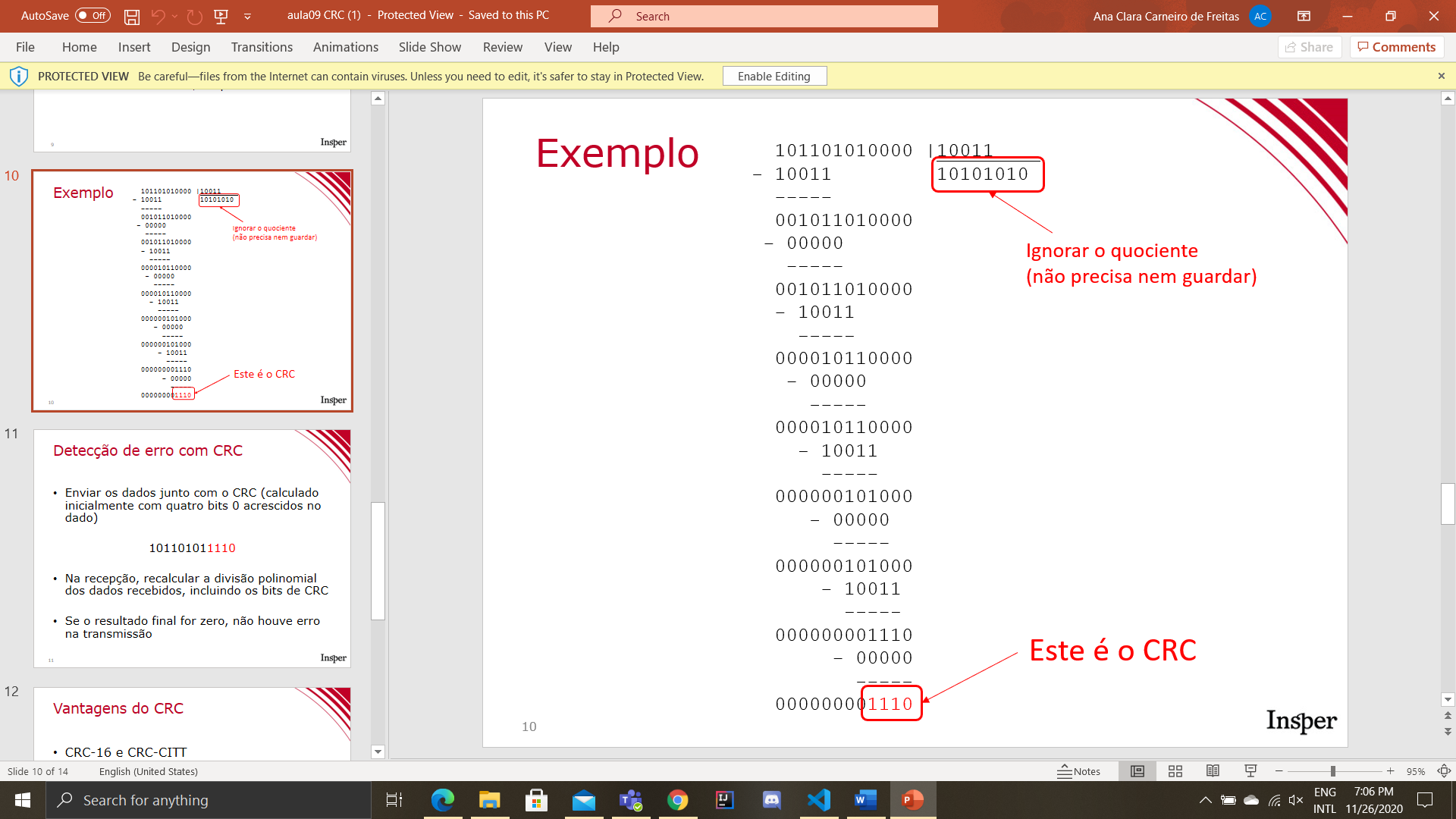
Resumo prova camada física

CRC

* É uma conta de divisão de polinômios realizada tanto pelo client quanto pelo server com o intuito de verificar se a informação transmitida e recebida é a mesma, tal divisão pode ser realizada apenas com o payload ou com o pacote completo.
* Os bits da sua sequencia são divididos por uma chave, a qual é definida pelo programador, gerando um resto único (**CRC**), dessa forma, por ser valores únicos, os dois lados tem que dar igual.
* O fato de ser uma divisão é o que faz o resultado ser único, caso fosse uma soma, dois erros poderiam se anular e com isso o erro poderia passar despercebido.
* Geralmente a chave tem 16 bits, o que resulta num polinomio de quarto grau, a partir de tal valor (4), é definido o numero de zeros que devem ser adicionados à sequencia para que a conta dê certo. Além disso, o resto gerado é da mesma ordem que a chave.

Quanto mais bits tiver em uma chave, mais confiável a resposta é.

* 10110101 = x^7 + x^5 + x^4 + x^2 + x^0



* Esse CRC geralmente é enviado junto com as informações do head
* A vantagem de fazer essa verificação por pacote é que se algum erro acontecer da pra saber em que pacote foi e, consequentemente, lidar com o reenvio a partir daquele pacote, sem ter que reiniciar toda a mensagem.

Serialização UART

* UART significa protocolo universal de transmissão e recepção assíncronos, o qual não possui o conhecimento de quando a informação vai chegar, sendo necessário uma certa comunicação para avisar. O UART começa com 1, dessa forma quando o sinal cai para 0 significa que o start bit chegou, dando inicio à transmissão.
* O **Baudrate** é o número de bits/ segundos, geralmente definido na interface física, sendo ele o inverso do período (baudrate = 1/período)
* **Start bit** é utilizado para avisar que a mensagem vai começar
* **TX** parte responsável por transformar a mensagem em bits
* **RX** parte responsável por analisar a tensão 1,5 período depois do start bit
* **Parity bit**  serve como uma forma de verificação para o recptor saber se a transmissão de bits veio certa (pode ser dispensável)
* **Parity bit par:** primeiro deve-se somar seus oito bits de informação, se esse numero for par, então o parity bit é 1
* **Parity bit impar:** primeiro deve-se somar seus oito bits de informação, se esse numero for impar, então o parity bit é 1
* **Stop bit** serve para avisar que a mensagem acabou, dessa forma fica um período no 0

Transmissão de dados

* Transdução: transforma uma grandeza mecânica em uma elétrica, como por exemplo a variação da pressão no tempo de um áudio para tensão no tempo. Tal transformada é responsável por transformar um sinal acústico em elétrico.
* Assim, quando esse sinal é transformado em elétrico é possível guardá-lo digitalmente por meio de uma **conversão analógico-digital (A/D)**. Essa conversão, para que seja possível guardar o sinal inteiro é necessário **amostrar**, que é basicamente olhar o gráfico em um determinado instante e guardar seu valor, dessa forma quanto mais amostras forem colhidas em um menor intervalo de tempo mais preciso seu áudio ficará, pois existirão menos gaps.
* Diferença entre **analógico** e **digital:**
* Analógico: possui valores em todos os instantes, ou seja, possui infinitos valores
* Digital: só possui valores nos intervalos de tempos em que as amostras foram feitas, possuindo finitos valores
* **Período de amostragem**: tempo decorrido entre duas amostras
* **Frequência de amostragem**: quantidade de amostras medidas em um segundo (quanto maior a frequência mais preciso é)
* Tensão -> amostra = conversor A/D
* Amostra -> tensão = conversor D/A (não é tão bom pois durante sua conversão ele fica um pouco quadrado pois algumas informações foram perdidas nos intervalos de tempo)
* **Bit depth ou Resolução** são quantos bits são utilizados para armazenar cada amostra (quantos mais bits forem utilizados para guardar a amostra mais precisão você vai ter pois terão mais bits disponíveis para armazenamento)
* **Amostra em alta resolução**: cada amostra feita tinha muitos bits disponíveis para guardar a informação, ficando com mais precisão
* **Range** é o alcance do que está sendo gravado, sendo os valores mínimos e máximos que podem ser guardados (quanto menor o range melhor, pois, considerando que a mesma quantidade de bit depth serão utilizadas, a escala ficará menor, possibilitando uma maior precisão
* **Menor variação de sinal possível** é o calculo que considera o bit depth (número dos possíveis valores) com o range da amostra, mostrando assim qual seria a menor variação possível
* **Variação discreta**: a partir da variação encontrada é possivel descobrir quais valores poderiam ser descartados

**Menor variação = range / numero de possiveis valores**

* **Taxa de amostragem insuficiente** é capaz de gerar sinais diferentes com os mesmos resultados de amostragem, resultando numa impossibilidade de reproduzir o som. Uma forma de resolver isso seria colhendo mais amostras, ou seja, aumentando a taxa de amostragem.
* **Fenomeno de aliasing** significa um sinal falso, outro sinal
* PARA UMA MAIOR PRECISÃO: maior frequencia de amostragem, mais bit depths, menor range, menor intervalo de tempo

Transformada de Fourier

* Consiste na somatória de diversas senoides com tamanhos e frequencias diferentes que resultam no sinal desejado.
* Cada senoide que compõe o sinal chama componente ou armônico
* Quando plotado a transformada exitem uma série de frequências negativas as quais são apenas o espelho do positivo, podendo ser ignorados pois não possuem significado físico
* Cada **pico** do gráfico corresponde a uma senoide, quanto maior esse pico mais relevante a senoide é na somatória

Filtros

* É uma função que olha o sinal gerado e elimina algumas frequencias
* **Filtro passa baixo**: só deixa passar frequências baixas
* **Filtro passa alto**: só deixa passar frequências altas
* **Filtro passa faixa**: recorta o gráfico deixando apenas as frequências do meio

Modulação AM

* Modulação feita pela multiplicação do sinal com a portadora, fazendo com que o sinal final tenha sua amplitude modificada

Modulação FM

* Essa modulação ocorre por meio da soma do sinal com a portadora resultando numa mudança de frequência. Isso é possível pois o sinal é muito lento ao ser comparado com a portadora, assim, ao somá-las a portadora consegue mudar sua frequência com agilidade, afinal a frequencia da portadora é muito maior que a do sinal.
* A frequencia não pode sair muito pra fora da banda de frequencia limite, dessa forma é feito uma relação amplitude e frequencia com um limite aceitável do valor

Modulação Digital

* Formas de fazer uma transmissão que consuma menos energia: sumir com a portadora e com as frequencias repetidas.
* Não é possível transmitir uma onda quadrada com seu baudrate definido pois as ondas possuem picos, dessa forma, ao fazer o fourier seria necessário uma banda muito larga para reproduzir o sinal, afinal pra uma variação tão brusca de frequência são necessários muitas senoides
* Como enviar esses bits?
* **ASK** envio da senoide com variação de amplitude na medida que fossem recebidos 0s ou 1s. Entretanto possui a limitação de que a frequência da portadora limita a transmissão do sinal, não conseguindo trabalhar com frequencias muito altas.

**X(t) = A cos (2.pi.f.t)**

* **PSK** envio de uma senoide com variação de fase de acordo com os 0s e 1s, sendo elas defasadas em 180 graus

**X(t) = A cos (2.pi.f.t + fase(t))**

* **FSK** envio de uma senoide com uma variação de frequência de acordo com 0s e 1s, entretanto possui uma limitação com a frequencia da da portadora, afinal é necessário alguns períodos para descobrir o bit.

**X(t) = A cos (2.pi.(f+fo).t )**

* **QAM** modalidade que mistura a modulação de amplitude com a de fase, adiciona mais fase e mais amplitudes podendo transmitir mais bits ao mesmo tempo, o que aumenta a velocidade da transmissão
* **Erros:** quanto mais ruídos tiver na transmissão maior será a taxa de erro

**Eb(energia do sinal por bit) / No(energia do ruido por bit)**

* Maior o eb/no melhor o sinal
* Menor o eb/no pior o sinal (mais erros)