

ECAN-Lite-4CH 产品说明书

EtherCAT 转 4 路 CAN FD 网关模块

版本: V1.0

日期: 2026-03

一、产品概述

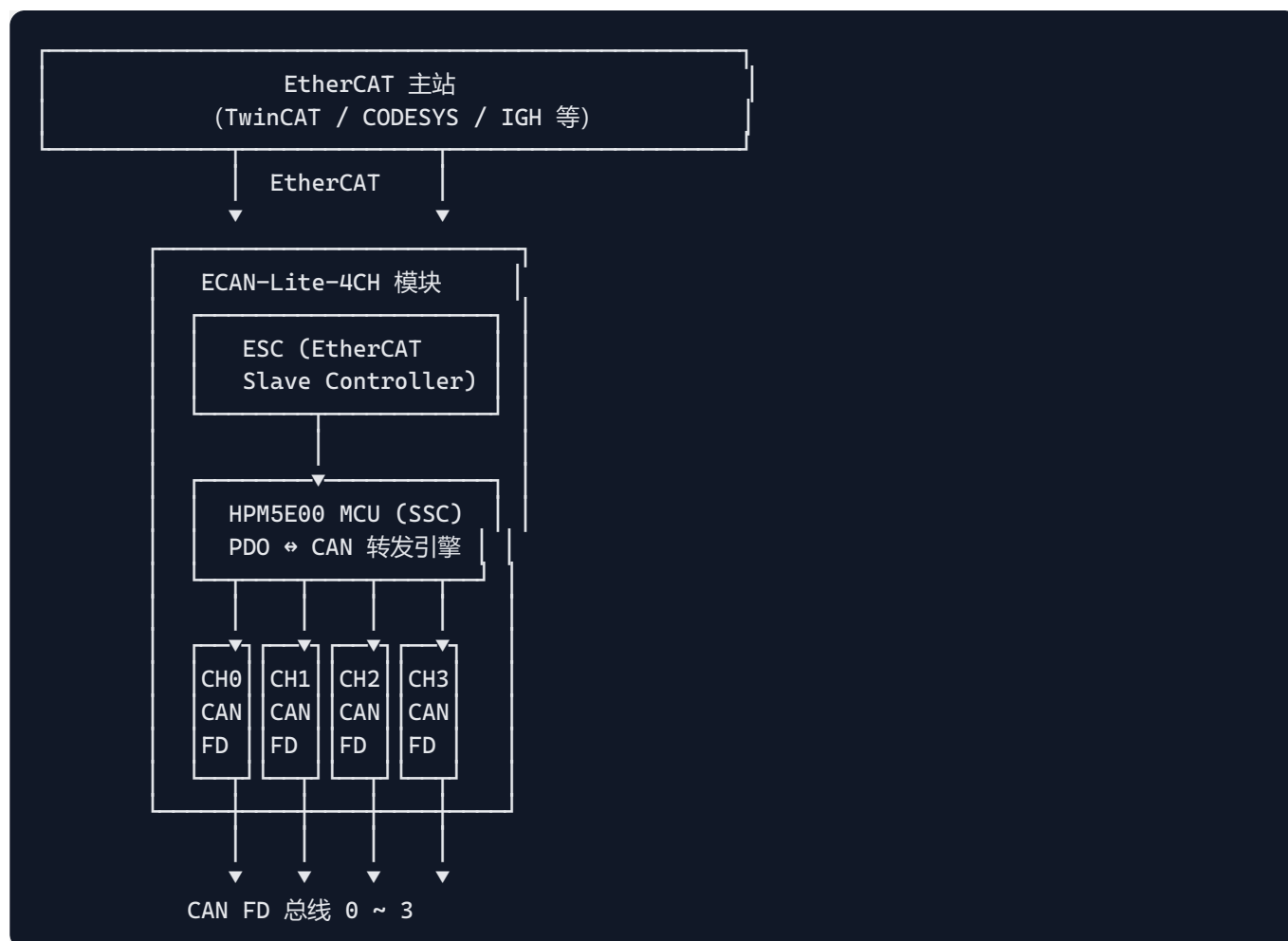
ECAN-Lite-4CH 是一款基于先楫 HPM5E00 系列高性能 RISC-V MCU 的工业级网关模块，实现 **EtherCAT** 总线与 **4 路独立 CAN FD** 通道之间的实时双向数据转发。

本产品作为 EtherCAT 从站设备，可无缝集成到 TwinCAT、CODESYS、IGH 等主流 EtherCAT 主站系统中，将 EtherCAT 网络的确定性实时通信能力延伸至 CAN / CAN FD 现场总线，适用于机器人、工业自动化、发散电机运动控制、电池管理系统 (BMS) 等需要同时使用 EtherCAT 和 CAN FD 总线的应用场景。

核心特性

特性	说明
EtherCAT 接口	2 端口 EtherCAT 从站 (100 Mbps, 支持线型拓扑)
CAN FD 通道数	4 路独立 CAN FD 通道
协议支持	CAN 2.0A / CAN 2.0B / CAN FD (最大 64 字节数据域)
波特率	仲裁段: 最高 1 Mbps; 数据段: 最高 8 Mbps (CAN FD)
过程数据	每通道每周期可传输 2 帧 CAN/CAN FD 报文
通道配置	支持通过 SDO 在线配置波特率、过滤器、时间戳等参数
参数持久化	通道配置可保存至 Flash, 上电自动恢复
接收过滤器	每通道最多 64 条标准 ID 过滤规则 + 24 条扩展 ID 过滤规则
时间戳	支持 MCAN 硬件时间戳与 TSU 高精度时间戳
寄存器诊断	通过 SDO 可直接读写底层 MCAN 寄存器 (调试用途)
同步模式	支持 Free Run / SM-Sync / DC-Sync (Sync0/Sync1)
主控芯片	HPMicro HPM5E3Y (RISC-V, 480 MHz, 1 MB Flash)

二、系统架构



三、接口说明

3.1 EtherCAT 接口

- **端口数量:** 2 个 RJ45 端口 (Port 0 / Port 1)
- **速率:** 100 Mbps 全双工
- **拓扑支持:** 线型菊花链拓扑
- **LED 指示灯:**
 - **PWR LED (红色):** 电源指示灯, 上电常亮
 - **LINK0 LED (绿色):** EtherCAT Port 0 PHY 连接状态指示
 - **LINK1 LED (绿色):** EtherCAT Port 1 PHY 连接状态指示

3.2 CAN FD 接口

- **通道数量:** 4 路独立通道 (CH0 ~ CH3)
- **接口类型:** 标准 CAN 收发器接口
- **信号定义:** CANH / CANL
- **终端电阻:** 板载可贴 120Ω 终端电阻

四、EtherCAT 对象字典

本产品的 EtherCAT 对象字典遵循 CoE（CANopen over EtherCAT）协议，主要对象索引分配如下：

4.1 对象索引总览

索引范围	对象名称	说明
0x1600	Output Mapping 0	RxPDO 映射（主站 → 从站）
0x1A00	Input Mapping 0	TxPDO 映射（从站 → 主站）
0x1C12	SM2 Assignment	SyncManager 2（输出）分配
0x1C13	SM3 Assignment	SyncManager 3（输入）分配
0x6000	Input data	TxPDO 输入过程数据
0x7000	Output data	RxPDO 输出过程数据
0x8000	System Config	系统配置对象
0x8001~0x8004	MCAN Channel 0~3 Config	各通道 CAN FD 配置
0x9000~0x9003	MCAN Channel 0~3 Information	各通道运行状态信息
0xA000~0xA003	MCAN Channel 0~3 Register Service	各通道寄存器读写服务

4.2 系统配置对象（0x8000）

子索引	名称	类型	访问	说明
1	Reboot command	UINT8	RW	写 1 触发系统重启
3	CPU voltage	UINT16	RW	CPU 电压（只读参考值）
4	CPU clock hz	UINT32	RW	CPU 主频（Hz）
5	MCAN0 clock hz	UINT32	RW	CH0 MCAN 源时钟（Hz）
6	MCAN1 clock hz	UINT32	RW	CH1 MCAN 源时钟（Hz）
7	MCAN2 clock hz	UINT32	RW	CH2 MCAN 源时钟（Hz）
8	MCAN3 clock hz	UINT32	RW	CH3 MCAN 源时钟（Hz）
9	Config version	UINT32	RW	配置版本号
10	Last apply result	UINT32	R	最后一次应用配置的结果状态

4.3 通道配置对象（0x8001 ~ 0x8004）

每个通道拥有独立的配置对象，子索引结构相同。以 0x8001（CH0）为例：

命令控制

子索引	名称	类型	访问	说明
1	Apply command	UINT8	RW	写 1 应用当前参数到运行时
2	Apply status	UINT8	R	应用结果 (0=成功, 1=失败)
3	Flash command	UINT8	RW	写入保存命令, 将配置持久化到 Flash
4	Flash status	UINT8	R	Flash 操作结果
5	Command sequence	UINT32	R	命令序列号 (自增计数)

通信参数

子索引	名称	类型	默认值	说明
7	Bit timing mode	UINT8	0	0=自动计算; 1=手动设定时序参数
8	Node mode	UINT8	0	0=Normal; 1=Listen Only; 2=Loopback 内部; 3=Loopback 外部
9	Non-ISO enable	UINT8	0	启用 Non-ISO CAN FD 模式
10	Transmit pause	UINT8	0	发送暂停使能
11	Edge filter enable	UINT8	0	边沿过滤器使能
12	Protocol exception disable	UINT8	0	禁用协议异常处理
13	Wide message marker enable	UINT8	0	宽消息标记使能
14	External timestamp use	UINT8	0	使用外部时间戳
15	FD enable	UINT8	1	1=启用 CAN FD; 0=仅 CAN 2.0
16	TDC enable	UINT8	0	传输延迟补偿使能
17	Restricted mode disable	UINT8	0	禁用受限操作模式
18	Disable auto retransmission	UINT8	0	禁用自动重传
19	Nominal bitrate	UINT32	500000	仲裁段波特率 (bps)
20	Data bitrate	UINT32	5000000	数据段波特率 (bps, CAN FD)
21	CAN2.0 sample point min x10	UINT16	750	CAN 2.0 采样点下限 (×10, 750=75.0%)
22	CAN2.0 sample point max x10	UINT16	875	CAN 2.0 采样点上限 (×10, 875=87.5%)
23	CAN FD sample point min x10	UINT16	750	CAN FD 采样点下限 (×10)
24	CAN FD sample point max x10	UINT16	875	CAN FD 采样点上限 (×10)

手动时序参数 (Bit timing mode = 1 时有效)

子索引	名称	类型	说明
25	Nom BRP	UINT16	仲裁段波特率预分频
26	Nom TSEG1	UINT8	仲裁段时间段 1
27	Nom TSEG2	UINT8	仲裁段时间段 2
28	Nom SJW	UINT8	仲裁段同步跳转宽度
29	Data BRP	UINT8	数据段波特率预分频
30	Data TSEG1	UINT8	数据段时间段 1
31	Data TSEG2	UINT8	数据段时间段 2
32	Data SJW	UINT8	数据段同步跳转宽度
33	TDC offset	UINT8	TDC 偏移
34	TDC filter	UINT8	TDC 滤波

全局过滤器配置

子索引	名称	类型	默认值	说明
36	Accept non-match std	UINT8	0	不匹配标准帧处理 (0=接收到 FIFO0, 1=接收到 FIFO1, 2=丢弃)
37	Accept non-match ext	UINT8	0	不匹配扩展帧处理
38	Reject remote std	UINT8	1	拒绝标准帧远程帧
39	Reject remote ext	UINT8	1	拒绝扩展帧远程帧
40	Ext ID mask	UINT32	0x1FFFFFFF	扩展 ID 全局掩码

过滤器条目编辑窗口

过滤器通过"编辑窗口"方式操作，每次选中一条条目进行查看/编辑：

子索引	名称	类型	访问	说明
41	Std list max	UINT8	R	标准过滤器条目上限（最大 64）
42	Ext list max	UINT8	R	扩展过滤器条目上限（最大 24）
43	Std current index	UINT8	RW	当前标准过滤器编辑索引
44	Ext current index	UINT8	RW	当前扩展过滤器编辑索引
45	Std entry cmd	UINT8	RW	标准条目命令（1=加载, 2=保存, 3=清除, 4=写入 Flash）
46	Ext entry cmd	UINT8	RW	扩展条目命令
47	Std entry status	UINT8	R	标准条目命令执行结果（0=成功）
48	Ext entry status	UINT8	R	扩展条目命令执行结果（0=成功）
49	Std filter count	UINT8	RW	启用的标准过滤器条目数量
50	Std filter type	UINT8	RW	标准过滤器类型（Range/Dual ID/Classic）
51	Std filter action	UINT8	RW	标准过滤器动作（接收到 FIFO0/FIFO1 等）
52	Std filter sync	UINT8	RW	标准过滤器同步消息标志
53	Std ID1	UINT32	RW	标准过滤器 ID1（过滤 ID 或范围起始 ID）
54	Std ID2	UINT32	RW	标准过滤器 ID2（掩码或范围结束 ID）
55	Ext filter count	UINT8	RW	启用的扩展过滤器条目数量
56	Ext filter type	UINT8	RW	扩展过滤器类型
57	Ext filter action	UINT8	RW	扩展过滤器动作
58	Ext filter sync	UINT8	RW	扩展过滤器同步消息标志
59	Ext ID1	UINT32	RW	扩展过滤器 ID1
60	Ext ID2	UINT32	RW	扩展过滤器 ID2

缓冲区配置

子索引	名称	类型	默认值	说明
61	RX FIFO0 op mode	UINT8	0	FIFO0 操作模式 (0=阻塞, 1=覆写)
62	RX FIFO0 size	UINT8	48	RX FIFO0 深度
63	RX FIFO1 op mode	UINT8	0	FIFO1 操作模式 (0=阻塞, 1=覆写)
64	RX FIFO1 size	UINT8	12	RX FIFO1 深度
65	RX FIFO0 data size	UINT8	64	RX FIFO0 数据域大小 (字节)
66	RX FIFO1 data size	UINT8	64	RX FIFO1 数据域大小 (字节)
67	RX buffer count	UINT8	12	RX Buffer 数量
68	RX buffer data size	UINT8	64	RX Buffer 数据域大小 (字节)
69	TX FIFO queue mode	UINT8	0	TX FIFO/Queue 模式 (0=FIFO, 1=Queue)
70	TX FIFO queue size	UINT8	16	TX FIFO 深度
71	TX dedicated buffer count	UINT8	16	TX 专用缓冲区数量
72	TX buffer data size	UINT8	64	TX 缓冲区数据域大小 (字节)
73	TX event FIFO size	UINT8	32	TX 事件 FIFO 深度

时间戳配置

子索引	名称	类型	说明
74	Timestamp selection	UINT8	时间戳源选择
75	Timestamp prescaler	UINT8	时间戳预分频
76	TSU enable	UINT8	TSU 时间戳单元使能
77	TSU use external timebase	UINT8	使用外部时基源
78	TSU capture on SOF	UINT8	在 SOF（帧起始）时捕获时间戳
79	TSU enable 64-bit	UINT8	启用 64 位时间戳
81	TSU prescaler	UINT16	TSU 预分频系数
82	TSU external timebase source	UINT8	TSU 外部时基源选择
83	TSU TBSEL option	UINT8	TSU 时基选择选项
84	Timeout enable	UINT8	超时使能
85	Timeout select	UINT8	超时源选择
87	Timeout period	UINT16	超时周期

4.4 通道信息对象 (0x9000 ~ 0x9003)

只读对象，返回各通道 MCAN 控制器的实时状态寄存器原始值：

子索引	名称	类型	说明
1	ECR raw	UINT32	错误计数寄存器（TEC / REC）
2	PSR raw	UINT32	协议状态寄存器
3	IR raw	UINT32	中断寄存器
4	RXF0S raw	UINT32	RX FIFO0 状态
5	RXF1S raw	UINT32	RX FIFO1 状态
6	TXFQS raw	UINT32	TX FIFO/Queue 状态
7	TXEFS raw	UINT32	TX Event FIFO 状态
8	HPMS raw	UINT32	高优先级消息状态

4.5 寄存器服务对象 (0xA000 ~ 0xA003)

用于直接读写底层 MCAN 控制器寄存器（调试/诊断用途）：

子索引	名称	类型	访问	说明
1	Offset	UINT16	RW	寄存器偏移地址 (0x0004~0x0404)
2	command	UINT8	RW	1=读取, 2=写入
3	status	UINT8	R	操作结果状态
4	Write value	UINT32	RW	待写入值
5	Read value	UINT32	R	读取结果
6	Timestamp low	UINT32	R	操作时间戳 (低 32 位)
7	Timestamp high	UINT32	R	操作时间戳 (高 32 位)

五、过程数据（PDO）映射

5.1 TxPDO（从站 → 主站，输入方向）

TxPDO 用于将 CAN FD 总线接收到的报文上传至 EtherCAT 主站。

总大小：784 字节 (4 通道 × 196 字节/通道)

每通道 196 字节结构：

通道头部（28 字节）

偏移	长度	名称	说明
0	2	Session ID	会话标识
2	2	Counter	帧计数器
4	4	Timestamp (s)	秒级时间戳
8	4	Timestamp (ns)	纳秒级时间戳
12	2	State Bits	通道状态位
14	1	RX FIFO0 Status	RX FIFO0 状态
15	1	RX FIFO1 Status	RX FIFO1 状态
16	1	TX FIFO Status	TX FIFO 状态
17	1	TX Event Status	TX 事件状态
18	1	ECR TEC	发送错误计数
19	1	ECR REC	接收错误计数
20	2	PSR	协议状态
22	6	Reserved	保留

CAN 帧数据 (84 字节 × 2 帧)

每帧结构:

偏移	长度	名称	说明
0	4	Last TX TimeOffset	上一次发送时间偏移
4	1	Last TX Sequence	上一次发送序号
5	1	Last TX Marker	上一次发送标记
6	2	RX Sequence	接收有效帧计数
8	4	RX TimeOffset	接收时间偏移
12	2	Status	帧状态
14	2	RX Ctrl	接收控制字 (含 DLC、BRS、FDF 等位域)
16	4	CAN ID	CAN 报文 ID
20	64	Data	CAN 数据域 (最大 64 字节)

RX Ctrl 控制字位域:

位	名称	说明
[3:0]	DLC	数据长度码
[4]	BRS	波特率切换标志
[5]	FDF	FD 格式标志
[6]	SI	帧来源索引
[14:8]	FIDX	匹配过滤器索引
[15]	ANMF	非匹配标志

5.2 RxPDO (主站 → 从站, 输出方向)

RxPDO 用于从 EtherCAT 主站下发 CAN FD 报文至指定通道。

总大小: 672 字节 (4 通道 × 168 字节/通道)

每通道 168 字节 = 2 帧 × 84 字节/帧。

每帧结构:

偏移	长度	名称	说明
0	2	Session ID	会话标识
2	1	TX Sequence	发送序号（用于去重）
3	1	TX Marker	发送标记
4	10	User Tag	用户自定义标签
14	2	TX Ctrl	发送控制字
16	4	CAN ID	CAN 报文 ID
20	64	Data	CAN 数据域（最大 64 字节）

TX Ctrl 控制字位域：

位	名称	说明
[3:0]	DLC	数据长度码
[4]	BRS	波特率切换使能
[5]	FDF	FD 格式使能
[6]	TSCE	时间戳捕获使能
[7]	EFC	事件 FIFO 控制
[15:8]	DI	数据索引

六、快速入门

6.1 硬件准备

1. **ECAN-Lite-4CH 模块**
2. **PC**：安装 TwinCAT 3.1（Build 4024.56 或更高），配备兼容 Intel 网卡
3. **CAN FD 分析仪**（如 PCAN、Vector CANalyzer 等）或 CAN FD 终端设备
4. **网线**：CAT5e 及以上规格
5. **CAN 总线线缆**：带 120Ω 终端电阻

6.2 接线说明

1. 使用网线将 PC 网口连接到模块 EtherCAT Port 0
2. 将 CAN FD 设备连接到所需通道的 CANH/CANL 引脚
3. 确保 CAN 总线两端配置 120Ω 终端电阻
4. 通过 USB 或外部电源为模块供电

6.3 TwinCAT 配置步骤

步骤 1: 安装 ESI 文件

将产品附带的 ESI 文件（`ECAT-CANFD.xml`）复制到 TwinCAT 安装目录：

```
C:\TwinCAT\3.1\Config\Io\EtherCAT\
```

提示： ESI 文件中 `<Device Physics="YY">` 属性应与实际使用的 EtherCAT 端口数匹配。2 端口为 `"YY"`，3 端口为 `"YYY"`。

步骤 2: 创建 TwinCAT 工程

1. 打开 TwinCAT，选择 **File → New → Project → TwinCAT Project**
2. 首次使用需安装 TwinCAT 实时网卡驱动

步骤 3: 扫描从站设备

1. 在项目树中右键 **Device → Scan**
2. 选择已安装实时驱动的网卡
3. TwinCAT 将自动识别 ECAN-Lite-4CH 模块

步骤 4: 配置 CAN FD 通道参数

在 TwinCAT 中双击从站设备，进入 **CoE - Online** 页面：

1. **设置波特率：** 写入 `0x8001:Nominal bitrate` = 仲裁段波特率，`0x8001:Data bitrate` = 数据段波特率
2. **启用 CAN FD：** 确认 `0x8001:FD enable` = 1
3. **应用配置：** 写入 `0x8001:Apply command` = 1
4. **保存到 Flash (可选)：** 写入 `0x8001:Flash command` = 1

重复以上操作配置 0x8002 ~ 0x8004 对应 CH1 ~ CH3。

步骤 5: 启动从站

1. 点击 **Activate Configuration** 激活配置
2. 切换到 **Run Mode**
3. 确认从站进入 **OP (Operational)** 状态
4. LINK LED 常亮绿色表示对应端口链路已建立

6.4 数据收发示例

从主站发送 CAN FD 报文

在 TwinCAT 中写入 RxPDO (0x7000) 对应通道的帧数据：

1. 设置 `CAN ID` = 目标 CAN ID
2. 设置 `TX Ctrl` 位域：DLC、FDF=1 (CAN FD)、BRS=1 (波特率切换)

3. 填充 **Data** 字段 (最多 64 字节)
4. 更新 **TX Sequence** 触发发送

从总线接收 CAN FD 报文

读取 TxPDO (0x6000) 对应通道的帧数据:

1. 检查 **RX Sequence** 是否递增 (表示有新帧到达)
2. 读取 **CAN ID**、**RX Ctrl**、**Data** 字段获取报文内容
3. 通道头部提供 **ECR TEC/REC** 错误计数用于总线诊断

七、参数配置流程

7.1 在线修改波特率

1. 写入 0x8001:Nominal bitrate = 新的仲裁段波特率 (如 1000000)
2. 写入 0x8001:Data bitrate = 新的数据段波特率 (如 5000000)
3. 写入 0x8001:Apply command = 1 (立即生效)

7.2 配置接收过滤器

以为 CH0 添加一条标准 ID 过滤规则为例:

1. 写入 0x8001:Std current index = 0 → 选中标准过滤器条目 #0
2. 写入 0x8001:Std entry cmd = 1 → 加载当前条目到编辑窗口 (Load)
3. 写入 0x8001:Std filter type = 过滤类型 → 如 Range / Dual ID / Classic
4. 写入 0x8001:Std filter action = 过滤动作 → 如 接收到 FIFO0
5. 写入 0x8001:Std ID1 = ID1 → 过滤 ID 或范围起始 ID
6. 写入 0x8001:Std ID2 = ID2 → 掩码或范围结束 ID
7. 写入 0x8001:Std entry cmd = 2 → 保存条目 (Save)
8. 写入 0x8001:Std filter count = 1 → 设置启用的过滤器数量
9. 写入 0x8001:Apply command = 1 → 使过滤器生效

7.3 持久化配置到 Flash

1. 完成通道参数配置并 Apply 成功后
2. 写入 0x8001:Flash command = 1
3. 读取 0x8001:Flash status 确认 = 0 (成功)

配置保存后, 设备下次上电将自动加载 Flash 中的参数, 无需重新配置。

7.4 恢复默认配置

如需恢复出厂默认参数:

1. 不执行 Flash 保存操作
2. 写入 **0x8000:Reboot command** = 1 触发系统重启
3. 若 Flash 中无有效数据, 设备将使用内置默认参数启动

八、LED 指示灯说明

LED	颜色	状态	含义
PWR	红色	灭	未供电
PWR	红色	常亮	电源正常
LINK0	绿色	灭	EtherCAT Port 0 未连接
LINK0	绿色	常亮	EtherCAT Port 0 PHY 链路已建立
LINK1	绿色	灭	EtherCAT Port 1 未连接
LINK1	绿色	常亮	EtherCAT Port 1 PHY 链路已建立

九、技术规格

参数	规格
主控芯片	HPMicro HPM5E3Y (RISC-V, 480 MHz)
Flash	1 MB 片内 Flash
EtherCAT 端口	2 × RJ45 (100 Mbps)
CAN FD 通道	4 路独立通道
CAN 协议	CAN 2.0A / CAN 2.0B / CAN FD (ISO 11898-1:2015)
仲裁段波特率	最高 1 Mbps
数据段波特率	最高 8 Mbps
CAN 数据域	最大 64 字节 (CAN FD)
TxPDO 大小	784 字节 (4 × 196B)
RxPDO 大小	672 字节 (4 × 168B)
每通道每周期帧数	2 帧
标准过滤器	每通道最多 64 条
扩展过滤器	每通道最多 24 条
EtherCAT 同步	Free Run / SM-Sync / DC-Sync
配置持久化	Flash 存储, 支持 CRC32 校验
工作温度	-20°C ~ +70°C (工业级)
供电电压	5V DC (通过 USB 或外部电源)

十、常见问题

Q1: 从站无法进入 OP 状态?

- 检查 EtherCAT 网线连接是否正常
- 确认 ESI 文件已正确安装
- 检查 TwinCAT 实时网卡驱动是否已安装
- 查看 PWR LED 是否常亮（电源正常）、LINK LED 是否常亮（链路已建立）

Q2: CAN 总线无法通信?

- 确认通道波特率配置与对端设备一致
- 检查 CAN 总线终端电阻（总线两端各 120Ω）
- 通过 0x9000 对象读取 ECR（错误计数），确认无总线错误
- 确认 `Apply command` 已执行（`0x8001:Apply command` = 1）

Q3: 配置重启后丢失?

- 需执行 `0x8001:Flash command` = 1 将配置持久化到 Flash
- 读取 `0x8001:Flash status` 确认保存成功

Q4: 如何诊断 CAN 控制器状态?

- 读取 0x9000~0x9003 通道信息对象获取实时状态
- 使用 0xA000~0xA003 寄存器服务对象直接读取 MCAN 寄存器
- TxPDO 通道头部包含 TEC/REC 错误计数和 PSR 协议状态

Q5: 如何切换 CAN 2.0 与 CAN FD 模式?

- 写入通道配置对象的 `FD enable` 字段（如 `0x8001:FD enable`）：
- 0 = 仅 CAN 2.0 模式（最大 8 字节数据）
- 1 = CAN FD 模式（最大 64 字节数据）
- 执行 `Apply command` = 1 使配置生效

十一、修订历史

版本	日期	说明
V1.0	2026-03	初始版本