更多高級配置

##### 2.3.1. 配置加速度和陀螺儀頻寬

### 3. 數據輸出協議(自定義二進制協議)

#### 3.1. 數據幀格式

#### 3.2. 出廠預設輸出

#### 3.3. 數據域內容

##### 3.3.1. 浮點型IMU數據幀(0x91)

##### 3.3.2. 整型IMU數據幀(0x92)

#### 3.4. CRC

#### 3.5. 數據幀結構示例

- 3. 6. 最大傳輸速率
- 4. RS-485數據協議與指令(Modbus協議)
  - 4. 1. 數據幀格式
    - 4. 1. 1. 讀暫存器(0x03)
    - 4. 1. 2. 寫暫存器(0x06)
    - 4. 1. 3. CRC校驗
  - 4. 2. 暫存器列表
  - 4. 3. 常用配置
    - 4. 3. 1. 控制暫存器說明(0x00)
    - 4. 3. 2. 配置波特率(0x04)
    - 4. 3. 3. 設定安裝方式 ( 0xA6 )
    - 4. 3. 4. 設定水平校準(0xA5)
    - 4. 3. 5. 設定6軸或9軸模式
  - 4. 4. 讀取模組版本資訊
  - 4. 5. 讀取感測器數據
  - 4. 6. Modbus ID自動分配
- 5. CAN數據協議與指令(CANopen協議)
  - 5. 1. CANopen 預設設定
  - 5. 2. CANopen TPDO
  - 5. 3. 使用上位機連線CAN裝置
  - 5. 4. 配置指令
    - 5. 4. 1. LLS(Layer Setting Services)協議
      - 修改波特率(重新上電生效)
      - 修改節點ID
      - 儲存配置
    - 5. 4. 2. NMT(Network Management)協議
      - 全域性使能/關閉數據輸出(開啟非同步觸發)
      - 復位
    - 5. 4. 3. SDO(Service Data Object)協議
      - 修改/關閉/開啟數據輸出速率
      - 設定傾角儀輸出正負號
      - 設定傾角儀零點(相對位置輸出)
      - 設定傾角輸出解析度(定製功能)
      - 恢復出廠設定
    - 5. 4. 4. 同步協議
      - 配置TPDO為同步模式
- 6. CAN數據協議與指令( SAE-J1939協議)
  - 6. 1. PGN訊息列表
    - 6. 1. 1. PGN65332(FF34) 加速度
    - 6. 1. 2. PGN65335(FF37) 角速度

- 6. 1. 3. PGN65345(FF41) 航向角
- 6. 1. 4. PGN65354(FF4A) 傾角儀輸出
- 6. 2. 配置指令
  - 6. 2. 1. 配置格式
  - 6. 2. 2. 配置模組
- 7. CAN數據協議與指令( NMEA2000協議)
- 8. 地磁校準
  - 8. 1. 地磁校準步驟
  - 8. 2. 配置地磁校準(手動地磁校準)
  - 8. 3. 磁干擾的知識
    - 8. 3. 1. 空間磁場干擾
    - 8. 3. 2. 感測器座標系下的干擾
  - 8. 4. 地磁使用注意事項

# 1. 模組配置概述

產品的預設配置已經可以滿足大多數使用者需求，因此在使用產品前需要仔細閱讀此章節，結合自身的使用需求，判斷是否需要進行使用者配置。

## 1.1. 地磁輔助場景(AHRS/9軸模式)

在絕對大多數情況下，機器人及室內環境，AHRS(9軸)模式很容易受到干擾導致航向角產生誤差。在少數空曠且無磁場干擾的環境下，可以嘗試使用地磁輔助模式，比如無人機，在使用前需要先將模組配置為地磁輔助模式並且進行地磁校準。詳見地磁校準章節。

序列介面配置請參考 CONFIG-模式配置

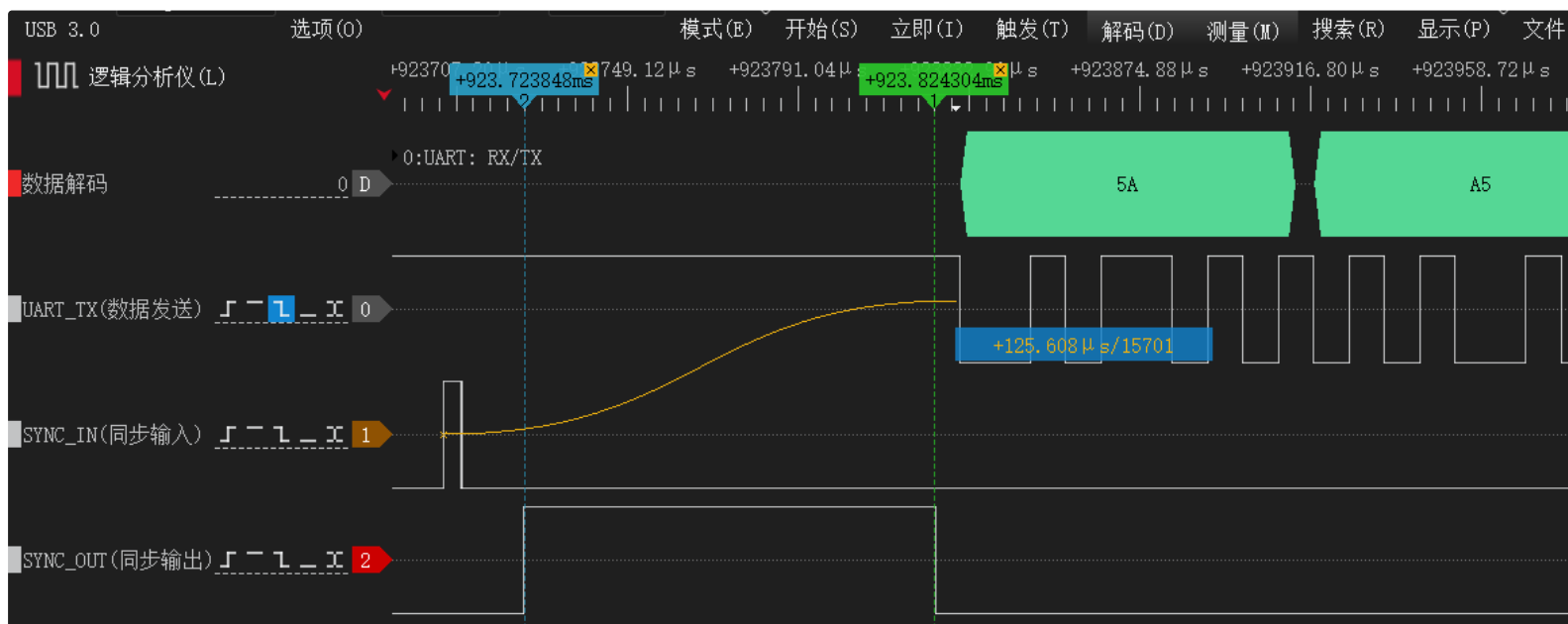
## 1.2. 同步輸入與同步輸出

### 1.2.1. 同步輸入(SIN)

同步脈衝輸入(SYNC\_IN): 輸入引腳，空閑狀態為高電平。當模組檢測到下降沿時，會輸出一幀數據(如果為ONMARK觸發)。同時，一些協議可以輸出PPS同步時間戳。PPS同步時間戳指的是: 當模組檢測到最近一次下降沿訊號時開始，到目前一幀數據採樣時所經過的時間。該引腳不使用可懸空。

### 1.2.2. 同步輸出(SOUT)

- 數據同步輸出(SYNC\_OUT): 輸出引腳，無數據輸出時為低電平(空閑)，一幀數據開始發送時會發送一個高脈衝



在使用同步輸入功能時，先要通過串列埠輸入指令UNLOGALL來取消所有非同步定時輸出，同時使用LOG <MSG> ON MARK 1 命令打開觸發輸入，具體請參見LOG命令

## 2. 模組配置命令

模組配置採用ASCII字串命令，每條命令必須以回車換行 `\r\n` 結束(類似AT指令)，才能被系統識別。

### 2.1. 配置命令總覽

指令	功能	備註
REBOOT	復位模組	和重上電等效
SAVECONFIG	儲存所有配置參數	立即生效
SERIALCONFIG	波特率設定	立即生效
CONFIG	設定使用者參數及模式	立即生效
LOG	列印模組資訊或配置輸出數據	立即生效
UNLOGALL	關閉所有定時輸出訊息	立即生效
FRESET	恢復出廠設定	立即生效

所有配置指令需要復位或重新上電后才能生效

### 2.2. 指令詳解

#### 2.2.1. REBOOT

復位模組，立即生效，和重新上電效果相同。

#### 2.2.2. SAVECONFIG

儲存所有使用者配置到Flash。

#### 2.2.3. SERIALCONFIG

設定串列埠波特率，可選值：9600/115200/256000/460800/921600

配置串列埠波特率為115200

`SERIALCONFIG 115200`

`SAVECONFIG`

使用此指令需要特別注意，輸入錯誤波特率後會導致無法和模組通訊。

#### 2.2.4. CONFIG

用於配置模組工作參數，絕大多數命令執行后立即生效，且需要 `SAVECONFIG` 才能掉電儲存

工作模式配置: 6軸 或者9軸(地磁輔助)模式

- `CONFIG ATT MODE 0` 配置模組為6DOF模式
- `CONFIG ATT MODE 1` 配置模組為AHRS(9軸)模式

水平校準

- `CONFIG ATT RST 3` 自動校平: 如果目前俯仰角/橫滾角接近 $0^{\circ}, 0^{\circ}$ (水平正面放置), 則自動校準到 $0, 0$ 。如果目前俯仰角/橫滾角接近 $0^{\circ}$  或  $180^{\circ}$ (水平倒放), 則自動校準到  $0^{\circ}, 180^{\circ}$ 。適用於機器人安裝環境。其中, 「接近」定義為 Pitch Roll均小於 $15^{\circ}$
- `CONFIG ATT RST 5` 取消水平校平: 清除目前俯仰橫滾角校平設定(恢復預設)

執行 `CONFIG ATT RST` 命令時，模組需要保持靜止，如果模組在運動中執行該命令，則有可能造成較大校平誤差

## 手動設定航向角

使用者手動設定航向角: `CONFIG YAW <MODE> <VAL>` · MODE: 0:相對模式 1:絕對模式 · YAW: 設定值,範圍(-180 到 180,單位: deg) · 此命令立即生效 · 掉電不儲存。

- `CONFIG YAW 1 16` : 絕對設定模式: 將航向角設定為 16 deg
- `CONFIG YAW 0 1.5` : 相對設定模式 · 例: 如目前航向角為30 deg · 則設定后航向角變為 30 + 1.5= 31.5 deg

## 座標系旋轉(改變安裝方式)

`CONFIG IMU URFR C00,C01,C02,C10,C11,C12,C20,C21,C22`

其中  $C_{nm}$  支援浮點數

$$\begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix}_B = \begin{bmatrix} C00 & C01 & C02 \\ C10 & C11 & C12 \\ C20 & C21 & C22 \end{bmatrix} \cdot \begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix}_U$$

其中  $\begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix}_U$  為旋轉后的感測器座標系下感測器數據 ·  $\begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix}_B$  為旋轉前感測器座標系下感測器數據

下面是幾種常用旋轉舉例：

- 新感測器座標系為繞原座標系X軸旋轉-90°(Y軸正方向朝下的垂直安裝) · 配置命令：  
`CONFIG IMU URFR 1,0,0,0,1,0,-1,0`
- 新感測器座標系為繞原座標系X軸旋轉 90°(Y軸正方向朝上的垂直安裝) · 配置命令：  
`CONFIG IMU URFR 1,0,0,0,-1,0,1,0`
- 新感測器座標系為繞原座標系X軸旋轉180° · 配置命令：  
`CONFIG IMU URFR 1,0,0,0,-1,0,0,-1`
- 新感測器座標系為繞原座標系Y軸旋轉 90°(X軸正方向朝上的垂直安裝) · 配置命令：  
`CONFIG IMU URFR 0,0,-1,0,1,0,1,0`
- 新感測器座標系為繞原座標系Y軸旋轉-90°(X軸正方向朝下垂直安裝) · 配置命令：  
`CONFIG IMU URFR 0,0,1,0,1,0,-1,0`
- 新感測器座標系為繞原座標系Y軸旋轉180° · 配置命令：  
`CONFIG IMU URFR -1,0,0,0,1,0,0,-1`
- 新感測器座標系為繞原座標系Z軸旋轉90° · 配置命令：  
`CONFIG IMU URFR 0,-1,0,1,0,0,0,1`
- 新感測器座標系為繞原座標系Z軸旋轉-90° · 配置命令：  
`CONFIG IMU URFR 0,1,0,-1,0,0,0,1`
- 水平 · Z軸朝上安裝(預設值)：  
`CONFIG IMU URFR 1,0,0,0,1,0,0,1`

- 設定URFR后需要軟體復位或重新上電才能生效 · 不需要每次上電都發送該指令。
- 如何確定URFR參數:(以繞原座標系X軸旋轉-90°(Y軸正方向朝下的垂直安裝)為例) · 可寫出轉換後坐標與轉換前坐標的關係:

- $X_U = X_B$
- $Y_U = -Z_B$
- $Z_U = Y_B$

從而可以寫出轉換矩陣 =

1 0 0 (轉換后的X = 轉換前的X)  
0 0 -1(轉換后的Y = 轉換前的-Z)  
0 1 0(轉換后的Z = 轉換前的Y)

按照上述URFR定義公式 · URFR參數需要的實際上是上述矩陣的轉置 · 即:

1 0 0  
0 0 1  
0 -1 0

## 多功能IO複用功能配置

模組具有多個多功能引腳: IO1-IO9這些引腳可用作不同的功能，不同功能之間通過配置命令切換。

### CONFIG <PMUX> <IO>

- PMUX: 複用功能：PMUX1 - PMUX5
- IO: 引腳號: IO1-IO9

例：將IO2配置為PMUX3(LED): CONFIG PMUX3 IO2

複用功能編號	複用名稱	說明	預設IO
PMUX1	SIN	同步脈衝輸入(PPS_SYNC_PIN): 輸入引腳, 空閑狀態為高電平。當模組檢測到下降沿時, 會輸出一幀數據(如果為ONMARK觸發)。同時, 一些協議可以輸出PPS同步時間戳。PPS同步時間戳指的是: 當模組檢測到最近一次下降沿訊號時開始, 到目前一幀數據採樣時所經過的時間。該引腳不使用可懸空。	IO1
PMUX2	SOUT	同步輸出:無數據輸出時為低電平(空閑), 一幀數據開始發送時會發送一個高脈衝, 脈衝寬度位80us, 用於同步數據	IO2
PMUX3	LED	輸出, 執行狀態指示燈	IO5
PMUX4	SOUT_DIV	同步輸出分頻: 例:當為8分頻(預設)時: 數據輸出速率為100Hz, 則SOUT輸出頻率為100Hz, 則該引腳輸出頻率為12.5Hz(週期80ms), 用於觸發相機等操作	IO6
PMUX5	ALARM	此功能用於輸出報警訊號, 可以設定當X或Y角度超過一定範圍后觸發報警。詳見設定報警參數	IO9

- 並不是所有IO引腳在本產品上都引出, 具體參見產品使用者手冊硬體部分。
- 韌體版本需  $\geq 1.5.4$ 才支援此功能

預設出廠IO複用功能如下:

- IO1: PMUX1(SYNC\_IN)
- IO2: PMUX2(SOUT)
- IO5: PMUX3(LED)
- IO6: PMUX4(SOUT\_DIV)
- IO9: PMUX5(ALARM)

### 設定IO報警觸發參數(PMUX5: ALARM)

此功能用於輸出報警訊號, 可以設定當X或Y角度超過一定範圍后觸發報警。用於報警觸發功能。當報警時, 對應IO引腳輸出高電平, 否則輸出低電平。

- CONFIG PMUX5 AIXS,MIN,MAX,THR,ENABLE 設定角度報警參數
  - AIXS: 軸選擇: 『X』: X軸, 『Y』: Y軸
  - MIN, MAX: 安全區域範圍, 如 MAX= -5, MAX=5 則 -5° ~ 5°內不報警, 超出此範圍則報警。
  - THR: 保留值, 固定為0
  - ENABLE: 0:關閉,1:使能

例: CONFIG PMUX5 Y,-5,5,0,1: 使能Y軸報警功能, 當角度低於-5°或者高於5°時觸發報警。

### 設定SOUT\_DIV分頻係數(PMUX4: SOUT\_DIV)

例: CONFIG PMUX4 DIV4 : SOUT\_DIV輸出同步輸出的4分頻, 分頻係數支援 1-64

## 2.2.5. LOG

ENABLE/DISABLE: 全域性打開/關閉數據輸出

LOG ENABLE 全域性使能數據幀輸出(預設)

LOG DISABLE 全域性禁止數據幀輸出

## VERSION: 顯示模組版本資訊

**LOG VERSION** 列印韌體版本資訊

## USRCONFIG: 顯示使用者配置資訊

**LOG USRCONFIG** 列印使用者配置資訊，用於檢視配置是否被寫入成功。

```
1 | ATT_MODE: 0 /* 工作模式: 0:6軸 · 1:9軸 */  
2 | ...
```

## COMCONFIG: 顯示串列埠配置資訊

**LOG COMCONFIG** 列印串列埠及輸出協議配置資訊

### 配置數據幀輸出及設定輸出頻率

**LOG <MSG> <TYPE> <PERIOD>**

- MSG: IMU91, HI91(同IMU91), HI92
- TYPE: ONTIME: 定時輸出, ONMARK: 外部觸發同步輸出
- PERIOD: 輸出幀週期，單位為s, 取值範圍: 1(1Hz) · 0.5(2Hz) · 0.1(10Hz) · 0.02(50Hz) · 0.01(100Hz), 0.005(200Hz), 0.002(500Hz),以此類推

例(定時100Hz輸出):

- **LOG IMU91 ONTIME 0.01** 將目前串列埠的 91數據包輸出週期設定為0.01s(100Hz)
- **LOG HI92 ONTIME 0.05** 將目前串列埠的 92數據包輸出週期設定為0.05s(20Hz)

例(關閉輸出):

- **LOG IMU91 ONTIME 0** 關閉91數據包輸出

當輸出幀率設定為比較高時(比如500Hz),預設的115200波特率不滿足輸出頻寬要求，此時需要將模組波特率設高(比如921600)后，模組才能正確的輸出數據。  
波特率參數設定好后掉電儲存，復位模組生效。上位機或者其他主機的波特率也要做相應修改。

### 配置為同步輸出模式

同步觸發輸出:

- **UNLOGALL** (可選)先取消所有數據輸出。
- **LOG IMU91 ONMARK 1** 將目前串列埠的 HI91數據包設定為同步觸發模式，設定同步觸發模式后，模組收到SYNC\_IN引腳的脈衝訊號或收到LOG <MSG> ONMARK ONCE指令後會輸出一幀數據。
- **LOG IMU91 ONMARK ONCE** (可選)手動觸發一次數據輸出，和SYNC\_IN脈衝效果相同

### 2.2.6. UNLOGALL

將所有定時輸出訊息輸出頻率設為0(不輸出)

### 2.2.7. FRESET

恢復出廠設定

## 2.3. 更多高級配置

除了以上常用配置外，模組還可以配置一些高級效能參數，這些參數一般情況不需要修改，出廠前已經調節到最優。

### 2.3.1. 配置加速度和陀螺儀頻寬

**CONFIG IMU ABW <VAL>** 調整加速度計頻寬 VAL: 0=5Hz, 1=10Hz, 2=20Hz, 3=40Hz, 4=80Hz, 5=125Hz, 6=230Hz, 預設)

**CONFIG IMU GBW <VAL>** 調整陀螺儀頻寬 VAL: 0 =12Hz, 3=47Hz, 4=80Hz, 5=116Hz(預設), 6=230Hz

例: 將陀螺儀濾波器3db截至頻寬調整為47Hz: CONFIG IMU GBW 3

- 不要配置為手冊上未標註值, 否則可能造成輸出數據異常, 改變加速度計和陀螺的頻寬並不能帶來精度的提升, 如為提高精度, 不建議修改。
- 為了減小靜態噪聲或者振動帶來的影響可以去適當降低加速度計頻寬。
- 調低頻寬的代價是輸出數據相位延遲更大, 在一些要求實時高動態的應用中不應將頻寬調的過低。

### 3. 數據輸出協議(自定義二進制協議)

該協議是超覈定義的二進制協議，可以輸出全部感測器資訊，支援該協議的介面:RS-232/TTL/USB(虛擬串列埠)。預設串列埠格式為N-8-N-1(8位數據位，1位停止位，0位校驗位)

#### 3.1. 數據幀格式

模組上電后，按照預設幀率(100Hz)輸出幀數據，幀格式如下：

域名稱	值	長度(位元組)	說明
幀頭	0x5A	1	0x5A
幀型別	0xA5	1	0xA5
數據域長度	1-512	2	幀中數據域的長度，LSB(低位元組在前) 長度表示數據域的長度(不包含幀頭，幀型別，長度，CRC)
CRC校驗	-	2	除CRC位元組外其餘所有欄位(幀頭,幀型別,長度,數據域)的16位CRC校驗和。 LSB(低位元組在前)
數據域	-	1-512	一幀攜帶的數據，由若干個子數據包組成，數據包包含數據包標籤和數據兩部分。 標籤決定了數據的型別及長度。

#### 3.2. 出廠預設輸出

出廠預設輸出: 浮點型IMU數據幀(0x91)

#### 3.3. 數據域內容

##### 3.3.1. 浮點型IMU數據幀(0x91)

數據域共76位元組。包含模組ID、溫度、IMU的原始數據、地磁、氣壓、融合后的姿態數據等。

開啟數據幀示例: LOG HI91 ONTIME 1, 具體詳見配置指令章節

位元組偏移	名稱	數據型別	大小(Byte)	單位	比例因子	說明
0	tag	uint8_t	1	-	-	數據包標籤:0x91
1	pps_sync_stamp	uint16_t	2	ms	1	PPS脈衝同步時間戳，用於精確時間同步，該值定義為：最近一次檢測到PPS同步引腳脈衝沿到本幀數據採樣時刻所經過的時間，範圍 0-8192,當超過最大值時會自動回滾到0，例：如果使用者在PPS引腳上輸入的是標準秒脈衝，則該值會在0-1000之間。
3	temperature	int8_t	1	°C	1	模組平均溫度
4	air_pressure	float	4	Pa	1	氣壓
8	system_time	uint32_t	4	ms	1	節點本地時間戳資訊，從系統開機開始累加，每毫秒增加1
12	acc_b	float	4*3	G	1	經過出廠校準后的加速度,順序為：XYZ軸。 1G=1x當地重力加速度,可近似為9.8 m/s^2
24	gyr_b	float	4*3	deg/s(dps)	1	經過出廠校準后的角速度,順序為：XYZ軸
36	mag_b	float	4*3	uT	1	磁強度,順序為：XYZ軸
48	roll	float	4	deg	1	橫滾角
52	pitch	float	4	deg	1	俯仰角
56	yaw	float	4	deg	1	航向角
60	quat	float	4*4	-	-	節點四元數集合,順序為WXYZ

### 3.3.2. 整型IMU數據幀(0x92)

共48位元組，比浮點型數據幀數據量更小。開啟數據幀示例: LOG HI92 ONTIME 1, 具體詳見配置指令章節

位元組偏移	名稱	數據型別	大小 (Byte)	單位	比例因子	說明
0	tag	uint8_t	1	-	-	數據包標籤:0x92
1	status	uint16_t	2	-	-	狀態字，保留
3	temperature	int8_t	1	°C	1	系統平均溫度
4	pps_sync_stamp	uint16_t	2	ms	1	PPS脈衝同步時間戳，用於精確時間同步，該值定義為：最近一次檢測到PPS同步引腳脈衝沿到本幀數據採樣時刻所經過的時間，範圍 0-8192,當超過最大值時會自動回滾到0。例：如果使用者在PPS引腳上輸入的是標準秒脈衝，則該值會在0-1000之間。
6	air_pressure	int16_t	2	Pa	1	大氣壓+100000Pa：如2000表示 102000Pa
8	reserved	-	2	-	-	保留
10	gyr_b	int16_t*3	6	rad/s	0.001	IMU角速度: X,Y,Z軸(出廠校準后)
16	acc_b	int16_t*3	6	m/s^(2)	0.0048828	IMU加速度: X,Y,Z軸(出廠校準后)
22	mag_b	int16_t*3	6	uT	0.030517	IMU磁強度: X,Y,Z軸(出廠校準后)
28	roll	int32_t	4	deg	0.001	橫滾角
32	pitch	int32_t	4	deg	0.001	俯仰角
36	yaw	int32_t	4	deg	0.001	航向角
40	quat	int16_t*4	8	-	0.0001	節點四元數集合,順序為WXYZ

### 3.4. CRC

16-bit CRC實現例程：

```

1  /*
2     correctCrc: previous crc value, set 0 if it's first section
3     src: source stream data
4     lengthInBytes: length
5  */
6  static void crc16_update(uint16_t *correctCrc, const uint8_t *src, uint32_t lengthInBytes)
7  {
8     uint32_t crc = *correctCrc;
9     uint32_t j;
10    for (j=0; j < lengthInBytes; ++j)
11    {
12        uint32_t i;
13        uint32_t byte = src[j];
14        crc ^= byte << 8;
15        for (i = 0; i < 8; ++i)
16        {
17            uint32_t temp = crc << 1;
18            if (crc & 0x8000)
19            {
20                temp ^= 0x1021;
21            }
22            crc = temp;
23        }
24    }
25    *correctCrc = crc;
26 }

```

### 3.5. 數據幀結構示例

使用串列埠助手採樣一幀數據,共82位元組,前6位元組為幀頭,長度和CRC校驗值。剩餘76位元組為數據域。假設數據接收到C語言陣列 `buf` 中。如下所示:

```
5A A5 4C 00 6C 51 91 00 A0 3B 01 A8 02 97 BD BB 04 00 9C A0 65 3E A2 26 45 3F 5C E7 30 3F E2 D4 5A C2 E5 9D A0 C1 EB 23 EE C2 78 77 99 41 AB AA D1 C1 AB
2A 0A C2 8D E1 42 42 8F 1D A8 C1 1E 0C 36 C2 E6 E5 5A 3F C1 94 9E 3E B8 C0 9E BE BE DF 8D BE
```

- 1. 判斷幀頭,得到數據域長度和CRC校驗值:

幀頭:5A A5

幀數據域長度:4C 00:  $(0x00 \ll 8) + 0x4C = 76$

CRC校驗值:6C 51:  $(0x51 \ll 8) + 0x6C = 0x516C$

- 2. 校驗CRC

```
1  uint16_t payload_len;
2  uint16_t crc;
3  crc = 0;
4  payload_len = buf[2] + (buf[3] << 8);
5
6  /* calculate 5A A5 and LEN field crc */
7  crc16_update(&crc, buf, 4);
8
9  /* calculate payload crc */
10 crc16_update(&crc, buf + 6, payload_len);
```

得到CRC值為0x516C,與幀中攜帶CRC值相同。CRC校驗通過。

- 3. 接收數據

從 `0x91` 開始為數據包的數據域,定義數據結構體和常用轉換宏:

```
1  #include "stdio.h"
2  #include "string.h"
3  /* common type conversion */
4  #define U1(p) (*(uint8_t*)(p))
5  #define I1(p) (*(int8_t*)(p))
6  #define I2(p) (*(int16_t*)(p))
7  static uint16_t U2(uint8_t *p) {uint16_t u; memcpy(&u,p,2); return u;}
8  static uint32_t U4(uint8_t *p) {uint32_t u; memcpy(&u,p,4); return u;}
9  static int32_t I4(uint8_t *p) {int32_t u; memcpy(&u,p,4); return u;}
10 static float R4(uint8_t *p) {float r; memcpy(&r,p,4); return r;}
11 typedef struct
12 {
13     uint8_t tag; /* item tag: 0x91 */
14     float acc[3]; /* acceleration */
15     float gyr[3]; /* angular velocity */
16     float mag[3]; /* magnetic field */
17     float eu1[3]; /* attitude: eular angle */
18     float quat[4]; /* attitude: quaternion */
19     float pressure; /* air pressure */
20     uint32_t timestamp;
21 }imu_data_t;
```

接收數據,從 `buf[6]=0x91` 開始為payload部分:

```
1  imu_data_t i0x91 = {0};
2  int offset = 6; /* payload strat at buf[6] */
3  i0x91.tag = U1(buf+offset+0);
4  i0x91.pressure = R4(buf+offset+4);
5  i0x91.timestamp = U4(buf+offset+8);
6  i0x91.acc[0] = R4(buf+offset+12);
7  i0x91.acc[1] = R4(buf+offset+16);
8  i0x91.acc[2] = R4(buf+offset+20);
9  i0x91.gyr[0] = R4(buf+offset+24);
10 i0x91.gyr[1] = R4(buf+offset+28);
11 i0x91.gyr[2] = R4(buf+offset+32);
```

```

12 i0x91.mag[0] = R4(buf+offset+36);
13 i0x91.mag[1] = R4(buf+offset+40);
14 i0x91.mag[2] = R4(buf+offset+44);
15 i0x91.eu1[0] = R4(buf+offset+48);
16 i0x91.eu1[1] = R4(buf+offset+52);
17 i0x91.eu1[2] = R4(buf+offset+56);
18 i0x91.quat[0] = R4(buf+offset+60);
19 i0x91.quat[1] = R4(buf+offset+64);
20 i0x91.quat[2] = R4(buf+offset+68);
21 i0x91.quat[3] = R4(buf+offset+72);

```

列印接收到的數據:

```

1 printf("%-16s0x%X\r\n", "tag:", i0x91.tag);
2 printf("%-16s%8.4f %8.4f %8.4f\r\n", "acc(G):", i0x91.acc[0], i0x91.acc[1],
i0x91.acc[2]);
3 printf("%-16s%8.3f %8.3f %8.3f\r\n", "gyr(deg/s):", i0x91.gyr[0], i0x91.gyr[1],
i0x91.gyr[2]);
4 printf("%-16s%8.3f %8.3f %8.3f\r\n", "mag(uT):", i0x91.mag[0], i0x91.mag[1],
i0x91.mag[2]);
5 printf("%-16s%8.3f %8.3f %8.3f\r\n", "eu1(deg):", i0x91.eu1[0], i0x91.eu1[1],
i0x91.eu1[2]);
6 printf("%-16s%8.3f %8.3f %8.3f %8.3f\r\n", "quat:", i0x91.quat[0], i0x91.quat[1],
i0x91.quat[2], i0x91.quat[3]);
7 printf("%-16s%8.3f\r\n", "pressure(pa):", i0x91.pressure);
8 printf("%-16s%d\r\n", "timestamp(ms):", i0x91.timestamp);

```

列印出的解析結果:

```

1 tag: 0x91
2 acc(G): 0.2242 0.7701 0.6910
3 gyr(deg/s): -54.708 -20.077 -119.070
4 mag(uT): 19.183 -26.208 -34.542
5 eu1(deg): 48.720 -21.014 -45.512
6 quat: 0.855 0.310 -0.310 -0.277
7 pressure(pa): -0.000
8 timestamp(ms): 310205

```

### 3.6. 最大傳輸速率

協議	位元組數	9600bps	115200bps	230400bps	256000bps	460800bps	921600bps
91	76	10Hz	100Hz	250Hz	250Hz	500Hz	1000Hz
92	48	10Hz	200Hz	250Hz	250Hz	500Hz	1000Hz

## 4. RS-485數據協議與指令(Modbus協議)

- 支援該協議的介面: RS485
- 預設串列埠配置: 115200 - N8N1
- Modbus 指令：
  - RS485通訊協議遵循Modbus RTU協議規範，數據以暫存器為單位進行發送和接收，每個暫存器佔用2個位元組，採用大端模式(高位元組在前)
  - 寫入: 0x06 (Write Single Register) :寫單個暫存器(每個Modbus暫存器為2個位元組)
  - 讀取: 0x03 (Read Holding Registers): 讀取單個或多個暫存器數據
  - 自定義功能碼: 0x50, 用於 Modbus ID自動分配，方便量產部署，韌體升級等
- Modbus裝置地址可修改, 出廠預設: 80 (0x50)

### 4.1. 數據幀格式

#### 4.1.1. 讀暫存器(0x03)

主機發送:

域名稱	值	說明
ID	1-0xFF	Modbus裝置地址
FUN_CODE	0x03	命令碼
ADDR_H	-	要讀取的暫存器地址高8位
ADDR_L	-	要讀取的暫存器地址低8位
LEN_H	-	要讀取暫存器長度高8位(以暫存器個數為單位)
LEN_L	-	要讀取暫存器長度低8位(以暫存器個數為單位)
CRC_L	-	CRC低8位
CRC_H	-	CRC高8位

從機(模組)返回:

域名稱	值	說明
ID	1-0xFF	Modbus裝置地址
FUN_CODE	0x03	命令碼
LEN	-	返回暫存器數據的長度(不算ID·FUN_CODE,LEN,CRC欄位)以位元組為單位
DATAH	-	返回數據高8位
DATAL	-	返回數據低8位
----	-	返回數據高8位
----	-	返回數據低8位
CRC_L	-	CRC低8位
CRC_H	-	CRC高8位

#### 4.1.2. 寫暫存器(0x06)

域名稱	值	說明
ID	1-0xFF	Modbus裝置地址
FUN_CODE	0x06	命令碼
ADDR_H	-	暫存器地址高8位
ADDR_L	-	暫存器地址低8位
DATA_H	-	寫入數據高8位
DATA_L	-	寫入數據低8位
CRC_L	-	CRC低8位
CRC_H	-	CRC高8位

從機返回:

域名稱	值	說明
ID	1-0xFF	Modbus裝置地址
FUN_CODE	0x06	命令碼
ADDR_H	-	暫存器地址高8位
ADDR_L	-	暫存器地址低8位
DATA_H	-	寫入數據高8位
DATA_L	-	寫入數據低8位
CRC_L	-	CRC低8位
CRC_H	-	CRC高8位

#### 4.1.3. CRC校驗

- 線上計算CRC : <https://www.23bei.com/tool/59.html>
- C程式碼 :

```

1 static const uint16_t modbus_crc_table[256] = {
2     0x0000, 0xc0c1, 0xc181, 0x0140, 0xc301, 0x03c0, 0x0280, 0xc241,
3     0xc601, 0x06c0, 0x0780, 0xc741, 0x0500, 0xc5c1, 0xc481, 0x0440,
4     0xcc01, 0x0cc0, 0xd80, 0xcd41, 0xf00, 0xfc1, 0xe81, 0xe40,
5     0xa00, 0xac1, 0xb81, 0xb40, 0x901, 0x9c0, 0x880, 0x841,
6     0xd801, 0x18c0, 0x1980, 0xd941, 0x1b00, 0x1bc1, 0xda81, 0x1a40,
7     0x1e00, 0x1dec1, 0xdf81, 0x1f40, 0xdd01, 0xdc0, 0x1c80, 0xdc41,
8     0x1400, 0xd4c1, 0xd581, 0x1540, 0xd701, 0x17c0, 0x1680, 0xd641,
9     0xd201, 0x12c0, 0x1380, 0xd341, 0x1100, 0xd1c1, 0xd081, 0x1040,
10    0xf001, 0x30c0, 0x3180, 0xf141, 0x3300, 0xf3c1, 0xf281, 0x3240,
11    0x3600, 0xf6c1, 0xf781, 0x3740, 0xf501, 0x35c0, 0x3480, 0xf441,
12    0x3c00, 0xfcc1, 0xfd81, 0x3d40, 0xff01, 0x3fc0, 0x3e80, 0xfe41,
13    0xfa01, 0x3ac0, 0x3b80, 0xfb41, 0x3900, 0xf9c1, 0xf881, 0x3840,
14    0x2800, 0xe8c1, 0xe981, 0x2940, 0xeb01, 0x2bc0, 0x2a80, 0xea41,
15    0xee01, 0x2ec0, 0x2f80, 0xef41, 0x2d00, 0xedc1, 0xec81, 0x2c40,
16    0xe401, 0x24c0, 0x2580, 0xe541, 0x2700, 0xe7c1, 0xe681, 0x2640,
17    0x2200, 0xe2c1, 0xe381, 0x2340, 0xe101, 0x21c0, 0x2080, 0xe041,
18    0xa001, 0x60c0, 0x6180, 0xa141, 0x6300, 0xa3c1, 0xa281, 0x6240,
19    0x6600, 0xa6c1, 0xa781, 0x6740, 0xa501, 0xa5c0, 0xa480, 0xa441,
20    0x6c00, 0xacc1, 0xad81, 0x6d40, 0xaf01, 0x6fc0, 0x6e80, 0xae41,
21    0xaa01, 0x6ac0, 0x6b80, 0xab41, 0x6900, 0xa9c1, 0xa881, 0x6840,
22    0x7800, 0xb8c1, 0xb981, 0x7940, 0xbb01, 0x7bc0, 0x7a80, 0xba41,
23    0xbe01, 0x7ec0, 0x7f80, 0xbf41, 0x7d00, 0xbdc1, 0xbc81, 0x7c40,
24    0xb401, 0x74c0, 0x7580, 0xb541, 0x7700, 0xb7c1, 0xb681, 0x7640,
25    0x7200, 0xb2c1, 0xb381, 0x7340, 0xb101, 0xb1c0, 0x7080, 0xb041,
26    0x5000, 0x90c1, 0x9181, 0x5140, 0x9301, 0x53c0, 0x5280, 0x9241,
27    0x9601, 0x56c0, 0x5780, 0x9741, 0x5500, 0x95c1, 0x9481, 0x5440,
28    0x9c01, 0x5cc0, 0x5d80, 0x9d41, 0x5f00, 0x9fc1, 0x9e81, 0x5e40,

```

```

29     0x5a00, 0x9ac1, 0x9b81, 0x5b40, 0x9901, 0x59c0, 0x5880, 0x9841,
30     0x8801, 0x48c0, 0x4980, 0x8941, 0x4b00, 0x8bc1, 0x8a81, 0x4a40,
31     0x4e00, 0x8ec1, 0x8f81, 0x4f40, 0x8d01, 0x4dc0, 0x4c80, 0x8c41,
32     0x4400, 0x84c1, 0x8581, 0x4540, 0x8701, 0x47c0, 0x4680, 0x8641,
33     0x8201, 0x42c0, 0x4380, 0x8341, 0x4100, 0x81c1, 0x8081, 0x4040
34 };
35
36 uint16_t modbus_crc_calc(uint8_t *buf, uint16_t len)
37 {
38     uint16_t crc = 0xFFFFU;
39     uint8_t nTemp;
40
41     while (len--)
42     {
43         nTemp = *buf++ ^ crc;
44         crc >>= 8;
45         crc ^= modbus_crc_table[(nTemp & 0xFFU)];
46     }
47
48     return(crc);
49 }

```

## 4.2. 暫存器列表

地址 (Hex)	地址 (Dec)	名稱	功能	R/W	說明
0x00	0	CTRL	控制	W	參見Modbus 設定模組章節
0x04	4	UART1_BAUD	波特率	R/W	串列埠波特率
0x05	5	MD_ID	Modbus ID	R/W	Modbus ID: 有效範圍: 1-128
0x06	6	HEADING_MODE	航向角模式	R/W	0: 6軸模式(相對航向 · 航向角上電為0). 1: 9軸模式(地磁融合 · 絕對航向)
0x34	52	ACCX	加速度X	R	單位G(1G=1重力加速度), 比例因子: 0.00048828
0x35	53	ACCY	加速度Y	R	單位G(1G=1重力加速度), 比例因子: 0.00048828
0x36	54	ACCZ	加速度Z	R	單位G(1G=1重力加速度), 比例因子: 0.00048828
0x37	55	GYRX	角速度X	R	單位deg/s, 比例因子: 0.061035
0x38	56	GYRY	角速度Y	R	單位deg/s, 比例因子: 0.061035
0x39	57	GYRZ	角速度Z	R	單位deg/s, 比例因子: 0.061035
0x3A	58	MAGX	磁強度X	R	單位uT, 比例因子: 0.030517
0x3B	59	MAGY	磁強度Y	R	單位uT, 比例因子: 0.030517
0x3C	60	MAGZ	磁強度Z	R	單位uT, 比例因子: 0.030517
0x3D	61	R_H	橫滾角高 16位	R	單位deg, 比例因子: 0.001
0x3E	62	R_L	橫滾角低 16位	R	單位deg, 比例因子: 0.001
0x3F	63	P_H	俯仰角高 16位	R	單位deg, 比例因子: 0.001
0x40	64	P_L	俯仰角低 16位	R	單位deg, 比例因子: 0.001
0x41	65	Y_H	航向角高 16位	R	單位deg, 比例因子: 0.001
0x42	66	Y_L	航向角低 16位	R	單位deg, 比例因子: 0.001
0x43	67	TEMP	溫度	R	單位°C, 比例因子: 0.01
0x44	68	PRS_H	氣壓高16 位	R	單位Pa, 比例因子: 0.01
0x45	69	PRS_L	氣壓低16 位	R	單位Pa, 比例因子: 0.01
0x46	70	Q0	四元數 QW	R	四元數 · 比例因子: 0.0001
0x47	71	Q1	四元數QX	R	四元數 · 比例因子: 0.0001
0x48	72	Q2	四元數QY	R	四元數 · 比例因子: 0.0001
0x49	73	Q3	四元數QZ	R	四元數 · 比例因子: 0.0001
0x4A	74	SA_X	傾角儀X 軸角度	R	雙軸傾角儀產品: X角度, ±180 · 單位deg, 比例因子: 0.005493 單軸傾角儀產品: X角度, 0-360, 單位deg, 比例因子: 0.005493
0x4B	75	SA_Y	傾角儀Y 軸角度	R	雙軸傾角儀Y角度 · ±90, 單位deg, 比例因子: 0.005493
0x70- 0x77	112- 119	PNAME	裝置名	R	裝置名字串, ASCII碼 · 共佔8個暫存器
0x78	120	SW_VERSION	軟體版本	R	軟體版本
0x79	121	BL_VERSION	BL版本	R	BL版本

地址 (Hex)	地址 (Dec)	名稱	功能	R/W	說明
0x7F- 0x82	127- 130	SN	產品唯一 序列號	R	產品唯一序列號, 占4暫存器
0xA5	165	SET_LV	自動校平	W	3: 執行一次自動調平: 如果目前俯仰角/橫滾角接近0°,0°(水平正面放置), 則自動校準到0,0. 前如果目前俯仰角/橫滾角接近0° 或 180°(水平倒放) · 則自動校準到 0°,180° 適用於機器人安裝環境。其中, 「接近」定義為 Pitch Roll均小於15° 5: 取消自動調平 · 恢復絕對測量角度 其他值: 無效
0xA6	166	URFR	安裝設定	W	0: 設定為水平安裝(預設模式) 1: 垂直安裝:Y軸正方向朝下 2: 垂直安裝:Y軸正方向朝上 3: 垂直安裝:X軸正方向朝上 4: 垂直安裝:X軸正方向朝下

### 4.3. 常用配置

以下所有配置示例預設Modbus地址為0x50(出廠預設), 如果Modbus ID已經被使用者修改, 則ID欄位和CRC欄位需要更改。

#### 4.3.1. 控制暫存器說明(0x00)

命令	CTL暫存器寫入值	命令(Hex) ID=0X50(出廠預設)
儲存所有配置參數到Flash	0x0000	50 06 00 00 00 00 84 4B
恢復出廠設定	0x0001	50 06 00 00 00 01 45 8B
復位	0x00FF	50 06 00 00 00 FF C4 0B

#### 4.3.2. 配置波特率(0x04)

配置目標波特率	指令(Hex) ID=0X50(出廠預設)
4800	50 06 00 04 00 00 C5 8A
9600	50 06 00 04 00 01 04 4A
19200	50 06 00 04 00 02 44 4B
38400	50 06 00 04 00 03 85 8B
57600	50 06 00 04 00 04 C4 49
115200	50 06 00 04 00 05 05 89
230400	50 06 00 04 00 06 45 88
460800	50 06 00 04 00 07 84 48
921600	50 06 00 04 00 08 C4 4C

#### 4.3.3. 設定安裝方式 ( 0xA6 )

配置目標安裝方式	指令(Hex) ID=0X50(出廠預設)
0: 設定為水平安裝(預設模式)	50 06 00 A6 00 00 64 68
1: 垂直安裝:y軸正方向朝下	50 06 00 A6 00 01 A5 A8
2: 垂直安裝:y軸正方向朝上	50 06 00 A6 00 02 E5 A9
3: 垂直安裝:x軸正方向朝上	50 06 00 A6 00 03 24 69
4: 垂直安裝:x軸正方向朝下	50 06 00 A6 00 04 65 AB

#### 4.3.4. 設定水平校準(0xA5)

- 啟動自動水平校平: 50 06 00 A5 00 03 D4 69
- 取消自動水平校平: 50 06 00 A5 00 05 54 6B

#### 4.3.5. 設定6軸或9軸模式

- 設定為6軸模式: 50 06 00 06 00 00 64 4A
- 設定為9軸模式: 50 06 00 06 00 01 A5 8A

### 4.4. 讀取模組版本資訊

1.讀取模組產品名, 軟體版本及SN號

**TX**(主機發送): 50 03 00 70 00 14 49 9F

ID=0x50, CMD=0x03, 讀取起始地址0x70, 讀取長度:0x14, CRC: 0x499F

**RX**(從機響應): 50 03 28 48 49 31 34 52 32 4E 2D 34 38 35 2D 30 30 30 00 00 98 00 6B 00 00 00 00 00 00 00 04 7D 95 5F 8D 2A 17 08 00 00 4D 0C

50 03 28: 從機地址0x50, 命令碼:0x03, 數據部分共0x28= 40位元組, 48 49 31 34 52 32 4E 2D 34 38 35 2D 30 30 30 00 00 98 00 6B 00 00 00 00 00 00 00 00 04 7D 95 5F 8D 2A 17 08 00 00, 數據段: 產品名: CH10x(M), 軟體版本:0x98(152), BL版本:0x6B(107), SN:047D955F8D2A1708, 4D 0C: CRC校驗

## 4.5. 讀取感測器數據

TX(主機發送): 50 03 00 34 00 18 09 8F

ID=0x50, CMD=0x03, 讀取起始地址0x34, 讀取長度:0x18, CRC: 0x098F

RX(從機響應): 50 03 30 FF 01 03 B0 06 50 FC C9 FF 7C 00 91 01 D5 FD DB FD 27 00 00 21 FF 00 00 7F F6 FF FD 73 E7 00 00 00 00 00 10 A6 0D 59 DD 4E 86 A8 06 30 17 82 1E CE

0x30 = 從機返回48位元組 · 第一個暫存器值為0xFF01=-255, 第二暫存器值為0x03B0=944 ...以此類推

- 加速度: X=-255, Y=944, Z=1616 ->乘比例因子后結果: X= -0.1245, Y=0.4609, Z=0.7891G (1G=1重力加速度 · 可取9.8m/s^(2))
- 角速度: X=-823, Y=-132, Z=145 -> 乘比例因子后結果: X=-50.2318, Y=-8.0566, Z=8.8501deg/s
- 磁場: X=469, Y=-549, Z=-729 -> 乘比例因子后結果: X=14.3125, Y=-16.7538, Z=-22.2469uT
- 歐拉角: 橫滾(Roll)=8703, 俯仰角(Pitch)=32758, 航向角(Yaw)=-166937 -> 乘比例因子后結果: Roll= 8.703deg, Pitch=32.758deg, Yaw=-166.937deg

## 4.6. Modbus ID自動分配

ID地址自動分配機制用於量產部署時多個模組掛接在同一個485匯流排的情況 · 模組可配合上位機完成ID自動地址分配 · 該功能只開放給量產客戶 · 具體資料請與我們聯繫。

0x50 自定義指令格式為: [ADDR] 0x50 [SUB\_CMD] [DATA\_LEN] [DATA]

目前支援的自定義指令列表:

1. 通過SN號設定ID地址(0x0031): 格式: 00 50 00 31 00 0A [SN] [NEW\_ADDR] CRC  
SN: 裝置唯一序列號,8位元組  
新的ID地址: 1-255, 2位元組
2. ID地址隨機化產生(0x0030): 該指令會強制匯流排上的所有模組拋棄原來ID且產生新的介於MIN\_ADDR和MAX\_ADDR之間的新ID 格式: 00 50 00 30 00 06 [MIN] [MAX] FF FF CRC  
MIN: 產生隨機ID的最小值, 2位元組。  
MAX: 產生隨機ID的最大值 2位元組。

## 5. CAN數據協議與指令(CANopen協議)

CAN介面符合CANopen協議，所有通訊均使用標準數據幀，使用TPDO1-7 傳輸數據。不接收/發送遠端幀和拓展數據幀，所有TPDO採用非同步定時觸發模式。

### 5.1. CANopen 預設設定

預設配置	值
CAN 波特率	500KHz
CANopen節點ID	8
初始化狀態	Operational
心跳包	無
TPDO輸出速率	1Hz - 200Hz(每個TPDO)

### 5.2. CANopen TPDO

通道	幀ID	數據長度 ( DLC )	傳輸方式	輸出頻率 ( Hz )	數據	說明
TPDO1	0x180+ID	6	非同步定時 (0xFE)	100	加速度	型別:int16 低位元組在前，每個軸2位元組，共6位元組分別為X,Y,Z軸加速度，單位為mG(0.001G)
TPDO2	0x280+ID	6	非同步定時 (0xFE)	100	角速度	型別:int16 低位元組在前，每個軸2位元組，共6位元組分別為X,Y,Z軸角速度，單位為0.1dps(°/s)
TPDO3	0x380+ID	6	非同步定時 (0xFE)	100	歐拉角	型別:int16 低位元組在前，每個軸2位元組，共6位元組順序分別為 橫滾角:Roll, 俯仰角:Pitch, 航向角:Yaw。單位為0.01°
TPDO4	0x480+ID	8	非同步定時 (0xFE)	100	四元數	型別:int16 低位元組在前,每個元素2位元組，共8位元組分別為 $q_w$ $q_x$ $q_y$ $q_z$ 。單位四元數擴大10000倍后結果。如四元數為1,0,0,0時，輸出10000,0,0,0.
TPDO6	0x680+ID	4	非同步定時 (0xFE)	20	氣壓	型別:int32 共4位元組。單位Pa
TPDO7	0x780+ID	8	非同步定時 (0xFE)	100	傾角儀角度	型別:int32 低位元組在前，每個軸4位元組，共8位元組順序分別為 X軸，Y軸。單位為0.01°

以加速度和角速度為例解析數據

加速度CAN幀：ID=0x188，DATA = 4A 00 1F 00 C8 03

- ID=0x188: ID為8的裝置發送的加速度數據幀
- 加速度X軸 = 0x004A = 74 = 74mG
- 加速度Y軸 = 0x001F = 731 = 31mG
- 加速度Z軸 = 0x03C8 = 968 = 968mG

角速度CAN幀：ID=0x288，DATA = 15 00 14 01 34 00

- ID=0x288: ID為8的裝置發送的角速度數據幀
- 角速度X軸 = 0x0015 = 21 = 2.1dps
- 角速度Y軸 = 0x0114 = 276 = 27.6dps
- 角速度Z軸 = 0x0034 = 52 = 5.2dps

### 5.3. 使用上位機連線CAN裝置

使用PCAN-View工具，配合PCAN，可以在接收框(Rx Message)中顯示收到的CAN訊息及幀率，如下圖所示:

CAN-ID	Type	Length	Data	Cycle Time ^	Count
688h		4	00 00 00 00	102.6	27
488h		8	E0 26 FB 02 0E 02 1A 01	10.2	270
388h		6	48 02 7B 03 17 01	10.1	270
288h		6	00 00 00 00 00 00	10.1	270
188h		6	9B FF 94 00 BD 03	10.2	270

## 5.4. 配置指令

所有配置更改后需要發送儲存配置指令才能儲存到Flash

### 5.4.1. LLS(Layer Setting Services)協議

- 主機發送LLS命令到從機: `ID = 0x07E5, DATA = CS,00,00,00,00,00,00,00` 其中CS為命令欄位 · DATA為數據欄位
- 從機回覆LLS命令到主機: `ID = 0x07E4, DATA = CS, ECODE, 00, 00, 00, 00, 00, 00` 其中ECODE為錯誤碼 · 0表示無錯誤。

#### 修改波特率(重新上電生效)

- CAN波特率修改為1000 kbit/s: `ID=0x07E5 · DATA=13,00,00,00,00,00,00,00`
- CAN波特率修改為800 kbit/s: `ID=0x07E5 · DATA=13,00,01,00,00,00,00,00`
- CAN波特率修改為500 kbit/s: `ID=0x07E5 · DATA=13,00,02,00,00,00,00,00`
- CAN波特率修改為250 kbit/s: `ID=0x07E5 · DATA=13,00,03,00,00,00,00,00`
- CAN波特率修改為125 kbit/s: `ID=0x07E5 · DATA=13,00,04,00,00,00,00,00`
- CAN波特率修改為100 kbit/s: `ID=0x07E5 · DATA=13,00,05,00,00,00,00,00`

#### 修改節點ID

`ID=0x07E5 · DATA=11,ID,00,00,00,00,00,00`

ID修改範圍：1-64 · 生效后發送啟動節點命令(比如節點啟動命令數據變為01 09)和SDO指令(發送CAN幀ID變為0x609時注意為新的地址)

#### 儲存配置

`ID=0x07E5 · DATA=17,00,00,00,00,00,00,00`

所有配置更改后需要發送儲存配置指令才能儲存到Flash

### 5.4.2. NMT(Network Management)協議

#### 全域性使能/關閉數據輸出(開啟非同步觸發)

使用CANopen NMT協議幀:

- 使能全域性數據輸出 `ID=0x00 · DATA=01 08`
- 關閉全域性數據輸出 `ID=0x00 · DATA=02 08`

#### 復位

- 復位節點 `ID=0x00 · DATA=81 08`

### 5.4.3. SDO(Service Data Object)協議

SDO格式:

- 主機發送SDO命令到從機:

CAN_ID	CS命令符(1B)	數據字典索引(2B)	子索引(1B)	數據(4B)
0x600+ID	0x23(寫4B)	低位在前	子索引	數據 · 低位在前

- 從機回覆SDO命令到主機:

CAN_ID	SDO命令(1B)	數據字典索引(2B)	子索引(1B)	數據(4B)
0x580+ID	0x60(寫成功應答)	低位在前	子索引	數據·低位在前

以下配置操作均使用快速SDO來寫數據字典, 其中TPDO通道與其對應的參數索引為:

通道	幀ID	參數索引地址	說明
TPDO1	0x180+ID	0x1800	加速度
TPDO2	0x280+ID	0x1801	角速度
TPDO3	0x380+ID	0x1802	歐拉角
TPDO4	0x480+ID	0x1803	四元數
TPDO6	0x680+ID	0x1804	氣壓
TPDO7	0x780+ID	0x1805	傾角儀輸出

### 修改/關閉/開啟數據輸出速率

此項配置立即生效

- ID=0x608 · DATA=23,00,18,05,00,00,00,00 關閉加速度輸出
- ID=0x608 · DATA=23,00,18,05,05,00,00,00 加速度200Hz輸出
- ID=0x608 · DATA=23,00,18,05,0A,00,00,00 加速度100Hz輸出
- ID=0x608 · DATA=23,00,18,05,14,00,00,00 加速度50Hz輸出
- ID=0x608 · DATA=23,00,18,05,32,00,00,00 加速度20Hz輸出
- ID=0x608 · DATA=23,00,18,05,64,00,00,00 加速度10Hz輸出(最低10Hz)
- ID=0x608 · DATA=23,01,18,05,00,00,00,00 關閉角速度輸出
- ID=0x608 · DATA=23,01,18,05,05,00,00,00 角速度200Hz輸出
- ID=0x608 · DATA=23,01,18,05,0A,00,00,00 角速度100Hz輸出
- ID=0x608 · DATA=23,01,18,05,14,00,00,00 角速度50Hz輸出
- ID=0x608 · DATA=23,01,18,05,32,00,00,00 角速度20Hz輸出
- ID=0x608 · DATA=23,01,18,05,64,00,00,00 角速度10Hz輸出(最低10Hz)
- ID=0x608 · DATA=23,02,18,05,00,00,00,00 關閉歐拉角輸出
- ID=0x608 · DATA=23,02,18,05,05,00,00,00 歐拉角200Hz輸出
- ID=0x608 · DATA=23,02,18,05,0A,00,00,00 歐拉角100Hz輸出
- ID=0x608 · DATA=23,02,18,05,14,00,00,00 歐拉角5Hz輸出
- ID=0x608 · DATA=23,02,18,05,32,00,00,00 歐拉角20Hz輸出
- ID=0x608 · DATA=23,02,18,05,64,00,00,00 歐拉角10Hz輸出(最低10Hz)
- ID=0x608 · DATA=23,03,18,05,00,00,00,00 關閉四元數輸出
- ID=0x608 · DATA=23,03,18,05,05,00,00,00 四元數200Hz輸出
- ID=0x608 · DATA=23,03,18,05,0A,00,00,00 四元數100Hz輸出
- ID=0x608 · DATA=23,03,18,05,14,00,00,00 四元數50Hz輸出
- ID=0x608 · DATA=23,03,18,05,32,00,00,00 四元數20Hz輸出
- ID=0x608 · DATA=23,03,18,05,64,00,00,00 四元數10Hz輸出(最低10Hz)
- ID=0x608 · DATA=23,04,18,05,00,00,00,00 關閉氣壓輸出
- ID=0x608 · DATA=23,04,18,05,05,00,00,00 氣壓200Hz輸出
- ID=0x608 · DATA=23,04,18,05,0A,00,00,00 氣壓100Hz輸出
- ID=0x608 · DATA=23,04,18,05,14,00,00,00 氣壓50Hz輸出
- ID=0x608 · DATA=23,04,18,05,32,00,00,00 氣壓20Hz輸出
- ID=0x608 · DATA=23,04,18,05,64,00,00,00 氣壓10Hz輸出(最低10Hz)

以 TPDO1(加速度)輸出速率為100Hz(每10ms輸出一次)為例: 0x23為SDO寫四個位元組指令。0x00, 0x18為寫0x1800索引。0x05為子索引。0x00, 0x0A=(0x00<<8) + 0x0A = 10(單位為ms)·後面不足補0。

#### 設定傾角儀輸出正負號

- ID=0x608 · DATA=23,9E,20,00,00,00,00,00 X軸正負號為出廠預設方向
- ID=0x608 · DATA=23,9E,20,00,01,00,00,00 X軸正負號反向
- ID=0x608 · DATA=23,9F,20,00,00,00,00,00 Y軸正負號為出廠預設方向
- ID=0x608 · DATA=23,9F,20,00,01,00,00,00 Y軸正負號反向

#### 設定傾角儀零點(相對位置輸出)

- ID=0x608 · DATA=23,A5,20,00,02,00,00,00 寫入后即設定目前位置為輸出零點(X=0,Y=0)
- ID=0x608 · DATA=23,A5,20,00,05,00,00,00 寫入后取消零點配置 · 輸出真實的X · Y角度(相當於X,Y offset=0)

#### 設定傾角輸出解析度(定製功能)

- ID=0x608 · DATA=23,A2,20,00,00,00,00,00 傾角數據輸出解析度為 0.01°(預設)
- ID=0x608 · DATA=23,A2,20,00,01,00,00,00 傾角數據輸出解析度為 0.02°
- ID=0x608 · DATA=23,A2,20,00,02,00,00,00 傾角數據輸出解析度為 0.1°

#### 恢復出廠設定

- ID=0x608 · DATA=23 00 20 00 01 00 00 00 恢復出廠設定 · 將所有參數包括波特率 · 節點ID等恢復至出廠值 · 重新上電生效

### 5.4.4. 同步協議

#### 配置TPDO為同步模式

- 關閉所有TPDO(設定TPDO輸出速率為0) ·
- 發送CANopen同步幀 · CANopen 同步幀: ID:80 · DATA:空`

## 6. CAN數據協議與指令( SAE-J1939協議)

模組預設輸出協議為CANOpen, 如需SAE J1939協議, 請聯繫我司。

PGN	描述
通訊模式	廣播通訊
預設傳輸時間間隔	100ms
數據長度	每個PGN8 位元組
PF(PDU format)	0xFF
PS(PDU specific)	PF > 0xF0時為拓展PGN地址(GE), 否則為目的地址(DA)
優先順序	3
預設J1939地址	0x08
數據格式	所有幀中數據格式採用LSB(低位在前),無特殊說明均為有符號整型

### 6.1. PGN訊息列表

#### 6.1.1. PGN65332(FF34) 加速度

CANID=0x0CFF3408

名稱	位置(byte)	說明
加速度X	0-1	單位G(1G=1重力加速度), 比例因子:0.00048828
加速度Y	2-3	單位G(1G=1重力加速度), 比例因子:0.00048828
加速度Z	4-5	單位G(1G=1重力加速度), 比例因子:0.00048828
保留	6-7	-

#### 6.1.2. PGN65335(FF37) 角速度

CANID=0x0CFF3708

名稱	位置(byte)	說明
角速度X	0-1	單位deg/s, 比例因子:0.061035
角速度Y	2-3	單位deg/s, 比例因子:0.061035
角速度Z	4-5	單位deg/s, 比例因子:0.061035
保留	6-7	

#### 6.1.3. PGN65345(FF41) 航向角

CANID=0x0CFF4108

SPN 名稱	SPN 位置(byte)	說明
航向角(Yaw)	0-3	0-360, 單位°, 比例因子:0.001, 順時針為正
保留	4-7	

#### 6.1.4. PGN65354(FF4A) 傾角儀輸出

CANID=0x0CFF4A08 (只適用於輸出J1939協議的傾角儀產品)

名稱	位置(byte)	說明
X傾角角度	0-3	範圍0-360或 ±180, 單位:deg, 比例因子:0.001
Y傾角角度	4-7	範圍0-360或 ±180, 單位:deg, 比例因子:0.001

## 6.2. 配置指令

### 6.2.1. 配置格式

主機發送: ADDR+ CMD + STATUS + VAL, 從機響應: ADDR+ CMD + STATUS + VAL

欄位	大小(Byte)	說明
ADDR	2	暫存器地址
CMD	1	0x06:寫入, 0x03:讀取
STATUS	1	保留
VAL	4	當寫入時: 寫入的值 · 當讀取時: 保留

### 6.2.2. 配置模組

29'b 拓展幀地址	數據	描述	說明
0x0CEF08xx	34 01 06 00 [VAL]	VAL: 4位元組	PGN:FF34(加速度) 發送間隔 · 單位ms, 範圍:5 -1000
0x0CEF08xx	37 01 06 00 [VAL]	VAL: 4位元組	PGN:FF37(角速度) 發送間隔 · 單位ms, 範圍:5 -1000
0x0CEF08xx	41 01 06 00 [VAL]	VAL: 4位元組	PGN:FF41(航向角) 發送間隔 · 單位ms, 範圍:5 -1000
0x0CEF08xx	4A 01 60 00 [VAL]	VAL: 4位元組	PGN:FF4A(傾角儀輸出) 發送間隔 · 單位ms, 範圍:5 -1000
0x0CEF08xx	00 00 06 00 00 00 00 00	-	儲存所有配置參數到Flash
0x0CEF08xx	00 00 06 00 01 00 00 00	-	恢復出廠設定
0x0CEF08xx	00 00 06 00 FF 00 00 00	-	復位
0x0CEF08xx	9A 00 06 00 [VAL]	VAL: 4位元組	配置波特率(儲存設定 · 復位生效): 0:1000K, 1:800K, 2:500K, 3:250K, 4:125K
0x0CEF08xx	9C 00 06 00 [VAL]	VAL: 4位元組	設定J1939 節點ID: 1-128
0x0CEF08xx	A5 00 06 00 [VAL]	VAL: 4位元組	設定零位, 0x02:設定目前位置為零位 · 0x05 : 取消零位設定 · 輸出絕對物理角度
0x0CEF08xx	9E 00 06 00 [VAL]	VAL: 4位元組	設定X軸正負方向, 0:預設 1:反向
0x0CEF08xx	9F 00 06 00 [VAL]	VAL: 4位元組	設定Y軸正負方向, 0:預設 1:反向
0x0CEF08xx	A3 00 06 00 [VAL]	VAL: 4位元組	(定製功能)設定XY傾角輸出解析度0: 0.001°(預設), 1: 0.01°, 2: 0.1°

地址域中xx: J1939協議中的源地址 · 可為任意位元組。

數據域中xx: 任意位元組

例: ID=0x0CEF0855, DATA = 37 01 06 00 64 00 00 00 : 將PGN:FF37設定為100ms週期(10Hz)

## 7. CAN數據協議與指令( NMEA2000協議)

定製協議, 如需NMEA2000協議, 請聯繫我司。

## 8. 地磁校準

### 8.1. 地磁校準步驟

- 首先確認校準前提: 目前測試下航向角精度達不到要求。
- 模組安裝環境有磁場干擾, 這種干擾是固定的, 並且這個干擾磁場與模組安裝之後不會再發生距離變化(例: 模組安裝在一個鐵材料之上, 因為鐵會有磁場干擾, 這時就需要把鐵與模組一起旋轉校準, 並且這個鐵在使用當中是不會和模組再分開的, 一旦分開是需要再重新校準。如果這個鐵大小是不固定的, 或與模組的距離變化也不是固定的, 這種干擾是無法校準, 只能避而遠之安裝, 安全距離控制在 40CM 以上)

模組內部自帶主動地磁校準系統, 不需要使用者發送任何指令, 該系統在後臺自動採集一段時間內地磁場數據, 並做分析比較, 剔除異常數據, 一旦數據足夠, 就會嘗試地磁校準。所以, 當使用地磁輔助(9軸)模式時, 不需要使用者任何干預即可完成地磁校準。但是模組仍然提供介面來讓使用者檢查目前校準狀態。**自動校準的前提是需要模組有充分的姿態變化(緩慢的讓模組經歷儘量多的姿態變化)**, 內部校準系統才能蒐集不同姿態下的地磁場資訊, 從而完成校準, 靜止狀態下是無法進行地磁校準的。

當首次使用模組並且需要使用AHRS ( 9軸 ) 模式時, 應進行如下校準操作:

1. 首先需要將模組切換到9軸模式, 才能開啟地磁校準。
2. 檢查周圍是否存在磁場干擾: 實驗室鐵質或者含有鐵質的桌子、電腦、電機、手機等旁邊都屬於常見的干擾區域。建議將模組拿到室外空曠處, 即使沒有條件拿到室外, 儘量將模組遠離干擾源。
3. 在儘量小範圍內(位置儘量不動, 只是旋轉), 緩慢的讓模組旋轉, 讓模組經歷儘量多的姿態位置(每個軸至少都旋轉360°, 持續約1分鐘), 即可完成校準, 如果始終沒能成功校準模組, 說明周圍地磁場干擾比較大。
4. 如果客戶安裝位置改變(比如上一次校準是拿著模組單獨去校準的, 使用的時候卻是安裝在目標裝置上), 則需要帶著目標裝置進行重新校準。
5. 如何檢視地磁校準結果: 輸入 LOG MAGCONFIG, 返回結果如下:

```
1 | ...
2 | MAG_BIS=
3 |     15.723
4 |     -0.320
5 |     -14.014
6 | OK
```

如果MAG\_BIAS顯示三個值不是0,0,0, 代表校準成功。

### 8.2. 配置地磁校準(手動地磁校準)

預設情況下, 只要配置為9軸模式, 模組就會在後臺不停的嘗試估計地磁校準參數。儘管如此, 本產品依然提供了關閉地磁校準的API:



- **CONFIG IMU MCAL 1**: 打開實時地磁校準(立即生效, 掉電不儲存, 出廠預設)
- **CONFIG IMU MCAL 0**: 關閉實時地磁校準(立即生效, 掉電不儲存)

如果您只想手動執行一次地磁校準, 而不想讓模組實時嘗試地磁校準, 可以按如下步驟操作:

1. 在無地磁干擾的情況下(室外)讓模組進行2-4分鐘緩慢8字運動, 讓模組完成一次自動地磁校準。
2. 輸入 **CONFIG IMU MCAL 0**, 並且 **SAVECONFIG** 儲存配置。這樣就關閉了模組地磁校準功能, 且掉電儲存。

### 8.3. 磁干擾的知識

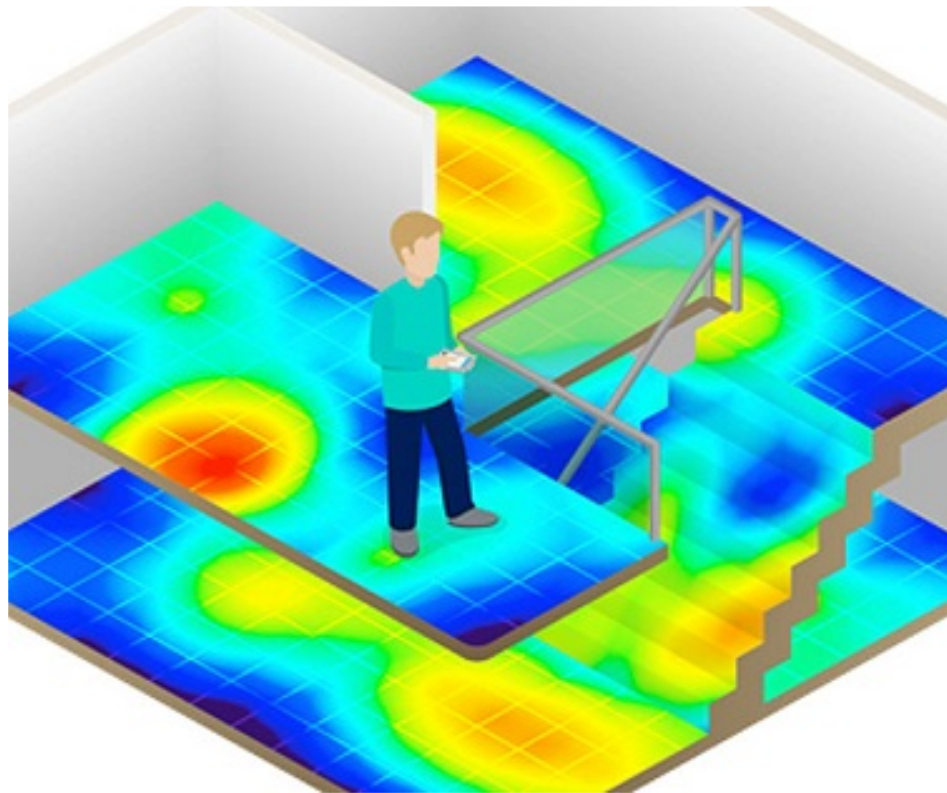
地磁干擾可分為空間磁場干擾與感測器座標系下的磁場干擾, 如下圖所示

Distortions that move with the sensor	Distortions that do not move with the sensor
 <ul style="list-style-type: none"> <li>• Calibration errors</li> <li>• Hard iron effects</li> <li>• Soft iron effects</li> <li>• Etc.</li> </ul>	 <ul style="list-style-type: none"> <li>• Spatial distortions</li> <li>• Temporal distortions</li> <li>• Etc.</li> </ul>

### 8.3.1. 空間磁場干擾

定義：磁場干擾不隨感測器運動而運動，處於世界座標系下

典型干擾源：各種固定的地磁干擾源，傢俱，家用電器，線纜，房屋內的鋼筋結構等。一切不隨磁感測器運動而運動的干擾源，下圖是典型的室內磁場分佈圖。



對模組的影響：無論磁場感測器是否校準的好，這些空間磁場的干擾(或者說環境磁場不均勻)都會使得空間地磁場發生畸變。地磁補償會錯誤，無法獲得正確的航向角。他們是造成室內地磁融合難以使用的主要原因。這種干擾不能被校準，會嚴重影響地磁效能。空間磁場干擾在室內尤其嚴重。

應對措施：只能儘量避免這種干擾源

### 8.3.2. 感測器座標系下的干擾

定義：地磁場干擾源隨感測器運動而運動

典型干擾源：與模組固定在一起的PCB板子，儀器裝置等。他們和磁感測器視為同一個剛體，隨磁感測器運動而運動。

對模組的影響：對感測器造成硬磁/軟磁干擾。這些干擾可以通過地磁校準演算法加以很好的消除。

應對措施：對模組進行地磁校準。

## 8.4. 地磁使用注意事項

- 1 在室內環境下，空間磁場干擾尤其嚴重，而且空間磁干擾並不能通過校準來消除。在室內環境下儘管模組內建均質磁場檢測及遮蔽機制，但地磁輔助(9軸)模式航向角的準確度很大程度上取決於室內磁場畸變程度，如果室內磁場環境很差(如電腦機房旁，實驗室，車間，地下車庫等)，即使校準后航向角精度可能還不如6軸模式甚至會出現大角度誤差。
- 2
- 3 模組的自動地磁校準系統只能處理和模組安裝在一起的，固定的磁場干擾。安裝環境如果有磁場干擾，這種干擾必須是固定的，並且這個干擾磁場與模組安裝之後不會再發生距離變化(例：模組安裝在一個導磁剛體(機器人/機械裝置/車輛/船舶等)之上，以機器為例：因為機器人金屬材料會有磁場干擾，這時就需要把機器人與模組一起旋轉校準，並且模組在使用當中是不會和機器人再分開的(發生相對位移)，一旦分開是需要再重新校準。