iAHRS V2 통신 프로토콜 설명서 (쉬운 버전)

Copyright © 2026 ROBOR Co., Ltd. All rights reserved. — 주식회사 로버

개정 2026-10-20 · 펌웨어 **2.1.21** · PC 프로그램 **UI 1.21.0** 기준 · 대상 보드 iAHRS V2 (RB-SDA-v2, 회로도 Rev 2.1)

지금 배포 중인 펌웨어는 **2.1.21** 입니다(2026-10-20). 2.1.15~2.1.21 의 명령·응답 형식은 2.1.14 와 같고(2.1.20: `algo` 0/1/2·`gv` 추가, 2.1.21: `cn`·`cb` 추가 — 2.14 절), 2.1.17 은 `b1` 의 허용 속도만 230400 까지로 줄였습니다(§2.13). 이 문서의 **응답 예시에 나오는 버전 번호는 2.1.20 기준**입니다(예: `S/W Ver 2.1.20`, SDO 0x100A = "2.1.20", 0x1018:3 = 0x00020114). 2.1.12~2.1.14 가 들어 있는 보드는 자이로 영점 자동 맞춤 결함이 있으니 최신으로 업데이트하세요.

이 문서는 "어떻게 연결해서, 무엇을 보내면, 무엇이 돌아오는가"를 처음 보는 사람 기준으로 설명합니다. PLC·로봇 제어기·직접 만든 프로그램으로 센서를 읽으려는 분, PC 프로그램의 콘솔에서 명령을 직접 쳐 보려는 분을 위한 문서입니다. 응답 문구는 해당 버전의 펌웨어 소스와 대조했습니다.

> 함께 보는 문서: 「iAHRS V2 사용설명서」(연결·화면·설치·LED·문제 해결) · 「iAHRS V2 UI 사용가이드」(화면 버튼 하나하나). 추가 연동 자료는 홈페이지 기술지원으로 요청하세요.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

목차

iAHRS V2 통신 프로토콜 설명서 (쉬운 버전) **1**

1. 한눈에 보기 3

2. 시리얼(USB·RS-232·RS-485) — iAHRS ASCII 프로토콜 5

3. CAN — CANopen 쉽게 이해하기 26

4. PC 프로그램(UI 1.21.0)에서 쓰기 31

5. 자주 묻는 것 36

부록 A. 펌웨어 버전별 프로토콜 변화 38

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

1. 한눈에 보기

1.1 연결 통로 네 가지

통로	커넥터	기본 속도	언제 쓰나	프로토콜
USB	USB-C (CP2102)	115200 8N1	PC 프로그램, 설정·보정·펌웨어 업데이트	iAHRS ASCII (텍스트 명령)
RS-232	J2 2·3 번 핀 (R20/R22 실장, 기본 모델)	115200 8N1	산업용 PC, PLC 시리얼	iAHRS ASCII (USB 와 같음)
RS-485	J2 2·3 번 핀 (R11/R12 실장, `_2` 모델)	RS-232 와 같은 속도(`b1`)	긴 케이블, 노이즈가 많은 곳	iAHRS ASCII (반이중)
CAN	J2 2·3 번 핀 (R3/R4 실장, `_1` 모델)	500 kbit/s 기본 (`cb=` 125~1000)	로봇·차량 제어기, PLC	**CANopen** (CiA 301, Node-ID 1 기본, `cn=` 1~127)

- J2 커넥터는 4 핀입니다: 1 = 전원 입력(5 V), 2 = H_A, 3 = L_B, 4 = GND. 2·3 번 핀은 CAN(CANH/CANL), RS-485(A/B), RS-232(TX/RX) 중 **보드에 실장된 저항 조합 하나만** 살아 있습니다. 어느 것인지는 주문한 모델로 정해집니다(모델명 끝 없음 = RS-232 기본 모델, `_1` = CAN, `_2` = RS-485).
- USB 와 J2 는 동시에 쓸 수 있습니다. 명령은 어느 시리얼 포트로 보내도 되고, **명령 응답과 스트림은 세 시리얼 포트(USB·RS-232·RS-485) 모두로** 나갑니다. 그래서 USB 에 PC 프로그램을 연결해 둔 채 RS-232 로 PLC 를 붙이면, PC 프로그램이 묻는 값의 응답도 PLC 쪽에 함께 들어옵니다. PC 프로그램은 연결 중 0.25 초마다 `pv`, `mt`, `s`, `ss` 등을 하나씩 돌아가며 묻습니다. 받는 쪽은 모르는 줄을 버리도록 만드세요.
- CAN 은 버스 양 끝에 120 Ω 종단저항이 있어야 합니다. 보드 안에는 종단저항이 없습니다(R8 미실장).
- 펌웨어 업데이트는 USB 또는 RS-232 로 합니다. RS-485 로는 할 수 없습니다(2.15 절).

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

1.2 제품·버전 체계

하드웨어	제품명	센서	펌웨어 버전	비고
V1	RB-SDA-v1	9축 센서	1.x	기존 양산 제품 (구형 iAHRS 프로토콜)
V2	**RB-SDA-v2**	**관성센서 + 3축 지자계**	**2.x.y**	이 문서. 현재 배포 펌웨어 2.1.21
V3	RB-SDA-v3	6축 관성센서 + 지자계	3.x.y	iAHRS V3. CAN 없음 (별도 설명서)

규칙: **펌웨어 앞자리 = 하드웨어 버전**입니다. V2 보드에는 2.x.y 펌웨어만 올라가고, 기능 추가·수정은 뒷자리(x.y)로 올라갑니다.

장치는 `vr` 응답의 `H/W Ver V2` 줄로 하드웨어 버전을, `S/W Ver 2.1.20` 줄로 펌웨어 버전을 알려 줍니다. PC 프로그램은 이 값을 읽어 상단 상태 줄에 `HW V2 · FW v2.1.20 ✓` 로 표시합니다(4.2 절).

- 장치 펌웨어가 프로그램이 아는 최신보다 낮으면 `최신 v2.1.20 ⚠` 처럼 최신 번호가 붙고 [펌웨어 업데이트 ↑] 버튼이 강조됩니다.
- 앞자리가 하드웨어와 다르면 `계열 불일치 ⚠` 가 뜹니다. 다른 계열의 펌웨어 파일(`AHRS_V3_…`)은 업데이트 전에 프로그램이 거부합니다.
- 부트로더도 제품 계열마다 서명 키가 달라 다른 계열의 파일을 받지 않습니다.

1.3 지금 어느 통로·속도로 연결됐나 — hp

`hp` 를 보내면 장치가 **그 명령이 들어온 포트와 속도**를 답합니다(FW 2.0.2 이상).

```
hp=USB,115200        <- USB 포트로 받았고, USB 속도(b2)는 115200
hp=RS232,115200      <- RS-232 포트, 속도는 b1
hp=RS485,115200      <- RS-485 포트, 속도는 b1 (RS-232 와 같은 값)
```

PC 프로그램은 연결 직후 자동으로 `hp` 를 보내 상단 상태 줄에 `USB @ 115200 bps` 처럼 표시하고, 콘솔에 `통신 방식: USB @ 115200 bps (CP2102)` 를 남깁니다. 괄호 안은 보드의 USB 칩 이름입니다(포트 이름에 CH9102 가 보이면 CH9102). RS-232·RS-485 로 연결했을 때는 칩 이름이 붙지 않습니다. CAN 으로 연결했을 때는 `CAN 500 kbit/s · PCAN-USB · Node 1` 로 표시합니다.

1.4 축과 각도의 기준

- **X = 앞, Y = 왼쪽, Z = 위** 입니다(오른손 좌표계). 케이스 윗면 라벨(보드 실크)의 X·Y 화살표가 기준이고, 커넥터가 있는 면이 뒤(−X)입니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

- a·g·m 의 x, y, z 는 이 축 기준입니다. 보드를 반듯하게 놓으면 a 가 0, 0, +1 근처로 읽힙니다.
- 롤·피치·요는 각각 X·Y·Z 축 기준의 회전입니다. 부호는 장치를 직접 기울이며 e 를 읽어 확인하세요.
- 요의 기준은 지자계 융합에 따라 다릅니다. mag=0(기본)이면 자북과 상관없는 상대 방향이고, mag=1 이면 자북 기준입니다. mag=1 에서 편각(dec)을 넣으면 진북 기준이 됩니다(2.4 절).
- 장치를 비스듬히 달았다면 rm(장착 회전)이나 za(영점)로 맞춥니다(2.3·2.4 절).

iAHRS V2 센서 축 — 케이스 윗면 라벨의 화살표 기준 (V2 케이스 3D 렌더 그림)

2. 시리얼(USB·RS-232·RS-485) — iAHRS ASCII 프로토콜

2.1 기본 규칙

1. 명령은 **영문 텍스트 + 줄바꿈**입니다. 줄바꿈은 CR, LF, CR+LF 모두 됩니다. 예: vr + Enter. 소문자로 보내는 것을 권장합니다.
2. 읽기는 이름만(mag), 쓰기는 이름=값(mag=1)입니다. 장치는 항상 이름=값 형태로 **현재 값**을 돌려줍니다. 범위 밖의 값을 쓰면 무시하고 현재 값을 그대로 돌려주므로, 응답 값을 보고 적용됐는지 확인하세요. 단, 값 여러 개를 받는 rm, geo, cw 는 범위 밖이거나 형식이 틀리면 !ERR 로 답합니다.
3. 모르는 명령이나 형식이 틀린 쓰기에는 !ERR 로 답합니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

4. **한 줄은 최대 63 자**입니다(줄바꿈 제외, FW 2.1.11 이상 — 그 전은 47 자). 더 긴 줄은 응답 없이 버려집니다.
5. 장치는 **한 번에 한 줄만** 보관합니다. 앞 명령의 응답을 받은 뒤 다음 명령을 보내세요. 응답을 기다리지 않고 연달아 보낼 때도 **120 ms 이상** 간격을 두세요(PC 프로그램도 이 간격을 씁니다).
6. 실행형 명령(`ca`, `cm`, `fw` 등)은 먼저 명령 이름을 한 줄로 되돌려 주고(`ca`), 결과는 끝난 뒤 따로 옵니다(`ca done. …`).
7. 시간이 걸리는 명령(`ca` 약 2~2.5 초, `fw`·`fd` 약 1 초, `cr` 약 0.2 초) 동안에는 스트림이 잠시 멈춥니다.

터미널 프로그램(PuTTY, Tera Term)에서 115200 8N1, 로컬 에코 켬, CR+LF 개행으로 설정하면 바로 대화할 수 있습니다. PC 프로그램의 콘솔 명령 입력줄을 써도 됩니다(4.1 절).

2.2 처음 켜면 나오는 글

```
B:14000000 f=00000000                          <- 부트로더: 리셋 원인 코드 (USB 에만 나옴)
=== ROBOR iAHRS V2 (iAHRS protocol) ===
[OK ] UART up @115200
[OK ] settings restored from flash (fw)        <- 저장된 설정 복원
[OK ] CAN 500 kbit/s, selftest PASS
[OK ] CANopen node 1: pre-operational, TPDO 0x181..0x481 after NMT start
[INFO] bootloader BL1: firmware update over USB and RS-232 (not CAN)
IMU       WHO_AM_I=0xE9 (expect 0xE9) -> OK    <- 가속도·자이로 센서
MAG       PROD_ID=0x30 (expect 0x30) -> OK     <- 지자계 센서
[OK ] blackbox: boot #4, 4 events, ~0 h total
[OK ] ADC: VDDA=3.30 V, MCU 31.6 C
Calibrating gyro, keep still...                <- 약 1 초, 움직이지 마세요
Gyro bias: 0.024 0.013 -0.189 dps
Ready (6-axis; mag present but OFF - 'mag=1' to fuse it). iAHRS protocol; send 'h' for
help.
```

- `Ready` 줄이 나오면 명령을 받을 수 있습니다. 지자계 융합은 기본 꺼짐이라 `6-axis` 로 시작합니다. 지자계를 켜고(`mag=1`) 저장한 보드는 `Ready (9-axis)` 로 나옵니다.
- CANopen 줄에는 `0x181..0x481` 만 적혀 있지만 지자기 PDO `0x681` 도 함께 나갑니다(3.5 절).
- `[INFO] bootloader` 줄은 보드에 든 부트로더 종류입니다. `BL2` 면 `firmware update over USB, RS-232 and CAN` 으로 나옵니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

가끔 나오는 줄:

줄	뜻과 조치
`[OK ] factory defaults loaded (fd)`	사용자 저장값이 없어 공장값으로 시작했습니다
`[OK ] b2=230400 (saved boot baud)`	저장된 통신속도로 바꿉니다. 이 줄까지는 115200 으로 나옵니다. RS-232·RS-485 속도는 `[OK ] b1=… (saved boot baud)` 로 나옵니다
`[OK ] watchdog armed (~4 s)`	`wd=1` 이 저장된 보드입니다. 시작에 실패하면 `[WARN] watchdog init failed`
`[INFO] gyro bias from quiet window 2/4 (vibration during calibration)`	부팅 자이로 보정 중 진동이 있어 가장 조용한 구간을 썼습니다
`[WARN] moving during gyro calibration - retrying, keep the device still...`	부팅 자이로 보정 중 움직였습니다. 한 번 더 잽니다. 가만히 두세요
`[WARN] vibration during gyro calibration - approximate bias used (std 0.40 dps). Run 'ca' when still.`	근사값을 썼습니다. 가만히 둔 뒤 `ca` 를 보내세요
`[WARN] gyro bias NOT calibrated (device kept moving). Place it still and run 'ca'.`	자이로 영점을 못 잡았습니다. 가만히 둔 뒤 `ca`
`[ERR] IMU IREG setup not verified (0x02: 1=pin7 2=gAAF 4=aAAF 8=gLPF 16=aLPF)`	센서 칩 내부 필터 설정이 확인되지 않았습니다(FW 2.1.14 이상). 측정은 계속됩니다. `imf` 로 확인하고(2.10 절), 반복되면 문의하세요
`[ERR] sensor cfg write failed (0x04); previous setting kept.`	저장된 센서 설정(`as gs odr gl al`)을 칩에 넣지 못했습니다(2.10 절)
`MAG PROD_ID=0x00 (expect 0x30) -> FAIL + Ready (6-axis, no mag)`	지자계가 인식되지 않았습니다
`[ERR] IMU not responding. SPI2: …`	가속도·자이로 센서가 인식되지 않았습니다(빨간 LED 2 회 깜빡임). 이때는 `Ready` 줄이 나오지 않지만 명령은 받습니다. 장치는 1 초마다 센서를 다시 찾습니다
`[WARN] CAN start failed` / `[WARN] CANopen stack init failed`	CAN 컨트롤러 또는 CANopen 이 시작되지 않았습니다. 시리얼은 정상입니다
`[WARN] HSE 25 MHz failed to start - running on the HSI PLL (CAN timing degraded)`	보드의 수정 발진기가 시작되지 않아 내부 클럭으로 동작합니다. CAN 통신이 불안정할 수 있습니다. 반복되면 문의하세요
`[WARN] ADC init failed - pv/mt use fallbacks`	전압·MCU 온도 측정이 시작되지 않았습니다. `pv·mt` 값을 믿지 마세요

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

동작 중에 나올 수 있는 줄:

줄	뜻과 조치
`[WARN] IMU stopped responding (read errors) - reinit pending`	가속도·자이로 센서와의 통신이 끊겼습니다(`(no data ready)` 로 끝나기도 함). 상태 워드에 0x02000000 이 켜지고, CAN 이면 EMCY 가 나갑니다(3.8 절)

- 센서가 끊기면 장치가 1 초마다 센서를 다시 시작해 봅니다. 되살아나면 자이로 영점을 다시 잡고 측정을 이어 갑니다.
- 이때 위 표의 자이로 보정 줄(`[WARN] moving during gyro calibration …` 등)이 다시 나올 수 있으니 가만히 두세요. 별도의 "복구됨" 줄은 없습니다. `ss` 의 `imu:1` 이나 상태 워드로 확인하세요.

2.3 자주 쓰는 명령

하고 싶은 것	보내는 것	돌아오는 것 (예)
장치 확인	`vr`	`ROBOR iAHRS V2` / `H/W Ver V2` / `S/W Ver 2.1.20 (iAHRS, <빌드 날짜>)` (빌드 날짜는 `Oct 10 2026` 같은 형식)
자세(오일러) 한 번 읽기	`e`	`e=0.386,-0.301,35.920` (롤, 피치, 요 °)
쿼터니언	`q`	`q=0.999985,0.003300,-0.002682,0.003571` (w, x, y, z)
가속도 / 자이로 / 지자계	`a g m`	`a=0.007,0.006,0.997` g · `g=0.030,-0.020,0.070` °/s · `m=109.49,-81.57,-220.84` μT
온도 / 입력전압	`t pv`	`t=30.8` ℃ · `pv=5.08` V
상태 워드	`s`	`s=0x2048` (2.11 절)
도움말	`h`	명령 목록 3 줄 (2.4 절)
연속 출력 시작	`sp=20`	`sp=20`, 이후 20 ms 마다 한 줄 (2.5 절)
연속 출력 정지	`sp=0`	`sp=0`
지금 자세를 0°로	`za`	`za`
영점 해제	`ra`	`ra`
지자계 융합 켜기/끄기	`mag=1` / `mag=0`	`mag=1` — 기본 꺼짐. 켜면 요(yaw)가 자북 기준
설정을 플래시에 저장	`fw`	`fw` → `fw: saving settings to flash...` → `fw done. settings saved (kept after power-off).`
재시작	`rd`	`rd` 후 재부팅
펌웨어 업데이트 모드	`fu`	`fu` → `fu: rebooting to bootloader (115200 8N1)...` (2.15 절)

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

영점(za)은 지자계 융합 상태에 따라 다르게 동작합니다.

- mag=0(6 축, 기본): 롤·피치·요를 **모두** 0 으로 맞춥니다.
- mag=1(9 축): 롤·피치만 0 으로 맞추고, **요는 자북 기준 방위를 그대로 유지**합니다.
- 영점은 RAM 에만 있습니다. 전원을 꺼도 유지하려면 fw 로 저장하세요. ra 는 영점을 지우고 자세 계산을 다시 시작합니다.
- 영점은 센서를 물리적으로 바로 세우는 것이 아니라, 지금 자세를 0°로 부르도록 기준만 바꾸는 것입니다. PC 프로그램의 [영점]·[영점 해제]가 za·ra 를 보냅니다.

영점(za) — 지금 자세를 0°로 만들기. 저장을 해야 전원을 껐다 켜도 유지됩니다

2.4 명령 전체 목록 — h

h 를 보내면 아래 3 줄이 그대로 나옵니다(FW 2.1.14 이상).

```
read: a g m e q t pv mt ag v p vd f sc pg vx an eu dv aa hvv hrs sn s can pt imf ss hp
cfg : so sp sd sb b1 b2 beta gate mag bt dec rm geo | la vb fft sh vel pos step mr ad
wd au cs nmea bb hi hv ntc | as gs odr gl al | algo vdx ai
cmd : za ra cm ca ct rc st rp rs lg lgc slp fw fd fu rd vr c=<n> cr cw=<g,x,y,z>
```

읽기 명령 — 이름만 보내면 이름=값 으로 답합니다.

명령	돌려주는 값	단위·조건
a / g / m	가속도 / 자이로 / 지자계 x, y, z	g / °/s / μT (보정 적용값)
e / q	롤·피치·요 / 쿼터니언 w, x, y, z	° (영점·장착 회전·편각 적용)
t / mt	센서 온도 / MCU 온도	°C
pv	입력 전압(VIN)	V
ag	선형 가속도(중력 제거)	g, la=1 일 때
v / p	속도 / 위치	m/s / m, vel=1 / pos=1 일 때 (오래 쓰면 오차가 커짐)
vd	진동 RMS x, y, z	g, vb=1 일 때

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

명령	돌려주는 값	단위·조건
f	주 진동 주파수, 진폭	Hz, fft=1 일 때
sc / pg	걸음 수 / 최대 가속도 크기	step=1 일 때 / g (rs 로 초기화)
vx	크레스트 팩터, 첨도	vdx=1 일 때
an	이상 점수, 이상 여부(0/1)	ai=1 일 때
eu	자세 추정 오차(1σ) 롤, 피치, 요 — algo=2 에서는 eu=unavailable	° (ESKF)
gv	GuardedVQF 상태(FW 2.1.20+): gv=<algo>,0x<상태워드>,<가속 품질>,<헤딩 품질> — 품질은 0~1 경험값(확률 아님). 상태 비트: 1 초기화, 2 가속 사용, 4 가속 거부, 8 자기 사용, 16 자기 거부, 32 헤딩 유효, 64 자기 오래됨, 128 시간 오류, 512 저하, 2048 자기 획득 중	—
dv	Δ 각도 x, y, z + Δ 속도 x, y, z	° / m/s, 읽을 때마다 0 부터 다시 셈
aa	각가속도 x, y, z	°/s²
hvv	히브(상하 변위)	m, hv=1 일 때
hrs	부팅 횟수, 누적 시간, 이벤트 수, 기록 용량	블랙박스 기준
sn	시리얼번호 (MCU 고유 ID, 16 진수 24 자리)	
s ss can pt imf	상태 워드와 진단	2.10·2.11 절
hp	연결 포트와 속도	1.3 절

설정 명령 — 이름만 보내면 읽기, 이름=값 이면 쓰기입니다.

명령	뜻	값 (기본값)	fw 로 저장
sp	스트림 주기 [ms]	0~60000, 0 = 정지 (0)	안 됨
sd	스트림에 넣을 필드	비트 마스크 (0xDD)	안 됨
so	스트림 포트 (호환용, 실제로는 세 포트 모두)	0 / 1 (1)	안 됨
sb	28 바이트 바이너리 스트림	0 / 1 (0)	안 됨
b1 / b2	RS-232·RS-485 / USB 통신속도	2.13 절 (115200)	됨 (다음 부팅 속도)
cn / cb	CANopen 노드 ID / CAN 속도 kbit/s (2.1.21+)	1~127 (1) / 125·250·500·1000 (500) — 2.14 절	됨 (쓰면 바로 재시작, 저장은 fw)
beta	Madgwick 게인	0~5 (0.20)	됨
gate	가속도 외란 게이트	0 / 1 (1)	됨
mag	지자계 융합	0 / 1 (0)	됨

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

명령	뜻	값 (기본값)	fw 로 저장
bt	자이로 영점 자동 맞춤 (아래 설명)	0 / 1 (1)	됨
dec	자기 편각 [°] (한국 약 −9, 아래 설명)	−90~90 (0)	됨
rm	장착 회전 롤, 피치, 요 [°] — 예 rm=0,0,90	롤·요 ±180, 피치 ±90 (0,0,0)	됨
geo	위도, 경도를 넣으면 편각 자동 계산 (쓰기 전용) — 예 geo=37.5,127.0	응답 geo=… dec=… inc=… (WMM2025)	dec 로 저장
algo	융합 방식: 1 = ESKF, 0 = Madgwick, 2 = GuardedVQF(실험, FW 2.1.20+; 그 외 값은 !ERR)	0 / 1 / 2 (1)	됨
as gs odr gl al	센서 설정	2.10 절	됨
기능 토글 (아래 표)	켜기 / 끄기	0 / 1 (bb 만 1, 나머지 0)	됨

기능 토글	뜻
la	선형 가속도(중력 제거) 계산
vb	진동 RMS
fft	주 진동 주파수(FFT)
sh	충격·자유낙하 감지
vel / pos	ZUPT 속도 / 위치 (드리프트가 큼)
step	만보계
mr	자기 외란 제거
ad	적응형 게인
wd	워치독 (저장 후 다음 부팅부터)
au	부팅하면 자동으로 스트림 시작 (저장 필요, 2.5 절)
cs	스트림 줄 끝에 체크섬
nmea	NMEA 0183 출력 (CSV 대신)
bb	블랙박스 이벤트 기록 (기본 켬)
hi	온라인 하드아이언 자동 추적
hv	히브 추정
ntc	적응형 진동 노치 필터
vdx	진동 진단 (vx 값)
ai	엣지 이상 감지 (an 값)

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

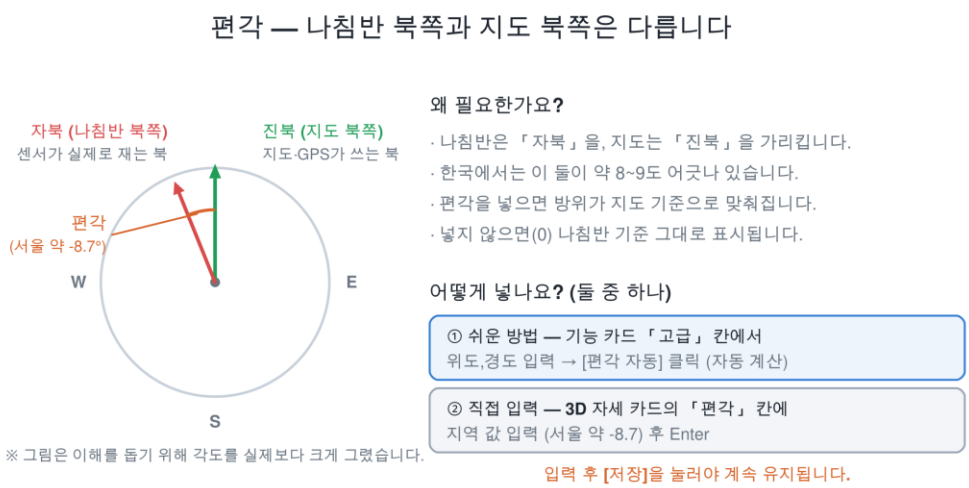
`bt` — 자이로 영점 자동 맞춤

- 장치는 **정지해 있는 구간에서만** 자이로 영점을 맞춥니다(`bt=1`, 기본). 1 초 단위로 보아 흔들림과 기울기 변화가 잡음 수준일 때를 정지로 봅니다.
- 움직이는 동안 들어온 값은 확정하기 전까지 되돌릴 수 있게 따로 보관합니다. 그래서 부드럽게 멈추거나 흔들린 뒤에도 정지해 있는 동안 방향(요) 표시가 흐르지 않습니다.
- 한계: 지자계를 끈 상태(`mag=0`)에서 **아주 느린(초당 0.4° 미만) 일정한 회전**은 영점 변화와 구분할 수 없어, 실제보다 화면에 덜 나타날 수 있습니다. 방향이 중요한 용도는 지자계를 쓰세요(`cm` 보정 → `mag=1` → `fw`).
- 지그 자동 캘리브레이션은 실행하는 동안 자동으로 `bt=0` 으로 바꾸고, 끝나면 다시 켭니다.
- 이 동작은 FW 2.1.15 이상입니다. 2.1.12~2.1.14 는 흔들림 뒤 방향이 계속 돌 수 있는 결함이 있으니 최신(2.1.21)으로 업데이트하세요.
- PC 프로그램은 연결할 때 장치가 `bt=0` 이면 한 번 `bt=1` 로 켭니다(4.2 절). 양산용 프로그램에서는 3D 카드의 '바이어스 추적' 체크로 켜고 끕니다.

`dec` · `geo` — 편각

- `dec` 은 자북과 진북의 차이(편각)입니다. 0 이면 요가 자북 기준, 지역 값을 넣으면 진북 기준입니다. 한국은 약 −9° 입니다.
- `geo=<위도>,<경도>` 를 보내면 장치 안의 세계자기모델(WMM2025)로 편각을 계산해 `dec` 에 바로 넣고, `geo=37.500,127.000 dec=-9.02 inc=54.60 (WMM2025)` 처럼 답합니다(값은 예). `inc` 는 복각입니다.
- 위도 ±90, 경도 ±180 밖이면 `!ERR` 입니다. 계산 기준 연도는 펌웨어 빌드 날짜입니다.
- 넣은 편각은 RAM 에만 있습니다. 계속 쓰려면 `fw` 로 저장하세요.
- PC 프로그램에서는 기능 카드 '고급'의 [편각 자동]이 `geo=` 를 보내고, 응답의 편각을 3D 카드의 '편각' 칸과 진북/자북 표시에 반영합니다. '편각' 칸에 직접 숫자를 넣으면 `dec=` 를 보냅니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr



편각 — 나침반 북쪽(자북)과 지도 북쪽(진북)의 차이. [편각 자동]은 geo=, '편각' 칸은 dec= 를 보냅니다

실행 명령

명령	하는 일	응답
`za` / `ra`	영점 / 영점 해제	`za` / `ra`
`ca cm ct rc`	보정	2.7~2.9 절
`st`	가속도·자이로 센서 셀프테스트	`st PASS (0x0F: id+ accel+ gyro+ temp+)` — 실패 항목은 `-` 로 표시
`rp`	속도·위치를 0 으로	`rp done. velocity+position zeroed.`
`rs`	걸음 수·최대 가속도 초기화	`rs done. step count + peak reset.`
`lg`	블랙박스 이벤트 최근 20 건	`lg: …` 여러 줄 뒤 `lg done.`
`lgc`	블랙박스 기록 지우기 (약 1 초)	`lgc: erasing event log...` → `lgc done. log cleared.` (실패 `lgc FAILED.`)
`slp`	딥슬립. 아무 바이트나 받으면 깨어남	`slp: entering deep sleep (send any byte to wake)...` → 깨면 `wake. (deep sleep exit)`
`fw` / `fd`	설정 저장 / 공장값 복원	2.6 절
`fu` / `rd`	부트로더로 재시작 / 재시작	2.15 절 / `rd`
`vr`	장치·버전 확인	1.2 절
`cr` / `cw=…`	보정값 백업·복원	2.12 절
`c=<n>`	숫자 명령 (구형 호환)	2.16 절

`slp` 는 워치독이 켜진 보드(`wd=1`)에서는 `slp FAILED. watchdog armed (wd=1) - it would reset in STOP.` 로 거절합니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

2.5 연속 출력(스트림) — sp · sd

`sd` 로 어떤 값을 넣을지 고르고, `sp` 로 주기(ms)를 정합니다. 기본 `sd=0xDD` 이면 한 줄이 이렇게 옵니다.

```
33718,0.003,0.006,0.992,0.216,-0.072,-0.046,109.49,-81.57,-220.84,0.377,-0.309,0.408,0.999985,0.003300,-0.002682,0.003571
 시간 ms | 가속도 x y z [g] | 자이로 x y z [°/s] | 지자계 x y z [μT] | 롤 피치 요 [°] | 쿼터니언 w x y z
```

`sd` 비트를 켜면 아래 표의 **위에서부터 순서대로** 필드가 붙습니다.

비트	필드	개수
0x0001	시간 (부팅 후 ms)	1
0x0002	온도 [°C]	1
0x0004	가속도 x, y, z [g]	3
0x0008	자이로 x, y, z [°/s]	3
0x0010	지자계 x, y, z [μT]	3
0x0020	선형 가속도 x, y, z [g] (`1a=1` 이 아니면 0)	3
0x0040	롤, 피치, 요 [°]	3
0x0080	쿼터니언 w, x, y, z	4
0x0100	속도 x, y, z [m/s]	3
0x0200	위치 x, y, z [m]	3
0x0400	진동 주파수·진폭 3 쌍 (첫 쌍만 값, 나머지 0)	6
0x0800	진동 RMS x, y, z [g]	3

- 예: `sd=0x40` 이면 한 줄이 `롤,피치,요` 만, `sd=0x41` 이면 앞에 시간이 붙습니다. `sd` 는 16 진(`0x41`)·10 진(`65`) 모두 받고, 응답은 항상 `sd=0x41` 처럼 16 진입니다.
- `sp` 는 10 진수 ms 입니다. `sp=0` 이면 멈춥니다.
- 스트림은 세 시리얼 포트 모두로 나갑니다. RS-485 는 반이중이라 스트림이 흐르는 동안 보낸 명령은 충돌할 수 있습니다. 먼저 `sp=0` 으로 멈추세요.

> 115200 bps 에서 기본 17 개 필드 한 줄(약 100 바이트)은 **50 Hz(`sp=20`)가 한계**입니다. `sp=10`(100 Hz)을 걸어도 실제로는 줄이 빠지며 약 50 Hz 만 나옵니다.
> 100 Hz 가 필요하면 먼저 통신속도를 230400 으로 올리세요(2.13 절). PC 프로그램도 115200 에서는 `sp=20`, 230400 이상에서는 `sp=10` 을 씁니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

다른 출력 형식:

설정	출력
`cs=1`	CSV 줄 끝에 `*HH` (`*` 앞 글자 전체의 XOR, 16 진 2 자리). PC 프로그램(UI 1.20.0+)은 이 값을 검사해 틀린 줄은 버리고 콘솔에 오류 건수를 남깁니다
`nmea=1`	CSV 대신 `$HEHDT,<방위>,T*hh` 와 `$PASHR,,<방위>,T,<롤>,<피치>,<히브>,<σ 롤>,<σ 피치>,<σ 요>,1*hh`
`sb=1`	28 바이트 고정 바이너리 프레임: `AA 55`, 순번 1, 시간 ms 4, 가속 x y z, 자이로 x y z, 쿼터니언 w x y z (각 int16 리틀엔디언), XOR 1. 가속 ÷8192 [g], 자이로 ÷16 [°/s], 쿼터니언 ÷30000. XOR 는 3 번째 바이트(순번)부터 27 번째 바이트까지

우선순위는 `sb=1` → `nmea=1` → CSV 순입니다. CSV 를 받으려면 `sb·nmea` 가 0 이어야 합니다.

저장되지 않는 것: `sp`, `sd`, `so`, `sb` 는 `fw` 로 저장되지 않습니다. 재부팅하면 스트림은 꺼지고 `sd` 는 0xDD 로 돌아갑니다. `au=1` 을 `fw` 로 저장하면 부팅 직후 `sp=10` 으로 스트림이 시작되지만, `sd` 는 0xDD(17 필드)입니다(115200 에서는 실제 약 50 Hz). PC 프로그램은 연결할 때마다 자기 설정(기능 카드의 '동기 데이터 선택 (sd)')으로 `sp·so·sd` 를 다시 보냅니다.

2.6 설정 저장·공장값 복원 — fw · fd

보내는 것	응답 (성공)	응답 (실패)
`fw`	`fw` → `fw: saving settings to flash...` → `fw done. settings saved (kept after power-off).`	`fw FAILED. flash erase/program error.`
`fd`	`fd` → `fd: restoring factory defaults...` → `fd done. factory defaults restored.`	`fd FAILED. flash error.` 또는 `fd FAILED. no factory record stored.` (공장값이 없음)

- 저장은 약 1 초 걸리고, 그동안 스트림이 멈춥니다.
- **결과 줄(`fw done.` / `fw FAILED.`)을 반드시 확인하세요.** PC 프로그램도 이 줄을 받아야만 저장 성공으로 처리합니다(응답이 없거나 FAILED 면 실패로 처리).
- `fd` 는 출하 때 넣어 둔 공장값(보정·설정)을 현재 값과 사용자 저장 영역에 덮어씁니다. 지금 연결 속도는 바뀌지 않습니다.
- 부팅할 때는 사용자 저장값을 먼저 읽고, 없으면 공장값을 읽습니다.
- 공장값을 새로 기록하는 명령은 양산 공정 전용이라 이 문서에서 다루지 않습니다. 성공 응답은 `fd done. factory defaults stored.` 입니다. PC 프로그램(양산용)의 [공장값 저장]도 이 줄을 받아야만 성공으로 표시하고, 실패·무응답은 따로 알립니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

`fw` 로 저장되는 것:

- 보정값: 지자계(하드·소프트아이언), 가속도·자이로의 바이어스·스케일·3×3 행렬, 온도계수
- 영점(`za`), 장착 회전(`rm`), 편각(`dec`), 온라인 하드아이언 추적값
- 융합 설정(`beta`, `gate`, `mag`, `bt`, `algo`), 기능 토글 전부, 센서 설정(`as gs odr gl al`)
- 통신속도 `b1`·`b2` (다음 부팅 때 그 속도로 바뀜)

저장되지 않는 것: 자이로 영점(부팅마다 다시 잼), 스트림 설정 `sp`·`sd`·`so`·`sb`.

2.7 가속도·자이로 보정 — ca

`ca` 는 **가만히 둔 상태에서 약 2~2.5 초 동안**(출력속도 400 Hz 기준, 최대 6 초) 측정합니다. 장치 문구에는 `~2 s` 로 나옵니다. 무엇을 보정할지는 장치가 상황을 보고 정합니다.

1. **정지 상태**: 자이로 영점(바이어스)을 새로 잡습니다.
2. **한 면이 정확히 위·아래를 향함**: 그 면의 가속도 값을 기억합니다. 같은 축의 반대 면까지 재면 그 축의 바이어스·스케일을 계산합니다. 6 면을 모두 재면 가속도 3×3 행렬(장착 기울기 포함)을 계산합니다.
3. **직전 `ca` 뒤 60 초 안에 한 축만 360° 돌렸음**: 그 축의 자이로 스케일을 계산합니다(이때는 자이로 영점을 새로 잡지 않습니다). 세 축을 양쪽 방향으로 모두 돌리면 자이로 3×3 행렬을 계산합니다.

응답 순서: `ca` → `ca: collecting (~2 s), keep still...` → 결과 줄. 결과 줄이 두 줄일 때가 있습니다(아래).

결과 줄 (예)	뜻
`ca done. Gyro bias: 0.012 -0.004 0.008 dps`	자이로 영점만 갱신 (면 판정 없음)
`ca done. Gyro bias: …; accel +Z face stored (opposite face pending)`	+Z 면을 기억함. 반대 면(–Z)을 재야 계산됨
`ca done. Gyro bias: …; accel -Z face (bias=0.0021 scale=0.9987)`	Z 축 양면 완료. 그 축의 바이어스·스케일
`ca: accel 3x3 matrix solved; mounting r=0.12 p=-0.85 y=0.03 deg`	6 면 완료. 가속도 행렬과 장착 기울기 계산
`ca done. gyro Z scale=1.0012 (measured 359.6' per 360')`	Z 축 360° 회전의 자이로 스케일
`ca: gyro 3x3 matrix solved (scale+cross-axis+alignment)`	세 축 양방향 완료. 자이로 행렬 계산

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

실패·경고 줄	뜻과 조치
`ca FAILED. sensor not ready.`	센서에서 샘플을 충분히 못 받았습니다
`ca FAILED. device moved (var a:…, g:…).`	측정 중 움직였습니다. 가만히 두고 다시
`ca FAILED. gyro mean too large (…dps) - platform turning?`	받침대가 돌고 있습니다(자이로 평균 5 °/s 초과)
`ca FAILED. accel 3x3 matrix solve (degenerate faces) - face data kept, re-measure a face.`	6 면 행렬 계산 실패(FW 2.1.13 이상). 잰 면 데이터는 남아 있으니 한 면만 다시 재면 다시 계산합니다
`ca WARNING: accel X face pair span 0.812 g outside [1,3] - fold skipped.`	행렬이 있는 보드에서 한 축 양면만 다시 잰 경우(2 면 재보정), 두 면이 서로 맞지 않아 반영하지 않았습니다. 이 줄이 결과 줄보다 먼저 오고, 결과 줄은 `(fold skipped: faces inconsistent)` 로 끝납니다

결과 줄이 두 줄 오는 경우

- **6 면째**: 먼저 그 면의 `ca done. Gyro bias: …; accel … face (…)` 가 오고, 바로 다음 줄에 행렬 결과가 옵니다. 성공이면 `ca: accel 3x3 matrix solved; …`, 실패면 `ca FAILED. accel 3x3 matrix solve …` 입니다.
- 실패 줄이 오면 앞의 `ca done.` 을 보았더라도 **실패로 처리**하세요. 이때 자이로 영점과 이번 면 값은 이미 바뀌었고, 행렬만 적용되지 않았습니다.
- **360° 회전**: `ca done. gyro Z scale=…` 다음에, 세 축 양방향이 모두 모였으면 `ca: gyro 3x3 matrix solved …` 가 이어집니다.
- 받는 프로그램은 `ca done.` 을 받은 뒤에도 1~2 초 정도 다음 줄을 더 기다리는 것이 안전합니다.

PC 프로그램(UI 1.20.0)의 [자이로 보정] 창은 `ca FAILED` 줄을 받으면 바로 실패로, `ca WARNING` 줄은 경고로 표시합니다. 6 면째 행렬 계산 실패도 실패로 바꿔 표시하고 [저장하고 닫기]를 숨깁니다.

- **2 면 재보정**(FW 2.1.13 이상): 6 면 행렬이 들어간 보드에서 한 축의 양면(예: +X, −X)만 다시 재면, 그 축의 행만 고쳐 출력에 반영합니다. 보드가 많이 틀어졌으면(두 면 차가 1~3 g 밖) 거절하므로 `rc` 후 6 면을 다시 재세요.
- 보정 결과는 RAM 에만 있습니다. `fw` 로 저장하세요.

2.8 지자계 보정 — cm

`cm` 을 보낸 뒤 **약 30 초 동안 모든 방향으로 천천히(8 자 그리듯) 돌립니다.** 30 초가 지나면 장치가 스스로 끝내고 결과를 알려 줍니다. 진행 중에 `cm` 을 한 번 더 보내면 중단합니다. 지자계가 인식되지 않은 보드에서는 `cm` 이 동작하지 않습니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

응답 순서: `cm` → `cm: rotate slowly in all directions (~30 s)...` → 30 초 뒤 결과 줄.

성공: `cm done. off: 12.31 -40.22 5.87 uT  scale: 1.012 0.987 1.003 (radius 48.0 uT, fit residual 0.6%)` (off = 하드아이언 오프셋, scale = 축별 배율, radius = 주변 자기장 크기, fit residual = 구 모양에서 벗어난 정도)

실패하면 이전 보정값을 그대로 유지합니다. 실패 줄은 모두 `cm FAILED. <사유>; previous calibration kept.` 형식입니다.

사유 (FAILED 뒤 문구)	뜻	조치
`aborted (cm re-sent while running)`	진행 중 `cm` 을 다시 보내 중단함	다시 `cm`
`too few samples`	샘플이 너무 적음	30 초 동안 계속 돌리기
`insufficient 3-axis coverage`	어느 축의 값 범위가 너무 좁음	세 축 모두 크게 돌리기
`rotation too planar (one axis only?) - cover ALL directions`	한 축으로만 돌림	모든 방향으로
`fit singular`	계산이 성립하지 않음	다시 시도
`data does not form an ellipsoid - cover ALL directions`	데이터가 구(타원체) 모양이 아님	모든 방향으로 고르게
`implausible soft-iron shape (move away from large metal)`	찌그러짐이 너무 큼(축 비 2 배 초과)	큰 금속에서 떨어져서
`field magnitude out of range (magnetised surroundings?)`	자기장 크기가 8~200 μT 밖	자석·자화된 물체 확인
`coverage too one-sided - rotate through ALL directions (figure-8)`	한쪽 방향만 돌림(8 분면 중 6 개 미만)	8 자로 모든 방향
`coverage concentrated on one side (cap-only rotation?) - rotate through ALL directions (figure-8)`	한쪽 반구 위주로만 돌림(FW 2.1.12)	8 자로 모든 방향
`poor ellipsoid fit (residual 9.3%) - rotate slowly, away from metal`	구 모양에서 8% 넘게 벗어남	천천히, 금속에서 떨어져서

- 성공한 뒤 `mag=1` 로 지자계 융합을 켜고 `fw` 로 저장하세요.
- 보정 중에는 상태 워드에 0x0001(보정 진행 중)이 켜지고, 끝나면 0x0008(성공) 또는 0x00010000(실패)이 남습니다.

2.9 온도계수 보정 — ct, 보정 초기화 — rc

`ct` 는 **가만히 둔 채 온도를 40 °C 이상 바꾸는 동안**(최대 10 분) 온도에 따른 흐름을 잽니다. 진행 중 `ct` 를 다시 보내면 중단합니다. 시작하고 1 분이 지난 뒤 온도 변화가 40 °C 이상이 되면 끝납니다. 진행 중에는 5 초마다 진행 줄이 나옵니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

```
ct: temperature calibration started. Keep the device perfectly still
    and change its temperature by 40'C or more (10 min max). 'ct' again aborts.
ct: 35s dT=12.3'C (T=41.0'C, need >=40)        <- 5 초마다 (값은 예)
ct done. coef/'C a:…,…,… g:…,…,… m:…,…,…
ct: run 'ca' to refresh the bias, then 'fw' to save.
```

줄	뜻
`ct WARNING: mag windows insufficient (n=12 dT=18.0'C) - mag coef kept.`	지자계 샘플이 부족해 지자계 계수는 바꾸지 않고 기존 값을 유지함(FW 2.1.13 이상). 가속도·자이로 계수는 갱신됨. 이 줄은 `ct done.` 줄 **다음에** 옵니다
`ct FAILED. temperature swing under 40'C (or aborted).`	온도 변화가 40 °C 미만이었거나 중단함

PC 프로그램(UI 1.20.0)의 [온도 보정] 창은 `ct FAILED` 를 받으면 바로 실패로 표시하고, `ct done.` 뒤에 오는 `ct WARNING` 도 받아 경고로 표시합니다.

`rc` 는 보정값을 처음 상태로 돌립니다: `rc done. bias/scale/matrix reset (tco kept; run ca/cm, then fw to save).`

초기화되는 것	그대로 남는 것
가속도 바이어스·스케일·3×3 행렬, 6 면 보정으로 계산한 장착 기울기(`gr`)	온도계수(tco)
자이로 스케일·3×3 행렬, 모으던 면·회전 데이터	지금의 자이로 영점(바이어스)
지자계 오프셋·스케일·소프트아이언 행렬, 온라인 하드아이언 학습값	영점(`za`), 장착 회전(`rm`), 편각(`dec`) 등 설정

- `rc` 도 RAM 에만 적용됩니다. `rc` 뒤에는 필요한 `ca`·`cm` 을 다시 하고 `fw` 로 저장하세요. 저장하지 않고 재부팅하면 마지막 저장값으로 돌아갑니다.
- PC 프로그램(UI 1.20.0)의 [보정 초기화] 확인 창도 같은 내용(초기화되는 것과 남는 것)을 보여 줍니다.

PC 프로그램에서는 [캘리브레이션 마법사], [지자계 보정] 같은 버튼이 이 명령들을 순서대로 보내 줍니다. 보정 절차는 사용설명서(배포용·양산용)를 보세요.

2.10 센서 설정 — as · gs · odr · gl · al, 필터 확인 — imf

명령	뜻	값 (기본값 굵게)
`as`	가속도 측정 범위	0 = ±2 g, **1 = ±4 g**, 2 = ±8 g, 3 = ±16 g
`gs`	자이로 측정 범위	0 = 250, 1 = 500, 2 = 1000, **3 = 2000** °/s
`odr`	센서 출력 속도	0 = 100, 1 = 200, **2 = 400**, 3 = 800 Hz

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

명령	뜻	값 (기본값 굵게)
`gl` / `al`	자이로 / 가속도 칩 내장 저역 필터 대역	**0 = ODR/4**, 1 = ODR/8, 2 = ODR/16, 3 = ODR/32, 4 = ODR/64, 5~7 = ODR/128

출력 속도 400 Hz 일 때 `gl`/`al` 대역은 0 = 100 Hz, 1 = 50 Hz, 2 = 25 Hz, 3 = 12.5 Hz, 4 = 6.25 Hz, 5~7 = 3.1 Hz 입니다. 숫자가 클수록 부드럽지만 늦어집니다.

- **FW 2.1.14 부터 `gl`/`al` 의 뜻이 바뀌었습니다.** 2.1.13 이하에서는 0 이 '필터 없음'이었고, 1 부터 ODR/4 였습니다. 지금은 0 이 가장 넓은 필터(ODR/4)이고 '필터 없음'은 고를 수 없습니다. 예전 펌웨어가 저장한 값은 업데이트 후 첫 부팅 때 자동으로 바뀌어, 저장했던 필터가 그대로 유지됩니다(0 만 '필터 없음' → ODR/4). 펌웨어를 이전 버전으로 되돌려도 보정값은 지워지지 않습니다.
- FW 2.1.14 부터 센서 칩의 안티앨리어싱 필터(AAF)가 항상 켜집니다. 사용자가 바꿀 수 없습니다.
- 설정은 쓴 뒤 칩에서 다시 읽어 확인합니다. 실패하면 `[ERR] sensor cfg write failed (0x04); previous setting kept.` 가 나오고 이전 설정이 유지됩니다. 괄호 안 숫자는 1 = `as`, 2 = `gs`, 4 = `odr`, 8 = `gl`/`al` 의 합입니다.
- PC 프로그램(양산용)에서는 기능 카드의 '센서 설정 (sensor)' 칸에서 바꿉니다. V2 장치를 연결하면 출력속도는 100/200/400/800 Hz 로, LPF 는 지금 출력속도 기준 실제 Hz 로 표시됩니다(FW 2.1.13 이하에서는 0 번이 '필터 없음 ⚠').

`imf` (FW 2.1.14 이상, 읽기 전용)는 칩 안의 필터가 실제로 켜졌는지 보여 줍니다.

```
imf=gsrc:0x5B,asrc:0x36,glpf:0x81,alpf:0x01,odr:400
```

필드	정상인지 보는 법
`gsrc`	자이로 AAF. 비트 6~5 가 2 이면 켜짐 (0x5B = 정상)
`asrc`	가속도 AAF. **아래 2 비트(비트 1~0)가 2 진수 10 이면 켜짐**. 위쪽 비트는 보드마다 다를 수 있어 보지 않습니다(실측 0x36, 예전 예시 0x16 — 둘 다 정상)
`glpf` / `alpf`	자이로 / 가속도 LPF. 아래 3 비트가 1~6 이면 필터 있음 (1 = ODR/4 … 6 = ODR/128), 0 이면 필터 없음
`odr`	지금 출력 속도 [Hz]

위 예는 기본 설정(`gl=0`, `al=0`, 400 Hz)에서 실제 보드로 읽은 정상 값입니다. 정상 기준은 `gsrc:0x5B`, `asrc` 아래 2 비트 = 10, `glpf:0x81`, `alpf:0x01` 입니다. 센서가 응답하지 않으면 `!ERR` 입니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

2.11 진단 명령 — ss · s · can · pt · gr

`ss` — 센서 칩별 상태 (FW 2.1.1 이상, `rdp` 2.1.2 이상, `mfresh` 2.1.12 이상)

```
ss=imu:1,mag:1,imu_id:0xE9,mag_id:0x30,482113,120530,0,0,0,rdp:0xBB,mfresh:1
```

순서	필드	뜻
1	`imu:`	가속도·자이로 센서 1 = 정상, 0 = 없음·고장
2	`mag:`	지자계 1 = 부팅 때 인식됨, 0 = 없음
3·4	`imu_id:` / `mag_id:`	부팅 때 읽은 칩 ID (정상 0xE9 / 0x30)
5·6	숫자	부팅 후 받은 센서 샘플 수 (가속도·자이로 / 지자계)
7·8	숫자	읽기 오류 수 (가속도·자이로 / 지자계)
9	숫자	가속도·자이로 샘플 시간 끊김 수
10	`rdp:`	플래시 읽기보호: 0xAA = 잠기지 않음, 0xBB = 양산 잠금(레벨 1), 0xCC = 레벨 2
11	`mfresh:`	지자계의 마지막 샘플이 0.2 초 안에 들어왔으면 1. 0 이면 지자계가 동작 중에 멈춘 것

- `mag:1` 인데 `mfresh:0` 이면 지자계 칩은 있지만 샘플이 끊겼습니다. 이때 펌웨어는 자동으로 6 축으로 계산합니다.
- `imu:0` 일 때는 지자계 샘플도 받지 않으므로 `mfresh` 를 믿지 마세요.
- PC 프로그램은 V2 장치(펌웨어 2.1.1 이상)이거나 이번 연결에서 `ss=` 로 답한 장치에만 `ss` 를 주기적으로 묻습니다. `ss` 가 없는 장치(V3 등)에는 보내지 않습니다. 답을 받아 제목 아래 센서 칩 배지(`IMU ✓ · MAG ✓`)를 표시합니다. `mfresh:0` 이면 `MAG ⚠ 샘플 끊김 (mfresh:0)`, `rdp:0xAA` 면 `⚠ 읽기보호 없음 (RDP 0xAA) - 출하 전 잠금 필요` 가 나옵니다.

`s` — 상태 워드 (`s=0x2048` 처럼 16 진수)

비트	뜻
0x00000001	보정 진행 중 (`cm`, `ct`)
0x00000008	마지막 보정 성공
0x00000010	자유낙하 (`sh=1`)
0x00000020	충격 (`sh=1`)
0x00000040	스트림 출력 중 (`sp` > 0)
0x00000080	이상 감지 (`ai=1`)
0x00002000	정지 상태
0x00004000	움직이는 중
0x00010000	마지막 보정 실패

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

비트	뜻
0x00020000	입력 전압 경고 (3.7~14 V 밖)
0x00200000	자기 교란
0x02000000	가속도·자이로 센서 오류
0x04000000	지자계 없음·오류

- 0x00020000(입력 전압 경고)은 예전 제품과 맞춘 넓은 기준(3.7~14 V)입니다. **이 보드의 입력은 5 V 전용**(정격 4.75~5.25 V, 5.5 V 이하)이므로 이 비트만 믿지 말고 `pv` 값을 직접 보세요. PC 프로그램은 `pv` 를 5 V 전용 기준으로 색칠합니다(4.2 절).

`can` — **CAN 상태** (USB 로 CAN 쪽 상태를 볼 때)

```
can=1,1520,0,1,0,3,0,127
```

순서대로 컨트롤러 동작(1/0), 송신 프레임 수, 버린 프레임 수, 자가진단(1 = PASS), 버스오프 횟수, 마지막 오류 코드, 수신 오류 수, NMT 상태입니다. 마지막 오류 코드는 0 = 없음, 1 = Stuff, 2 = Form, **3 = ACK 없음(버스에 다른 노드가 없으면 정상)**, 4 = Bit1, 5 = Bit0, 6 = CRC 입니다. NMT 상태는 0 = 부트업, 4 = Stopped, 5 = Operational, 127 = Pre-operational, 255 = CANopen 스택 없음입니다.

그 밖의 진단

명령	응답	뜻
`pt`	`pt=<평균 µs>,<최대 µs>,<ODR Hz>`	센서 샘플 하나당 자세 계산 시간
`gr`	`gr=<롤>,<피치>,<요>`	가속도 6 면 행렬에서 계산한 장착 기울기 [°]
`sn`	`sn=<16 진 24 자리>`	시리얼번호
`hrs`	`hrs=<부팅 횟수>,<누적 시간>,<이벤트 수>,<용량>`	블랙박스 누적 기록

2.12 보정값 백업·복원 — cr · cw

`cr` 은 보정 계수 전체를 그룹별로 한 줄씩 내보내고, 마지막에 `cr done.` 을 보냅니다(21 줄, 약 0.2 초 동안 스트림 멈춤). 그룹 0~8 은 소수 7 자리 유효숫자(`%.7g`), 그룹 9~19 는 float 값을 정확히 되살리는 9 자리(`%.9g`)입니다(FW 2.1.11 이상).

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

```
cr=0,0.001234,-0.003412,0.002105
cr=1,0.012,-0.004,0.008
…
cr=9,1.00213456,-0.00141234,0.00321678
…
cr=19,1,1,0
cr done.
```

(값은 예)

그룹	내용
0 / 1 / 2	바이어스: 가속도 [g] / 자이로 [°/s] / 지자계 하드아이언 오프셋 [μT]
3 / 4 / 5	축별 스케일: 가속도 / 자이로 / 지자계
6 / 7 / 8	온도계수 [/°C]: 가속도 / 자이로 / 지자계
9 / 10 / 11	가속도 3×3 행렬 1·2·3 행
12	가속도 오프셋 (행렬식)
13 / 14 / 15	자이로 3×3 행렬 1·2·3 행
16 / 17 / 18	지자계 소프트아이언 3×3 행렬 1·2·3 행
19	행렬 사용 표시 [가속도 행렬, 자이로 행렬, 0] (1 = 사용)

- 그룹 2 는 온라인 하드아이언 추적(`hi`)으로 배운 값까지 합친 값입니다(FW 2.1.13 이상). 그래서 `/c` 표시값과 조금 다를 수 있지만, 복원하면 출력은 같습니다.

`cw=<그룹>,<x>,<y>,<z>` 는 한 그룹을 장치에 다시 씁니다. 성공하면 `cw=<그룹> ok`, 틀리면 `!ERR` 입니다.

- 그룹은 0~19, 값은 유한한 숫자이고 절댓값이 10000 미만이어야 합니다(nan·inf 거부).
- **한 줄 63 자 한계**를 지키세요(2.1 절). 넘으면 응답 없이 버려집니다. 값은 불필요한 0 을 빼고 짧게 쓰세요(예: `1.0021346`, `-1.234e-05`).
- 복원 순서: **0~8 → 9~18 → 19(맨 마지막)**. 19 를 쓰면 행렬이 켜지고 장착 기울기를 다시 계산하며 온라인 하드아이언 추적값을 지웁니다.
- 쓴 값은 RAM 에만 있습니다. `cr` 로 다시 읽어 맞는지 확인한 뒤 `fw` 로 저장하세요.
- 그룹 1(자이로 영점)은 써도 `fw` 로 저장되지 않습니다. 자이로 영점은 부팅할 때마다 다시 잽니다.

PC 프로그램(양산용)의 [보정값 내보내기]·[보정값 가져오기]가 이 과정을 자동으로 합니다. 가져오기는 쓴 그룹을 모두 다시 읽어 대조한 뒤에만 저장하고, `fw done.` 응답을 확인합니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

2.13 통신속도 바꾸기 — b1 · b2

명령	포트	쓸 수 있는 값
`b1=<속도>`	RS-232 와 RS-485 (같은 값)	9600, 19200, 38400, 57600, 115200, 230400 — **230400 까지**(FW 2.1.17+. RS-232 송수신기 TRS3221E 의 한계 250 kbit/s. RS-485 THVD1420 은 더 빠르지만 `b1` 하나를 두 포트가 같이 쓰므로 RS-232 한계가 상한)
`b2=<속도>`	USB	위 값 + 460800, 921600 (USB 브리지 CP2102 한계 921600)

- 장치는 **먼저 옛 속도로 응답**(`b2=230400`)한 뒤 새 속도로 바꿉니다. 응답을 받은 다음 PC 쪽 속도를 바꾸세요.
- 목록에 없는 값은 무시하고 현재 값을 돌려줍니다. `b1=460800` 처럼 포트 한계를 넘는 값도 같습니다(`b1=115200` 처럼 지금 값이 그대로 돌아옴). 2.1.16 이하 펌웨어는 `b1` 에 460800·921600 을 받아들였는데, 그 속도로 저장된 보드를 2.1.20 로 올리면 부팅 때 그 값을 쓰지 않고 115200 으로 시작합니다.
- 바이너리 쓰기(인덱스 11·12)도 같은 규칙입니다.
- PC 프로그램도 RS-232·RS-485 포트에서는 230400 을 넘는 값을 보내지 않습니다.
- `fw` 로 저장하면 다음 부팅부터 그 속도로 시작합니다. 부팅 첫 글은 115200 으로 나오고, `[OK ] b2=230400 (saved boot baud)` 줄 뒤에 바뀝니다.
- PC 프로그램에서는 상단의 **통신속도** 목록으로 바꿉니다. '자동'이면 연결할 때 장치 속도를 찾아 그대로 씁니다. 연결 중에 다른 값을 고르면 **지금 연결된 포트만** 바꾸고, 새 속도에서 응답이 없으면 이전 속도로 되돌립니다. 지금 값은 기능 카드 '시스템 (system)'의 [속도 읽기]로 확인합니다.

2.14 CAN 노드 ID·속도 바꾸기 — cn · cb (2.1.21+)

한 CAN 버스에 센서를 여러 대 달거나(연구용 다중 센서, 로봇 관절별 센서) 장비의 버스 속도가 500 kbit/s 가 아닐 때 씁니다.

- `cn` → `cn=1` : CANopen 노드 ID 읽기. `cn=3` 처럼 **1~127** 로 씁니다. 모든 CANopen ID 가 함께 움직입니다 — TPDO 0x180+ID … 0x680+ID, 하트비트 0x700+ID, SDO 0x600+ID(요청)/0x580+ID(응답), EMCY 0x080+ID. 예: `cn=3` 이면 자세 PDO 가 0x183·0x283·0x383·0x483·0x683, 하트비트 0x703.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

- `cb` → `cb=500` : CAN 속도(kbit/s). **125 / 250 / 500 / 1000** 만 받습니다. 버스의 모든 장치·마스터와 같아야 합니다.
- 쓰면 응답(`cn=3`) 바로 뒤에 `can: restarted - node 3, 500 kbit/s (fw to keep)` 가 나옵니다. 장치는 CAN 컨트롤러를 새 속도로 다시 켜고 CANopen 스택을 새 ID 로 다시 올립니다. 마스터에는 **부트업(0x700+ID, `00`)이 다시 보이고 Pre-operational** 로 돌아가므로 NMT Start 를 다시 보내야 합니다.
- 범위 밖(`cn=0`, `cn=128`, `cb=800`)은 `!ERR cn=1..127` / `!ERR cb=125|250|500|1000` 로 거부하고 값은 그대로입니다.
- `fw` 로 저장해야 다음 부팅에도 유지됩니다(저장 안 하면 재부팅 때 저장된 값으로 돌아감). 부팅 배너에 실제 값이 나옵니다: `[OK ] CAN 250 kbit/s, selftest PASS` · `[OK ] CANopen node 3: pre-operational, TPDO 0x183..0x483 after NMT start`.
- 2.1.20 이하 펌웨어로 내리면 이 설정은 무시되어 노드 1·500 kbit/s 로 돌아갑니다(설정 블록의 예비 영역에 저장되므로 다른 설정은 영향 없음).
- PC 프로그램(1.21.0+): 기능 카드 **시스템** 그룹의 `CAN 노드 ID (cn)`·`CAN 속도 (cb)` 행에 현재 값이 보이고, 그 아래 `CAN ID` 입력·`CAN 속도` 선택·**[CAN 적용]** 으로 보냅니다(그 뒤 [저장]). PCAN 으로 받는 쪽은 CAN 카드의 **브리지 노드·속도·[브리지 설정]** 을 장치 값에 맞춥니다.
- **속도(`cb`)를 바꾼 뒤에는 PCAN-USB 를 한 번 뽑았다 꽂으세요.** 어댑터가 이전 속도로 열려 있던 동안 오류 프레임을 겪으면 오류 회복 상태로 들어가, 새 속도로 다시 열어도 한동안 PDO 를 20 % 정도만 받고(각 PDO 8~11 Hz) 하트비트를 놓칩니다(2026-10-20 실측: 센서는 250 프레임/초 정상 송신, PCAN 수신만 저하). 뽑았다 꽂으면 바로 47~50 Hz 로 돌아옵니다. 센서 쪽 확인은 USB 로 `can` 을 두 번 읽어 송신 카운터가 초당 약 250 씩 느는지 보면 됩니다.

2.15 펌웨어 업데이트가 내부에서 하는 일 — fu

1. `fu` → 장치가 `fu: rebooting to bootloader (115200 8N1)...` 를 보내고 부트로더로 재시작합니다.
2. 부트로더가 USB 와 RS-232 로 115200 bps 에서 `BL1` 을 0.5 초마다 보냅니다.
3. PC 가 이미지를 1 KB 씩 보냅니다. 장치는 CRC 와 서명을 확인한 뒤 새 펌웨어로 시작합니다.

- PC 프로그램의 '콘솔 · 기록' 카드 [펌웨어 업데이트] 버튼이 이 절차를 대신합니다. 파일 이름은 `AHRS_V2_v2.x.y_<날짜>_App_signed.bin` 입니다(지금 배포본 `AHRS_V2_v2.1.21_20261020_App_signed.bin`).

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

- 업데이트는 USB 또는 RS-232 로만 됩니다(RS-485 불가).
- 전송 중 전원이 끊겨도 부트로더는 남아 있으므로 다시 시도하면 됩니다.
- CAN 으로 업데이트하는 기능(3.7 절 코드 12)은 CAN 부트로더(BL2)가 든 보드에서만 되며, 아직 실기 검증 전입니다.

2.16 구형 프로그램 호환 기능

새로 연동할 때는 쓰지 않아도 됩니다. 구형 프로그램·지그와 맞추려고 남겨 둔 기능입니다.

기능	내용
`c=<n>`	숫자 명령. 1 = `fw`, 2 = `fd`, 3 = `ca`, 4 = `cm`, 5 = `za`, 7 = `ra`, 8 = `rp`, 9 = `rc`, 45 = `ct`, 99 = `rd`. 응답 `c=<n>`, 그 밖의 숫자는 `!ERR`
`/` 명령	`/` 목록, `/d` 현재 값, `/k` 자세 오차, `/c` 보정 계수 표시(PC 프로그램 [보정값 보기]가 사용) 등
`*`	스트림을 3 초 동안 멈춤 (보낼 때마다 연장)
바이너리 프레임	0x02 로 시작하는 구형 공식 프로그램용 프레임. 일반 연동에는 쓰지 않음

구형 프로그램 연동이 필요하면 제품 모델과 펌웨어 버전을 기술지원에 알려 주세요.

3. CAN — CANopen 쉽게 이해하기

3.1 용어 다섯 개만 알면 됩니다

용어	뜻	이 장치에서
NMT (Network Management)	마스터가 노드에게 "시작해/멈춰" 하는 명령. ID 0x000	켜지면 **Pre-operational**(대기). 마스터가 Start 를 보내야 데이터가 나옴
PDO (Process Data Object)	노드가 **주기적으로 알아서 보내는** 실시간 데이터	자세·센서값을 **5 개 메시지**로 **50 Hz** 송출
SDO (Service Data Object)	마스터가 **물어보고 답을 받는** 읽기/쓰기	장치명, 버전, 센서 float 값, 진단 카운터, 보정·저장 명령
하트비트	노드가 "살아 있다"고 주기적으로 알림	ID 0x701, 1 초마다
EMCY (Emergency)	노드의 비상 메시지	센서 오류 때 ID 0x081

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

3.2 기본 설정

- 노드 ID **1**, 속도 **500 kbit/s** 가 출하 기본값, 11 비트 표준 ID. **2.1.21 부터 `cn=`(1~127)·`cb=`(125/250/500/1000) 로 바꿀 수 있고 `fw` 로 저장됩니다(2.14 절)**. 아래의 0x181·0x701 같은 ID 는 노드 1 기준이며 노드 N 이면 0x180+N·0x700+N 입니다.
- 숫자는 모두 리틀엔디언(낮은 바이트 먼저)입니다.
- 버스 양 끝에 120 Ω 종단저항이 필요합니다(보드 안에는 없음).
- 하트비트 주기·PDO 설정 같은 **통신 설정은 RAM 에만** 있어 재부팅하면 기본값으로 돌아갑니다. 센서 보정·설정은 제어 코드 5(3.7 절)로 플래시에 저장합니다.

3.3 켜자마자 보이는 것

- 부팅 직후 ID **0x701**, 데이터 `00` (부트업 알림)
- 그 뒤 1 초마다 ID 0x701, 데이터 `7F` (하트비트: 7F = Pre-operational)

3.4 데이터 받기 — NMT Start 한 줄이면 끝

```
보낼 것 :  ID 0x000  DLC 2  데이터 01 01     (01 = Start, 01 = 노드 1)
받는 것 :  ID 0x701  데이터 05                (하트비트가 05 = Operational 로 바뀜)
           ID 0x181 0x281 0x381 0x481 0x681 이 각각 20 ms 마다 옴
```

NMT 명령 (ID 0x000, 2 바이트)	동작	하트비트
`01 01`	Start — PDO 송출 시작	05
`80 01`	Pre-operational — PDO 멈춤, SDO·제어 가능	7F
`02 01`	Stop — 하트비트 외 대부분 멈춤	04
`81 01`	장치 재부팅	00 → 7F
`82 01`	통신 설정 초기화	00 → 7F

3.5 PDO 5 개 해석표 (각 8 바이트, int16 리틀엔디언)

ID	바이트 0-1	2-3	4-5	6-7	단위 환산
0x181	쿼터니언 W	X	Y	Z	값 ÷ 32767
0x281	롤	피치	요	상태 워드 하위 16 비트 (부호 없음)	각도 = 값 ÷ 100 [°]
0x381	가속도 X	Y	Z	온도	가속도 = 값 ÷ 1000 [g], 온도 = 값 ÷ 100 [°C]
0x481	자이로 X	Y	Z	입력전압 (부호 없음)	자이로 = 값 ÷ 10 [°/s], 전압 = 값 ÷ 1000 [V] (0 = 측정값 없음)

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

ID	바이트 0-1	2-3	4-5	6-7	단위 환산
0x681	지자기 X	Y	Z	지자기 상태 (부호 없음)	지자기 = 값 ÷ 10 [μT] (FW 2.0.1 이상)

- 자세(0x181, 0x281)는 시리얼 `e·q` 와 같은 기준입니다(영점·장착 회전·편각 적용).
- **0x681 지자기 상태**: bit0 = 9 축 융합 중(`mag=1`), bit1 = 지자계 센서 정상, bit2 = 자기 교란, 상위 8 비트 = 지자계 샘플 카운터(하위 8 비트).
- 0x681 은 CiA 301 기본 PDO ID 가 아닌 제조사 지정 ID(0x680 + 노드 번호)입니다. 0x680~0x6FF 는 표준에서 비어 있는 구간이라 다른 노드의 기본 PDO·SDO·하트비트와 겹치지 않습니다.
- 상태 워드 하위 16 비트의 뜻은 2.11 절 표와 같습니다(0x0008 보정 성공, 0x2000 정지, 0x4000 움직임 등). 32 비트 전체는 SDO 0x2400 으로 읽습니다.

예: 0x281 에 `81 00 E2 FF 07 0E 08 20` 이 오면 롤 = 0x0081 = 129 → 1.29°, 피치 = 0xFFE2 = −30 → −0.30°, 요 = 0x0E07 = 3591 → 35.91°, 상태 0x2008(정지 + 보정 성공).

3.6 SDO 로 값 읽기 (물어보기)

요청은 ID **0x601**, 응답은 **0x581**. "읽어 줘"는 첫 바이트 `40`, 그다음 인덱스(리틀엔디언 2 바이트), 서브인덱스입니다.

4 바이트 이하의 값(float 등)은 한 번에 옵니다.

```
가속도 X (float) :  요청 601  40 00 21 01 00 00 00 00      (인덱스 0x2100, 서브 1)
                    응답 581  43 00 21 01 xx xx xx xx      (43 = 4 바이트 값, 뒤 4 바이트 = float32)
```

긴 문자열(장치명, 펌웨어 버전)은 나눠서 옵니다. 펌웨어 버전 "2.1.20"(6 바이트)는 이렇게 읽습니다.

```
요청 601  40 0A 10 00 00 00 00 00      (인덱스 0x100A)
응답 581  41 0A 10 00 06 00 00 00      (41 = 나눠서 보냄, 전체 6 바이트)
요청 601  60 00 00 00 00 00 00 00      (첫 조각 주세요)
응답 581  03 32 2E 31 2E 32 30 00      (03 = 마지막 조각, 데이터 "2.1.20")
```

자주 읽는 인덱스:

인덱스	내용	형식
0x1008	장치명 `ROBOR iAHRS V2`	문자열
0x1009	CAN 사양 버전 `2.1`	문자열
0x100A	펌웨어 버전 (예: `2.1.20`)	문자열

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

인덱스	내용	형식
0x1017	하트비트 주기 [ms] (기본 1000), 쓰기 가능	uint16
0x1018:3	리비전 번호 (예: 0x00020114 = 2.1.20)	uint32
0x1018:4	시리얼 번호 (MCU 고유 ID 로 만든 값)	uint32
0x2100:1~3 / 0x2101:1~3 / 0x2102:1~3	가속도 [g] / 자이로 [°/s] / 지자계 [μT]	float32
0x2103 / 0x2104	온도 [°C] / 입력전압 [V]	float32
0x2105	마지막 센서 샘플 시각 [μs]	uint32
0x2200:1~5	가속도·자이로 샘플 수, 지자계 샘플 수, 가속도·자이로 읽기 오류, 지자계 읽기 오류, 샘플 시간 끊김	uint32
0x2200:6~10	CAN 버린 프레임, CAN 자가진단, 버스오프 횟수, 마지막 오류 코드, 수신 오류	uint32
0x2400	상태 워드 32 비트 전체 (2.11 절)	uint32

잘못 읽거나 쓰면 응답 첫 바이트가 `80`(abort)이고 뒤 4 바이트가 이유입니다.

abort 코드	뜻
0x06020000	없는 인덱스
0x06090011	없는 서브인덱스
0x06010002	읽기 전용에 쓰기
0x06070010	데이터 길이가 맞지 않음
0x06090030	값이 범위 밖
0x08000022	지금 상태에서는 받을 수 없음 (Pre-operational 필요, 또는 앞 명령 처리 중)

3.7 SDO 로 보정·저장 시키기 — 0x2300

0x2300 에 **코드 한 바이트**를 쓰면 됩니다. 결과는 0x2301(숫자)과 0x2302(문장)로 확인합니다.

```
자이로 보정 :  601  2F 00 23 00 01 00 00 00     (0x2300 에 1 쓰기, 응답 581 60 00 23 00 …)
결과 확인   :  601  40 01 23 00 00 00 00 00     (0x2301 읽기: 0 대기, 1 진행 중, 2 성공,
128 실패)
설명 문장   :  0x2302 읽기 (예: "issued ca" → "calibration done")
영점        :  601  2F 00 23 00 0A 00 00 00     (0x2300 에 10 쓰기, 측정 중에도 가능)
영점 해제   :  601  2F 00 23 00 0B 00 00 00     (0x2300 에 11 쓰기)
```


© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

코드	동작	같은 시리얼 명령	받는 NMT 상태	0x2301 결과
1	자이로·가속도 보정 (정지, 약 2~2.5 초)	`ca`	Pre-operational	1 → 2 / 128
2	지자계 보정 시작 (30 초 회전, 자동 종료)	`cm`	Pre-operational	1 → 2 / 128
3	쓰지 않음 (`cm` 은 스스로 끝남)	—	Pre-operational	128
4	지자계 보정 중단	`cm` 다시 보내기	Pre-operational (코드 2 진행 중)	128 (중단 = 실패, 이전 보정 유지)
5	설정 플래시 저장	`fw`	Pre-operational	1 → 2 (기록됨) / 128 (기록 실패)
6	공장값 복원	`fd`	Pre-operational	1 → 2 / 128
7	보정값 초기화	`rc`	Pre-operational	2
8 / 9	지자계 융합 켬 / 끔	`mag=1` / `mag=0`	Pre-operational	2
10	영점 (지금 자세를 0°로)	`za`	Pre-operational, **Operational** (FW 2.0.6 이상)	2
11	영점 해제	`ra`	Pre-operational, **Operational** (FW 2.0.6 이상)	2
12	부트로더로 재시작 (CAN 펌웨어 업데이트)	`fu`	Pre-operational, BL2 부트로더 필요 (FW 2.1.0 이상)	BL1 이면 128

- **범위 밖 코드(0, 13 이상)**는 abort 0x06090030, 지금 상태에서 받을 수 없는 코드는 0x08000022 입니다.
- 코드 1~9·12 는 Pre-operational 에서만 받습니다. 먼저 `80 01` 로 바꾸세요. 10·11 은 측정 중(Operational)에도 받습니다.
- 앞 명령(1, 2, 5, 6)이 진행 중이면 새 코드는 0x08000022 로 거절됩니다. 단 코드 2 진행 중의 코드 4(중단)는 받습니다.
- **저장(5)·공장값 복원(6)의 결과**: 쓰고 나면 0x2301 이 1(진행)이고, 플래시 기록이 끝나면 2(기록됨) 또는 128(기록 실패)이 됩니다. **1 이 아닐 때까지 읽으세요**(보통 0.1 초 안). 5 초 안에 결과가 없으면 128 과 `ERR no result of the flash write` 가 됩니다.
- 보정(1, 2)은 끝날 때까지 1 이고, 90 초 안에 끝나지 않으면 128 과 `ERR timeout (IMU absent?)` 가 됩니다.
- 코드 1(`ca`)이 6 면째에서 행렬 계산에 실패하면(2.7 절) 128(실패)이 됩니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

- 장치가 다른 보정을 하는 중에 들어온 요청은 받아 두었다가 그 보정이 끝난 뒤 실행합니다.
- 0x2302 문장 예: `issued fw` → `fw done` / `fw FAILED (flash error)`, `calibration done` / `calibration FAILED`, `ERR cm finishes by itself (~30 s)`(코드 3), `ERR no cm running`(진행 중이 아닌데 코드 4), `ERR bootloader without CAN (BL1): update over USB`(코드 12, BL1 보드).
- 영점(10)은 시리얼 `za` 와 같아서, 지자계 융합이 켜져 있으면 요는 유지하고 롤·피치만 0 으로 맞춥니다(2.3 절). 영점은 RAM 에만 있으니 전원을 꺼도 유지하려면 Pre-operational 로 바꾼 뒤 5(저장)를 쓰세요.

3.8 비상 메시지(EMCY)·정의 파일

- 가속도·자이로 센서 접근 오류가 0.5 초 넘게 이어지면 ID **0x081** 로 EMCY(오류 코드 0x1000)를 보냅니다. 오류가 풀리면 해제 메시지를 보냅니다.
- 제품과 펌웨어 버전에 맞는 PDO·객체 정의 파일은 기술지원에 요청하세요. 제공된 파일의 적용 버전을 확인한 뒤 CANopen 마스터 소프트웨어나 CAN 분석기에 불러오세요.
  - `IAHRS_V2_CANopen.eds` (객체사전 전체), `IAHRS_V2.dbc` (PDO 디코딩), `canopen_objects.json`
- CANopen 연동 시 이 문서의 객체 사전과 실제 장치 버전을 함께 확인하세요.

4. PC 프로그램(UI 1.21.0)에서 쓰기

PC 프로그램은 `AHRS_Monitor.exe`(배포용), `AHRS_Monitor_EN.exe`(영문)입니다. 이 장에서는 통신과 관련된 부분만 설명합니다. 화면 전체는 「iAHRS V2 사용설명서」와 「iAHRS V2 UI 사용가이드」를 보세요.

- 영문판은 화면과 콘솔 안내가 모두 영어입니다. 장치가 보내는 응답 문구는 원래 영어라 어느 판에서나 같습니다.
- 창 제목은 연결 전·해제 후에 `iAHRS V2 / V3 모니터`(양산용은 `iAHRS V2 / V3 — 양산 · 캘리브레이션`)이고, 장치를 알아보면 `iAHRS V2 모니터` 처럼 바뀝니다.

4.1 콘솔 명령 입력줄

'콘솔 · 기록' 카드 아래의 입력줄(>)에 명령을 쓰고 Enter 또는 [전송]을 누르면 장치로 그대로 보냅니다. ↑ ↓ 키로 이전 명령을 다시 불러옵니다. 보낸 명령은 `-> vr` 처럼 표시되고, 그 아래에 장치 응답이 나옵니다. 터미널 프로그램 없이 이 문서의 명령을 바로 시험할 수 있습니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

콘솔 명령 입력줄 — vr·ss·e 응답 예 (데모 화면, UI 1.20.0 · FW 2.1.20 응답)

- 같은 카드 위쪽에 [● 기록], [CSV 저장], [기록 목록], [연결 진단], [펌웨어 업데이트] 버튼이 있습니다.
- 스트림 값은 콘솔에 찍지 않고 3D·계기·그래프로 보냅니다. 응답이 많은 명령을 칠 때 스트림이 방해되면 먼저 `sp=0` 을 보내세요.
- 연결이 안 되거나 값이 안 들어오면 [연결 진단]을 누르세요. 연결 단계(포트 열기 → 속도 탐색 → 장치 식별 → 설정 동기화 → 수신 확인), 실패 사유, 응답 지연이 보이고, [진단 파일 저장]으로 문의용 파일을 만들 수 있습니다.

4.2 상단 상태 표시

연결하면 프로그램이 `vr`, `hp`, `ss`, `sn` 등을 자동으로 물어 상단에 표시합니다.

표시	어디서 오나	예
제품·융합 방식 (제목 아래 부제목)	`vr` 응답, `algo·mag` 값	연결 전 `장치 연결 전`, 알아보는 중 `장치 확인 중 · ESKF 6 축`, 알아본 뒤 `RB-SDA-v2 · IMU + MAG · ESKF 6 축`
연결·수신 상태	연결 결과, 마지막으로 값을 받은 시각	`정상 수신 · 마지막 수신 0.02 초 전` (값이 끊기면 `수신 중단`)
통신 방식·속도	`hp` 응답 (1.3 절)	`USB @ 115200 bps`, CAN 연결이면 `CAN 500 kbit/s · PCAN-USB · Node 1`
수신 속도	초당 받은 줄 수	`50.0 Hz`
시리얼번호	`sn` 응답	앞 4 자리 … 뒤 4 자리 (마우스를 올리면 24 자리 전체)
하드웨어·펌웨어	`vr` 응답 (1.2 절)	`HW V2 · FW v2.1.20 ✓` (연결 전에는 이 프로그램이 아는 최신 버전 `최신 FW V2 2.1.20 · V3 3.0.15` — 프로그램 버전에 따라 다름)
UI 버전	프로그램 자체	`UI v1.20.0 (b 빌드일)`
누적 가동	`hrs` 응답	`가동 12h · 34 회`

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

표시	어디서 오나	예
센서 칩 상태 (제목 아래)	부팅 글, `ss`, `s`, `st`	`IMU ✓ · MAG ✓`
읽기보호 경고 (제목 아래)	`ss` 의 `rdp:0xAA`	`⚠ 읽기보호 없음 (RDP 0xAA) - 출하 전 잠금 필요`

설정을 바꾸고 아직 `fw` 로 저장하지 않았으면 화면 위에 저장 안내 줄(`변경 N 건이 장치에 적용됐지만 저장되지 않았습니다.`)과 [되돌리기]·[저장] 버튼이 나타납니다.

프로그램이 스스로 보내는 명령 중 장치 설정을 바꾸는 것:

- 연결 직후 장치가 `bt=0` 이면 한 번 `bt=1` 로 켜고 콘솔에 `(자동) 자이로 바이어스 추적 켬 (bt=1) — yaw 드리프트 방지` 를 남깁니다. 저장(`fw`)은 하지 않습니다. 지그 자동 캘리브레이션이 실행 중일 때는 켜지 않습니다(2.4 절 `bt`).
- 연결할 때마다 스트림 설정 `sp·so·sd` 를 프로그램 설정으로 다시 보냅니다(2.5 절).
- 3D 카드의 '지자계' 체크를 바꾸면 `mag=` 를 보내고, 1.5 초 뒤 `za` 를 자동으로 보내 영점을 다시 잡습니다(콘솔 `(자동) 영점 재설정 — 지자계 전환 반영`).

기능 카드 '라이브'의 **입력전원 V**(`pv`)는 5 V 전용 입력 기준으로 색이 바뀝니다. 외부 전원은 실제보다 약 0.2 V 높게 읽힙니다.

읽힌 값	색	뜻
4.1 ~ 5.45 V	보통	정상
5.45 V 초과	주황	정격(5.25 V)보다 높음
5.7 V 초과	빨강	절대 최대(5.5 V)에 닿았거나 넘음. 바로 전원을 확인하세요
4.1 V 미만	주황	전원 부족
3.7 V 미만	빨강	전원 부족(장치의 입력 전압 경고 기준과 같음)

4.3 USB 없이 CAN 으로 받기 — [CAN 연결]

상단 툴바의 **[CAN 연결]**을 누르면 USB 케이블 없이 CAN 버스(PCAN-USB)로 센서값을 받습니다(V2 전용).

1. 프로그램이 장치에 NMT Start 를 보내 Operational 로 만듭니다(PDO 5 종이 50 Hz 로 흐름).
2. 브리지가 PDO 를 풀어 USB 스트림과 같은 형식으로 넘겨 주므로, 3D·계기·그래프·CSV 기록이 그대로 동작합니다.
3. SDO 0x100A 로 펌웨어 버전을 읽어 `HW V2 · FW v2.1.20 ✓` 처럼 표시합니다. 통신 방식 칸에는 `CAN 500 kbit/s · PCAN-USB · Node 1` 과 수신 Hz 가 보입니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

4. 오른쪽 도크의 CAN / CANopen 카드가 자동으로 펼쳐집니다(4.4 절).

항목	USB 연결	CAN 연결
자세(롤·피치·요·쿼터니언), 가속도, 자이로	O	O (PDO)
지자기(mx, my, mz)	O	O (0x681, FW 2.0.1 이상)
온도·입력전압·상태	O	CAN 카드에 표시
[영점]·[영점 해제]·[저장]	O	O — 같은 버튼 (FW 2.0.6 이상)
보정 시작·중단, 공장값 복원, 융합 켬/끔	O	CAN 카드의 제어(0x2300)로
설정 변경, [캘리브레이션 마법사], [펌웨어 업데이트]	O	X — USB 로

- CAN 연결에서 [저장]을 누르면 Pre-operational 로 바꿔 코드 5 로 저장한 뒤 다시 Start 합니다. 그동안 측정값 전송이 1~2 초 멈춥니다. 결과는 콘솔에 세 가지 중 하나로 나옵니다.
  - `fw done. settings saved (CAN control 5)` — 저장됨
  - `fw FAILED. (CAN control 5) …` — 저장 실패(뒤에 사유)
  - `fw sent, result not confirmed (CAN control 5) …` — 명령은 보냈지만 결과를 읽지 못함. USB 로 연결해 확인하세요
- [연결 해제]로 끝냅니다. 장치는 Operational 로 남아 PDO 를 계속 보냅니다. 멈추려면 CAN 카드의 [Pre-op]를 누르세요.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

4.4 CAN / CANopen 카드

V2 장치가 연결되면 오른쪽 도크에 **CAN / CANopen** 탭이 생깁니다. [펼치기 ◂]를 누르면 카드가 옆으로 펼쳐지고, [접기 ▸]로 닫습니다.

CAN / CANopen 카드 — 데모 화면 (브리지 응답을 모의로 넣어 찍은 그림, 실제 장치 값 아님)

카드는 세 칸입니다.

1. **장치 CAN 상태 (USB 'can' 명령, 2 초마다)** — USB 만 있으면 됩니다. 컨트롤러, 자가진단, NMT 상태, 송신, 드롭, 버스오프, 마지막 오류, 수신 오류를 보여 줍니다(2.11 절 can). "마지막 오류: ACK 없음(상대 노드 없음)"은 버스에 다른 노드가 없을 때 정상입니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

2. **CANopen 마스터 (브리지)** — PC 에 PCAN-USB 가 꽂혀 있어야 합니다. 노드 상태, 하트비트, PDO 속도(0x181~0x681 각각의 Hz)를 보여 줍니다. ▶ Start, Pre-op, Stop, 통신 리셋 버튼이 있고, 롤/피치/요, 쿼터니언, 가속도, 자이로, 지자기, 온도·VIN·상태, 비상 메시지가 실시간으로 보입니다.
3. **SDO 읽기 · 보정 제어 (브리지)** — 인덱스(hex)·서브·형식(float32 / unsigned / signed / 문자열)을 넣고 [읽기]를 누르면 아무 값이나 읽습니다(예: 2100:1 가속도 X, 1008:0 장치명). '제어 (0x2300)' 목록에서 코드 1, 2, 4~11 을 골라 [실행]을 누르면 3.7 절의 제어를 보내고, 결과(진행 중 / 성공 / 실패)와 0x2302 문장을 1 초마다 보여 줍니다. 1~9 는 먼저 [Pre-op]를 누르세요.

PCAN-USB 가 없으면 "PCAN-USB 가 연결되지 않았습니다. 꽂으면 자동으로 켜집니다." 가 나옵니다.

4.5 CAN 브리지

- `AHRS_Monitor.exe` / `AHRS_Monitor_EN.exe` 는 실행되면 CAN 브리지(`can_bridge.exe`)를 스스로 띄우고, 종료할 때 같이 닫습니다. PCAN-USB 만 꽂으면 됩니다.
- 브리지가 아직 안 떠 있으면 카드에 "CAN 브리지가 아직 시작되지 않았습니다 (앱이 자동으로 띄웁니다. 계속 꺼져 있으면 앱을 다시 실행하세요)." 가 나옵니다.
- PC 에 PEAK 드라이버가 설치되어 있어야 합니다.
- 브리지는 이 PC 안에서만 접근되는 작은 서버(127.0.0.1:8766)입니다. 다른 CAN 프로그램(PCAN-View 등)과 동시에 PCAN-USB 를 쓸 수 없습니다.
- 브라우저 기반 연동 환경에서 별도 CAN 브리지가 필요하면 기술지원에 해당 모델용 실행 자료를 요청하세요.

5. 자주 묻는 것

- **CAN 에서 아무것도 안 온다** → 종단저항(양 끝 120 Ω), 500 kbit/s 설정, NMT Start(`000: 01 01`)를 보냈는지 확인하세요. CAN 카드의 "마지막 오류"가 계속 ACK 없음이면 상대 노드가 없는 것입니다.
- **PDO 가 오긴 오는데 느리다(각 10 Hz 안팎, 하트비트 간격이 몇 초)** → 속도(`cb`)를 바꾼 직후 PCAN 어댑터가 오류 회복 중인 상태입니다. PCAN-USB 를 뽑았다 꽂은 뒤 다시 [CAN 연결]하세요(2.14 절). 센서는 정상 송신 중이므로 USB `can` 의 송신 카운터는 초당 약 250 씩 늘어납니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

- **CAN 제어(0x2300)가 abort 0x08000022 로 거절된다** → 코드 1~9·12 는 Pre-operational 에서만 받습니다. `80 01` 을 보낸 뒤 다시 쓰세요. 앞 명령이 진행 중이어도 거절됩니다.
- **요(yaw)가 천천히 돈다** → 먼저 `vr` 로 펌웨어를 보세요. 2.1.12~2.1.14 이면 최신(2.1.21)으로 업데이트하세요. 지자계 융합이 꺼진 6 축 모드(`mag=0`)에서는 요가 상대각입니다. 자이로 영점 자동 맞춤(`bt=1`)이 정지 구간에서 영점을 맞추므로 가만히 있을 때는 흐르지 않지만, 초당 0.4° 미만의 아주 느린 일정한 회전은 영점 변화와 구분할 수 없어 덜 나타날 수 있습니다(2.4 절). 방향이 중요하면 `cm` 보정 후 `mag=1`, `fw`.
- **정지해 있는데 롤·피치가 1 초마다 툭툭 튄다(0.1~0.4°)** → 융합이 Madgwick(`algo=0`)인지 `algo` 로 확인하세요. Madgwick 은 정지 상태에서도 보정 걸음이 일정 크기(β·dt)라 50 Hz 출력에서 롤·피치가 0.1~0.4° 씩 흔들릴 수 있습니다(2026-10-09 실측: 명령을 보내지 않아도 3 분에 0.2° 넘는 변화 72 회, 명령을 보내도 60 회 — 명령 응답이 원인이 아니라 Madgwick 방식 자체의 특성). ESKF 는 같은 조건에서 변화 0 회·떨림 0.01° 이하. `algo=1`(ESKF, 기본) → `fw`. PC 프로그램에서는 기능 카드의 'ESKF 융합'을 켜고 [저장].
- **지자계 값이 200 μT 넘게 나온다** → 주변 자석·철·전류의 영향이거나 하드아이언입니다. `cm` 보정이 상쇄합니다. 보정 전 값은 방위에 쓰지 마세요.
- **`cm FAILED.` 가 나온다** → 2.8 절 표에서 사유를 찾아 조치하세요. 실패해도 이전 보정값은 그대로입니다.
- **센서 칩 배지에 '샘플 끊김 (mfresh:0)'이 뜬다** → 지자계가 동작 중에 멈췄습니다. 장치는 자동으로 6 축으로 계산합니다. 전원을 다시 넣어도 반복되면 문의하세요.
- **USB 스트림이 50 Hz 밖에 안 나온다** → 2.5 절. 115200 에서는 50 Hz 가 한계입니다. `b2=230400` 으로 올리세요.
- **RS-485 에서 응답이 겹친다** → 반이중이므로 장치가 스트림을 보내는 동안 명령을 보내면 충돌합니다. `sp=0` 으로 멈추고 명령한 뒤 다시 켜세요.
- **PLC 에 `pv=`, `s=` 같은 줄이 섞여 들어온다** → USB 에 연결된 PC 프로그램이 묻는 값의 응답입니다. 응답은 세 포트 모두로 나갑니다(1.1 절). 모르는 줄은 버리도록 만드세요.
- **긴 명령(`cw=…`)에 응답이 없다** → 한 줄 63 자를 넘었습니다(2.1 절). 값을 짧게 쓰세요.
- **명령에 `!ERR` 가 온다** → 없는 명령이거나 형식이 틀렸습니다. `h` 로 목록을 확인하세요. 구형 펌웨어에는 없는 명령일 수 있습니다(부록 A).
- **설정이 재부팅하면 사라진다** → `fw` 를 보내지 않았습니다(CAN 에서는 코드 5). 스트림 설정 `sp`·`sd` 는 원래 저장되지 않습니다.

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

- **`gl`/`al` 숫자의 뜻이 예전과 다르다** → FW 2.1.14 부터 0 이 ODR/4 입니다(2.10 절).
- **정지 노이즈·진동이 크다 / 칩 필터가 켜졌는지 확인하고 싶다** → `imf` 를 보내세요(2.10 절). `gsrc:0x5B`, `asrc` 아래 2 비트 = 10, `glpf:0x81`, `alpf:0x01`(기본 `gl=0·al=0`)이면 칩 안의 안티앨리어싱 필터와 저역 필터(100 Hz)가 켜진 정상 상태입니다. 2.1.13 이하는 이 필터가 꺼져 있었으니 업데이트하세요.
- **`ca done.` 을 받았는데 PC 프로그램은 실패라고 한다** → 6 면째에 `ca done.` 다음 줄로 `ca FAILED. accel 3x3 matrix solve …` 가 온 경우입니다(2.7 절). 한 면을 다시 재세요.
- **`rc` 를 했는데 자이로 영점과 온도계수가 그대로다** → 정상입니다. `rc` 는 둘 다 지우지 않습니다(2.9 절). 자이로 영점은 정지 상태에서 `ca` 로 다시 잽니다.
- **PC 프로그램의 입력전원 칸이 주황·빨강이다** → 이 보드는 5 V 전용 입력입니다. 4.2 절 표의 기준으로 전원을 확인하세요.

부록 A. 펌웨어 버전별 프로토콜 변화

연동하는 장치의 펌웨어가 낮으면 아래 기능이 없을 수 있습니다. 버전은 `vr` 또는 SDO 0x100A 로 확인합니다.

펌웨어	프로토콜에서 바뀐 것
2.0.1	CAN 지자기 PDO 0x681
2.0.2	`hp` (연결 포트·속도)
2.0.3	스트리밍 중 명령 응답이 빠지던 문제 수정 (필수 업데이트)
2.0.6	CANopen 제어 10(영점)·11(영점 해제)을 Operational 에서도 받음. 저장(5)·공장값 복원(6)의 실제 기록 결과를 0x2301 로 알림
2.1.1	`ss` 칩별 상태, CANopen 제어 12(CAN 업데이트, BL2 필요), 부팅 자이로 보정 약 1 초
2.1.2	`ss` 끝에 `rdp:`
2.1.4·2.1.5	사용 금지 버전 (2.1.6 으로 대체). 2.1.7~2.1.10 은 배포되지 않음
2.1.11	`cr`/`cw` 그룹 0~19, 한 줄 63 자, `cw` 비정상 값 거절(`!ERR`), `cm` 타원체 피팅과 `cm FAILED.` 사유 표시, SDO 0x100A 가 실제 버전을 따름
2.1.12	`ss` 끝에 `mfresh:`, `cm` 한쪽 반구 회전 거절(`coverage concentrated on one side`)
2.1.13	6 면 행렬 계산 실패 시 `ca FAILED.`, 2 면 재보정이 행렬에 반영(`ca WARNING`), `ct WARNING`(지자계 계수 유지), `[ERR] sensor cfg write failed`, `cr` 그룹 2 에 온라인 하드아이언 포함, 끊긴 바이너리 프레임 뒤 명령 복구

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr

펌웨어	프로토콜에서 바뀐 것
2.1.14	`imf` 신설, `gl`/`al` 새 의미(0 = ODR/4, '필터 없음' 선택 불가), 센서 칩 AAF 항상 켬, `[ERR] IMU IREG setup not verified`, SDO 0x100A = "2.1.14"
2.1.15	명령·응답 형식은 2.1.14 와 같음. 자이로 영점 자동 맞춤(`bt`) 새 방식(정지 구간에서만 맞춤, 움직임에서 온 값은 확정 전까지 되돌릴 수 있게 보관 — 2.4 절), `geo=` 계산 때 내부 메모리(스택)가 넘칠 수 있던 문제 수정. SDO 0x100A = "2.1.15", 0x1018:3 = 0x0002010F
2.1.17 (2026-10-09 배포, 2.1.20 으로 대체)	명령·응답 형식은 2.1.14 와 같음. `b1` 은 230400 까지만 수용(2.13 절) — 넘는 값은 `b1=<지금 값>` 으로 응답. `b2` 는 921600 까지
2.1.21 (현재 배포, 2026-10-20)	`cn`(CANopen 노드 ID 1~127)·`cb`(CAN 속도 125/250/500/1000 kbit/s) 신설 — 쓰면 CAN 이 바로 다시 시작되고 `fw` 로 저장(2.14 절). 범위 밖은 `!ERR`. 그 외 명령·응답 형식 변화 없음
2.1.20 (2026-10-10 배포, 2.1.21 로 대체)	`algo=` 가 0/1/2 만 받고 그 외는 `!ERR`(이전엔 0 외 모든 값 → 1). `gv` 조회 신설. `algo=2` 에서 `eu=unavailable`, 바이너리 오차 3 개는 NaN. 그 외 형식 변화 없음
2.1.19 (2.1.20 에 포함)	형식 변화 없음. 응답이 송신 큐를 거쳐 나가므로 응답 중에도 스트림이 끊기지 않고(`imu_gaps` 증가 없음), `b1=`/`b2=` 응답은 **바뀌기 전 속도로** 끝까지 나간 뒤 속도가 바뀜, `rd`·`fu` 응답도 재부팅 전에 끝까지 나감(10-09 실기 확인). 스트림은 포트 큐에 45 ms 분량 이상 밀리면 그 줄을 버림(이전의 '바쁘면 버림'과 같은 정책)
2.1.18 (2.1.20 에 포함)	형식 변화 없음. 상태 워드 `ST_MAG_DISTURB`(0x00200000)가 자기장 **방향** 외란 거부도 포함(이전엔 크기만). 융합 동작 변경은 양산용 설명서 부록 E
2.1.16	명령·응답 형식은 2.1.14 와 같음. Madgwick(`algo=0`)에서 명령 응답 중 계산이 밀릴 때 롤·피치가 튀던 크기를 줄임(ESKF 는 변화 없음). 2.1.15 내용 포함. SDO 0x100A = "2.1.16", 0x1018:3 = 0x00020110. 실기 검증 중

© 2026 ROBOR Co., Ltd. All rights reserved. | 주식회사 로버 · www.robor.co.kr