**Desenho do pacote**

Imagem anexada

Dois bytes do Head guardam o tamanho, e um significa o começo.

O End of Packet não representa muito por enquanto, pois não foi usado em nosso método. Contudo, representa 1 byte do packet e será usado futuramente.

**Cálculo do Overhead**

[overhead](https://github.com/elisamalzoni/Proj-1-Comunicacao/blob/master/2-COM-Datagrama/overhead.png)

O Overhead é a razão entre o tamanho total e a carga útil. Neste caso, é payload+4/payload. Ou seja, 1+(4/tamanho do payload).

Para um payload de tamanho 3000, esse valor da 1.00133

**Cálculo do tempo teórico de transmissao de qualquer imagem**

A velocidade padrão de transmissão pela porta serial é de 9600b/s. Multiplica-se por 8 para obter os bits, e soma-se o valor do head+eop (4\*8).

**Cálculo do Thoughput**

O tamanho da imagem somado aos valores de bytes do Head e do EOP fazem o Troughput. No caso, tamanho + 4.

## Descrição do HandShake implementado

O client envia um comando do tipo SYN e quando o server recebe, o server responde com dois comandos, um ACK e um SYN. Se o cliente passar 2 segundos sem ter recebido esses comandos, ele manda outro SYN e assim continua até que os receba. Quando ele recebe esses dois comandos o client manda um ACK e se o server recebe o ACK a comunicação está feita.

## Descrever os pacotes (SYN, ACK, NACK e dados)

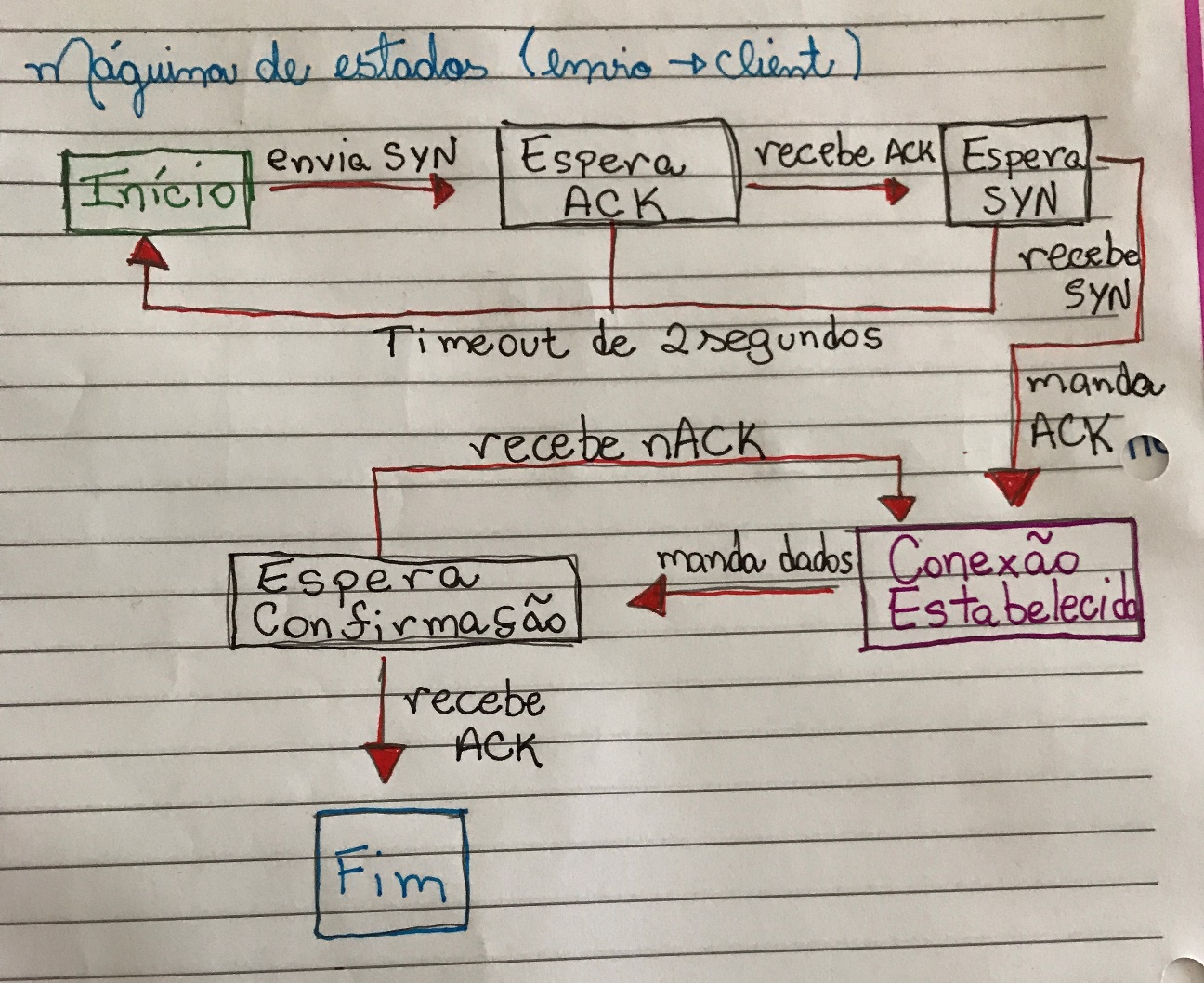
## C:\Users\vitor\AppData\Local\Microsoft\Windows\INetCache\Content.Word\image1.jpg

## C:\Users\vitor\AppData\Local\Microsoft\Windows\INetCache\Content.Word\image2.jpg

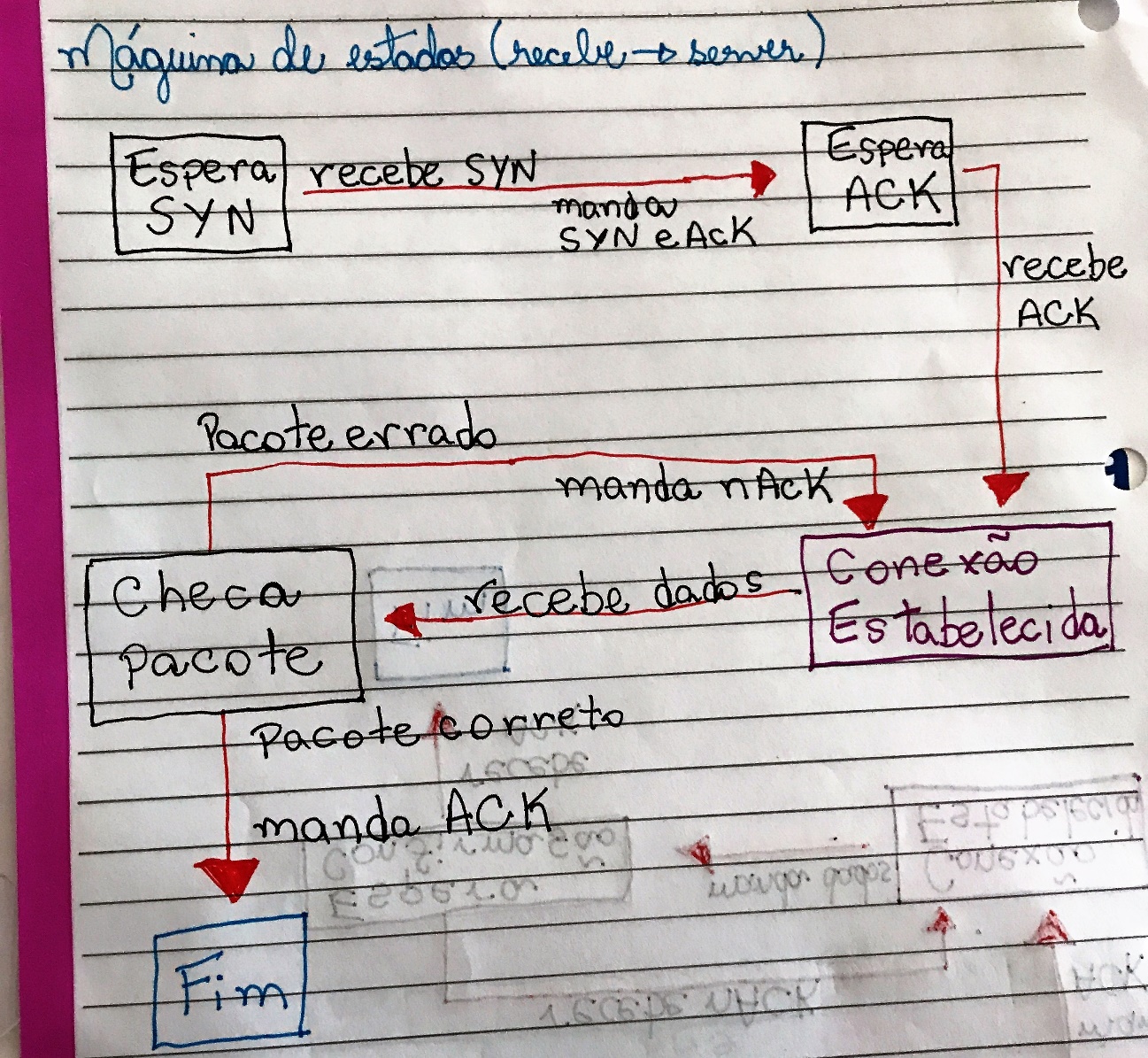
## C:\Users\vitor\AppData\Local\Microsoft\Windows\INetCache\Content.Word\image3.jpg

## C:\Users\vitor\AppData\Local\Microsoft\Windows\INetCache\Content.Word\image4.jpg

## Diagrama do envio de pacotes como uma máquina de estados



## Diagrama a recepção de pacotes como uma máquina de estados



## Descrever o tempo de timeout utilizado (e o porque desse valor)

O timeout utilizado foi de 2 segundos, a melhor otimização encontrada pela dupla ao perceber a recepção dos dados através da observação dos prints do recebimento.

## Diferença entre pacotes de comando (SYN,ACK,NACK) de pacote de dados

Os pacotes possuem um byte reservado no HEAD que indica seu tipo (dados, SYN, ACK ou NACK) através dos números 0x03, 0x00, 0x01 e 0x02 respectivamente. Como os pacotes de comando SYN, ACK e NACK não possuem payload, seu size é 0x00.