

# EG8062 芯片数据手册

逆变发电机专用芯片

R

### 版本变更记录

| 版本号  | 日期               | 描述             |
|------|------------------|----------------|
| V1.0 | 2022 年 10 月 10 日 | EG8062 数据手册初稿。 |
|      |                  |                |
|      |                  |                |

# 目录

|                                 |    |
|---------------------------------|----|
| 目录 .....                        | 3  |
| 1. 特点 .....                     | 5  |
| 2. 描述 .....                     | 6  |
| 3. 应用领域 .....                   | 6  |
| 4. 引脚 .....                     | 7  |
| 4.1 引脚定义 .....                  | 7  |
| 4.2 引脚描述 .....                  | 8  |
| 5. 结构框图 .....                   | 10 |
| 6. 典型应用电路 .....                 | 11 |
| 6.1 1KW/230V 逆变发电机典型应用图 .....   | 11 |
| 6.2 1KW/120V 逆变发电机典型应用图 .....   | 12 |
| 6.3 2KW/230V 逆变发电机典型应用图 .....   | 13 |
| 6.4 2KW/120V 逆变发电机典型应用图 .....   | 14 |
| 6.5 3.5KW/230V 逆变发电机典型应用图 ..... | 15 |
| 6.6 3.5KW/120V 逆变发电机典型应用图 ..... | 16 |
| 7. 电气特性 .....                   | 17 |
| 7.1 极限参数 .....                  | 17 |
| 7.2 典型参数 .....                  | 18 |
| 8. 应用设计 .....                   | 21 |
| 8.1 PWM 调制方式 .....              | 21 |
| 8.2 输出电压反馈 .....                | 21 |
| 8.3 输出电流反馈 .....                | 23 |
| 8.4 温度反馈 .....                  | 24 |
| 8.5 直流母线电压反馈 .....              | 24 |
| 8.6 发动机油门控制系统 .....             | 25 |
| 9. 保护功能 .....                   | 27 |
| 9.1 输出过载保护 .....                | 27 |
| 9.2 输出过流保护 .....                | 27 |
| 9.3 直流母线电压过压、欠压保护 .....         | 27 |
| 9.4 PCB 过温保护 .....              | 27 |
| 9.5 功率管过温保护 .....               | 27 |
| 9.6 短路保护 .....                  | 27 |
| 9.7 MOS 管峰值电流保护 .....           | 28 |
| 10. 并机应用 .....                  | 29 |
| 11. 通讯功能 (UART) .....           | 30 |
| 11.1 串口描述 .....                 | 30 |
| 11.2 APP 功能 .....               | 30 |
| 11.2.1 APP 消息发送 .....           | 30 |
| 11.2.2 APP 消息接收 .....           | 31 |
| 11.3 CFG 功能 .....               | 34 |
| 11.3.1 CFG 请求消息 .....           | 34 |

|        |                     |    |
|--------|---------------------|----|
| 11.3.2 | CFG 应答消息 .....      | 34 |
| 11.3.3 | 0x10 服务-会话切换 .....  | 35 |
| 11.3.4 | 0x22 服务-读 DID ..... | 35 |
| 11.3.5 | 0x2E 服务-写 DID.....  | 36 |
| 11.3.6 | 0x21 服务-读 CFG ..... | 38 |
| 11.3.7 | 0x2D 服务-写 CFG ..... | 38 |
| 11.3.8 | 0x2F 服务-IO 控制 ..... | 42 |
| 11.3.9 | 0x80 服务-校准 .....    | 43 |
| 12.    | 封装尺寸 .....          | 45 |
| 12.1   | LQFP64 .....        | 45 |

# EG8062 芯片数据手册 V1.0

## 1. 特点

- 采用电流模式、中心对齐 PWM 调制方式，能带感性和容性负载
- SPWM 载波频率 20KHz，适合大功率 MOS 管和 IGBT 管的应用
- 集成了两路 600V 半桥高压 MOS 管驱动器，驱动能力为  $\pm 2A$
- 集成四路独立的 MOS 管峰值电流保护电路及内置四路 200mV 基准源的比较器供用户设定保护值
- 输出电压和输出电流是每个 PWM 周期实时处理，能实现精确跟踪
- 集成发动机油门控制，转速随负载的增加而自动增加
- 支持多机并机功能
- 逆变器保护功能：
  - 直流母线电压过压和欠压保护
  - 交流输出欠压保护
  - 输出过载保护
  - 输出过流保护
  - PCB 过温保护和 IGBT 过温保护
  - 输出短路保护
- 串口通讯可设置参数：
  - 50Hz 纯正弦波固定频率
  - 60Hz 纯正弦波固定频率
  - 交流输出电压
  - 死区时间
  - 温度保护值
  - 额定功率保护值
  - 额定电流保护值
  - ECO 节能模式
  - 故障复位
- 串口通讯可读参数：
  - 发动机转速
  - 交流输出电压
  - 交流输出频率
  - 交流输出功率
  - 交流输出电流
  - 直流母线电压
  - 故障代码

## 2. 描述

EG8062 是一款采用电流模式、中心对齐 PWM 调制方式和内置两路 600V 半桥高压 MOS 驱动器、油门控制模块和面板控制模块等一体的数模结合芯片，专用于逆变发电机产品。

该芯片采用 CMOS 工艺，内部的主要模块为 SPWM 正弦发生器、死区时间控制电路、多路反馈及保护处理电路、UART 串口通讯模块、步进电机控制模块和发电机面板控制及指示电路等功能。

该芯片采用 LQFP64 封装，只需少量的外围器件，即可实现空载波形失真率小于 1.5%、满载波形失真率小于 3% 和高精度输出电压的特性，能满足逆变发电机行业的波形要求。

集成了两路 600V 半桥高压 MOS 驱动器，驱动器的输出电流能力为  $\pm 2A$ ，内置四路独立的逐周 PWM 关断保护，可有效防止在极端情况下过高的峰值电流而损坏 MOS 的情况。另外提供了两路 SD，分别为 SD1，和 SD2，SD1 是驱动器 1 H01 和 L01 的逐周关断引脚，SD2 是驱动器 2 H02 和 L02 的逐周关断引脚，结合 MCU 和 SD 功能可实现逆变器的短路保护功能。

EG8062 内置了完善的各项保护功能，提供了直流母线过压和欠压保护、交流输出欠压保护、过载报警提示、过载保护指示、过流保护指示、过温保护指示及短路保护指示等。

EG8062 内置了 UART 串口通讯功能，用户通过串口可设配置参数，及复位功能等，也可以通过串口读出逆变器运行的状态及相关数据。

## 3. 应用领域

- 数码发电机
- 变频发电机
- 储能电源

# 4. 引脚

## 4.1 引脚定义

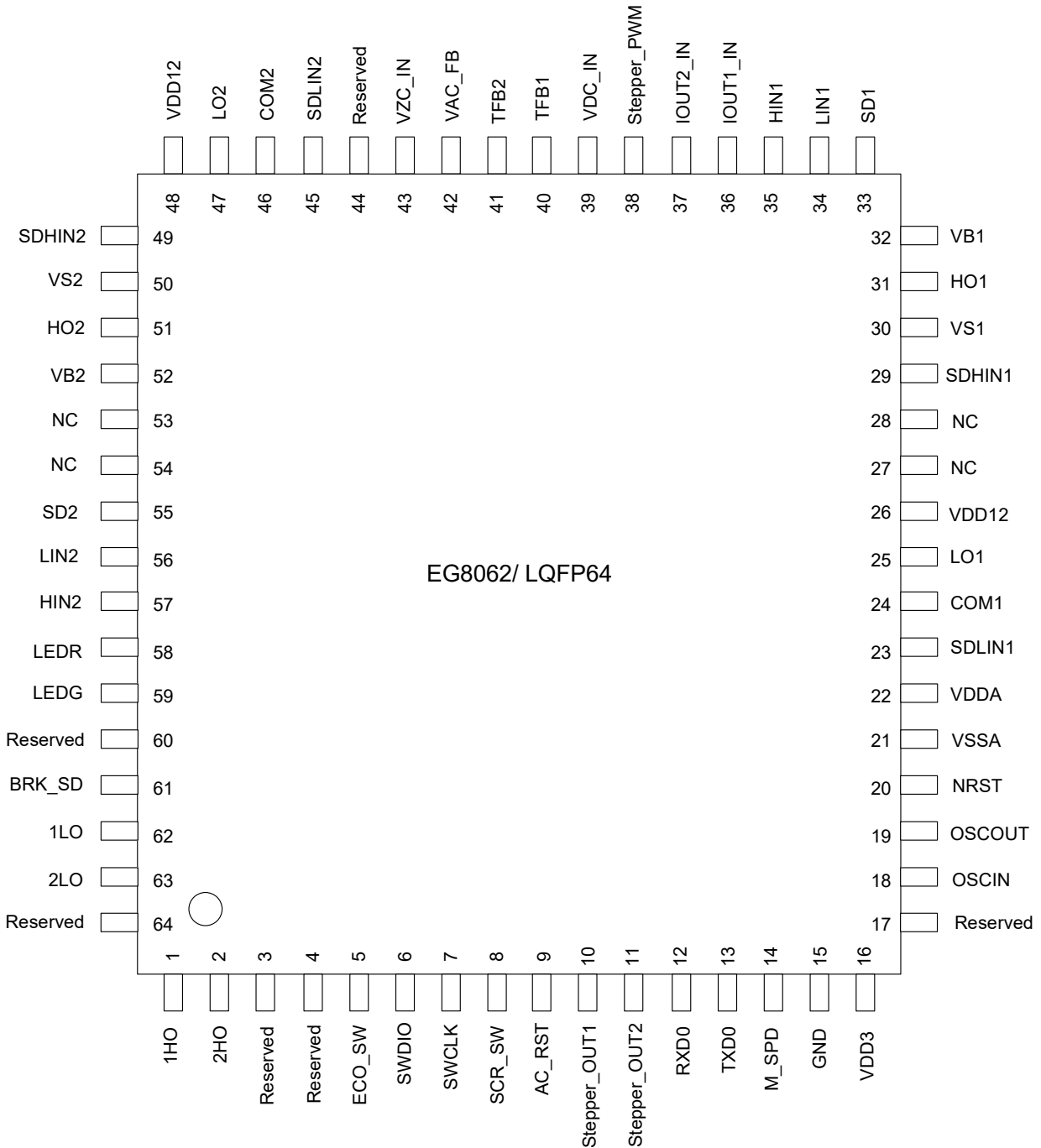


图 4-1. EG8062 管脚定义

## 4.2 引脚描述

| 引脚序号 | 引脚名称         | I/O   | 描述   |
|------|--------------|-------|--|
| 1    | 1HO          | O     | 右桥臂上管 SPWM 输出信号，输出到芯片的引脚 35（HIN1）            |
| 2    | 2HO          | O     | 左桥臂上管 SPWM 输出信号，输出到芯片的引脚 57（HIN2）            |
| 3    | Reserved     | -     | 保留   |
| 4    | Reserved     | -     | 保留   |
| 5    | ECO_SW       | I     | 节能模式控制信号输入，输入“1”为节能模式开，“0”为节能模式关             |
| 6    | SWDIO        | I     | 固件升级数据                                       |
| 7    | SWCLK        | I     | 固件升级时钟                                       |
| 8    | SCR_SW       | O     | 可控硅 ON/OFF 控制，需外接 PC817 光耦，高电平开启可控硅，低电平关闭可控硅 |
| 9    | AC_RST       | I     | 输出复位，默认上拉 3.3V 高电位，输入低电平脉冲复位输出               |
| 10   | Stepper_OUT1 | O     | 步进电机控制信号 1，需外接步进电机驱动芯片                       |
| 11   | Stepper_OUT2 | O     | 步进电机控制信号 2，需外接步进电机驱动芯片                       |
| 12   | RXD          | I     | 串口通讯数据接收端                                    |
| 13   | TXD          | O     | 串口通讯数据发送端                                    |
| 14   | M_SPD        | I     | 发动机转速信号输入                                    |
| 15   | GND          | GND   | 芯片的地端  |
| 16   | VDD3         | Power | 芯片的 +3.3V 电源端                                |
| 17   | Reserved     | -     | 保留   |
| 18   | OSC_IN       | I     | 4M 晶体振荡器引脚 1                                 |
| 19   | OSC_OUT      | O     | 4M 晶体振荡器引脚 2                                 |
| 20   | NRST         | I     | 复位脚，需外接 0.1uF 电容到 GND                        |
| 21   | VSSA         | GND   | 芯片的模拟地                                       |
| 22   | VDDA         | Power | 芯片的模拟电路，+3.3V 电源输入端                          |
| 23   | SDLIN1       | I     | 驱动器 1 的低端 MOS 管峰值电流保护输入端，内部基准电压为 200mV       |
| 24   | COM1         | I     | 驱动芯片 1 的功率地                                  |
| 25   | LO1          | O     | 驱动器 1 的低端门极驱动输出                              |
| 26   | VDD12        | Power | 驱动器 1 的电源，电压范围为 10V-20V                      |
| 27   | NC           | -     | 空脚，用于高压隔离                                    |
| 28   | NC           | -     | 空脚，用于高压隔离                                    |
| 29   | SDHIN1       | I     | 驱动器 2 的高端 MOS 管峰值电流保护输入端，内部基准电压为 200mV       |
| 30   | VS1          | O     | 驱动器 1 的高端悬浮端输出                               |
| 31   | HO1          | O     | 驱动器 1 的高端门极驱动输出                              |
| 32   | VB1          | Power | 驱动器 1 的悬浮电源，需外接 10uF 的自举电容                   |
| 33   | SD1          | I     | 驱动器 1 的 SD 控制端                               |
| 34   | LIN1         | I     | 驱动器 1 的低端控制输入脚，需连接到引脚 62（1L0）                |
| 35   | HIN1         | I     | 驱动器 1 的高端控制输入脚，需连接到引脚 1（1HO）                 |
| 36   | IOUT1_IN     | I     | 右桥臂正弦电流采样输入端                                 |
| 37   | IOUT2_IN     | I     | 左桥臂正弦电流采样输入端                                 |
| 38   | Stepper_PWM  | I     | 步进电机使能脚，高电平输出开启驱动器，低电平输出关闭驱动器                |

|    |          |       |  |
|----|----------|-------|--|
| 39 | VDC_IN   | I     | 直流母线电压反馈输入，低于 1.8V 欠压关断，高于 2.8V 过压关断   |
| 40 | TFB1     | I     | 温度反馈输入通道 1                             |
| 41 | TFB2     | I     | 温度反馈输入通道 2                             |
| 42 | VAC_FB   | I     | 输出交流电压反馈输入，用于电压环稳压                     |
| 43 | VZC_IN   | I     | 交流电压过零信号输入                             |
| 44 | Reserved | -     | 保留                                     |
| 45 | SDLIN2   | I     | 驱动器 2 的低端 MOS 管峰值电流保护输入端，内部基准电压为 200mV |
| 46 | COM2     | GND   | 驱动器 2 的功率地                             |
| 47 | LO2      | O     | 驱动器 2 的低端门极驱动输出                        |
| 48 | VDD12    | Power | 驱动器 2 的电源，电压范围为 10V-20V                |
| 49 | SDHIN2   | I     | 驱动器 2 的高端 MOS 管峰值电流保护输入端，内部基准电压为 200mV |
| 50 | VS2      | O     | 驱动器 2 的高端悬浮端输出                         |
| 51 | HO2      | O     | 驱动器 2 的高端门极驱动输出                        |
| 52 | VB2      | Power | 驱动器 2 的悬浮电源，需外接 10uF 的自举电容             |
| 53 | NC       | -     | 空脚，用于高压隔离                              |
| 54 | NC       | -     | 空脚，用于高压隔离                              |
| 55 | SD2      | I     | 驱动器 2 的 SD 控制端                         |
| 56 | LIN2     | I     | 驱动器 2 的低端控制输入脚，连接到引脚 63（2LO）           |
| 57 | HIN2     | I     | 驱动器 2 的高端控制输入脚，连接到引脚 2（2HO）            |
| 58 | LEDR     | O     | 故障 LED 指示输出                            |
| 59 | LEDG     | O     | 运行 LED 指示输出                            |
| 60 | Reserved | -     | 保留                                     |
| 61 | BRK_SD   | I     | SD 控制端，用于检测负载短路保护                      |
| 62 | 1LO      | O     | 右桥臂下管 SPWM 输出信号，输出到芯片的引脚 34（LIN1）      |
| 63 | 2LO      | O     | 左桥臂下管 SPWM 输出信号，输出到芯片的引脚 56（LIN2）      |
| 64 | Reserved | -     | 保留                                     |

## 5. 结构框图

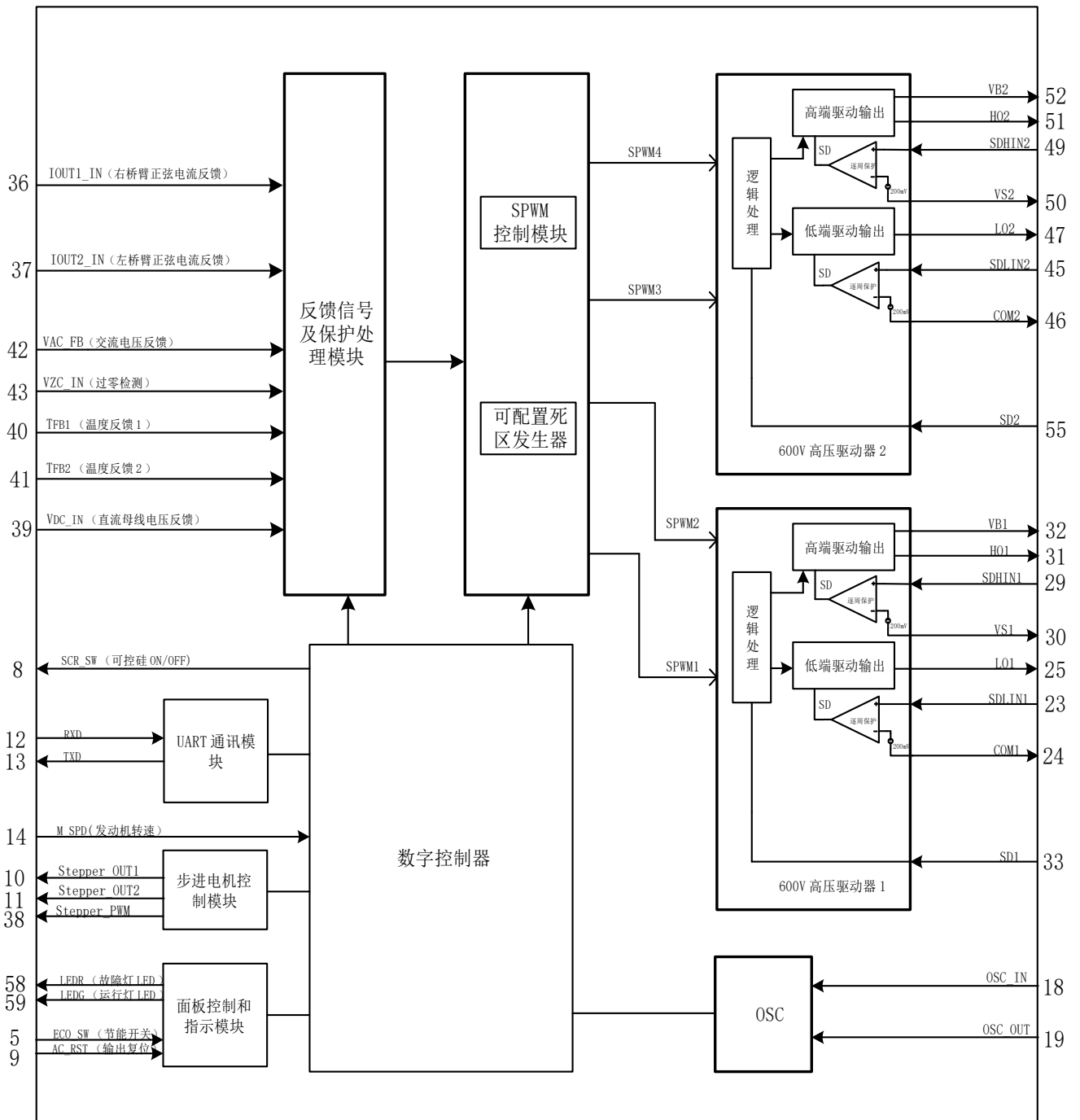


图 5-1. EG8062 结构框图

# 6. 典型应用电路

## 6.1 1KW/230V逆变发电机典型应用图

1KW/230V逆变发电机典型应用图

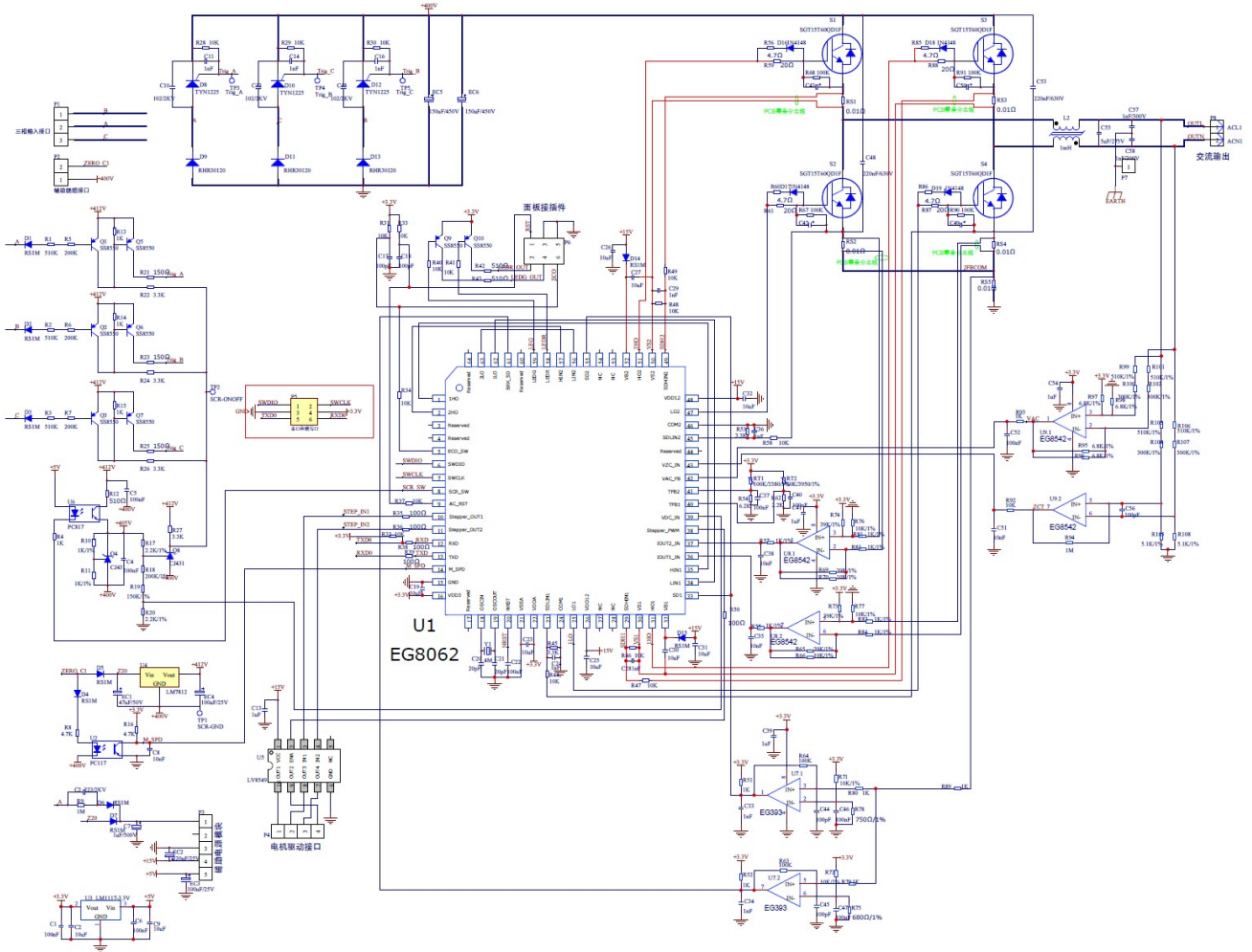


图 6-1. 1KW/230V 逆变发电机典型应用图

## 6.2 1KW/120V逆变发电机典型应用图

1KW/120V逆变发电机典型应用图

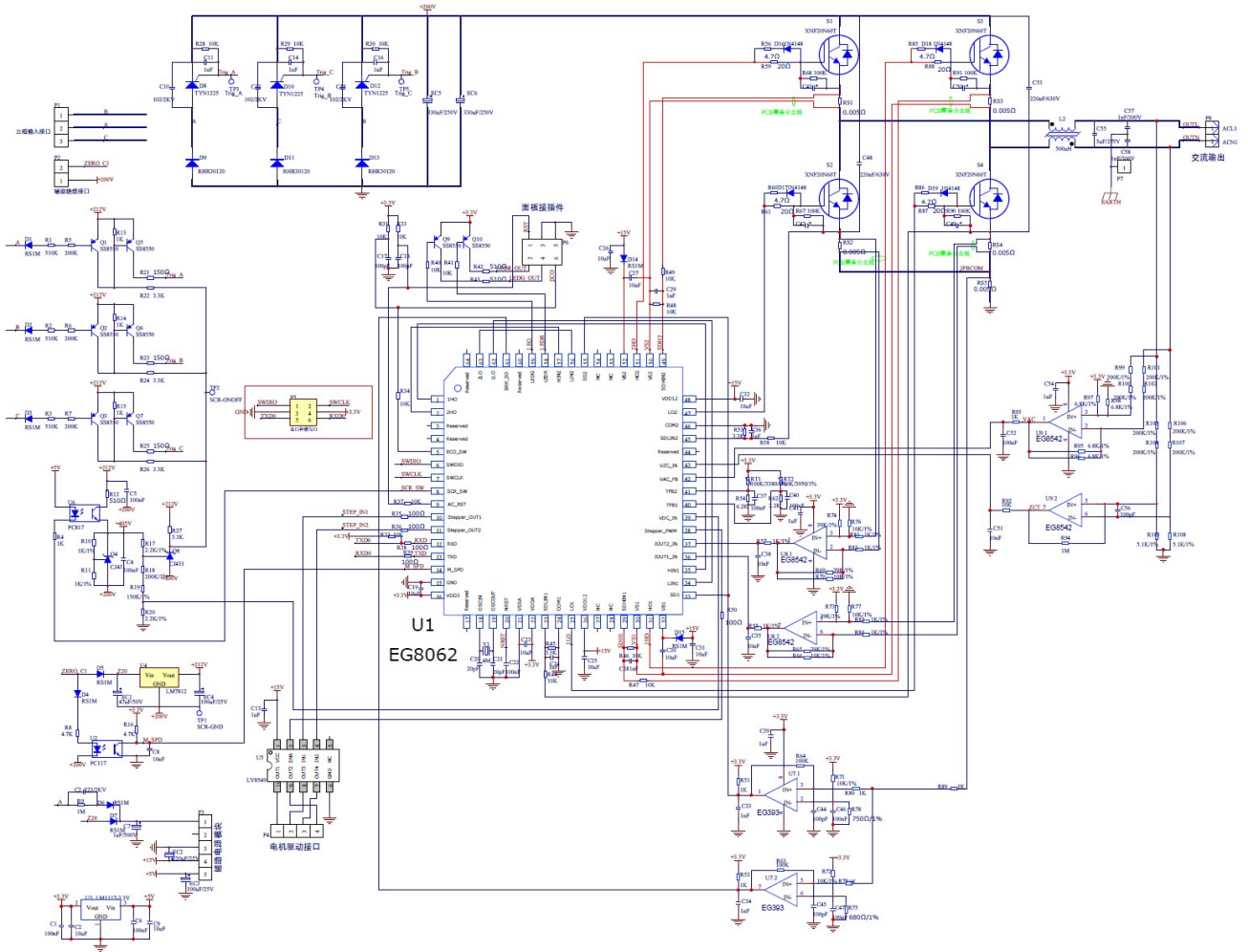


图 6-2 1KW/120V 逆变发电机典型应用图

6.3 2KW/230V逆变发电机典型应用图

2KW/230V逆变发电机典型应用图

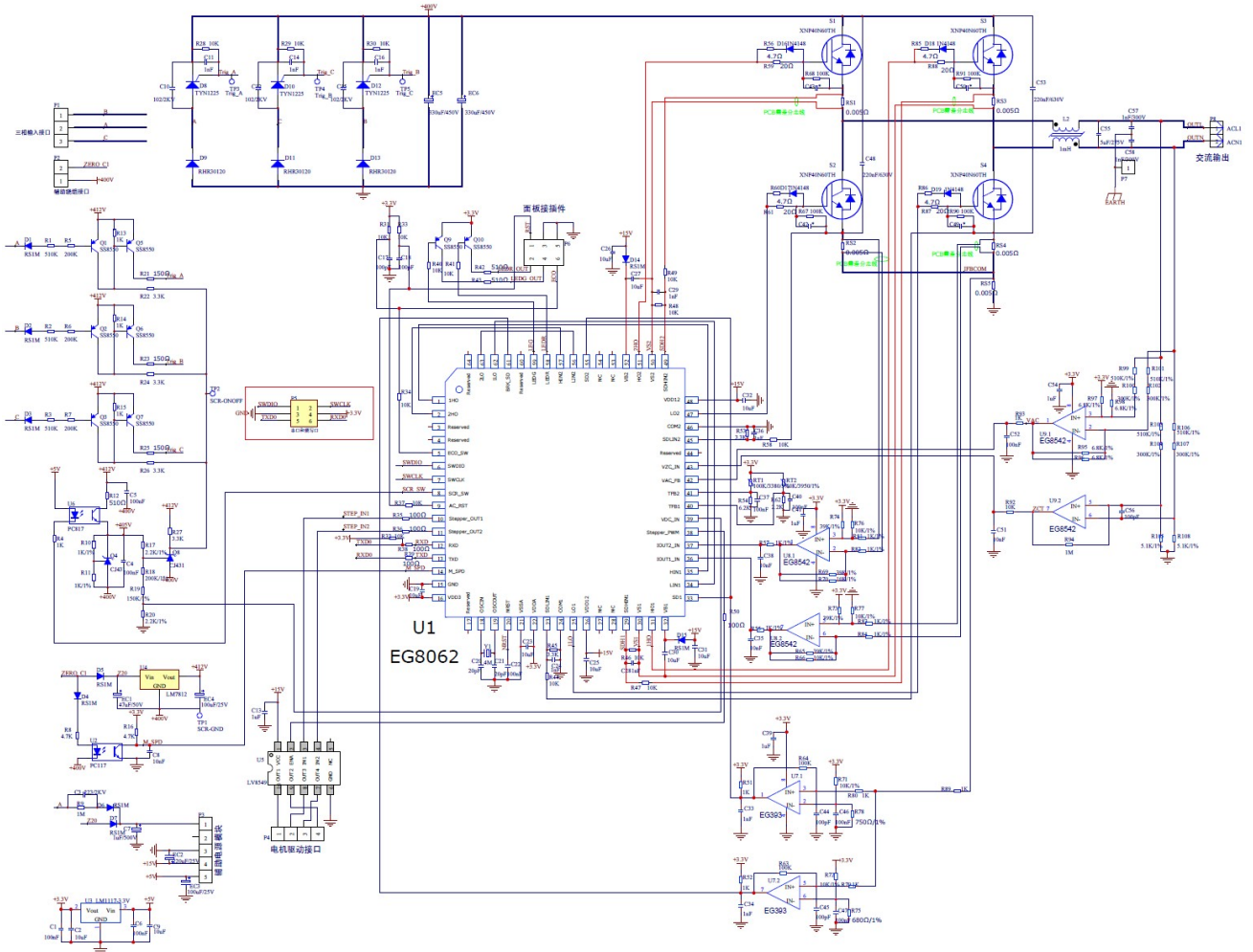


图 6-3 2KW/230V 逆变发电机典型应用图

### 6.4 2KW/120V逆变发电机典型应用图

2KW/120V逆变发电机典型应用图

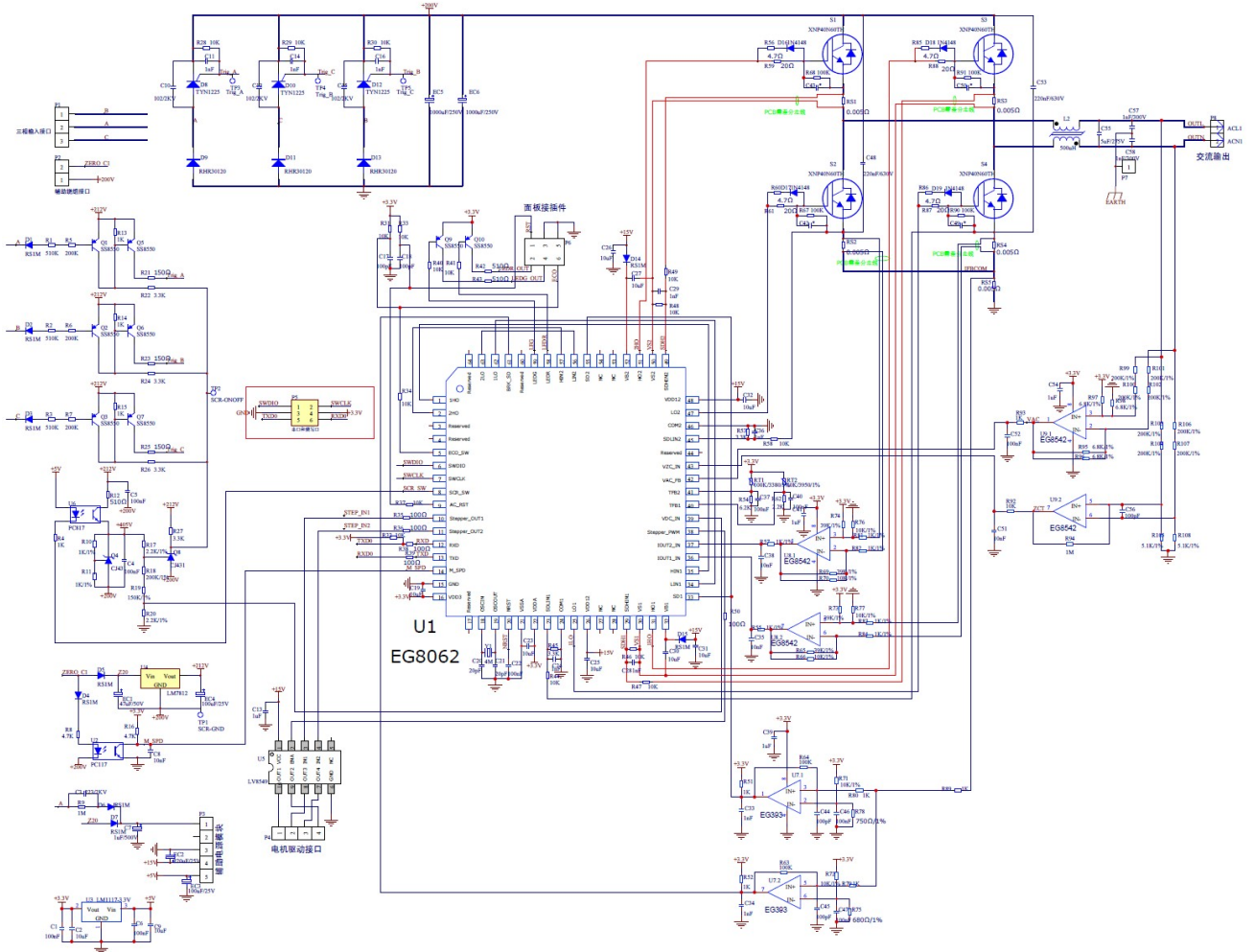


图 6-4 2KW/120V 逆变发电机典型应用图

### 6.5 3.5KW/230V逆变发电机典型应用图

3.5KW/230V逆变发电机典型应用图

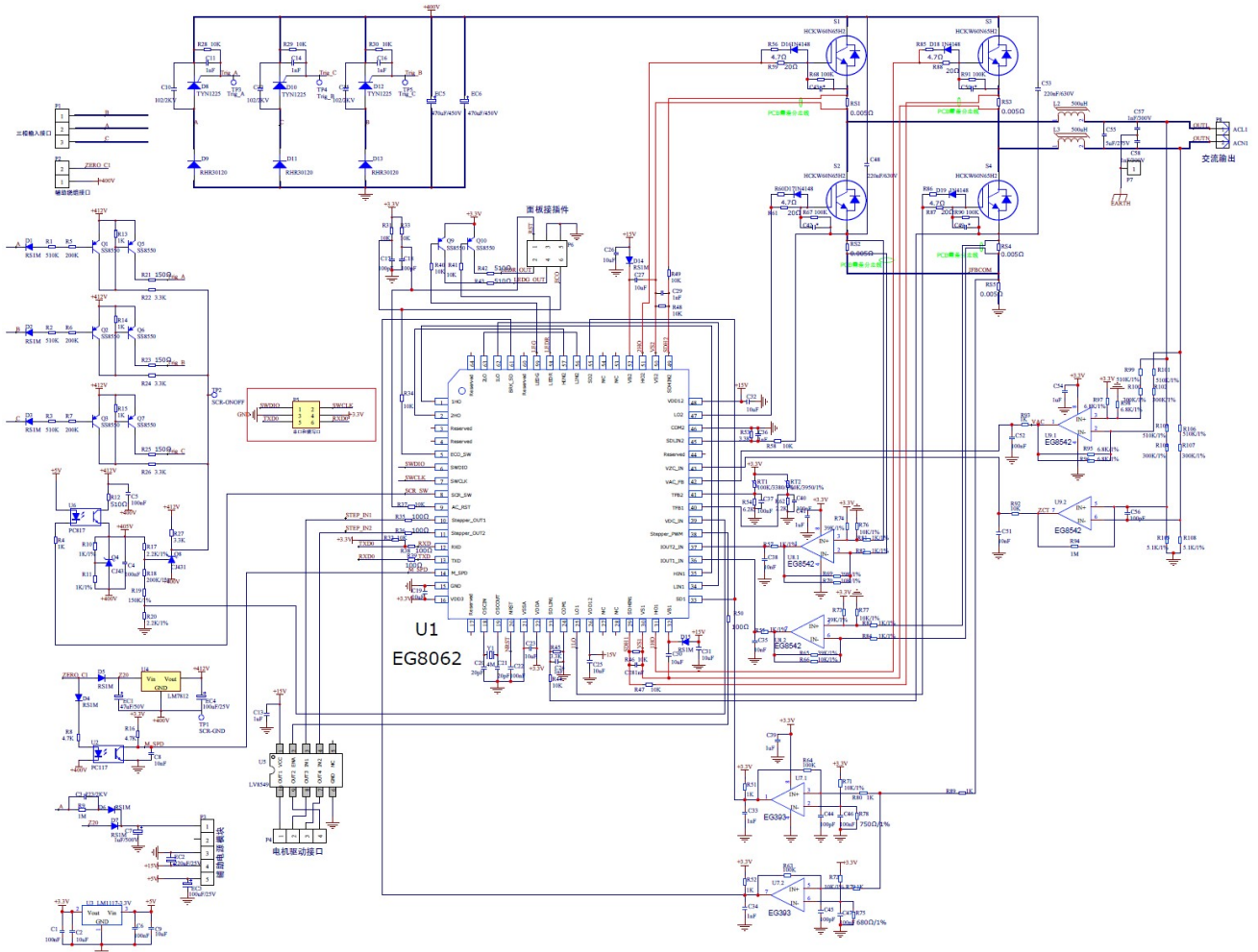


图 6-5 3.5KW/230V 逆变发电机典型应用图

6.6 3.5KW/120V逆变发电机典型应用图

3.5KW/120V逆变发电机典型应用图

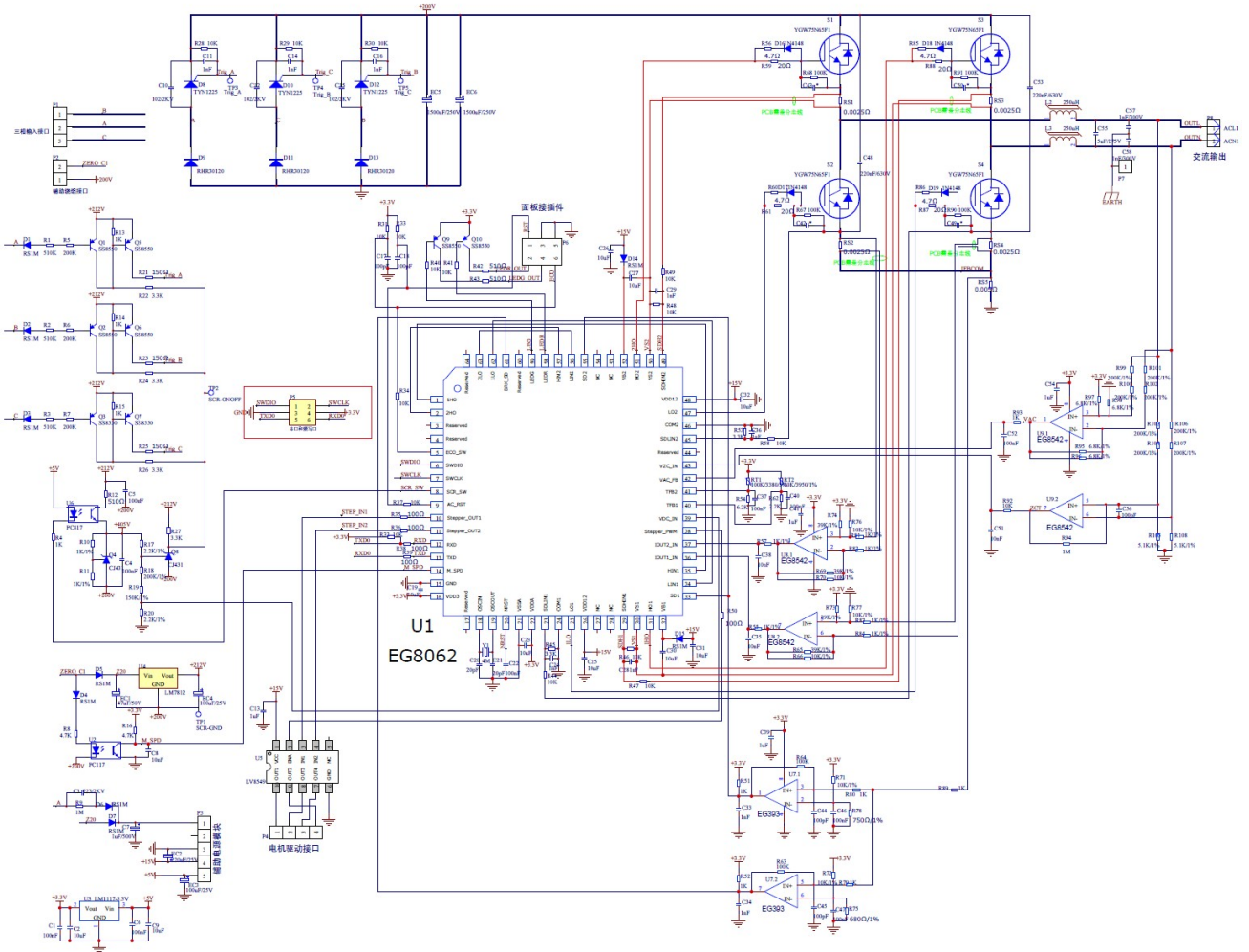


图 6-6 3.5KW/120V 逆变发电机典型应用图

## 7. 电气特性

### 7.1 极限参数

无另外说明，在 TA=25℃ 条件下

| 符号                | 参数名称          | 测试条件                    | 最小     | 最大        | 单位 |
|-------------------|---------------|-------------------------|--------|-----------|----|
| <b>高压 MOS 驱动器</b> |               |                         |        |           |    |
| VDD12             | 驱动器功率电源       | VDD12 引脚相对 COM 的电压      | -0.3   | 20        | V  |
| VB1, VB2          | 自举高端电源        | -                       | -0.3   | 600       | V  |
| HO1, HO2          | 高端输出          | -                       | VS-0.3 | VB+0.3    | V  |
| VS1, VS2          | 悬浮地端          | -                       | VB-20  | VB+0.3    | V  |
| SDHIN1, SDHIN2    | 高端 SD 比较器输入正端 | -                       | VS-0.3 | VS+5      | V  |
| LO1, LO2          | 低端输出          | -                       | -0.3   | VCC12+0.3 | V  |
| SDLIN1, SDLIN2,   | 低端 SD 比较器输入正端 | -                       | -0.3   | 5.5       | V  |
| SD1, SD2          | SD 逻辑控制输入     | -                       | -0.3   | 5.5       | V  |
| <b>数字控制器</b>      |               |                         |        |           |    |
| VDD3, VDDA        | 电源            | VDD3, VDDA 引脚相对 GND 的电压 | -0.3   | 3.6       | V  |
| I/O               | 所有输入输出端口      | 所有 I/O 引脚对 GND 的电压      | -0.3   | 3.6       | V  |
| Isink             | 输出引脚的最大输出灌电流  | -                       | -      | 25        | mA |
| Isouce            | 输出引脚的最大输出拉电流  | -                       | -      | -25       | mA |
| TA                | 环境温度          | -                       | -45    | 105       | ℃  |
| Tstr              | 储存温度          | -                       | -65    | 125       | ℃  |

注：超出所列的极限参数可能导致芯片内部永久性损坏，在极限的条件长时间运行会影响芯片的可靠性。

## 7.2 典型参数

无另外说明，在  $T_A=25^{\circ}\text{C}$ ,  $\text{OSC}=4\text{MHz}$

| 符号                       | 参数名称                       | 测试条件  | 最小  | 典型   | 最大  | 单位 |
|--------------------------|----------------------------|---|-----|------|-----|----|
| VCC12                    | 驱动器功率电源                    | VCC12 引脚相对 COM 的电压  | 10  | 15   | 18  | V  |
| VDDA, VDD3               | 控制器电源                      | VDDA 和 VDD3 引脚相对 GND 的电压  | 2.4 | 3.3  | 3.6 | V  |
| Ivcc12                   | VCC12 静态电流                 | VCC12=15V   | -   | 1    | 1.2 | mA |
| Ivdd                     | VDDA, VDD3 静态电流            | VDD3=3.3V   | -   | 10   | 20  | mA |
| <b>高压 MOS 驱动器</b>        |                            |   |     |      |     |    |
| VB1, VB2                 | 高端驱动器供电电源                  | 相对 COM 端电压  | 10  |      | 600 | V  |
| SDHIN1, SDHIN2           | 高端电流比较器                    | 相对 VS 电压  | -   | 200  | -   | mV |
| SDLIN1, SDLIN2           | 低端电流比较器                    | 相对 COM 端电压  | -   | 200  | -   | mV |
| SD1, SD2                 | SD 逻辑控制输入                  | 相对 GND 的电压  | 0   |      | 3.6 | V  |
| L01、L02、H01、H02<br>输出拉电流 | I <sub>O+</sub>            | V <sub>O</sub> =0V, V <sub>IN</sub> = V <sub>IH</sub><br>PW≤10uS  | 1.8 | 2    | -   | A  |
| L01、L02、H01、H02<br>输出灌电流 | I <sub>O-</sub>            | V <sub>O</sub> =12V, V <sub>IN</sub> = V <sub>IL</sub><br>PW≤10uS | 2   | 2.5  | -   | A  |
| <b>反馈</b>                |                            |   |     |      |     |    |
| VAC_FB                   | 电压反馈通道<br>直流共模分量           | VDD3=3.3V   | -   | 1.65 | -   | V  |
|                          | 电压反馈通道<br>正弦基准源幅值          | VDD3=3.3V   | -   | 1.36 | -   | V  |
| IOUT1_IN                 | 电流反馈通道<br>1<br>放大器输入       | VDD3=3.3V   | -   |      | 320 | mV |
| IOUT2_IN                 | 电流反馈通道<br>2<br>放大器输入正<br>端 | VDD3=3.3V   | -   |      | 320 | mV |
| TFB1                     | 温度保护基准                     | VDD3=3.3V   | -   | 2.2  | -   | V  |

|                                |                   |                      |      |     |     |   |
|--------------------------------|-------------------|----------------------|------|-----|-----|---|
|                                | 电压 1              |                      |      |     |     |   |
| TFB2                           | 温度保护基准<br>电压 2    | VDD3=3.3V            | -    | 2.1 | -   | V |
| VDC_IN                         | 直流母线电压<br>保护      | VDD3=3.3V            | 1.8  | -   | 2.8 | V |
| <b>UART 通讯</b>                 |                   |                      |      |     |     |   |
| RXD, TXD                       | Vout (H)<br>输出高电位 | VDD3=3.3V, IOH=-10mA | -    | 3.3 | -   | V |
|                                | Vout (L)<br>输出低电位 | VDD3=3.3V, IOL=10mA  | -    | 0   | 0.3 | V |
| <b>步进电机控制</b>                  |                   |                      |      |     |     |   |
| Stepper_OUT1                   | Vout (H)<br>输出高电位 | VDD3=3.3V, IOH=-10mA | -    | 3.3 | -   | V |
|                                | Vout (L)<br>输出低电位 | VDD3=3.3V, IOL=10mA  | -    | 0   | 0.3 | V |
| Stepper_OUT2                   | Vout (H)<br>输出高电位 | VDD3=3.3V, IOH=-10mA | -    | 3.3 | -   | V |
|                                | Vout (L)<br>输出低电位 | VDD3=3.3V, IOL=10mA  | -    | 0   | 0.3 | V |
| Stepper_PWM                    | Vin (H)<br>输入高电位  | VDD3=3.3V            | -    | 3.3 | -   | V |
|                                | Vin (L)<br>输入低电位  | VDD3=3.3V            | -0.3 | 0   | 0.7 | V |
| <b>可控硅 ON/OFF 控制及发动机转速输入信号</b> |                   |                      |      |     |     |   |
| SCR_SW                         | Vout (H)<br>输出高电位 | VDD3=3.3V, IOH=-10mA | -    | 3.3 | -   | V |
|                                | Vout (L)<br>输出低电位 | VDD3=3.3V, IOL=10mA  | -    | 0   | 0.3 | V |
| M_SPD                          | Vin (H)<br>输入高电位  | VDD3=3.3V            | -    | 3.3 | -   | V |
|                                | Vin (L)<br>输入低电位  | VDD3=3.3V            | -0.3 | 0   | 0.7 | V |
| <b>面板控制和指示模块</b>               |                   |                      |      |     |     |   |
| LEDR, LEDG                     | Vout (H)<br>输出高电位 | VDD3=3.3V, IOH=-10mA | -    | 3.3 | -   | V |
|                                | Vout (L)<br>输出低电位 | VDD3=3.3V, IOL=10mA  | -    | 0   | 0.3 | V |
| ECO_SW                         | Vin (H)<br>输入高电位  | VDD3=3.3V            | -    | 3.3 | -   | V |
|                                | Vin (L)<br>输入低电位  | VDD3=3.3V            | -0.3 | 0   | 0.7 | V |

|        |                              |           |      |     |     |   |
|--------|------------------------------|-----------|------|-----|-----|---|
|        |                              |           |      |     |     |   |
| AC_RST | V <sub>in</sub> (H)<br>输入高电位 | VDD3=3.3V | -    | 3.3 | -   | V |
|        | V <sub>in</sub> (L)<br>输入低电位 | VDD3=3.3V | -0.3 | 0   | 0.7 | V |

## 8. 应用设计

### 8.1 PWM 调制方式

EG8062 采用中间对齐 PWM 调制方式，调制频率为 20KHz，该调制方式的优点是 H 桥上开关管的频率为 20KHz，输出电感和输出电容上的开关频率是 PWM 频率的二倍（40KHz），跟传统逆变器的单极性或双极性调制方式相比，在相同功率下，MOS 管或 IGBT 管上的开关损耗相同，作用在输出电感和电容上的频率是传统的 2 倍，这种调制方式可以降低电感的体积和线径。

### 8.2 输出电压反馈

EG8062 是通过外部差分运放，实时采样交流输出电压，输出电压调整率为每个 PWM 周期 (50uS) 时间，跟传统逆变器的峰值电压采样相比，EG8062 的输出电压精度和动态响应时间大大提高，波形失真度能在 3% 以内。

EG8062 内部的正弦电压基准为直流偏移量 1.65V+ 幅值 1.36V 的 50Hz 或 60Hz 正弦波，通过差分运放的放大和直流偏移量 1.65V 叠加后，送入到 EG8062 的 42 脚 VAC\_FB，经误差计算后，然后调整相应的输出电压。具体应用电路图可参考图 8.2a 和图 8.2b。

应用时必须保证差分运放外接电阻的对称性，即  $(R32+R31)=(R37+R36)$ ， $(R39//R40)=(R29//R28)$

当设定输出电压值时，下列的计算公式供参考设计：

**用于 230V 输出电压的计算步骤，电路结构图如图 8.2a：**

第一步、计算直流偏移量：运放输出  $V_{out\_DC} = \frac{R28}{R28+R29} \times 3.3V = 6.8K/13.6K \times 3.3V \approx 1.65V$ （参考图 8.2a）

第二步、计算运放放大倍数： $A = (R39//R40)/(R32+R31)$

第三步、计算输出电压： $V_{out\_AC} = A \times V_{in}$

$$V_{out\_AC} = \frac{R39 // R40}{R32 + R31} \times (V_{ACL} - V_{ACN}) \approx \frac{1}{238} \times (V_{ACL} - V_{ACN}) \quad (\text{参考图 8.2a})$$

从上述公式得出，该运放的放大倍数为 1/238 倍，比如当交流 230V 输出，峰值电压为 325V，经差分运放 1/238 倍的放大后，输入到 EG8062 的 42 脚，其幅度为  $V_{AC\_FB} = V_{out\_DC} + V_{out\_AC} = 1.65V + 1.365V = 3.015V$ ，然后经内部电路误差计算后，再调整输出电压。

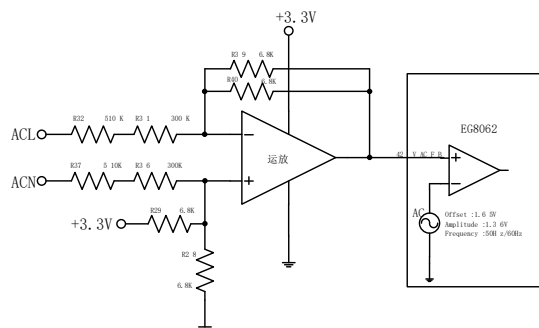


图 8.2a 用于交流 230V 输出电压的反馈原理图

用于 120V 输出的计算步骤，电路结构图如图 8.2b 的参数：

第一步、计算直流偏移量：运放输出  $V_{out\_DC} = \frac{R_{28}}{R_{28}+R_{29}} \times 3.3V = 6.8K/13.6K \times 3.3V \approx 1.65V$ （参考图 8.2b）

第二步、计算运放放大倍数： $A = (R_{39} // R_{40}) / (R_{32} + R_{31})$

第三步、计算输出电压： $V_{out\_AC} = A \times V_{in}$

$$V_{out\_AC} = \frac{R_{39} // R_{40}}{R_{32} + R_{31}} \times (V_{ACL} - V_{ACN}) \approx \frac{1}{118} \times (V_{ACL} - V_{ACN}) \quad (\text{电阻值参考图 8.2b})$$

从上述公式得出，该运放的放大倍数为 1/118 倍，比如当交流 120V 输出的峰值电压为 170V 时，经差分运放 1/118 倍的放大后，输入到 EG8062 的 42 脚，其幅度为  $V_{AC\_FB} = V_{out\_DC} + V_{out\_AC} = 1.65V + 1.44V = 3.09V$ ，然后经内部电路误差计算后，再调整输出电压。

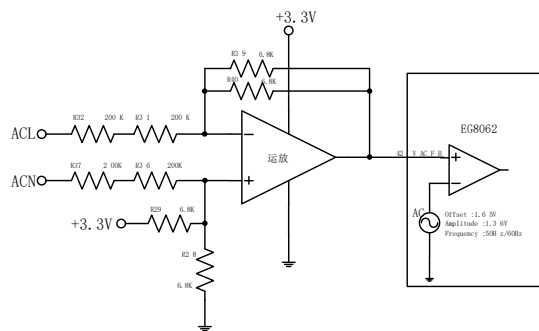


图 8.2b 用于交流 120V 输出电压的反馈原理图

### 8.3 输出电流反馈

EG8062 采用了交替轮流方式采样输出电流，电路结构如图 8.3a 所示，IOUT2\_IN 和 IOUT1\_IN 分别用于采样 RS2 和 RS4 电阻上的电流，RS2 和 RS4 电阻上的电流信号需经外部运放放大后送入 IOUT2\_IN 和 IOUT1\_IN 引脚进行运算，如图 8.3a 的电路，默认配置增益为 7.95 倍放大，没有特殊需要的情况下，建议不要运放的增益配置，否则会影响电流、功率保护和显示等，PCB 走线时，RS2 和 RS4 到运放输入端的走线需走差分信号线。

EG8062 的电流反馈值设置了最大饱和电流限制，最大饱和电流为： $I_{max}=3000mV/7.95/R_s$  如  $R_s$  选  $0.01\Omega$  时，可得出最大电流  $I_{max}=3000mV/7.95/0.01\Omega=37.73A$ ，该电流值是比较适合 1.5KW 的逆变器应用，不同功率应用时，可选取不同阻值的采样电阻值，可参考上述的公式进行计算。

应用时，IOUT2\_IN 和 IOUT1\_IN 脚不能做悬空或接地处理，必须要严格按照 8.3a 所示的接法，否则不能输出正常正弦波形。

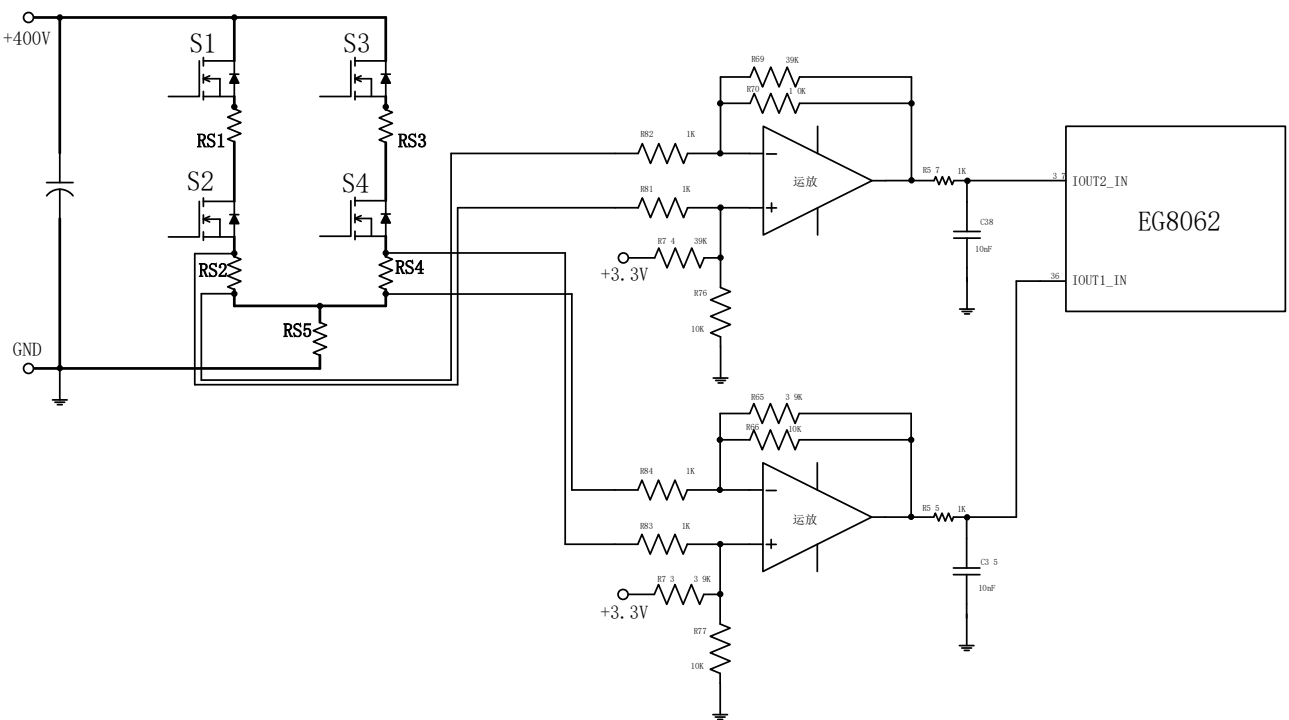


图 8.3a 输出电流反馈原理图

在批量生产时，考虑硬件精度的误差导致输出电压和输出电流的偏差，屹晶微公司提供了上位机软件，供输出电压和输出电流校正，用户可以到我司的网站或联系我们，下载相应的上位机软件。

### 8.4 温度反馈

EG8062 提供了两路温度反馈 TFB1 和 TFB2 用于温度检测及保护，TFB1 主要用于 PCB 板级的温度保护，TFB2 主要用于功率管的温度保护，电路结构如图 8.4a 所示。

TFB1 的检测电路由 RT1 和电阻 R60 组成一个简单的分压电路，RT1 选用 25°C 对应阻值 10K（B 常数值为 3950）的 NTC 热敏电阻，下拉电阻 R60 选用 2.2KΩ，TFB1 引脚的过温电压值设定在 2.2V，对应温度保护值 85°C 左右（此时 NTC 的电阻值为 1.08K），退出过温的迟滞值为 10°C，即低于 75°C 退出过温保护。

TFB2 的检测电路由 RT2 和电阻 R61 组成一个简单的分压电路，TFB2 主要用功率管的温度保护，RT2 选用 25°C 对应阻值 100K（B 常数值为 3950）的 NTC 热敏电阻，下拉电阻 R61 选用 6.2KΩ，TFB1 引脚的过温电压值设定在 2.1V，对应温度保护值 125°C 左右（此时 NTC 的电阻值为 3.58K），退出过温的迟滞值为 10°C，即低于 115°C 退出过温保护。

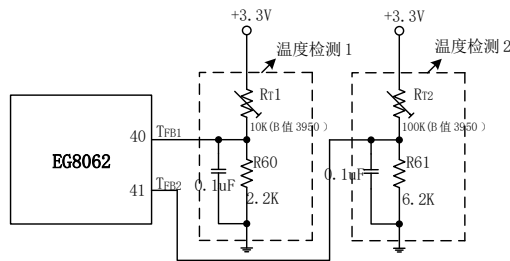


图 8.4a EG8062 的温度检测电路

### 8.5 直流母线电压反馈

为了防止过高或过低的母线电压输入到逆变器而引起的损坏，EG8062 芯片内置了母线电压检测电路，提供了直流母线电压过压和欠压两种关断功能，电路结构图如图 8.5a 所示，通过 EG8062 的 39 脚，外接分压电阻来实现母线电压的检测。

EG8062 芯片内部的过压关断值为 2.8V，按图 8.5a 的参数，对应的过压保护值为 440V，延时时间为 500mS，退出过压保护的迟滞值为 10V，即低于 430V 退出过压保护。

欠压关断值为 1.8V，按图 8.5a 的参数，对应的欠压保护值为 290V，延时时间为 3S，退出欠压保护的迟滞值为 30V，即高于 320V 退出欠压保护。

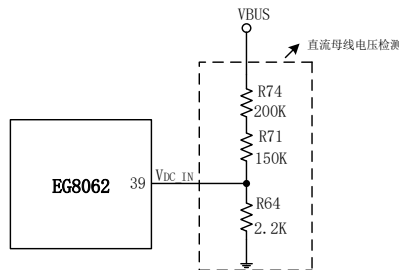


图 8.5a EG8062 的直流母线电压检测电路

## 8.6 发动机油门控制系统

EG8062 的油门控制部分主要由三部分电路模块组成：转速检测电路、数字 PID 控制模块和步进电机驱动输出电路。

转速检测电路如图 8.8a 所示，该设计是采用了 PC817 光电耦合器把发电机辅助绕组上的交流电压转换为方波脉冲信号，并送入 EG8062 的 14 脚（M\_SPD）进行数字处理，当确定电机的极对数后，就可以根据方波的频率计算出对应的转速大小。

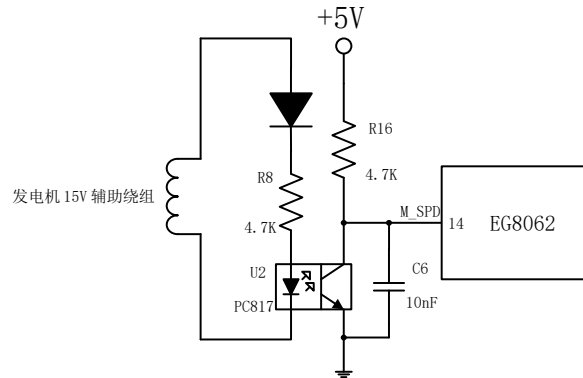


图 8.8a EG8062 的转速检测电路

发动机的油门控制系统结构原理如图 8.8b 所示，发动机转速控制系统由数字 PID 控制器和步进电机电路和组成。EG8062 的转速设定值由发动机转速曲线特性对应输出功率决定，转速会随负载的增大而自动增加。

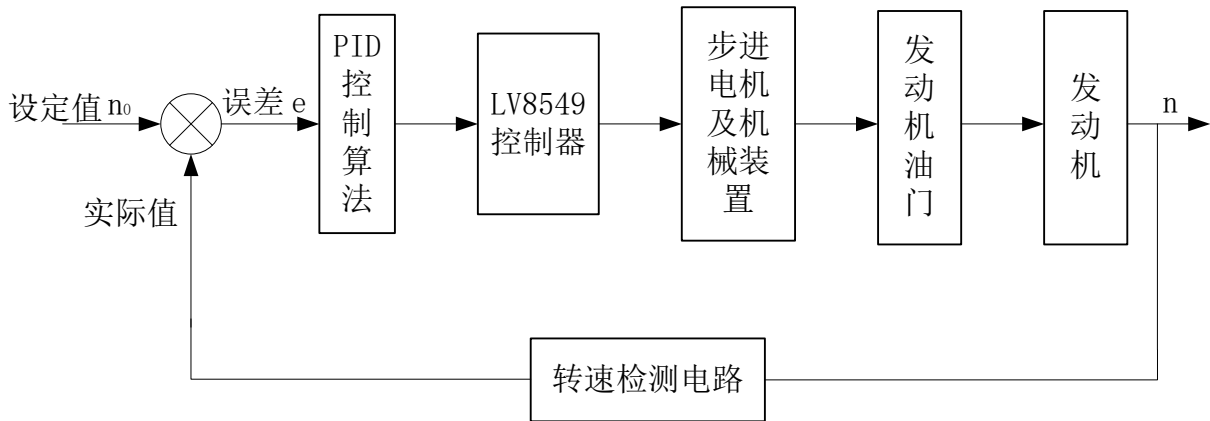


图 8.8b EG8062 的油门控制系统

EG8062 支持 4 线步进电机应用，步进电机驱动电路如图 8.8c 所示，该驱动器内置了两路 H 桥驱动，具有 1A 的驱动能力，可以直接驱动双极步进电机。步进电机的控制主要由 EG8062 的 Stepper\_PWM, Stepper\_OUT1 和 Stepper\_OUT2 来实现，Stepper\_PWM 是控制步进电机的开启和关闭，Stepper\_OUT1 和 Stepper\_OUT2 组合输出实现步进电机的正反转，步进电机的速度控制主要由 EG8062 内部的细分算法来实现，结合转速检测电路、数字 PID 控制模块、步进电机驱动及机械装置来实现发电机的油门控制。

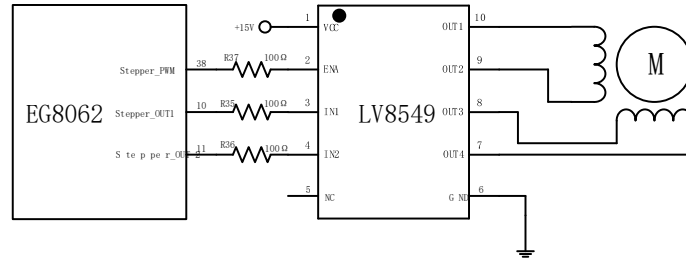


图 8.8c EG8062 步进电机驱动电路

## 9. 保护功能

EG8062 内置了完善的系统保护功能，提供了输出过载保护、输出过流保护、直流母线电压过压和欠压保护、输出电压欠压保护、过温保护及短路保护等。

同时提供了两种复位方式，一种是硬件复位，由芯片的 9 脚 AC\_RST 低脉冲有效进行复位，另一种由 UART 串口命令复位，通过逆变器开启寄存器。

下列的各项保护参数值是基于 1KW 应用图而设定的，如用户有特殊要求，屹晶微可以提供修改。

### 9.1 输出过载保护

EG8062 具有过载保护功能，过载功率大于 1100W 时，红灯 LED 指示灯开始闪烁，过载功率大于 1200W 并持续 60 秒时，逆变器关断，并输出红灯 LED 常亮，同时用户通过串口可以读出相应的保护信息。

### 9.2 输出过流保护

EG8062 具有过流保护功能，过流值大于 4.6A 时，红灯 LED 指示灯开始闪烁，过流值大于 5.5A 并持续 60 秒时，逆变器关断，并输出红灯 LED 常亮，同时用户通过串口可以读出相应的保护信息。

### 9.3 直流母线电压过压、欠压保护

EG8062 具有母线电压过压和欠压保护功能，直流电压低于 290V 或高于 440V 时，逆变器关断，并输出红灯 LED 常亮，提供用户通过串口可以读出相应的保护信息。

### 9.4 PCB 过温保护

EG8062 具有 PCB 过温保护功能，PCB 温度大于 85°C 时，逆变器关断，并输出红灯 LED 常亮，同时用户通过串口可以读出相应的保护信息。

### 9.5 功率管过温保护

EG8062 具有功率管过温保护功能，功率管温度大于 125°C 时，逆变器关断，并输出红灯 LED 常亮，同时用户通过串口可以读出相应的保护信息。

### 9.6 短路保护

EG8062 具有输出短路保护功能，短路保护时间小于 30mS，当输出短路发生时，逆变器关断，并输出红灯 LED 常亮，同时用户通过串口可以读出相应的保护信息。

### 9.7 MOS管峰值电流保护

EG8062 集成了四路独立的 MOS 管峰值电流保护电路及内置四路 200mV 基准源的比较器，供用户设定 MOS 管的峰值电流保护值，电路结构图如图 9.7a 所示。

当设定四路 MOS 管的峰值电流保护值时，下列的计算公式供参考设计：

MOS 管 S1 的峰值电流保护值为： $I_{s1\_peak}=200mV*(1+R26/R25)/RS1$

比如如图 9.7a 所示的参数， $R26=10K$ ， $R25=10K$ ， $RS1=10m\Omega$  时，

$I_{s1\_peak}=200mV*(1+10K/10K)/10m\Omega=40A$

MOS 管 S2 的峰值电流保护值为： $I_{s2\_peak}=200mV*(1+R36/R30)/(RS2+RS5)$

比如如图 9.7a 所示的参数， $R36=10K$ ， $R30=3.3K$ ， $RS2=10m\Omega$ ， $RS5=10m\Omega$  时，

$I_{s2\_peak}=200mV*(1+10K/3.3K)/(10m\Omega+10m\Omega)=40A$

MOS 管 S3 的峰值电流保护值为： $I_{s3\_peak}=200mV*(1+R33/R29)/RS3$

比如如图 9.7a 所示的参数， $R33=10K$ ， $R29=10K$ ， $RS3=10m\Omega$  时，

$I_{s3\_peak}=200mV*(1+10K/10K)/10m\Omega=40A$

MOS 管 S4 的峰值电流保护值为： $I_{s4\_peak}=200mV*(1+R24/R23)/(RS4+RS5)$

比如如图 9.7a 所示的参数， $R36=10K$ ， $R30=3.3K$ ， $RS2=10m\Omega$ ， $RS5=10m\Omega$  时，

$I_{s4\_peak}=200mV*(1+10K/3.3K)/(10m\Omega+10m\Omega)=40A$

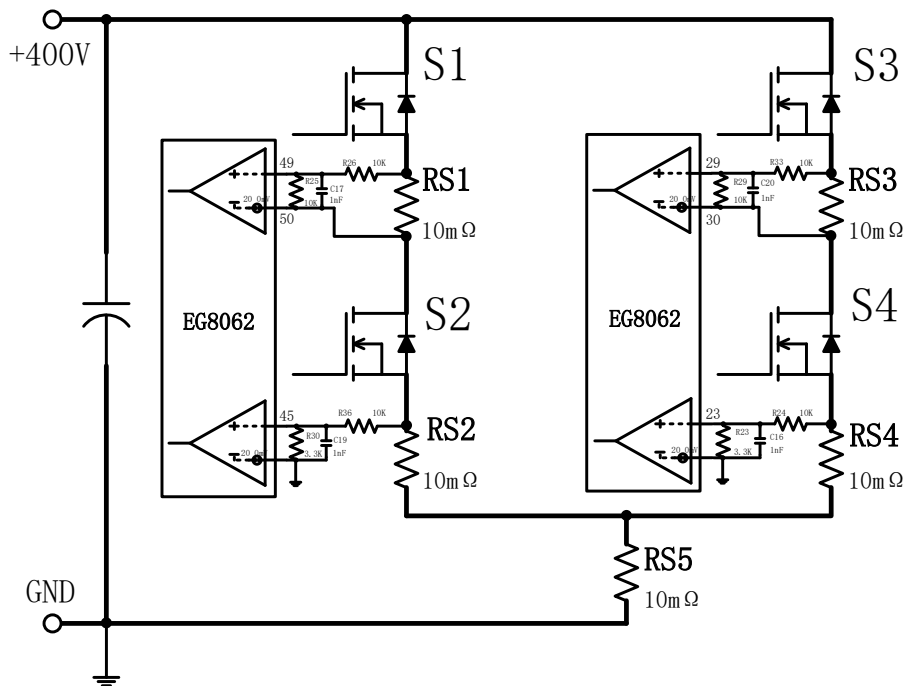


图 9.7a 四路独立的MOS管峰值电流保护电路

## 10. 并机应用

EG8062 支持多机逆变器的应用，主要用于多台发电机并联来获得更高的供应能力，并机应用示意图如图 10.1a 所示，并机时无需区分 L 和 N 线来实现并联，也无需通讯线实现输出波形的同步。

并联时需注意，EG8062 支持在发电机未开启的情况下进行并联，暂不支持在两台发电机开启的情况下进行并联，也不支持任何一台发电机并联到市电上。

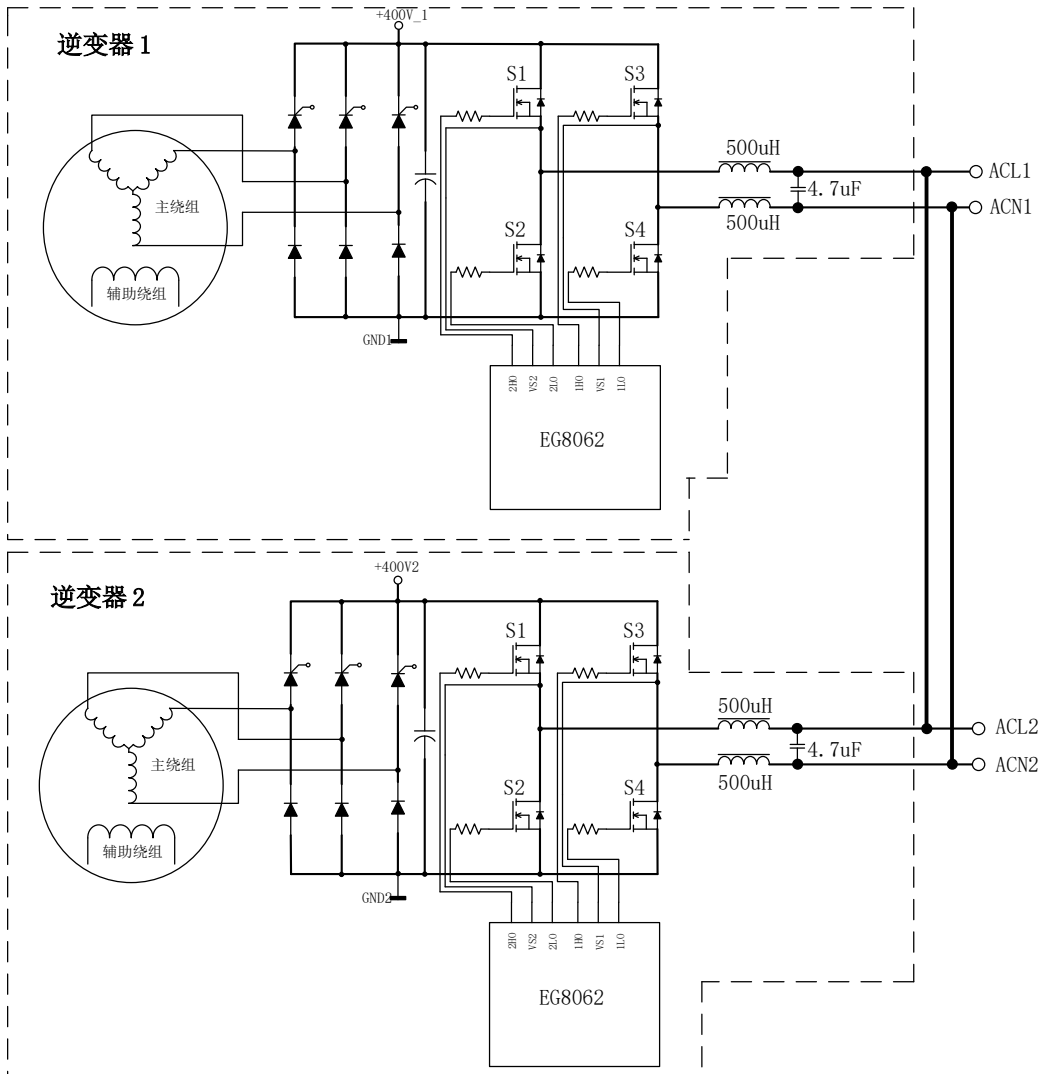


图 10.1a 两台发电机并机应用示意图

# 11. 通讯功能（UART）

## 11.1 串口描述

串口配置：（9600.8.N.1）  
 波特率：9600  
 数据位：8 位  
 校验位：无  
 停止位：1

### 通讯功能：

串口通讯功能分为 APP 功能和 CFG 功能两部分。APP 功能为正常应用功能，包含芯片主动发送状态消息，和接收外部控制命令的功能。CFG 功能为高级配置功能，主要实现芯片的工作模式配置、参数校准等功能。APP 功能通常应用在逆变系统工作时，而 CFG 功能通常应用在逆变系统停机状态下。通过 CFG 功能配置的参数，会存储在芯片内部的 FLASH 空间中，在芯片上电时自动加载。

## 11.2 APP 功能

APP 功能为正常应用功能，包含芯片主动发送状态消息，和接收外部控制命令的功能。APP 功能通常应用在逆变系统工作时，持续向外发送状态消息，并实时接收外部命令，执行相应控制操作。

### 11.2.1 APP 消息发送

芯片上电后，会间隔 100ms 周期持续向外部发送状态消息，长度为 16 个字节。

#### 状态消息：

| 状态消息（100ms 周期发送） |          |   |
|------------------|----------|---|
| BYTE0            | 报头       | 0x55  |
| BYTE1            | 输出电压高字节  | <b>输出电压：</b> 电压数据为 2 个字节表示，最小分辨率是 0.1V。<br>举例：[0x08,0xCF] 用两个 16 进制表示电压，0x08 转换到 10 进制是 8，0xCF 转换到 10 进制是 207，数据的十进制值为 $8*256+207=2255$ ，得出的电压 $V=2255*0.1V=225.5V$ 。   |
| BYTE2            | 输出电压低字节  |   |
| BYTE3            | 输出电流高字节  | <b>输出电流：</b> 电流数据为 2 个字节表示，最小分辨率是 0.01A。<br>举例：[0x02,0xCF] 用两个 16 进制表示电压，0x02 转换到 10 进制是 2，0xCF 转换到 10 进制是 207，数据的十进制值为 $2*256+207=719$ ，得出的电流 $I=719*0.01A=7.19A$ 。    |
| BYTE4            | 输出电流低字节  |   |
| BYTE5            | 输入电压高字节  | <b>输入电压：</b> 电压数据为 2 个字节表示，最小分辨率是 0.1V。<br>举例：[0x0E,0x83] 用两个 16 进制表示电压，0x0E 转换到 10 进制是 14，0x83 转换到 10 进制是 131，数据的十进制值为 $14*256+131=3715$ ，得出的电压 $V=3715*0.1V=371.5V$ 。 |
| BYTE6            | 输入电压低字节  |   |
| BYTE7            | 发动机转速高字节 | <b>发动机转速：</b> 转速数据为 2 个字节表示，最小分辨率是 1RPM。  |

|        |           |   |
|--------|-----------|---|
| BYTE8  | 发动机转速低字节  | 举例：[0x04, 0xCF] 用两个 16 进制表示电压，0x04 转换到 10 进制是 4，0xCF 转换到 10 进制是 207，数据的十进制值为 $4*256+207=1231$ ，得出的转速 $n=1231*1RPM=1231RPM$ 。  |
| BYTE9  | IGBT 温度   | <b>IGBT 温度：</b> 温度数据为 1 个字节，有符号数，最小分辨率是 1℃。<br>举例：[0x16] 转换为 10 进制是 20，温度 =20℃<br>[0xF0] 转换为 10 进制是 -16，温度 =-16℃  |
| BYTE10 | 故障码       | <b>错误代码：</b><br>0x00：未定义<br>0x01：过载保护<br>0x02：短路保护<br>0x03：输出欠压保护<br>0x04：直流母线电压过压保护<br>0x05：直流母线电压过低保护<br>0x06：相位错误<br>0x07：温度保护<br>0x08：过流保护<br>0x09：频率不匹配<br>0x0A：发动机转速太低<br>0x0C：关闭输出 |
| BYTE11 | 环境温度      | <b>逆变器温度：</b> 温度数据为 1 个字节，有符号数，最小分辨率是 1℃。<br>举例：[0x16] 转换为 10 进制是 20，温度 =20℃<br>[0xF0] 转换为 10 进制是 -16，温度 =-16℃  |
| BYTE12 | 输出功率高字节   | <b>输出功率：</b> 功率数据为 2 个字节表示，最小分辨率是 1W。<br>举例：0x06, 0x40 用两个 16 进制表示电压，0x06 转换到 10 进制是 6，0x40 转换到 10 进制是 64，数据的十进制值为 $6*256+64=1600$ ，得出的功率 $P=1600*1W=1600W$ 。                             |
| BYTE13 | 输出功率低字节   |   |
| BYTE14 | CRC 校验高字节 | <b>循环冗余校验是 <math>CRC16=f(X^6+X^5+X^2+1)</math></b><br>对前 14 个字节 BYTE0-BYTE13 执行 CRC16 运算，BYTE14= 校验结果高字节，BYTE15= 校验结果低字节。   |
| BYTE15 | CRC 校验低字节 |   |

### 11.2.2 APP 消息接收

芯片可接收的 APP 消息共有四条。

**逆变关闭：**收到逆变关闭消息后，关闭逆变输出。

**逆变开启：**收到逆变开启消息后，清除故障状态，启动逆变输出。

**ECO 节能开启：**收到 ECO 节能开启消息后，开启 ECO 节能功能，轻载时执行发动机低转速运行。

**ECO 节能关闭：**收到 ECO 节能关闭消息后，关闭 ECO 节能功能，轻载时执行发动机高转速运行。

消息长度同样为 16 字节，超时 50ms 接收，即外部发送数据时，两个字节之间的时间间隔应小于 50ms，如超过 50ms，则判断为当前消息结束，为提高通讯效率，两个字节之间的时间间隔可以尽量小。两组消息之间时间间隔应大于 50ms，为避免接收乱帧，推荐两组消息之间时间间隔大于 100ms。

**逆变关闭消息：**

| 逆变关闭（超时 50ms 接收） |           |  |
|------------------|-----------|--|
| BYTE0            | 命令字段 1    | 0x0F   |
| BYTE1            | 命令字段 2    | 0xF0   |
| BYTE2            | 命令字段 3    | 0x5A   |
| BYTE3            | 命令字段 4    | 0x36   |
| BYTE4            | 保留        | 0x00   |
| BYTE5            | 保留        | 0x00   |
| BYTE6            | 保留        | 0x00   |
| BYTE7            | 保留        | 0x00   |
| BYTE8            | 保留        | 0x00   |
| BYTE9            | 保留        | 0x00   |
| BYTE10           | 保留        | 0x00   |
| BYTE11           | 保留        | 0x00   |
| BYTE12           | 保留        | 0x00   |
| BYTE13           | 保留        | 0x00   |
| BYTE14           | CRC 校验高字节 | 循环冗余校验是 $CRC16=f(X^6+X^{15}+X^2+1)$<br>对前 14 个字节 BYTE0-BYTE13 执行 CRC16 运算，BYTE14= 校验结果高字节，<br>BYTE15= 校验结果低字节。 |
| BYTE15           | CRC 校验低字节 |  |

**逆变开启消息：**

| 逆变开启（超时 50ms 接收） |           |  |
|------------------|-----------|--|
| BYTE0            | 命令字段 1    | 0x7D   |
| BYTE1            | 命令字段 2    | 0xD7   |
| BYTE2            | 命令字段 3    | 0xFE   |
| BYTE3            | 命令字段 4    | 0xDA   |
| BYTE4            | 保留        | 0x00   |
| BYTE5            | 保留        | 0x00   |
| BYTE6            | 保留        | 0x00   |
| BYTE7            | 保留        | 0x00   |
| BYTE8            | 保留        | 0x00   |
| BYTE9            | 保留        | 0x00   |
| BYTE10           | 保留        | 0x00   |
| BYTE11           | 保留        | 0x00   |
| BYTE12           | 保留        | 0x00   |
| BYTE13           | 保留        | 0x00   |
| BYTE14           | CRC 校验高字节 | 循环冗余校验是 $CRC16=f(X^6+X^{15}+X^2+1)$<br>对前 14 个字节 BYTE0-BYTE13 执行 CRC16 运算，BYTE14= 校验结果高字节，<br>BYTE15= 校验结果低字节。 |
| BYTE15           | CRC 校验低字节 |  |

**ECO 节能开启消息：**

| ECO 节能开启（超时 50ms 接收） |           |  |
|----------------------|-----------|--|
| BYTE0                | 命令字段 1    | 0x2A   |
| BYTE1                | 命令字段 2    | 0xA2   |
| BYTE2                | 命令字段 3    | 0x00   |
| BYTE3                | 保留        | 0x00   |
| BYTE4                | 保留        | 0x00   |
| BYTE5                | 保留        | 0x00   |
| BYTE6                | 保留        | 0x00   |
| BYTE7                | 保留        | 0x00   |
| BYTE8                | 保留        | 0x00   |
| BYTE9                | 保留        | 0x00   |
| BYTE10               | 保留        | 0x00   |
| BYTE11               | 保留        | 0x00   |
| BYTE12               | 保留        | 0x00   |
| BYTE13               | 保留        | 0x00   |
| BYTE14               | CRC 校验高字节 | 循环冗余校验是 $CRC16=f(X^6+X^{16}+X^2+1)$<br>对前 14 个字节 BYTE0-BYTE13 执行 CRC16 运算，BYTE14= 校验结果高字节，<br>BYTE15= 校验结果低字节。 |
| BYTE15               | CRC 校验低字节 |  |

**ECO 节能关闭消息：**

| ECO 节能关闭（超时 50ms 接收） |           |  |
|----------------------|-----------|--|
| BYTE0                | 命令字段 1    | 0x2A   |
| BYTE1                | 命令字段 2    | 0xA2   |
| BYTE2                | 命令字段 3    | 0x01   |
| BYTE3                | 保留        | 0x00   |
| BYTE4                | 保留        | 0x00   |
| BYTE5                | 保留        | 0x00   |
| BYTE6                | 保留        | 0x00   |
| BYTE7                | 保留        | 0x00   |
| BYTE8                | 保留        | 0x00   |
| BYTE9                | 保留        | 0x00   |
| BYTE10               | 保留        | 0x00   |
| BYTE11               | 保留        | 0x00   |
| BYTE12               | 保留        | 0x00   |
| BYTE13               | 保留        | 0x00   |
| BYTE14               | CRC 校验高字节 | 循环冗余校验是 $CRC16=f(X^6+X^{16}+X^2+1)$<br>对前 14 个字节 BYTE0-BYTE13 执行 CRC16 运算，BYTE14= 校验结果高字节，<br>BYTE15= 校验结果低字节。 |
| BYTE15               | CRC 校验低字节 |  |

### 11.3 CFG功能

**CFG 功能**为高级配置功能，主要实现芯片的工作模式配置、参数校准等功能。CFG 功能通常应用在逆变系统停机状态下。通过 CFG 功能配置的参数，会存储在芯片内部的 FLASH 空间中，在芯片上电时自动加载。

CFG 功能需要外部发送请求消息，芯片响应请求服务并回复应答消息。

发送和接收均采用 16 字节固定长度，消息以 ASCII 码'E'、'G'开头，CRC16 结尾。为区分 APP 消息和 CFG 消息，CRC 校验结果稍有不同，APP 消息的 CRC 校验结果= $f(X^{16}+X^{15}+X^2+1)$ 。而 CFG 消息的 CRC 校验结果相当于在 APP 校验基础上加 1，即 CFG 消息的校验结果= $f(X^{16}+X^{15}+X^2+1)+1$ 。

#### 11.3.1 CFG 请求消息

**CFG 请求消息格式：**

| CFG 请求消息（超时 50ms 接收） |                    |  |
|----------------------|--------------------|--|
| BYTE0                | 报头 1               | 0x45 - 'E'   |
| BYTE1                | 报头 2               | 0x47 - 'G'   |
| BYTE2                | 服务编码(SID)          | 主机请求的服务内容  |
| BYTE3                | 子功能(sfun)/地址(addr) | 当前服务下的子功能或地址   |
| BYTE4                | 请求数据 1             |  |
| BYTE5                | 请求数据 2             |  |
| BYTE6                | 请求数据 3             |  |
| BYTE7                | 请求数据 4             |  |
| BYTE8                | 请求数据 5             |  |
| BYTE9                | 请求数据 6             |  |
| BYTE10               | 请求数据 7             |  |
| BYTE11               | 请求数据 8             |  |
| BYTE12               | 请求数据 9             |  |
| BYTE13               | 请求数据 10            |  |
| BYTE14               | CRC 校验高字节          | <b>循环冗余校验是 <math>CRC16=f(X^6+X^{15}+X^2+1)+1</math></b><br>对前 14 个字节 BYTE0-BYTE13 执行 CRC16 运算，BYTE14= 校验结果高字节， |
| BYTE15               | CRC 校验低字节          | BYTE15= 校验结果低字节。   |

#### 11.3.2 CFG 应答消息

**CFG 应答消息格式：**

| CFG 请求消息（超时 50ms 接收） |                    |                |
|----------------------|--------------------|----------------|
| BYTE0                | 报头 1               | 0x45 - 'E'     |
| BYTE1                | 报头 2               | 0x47 - 'G'     |
| BYTE2                | 服务编码(SID)          | 主机请求的 CFG 服务编码 |
| BYTE3                | 子功能(sfun)/地址(addr) | 当前服务下的子功能或地址   |
| BYTE4                | 应答数据 1             |                |

|        |           |  |
|--------|-----------|--|
| BYTE5  | 应答数据 2    |  |
| BYTE6  | 应答数据 3    |  |
| BYTE7  | 应答数据 4    |  |
| BYTE8  | 应答数据 5    |  |
| BYTE9  | 应答数据 6    |  |
| BYTE10 | 应答数据 7    |  |
| BYTE11 | 应答数据 8    |  |
| BYTE12 | 应答数据 9    |  |
| BYTE13 | 应答数据 10   |  |
| BYTE14 | CRC 校验高字节 | <p>循环冗余校验是 <math>CRC16=f(X^6+X^{15}+X^2+1)+1</math></p> <p>对前 14 个字节 BYTE0-BYTE13 执行 CRC16 运算，BYTE14= 校验结果高字节，BYTE15= 校验结果低字节。</p> |
| BYTE15 | CRC 校验低字节 |  |

### 11.3.3 0x10 服务-会话切换

0x10 服务是切换会话服务，通信会话主要分为默认会话（ 01）、扩展会话（ 03）和编程会话（02）， 其中编程会话暂不对用户开放。

主机 0x10 请求消息：

| Byte0 | Byte1 | Byte2 | Byte3   | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
|-------|-------|-------|---------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| 0x45  | 0x47  | 0x10  | Session | 0x00  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00   | 0x00   | 0x00   | 0x00   |        | CRC16  |

addr 是 DID 的地址，不同的地址存储不同的 DID 信息。

从机回复：

| Byte0 | Byte1 | Byte2 | Byte3   | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
|-------|-------|-------|---------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| 0x45  | 0x47  | 0x22  | Session | 0x00  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00   | 0x00   | 0x00   | 0x00   |        | CRC16  |

若从机回复 Byte4=0xCC，表示切换会话成功，回复 Byte4=0xEE，表示切换会话失败。

#### 默认会话（Session = 01）：

默认会话下，从机周期性发送 APP 报文。通信默认在默认会话。

#### 扩展会话（Session = 03）：

扩展会话下，从机停止周期性发送 APP 报文。通常在需要执行 DID 或 CFG 读写操作时，为避免 APP 报文占用通讯资源，切换至扩展会话；操作完成后，再切换回默认会话。

#### 编程会话（Session = 02）：

程序刷写返回 bootloader 时使用，暂不对用户开放。

### 11.3.4 0x22 服务-读 DID

0x22 服务是读 DID 服务，系统的配置参数、版本信息等都存储在 DID 中，通过请求 0x22 服务，主机可以读取芯片的配置参数和版本信息灯内容。

主机 0x22 请求消息:

|       |       |       |       |       |       |       |       |       |       |        |        |        |        |        |        |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
| 0x45  | 0x47  | 0x22  | addr  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00   | 0x00   | 0x00   | 0x00   |        | CRC16  |

addr 是 DID 的地址，不同的地址存储不同的 DID 信息。

从机回复:

|       |       |       |       |       |       |       |       |       |       |        |        |        |        |        |        |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
| 0x45  | 0x47  | 0x22  | addr  | d1    | d2    | d3    | d4    | d5    | d6    | d7     | d8     | d9     | d10    |        | CRC16  |

若从机回复 d1~d10 全为 0xFF，那么表示读取 DID 失败。

### 11.3.5 0x2E 服务-写 DID

0x2E 服务是写 DID 服务，通过请求 0x2E 服务，主机可以将配置参数和版本信息等内容写入芯片。

主机 0x2E 请求消息:

|       |       |       |       |       |       |       |       |       |       |        |        |        |        |        |        |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
| 0x45  | 0x47  | 0x2E  | addr  | d1    | d2    | d3    | d4    | d5    | d6    | d7     | d8     | d9     | d10    |        | CRC16  |

addr 是 DID 的地址，不同的地址存储不同的 DID 信息。

从机回复:

|       |       |       |       |       |       |       |       |       |       |        |        |        |        |        |        |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
| 0x45  | 0x47  | 0x2E  | addr  | resp  |       |       |       |       |       |        |        |        |        |        | CRC16  |

resp = 1 : 写入成功

resp = 0 : 写入失败

DID 信息表:

| DID 信息表 |        |     |     |       |  |
|---------|--------|-----|-----|-------|--|
| ADDR    | DID    | W/R | LEN | TYPE  | 描述   |
| 0x03    | EngCfg | w/r | 1   | hex   | 此 DID 为发动机参数组选择:<br>d1 = EngCfg<br>EngCfg = 0 自定义<br>EngCfg = 1 参数一<br>EngCfg = 2 参数二<br>... |
| 0x04    | InvCfg | w/r | 1   | hex   | 此 DID 为逆变参数组选择:<br>d1 = InvCfg<br>InvCfg = 0 自定义<br>InvCfg = 1 参数一<br>InvCfg = 2 参数二<br>...  |
| 0x05    | VinAdj | w/r | 4   | float | 此 DID 为母线电压校准因子:<br>d1 = (VinAdj >>24) & 0xFF<br>d2 = (VinAdj >>16) & 0xFF                   |

|      |             |     |    |       |  |
|------|-------------|-----|----|-------|--|
|      |             |     |    |       | $d3 = (VinAdj \gg 8) \& 0xFF$<br>$d4 = VinAdj \& 0xFF$   |
| 0x06 | VoutAdj     | w/r | 4  | float | 此 DID 为右桥臂电压校准因子:<br>$d1 = (VoutAdj \gg 24) \& 0xFF$<br>$d2 = (VoutAdj \gg 16) \& 0xFF$<br>$d3 = (VoutAdj \gg 8) \& 0xFF$<br>$d4 = VoutAdj \& 0xFF$  |
| 0x07 | IoutAdj1    | w/r | 4  | float | 此 DID 为左桥臂电流校准因子:<br>$d1 = (IoutAdj1 \gg 24) \& 0xFF$<br>$d2 = (IoutAdj1 \gg 16) \& 0xFF$<br>$d3 = (IoutAdj1 \gg 8) \& 0xFF$<br>$d4 = IoutAdj1 \& 0xFF$  |
| 0x08 | IoutAdj2    | w/r | 4  | float | 此 DID 为右桥臂电流校准因子:<br>$d1 = (IoutAdj2 \gg 24) \& 0xFF$<br>$d2 = (IoutAdj2 \gg 16) \& 0xFF$<br>$d3 = (IoutAdj2 \gg 8) \& 0xFF$<br>$d4 = IoutAdj2 \& 0xFF$  |
| 0x09 | ProductDate | w/r | 4  | BCD   | 此 DID 可以写入整机生产日期, 如 2019 年 10 月 15 日:<br>$d1 = 0x20$<br>$d2 = 0x19$<br>$d3 = 0x10$<br>$d4 = 0x15$  |
| 0x0A | SerialNo    | w/r | 10 | ASCII | 此 DID 可以写入整机序列号, 如 HS 厂生产的 2019 年第 21 周第 0001 号产品: "HS19210001"<br>$d1 = 'H'$<br>$d2 = 'S'$<br>$d3 = '1'$<br>$d4 = '9'$<br>$d5 = '2'$<br>$d6 = '1'$<br>$d7 = '0'$<br>$d8 = '0'$<br>$d9 = '0'$<br>$d10 = '1'$ |
| 0x0B | PartNo      | r   | 10 | ASCII | 此 DID 表示芯片型号:<br>"EG8060 "   |
| 0x0C | ChipID      | r   | 10 | hex   | 此 DID 包含芯片 ID, 每颗芯片唯一:<br>如: 0x0123456789ABCDEF0123<br>$d1 = 0x01$<br>$d2 = 0x23$<br>$d3 = 0x45$<br>$d4 = 0x67$<br>$d5 = 0x89$   |

|      |             |   |    |       |  |
|------|-------------|---|----|-------|--|
|      |             |   |    |       | d6 = 0xAB<br>d7 = 0xCD<br>d8 = 0xEF<br>d9 = 0x01<br>d10 = 0x23 |
| 0x0D | UsartVer    | r | 10 | ASCII | 串口通讯协议版本号，如 V1.0 版本 19 年 10 月 15 日发行：<br>“1.0.191015”          |
| 0x0E | SoftwareVer | r | 10 | ASCII | 芯片固件版本号，如 V2.1 版本 19 年 10 月 15 日发行：<br>“2.1.191015”            |
| 0x0F | HardwareVer | r | 10 | ASCII | 芯片硬件版本号，如 V3.2 版本 19 年 10 月 15 日发行：<br>“3.2.191015”            |

### 11.3.6 0x21 服务-读 CFG

0x21 服务是读 CFG 服务，通过请求 0x21 服务，主机可以读取芯片的配置，包括输出电压、频率、保护参数等。

主机 0x2F 请求消息：

| Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| 0x45  | 0x47  | 0x21  | addr  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00   | 0x00   | 0x00   | 0x00   |        | CRC16  |

addr 是 CFG 寄存器地址。

从机回复：

| Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| 0x45  | 0x47  | 0x21  | addr  | d1    | d2    | d3    | d4    | d5    | d6    | d7     | d8     | d9     | d10    |        | CRC16  |

resp = addr：请求成功，d1-d10 是读出的寄存器值。

resp = 0xFF：请求失败

### 11.3.7 0x2D 服务-写 CFG

0x2D 服务是写 CFG 服务，通过请求 0x2D 服务，主机可以写入芯片的配置，包括输出电压、频率、保护参数等。

主机 0x2F 请求消息：

| Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| 0x45  | 0x47  | 0x2F  | sfun  | d1    | d2    | d3    | d4    | d5    | d6    | d7     | d8     | d9     | d10    |        | CRC16  |

addr 是 CFG 寄存器地址，d1-d10 是写入的寄存器值。

从机回复：

| Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| 0x45  | 0x47  | 0x2F  | addr  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00   | 0x00   | 0x00   | 0x00   |        | CRC16  |

resp = addr：请求成功

resp = 0xFF：请求失败

CFG 寄存器表

| CFG 寄存器表(逆变参数) |                              |     |     |       |  |
|----------------|------------------------------|-----|-----|-------|--|
| ADDR           | CFG                          | W/R | LEN | TYPE  | 描述   |
| 0x00           | (InvCfg_Save)                | w   | 0   | (cmd) | 逆变器参数保存, 包含 0x01-0x18 所有的设定参数, 此操作将触发写存储器, 内核会暂停运行, 逆变输出关闭, 发动机进入怠速模式, 写入后需要重新启动。  |
| 0x01           | Vset                         | w/r | 2   | Hex   | 输出电压设定值, Vset=2200 表示设定输出电压 220.0V:<br>d1 = (Vset>>8) & 0xFF<br>d2 = Vset & 0xFF   |
| 0x02           | Fset                         | w/r | 2   | Hex   | 输出频率设定值, Fset=5000 表示设定输出电压 50.00Hz:<br>d1 = (Fset>>8) & 0xFF<br>d2 = Fset & 0xFF  |
| 0x03           | Pset                         | w/r | 2   | Hex   | 最大输出功率设定值, Pset=1000 表示设定最大输出功率 1000W:<br>d1 = (Pset>>8) & 0xFF<br>d2 = Pset & 0xFF  |
| 0x10           | prt_vdch<br>prt_vdch_delay   | w/r | 4   | Hex   | 输入过压保护设定, prt_vdch=4400, prt_vdch_delay=5 表示持续 5 秒输入电压大于 440V 判定为输入过压:<br>d1 = (prt_vdch >>8) & 0xFF<br>d2 = prt_vdch & 0xFF<br>d3 = (prt_vdch_delay >>8) & 0xFF<br>d4 = prt_vdch_delay & 0xFF         |
| 0x11           | prt_vdcl<br>prt_vdcl_delay   | w/r | 4   | Hex   | 输入欠压保护设定, prt_vdcl=2500, prt_vdcl_delay=10 表示持续 10 秒输入电压小于 250V 判定为输入欠压:<br>d1 = (prt_vdcl >>8) & 0xFF<br>d2 = prt_vdcl & 0xFF<br>d3 = (prt_vdcl_delay >>8) & 0xFF<br>d4 = prt_vdcl_delay & 0xFF       |
| 0x12           | prt_vouth<br>prt_vouth_delay | w/r | 4   | Hex   | 输出过压保护设定, prt_vouth=2400, prt_vouth_delay=5 表示持续 5 秒输出电压大于 240V 判定为输出过压:<br>d1 = (prt_vouth >>8) & 0xFF<br>d2 = prt_vouth & 0xFF<br>d3 = (prt_vouth_delay >>8) & 0xFF<br>d4 = prt_vouth_delay & 0xFF   |
| 0x13           | prt_voutl<br>prt_voutl_delay | w/r | 4   | Hex   | 输出欠压保护设定, prt_voutl=1800, prt_voutl_delay=10 表示持续 10 秒输出电压小于 180V 判定为输出欠压:<br>d1 = (prt_voutl >>8) & 0xFF<br>d2 = prt_voutl & 0xFF<br>d3 = (prt_voutl_delay >>8) & 0xFF<br>d4 = prt_voutl_delay & 0xFF |
| 0x14           | prt_iout1                    | w/r | 6   | Hex   | 第一级过载保护设定, prt_iout1=500, prt_pout1=1100,  |

|               |   |     |   |     |   |
|---------------|---|-----|---|-----|---|
|               | prt_pout1<br>prt_pout1_delay                        |     |   |     | prt_pout1_delay=60 表示持续 60 秒输出电流大于 5A 或输出功率大于 1100W 判定为过载：<br>d1 = (prt_iout1 >>8) & 0xFF<br>d2 = prt_iout1 & 0xFF<br>d3 = (prt_pout1 >>8) & 0xFF<br>d4 = prt_pout1 & 0xFF<br>d5 = (prt_pout1_delay >>8) & 0xFF<br>d6 = prt_pout1_delay & 0xFF  |
| 0x15          | prt_iout2<br>prt_pout2<br>prt_pout2_step            | w/r | 6 | Hex | 第二级过载保护设定，prt_iout2=545，prt_pout2=1200，prt_pout2_step=6 表示持续 (prt_pout1_delay/prt_pout2_step)=10 秒输出电流大于 5.45A 或输出功率大于 1200W 判定为过载：<br>d1 = (prt_iout2 >>8) & 0xFF<br>d2 = prt_iout2 & 0xFF<br>d3 = (prt_pout2 >>8) & 0xFF<br>d4 = prt_pout2 & 0xFF<br>d5 = (prt_pout2_step >>8) & 0xFF<br>d6 = prt_pout2_step & 0xFF |
| 0x16<br>(不可用) | prt_iout3<br>prt_pout3<br>prt_pout3_step            | w/r | 6 | Hex | 第二级过载保护设定，prt_iout3=545，prt_pout3=1200，prt_pout3_step=60 表示持续 (prt_pout1_delay/prt_pout3_step)=1 秒输出电流大于 5.90A 或输出功率大于 1300W 判定为过载：<br>d1 = (prt_iout3 >>8) & 0xFF<br>d2 = prt_iout3 & 0xFF<br>d3 = (prt_pout3 >>8) & 0xFF<br>d4 = prt_pout3 & 0xFF<br>d5 = (prt_pout3_step >>8) & 0xFF<br>d6 = prt_pout3_step & 0xFF |
| 0x17          | prt_temp_board<br>prt_temp_igbt<br>prt_temp_delay   | w/r | 6 | Hex | 温度保护设定，prt_temp_board =85，prt_temp_igbt=130，prt_temp_delay=10 表示持续 10 秒板子温度大于 85℃ 或 IGBT 温度大于 130℃ 判定为系统过热：<br>d1 = (prt_temp_board >>8) & 0xFF<br>d2 = prt_temp_board & 0xFF<br>d3 = (prt_temp_igbt >>8) & 0xFF<br>d4 = prt_temp_igbt & 0xFF<br>d5 = (prt_temp_delay >>8) & 0xFF<br>d6 = prt_temp_delay & 0xFF       |
| 0x18          | prc_short_vol<br>prc_short_delay<br>prt_short_reset | w/r | 6 | Hex | 温度保护设定，prc_short_vol=200，prc_short_delay =150，prt_short_reset=1000 表示持续 150 个 PWM 周期输出电压小于 20V 判定为输出短路，当母线电压小于 100V 时自动清除输出短路保护：<br>d1 = (prt_temp_board >>8) & 0xFF<br>d2 = prt_temp_board & 0xFF  |

|  |  |  |  |  |   |
|--|--|--|--|--|---|
|  |  |  |  |  | <p>d3 = (prt_temp_igbt &gt;&gt;8) &amp; 0xFF</p> <p>d4 = prt_temp_igbt &amp; 0xFF</p> <p>d5 = (prt_temp_delay &gt;&gt;8) &amp; 0xFF</p> <p>d6 = prt_temp_delay &amp; 0xFF</p> |
|--|--|--|--|--|---|

| CFG 寄存器表(发动机参数) |   |     |     |       |  |
|-----------------|---|-----|-----|-------|--|
| ADDR            | CFG   | W/R | LEN | TYPE  | 描述   |
| 0x80            | (EngCfg_Save)                               | w   | 0   | (cmd) | 逆变器参数保存，包含 0x81-0x83 所有的设定参数，此操作将触发写存储器，内核会暂停运行，逆变输出关闭，发动机进入怠速模式，写入后需要重新启动。  |
| 0x81            | Kp<br>Ki<br>circle<br>step_max<br>step_max1 | w/r | 10  | Hex   | <p>发动机油门控制设定，</p> <p>Kp、Ki 为控制环比例系数和积分系数</p> <p>d1 = (Kp &gt;&gt;8) &amp; 0xFF</p> <p>d2 = Kp &amp; 0xFF</p> <p>d3 = (Ki &gt;&gt;8) &amp; 0xFF</p> <p>d4 = Ki &amp; 0xFF</p> <p>Circle 为发动机控制周期，Circle = 100 标识 100ms 执行一次控制换运算</p> <p>d5 = (circle &gt;&gt;8) &amp; 0xFF</p> <p>d6 = circle &amp; 0xFF</p> <p>step_max 为运行中油门单次最大步进，step_max=30 表示单次环路运算，油门最多只动作 30 步：</p> <p>d7 = (step_max &gt;&gt;8) &amp; 0xFF</p> <p>d8 = step_max &amp; 0xFF</p> <p>step_max1 为启动时油门单次最大步进，step_max1=30 表示发动机启动时单次环路运算，油门最多只动作 30 步：</p> <p>d9 = (step_max1 &gt;&gt;8) &amp; 0xFF</p> <p>d10 = step_max1 &amp; 0xFF</p> |
| 0x82            | rpm_min<br>rpm_eco<br>rpm_max<br>rpm_istep  | w/r | 8   | Hex   | <p>转速-电流曲线设定：</p> <p>rpm_min 表示待机转速</p> <p>d1 = (rpm_min &gt;&gt;8) &amp; 0xFF</p> <p>d2 = rpm_min &amp; 0xFF</p> <p>rpm_eco 表示节能模式关闭时的待机转速</p> <p>d3 = (rpm_eco &gt;&gt;8) &amp; 0xFF</p> <p>d4 = rpm_eco &amp; 0xFF</p> <p>rpm_max 表示最高转速</p> <p>d5 = (rpm_min &gt;&gt;8) &amp; 0xFF</p>   |

|      |                     |     |   |     |  |
|------|---------------------|-----|---|-----|--|
|      |                     |     |   |     | $d6 = rpm\_min \& 0xFF$<br><br>$rpm\_istep$ 表示转速电流比,<br>发动机转速设定 = $rpm\_min + iac * rpm\_istep$<br>$d7 = (rpm\_istep \gg 8) \& 0xFF$<br>$d8 = rpm\_istep \& 0xFF$  |
| 0x83 | rpm_filter<br>polar | w/r | 4 | Hex | 转速检测设定:<br>$rpm\_filter$ 为转速滤波, $rpm\_filter=10$ 表示 10 次求平均<br>$d1 = (rpm\_filter \gg 8) \& 0xFF$<br>$d2 = rpm\_filter \& 0xFF$<br><br>$polar$ 为极对数, 转速( $rpm$ )=辅助绕组频率*60/极对数<br>$d3 = (polar \gg 8) \& 0xFF$<br>$d4 = polar \& 0xFF$ |

### 11.3.8 0x2F 服务-IO 控制

0x2F 服务是 IO 控制服务, 通过请求 0x2F 服务, 主机可以控制芯片工作在特殊的状态, 如测试模式等。

主机 0x2F 请求消息:

|       |       |       |       |       |       |       |       |       |       |        |        |        |        |        |        |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
| 0x45  | 0x47  | 0x2F  | sfun  | ctl   | d2    | d3    | d4    | d5    | d6    | d7     | d8     | d9     | d10    | CRC16  |        |

sfun 是不同的 IO 服务子功能, ctl 是当前子功能下的控制字。

从机回复:

|       |       |       |       |       |       |       |       |       |       |        |        |        |        |        |        |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
| 0x45  | 0x47  | 0x2F  | sfun  | resp  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00   | 0x00   | 0x00   | 0x00   | CRC16  |        |

resp = ctl : 请求成功

resp = 0xFF : 请求失败

IO 控制功能表:

| IO 控制功能表 |      |                |
|----------|------|----------------|
| SFUN     | CTL  | 描述             |
| 0x02     | 0x01 | 油门步进电机正转一周     |
|          | 0x02 | 油门步进电机反转一周     |
| 0x02     | 0x00 | 逆变电路闭环输出       |
|          | 0x03 | 逆变电路开环 SPWM 输出 |

|      |      |           |
|------|------|-----------|
| 0x03 | 0x01 | 整流电路可控硅关闭 |
|      | 0x02 | 整流电路可控硅开启 |

### 11.3.9 0x80 服务-校准

0x80 服务是输出校准服务，通过请求 0x80 服务，用户能够将芯片读到的反馈值校准到实际值。这样可以避免因外部反馈电路差异造成的反馈信号增益不匹配。主要校准的内容有三条：输出电压反馈校准，左桥臂电流反馈校准，右桥臂电流反馈校准。校准服务下，从机无应答信息。

输出电压反馈校准：

将外部校准仪表测量的输出电压通过 80 服务 01 子功能传输给芯片，芯片根据接收到的电压值重新换算电压反馈增益，使反馈电压（芯片 ADC 读取）等于输出电压（仪表检测）。

| Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| 0x45  | 0x47  | 0x80  | 0x01  | vouth | voutl | 0x00  | 0x00  | 0x00  | 0x00  | 0x00   | 0x00   | 0x00   | 0x00   | 0x00   | CRC16  |

vouth 为输出电压高字节，voutl 为输出电压低字节，两个字节组合成 16 位无符号数的电压值，分辨率为 0.1V。

如校准仪表上显示的输出电压值为 230V，那么需要写入的输出电压=230 / 0.1 =2300，转换为十六进制为 0x08FC。得到 vouth = 0x08，voutl = 0xFC。

左桥臂电流反馈校准：

主机请求 80 服务 02 子功能时，芯片会将输出切换到直流，且左桥臂输出正电压。此时通过接入负载，通常校准点选在额定电流的一半，将校准仪表的电流值传输给芯片。所以电流校准时通常要请求两次 80 服务 02 子功能，第一次用于切换输出为直流（iouth,ioutl 可以写入额定电流的一般），接通负载，第二次将仪表电流写入（iouth,ioutl）传输给芯片。

| Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| 0x45  | 0x47  | 0x80  | 0x02  | iouth | ioutl | 0x00  | 0x00  | 0x00  | 0x00  | 0x00   | 0x00   | 0x00   | 0x00   | 0x00   | CRC16  |

iouth 为输出电压高字节，ioutl 为输出电压低字节，两个字节组合成 16 位无符号数的电流值，分辨率为 0.01A。

如校准仪表上显示的输出电压值为 10A，那么需要写入的输出电流=10/ 0.01 =1000，转换为十六进制为 0x03E8。得到 iouth = 0x03，ioutl = 0xE8。

右桥臂电流反馈校准：

主机请求 80 服务 03 子功能时，芯片会将输出切换到直流，且右桥臂输出正电压。此时通过接入负载，通常校准点选在额定电流的一半，将校准仪表的电流值传输给芯片。所以电流校准时通常要请求两次 80 服务 03 子功能，第一次用于切换输出为直流（iouth,ioutl 可以写入额定电流的一般），接通负载，第二次将仪表电流写入（iouth,ioutl）传输给芯片。

| Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| 0x45  | 0x47  | 0x80  | 0x03  | iouth | ioutl | 0x00  | 0x00  | 0x00  | 0x00  | 0x00   | 0x00   | 0x00   | 0x00   | 0x00   | CRC16  |

iouth 为输出电压高字节，ioutl 为输出电压低字节，两个字节组合成 16 位无符号数的电流值，分辨率为 0.01A。

如校准仪表上显示的输出电压值为 10A，那么需要写入的输出电流= $10 / 0.01 = 1000$ ，转换为十六进制为 0x03E8。得到 iouth = 0x03， ioutl = 0xE8。

保存校准值：

主机请求 80 服务 00 子功能时，芯片会将之前校准的输出电压、左桥臂电流、右桥臂电流增益写入芯片 DID 存储区，并在每次上电时加载。在写存储区的过程中，芯片会关闭输出，待写入完成后，重新开启输出，且工作在闭环模式。

| Byte0 | Byte1 | Byte2 | Byte3 | Byte4 | Byte5 | Byte6 | Byte7 | Byte8 | Byte9 | Byte10 | Byte11 | Byte12 | Byte13 | Byte14 | Byte15 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| 0x45  | 0x47  | 0x80  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00  | 0x00   | 0x00   | 0x00   | 0x00   | CRC16  |        |

# 12. 封装尺寸

## 12.1 LQFP64

