

1. 实验名称及目的

1.1. 实验名称

dll 模型外部通信实验之获取平台 rfly_px4 uORB 消息

1.2. 实验目的

当订阅了 rfly_px4 uORB 消息，并使用平台最大模板进行硬件在环仿真时，可以通过监听 UDP40101 系列端口接收 rfly_px4 消息。

1.3. 关键知识点

详细实验原理见：[\4.RflySimModel\0.ApiExps\3.ExtCtrlAPI\intro.pdf](#)

无论应用程序使用 TCP（传输控制协议）传数据，还是 UDP（User Datagram Protocol 用户数据报协议）传数据，都要监听一个端口，这个端口用来区分应用程序，故端口不能冲突。所以，无论是 TCP 还是 UDP 包头里面都应该有端口号，根据端口号，将数据交给相应的应用程序。

源端口号（16位）	目的端口号（16位）
UDP长度（16位）	UDP校验和（16位）
实际数据内容	

每个 UDP 报文分为 UDP 报头和 UDP 数据区两部分。报头由 4 个 16 位长（2 字节*4=8 字节）字段组成，分别说明该报文的源端口、目的端口、报文长度和校验值。

- 源端口号(Source Port):这个字段占据 UDP 报文头的前 16 位，通常包含发送数据报的应用程序所使用的 UDP 端口。接收端的应用程序利用这个字段的值作为发送响应的目的地址。这个字段是可选项，有时不会设置源端口号。没有源端口号就默认为 0，通常用于不需要返回消息的通信中。

- 目标端口号(Destination Port): 表示接收端端口，字段长为 16 位

- 长度(Length): 该字段占据 16 位，表示 UDP 数据报长度，包含 UDP 报文头和 UDP 数据长度。因为 UDP 报文头长度是 8 个字节，所以这个值最小为 8，最大长度为 65535 字节。

- 校验和(Checksum): UDP 使用校验和来保证数据安全性，UDP 的校验和也提供了差错检测功能，差错检测用于校验报文段从源到目标主机的过程中，数据的完整性是否发生了改变。

2. 实验效果

以最大模板启动硬件在环仿真，待仿真初始化完成（GPS 3D fixed）后，监听 UDP40101 系列端口接收 rfly_px4 消息。

3. 文件目录

例程目录：[\[安装目录\]\RflySimAPIs\4.RflySimModel\0.ApiExps\3.ExtCtrlAPI\3.ExtCtrlAPI-UDP40100\](#)

文件夹/文件名称		说明
DLLModelTemp	Exp2_MaxModelTemp.dll	由最大模型生成的动态链接库
	Exp2_MaxModelTemp.slx	最大模型源程序
	Exp2_MaxModelTempHIT L.bat	最大模型硬件在环仿真启动脚本
PX4ExtMsgReceiver.slx	\	rfly_px4 uORB 消息文件
PX4ExtMsgSender.slx	\	外部通信接口程序

4. 运行环境

序号	软件要求	硬件要求	
		名称	数量
1	Windows 10 及以上版本	笔记本/台式电脑 ^①	1
2	RflySim 工具链	Pixhawk 6X 或其它飞控 ^②	1
3	MATLAB 2017b 及以上 ^③	数据线	1

①：推荐配置请见：<https://rflysim.com/>

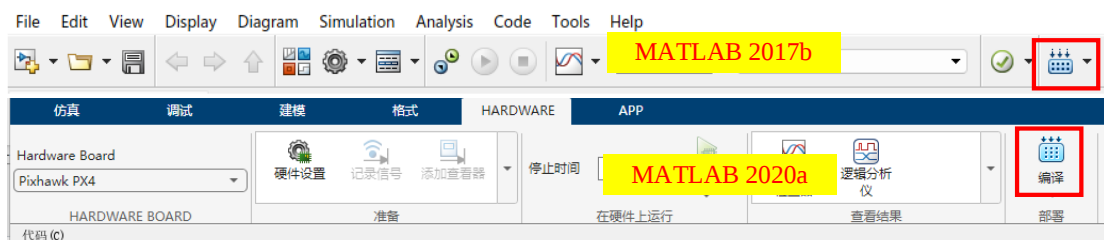
②：若使用 Pixhawk 6X 飞控，平台安装时的编译命令为：px4_fmu-v6x_default，推荐 PX4 固件版本为：1.12.3。其他配套飞控及编译命令请见：<https://rflysim.com/doc/zh/1/Hardware.html>。

5. 实验步骤

5.1. 固件烧录

Step 1: 编译

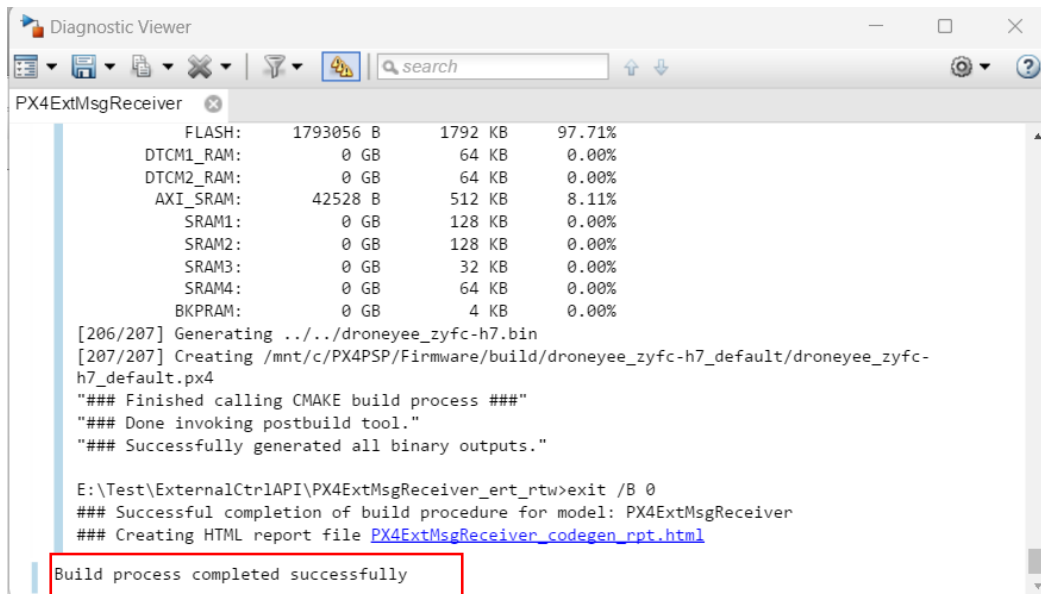
打开 MATLAB 软件，在 MATLAB 中打开 PX4ExtMsgReceive.slx 文件，在 Simulink 中，点击编译命令。



注意：编译 PX4ExtMsgReceive.slx 文件与下面的编译 Exp2_MaxModelTemp.slx 文件不同，编译 PX4ExtMsgReceive.slx 文件需要设置 Pixhawk 硬件，本文件中已设置。

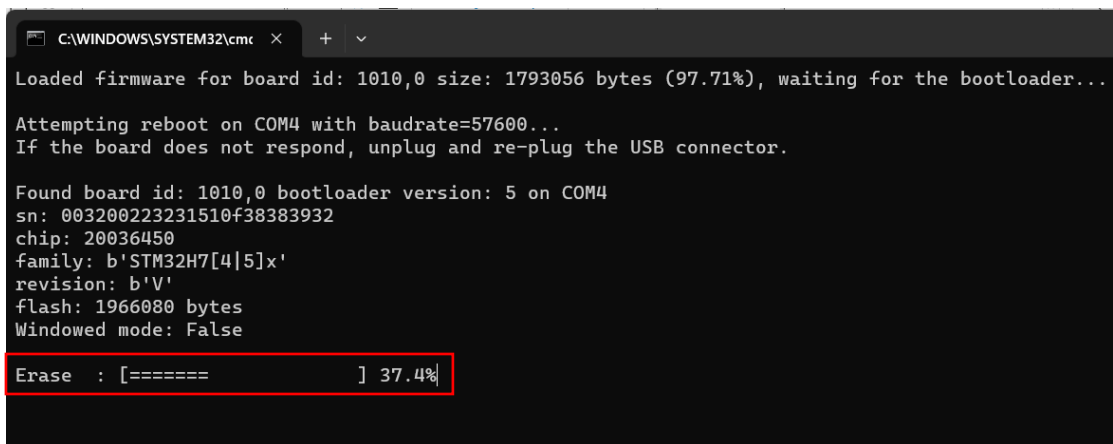
Step 2: 等待编译成功

在 Simulink 的下方点击 View diagnostics 指令，即可弹出诊断对话框，可查看编译过程。在诊断框中弹出 Build process completed successfully，即表示编译成功。



Step 3: 烧录固件

点击 Simulink 的 Code 选项，在下拉条目中选择 PX4 PSP: Upload code to Px4FMU，弹出命令框后将飞控通过 USB 线连接至电脑，开始固件烧录。



当进度条到达 100%时，表明固件烧录完毕。

```
C:\WINDOWS\SYSTEM32\cmd x + v
Loaded firmware for board id: 1010,0 size: 1793056 bytes (97.71%), waiting for the bootloader...

Attempting reboot on COM4 with baudrate=57600...
If the board does not respond, unplug and re-plug the USB connector.

Found board id: 1010,0 bootloader version: 5 on COM4
sn: 003200223231510f38383932
chip: 20036450
family: b'STM32H7[4]5x'
revision: b'V'
flash: 1966080 bytes
Windowed mode: False

Erase : [=====] 100.0%
Program: [=====] 100.0%
Verify : [=====] 100.0%
Rebooting. Elapsed Time 75.586
```

5.2. DLL 模型生成

Step 1: 修改模型并编译

打开 DLLModelTemp 文件夹，在 MATLAB 中打开 Exp2_MaxModelTemp.slx 文件，将 simulink 文件中的 ExtToPx4 接口数据输入修改为自定义 16 维数据。



double ExtToUE4[16]; // This signal will be sent to UE4 as the 9 to 26 of actuator's inputs. Besides, this value can be shown in UE4's D mode, so you can observe the value of the model through UE4.

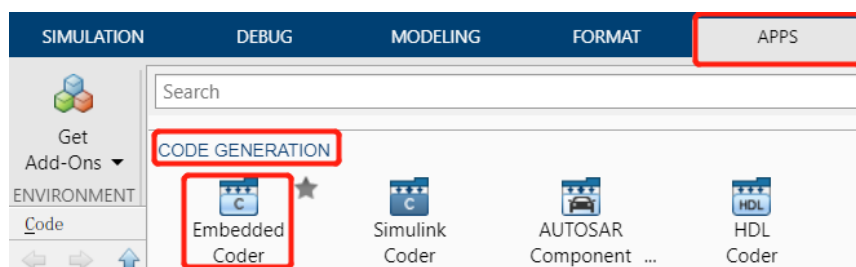
float ExtToPX4[16]; // this value will be sent to PX4 with uORB msg rfly_ext. So you can transfer some sensor data to you generated PX4 controller.

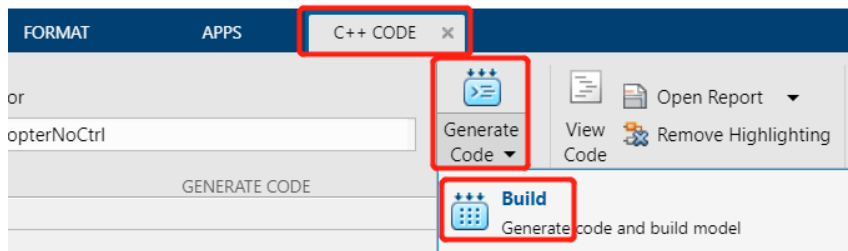
修改完成后，在 Simulink 中保存，后进行编译。

对于 MATLAB 2019a 及之前版本，工具栏样式见下图，直接点击它的编译按钮“Build”即可。



对于 2019b 及之后版本，点击 APPS - CODE GENERATION - Embedded Coder 才能弹出代码生成工具栏，在其中如下图所示点击“C++CODE”-“Generate Code”-“Build”按钮就能编译生成代码。





在 Simulink 的下方点击 View diagnostics 指令，即可弹出诊断对话框，可查看编译过程。
在诊断框中弹出 Build process completed successfully，即表示编译成功

```

* Build @ 4 @ 3 Clear
11:28 AM Elapsed: 21 sec

*** Starting build procedure for model: Exp2_MaxModelTemp
*** Generating code and artifacts to 'Model specific' folder structure

* Code Generation @ 2 @ 3
Elapsed: 15 sec

*** Generating code into build folder: E:\Test\ExternalCtrlAPI\demo3\DLLModelTemp\Exp2_MaxModelTemp_ert_rtw
Output port 1 of 'Exp2_MaxModelTemp/ExtToP4' is not connected.
Component Simulink|Category: Block warning
Unconnected output line found on 'Exp2_MaxModelTemp/SensorModel/HILGPSModel1/Bus_Creator1' (output port: 1)
Component Simulink|Category: Block warning
The output(s) read after the base-rate model step reflects intervening minor time steps. To observe data that is a snapshot of output(s) at major time steps, do one of the following:
* Place a Zero-Order Hold block before the continuous output port 'Exp2_MaxModelTemp/MavLinkSensor'.
* Clear Single output/update_function, and read model output values after model_output call and before model_update call.
For more details, see "Single output/update_function" in the Simulink documentation.
Component Simulink|Category: Block warning

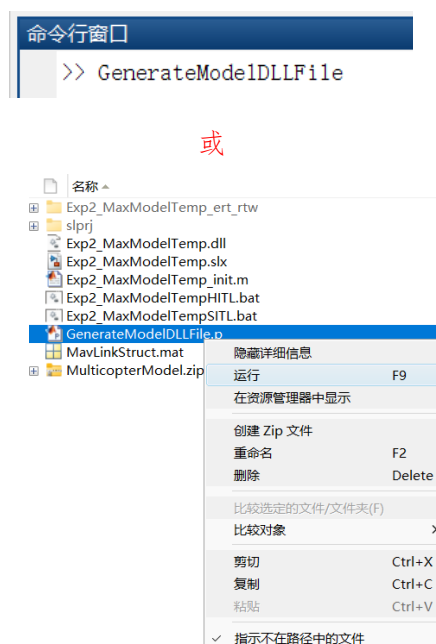
*** Invoking Target Language Compiler on Exp2_MaxModelTemp.rtw
*** Using System Target File: D:\MATLAB\R2017b\rtw\c\vert\ert.tlc
*** Loading TLC function libraries
*** Initial pass through model to cache user defined code
*** Caching model source code
.....
*** Writing header file Exp2_MaxModelTemp.h
*** Writing source file Exp2_MaxModelTemp.cpp
*** Writing header file rtwtypes.h
*** Writing source file ert_main.cpp
..
*** TLC code generation complete.
.....[警告: 名称不存在或不是目录: E:\OneDrive\RFlySimSource\RFlySimGit\rflysimapis_beta\RFlySimAPIs\OtherVehicleTypes\DLLModelTemp]
*** Successful completion of code generation for model: Exp2_MaxModelTemp

Build process completed successfully

```

Step 2: 生成 DLL 文件

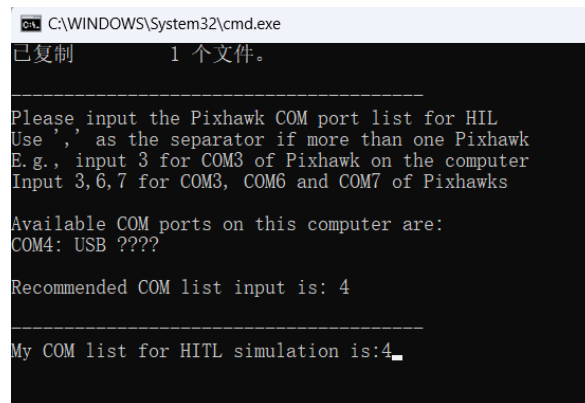
右键运行 GenerateModelDLLFile.p 文件或在命令行窗口中输入 GenerateModelDLLFile 后回车，得到最大模型动态链接库 Exp2_MaxModelTemp.dll。



5.3. 硬件在环仿真

Step 1: 开启仿真

右键以管理员身份运行 Exp2_MaxModelTempHITL.bat，输入飞控和电脑连接的端口号，回车启动硬件在环仿真。



```
C:\WINDOWS\System32\cmd.exe
已复制 1 个文件。

-----
Please input the Pixhawk COM port list for HITL
Use ',' as the separator if more than one Pixhawk
E.g., input 3 for COM3 of Pixhawk on the computer
Input 3,6,7 for COM3, COM6 and COM7 of Pixhawks

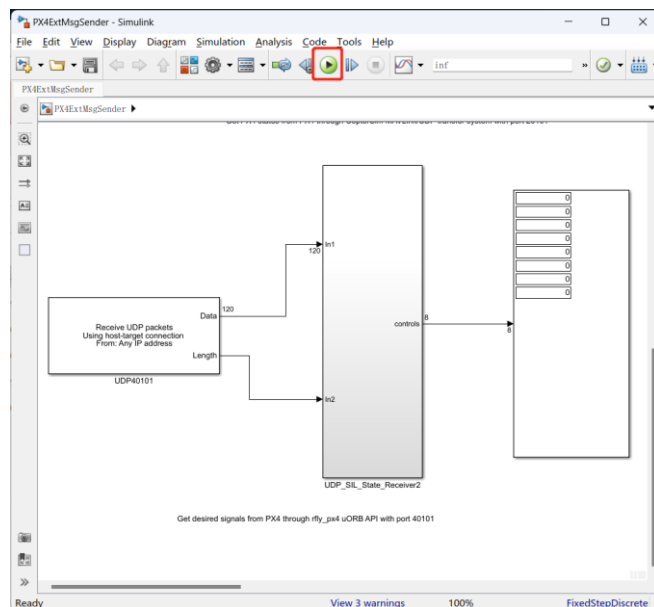
Available COM ports on this computer are:
COM4: USB ???

Recommended COM list input is: 4

-----
My COM list for HITL simulation is:4_
```

Step 2: 运行 Simulink 模型

返回上一级文件夹，以 MATLAB 打开 PX4ExtMsgSender.slx，并运行。



Step 3: 观察结果

正常现象为，PX4ExtMsgSender.slx 运行后，可以在 UDP_SIL_State_Receiver2 模块的 Display 实时观察到 rfly_px4 消息，该消息为 8 维。

6. 参考资料

- [1]. 自动代码生成外部通信接口 [../PX4PSP/RflySimAPIs/5.RflySimFlyCtrl/API.pdf](http://PX4PSP/RflySimAPIs/5.RflySimFlyCtrl/API.pdf)
- [2]. uORB Read and Write—uORB 消息读取和写入库 [../PX4PSP/RflySimAPIs/5.RflySimFlyCtrl/API.pdf](http://PX4PSP/RflySimAPIs/5.RflySimFlyCtrl/API.pdf)

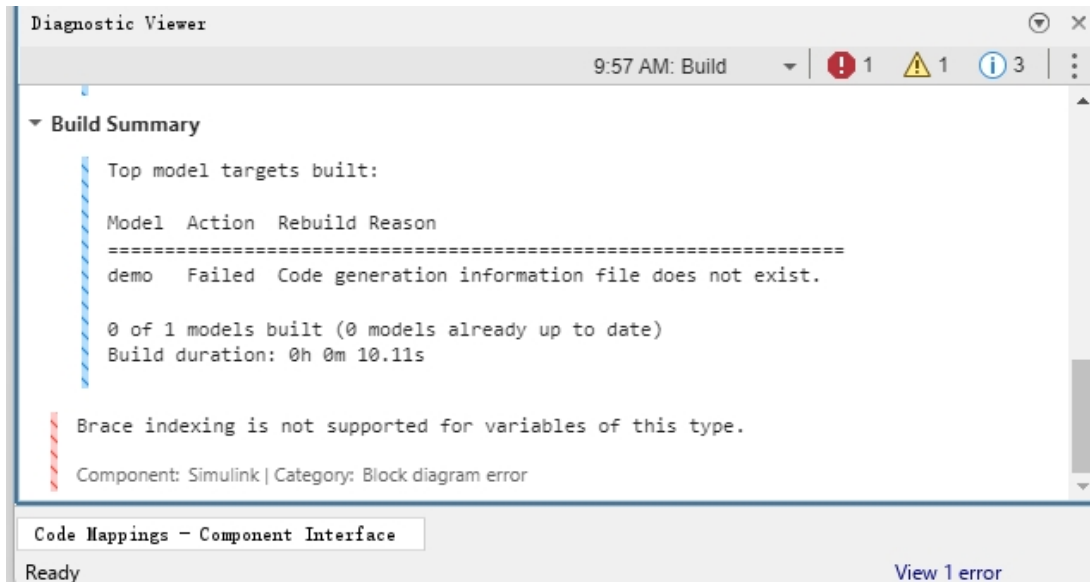
[3]. DLL/SO 模型与通信接口 [..\PX4PSP\RflySimAPIs\4.RflySimModel\API.pdf](#)

[4]. 外部控制接口 [..\PX4PSP\RflySimAPIs\4.RflySimModel\API.pdf](#)

[5].

7. 常见问题

Q1: 未正确安装 visual studio c++编译环境并配置 mex, 导致 Simulink 文件编译失败



A1: 首先将低于当前 MATLAB 版本的 Visual Studio C++编译环境安装到 VS 默认安装目录, 然后在 MATLAB 的命令行窗口中输入指令 “mex -setup”, 一般来说会自动识别并安装上支持的编译器, 命令行显示 “MEX 配置使用 ‘Microsoft Visual C++ 2017’ 以进行编译” 的字样说明安装正确。详细环境配置参考” [RflySim 平台安装目录]\RflySimAPIs\4.RflySimModel\API.pdf “中的环境配置。



Q2: 编译报错，无法加载库文件



A2: 这可能是由于安装平台时 PX4PSP 工具箱未更新到最新版，更新 RflySim 安装包后按照如下配置重新安装平台即可

Toolbox one-key installation script: RflySimA...

(1) Software package installation directory
C:\PX4PSP

(2) PX4 firmware compiling command: firmware versions <= PX4-1.8 use format px4fmu-v3_default; >= PX4-1.9 use format px4_fmu-v3_default
px4_fmu-v6c_default

(3) PX4 firmware version (1: PX4-1.7.3, ... , 6: PX4-1.12.3, 7: PX4-1.13.2, 8: PX4-1.14.4, 9: PX4-1.15.0)
9

(4) PX4 firmware compiling toolchain (1: WinWSL[suitable for all versions], 2: Msys2[suitable for <= PX4-1.8], 3: Cygwin[for >=PX4-1.8])
1

(5) Whether to reinstall PSP toolbox (yes to reinstall and no to remain current installation)
yes

(6) Whether to reinstall the dependent software packages (CopterSim, QGroundControl, CopterSim, etc. About 5 minites)
no

(7) Whether to reinstall the selected compiling toolchain (yes to reinstall and no to remain unchanged, about 5 minites)
no

(8) Whether to reinstall the selected PX4 firmware source code (yes to reinstall and no to remain unchanged, about 5 minites)
no

(9) Whether to pre-compile the selected firmware with the selected command (yes to compile and no to remain unchanged, about 5 minites)
no

(10) Whether to block the actuator outputs in the PX4 firmware code ("yes" to use Simulink controller, "no" to use PX4 official controller)
no

OK Cancel