**M线功率板软件设计说明**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **文件信息** | | | | | | | |
| **项目**  **阶段** | □全生命周期 | □概念  阶段 | ■计划  阶段 | □开发  阶段 | □验证  阶段 | □发布阶段 | □生命周期阶段 |
| **版本号** |  | | | **密级** | □内部公开 ■秘密 □机密 □绝密 | | |
| **主责**  **部门** |  | | | **保存**  **部门** |  | | |
| **拟制** |  | | | **拟制**  **日期** |  | | |
| **审核人** |  | | | **审核**  **日期** |  | | |
| **批准人** |  | | | **批准**  **日期** |  | | |



上海高仙自动化科技发展有限公司

版权所有，侵权必究 All rights reserved

**修订记录**

|  |  |  |  |
| --- | --- | --- | --- |
| 版本号 | 修订日期 | 修订内容 | 修订人 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

*文档版本号由“Vx.y[z]”组成，其中：*

*a） x为主版本号，取值范围为“0～9”。文件进行重大修订时主版本号递增1；*

*b） y为次版本号，取值范围为“0～9”。文件每修改一次时次版本号递增1；主版本号发生改变时，次版本号重新置0；*

*c） z为可选择的当日版本号，在同一天材料的不断修改则需在版本号后面标上英文字母，取值范围为“A～Z”*

目录

[1 引言 5](#_Toc116112415)

[1.1 目的与背景 5](#_Toc116112416)

[1.2 术语（名词解释） 5](#_Toc116112417)

[1.3 预期读者 5](#_Toc116112418)

[1.4 参考资料 5](#_Toc116112419)

[2 设计概述 6](#_Toc116112420)

[2.1 任务和目标 6](#_Toc116112421)

[2.2 运行环境 6](#_Toc116112422)

[2.3 硬件环境 6](#_Toc116112423)

[2.4 软件环境 6](#_Toc116112424)

[2.5 需求概述 7](#_Toc116112425)

[3 系统详细设计 10](#_Toc116112426)

[3.1 M线功率板软件设计说明 10](#_Toc116112427)

[3.1.1 设计思想 10](#_Toc116112428)

[3.1.2 设计原则 10](#_Toc116112429)

[3.1.3 详细设计工具 10](#_Toc116112430)

[3.1.4 总体架构设计 10](#_Toc116112431)

[3.2 M线功率板软件设计 13](#_Toc116112432)

[**3.2.1 混合型驱动板原理说明** 13](#_Toc116112433)

[3.2.2 混合型驱动板接口说明 14](#_Toc116112434)

[3.3 驱动器软件功能设计 15](#_Toc116112435)

[3.3.1 驱动器软件调度系统 15](#_Toc116112436)

[3.3.2 驱动器系统接口与通信功能 15](#_Toc116112437)

[4 系统性能设计 24](#_Toc116112438)

[4.1 速度环控制实现 24](#_Toc116112439)

[4.2 位置控制实现 24](#_Toc116112440)

[5 驱动器系统监控、保护和报警功能 25](#_Toc116112441)

# 1 引言

## 1.1 目的与背景

根据M线需求，进行灵活定制的一款可以驱动14路电机、总功率可达700W的混合型驱动器和一款可配置远程唤醒模块，可以对其他外设进行上下电的低功耗MCU。本文主要对该混合型驱动器低功耗MCU进行全面的软件详细设计说明。M线功率板软件设计说明以模块化设计为基础，以实现一控多驱为目标，能够灵活结合公司目前的实际情况，适应公司现有产品或将来新产品。为后续系统详细设计和开发提供设计指导。

|  |  |
| --- | --- |
| 缩写、术语 | 解 释 |
| BLDC | Brushless Direct Current Motor (无刷直流电机) |
| PWM | Pulse Width Modulation (脉冲宽度调制) |
| PI | proportion-integral比例-积分 |
| CAN | 控制器局域网络 |
| MCU | Microcontroller Unit 微控制单元 |
| CANopen | 通讯协议 |

## 1.3 预期读者

系统设计人员、硬件开发人员、线束设计人员、硬件测试人员、软件开发人员。

## 1.4 参考资料

《驱动器报警指示灯说明》

《CANopen协议》

# 2 设计概述

## 2.1 任务和目标

本设计是为了解决公司现有机型由于电机种类多、成本高、控制系统冗余的问题。为设计一个适应公司的产品的专用驱动器，包含２路无刷电机、3路H桥驱动双向电机、４路单向电磁阀、２路单向电机和3路外驱驱动电机（1风机+2边刷）总共１４路电机接口，适应公司总功率５００W以下的电机驱动控制。

## 2.2 运行环境

## 2.3 硬件环境

1.  工作温度：-20℃ ~ 60℃（长期工作温度），40℃ ~ 65℃（短期工作温度）；

2.  工作湿度：10%-95%RH 无冷凝；

3.  存储温度：-20℃ ~ 65℃；

4.  存储湿度：10%-95%RH 无冷凝；

5.  海拔高度：当海拔高度超过2000m时，无风扇机型最高工作环境温度依每10℃/1000m比例下降，有风扇机型最高工作环境温度依每5℃/1000m比例下降。

## 2.4 软件环境

基于ARM CORTEX-M4内核的嵌入式开发环境

1)        运行系统：Windows10/8/7

2)        软件开发工具：Keil5

3)        软件调试烧录工具：J-Link/ST-Link

## 2.5 需求概述

1.  正常运行电机驱动信号驱动十四路电机；

2.  具备电机刹车制动保护功能；

3.  电机霍尔传感器信号接口；

4.  保护功能：系统过压欠压保护，电机过流报警保护等

5.  电机电流采样反馈；

6.  CAN接口；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | *单向有刷1* | 满载工作电压范围(*V*) | *20~30* |
| 输入额定电压(*V*) | *24* | 输入电压范围(*V*) | *18~32* |
| 输出额定功率(*W*) | *120* | 输出峰值功率(*W*) | *150* |
| 载波频率(*kHz*) | *10* | 输出频率(*Hz*) | *1~10000* |
| 反馈信号类型 | *无* | 温度传感器类型 | *无* |
| *HOC*(*A*) | □ | *SOC*(*A*) | ☑相电流4.5A |
| *OV*(*V*) | *36* | *UV*(*V*) | *16* |
| *OVR*(*V*) | *30* | *UVR*(*V*) | *20* |
| *OT*(*℃*) | □ | *OTR*(*℃*) | □ |
| *SC* | ☑ | *DP* | □ |
| *HF* | □ | 其它 | *定载频* |
| 备注 |  | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 类型 | | *单向有刷2* | 满载工作电压范围(*V*) | *20~30* | | |
| 输入额定电压(*V*) | | *24* | 输入电压范围(*V*) | *18~32* | | |
| 输出额定功率(*W*) | | *150* | 输出峰值功率(*W*) | *200* | | |
| 载波频率(*kHz*) | | *10* | 输出频率(*Hz*) | *1~10000* | | |
| 反馈信号类型 | | *无* | 温度传感器类型 | *无* | | |
| *HOC*(*A*) | | □ | *SOC*(*A*) | ☑相电流2A | | |
| *OV*(*V*) | | *36* | *UV*(*V*) | *16* | | |
| *OVR*(*V*) | | *30* | *UVR*(*V*) | *20* | | |
| *OT*(*℃*) | | □ | *OTR*(*℃*) | □ | | |
| *SC* | | ☑ | *DP* | □ | | |
| *HF* | | □ | 其它 | *变载频* | | |
| 备注 | |  | | | | |
| 类型 | | *H桥有刷* | 满载工作电压范围(*V*) | *20~30* | |
| 输入额定电压(*V*) | | *24* | 输入电压范围(*V*) | *18~32* | |
| 输出额定功率(*W*) | | *50* | 输出峰值功率(*W*) | *100* | |
| 载波频率(*kHz*) | | *10* | 输出频率(*Hz*) | *1~10000* | |
| 反馈信号类型 | | *HALL* | 温度传感器类型 | *无* | |
| *HOC*(*A*) | | □ | *SOC*(*A*) | ☑相电流5A | |
| *OV*(*V*) | | *36* | *UV*(*V*) | *16* | |
| *OVR*(*V*) | | *30* | *UVR*(*V*) | *20* | |
| *OT*(*℃*) | | □ | *OTR*(*℃*) | □ | |
| *SC* | | ☑ | *DP* | □ | |
| *HF* | | □选配 | 其它 |  | |
| 备注 | |  | | | |
| 类型 | | *无刷* | 满载工作电压范围(*V*) | *20~30* |
| 输入额定电压(*V*) | | *24* | 输入电压范围(*V*) | *18~32* |
| 输出额定功率(*W*) | | *400* | 输出峰值功率(*W*) | *150* |
| 载波频率(*kHz*) | | *10* | 输出频率(*Hz*) | *1~10000* |
| 反馈信号类型 | | *HALL* | 温度传感器类型 | *无* |
| *HOC*(*A*) | | □ | *SOC*(*A*) | ☑相电流15A |
| *OV*(*V*) | | *36* | *UV*(*V*) | *16* |
| *OVR*(*V*) | | *30* | *UVR*(*V*) | *20* |
| *OT*(*℃*) | | □ | *OTR*(*℃*) | □ |
| *SC* | | ☑ | *DP* | ☑ |
| *HF* | | ☑ | 其它 |  |
| 备注 | |  | | |
| 驱动控制器通讯方式\工作方式 | | | | |
| CAN | ☑ | | | |
| RS485 | □ | | | |
| RS232 | □ | | | |
| SPI | □ | | | |
| 备注 |  | | | |
| 工作环境温度 | -20℃~65℃ | | | |
| 存储环境温度 | -20℃~65℃ | | | |
| 工作环境湿度 | 20%~90% | | | |
| 污染等级 |  | | | |
| 散热方式 | 自然散热 | | | |
| 海拔高度 |  | | | |
| 备注 |  | | | |

# 3 系统详细设计

## 3.1 M线功率板软件设计说明

本文所述的混合型驱动板驱动系统核心处理器采用基于CORTEX-M4 ARM内核的GD32F407作为主控MCU，实现电机驱动、电机霍尔传感器信号接收处理、相电流报警、标准CANOPEN通信与串口调试功能。

### 3.1.1 设计思想

各程序模块采用功能模块的思想设计，进行分层任务运行，总体软件包含四个主要环路：PID环路，安全环路，电机控制环路，以及UI通信环路。

### 3.1.2 设计原则

在软件开发中，为了提高软件系统的可维护性和可复用性，增加软件的可扩展性和灵活性，开闭原则、依赖倒置原则、单一职责原则、接口隔离原则等原则来开发程序，从而提高软件开发效率、节约软件开发成本和维护成本。

### 3.1.3 详细设计工具

|  |  |
| --- | --- |
| **资源** | **描述** |
| 操作系统 | Windows10/8/7 |
| 数据库 | Keil：hal库、标准库 |
| 测试工具 | 示波器、电流探头、网络与串口调试工具 |
| 监控工具 | 网络与串口调试工具 |

### 3.1.4 总体架构设计

混合型驱动的总体设计主要由单个MCU控制,驱动多达十四路电机，保证各个电机独立稳定运行，适应公司特殊需求。为公司混合型驱动提供一种性价比较高的驱动方案。混合型驱动的总体框图如图1所示 ：



图1 M线功率板总体框图

四路双向推杆有刷电机采用H桥驱动，两路单向水泵和四路开通阀有刷电机采用半桥驱动，两路滚刷无刷电机采用三相全桥驱动。具体见表3-1：

表3-1 电机规格表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **电机类型及功能** | **电机反馈信号** | **电机数量** | **电机额定功率** | **驱动控制器类型** |
| 无刷/边刷 | 霍尔 | 2 | / | 485通信/PWM信号 |
| 无刷/吸风 | 霍尔 | 1 | 400W | 485通信/PWM信号 |
| 有刷/推杆 | 霍尔 | 3 | 50W | H桥 |
| 有刷/水泵 | 无 | 2 | 150W | 单驱 |
| 有刷/开通阀 | 无 | 4 | 120W | 单驱 |
| 无刷/滚刷 | 霍尔 | 2 | 400W/160W | 三相全桥 |

详细的硬件系统框图如图2所示：

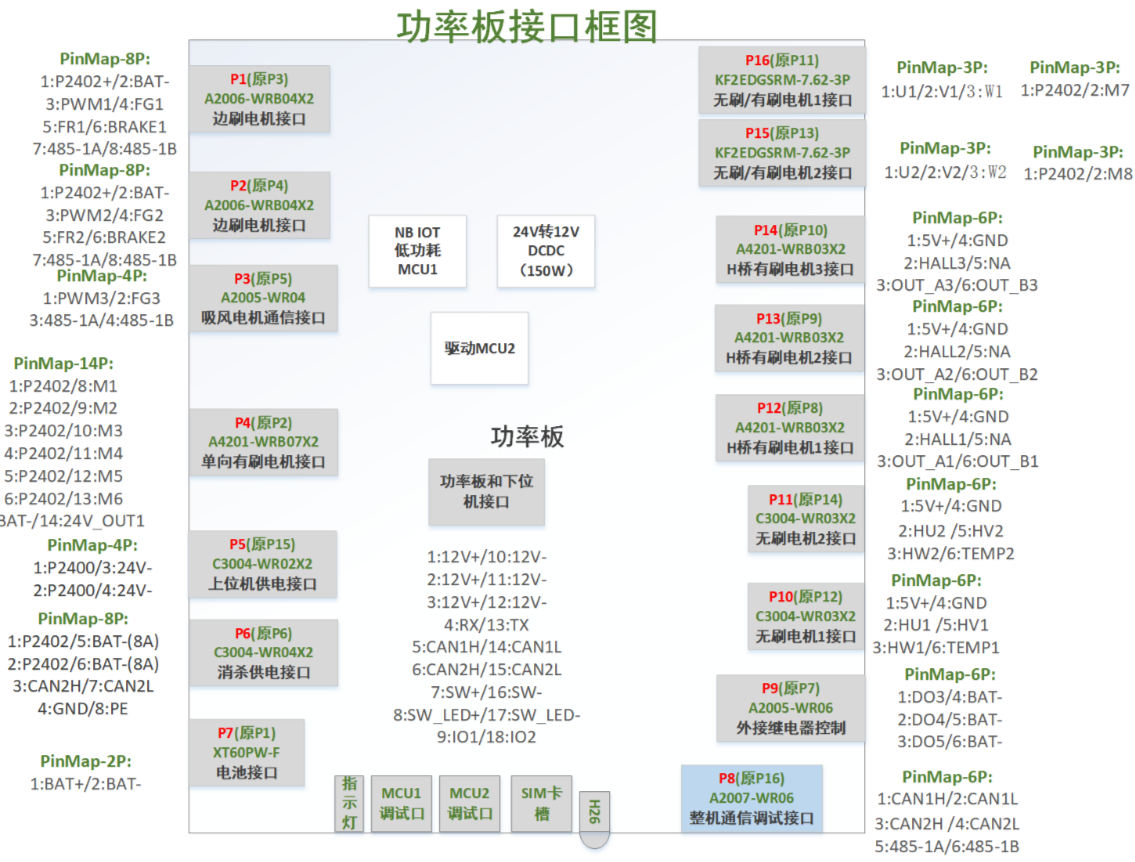


图2 硬件系统框图

 该框图按功能划分主要包括：电源、电机驱动、主功率电路、通讯、接口等。DCDC电源包括两路BUCK电源和一路LDO电源。两路BUCK电源分别为24V转15V，15V转5V，其中15V主要给驱动IC供电以及作为5VBUCK电源输入，5V主要给霍尔供电以及作为3.3VLDO电源输入。LDO电源主要给MCU和电流检测运放供电。

电机驱动部分包括三路有刷推杆电机驱动、两路有刷水泵电机驱动（一路可接电磁泵）、四路电磁开通阀、两路无刷滚刷电机驱动以及二路边刷和一路风机控制信号。

采样部分主要包括驱动输出电流采样、母线电压采样、电机温度采样等。

支持CAN通讯，用来与下位机通讯。

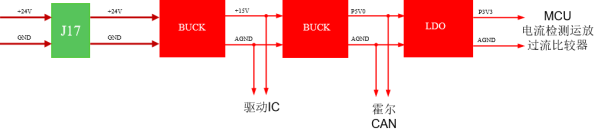
接口部分包括24V电源输入、CAN通讯、两路无刷电机驱动输出、两路无刷电机霍尔反馈信号、四路H桥有刷电机输出、6路单相有刷电机输出、外部吸风电机与边刷电机接口、烧录与调试接口等。

## 3.2 M线功率板软件设计

3.2.1 M线功率板原理说明

本混合型驱动板设计主要实现整机大部分电机驱动控制，电机霍尔传感器信号接收与处理、相电流报警信号处理、CAN标准CANopen从站和串口调试功能，具体硬件原理框图如图3所示。

图3 硬件原理框图

图4 电源模块

M线功率板输入电源为来自电源管理板的24V直流电源，输入至系统后级DC-DC Buck电路。24V分别经后级DC-DC电路与LDO电路分别生成15V，5V与3.3V电平。

15V直接供至电机驱动IC FD2103S作为电机的驱动母线电压，UVW三相相电流经过采样电阻进行采样，反馈接至STM32F407VET6。STM32F407VET6为意 法半导体出品的一款基于ARM Cortex-M4内核的32位CPU，主频高达168MHz，512KB FLASH、192+4KB SRAM，3个12位2.4 MSPS A/D转换器，多达24个通道，2个12位D/A转换器，多达3个I2C接口、6个UART接口、3个SPI接口、2个CAN接口、片上带USB2.0 PHY控制器，带10/100M以太网MAC，支持IEEE 1588V2，MII/RMII。

### 3.2.2 M线功率板接口说明

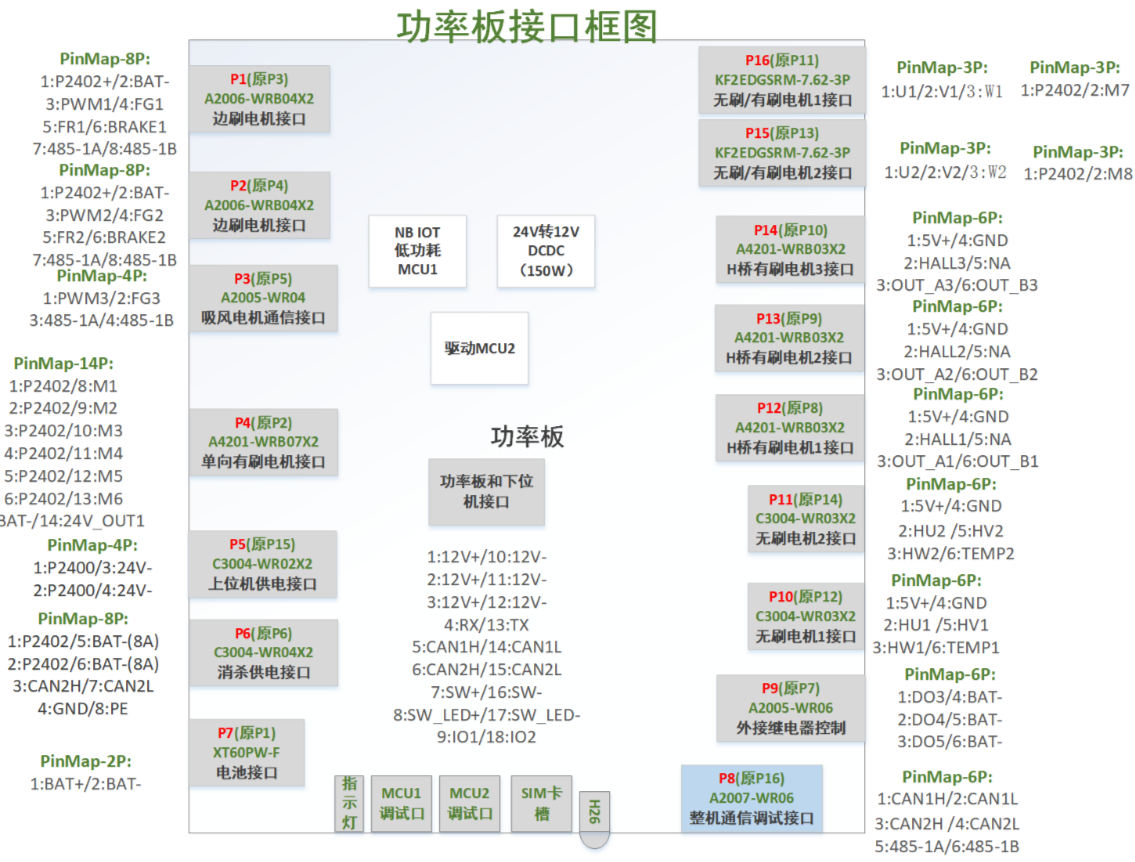


图5混合驱动板接口图

## 3.3 驱动器软件功能设计

### 3.3.1 驱动器软件调度系统

按照驱动器软件的基本任务，如图4给出了驱动器运行整体软件框图，对于整体程序流程上，电机库控制过程都发生在中断中，区别于普通顺序控制流程，也无任务调度；这样做可以将电机控制做到实时控制，利于驱动实时控制。



图6 驱动器软件框架图

### 3.3.2 驱动器系统接口与通信功能

#### 3.3.2.1 系统初始化及main函数

在 main.c 中主要是系统的初始化函数，包含检测时钟配置、相关应用引脚配置、相关定时器功能配置、ADC初始化配置、串口初始化配置、中断优先级配置等。



图7 系统初始化配置

#### 3.3.2.2 电机控制执行时序



图8 电机控制执行时序

驱动器作为整机系统当中的驱动部分运行CANopen从站，并完全遵循CANopen协议，完全受CAN网络控制，接收相应指令执行对应动作，并将当前的状态通过CAN反馈。

电机收到运行指令之后，先判断是否存在未清除错误，保证在无故障的情况下启动电机，电机运行状态机在嘀嗒中断当中2ms执行一次。

无刷电机经过速度规划输出参考速度，速度环通过计算，将实际速度迅速靠近参考速度，最终参考速度等于设置速度，实际速度跟随参考速度达到速度设定值。

推杆电机经过对比当前位置与实际位置的偏差调整自身占空比，通过调整PWM，达到实际位置。

单向电机通过对比所下发的PWM值，规划处比较缓和的PWM上升曲线，最终达到实际下发的PWM值。

ADC在驱动器上电时持续采样，并通过DMA搬运数据，TIM1更新中断100us监控电机当前电流，执行周期为100us，出错之后就进入相应的出错管理。

#### 3.3.2.3 中断

M线功率板板由于电机控制种类多，基本用完GD32F407VET6所有引脚和功能，中断种类多，时序控制复杂，为保证驱动器功能正常执行，各个中断执行时间和中断时间必须严格控制，以下为驱动器各种中断及其实现功能。

表3-4 中断说明表

|  |  |
| --- | --- |
| **中断类型** | **实现功能** |
| TIM1更新中断 | 用于电流保护 |
| Sysstick中断 | Systick整个系统的电机控制都在嘀嗒中断当中运行，包括速度环、位置环、上电标定、霍尔学习、过压欠压保护、看门狗喂狗 |
| 外部中断0\_1\_2 | 推杆电机霍尔信号 |
| 外部中断3 | 风机hall信号 |
| 外部中断4 | 无刷电机2刹车保护 |
| 外部中断5\_6\_7 | 无刷电机1霍尔中断 |
| 外部中断9 | 单向电机10 11 12 13刹车 |
| 外部中断10 | 推杆0 1 2刹车 |
| 外部中断11 | 单向电机3 4刹车 |
| 外部中断13\_14\_15 | 无刷电机2霍尔中断 |
| TIM1BRAKE中断 | 无刷电机1刹车保护 |
| TIM7中断 | 单向电机J13J14速度计算 |
| TIM9中断 | 单向电机测速捕获中断 |
| CAN接收中断 | 下位机控制 |
| 串口接收中断 | 开发调试 |
| DMA中断 | 搬运ADC电流采集值 |
| TIM6中断 | 为CANopen协议栈提供时基 |

#### 3.3.2.4 通信接口及协议

1、串口通信

根据通讯使用的电平标准不同，串口通信可分为TTL标准及RS-232标准，见如下表格18：

表3-5 常见串口通信电平

|  |  |
| --- | --- |
| 通讯标准 | 电平标准（发送端） |
| 5V TTL | 逻辑1:：2.4V ~ 5V —— 逻辑0： 0 ~ 0.5V |
| RS-232 | 逻辑1:：-15V ~ -3V —— 逻辑0： +3V ~ +15V |
| RS-485 | 逻辑1：+2~+6——逻辑0：-2~-6V |

我们常见的电子电路中常使用TTL的电平标准，如我们的计算机内部识别的就是TTL电平，理想状态下，使用5V表示二进制逻辑1，使用0V表示逻辑0，所以说计算机CPU也只能识别二进制的0和1；但TTL电平有个缺点就是它不适合远距离传输，所以为了增加串口通讯的远距离传输及抗干扰能力，我们通常会使用RS-232或者RS-485电平。

TTL串口通信主要用于调试程序，监视电机运行的数据。

2、通信协议

CANopen是一种架构在控制局域网路（Controller Area Network, [CAN](https://baike.baidu.com/item/CAN/4698193)）上的高层通讯协定，包括通讯子协定及设备子协定常在[嵌入式系统](https://baike.baidu.com/item/%E5%B5%8C%E5%85%A5%E5%BC%8F%E7%B3%BB%E7%BB%9F)中使用，也是工业控制常用到的一种[现场总线](https://baike.baidu.com/item/%E7%8E%B0%E5%9C%BA%E6%80%BB%E7%BA%BF)。驱动器在 CANopen 总线网络（支持“CIA 301”和“CIA 402”）中作为从站使用，参数功能通过使用对象字典“制造商指定数据”区实现，所有的参数、参数值和功能都是通过 Index 和 Sub-Index 组成的地址来访问和存取。CANopen 通讯模型定义了如下几种报文（通讯对象）：

表3-6 CANopen 通讯对象

|  |  |  |
| --- | --- | --- |
| SDO | Service Data Object | 用于非时间关键数据，如驱动器参数的设置 |
| PDO | Porcess Data Objec | 快速过程数据交互（如：实际位置） |
| EMCY | Emergency Message | 故障信息传输 |
| SYNC | Synchronization Message | 同步信息：多个 CAN 节点的同步 |
| NMT | Network Management | 网络服务：例如，可以同时激活所有的 CAN 节点 |

NMT 管理报文NMT（Network Management）管理报文用来实现 CANopen 网络中主节点对从节 点监控和管理，如初始化、启动和停止节点，侦测失效节点。这种服务主要采用主从通 讯模式来实现，此消息不需要应答。只有主节点能够发送 NMT Modual Control 报文。

表3-7 NMT 消息格式

|  |  |  |  |
| --- | --- | --- | --- |
| COB-ID | DLC | Byte0 | Byte1 |
| 0x000 | 2 | 命令字 | Node-ID |

COB-ID 为 0x000，Byte0 为命令字，占一个字节，Byte 1 为 CANopen 网络设备节点地址，0 表示广播。

例：发送：000h 01h 06h // 将节点 6 置为可操作状态；

发送：000h 01h 00h // 将所有节点置为可操作状态；

NMT 节点保护（NMT Node Guarding）通过节点保护服务，MNT 主节点可以检查每个节点的当前状态，当这些节点没有数 据传送时这种服务尤其有意义。节点保护分为心跳报文（Heartbeat）和心跳监护（Life Guard）。

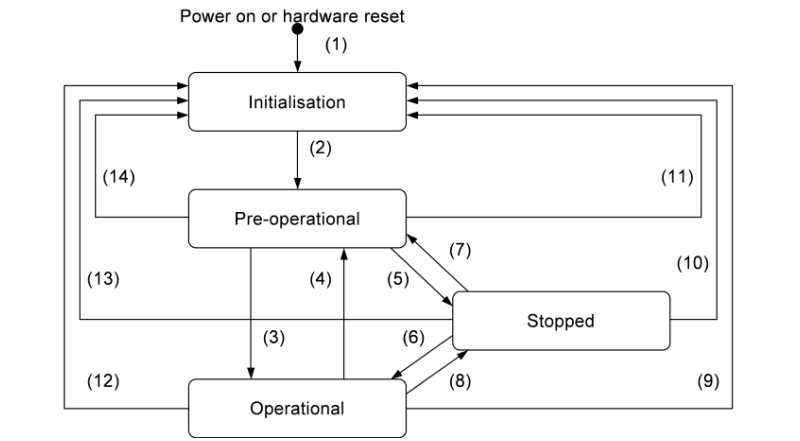


图9 NMT 状态机

心跳报文（Heartbeat） 驱动器上电后，会主动上报一包上线报文。通过设置心跳，可以以一个固定的频率 发送心跳报文，用于告诉主机自己的状态。报文的格式：COB-ID 为 0x700+ ID，数据 为一字节的状态数据。例如 ID 为 2 的驱动器在接入 CAN 总线后会主动发送报文：0x702 00。

表 3-7 心跳报文格式表

|  |  |  |  |
| --- | --- | --- | --- |
| COB-ID | DLC | Byte0 | 状态含义 |
| 0x700+Node\_ID | 1 | 0x00 | BOOTUP 启动状态 |
| 0x04 | STOPPED 停止 |
| 0x05 | OPERATIONAL 可操作 |
| 0x7F | PRE-OPERATIONAL 预操作 |

心跳间隔为字典 0x1017 设置值(单位 ms)，设置 0x1017 为 0，则禁止心跳报文， 设置 0x1017 为非 0，则使能心跳报文。例如：

|  |  |  |  |
| --- | --- | --- | --- |
|  | COB-ID | DATA | 注释 |
|  | 602 | 2B 17 10 00 00 00 00 00 | 禁止心跳报文 |
| 发送 | 602 | 2B 17 10 00 64 00 00 00 | 使能心跳报文，心跳间隔 100ms |

SDO 服务数据对象

SDO（Service Data Object）主要用来访问节点的对象字典，主站问，从站答。主 要用来在设备之间传输低优先级的对象，典型是对从设备进行配置、管理，比如模式配 置、PDO 配置等，类似于 Modbus 的一问一答体系。

表3-8 指令结构表

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COB\_ID | | Byte0 | Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | Byte 7 |
| CMD | INDEX | | SUB  INDEX | DATAs | | | |
| 问 | 0x600+ Node\_ID | 0x23 写 4byte | Fn 参数地址  Dn 参数地址  Pn 参数地址  对象字典 Index | | DATA1 | DATA2 | DATA3 | DATA4 |
| 0x2B 写 2byte | 0x00 | 0x00 |
| 0x2F 写 1byte | 0x00 |
| 0x40 读数据 | 0x00 |
| 答 | 0x580+ Node\_ID | 0x43 回 4byte | DATA1 | DATA2 | DATA3 | DATA4 |
| 0x4B 回 2byte |
| 0x4F 回 1byte | 0x00 | 0x00 |
| 0x60 写回复 | 0x00 |
| 0x80 错误码 | DATA2 | DATA3 | DATA4 |

PDO 过程数据对象

PDO（Process Data Object）是用来发送（TPDO）或者接收（RPDO）数据的， 实现实时数据的传输，只传不回。1 个 PDO 单次最多传输 8 个字节的数据。

表3-9 PDO 的对象字典参数表

|  |  |  |  |
| --- | --- | --- | --- |
| 映射参数 Index | | 通信参数 Index | |
| RPDO1~4 | TPDO1~4 | RPDO1~4 | TPDO1~4 |
| 1600h~1603h | 1A00h~1A03H | 1400h~1403h | 1800h~1803h |

驱动器运行标准的CANopen从站，通讯协议详见标准CANopen协议。

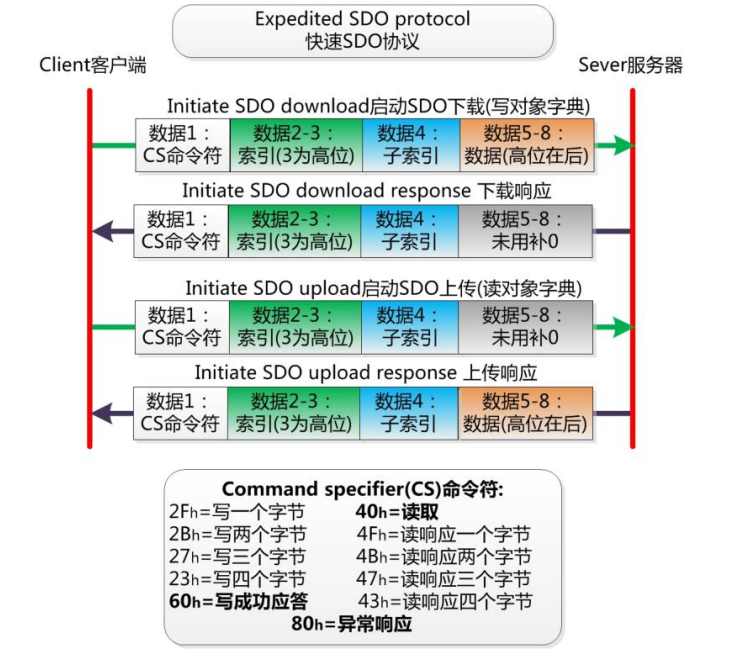


图10 SDO协议图

# 4 系统性能设计

## 4.1 速度环控制实现

速度环路控制用来实现驱动器速度控制，包括速度环调节、速度命令给定及反馈处理。速度控制总框图如下图11：

图11 速度环说明图

## 4.2 位置控制实现

位置环路控制用来实现驱动器对于推杆电机的位置控制，如下图12所示



图12 位置控制说明图

# 5 驱动器系统监控、保护和报警功能

驱动器工作系统监控保护报警功能包括：如下图所示：



图12 驱动器保护报警结构图

 驱动器工作系统监控保护报警功能具体错误类型与错误代码如下表所示：

表3-10 驱动器错误代码表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 驱动器报警指示灯说明 | | 错误类型 | 检查电机 | 对应电机 | 错误代码 |
| 上面绿灯 | 上电点亮 | 上电指示灯 |  | | |
| 下面绿灯 | 常亮+闪烁一次 | 代码运行指示灯 |
| 上面红灯 | 快速闪烁1次 | 推杆电机1过流 | 检查J1 | 0号电机 | 0X00000001 |
| 快速闪烁2次 | 推杆电机2过流 | 检查J2 | 1号电机 | 0X00000002 |
| 快速闪烁3次 | 推杆电机3过流 | 检查J3 | 2号电机 | 0X00000004 |
| 快速闪烁4次 | 推杆电机4过流 | 检查J4 | 9号电机 | 0X00000200 |
| 快速闪烁5次 | BLDC1过流 | 检查J6 | 5号电机 | 0X00000020 |
| 快速闪烁6次 | BLDC2过流 | 检查J7 | 6号电机 | 0X00000040 |
| 快速闪烁7次 | 单向电机1过流 | 检查J8\_1 | 3号电机 | 0X00000008 |
| 快速闪烁8次 | 单向电机2过流 | 检查J8\_2 | 4号电机 | 0X00000010 |
| 下面红灯 | 慢速闪烁 | 霍尔报错 | 检查J5 | 5号电机霍尔 | 0X00000400 |
| 上面红灯 | 慢速闪烁 | 霍尔报错 | 检查J7 | 6号电机霍尔 | 0X00008000 |
| 两个红灯 | 慢速同频率闪烁 | CAN掉线 | 检查J14 |  | 0X00100000 |
| 两个红灯 | 常亮 | 母线过压 | 检查母线电压 |  | 0X00000080 |
| 下面红灯 | 常亮 | 母线欠压 | 检查母线电压 |  | 0X00000100 |
| 下面红灯 | 快速闪烁一次 | BLDC1硬件过流 | 检查短路 | 5号电机 | 0X20000000 |
| 快速闪烁二次 | BLDC2硬件过流 | 检查短路 | 6号电机 | 0X40000000 |
| 快速闪烁三次 | 电机3\_4硬件过流 | 检查短路 | 电机3\_4 | 0X04000000 |
| 快速闪烁四次 | 推杆0\_1\_2\_9硬件过流 | 检查短路 | 推杆0\_1\_2\_9 | 0X08000000 |
| 快速闪烁五次 | 电机10\_11\_12\_13硬件过流 | 检查短路 | 电机10\_11\_12\_13 | 0X10000000 |

# 6 低功耗MCU系统详细设计

## 6.1 低功耗MCU总体架构

本文所述的混合型驱动板驱动系统核心处理器采用基于CORTEX-M4 ARM内核的GD32E230C8T6作为主控MCU，实现远程唤醒串口数据和钥匙IO输入数据读取，控制上位机，功率板驱动器，轮毂驱动器等外设上下电。总体架构如下图。



## 6.2 低功耗MCU接口说明

K0控制：控制上位机RK3588（24V供电）上下电的开关，受低功耗MCU控制；远程唤醒拷包时可以只打开该开关；

K1控制：为24V转12V电源模块使能管脚，低功耗MCU通过IO控制；如果使用的是现有金升阳外购DC模块则低功耗MCU输出3.5V~12V电源模块输出开启，输出0V~1.2V时电源模块输出关闭；如果使用自研DC电源模块，则根据电路设计相关控制电路；

K2控制：为功率板驱控相关外设供电控制开关，受低功耗MCU控制；

K3控制：为轮毂和吸风电机上下电控制开关，该控制开关使用外置继电器实现轮毂和吸风电机的上下电，不集成在功率板上；同时受低功耗MCU和功能安全板控制，两者或关系，逻辑关系通过硬件电路实现，同时兼容无功能安全板方案；

K4控制：为自动充电控制开关，该控制开关使用外置继电器实现自动充电口上下电，不集成在功率板

## 6.3 低功耗MCU软件架构

