

Nav-Tools 使用手册

目录

1 简介	3
2 基本使用	3
2.1 菜单栏	3
2.2 工具栏	4
2.3 状态栏	5
3 机器人应用 (Robot)	6
3.1 Flow 数据流	6
3.1.1 日志组件 (Console)	7
3.1.2 时序数据 (Data)	8
3.1.3 轨迹数据 (Deviation)	9
4 感知模块 (PERC)	10
4.1 超声波避障 (Ultrasonic)	10
4.1.1 日志组件 (Console)	10
4.1.2 时序数据 (Data)	11
5 定位模块 (POS)	12

5.1 GNSS	12
5.1.1 日志组件（Console）	12
5.1.2 轨迹数据（Deviation）	13
5.1.3 信号数据（Signal）	13
5.1.4 星空视图（Sky）	14
6 规控模块（PNC）	15
6.1 Motor	15
6.1.1 配置组件（Config）	16
6.1.2 日志组件（Console）	17
6.1.3 时序数据（Data）	17
6.2 FollowSim.....	18
6.2.1 仪表盘组件（Dashboard）	18
6.2.2 时序数据（Data）	19
6.2.3 控制面板（Config）	19

1 简介

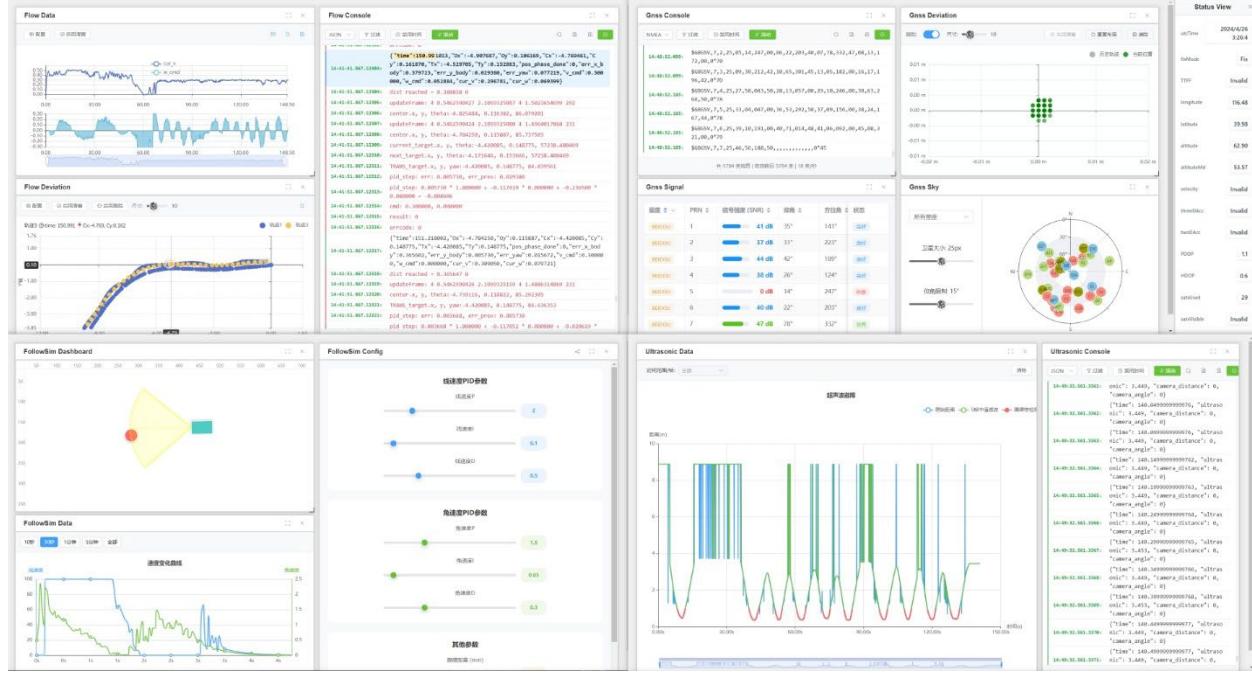


图 2-1: Nav-Tools 示意图

Nav-Tools 是一款专为机器人开发者打造的桌面级可视化工作台，全面覆盖机器人复合应用与主流传感器。目前已集成通用数据流 Flow、GNSS 定位模块、超声波避障、PID 跟随仿真和电机驱动等功能，支持布局与数据字段的灵活自定义，极大提升调试效率与开发体验。

2 基本使用

2.1 菜单栏

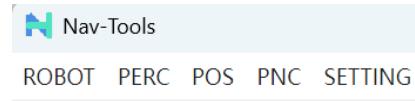


图 2-1: Nav-Tools 菜单栏

菜单栏汇集了当前所有的应用与模块。

1. ROBOT (机器人应用)
 - (a) Flow (数据流) : 整合了机器人开发中通用且常见的可视化需求
2. PERC (感知模块)
 - (a) Ultrasonic (超声波) : 避障滤波处理
3. POS (定位模块)
 - (a) GNSS (卫星导航) : 基本实现了常见的 GNSS 上位机功能
4. PNC (规划与控制)
 - (a) Motor (电机驱动) : 实现了电机驱动的控制下发与数据可视化
 - (b) FollowSim (PID 跟随仿真) : 实现了基于 PID 的跟随机器人仿真
5. SETTING (设置)
 - (a) 页面刷新设置
 - (b) 视图放缩设置

2.2 工具栏



图 2-2: Nav-Tools 工具栏

1. 工具栏拖拽锚点: 工具栏能吸附在上下左右四个方向的边框上
2. 设备连接开关: 用于开启设备连接，并接通实时数据，不过要求在 Input 中进行配置设备信息
3. 设备配置（输入配置）: 当前支持串口输入、文件输入（文件类型支持拖拽）
4. 状态栏视图: 用于显示或隐藏状态栏，每个应用均包含状态栏
5. 可视化组件: 根据应用的不同，将加载不同的可视化组件，例如
 - (a) Flow 应用加载 Data、Console 和 Deviation 组件；
 - (b) GNSS 应用 Console、Deviation、Signal 和 Sky 组件。

6. 页面布局

- (a) Save: 当页面布局更改时，触发是否保存
- (b) Auto: 将当前组件调节为紧凑布局
- (c) Reset: 恢复默认组件布局

2.3 状态栏

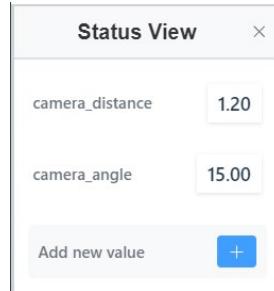


图 2-3: Nav-Tools 状态栏

根据应用的不同，状态栏将加载不同的数据状态。

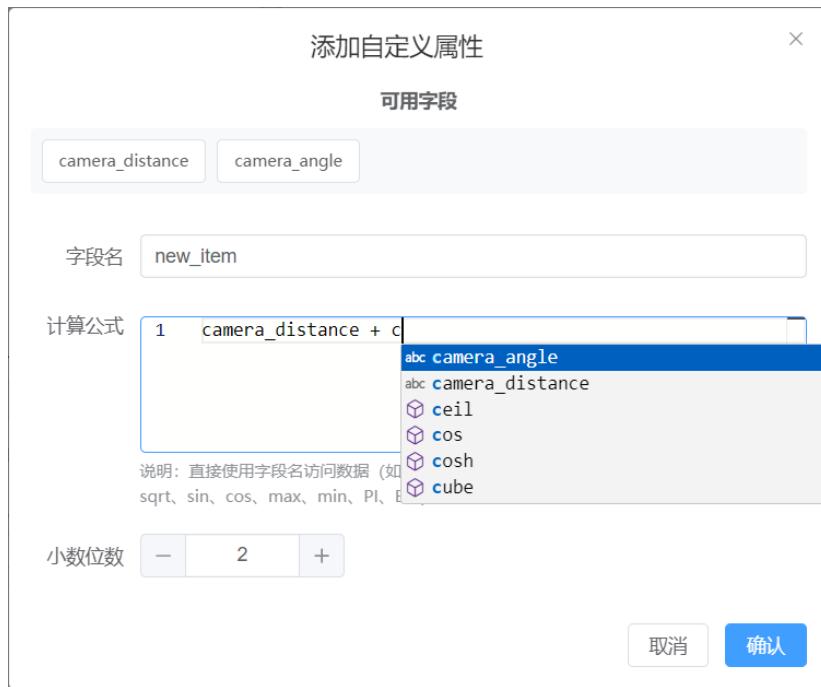


图 2-4: 状态栏自定义属性

部分应用（如 Flow）支持 Add new value 功能，可以基于现有数据创建新的字段。

3 机器人应用 (Robot)

3.1 Flow 数据流

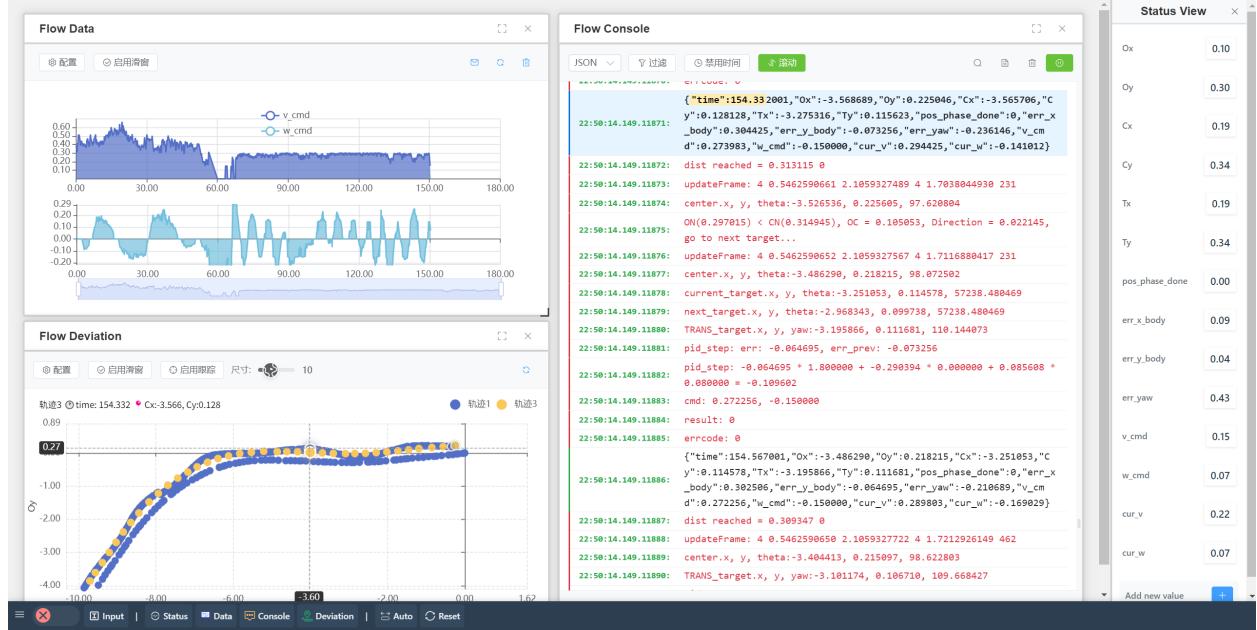


图 3-1: 机器人 Flow 数据流

Flow 考虑了机器人常见的可视化需求，例如数据的时序变化、原始日志和轨迹偏移。Flow 要求的数据采用 JSON 格式，并以换行符分隔，每行一个 JSON 对象，以下为一段示例数据：

```
{
  "time": 0.00, "camera_distance": 1.20, "camera_angle": 0.5, "pid_left_speed": 0.30, "pid_right_speed": 0.30, "motor_left_speed": 0.28, "motor_right_speed": 0.28},
  {"time": 0.05, "camera_distance": 1.18, "camera_angle": 0.4, "pid_left_speed": 0.3, "pid_right_speed": 0.3, "motor_left_speed": 0.29, "motor_right_speed": 0.29},
  {"time": 0.10, "camera_distance": 1.15, "camera_angle": 0.3, "pid_left_speed": 0.31, "pid_right_speed": 0.3, "motor_left_speed": 0.3, "motor_right_speed": 0.29}
}
```

3.1.1 日志组件 (Console)



图 3-2: 日志组件 (Console)

1. 控制区

- (a) 数据过滤: 用于过滤感兴趣的数据 (当前支持 JSON、NMEA)
- (b) 启用时间/禁用时间: 启用或禁用时间戳
- (c) 置底/滚动: 将窗口吸附在底部或使用鼠标手动滚动窗口;
- (d) 搜索: 字段搜索
- (e) 保存: 保存日志
- (f) 清除: 清空控制台
- (g) 继续/暂停: 继续或暂停呈现控制数据 (仅用于实时数据)

2. 内容区

- (a) 黑色字体为有效数据, 红色字体为其他数据、黄色字体为搜索目标。

3. 状态区

- (a) 消息总数
- (b) 有效数目
- (c) 数据接收速率

3.1.2 时序数据 (Data)



图 3-3: 时序数据 (Data)

1. 配置
 - (a) 视图布局（支持单图单 Y 轴、单图双 Y 轴、双图单 Y 轴、双图双 Y 轴）
 - (b) 字段选择（Flow 会自动从 JSON 中解析字段）
 - (c) 取色器与填充（可自定义颜色及是否填充）
2. 启用/关闭滑窗：滑窗会显示最新的 100 条数据（实时工作时，将强制启用滑窗以保证应用性能）
3. 消息格式
4. 刷新：恢复放缩状态
5. 清除：清空所有数据

3.1.3 轨迹数据 (Deviation)

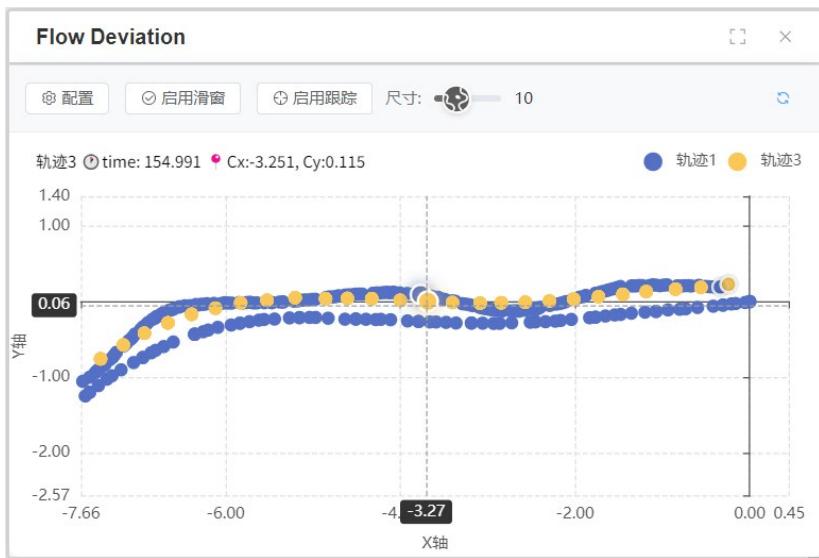


图 3-4: 轨迹数据 (Deviation)

1. 配置: 配置轨迹字段与颜色, 当前最多支持 4 条轨迹
2. 启用/关闭滑窗: 滑窗会显示最新的 100 条数据 (实时工作时, 将强制启用滑窗以保证应用性能)
3. 启用/关闭跟踪: 启用跟踪将保持最新点始终在 (0,0) 点
4. 尺寸: 调节轨迹尺寸
5. 刷新: 恢复放缩状态

4 感知模块 (PERC)

4.1 超声波避障 (Ultrasonic)



图 4-1: 超声波避障 (Ultrasonic)

4.1.1 日志组件 (Console)

同 FlowConsole。

4.1.2 时序数据 (Data)

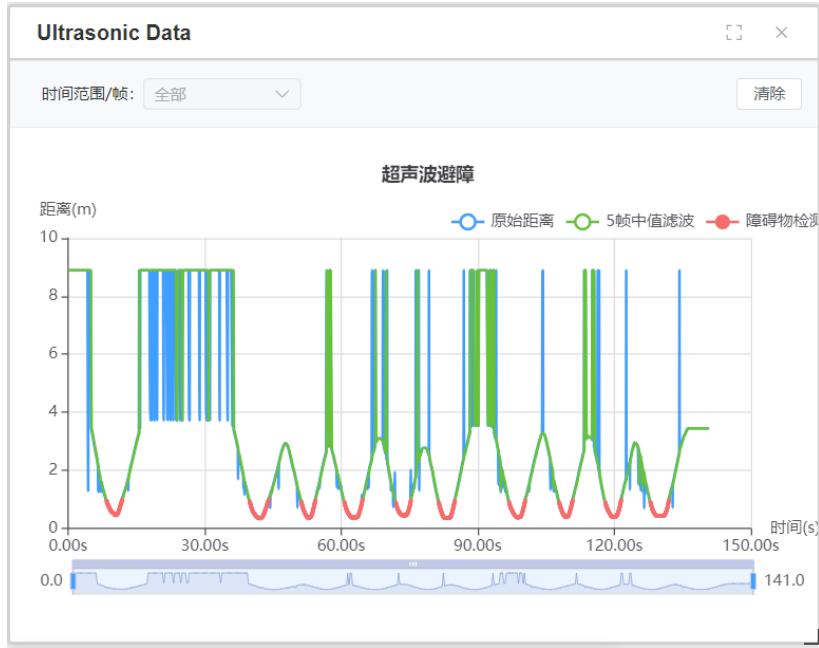


图 4-2: 时序数据 (Data)

由于 Ultrasonic 内置算法，因此要求原始数据相对固定，具体的要求必须为 JSON 格式，同时包含 ultrasonic 字段：

```
{"time": 140.449999999977, "ultrasonic": 3.449}  
 {"time": 140.499999999977, "ultrasonic": 3.449}  
 {"time": 140.549999999977, "ultrasonic": 3.453}
```

控制按钮：

1. 时间范围
2. 清除

5 定位模块 (POS)

5.1 GNSS

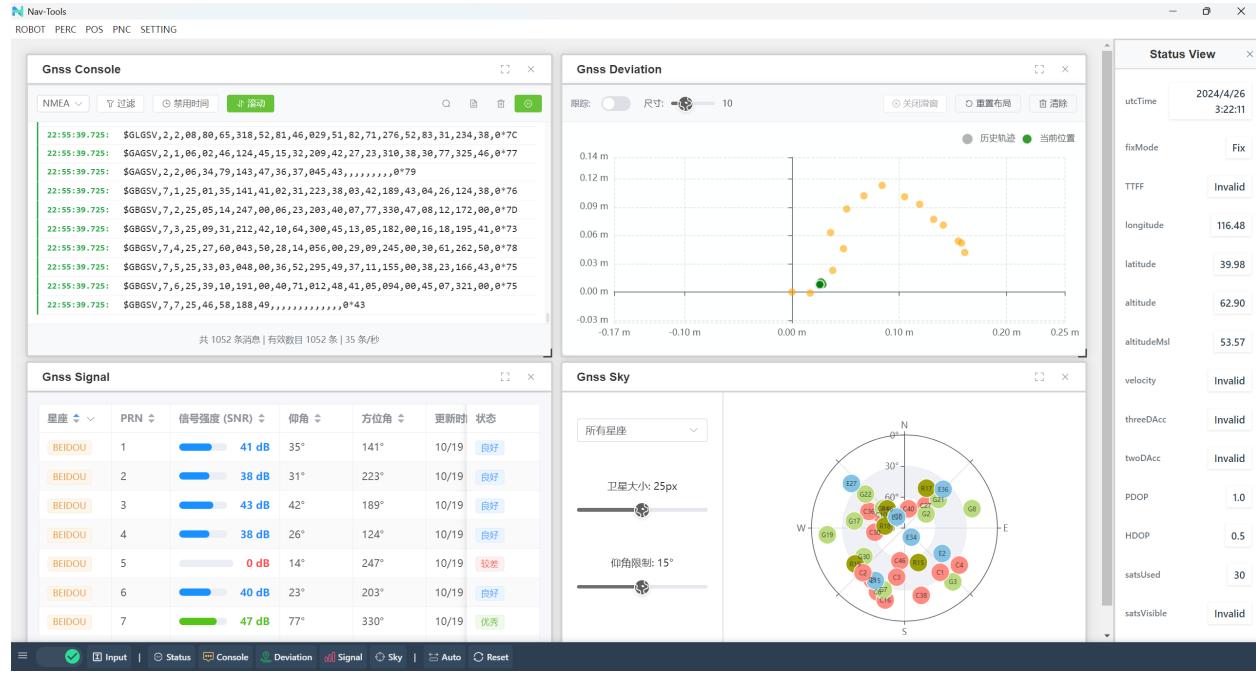


图 5-1: GNSS 模块

基本实现了常见的 GNSS 上位机功能。

5.1.1 日志组件 (Console)

同 FlowConsole，不过这里视 NMEA 格式为有效数据。

5.1.2 轨迹数据 (Deviation)

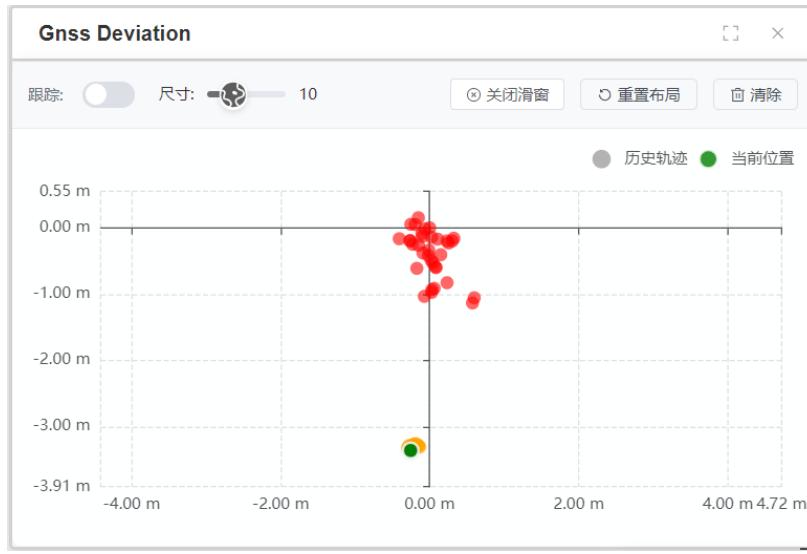


图 5-2: 轨迹数据 (Deviation)

轨迹组件基本与 Flow Deviation 相同，不过与常规 GNSS 上位机相同，它只包含一条轨迹，另外红色表示单点解、橙色表示浮点解、绿色表示固定解。

5.1.3 信号数据 (Signal)



图 5-3: 信号数据 (Signal)

主要呈现卫星的信号强度信息，相较于一些 GNSS 上位机中常用的柱状图，这里使用了表格信息。

5.1.4 星空视图 (Sky)



图 5-4: 星空视图 (Sky)

呈现卫星分布清空，可以调节星座、卫星图标大小和仰角限制。

6 规控模块 (PNC)

6.1 Motor

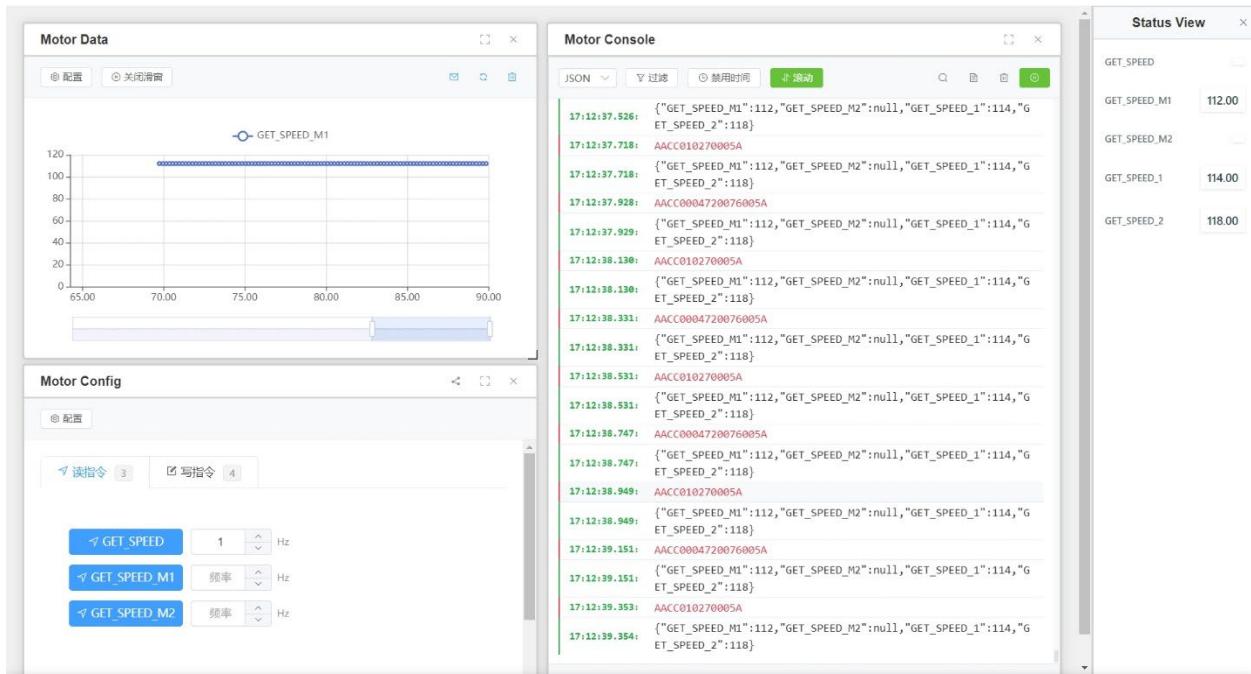


图 6-1: 电机驱动模块 (Motor)

6.1.1 配置组件 (Config)

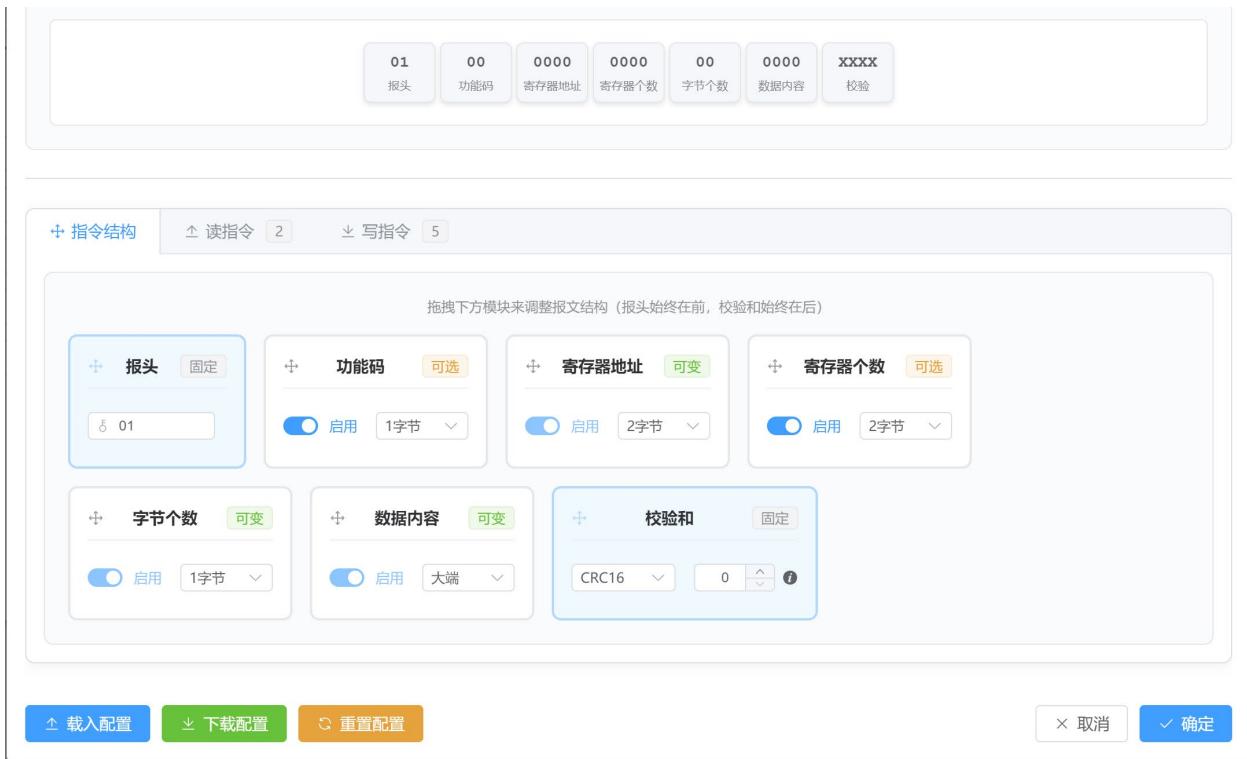


图 6-2: 电机驱动指令配置

Motor 模块提供了灵活的指令配置项，包括指令结构和命令主体，并以读指令和写指令对命令进行了划分。设置好的指令配置可以将其导出或载入。



图 6-3: 命令渲染界面

界面主体将会根据配置项中的命令来生成相应的按钮，用于实现与下位的电机驱动器进行交互。

6.1.2 日志组件（Console）



图 6-4: 指令与 Console 的同步

MotorConsole 复用了 FlowConsole 组件，不过这里增加了 16 进制字节数组到 JSON 的转换，以便数据可视化。另外 MotorConfig 中下发的指令也会同步到 Console 组件中。

6.1.3 时序数据（Data）

同 FlowData。

6.2 FollowSim

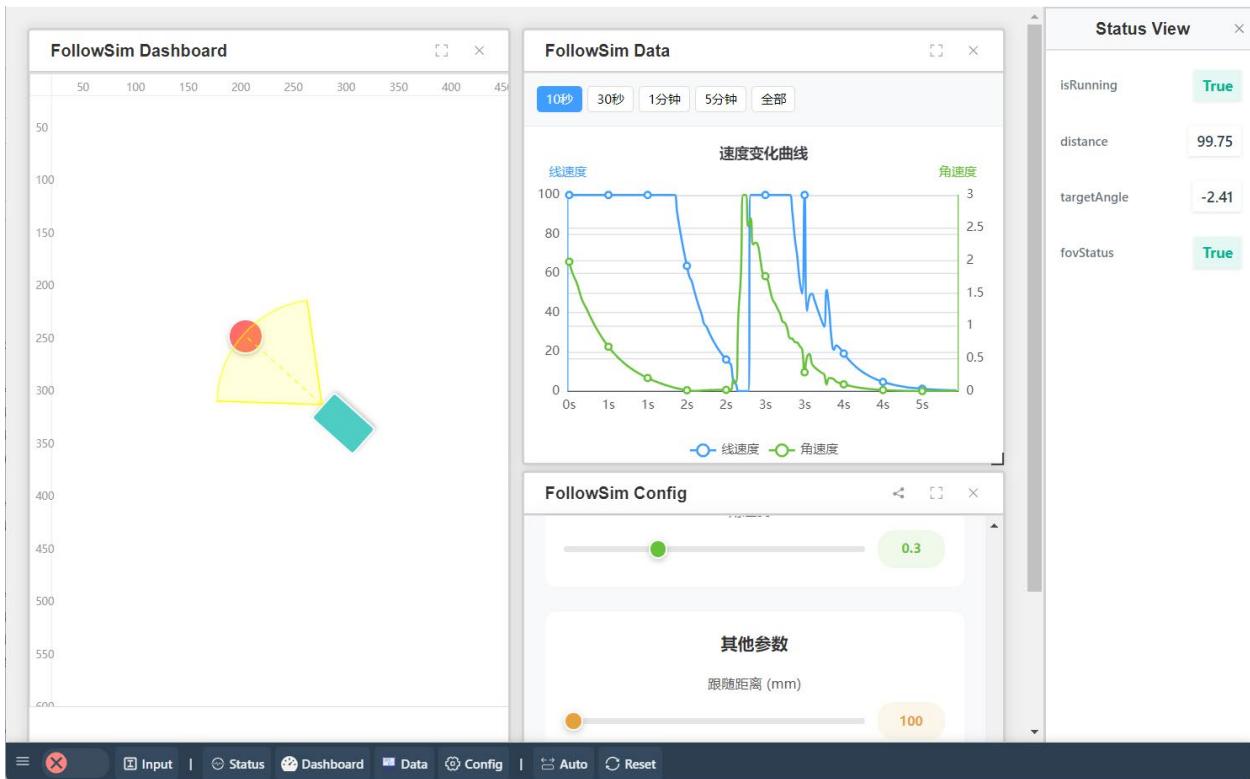


图 6-5: PID 跟随仿真 (FollowSim)

6.2.1 仪表盘组件 (Dashboard)

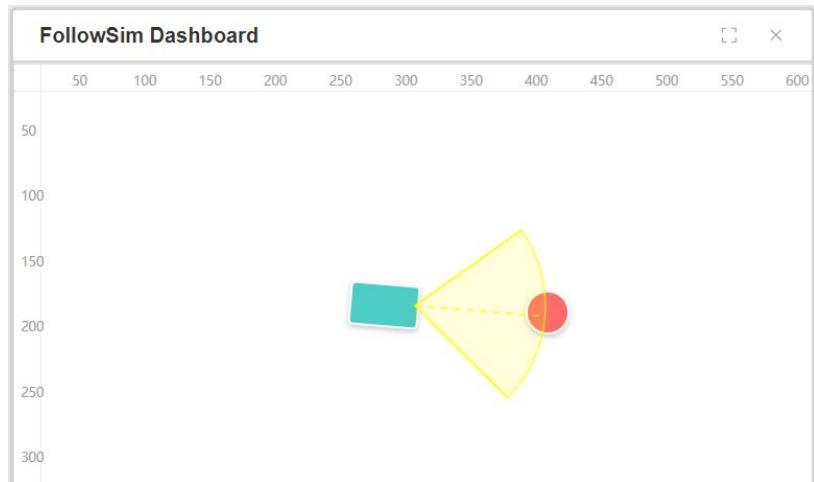


图 6-6: 仪表盘组件 (Dashboard)

呈现 PID 跟随的二维动画。

6.2.2 时序数据 (Data)

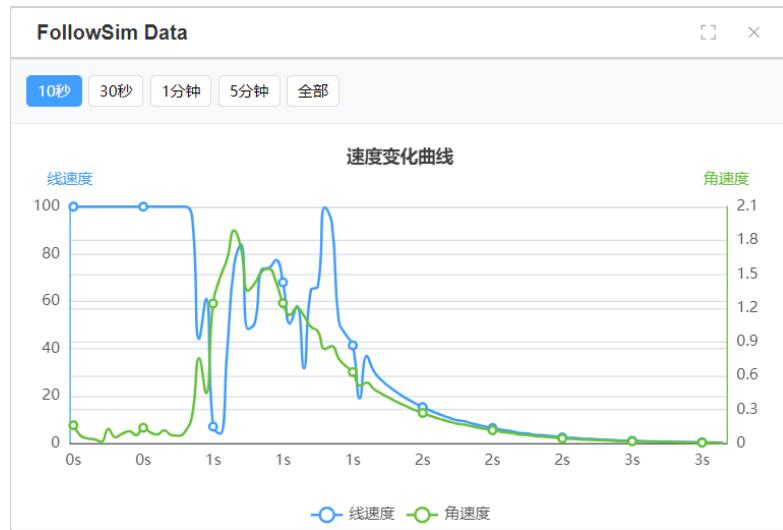


图 6-7: 时序数据组件 (Data)

主要绘制线速度和角速度的变化。

6.2.3 控制面板 (Config)



图 6-8: 控制面板 (Config)

主要包含线速度 PID、角速度 PID 和其他相关参数。