

Tutorial/documentação das comunicações feitas para SIL(como implementar as comunicações).

1. INTRODUÇÃO

Neste trabalho é desenvolvido a comunicação para envio e recebimento de dados entre o simulador de vôo X-Plane e o *software* MatLab/Simulink.

2. MATLAB

O MATLAB (MATrix LABoratory) é um *software* desenvolvido pela companhia *The MathWorks* para cálculo científico que permite a realização de algoritmos numéricos sobre matrizes. É possível a criação e manipulação das matrizes sem dimensionamento prévio, onde, a manipulação é feita de forma interativa.

O *software* dispõe de diversas extensões (*toolboxes*), sendo o mais conhecido o Simulink, uma interface de diagrama de blocos amplamente utilizada para modelamento, simulação e análise de sistemas dinâmicos.

A versão utilizada neste tutorial é a 8.1.0.604 (R2013a).

3. X-PLANE

O X-Plane é um simulador de voo desenvolvido pela empresa *Laminar Research*. Sua biblioteca conta com diversas modelos de aeronaves e aeroportos. O *software* ainda dispõe de ferramentas para remodelar aeronaves, cenários e até mesmo as condições climáticas.

O X-Plane também oferece ao usuário a possibilidade de construir seus próprios modelos de aeronave. A qualidade da simulação está diretamente ligada ao detalhamento de informações utilizado na criação destes modelos.

Devido aos seus modelos de aeronaves extremamente precisos, o *software* pode produzir simulações com alto grau de semelhança com um voo real, tornando-se uma ferramenta útil para prever e testar o desempenho de uma aeronave de acordo com suas características.

A versão utilizada neste tutorial é a 10.02r1. A Figura 1 mostra a simulação de uma aeronave Boeing 747-400 em voo.



Figura 1 – Boeing 747-400 no X-Plane

4. A COMUNICAÇÃO ENTRE SIMULINK E X-PLANE

O X-Plane tem a capacidade de trocar informações com dispositivos externos, uma das possibilidades é implementar esta comunicação pelo protocolo UDP para enviar e receber dados.

O MatLab/Simulink em sua biblioteca apresenta blocos para realizar uma comunicação UDP. Para a decodificação do pacote UDP recebido é possível criar trechos de código em linguagem C.

Os pontos de acesso do UDP comumente chamados por “Portas de Protocolos” ou “portas”, onde em cada transmissão de dados UDP identifica-se o endereço de IP e o número de porta do destino e da fonte de mensagem.

5. CONFIGURAÇÃO X-PLANE

Foi feita a comunicação do simulador com o Simulink utilizando dois computadores e utilizando apenas um computador. O tutorial da configuração de rede do X-Plane para os dois casos é apresentado mais abaixo.

5.1 REDE ENTRE DOIS COMPUTADORES

Para efetivar a conexão entre o simulador X-Plane e o MatLab/Simulink que se encontram em computadores diferentes é necessário criar um rede entre os dois computadores utilizando suas portas Ethernet e atribuindo um endereço IP específico para cada computador.

A Figura 2 ilustra a rede de área local a ser implementada para fazer a comunicação entre os dois computadores.

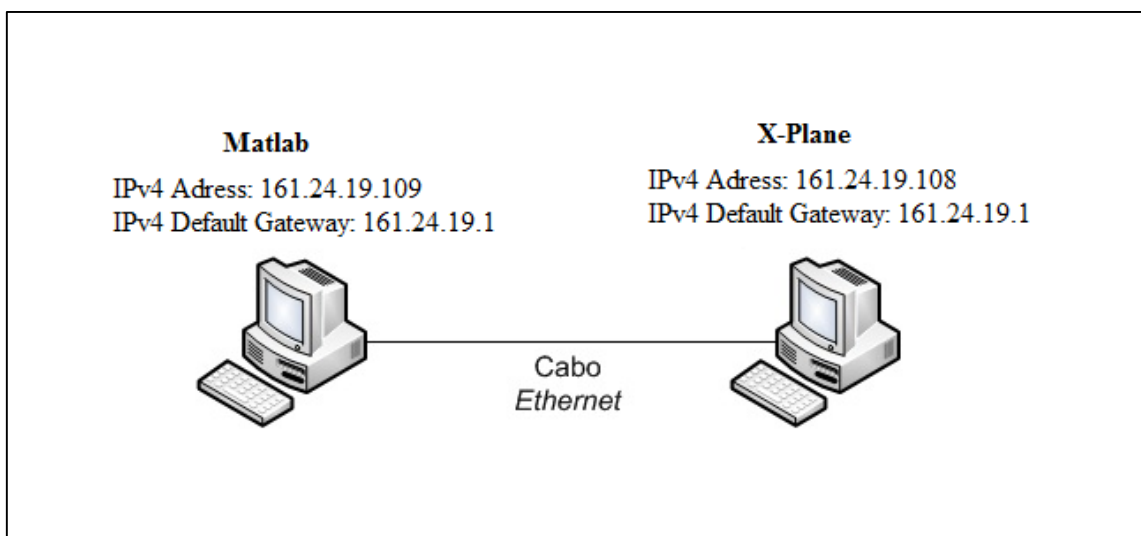


Figura 2 - Rede entre dois computadores

Neste tipo de configuração é necessário utilizar quatro pontos de acesso, dois para o Simulink (P1 e P3) e dois para o X-Plane(P2 e P4), conforme a Figura 3.

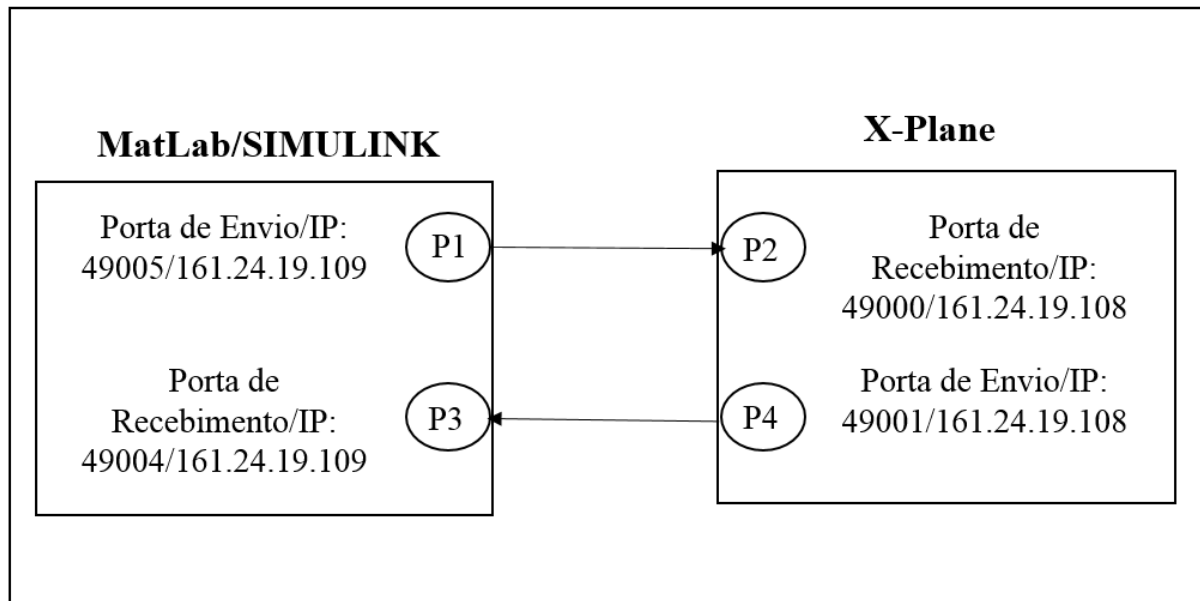


Figura 3 - Configuração dos pontos de acesso

A configuração dos pontos de acesso no X-Plane pode ser acessada através do menu principal, *Settings / Net Connection / Data*. As configurações do endereço de IP e da porta do dispositivo que envia os dados para o X-Plane, assim, como o *default* das portas que o X-Plane envia e recebe dados são ilustradas pela Figura 4.

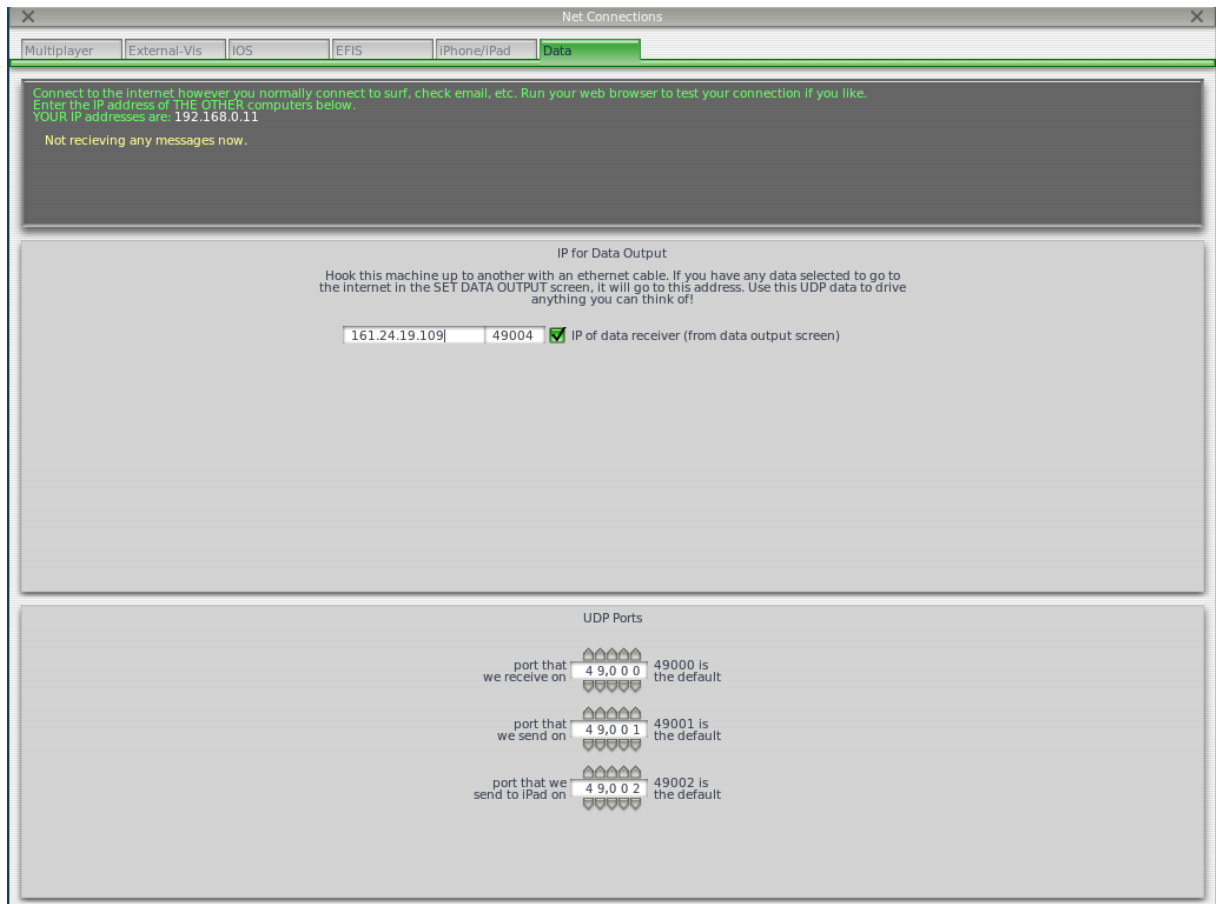


Figura 4 - Interface para configurar Endereço de IP e porta de Saída/Entrada do X-Plane

5.2 REDE ENTRE UM COMPUTADOR

Nesta plataforma de comunicação estabelecida entre o X-Plane e o Matlab são executadas em único computador utilizando o IP referente ao *localhost* da máquina. *Localhost* é um dispositivo *loopback* ao qual é atribuído endereço de IP “127.0.0.1” no IPv4 para testar a comunicação consigo mesmo.

Neste tipo de configuração é necessário utilizar quatro pontos de acesso, dois para o Simulink (P1 e P3) e dois para o X-Plane (P2 e P4), e um único endereço de IP, conforme a Figura 5.

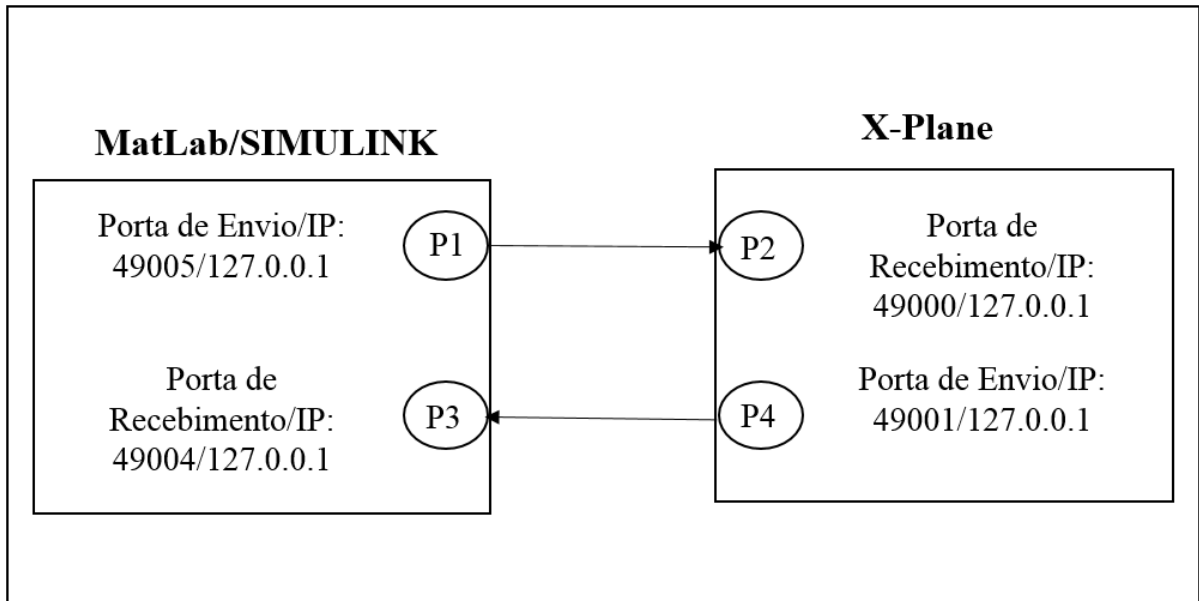


Figura 5 - Configuração dos pontos de acesso

5.3 SELECIONANDO PARÂMETROS NO X-PLANE

A interface de seleção de parâmetros no X-Plane pode ser acessada do menu principal, “*Settings / Data Input&Output/ Data Set*” e permite fazer a seleção dos parâmetros que serão enviados pela porta 49001 do simulador. O X-Plane fornece quatro formas de visualização de dados:

- Protocolo UDP
- Gravação de dados em disco
- Geração de gráficos
- Na tela do cockpit da aeronave em simulação

Para a seleção da opção desejada, basta habilitar o *checkbox* correspondente ao parâmetro e a forma de visualização de dados. A Figura 6 ilustra uma seleção de um parâmetros para ser enviado pela rede ethernet da interface do X-Plane.

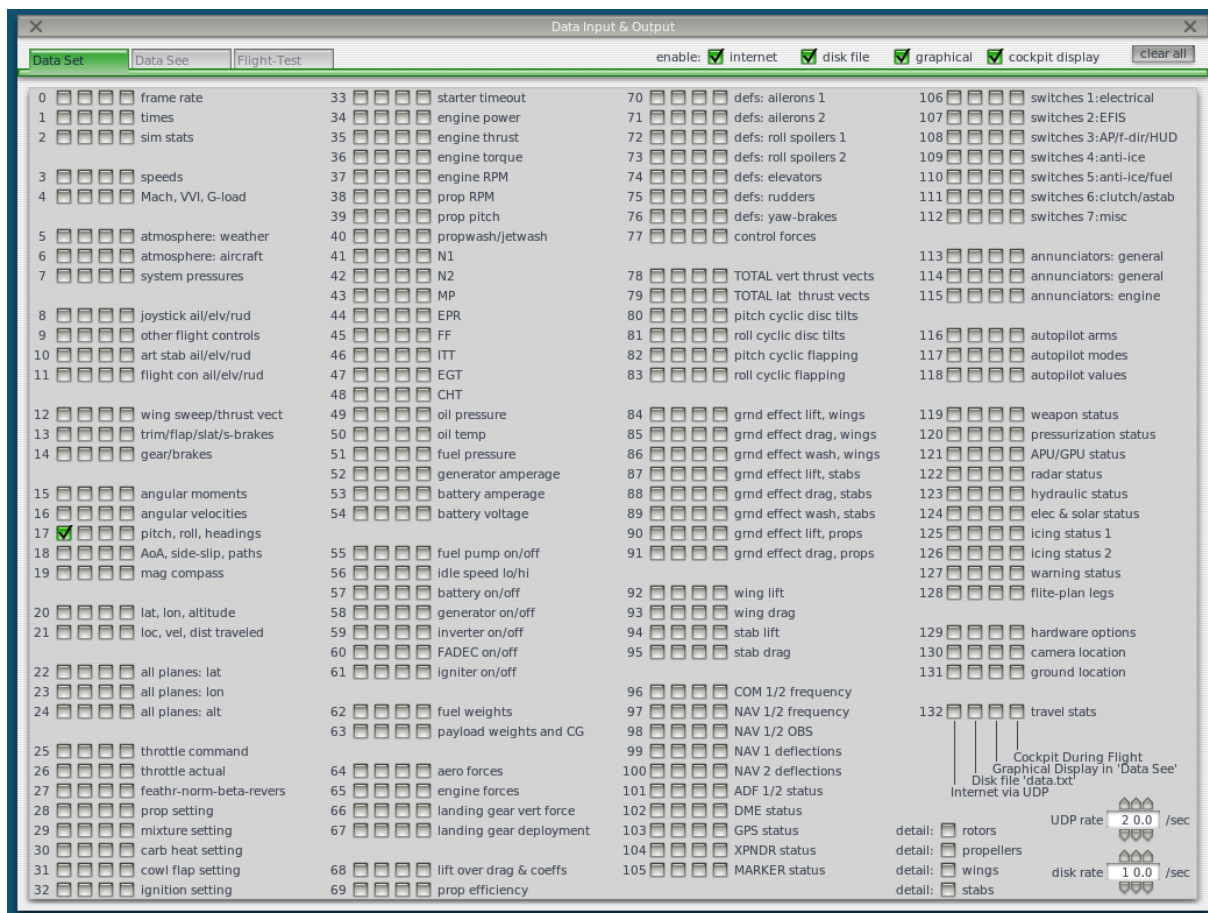


Figura 6 - Tela para seleção de dados de entrada e saída no X-Plane

6. PACOTE DE DADOS

Basicamente o pacote de dados para um único identificador é composto de uma sequência de 41 bytes. Marcando o identificador 17 (*pitch, roll, headings*) da interface “*Settings / Data Input&Output/ Data Set*” apresentada na seção anterior, a organização do pacote de dados de entrada e saída do X-Plane obedece o padrão mostrado na Figura 7.

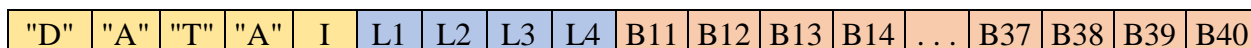


Figura 7 - Pacote de comunicação para a comunicação entre o X-Plane

Onde os 4 primeiros bytes recebem os valores {68,65,84,65} referente aos caracteres “DATA” e indicam que este é um pacote de dados. O byte 5 é uma diretiva interna.

Os próximos quatro bytes (L1, L2, L3, L4) representam o identificador dos parâmetros que estão sendo transmitidos. Tomando como exemplo a seleção de parâmetros utilizada na Figura 6 os valores recebidos para L1, L2, L3 e L4 seriam {17,18,19,20} respectivamente.

Os 8 conjuntos de 4 bytes seguintes (B11,B12,B13,B14 até B37,B38,B39,B40) representam os dados expressos em ponto flutuante de precisão simples. Desta forma para o identificador 17 os bytes {B11,B12,B13,B14} seriam dados referentes ao *pitch*, {B15,B16,B17,B18} seriam dados referentes aos *ailerons*, {B15,B16,B17,B18} são os dados do *heading true* e {B19,B20,B21,B22} são os dados de *heading magnético*.

Caso mais de um identificador seja marcado, este será composto apenas de 36 bytes contendo 4 bytes (L1,L2,L3,L4) relativos ao identificador dos parâmetros e os 8 conjuntos de 4 bytes restantes representando os dados expressos em ponto flutuante de precisão simples. Deste modo, o pacote de dados conterá um total de 77 bytes para 2 identificadores.

O número total de bytes do pacote de dados enviados pelo X-Plane pode ser calculado utilizando a Equação 1.

$$Nb = 5 + Np * 36 \quad (1)$$

Onde,

Nb Número de bytes do pacote de dados enviado

Np Número de identificadores

7. CONFIGURAÇÃO MATLAB

O Simulink apresenta alguns blocos que enviam e recebem dados através do protocolo UDP. A Figura 7 ilustra um diagrama de blocos para receber os dados selecionados na Figura 6.

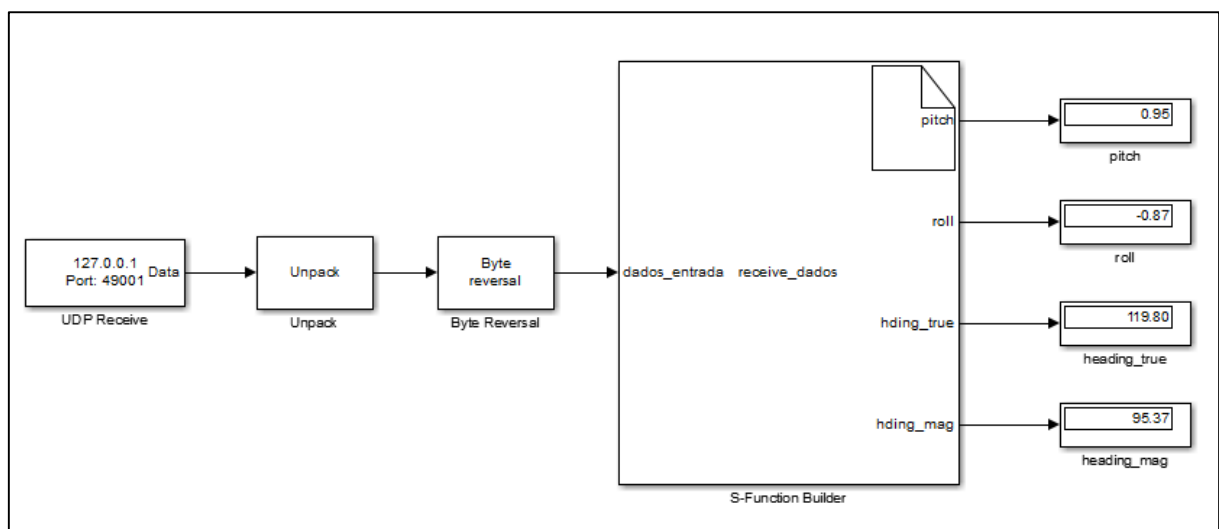


Figura 8 – Modelo desenvolvido em Simulink para receber os dados via comunicação UDP pelo X-Plane

A recepção do pacote de dados do X-Plane é executada pelos blocos do Matlab/Simulink, cuja configuração é apresentada a seguir.

- ***UDP Receive***

Pertence a biblioteca *Instrument Control* Toolbox do Matlab/Simulink e é responsável por receber os pacotes enviados pelo X-Plane, de forma que sua configuração fique de acordo com a Figura 9.

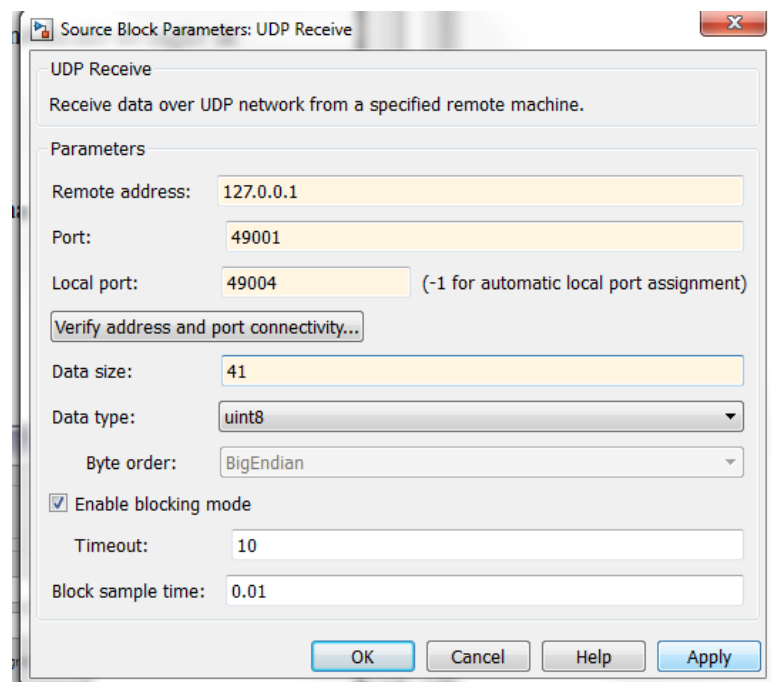


Figura 9 - *UDP Receive*

- ***Unpack***

Pertence à biblioteca *xPC Target/UDP* do Matlab/Simulink e desempacota os dados enviados pelo X-Plane. Sua configuração é feita de acordo com a Figura 10.

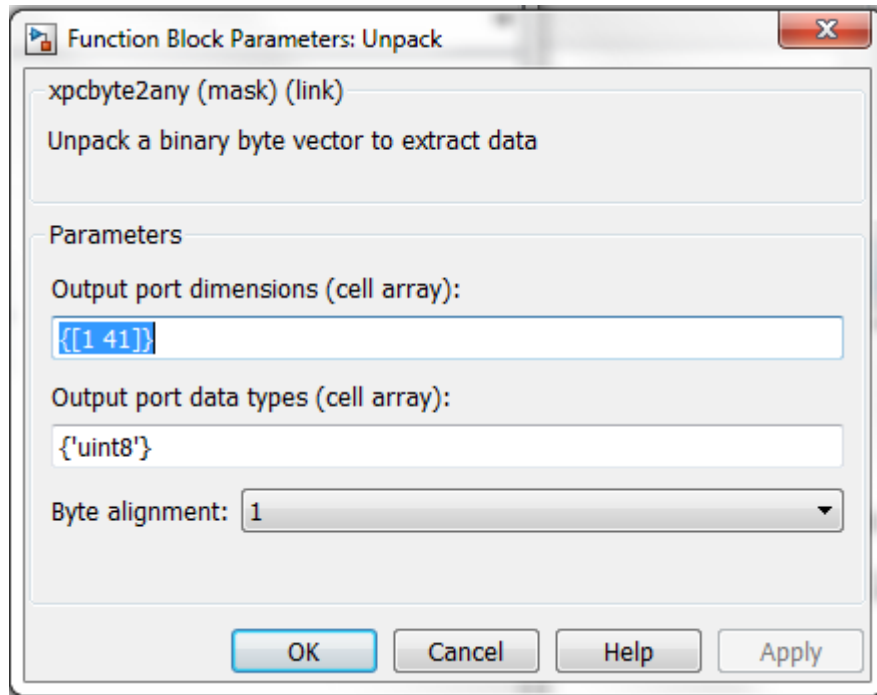


Figura 10 - *Unpack UDP*

- *Byte Reversal*

Pertence à biblioteca *xPC Target/UDP* do Matlab/Simulink e é utilizado para reverter os Bytes do pacote (necessário devido ao tipo de processador utilizado pela máquina).

- *S-Function Builder*

Depois de todo o tratamento o pacote recebido entra em um S Function Builder que é um bloco que permite gerar um código em C que roda diretamente no Simulink. Este bloco pertence a biblioteca *Simulink / User-Defined Functions*. As abas que devem ser configuradas devem seguir as Figuras seguintes.

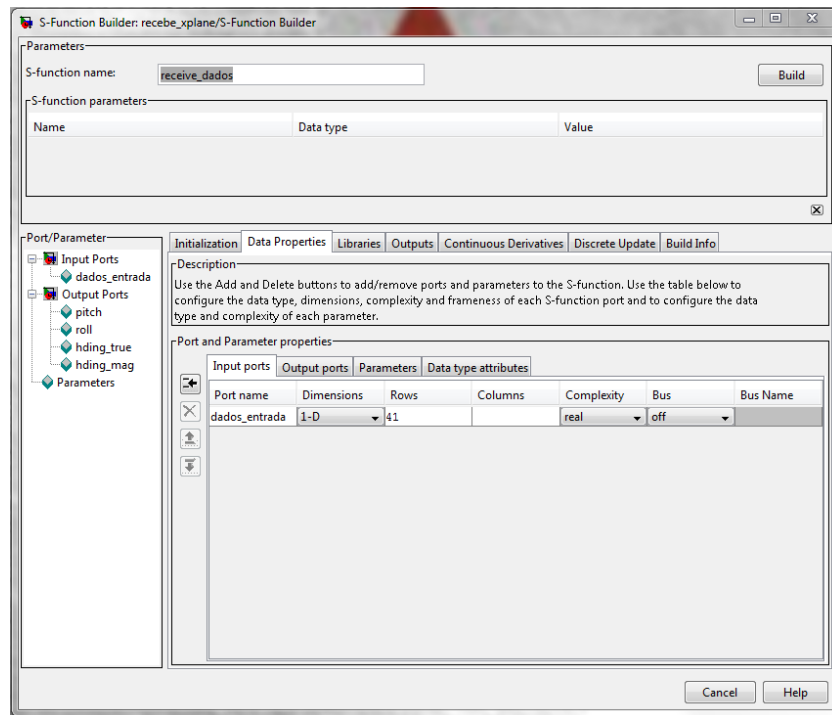


Figura 11 - S-Function Builder - Data Properties - Input Ports

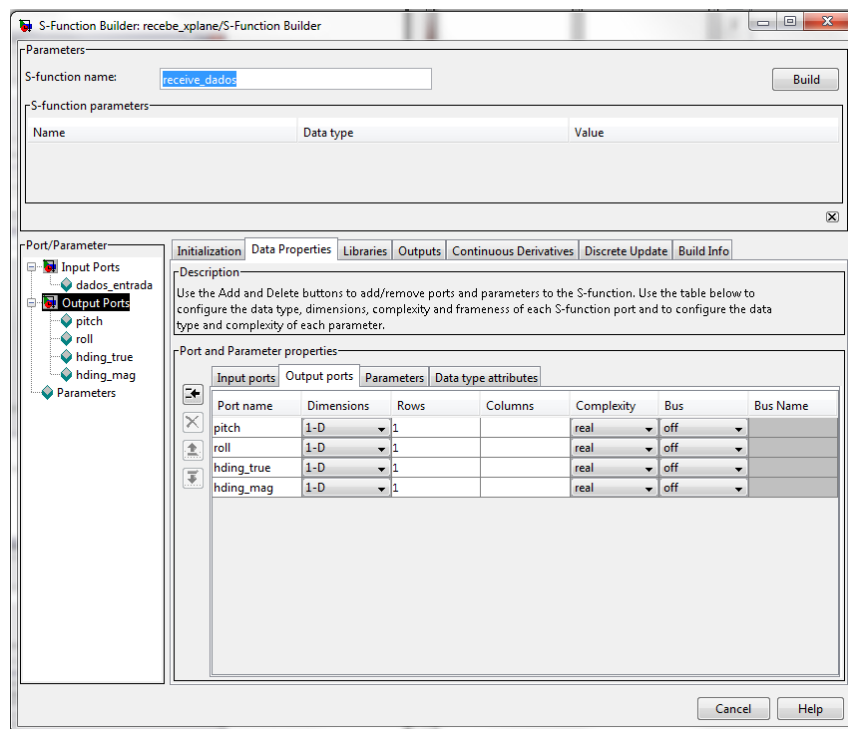


Figura 12 - S-Function Builder - Data Properties - Output Ports

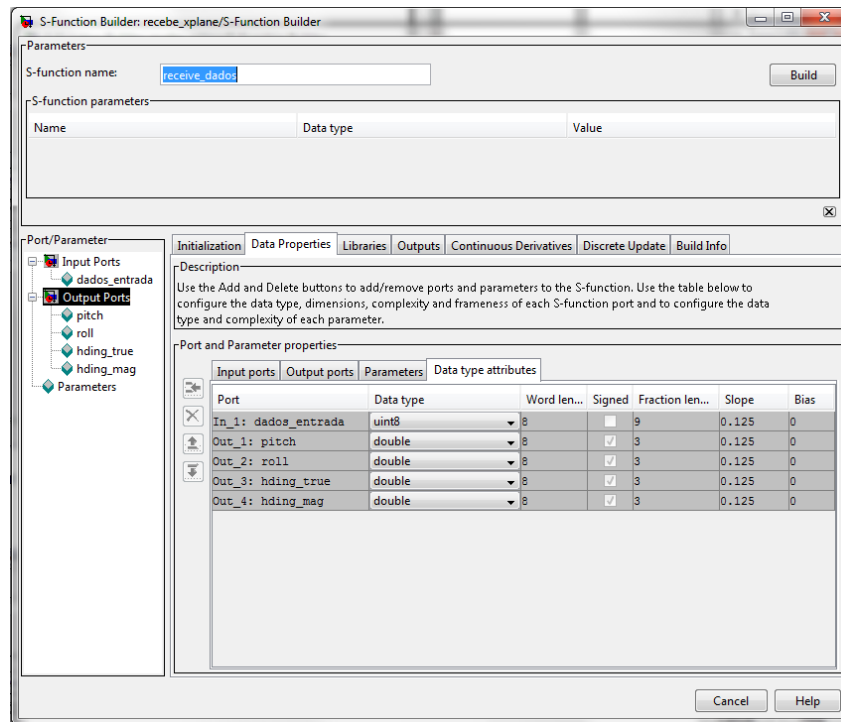


Figura 13 - S-Function Builder - Data Properties – Data type attributes

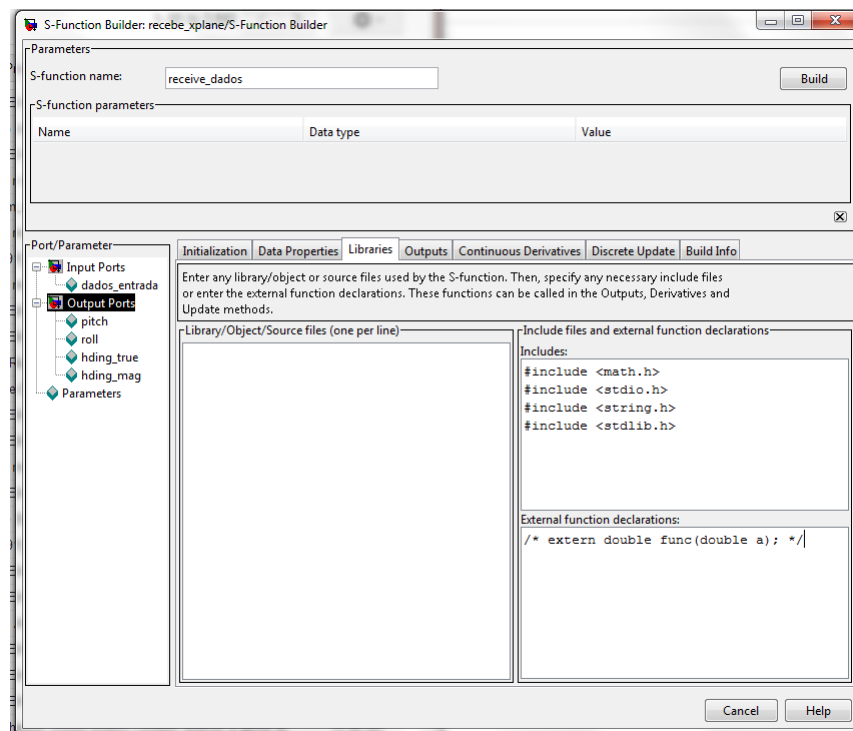


Figura 14 - S-Function Builder – Libraries

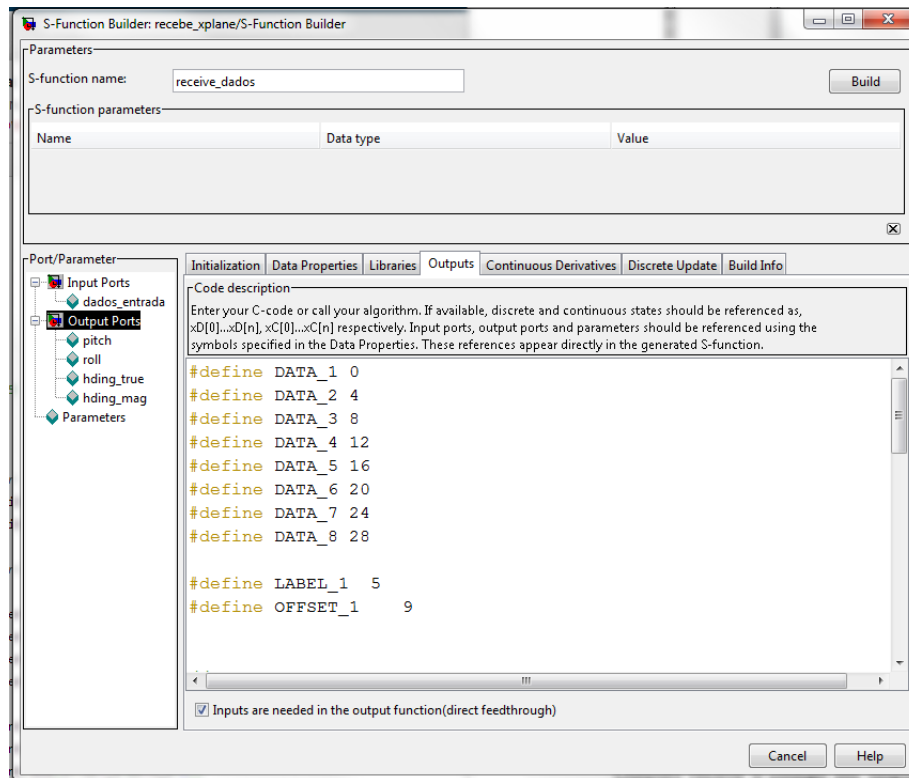


Figura 15 - S-Function Builder – Outputs

A aba Outputs recebe o código em linguagem C responsável pelo tratamento do pacote de dados recebido e o cálculo das variáveis de saída. Em resumo, o código analisa a estrutura do pacote de dados recebido agrupando os bytes, extraíndo os valores dos parâmetros e associando-os às variáveis de saída do bloco.

O código desenvolvido em linguagem C para tratar os dados da Figura 6 está disponível no endereço < <http://pastebin.com/A0SEecBF> >.