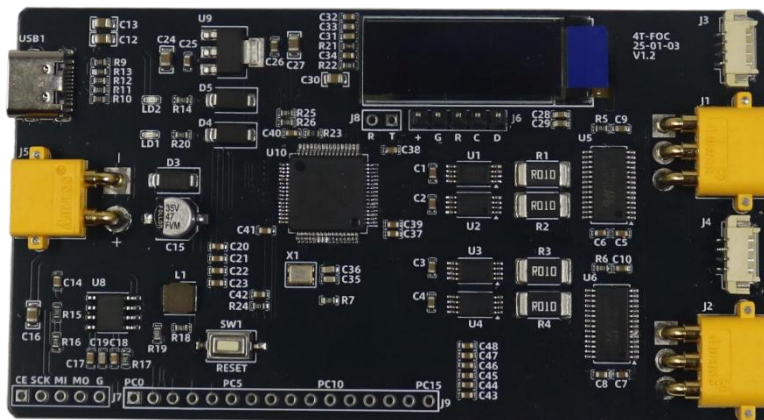


无刷电机控制驱动板

用户手册



四梯科技有限公司

目录

一、 总体概述3

 1.1 产品特点 3

 1.2 资源配置 3

 1.3 开发环境 4

 1.4 订购信息 4

 1.5 获取支持 4

 1.6 版本信息 4

二、 硬件规格详情 5

 2.1 供电方式 5

 2.2 主控单元 5

 2.3 时钟与复位系统 6

 2.4 通信接口 6

 2.5 人机交互接口 6

 2.6 传感器与执行器 6

 2.7 物理特性 6

三、 接线详情6

四、 测试程序简介 8

五、 硬件资源布局 10

4T_BLDC_G4Q1

一、总体概述

本 4T_BLDC_G4Q1 电机控制开发板由四梯科技有限公司设计推出，是一款面向工业应用与高端项目开发的高性能无刷直流电机驱动平台。

板卡以 ARM Cortex-M4 内核的高性能 STM32G431 微控制器为核心，配备两路独立的 MS8313 栅极驱动器，可精准驱动大功率 MOSFET，轻松驾驭单路或双路 BLDC 电机的复杂控制任务。为满足精准闭环控制与先进算法实现的需求，可通过 I2C 接口读取 AS5600 磁编码器用于实时转子位置反馈，并采用专业级 INA240 电流采样芯片，为实现精准的磁场定向控制提供了可靠的硬件保障。

在此基础之上，该开发板不仅是一个功能完备的电机驱动核心，更是一个高效的工程原型验证平台。其稳健的电源与驱动设计、丰富的扩展接口，使其能够无缝对接各类传感器与上位机系统，帮助开发者快速完成从算法仿真、原型验证到系统集成的高级阶段。

无论是用于高性能无人机推进系统、精密工业传动设备，还是对动态响应有苛刻要求的机器人关节驱动，本开发板都能为您提供强大的硬件支持与灵活的二次开发空间，显著缩短项目周期，助您将创新构想加速转化为成熟的工程现实。

1.1 产品特点

- 核心架构：板载高性能 STM32G431 微控制器，为复杂的电机控制算法（如 FOC、SVPWM）提供充足的算力支撑。
- 双路驱动设计：集成两路独立的 MS8313 栅极驱动器，可同时驱动两个 BLDC 电机，支持双电机协同或独立控制的高级应用场景。
- 高精度反馈系统：板载 AS5600 磁编码器，提供精准的转子位置信息；配合基于 INA240 的专业级电流采样电路，为闭环控制构建了完整的实时传感数据链。
- 工业级接口与易用性：提供 I2C 等标准通信接口，便于与外部传感器或上位机扩展连接。设计兼顾稳健性与便捷性，助力开发者快速进行原型验证与集成。

1.2 资源配置

- 主控 MCU：STM32G431 (ARM Cortex-M4)
- 电机驱动：两路 MS8313 栅极驱动器
- 位置传感器：AS5600 磁编码器
- 电流检测：INA240 高侧电流采样放大器
- 通信接口：I2C、USART、USB 等

4T_BLDC_G4Q1

- 6) 调试接口：SWD/JTAG
- 7) 电源管理：宽电压输入 DC-DC 与稳压电路

1.3 开发环境

- 1) 集成开发环境 (IDE)：STM32CubeIDE / Keil MDK
- 2) 关键开发工具：STM32CubeMX (用于引脚配置、时钟树初始化及代码生成)
- 3) 电机控制库：可结合 ST 官方电机控制软件开发套件进行高阶开发

1.4 订购信息

- 1).官方淘宝：<https://gxct.taobao.com/>
- 2).四梯商城：<https://4t.wiki/mall>

1.5 获取支持

请通过以下方式联系我们，获取更多硬件学习资源和技术支持。

- 1).技术支持：tech@4t.wiki
- 2).交流社区：<https://www.4t.wiki/community>
- 3).学习资源：<https://www.4t.wiki/curriculum>
- 4).Github 仓库地址：https://github.com/4T-tech/4T_BLDC_G4Q1
- 5).Gitee 仓库地址：https://gitee.com/fourT-tech/4T_BLDC_G4Q1

打开 4t.wiki 网站，获取更多资讯。

1.6 版本信息

版本编号	日期	修改内容	页码
V1.0	2025-10	新修订	
V1.1	2025-12	新增仓库地址，修改系列名称，补充图片	

4T_BLDC_G4Q1

二、硬件规格详情

4T_BLDC_G4Q1 包括两部分，分别为电机驱动控制板 4T_FOC_G4Q1 与磁编码器控制板 4T_AS5600。

其中 4T_FOC_G4Q1 中的 MCU 负责相电流检测，输出控制信号，通过 MS8313 驱动无刷电机的三相供电，还配备有 OLED 屏幕可实时显示当前的运行状态，以及 DEBUG 串口的引出方便调试，设备使用 SWD 方式进行下载。双路驱动设计可以支持双电机协同或独立控制的高级应用场景。

而 4T_AS5600 搭配无刷电机 2804 使用，经由螺柱安装后，可通过 I2C 接口配合 4T_FOC_G4Q1 获取无刷电机的位置信息，也保留模拟输出接口，默认 DIR 上拉为 CCW。

系统框图如下所示：

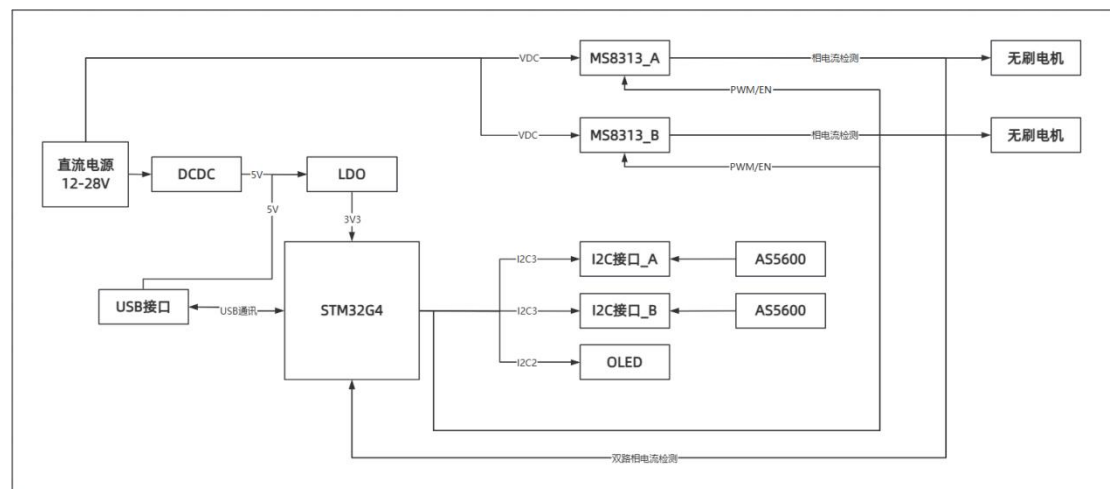


图 2.1 4T_BLDC_G4Q1 系统框图

2.1 供电方式

- 供电接口 1：Type-C USB 接口----USB1
输入电压：5V DC $\pm 5\%$
额定电流：500mA
- 供电接口 2：DC 直流接口，XT30PW-M30.G.Y
输入电压：3.5V-28V DC $\pm 5\%$
额定电流：3000mA

2.2 主控单元

- 型号：STM32G431
- 架构：ARM Cortex-M4F，32 位，带 FPU 和 DSP 指令
- 存储器：
Flash ROM:128 KB

4T_BLDC_G4Q1

RAM:32KB

- 工作电压：3.3V

2.3 时钟与复位系统

- 时钟源：配备独立 24MHZ 晶振
- 复位电路：配备复位电路与按键

2.4 通信接口

- 串口：RX 与 TX 进行 Debug
- I2C：实现与 AS5600 等设备的 I2C 通讯
- SPI：可用以驱动 SPI 屏幕等
- SWD：下载调试接口

2.5 人机交互接口

- 显示：0.96 寸 OLED

2.6 传感器与执行器

- 传感器：
INA240A2 电流检测*4
AS5600 I2C 接口*2
- 执行器接口：无刷电机接口 MR30PW-M30.G.Y*2

2.7 物理特性

- PCB 尺寸：99mm*54mm*1.6mm
- 产品尺寸：112mm*54mm*15mm

三、接线详情

STM32G431			
PIN	名称	接线	备注
1	VBAT	VDDA	3V3 供电
2	PC13	扩展接口	
3	PC14	扩展接口	
4	PC15	扩展接口	
5	PF0	外部晶振+	
6	PF1	外部晶振-	
7	PG10	NRESET	复位引脚
8	PC0	扩展接口	
9	PC1	扩展接口	

4T_BLDC_G4Q1

10	PC2	扩展接口	
11	PC3	扩展接口	
12	PA0	M1_CSA	M1 电流检测输出 A
13	PA1	M1_CSB	M1 电流检测输出 B
14	PA2	UART_TX	Debug 串口
15	VSS	GND	
16	VDD	VDDA	
17	PA3	UART_RX	Debug 串口
18	PA4	M1_EN	M1 使能
19	PA5	M1_A	M1_PWM 输出 A
20	PA6	M1_B	M1_PWM 输出 B
21	PA7	M1_C	M1_PWM 输出 C
22	PC4	扩展接口	
23	PC5	扩展接口	
24	PB0	M2_CSA	M2 电流检测输出 A
25	PB1	M2_CSB	M2 电流检测输出 B
26	PB2	NC	
27	VSSA	GND	
28	VREF+	VDD	
29	VDDA	VDD	
30	PB10	NC	
31	VSS	GND	
32	VDD	VDD	
33	PB11	NC	
34	PB12	SPI2_NSS/扩展接口	
35	PB13	SPI2_SCK/扩展接口	
36	PB14	SPI2_MISO/扩展接口	
37	PB15	SPI2_MOSI/扩展接口	
38	PC6	扩展接口	
39	PC7	扩展接口	
40	PC8	I2C3_SCL/扩展接口	
41	PC9	I2C3_SDA/扩展接口	
42	PA8	I2C2_SDA	

4T_BLDC_G4Q1

43	PA9	I2C2_SCL	
44	PA10	NC	
45	PA11	USB_DM	
46	PA12	USB_DP	
47	VSS	GND	
48	VDD	VDD	
49	PA13	SWDIO	
50	PA14	SWCLK	
51	PA15	M2_A	M2_PWM 输出 A
52	PC10	扩展接口	
53	PC11	扩展接口	
54	PC12	扩展接口	
55	PD2	NC	
56	PB3	NC	
57	PB4	M2_EN	M2 使能
58	PB5	M2_B	M2_PWM 输出 B
59	PB6	M2_C	M2_PWM 输出 C
60	PB7	NC	
61	PB8	下拉	
62	PB9	NC	
63	VSS	GND	
64	VDD	VDD	

四、测试程序简介

测试程序中使用 LL 库实现在 STM32G431 上对无刷电机的驱动。默认使用电压调节以及角度控制，实现电机的角度定位与保持。

在主函数中，首先实现各类基础外设的初始化工作，包括 GPIO，DMA，ADC，定时器的初始化等。

```
MX_GPIO_Init();
MX_DMA_Init();
MX_ADC1_Init();
MX_ADC2_Init();
MX_TIM2_Init();
MX_TIM3_Init();
```

4T_BLDC_G4Q1

```
MX_I2C3_Init();
MX_USART2_UART_Init();
MX_I2C2_Init();
```

其次则是 FOC 控制相关的初始化内容，包括 PWM 的启动，PID 各种参数的初始化，以及实际应用情况的初始化。

```
HAL_TIM_PWM_Start(&htim2,TIM_CHANNEL_1);
HAL_TIM_PWM_Start(&htim3,TIM_CHANNEL_1);
HAL_TIM_PWM_Start(&htim3,TIM_CHANNEL_2);
HAL_Delay(1000);
MagneticSensor_Init();

LPF_init();           //LPF 参数初始化
PID_init();           //PID 参数初始化
.....
```

参数设置完成后，会继续初始化电机，还会通过位置传感器来对电机进行位置和方向的定位。这一部分会观察到电机会有小幅度的慢速摆动，电流会有小幅度的上升。

```
Motor_init();
Motor_initFOC(0,UNKNOWN);
```

初始化完成后则是在主循环中持续进行 foc 的控制。此时观察到的电机是静止的。当扭动电机时，会产生反向的力保持电机位置不变，同时电流会持续增大直到返回原位。

```
move(target);
loopFOC();
```

五、硬件资源布局

