

Projeto 1 - Protocolo de Enlace

Alline Domingos e Natália Miranda

Engenharia de Telecomunicações

Instituto Federal de Santa Catarina

Outubro de 2018

Introdução

O presente relatório tem como principal objetivo apresentar as especificações do protocolo de enlace desenvolvido na disciplina de Protocolo por meio da linguagem python com intuito de aplicar na prática conteúdos abordados na disciplina de Redes de Computadores I e II.

Sendo assim, um protocolo de comunicação ponto-a-ponto foi implementado com requisitos de tamanho de quadro, formato de quadro, e mecanismos capaz de garantir controle de sequência, tratamento de erros, sincronismos, estabelecimento de conexão e controle de sessão.

Este relatório está dividido em 3 seções, onde na primeira seção apresenta-se as especificações da aplicação desenvolvida, em seguida na segunda seção apresenta-se cada classe da aplicação, e por fim faz-se uma breve explicação de como utilizar a aplicação desenvolvida.

1. Especificação

Para que fosse possível a realização da atividade que todos os protocolos desenvolvidos por cada equipe fossem capazes de enviar e receber pacotes de dados entre si utilizando um transceiver RF APC220 como UART (Universal Asynchronous Receiver/Transmitter) foram definidas algumas especificações para o protocolo desenvolvido como o tamanho limite da mensagem de 8 a 256 bytes, garantia de entrega, controle de sequência e tratamento de erros. Sendo assim, o formato do quadro é ilustrado na figura 1 abaixo e algumas especificações do mesmo são descritas a seguir:

ID CONTROLE	ID SESSÃO	ID PROTOCOLO	DADO	FCS
-------------	-----------	--------------	------	-----

Figura 1 - Quadro especificado no protocolo

- **ID CONTROLE** : Como mecanismo de garantia de entrega, para este protocolo utilizou-se a técnica de Stop-and-Wait (Pare-e-Espere), onde só transmite-se o próximo pacote após receber uma confirmação (ACK) do último pacote enviado. Ao enviar um pacote, um tempo de timeout de aplicação para obtenção de uma resposta, caso isto não ocorra, este pacote será transmitido novamente num total de 4 tentativas. Um característica importante de se ressaltar do ID CONTROLE e a representação de seus bits que são:
 - ➔ 3 bit: Representa o número de sequência.
 - ➔ 7 bit : Representa o tipo informação, se é ACK(1) ou DATA(0).

- **ID SESSÃO:** Utilizado para garantir que os pacotes recebidos por um ponta sejam correspondentes a ponta com qual se iniciou a comunicação faz-se necessário utilizar um identificador único entre os pacotes que serão trocados com intuito de garantir que um pacote com um ID SESSÃO diferente do id definido durante a negociação de estabelecido de enlace seja descartado, pois ele não faz parte daquela comunicação.
- **ID PROTOCOLO:** Utilizado para identificar qual tipo de conteúdo está sendo transportado pelo quadro. Sendo assim:
 - 0: Quadro de dados ou ACK
 - 255: Quadro de gerenciamento de sessão.
- **DADO:** Campo do quadro que pode possuir o conteúdo da mensagem enviada pelo transmissor (payload), ou pode conter as mensagens de transições do gerenciamento de sessão (CR,CC,DR,DC).
- **FCS:** Campo do quadro que conteúdo código de CRC, que tem por objetivo identificar se o pacote foi corrompido durante seu transporte.

2. Classes

Para o desenvolvimento desta aplicação foi necessário a criação e implementação de três classes que representaram cada subcamada do protocolo de enlace. As seções a seguir trazem uma descrição detalhada acerca da função de cada classe/subcamada.

2.1 Enquadramento

A classe enquadramento é a subcamada deste protocolo responsável por delimitar os quadros na interface com a camada física que oferece um serviço de envio e recepção de bytes sem nenhuma estrutura, portanto à camada de enlace tem como função delimitar a sequências de bytes obtidas em um frame. Para isso, neste trabalho foi utilizada a técnica sentinela que define o início e/ou fim de um quadro.

Nesta técnica, utiliza-se os bytes especiais 0x7E para delimitar quadro, e o byte 0x7D como byte de escape, logo, caso haja a ocorrência de um byte do tipo 0x7E ou 0x7D no meio do quadro de mensagem, é realizada uma operação lógica do tipo XOR (ou exclusivo) com byte 0x20 que resultará os byte 0x5E ou 0x5D, além disso um byte 7D deve ser colocado antes dos bytes modificados para indicar ao receptor que houve a alteração na informação do pacote antes do envio.

Outra função importante realizada pela classe enquadramento é o controle de erro que é realizado por meio da técnica de detecção de erros conhecida como CRC (Cyclic Redundancy Check), que executa um cálculo do pacote antes do envio, gera um resultado e adiciona ao frame. Na recepção este cálculo é realizado novamente, e caso o valor do CRC recebido não está de acordo com valor do CRC calculado pelo receptor o pacote é descartado, pois foi corrompido. A figura 3 abaixo ilustra o quadro montado pela quase enquadramento.

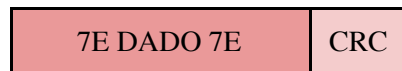


Figura 2 - Quadro do enquadramento

2.2 Garantia de Entrega (ARQ)

A classe ARQ é a subcamada deste protocolo responsável por pela garantia de entrega das mensagem ao destinatário, ou seja, enquanto a entrega de uma mensagem não for confirmada no transmissor ele não poderá enviar a próxima mensagem.

Para isso, o ARQ gerência por meio do ID controle o tipo de quadro que está sendo enviado, ou seja, por meio do bit 7 do byte deste ID ele identifica se o quadro recebido é do tipo ACK ou DATA, e através do 3 bit do byte deste ID ele reconhece a sequência do pacote, ou seja, ele verifica se é um ACK 0 (esperado pacote), ACK 1 (confirmação de recepção de pacote), DATA 0 (pacote de dado 0) ou DATA 1 (pacote de dado 1).

Além disso, outra função da classe ARQ, é a verificação se o ID de sessão do pacote recebido correspondem conexão realizada na classe sessão.

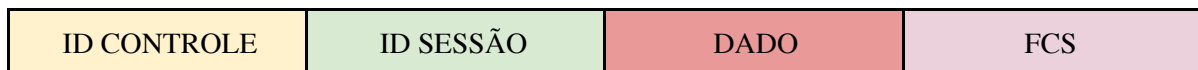


Figura 3 - Quadro do ARQ

2.3 Sessão

A classe ARQ é a subcamada deste protocolo responsável por pelo estabelecimento e encerramento de conexão entre o transmissor e receptor, ou seja, ele é o gerenciador da conexão do enlace.

Sendo assim, para gerenciar esta negociação de estabelecimento ou encerramento conexão a classe sessão utiliza as flags CR (solicitação de estabelecimento de sessão, 0), CC(confirmação de estabelecimento de sessão, 1), DR(solicitação de desconexão de sessão, 4) e DC(confirmação de desconexão de sessão, 5) para este fim, ou seja, a partir do valor da flag detectada ela saberá se deve encaminhar/aguardar um quadro do ARQ.

Outras funções importantes da classe sessão, é a geração do ID de sessão que será o identificador do enlace e o controle do ID protocolo que indicará qual tipo de conteúdo do quadro enviado/recebido.

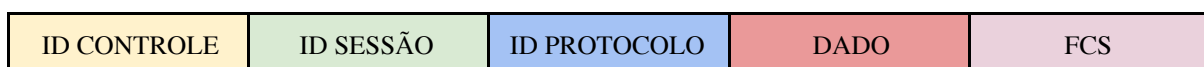


Figura 4 - Quadro da sessão

3. Conclusão

O protocolo de enlace ponto-a-ponto apresentado neste documento foi desenvolvido com intuito de resgatar conceitos abordados nas disciplinas de Redes de Computadores I e II durante o curso.

Para isso, a equipe optou pela linguagem de programação Python para o desenvolvimento do projeto.

Ao decorrer do desenvolvimento algumas dificuldades foram encontradas por não obter-se muita familiaridade com a linguagem, entretanto esta dificuldade foi superada e parte do protocolo desenvolvido (classes enquadramento e ARQ) apresentaram resultados satisfatórios. Já a classe sessão apresentou problemas quando integrada às classe ARQ, portanto o projeto funcionou apenas parcialmente devido a dificuldades encontradas pela equipe na implementação do gerenciamento de sessão.

Sendo assim, foi possível com este projeto entender por meio da prática as sutilezas e detalhes que envolve o bom e perfeito funcionamento de um protocolo de enlace ponto-a-ponto, pois ficou nítido que quando um usuário utilizando uma aplicação para enviar uma mensagem a um destinatário x, o tratamento da mensagem (montagem do quadro), controle de erros, a garantia de entrega e gerenciamento de sessão é de responsabilidade do protocolo que possui detalhes que agregam complexidades a sua implementação .