

MP6543 + MA600A + ESP32 硬件诊断程序

版本: 1.0
适配: ESP32 Arduino Core 2.x / 3.0+
功能: 对 MP6543 三相无刷电机驱动器和 MA600A 14bit 磁编码器进行硬件级诊断

目录

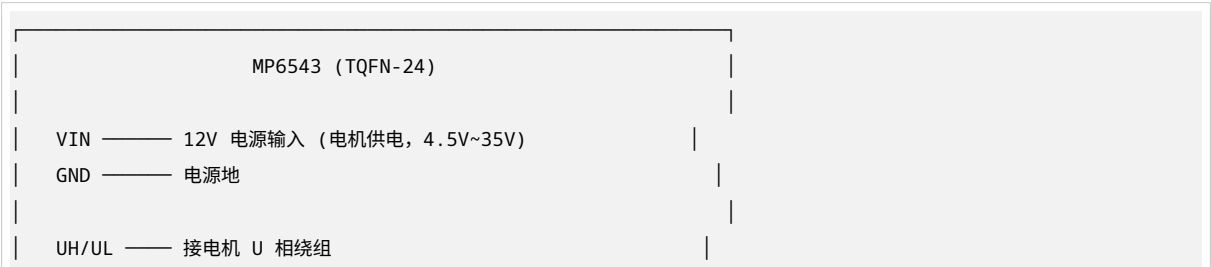
- 1. 硬件清单
- 2. 引脚接线图
- 3. Arduino IDE 配置
- 4. 程序功能说明
- 5. 使用步骤
- 6. 常见问题排查
- 7. 技术细节

硬件清单

序号	器件	型号/规格	数量	说明
1	主控板	ESP32-S3 / ESP32-WROOM	1	支持 Arduino 框架
2	电机驱动器	MP6543H / MP6543	1	三相半桥驱动, 内置 电流检测
3	磁编码器	MA600A	1	14bit 绝对式, SPI 接口
4	无刷电机	8mm 空心杯 / 任意三相无刷	1	极对数需与代码匹配
5	磁铁	径向充磁, 直径 6~8mm	1	安装在电机轴末端, 对准编码器
6	电流采样运放	AD820 / MCP6002 / 任意单电源 运放	3	用于放大 MP6543 差分电流信号
7	电阻	10K Ω (上拉)	若干	CS、nFAULT 上拉
8	电容	100nF (去耦)	若干	电源去耦
9	杜邦线	母对母 / 公对母	若干	连接各模块

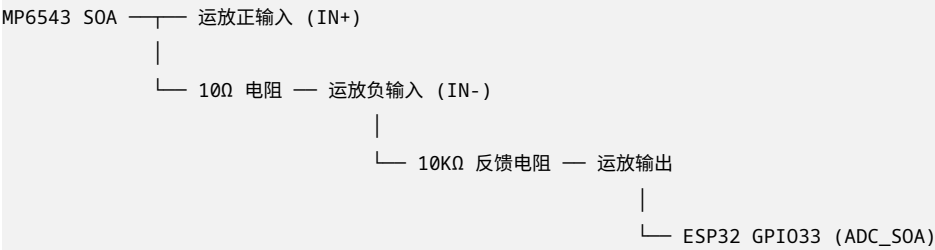
引脚接线图

1. MP6543 驱动器接线



VH/VL	——	接电机 V 相绕组	
WH/WL	——	接电机 W 相绕组	
INA	——	ESP32 GPIO15 (PWMA_PIN)	← PWM 信号输入
INB	——	ESP32 GPIO16 (PWMB_PIN)	← PWM 信号输入
INC	——	ESP32 GPIO17 (PWMC_PIN)	← PWM 信号输入
EN	——	ESP32 GPIO10 (EN_PIN)	← 使能控制 (高有效)
nFAULT	——	ESP32 GPIO4 (nFAULT_PIN)	← 故障输出 (低有效, 需上拉)
SOA	——	运放输入 A	← A 相电流检测 (差分信号)
SOB	——	运放输入 B	← B 相电流检测 (差分信号)
SOC	——	运放输入 C	← C 相电流检测 (差分信号)
VREF	——	3.3V (参考电压)	
nSLEEP	——	接 VCC (常醒) 或 GPIO (休眠控制)	

电流采样运放接线 (以 AD820 为例, 每相一个):



运放电源: V+ = 3.3V, V- = GND
偏置电压: Vref/2 = 1.65V (通过两个 10KΩ 电阻分压)
零点输出: ~1.65V (无电流时)

2. MA600A 编码器接线

MA600A (SOP-8)			
VDD	——	3.3V (编码器供电)	
GND	——	地	
CS	——	ESP32 GPIO6 (SPI_CS)	
SCK	——	ESP32 GPIO7 (SPI_SCK)	
MOSI	——	ESP32 GPIO8 (SPI_MOSI/SDI)	
MISO	——	ESP32 GPIO9 (SPI_MISO/SDO)	
注意: CS 引脚必须外接 4.7K~10KΩ 上拉电阻			
到 3.3V, 防止上电时误触发			

3. 完整接线汇总表

信号名	ESP32 引脚	连接目标	功能	备注
PWMA	GPIO15	MP6543 INA	A 相 PWM	20kHz, 10bit
PWMB	GPIO16	MP6543 INB	B 相 PWM	20kHz, 10bit
PWMC	GPIO17	MP6543 INC	C 相 PWM	20kHz, 10bit
EN	GPIO10	MP6543 EN	驱动器使能	高电平有效
nFAULT	GPIO4	MP6543 nFAULT	故障检测	开漏, 需上拉
ADC_SOA	GPIO33	运放输出 A	A 相电流	ADC1_CH5
ADC_SOB	GPIO34	运放输出 B	B 相电流	ADC1_CH6
ADC_SOC	GPIO35	运放输出 C	C 相电流	ADC1_CH7
SPI_SCK	GPIO7	MA600A SCK	SPI 时钟	最大 8MHz
SPI_MOSI	GPIO8	MA600A MOSI	SPI 主出	数据输入编码器
SPI_MISO	GPIO9	MA600A MISO	SPI 主入	数据输出编码器
SPI_CS	GPIO6	MA600A CS	SPI 片选	低有效, 需上拉

Arduino IDE 配置

1. 安装 ESP32 开发板支持

- 打开 Arduino IDE → 文件 → 首选项
- 在 附加开发板管理器网址中添加:

```
https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json
```

- 打开 工具 → 开发板 → 开发板管理器
- 搜索 ESP32, 安装 **esp32 by Espressif Systems** (推荐 3.0.0+)

2. 选择开发板配置

配置项	推荐设置	说明
开发板	ESP32S3 Dev Module	根据实际芯片选择
CPU Frequency	240MHz	最高性能
Flash Mode	QIO 80MHz	四线 IO 模式
Flash Size	8MB (64Mb)	根据实际 Flash 选择
Partition Scheme	Huge APP (3MB No OTA)	最大程序空间
PSRAM	Enabled	启用外部 RAM (如有)
Upload Speed	921600	最高上传速度
Port	选择对应 COM 口	根据设备管理器确定

3. 编译上传

- 选择正确的 端口 (工具 → 端口)
- 点击 上传按钮 (或按 Ctrl+U)
- 上传完成后, 打开 串口监视器 (Ctrl+Shift+M)
- 设置波特率为 **115200**
- 按 **RST** 按钮重启 ESP32

程序功能说明

程序包含 4 个独立诊断测试，通过串口命令触发：

测试 1: MA600A 编码器通信测试 (1)

目的: 验证编码器 SPI 通信和角度读取是否正常

步骤:

1. 初始化 HSPI 总线 (8MHz, Mode3)
2. 连续读取两次角度值，检查一致性
3. 20 次连续读取，观察角度变化

通过标准:

- 两次读取值不为 0x0000 或 0xFFFF
- 手动转动电机轴时，角度值平滑变化

失败排查:

- 检查 CS 引脚是否有 4.7K Ω 上拉电阻
- 检查 SPI 四线连接是否正确 (SCK/MOSI/MISO/CS)
- 检查编码器供电是否为 3.3V
- 检查磁铁是否在编码器正上方 0.5~2mm

测试 2: MP6543 电流采样测试 (2)

目的: 验证电流采样电路和故障检测功能

步骤:

1. 未使能驱动器时读取三相零点电压 (~1.65V)
2. 使能驱动器后再次读取，确认无漂移
3. 检查 nFAULT 引脚状态

通过标准:

- 三相零点电压均在 1.5V~1.8V 范围内
- 使能前后零点变化 < 0.2V
- nFAULT 为高电平 (无故障)

失败排查:

- 零点偏差大: 检查运放偏置电压是否为 $VCC/2$
- nFAULT 为低: 检查是否过流/过温，或 nFAULT 引脚是否悬空未上拉

测试 3: PWM 输出测试 (3)

目的: 验证三路 PWM 信号是否正常输出到 MP6543

步骤:

1. 初始化三个 LEDC 通道 (20kHz, 10bit)
2. 依次给 A/B/C 相输出 30% 占空比 PWM, 持续 1 秒

通过标准:

- 用示波器测量 PWMA/PWMB/PWMC 引脚, 可观察到 20kHz 方波
- 用万用表直流档测量, 电压约为 $3.3V \times 30\% = 1.0V$

失败排查:

- 无 PWM 输出: 检查引脚是否被其他功能占用
- 频率不对: 检查 LEDC 通道配置

测试 4: 开环转动测试 (4)

目的: 验证电机和驱动器整体是否能驱动电机转动

步骤:

1. 初始化 PWM 并使能驱动器
2. 按六步换相顺序输出 PWM, 每步 50ms
3. 执行 60 步 (10 圈)

安全警告:

- ⚠️ 电机将会转动! 确保轴自由无负载!
- ⚠️ 开环无反馈, 若相序错误电机抖动但不转

通过标准:

- 电机按预期方向平稳转动
- 无异常噪音或发热

失败排查:

- 电机不转但有嗡嗡声: 相序错误, 交换任意两相接线
- 电机抖动: 换相速度过快/过慢, 调整 `step_delay`
- 电机发热: 降低 `pwm_val` 占空比

使用步骤

首次使用流程

1. 按接线图完成硬件连接
 - └─ 特别注意: CS 和 nFAULT 引脚必须接上拉电阻!
2. 打开 Arduino IDE, 选择正确的开发板和端口

3. 上传 Hardware_Diagnostic.ino 到 ESP32
4. 打开串口监视器 (波特率 115200)
5. 按 ESP32 的 RST 按钮重启
 - 应看到程序标题和版本信息
6. 依次执行测试:
 - 发送 '1' → 检查编码器通信
 - 发送 '2' → 检查电流采样
 - 发送 '3' → 检查 PWM 输出 (可用示波器验证)
 - 发送 '4' → 开环转动测试 (注意安全!)
7. 全部通过后, 可上传 FOC 控制程序

命令列表

命令	功能	安全等级
1	编码器通信测试	★ 安全
2	电流采样测试	★ 安全
3	PWM 输出测试	★ 安全 (电机不转)
4	开环转动测试	⚠ 警告 (电机会转)
a	执行全部测试	⚠ 包含转动测试

🔍 常见问题排查

Q1: 编码器读取始终为 0 或 65535

可能原因:

- CS 引脚未接上拉电阻, 导致上电时芯片被误选中进入异常状态
- SPI 线序错误 (MOSI/MISO 接反)
- 编码器供电不足 (MA600A 需要 3.0~3.6V)
- 磁铁未对准或距离过远 (>2mm)

解决方案:

- 在 CS 引脚和 3.3V 之间焊接 4.7KΩ 电阻
- 用万用表通断档确认 SPI 四线连接
- 测量编码器 VDD 引脚电压是否为 3.3V
- 调整磁铁位置, 确保在编码器感应区域内

Q2: 电流采样零点偏差很大 (>2.0V 或 <1.0V)

可能原因:

- 运放偏置电压未设置为 $VCC/2$
- 采样电阻或增益电阻值错误
- MP6543 未正确使能

解决方案:

1. 检查运放同相输入端的偏置电路 (两个等值电阻分压)
2. 确认反馈电阻阻值与设计的增益匹配
3. 确保 EN 引脚已拉高使能驱动器

Q3: PWM 测试时电机意外转动

可能原因:

- MP6543 的 INL 引脚 (下桥臂逻辑输入) 未正确处理
- 三相 PWM 时序错误导致直流母线短路

解决方案:

1. 确保 MP6543 的 nSLEEP 引脚接高电平
2. 若使用 MP6543H (带内部逻辑), INL 引脚可悬空或接高
3. 测试时先断开电机, 仅测量 PWM 波形

Q4: 开环转动时电机抖动但不转

可能原因:

- 相序错误 (三相接线顺序不对)
- 换相速度不匹配 (过快或过慢)
- 电压不足 (占空比太低)

解决方案:

1. 交换任意两相电机线 (如 UV), 重新测试
2. 调整 step_delay (50ms $\rightarrow$ 100ms 或 20ms)
3. 增大 pwm_val (200 $\rightarrow$ 400, 即 40% 占空比)

Q5: 程序编译报错 “ledcAttachChannel was not declared”

可能原因:

- ESP32 Arduino Core 版本为 2.x, 不支持 3.0+ 的 API

解决方案:

- 代码已通过兼容性宏自动适配, 若仍报错:
1. 升级 Arduino IDE 的 ESP32 支持包到 3.0+
 2. 或手动修改宏定义, 使用 2.x 的 ledcSetup + ledcAttachPin

技术细节

MA600A SPI 通信协议

参数	值	说明
时钟极性 (CPOL)	1	空闲时时钟为高电平
时钟相位 (CPHA)	1	第二个边沿采样
数据位序	MSB First	高位在前
最大时钟	8MHz	实测稳定频率
帧格式	16bit	命令 + 数据

数据格式:

Bit 15	Bit 14	Bit 13~0
▼	▼	▼
PARC	ODD	角度值 (14bit)
(校验)	(奇偶)	0~16383 = 0°~360°

MP6543 电流采样公式

$V_{out} = V_{ref} + (I_{phase} \times R_{sense} \times Gain)$
其中:
$V_{ref} = 1.65V \text{ (运放偏置, } VCC/2)$
$R_{sense} = \text{MP6543 内部检测电阻 (典型 } 0.15\Omega)$
$Gain = \text{外部运放增益 (如 } 10V/V)$
电流计算:
$I_{phase} = (V_{out} - V_{ref}) / (R_{sense} \times Gain)$

六步换相原理

Step 0: A+ B- (U 相上桥导通, V 相下桥导通)
Step 1: A+ C-
Step 2: B+ C-
Step 3: B+ A-
Step 4: C+ A-
Step 5: C+ B-

每完成 6 步, 转子旋转一个电角度周期 ($60^\circ \times \text{极对数} = \text{机械角度}$)。

ESP32 LEDC 通道分配

通道	引脚	功能	频率	分辨率
0	GPIO15	A 相 PWM	20kHz	10bit
1	GPIO16	B 相 PWM	20kHz	10bit
2	GPIO17	C 相 PWM	20kHz	10bit

参考资料

- [MP6543 数据手册](#)
- [MA600A 数据手册](#)
- [ESP32 LEDC API 文档](#)
- [SimpleFOC 官方文档](#)

版本历史

版本	日期	说明
1.0	2026-07-20	初始版本, 支持 Core 2.x/3.0+ 兼容性

免责声明

本程序仅用于硬件诊断和学习目的。开环转动测试可能导致电机高速旋转, 请确保:

- 电机轴无机械负载
- 电源电流容量充足
- 操作人员远离旋转部件
- 紧急情况下可立即切断电源

作者不对因使用本程序造成的任何设备损坏或人身伤害负责。