

HC32M140 系列

风机无传感器控制方案

用户手册

适用产品
本编程器支持芯片型号如下

系列	型号
HC32M140	HC32M140F8TA HC32M140J8UA HC32M140J8TA HC32M140KATA

目 录

1. 简介	4
1.1 概览.....	4
2. 系统环境	4
2.1 硬件环境.....	4
2.2 软件环境.....	4
3. 硬件部分	5
3.1 硬件框图.....	5
3.2 主要特征.....	5
3.3 硬件电路.....	6
3.3.1 芯片介绍.....	6
3.3.2 电源部分.....	6
3.3.3 UART 电路/PWM 电路.....	6
3.3.4 电位器调速电路.....	6
3.3.5 ADC 采样.....	6
3.3.6 三相逆变器.....	6
3.3.7 过流保护.....	6
4. 软件设计	7
4.1 软件框图.....	7
4.2 软件功能.....	8
4.3 软件结构.....	10
4.4 文件描述.....	12
5. 系统中断	13
5.1 中断函数.....	13
5.2 中断优先级设置.....	14
6. 系统功能函数	15
6.1 结构体.....	15
6.2 系统函数.....	17
7. 系统参数	25
7.1 硬件参数.....	25
7.2 用户参数.....	26
7.2.1 用户参数定义.....	27
7.2.2 用户参数计算.....	29
8. 系统数据处理	32
8.1 数据标么.....	32
8.1.1 数据标么的方法.....	32
8.1.2 标么基值的设定.....	32
8.2 Q 格式化数据.....	33
8.2.1 Q 格式化方法.....	33
8.2.2 Q 格式化数据范围.....	33
9. 参数设定与程序编译下载	34
9.1 实物连接.....	34
9.2 打开工程.....	36
9.3 用户配置.....	37
9.3.1 中断参数配置.....	37
9.3.2 硬件参数配置.....	37
9.3.3 电机运行相关参数的配置.....	38
9.4 代码编译.....	41

9.5	代码下载.....	42
9.6	程序运行.....	42
9.7	电机运行控制以及运行时参数观测.....	43
9.8	调试退出.....	44
10.	实例讲解	45
10.1	速度控制.....	45
10.2	转向控制.....	47
10.3	PI 参数调节	48
10.4	速度 PI 参数调节	50
11.	附录	51
11.1	原理图.....	51
12.	版本信息 & 联系方式.....	52

1. 简介

本手册介绍了 HDSC 基于 HC32M140 系列芯片的风机无传感器控制解决方案。

1.1 概览

通过该手册可以了解到的信息有：风机的无传感器控制的硬件情况、控制系统的软件框架、软件功能、软件结构、以及控制方案中的数据处理方式，同时还能够了解到如何进行相关参数的测试与计算以及使用风机的无传感控制方案的示例讲解。

2. 系统环境

2.1 硬件环境

表 1 硬件开发环境

系统	描述
MCU	HC32M140
硬件	EV-MD050-HV-02-V10
调试器	JLINK V9隔离仿真器
电机类型	PMSM/BLDC

2.2 软件环境

表 2 软件开发环境

类型	描述
操作系统	Windows 7 64-bit
编译器	IAR EWARM V7.7
编译语言	C

3. 硬件部分

本章介绍风机无传感器控制系统 Demo 板的硬件框图、功能特征以及各部分电路的情况。

3.1 硬件框图

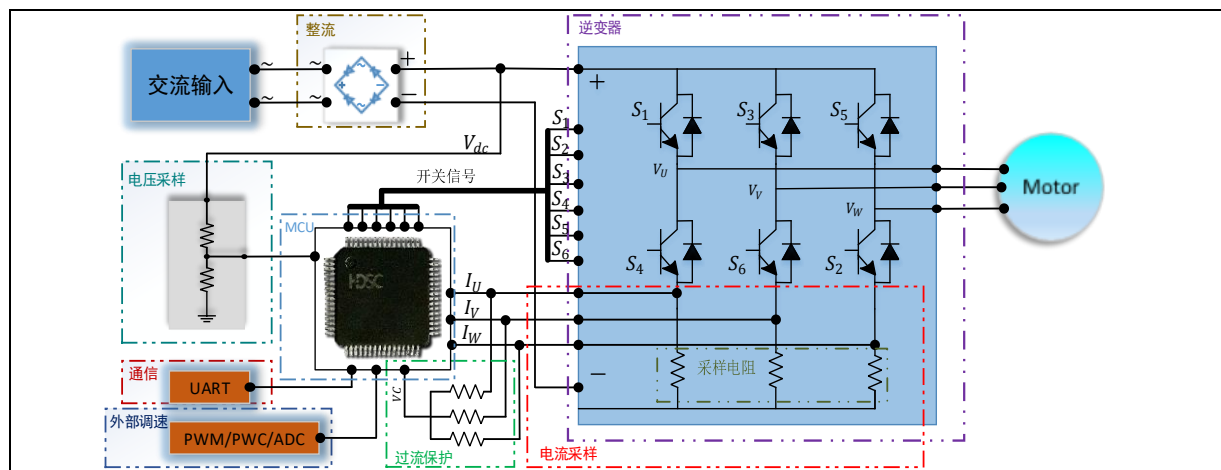


图 1 硬件系统框图

3.2 主要特征

- 1) 输入：220VAC $\pm 10\%$
- 2) 最大功率可达 50W
- 3) MCU 内置放大器，比较器，削减 BOM 成本
- 4) 隔离的 UART 和 PWM/PWC，保证在通信时设备的安全
- 5) 支持多种速度控制的方式（PWM、UART、电位器）
- 6) 完善的系统保护功能（过压/欠压/过流/堵转/缺相等）
- 7) 支持异常情况紧急停机功能
- 8) 支持运行 Sensorless FOC 控制，运行噪音低，电流谐波小
- 9) 支持各种工况下的电机启动（顺风/逆风/静止启动）

3.3 硬件电路

3.3.1 芯片介绍

Demo 板使用 HC32M140 作为主控芯片，该芯片是本公司为电机控制专门设计开发的一块控制芯片，该芯片以 Cortex-M0+ 为内核，最大工作频率可达 48MHz。

3.3.2 电源部分

将交流电整流为直流电，然后将电流稳压成需要的电平。

3.3.3 UART 电路/PWM 电路

使用隔离的方式进行搭建，保证在高电压下能够安全使用。

3.3.4 电位器调速电路

使用 ADC 采样获取电位器阻值，根据阻值的不同来控制风机的速度。

3.3.5 ADC 采样

ADC 采样分为电压采样和电流采样，电压采样使用电阻分压的方式进行采样，完成 V_{dc} 电压采样；电流采样通过使用芯片内部集成的放大器构建电流采样电路，完成对电机三相电流的采样。

3.3.6 三相逆变器

完成开关信号到三相电压的转换。

3.3.7 过流保护

使用芯片内部集成的比较器对电机的电流的进行监测，当电机电流过大时配合紧急停止功能完成电机的紧急停机功能，保护电路不被烧毁。

注意：

- 电机驱动板原理图详见附录原理图。

4. 软件设计

本章对风机无传感器控制方案的软件框图、软件功能以及软件结构等情况进行讲解。

4.1 软件框图

图 2 为风机无传感器控制方案的软件框图。

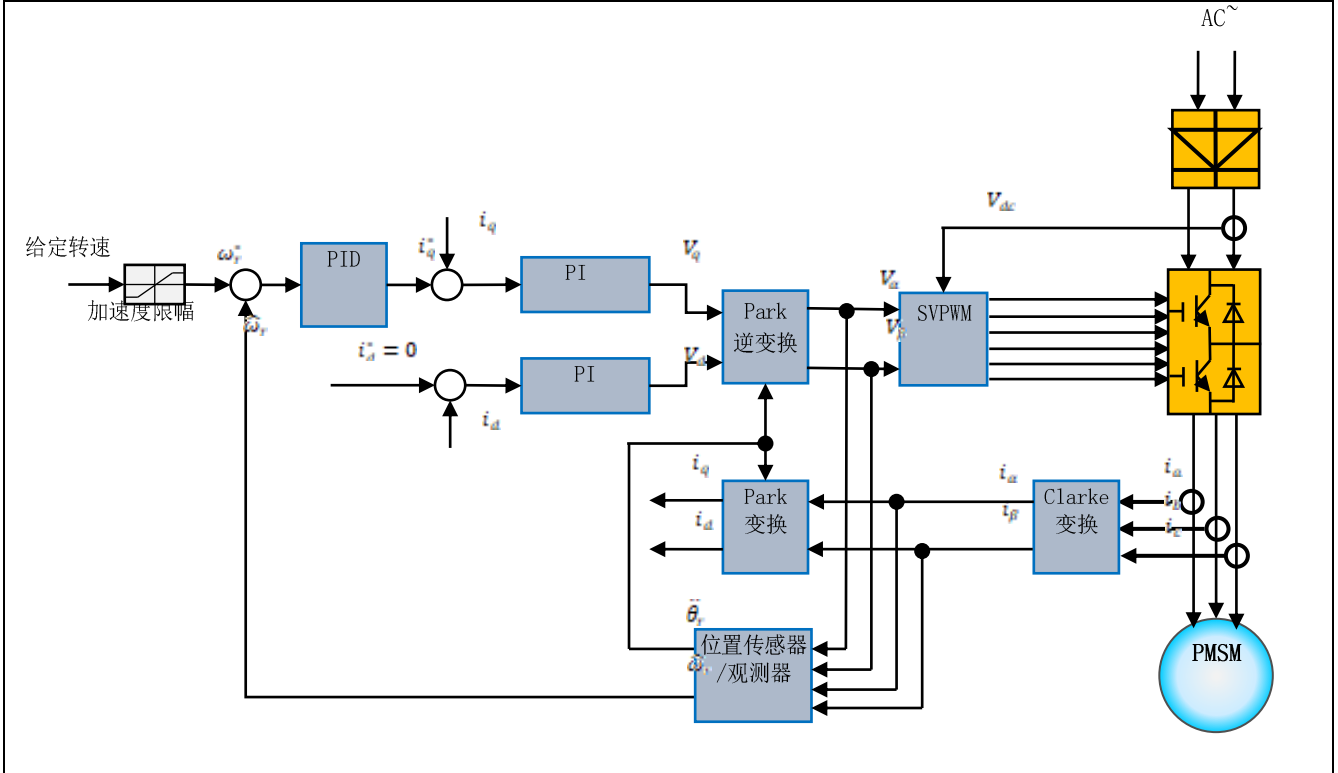


图 2 风机无传感器控制软件框图

4.2 软件功能

控制程序具有的功能如表 3 所示：

表 3 无传感器 FOC 控制程序功能

NO	功能	描述	备注
1	坐标变换	Clarke变换	
		Clarke逆变换	
		Park变换	
		Park逆变换	
2	数学计算	三角函数正弦值计算	
		三角函数余弦值计算	
		三角函数反正切值计算	
3	SVPWM	SVPWM矢量作用时间计算	
		SVPWM开关计数值计算	
4	数字滤波	一阶滤波	
		算术平均值滤波	
5	PID控制	增量式PID控制	
		位置式PID控制	
6	FOC控制	FOC控制算法	
7	硬件自检测	检测ADC电流采样是否正确	
8	ADC采样	电压采样	
		电流采样	
9	死区电压计算	计算死区损失电压	
10	速度、位置估计	电机的转速、位置估算	
11	速度控制	电机转速设置	
12	保护	欠压保护	
		过压保护	
		A/D偏移保护	
		堵转保护	
		过流保护	

NO	功能	描述	备注
13	不停机换向	在不停机的状态下进行电机的换向操作	
14	异常停机	工作状态异常，快速停机	
15	电流电压限幅	限制电流和电压的幅值	
16	软件时间事件	1ms事件、5ms事件、50ms事件等	
17	启动控制	电机任意位置启动	
18	停机控制	电机停止转动	
19	状态记录	记录电机运行过程的状态	
20	加减速控制	速度加减控制	
21	制动控制	刹车	
22	通信	UART发送接收数据	

4.3 软件结构

风机无传感器控制的软件架构如图 3 所示。

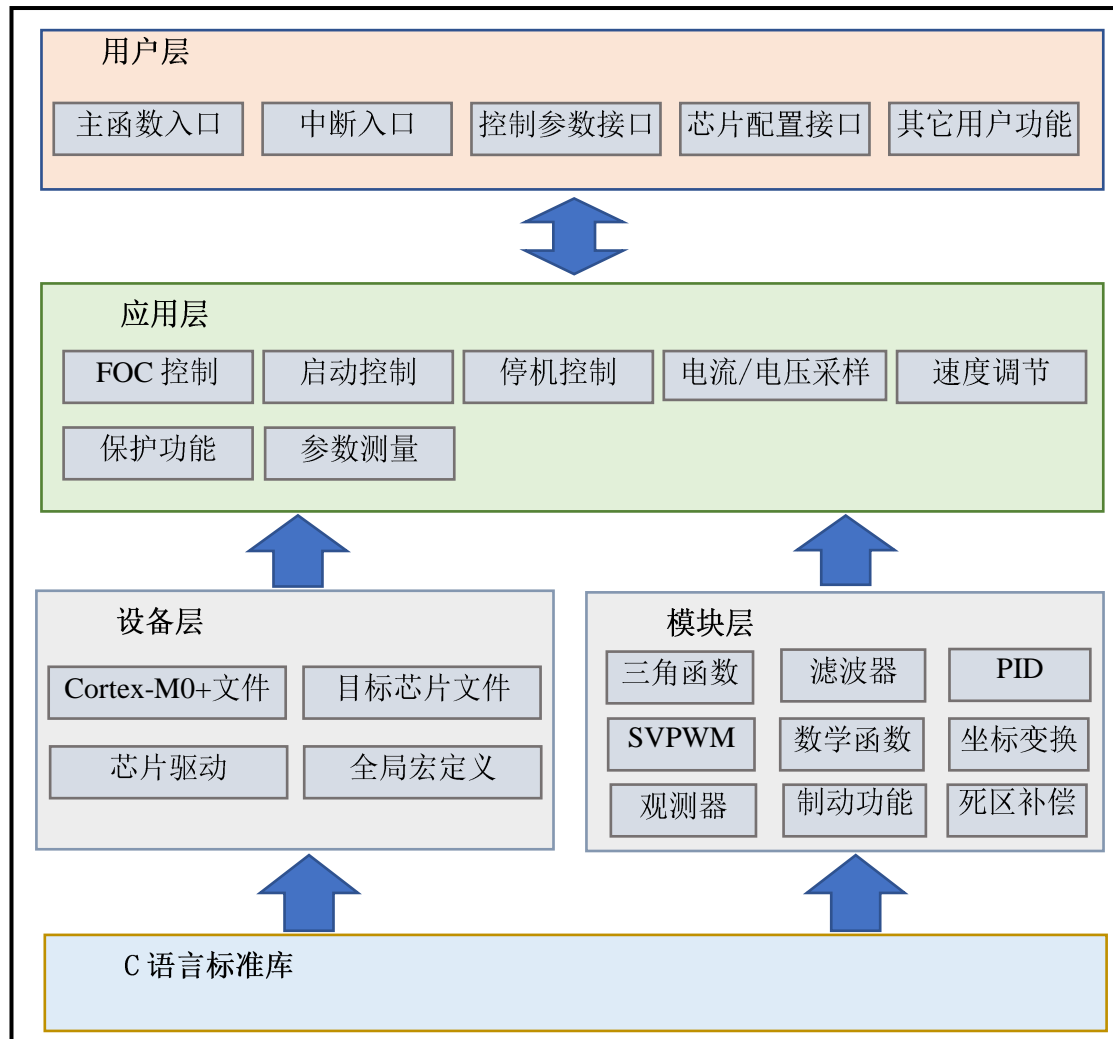


图 3 风机无传感器控制软件架构

风机无传感器控制的文件结构的描述如表 4 所示。

表 4 风机无传感器控制文件结构

Layer	对应文件夹	描述	备注
device	h01_device	MCU驱动文件（Clk、Swwdt、Hwwdt、GPIO、Timer4等）	
	s01_device		
module	h02_module	基本FOC控制算法中的各个模块（坐标变换、Filter、PID、SVPWM、刹车控制等）	
	s02_module		
application	h03_application	电机控制应用功能文件夹（ADC采样、启动控制、保护、速度设置等）	
	s03_application		
user	h04_user	用户配置文件夹（电机参数、PID参数、硬件参数等）	

风机无传感控制文件结构在 IAR 工程中的情况如下。

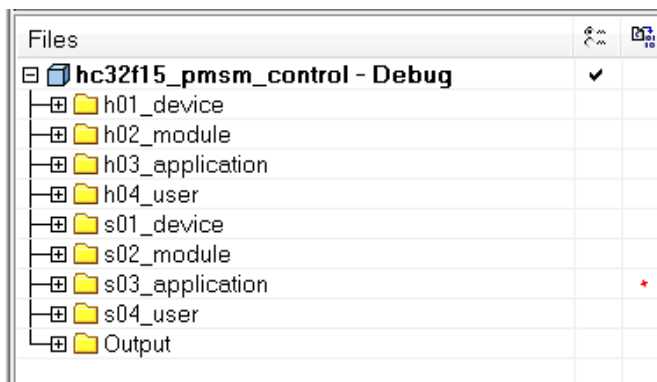


图 4 IAR 工程中风机无传感器控制文件结构

注意：

- 其中.a 文件放在以 S 开头的文件夹中，.h 文件放在 h 开头的文件夹中。

4.4 文件描述

IAR 工程中无传感器控制详细的文件描述如下。

表 5 详细文件描述

文件夹	文件	描述
h02_module	coordinate_transform.h	坐标变换 (Park、Clarke、InvPark、InvClarke)
	base_types.h	数据类型的定义
	filter.h	数字滤波 (一阶滤波、算术平均值滤波)
	module_math.h	算术运算 (正余弦值计算、反正切值计算等)
	pid.h	PID控制 (增量式PID、位置式PID)
	svpwm.h	SVPWM算法
	brake.h	刹车控制
	dead_time_comp.h	IGBT死区损失电压计算
	observer_bemf.h	电机速度、位置估计
	module_include.h	h02_module层头文件的声明, 运行状态定义
h03_application	adc_sample.h	ADC采样 (V_{ac} 电压、三相电流)
	motor_timer.h	电机运行时间事件记录
	motor_control.h	电机控制 (转速、停机控制等)
	protection.h	电机保护 (过压保护、欠压保护、过流保护、堵转保护等)
	speed_set.h	电机转速设置
	startup_control.h	电机启动控制
	stop_control.h	停机控制
h04_user	init_mcu.h	MCU初始化设置 (ADC模块、Timer4模块等)
	hardware_config.h	硬件参数配置
	isr.h	中断处理函数
	main.h	主函数
	startup_hc32f_m14x.s	MCU启动文件
	user_interface.h	用户配置文件

5. 系统中断

本章主要讲述控制系统中的系统中断函数的使用情况以及如何进行相关的配置。

5.1 中断函数

风机无传感器控制中涉及的中断函数如下：

表 6 中断函数列表

中断函数	描述
void NMI_Handler (void)	系统中断（包含硬件看门狗中断）
void SWDT_IRQHandler(void)	软件看门狗中断
void TIM4CNT_IRQ(void)	Timer4计数中断
void TIM4RT_EMI_IRQHandler(void)	Timer4紧急停机中断
void ADC_IRQHandler(void)	ADC中断
void MSC5RX_IRQHandler(void)	串口接收中断
void MSC5TX_IRQHandler(void)	串口发送中断

中断函数的实现函数位于 s04_user\isr.c 文件中，如图 5 所示。

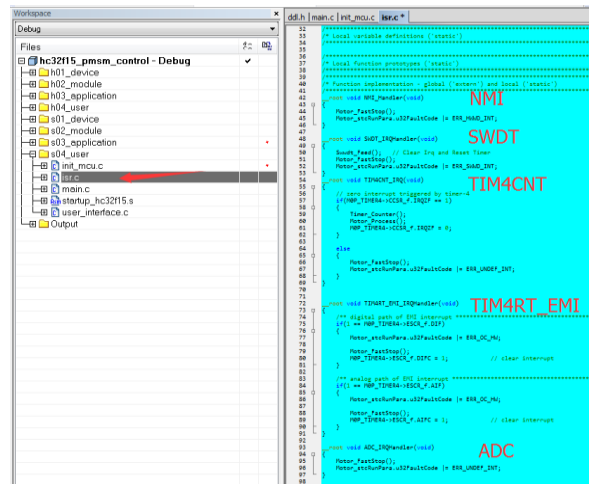


图 5 中断函数

若用户需要修改控制系统中断函数的名称，需要进入 h01_device\ddl.h 文件，将对应中断函数的宏定义名进行修改（建议用户使用默认函数名），如图 6 所示。

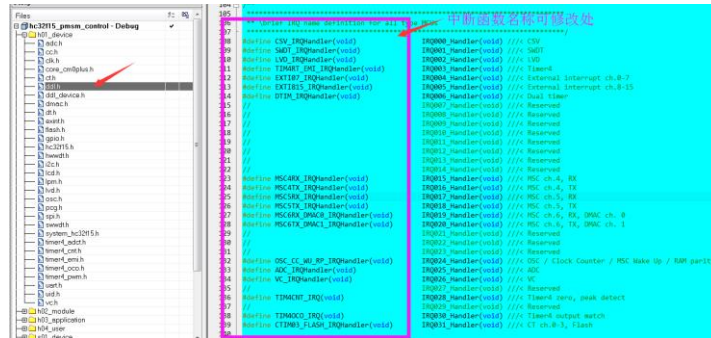


图 6 中断函数名称修改

5.2 中断优先级设置

软件中使用的中断及其使能位于 s04_user\init_mcu.c 文件中，如图 7 所示：

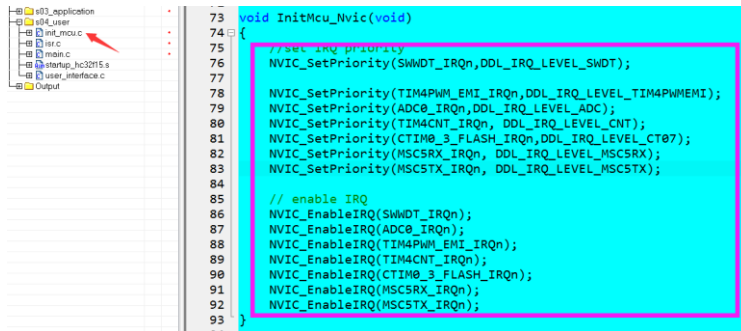


图 7 中断及其使能

控制系统中默认的中断优先级为：NMI > SWWDT = TIM4RT_EMI > ADC > TIM4CNT > CTIM。若用户需要修改中断优先级，需要对 h01_device\ddl.h 文件中对中断优先级的宏定义作相应修改便可（建议用户使用默认的中断优先级）。如图 8 所示：

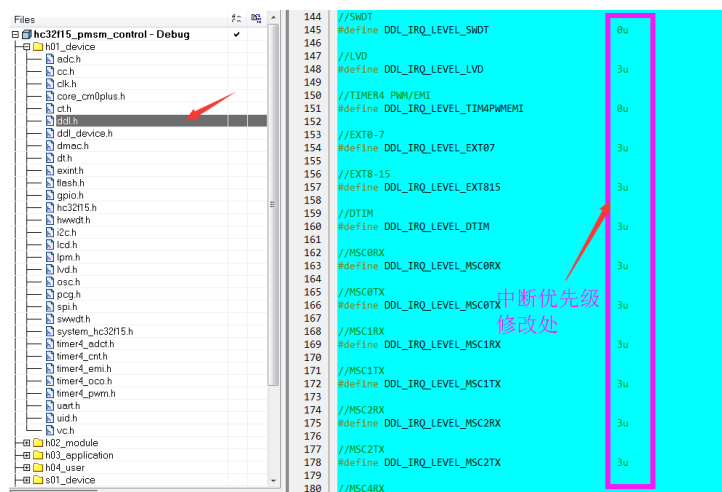


图 8 中断优先级修改

6. 系统功能函数

本章主要对风机无传感器控制中的结构体以及函数的使用情况进行描述。

6.1 结构体

在控制系统中通过定义结构体来存储数据，情况如下：

表 7 结构体列表

结构体类型	结构体名	描述	备注
stc_adc_sample_t	Adc_stcSample	ADC采样	
stc_uvw_t	Motor_stcIuvw	电机电流	
stc_ab_t	Motor_stcIab		
stc_dq_t	Motor_stcIdq		
	Motor_stcIdqRef		
stc_ab_t	Motor_stcVab	电机电压	
	Motor_stcVabZ		
	Motor_stcVabRef		
stc_dq_t	Motor_stcVdq		
	Motor_stcVdqRef		
	Motor_stcVdqReal		
stc_inc_pid_t	Motor_stcWrPid	转速PI调节	
	Motor_stcIdPid	<i>d</i> 轴电流PI调节	
	Motor_stcIqPid	<i>q</i> 轴电流PI调节	
stc_cmd_speed_set_t	Motor_stcCmdSpdSet	速度参数设置	
stc_speed_traj_t	Motor_stcSpdTraj		
stc_prot_vbus_t	Motor_stcProtVbus	<i>dc</i> 电压保护参数	
stc_prot_oc_t	Motor_stcOverCurrent	过流保护	
stc_prot_lock_rotor_t	Motor_stcLockRotor	堵转保护	
stc_motor_timer_t	Motor_stcTimer	时间事件	
stc_bemf_observer_t	Motor_stcBemfObs	位置、速度估计	
stc_dt_comp_t	Motor_stcDtComp	死区电压估算	
stc_svpwm_calc_t	Motor_stcSvpwm	SVPWM	

结构体类型	结构体名	描述	备注
stc_pwm_gen_t	Motor_stcPwmGen		
stc_motor_run_para_t	Motor_stcRunPara	电机运行参数	
stc_startup_t	Motor_stcStartup	电机启动	
stc_stop_ctrl_t	Motor_stcStopCtrl	电机停机	
stc_auto_run_t	Motor_stcAutoRun	电机自动运行	
stc_periodic_avg_t	Motor_stcVbusAvgF	V_{dc} 电压算术平均滤波	
	Motor_stcSpdAvgF	V_{dc} 电压算术平均滤波	
stc_1stLpf_t	Motor_stcVdLpf	d 轴电压低通滤波	
	Motor_stcVqLpf	q 轴电压低通滤波	
stc_motor_cv_limit_t	Motor_stcLimit	电机电压幅值限制	
stc_brake_ctrl_t	Motor_stcBrake	电机制动控制结构体	

6.2 系统函数

控制系统中涉及的函数情况描述如下：

brake.h

函数名	void Brake_Init(stc_brake_cfg_t stcCfg, stc_brake_ctrl_t* pstc)
描述	制动控制初始化
输入参数	制动控制配置结构体、制动控制结构体指针
返回值	none

函数名	void Brake_Ctrl(int32_t i32Q12_Wr, int32_t i32Q12_Vbus, stc_brake_ctrl_t* pstc)
描述	制动控制
输入参数	转速、Vdc电压、制动控制结构体指针
返回值	none

coordinate_transform.h

函数名	void Transf_Clarke(stc_uvw_t stcuvw, stc_ab_t* pstcab)
描述	Clarke坐标变换
输入参数	abc结构体、alpha-beta结构体指针
返回值	none

函数名	void Transf_InvClarke(stc_ab_t stcab, stc_uvw_t* pstcuvw)
描述	Clarke坐标反变换
输入参数	alpha-beta结构体、abc结构体指针
返回值	none

函数名	void Transf_Park(stc_ab_t stcab, stc_dq_t* pstcdq)
描述	Park坐标变换
输入参数	alpha-beta结构体、dq结构体指针
返回值	none

函数名	void Transf_InvPark(stc_dq_t stcdq, stc_ab_t* pstcab)
描述	Park坐标反变换
输入参数	dq结构体、alpha-beta结构体指针
返回值	none

filter.h

函数名	void Filter_InitFirstLpf(stc_1stLpf_cfg_t stcCfg, stc_1stLpf_t* pstc)
描述	低通滤波初始化
输入参数	低通滤波器配置结构体、低通滤波结构体指针
返回值	none

函数名	int32_t Filter_1stLpf(int32_t i32Q12_Input, stc_1stLpf_t* pstc)
描述	低通滤波
输入参数	原始数据、低通滤波结构体指针

返回值	滤波后的数据
-----	--------

函数名	void Filter_InitPeriodAvg(stc_period_avg_cfg_t stcCfg, stc_period_avg_t* pstc)
描述	算术平均滤波初始化
输入参数	算术平均滤波配置结构体、低通滤波结构体指针
返回值	none

函数名	int32_t Filter_PeriodicAvg(int32_t i32Q12_Input, stc_periodic_avg_t* pstc)
描述	算术平均值滤波
输入参数	原始数据、算术平均值滤波结构体指针
返回值	滤波后的数据

module_math.h

函数名	int32_t Math_Sin(int32_t i32Q24_angle)
描述	三角正弦值函数计算
输入参数	角度
返回值	三角正弦值

函数名	int32_t Math_Cos(int32_t i32Q24_angle)
描述	三角余弦函数计算
输入参数	角度
返回值	三角余弦值

函数名	int32_t Math_Atan2(int32_t i32_x, int32_t i32_y)
描述	反正切值计算
输入参数	x值、y值
返回值	角度

函数名	int32_t Math_Sqrt(int32_t i32_x)
描述	开方运算
输入参数	开方值
返回值	开方结果

observer_bemf.h

函数名	void Bemf_Init(stc_bemf_obs_cfg_t stcCfg, stc_bemf_observer_t* pstc)
描述	观测器初始化
输入参数	观测器配置结构体、观测器结构体指针
返回值	none

函数名	void Bemf_OnlineInit(int32_t i32Q12_Wr, int32_t i32Q24_Theta, stc_bemf_observer_t* pstc)
描述	观测器在线初始化
输入参数	转速、角度、观测器结构体指针
返回值	none

函数名	void Bemf_UpdateBemfLpf(int32_t i32Q12_Wr, stc_bemf_observer_t* pstc)
描述	观测结果滤波
输入参数	转速、观测器结构体指针

返回值	none
-----	------

函数名	void Bemf_Observer(stc_ab_t stcVab, stc_ab_t stcIab, stc_bemf_observer_t* pstc)
描述	位置、速度观测
输入参数	alpha-beta轴电压、alpha-beta轴电流、观测器结构体指针
返回值	none

pid.h

函数名	int32_t PID_IncPid(int32_t i32Q12_Ref, int32_t i32Q12_Fb, stc_inc_pid_t* pstc)
描述	增量式PID
输入参数	当前值、反馈值、增量式PID结构体指针
返回值	PID计算结果

函数名	void PID_InitIncPid(stc_pid_cfg_t stcCfg, stc_inc_pid_t* pstc)
描述	增量式PID初始化
输入参数	PID配置结构体、增量式PID结构体指针
返回值	none

svpwm.h

函数名	boolean_t Svpwm_InitCalc(int32_t i32PeakCnt, int32_t i32Q15_MaxDuty, stc_svpwm_calc_t* pstc)
描述	svpwm初始化
输入参数	最大计数值, SVPWM最大占空比、SVPWM结构体指针
返回值	初始化结果

函数名	void Svpwm_CalcDuration(int32_t i32Q12_Va, int32_t i32Q12_Vb, int32_t i32Q12_Vbus, stc_svpwm_calc_t* pstc)
描述	SVPWM电压矢量作用时间计算
输入参数	alpha电压、beta电压、VDC电压、svpwm结构体指针
返回值	none

函数名	void Svpwm_7SegSymmPwm(uint8_t u8Sector, uint16_t u16T0, uint16_t u16T1, uint16_t u16T2, stc_pwm_gen_t* pstc)
描述	七段SVPWM计数值的计算
输入参数	扇区、零矢量时间、非零矢量时间T1,T2、svpwm开关时间结构体指针
返回值	none

adc_sample.h

函数名	void Adc_InitSample(void)
描述	adc采样初始化
输入参数	none
返回值	none

函数名	void Adc_Sample(stc_adc_sample_t* pstc)
描述	adc采样
输入参数	ADC采样结构体指针

返回值	none
-----	------

dead_time_comp.h

函数名	void DtComp_Init(stc_dt_comp_t* pstc)
描述	死区补偿初始化
输入参数	死区补偿结构体指针
返回值	none

函数名	void DtComp_Calc(int32_t i32Q12_Vbus, stc_uvw_t stcIuvwF, stc_dt_comp_t* pstc)
描述	死区电压计算
输入参数	VDC电压、abc三相滤波电流结构体指针、死区补偿结构体指针
返回值	none

motor_control.h

函数名	void Motor_HwInit(void)
描述	系统硬件初始化
输入参数	none
返回值	none

函数名	void Motor_SwInit(void)
描述	系统软件参数初始化
输入参数	none
返回值	none

函数名	void Motor_InitStartup(void)
描述	电机启动初始化
输入参数	none
返回值	none

函数名	void Motor_InitStop(void)
描述	电机停机初始化
输入参数	none
返回值	none

函数名	void Motor_FastStop(void)
描述	电机快速停机
输入参数	none
返回值	none

函数名	void Motor_ClrRunPara(void)
描述	电机运行参数清除
输入参数	none
返回值	none

函数名	void Motor_CVLimit(void)
描述	电机电流电压限幅

输入参数	none
返回值	none

函数名	void Motor_Process(void)
描述	FOC控制算法
输入参数	none
返回值	none

motor_timer.h

函数名	void Timer_Init(int32_t i32Fs, stc_motor_timer_t* pstc)
描述	软件时间初始化
输入参数	采样频率、软件时间结构体指针
返回值	none

函数名	void Timer_Counter(stc_motor_timer_t* pstc)
描述	软件时间计数
输入参数	软件时间结构体指针
返回值	none

protection.h

函数名	void Prot_InitMonitorVbus(stc_prot_vbus_cfg_t stcCfg, stc_prot_vbus_t* pstc)
描述	监控电压保护初始化
输入参数	监控电压配置结构体，电压保护结构体指针
返回值	none

函数名	void Prot_MonitorVbus(int32_t i32Q12_Vbus, stc_prot_vbus_t* pstc)
描述	V_{dc} 电压保护
输入参数	V_{dc} 电压、 V_{dc} 电压保护结构体指针
返回值	none

函数名	void Prot_InitOverCurrent(stc_prot_oc_cfg_t stcCfg, stc_prot_oc_t* pstc)
描述	过流保护初始化
输入参数	过流保护配置结构体、电流保护结构体指针
返回值	none

函数名	void Prot_MotorOverCurrent(stc_uvw_t stcIuvw, stc_prot_current_t* pstc)
描述	过流流保护
输入参数	abc三相电流、电流保护结构体指针
返回值	none

函数名	void Prot_InitLockRotor(stc_prot_lock_rotor_cfg_t stcCfg, stc_prot_lock_rotor_t* pstc)
描述	电机堵转保护初始化
输入参数	堵转保护配置结构体、堵转保护结构体指针
返回值	none

函数名	void Prot_LockRotor(int32_t i32Q12_WrF, int32_t i32Q12_EdF, int32_t
-----	---

	i32Q12_EqF, stc_prot_lock_rotor_t* pstc)
描述	电机堵转保护
输入参数	滤波转速、d轴滤波反电动势、Q轴滤波反电动势、堵转保护结构体指针
返回值	none

speed_set.h

函数名	void Speed_InitCmdSpdSet(stc_cmd_speed_set_t* pstc)
描述	转速初始化
输入参数	电机转速结构体指针
返回值	none

函数名	void Speed_InitSpdTraj(stc_speed_traj_t* pstc)
描述	电机加减速控制初始化
输入参数	电机加减速控制结构体指针
返回值	none

函数名	void Speed_CmdSpdSet(stc_cmd_speed_set_t* pstc)
描述	电机速度设置
输入参数	电机速度结构体指针
返回值	none

函数名	int32_t Speed_TgtSpdTraj(int32_t i32Q12_CmdWr, stc_speed_traj_t* pstc)
描述	电机加减速控制
输入参数	电机目标转速、转矩结构体指针
返回值	改变后的速度

startup_control.h

函数名	void Startup_Init(stc_startup_t* pstc)
描述	启动控制初始化
输入参数	电机启动结构体指针
返回值	none

函数名	void Startup_Control(stc_startup_t* pstc)
描述	启动控制
输入参数	电机启动控制结构体指针
返回值	none

stop_control.h

函数名	void Stop_Init(stc_stop_ctrl_cfg_t stcCfg, stc_stop_ctrl_t* pstc)
描述	停机控制初始化
输入参数	电机停机控制配置结构体、电机停机控制结构体指针
返回值	none

函数名	void Stop_Control(stc_stop_ctrl_t* pstc)
描述	停机控制
输入参数	电机停机控制结构体指针
返回值	none

init_mcu.h

函数名	void SystemInit(void)
描述	MCU系统初始化
输入参数	none
返回值	none

函数名	void InitMcu_Nvic(void)
描述	中断初始化
输入参数	none
返回值	none

函数名	uint32_t InitMcu_Clock(void)
描述	系统时钟初始化
输入参数	none
返回值	0

函数名	void InitMcu_InitMotorTimer(uint16_t u16_CPSR, uint16_t u16_DeadTimeCnt)
描述	timer4初始化
输入参数	最大计数值、死区时间
返回值	none

函数名	void InitMcu_Adc(void)
描述	ADC初始化
输入参数	none
返回值	none

函数名	void InitMcu_StartMotorTimer(boolean_t bEnable)
描述	定时器启动停止控制
输入参数	使能值 (TRUE/FALSE)
返回值	none

函数名	void InitMcu_SetZeroIsrMaskCnt(uint8_t u8ZIM)
描述	中断屏蔽次数设置
输入参数	中断屏蔽次数
返回值	none

函数名	uint8_t InitMcu_ReadZeroIsrMaskCnt(void)
描述	中断屏蔽次数读取
输入参数	none
返回值	中断屏蔽次数

函数名	void InitMcu_UpdateMotorPwmCycle(uint16_t u16_Cpsr)
描述	定时器计数值更新
输入参数	更新的值
返回值	none

函数名	uint16_t InitMcu_ReadMotorPwmCycle(void)
-----	--

描述	定时器周期读取
输入参数	none
返回值	定时器周期

函数名	void InitMcu_UpdateMotorPwmDuty(uint16_t u16_Occr0, uint16_t u16_Occr1, uint16_t u16_Occr2, uint16_t u16_Occr3, uint16_t u16_Occr4, uint16_t u16_Occr5)
描述	定时器计数值更新
输入参数	timer4的6个通道的计数值
返回值	none

函数名	void InitMcu_EnMotorPwm(boolean_t bEnable)
描述	定时器使能
输入参数	定时器使能/失能
返回值	none

函数名	void InitMcu_Gpio(void)
描述	GPIO初始化
输入参数	none
返回值	none

函数名	void InitMcu_UART(int32_t i32BautRate, M0P_MSC_UART_TypeDef* pUARTx)
描述	串口初始化
输入参数	波特率, 串口结构体指针
返回值	none

7. 系统参数

风机无传感器控制系统中所定义的硬件参数和用户参数，用户在使用该控制系统的时候，需要结合实际情况对这些参数进行修改才能够保证控制系统正常的工作。

7.1 硬件参数

在 h04_user\hardware_config.h 文件中可对 Demo 板上晶振频率，ADC 采样通道等硬件参数进行设置，如图 9 所示。

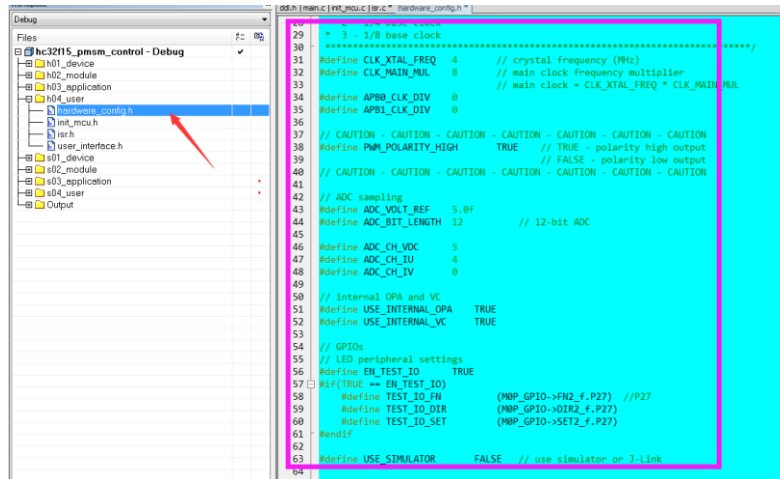


图 9 硬件参数

硬件参数的情况描述如下：

表 8 硬件参数列表

所属部分	宏定义名	描述	值	单位	备注
CLK	CLK_XTAL_FREQ	晶振频率	4	MHz	
	CLK_MAIN_MUL	主时钟倍频系数			主时钟频率 =CLK_XTAL_FREQ* CLK_MAIN_MUL
	APB0_CLK_DIV	APB0分频系数			APB0时钟频率=主时钟 频率>>APB0_CLK_DIV
	APB1_CLK_DIV	APB1分频系数			APB1时钟频率=主时钟 频率>>APB1_CLK_DIV
PWM	PWM_POLARITY_HI GH	默认输出电压极性	TRUE /FALSE		TRUE:high、FALSE:low
ADC	ADC_VOLT_REF	ADC采样参考电压	5	V	
	ADC_BIT_LENGTH	ADC采样位数	12		
	ADC_CH_VDC	V _{adc} 采样通道	5		
	ADC_CH_IU	U相电流采样通道	4		
	ADC_CH_IV	V相电流采样通道	0		
OPA	USE_INTERNAL_OPA	是否使用内部运放	TRUE		
VC	USE_INTERNAL_VC	是否使用内部比较器	TRUE		
GPIO	EN_TEST_IO	LED连接IO口设置	TRUE		TRUE:设置、FALSE:不

所属部分	宏定义名	描述	值	单位	备注
			/FALSE		设置
DEBUG	USE_SIMULATOR	DEBUG模式	TRUE /FALSE		TRUE:simulink、 FALSE:J-Link

7.2 用户参数

文件 s04_user\user_interface.c 用来配置风机无传感器控制系统中所使用的相关用户参数，如图 10 所示，包括采样增益、载波频率、风扇电机转速范围、正常工作电压范围等数据，通过改变这些参数来改变控制系统的性能。

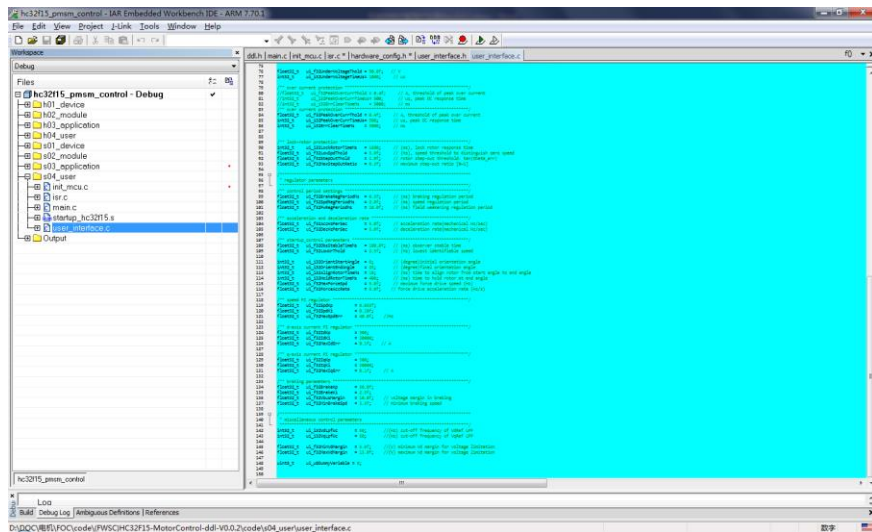


图 10 用户参数

7.2.1 用户参数定义

在风机无传感器控制系统中，用户参数的定义情况如下：

表 9 用户参数列表

所属部分	参数名	描述	单位	备注
载波	ui_i32CarrierFreq	载波频率	Hz	
	ui_f32DeadTimeUs	IGBT开关死区时间	us	
	ui_f32MaxDutyRatio	SVPWM最大占空比		
ADC	ui_i32AdcCheckDelayMs	ADC偏移检测时间	ms	
	ui_f32IuvwSampleRs	电流采样电阻阻值	Ohm	
	ui_f32IuvwSampleK	电流采样增益		
	ui_i32IuvwRefOffset	电流采样参考偏移		
	ui_i32IuvwMaxOffsetBias	电流采样最大偏移误差		
	ui_f32VbusSampleK	V_{ac} 电压采样增益		
电机参数	ui_i32PolePairs	电机极对数		
	ui_f32Rs	电机定子电阻阻值	Ohm	
	ui_f32Ld	电机直轴电感	mH	
	ui_f32Lq	电机交轴电感	mH	
	ui_f32Ke	反电动势系数	V/kRPM	
	ui_f32BaseVoltage	电压标么基值	V	
	ui_f32BaseCurrent	电流标么基值	A	
	ui_f32BaseFrequency	频率标么基值	Hz	
	ui_i32MinSpeedRpm	电机最小转速	RPM	
	ui_i32MaxSpeedRpm	电机最大转速	RPM	
	ui_f32MaxIs	最大电流	A	
DC电压保护	ui_f32AbnormalHighVbus	第一层过压	V	
	ui_f32AbnormalLowVbus	第一层欠压	V	
	ui_i32AbnormalVbusTimeUs	第一层过压/欠压允许时间	us	
	ui_f32OverVoltageThold	第二层过压	V	
	ui_i32OverVoltageTimeUs	第二层过压允许时间	us	
	ui_f32UnderVoltageThold	第二层欠压	V	
	ui_i32UnderVoltageTimeUs	第二层欠压允许时间	us	
过流保护	ui_f32PeakOverCurrThold	最大超调电流	A	
	ui_i32PeakOverCurrTimeUs	电流超调允许时间	us	
	ui_i32ErrClearTimeMs	电流超调标志清除时间	ms	
堵转保护	ui_i32LockRotorTimeMs	堵转响应时间	ms	
	ui_f32LowSpdThold	零速区分阈值	Hz	
	ui_f32StepOutThold	转子步进阈值		
	ui_f32MaxStepOutRatio	最大转子步进阈值比例		
调节参数	ui_f32BrakeRegPeriodMs	制动反馈周期	ms	
	ui_f32SpdRegPeriodMs	速度调节周期	ms	
	ui_f32FwRegPeriodMs	弱磁调节周期	ms	
	ui_f32AccHzPerSec	每秒增加的转速	Hz/sec	
	ui_f32DecHzPerSec	每秒减小的转速	Hz/sec	
	ui_f32ObsStableTimeMs	启动观测稳定时间	ms	
	ui_f32LowWrThold	启动最小转速	Hz	
	ui_i32OrientStartAngle	电机启动角度		
	ui_i32OrientEndAngle	电机启动结束角度		

所属部分	参数名	描述	单位	备注
	ui_i32AlignRotorTimeMs	电机启动完成时间	ms	
	ui_i32HoldRotorTimeMs	电机启动结束时转子保持时间	ms	
	ui_f32MaxForceSpd	电机启动时最大驱动速度	Hz	
	ui_f32ForceAccRate	电机启动时转速增加比例	Hz/s	
	ui_f32SpdKp	速度PI调节Kp系数		
	ui_f32SpdKi	速度PI调节Ki系数		
	ui_f32MaxSpdErr	速度PI调节时速度最大允许误差	Hz	
	ui_f32IdKp	电流PI调节D轴Kp系数		
	ui_f32IdKi	电流PI调节D轴Ki系数		
	ui_f32MaxIdErr	电流PI调节D轴电流最大允许误差	A	
	ui_f32IqKp	电流PI调节Q轴Kp系数		
	ui_f32IqKi	电流PI调节Q轴Ki系数		
	ui_f32MaxIqErr	电流PI调节D轴电流最大允许误差	A	
	ui_f32BrakeKp	制动PI调节Kp系数		
	ui_f32BrakeKi	制动PI调节Ki系数		
	ui_f32VbusMargin	制动PI调节电压裕值		
	ui_f32MinBrakeSpd	制动PI调节最小制动速度		
低通滤波参数	ui_i32VdLpfWc	V_d 低通滤波截止频率	Hz	
	ui_i32VqLpfWc	V_q 低通滤波截止频率	Hz	
	ui_f32MinVdMargin	V_d 电压限幅最大裕值	V	
	ui_f32MaxVdMargin	V_q 电压限幅最小裕值	V	

7.2.2 用户参数计算

在进行用户参数的配置时，若参数未知，如电机参数、ADC 采样增益，此时就需要通过相关的测试和计算获取，具体方法如下。

电机参数计算

在风机无传感器控制中，需要配置的电机参数有极对数 P 、定子电阻 R_s 、直轴电感 L_d 、交轴电感 L_q 、反电动势系数 K_e 这五个参数，它们的计算方法如下（风机类极对数及反电动势系数不能用此方式得出）。

1. 极对数 P

测量方法：手动转动电机转子一周，测试电机三相端子线电压波形，记录正弦波周期数 M 。

计算公式： $P = M$

2. 定子电阻 R_s

测量方法：使用万用表测试电机任意两相之间的电阻 R 。

计算方法： $R_s = R/2$

3. 直轴电感 L_d 、交轴电感 L_q

测量方法：使用 LCR 电桥，设定好频率，调到 L 档，接到电机的其中两根线，用手慢慢旋转电机，并观察测试值 L 。

计算方法：如果是表贴式，则发现电感基本不变，则每相 $L_d = L_q = L/2$ ；如果是内嵌式，则 L_d 小于 L_q ，因此转动过程中测试较大的为 $L_q = L_{max}/2$ ，较小的为 $L_d = L_{min}/2$ 。

4. 反电动势系数 K_e

测量方法：快速转动电机转子，测试电机任意两相端子线电压波形，并选择相对稳定的线电压幅度作为测量范围，记录电压峰峰值 V_{p-p} 和周期 T 。

计算公式： $K_e = \frac{V_{p-p} \cdot P \cdot T}{2 \cdot \sqrt{2} \cdot 60}$

其中， P 为电机极对数。

ADC 采样

ADC 采样部分需要获取的参数有电压采样增益 V_{dck} 、电流采样电阻 R_{sample} 、电流采样增益 I_{uvwk} ，通过硬件原理图（见附录：原理图）可以计算出它们的值，方法如下：

1. 电压采样增益 V_{dck}

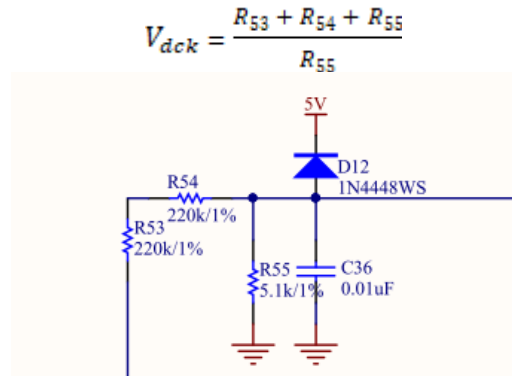


图 11 为 V_{dc} 电压采样电路，其对应的电压采样增益 V_{dck} 的计算方法为：

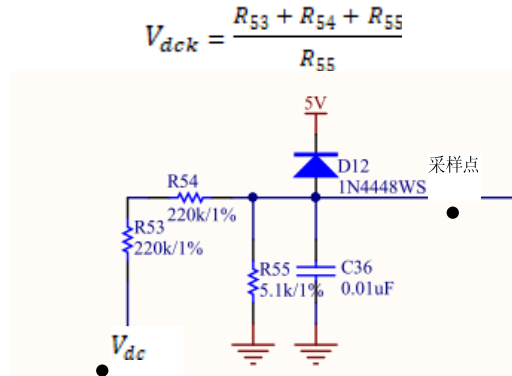


图 11 V_{dc} 电压采样电路

2. 电流采样电阻 R_{sample}

原理图中的采样电阻为 R_{76} 、 R_{77} 、 R_{79} ，设置为它们的阻值。

3. 电流采样增益 I_{uvwk} 、偏置电压 V_{offset}

芯片内部集成了放大器，只需要通过在外围搭建外围电路来控制放大增益便可，其外围电路如图 12 所示，电流采样电路对应的完整电路如图 13 所示。电流采样增益 I_{uvwk} 的计算公式为：

$$I_{uvwk} = \frac{R_{64} * R_{66} * (R_{62} + R_{68} + R_{72})}{(R_{68} + R_{72}) * ((R_{69} + R_{73}) * (R_{66} + R_{64}) + R_{64} * R_{66})} \approx \frac{R_{62}}{R_{68} + R_{72}}$$

偏置电压 V_{offset} 为电流为 0 时， V_{out} 所对应的电压，其计算公式如下：

$$V_{out} = \frac{(R_{72} + R_{68} + R_{62})V^-}{R_{72} + R_{68}}$$

备注：在电路工作正常的情况下，图 13 中存在如下关系 $V^- = V^+$ ， $V_{out} = 2.5V$ 。

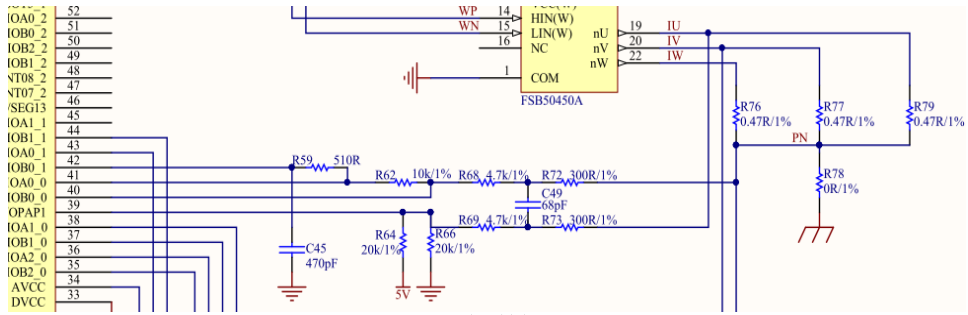


图 12 电流采样电路外围电路

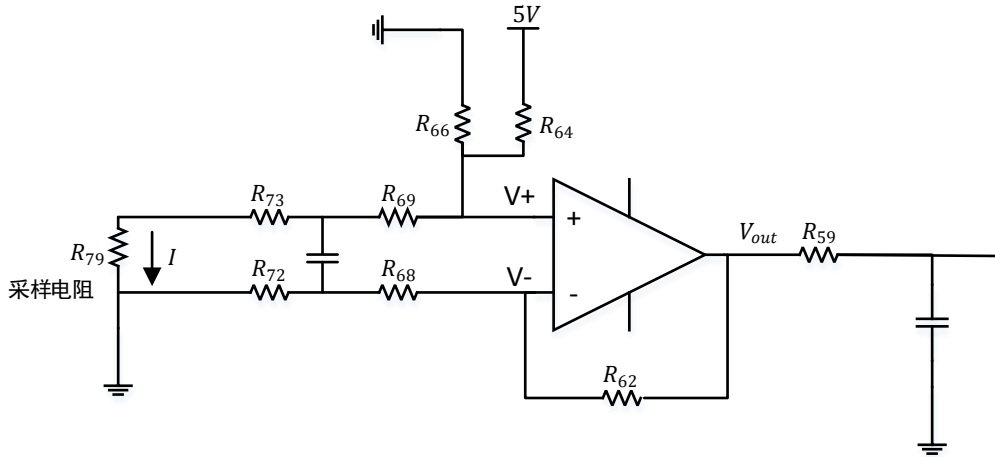


图 13 电流采样电路

8. 系统数据处理

由于 MCU 处理浮点数据效率不高，在风机无传感器控制方案中将浮点数据转换为定点数，即 Q 格式化数据，来提高 MCU 的数据处理效率，为了防止数据在运算时出现溢出的情况，在进行定点数转换前，采用标幺的形式对数据进行处理，待数据处理完成后再将该数据转换为定点数。

8.1 数据标幺

8.1.1 数据标幺的方法

数据标幺是进行数据处理的第一步，标幺的方法如下：

令 x 为待标幺变量， $x_B = \text{const}$ 为变量基值，则数据 x 的标幺定义为：

$$x' = \frac{x}{x_B}$$

其中 x' 为标幺后的变量。例如取 x_B 为 $\max[\text{abs}(x)]$ ，则标幺后的数据 $x' \in [-1 \ 1]$

永磁同步电机中物理量的标幺可通过定义三个独立的基值实现，其他变量则可通过这三个独立基值确定。本系统选取电压、电流、和角频率作为系统基值，分别定义为：

- V_B ：电压基值(V)，定义为额定相电压幅值
- I_B ：电流基值(A)，定义为额定相电流的幅值
- ω_B ：转子角频率基值(rad/s)，定义为转子的额定电转速

根据电压、电流、和角频率，可推导出 PMSM 中的其它物理量基值分别为：

- 电阻基值： $R_B = \frac{V_B}{I_B}$
- 电感基值： $L_B = \frac{V_B}{\omega_B I_B}$
- 磁链基值： $\lambda_B = \frac{V_B}{\omega_B}$
- 时间基值： $T_B = \frac{1}{\omega_B}$
- 功率基值： $S_B = \frac{3}{2} V_B I_B$
- 机械转速基值： $\Omega_B = \frac{\omega_B}{P}$ ，其中 P 为电机极对数
- 转矩基值： $\Gamma_B = \frac{3P}{2} \frac{V_B I_B}{\omega_B}$

8.1.2 标幺基值的设定

由 8.1.1 可知，在进行数据标幺时，基值的选取较为关键，只有基值选取合适，才能够保证在转换为定点数后进行数据处理时既不会溢出，同时也能够保证计算的准确性。设定的标幺基值只要能够满足表 10 中所示的标幺后数据的范围即可。

8.2 Q 格式化数据

8.2.1 Q 格式化方法

Q 格式数据将浮点数以定点数的形式存储，常用于不支持浮点运算的芯片应用中。对于 32 位有符号数的 QN 格式数据，系统存储的数据以原始数据乘以 2^N 后取整保存，如：

$$QN(x) = \text{round}(x \cdot 2^N)$$

此时，数据的分辨率为 2^{-N} ，可表示数据范围 $[-2^{31-N}, 2^{31-N} - 1]$ 。例如 Q15 格式保存的数据分辨率为 $2^{-15} = 0.0000305$ ，数据范围为 $[-65536, 65535]$ 。

8.2.2 Q 格式化数据范围

考虑 8.1 节中讨论的标幺化系统，在风扇无传感器控制系统中，标幺后系统物理量的数据范围约为 $[-2, 2]$ ，为提高数据精度，保证运算中乘法不溢出，并预留足够裕度，标幺后的数据范围及其 QN 格式数据如所示。

表 10 主要参数 QN 格式定义

变量	类型	Q格式	标幺后数据范围
电流	int32_t	Q12	-8 ~ 8
电压	int32_t	Q12	-8 ~ 8
频率	int32_t	Q12	-8 ~ 8
角度	int32_t	Q24	-128 ~ 127

9. 参数设定与程序编译下载

本节主要对下载程序前的相关准备以及如何进行程序编译下载。

9.1 实物连接

准备好所需设备，如图 14 所示：

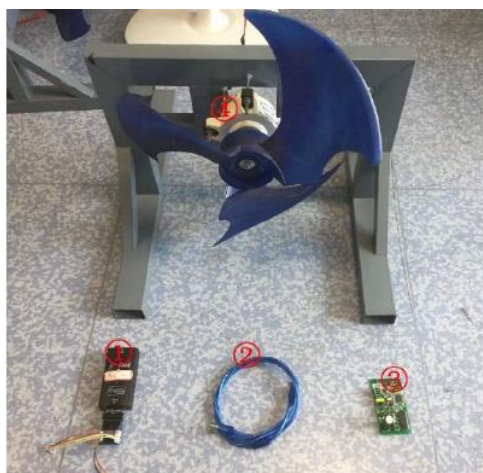


图 14 所需物品

在图 14 中共有 4 样设备：

①JLINK V9 隔离仿真器、②下载线、③电机驱动板、④PMSM

按照 PMSM-电机驱动板- JLINK V9 隔离仿真器-下载线-电脑的顺序连接好各设备，其中，电机驱动板各个接口如图 15 所示：

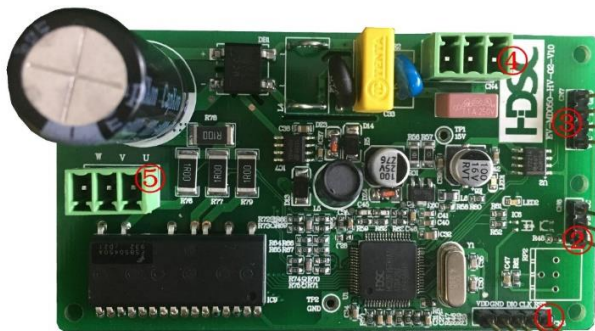


图 15 驱动板各个接口

如图 15 所示，电机驱动板上一共有 5 个接口，他们分别是：

- 1: 程序下载口和调试口
- 2: PWM/PWC 外部调速接口
- 3: UART 接口
- 4: 电源接口
- 5: PMSM 接口

其连接图如下：

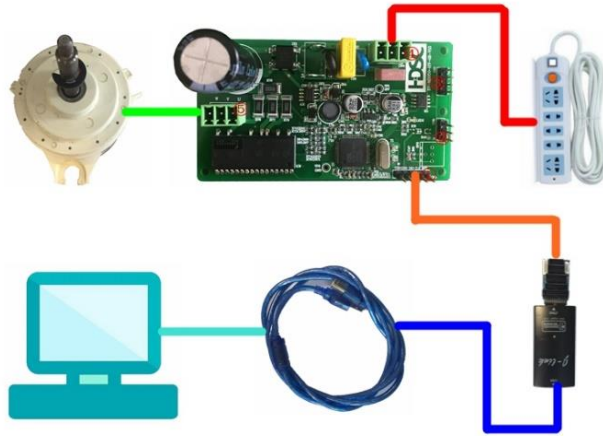


图 16 连接示意图

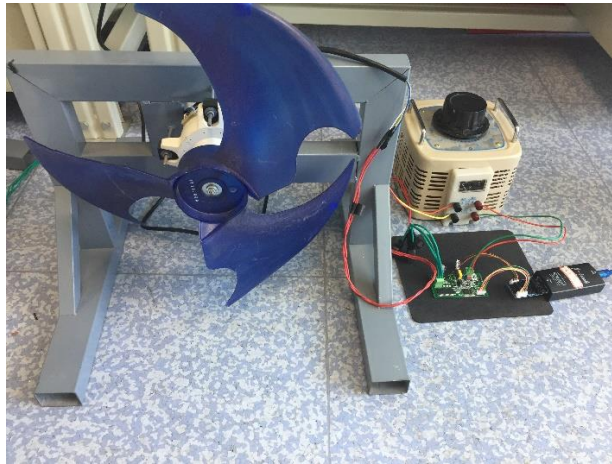


图 17 实物连接图

9.2 打开工程

1、打开..\editor\EWARM 文件夹，找到 HC32M14_pmsm_control.eww 文件。

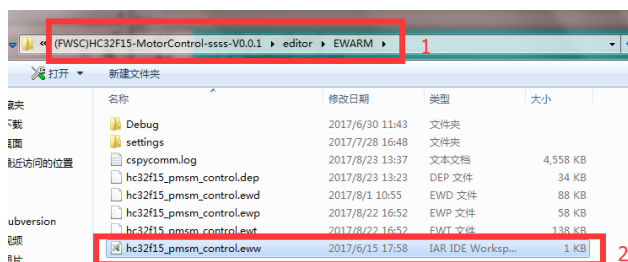


图 18 工程文件所处位置

2、使用 IAR EWARM 打开 HC32M14_pmsm_control.eww 文件。

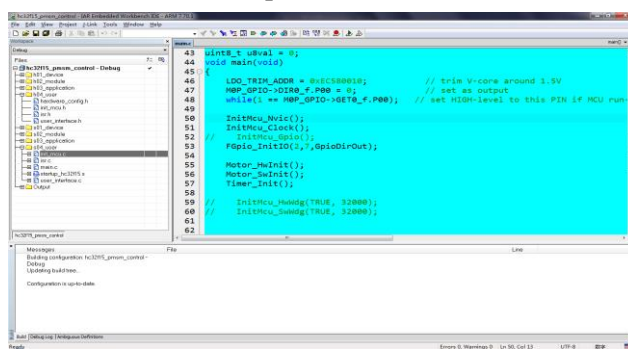


图 19 工程打开效果

9.3 用户配置

在进行程序下载前需要根据电机和硬件的实际情况进行相关配置，才能保证电机能够正常运行。

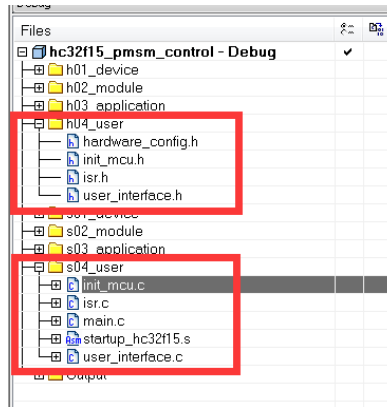


图 20 用户配置文件位置分布

图 20 为相关配置文件所处的位置，用户进行配置的时候，主要是配置 MCU 系统中断参数（s04_user\isr.c），硬件参数（h04_user\hardware_config.h），电机参数以及程序运行时的相关变量的相关参数（s04_user\user_interface.c）。

9.3.1 中断参数配置

中断参数配置需要设置的地方有：

- 1、风机无传感器控制系统中所使用到的中断的中断优先级
- 2、使用到的中断的使能

注意：

- 详细讲解请参考章节系统中断部分。

9.3.2 硬件参数配置

硬件参数配置部分主要是根据无传感器控制系统所使用的硬件来进行设置，其配置在文件 h04_user\hardware_config.h 中，如[错误!未找到引用源。](#)所示。主要配置以下几个地方：

1. 晶振频率：驱动板上所使用的晶振的频率（单位：MHz）
2. 主时钟倍频系数
3. APB0 时钟分频系数
4. APB1 时钟分频系数
5. PWM 输出极性设置
6. ADC 参考电压设定（单位：V）
7. ADC 采样位数设置
8. 采样通道设置（ V_{dc} 、U 相电流、V 相电流）

注意：

- 硬件参数的详细讲解，请参考章节硬件参数。

```
#define CLK_XTAL_FREQ      4      // crystal frequency (MHz)
#define CLK_MAIN_MUL      6      // main clock frequency multiplier
// main clock = CLK_XTAL_FREQ * CLK_MAIN_MUL
#define APB0_CLK_DIV      0
#define APB1_CLK_DIV      0
#define PWM_POLARITY_HIGH TRUE // TRUE - polarity high output
                                // FALSE - polarity low output

// ADC sampling
#define ADC_VOLT_REF      5.0f
#define ADC_BIT_LENGTH    12     // 12-bit ADC
#define ADC_CH_VDC        5
#define ADC_CH_IU         4
#define ADC_CH_IV         0
```

图 21 硬件参数

9.3.3 电机运行相关参数的配置

电机运行相关参数位于文件 `s04_user\user_interface.c` 中，相关参数的描述请参考章节用户参数定义。

在工程中已经提前设计好电机运行相关的参数，用户主要考虑进行设置的有以下几个部分：(1) 载波频率、死区等 `Timer4` 参数设置、(2) 标么基值部分、(3) 电机参数部分、(4) `ADC` 采样部分 (5) 电压限制部分、(6) 电流限制部分、(7) 转速限制部分。

这七个部分需要根据设计情况进行修改，其他参数可根据需要酌情修改，或者直接使用默认参数即可（建议用户使用默认参数）。它们的具体情况介绍如下：

a) 载波频率、死区等 `Timer4` 参数设置

这部分参数主要是和 `Timer4` 寄存器相关，配置这些参数可以设定控制系统的周期，功率管进行开关切换的时候死区时间，逆变器最大输出电压。如错误!未找到引用源。所示。

```
int32_t ui_i32CarrierFreq    = 10000; // (Hz) carrier frequency of PWM
float32_t ui_f32DeadTimeUs   = 2.0f;  // (us) dead time of IPM
float32_t ui_f32MaxDutyRatio = 0.90f; // maximum duty ratio of SVPWM
```

图 22 定时器参数

错误!未找到引用源。中所示的三个参数与定时器相关，它们的意义如下：

变量名	单位	设定范围	作用
<code>ui_i32CarrierFreq</code>	Hz	8~16K	载波频率
<code>ui_f32DeadTimeUs</code>	us	>1us	死区时间
<code>ui_f32MaxDutyRatio</code>		< 0.95	SVPWM最大占空比

b) 标么基值部分

在控制程序中，使用定点数来处理数据，为了在将数据转换为定点数后不会溢出，需要在进行转换前将数据标么。数据标么需要进行基值的设定，在控制系统中，需要对电压、电流以及频率这三个量的标么基值进行设定，表么基值的设定在文件 `s04_user\user_interface.c` 中，电压标么基值、电流标么基值、角频率标么基值对应的用户参数分别为：`ui_f32BaseVoltage`、`ui_f32BaseCurrent`、`ui_f32BaseFrequency`（建议用户使用默认的标么基值），如图 23 所示。

```
float32_t ui_f32BaseVoltage = 100.0f; // (V) base voltage
float32_t ui_f32BaseCurrent = 0.5f; // (A) base current
float32_t ui_f32BaseFrequency = 100.0f; // (Hz) base frequency
```

图 23 标幺基值参数

如何对数据进行标幺，以及标幺基值如何选取请参考章节系统数据处理。

c) 电机参数部分

电机参数部分需要设定的量有电机的极对数、定子电阻、直轴电感、交轴电感、反电动势系数，它们分别对应的变量名为：`ui_i32PolePairs`、`ui_f32Rs`、`ui_f32Ld`、`ui_f32Lq`、`ui_f32Ke`，如图 24 所示。

```
int32_t ui_i32PolePairs = 4;
float32_t ui_f32Rs = 26.5f; // Ohm
float32_t ui_f32Ld = 170.0f; // mH
float32_t ui_f32Lq = 170.0f; // mH
float32_t ui_f32Ke = 87.9f; // V/kRPM
```

图 24 电机参数

在设置电机参数的时候，若用户已经知道所使用电机的参数，则只需要直接对 `s04_user\user_interface.c` 文件中的电机参数进行修改即可。若电机参数未知，用户需要通过测量计算出电机的参数，然后再进行参数的修改，电机参数的测量和计算方法请参考章节电机参数计算。

d) ADC 采样部分

ADC 采样部分需要设置的量有电流采样电阻阻值、电流采样增益、电压采样增益，它们分别对应的用户参数为：`ui_f32VbusSampleK`、`ui_f32IuvwSampleRs`、`ui_f32IuvwSampleK`，如图 25 所示。

```
float32_t ui_f32IuvwSampleRs = 1.00f; // (Ohm) sample resistance
float32_t ui_f32IuvwSampleK = 2.0f; // current sample gain
float32_t ui_f32VbusSampleK = 87.2745f; // DC bus voltage sample gain
```

图 25 ADC 采样参数

关于 ADC 采样部分参数的设置方法请参考章节 ADC 采样。

e) 电压限制部分

电压限制部分需要设定参数有第一层过压 `ui_f32AbnormalHighVbus`、第一层欠压 `ui_f32AbnormalLowVbus`、第二层过压 `ui_f32OverVoltageThold`、第二层欠压 `ui_f32UnderVoltageThold` 这四个量，如图 26 所示，用户在设定的时候需要根据系统运行时的电压情况进行设置。

```
float32_t   ui_f32AbnormalHighVbus      = 380.0f; // V
float32_t   ui_f32AbnormalLowVbus       = 180.0f; // V
float32_t   ui_f32OverVoltageThold      = 360.0f; // V
float32_t   ui_f32UnderVoltageThold     = 200.0f; // V
```

图 26 电压限制参数

f) 电流限制部分

电流限制部分主要用来设置最大的工作电流，如图 27 所示。

```
float32_t   ui_f32MaxIs                 = 0.20f; // (A) maximum operation
```

图 27 电流限制参数

g) 转速限制部分

转速限制部分主要用来设置电机的转速范围，如图 28 所示。

```
int32_t     ui_i32MinSpeedRpm          = 200; // (RPM) minimum speed
```

图 28 转速限制参数

9.4 代码编译

点击 IAR EWARM 选中工程名: HC32M14_pmsm_control-Debug,鼠标右键调出工程选项栏如图 29 所示,选择菜单栏中的 Clean 选项,待 build 信息栏出现图 30 所示效果后,再次调出图 29 中的菜单栏,点击 Rebuild All 选项进行代码的编译,编译完成后,build 信息栏会出现图 31 效果,并且 build 信息栏中显示 errors 和 warning 的数量均为 0,表明程序没有错误警告,可以进行程序的下载操作。

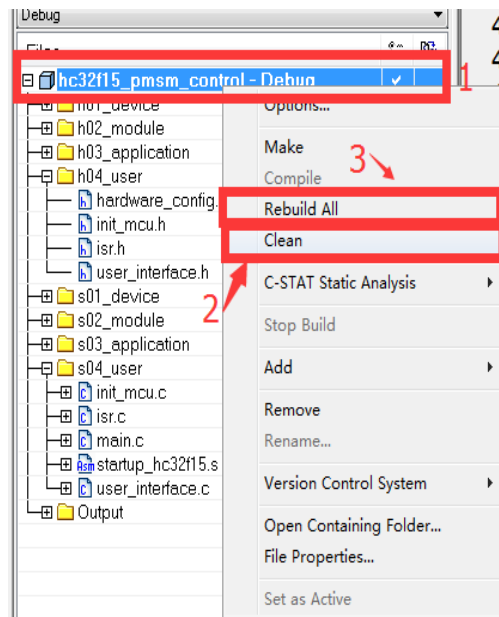


图 29 代码编译

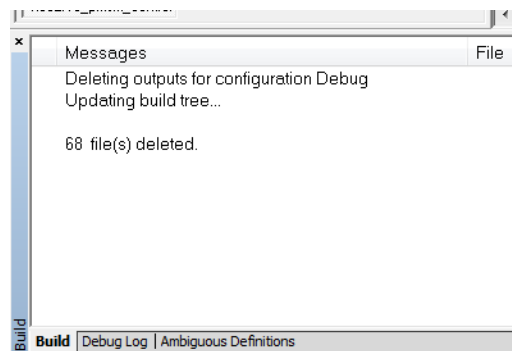


图 30 编译 Clean 效果

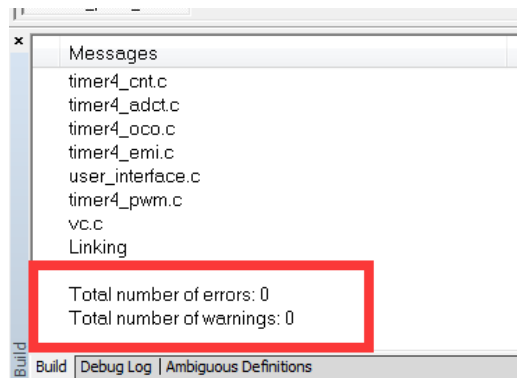



图 31 编译结果

9.5 代码下载

点击菜单栏中的下载编译按钮，进行程序的下载，下载完成后出现图 32 效果。

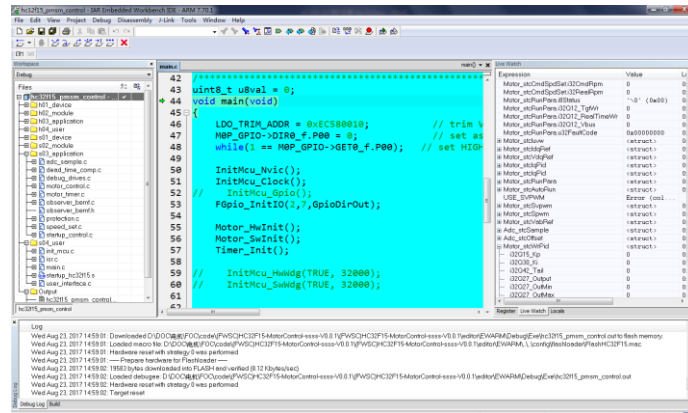


图 32 代码下载效果

9.6 程序运行

点击快速运行按钮运行程序进入 Debug 模式，运行后的效果如图 33 所示。

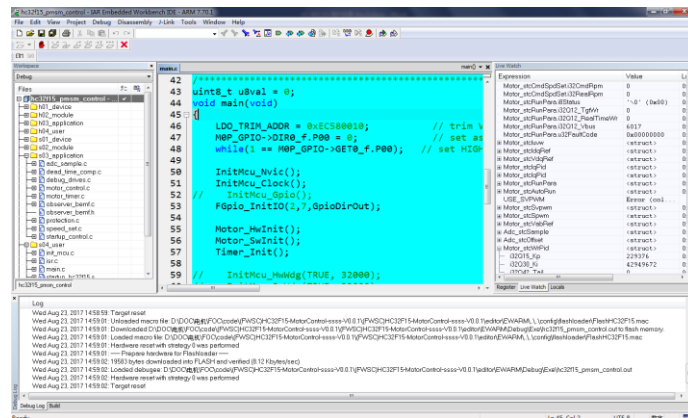


图 33 程序运行效果

9.7 电机运行控制以及运行时参数观测

1. 为了保证安全，程序默认状态下电机转速为 0，需要在 Debug 模式下进行速度的设置。在 View 菜单栏中点击 Live Watch，如图 34 所示，调出 Live Watch 菜单栏，调出后的效果如图 35 所示。

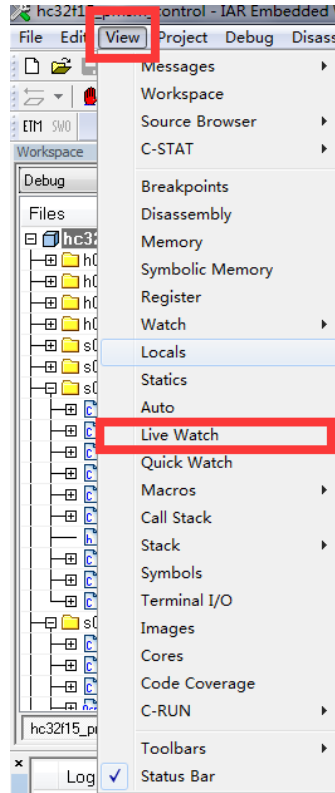


图 34 Live Watch

Expression	Value	Location	Type
Motor_stcCmdSpdSet.i32CmdRpm	0	0x20000614	int32_t
Motor_stcCmdSpdSet.i32RealRpm	0	0x20000618	int32_t
Motor_stcRunPara.i8Status	'\0' (0x00)	0x20000764	int8_t
Motor_stcRunPara.i32Q12_TgtWr	0	0x2000074C	int32_t
Motor_stcRunPara.i32Q12_RealTimeWr	0	0x20000750	int32_t
Motor_stcRunPara.i32Q12_Vbus	0	0x2000076C	int32_t
Motor_stcRunPara.u32FaultCode	0x00000000	0x20000760	uint32_t
Motor_stcluvw	<struct>	0x200007E4	stc_uvw
Motor_stcdldqRef	<struct>	0x200006D0	stc_dq
Motor_stcVdqRef	<struct>	0x20000680	stc_dq
Motor_stclqPid	<struct>	0x20000530	stc_inc
Motor_stclqPid	<struct>	0x20000530	stc_inc
Motor_stcRunPara	<struct>	0x20000748	struct <
Motor_stcAutoRun	<struct>	0x20000664	stc_aut
USE_SVPWM	Error (col...		
Motor_stcSvpwm	<struct>	0x200006F0	stc_svp
Motor_stcSpwm	<struct>	0x20000710	stc_spm
Motor_stcVabRef	<struct>	0x20000580	stc_ab
Adc_stcSample	<struct>	0x20000690	stc_ad
Adc_stcOffset	<struct>	0x20000880	stc_ad

图 35 Live Watch 窗口信息栏

2. 在 Live Watch 中增加在程序中所定义的相关变量，如图 35 所示。
3. 改变目标转速的值（Motor_stcCmdSpdSet.i32CmdRpm）为 200，可以看到电机开始转动，并且在 Live Watch 信息栏中电机运行时的相关参数也被实时显示出来，如电机的实时转速（Motor_stcCmdSpdSet.i32RealRpm）、Vdc 的电压值（Motor_stcRunPara.i32Q12_Vbus）。如图 36 所示。

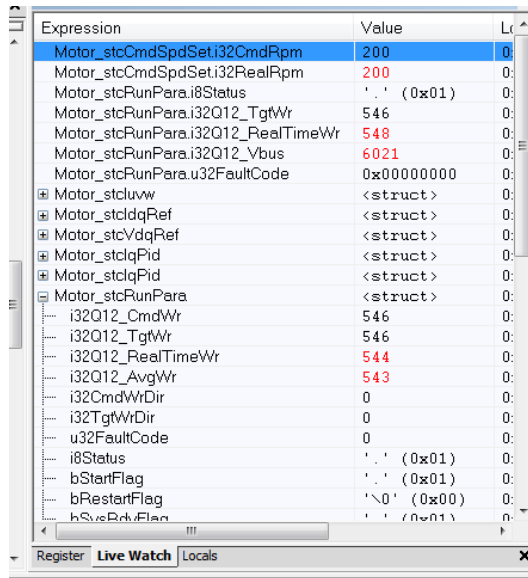



图 36 Live Watch 数据改变

注意：

- 在进行电机的调试时，需要观测电机运行时的参数，对电机的运行是否正常、程序出错时问题所在之处等一系列情况的观测尤为重要，如图 36 中通过 `Motor_stcCmdSpdSet.i32RealRpm` 的值我们能够知道电机的实时转速。所以在进行观测参数时只需要在 Live Watch 栏中输入相应的变量名，便能够知道该变量当前的值，帮助我们进行调试。在风扇无传感器控制中，所有参数均包含在结构体中，关于结构体的详细讲解请参考章节结构体。

9.8 调试退出

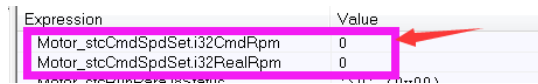
为了安全起见，在退出调试前重新设置 `Motor_stcCmdSpdSet.i32CmdRpm` 的值为 0 停止电机，然后点击退出按钮  退出调试。

10. 实例讲解

本章对风机无传感控制方案中功能的实现步骤进行简单介绍，具体情况如下。

10.1 速度控制

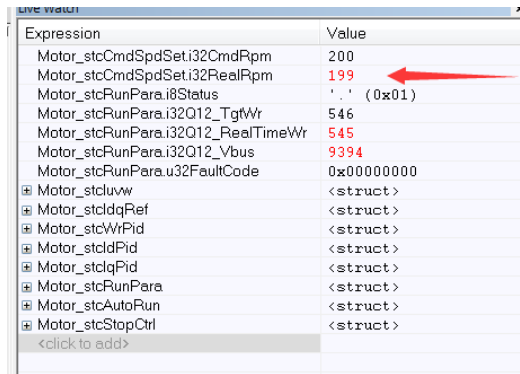
1. 按照章节 9.1 准备好所需要的实物。
2. 若电机参数未知，则按章节 7.2.2 所述，完成对电机的极对数 P 、定子电阻 R_s 、直轴电感 L_d 、交轴电感 L_q 、反电动势系数 K_e 的测试。若电机参数已知，则直接跳过步骤 2。
3. 按照章节 9.1 连接好实物。
4. 按照章节 9.2 到章节 9.6 所述，完成相关参数的配置修改、程序的下载编译运行。
5. 按照章节 9.7 调出 Live Watch 菜单栏，并在菜单栏中增加变量 Motor_stcCmdSpdSet.i32CmdRpm(目标转速)、Motor_stcCmdSpdSet.i32RealRpm(实际转速)，如图 37 所示。



Expression	Value
Motor_stcCmdSpdSet.i32CmdRpm	0
Motor_stcCmdSpdSet.i32RealRpm	0

图 37 速度控制相关变量

6. 将 Motor_stcCmdSpdSet.i32CmdRpm 的值设置为 200，电机开始转动，Motor_stcCmdSpdSet.i32RealRpm 的值逐渐增加，并快速接近设定的目标转速值，如图 38 所示。



Expression	Value
Motor_stcCmdSpdSet.i32CmdRpm	200
Motor_stcCmdSpdSet.i32RealRpm	199
Motor_stcRunPara.i8Status	'.' (0x01)
Motor_stcRunPara.i32Q12_TgtWr	546
Motor_stcRunPara.i32Q12_RealTimeWr	545
Motor_stcRunPara.i32Q12_Vbus	9394
Motor_stcRunPara.u32FaultCode	0x00000000
Motor_stcluww	<struct>
Motor_stcldqRef	<struct>
Motor_stcWrPid	<struct>
Motor_stcldPid	<struct>
Motor_stclqPid	<struct>
Motor_stcRunPara	<struct>
Motor_stcAutoRun	<struct>
Motor_stcStopCtrl	<struct>
<click to add>	

图 38 转速 200RPM

7. 将 Motor_stcCmdSpdSet.i32CmdRpm 的值设置为 300，电机以更快的速度转动，Live Watch 菜单栏中相关数据的变化情况如图 39 所示。同时也可以通过示波器观测到电机其中一相电流在由 200 改变到 300 以后，其电流的幅值和频率更大，如图 40 所示。

Expression	Value
Motor_stcCmdSpdSet.i32CmdRpm	300
Motor_stcCmdSpdSet.i32RealRpm	300
Motor_stcRunPara.i8Status	'.' (0x01)
Motor_stcRunPara.i32Q12_TgtWr	819
Motor_stcRunPara.i32Q12_RealTimeWr	822
Motor_stcRunPara.i32Q12_Vbus	9346
Motor_stcRunPara.u32FaultCode	0x00000000
Motor_stcluww	<struct>
Motor_stcldqRef	<struct>
Motor_stcWrPid	<struct>
Motor_stcldPid	<struct>
Motor_stcldqPid	<struct>
Motor_stcRunPara	<struct>
Motor_stcAutoRun	<struct>
Motor_stcStopCtrl	<struct>
<click to add>	

图 39 转速 300RPM

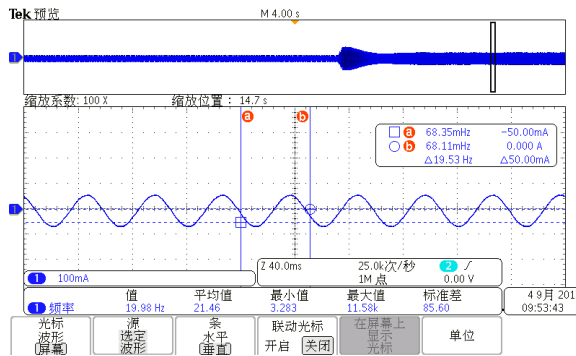


图 40 电流变化情况

8. 待速度控制调试完成后，按照章节 9.8 所述，先将转速设置为 0，然后退出调试界面。

10.2 转向控制

1. 按照章节速度控制中的步骤 1 到步骤 5 完成实物连接、参数设置、程序下载等一些列操作。
2. 设定转速为 200，待电机真实转速稳定后，改变转速为-200，可以看到电机开始反方向进行旋转，并逐渐趋于目标转速，如图 41 所示。同时，也可以通过示波器观测到电机其中一相的电流变化情况。如图 42 所示。

Expression	Value
Motor_stcCmdSpdSet.i32CmdRpm	-200
Motor_stcCmdSpdSet.i32RealRpm	-199
Motor_stcRunPara.i8Status	' ' (0x01)
Motor_stcRunPara.i32Q12_TgtWr	546
Motor_stcRunPara.i32Q12_RealTimeWr	545
Motor_stcRunPara.i32Q12_Vbus	8975
Motor_stcRunPara.u32FaultCode	0x00000000
Motor_stcluvw	<struct>
Motor_stcldqRef	<struct>
Motor_stcWrPid	<struct>
Motor_stcldPid	<struct>
Motor_stclqPid	<struct>
Motor_stcRunPara	<struct>
Motor_stcAutoRun	<struct>
Motor_stcStopCtrl	<struct>
<click to add>	

图 41 电机转速-200RPM

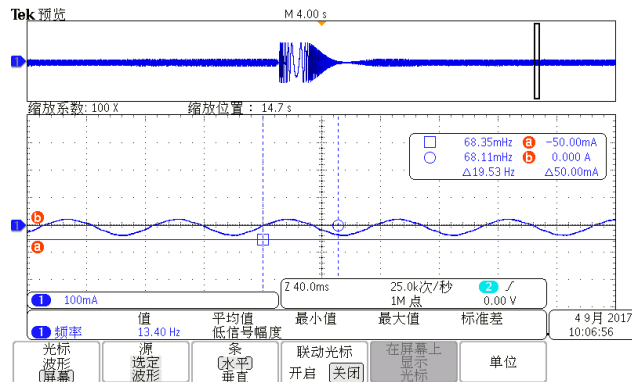
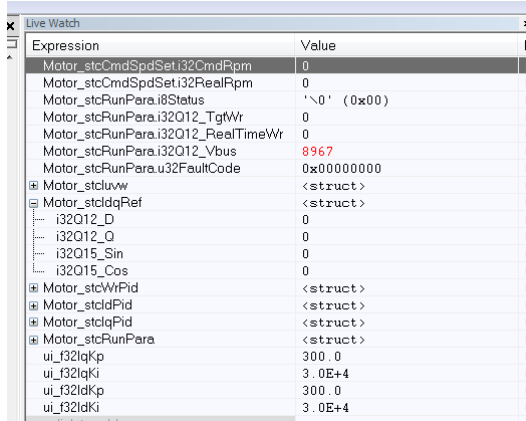


图 42 转向时电流变化情况

3. 待转向控制调试完成后，按照章节调试退出所述，先将转速设置为 0，然后退出调试界面。

10.3 PI 参数调节

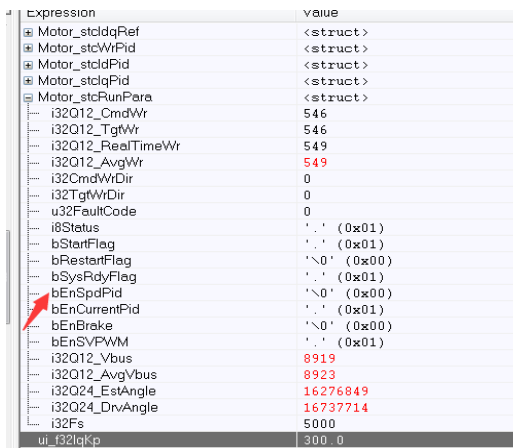
- 按照章节速度控制中的步骤 1 到步骤 4 完成实物连接、参数设置、程序下载等一系列操作。
- 在 Live Watch 菜单栏中增加 `ui_f32IdKp`、`ui_f32IdKi`、`ui_f32IqKp`、`ui_f32IqKi`、`Motor_stcIdqRef`，如图 43 所示。



Expression	Value	L
Motor_stcCmdSpdSet.i32CmdRpm	0	0
Motor_stcCmdSpdSet.i32RealRpm	0	0
Motor_stcRunPara.i8Status	'\0' (0x00)	0
Motor_stcRunPara.i32Q12_TgtWr	0	0
Motor_stcRunPara.i32Q12_RealTimeWr	0	0
Motor_stcRunPara.i32Q12_Vbus	8967	0
Motor_stcRunPara.u32FaultCode	0x00000000	0
Motor_stcIdqRef	<struct>	0
Motor_stcIdqRef.i32Q12_D	0	0
Motor_stcIdqRef.i32Q12_Q	0	0
Motor_stcIdqRef.i32Q15_Sin	0	0
Motor_stcIdqRef.i32Q15_Cos	0	0
Motor_stcWrPid	<struct>	0
Motor_stcIdPid	<struct>	0
Motor_stcIqPid	<struct>	0
Motor_stcRunPara	<struct>	0
ui_f32IqKp	300.0	0
ui_f32IqKi	3.0E+4	0
ui_f32IdKp	300.0	0
ui_f32IdKi	3.0E+4	0

图 43 电流 PI 调节参数

- 以 Q 轴 PI 控制为例，设定转速为 0，然后改变 Live Watch 菜单栏中 `ui_f32IqKp`、`ui_f32IqKi` 的值。如图 43 所示。
- 设定目标转速为 200，待电机运行一会后将结构体 `Motor_stcRunPara` 中的成员 `bEnSpdPid` 的值设置为 0，失能速度 PI 控制，如图 44 所示。



Expression	Value
Motor_stcIdqRef	<struct>
Motor_stcWrPid	<struct>
Motor_stcIdPid	<struct>
Motor_stcIqPid	<struct>
Motor_stcRunPara	<struct>
Motor_stcRunPara.i32Q12_CmdWr	546
Motor_stcRunPara.i32Q12_TgtWr	546
Motor_stcRunPara.i32Q12_RealTimeWr	549
Motor_stcRunPara.i32Q12_AvgWr	549
Motor_stcRunPara.i32CmdWrDir	0
Motor_stcRunPara.i32TgtWrDir	0
Motor_stcRunPara.u32FaultCode	0
Motor_stcRunPara.i8Status	'\0' (0x01)
Motor_stcRunPara.bStarFlag	'\0' (0x01)
Motor_stcRunPara.bRestartFlag	'\0' (0x00)
Motor_stcRunPara.bSysPdyFlag	'\0' (0x01)
Motor_stcRunPara.bEnSpdPid	'\0' (0x00)
Motor_stcRunPara.bEnCurrentPid	'\0' (0x01)
Motor_stcRunPara.bEnBrake	'\0' (0x00)
Motor_stcRunPara.bEnSVPWM	'\0' (0x01)
Motor_stcRunPara.i32Q12_Vbus	8919
Motor_stcRunPara.i32Q12_AvgVbus	8923
Motor_stcRunPara.i32Q24_EstAngle	16276849
Motor_stcRunPara.i32Q24_DrvAngle	16737714
Motor_stcRunPara.i32Fs	5000
ui_f32IqKp	300.0

图 44 速度 PI 控制失能

- 改变 `Motor_stcIdqRef` 结构体中变量 `i32Q12_Q` 的值为 200，运行一段时间后，改变 `i32Q12_D` 的值为 800（该值由实际情况决定，不能超过系统所能承受的最大电流，电流的改变由小电流突然变为大电流），如图 45 所示。待 `i32Q12_Q` 的值改变后，通过示波器观察电流波形的变化情况，如图 46 所示。

Expression	Value	L
Motor_stcCmdSpdSet.i32CmdRpm	200	0
Motor_stcCmdSpdSet.i32RealRpm	403	0
Motor_stcRunPara.i8Status	'.' (0x01)	0
Motor_stcRunPara.i32Q12_TgWr	546	0
Motor_stcRunPara.i32Q12_RealTimeWr	1104	0
Motor_stcRunPara.i32Q12_Vbus	8823	0
Motor_stcRunPara.u32FaultCode	0x00000000	0
Motor_stcIuvw	<struct>	0
Motor_stcIqRef	<struct>	0
i32Q12_D	0	0
i32Q12_Q	800	0
i32Q15_Sin	0	0
i32Q15_Cos	0	0
Motor_stcWrPid	<struct>	0
Motor_stcIdPid	<struct>	0
Motor_stcIqPid	<struct>	0
Motor_stcRunPara	<struct>	0
ui_f32IqKp	300.0	0
ui_f32IqKi	3.0E+4	0
ui_f32IdKp	300.0	0
ui_f32IdKi	3.0E+4	0
<click to add>		

图 45 电流 PI 控制电流改变

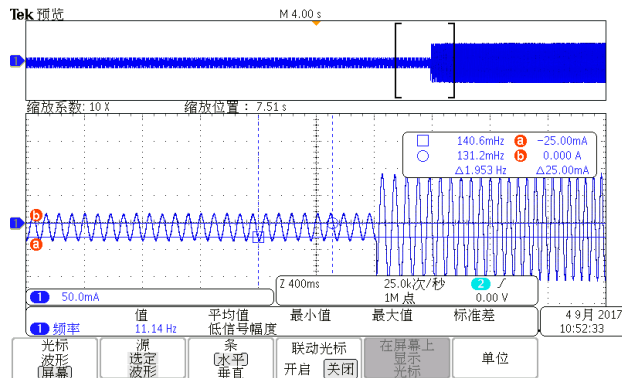


图 46 电流 PI 参数调节电流变化情况

6. 重复步骤 3 到步骤 5，直到电流的变化情况满足预期，退出调试，并将调试时 ui_f32IqKp、ui_f32IqKi 所使用的值写入到 s04_user\user_interface.c 中的对应位置，如图 47 所示。

```

/** d-axis current PI regulator *****/
float32_t ui_f32IdKp = 300;
float32_t ui_f32IdKi = 30000;
/** q-axis current PI regulator *****/
float32_t ui_f32IqKp = 300;
float32_t ui_f32IqKi = 30000;

```

图 47 电流 PI 参数设定

注意：

- D 轴 PI 参数调节请参考 Q 轴 PI 调节。

10.4 速度 PI 参数调节

- 待电流 PI 调节完成后，按照章节速度控制中的步骤 1 到步骤 4 完成实物连接、参数设置、程序下载等一些列操作。
- 在 Live Watch 菜单栏中增加 ui_f32SpdKp、ui_f32SpdKi，如图 48 所示。

ui_f32IqKp	300.0	0
ui_f32IqKi	3.0E+4	0
ui_f32IdKp	300.0	0
ui_f32IdKi	3.0E+4	0
ui_f32SpdKp	1.20000001E-2	0
ui_f32SpdKi	1.99999995E-2	0

图 48 速度 PI 调节参数

- 设定转速为 0，然后改变 Live Watch 菜单栏中 ui_f32SpdKp、ui_f32SpdKi 的值。如图 48 所示。
- 设定转速为 200，待电机运行稳定后，设定转速为 600（根据具体情况确定转速的改变值），然后使用示波器观察在电机转速改变时的电流变化情况，如图 49 所示。

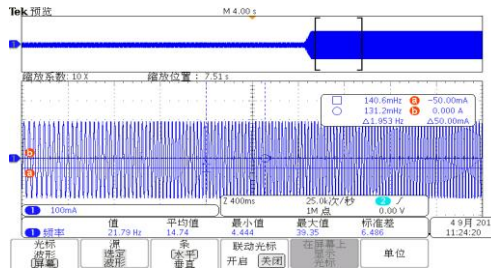


图 49 速度 PI 调节电流变化情况

- 重复步骤 2 到步骤 4，直到速度 PI 控制参数满足要求，退出调试。
- 在 s04_user\user_interface.c 中，改变 ui_f32SpdKp、ui_f32SpdKi 的值为调试时所使用的值，如图 50 所示。

```

/** speed PI regulator *****/
float32_t ui_f32SpdKp = 0.012f;
float32_t ui_f32SpdKi = 0.02f;
    
```

图 50 速度 PI 参数设定

11. 附录

11.1 原理图

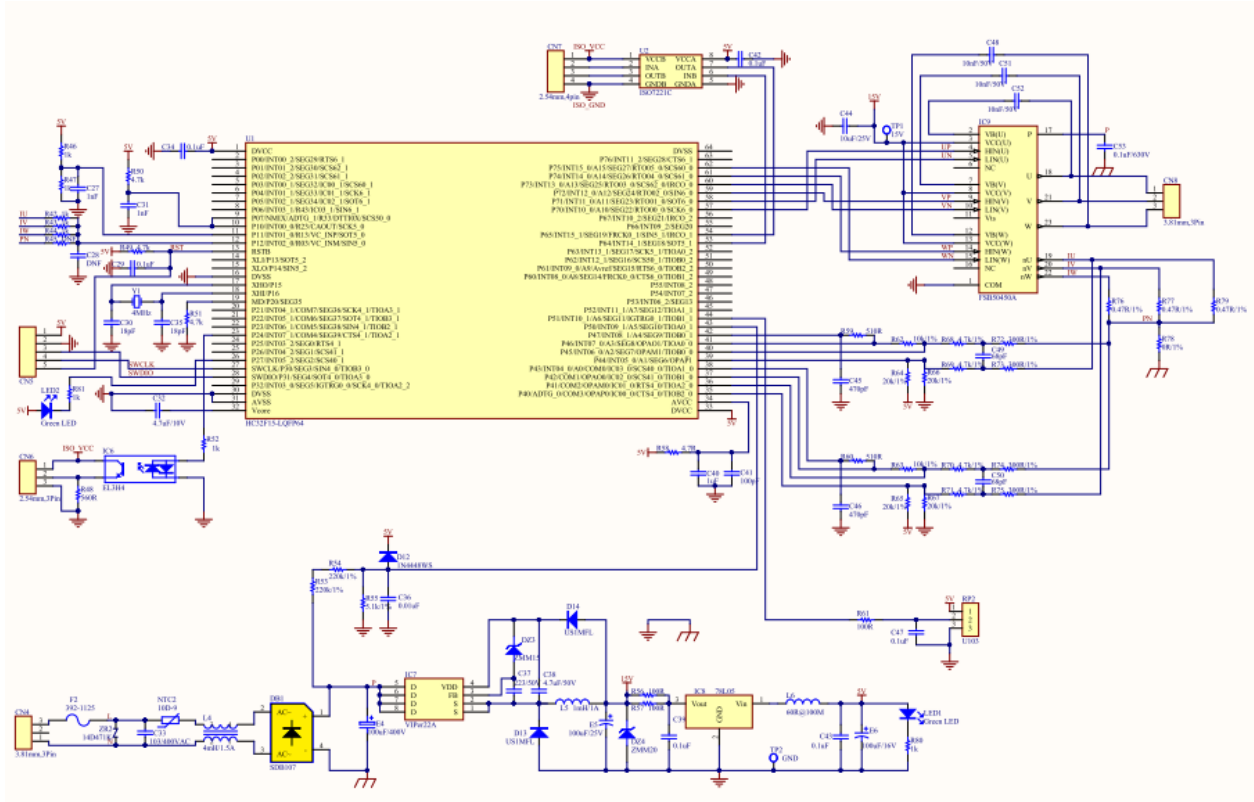


图 51 原理图

12. 版本信息 & 联系方式

日期	版本	修改记录
2019/3/13	Rev1.0	初版发布。



如果您在购买与使用过程中有任何意见或建议，请随时与我们联系。

Email: mcu@hdsc.com.cn

网址: www.hdsc.com.cn

通信地址: 上海市张江高科园区碧波路 572 弄 39 号

邮编: 201203

