

CAMADA FÍSICA DA COMPUTAÇÃO

ENGENHARIA DA COMPUTAÇÃO - Rodrigo Carareto – 0#07E2/01

PROJETO 4 – Protocolo de comunicação UART ponto a ponto.

A situação é a seguinte: vamos nos imaginar em 2023. Agora você é um(a) engenheiro(a) de computação recém contratado(a) em seu novo trabalho, onde lhe é atribuído a tarefa (em dupla) de implementar um software em dois equipamentos que se comunicam serialmente com padrão UART. A comunicação deve ser feita para envio de um arquivo de um cliente para um servidor. Sem muitos detalhes, foi criado um primeiro esboço de como a camada superior da comunicação (aplicação) deve funcionar. Você poderá criar, mas esse esboço deve ser respeitado. O que chegou em suas mãos foi o documento mostrado a seguir. Você pode criar diferentes maneiras para implementar essa comunicação, ficando por sua conta as estratégias para ler e enviar corretamente os pacotes, evitando obtenção de pacotes não inteiros ou misturados, ou qualquer outro tipo de problema.

IMPORTANTE: A MÉTRICA PARA SEU SUCESSO É O THROUGHPUT !

Documento de definição do protocolo de comunicação

Padrao:

UART, baudrate 115200, sem bit de paridade

Datagrama

Cada envio deve ser feito como um datagrama completo, contendo head, payload e eop. O tamanho do payload não pode ultrapassar 128bytes e o tamanho do datagrama não deve ultrapassar 148 bytes.

Cada datagrama enviado deve conter no head **obrigatoriamente** o **tipo de mensagem** que ele representa e o **tamanho do payload**. Um único byte deve ser reservado no head para conter o número representativo do tipo de mensagem. Dado o tipo de mensagem, algumas outras informações devem estar no head, como definido a seguir.

Tipos de mensagens

TIPO 1 – Esta mensagem representa um chamado do cliente enviado ao servidor convidando-o para a transmissão. Nesse caso, o head deve conter um byte com o número 1, indicando mensagem tipo 1, e outro byte com um identificador. O identificador é o número do servidor, sendo que quando este receber uma mensagem tipo 1, verifica se é para ele mesmo o envio. A mensagem tipo 1 deve conter o número total de pacotes que pretende-se enviar!

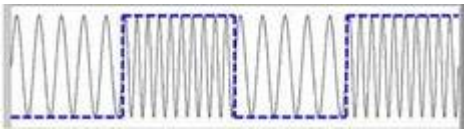
TIPO 2 – Essa mensagem é enviada pelo servidor ao cliente, após o primeiro receber uma mensagem tipo 1 com o número identificador correto. Deve conter no head o número 2 no byte reservado ao tipo de mensagem. O significado de uma mensagem tipo 2 é que o servidor está ocioso e, portanto, pronto para receber o envio dos pacotes.

TIPO 3 – A mensagem tipo 3 é a mensagem de dados. Este tipo de mensagem contém de fato um bloco de dados a ser enviado (payload). Deve conter o número 3 no byte reservado ao tipo de mensagem. Essa mensagem deve conter também **o número do pacote que envia (começando do 1) e o total de pacotes a serem enviados**.

TIPO 4 – Essa mensagem é enviada do servidor para o cliente toda vez que uma mensagem tipo 3 é recebida pelo servidor e averiguada. Quando verificado que a mensagem é realmente o pacote que o servidor estava esperando e que tal mensagem chegou em perfeitas condições (eop no local correto), o servidor envia então a mensagem tipo 4, afirmando que recebeu o pacote. Essa mensagem deve ter o número 4 no byte reservado ao tipo de mensagem. Além disso, deve conter o número do último pacote recebido e já aferido.

TIPO 5 – É uma mensagem de time out. Toda vez que o limite de espera exceder o timer dedicado a isso, em qualquer um dos lados, deve se enviar essa mensagem e finalizar a conexão. Essa mensagem deve ter o número 5 no byte reservado ao tipo de mensagem.

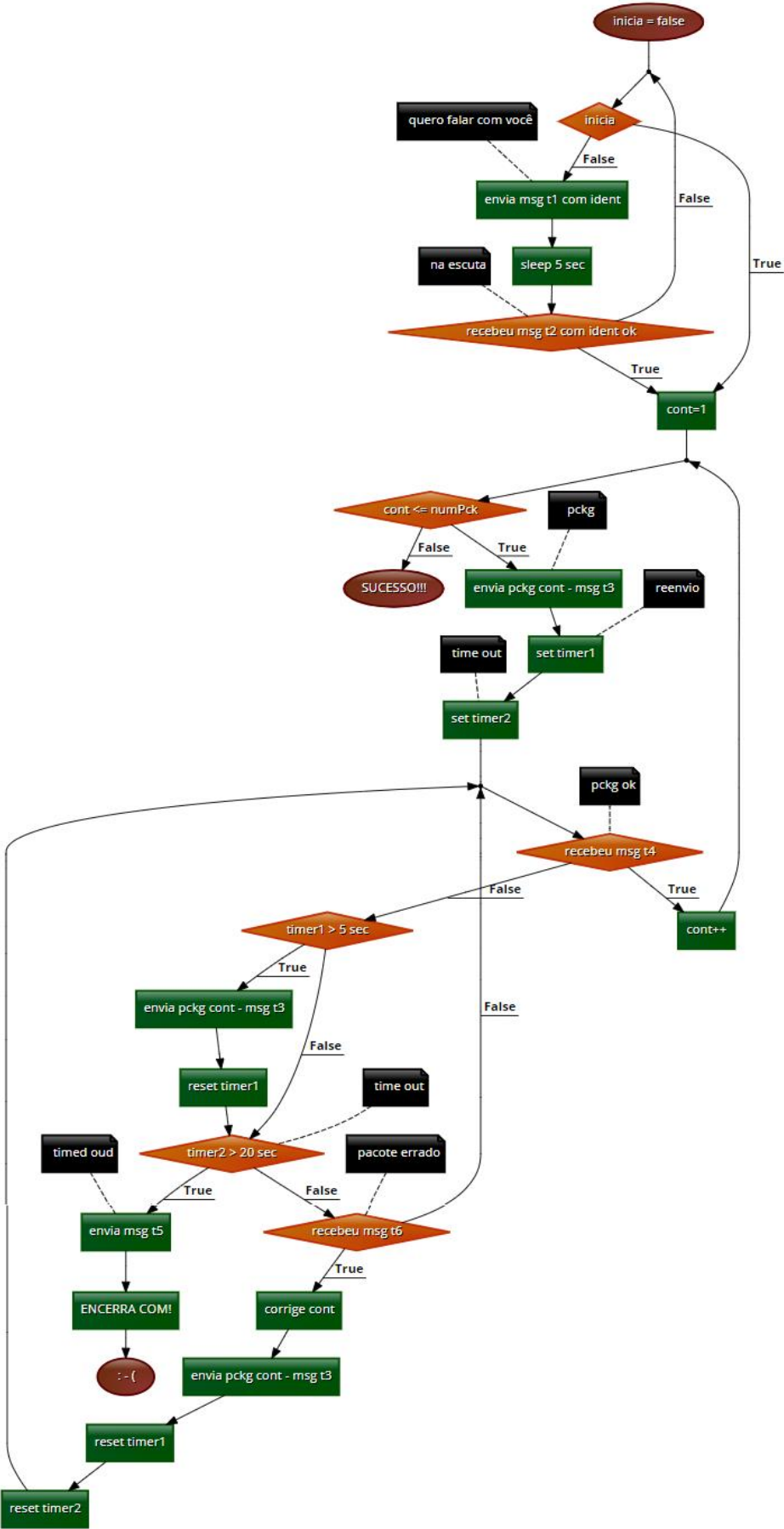
TIPO 6 – É uma mensagem de erro. O servidor deve enviar esta mensagem ao cliente toda vez que receber uma mensagem tipo 3 inválida, seja por estar com bytes faltando, fora do formato correto ou por não ser o pacote esperado pelo servidor (pacote repetido ou fora da ordem). Essa mensagem deve ter o número 6 no byte reservado ao tipo de mensagem. Além disso, deve conter o número correto do pacote esperado pelo servidor, independentemente do problema que invalidou a mensagem. Isso orienta sempre o cliente para o reenvio.

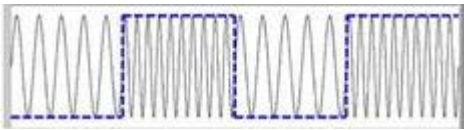


CAMADA FÍSICA DA COMPUTAÇÃO

ENGENHARIA DA COMPUTAÇÃO - Rodrigo Carareto – 0#07E2/01

PROTOCOLO CLIENTE





CAMADA FÍSICA DA COMPUTAÇÃO

ENGENHARIA DA COMPUTAÇÃO - Rodrigo Carareto – 0#07E2/01

PROTOCOLO SERVER

