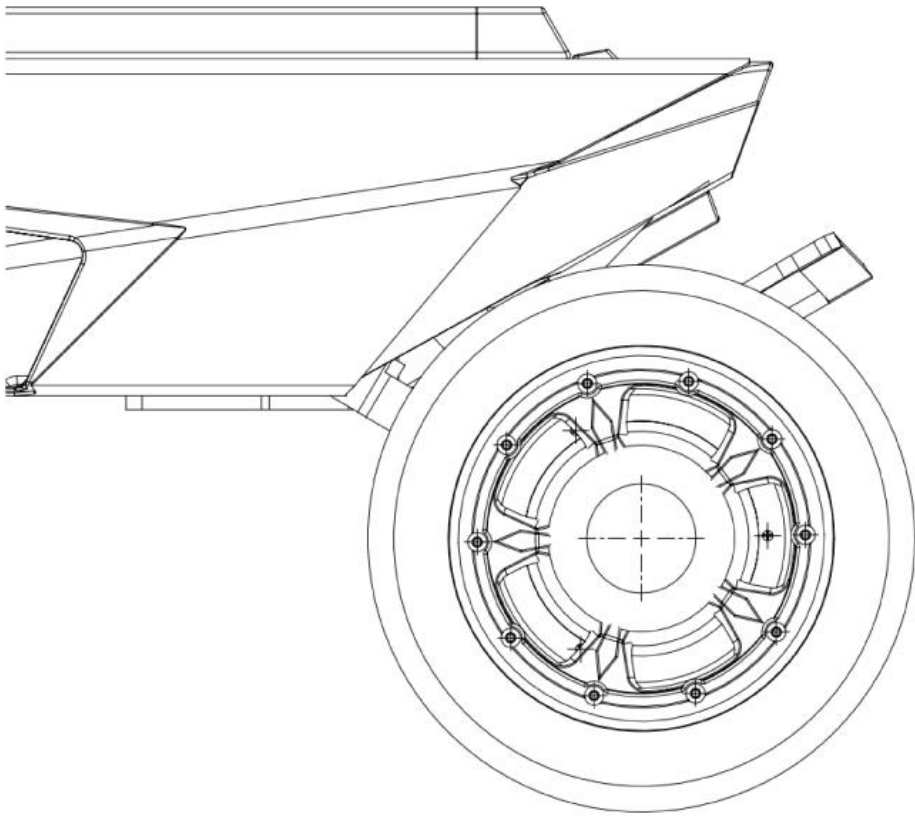




SCOUT MINI

用户手册

SCOUT MINI User Manual



| 编号 | 版本 | 日期 | 修改人 | 审核人 | 备注 |
|----|---------|------------|-----|-----|---|
| 1 | V.2.0.0 | 2023/03/12 | 吴忠义 | | 文件创建, 更新图片 |
| 2 | V.2.0.1 | 2023/04/19 | 吴忠义 | | 1、同步 GitHub 指令 2、更新固件下载地址 3、新版遥控器更新 |
| 2 | V.2.0.2 | 2023/6/2 | 罗俊 | | 1、遥控器 SWC、SWD 映射改变 2、增加速度模式说明 |

本章包含重要的安全信息，在机器人第一次通电前，任何个人或者机构在使用设备之前必须阅读并理解这些信息。有任何相关使用的疑问都可以联系我们 support@agilex.ai 必须遵守并执行本手册其他章节中的所有组装说明和指南，这一点非常重要。应特别注意与警告标志相关的文本。

重要安全信息

本手册中的信息不包含设计、安装和操作一个完整的机器人应用，也不包含所有可能对这一完整的系统的安全造成影响的周边设备。该完整的系统的设计和使用需符合该机器人安装所在国的标准和规范中确立的安全要求。

SCOUTMINI 的集成商和终端客户有责任确保遵循相关国家的切实可行的法律法规，确保完整的机器人应用实例中不存在任何重大危险。这包括但不限于以下内容：

有效性和责任

- 对完整的机器人系统做一个风险评估。
- 将风险评估定义的其他机械的附加安全设备连接在一起。
- 确认整个机器人系统的外围设备包括软件和硬件系统的设计和安装准确无误。
- 本机器人不具备一个完整的自主移动机器人具备的自动防撞、防跌落、生物接近预警等相关安全功能但不局限于上述描述，相关功能需要集成商和终端客户遵循相关规定和切实可行的法律法规进行安全评估，确保开发完成的机器人在实际应用中不存在任何重大危险和安全隐患。
- 收集技术文件中的所有文档：包括风险评估和本手册。
- 在操作和使用设备之前已经知晓可能存在的安全风险。

环境

- 首次使用，请先仔细阅读本手册，了解基本操作内容与操作规范。
- 遥控操作，选择相对空旷区域使用，车上本身是不带任何自动避障传感器。
- 在-10℃~45℃的环境温度中使用。
- 如果车辆非单独定制 IP 防护等级，车辆防水、防尘能力为 IP22。

检查

- 确保各设备的电量充足。
- 确保车辆无明显异常。
- 检查遥控器的电池电量是否充足

操作

- 保证使用时周围区域相对空旷。
- 在视距内遥控控制。
- SCOUTMINI 最大的载重为 10KG，在使用时，确保有效载荷不超过 10KG。
- SCOUTMINI 安装外部扩展时，确认扩展的质心位置，确保在旋转中心。

- 当设备低电量报警时请及时充电。
- 当设备出现异常时，请立即停止使用，避免造成二次伤害。
- 当设备出现异常时，请联系相关技术人员，请勿擅自处理。
- 请根据设备的 IP 防护等级在满足防护等级要求的环境中使用。
- 请勿直接推车。
- 充电时，确保周围环境温度大于 0℃。

保养

- 轮胎磨损严重，请及时更换。
- 如果长时间不使用电池，需要按照 2 到 3 个月对电池进行周期性充电。

注意事项 Attention

本部分包含一些使用和开发 SCOUT MINI 应该注意的一些事项。

电池注意事项

- SCOUT MINI 产品出厂时电池并不一定是满电状态的，具体电池电量可以通过 SCOUT MINI 底盘尾部电压显示表显示或者 CAN 总线通信接口读取得到，充电时间以充电器亮绿色指示灯表示充电完毕，但是绿灯亮起后电池依然会以 0.1A 的电流缓慢充电，可以再充 30 分钟左右；
- 请不要在电池使用殆尽以后再进行充电，在 SCOUT MINI 提示电量低的情况下请及时充电；
- 静态存放条件：存储的最佳温度为-10℃~45℃，电池在不使用的情况下存放，必须是 2 个月左右充放电一次，然后使电池处于满电压状态进行存放，请勿将电池放入火中，或对电池加热，请勿在高温下存储电池；
- 充电：必须使用配套的锂电池专用充电器进行充电，请勿在 0℃ 以下给电池充电，请勿使用非原厂标配的电池、电源、充电器。

使用环境注意事项

- SCOUT MINI 室内外工作温度为-10℃ ~ 45℃，请勿在室内外温度低于-10℃、高于 45℃环境中使用；
- SCOUT MINI 的使用环境的相对湿度要求是：最大 80%，最小 30%；
- 请勿在存在腐蚀性、易燃性气体的环境或者靠近可燃性物质的环境中使用；
- 不要存在在加热器或者大型卷线电阻等发热体周围；除特别定制版（IP 防护等级定制），SCOUT MINI 不具有防水功能，请勿在有雨、雪、积水的环境使用；
- 建议使用环境海拔高度不超过 1000M；
- 建议使用环境昼夜温差不得超过 25℃；

电气外部扩展注意事项

- 扩展电源电流不超过 5A,总功率不超过 120W；
- 当系统检测到电池电压低于安全电压以后，外部电源扩展会被主动切断，所以如果外部扩展设备涉及到重要数据的存储且无掉电保护，建议用户注意。

安全注意事项

- 使用过程有疑问，请按照相关说明手册进行操作或者咨询相关技术人员；
- 使用设备前，注意现场情况，避免误操作导致人员安全问题发生；
- 请勿未经技术支持和允许，私自改装内部设备结构。

其他注意事项

- SCOUT MINI 前后为塑料件，请勿直接捶打，否则容易损坏；
- 搬运以及设置作业时，请勿落下或者倒置；
- 非专业人员，请不要私自拆卸。

1 SCOUT MINI 简介 Introduction

SCOUT MINI 智能移动底盘采用四轮四驱，具备强悍的越野性能，身形小巧，真正实现“灵巧似燕，驰骋如心”。SCOUT MINI 继承了 SCOUT 四轮差速底盘系列四轮驱动、独立悬挂、原地自转等优点，并在轮毂电机的设计上取得了创新，底盘最小转弯半径为 0M，爬坡角度接近 30 度。SCOUT MINI 在体积上比 SCOUT 缩小一半的同时，仍具备卓越的越野性能，同时突破性实现了 10.8KM/h 的高速精准稳定可控的动力控制系统。SCOUT MINI 开发平台 自带控制核心，支持标准 CAN 总线通讯，可接入标准 CAN 总线通讯，以及各类外部设备，在此基础上支持 ROS 等二次开发和更高级的机器人开发系统接入。配置有标准航模遥控器，24V15AH 锂电池动力系统，续航里程可达 10KM。立体相机、激光雷达、GPS、IMU、机械手等设备可选择加装至 SCOUT MINI 作为扩展应用。SCOUT MINI 可被应用到无人巡检、安防、科研、勘探、物流等领域。

1.1 产品列表

| 名称 | 数量 |
|------------------|----|
| SCOUT MINI 机器人本体 | *1 |
| 电池充电器(AC 220V) | *1 |
| 航空插头公头（4Pin） | *1 |
| USB 转 CAN | *1 |
| 遥控器 | *1 |
| USB 转 RS232 | *1 |

1.2 性能参数

| 参数类型 | 项目 | 指标 |
|------|----------------|---------------|
| 机械参数 | 长 x 宽 x 高 (mm) | 612x580x245 |
| | 轴距 (mm) | 451 |
| | 轮距 (mm) | 490 |
| | 车体重量 (Kg) | 23 |
| | 额定载重 | 10Kg |
| | 电池类型 | 锂电池 24V 15Ah |
| | 电机 | 直流无刷 4 X 250W |
| | 驱动形式 | 四轮独立驱动 |
| | 悬架形式 | 摇臂独立悬架 |
| | 转向 | 四轮差速转向 |
| | 安全装备 | 伺服刹车/防撞管 |
| 性能参数 | 减速比 | 1: 4.3 |
| | 空载最高车速 (km/h) | 10 |
| | 最小转弯半径 | 0 |
| | 工作温度 | -10C°~+45C° |
| | 最大爬坡能力 | 30° |
| | 最小离地间隙 (mm) | 115 |

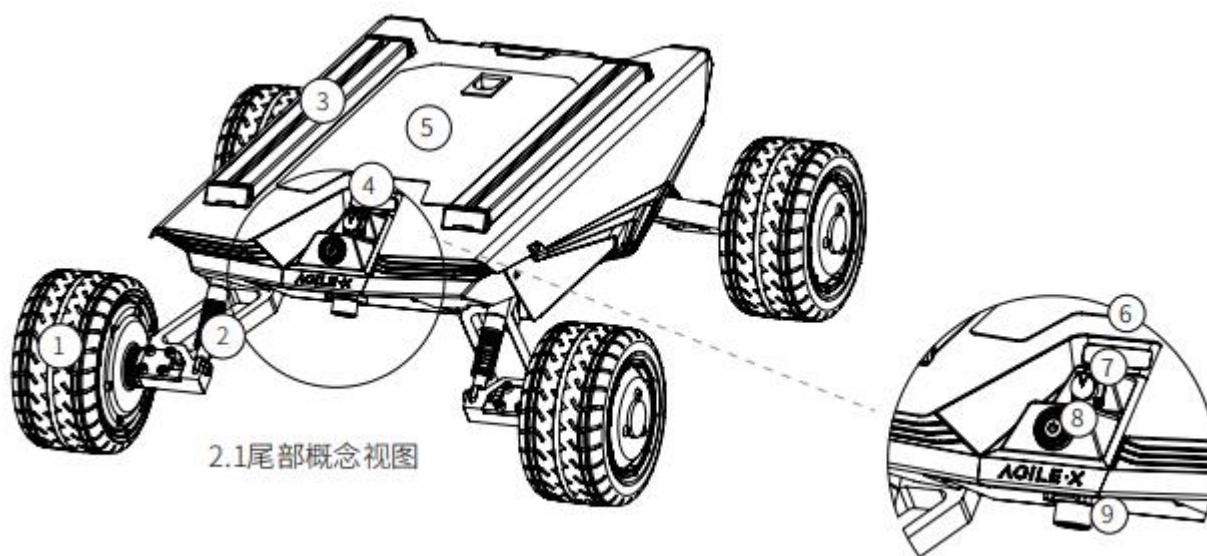
| | | |
|------|------|------------------|
| 控制参数 | 控制模式 | 遥控控制 控制指令模式 |
| | 遥控器 | 2.4G / 极限距离 200m |
| | 通讯接口 | CAN |

1.3 开发所需

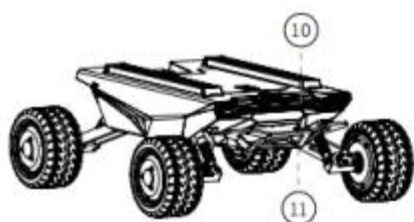
SCOUT MINI 出厂时可选配 FS 遥控器，用户可以通过遥控器控制 SCOUT MINI 移动机器人底盘，完成移动和旋转操作；SCOUT MINI 配备了 CAN 接口，用户可以通过 CAN 接口进行二次开发。

2 基本介绍 The Basics

本部分内容将会对 SCOUT MINI 移动机器人底盘作一个基本的介绍，便于用户和开发者对于 SCOUT MINI 底盘有一个基本的认识。如下图 2.1 与 2.2 所示，为整个移动机器人底盘的概览视图。



2.1 尾部概念视图



- | | |
|-----------|------------|
| 1. 轮胎 | 7. CAN扩展接口 |
| 2. 减震弹簧 | 8. 电源按键 |
| 3. 扩展支架 | 9. 充电接口 |
| 4. 控制接口区 | 10. 前部灯光 |
| 5. 电气仓室面板 | 11. 前部防撞护栏 |
| 6. 电量显示液晶 | |

2.2 前部概念视图

SCOUT MINI 整体上采用了模块化和智能化的设计思想，在动力模块上采用填充实心胎与独立悬挂的复合设计，再加上动力强劲的轮毂电机，使得 SCOUT MINI 机器人底盘开发平台具有很强的通过性和地面适应性，可在不同的地面灵活运动；轮毂电机使得整体省去了复杂的传动结构设计，使得车型变小成为可能。车体前侧安装安全防撞栏，可在发生紧急事故时，保护车体前侧，减缓对车体的损伤。车体前侧安装有灯光，前侧

采用白光设计，可进行照明。

在汽车的尾部配置了开放的电气接口和通讯接口，方便客户进行二次开发，电气接口在设计选型上采用了航空防水接插件，一方面利于用户的扩展和使用，另外一方面使得机器人平台可以在一些严苛的环境中使用。在车体顶部安装有标准铝型材扩展支架，方便用户搭载外部设备扩展使用。

2.1 机器人状态指示

用户可以通过安装在 SCOUT MINI 上的电压表、电源以及灯光来确定车体的状态。

2.2 电气接口说明

在 SCOUT MINI 尾部简约的设计，所有的电气接口均在尾部。其接口包括电压显示交互模块、扩展接口、电源按键以及充电接口。各个模块在尾部的位置如图 2.3 所示。

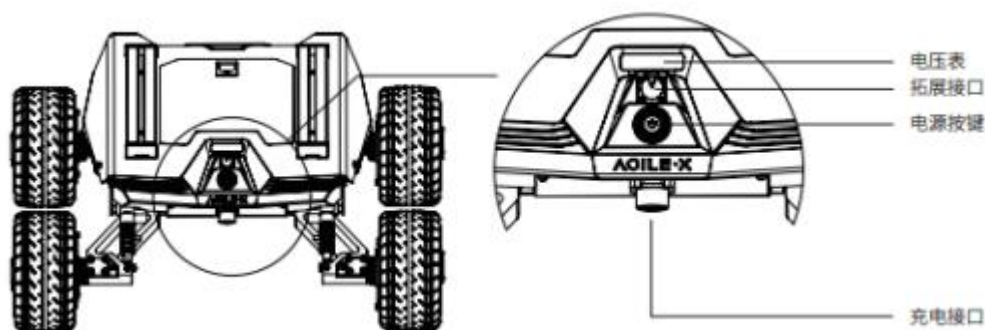
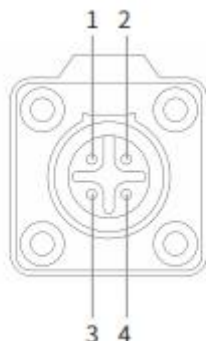


图 2.3 尾部电气面板示意图

SCOUT MINI 配置的航空扩展接口既配置了一组电源接口也配置了一组 CAN 通讯接口。便于使用者给扩展设备 提供电源，以及通讯使用。其具体引脚定义图 2.4。



| 引脚编号 | 引脚类型 | 功能及定义 | 备注 |
|------|------|-------|-------------------------------|
| 1 | 电源 | VCC | 电源正，电压范围 23-29.25V,最大电流 5A |
| 2 | 电源 | GND | 电源负 |
| 3 | CAN | CAN_H | CAN 总线高 |
| 4 | CAN | CAN_L | CAN 总线低 |

图 2.4 引脚定义图

2.3 遥控说明

富斯遥控器为 SCOUT MINI 产品选配配件，用户可根据实际需求选配，使用遥控器可以轻松控制 SCOUT MINI 通用机器人底盘，在本产品中我们采用左手油门的设计。其定义及其功能可参考图 2.5。

遥控器出厂已经预置了按键的映射，请勿随意更改按键映射，更改可能会导致无法正常控制。拨杆 SWB 切换控制模式，SWC 控制速度模式，SWD 手动灯光控制开关，左摇杆控制前进后退，右摇杆控制车子左旋转和右旋转。值得注意的是，在内部控制上移动底盘是根据百分比映射的，因此当摇杆处于同一个位置时，其速度是恒定的。



图 2.5 富斯遥控器按键示意图

遥控界面说明：

- Scout：车型
- Vol: 电池电压
- Car: 底盘状态
- Batt: 底盘电量百分比
- P:驻车
- Remoter: 遥控器电量
- Fault Code: 错误信息 (参考故障信息说明表)

2.4 遥控运动控制和指令控制说明

我们将地面移动车辆根据 ISO 8855 标准建立如图 2.6 的坐标参考系。

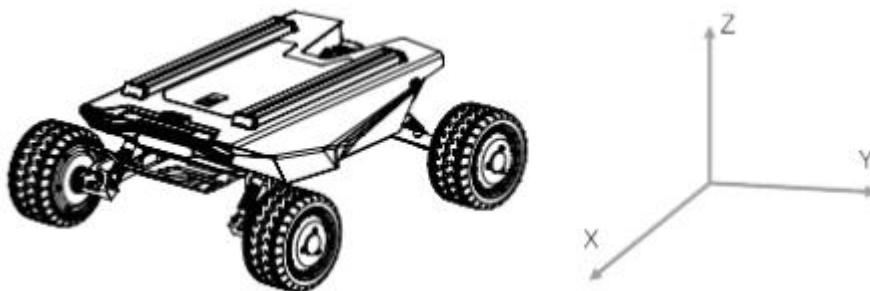
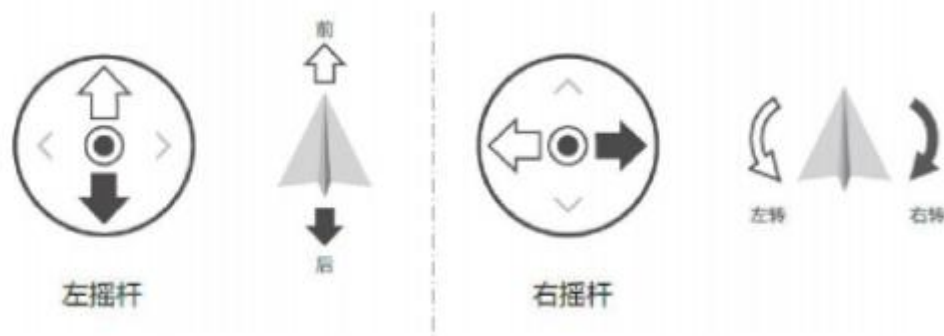


图 2.6 车身参考坐标系示意图

正如 3.0 所展示的，SCOUT MINI 车体与建立的参考坐标系 X 轴为平行状态。

在遥控器控制模式下，遥控器左摇杆往前推动则为往 X 正方向运动，遥控器左摇杆往后推动则往 X 负方向运动，遥控器左摇杆推动至最大值时，往 X 正方向运动速度最大，遥控器左摇杆推动至最小值时，往 X 负方向运动速度最大；遥控器右摇杆左右控制车体的旋转运动，遥控器右摇杆往左推动车体则由 X 轴正反向往 Y 正方向旋转，遥控器右摇杆往右推动车体则由 X 轴正方向往 Y 负方向旋转，遥控器右摇杆往左推动至最大值时，逆时针方向旋转线速度 最大，遥控器右摇杆往右推动至最大值时，顺时针旋转线速度最大。

在控制指令模式下，线速度的正值表示往 X 轴正方向运动，线速度的负值表示往 X 轴负方向运动；角速度的正值表示车体由 X 轴正方向往 Y 轴正方向运动，角速度的负值表示车体由 X 轴正方向往 Y 轴负方向运动。



2.5 灯光控制说明

SCOUT MINI 前后均配置了灯光，为了方便用户，SCOUT MINI 对外开放灯光控制接口。同时为了节省能源，在遥控器预留灯光控制接口。目前遥控器的灯光模式有 2 种，模式切换可以通过 SWD 拨杆切换：

模式控制说明：SWD 拨杆拨至最下方为常闭模式，最上方为呼吸灯模式。

常闭模式：在常闭的模式下，如果底盘静止，灯光会关闭；如果底盘在正常速度行驶状灯光会打开；

常开模式：在常开的模式下，如果底盘静止不动，灯光常开；如果在运动模式下，灯光打开；

呼吸灯模式：灯光为呼吸灯模式。

2.6 速度模式说明

为方便用户，SCOUT MINI 设置了 3 种速度模式，分别为低速、中速和高速，可通过遥控器 SWC 拨杆进行切换。

模式控制说明：SWC 拨杆拨至最上方为中速模式，中间为低速模式，最下方为高速模式。

低速模式：最大速度为 0.5 m/s。

中速模式：最大速度为 1.5 m/s。

高速模式：最大速度为 3.0 m/s。

3 使用与开发 Getting Started

本部分主要介绍 SCOUT MINI 平台的基本操作与使用，介绍如何通过外部 CAN 口，通过 CAN 总线协议来对车体进行二次开发。

3.1 使用与操作

检查：

- 检查车体状态。检查车体是否有明显异常；如有，请联系售后支持；

启动：

- 按下 SCOUT MINI 电源按键，等待数秒即可；
- 将 SWB 拨至中间；
- 可尝试手动切换灯光模式，确定模式选择是否正确；
- 尝试将左边摇杆轻轻往前推，推一小部分即可，可见小车缓慢速度往前移动；
- 尝试将左边摇杆轻轻往后推，推一小部分即可，可见小车缓慢速度往后移动；
- 释放左边摇杆，小车停下；

- 尝试将右边摇杆轻轻往左推，推一小部分即可，可见小车缓慢往左旋转；
- 尝试将右边摇杆轻轻往右推，推一小部分即可，可见小车缓慢往右旋转；
- 释放右边摇杆，小车停下；
- 可尝试在相对空旷的区域自由控制，熟悉车辆移动速度。

关闭操作：

- 按下 SCOUT MINI 电源按键，释放即可。

遥控控制基本操作流程：

- 正常启动 SCOUT MINI 底盘后，启动遥控器，将控制模式选择为遥控控制模式，即可通过遥控器控制 SCOUT MINI 平台运动。

3.2 充电

- SCOUT MINI 产品默认随车配备一个 10A 的充电器，可满足客户的充电需求。默认关机充电，正常充电时，底盘没有指示灯说明。具体指示灯请看充电器上说明。
- 确保 SCOUT MINI 底盘处于停机断电状态。
- 将充电器的插头插入车尾充电接口；
- 将充电器连接电源，将充电器开关打开，即可进入充电状态。

注意：当前电池从 22V 充满电状态大约需要 1.5 小时，电池充满电电压约为 29.2V；充电时间计算 $15Ah/10A=1.5H$

3.3 开发

SCOUT MINI 产品针对用户的开发提供了 CAN 的接口，用户可用 CAN 指令对车体进行指令控制。

3.3.1 CAN 线的连接

SCOUT MINI 随车发货提供了一个航空插头公头如 图 3.2,线的定义可参考表 2.2。注：当前 SCOUT MINI 版本对外扩展接口仅尾部接口开放。此版本中电源最大可提供 5A 的电流。



图 3.2 航空插头公头示意图

3.3.2 CAN 指令控制的实现

正常启动 SCOUT MINI 移动机器人底盘，打开富斯遥控器，然后将控制模式切换至指令控制，即将富斯遥控器 SWB 模式选择拨至最上方，此时 SCOUT MINI 底盘会接受来自 CAN 接口的指令，同时主机也可

以通过 CAN 总线回馈的实时数据，解析当前底盘的状态，具体协议内容参考 CAN 通讯协议。

3.3.3 CAN 接口协议

SCOUT MINI 产品中 CAN 通信标准采用的是 CAN2.0B 标准，通讯波特率为 500K，报文格式采用 MOTOROLA 格式。通过外部 CAN 总线接口可以控制底盘的移动的线速度以及旋转的角速度；SCOUT MINI 会实时反馈当前的运动状态信息以及 SCOUT MINI 底盘的状态信息等。

协议包含系统状态回馈帧、运动控制回馈帧、控制帧，协议内容具体如下：

系统状态回馈指令包含了当前车体状态回馈、控制模式状态回馈、电池电压回馈以及故障回馈，协议内容如表 3.1 所示。

表格 3.1 SCOUT MINI 底盘系统状态回馈帧

| 指令名称 | 系统状态回馈指令 | | | |
|----------|--------------|----------------|---|-----------|
| 发送节点 | 接收节点 | ID | 周期 (ms) | 接受超时 (ms) |
| 线控底盘 | 决策控制单元 | 0x211 | 20ms | 无 |
| 数据长度 | 0x08 | | | |
| 位置 | 功能 | 数据类型 | 说明 | |
| byte [0] | 当前车体状态 | unsigned int8 | 0x00 系统正常 0x02 系统异常 | |
| byte [1] | 模式控制 | unsigned int8 | 0x00 待机模式 0x01 CAN 指令控制模式 0x02 串口控制模模式[1] 0x03 遥控控制模模式 | |
| byte [2] | 电池电压高八位 | unsigned int16 | 实际电压 X 10 (精确到 0.1V) | |
| byte [3] | 电池电压低八位 | unsigned int16 | 实际电压 X 10 (精确到 0.1V) | |
| byte [4] | 保留 | - | 0x00] | |
| byte [5] | 故障信息 | unsigned int8 | 详见备注[故障信息说明] | |
| byte [6] | 保留 | - | 0x00 | |
| byte [7] | 计数校验 (count) | unsigned int8 | 0~255 循环计数，每发送一条指令计数加一次 | |

表格 3.2 故障信息说明表

| 故障信息说明 | | |
|--------|---|----|
| 字节 | 位 | 含义 |
| | | |

| | | |
|----------|---------|-------------------------------------|
| byte [5] | bit [0] | 电池欠压故障 (0:无故障 1: 故障) 保护电压为 20.5V |
| | bit [1] | 电池欠压警告[2] (0:无警告 1: 警告) 报警电压为 22.5V |
| | bit [2] | 遥控器失联保护 (0: 正常, 1: 遥控器失联) |
| | bit [3] | 驱动 1 通讯故障(0: 无故障, 1: 故障) |
| | bit [4] | 驱动 2 通讯故障(0: 无故障, 1: 故障) |
| | bit [5] | 驱动 3 通讯故障(0: 无故障, 1: 故障) |
| | bit [6] | 驱动 4 通讯故障(0: 无故障, 1: 故障) |
| | bit [7] | 预留, 默认 0 |

[1]: 机器人底盘固件版本 V1.2.8 后续版本支持, 之前版本需要升级固件方可支持

[2]: 电池欠压警告标志置位时蜂鸣器响, 但是底盘控制不受影响, 欠压故障后会切断动力输出
运动控制反馈帧指令包含了当前车体的运动线速度、运动角速度反馈, 协议具体内容如表 3.3 所示。

表格 3.3 运动控制反馈帧

| 指令名称 | 运动控制反馈指令 | | | |
|----------|----------|--------------|------------------------------|-----------|
| 发送节点 | 接收节点 | ID | 周期 (ms) | 接受超时 (ms) |
| 线控底盘 | 决策控制单元 | 0x221 | 20ms | 无 |
| 数据长度 | 0x08 | | | |
| 位置 | 功能 | 数据类型 | 说明 | |
| byte [0] | 移动速度高八位 | signed int16 | 实际速度 X 1000 (精确到 0.001m/s) | |
| byte [1] | 移动速度低八位 | signed int16 | | |
| byte [2] | 旋转速度高八位 | signed int16 | 实际速度 X 1000 (精确到 0.001rad/s) | |
| byte [3] | 旋转速度低八位 | signed int16 | | |
| byte [4] | 保留 | - | 0x00 | |
| byte [5] | 保留 | - | 0x00 | |
| byte [6] | 保留 | - | 0x00 | |
| byte [7] | 保留 | - | 0x00 | |

运动控制帧包含了线速度控制开度、角速度控制开度, 其具体协议内容如表 3.4 所示。

表格 3.4 运动控制指令控制帧

| 指令名称 | 控制指令 | | | |
|--------|------|-------|---------|-----------|
| 发送节点 | 接收节点 | ID | 周期 (ms) | 接受超时 (ms) |
| 决策控制单元 | 底盘节点 | 0x111 | 20ms | 500ms |
| 数据长度 | 0x08 | | | |
| 位置 | 功能 | 数据类型 | 说明 | |

| | | | |
|----------|--------|--------------|--|
| byte [0] | 线速度高八位 | signed int16 | 车体行进速度, 单位 mm/s, 值域 [-3000,3000] |
| byte [1] | 线速度低八位 | signed int16 | |
| byte [2] | 角速度高八位 | signed int16 | 车体旋转角速度, 单位 0.001rad/s, 值域[-2523,2523] |
| byte [3] | 角速度低八位 | signed int16 | |
| byte [4] | 保留 | - | 0x00 |
| byte [5] | 保留 | - | 0x00 |
| byte [6] | 保留 | - | 0x00 |
| byte [7] | 保留 | - | 0x00 |

模式设定帧用于设定终端的控制接口, 其具体协议内容如表 3.5 所示。

表格 3.5 控制模式设定帧

| 指令名称 | 控制模式设定指令 | | | |
|----------|----------|---------------|------------------------------|-----------|
| 发送节点 | 接收节点 | ID | 周期 (ms) | 接受超时 (ms) |
| 决策控制单元 | 底盘节点 | 0x421 | 无 | 无 |
| 数据长度 | 0x01 | | | |
| 位置 | 功能 | 数据类型 | 说明 | |
| byte [0] | CAN 控制使能 | unsigned int8 | 0x00 待机模式 0x01 CAN 指令模式使能 | |

控制模式说明:

SCOUT MINI 在开机上电, 遥控器未连接的情况下, 控制模式默认是待机模式, 此时底盘只接收控制模式指令, 速度指令不做响应, 要使用 CAN 控制就需要先使能 CAN 控制模式。若打开遥控器, 遥控器具有最高权限, 可以屏蔽指令的控制, 可以切换控制模式。

状态置位帧用于清除系统错误, 其具体协议内容如表 3.6 所示

表格 3.6 状态置位帧

| 指令名称 | 状态设定指令 | | | |
|----------|--------|---------------|---|-----------|
| 发送节点 | 接收节点 | ID | 周期 (ms) | 接受超时 (ms) |
| 决策控制单元 | 底盘节点 | 0x441 | 无 | 无 |
| 数据长度 | 0x01 | | | |
| 位置 | 功能 | 数据类型 | 说明 | |
| byte [0] | 错误清除指令 | unsigned int8 | 0x00 清除所有错误 0x01 清除电机 1 错误 0x02 清除电机 2 错误 0x03 清除电机 3 错位 0x04 清除电机 4 错误 | |

[注 3]示例数据, 以下数据仅供测试使用

1、小车以 0.15m/S 的速度前进

| | | | | | | | |
|----------|----------|----------|----------|----------|----------|----------|----------|
| byte [0] | byte [1] | byte [2] | byte [3] | byte [4] | byte [5] | byte [6] | byte [7] |
| 0x00 | 0x96 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |

2、小车以 0.2RAD/S 旋转

| | | | | | | | |
|----------|----------|----------|----------|----------|----------|----------|----------|
| byte [0] | byte [1] | byte [2] | byte [3] | byte [4] | byte [5] | byte [6] | byte [7] |
| 0x00 | 0x00 | 0x00 | 0xc8 | 0x00 | 0x00 | 0x00 | 0x00 |

除了底盘的状态信息会进行反馈以外，底盘反馈的信息还包括电机的电流信息、编码器数据以及温度信息。下面的帧反馈是电机的电流信息、编码器信息以及电机温度信息：

在底盘中四个电机编号对应为如下图所示：

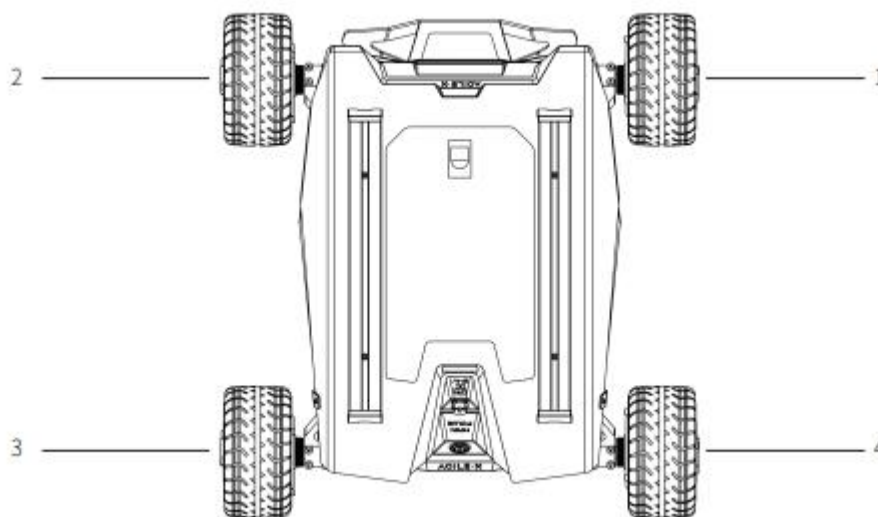


图 3.0 电机反馈 ID 示意图

表格 3.7 电机转速电流位置信息反馈

| 指令名称 | 电机驱动器高速信息反馈帧 | | | |
|----------|--------------|--------------|----------------|-----------|
| 发送节点 | 接收节点 | ID | 周期 (ms) | 接受超时 (ms) |
| 线控底盘 | 决策控制单元 | 0x251~0x254 | 20ms | 无 |
| 数据长度 | 0x08 | | | |
| 位置 | 功能 | 数据类型 | 说明 | |
| byte [0] | 电机转速高八位 | signed int16 | 电机当前转速 单位 RPM | |
| byte [1] | 电机转速低八位 | signed int16 | | |
| byte [2] | 电机电流高八位 | signed int16 | 电机当前电流 单位 0.1A | |

| | | | |
|----------|---------|--------------|------|
| byte [3] | 电机电流低八位 | signed int16 | |
| byte [4] | 保留 | - | 0x00 |
| byte [5] | 保留 | - | |
| byte [6] | 保留 | - | |
| byte [7] | 保留 | - | |

表格 3.8 电机驱动器信息反馈

| 指令名称 | 电机驱动器低速信息反馈帧 | | | |
|----------|--------------|----------------|-----------------|----------|
| 发送节点 | 接收节点 | ID | 周期 (ms) | 接收超时(ms) |
| 线控底盘 | 决策控制单元 | 0x261~0x264 | 100ms | 无 |
| 数据长度 | 0x08 | | | |
| 位置 | 功能 | 数据类型 | 说明 | |
| byte [0] | 驱动器电压高八位 | unsigned int16 | 当前驱动器电压 单位 0.1V | |
| byte [1] | 驱动器电压低八位 | unsigned int16 | | |
| byte [2] | 驱动器温度高八位 | signed int16 | 单位 1°C | |
| byte [3] | 驱动器温度低八位 | signed int16 | 单位 1°C | |
| byte [4] | 电机温度 | signed int8 | 单位 1°C | |
| byte [5] | 驱动器状态 | unsigned int8 | 详见表 3.9 | |
| byte [6] | 保留 | - | 0x00 | |
| byte [7] | 保留 | - | 0x00 | |

表格 3.9 驱动器状态

| 字节 | 位 | 说明 |
|---------|--------|------------------------|
| byte[5] | bit[0] | 电源电压是否过低 (0: 正常 1: 过低) |
| | bit[1] | 电机是否过温 (0: 正常 1: 过温) |
| | bit[2] | 电机是否过流 (0: 正常 1: 过流) |
| | bit[3] | 驱动器是否过温 (0: 正常 1: 过温) |
| | bit[4] | 保留 |
| | bit[5] | 保留 |
| | bit[6] | 保留 |
| | bit[7] | 保留 |

前部和外部的灯光也支持指令控制，下表为控制的指令

表格 3.10 灯光控制帧

| 指令名称 | 灯光控制帧 | | | |
|------|-------|----|---------|----------|
| 发送节点 | 接收节点 | ID | 周期 (ms) | 接收超时(ms) |

| | | | | |
|----------|--------------|---------------|--|-------|
| 决策控制单元 | 线控底盘 | 0x121 | 100ms | 500ms |
| 数据长度 | 0x08 | | | |
| 位置 | 功能 | 数据类型 | 说明 | |
| byte [0] | 灯光控制使能标志 | unsigned int8 | 0x00 控制指令无效 0x01 灯光控制使能 | |
| byte [1] | 前侧灯光模式 | unsigned int8 | 0x00 常关 0x01 常开 0x02 呼吸灯模式 0x03 客户自定义亮度 | |
| byte [2] | 前侧灯光自定义亮度 | unsigned int8 | [0,100],其中 0 为不亮, 100 最亮[5] | |
| byte [3] | 保留 | - | 0x00 | |
| byte [4] | 保留 | - | 0x00 | |
| byte [5] | 保留 | - | 0x00 | |
| byte [6] | 保留 | - | 0x00 | |
| byte [7] | 计数校验 (count) | unsigned int8 | 0~255 循环计数, 每发送一条指令计数加一次 | |

注[5]: 此值只在自定义模式下有效

表格 3.11 灯光控制反馈帧

| 指令名称 | 灯光控制反馈帧 | | | |
|----------|--------------|---------------|--|----------|
| 发送节点 | 接收节点 | ID | 周期 (ms) | 接收超时(ms) |
| 线控底盘 | 决策控制单元 | 0x231 | 500ms | 无 |
| 数据长度 | 0x08 | | | |
| 位置 | 功能 | 数据类型 | 说明 | |
| byte [0] | 当前灯光控制使能标志 | unsigned int8 | 0x00 控制指令无效 0x01 灯光控制使能 | |
| byte [1] | 当前前侧灯光模式 | unsigned int8 | 0x00 常关 0x01 常开 0x02 呼吸灯模式 0x03 客户自定义亮度 | |
| byte [2] | 当前前侧灯光自定义亮度 | unsigned int8 | [0,100],其中 0 为不亮, 100 最亮 | |
| byte [3] | 保留 | - | 0x00 | |
| byte [4] | 保留 | - | 0x00 | |
| byte [5] | 保留 | - | 0x00 | |
| byte [6] | 保留 | - | 0x00 | |
| byte [7] | 计数校验 (count) | unsigned int8 | 0~255 循环计数, 每发送一条指令计数加一次 | |

表格 3.12 系统版本信息查询帧

| 指令名称 | 系统版本信息查询指令 |
|------|------------|
|------|------------|

| 发送节点 | 接收节点 | ID | 周期 (ms) | 接收超时(ms) |
|----------|--------|---------------|---------|----------|
| 决策控制单元 | 底盘节点 | 0x411 | 无 | 无 |
| 数据长度 | 0x01 | | | |
| 位置 | 功能 | 数据类型 | 说明 | |
| byte [0] | 查询系统版本 | unsigned int8 | 固定 0x01 | |

表格 3.13 系统版本信息查询帧

| 指令名称 | 系统版本信息反馈帧 | | | |
|----------|-------------|----------------|-------------------|----------|
| 发送节点 | 接收节点 | ID | 周期 (ms) | 接收超时(ms) |
| 线控底盘 | 决策控制单元 | 0x41A | 无 | 无 |
| 数据长度 | 0x08 | | | |
| 位置 | 功能 | 数据类型 | 说明 | |
| byte [0] | 主控硬件版本号高八位 | unsigned int16 | 高八位为主版本号,低八位为次版本号 | |
| byte [1] | 主控硬件版本号低八位 | | | |
| byte [2] | 驱动器硬件版本号高八位 | unsigned int16 | 高八位为主版本号,低八位为次版本号 | |
| byte [3] | 驱动器硬件版本号低八位 | | | |
| byte [4] | 主控软件版本号高八位 | unsigned int16 | 高八位为主版本号,低八位为次版本号 | |
| byte [5] | 主控软件版本号低八位 | | | |
| byte [6] | 驱动器软件版本号高八位 | unsigned int16 | 高八位为主版本号,低八位为次版本号 | |
| byte [7] | 驱动器软件版本号低八位 | | | |

表格 3.14 里程计信息反馈

| 指令名称 | 里程计信息反馈 | | | |
|----------|--|--------------|------------------|----------|
| 发送节点 | 接收节点 | ID | 周期 (ms) | 接收超时(ms) |
| 线控底盘 | 决策控制单元 | 0x311 | 20ms | 无 |
| 数据长度 | 0x08 | | | |
| 位置 | 功能 | 数据类型 | 说明 | |
| byte [0] | 左轮里程计最高位 左轮里程计低高位 左轮里程计次低位 左轮里程计最低位 | signed int32 | 底盘左轮里程计反馈 单位: mm | |
| byte [1] | | | | |
| byte [2] | | | | |
| byte [3] | | | | |
| byte [4] | 右轮里程计最高位 右轮里程计次高位 右轮里程计次低位 右轮里程计最低位 | signed int32 | 底盘右轮里程计反馈 单位: mm | |
| byte [5] | | | | |
| byte [6] | | | | |
| byte [7] | | | | |

表格 3.15 遥控器信息反馈

| 指令名称 | 遥控器信息反馈帧 | | | |
|------|----------|--|--|--|
|------|----------|--|--|--|

| 发送节点 | 接收节点 | ID | 周期 (ms) | 接收超时(ms) |
|---------|----------|---------------|---|----------|
| 线控底盘 | 决策控制单元 | 0x241 | 20ms | 无 |
| 数据长度 | 0x08 | | | |
| 位置 | 功能 | 数据类型 | 说明 | |
| byte[0] | 遥控 SW 反馈 | unsigned int8 | bit[0-1]: SWA : 2-上档 3-下档 bit[2-3]: SWB: 2-上档 1-中档 3-下档 bit[4-5]: SWC: 2-上档 1-中档 3-下档 bit[6-7]: SWD: 2-上档 3-下档 | |
| byte[1] | 右边拨杆左右 | signed int8 | 值域: [-100,100] | |
| byte[2] | 右边拨杆上下 | signed int8 | 值域: [-100,100] | |
| byte[3] | 左边拨杆上下 | signed int8 | 值域: [-100,100] | |
| byte[4] | 左边旋钮左右 | signed int8 | 值域: [-100,100] | |
| byte[5] | 左边旋钮 VRA | signed int8 | 值域: [-100,100] | |
| byte[6] | 保留 | - | 0x00 | |

3.4 固件升级

为了方便解决用户对 SCOUT MINI 所使用的固件版本进行升级, 给客户带来更加完善的体验, SCOUT MINI 提供了固件升级的硬件接口以及与之对应的客户端软件。其客户端界面如图 3.3 所示。

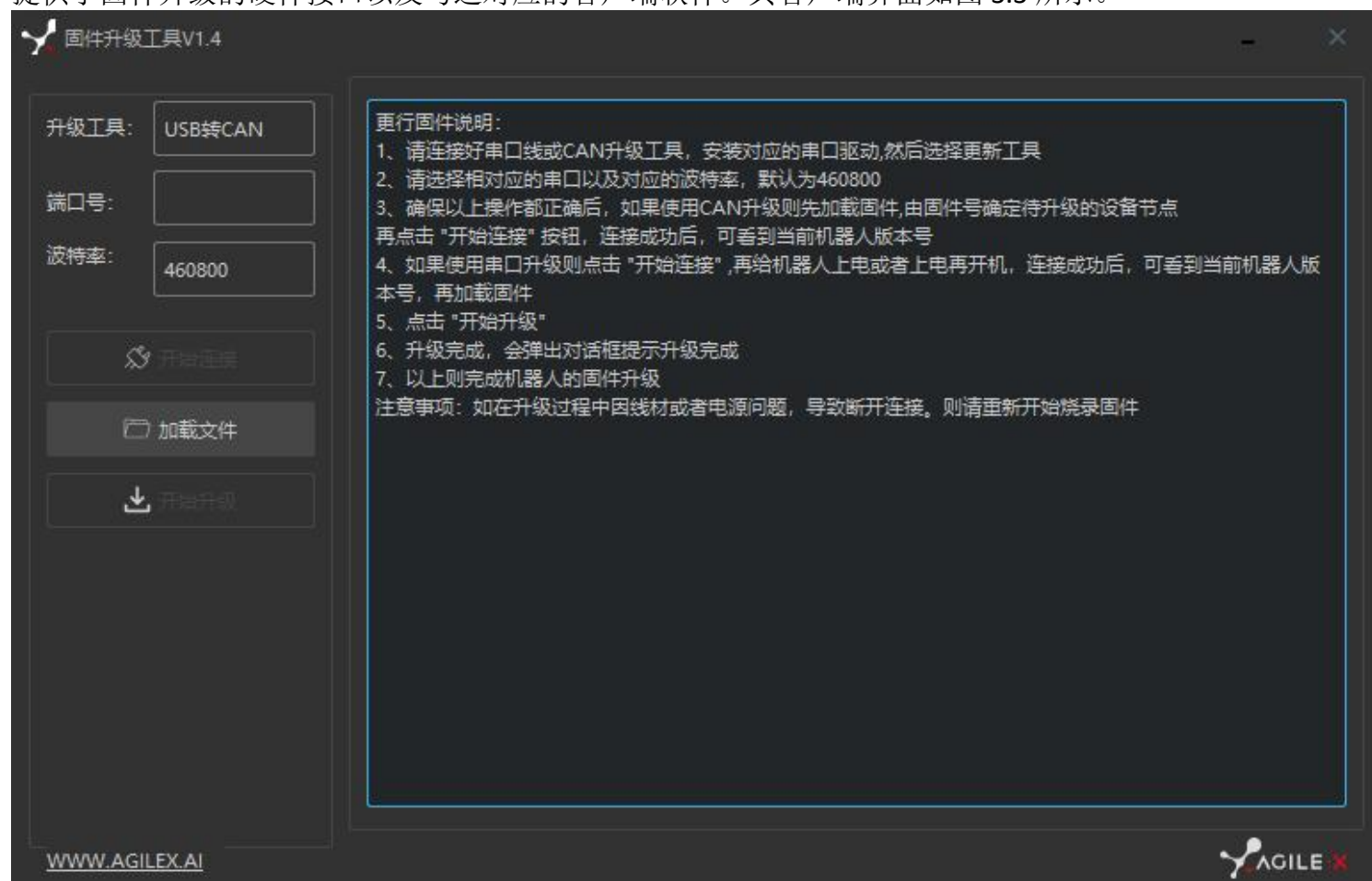


图 3.3 固件升级客户端界面

升级准备

- 串口线 X 1
- USB 转串口 X 1
- SCOUT2.0 底盘 X 1
- 电脑(WINDOWS 操作系统) X 1

固件升级软件:

- 链接: <https://pan.baidu.com/s/1wdjrr4swhqLT9PcsrU-nzg?pwd=i34d> 提取码: i34d

升级过程

- 连接前保证机器人底盘电源处于断开状态;
- 使用串口线连接至 SCOUT MINI 底盘内部串口;
- 串口线连接至电脑;
- 打开客户端软件;
- 选择端口号;
- SCOUT MINI 底盘上电, 立即点击开始连接 (SCOUT MINI 底盘会在上电前 3S, 如果时间超过 3S 则会进入应用程序); 若连接成功, 会在文本框提示“连接成功”;
- 加载 BIN 文件;
- 点击升级, 等待升级完成的提示即可;
- 断开串口线, 底盘断电, 再次通电即可。

3.5 SCOUT MINI 2.0 ROS Package 使用示例

ROS 提供一些标准操作系统服务, 例如硬件抽象, 底层设备控制, 常用功能实现, 进程间消息以及数据包管理。ROS 是基于一种图状架构, 从而不同节点的进程能接受, 发布, 聚合各种信息 (例如传感, 控制, 状态, 规划等等)。目前 ROS 主要支持 UBUNTU

开发准备

硬件准备

- CANlight can 通讯模块 X1
- Thinkpad E470 笔记本电脑 X1
- AGILEX SCOUT MINI 2.0 移动机器人底盘 X1
- AGILEX SCOUT MINI 2.0 配套遥控器 FS-i6s X1
- AGILEX SCOUT MINI 2.0 顶部航空插座 X1

使用示例环境说明

- Ubuntu 16.04 LTS (此为测试版本, 在 Ubuntu 18.04 LTS 测试过)
- ROS Kinetic (后续版本亦测试过)
- Git

硬件连接与准备

- 将 SCOUT MINI 顶部航空插头或者尾部插头 CAN 线引出, 将 CAN 线中的 CAN_H 和 CAN_L 分别与 CAN_TO_USB 适配器相连;
- 打开 SCOUT MINI 移动机器人底盘旋钮开关, 检查来两侧的急停开关是否释放;
- 将 CAN_TO_USB 连接至笔记本的 usb 口。连接示意如图 3.4 所示。



图 3.4 CAN 线连接示意图

ROS 安装和环境设置: 安装具体可以参考: <http://wiki.ros.org/kinetic/Installation/Ubuntu>

测试 CANABLE 硬件与 CAN 通讯:

- 使能 `gs_usb` 内核模块

```
$ sudo modprobe gs_usb
```

- 设置 500k 波特率和使能 `can-to-usb` 适配器

```
$ sudo ip link set can0 up type can bitrate 500000
```

- 如果在前面的步骤中没有发生错误, 您应该可以使用命令立即查看 `can` 设备

```
$ ifconfig -a
```

- 安装并使用 `can-utils` 来测试硬件

```
$ sudo apt install can-utils
```

- 若此次 `can-to-usb` 已经和 SCOUT MINI 2.0 机器人相连, 且小车已经开启的情况下, 使用下列指令可以监听来自 SCOUT MINI 2.0 底盘的数据了

```
$ candump can0
```

- 参考来源:

https://github.com/agilexrobotics/agx_sdk

AGILEX SCOUT MINI 2.0 ROS PACKAGE 下载与编译:

- 下载 `ros` 依赖包

```
$ sudo apt install -y libasio-dev
```

```
$ sudo apt install -y ros-$ROS_DISTRO-teleop-twist-keyboard
```

- 克隆编译 `scout_mini_ros 2.0` 源码

```
$ cd ~/catkin_ws/src
```

```
$ git clone https://github.com/agilexrobotics/ugv_sdk.git
```

```
$ git clone https://github.com/agilexrobotics/scout_ros.git
```

```
$ cd ..
```

```
$ catkin_make
```

参考来源:

https://github.com/agilexrobotics/scout_base

启动 ROS 节点:

- 启动基础节点

```
$ roslaunch scout_bringup scout_mini_robot_base.launch
```

- 启动键盘远程操作节点

```
$ roslaunch scout_bringup scout_teleop_keyboard.launch
```

5 常见问题与解决 Q&A

Q: SCOUT MINI 启动正常, 使用遥控器控制车体不移动?

A: 首先确认供电是否正常; 然后确认遥控器的左侧上方模式选择开关选择的控制模式是否正确。

Q: SCOUT MINI 遥控控制正常, 底盘状态、运动信息反馈正常, 下发控制帧协议, 车体控制模式无法切换, 底盘不响应控制帧协议?

A: 正常情况下, SCOUT MINI 若可以通过遥控器控制正常情况下, 说明底盘运动控制正常, 可以接收到底盘的反馈帧, 说明 CAN 扩展链路正常。默认模式是待机模式, 需要先切换指令控制模式, 才可以发送指令控制运动帧。

Q: SCOUT MINI 在运行中发出“滴-滴-滴...”的声音, 该如何处理?

A: 若 SCOUT MINI 发出连续的“滴-滴-滴...”表明电池已经处于警报电压状态, 请及时充电出现相关声音以后, 亦可能是内部出现相关错误了, 可以通过 CAN 总线检查相关的错误代码, 或与相关技术人员沟通。

Q: SCOUT MINI 在运行过程中出现轮胎磨损情况是属于正常现象嘛？

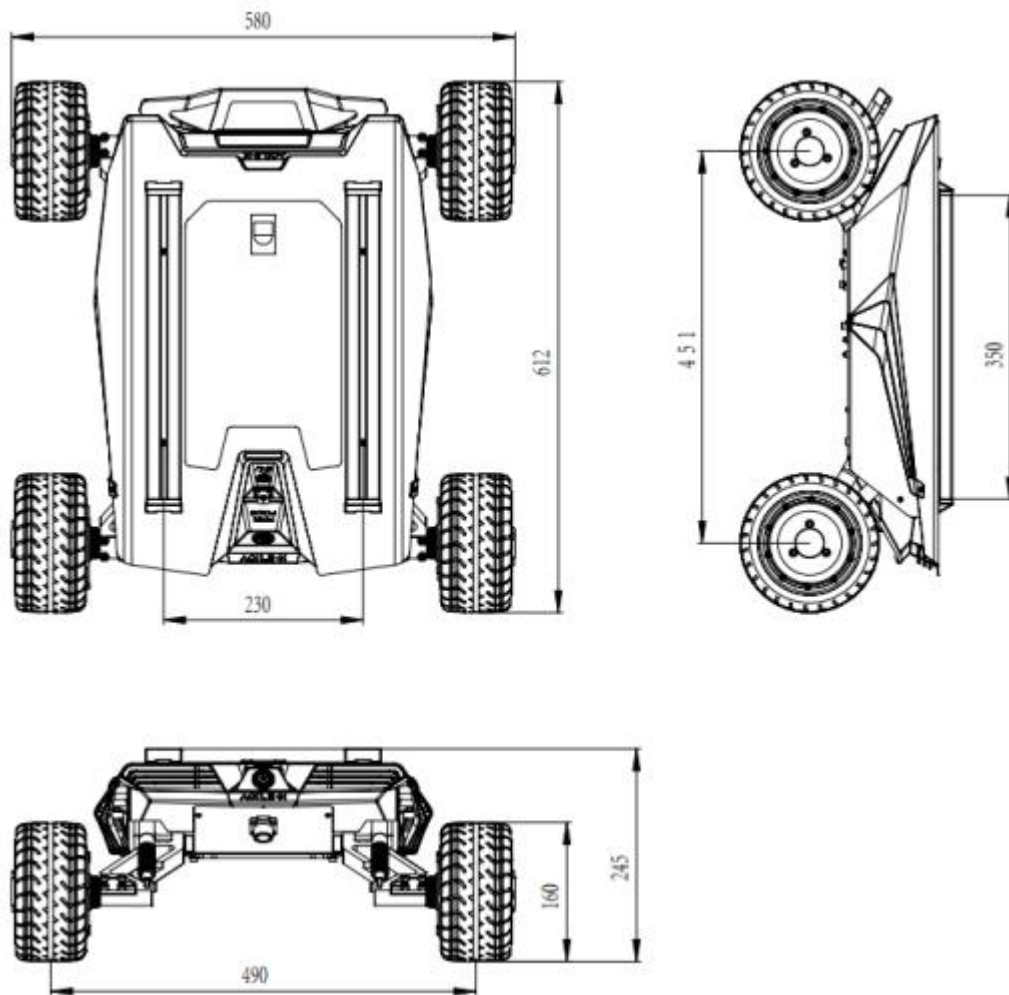
A: SCOUT MINI 在运行过程中出现轮胎磨损属于正常现象。由于 SCOUT MINI 采用的是四轮差速转向的设计，在车体旋转的过程中会出现滑动摩擦和滚动摩擦并存的情况，如果地面不光滑，表面粗糙，这个时候对轮胎表面存在磨损情况。为了减少磨损或者减缓磨损的过程，可以采用小角度转弯的形式，尽量减少原地旋转的形式。

Q:通过 CAN 总线进行相关通讯时，底盘反馈指令正常，下发控制小车无响应？

A: SCOUT MINI 的内部有通讯保护机制，底盘在处理来自外部的 CAN 控制指令时存在超时保护机制，假设小车收到一帧通讯协议以后，小车超过 500MS 未收到下一帧控制指令，小车会进入通讯保护，速度为 0，所以来自上位机的指令必须是周期性的发布。

6 产品尺寸 Product Dimensions

6.1 产品外形尺寸说明图



6.2 顶部扩张支架尺寸说明图

