

| dll模型扩展外部数据输入接口系列实验

| 1. 实验目的

本系列实验用于说明如何向 DLL 模型注入扩展外部数据，涵盖

`inSILIntsFloats`、`inDoubCtrls` 和 `inSIL28d` 三类典型输入接口，帮助用户完成故障注入、参数传递、自定义控制量输入和数据链路验证。

| 2. 实验要求

- 软件要求：Windows 10及以上版本；RflySim工具链；MATLAB/Simulink；Python 运行环境 [\[1\]](#)。
- 硬件要求：笔记本/台式电脑1台；若需与飞控、外部控制器或日志系统联调，请准备相应通信环境 [\[2\]](#)。

| 3. 实验地址

例程目录：[\[安装目录\]\RflySimAPIs\4.RflySimModel\0.ApiExps\11.inSILAPI](#)

- [../././API.pdf](#)：模块接口说明。

| 4. 实验内容或步骤

| 4.1 建议学习顺序

1. 先阅读 [1.InSILIntsFloats](#)，理解整数标志位与浮点参数混合注入方式。
2. 再阅读 [2.inDoubCtrls](#) 和 [3.inSIL28d](#)，理解 28 维双精度数据的两种输入链路。
3. 完成输入注入后，再结合 `outCopterData`、`ExtCtrlAPI` 或综合模型实验做端到端联调。

4.2 子实验概览

目录	主题	说明
1.inSILIntsFloats	dll模型外部数据输入接口inSILIntsFloats	适合做标志位加参数位混合注入，如故障触发和离散事件控制。
2.inDoubCtrls	dll模型外部输入接口inDoubCtrls	面向 28 维双精度数据输入，常用于控制量或自定义状态量注入。
3.inSIL28d	dll模型外部输入接口inSIL28d	面向模型侧 28 维双精度扩展接口的直接输入与转发验证。

4.3 典型使用方式

- 当 DLL 模型需要同时接收离散标志位和连续参数时，优先使用 `inSILIntsFloats`。
- 当模型只需一组连续双精度输入时，可根据模型接口设计选择 `inDoubCtrls` 或 `inSIL28d`。
- 对应的 Python 发送函数通常为 `sendSILIntFloat`、`sendSILIntDouble` 和 `sendInDoubCtrls`，接口验证完成后建议配合日志或 `outCopterData` 做回读确认。

5. 关键知识点

- `inSILIntsFloats` 适合做“标志位 + 参数位”混合输入，例如外部故障注入和事件触发。
- `inDoubCtrls` 与 `inSIL28d` 都是 28 维双精度输入接口，适合连续控制量、参数或扩展状态输入。
- 使用扩展输入接口前，通常需要在 Simulink 模型中显式增加对应 `Inport`，重新生成 DLL 后才能被 CopterSim 正确识别。
- 数据注入链路往往需要和 `outCopterData`、UDP 30100/30101 系列端口一起验证，才能确认输入是否真正到达模型内部。

6. 参考资料

1. [RflySim官方文档](#)
2. [模块接口说明 API.pdf](#)
3. [dll模型自定义输出接口outCopterData](#)
4. [dll模型外部控制UDP通信端口](#)

7. 常见问题

Q1: 应该选 inSILIntsFloats、inDoubCtrls 还是 inSIL28d?

A1: 如果你既需要离散标志位又需要连续参数, 优先选 `inSILIntsFloats` ; 如果只需要 28 维连续双精度输入, 则根据模型中预留的接口名称选择 `inDoubCtrls` 或 `inSIL28d` 。

Q2: 为什么发送了数据, 但模型里没有生效?

A2: 通常需要检查三点: 一是 Simulink 模型是否已经添加了对应名称和维度的输入端口; 二是 DLL 是否在修改后重新生成; 三是 UDP 端口和数据打包格式是否与接口协议一致。

-
1. <https://rflysim.com/> ↩
 2. 推荐配置请见: <https://rflysim.com/doc/zh/HowToInstall.pdf> ↩