

冰箱压缩机参考设计

用户指南

目录

安全须知	2
文档版本历史	3
术语表	4
1 简介	5
1.1 系统和工具要求	6
1.1.1 基本要求	6
1.1.2 高级要求	6
2 硬件说明	7
2.1 简介	7
2.2 关键功能模块	8
2.2.1 EMI滤波器和保护电路	9
2.2.2 整流器	10
2.2.3 辅助电源	10
2.2.4 MCU	11
2.2.5 逆变器电路	12
2.2.6 反馈电路	13
2.2.7 通信端口	14
2.3 用户接口	14
2.3.1 连接器和插座	14
2.3.2 LED指示灯	16
2.4 电气规范	17
3 设置和运行	18
3.1 基本操作	18
3.2 使用诊断内核运行	21
3.3 通过测试用具进行调试和测试	27
3.3.1 通过测试用具进行调试和测试操作	28
3.3.2 测试用具设置的简要汇总	30
4 运行其他压缩机	31
4.1 修改理论算术参数	31
4.2 调试参数	34
4.2.1 调试启动参数	34
4.2.2 调试PI参数	35
5 已知问题	37
附录A. 标志说明	38
附录B. 原理图	39

安全须知

应遵守产品随附的安全须知和操作说明，以避免安全隐患。



警告——输入交流市电、输出端子和其他接口彼此未隔离。施加输入市电电压时，请勿将任何非隔离器件连接到开发板。

警告——无论施加的输入市电电压如何，输出端子相对于地的最高电压均为**410V**。这些端子在工作期间为火线，并且在断开电源后仍保持**3分钟**。在此期间切勿尝试接触端子或移除端子盖。

小心——本系统只能由了解电击危险并已阅读和理解用户说明的合格人员安装、操作、维修和修改。用户执行的任何维修或修改由用户自行承担风险，并且所有保修都将失效。

小心——参考设计板设计为通过非锁定插头与交流市电连接。由于参考设计板没有市电开关，只能通过此插头与电源断开连接，因此用户在操作期间必须能无障碍地使用此插头。

文档版本历史

版本A——2019年12月4日
这是本文档的初始版本。

术语表

ADC	——模数转换器
ATPLL	——角度跟踪锁相环
DSC	——数字信号控制器
RCDRD	——冰箱压缩机开发参考设计
RCDB	——冰箱压缩机开发板
FOC	——磁场定向控制
IPMSM	——内置永磁同步电机
LDO	——低压差稳压器
LED	——发光二极管
MCU	——单片机
MTPA	——最大转矩/电流
Op-Amp	——运算放大器
PI	——比例-积分
PWM	——脉宽调制
RMS	——均方根
SPMSM	——表面永磁同步电机
SVPWM	——空间矢量脉宽调制
TTL	——晶体管-晶体管逻辑
UART	——通用异步收发器
USB	——通用串行总线

1 简介

与传统的单相交流电机相比，PMSM电机的效率更高、噪声更低，因此被广泛应用于家用电冰箱压缩机中。典型的冰箱电气控制系统包含2个控制板，一个用于控制冰箱系统，另一个用于控制压缩机。

本冰箱压缩机开发参考设计（Refrigerator Compressor Development Reference Design, RCDRD）旨在帮助工程师使用dsPIC®数字信号控制器（Digital Signal Controller, DSC）开发PMSM压缩机控制应用。本开发参考设计的目标是为内置永磁同步电机（Interior Permanent Magnet Synchronous Motor, IPMSM）和表面永磁同步电机（Surface Permanent Magnet Synchronous Motor, SPMSM）压缩机提供一种极具竞争力的无传感器控制解决方案，兼具硬件和软件。

额定连续输出电流为0.65A（RMS）。当在室温环境下以187V至264V单相输入电压运行时，最高可达到约250W的连续输出功率。对于压缩机的瞬时高压操作，峰值输出功率为350W。有关RCDRD硬件的更多详细信息，请参见第2章“硬件说明”。

本文档介绍如何使用RCDRD原型1。下面的图1-1给出了此参考设计的图片。

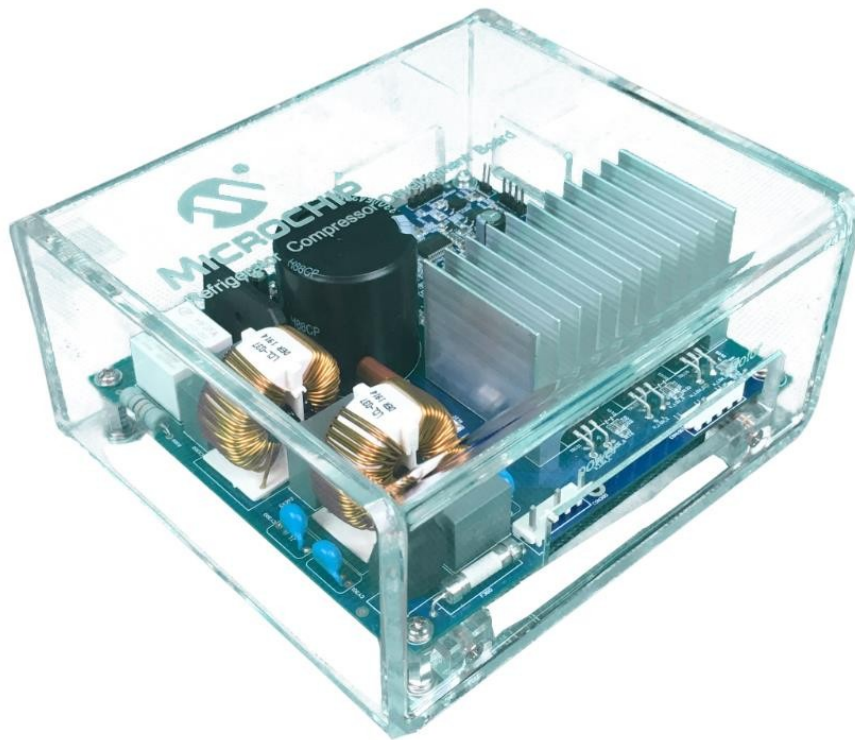


图1-1 冰箱压缩机开发参考设计

1.1 系统和工具要求

1.1.1 基本要求

- 冰箱压缩机开发板（Refrigerator Compressor Development Board, RCDB）。
- MPLAB® X V5.10或更高版本。
- 编译器XC16 V1.33或更高版本。
- PICKit™ 3/PICKit 4在线调试器（部件编号：PG164130/PG164140）。

1.1.2 高级要求

以下要求适用于诊断和测试用具功能。

- MCP2200隔离式USB转UART演示板（部件编号：ADM00276）。



- 4条杜邦线。
- X2C Scope插件V1.30。

2 硬件说明

2.1 简介

RCDB由220V交流电网供电，使用dsPIC33EP64MC202控制压缩机电机。输入的220V交流电压经过滤波器和保护电路，然后通过整流器电路转换为直流电压。辅助电源电路为MCU、反馈电路和三相逆变桥电路提供非隔离15V和3.3V电压。下面的图2-1给出了开发板的图片，而图2-2给出了框图。



图2-1 冰箱压缩机开发板

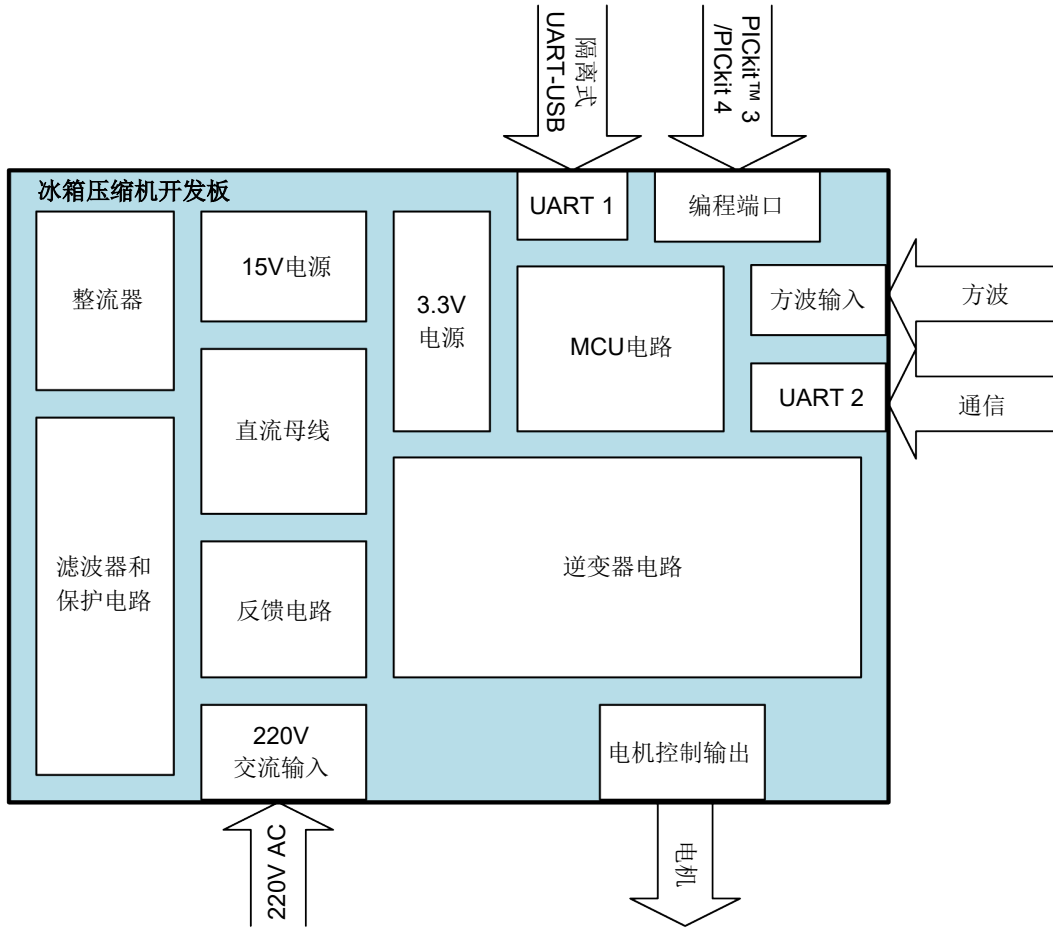


图2-2 RCDB框图

2.2 关键功能模块

RCDB具有各种功能模块：EMI滤波器和保护电路、整流器、辅助电源、MCU、逆变器、反馈电路以及通信端口。图2-3和表2-1中标记了这些模块。

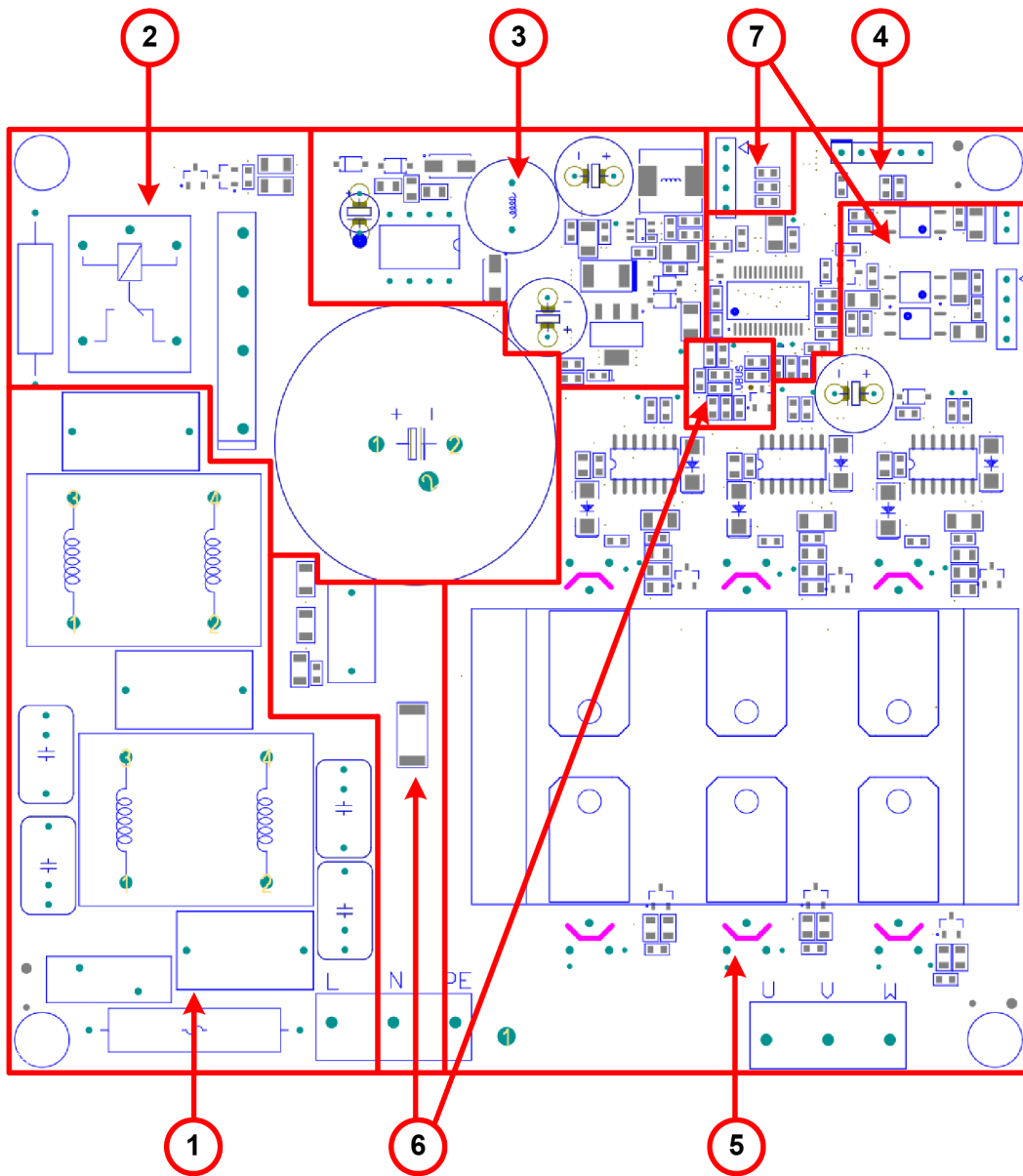


图2-3 RCDB功能模块

表2-1 RCDB功能模块

① EMI滤波器和保护电路	⑤ 三相逆变桥电路
② 整流器	⑥ 反馈电路
③ 辅助电源	⑦ 通信端口
④ MCU	

2.2.1 EMI滤波器和保护电路

开发板的输入电源先经过EMI滤波器和保护模块，如下面的图2-4所示。F300是一个250 VAC/10A快速熔断器，在电流达到其额定值的1000%时，断开时间为0.03s。Vst300是一

个名称为TVR14561D的氧化锌压敏电阻，用于浪涌保护。EMI滤波器包含两级。如果只有一级滤波器有效，则应将CY300、CY303、CX302、L301和CX300脱焊，并将L301短路。

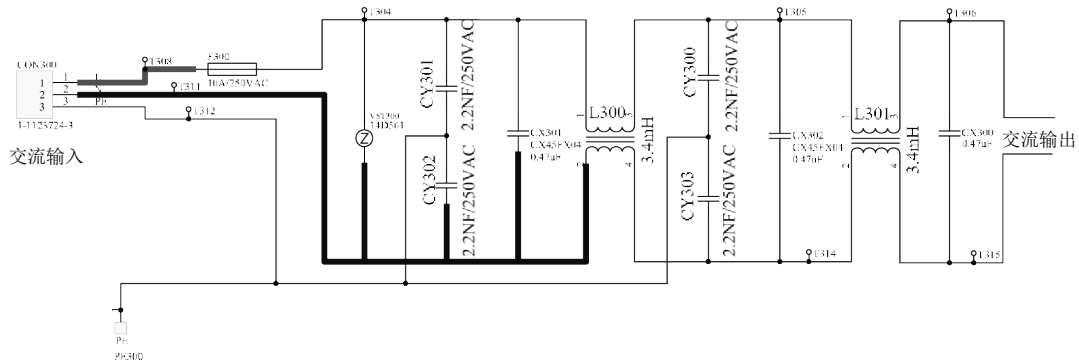


图2-4 EMI滤波器和保护电路

2.2.2 整流器

此模块包括一个浪涌电流保护电路、一个单相全桥整流器和一个直流母线电容，如下面的图2-5所示。开发板上电时，功率电阻R305会抑制浪涌电流。直流母线电容完全充电后，继电器RL300会将功率电阻短路。单相桥式整流器DB300会将工频输入交流电压转换为恒定的直流电压。

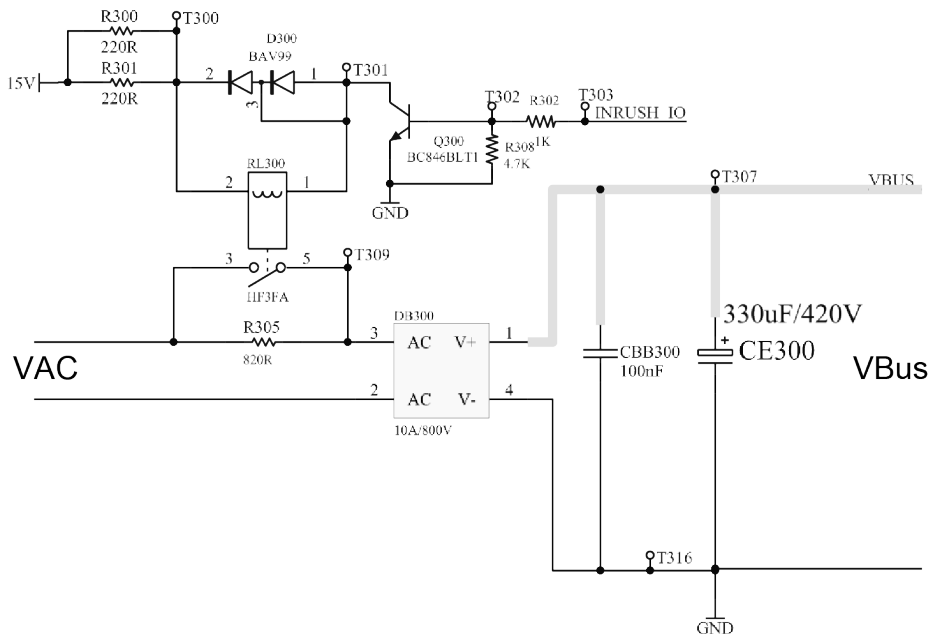


图2-5 整流器电路

2.2.3 辅助电源

辅助电源将分别为IGBT驱动器和dsPIC33EP64MC202生成15V和3.3V电压。

图2-6给出了辅助电源的框图。15V电源轨将整流器输出直接降压得到。这样可以省去反激方法中使用的变压器和高额定电压元件。MCP16331用于将此15V电压降至5V。MIC5239 LDO将电压从5V稳压至3.3V，以便为控制芯片dsPIC33EP64MC202提供干净的电源。除了控制芯片之外，此3.3V电源轨还将为其他数字和模拟控制电路供电，例如ADC参考和反馈电路。这种方法可提供良好的辅助电源，而功耗却很低。

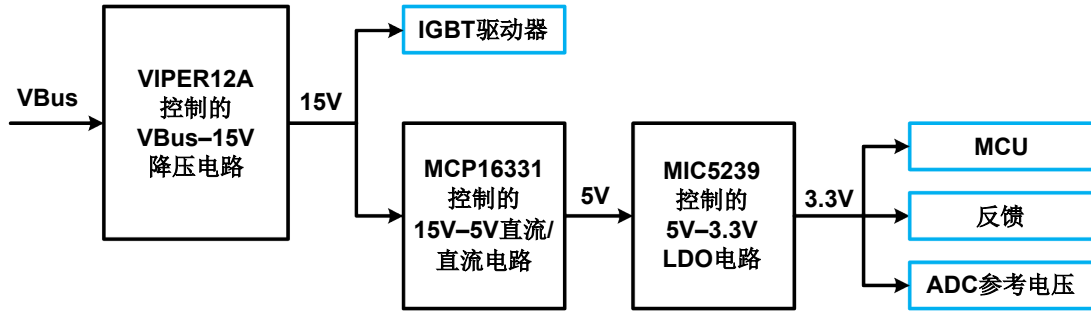


图2-6 辅助电源

有关这三个辅助电路的详细信息，请参见附录B. 原理图。

2.2.4 MCU

本设计中选择的是dsPIC33EP64MC202。另外，还有一种方法是由单个芯片同时控制冰箱系统和压缩机。Microchip提供了dsPIC33EP或dsPIC33CK系列的许多其他产品，可以很好地满足应用的需求。

下表列出了本设计中的芯片引脚功能。

表2-2 dsPIC33EP64MC202引脚功能（SSOP）

引脚编号	引脚功能	有效功能	说明	I/O	ANSEL	PPS
1	MCLR	MCLR	RESET			
2	AN0/OA2OUT/RA0	OA2OUT	IOOUT		1	
3	AN1/C2IN1+/RA1	C2IN1+	I+		1	
4	PGED3/VREF-/AN2/C2IN1-/SS1/RPI32/CTED2/RB0	C2IN1-	I-		1	
5	PGEC3/VREF+/AN3/OA1OUT/RPI33/CTED1/RB1	AN3	VBUS	I	1	
6	PGEC1/AN4/C1IN1+/RPI34/RB2	RPI34	RX1	I		1
7	PGED1/AN5/C1IN1-/RP35/RB3	RP35	TX1	O		1
8	VSS	VSS	VSS			
9	OSC1/CLKI/RA2	RA2	LED1	O		
10	OSC2/CLKO/RA3	RA3	时钟输出/测试点	I/O		
11	FLT32/RP36/RB4	RB4	测试点	I/O		
12	CVREF20/RP20/T1CK/RA4	RA4	继电器	O		

引脚编号	引脚功能	有效功能	说明	I/O	ANSEL	PPS
13	VDD	VDD	VDD			
14	PGED2/ASDA2/RP37/RB5	PGED2	调试	I		
15	PGEC2/ASCL2/RP38/RB6	PGEC2	调试	I		
16	SCK1/RP39/INT0/RB7	RP39	RX2	I		1
17	TCK/CVREF10/ASCL1/SDO1/RP40/T4CK/RB8	RP40	TX2	O		1
18	TMS/ASDA1/SDI1/RP41/RB9	PR41	Square_INPUT	I		1
19	VSS	VSS	VSS			
20	VCAP	VCAP	VCAP			
21	TDO/RP42/PWM3H/RB10	PWM3H	PWM3H	O		
22	TDI/RP43/PWM3L/RB11	PWM3L	PWM3L	O		
23	RPI44/PWM2H/RB12	PWM2H	PWM2H	O		
24	RPI45/PWM2L/CTPLS/RB13	PWM2L	PWM2L	O		
25	RPI46/PWM1H/T3CK/RB14	PWM1H	PWM1H	O		
26	RPI47/PWM1L/T5CK/RB15	PWM1L	PWM1L	O		
27	AVSS	AVSS	AVSS			
28	AVDD	AVDD	AVDD			

2.2.5 逆变器电路

三相逆变器采用分立IGBT和单电流检测电阻以节省系统成本。其中包括：

- 6个600V/15A IGBT
- 3个600V半桥IGBT驱动器MIC4608
- 单电流检测电阻，用于电机相电流检测
- PWM开关频率为2.5 kHz

MCU将PWM信号提供给3个半桥驱动器以开关6个IGBT，从而为电机各相供电。单电流检测电阻在负直流母线上以串联方式连接，用于电流反馈和过流保护。

为简单起见，图2-7中仅给出了三相逆变器电路中的一相。

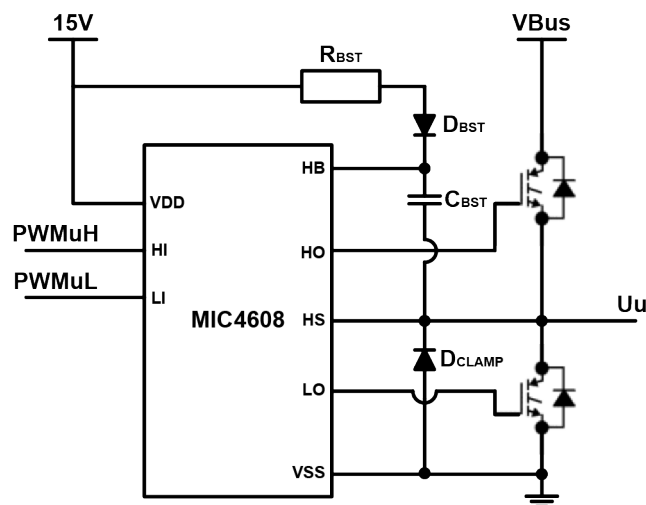


图2-7 单相逆变桥电路图

自举电路

MIC4608的上桥臂驱动器用于驱动源极端子以HS引脚为参考的浮动N沟道IGBT。MIC 4608中的电平转换电路将以VSS引脚为参考的下桥臂电路与以HS引脚为参考的上桥臂驱动器隔离。当HS引脚的电压升高时，上桥臂驱动器由自举电容（ C_{BST} ）供电。

HS节点钳位

建议在开关节点与HS引脚之间使用二极管钳位，以最大程度地减少HS引脚上较大的负毛刺或脉冲。

有关自举电路和HS节点钳位的更多详细信息，请参见MIC4608数据手册。

2.2.6 反馈电路

通过对直流母线电压和电机相电流进行采样，实现电机控制和保护。

- **直流母线电压反馈。**它由分压器和低通RC滤波器组成。分压器将直流母线电压分压为3.3V/443.3V以匹配MCU逻辑电平。
- **电机电流反馈。**电流检测电阻位于三个下桥臂开关的发射极和“DC-”之间。dsPIC33EP64MC202提供内部运算放大器（运放）和比较器，用于放大电机电流和进行过流保护。运放的增益设置为4，电流检测电阻电压信号偏移0.2 VDC。因此，电机的相电流范围为-4.125A至+4.125A。

图2-8给出了反馈电路的位置。

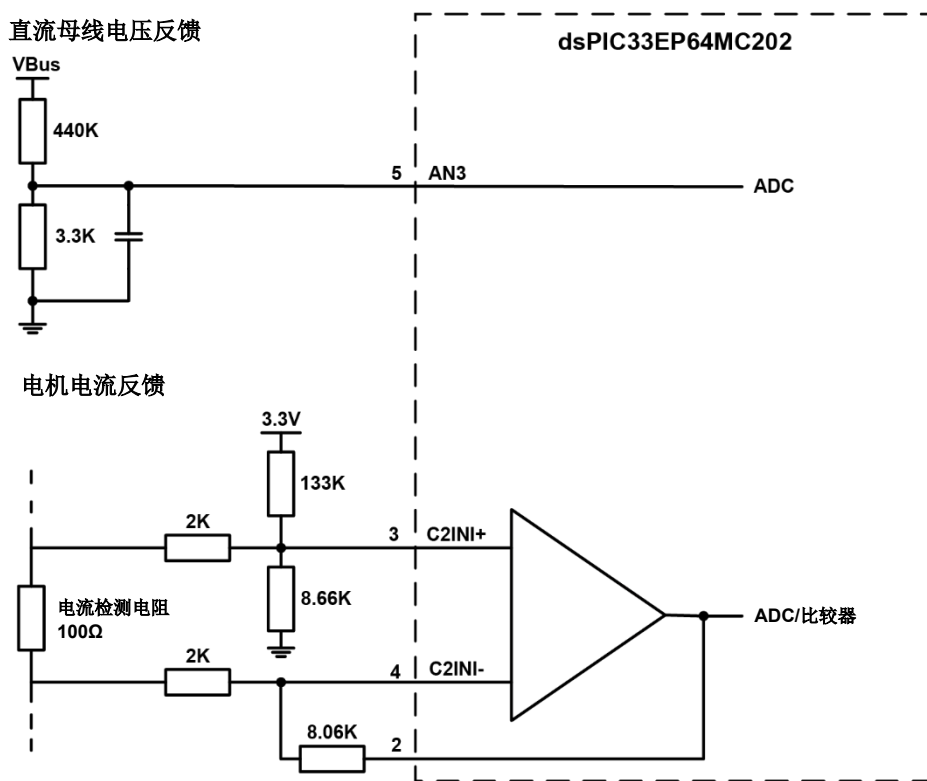


图2-8 反馈电路图

2.2.7 通信端口

RCDB提供了三个端口，用于与调试工具和/或系统控制板之间相互收发数据。所有这些端口均由3.3V电源轨供电。

- **UART1**通信端口是非隔离式端口，能够直接转换板上dsPICDSC输入和输出的UART信号。此端口用于与计算机/MPLAB X IDE通信。

注：UART1端口是非隔离式端口。直接将此端口与其他非隔离系统连接时，损坏电路板的风险较大。强烈建议将此端口与MCP2200隔离式USB-UART演示板或其他隔离式USB-UART转换器连接。

- **UART2**通信端口用于与冰箱系统控制板通信，以获取命令和运行状态信息。此端口由两个光耦合器隔离。
- **方波接收器**端口提供了另一种从主控制板接收速度命令的方法。此端口由一个光电晶体管隔离。

图2-9给出了通信端口的的位置。

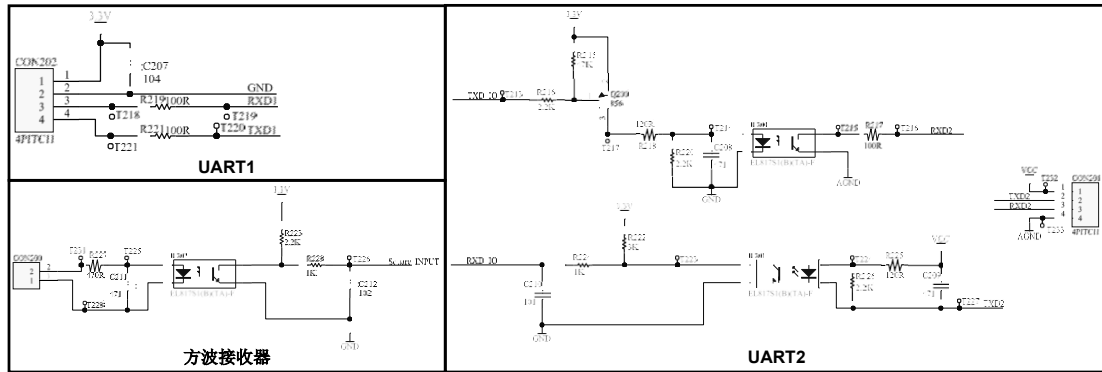


图2-9 通信端口

2.3 用户接口

2.3.1 连接器和插座

(1) 电源插座

- 交流电源插座（CON300）
- 压缩机的三相逆变器连接器（CON100）

(2) 信号插座

- 隔离式方波接收器插座（CON200）
- 隔离式UART2通信插座（CON201）
- 非隔离式UART1通信插座（CON202）
- PICKit 3/PICKit 4在线调试器插座（CON203）

UART插座有4个端子。方波插座有2个端子。调试器插座有5个端子。表2-3列出了每个端子的功能。

表2-3 信号插座的端子功能

插座编号	插座名称	引脚	端子功能
SON200	隔离式方波接收器	1	+
		2	隔离式GND
CON201	隔离式UART2	1	隔离式VCC
		2	TXD2
		3	RXD2
		4	隔离式GND
CON202	非隔离UART1	1	3.3V
		2	GND
		3	RXD1
		4	TXD1
CON203	PICKit™ 3/PICKit 4在线调试器插座	1	MCLR
		2	3.3V
		3	GND
		4	PGD
		5	PGC

所有插座的位置均可在图2-10中找到。

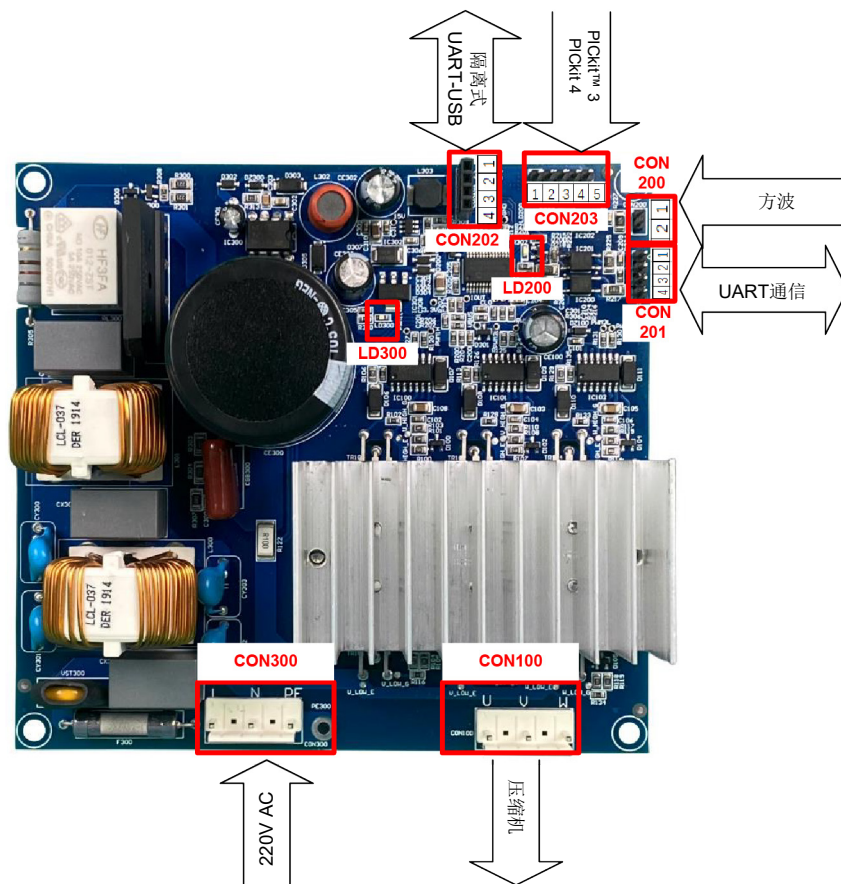


图2-10 所有插座和LED指示灯

2.3.2 LED指示灯

开发板上有两个LED指示灯。图2-10中给出了两个LED的位置。

LD300指示3.3V电源轨是否可用。当3.3V电源轨就绪时，此LED点亮。

LD200指示开发板的运行状态和错误消息。开发板上电后，此LED始终闪烁。客户可通过LED闪烁获得运行信息。当RCDB按预期运行（例如压缩机正常运行）时，下表列出了其所有可能的运行状态。

表2-4 运行状态LED闪烁模式

系统状态	LED闪烁模式
重启	熄灭
正在启动	15/16点亮@0.625 Hz
正在运行	1/16点亮@0.625 Hz
正在停止	15/16点亮@0.625 Hz
已停止	50%点亮@0.625 Hz

如果驱动压缩机的开发板处于故障状态，则LED将按以下错误代码形式闪烁。

表2-5 LED闪烁模式错误代码

故障类型	LED闪烁模式
陷阱	
ERR_OSC_FAIL	+ + - - -
ERR_ADDRESS_ERROR	+ + + - - -
ERR_HARD_TRAP	+ + + + - - -
ERR_STACK_ERROR	+ - + + - - -
ERR_MATH	+ + - + + - - -
ERR_RESERVED_TRAP5	+ + + - + + - - -
ERR_SOFT_TRAP	+ + + + - + + - - -
ERR_RESERVED_TRAP7	+ - + + + - - -
应用程序错误	
ERR_STALL	+ - + - + + - - -
ERR_INVALID_STARTUP_FSM_STATE	+ + - + - + + - - -
ERR_HW_OVERCURRENT	+ + + - + - + + - - -
ERR_DCLINK_OVERVOLTAGE	+ + + + - + - + + - - -
ERR_DCLINK_UNDERVOLTAGE	+ - + + - + + - - -
ERR_DIRECTREVERSE	+ + - + + - + + - - -
ERR_SPEEDERROR	+ + + - + + - + + - - -
复位错误	
ERR_RCON_TRAPR	+ - + - + + + - - -

故障类型	LED闪烁模式
ERR_RCON_IOPUWR	+ + - - + + - - -
ERR_RCON_CM	+ + + - - + + + - - -
ERR_RCON_WDTO_ISR	+ + + + - + - + + + - - -
ERR_RCON_WDTO_MAINLOOP	+ - + + - + + + - - -
ERR_UNEXPECTED_INTERRUPT_BASE	+ - + - - + - + + - - -

在上表中，“+”表示LED以50%占空比 @ 1.67Hz闪烁一次。“-”表示LED在1.67 Hz的整个周期内熄灭。

2.4 电气规范

下表列出了RCDB开发板的电气规范：

表2-6 RCDB开发板的电气规范

编号	参数	规范
1	额定输入电压	220 VAC 50/60 Hz
2	输入电压范围	90V~264 VAC
3	数字电压	3.3V
4	额定功率	250W
5	最大功率	250W
6	PWM频率范围	2.5 kHz~15 kHz
7	工作温度	-10°C ~ +60°C
8	开发板效率	94%
9	开发板待机功率	0.5W
10	开发板尺寸	130 mm × 120 mm × 40 mm
11	整流器	有
12	辅助电源	有
13	PFC电路	无
14	电源电路	分立
15	电流检测	dsPIC® DSC中的单电流检测电阻和内部运放
16	直流母线电压检测	有，dsPIC DSC中的内部运放
17	温度检测	无
18	硬件过流保护	有，dsPIC DSC中的内部比较器
19	直流母线的浪涌电流保护	有，功率电阻和继电器

3 设置和运行

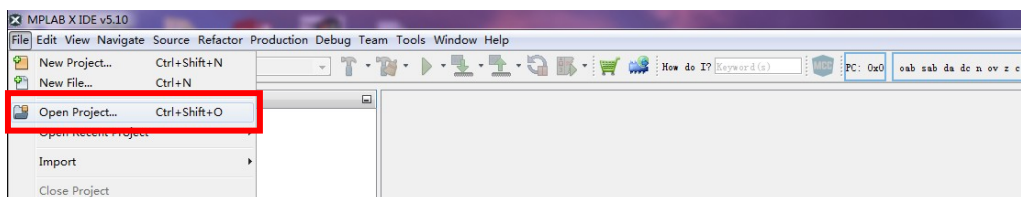
RCDRD的软件已针对基本操作进行了配置。本章介绍如何设置软件、运行压缩机和调试代码。

3.1 基本操作

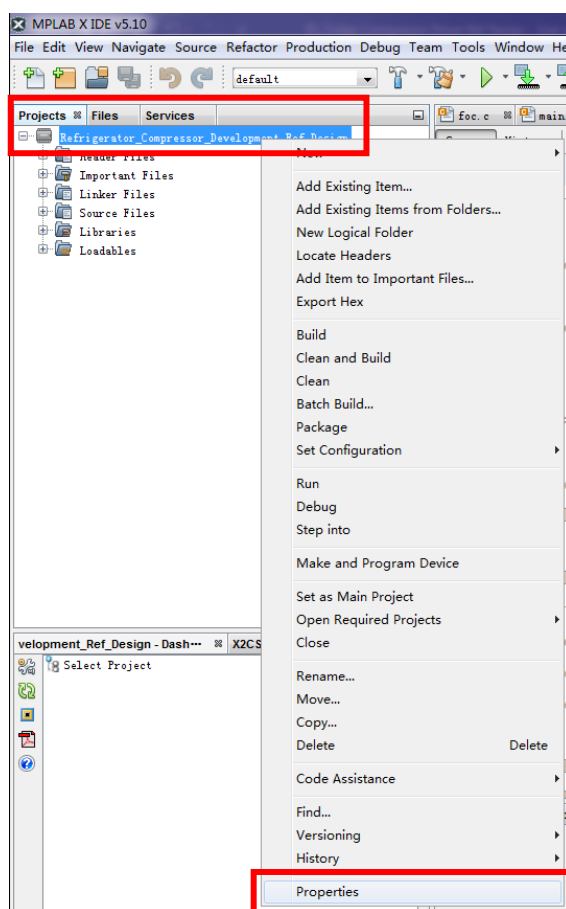
基本操作包括打开软件项目、配置和编译代码，以及对开发板上的芯片进行编程。

(1) 启动MPLAB X IDE V5.10，然后打开项目“RCDRD_V1.0.X”。

注：此项目需要MPLAB X IDE V5.10或更高版本。

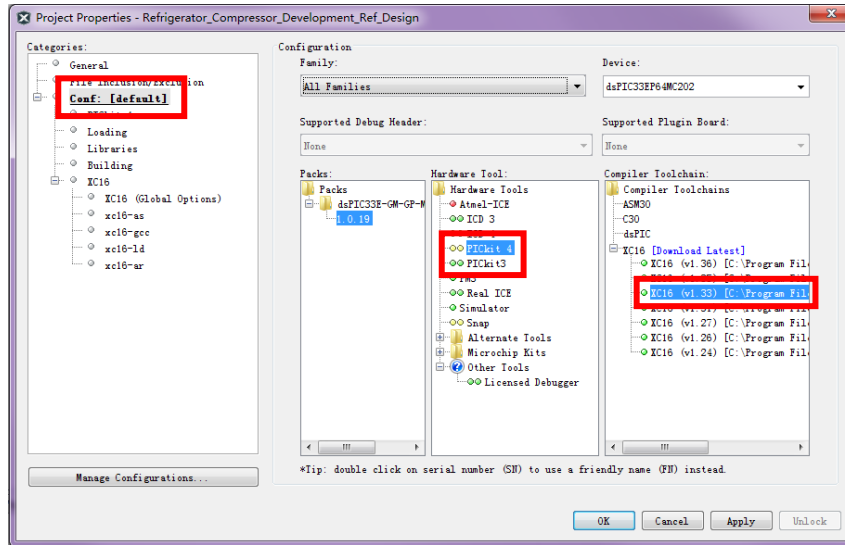


(2) 在左侧名为“Project”（项目）的选项卡上右键单击此项目，选择最后一项“Properties”（属性）。



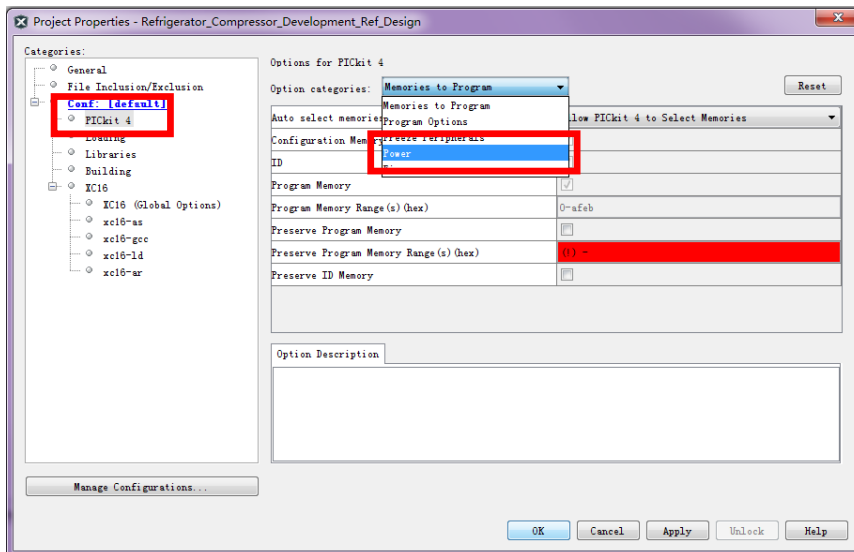
在“Project Properties”（项目属性）页面上，可以在“Hardware Tool”（硬件工具）部分选择编程器/调试器，并且在“Compiler toolchain”（编译器工具链）部分选择编译器（XC16 V1.33或更高版本）。将编程器/调试器连接到计算机后，即可在

Hardware Tools（硬件工具）列表中找到“PICKit 3”/“PICKit 4”项。单击“Apply”（应用），应用选择。

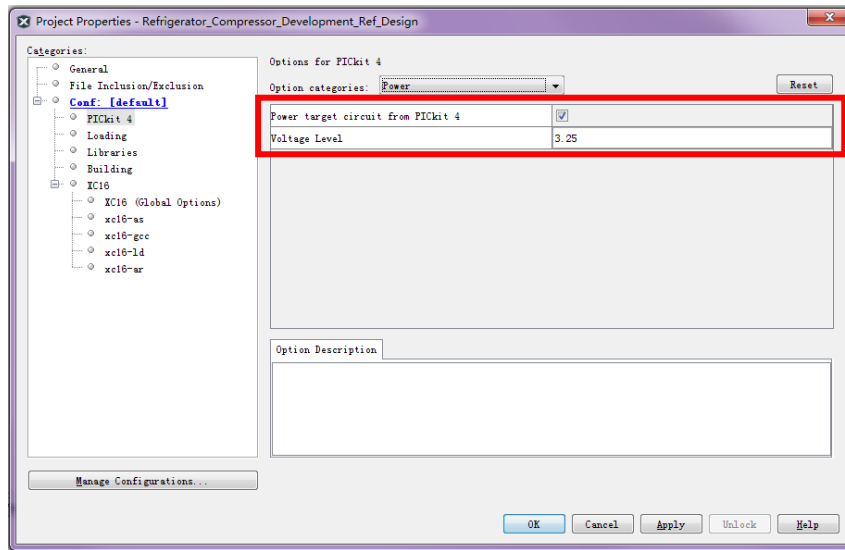


(3) RCDB开发板采用热地线设计。当它与任何其他非隔离工具和设备（例如计算机和示波器等）连接时，禁止为其供电。建议使用编程器/调试器对芯片进行编程，因为编程器/调试器可以为开发板提供3.3V电压来进行编程。

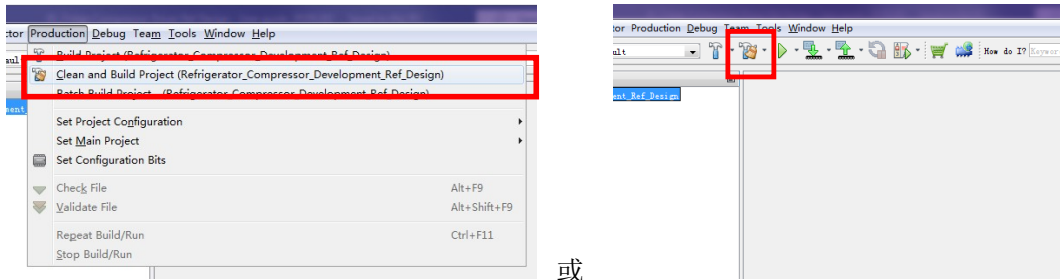
应用编程器/调试器选择（PICKit 3/PICKit 4）后，可以在“Project Properties”窗口的“Categories”（类别）列表中找到“PICKit 3”/“PICKit 4”项。



在“PICKit 3”/“PICKit 4”页面上的“Option categories”（选项类别）中，选择“Power”（电源）项。选中“Power target circuit from PICKit 3”（通过PICKit 3为目标电路供电）项，然后为“Voltage level”（电压值）项选择“3.25”。单击“OK”（确定），应用选择。

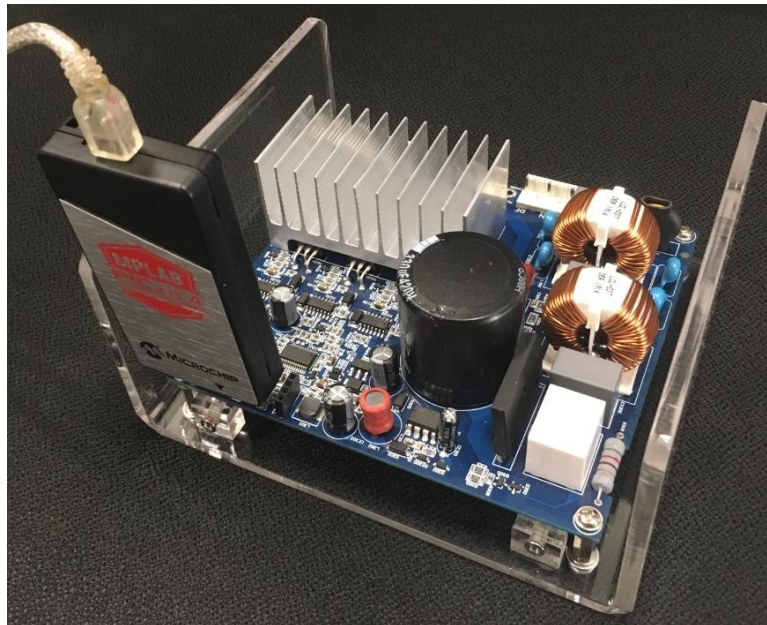


- (4) 单击工具栏或“Production”（生产）菜单中的“Clean and Build Project”（清除并编译项目）按钮，编译代码。

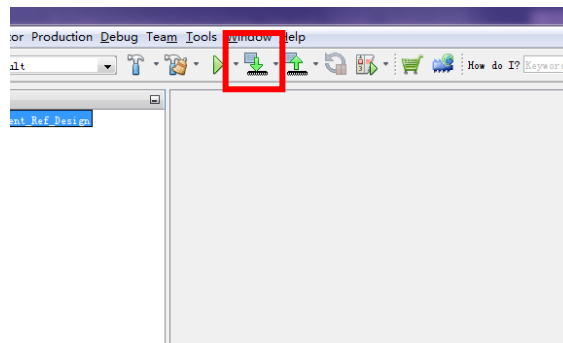


或

- (5) 成功编译之后，将PICKit 3/PICKit 4连接到RCDB CON203。



单击工具栏上的“Make and Program”（编译并编程）按钮，将代码下载到芯片中。



注：在执行此步骤之前，请确保交流电源已断开与开发板的连接。

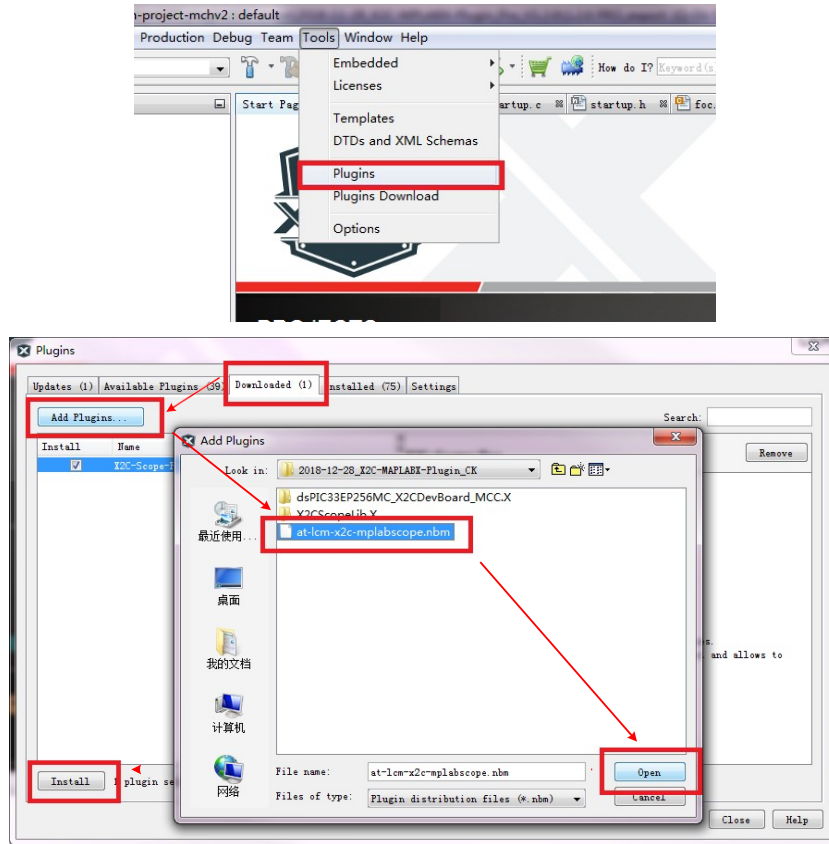
- (6) 断开PICKit 3与开发板的连接。确保RCDB未与任何其他非隔离开发板或设备连接。
- (7) 在CON100上连接压缩机或电机，在CON300上连接交流电源。为开发板上电。开发板现在处于“已停止”状态，LED会以50%占空比@0.625 Hz闪烁。



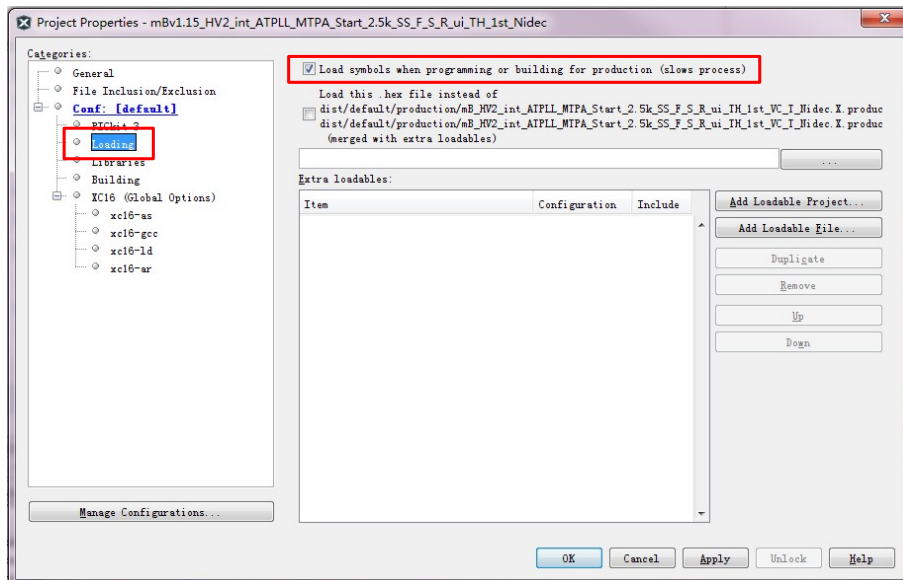
3.2 使用诊断内核运行

就RCDRD原型1而言，运行压缩机的唯一方法是使用代码中的诊断内核功能。与诊断内核功能配合使用的工具是X2C Scope插件和MCP2200隔离式USB转UART演示板。X2C Scope是MPLAB X IDE的插件，用于简化调试工作。它能够提供通过UART在dsPIC数据存储器中读写变量的完整功能，还可以在实时模式下绘制这些变量。

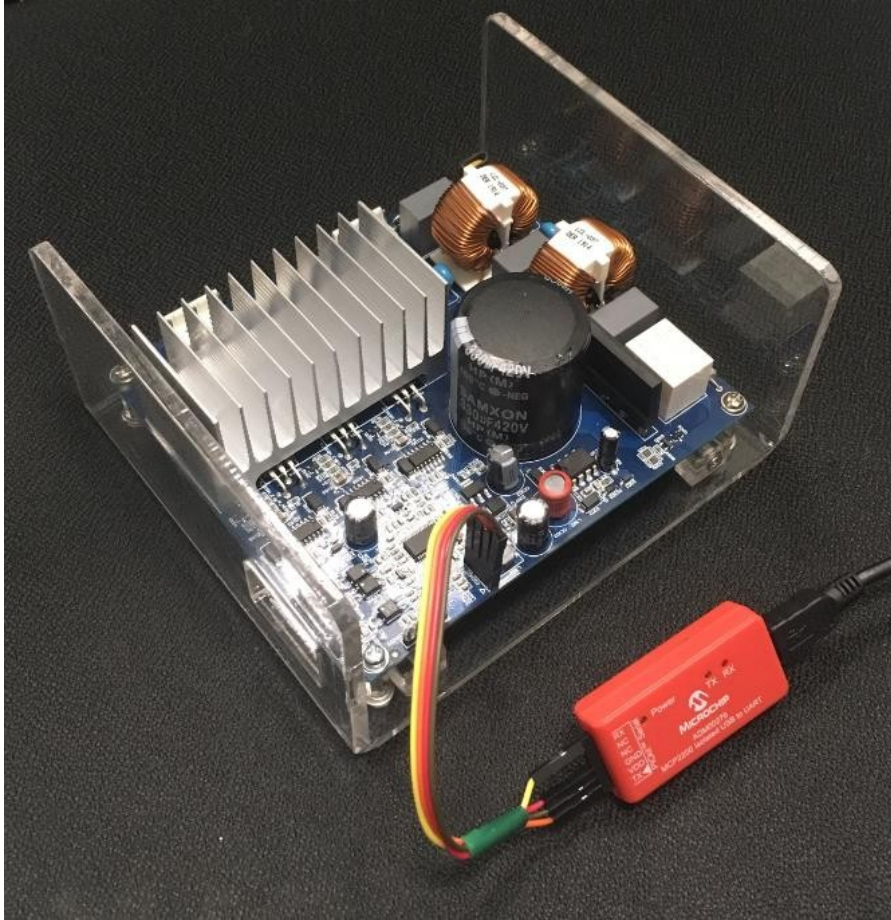
- (1) 安装X2C Scope插件。从“Tools”（工具）菜单中选择“Plugins”（插件）项。在“Downloaded”（已下载）页面上，单击“Add Plugins...”（添加插件...）按钮，选择“at-lcm-x2c-mplabscope.nbm”文件。单击“Install”（安装），安装X2C Scope插件。



- (2) 打开project properties（如3.1(2)中所示），在“Loading”（装入）页面的生产编译期间使能“Load symbols when programming or building for production (slows process)”（为生产进行编程或编译时装入符号（减慢过程））。



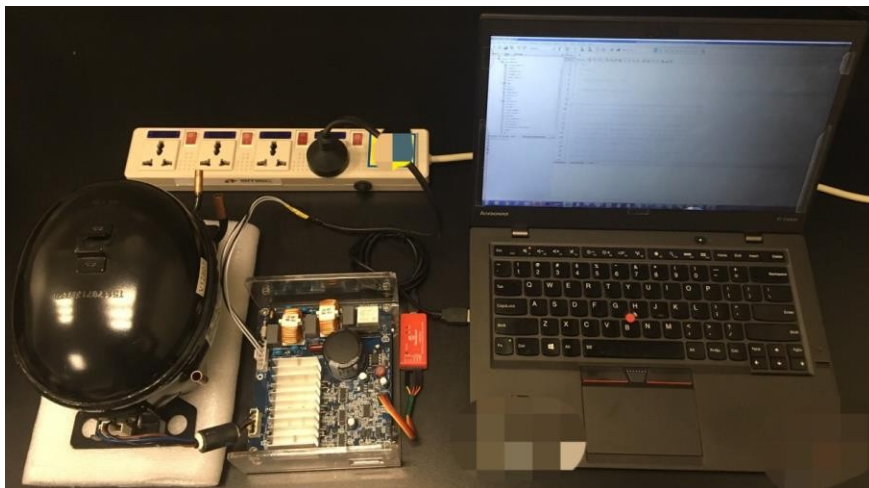
- (3) 通过杜邦线将MCP2200隔离式USB转UART演示板连接到计算机和RCDB。



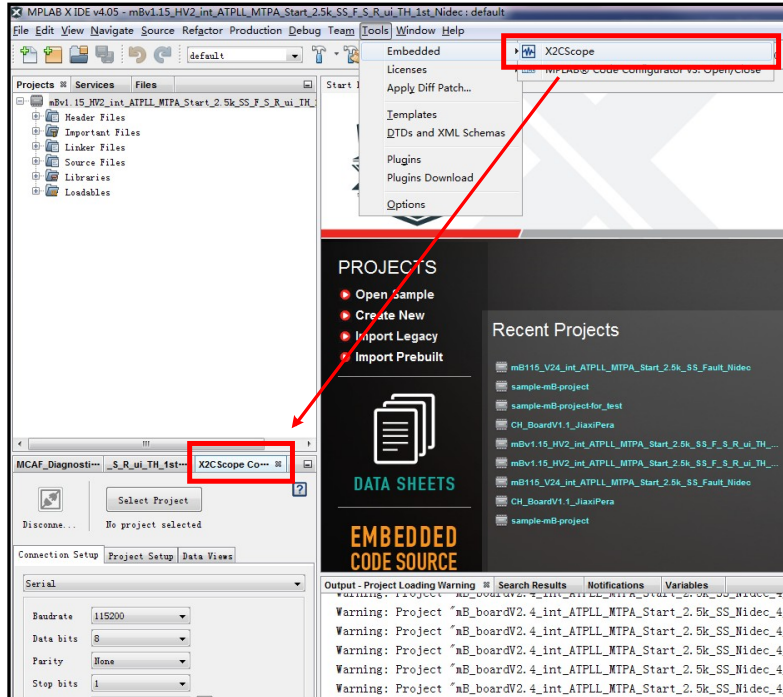
(4) 将代码下载到驱动板（如3.1(4)(5)(6)中所述），断开所有非隔离器件的连接。

注：每次开发板掉电和上电时均需要下载代码，以将X2C Scope连接到板上MCU。

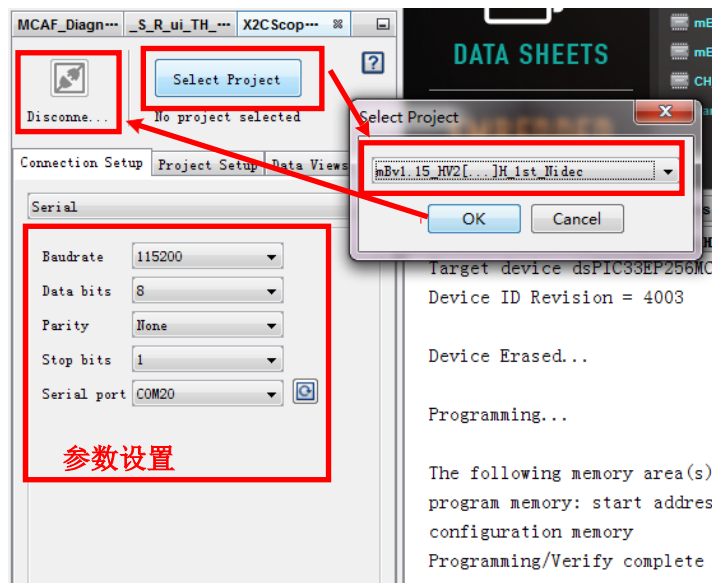
(5) 断开编程器/调试器的连接，然后为开发板上电。



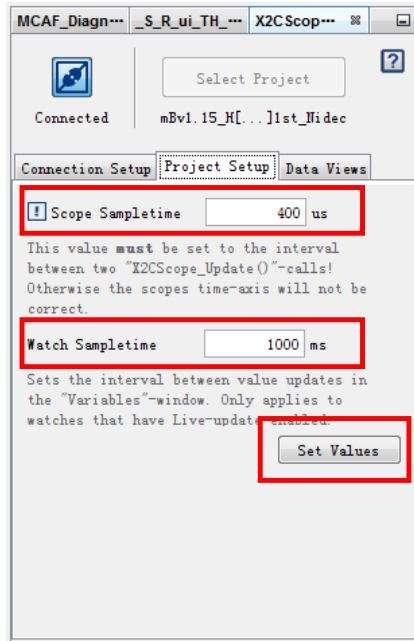
(6) 从“Tools”菜单上的“Embedded”（已安装工具）中单击“X2CScope”，启动X2C Scope。“X2C Scope Configuration”（X2C Scope配置）窗口将打开。



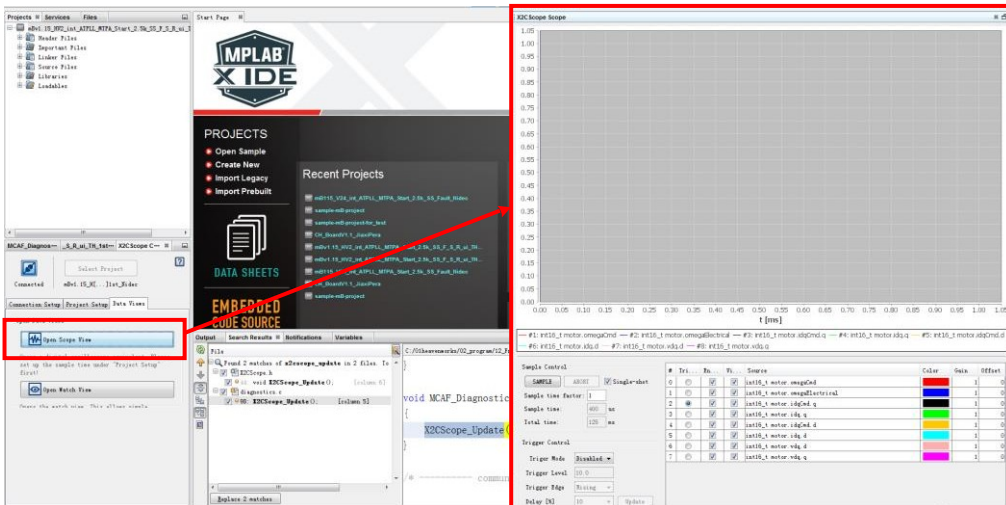
- (7) 按下图所示设置参数，在“Select Project”（选择项目）按钮中选择此RCDRD项目。然后单击“Disconne...”（断开...）按钮，将计算机连接到驱动板。

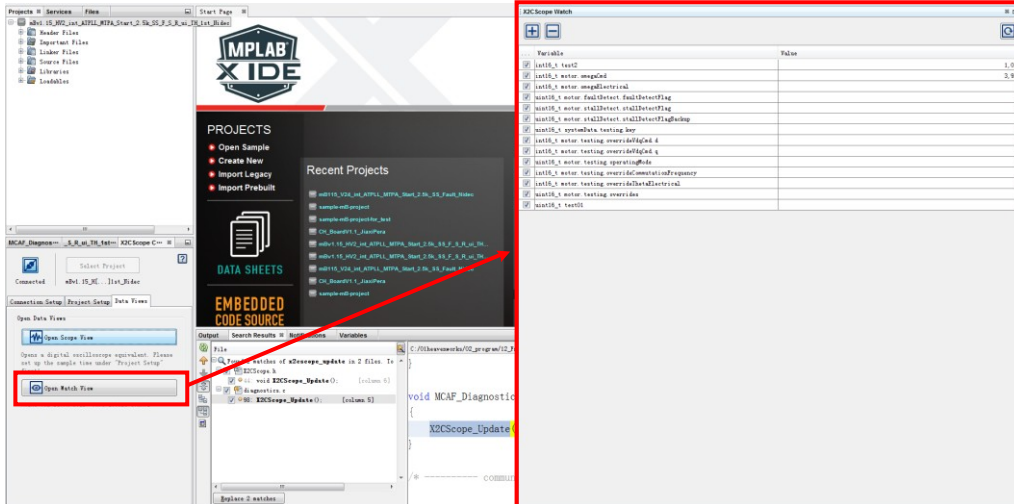


- (8) 在“Project Setup”（项目设置）页面上，“Scope Samplertime”（示波器采样时间）用于配置将在示波器时间轴中显示的采样时间间隔，其值必须与PWM周期相同。在RCDRD中，该值为400 μ s（2.5 kHz）。“X2C Scope Watch”（X2C Scope观察）窗口中的“Watch Samplertime”（观察采样时间）用于配置刷新时间间隔，其值应大于PWM周期。单击“Set Values”（设置值），应用设置。

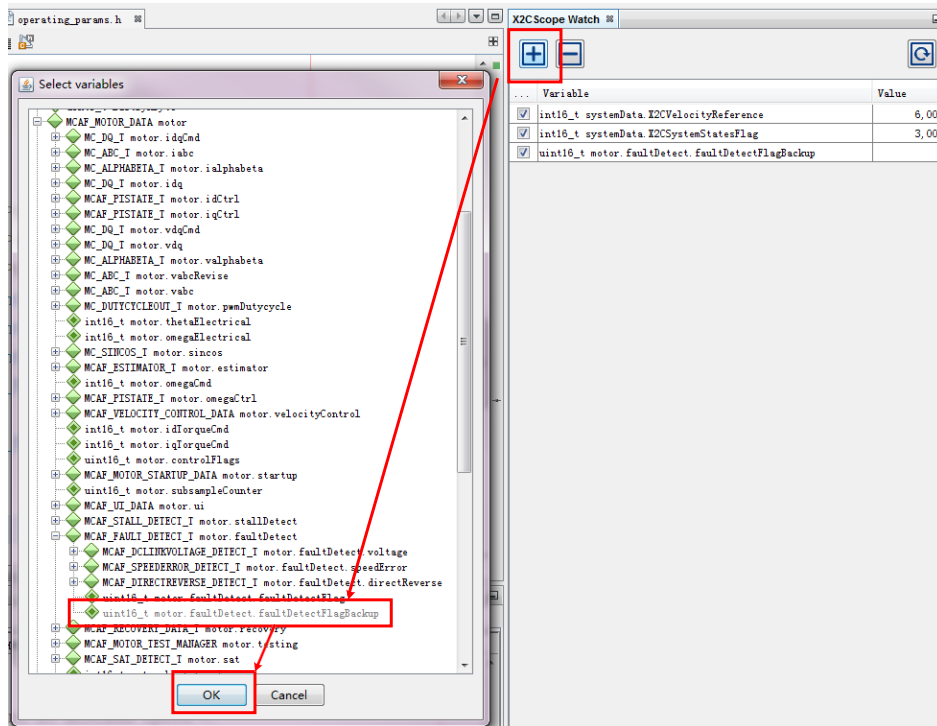


- (9) 在“Data Views”（数据视图）页面上，单击“Open Scope View”（打开示波器视图），打开“X2C Scope Scope”（X2C Scope示波器）窗口。此窗口用于显示变量的波形。单击“Open Watch View”（打开观察视图），打开“X2C Scope Watch”窗口。此窗口用于显示和更改变量值。这两个窗口中只能添加全局变量。

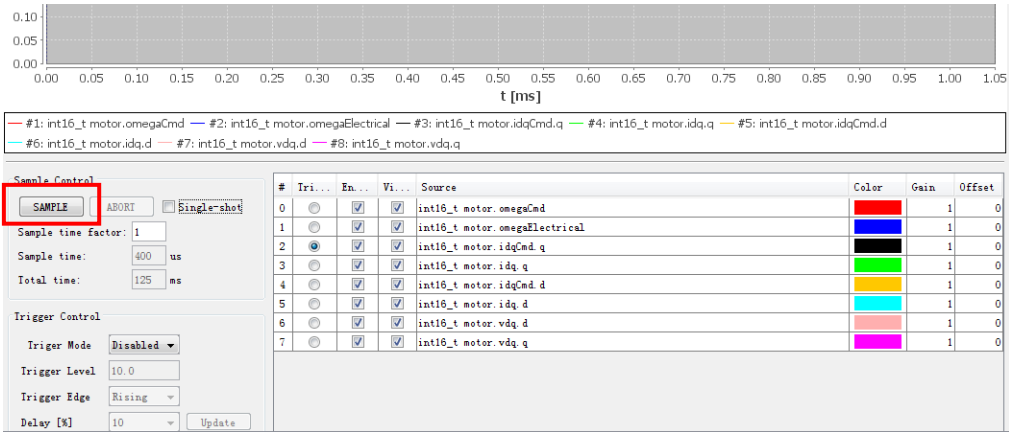




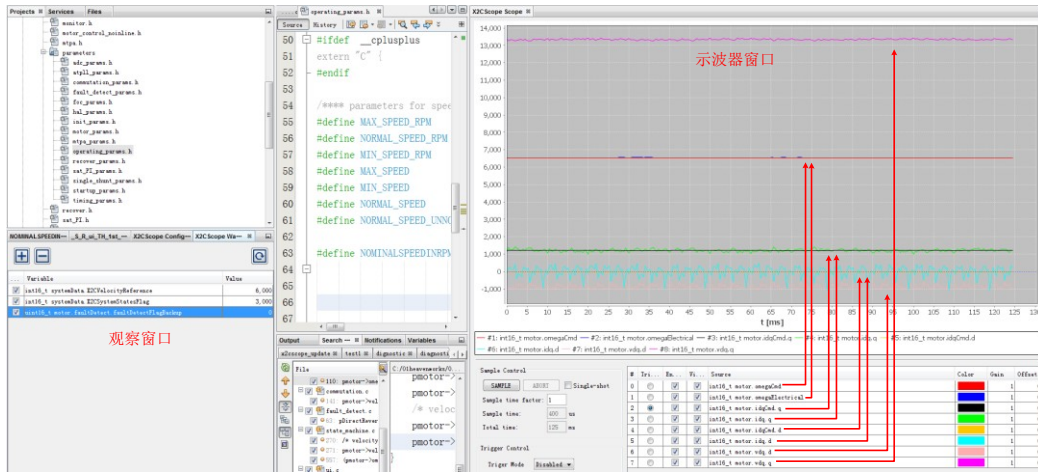
例如，下图的“X2C Scope Watch”窗口中选择了 *systemData.X2CVelocityReference*、*systemData.X2CSystemStatesFlag* 和 *motor.faultDetect.faultDetectFlagBackup* 这三个变量。变量 *systemData.X2CVelocityReference* 是压缩机电机的参考电磁速度（以 RPM 为单位）。变量 *systemData.X2CSystemStatesFlag* 是系统状态标志。变量 *motor.faultDetect.faultDetectFlagBackup* 是故障标志。有关标志的更多详细信息，请参见附录 A. 标志说明。另外，也可以添加其他关注的变量。



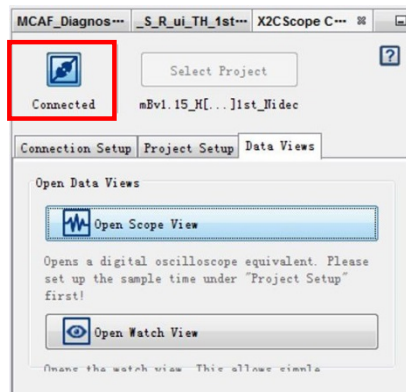
(10) 在“X2C Scope Scope”窗口中添加要观察的变量。例如，添加参考速度和反馈速度、参考id/iq和检测到的id/iq以及输出vd/vq等。单击“SAMPLE”（采样）按钮，开始采样变量并显示波形。



(11)使用大于最小速度的数据配置参考速度变量`systemData.X2CVelocityReference`。压缩机将开始运行。Scope（示波器）窗口将实时显示变量波形。



(12)操作结束时，切记单击“Connected”（已连接）按钮，断开X2C Scope与开发板的连接。

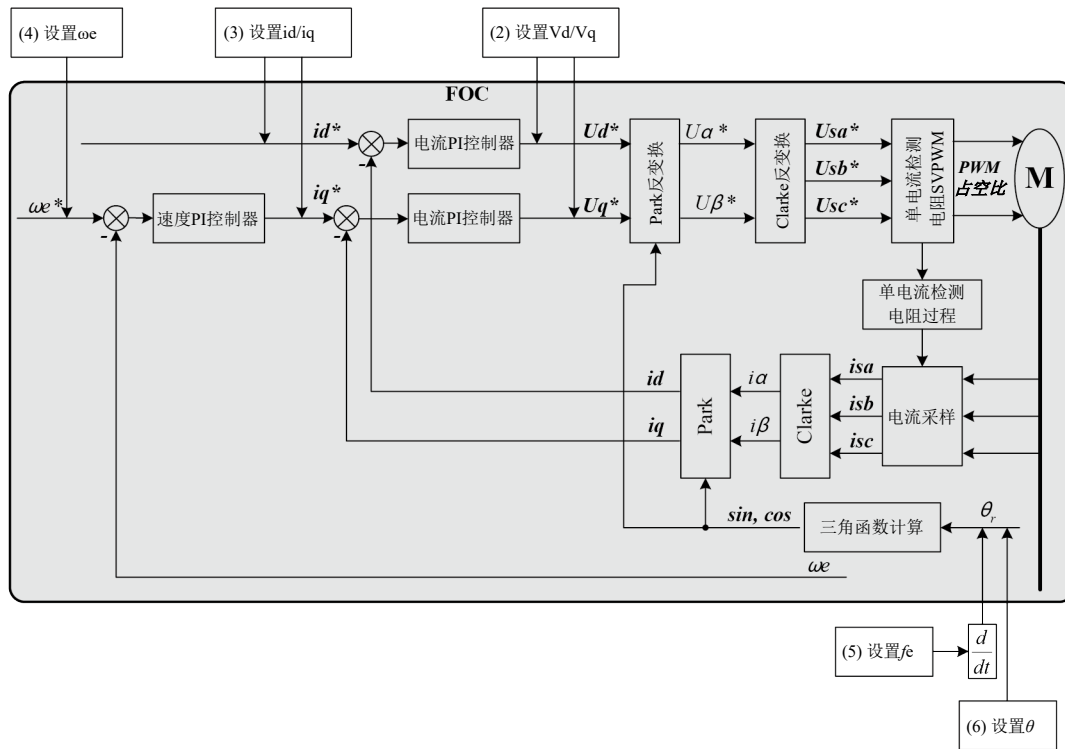


(13)有关X2C Scope的更多详细信息，请参见文件“X2C Scope Documentation”。

3.3 通过测试用具进行调试和测试

FDCRD提供了一个测试用具组件，可用于轻松修改换相和电机控制环的控制工作模式。它代表几个运行时参数，这些参数用于将系统置于某些测试模式。通过结合电机控制应用中

使用的其他可调参数，可以调试和测试压缩机及开发板以确保其正常运行。它与换相和电机控制环紧密耦合，如下图所示。



3.3.1 通过测试用具进行调试和测试操作

在“X2C Scope Watch”窗口中，可以设置所有调试和测试操作。

(1) 启动测试模式

设置密钥会使控制系统进入测试模式。否则，所有其他测试用具功能将不可用。

启动操作：

- 设置 `systemData.testing.key = TEST_GUARD_VALID = 53670`。

停止操作：

- 设置 `systemData.testing.key != TEST_GUARD_VALID != 53670`。

(2) 直接设置Vd和Vq

旁路速度环和电流环，并向其输出Vd/Vq馈入设置值。它用于调试和测试SVPWM占空比计算功能。此功能仅设置输出电压的幅值。

启动操作：

- 为 `motor.testing.overrideVdqCmd.d` 和 `motor.testing.overrideVdqCmd.q` 设置适当的值。这两个变量的值已归一化。
- 设置 `motor.testing.operatingMode = OM_FORCE_VOLTAGE_DQ = 1`。设置的Vd和Vq可用。

停止操作：

- 设置 `motor.testing.operatingMode = OM_DISABLED = 0`。没有输出电压。

相关功能：

- 结合功能(5)中的电磁频率可设置输出电压频率。
- 结合功能(6)中的电磁角可设置电机相的输出电压。

(3) 直接设置idCmd和iqCmd

旁路速度环并使能电流环，然后直接向电流参考id/iq馈入设置值。它可用于执行调试和测试电流环、调整PI参数以及校准电流采样等操作。此功能仅设置输出电流的幅值。

启动操作：

- 为*motor.testing.overrideIdqCmd.d*和*motor.testing.overrideIdqCmd.q*设置适当的值。请注意，这两个变量的值已归一化。
- 设置*motor.testing.operatingMode* = OM_FORCE_VOLTAGE_DQ = 2。随后，Id和Iq将按照我们的设置进行输出。

停止操作：

- 设置*motor.testing.operatingMode* = OM_DISABLED = 0。没有输出电流。

相关功能：

- 结合功能(5)中的电磁频率可设置输出电流频率。
- 结合功能(6)中的电磁角可设置电机相的输出电流。

(4) 直接设置参考速度

直接在FOC操作中设置参考速度，忽略原始参考速度。它用于调试和测试速度环并调整PI参数。

启动操作：

- 设置*motor.testing.operatingMode* = OM_NORMAL = 3。
- 为*motor.testing.overrideOmegaElectrical*设置适当的值。请注意，变量是电磁速度，单位为RPM。
- 设置*motor.testing.overrides* = TEST_OVERRIDE_VELOCITY_COMMAND = 1。
- 启动系统。启动周期后，驱动器将以设置的参考速度旋转电机。

停止操作：

- 设置*motor.testing.overrides* = 0。原始速度顺序可用。

(5) 直接设置电磁频率

直接在换相组件中设置电磁频率，忽略估算的电磁角。将此功能与功能(2)/功能(3)结合使用，可设置输出电压/电流的频率。如果将频率设置为0，则输出电压/电流为直流。

启动操作：

- 为*motor.testing.overrideCommutationFrequency*设置适当的值。请注意，变量是FOC的电磁频率，单位为Hz。
- 设置*motor.testing.overrides* = TEST_OVERRIDE_COMMUTATION = 2。
- 设置功能(2)或功能(3)。

停止操作：

- 设置 `motor.testing.overrides = 0`。估算的频率可用。请注意，仍存在由功能(2)或功能(3)设置的输出电压或电流。

(6) 设置电磁角

直接在换相组件中设置电磁角，忽略估算的电磁角。将此功能与功能(2)/功能(3)结合使用，可将电机相电压/电流控制为设置值。

启动操作：

- 为 `motor.testing.overrideThetaElectrical` 设置适当的值。请注意，该变量是FOC的电磁角，单位已归一化。
- 设置 `motor.testing.overrides = TEST_OVERRIDE_COMMUTATION = 4`。
- 设置功能(2)或功能(3)。

停止操作：

- 设置 `motor.testing.overrides = 0`。估算的电磁角可用。请注意，仍存在由功能(2)或功能(3)设置的输出电压或电流。

(7) 时间戳

16位时间戳数组是测试用具状态结构体的一部分。这些时间戳记录在各个位置中，以通过实时诊断工具对主控制过程进行性能分析。时间戳使用以系统时钟速率 ($F_{cy} = 70 \text{ MHz}$) 运行的定时器1，因此定时器值指示经过的系统时钟时间。时间戳数组变量的说明：

- `motor.testing.timestamps[0]`: 状态机启动前的执行时间
- `motor.testing.timestamps[1]`: 与状态无关的所有关键任务的执行时间。
- `motor.testing.timestamps[2]`: 确定下一个状态的执行时间。
- `motor.testing.timestamps[3]`: 更新状态并在该状态下执行适当操作的执行时间。
- `motor.testing.timestamps[4]`: 与状态无关的所有非关键任务的执行时间。
- `motor.testing.timestamps[7]`: 包括UI、监视器和看门狗等在内的其他过程的执行时间。

操作：

- 在X2C Scope中添加时间戳数组变量。

3.3.2 测试用具设置的简要汇总

下表中汇总了测试用具的功能和设置：

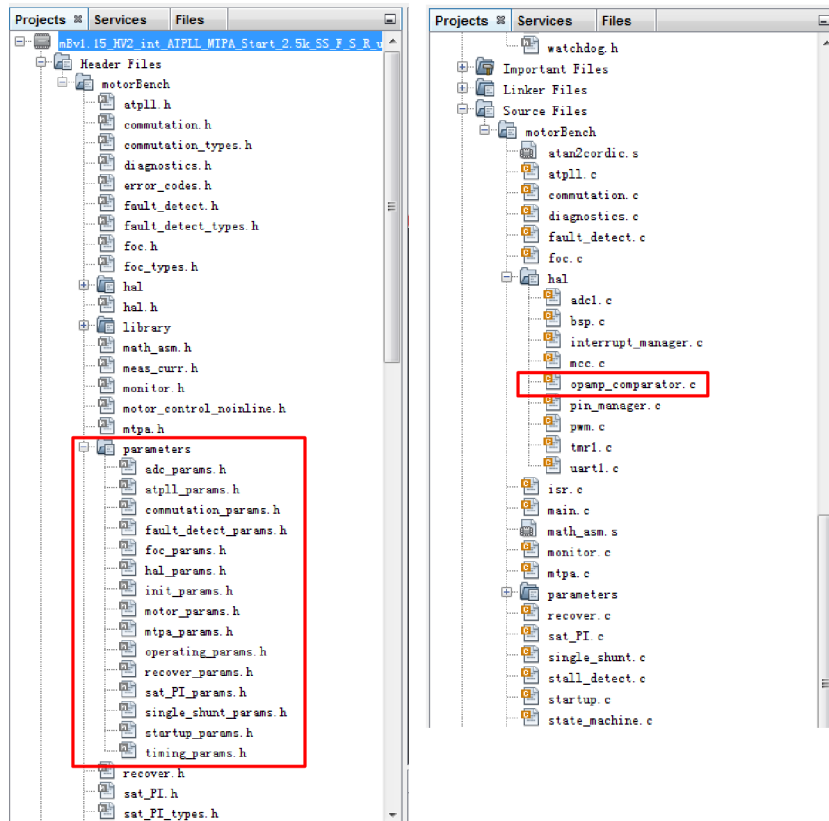
表3-1 测试用具功能和设置的简要汇总

设置 功能	开-关	参数设置							控制设置	
	system Data.testing.key	motor.testing.overrideVdqCmd.d	motor.testing.overrideVdqCmd.q	motor.testing.overrideIdqCmd.d	motor.testing.overrideIdqCmd.q	motor.testing.overrideOmegaElectrical	motor.testing.overrideCommutationFrequency	motor.testing.overrideThetaElectrical	motor.testing.operatingMode	motor.testing.overrides
设置Vd和Vq	53670	值	值	---	---	---	---	---	1	---
设置Id和Iq	53670	---	---	值	值	---	---	---	2	---
设置速度	53670	---	---	---	---	值	---	---	3	1
设置频率	53670	值	值	值	值	---	值	---	1/2	2
设定角度	53670	值	值	值	值	---	---	值	1/2	4
禁止	153670	---	---	---	---	---	---	---	---	---

4 运行其他压缩机

电机控制是一种极其依赖电机和应用程序的系统。每当机制发生变革时，压缩机电机控制都会在可靠启动和不平衡负载方面带来更大的挑战。RCDRD旨在提供一种简单的方法来运行其他部件编号的压缩机。本章介绍如何使用RCDRD来运行并非供开发人员用于调试和测试的压缩机。

要运行其他压缩机，需要修改某些参数以便与压缩机匹配。除了文件夹“hal”中的一个C源文件“opamp_comparator.c”之外，所有需要修改的代码文件均为文件夹“parameters”中的头文件。



4.1 修改理论算术参数

- (1) 由于dsPIC是一种定点控制器，因此不能直接在代码中使用电机参数，因为它们是小数数据。因此，我们必须将物理单位转换为定点PU格式（主要是Q15）。
 - “tuning_params.xlsx”用于实现这种转换。“tuning_params.xlsx”文件中有多个参数，具体说明如下。

表4-1 “tuning_params.xlsx” 中的参数类型

类型	属性	标记
	输入	硬件板参数的物理单位。
	输入	电机参数的物理单位。

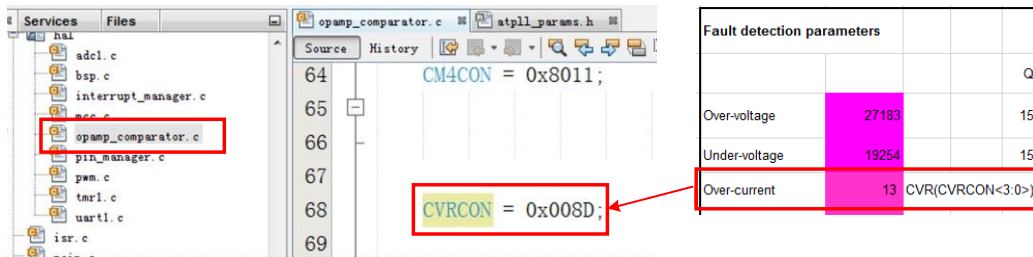
	输出	定点格式的开发板和电机参数。在“xxxx_params.h”中设置该常数。
	输出	定点格式的启动算法参数。在“startup_params.h”中设置该常数。
其他		临时计算数据。

要获得定点参数，应按如下所示在“tuning_params.xlsx”中输入电机参数。

Motor parameters	
Pole pairs	3 -
Stator resistance (Rs)	8.85 Ohm
Stator inductance (Ld)	0.0695 H
Stator inductance (Lq)	0.1125 H
Stator inductance (Ls)	0.091 H
Voltage constant (Kfi)	75.6892 Vpll/KRPM
Minimum speed	1200 RPM
Nominal speed	3000 RPM
Maximum speed	4500 RPM
J (rotor + load inertia)	0.001 Nm/(rad/s ²)
B (rotor viscous damping)	0.00003 Nm/(rad/s)
Tfr (rotor Coulomb friction torque)	0.004 Nm
Protect current	3.1 A

随后将在紫色背景单元格中生成定点格式的值。为避免Q15实现引起的饱和、分辨率损失和截断错误，建议定点数据应处于6000-26000范围内。如果定点数据超出建议范围（大于26000），则应调整换算值“Q”以符合建议范围。

- (2) 在C源文件“hal/opamp_comparator.c”中的寄存器CVR（CVRCON<3:0>）中修改过流阈值。



- (3) 在头文件“parameters/atpll_params.h”中修改估算器角度跟踪锁相环（Angle Tracking Phase-Locked Loop, ATPLL）的参数。

The screenshot shows the `atpll_params.h` file with several parameters defined. A table on the right lists these parameters with their values and units.

Motor parameters	Value	Unit
NORM_RS	4093	1 4093 15
NORM_LODTBASE	105209	16 6576 11
NORM_L1DTBASE	-24857	16 -1554 11
NORM_LSDBASE	105209	16 6576 11
NORM_INVKFIBAS E	17836	2 8918 14
NORM_DELTAT	14317	1 14317 15
WBASE_DT_LSGA IN	44977	2 22488 14

(4) 在头文件“parameters/fault_detect_params.h”中修改停转检测的软件过流阈值。

The screenshot shows the `fault_detect_params.h` file with a threshold for overcurrent stall detection. A table on the right lists the fault detection parameters.

Fault detection parameters	Value	Unit
Over-voltage	27183	15
Under-voltage	19254	15
Over-current	13	CVR(CVRCON<3:0>)
Stall over-current	24933	15

(5) 在头文件“parameters/motor_params.h”中修改电机参数。

The screenshot shows the `motor_params.h` file with various motor parameters defined. A table on the right lists these parameters with their values and units.

Motor parameters	Value	Unit
Pole pairs	3	-
Stator resistance (Rs)	8.85	Ohm
Stator inductance (Ld)	0.0695	H
Stator inductance (Lq)	0.1125	H
Stator inductance (Ls)	0.091	H
Voltage constant (Kf)	75.6892	Vp/IKRPM
Nominal speed (NOMINAL_SPEED_RPM)	3000	RPM
Maximum speed (MAXIMUM_SPEED_RPM)	4500	RPM
J (rotor + load inertia)	0.001	Nm/(rad/s ²)
B (rotor viscous damping)	0.00003	Nm/(rad/s)
Tfr (rotor Coulomb friction torque)	0.004	Nm

(6) 在头文件“parameters/mtpa_params.h”中修改最大转矩/电流（Maximum Torque per Ampere, MTPA）算法的参数。

The screenshot shows the `mtpa_params.h` file with MTPA parameters defined. A table on the right lists these parameters with their values and units.

Motor parameters	Value	Unit
NORM_RS	4093	1 4093 15
NORM_LODTBASE	105209	16 6576 11
NORM_L1DTBASE	-24857	16 -1554 11
NORM_LSDBASE	105209	16 6576 11
NORM_INVKFIBAS E	17836	2 8918 14
NORM_DELTAT	14317	1 14317 15
WBASE_DT_LSGA IN	44977	2 22488 14

(7) 在头文件“parameters/operating_params.h”中修改电机速度参数。

The screenshot shows the `operating_params.h` file with operating parameters defined. A table on the right lists these parameters with their values and units.

Motor parameters	Value	Unit
Pole pairs	3	-
Stator resistance (Rs)	8.85	Ohm
Stator inductance (Ld)	0.0695	H
Stator inductance (Lq)	0.1125	H
Stator inductance (Ls)	0.091	H
Voltage constant (Kf)	75.6892	Vp/IKRPM
Minimum speed	1200	RPM
Nominal speed	3000	RPM
Maximum speed	4500	RPM

(8) 在头文件“parameters/sat_PI_params.h”中修改速度环的PI饱和和阈值。

The screenshot shows the code editor for `sat_PI_params.h` with several lines highlighted in red boxes. Red arrows point from these boxes to a table on the right titled "Fault detection parameters".

Code Annotations:

- Line 55: `#define CURRENT_SATURATION_THRESHOLD_DOWN 22163` (comment: Q15, 2.48 A)
- Line 57: `#define CURRENT_SATURATION_THRESHOLD_UP 23548` (comment: Q15, 2.635)
- Line 68: `#define CURRENT_MAXIMUM_COMMAND 25773` (comment: // Q15, = 1.03 * ...)

Fault detection parameters table:

Parameter	Value	Unit
Over-voltage	27183	15
Under-voltage	19254	15
Over-current	13	CVR(CVRCOIN<3 D>)
Stall over-current	24933	15
PI saturation current up	22163	15
PI saturation current down	23548	15
Max CMD current	22828	15

(9) 在头文件“parameters/startup_params.h”中修改启动算法的参数。

The screenshot shows the code editor for `startup_params.h` with several lines highlighted in red boxes. Red arrows point from these boxes to a table on the right titled "Startup algorithm parameters".

Code Annotations:

- Line 57: `#define STARTUP_INITIALTORQUE 0`
- Line 58: `#define STARTUP_FINALTORQUE 8256`
- Line 59: `#define STARTUP_TORQUE_RAMPUP_RATE 16`
- Line 66: `#define STARTUP_ACCELO_VELOCITY_THRESHOLD 575`
- Line 67: `#define STARTUP_ACCEL1_VELOCITY_THRESHOLD 3932`
- Line 68: `#define STARTUP_ACCELERATION0 1747`
- Line 69: `#define STARTUP_ACCELERATION1 3281`
- Line 71: `#define STARTUP_DELTA_T_FACTOR 26214`
- Line 72: `#define STARTUP_DELTA_T_FACTOR_ACCELERATION 100`
- Line 75: `#define STARTUP_DAMPING_THRESHOLD 1573`
- Line 76: `#define STARTUP_DAMPING_IQMAX 1341`
- Line 77: `#define STARTUP_DAMPING_SHIFT 8`
- Line 78: `#define STARTUP_DAMPING_GAIN 5365`
- Line 81: `#define STARTUP_HOLDTIME 0`
- Line 84: `#define STARTUP_RAMPDOWN_DECAY_RATE 9914`
- Line 85: `#define STARTUP_RAMPDOWN_DECAY_RATE_Q 24`

Startup algorithm parameters table:

Parameters	value	revise	unit	normalization	revise normalization
drive_max	3.686687		A		
motor_rated_current	1.9		A		
imax	1.5		A		
STARTUP_FINALTORQUE	1.125	0.7	A	10054	6256
STARTUP_TORQUE_RAMPUP_RATE	4.376374	6.15	A/s	16	22
STARTUP_ACCELO_VELOCITY_THRESHOLD	18.38505		rad/s	575	
STARTUP_DELTA_T_FACTOR_ACCELERATION	1	100		1	100
STARTUP_ACCELERATION0	20.97563	213	rad/s ²	17206	1747
STARTUP_ACCEL1_VELOCITY_THRESHOLD	125.6637		rpm	3932	0
STARTUP_ACCELERATION1	48.88456	400	rad/s ²	40099	3281
STARTUP_DAMPING_THRESHOLD	50.26548		rad/s	1573	
STARTUP_DAMPING_GAIN	0.073381		A/(rad/s)	5365	
STARTUP_DAMPING_IQMAX	0.15		A	1341	
STARTUP_RAMPDOWN_DECAY_RATE	1.477231		1/s	9914	
STARTUP_RAMPDOWN_END_CURRENT	0.05625	0.7	A	503	6256
STARTUP_DELTA_T_FACTOR	26214.4			26214	

(10) 返回第3章“设置和运行”，编程并运行代码。

4.2 调试参数

4.2.1 调试启动参数

RCDRD中的启动算法专用于压缩机应用。它不仅有助于改善启动可靠性，还能减少振动。所有启动算法参数均位于头文件“parameters/startup_params.h”中。更换压缩机时，需要调试其中一些参数。

请参见表4-2“tuning_params.xlsx”的“启动算法参数”部分，“值”列是理论计算值，“归一化”列是归一化值。“修正”列是调试值，“修正归一化”列是归一化调试值。

表4-2 “tuning_params.xlsx”的“启动算法参数”部分

Startup algorithm parameters					
Parameters	value	revise	unit	normalization	revise normalization
drive.Imax	3.668667		A		
motor.rated_current	1.5		A		
Imax	1.5		A		
STARTUP_FINALTORQUE	1.125	0.7	A	10054	6256
STARTUP_TORQUE_RAMPUP_RATE	4.376374	6.15	A/s	16	22
STARTUP_ACCELO_VELOCITY_THRESHOLD	18.38505		rad/s	575	
STARTUP_DELTA_T_FACTOR_ACCELERATION	1	100		1	100
STARTUP_ACCELERATION0	20.97563	213	rad/s ²	17206	1747
STARTUP_ACCEL1_VELOCITY_THRESHOLD	125.6637		rad/s rpm	3932	0
STARTUP_ACCELERATION1	48.88456	400	rad/s ²	40099	3281
STARTUP_DAMPING_THRESHOLD	50.26548		rad/s	1573	
STARTUP_DAMPING_GAIN	0.073381		A/(rad/s)	5365	
STARTUP_DAMPING_IQMAX	0.15		A	1341	
STARTUP_RAMPDOWN_DECAY_RATE	1.477231		1/s	9914	
STARTUP_RAMPDOWN_END_CURRENT	0.05625	0.7	A	503	6256
STARTUP_DELTA_T_FACTOR	26214.4			26214	

更换压缩机时，上表中只有几个参数需要微调，具体说明如下。

表4-3 启动的关键参数

参数	说明	调试
STARTUP_FINALTORQUE	开环电流	如果启动负载很大，则增大此参数
STARTUP_ACCELERATION0	第一速度加速状态的加速度	如果需要更短的启动时间，则增大此参数
STARTUP_ACCELERATION1	第二速度加速状态的加速度	如果需要更短的启动时间，则增大此参数
STARTUP_RAMPDOWN_END_CURRENT	从开环切换到闭环的电流阈值	此参数始终等于压缩机应用中的开环电流

4.2.2 调试PI参数

更换压缩机时，可能需要调整电流环和速度环的PI控制器参数。所有PI参数均位于头文件“parameters/for_params.h”中。

```

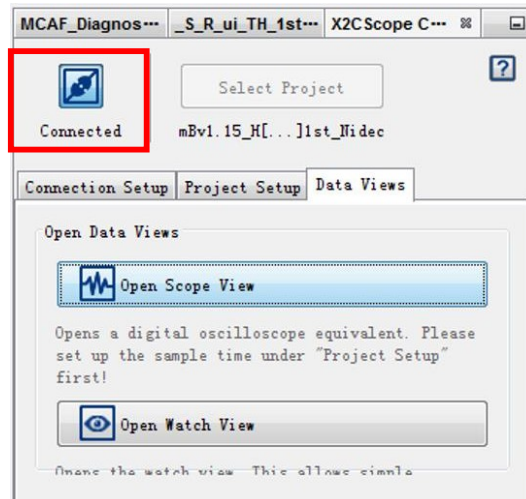
53
54 //***** PI Coefficients *****
55 /// Current loop
56 // phase margin = 80 deg
57 // PI phase at crossover = 45.000 deg
58 // crossover frequency = 1.547 k rad/s (246.207 Hz)
59 #define KIP 7000
60 #define KIP_Q 12 //
61 #define KII 200
62 #define KII_Q 15
63 /// Velocity loop
64 // phase margin = 50 deg
65 // PI phase at crossover = 10.000 deg
66 // crossover frequency = 90.725 rad/s (14.439 Hz)
67 #define KWP 30000
68 #define KWP_Q 13
69 #define KWI 2000
70 #define KWI_Q 15
    
```

Kxx_Q (KIP_Q、KII_Q、KWP_Q和KWI_Q) 是换算值。将Kp和Ki乘以 2^{Kxx_Q} 可以得到代码中使用的最终PI控制器数据。在调试工作的早期阶段，更改此换算值而不是Kp (Ki) 将有助于缩短调试时间，最终找到大致的Kp (Ki) 值。

5 已知问题

(1) 计算机无法通过X2C Scope连接到RCDB。有几种原因可能导致这种问题。

- **情形1:** 在不断开X2C Scope连接的情况下，在MPLAB X中打开一个新项目。



解决方案: 重新启动MPLAB X，然后将程序重新下载到开发板中。

- **情形2:** 有时芯片可能会进入看门狗超时保护状态。这可能是在修改代码时由某种错误引起。检查LED（LD200）的闪烁情况以获取芯片运行状态。

解决方案: 使RCDB掉电。重新连接UART1端口并为开发板上电。

附录A. 标志说明

以下各表格说明了RCDRD的所有状态标志。

表A-1 系统状态

系统状态	
变量	motor.state
标志	枚举
0	MCSM_RESTART
1	MCSM_STOPPED
2	MCSM_STARTING
3	MCSM_RUNNING
4	MCSM_STOPPING
5	MCSM_FAULT
6	MCSM_TEST_DISABLE
7	MCSM_TEST_ENABLE

表A-2 错误代码

错误代码	
变量	motor.faultDetect.faultDetectFlagBackup
标志	枚举
1	OVERVOLTAGE
2	UNDERVOLTAGE
4	OVERCURRENT
8	DIRECTREVERSE
16	SPEEDERROR
32	STALL

表A-3 停转检测代码

停转代码	
变量	motor.stallDetect.stallDetectFlagBackup
标志	枚举
1	OVERCURRENT
2	LOSS_OF_LOCK
4	LOW_SPEED
8	TORQUE_ANGLE

附录B. 原理图

