高速吹风筒应用手册

三相电机控制 MCU
FU6862L

峰岹科技(深圳)股份有限公司

目录

目录 **2**
1 概述 **4**
2 硬件 **5**
2.1 硬件原理图 5
2.1.1 芯片电路 6
2.1.2 电源电路 6
2.1.3 功率驱动电路 6
2.1.4 过零检测电路 7
2.1.5 运放配置电路 8
2.1.6 母线电压采样电路 8
3 软件架构 **9**
3.1 电机状态机流程图 9
3.2 程序流程图 11
3.3 程序说明 11
3.3.1 Main 函数: 11
3.3.2 1ms 定时中断 11
3.3.3 FOC 中断 11
3.3.4 CMP3 中断 12
3.3.5 外部中断 12
3.3.6 Timer3 中断 12
4 调试步骤 **13**
4.1 电机参数配置 13
4.1.1 电机参数 13
4.1.2 电机参数测量方法 13
4.1.3 对应程序 14
4.2 芯片内部参数配置 14
4.3 硬件参数配置 15
4.4 保护参数配置 16
4.5 启动参数配置 16
4.6 硬件驱动电路检测 18
4.7 电流环调试 18
4.8 速度环调试 19
4.9 增加按键功能 20

4.10 增加发热功能 20
4.11 可靠性测试 21
4.11.1 功能可靠性 21
4.11.2 保护可靠性 21
4.11.3 启动稳定性 21
5 功能介绍 22
5.1 启动调试 22
5.1.1 Omega 启动 22
5.1.2 启动常见问题&解决方式 23
5.2 保护介绍 23
5.2.1 过流保护 23
5.2.2 电压保护 24
5.2.3 缺相保护 24
5.2.4 启动保护 25
5.2.5 堵转保护 26
5.2.6 过温保护 26
5.2.7 功率保护 27
5.2.8 其他保护 27
6 其他常见功能调试 27
6.1 限功率功能 27
7 方案调试难点&解决方法 28
8 版权说明 29

1 概述

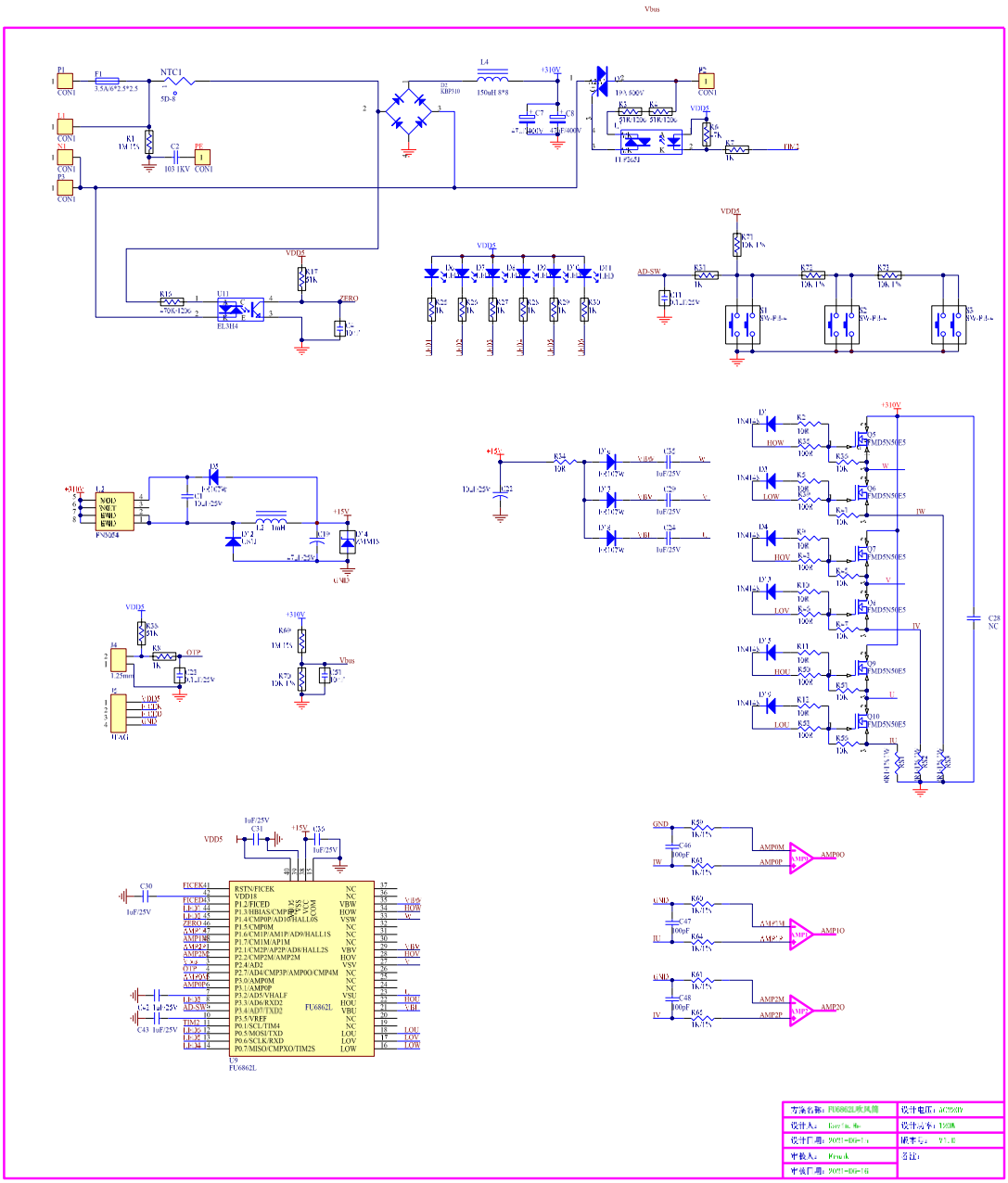
本调试手册详细介绍了如何使用峰岹科技的 FU6862L 芯片，在高速吹风筒专用 DEMO 板子上，对直流无刷高速吹风筒电机进行无感 FOC 驱动控制。阅读手册时，第二章节硬件原理跟第三章节软件原理可以大致先浏览一遍，重点放在第四章调试步骤。

涉及的软/硬件

软/硬件和模块	名称	章节	备注
软件	FU-AM-FU6862-B-059-SW-V1.0.00-20221212	全部	调试需在该工程软件上进行
硬件	FU-AM-FU6862-B-059-HW-V1.0.00-20221212	全部	调试需在该硬件上进行

2 硬件

2.1 硬件原理图



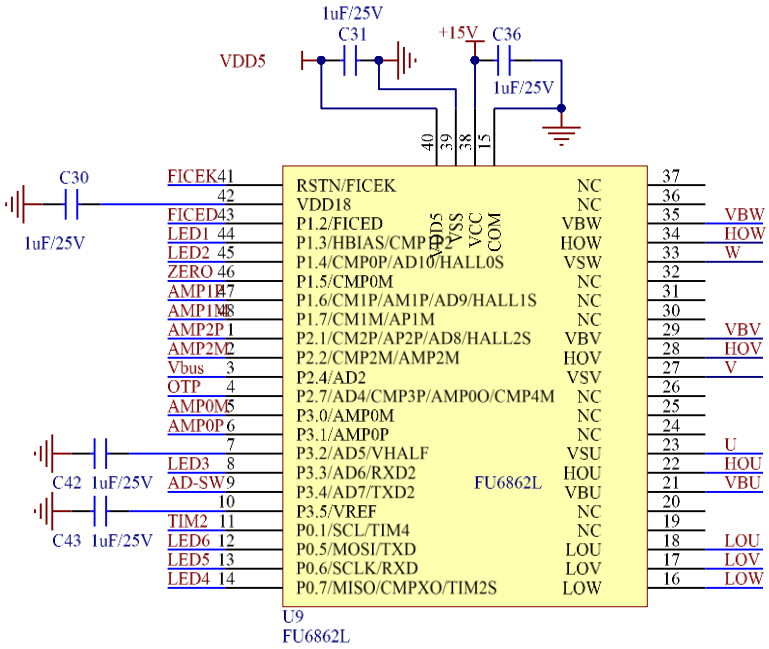
使用方式:

该板子为高速吹风筒应用 FU6862L 三电阻方案的专用 DEMO 板子，直接上电即可使用。

注意事项:

根据具体电机电压和电流大小，合理配置母线电压比，运放放大倍数，采样电阻。

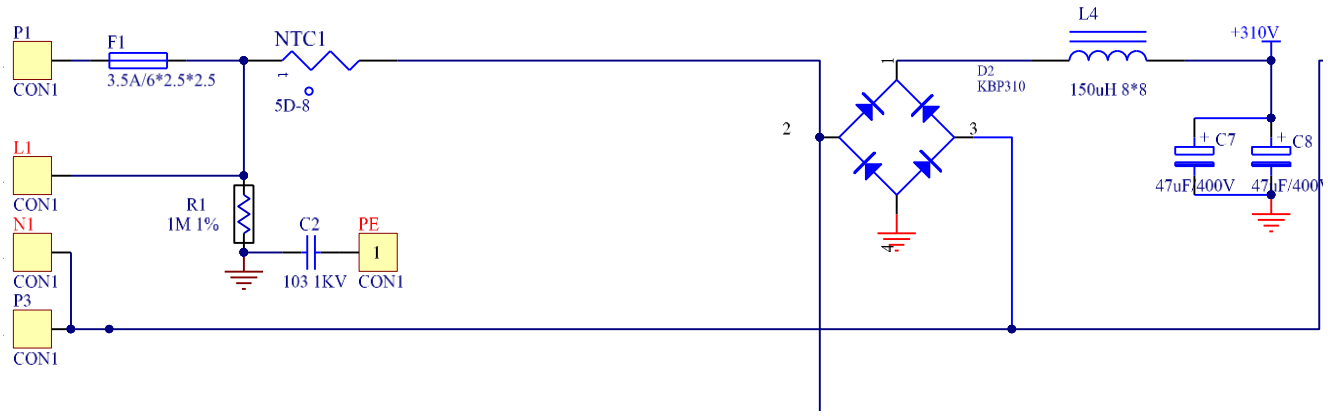
2.1.1 芯片电路



使用方式:

FU6862L 应用于交流高压 6-NMOSFET 驱动应用。

2.1.2 电源电路



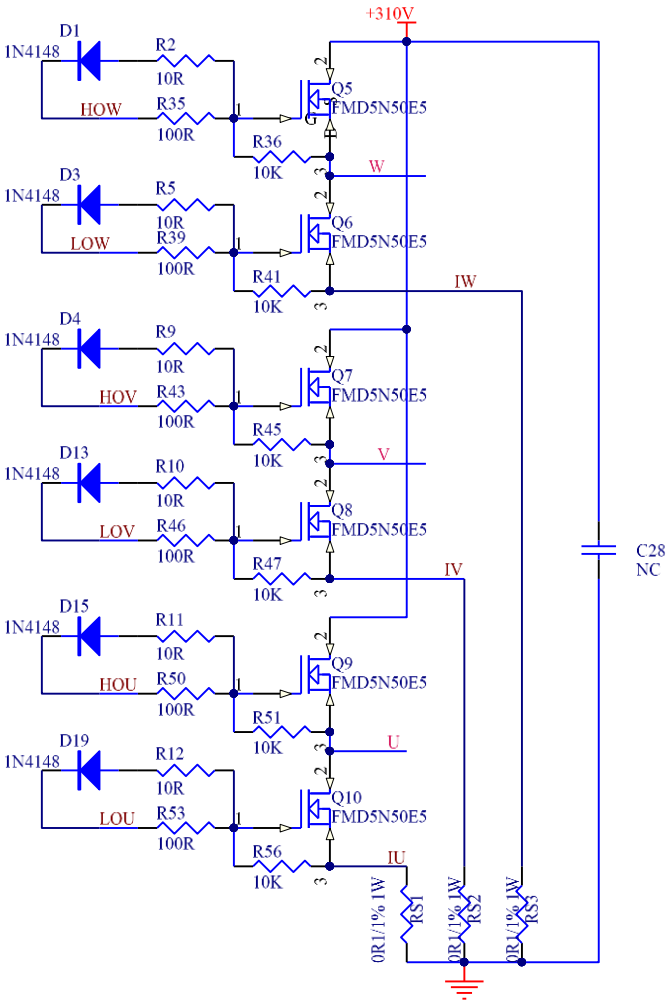
使用方式:

交流电源接 P1、P3 两端。

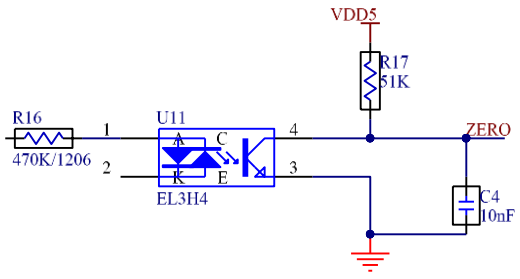
2.1.3 功率驱动电路

注意事项:

最大电流情况下，采样电阻功率不能超过额定功率的 80%。



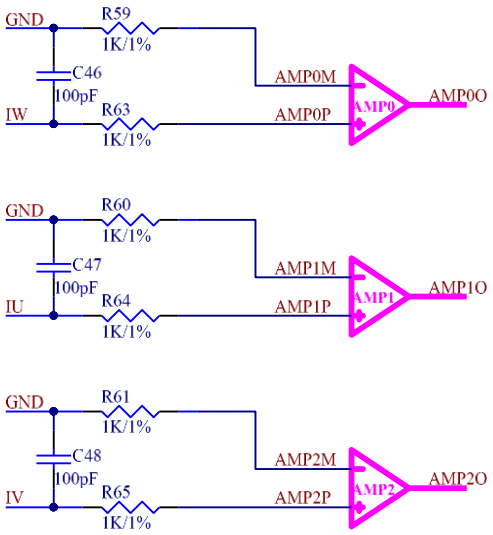
2.1.4 过零检测电路



注意事项:

该电路是检测过零信号的的，程序上是以外部中断的方式触发，所以过零信号需接到可作为外部中断的入口。C4 电容最好放在到靠近芯片端以减少干扰。

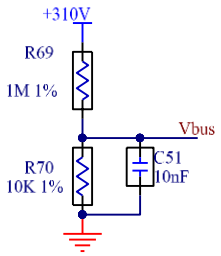
2.1.5 运放配置电路



注意事项:

1. C46、C47、C48、R59、R60、R61、R63、R64、R65 需要用 1%精度电阻；
2. 此原理图采用差分输入，程序上配置运放放大倍数；
3. 最大采样电流 = (VREF - VHALF)/放大倍数/采样电阻值；
4. 最大采样电流一般设置为最大母线电流的 4 倍左右。

2.1.6 母线电压采样电路



注意事项:

1. R69、R70 需要用 1%精度电阻；
2. 最大采样电压 = (R69 + R70)/(R70)*VREF；
3. 最大采样电压一般选择为 2 倍的最大应用电压，OVP 此处的电压需要低于 0.8*VREF。

3 软件架构

3.1 电机状态机流程图

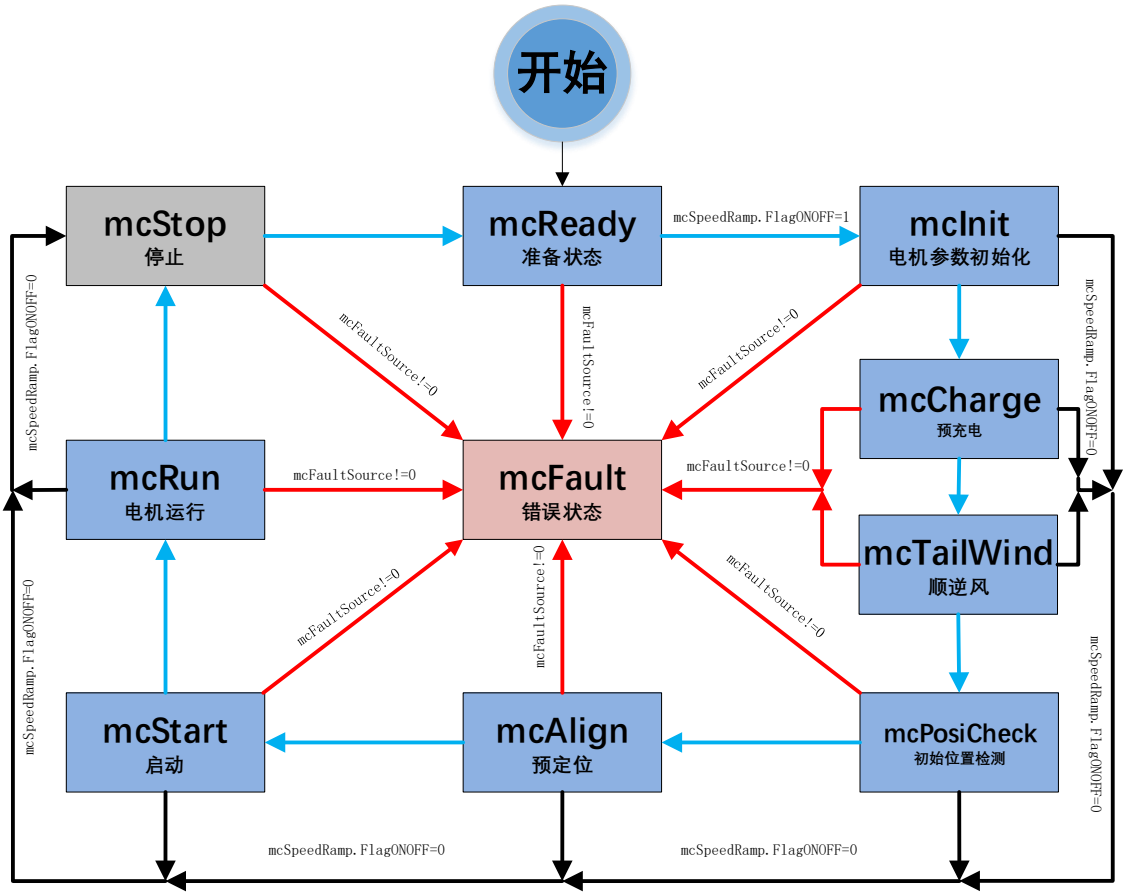


图 3-1 电机状态机流程图

如图所示，电机状态机分为三条路径:

1. 运行: mcReady -> mcInit -> mcCharge -> mcTailWind -> mcPosiCheck -> mcAlign -> mcStart -> mcRun；
2. 停机: mcInit、mcCharge、mcTailWind、mcPosiCheck、mcAlign、mcStart、mcRun 状态下如果检测到关机信号则会切入到 mcStop 状态进行降速关机；
3. 故障: 所有状态下发生故障均会跳转至 mcFault 状态，在 mcFault 状态将不再进行故障检测，因此不支持多故障并发的同时上报。

说明:

1. mcReady: 准备状态，等待开机命令，如果开机使能则跳转到 mcInit 状态；
2. mcInit: 相关变量和 PI 初始化，关闭电流，母线采样的外部 ADC 触发，然后跳转到下一状态；
3. mcCharge: 预充电采用的方式是 U 相上下桥臂先互补输出，再接着 U、V 相一起互补输出，最后 U、

V、W三相都互补输出。上下桥臂互补输出中实际有效的是下桥臂的电平输出，同时充电时间可调。

4. mcTailWind: 高速吹风筒不添加顺逆风判断，无论顺风或者逆风，高速吹风筒都是直接刹停，然后静止启动。
5. mcPosiCheck: 初始位置检测状态，检测电机的初始位置，再正常启动；高速吹风筒不添加初始位置检测，直接跳到预定位状态。
6. mcAlign: 预定位状态, 该状态下控制器输出恒定的电流将电机强行拖动到固定的角度上。定位结束则跳入下一个状态 mcStart；高速吹风筒程序上预定位时间给很短，相当于不定位直接强拉启动。
7. mcStart: 启动状态，该状态主要用于电机的启动代码配置，对相关寄存器代码与变量进行配置之后则转入下一个状态 mcRun。电机启动过程由 ME 内核实现；
8. mcRun: 运行状态，该状态包含: 电机启动阶段,电机运行阶段，电机速度的控制在该状态进行；
9. mcStop: 停机状态，该状态用于停机操作，速度降低到一定数值进行刹车，切入到 mcBreak 状态；
10. mcBreak:刹车状态，进行刹车处理（程序上刹车已在 mcStop 状态里处理），刹车结束切入到 mcReady 状态；
11. mcFault: 错误状态，当发生保护时，程序会记录错误源并且状态机会跳转到错误状态关机保护，当错误源被清掉时，会切入到 mcReady 状态，等待新的开机命令。

注意事项:

1. 电机状态机一共分为 8 个状态，状态之间只允许固定的状态跳转 例如: mcReady 状态只能向 mcInit 和 mcFault 状态跳转；
2. 特别的，mcTailWind，mcPosiCheck，mcAlign 三个状态都有使能位，当没使能时，直接跳转到下一个状态。例如: mcPosiCheck 没使能，mcAlign 也没使能时，mcTailWind 直接跳转到 mcStart 状态，高速吹风筒都直接跳过 mcTailWind，mcPosiCheck 状态，mcAlign 状态也是只给 1ms，相当于直接 Omega 启动。

3.2 程序流程图

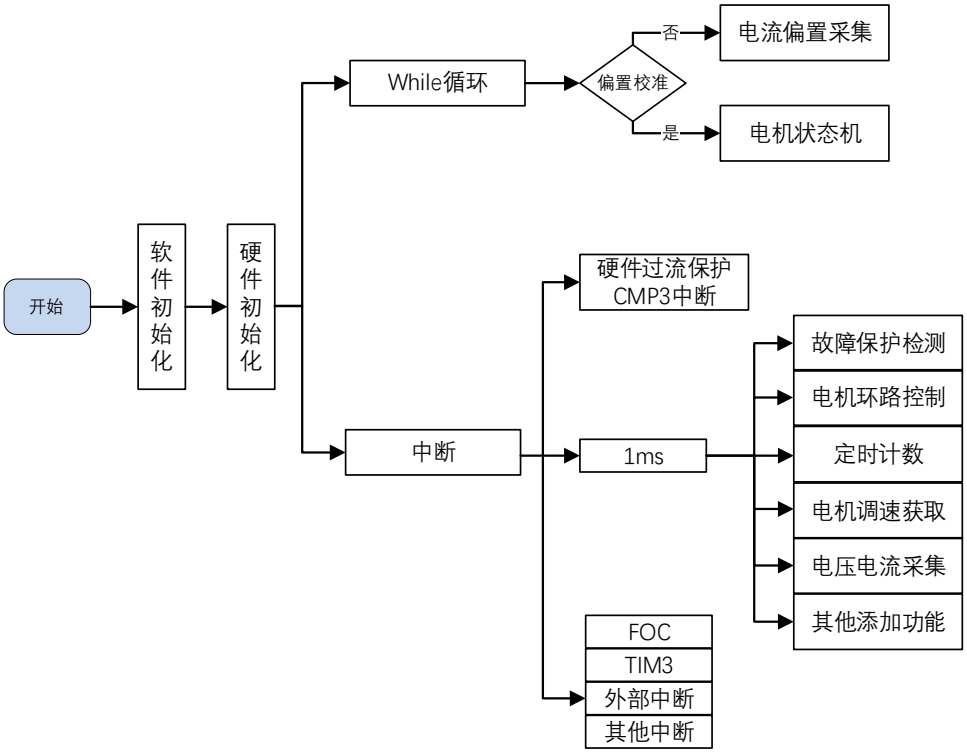


图 3-2 程序执行流程图

3.3 程序说明

3.3.1 Main 函数:

程序初始化->偏置电压检测 GetCurrentOffset () + 电机运行控制 MC_Control ()。

3.3.2 1ms 定时中断

程序中调速、故障保护检测、母线电流、母线电压采集等功能都在 1ms 中断中调用，包括以下函数:

Speed_response(); // 环路控制函数

TargetSpeed_Colletion (); // 按键调速功能

StarRampDealwith(); // 电机启动 ATO 爬坡控制

LEDControl (); //转速挡位赋值、发热挡位赋值、灯的显示

Fault_Detection (); // 故障检测

3.3.3 FOC 中断

FOC 中断，即载波中断，主要处理一些时序比较快的程序，如调用除法器等。

3.3.4 CMP3 中断

比较器 3 中断主要是处理硬件过流保护，具体原理可参考章节 5.2.1。

3.3.5 外部中断

外部中断用来采集过零信号，交流电过零的到来触发外部中断，然后通过 TIM3 延时控制 GPIO 口的关断与开通，从而达到控制发热的目的。

3.3.6 Timer3 中断

控制方式为斩波时：Timer3 中断主要是作为采集到过零之后做一个延时打开可控硅计数，过零触发外部中断之后，打开定时器 3 开始计数，延迟打开发热丝，延迟时间可调节以达到控制发热。

4 调试步骤

4.1 电机参数配置

4.1.1 电机参数

1. 电机极对数 Pole_Pairs；
2. 电机的相电阻 RS、相电感 LD、LQ，以及反电动势常数 Ke；
3. 电机速度基准，速度基准 MOTOR_SPEED_BASE = 2*电机额定转速。

4.1.2 电机参数测量方法

1. 极对数 Pole_Pairs: 电机设计时需给出的参数；
2. 相电阻 Rs: 万用表或者电桥测量电机两相线电阻 RL，相电阻 Rs = RL/2；
3. 相电感 Ls: 电桥测 1KHz 频率下的两相线电感 LL，相电感 Ls = LL/2；LD = LQ = Ls；
4. 反电动势常数 Ke: 示波器的探头接电机的一相，地接电机另外两相中的某一相，转动负载，测出反电动势波形。取中间的一个正弦波，测量其峰峰值 Vpp 和频率 f。计算公式如下:

$$Ke = 1000 * P * \frac{Vpp}{2 * 1.732 * 60 * f}$$

其中，P 为电机极对数。

示例，测量反电动势波形如下:

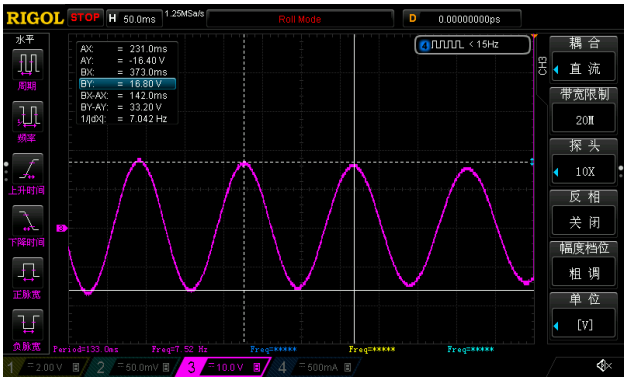


图 4-1 反电动势波形

测量峰峰值 Vpp 为 33.2V，频率 f 为 7.042Hz，极对数 P 为 4，则:

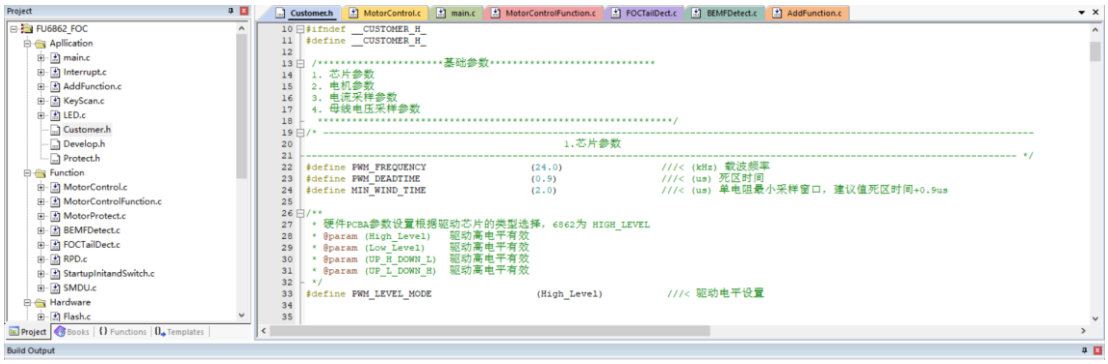
反电动势$Ke = 1000 * 4 * \frac{33.2}{2*1.732*60*7.042} = 90.73$(FU6862L 三电阻无需再测反电动势常数)

5. 速度基准 MOTOR_SPEED_BASE: 速度基准一般设置为电机最大转速的 2 倍左右，该值会影响启动等性能，一般需要提前定好之后，后面不要轻易改动。

4.1.3 对应程序



4.2 芯片内部参数配置





注意事项:

1. 载波频率一般需要设置为最大电周期 10 倍左右，载波频率会影响启动，MOS 温升等等，调试之前需要选择好合适的载波频率。吹风筒一般转速较高，可先用默认 24K 的调试；
2. 死区大小根据实际的 MOS 开关速度设置，保证没有直通风险；
3. 最小采样窗口设置，最小窗口最小需要大于 2 倍的死区，小于载波周期的 1/16，即 1000/16/PWM_FREQUENCY > MIN_WIND_TIME > 2*PWM_DEADTIME；
4. 正反转设置，根据实际接线设置，吹风筒电机反转会有高频噪声，且出风较正转小很多，如果电机反转了，则将该位取反即可。

4.3 硬件参数配置

1. 通过电机的电压范围和功率范围确认母线分压比、采样电阻值、放大倍数。
2. 电阻阻值跟放大倍数选取规则:
   1) 母线分压电阻:
      - 分压比不宜太小: 一般建议最大采集电压为 0.8*VREF，如某电机的最大电压为 30V，ADC 基准 VREF 为 4.5V，此时分压比建议不低于: 30/0.8/4.5 = 8.33；如果分压比大小，如分压比为 5，则 30V 时，经过分压后到 AD 口的电压为 6V，此时溢出了。
      - 分压比不宜太大: 分压比太大的话会导致 AD 采集电压精度不够，如最大电压为 30V，当分压比为 40 时，经过 AD 口的电压为 30V/40V = 0.75V，28V 时为 0.7V，此时精度比较低，而且 AD 还有 4.5 - 0.75 = 3.75V 的余量。
   2) 采样电阻与放大倍数:

      最大采集电流 = VREF/HW_RSHUNT/HW_AMPGAIN；这里要注意的是，最大采集电流不是电源上显示的电流(电源上显示的是滤波后的)，而是流经采样电阻的电流。
      - 采样电阻不宜太大: 太大的话容易导致采样溢出，或者本身的功率超过范围；2512 封装的采样电阻常见功率为 1W 或者 2W，1206 封装电阻的功率常见位 1/4W，选择时，要注意流经采样电阻的功率 I^2R 不要超过该功率。
      - 采样电阻不宜太小，太小的话精度不够
      - 放大倍数结合采样电阻调整，先确定了采样电阻，再去调整放大倍数

   其中，HW_RSHUNT 为采样电阻，HW_AMPGAIN 为放大倍数。
3. 母线分压比、采样电阻值、放大倍数对应填写到程序中(在 Customer.h 文件)。

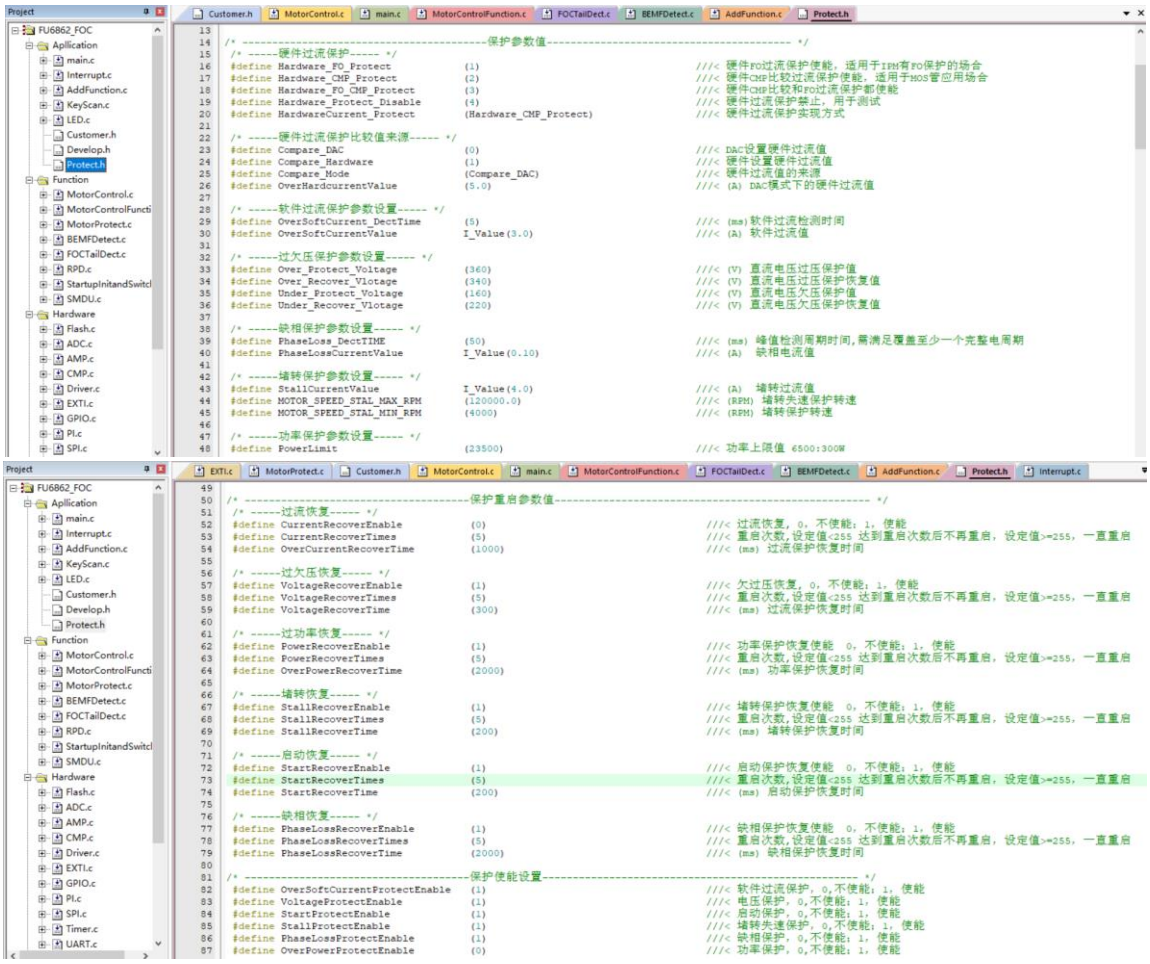


其中，

1) 母线分压比 = (RV1 + RV2 + RV3)/RV3；
2) VC1 为电压补偿系数，只应用在启动时刻，可先不动。

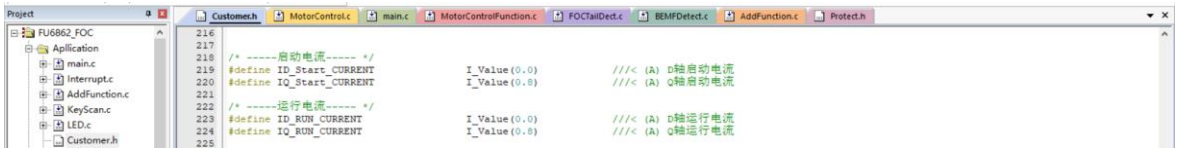
4.4 保护参数配置

1. 电流保护设置:
   - 硬件过流: 根据功率器件的最大电流值，设置硬件过流保护值，一般硬件过流保护值 OverHardcurrentValue 设置大于母线最大电流值，小于功率器件最大电流值。
   - 软件过流: OverSoftCurrentValue 一般设置比硬件过流小一点即可，软件过流为软件触发，保护时间不及硬件过流；
2. 设置过欠压保护跟保护恢复参数，详细设置参考章节 5.2.2；
3. 关闭上述保护的其他保护，防止启动的时候误触发，后面添加需要的保护再确认，其中过流保护是一定要开的，因此没有使能位；
4. 将参数对应填写到程序中(在 Protect.h 文件)。



4.5 启动参数配置

启动参数都先采用自带的**默认参数**，等启动有问题或者启动不顺的时候再做调整。启动常见的问题即参数调整可以参考章节 5.1。

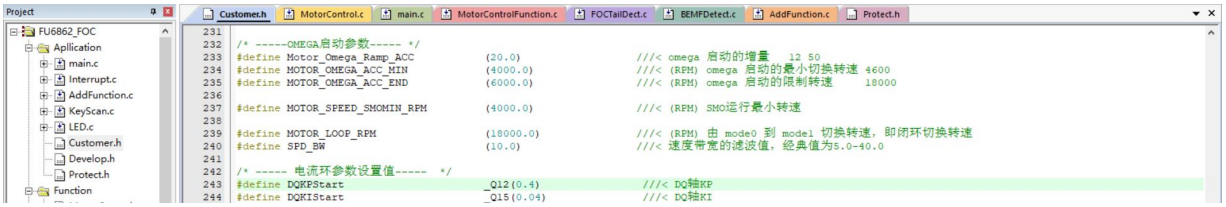


1. 启动电流: 一般 ID_Start_CURRENT 固定设置为 0，IQ_Start_CURRENT 根据实际电机设置确认；注意事项: IQ_Start_CURRENT，不能过小否则启动力矩太小导致启动失败。IQ_Start_CURRENT，不能过大否则启动过冲还会引入启动噪声。
2. 切换电流: IQ_RUN_CURRENT 只决定一瞬间的电流。通过实际观测相电流，可通过 IO 口翻转确认在切环瞬间是否存在电流不平滑，可以适当调整 IQ_RUN_CURRENT 解决；
3. 启动 ATO: 由于在较低转速下估算器输出存在误差，此时需要设置 ATO_BW 是(速度带宽滤波值)，以限制 FOC 估算器的最大转速输出；



注意事项: 对于高速吹风筒而言，启动的前 3 个 ATO 影响比较明显，需要根据实际情况调整。因为开的是 AO 观测器，所以第一个 ATO_BW 不用太大即可。

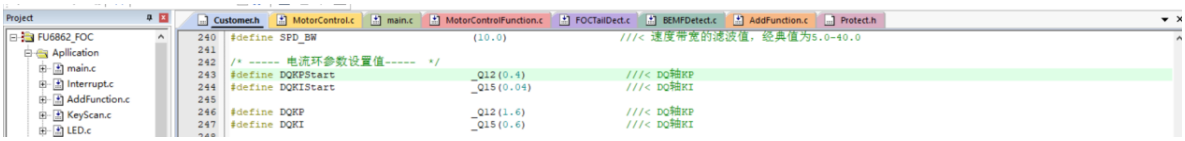
4. Omega 启动参数设置，影响启动的电流频率，即电机的启动加速度；



注意事项:

1) Motor_Omega_Ramp_ACC 参考值范围 10 ~ 50；
2) MOTOR_OMEGA_ACC_MIN 参考值范围 2000 ~ 10000；
3) MOTOR_OMEGA_ACC_END 参考值范围 3000 ~ 16000；
4) MOTOR_LOOP_RPM 需要大于 MOTOR_OMEGA_ACC_END，参考值范围 6000 ~ 16000；
5) MOTOR_SPEED_SMOMIN_RPM 参考值范围 3000 ~ 8000。

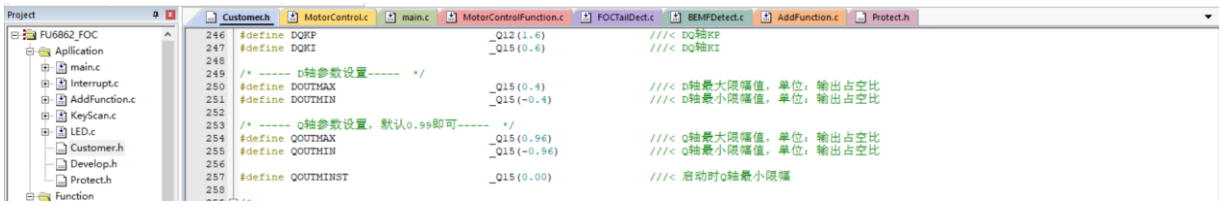
5. 电流环 PI: 电流环 PI 分启动的电流环 PI 跟运行时的电流环 PI；



注意事项:

1) 启动的电流环 PI，影响电机的启动；
2) 运行的电流环 PI，影响电流的稳定性，也影响效率，KPKI 给过小相电流会出现震荡；
3) DQKP 建议范围 0.4 ~ 2.0；
4) DQKI 建议范围 0.05 ~ 0.6。

6. DQ 轴最大输出限幅:D 轴影响电机的磁通，Q 轴影响电机的转矩。



注意事项:

1) FOC__UQ 反馈电机已经输出是否饱和；
2) FOC__UD 正得越多表示角度越超前，可以通过增加补偿角(FOC_THECOMP)让电机角度超前，此时能提升最大转速，FOC__UD 是一个正值；
3) 过多的超前角度，会导致关机时候电流过冲，可以通过低压预警关机处理，也可以通过快速欠压保护处理；
4) 过多的超前角度，会导致效率变差，相同功率下，相电流幅值更大，需要合理设置补偿角度。

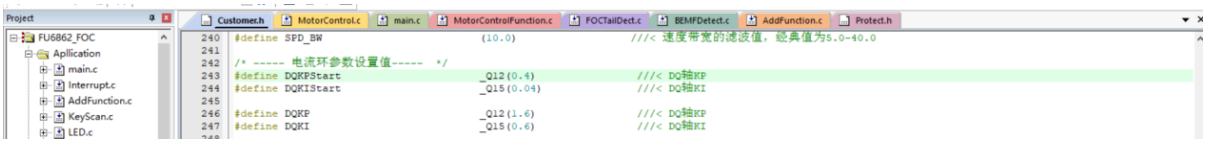
4.6 硬件驱动电路检测



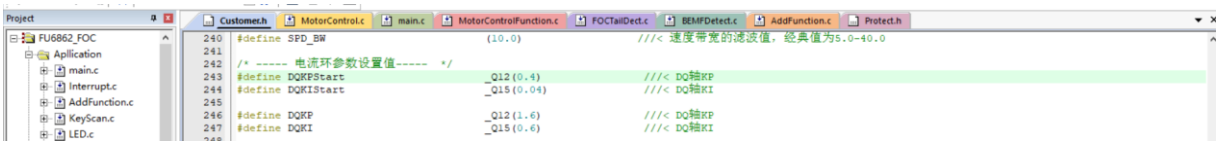
将IPMState模式选择为IPMtest，这个时候电机会跑到预定位状态，此时UVW三相会有固定的PWM波形输出，则硬件驱动电路正常。若没有输出，则要查找硬件问题。

4.7 电流环调试

1. 电流环启动时 KPKI 给小一点，得切入闭环时再把 KPKI 加大；



2. 运行时，母线电压会随着带载的大小而波动，这时相电流会出现震荡，需加大运行时的 KPKI，KP 一般到 1.5 左右，KI 一般 0.4 左右。（此值为参考值，可以根据实际去给）



常见问题即解决办法:

1) 运行过程中，触发过流保护；

解决: 看相电流波形是否异常，看是否是设定值比较小正常触发了过流保护。如果没异常的情况下，查看硬件布线等是否有问题。

2) 相电流波形有抖动。

解决: 调整电流环 PI(即 DQKP，DQKI)的值，电流环 PI 和电流采样对于电流波形的稳定性影响比较大。

4.8 速度环调试

1. 一般高速吹风筒都是用的恒转速环，因此将环路选择为速度环；



2. 设置速度环限幅最大值 SOUTMAX，如果设置过小会导致速度上不去，根据实际需要去做限幅；



3. 通过调整速度环 SKP、SKI、速度环爬坡增量、速度调节周期保证速度环稳定，启动响应快且不过冲。



4.9 增加按键功能

1. 一般吹风筒都是按键调速；

   1) 吹风筒程序上调速模式只有按键调速，板子正常情况下上电即会运行电机；



```
/**
 * @brief        调速信号处理包含：开关机控制、将调速信号处理成控制目标给定信号
 * @date         2022-07-14
 */
void TargetSpeed_Colletion(void)
{
    #if(SPEED_MODE == ONOFFTEST)
    {
        ONOFF_Starttest();
    }
    #elif(SPEED_MODE == KEYSCANMODE)
    {
        MCCtrl.PowerLimitValue = POWERLPFLIMIT;
        MCCtrl.FlagONOFF    = 1;
        KeyScan();                                          //获取按键值
    }
    #else
    {
        MCCtrl.PowerLimitValue = POWERLPFLIMIT;
        MCCtrl.TargetValue     = Motor_Speed_Low;       // 转速赋值
        MCCtrl.FlagONOFF       = 1;
    }
    #endif
    #if(FREnable ==1)
    {
        FRControl();
    }
    #endif
}
```

   2) 根据客户给的逻辑功能需求，修改逻辑；现在的按键逻辑如下: 风速 1->2->3->2->1…依此循环；风温: 0->1->2->3->2->1->0->1……依此循环(0 代表冷风)；一键冷风: 按下且一直按着冷风，松开恢复原来风温。

4.10 增加发热功能

1. 其中当设置 PORT0.0~PORT0.6 为数字 IO 输入，或者启用比较器 CMP4 时，可设置 EX0 = 1 使其作为外部中 0(INT0)。当设置 PORT1.0 ~ 1.7、PORT2.0 ~ 2.7 为数字 IO 输入时，可设置 EX1 = 1 及对应 P1IE/P2IE 使其共用外部中断 1(INT1)。外部中断 0 使能位 EX0，中断标志位 IF0，中断电平触发控制 IT0。由寄存器 LVSR 中的 EXT0CFG 指定外部中断 0 的来源，这些来源可以是 PORT0.0～PORT0.6 输入、比较器 CMP4 输出中的任一个，所有外部中断 0 的中断源共用一个中断入口、一个中断标志位。外部中断 1 使能位 EX1，16 个 PIN 的中断使能由寄存器 P1IE、P2IE 控制。对应的中断标志位为 P1IF、P2IF，中断电平触发控制为 IT1；

2. 外部中断配置，根据硬件设计配置过零触发 IO 口；



```
    ClrBit(P1_PU, P11);  // diasable P11 Pull Up
    // ClrBit(P1_AN, P11);  // diasable P11 analog function
    ClrBit(P1_IF, P11);        // clear P11 interrupt flag
    ClrBit(P1_IE, P11);   // config P11 as the source of EXTI1
    IT11 = 1;
    IT10 = 0;                           // 00: posedge mode interrupt, 01: negedge mode interrupt, 1x: edge-change mode interrupt
    PX11 = 0;
    PX10 = 0;                           // 中断优先级别0, 中断优先级最高
    EX1 = 1;                            // 使能外部中断1, P11位外部中断输入硬件
    EA = 1;                             // 使能全局中断
}

/**
 * @brief       过零点检测
 * @data        2022/08/12
 */
void ZeroCrossing_Init(void)
{
    ClrBit(P1_OE, P15);  // config P20 as input
    ClrBit(P1_PU, P15);  // diasable P20 Pull Up
    ClrBit(P1_AN, P15);  // diasable P20 analog function
    ClrBit(P1_IF, P15);        // clear P11 interrupt flag
    SetBit(P1_IE, P15);   // config P11 as the source of EXTI1
    IT11 = 0;
    IT10 = 0;                           // 00: posedge mode interrupt, 01: negedge mode interrupt, 1x: edge-change mode interrupt
    PX11 = 0;
    PX10 = 1;                           // 中断优先级别0, 中断优先级最高
    EX1 = 1;                            // 使能外部中断1, P20位外部中断输入硬件
    EA = 1;                             // 使能全局中断
}
```

触发了中断之后，要清对应的中断标志位。



```
void EXTERN_INT(void) interrupt 2
{
    static uint16 ScrOnTime = 0;                //发热开启时间

    if ((User.TPCtrlDealy < 4800) && (mcState == mcRun) && (mcFocCtrl.SpeedFlt > 12000))
    {
        ScrOnTime ++;
        if (ScrOnTime <= User.ScrOnPeriod)
        {
            if ((ScrOnTime < User.ScrOnAngleTimeDatum) && (User.ScrOnAngleTime != Temperature_Off))
            {
                GP01 = 0;                       //打开发热丝
            }
            else
            {
                GP01 = 1;                       //关闭发热丝
            }
        }
        else
        {
            ScrOnTime = 0;

            Temperature_Control();              //发热功率控制
        }
    }
    else
    {
        GP01 = 1;                               //关闭发热丝
    }
    ClrBit(P1_IF, P15);                         // 清零中断标志位
}
```

4.11 可靠性测试

4.11.1 功能可靠性

全部功能添加完成后，要再按照客户需求表重新测试确保没异常状态发生。

4.11.2 保护可靠性

保护添加之后，要验证保护都可以正常触发，且在电机运行时不会误触发。例如: 如果堵转保护的参数设置不合理，可能会导致电机在正常运行时也会误报堵转保护；或者是电机发生堵转后，不会触发堵转保护。

4.11.3 启动稳定性

在功能都基本调试完成之后，要做启动的可靠性测试，可先手动测试，手动测试没问题后，再进行老化测试。

老化测试步骤:

1. 将 SPEED_MODE，模式选为 ONOFFTEST；
2. 根据实际情况配置运行时间 StartON_Time 和停止时间 StartOFF_Time；
3. 调整 Motor_Speed_HIgh 的值可以修改启停的速度大小；
4. 先用工具堵住电机然后上电，看是否能正常触发启动保护，且保护后电机不会重启，即验证了启停时如果触发保护电机不会二次重启；
5. 再次上电进行老化测试即可。最后根据电机是否处于停止状态判断启动是否有异常，启动失败后电机会一直停机不再重启。一般测试 3000 次以上没问题则认为启动可靠。



```
/* ------------------------------------------------------------------------------------------------------
                                        9.调速参数设置
------------------------------------------------------------------------------------------------------ */
/**
 * 闭环方式选择
 * @param (PWMMODE)      PWM调速（未添加）
 * @param (ONOFFTEST)    启停测试工具
 * @param (NONEMODE)     直接给定值，不调速
 * @param (KEYSCANMODE)  按键控制模式
 */
#define SPEED_MODE                  (KEYSCANMODE)           ///< 闭环方式选择

/**
 * 测试模式下时间配置
 * @param (StartON_Time)     开启时间
 * @param (StartOFF_Time)    关闭时间
 */
#define StartON_Time                (3000)              ///< (ms) 启动运行时间
#define StartOFF_Time               (2000)              ///< (ms) 停止时间
```

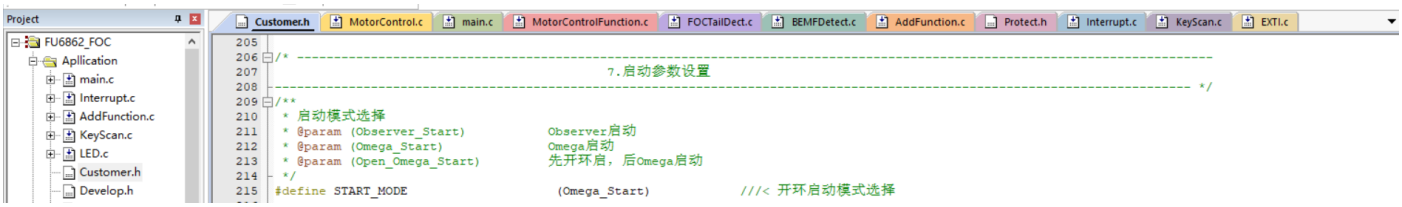
5 功能介绍

目前拿到初始版本程序，配置好电机参数，硬件参数后，给开机信号时，电机基本都能正常启动。若不能正常启动，则在排除是硬件问题的前提下，再调整启动参数。

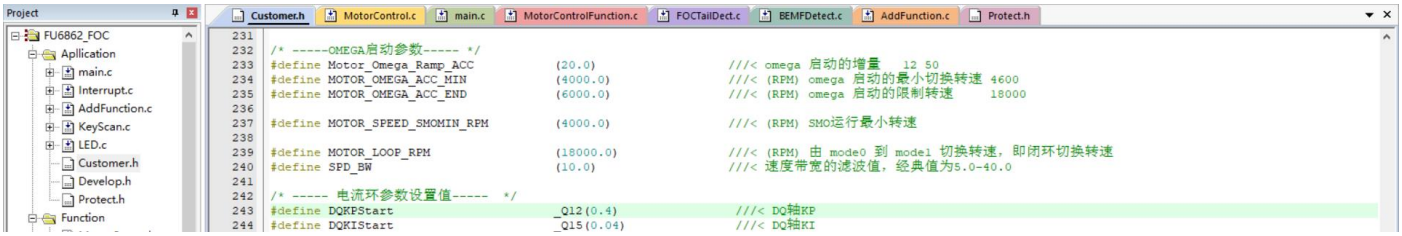
5.1 启动调试

5.1.1 Omega 启动

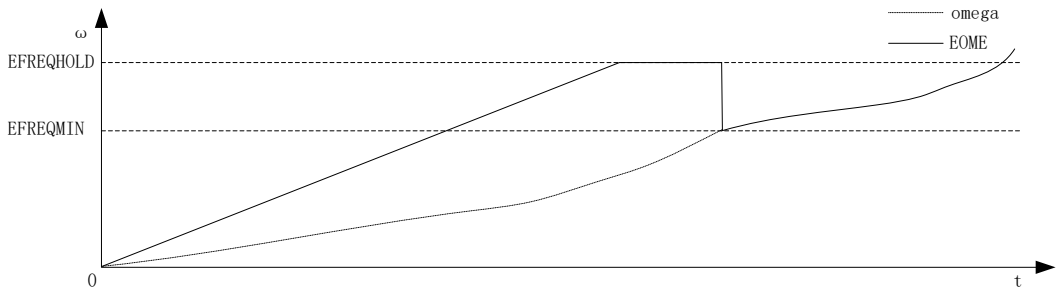
吹风筒选择 Omega 启动，程序对应默认即是该启动方式。



当估算器的估算速度 OMEGA 小于用户设定的最小值 FOC_EFREQMIN(对应 MOTOR_OMEGA_ACC_MIN 参数)，强制速度从 0 开始，每个运算周期与速度增量 FOC_EFREQACC(Motor_Omega_Ramp_ACC 参数)相加，同时根据 FOC_EFREQHOLD(MOTOR_OMEGA_ACC_END 参数)进行最大值限幅，输出强制速度作为最终速度 EOME 供角度计算模块算出估算器角度 ETHETA；当估算器的估算速度 OMEGA 大于等于 EFREQMIN 时，输出估算速度 OMEGA 作为最终速度 EOME。



启动过程如下图所示:



5.1.2 启动常见问题&解决方式

常见问题	解决方法
电机动一下后静止，且一直有电流输出	1.启动电流太小电机输出无法切换到下一拍换相，增大 IQ_Start_CURRENT； 2.估算器输出速度太小无法换相到下一拍，若排除 A 后此时依次加大 ATO_BW、ATO_BW_RUN、ATO_BW_RUN1、ATO_BW_RUN2； 3.排除 A、B 后，检查硬件电路运放 AMP0 部分是否有问题导致电流采样不准，估算器估算不正常； 4.也有可能是 Omega 加速度的频率太高导致，可以减小 Motor_Omega_Ramp_ACC。
电机转一下后停下且一直抖动	1.此种情况一般为 ATO_BW 值太大，导致估算器输出转速较高，启动时失步，此时依次减小 ATO_BW、ATO_BW_RUN、ATO_BW_RUN1、ATO_BW_RUN2； 2. Omega 启动参数也会有影响，SMO 运行最小转速 MOTOR_SPEED_SMOMIN_RPM，Omega 启动的限制转速 OTOR_OMEGA_ACC_EN，Omega 启动的最小切换转速 MOTOR_OMEGA_ACC_MIN 可以修改到合适的值
启动正序旋转一定角度后卡顿一下再正常旋转	1.此时可估计从启动到出现卡顿的时间，再设置对应时间的 ATO_BW 值，例如：启动 1s 后电机卡顿一下然后继续正常运行，1s 时间对应的值为 ATO_BW_RUN1、ATO_BW_RUN2，此时该 ATO_BW 值较小限制了电机加速，相应加大该值即可； 2.Omega 加速度太小时也会造成卡顿的情况，可以加大 Motor_Omega_Ramp_ACC。
电机启动反转后正转时持续抖动	1.电机启动反转一下后正转所需要的时间较长，此时 ATO_BW 已加到比较大的值，因此减小相应时间的 ATO_BW 值即可； 2.也有可能是 Omega 加速度的频率太高导致，可以修改 Motor_Omega_Ramp_ACC。

5.2 保护介绍

每个项目，不同电机，不同板子的保护值都会有所不同，各种保护的保护值都要根据实际项目去匹配。当发现保护，特别是堵转保护或缺相保护触发不了，或者正常运行时，误触发保护时，说明是保护设定值不合理导致的，此时要调整保护的设定值。

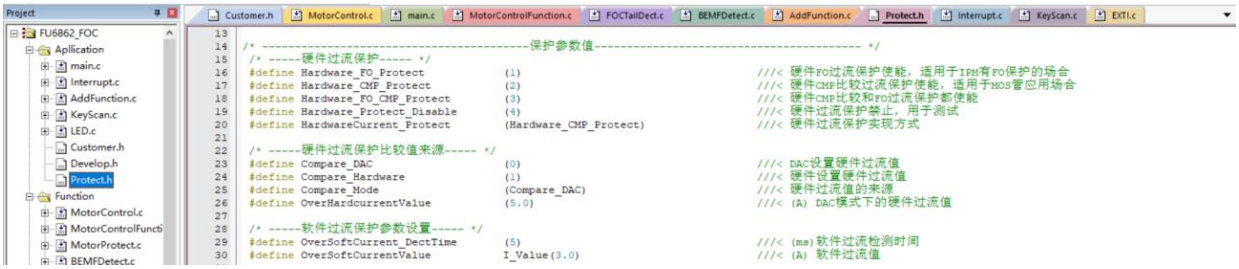
5.2.1 过流保护

1. 硬件过流保护；

芯片通过比较器 3 做硬件过流保护，检测方法：母线电流流经采样电阻，在采样电阻上形成一个电压,这个电压经过运算放大器放大送入比较器的正向输入端。比较器的负向输入端会被设置一个参考电压,这个参考电压可选择 DAC 产生或者由外部分压得到(目前都是用的 DAC 产生)。当母线电流增大到一定数值之后，就会导致比较器的正向输入端的电压高于负向输入端电压，这个时候就会触发 MCU 的比较器中断，MCU 发生中

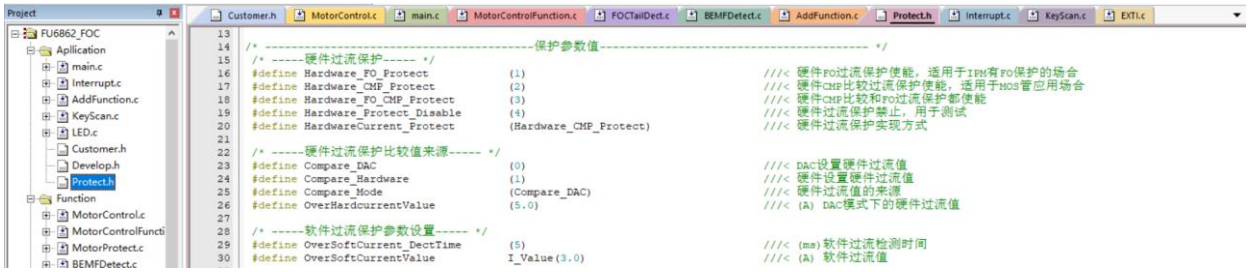
断并自动关闭 MOE(可选择自动或者不自动关闭 MOE，目前默认都是自动关闭 MOE)，从而完成过流保护。

硬件过流保护只需要修改保护值 OverHardcurrentValue 的大小即可。



2. 软件过流保护。

程序通过获取三相最大电流值，当最大电流值超过设定的软件过流保护值 OverSoftCurrentValue 时，则计一次；计数超过 3 次(可修改)时，则触发保护。



5.2.2 电压保护

程序通过AD2口检测电压，当检测到的电压超过设定值时，则报过压保护；此时当电压重新低于过压恢复值时，清除过压保护故障。当电压低于设定的欠压值时，则报欠压保护。此时当电压重新高于欠压恢复值时，清除欠压保护故障。



5.2.3 缺相保护

电机发生缺相时，三相电流是不对称的。因此可以通过在程序中检测一定时间内的三相电流的最大值，判断三相电流的最大值是否有不对称的情况来实现缺相保护。

具体程序实现方法：若检测到其中一相的最大电流大于另一相最大电流的3倍(可根据实际修改)，且该相最大电流大于设定的PhaseLossCurrentValue值，则判定为缺相。

注意事项：

有些方案在缺相时，由于缺的那一相会有毛刺的存在，可能会导致采集的最大电流值跟另外两相差不多，这时候通过上述方法可能检测不出来。解决方法：可以通过积分的方式，在一定时间内通过去比较电流累计值的大小去判断缺相。

5.2.4 启动保护

启动保护有三种方法检测:

1. 通过检测估算器计算出来的FOC_ESQU(估算器计算的反电动势的平方)判断，正常情况下，电机转速越高，FOC_ESQU会越大。在电机发生堵转时，电机失步的情况下，估算转速会很高，但是FOC_ESQU会很小，因此可以通过改方式判断；具体程序实现方法:判断FOC_ESQU的值还是小于设定值Stall_DectEsValue1；或者当估算转速高于设定值Motor_Max_Speed(可根据实际修改)，但是FOC_ESQU的值小于设定值5000(可根据实际修改)时，则判定为启动失败。



```
void Fault_Start(void)
{
    /********启动保护恢复*********/
    if (mcState == mcRun)
    {
        //方法一，5s内速度大于最大速度，同时反电动势值低于一定值
        if (mcFaultDect.StartSpeedCnt <= 800)
        {
            mcFaultDect.StartSpeedCnt++;

            if ((mcFocCtrl.SpeedFlt > Motor_Max_Speed) && (mcFocCtrl.EsValue < 5000))
            {
                mcFaultDect.StartSpeedCnt = 0;
                mcFaultSource = FaultStart;
                mcProtectTime.SecondStartTimes++;
                mcProtectTime.StartFlag  =  1;
            }
        }
```

2. 通过检测估算器计算出来的FOC_ESQU判断，在电机发生堵转或者电机失步的情况下，估算转速会很高，但是FOC_ESQU会很小，则认为发生了堵转或者失步，判定为启动失败；



```
        //方法二
        if (mcFaultDect.StartEsCnt <= 1000) //前1s，等待1.5s后，开始判断Es，如果超过一定次数，则失败
        {
            mcFaultDect.StartEsCnt++;
            mcFaultDect.StartDelay++;

            if (mcFaultDect.StartDelay >= 400)            // 1.5s
            {
                mcFaultDect.StartDelay = 400;

                if ((mcFocCtrl.EsValue < 1200)) //&&(mcFocCtrl.CtrlMode==0))
                {
                    mcFaultDect.StartESCount++;

                    if (mcFaultDect.StartESCount >= 40)  //50次Es小于设定值，启动失败
                    {
                        mcFaultSource = FaultStart;
                        mcProtectTime.SecondStartTimes++;
                        mcFaultDect.StartDelay = 0;
                        mcFaultDect.StartESCount = 0;
                        mcProtectTime.StartFlag  =  2;
                    }
                }
                else
                {
                    if (mcFaultDect.StartESCount > 0)
                    { mcFaultDect.StartESCount--; }
                }
            }
        }
        else
        {
            mcFaultDect.StartESCount = 0;
        }
```

3. 电机启动的时候，程序会在判断估算转速大于MOTOR_LOOP_RPM后，将Mode状态从0置为1，从固定电流启动，进入正常的环路。此时可以通过该Mode去判断是否发生了堵转。若在开机经过2000ms(可根据实际修改)时间后,Mode仍然处于0的状态，则认为电机启动失败。



```
            mcFaultDect.StartESCount = 0;
        }

        //方法三，长时间在CtrlMode=0状态
        if (mcFocCtrl.CtrlMode == 0)
        {
            mcFaultDect.StartFocmode++;

            if (mcFaultDect.StartFocmode >= 2000) //在MODE0模式超过2s起动失败
            {
                mcFaultDect.StartFocmode = 0;
                mcFaultSource = FaultStart;
                mcProtectTime.SecondStartTimes++;
                mcProtectTime.StartFlag  =  3;
            }
        }
    }
```

5.2.5 堵转保护

堵转保护有三种方法检测：

1. 通过检测估算器计算出来的FOC_ESQU(估算器计算的反电动势的平方)判断，正常情况下，电机转速越高，FOC_ESQU会越大。在电机发生堵转时，电机失步的情况下，估算转速会很高，但是FOC_ESQU会很小，因此可以通过改方式判断；

具体程序实现方法：当开机延时1200ms后，判断FOC_ESQU的值小于设定值2000同时当估算转速高于设定值1200转，或者FOC_ESQU的值小于设定值2000时，计数50ms, 达到计数值则判定为堵转。

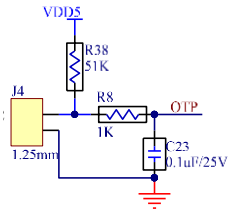


2. 通过检测估算转速，当估算转速超过设定转速Motor_Stall_Max_Spee，或者低于设定转速Motor_Stall_Min_Speed，则认为发生了堵转；



5.2.6 过温保护

过温保护常用的电路图如下所示，分压电阻通常会用一个NTC电阻，NTC放在出风口位置，该电阻随着温度的上升，阻值逐渐下降。在每个温度都会有对应一个阻值。OTP连接到芯片的一个AD端口。程序通过检测该AD口的电压，当该电压小于设定温度下的电压时，则表明NTC电阻温度超过了设定值，触发保护。



标准程序上没有这一段保护，可根据实际需要添加。

5.2.7 功率保护

当采集到功率滤波值大于功率保护设定值PowerLimit时，计数80ms,达到计数值即判断为功率过大。



5.2.8 其他保护

根据客户需求自行添加其他保护。

6 其他常见功能调试

6.1 限功率功能

使用恒转速控制时，当吹风筒入风口堵住时，负载变小电机会运行至较高转速会损坏轴承，而且电机散热较差导致电机损坏，故需要使用限功率功能对功率进行限定。

限功率方式: 双 PI 的方式限制功率，硬件 PI1 实现速度闭环，硬件 PI2 实现功率限制:



7 方案调试难点&解决方法

恒功率调试	
常见问题	**解决方法**
启动一直有异常	启动一直调试不好，软件问题排除后，可以查看硬件采样布线等是否有问题。
启动老化测试发现启动有失败	可以优化参数 Omega 启动参数，ATO 参数、电流环启动 KPKI、
电机速度响应较慢或者过快	1. 调试外环的 SKP,SKI； 2. 调节时间 SPEED_LOOP_TIME； 3. 如果只是加减速比较慢，就调节加减速的增减量。
运行中出现相电流有规律的振荡	母线电压震荡导致，可以加大电流环 KP、KI
转速或者功率达不到客户要求	1. 电流波形正弦的情况下，通过观测 FOC__UQ 是否饱和 2. FOC_UQ 饱和，且 FOC__UD 值比较大的话，通过调整补偿角 FOC_THECOMP(正负都调整看看)确认是否能达到客户需求； 3. FOC_UQ 没饱和，可以检查是否是电流环限幅 QOUTMAX、外环限幅 SOUTMAX 没有设的足够大，以及检查最大功率限幅值 POWERLPFLIMIT 是否过小。
电机运行到高转速后容易出现大电流的情况	1. 调整补偿角 FOC_THECOMP； 2. 检查硬件采样是否存在干扰。 3. 检查电机参数是否填正确
电流波形存在正弦度失真	1. 看采样偏置基准是否正常； 2. 修改电流环的 PI，即 DQKP,DQKI； 3. 修改载波频率(注意修改后会影响启动跟运行)。
发热不受控或者功率跳动大	1. 检查过零采样是否存在干扰，是否按照配置去触发外部中断，如果过零信号存在毛刺，会不停触发外部中断，导致发热不受控，一般原因为地线干扰大。 2. 检查驱动可控硅的信号是否正确，如果存在干扰，驱动信号会输出异常，导致发热不受控，一般原因为地线干扰大。
注意事项：修改参数后，基本都会影响启动和运行性能，解决好问题后要重新测试确认。	

8 版权说明

版权所有©峰岹科技（深圳）股份有限公司（以下简称：峰岹科技）。

为改进设计和/或性能，峰岹科技保留对本文档所描述或包含的产品（包括电路、标准元件和/或软件）进行更改的权利。本文档中包含的信息供峰岹科技的客户进行一般性使用。峰岹科技的客户应确保采取适当行动，以使其对峰岹科技产品的使用不侵犯任何专利。峰岹科技尊重第三方的有效专利权，不侵犯或协助他人侵犯该等权利。
本文档版权归峰岹科技所有，未经峰岹科技明确书面许可，任何单位及个人不得以任何形式或方式（如电子、机械、磁性、光学、化学、手工操作或其他任何方式），对本文档任何内容进行复制、传播、抄录、存储于检索系统或翻译为任何语种，亦不得更改或删除本内容副本中的任何版权或其他声明信息。

峰岹科技（深圳）股份有限公司
深圳市南山区科技中二路深圳软件园二期 11 栋 2 楼 203
邮编：518057
电话：0755-26867710
传真：0755-26867715
网址：www.fortiortech.com

峰岹科技（深圳）股份有限公司

ISO9001 质量管控文件

FU6862

吹风筒应用手册

文件版本管制

版本	修订者	主要修改内容	生效日期	审核/批准
V1.0.00	陈正孔	初稿	2022/07/07	
V1.0.01	陈正孔	修改硬件、调试步骤部分描述	2022/07/27	

会签单位	签核人员	签核时间
应用部		
品质部		