
ROBSTRIDE 04

目录

注意事项

1、请按照本文规定的工作参数使用，否则可能会对本产品造成严重的损坏！
2、在关节运行时不可切换控制方式，如需切换需要发送停止运行命令后再做切换。
3、使用前请检查各部件是否完好，如发生部件缺失、损坏请及时联系技术支持。
4、请勿随意拆卸电机，以免出现无法恢复的故障。
5、确保电机连接时无短路，接口按要求正确连接。

法律声明

在使用本产品前，请用户务必仔细阅读本手册，按照本手册内容操作本产品。如用户违反本手册内容使用本产品，造成的任何财产损失、人身伤害事故，本公司不承担任何责任。因本产品由众多零部件构成，切勿让儿童接触本产品，以免发生意外事故。为延长产品使用寿命，请勿在高温、高压环境中使用本产品。本手册在印刷时已尽可能的包含各项功能介绍和使用说明。但由于产品功能不断完善、设计变更等，仍可能与用户购买的产品有不符之处。

本手册与实际产品在颜色、外观等方面可能有所偏差，请以实际产品为准。本手册由北京灵足时代科技有限公司（下简称灵足）出版，灵足随时可能对本手册中不准确的最新信息进行必要的改进和更改，或对程序和/或设备进行改进。此类更改将上传电子版说明书到公司官网，详情可查看下载中心栏（www.robstride.com）。所有图片仅供功能说明参考，请以实物为准。

售后政策

本产品售后服务严格依据《中华人民共和国消费者权益保护法》、《中华人民共和国产品质量法》实行售后服务，服务内容如下：

1、保修期限及内容

（1）凡在线上渠道下单购买本产品的用户，可在自签收次日起七日内享受无理由退货服务。退货时用户须出示有效购买凭证，并退回发票。用户须保证退货商品保持原有品质和功能、外观完好、商品本身及配件的商标和各种标识完整齐全，如有赠品需一并退回。如果商品出现人为损坏、人为拆机、包装箱缺失、零配件缺失的情况，不予办理退货。退货时产生的物流费用由用户承担（收费标准见“售后服务收费标准”）。如果用户未结清物流费用，将按实际发生额从退款金额中扣除。自收到退货商品之日起七日内向用户返还已支付的货款。退款方式与付款方式相同。具体到账日期可能会受银行、支付机构等因素影响。

（2）本产品保修期为1年。

（3）自用户签收次日起7天内，发生非人为损坏性能故障，经由灵足售后服务中心检测确认后，为用户办理退货业务，退货时用户须出示有效购买凭证，并退回发票。如有赠品需一并退回。

（4）自用户签收次日起7天后至15天内，发生非人为损坏性能故障，经由灵足售后服务中心检测确认后，为用户办理换货业务，更换整套商品。换货后，商品本身三包期重新计算。

（5）自用户签收次日起15天后至365天内，经由灵足售后服务中心检测确认后，属于产品本身质量故障，可免费提供维修服务。更换的故障产品归灵足公司所有。无故障产品，将原样返回。本产品经过各项严格检测后出厂，如有非产品本身质量故障，我们将有权拒绝用户的退换货需求。本手册售后政策若与店铺售后政策不一致的，以店铺的售后政策为准。

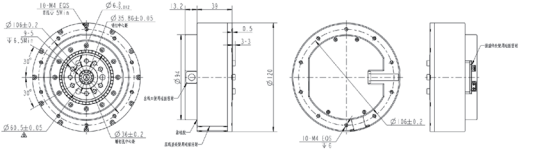
2、非保修条例以下情况不属于保修范围：

1. 超出保修条款所限定的保修期限。
2. 未按照说明书要求，错误使用造成的产品损坏损毁。
3. 不当的操作、维修、安装、改装、测试等不正当使用造成的损坏损毁。
4. 非质量故障引起的常规机械损耗、磨损。
5. 非正常工况下造成的损坏，包括但不限于跌落、撞击、液体浸入、剧烈撞击等。
6. 天灾（如水灾、火灾、雷击、地震等）或不可抗击力造成的损坏。
7. 超过峰值扭矩使用造成的损坏。
8. 非灵足原装正品或无法提供合法购买凭证。
9. 其他非产品的设计、技术、制造、质量等问题导致的故障或损坏。
10. 将本产品应用于商业用途。

如果出现上述情况，用户需自行支付费用。

1 电机规格参数

1.1 外形及安装尺寸

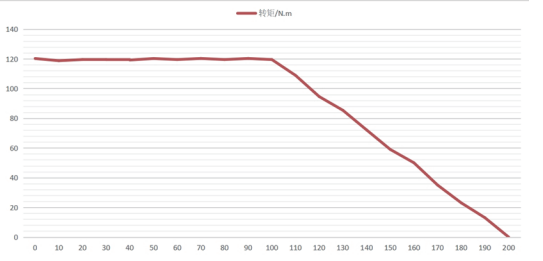


1.2 标准使用状态

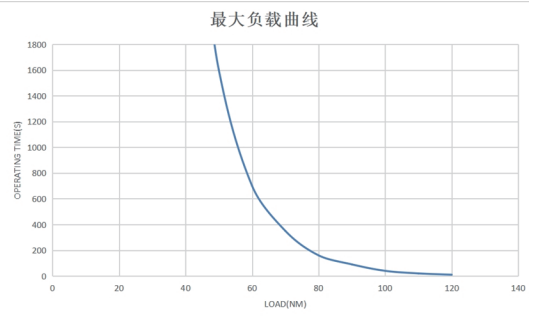
1.2.1 额定电压：48 VDC
1.2.2 使用电压范围：24V—60 VDC
1.2.3 额定负载 (CW)：40 N.m
1.2.4 运转方向：CW/CCW 从出轴方向看
1.2.5 使用姿势：出轴方向为水平或者垂直
1.2.6 标准使用温度：25±5°C
1.2.7 使用温度范围：-20～50°C
1.2.8 标准使用湿度：65%
1.2.9 使用湿度范围：5～85%,无凝露
1.2.10 保存温度范围：-30～70°C
1.2.11 绝缘等级：Class B

1.3 电气特性

1.3.1 空载转速：200 rpm±10%
1.3.2 空载电流：2 Arms
1.3.3 额定负载：40 N.m
1.3.4 额定负载转速：167rpm±10%
1.3.5 额定负载相电流(峰值)：27Apk±10%
1.3.6 峰值负载：120 N.m
1.3.7 最大负载相电流(峰值)：90Apk±10%
1.3.8 绝缘电阻/定子绕组：DC 500VAC, 100M Ohms
1.3.9 耐高压/定子与机壳：600 VAC, 1s, 2mA
1.3.10 电机反电势：16.9Vrms/krpm±10%
1.3.11 转矩常数：2.1N.m/Arms
1.3.12 T-N曲线



1.3.13 最大过载曲线
测试条件：
环境温度：25°C
绕阻极限温度：130°C（此为约束温度，实际为180度）
转速：24rpm



测试数据

Load	Operating time(s)
120	10
110	20
100	40
90	90
80	160
70	350
60	700
50	1600
46	2500
40	rated

1.4 机械特性
1.4.1 重量：1420g±20g
1.4.2 极数：42极
1.4.3 相数：3相
1.4.4 驱动方式：FOC
1.4.5 减速比：9：1

2 驱动器产品信息

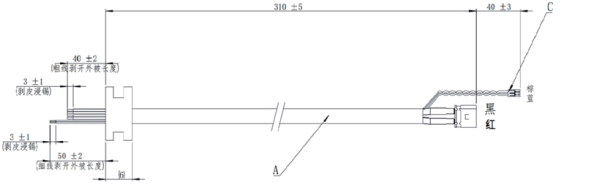
2.1 驱动器产品规格

产品规格	
额定工作电压	48VDC
允许最大电压	60VDC
额定工作 相电流	27Apk
最大允许 相电流	90Apk
待机功耗	≤40mA
CAN总线比特率	1Mbps
尺寸	Φ 84mm
工作环境温度	- 20°C 至 50°C
控制板允许最大温度	80°C
编码器分辨率	14bit （单圈绝对值）

2.2 驱动器接口定义

2.2.1 驱动器接口定义

2.2.2 驱动器线束定义



线束定义	
蓝	CAN_H
棕	CAN_L
黑	GND
红	VBAT+

2.2.3 驱动器接口推荐品牌及型号

板端型号	品牌厂家	线端型号	品牌厂家
XT30APW-M	AMASS（艾迈斯）	XT30UW-F	AMASS（艾迈斯）
GH1.25-2PWT	任意	GH1.25-T	任意

2.3 主要器件及规格

序号	项目	规格	数量
1	MCU　芯片	GD32F103RET6	1 PCS
2	驱动芯片	DRV8353SRTAT	1 PCS
3	磁编码器芯片	AS5047P	1 PCS
4	热敏电阻	LTS00　-104J395T19E010/ NCP18XH103F03RB	2 PCS
5	功率　MOS	ISC030N12NM6	12 PCS

3 上位机使用说明(请前往www.robstride.com官网下载中心下载)

3.1 硬件配置

关节电机采用CAN通信方式，通信线有两根，通过can转USB工具与调试器相连，调试器需要提前安装ch340驱动，默认工作在AT模式。需要注意的是，我们是根据特定的can转USB工具开发的调试器，因此需要用我们推荐的串口工具来进行调试器调试，如果想要移植到其他调试器平台可以参照说明书的第三章进行开发。

can转USB工具推荐使用YourCee的USB-CAN模块，对应串口协议的帧头为41 54，帧尾为0D 0A。

3.2 上位机界面及说明

主要包括：
A. 模块选择
- 设备模块
- 配置模块
- 分析模块
- 帮助模块

B. 子模块选择
设备模块包括
- 连接或断开电机设备
- 电机设备信息
- 电机编码器标定
- 修改电机CAN ID
- 设置电机的机械零位
- 电机程序升级

配置模块包括：
- 参数表，可以查看并修改电机参数
- 上传参数，可以将电机中参数上传到参数表中
- 下载参数，可以将参数表中数据下载到电机中
- 导出参数，可以将参数表中数据下载到本地
- 恢复出厂，可以将参数表中数据恢复出厂设置
- 清除警告，可以清除电机报错，如温度过高等

分析模块包括：
- 示波器，可以查看参数随时间变化曲线
- 频率，可以调整查看数据的频率
- 信道，可以配置查看的数据
- 开始、停止绘图
- 输出波形数据到本地

帮助模块包括：
- 使用说明，可以打开使用说明书
- 关于，可以查看软件信息

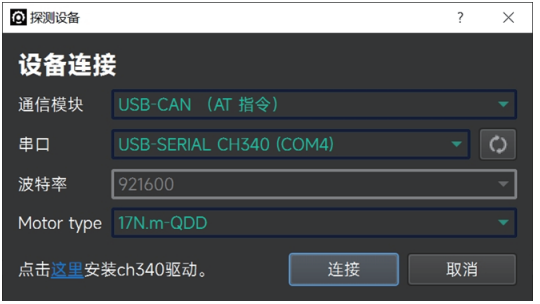
C. 电机信息查询
- 设备信息
- 参数表信息

D. 数据栏
- 日志信息
- 通信信息

E. 运行调试区
- 选择设备
- 便捷操作区，可以快速控制电机正反转
- 运动控制区，可以控制电机按各模式运行

F. 子模块显示区

3.3 电机设置
3.3.1 电机连接设置

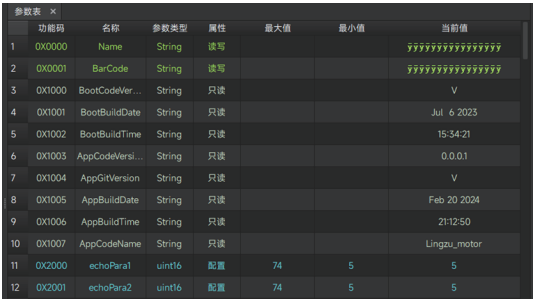


连接can转USB工具（安装ch340驱动，默认工作在AT模式），点击设备模块中的连接子模块，选择对应串口连接、电机类型，点击连接。

3.3.2 基本设置

(1) 修改电机id号。
(2) 电机磁编标定，电机板与电机重新安装，或电机三相线重新换顺序连接等，需要重新进行磁编标定。
(3) 设置零位（掉电丢失），设置当前位置为0。
(4) 电机程序升级，当电机程序有更新时，点击升级按钮选中升级文件即可进行升级。

3.3.3 参数表



成功连接电机后，点击配置模块中的参数表模块，日志中会显示全部参数加载成功，说明成功读取到电机相关参数（注：参数表需要在电机处于待机状态下进行配置，如果电机处于运行状态则无法进行参数表刷新），界面会显示电机的相关参数，蓝色的参数为电机内部的存储参数，可以在相应参数后面的当前值栏进行修改，点击下载参数可以将调试器中参数下载到电机中，点击上传参数可以将电机中的参数上传到调试器中，电机恢复绿色参数为观测参数，为采集得到的参数，可进行实时观测。

注：电机的转矩限制、保护温度、过温时间请勿随意更改。因违规操作本产品导致对人体造成伤害，或对关节造成不可逆的损伤，我司将不承担任何法律责任。

参数表							
功能码	名称	参数类型	属性	最大值	最小值	当前值（供参考）	备注
0X0000	Name	String	读/写			ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ	
0X0001	BarCode	String	读/写			ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ	
0X1000	BootCodeVersion	String	只读			0.1.5	
0X1001	BootBuildDate	String	只读			Mar 16 2022	
0X1002	BootBuildTime	String	只读			20:22:09	
0X1003	AppCodeVersion	String	只读			0.0.0.1	电机程序版本号
0X1004	AppGitVersion	String	只读			7b844b0fM	
0X1005	AppBuildDate	String	只读			Apr 14 2022	
0X1006	AppBuildTime	String	只读			20:30:22	
0X1007	AppCodeName	String	只读			Lingzu_motor	
0X2000	echoPara1	uint16	配置	74	5	5	
0X2001	echoPara2	uint16	配置	74	5	5	
0X2002	echoPara3	uint16	配置	74	5	5	
0X2003	echoPara4	uint16	配置	74	5	5	
0X2004	echoFreHz	uint32	读/写	10000	1	500	
0X2005	MechOffset	float	设定	7	-7	4.619583	低速端位置偏置值
0X2006	chasu_offset	float	读/写	6.28	0	4.52	差速偏置值
0X2007	ElecOffset	float	读/写	6.28	0	3.27	电角度偏置值
0X2008	I_FW_MAX	float	读/写	33	0	0	弱磁电流值，默认 0

0X2009	CAN_ID	uint8	设定	127	0	1	本节点 id
0X200a	CAN_MASTER	uint8	设定	127	0	0	can 主机 id
0X200b	CAN_TIMEOUT	uint32	读/写	100000	0	0	can 超时阈值，默认 0
0X200c	motorOverTemp	int16	读/写	1500	0	800	电机保护温度值，temp（度）*10
0X200d	overTempTime	uint32	读/写	1000000	1000	20000	过温时间
0X200e	GearRatio	float	读/写	64	1	9	传动比
0x200f	Kt_Nm/Amp	float	读/写	1	1	0	0
0X2010	Tq_caliType	uint8	读/写	1	0	1	转矩标定方法设定
0X2011	cur_filt_gain	float	读/写	1	0	0.6	电流滤波参数
0X2012	cur_kp	float	读/写	200	0	0.05	电流 kp
0X2013	cur_ki	float	读/写	200	0	0.05	电流 ki
0X2014	spd_kp	float	读/写	200	0	2	速度 kp
0X2015	spd_ki	float	读/写	200	0	0.021	速度 ki
0X2016	loc_kp	float	读/写	200	0	30	位置 kp
0X2017	spd_filt_gain	float	读/写	1	0	0.1	速度滤波参数
0X2018	limit_spd	float	读/写	200	0	2	位置模式速度限制
0X2019	limit_cur	float	读/写	90	0	90	位置、速度模式　电流限制
0X3000	timeUse0	uint16	只读			5	
0X3001	timeUse1	uint16	只读			0	
0X3002	timeUse2	uint16	只读			10	
0X3003	timeUse3	uint16	只读			0	

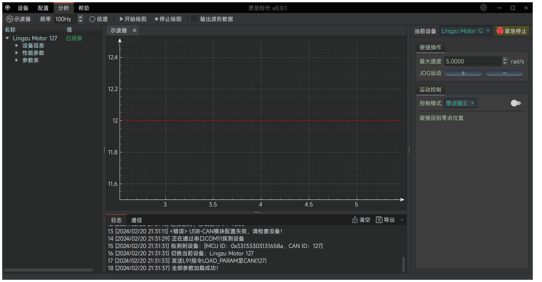
0X3004	encoderRaw	int16	只读			11396	磁编码器采样值
0X3005	mcuTemp	int16	只读			337	mcu 内部温度，*10
0X3006	motorTemp	int16	只读			333	电机 ntc 温度，*10
0X3007	vBus(mv)	uint16	只读			24195	母线电压
0X3008	adc1Offset	int32	只读			2084	adc 采样通道 1 零电流偏置
0X3009	adc2Offset	int32	只读			2084	adc 采样通道 2 零电流偏置
0X300a	adc1Raw	uint16	只读			1232	adc 采样值 1
0X300b	adc2Raw	uint16	只读			1212	adc 采样值 2
0X300c	VBUS	float	只读			24.195	母线电压 V
0X300d	cmdId	float	只读			0	id 环指令，A
0X300e	cmdIq	float	只读			0	iq 环指令，A
0X300f	cmdlocref	float	只读			0	位置环指令，rad
0X3010	cmdspdref	float	只读			0	速度环指令，rad/s
0X3011	cmdTorque	float	只读			0	转矩指令，nm
0X3012	cmdPos	float	只读			0	mit 协议角度指令
0X3013	cmdVel	float	只读			0	mit 协议速度指令
0X3014	rotation	int16	只读			1	圈数
0X3015	modPos	float	只读			4.363409	电机未计圈机械角度，rad
0X3016	mechPos	float	只读			0.777679	负载端计圈机械角度，rad
0X3017	mechVel	float	只读			0.036618	负载端转速,rad/s
0X3018	elecPos	float	只读			4.714761	电气角度

0X3019	ia	float	只读			0	U 线电流，A
0X301a	ib	float	只读			0	V 线电流，A
0X301b	ic	float	只读			0	W 线电流，A
0X301c	tick	uint32	只读			31600	
0X301d	phaseOrder	uint8	只读			0	标定方向标记
0X301e	iqf	float	只读			0	iq 滤波值，A
0X301f	boardTemp	int16	只读			359	板上温度，*10
0X3020	iq	float	只读			0	iq 原值，A
0X3021	id	float	只读			0	id 原值，A
0X3022	faultSta	uint32	只读			0	故障状态值
0X3023	warnSta	uint32	只读			0	警告状态值
0X3024	drv_fault	uint16	只读			0	驱动芯片故障值
0X3025	drv_temp	int16	只读			48	驱动芯片温度值，度
0X3026	Uq	float	只读			0	q 轴电压
0X3027	Ud	float	只读			0	d 轴电压
0X3028	dtc_u	float	只读			0	U 相输出占空比
0X3029	dtc_v	float	只读			0	V 相输出占空比
0X302a	dtc_w	float	只读			0	W 相输出占空比
0X302b	v_bus	float	只读			24.195	闭环中 vbus
0X302c	v_ref	float	只读			0	闭环 vq,vd 合成电压
0X302d	torque_fdb	float	只读			0	转矩反馈值，nm

0X302e	rated_i	float	只读			8	电机额定电流
0X302f	limit_i	float	只读			27	电机限制最大电流

3.3.4 示波器

该界面支持观看观察实时数据所生成的图谱，可观测的数据包括电机Id/Iq电流、温度、输出端实时转速、转子（编码器）位置、输出端位置等。点击分析模块中的示波器模块，信道内选定合适的参数（参数含义可参考3.3.3），设置输出频率后点击开始绘图即可观测数据图谱，停止绘图即可停止观测图谱。



3.3.5 通信框指令说明

通信框指令示例：

41 54 90 07 e8 0c 08 05 70 00 00 01 00 00 00 0d 0a

含义如下

41 54	90 07 e8 0c	08	05 70 00 00 01 00 00 00	0d 0a
帧头	扩展帧	数据位个数	数据帧	帧尾

其中扩展帧canid转译为真实canid需要经过以下转换：

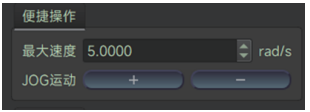
90 07 e8 0c转换成二进制为1001 0000 0000 0111 1110 1000 0000 1100，去掉右边的100，则为1 0010 0000 0000 1111 1101 0000 0001，将其转换为16进制，为 12 00 FD 01，对照通信协议说明，含义如下：

12（16 进制）	00	FD	01
通信类型 18（10 进制）	无含义	主机 id	电机 canid

3.3.6 can通信故障保护
当CAN_TIMEOUT值为0时，该功能不启用
当CAN_TIMEOUT值为非0时，当电机在一定时间段内没收到can指令时，电机进入reset模式，12000为1s

3.4控制演示

jog运行：
设置最大速度，点击运行后，点击JOG运行即可让电机正反运行



控制模式切换：
在运动模式界面可以进行电机控制模式的转换

3.4.1 零点模式

点击右侧开关按钮，电机会缓慢回到机械零位位置

3.4.2 运控模式

点击右侧开关按钮，然后设置五个参数值，点击开始或连续发送，电机将返回反馈帧并按目标指令运行；再次点击右侧开关按钮，电机将停机。

3.4.3 电流模式

手动切换电流模式，点击右侧开关按钮，然后设置Iq电流指令值，开始或连续发送，电机将跟随电流指令运行，再次点击右侧开关按钮，电机将停机。

点击控制模式右侧开关按钮，输入正弦化自动测试的幅值和频率，然后点击正弦化自动测试右侧开关按钮，电机的iq（A）会按设定的幅值和频率来运行。

3.4.4 速度模式

手动切速度模式，点击右侧开关按钮，然后设置速度指令值，开始或连续发送，电机将跟随速度指令运行，再次点击右侧开关按钮，电机将停机。

点击控制模式右侧开关按钮，输入正弦化自动测试的幅值和频率，然后点击正弦化自动测试右侧开关按钮，电机的速度（rad/s）会按设定的幅值和频率来运行。

3.4.5 位置模式

手动切换位置模式，点击右侧开关按钮，然后设置位置指令值（rad），开始或连续发送，电机将跟随目标位置指令运行，再次点击右侧开关按钮，电机将停机。可通过设置速度，修改位置跟随的最大速度。

点击控制模式右侧开关按钮，输入正弦化自动测试的幅值和频率，然后点击正弦化自动测试右侧开关按钮，电机的位置（rad）会按设定的幅值和频率来运行。

3.5 固件更新

第一步，点击设备模块的升级，选择待烧录bin文件；第二步，确认升级，电机开始更新固件，进度完成后，电机更新完成，自动重启。

4 驱动器通信协议及使用说明

电机通信为CAN 2.0通信接口，波特率1Mbps，采用扩展帧格式，如下所示：

数据域	29 位 ID			8Byte 数据区
大小	Bit28~bit24	bit23~8	bit7~0	Byte0~Byte7
描述	通信类型	数据区 2	目标地址	数据区 1

电机支持的控制模式包括：
运控模式:给定电机运控5个参数；
电流模式:给定电机指定的Iq 电流；
速度模式:给定电机指定的运行速度；
位置模式:给定电机指定的位置，电机将运行到该指定的位置；

4.1 通信协议类型说明

4.1.1 获取设备ID（通信类型0）；获取设备的ID和64位MCU唯一标识符

数据域	29 位 ID			8Byte 数据区
大小	Bit28~bit24	bit23~8	bit7~0	Byte0~Byte7
描述	0	bit15~8:用来标识主机 CAN_ID	目标电机 CAN_ID	0

应答帧：

数据域	29 位 ID			8Byte 数据区
大小	Bit28~bit24	bit23~8	bit7~0	Byte0~Byte7
描述	0	目标电机 CAN_ID	0XFE	64 位 MCU 唯一标识符

4.1.2 运控模式电机控制指令（通信类型1）用来向电机发送控制指令

数据域	29 位 ID			8Byte 数据区
大小	Bit28~bit24	bit23~8	bit7~0	Byte0~Byte7
描述	1	Byte2:力矩(0~65535) 对应（-120Nm~120Nm）	目标电机 CAN _ID	Byte0~1：目标角度[0~65535]对应(-4π~4π) Byte2~3：目标角速度[0~65535]对应(-15rad/s~15rad/s) Byte4~5：Kp [0~65535]对应(0.0~5000.0) Byte6~7：Kd [0~65535]对应(0.0~100.0) 以上数据转换后高字节在前，低字节在后

应答帧:应答电机反馈帧(见通信类型2)

4.1.3 电机反馈数据 (通信类型2) 用来向主机反馈电机运行状态

数据域	29 位 ID			8Byte 数据区
大小	Bit28~bit24	bit23~8	bit7~0	Byte0~Byte7
描述	2	Bit8~Bit15:当前电机 CAN ID bit21~16:故障信息（0 无 1 有） bit21: 未标定 bit20: HALL 编码故障 bit19: 磁编码故障 bit18: 过温 bit17: 过流 bit16: 欠压故障 bit22~23:模式状态 0 : Reset 模式[复位] 1 : Cali 模式[标定] 2 : Motor 模式[运行]	主机 CAN _ID	Byte0~1: 当前角度 [0~65535]对应(-4π~4π) Byte2~3: 当前角速度 [0~65535]对应(-15rad/s~15rad/s) Byte4~5:当前力矩 [0~65535]对应（-120Nm~120Nm） Byte6~7:当前温度：Temp(摄氏度）*10 以上数据高字节在前，低字节在后

4.1.4 电机使能运行 (通信类型3)

数据域	29 位 ID			8Byte 数据区
大小	Bit28~bit24	bit23~8	bit7~0	Byte0~Byte7
描述	3	bit15~8:用来标识主 CAN_ID	目标电机 CAN_ID	

应答帧:应答电机反馈帧(见通信类型2)
4.1.5 电机停止运行 (通信类型4)

数据域	29 位 ID			8Byte 数据区
大小	Bit28~bit24	bit23~8	bit7~0	Byte0~Byte7
描述	4	bit15~8:用来标识主 CAN_ID	目标电机 CAN_ID	正常运行时，data 区需清 0； Byte[0]=1 时：清故障；

应答帧:应答电机反馈帧(见通信类型2)
4.1.6设置电机机械零位(通信类型6)会把当前电机位置设为机械零位(掉电丢失)

数据域	29 位 ID			8Byte 数据区
大小	Bit28~bit24	bit23~8	bit7~0	Byte0~Byte7
描述	6	bit15~8:用来标识主 CAN_ID	目标电机 CAN_ID	Byte[0]=1

应答帧:应答电机反馈帧(见通信类型2)
4.1.7设置电机CAN_ID(通信类型7)更改当前电机CAN_ID,立即生效。

数据域	29 位 ID			8Byte 数据区
大小	Bit28~bit24	bit23~8	bit7~0	Byte0~Byte7
描述	7	bit15~8:用来标识主 CAN_ID Bit16~23: 预设置 CAN_ID	目标电机 CAN_ID	

应答帧:应答电机广播帧(见通信类型0)
4.1.8单个参数读取(通信类型17)

数据域	29 位 ID			8Byte 数据区
大小	Bit28~bit24	bit23~8	bit7~0	Byte0~Byte7
描述	17	bit15~8:用来标识主 CAN_ID Bit23~16:00 为读取成功 01 为读取失败	目标电机 CAN_ID	Byte0~1: index，参数列表详见 4.1.11 Byte2~3: 00 Byte4~7: 00 以上数据低字节在前，高字节在后

应答帧：

数据域	29 位 ID			8Byte 数据区
大小	Bit28~bit24	bit23~8	bit7~0	Byte0~Byte7
描述	17	bit15~8:目标电机 CAN_ID	主机 CAN_ID	Byte0~1: index，参数列表详见 4.1.11 Byte2~3: 00 Byte4~7: 参数数据，1 字节数据在 Byte4 以上数据低字节在前，高字节在后

4.1.9 单个参数写入（通信类型 18） （掉电丢失）

数据域	29 位 ID			8Byte 数据区
大小	Bit28~bit24	bit23~8	bit7~0	Byte0~Byte7
描述	18	bit15~8:用来标识主 CAN_ID	目标电机 CAN_ID	Byte0~1: index，参数列表详见 4.1.11 Byte2~3: 00 Byte4~7: 参数数据 以上数据低字节在前，高字节在后

应答帧：应答电机反馈帧(见通信类型 2)

4.1.10 故障反馈帧（通信类型 21）

数据域	29 位 ID			8Byte 数据区
大小	Bit28~bit24	bit23~8	bit7~0	Byte0~Byte7
描述	21	bit15~8:用来标识主 CAN_ID	电机 CAN_ID	Byte0~3: fault 值(非 0:有故障，0：正常） bit16:A 相电流采样过流 bit15~bit8:过载故障 bit7:编码器未标定 bit5:C 相电流采样过流 bit4:B 相电流采样过流 bit3:过压故障 bit2:欠压故障 bit1:驱动芯片故障 bit0:电机过温故障，默认 80 度 Byte4~7: warning 值 bit0：电机过温预警，默认 75 度

4.1.11 可读写单个参数列表

参数 index	参数名称	描述	类型	字节数	单位/说明	R/W 读写权限
0X7005	run_mode	0: 运控模式 1: 位置模式 2: 速度模式 3: 电流模式	uint8	1		W/R
0X7006	iq_ref	电流模式 Iq 指令	float	4	-90~90A	W/R
0X700A	spd_ref	转速模式转速指令	float	4	-15~15rad/s	W/R
0X700B	imit_torque	转矩限制	float	4	0~120Nm	W/R
0X7010	cur_kp	电流的 Kp	float	4	默认值 0.05	W/R

0X7011	cur_ki	电流的 Ki	float	4	默认值 0.05	W/R
0X7014	cur_filt_gain	电流滤波系数 filt_gain	float	4	0~1.0，默认值 0.06	W/R
0X7016	loc_ref	位置模式角度指令	float	4	rad	W/R
0X7017	limit_spd	位置模式速度限制	float	4	0~15rad/s	W/R
0X7018	limit_cur	速度位置模式电流限制	float	4	0~90A	W/R
0x7019	mechPos	负载端计圈机械角度	float	4	rad	R
0x701A	iqf	iq 滤波 值	float	4	-90~90 A	R
0x701B	mechVel	负载端转速	float	4	-15~15 rad/s	R
0x701C	VBUS	母线电压	float	4	V	R
0x701E	loc_kp	位置的 kp	float	4	默认值 30	W/R
0x701F	spd_kp	速度的 kp	float	4	默认值 5	W/R
0x7020	spd_ki	速度的 ki	float	4	默认值 0.00 5	W/R
0x7021	spd _filt_gain	速度滤波值	float	4	默认值 0.1	W/R

读取示例：
以读取loc_kp为例：
读取指令为

大小	Bit28~bit24	bit23~8	bit7~0	Byte0~Byte7
	11	00FD	7F	1E 70 00 00 00 00 00 00 00 00
描述	类型 17 16 进制 0x11	主机 id 0xFD	目标电机 CAN_ID 7F	Byte0~1: index，对应 loc_kp

反馈指令为

大小	Bit28~bit24	bit23~8	bit7~0	Byte0~Byte7
	11	007F	FD	1E 70 00 00 00 00 F0 41
描述	类型 17 16 进制 0x11	bit15~8:目标电机 CAN_ID 7F	主机 id 0xFD	Byte0~1: index，对应 loc_kp Byte4~7:loc_kp 值为 30，右高字节，（32 位单精度）16 进制 IEEE-754 标准浮点数

4.2控制模式使用说明

4.2.1 程序样例

以下提供各种模式控制电机实例（以gd32f303为例）

下面为各种实例调用库，函数与宏定义

```
#define P_MIN -12.5f
#define P_MAX 12.5f
#define V_MIN -15.0f
#define V_MAX 15.0f
#define KP_MIN 0.0f
#define KP_MAX 500.0f
#define KD_MIN 0.0f
#define KD_MAX 5.0f
#define T_MIN -120.0f
#define T_MAX 120.0f
struct exCanIdInfo{
uint32_t id:8;
uint32_t data:16;
uint32_t mode:5;
uint32_t res:3;
};
can_receive_message_struct rxMsg;
can_trasnmit_message_struct txMsg={
.tx_sfid = 0,
.tx_efid = 0xff,
.tx_ft = CAN_FT_DATA,
.tx_ff = CAN_FF_EXTENDED,
.tx_dlen = 8,
};
#define txCanIdEx (((struct exCanIdInfo)&(txMsg.tx_efid)))
#define rxCanIdEx (((struct exCanIdInfo)&(rxMsg.rx_efid))) //将扩展帧id解析为自定义数据结构
int float_to_uint(float x, float x_min, float x_max, int bits){
float span = x_max - x_min;
float offset = x_min;
if(x > x_max) x=x_max;
else if(x < x_min) x= x_min;
return (int) ((x-offset)*((float)((1<<bits)-1))/span);
}
#define can_txd() can_message_transmit(CAN0, &txMsg)
#define can_rxd() can_message_receive(CAN0, CAN_FIFO1, &rxMsg)
```

下面列举常见的通信类型发送：

1、电机使能运行帧（通信类型3）

```
void motor_enable(uint8_t id, uint16_t master_id)
{
```

```
txCanIdEx.mode = 3;
txCanIdEx.id = id;
txCanIdEx.res = 0;
txCanIdEx.data = master_id;
txMsg.tx_dlen = 8;
txCanIdEx.data = 0;
can_txd();
}
```

2、运控模式电机控制指令（通信类型1）

```
void motor_controlmode(uint8_t id, float torque, float MechPosition, float speed, float kp, float kd)
{
txCanIdEx.mode = 1;
txCanIdEx.id = id;
txCanIdEx.res = 0;
txCanIdEx.data = float_to_uint(torque,T_MIN,T_MAX,16);
txMsg.tx_dlen = 8;
txMsg.tx_data[0]=float_to_uint(MechPosition,P_MIN,P_MAX,16)>>8;
txMsg.tx_data[1]=float_to_uint(MechPosition,P_MIN,P_MAX,16);
txMsg.tx_data[2]=float_to_uint(speed,V_MIN,V_MAX,16)>>8;
txMsg.tx_data[3]=float_to_uint(speed,V_MIN,V_MAX,16);
txMsg.tx_data[4]=float_to_uint(kp,KP_MIN,KP_MAX,16)>>8;
txMsg.tx_data[5]=float_to_uint(kp,KP_MIN,KP_MAX,16);
txMsg.tx_data[6]=float_to_uint(kd,KD_MIN,KD_MAX,16)>>8;
txMsg.tx_data[7]=float_to_uint(kd,KD_MIN,KD_MAX,16);
can_txd();
}
```

3、电机停止运行帧（通信类型4）

```
void motor_reset(uint8_t id, uint16_t master_id)
{
txCanIdEx.mode = 4;
txCanIdEx.id = id;
txCanIdEx.res = 0;
txCanIdEx.data = master_id;
txMsg.tx_dlen = 8;
for(uint8_t i=0;i<8;i++)
{
txMsg.tx_data[i]=0;
}
can_txd();
}
```

4、电机模式参数写入命令（通信类型18，运行模式切换）

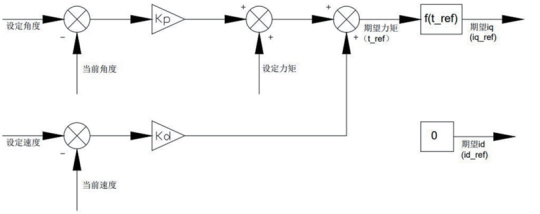
```
uint8_t runmode;
uint16_t index;
```

```
void motor_modechange(uint8_t id, uint16_t master_id)
{
txCanIdEx.mode = 0x12;
txCanIdEx.id = id;
txCanIdEx.res = 0;
txCanIdEx.data = master_id;
txMsg.tx_dlen = 8;
for(uint8_t i=0;i<8;i++)
{
txMsg.tx_data[i]=0;
}
memcpy(&txMsg.tx_data[0],&index,2);
memcpy(&txMsg.tx_data[4],&runmode, 1);
can_txd();
}
```

5、电机模式参数写入命令（通信类型18，控制参数写入）

```
uint16_t index;
float ref;
void motor_write(uint8_t id, uint16_t master_id)
{
txCanIdEx.mode = 0x12;
txCanIdEx.id = id;
txCanIdEx.res = 0;
txCanIdEx.data = master_id;
txMsg.tx_dlen = 8;
for(uint8_t i=0;i<8;i++)
{
txMsg.tx_data[i]=0;
}
memcpy(&txMsg.tx_data[0],&index,2);
memcpy(&txMsg.tx_data[4],&ref,4);
can_txd();
}
```

4.2.2 运控模式

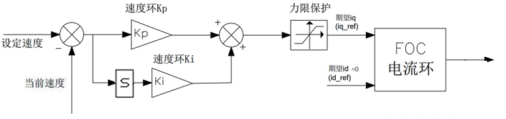


电机上电后默认处于运控模式；
发送电机使能运行帧（通信类型3）-->发送运控模式电机控制指令（通信类型1）-->收到电机反馈帧（通信类型2）

4.2.3 电流模式

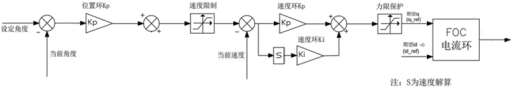
发送电机模式参数写入命令（通信类型18）设置runmode参数为3 ---> 发送电机使能运行帧（通信类型3）--> 发送电机模式参数写入命令（通信类型18）设置 iq_ref参数为预设电流指令

4.2.4 速度模式



发送电机模式参数写入命令（通信类型18）设置runmode参数为2 ---> 发送电机使能运行帧（通信类型3）--> 发送电机模式参数写入命令（通信类型18）设置 limit_cur参数为预设最大电流指令-->发送电机模式参数写入命令（通信类型18）设置 spd_ref参数为预设速度指令

4.2.5 位置模式



发送电机模式参数写入命令（通信类型18）设置runmode参数为1 --> 发送电机使能运行帧（通信类型3）--> 发送电机模式参数写入命令（通信类型18）设置 limit_spd参数为预设最大速度指令-->发送电机模式参数写入命令（通信类型18）设置 loc_ref参数为预设位置指令

4.2.6 停止运行

发送电机停止运行帧（通信类型4）

Precautions

1. Please use according to the working parameters specified in this article, otherwise it may cause serious damage to this product!
2. During joint operation, the control mode cannot be switched. To switch, a stop running command needs to be sent before switching.
3. Before use, please check if all components are intact. If any parts are missing or damaged, please contact technical support in a timely manner.
4. Please do not disassemble the motor randomly to avoid irreparable faults.
5. Ensure that there is no short circuit when connecting the motor and that the interface is correctly connected as required.

Legal Statement

Before using this product, users must carefully read this manual and operate the product according to its contents. If the user violates the content of this manual and uses this product, resulting in any property damage or personal injury accidents, our company shall not be held responsible. Due to the numerous components of this product, do not let children come into contact with it to avoid accidents. To extend the service life of the product, please do not use it in high temperature and high pressure environments. This manual has included various functional introductions and usage instructions as much as possible during printing. However, due to the continuous improvement of product functions and design changes, there may still be discrepancies with the products purchased by users.

This manual may deviate from the actual product in terms of color, appearance, etc. Please refer to the actual product for accuracy. This manual is published by RobStride Dynamics Co., Ltd (hereinafter referred to as Robstride Dynamics). RodStride may make necessary improvements and changes to the latest and inaccurate information in this manual, or make improvements to programs and/or equipment at any time. This type of change will upload an electronic version of the instruction manual to the company's official website. For details, please refer to the download center (www.robstride. com). All images are for reference only for functional specifications. Please refer to the actual product.

After sales policy

The after-sales service of this product is strictly in accordance with the Consumer Rights Protection Law of the People's Republic of China and the Product Quality Law of the People's Republic of China. The service content is as follows:

1. Warranty period and content

(1) Users who place orders for this product through online channels can enjoy a no reason return service within seven days from the day after signing for it. When returning goods, the user must present a valid purchase voucher and return the invoice. Users must ensure that the returned goods maintain their original quality and function, have an intact appearance, and that the trademarks and various markings of the goods and accessories are complete. If there are any gifts, they must be returned together. If the product is artificially damaged, dismantled, the packaging box is missing, or the spare parts are missing, no return will be processed. The logistics costs incurred during the return process are borne by the user (refer to the "After sales Service Fee Standards" for charging standards). If the user has not settled the logistics fees, they will be deducted from the refund amount based on the actual amount incurred. Within seven days from the date of receiving the returned goods, the paid payment shall be returned to the user. The refund method is the same as the payment method. The specific arrival date may be influenced by factors such as banks and payment institutions.

(2) The warranty period for this product is 1 year.

(3) Within 7 days from the day after the user signs for the item, if there is any non-human damage or performance malfunction, the customer must be confirmed by the Robstride Dynamics after-sales service center and return the item. When returning the item, the customer must present a valid purchase voucher and return the invoice. If there are any gifts, they should be returned together.

(4) Within 7 to 15 days from the day after the user signs for it, if there is any non-human damage or performance failure, the RodStride after-sales service center will conduct a replacement service for the user and replace the entire set of goods after testing and confirmation. After the exchange, the warranty period of the product itself is recalculated.

(5) Within 15 to 365 days from the day after the user signs for it, if it is confirmed by the Robstride Dynamics after-sales service center that the product itself has a quality fault, free repair services can be provided. The faulty products replaced belong to Robstride Dynamics Company. No faulty products, return as is. This product has undergone strict testing before leaving the factory. If there is any quality failure other than the product itself, we have the right to refuse the user's return or exchange request.

2. The following situations are not covered by warranty regulations:

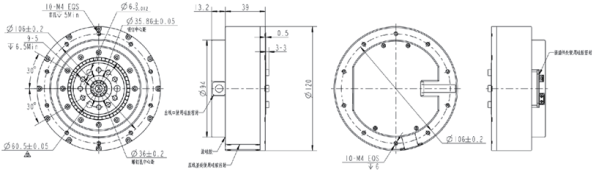
1. Exceeding the warranty period specified in the warranty terms.
2. Product damage caused by incorrect use without following the instructions.
3. Damage caused by improper operation, maintenance, installation, modification, testing, and other improper use.

4. Conventional mechanical losses and wear caused by non quality faults.
5. Damage caused by abnormal working conditions, including but not limited to falls, impacts, liquid immersion, severe impacts, etc.
6. Damage caused by natural disasters (such as floods, fires, lightning strikes, earthquakes, etc.) or uncontrollable forces.
7. Damage caused by exceeding peak torque usage.
8. Non genuine spiritual feet or inability to provide legal purchase vouchers.
9. Other malfunctions or damages caused by non product design, technology, manufacturing, quality, and other issues.

If the above situation occurs, users need to pay the fees themselves.

1. Motor specification parameters

1. Appearance and installation dimensions

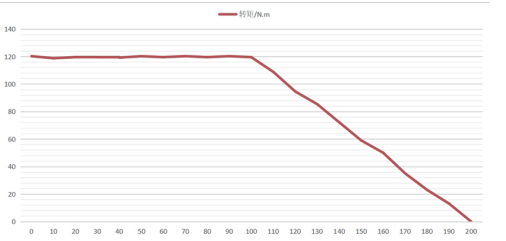


1.2 Standard Usage Status

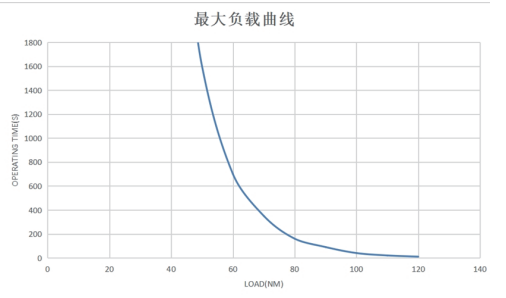
1.2.1 Rated voltage: 48 VDC
1.2.2 Voltage range: 24V -60 VDC
1.2.3 Rated load (CW): 40 N.m
1.2.4 Operating direction: CW/CCW viewed from the output shaft direction
1.2.5 Usage posture: The axis direction is horizontal or vertical
1.2.6 Standard operating temperature: 25 ± 5 ℃
1.2.7 Temperature range for use: -20~50 ℃
1.2.8 Standard operating humidity: 65%
1.2.9 Humidity range: 5-85%, no condensation
1.2.10 Storage temperature range: -30~70 ℃
1.2.11 Insulation level: Class B

1.3 Electrical characteristics

1.3.1 No load speed: 200 rpm ± 10%
1.3.2 No load current: 2Arms
1.3.3 Rated load: 40 N.m
1.3.4 Rated load speed: 167rpm ± 10%
1.3.5 Rated load phase current (peak): 27Apk ± 10%
1.3.6 Peak load: 120 N.m
1.3.7 Maximum load phase current (peak): 90apk ± 10%
1.3.8 Insulation resistance/stator winding: DC 500VAC, 100M Ohms
1.3.9 High voltage resistance/stator and cover: 600 VAC, 1s, 2mA
1.3.10 Motor back electromotive force: 16.9Vrms/krpm±10%
1.3.11 Torque constant: 2.1N.m/Arms
1.3.12 T-N curve



1.3.13 Maximum overload curve
Test conditions:
Environmental temperature: 25 ℃
Maximum temperature for winding resistance: 130 ℃ (this is the constraint temperature, actually 180 degrees)
Speed: 24rpm



test data

Load	Operating time(s)
120	10
110	20
100	40
90	90
80	160
70	350
60	700
50	1600
46	2500
40	rated

1.4 Mechanical characteristics
1.4.1 Weight: 1420g ± 20g
1.4.2 Number of Poles: 42 Poles
1.4.3 Number of phases: 3 phases
1.4.4 Drive mode: FOC
1.4.5 Reduction ratio: 9:1

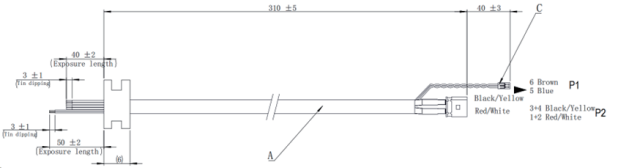
2 Drive Product Information

2.1 Driver Product Specifications

Product specifications	
Rated working voltage	48 VDC
Maximum allowable voltage	60 VDC
Rated working phase current	27Apk
Maximum allowable phase current	90 Apk
Standby power consumption	≤ 40 mA
CAN bus bit rate	1Mbps
Size	Φ 84 mm
Working environment temperature	-20 ℃ to 50 ℃
The maximum temperature allowed by the control board	80 ℃
Encoder resolution	14 bit (absolute value of a single lap)

2.2 Driver Interface Definition
2.2.1 Driver Interface Definition

2.2.2 Definition of drive harness sequence



Definition of drive harness sequence	
Blue	CAN_H
Brown	CAN_L
Black	GND
Red	VBAT+

2.2.3 Recommended brand and model of driver interface

Board end model	Brand manufacturer	Line end model	Brand manufacturer
XT30 APW -M	AMASS	XT30 UW -F	AMASS
GH1.25 -2PWT	Any	GH1.25 -T	Any

2.3 Main components and specifications

Number	project	Specifications	quantity
1	MCU chip	GD32F103RET6	1 PCS
2	Driver chip	DRV8353SRTAT	1 PCS
3	Magnetic encoder chip	AS5047P	1 PCS
4	Thermistor	LTS00 -104J395T19E010/ NCP18XH103F03RB	2 PCS
5	Power MOS	ISC030N12NM6	12 PCS

3 Instructions for using the Studio (please go to the download center on the official website of www.robstride.com)

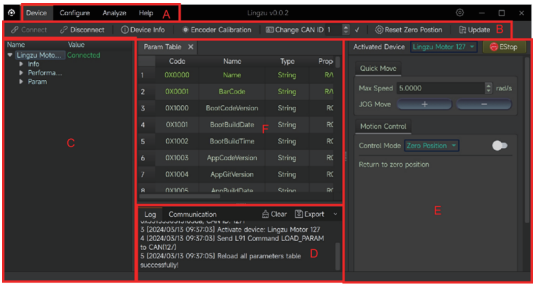
3.1 Hardware Configuration

The joint motor adopts CAN communication method, with two communication lines connected to the debugger through CAN to USB tool.

The debugger needs to install ch340 driver in advance and works in AT mode by default.

It should be noted that we developed the debugger based on a specific CAN to USB tool, so we need to use the recommended serial port tool for debugging. If you want to port to other debugger platforms, you can refer to Chapter 3 of the manual for development. It is recommended to use YourCee's USB-CAN module for the CAN to USB tool, with a frame header of 41 54 and a frame tail of 0D 0A corresponding to the serial protocol.

3.2 Studio interface and instructions



Mainly including:
A. Module selection
- Equipment module
- Configuration module
- Analysis module
- Help module

B. Sub module selection
Equipment modules include
- Connecting or disconnecting motor equipment
- Motor equipment information
- Motor encoder calibration
- Modify the motor CAN ID
- Set the mechanical zero position of the motor
- Motor program upgrade

The configuration module includes:
- Parameter table, can view and modify motor parameters
- Upload parameters, you can upload the parameters in the motor to the parameter table
- Download parameters, you can download the data from the parameter table to the motor
- Export parameters and download the data from the parameter table to the local location
- Restore to factory settings by resetting the data in the parameter table to factory settings
- Clear warning to clear motor errors, such as excessive temperature

The analysis module includes:
- Oscilloscope, which can view the curve of parameter changes over time
- Frequency, can adjust the frequency of viewing data
- Channel, can be configured to view data
- Start and stop drawing
- Output waveform data to local

The help module includes:
- Instructions for use, which can be opened
- Regarding, you can view software information

C. Motor information inquiry
- Device information
- Parameter Table Information

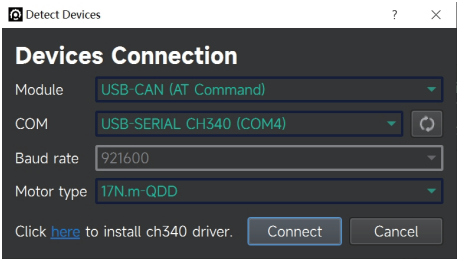
D. Data bar
- log information
- Communication information

E. Run debugging area
- Select device
- Convenient operation area, which can quickly control the forward and reverse rotation of the motor
- Motion control area, which can control the motor to operate in various modes

F. Sub module display area

3.3 Motor settings

3.3.1 Motor connection settings

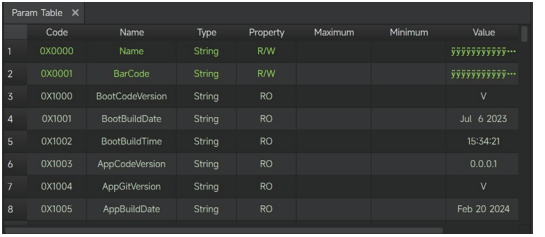


Connect the CAN to USB tool (install ch340 driver, default working in AT mode), click on the connection submodule in the device module,select the corresponding serial port connection and motor type, and click connect.

3.3.2 Basic Settings

(1) Modify the motor ID number.
(2) Motor magnetic encoder calibration, reinstallation of the motor board and motor, or replacement of the motor's three-phase line sequence connection, etc., requires re magnetic encoder calibration.
(3) Set the zero position (power loss) and set the current position to 0.
(4) Motor program upgrade. When there is an update to the motor program, click the upgrade button and select the upgrade file to proceed with the upgrade.

3.3.3 Parameter Table



	Code	Name	Type	Property	Maximum	Minimum	Value
1	0X0000	Name	String	R/W			ŷŷŷŷŷŷŷŷŷŷ···
2	0X0001	BarCode	String	R/W			ŷŷŷŷŷŷŷŷŷŷ···
3	0X1000	BootCodeVersion	String	RO			V
4	0X1001	BootBuildDate	String	RO			Jul 6 2023
5	0X1002	BootBuildTime	String	RO			15:34:21
6	0X1003	AppCodeVersion	String	RO			0.0.0.1
7	0X1004	AppGitVersion	String	RO			V
8	0X1005	AppBuildDate	String	RO			Feb 20 2024

After successfully connecting the motor, click on the parameter table module in the configuration module. The log will display all parameters loaded successfully, indicating that the motor related parameters have been successfully read (note: the parameter table needs to be configured when the motor is in standby mode, and cannot be refreshed if the motor is in running mode). The interface will display the motor related parameters, and the blue parameters are the stored parameters inside the motor, You can make modifications in the current alue column after the corresponding parameters. Clicking on the download parameter will download the parameters from the debugger to the motor. Clicking on the upload parameter will upload the parameters from the motor to the debugger. The green parameter of the motor is the observation parameter, and the collected parameter can be observed in real time.

Note: The torque limit, protection temperature, and over temperature time of the motor should not be changed arbitrarily. Our company will not bear any legal responsibility for any harm caused to the human body or irreversible damage to joints due to improper operation of this product.

Parameter table							
Function code	Name	Parameter type	Properties	Maximum value	Minimum value	Current value (for reference)	Remarks
0X0000	Name	String	Read/Write			ÿÿÿÿÿÿÿÿÿÿÿÿ	
0X0001	BarCode	String	Read/Write			ÿÿÿÿÿÿÿÿÿÿÿÿ	
0X1000	BootCodeVersion	String	Read Only			0.1.5	
0X1001	BootBuildDate	String	Read Only			Mar 16 2022	
0X1002	BootBuildTime	String	Read Only			20:22:09	
0X1003	AppCodeVersion	String	Read Only			0.0.0.1	Motor program version number
0X1004	AppGitVersion	String	Read Only			7b844b0fM	
0X1005	AppBuildDate	String	Read Only			Apr 14 2022	
0X1006	AppBuildTime	String	Read Only			20:30:22	
0X1007	AppCodeName	String	Read Only			Lingzu_motor	
0X2000	EchoPara1	Uint16	Configuration	74	5	5	
0X2001	EchoPara2		Configuration	74	5	5	
0X2002	EchoPara3	Uint16	Configuration	74	5	5	
0X2003	EchoPara4	Uint16	Configuration	74	5	5	
0X2004	EchoFreHz	Uint32	Read/Write	10000	1	500	

0X2005	MechOffset	Float	Read/Write	7	-7	4.619583	Low-speed end position offset value
0X2006	chasu_offset	Float	Read/Write	6.28	0	4.52	Differential offset value
0X2007	Elecoffset	Float	Read/Write	6.28	0	3.27	Electrical Angle Offset Value
0X2008	I_ FW_ MAX	Float	Read/Write	33	0	0	Weak magnetic current value, default to 0
0X2009	CAN_ ID	Uint8	Setting	127	0	1	This node ID
0X200a	CAN_ MASTER	Uint8	Setting	127	0	0	Can host ID
0X200b	CAN_ TIMEOUT	Uint32	Read/Write	100000	0	0	Can timeout threshold, default to 0
0X200c	MotorOverTemp	Int16	Read/Write	1500	0	800	Motor protection temper ature value, temp (degrees) * 10
0X200d	OverTempTime	Uint32	Read/Write	1000000	1000	20000	Over temperature time
0X200e	GearRatio	Float	Read/Write	64	1	9	Transmission ratio
0X200f	Kt_Nm/Amp	Float	Read/Write	1	1	0	0

0X2010	Tq_ CaliType	Uint8	Read/Write	1	0	1	Torque calibration method setting
0X2011	Cur_ Fit_ Gain	Float	Read/Write	1	0	0.6	Current filtering parameters
0X2012	Cur_ Kp	Float	Read/Write	200	0	0.05	Current kp
0X2013	Cur_ Ki	Float	Read/Write	200	0	0.05	Current ki
0X2014	Spd_ Kp	Float	Read/Write	200	0	2	Speed kp
0X2015	Spd_ Ki	Float	Read/Write	200	0	0.021	Speed ki
0X2016	Loc_ Kp	Float	Read/Write	200	0	30	Location kp
0X2017	Spd_ Fit_ Gain	Float	Read/Write	1	0	0.1	Speed filtering parameters
0X2018	Limit_ Spd	Float	Read/Write	200	0	2	Position mode speed limit
0X2019	Limit_ Cur	Float	Read/Write	90	0	90	Position and speed modes Current limitation
0X3000	TimeUse0	Uint16	Read Only			5	
0X3001	TimeUse1	Uint16	Read Only			0	
0X3002	TimeUse2	Uint16	Read Only			10	
0X3003	TimeUse3	Uint16	Read Only			0	
0X3004	EncoderRaw	Int16	Read Only			11396	Magnetic encoder sampling value

0X3005	McuTemps	Int16	Read Only			337	Internal temperature of MCU, * 10
0X3006	MotorTemps	Int16	Read Only			333	Motor NTC temperature, * 10
0X3007	VBus (mv)	Uint16	Read Only			24195	Bus voltage
0X3008	Adc1Offset	Int32	Read Only			2084	ADC sampling channel 1 zero current bias
0X3009	Adc2Offset	Int32	Read Only			2084	ADC sampling channel 2 zero current bias
0X300a	Adc1Raw	Uint16	Read Only			1232	ADC sampling value 1
0X300b	Adc2Raw	Uint16	Read Only			1212	ADC sampling value 2
0X300c	VBUS	Float	Read Only			24.195	Bus voltage V
0X300d	CmdId	Float	Read Only			0	ID Ring Instruction, A
0X300e	CmdIq	Float	Read Only			0	Iq ring instruction, A
0X300f	Cmdlocref	Float	Read Only			0	Position loop command, rad
0X3010	Cmdspdref	Float	Read Only			0	Speed loop command, rad/s

0X3011	CmdTorque	Float	Read Only			0	Torque command, nm
0X3012	CmdPos	Float	Read Only			0	Mit protocol angle instruction
0X3013	CmdVel	Float	Read Only			0	Mit protocol speed instruction
0X3014	Rotation	Int16	Read Only			1	Number of laps
0X3015	ModPos	Float	Read Only			4.363409	Motor uncounted mechanical angle, rad
0X3016	MechPos	Float	Read Only			0.777679	Mechanical angle of load end coil, rad
0X3017	MechVel	Float	Read Only			0.036618	Load end speed, rad/s
0X3018	ElecPos	Float	Read Only			4.714761	Electrical perspective
0X3019	Ia	Float	Read Only			0	U line current, A
0X301a	Ib	Float	Read Only			0	V-line current, A
0X301b	IC	Float	Read Only			0	W-line current, A
0X301c	Tick	Uint32	Read Only			31600	

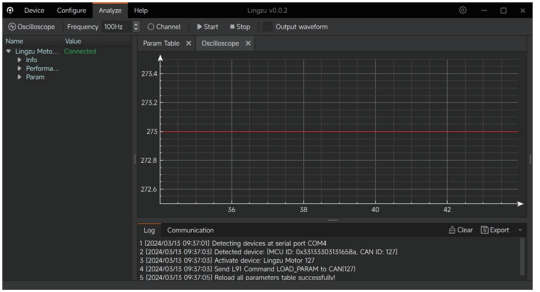
0X301d	PhaseOrder	Uint8	Read Only			0	Calibration direction markings
0X301e	IQF	Float	Read Only			0	
0X301f	BoardTemps	Int16	Read Only			359	On board temperature, * 10
0X3020	Iq	Float	Read Only			0	Iq original value, A
0X3021	ID	Float	Read Only			0	ID original value, A
0X3022	FaultSta	Uint32	Read Only			0	Fault state value
0X3023	WarnSta	Uint32	Read Only			0	Warning status value
0X3024	DRV_ Fault	Uint16	Read Only			0	Driver chip fault value
0X3025	DRV_ Temp	Int16	Read Only			48	Driver chip temperature value, degrees
0X3026	Uq	Float	Read Only			0	Q-axis voltage
0X3027	Ud	Float	Read Only			0	D-axis voltage
0X3028	DTC_ U	Float	Read Only			0	U-phase output duty cycle
0X3029	DTC_ V	Float	Read Only			0	V-phase output duty cycle

0X302a	DTC_ W	Float	Read Only			0	W-phase output duty cycle
0X302b	V_ Bus	Float	Read Only			24.195	Vbus in closed - loop
0X302c	V_ Ref	Float	Read Only			0	Closed loop vq, vd synthesis voltage
0X302d	Torque_ FDB	Float	Read Only			0	Torque feedback value, nm
0X302e	Rated_ I	Float	Read Only			8	Motor rated current
0X302f	Limit_ I	Float	Read Only			27	Motor limit maximum current

3.3.4 Oscilloscope

This interface supports viewing the graph generated by observing real-time data, including motor Id/Iq current, temperature, real-time output speed, rotor (encoder) position, output position, etc.

Click on the oscilloscope module in the analysis module, select the appropriate parameters in the channel (parameter meanings can refer to 3.3.3), set the output frequency, and click start drawing to observe the data graph. Stop drawing to stop observing the graph.



Example of communication box command:
41 54 90 07 e8 0c 08 05 70 00 00 01 00 00 00 0d 0a
The meaning is as follows

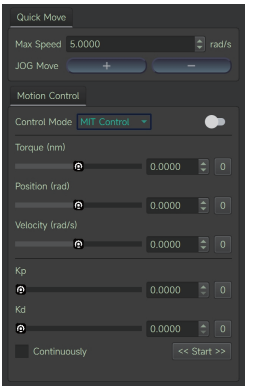
41 54	90 07 e8 0c	08	05 70 00 00 01 00 00 00 00	0d 0a
Frame header	Extended frame	Number of data bits	Data frame	End of frame

Translating the extended CAN ID to the real CAN ID requires the following conversion:
90 07 e8 0c is converted to binary as 1001 0000 0000 0111 1110 1000 0000 1100. If the 100 on the right is removed, it becomes 1 0010 0000 1111 1101 0000 0001. Convert it to hexadecimal and it becomes 12 00 FD 01. Refer to the communication protocol description and the meaning is as follows:

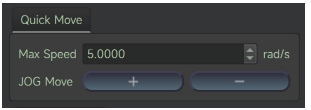
12 (hexadecimal)	00	FD	01
Communication type 18 (base 10)	Meaningless	Host ID	Motor CAN ID

3.3.5 CAN communication fault protection
When CAN_ When the TIMEOUT value is 0, this function is not enabled
When CAN_ When the TIMEOUT value is non-0, when the motor does not receive a can command within a certain period of time, the motor enters reset mode, and 12000 is 1 second

3.4 Control demonstration



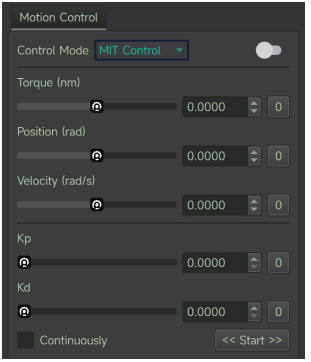
Jog Run:
Set the maximum speed, click run, then click JOG to run the motor in both forward and reverse directions



Control mode switching:
The motor control mode can be switched on the sports mode interface

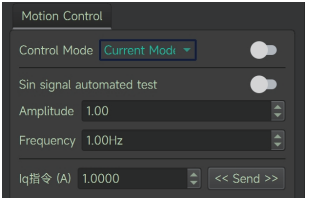
3.4.1 Zero point mode

Click the switch button on the right side, and the motor will slowly return to the mechanical zero position
3.4.2 Operation and control mode



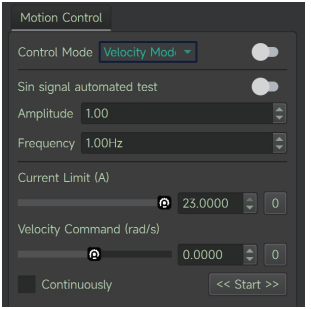
Click the switch button on the right side, then set five parameter values, click start or continue sending, the motor will return to the feedback frame and run according to the target command; Click the switch button on the right again, and the motor will stop

3.4.3Current mode



Manually switch the current mode, click the right switch button, and then set the Iq current command value. Start or continue sending, and the motor will follow the current command. Click the right switch button again, and the motor will stop.
Click the switch button on the right side of the control mode, input the amplitude and frequency of the sine automatic test, and then click the switch button on the right side of the sine automatic test. The iq (A) of the motor will run according to the set amplitude and frequency

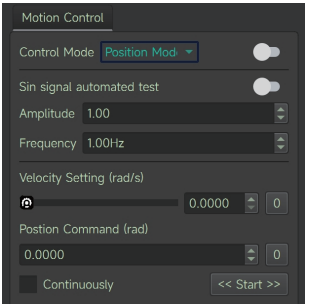
3.4.4 Speed mode



Manually switch speed mode, click the right switch button, then set the speed command value, start or continuously send, the motor will follow the speed command to run, click the right switch button again, the motor will stop.

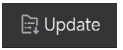
Click the switch button on the right side of the control mode, input the amplitude and frequency of the sine automatic test, and then click the switch button on the right side of the sine automatic test. The motor speed (rad/s) will run according to the set amplitude and frequency.

3.4.5 Position mode



Manually switch position mode, click the right switch button, then set the position command value (rad), start or continuously send, the motor will follow the target position command to run, click the right switch button again, the motor will stop. The maximum speed followed by the position can be modified by setting the speed.
Click the switch button on the right side of the control mode, input the amplitude and frequency of the sine automatic test, and then click the switch button on the right side of the sine automatic test. The position (rad) of the motor will run according to the set amplitude and frequency.

3.5 Firmware updates



The first step is to click on the upgrade of the device module and select the bin file to be burned; The second step is to confirm the upgrade and start updating the firmware of the motor. After the progress is completed, the motor update is completed and automatically restarts.

4. Driver Communication Protocol and Instructions for Use

The motor communication uses CAN 2.0 communication interface, with a baud rate of 1Mbps and an extended frame format, as shown below:

Data Domain		29 digit ID		8Byte data area
size	Bit28~bit24	Bit23 -8	Bit7~0	Byte0~Byte7
describe	Communication type	Data Area 2	Target address	Data Area 1

The control modes supported by the motor include:
Operation control mode: Given 5 parameters for motor operation control;
Current mode: Given the specified Iq current of the motor;
Speed mode: Given the specified operating speed of the motor;
Position mode: Given the specified position of the motor, the motor will run to that specified position;

4.1 Description of Communication Protocol Types
4.1.1 Obtain device ID (communication type 0); Obtain the device ID and 64 bit MCU unique identifier

Data Domain	29 digit ID			8Byte data area
size	Bit28~bit24	Bit23 -8	Bit7~0	Byte0~Byte7
describe	0	Bit15~8: Used to identify the host CAN_ ID	Target motor CAN_ ID	0

Response frame:

Data Domain	29 digit ID			8Byte data area
size	Bit28~bit24	Bit23 -8	Bit7~0	Byte0~Byte7
describe	0	Target motor CAN_ ID	0XFE	64 bit MCU unique identifier

4.1.2 Operation Control Mode Motor Control Command (Communication Type 1) is used to send control commands to the motor

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23 -8	Bit7~0	Byte0~Byte7
Description	1	Byte2: Moment (0~65535) Corresponding (-120 Nm~ 120 Nm)	Target motor CAN_ ID	Byte0~1: Target angle [0~65535] corresponds to (-4 π~4 π) Byte2~3: Target angular velocity [0~65535] corresponds to (-15 rad/s~ 15 rad/s) Byte4~5: Kp [0~65535] corresponds to (0.0~5000.0) Byte6~7: Kd [0~65535] corresponds to (0.0~100.0) After converting the above data, high bytes come first and low bytes come last

Response frame: Response motor feedback frame (see communication type 2)
4.1.3 Motor feedback data (communication type 2) is used to provide feedback on the motor operation status to the host

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23 -8	Bit7~0	Byte0~Byte7
Description	2	Bit8~Bit15: Current motor CAN ID Bit21~16: Fault information (0 none 1 available) Bit21: Uncalibrated Bit20: HALL encoding fault Bit19: Magnetic encoding fault Bit18: Over temperature Bit17: Overcurrent Bit16: Undervoltage fault Bit22~23: Mode status 0: Reset mode [reset] 1: Cali mode [calibration] 2: Motor mode [Run]	Host CAN_ ID	Byte0~1: Current angle [0~65535] corresponds to (-4 π~4 π) Byte2~3: Current angular velocity [0~65535] corresponds to (-15 rad/s~ 15 rad/s) Byte4~5: Current torque [0~65535] corresponds to (-120 Nm~ 120 Nm) Byte6~7: Current temperature: Temperature (Celsius) * 10 The above data has high bytes first and low bytes last

4.1.4 Motor Enable Operation (Communication Type 3)

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23 -8	Bit7~0	Byte0~Byte7
Description	3	Bit15~8: Used to identify the main CAN_ ID	Target motor CAN_ ID	

Response frame: Response motor feedback frame (see communication type 2)

4.1.5 Motor stop running (communication type 4)

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23 -8	Bit7~0	Byte0~Byte7
Description	4	Bit15~8: Used to identify the main CAN_ ID	Target motor CAN_ ID	During normal operation, the data area needs to be cleared to 0; When Byte [0]=1: Clear fault;

Response frame: Response motor feedback frame (see communication type 2)

4.1.6 Setting the mechanical zero position of the motor (communication type 6) will set the current motor position to tAhe mechanical zero position (power loss)

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23 -8	Bit7~0	Byte0~Byte7
Description	6	Bit15~8: Used to identify the main CAN_ ID	Target motor CAN_ ID	Byte [0]=1

Response frame: Response motor feedback frame (see communication type 2)
4.1.7 Setting Motor CAN_ ID (Communication Type 7) Change the current motor CAN_ ID, effective immediately.

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23 -8	Bit7~0	Byte0~Byte7
Description	7	Bit15~8: Used to identify the main CAN_ ID Bit16~23: Pre set CAN_ ID	Target motor CAN_ ID	

Response frame: Response motor broadcast frame (see communication type 0)
4.1.8 Single parameter reading (communication type 17)

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23 -8	Bit7~0	Byte0~Byte7
Description	17	Bit15~8: Used to identify the main CAN_ ID Bit23 -16:00 indicates successful read 01 indicates read failure	Target motor CAN_ ID	Byte0~1: index, parameter list detailed in 4.1.11 Byte2~3:00 Byte4~7:00 The above data has low bytes first and high bytes last

Response frame:

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23 -8	Bit7~0	Byte0~Byte7
Description	17	ID	Host CAN_ ID	Byte0~1: index, parameter list detailed in 4.1.11 Byte2~3:00 Byte4~7: Parameter data, 1 byte of data in Byte4 The above data has low bytes first and high bytes last

4.1.9 Single parameter writing (communication type 18) (power loss)

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23 -8	Bit7~0	Byte0~Byte7
Description	18	Bit15~8: Used to identify the main CAN_ ID	Target motor CAN_ ID	Byte0~1: index, parameter list detailed in 4.1.11 Byte2~3:00 Byte4~7: Parameter data The above data has low bytes first and high bytes last

Response frame: Response motor feedback frame (see communication type 2)

4.1.10 Fault Feedback Frame (Communication Type 21)

Data Domain	29 digit ID			8Byte data area
Size	Bit28~bit24	Bit23 -8	Bit7~0	Byte0~Byte7
Description	21	Bit15~8: Used to identify the main CAN_ ID	Motor CAN_ ID	Byte0~3: fault value (non 0: faulty, 0: normal) Bit16: A -phase current sampling overcurrent Bit15~bit8: Overload fault Bit7: Encoder not calibrated Bit5: C -phase current sampling overcurrent Bit4: B -phase current sampling overcurrent Bit3: Overvoltage fault Bit2: Undervoltage fault Bit1: Driver chip failure Bit0: Motor over temperature fault, default to 80 degrees Byte4~7: warning value Bit0: Motor over temperature warning, default to 75 degrees

4.1.11 Read and write a single parameter list

Parameter index	Parameter Name	Description	Type	Byte count	Unit/Description	R/W read and write permissio ns
0X7005	Run_ Mode	0: Operation control mode 1: Position mode 2: Speed mode 3: Current mode	Uint8	1		W/R
0X7006	Iq_ Ref	Current mode Iq command	Float	4	-90-90A	W/R
0X700A	Spd_ Ref	Speed mode speed command	Float	4	-15-15 rad/s	W/R
0X700B	Limit_ Torque	Torque limitation	Float	4	0-120 Nm	W/R
0X7010	Cur_ Kp	Kp of current	Float	4	Default value 0.05	W/R
0X7011	Cur_ Ki	Ki of current	Float	4	Default value 0.05	W/R
0X7014	Cur_ Fit_ Gain	Current filtering coefficient filter_ Gain	Float	4	0~1.0, default value 0. 06	W/R
0X7016	Ref	Position mode angle command	Float	4	Rad	W/R
0X7017	Limit_ Spd	Position mode speed limit	Float	4	0-15 rad/s	W/R
0X7018	Limit_ Cur	Speed position mode current limit	Float	4	0-90A	W/R
0x7019	mechPos	Load end-ring mechanical angle	float	4	rad	R
0x701A	iqf	The iq filter values	float	4	-90~90A	R
0x701B	mechVel	Load end speed	float	4	-15~15rad/s	R
0x701C	VBUS	busbar voltage	float	4	V	R
0x701E	loc_kp	positional kp	float	4	Default value 30	W/R
0x701F	spd_kp	Speed kp	float	4	Default value 5	W/R
0x7020	spd_ki	Speedki	float	4	Default value 0.005	W/R
0x7021	spd_filt_gain	Speed filter value	float	4	Default value 0.1	W/R

Reading example:
To read loc_ For example, kp:
The read instruction is

Size	Bit28~bit24	Bit23 -8	Bit7~0	Byte0~Byte7
	11	00FD	7F	1E 70 00 00 00 00 00 00 00
Description	Type 17 Hexadecimal 0x11	Host ID 0xFD	Target motor CAN_ ID 7F	Byte0~1: index, corresponding to loc_ Kp

The feedback command is

Size	Bit28~bit24	Bit23 -8	Bit7~0	Byte0~Byte7
	11	007F	FD	1E 70 00 00 00 00 00 F0 41
Description	Type 17 Hexadecimal 0x11	Bit 15~8: Target motor CAN_ ID 7F	Host ID 0xFD	Byte0~1: index, corresponding to loc_ Kp Byte4~7: loc_ Kp value is 30, right high byte, (32 -bit single precision) hexadecimal IEEE -754 standard floating -point number

4.2 Instructions for using control mode
4.2.1 Program Examples
The following provides examples of various mode control motors (taking gd32f303 as an example)
Below are various instance call libraries, function and macro definitions

```
#Define P_ MIN -12.5f
#Define P_ MAX 12.5f
#Define V_ MIN -15.0f
#Define V_ MAX 15.0f
#Define KP_ MIN 0.0f
#Define KP_ MAX 500.0f
#Define KD_ MIN 0.0f
#Define KD_ MAX 5.0f
#Define T_ MIN -120.0f
#Define T_ MAX 120.0f
Struct exCanIdInfo{
Uint32_ T id: 8;
Uint32_ T data: 16;
Uint32_ T mode: 5;
Uint32_ T res: 3;
};
Can_ Receive_ Message_ Struct rxMsg;
Can_ Trasnmit_ Message_ Struct txMsg={
. tx_ Sfid=0,
. tx_ Efid=0xff,
. tx_ Ft=CAN_ FT_ DATA,
. tx_ Ff=CAN_ FF_ Extended,
. tx_ Dlen=8,
};
#Define txCanIdEx (((struct exCanIdInfo)&(txMsg. tx_efid)))
#Define rxCanIdEx (((struct exCanIdInfo)&(rxMsg. rx_efid)))//Parse the extended frame ID to a custom data structure
Int float_ To_ Uint (float x, float x_min, float x_max, int bits){
Float span=x_ Max - x_ Min;
Float offset=x_ Min;
If (x>x_max) x=x_ Max;
Else if (x<x_min) x=x_ Min;
Return (int) (x-offset) * (float) ((1<<bits) -1))/span);
}
#Define can_ Txd() can_ Message_ Transmit (CAN0,&txMsg)
#Define can_ Rxd() can_ Message_ Receive (CAN0, CAN-FIFO1,&rxMsg)
```

Here are some common types of communication to send:

1. Motor Enable Run Frame (Communication Type 3)

```
Void motor_ Enable (uint8_t id, uint16_t master_id)
{
TxCanIdEx. mode=3;
TxCanIdEx. id=id;
TxCanIdEx. res=0;
TxCanIdEx. data=master_ ID;
TxMsg. tx_ Dlen=8;
TxCanIdEx. data=0;
Can_ Txd();
}
```

2. Operation control mode motor control command (communication type 1)

```
Void motor_ Control mode (uint8_t id, float torque, float MechPosition, float speed, float kp, float kd)
```

```
{
TxCanIdEx. mode=1;
TxCanIdEx. id=id;
TxCanIdEx. res=0;
TxCanIdEx. data=float_ To_ Uint (torque, T.MIN, T.MAX, 16);
TxMsg. tx_ Dlen=8;
TxMsg. tx_ Data [0]=float_ To_ Uint (MechPosition, P_MIN, P_MAX, 16)>>8;
TxMsg. tx_ Data [1]=float_ To_ Uint (MechPosition, P_MIN, P_MAX, 16);
TxMsg. tx_ Data [2]=float_ To_ Uint (speed, V-MIN, V-MAX, 16)>>8;
TxMsg. tx_ Data [3]=float_ To_ Uint (speed, V-MIN, V-MAX, 16);
TxMsg. tx_ Data [4]=float_ To_ Uint (kp, KP-MIN, KP-MAX, 16)>>8;
TxMsg. tx_ Data [5]=float_ To_ Uint (kp, KP-MIN, KP-MAX, 16);
TxMsg. tx_ Data [6]=float_ To_ Uint (kd, KD-MIN, KD-MAX, 16)>>8;
TxMsg. tx_ Data [7]=float_ To_ Uint (kd, KD-MIN, KD-MAX, 16);
Can_ Txd();
}
```

3. Motor stop running frame (communication type 4)

```
Void motor_ Reset (uint8_t id, uint16_t master_id)
{
TxCanIdEx. mode=4;
TxCanIdEx. id=id;
TxCanIdEx. res=0;
TxCanIdEx. data=master_ ID;
TxMsg. tx_ Dlen=8;
For (uint8_t i=0; i<8; i++)
{
TxMsg. tx_ Data [i]=0;
}
Can_ Txd();
}
```

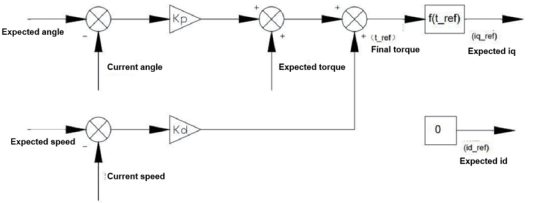
4. Motor mode parameter writing command (communication type 18, operation mode switching)

```
Uint8_ T runmode;
Uint16_ T index;
Void motor_ Modechange (uint8_t id, uint16_t master_id)
{
TxCanIdEx. mode=0x12;
TxCanIdEx. id=id;
TxCanIdEx. res=0;
TxCanIdEx. data=master_ ID;
TxMsg. tx_ Dlen=8;
For (uint8_t i=0; i<8; i++)
{
TxMsg. tx_ Data [i]=0;
}
Memcpy (&txMsg. tx_data [0],&index, 2);
Memcpy (&txMsg. tx_data [4],&runmode, 1);
Can_ Txd();
}
```

5. Motor mode parameter writing command (communication type 18, control parameter writing)

```
Uint16_ T index;
Float ref;
Void motor_ Write (uint8_t id, uint16_t master_id)
{
TxCanIdEx. mode=0x12;
TxCanIdEx. id=id;
TxCanIdEx. res=0;
TxCanIdEx. data=master_ ID;
TxMsg. tx_ Dlen=8;
For (uint8_t i=0; i<8; i++)
{
TxMsg. tx_ Data [i]=0;
}
Memcpy (&txMsg. tx_data [0],&index, 2);
Memcpy (&txMsg. tx_data [4],&ref, 4);
Can_ Txd();
}
```

4.2.2 Operation and control mode

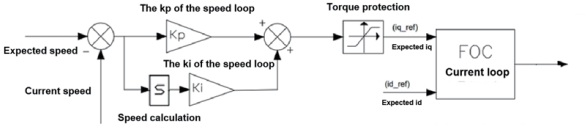


After the motor is powered on, it defaults to operation control mode;
Send motor enable running frame (communication type 3) -->Send operation control mode motor control command (communication type 1) -->Receive motor feedback frame (communication type 2)

4.2.3 Current mode

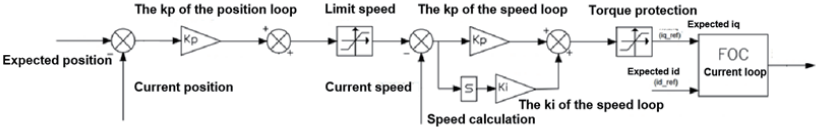
Send motor mode parameter writing command (communication type 18) Set runmode parameter to 3-->Send motor enable running frame (communication type 3) -->Send motor mode parameter writing command (communication type 18) Set iq_ The ref parameter is a preset current command

4.2.4 Speed mode



Send motor mode parameter writing command (communication type 18) Set runmode parameter to 2-->Send motor enable running frame (communication type 3) -->Send motor mode parameter writing command (communication type 18) Set limit_ The cur parameter is the preset maximum current command -->Send motor mode parameter writing command (communication type 18) Set SPD_ The ref parameter is a preset speed command

4.2.5 Position mode



Send motor mode parameter write command (communication type 18) to set the runmode parameter to 1 --> Send motor enable run frame (communication type 3) --> Send motor mode parameter write command (communication type 18) to set the limit_ spd parameter to the preset maximum speed instruction --> Send motor mode parameter write command (communication type 18) to set the loc_ref parameter to the preset position instruction.

4.2.6 Stopping operation
Send motor stop running frame (communication type 4)