

Para implementar esse sistema, será utilizado um micro controlador Arduino nano, um fotodiodo TSC 14, um resistor de 100Ω para limitar a corrente do Led e um Led branco de 5mm, o Led está conectado a protoboard através de um cabo de 1 metro. Abaixo podemos as ligações do circuito.

FIGURA LIGAÇÕES PINOS ARDUINO

Para efetuar o controle de um sistema de comunicação podemos utilizar uma FSM, o conceito de FSM é concebido como uma máquina abstrata que deve estar em um de seus finitos estados. A máquina está em apenas um estado por vez, esse estado é chamado de estado atual. Um estado armazena informações sobre o passado, isto é, ele reflete as mudanças desde a entrada em um estado, no início do sistema, até o momento presente. Uma transição indica uma mudança de estado e é descrita por uma condição que precisa ser realizada para que a transição ocorra. Uma ação é a descrição de uma atividade que deve ser realizada em um determinado momento. Cada estado descreve um nó de comportamento do sistema em que está à espera de uma condição para executar uma transição.

Esse sistema de comunicação terá 2 FSM, uma para o transmissor e outra para o receptor, ambas serão executadas em um micro controlador Arduino Nano, onde a FSM do receptor será executada no loop principal e a do transmissor será ativada por interrupções do timer, para gerar essas interrupções será utilizada a biblioteca TimerOne.h e a obtenção das informações referentes a erros de transmissão será via comunicação serial.

Nesse sistema, podemos definir 3 variáveis via código, a primeira delas no receptor e as demais no transmissor, a primeira delas é um limite de luz, que consiste no valor digital referente a luminosidade que representa o nível lógico baixo, o símbolo 0 no caso, ele varia de 0 a 1023. Sem comunicação, apenas com outra lâmpada fluorescente compacta de 25W, o valor digital da leitura da luminosidade pelo fotodiodo fica em torno de 20, então para testar a comunicação nesse ambiente eu defino esse valor como sendo 30, e para um ambiente sem iluminação externa defino como sendo 2. Alterei o código para que esse valor fosse sendo alterado automaticamente durante a execução, lendo os valores de alto e baixo, e a partir dessa média, e com um acréscimo de um fator que seria definido pelo usuário, como um offset, o sistema identificaria o melhor valor antes de cada recepção, mas para realizar os testes não utilizei isso para manter o mesmo valor fixo entre os testes e comparar apenas o que foi alterado.

Outro parâmetro que pode ser alterado é o tamanho do pacote de bits que será enviado, teoricamente esse valor pode ser qualquer valor maior ou igual a 1, mas para realizar os testes eu defino entre 3 até 9.

E podemos também definir o intervalo de interrupções do transmissor em microssegundos, definindo assim a frequência do transmissor, que novamente pode ser qualquer valor positivo, e para realizar os testes eu defino entre 1200 até 4000, testando assim transmissões com frequências entre 250Hz até 833Hz.

Transmissor

O transmissor é ativado através de interrupções de tempo, a cada interrupção ele pode mudar de estado ou transmitir o próximo símbolo e continuar no estado de transmissão. Ele é composto de 5 estados e uma função, essa função é responsável pela codificação Manchester

dos dados antes do envio e é executada no quarto estado, ela simplesmente pega cada bit a ser transmitido e salva nas posições pares de um vetor de símbolos que será enviado, e faz uma operação de negação de cada bit par e armazena na próxima posição ímpar, gerando assim um vetor de símbolos onde teremos a mesma quantidade de 0s e de 1s, o que é muito importante para um sistema de comunicação via luz, pois assim mantemos a média de luminosidade estável. Além disso, quando não está transmitindo nenhuma informação, eu optei por deixar o transmissor enviando o símbolo 1, mantendo o Led ligado para iluminação por exemplo.

No primeiro estado, o transmissor liga o Led e aguarda a próxima interrupção.

No segundo estado, o transmissor desliga o Led e verifica se existe algum dado a ser enviado, mas antes de enviar esse estado precisa ser executado de novo, já que eu implementei o receptor de uma maneira que onde eu preciso receber 2 bits 1 antes de começar a recepção de dados, então mesmo que eu tenha alguma informação a ser enviada, na próxima interrupção eu volto para o primeiro estado, e aí sim, na próxima interrupção se eu retorno ao segundo estado, e se tiver algum dado a ser enviado, na próxima interrupção ele vai para o terceiro estado, se não ele fica intercalando entre os dois primeiros estados.

No terceiro estado eu mantenho o Led desligado, alterando o padrão agora para o bit 0, que representa que o transmissor irá iniciar a transmissão de um dado.

No quarto estado eu ligo o Led, terminando o envio do bit 0 pela codificação Manchester e passo para o quinto estado, que a cada interrupção ira ler uma posição do vetor de símbolos a ser enviado, e irá ligar ou desligar o Led de acordo com esse vetor até que todo o pacote seja enviado e retorne para o primeiro estado.

Receptor

O receptor utiliza 4 funções e possui 6 estados, a função recebe executa a leitura do fotodiodo duas vezes, realiza o cálculo da média deslocando um bit para direita e retorna esse valor. A função recebe símbolo realiza a leitura do fotodiodo e retorna se recebeu o símbolo 1 ou 0, comparando o valor lido com aquele valor definido em limite de luz, essa função também realiza 2 leituras do fotodiodo e calcula a média antes de comparar. Existe uma função que realiza a decodificação da codificação Manchester, retornando os bits recebidos e uma para comparar os dados enviados e recebidos e registrar os erros, enviando ao final de cada teste de 1000 pacotes recebidos a quantidade de pacotes enviados e recebidos, para comparar o sincronismo entre o transmissor e receptor, além do total de bits recebidos corretamente e com erros, todas essas informações são enviadas através da porta serial.

No primeiro estado o receptor aguarda a recepção de um símbolo 1, nesse estado ele guardaria o valor alto para calcular o limite de luminosidade e vai imediatamente para o segundo estado, onde fica travado até receber o símbolo 0. Quando recebe o símbolo 0, ele verificaria o valor baixo e calcularia o novo limite de luz, como sendo a média dos valores lidos para alto e baixo mais o offset definido pelo usuário, e nesse estado ele inicia a contagem de um timer em microssegundos e vai imediatamente para o próximo estado.

No terceiro estado, ele aguarda a recepção de outro símbolo 1, quando esse símbolo é recebido e calcula o tempo do símbolo 0 e vai para o quarto estado, onde aguarda a recepção de outro símbolo 0, e dispara outro timer em microssegundos e muda para o quinto estado.

No quinto estado ele aguarda a recepção de um símbolo 1, quando esse símbolo é recebido ele efetua novamente o cálculo do tempo do símbolo 0, se a duração desse símbolo for 1.5 vezes maior, quer dizer que agora ele recebeu um bit 0 e antes um bit 1, já que o bit zero após um bit 1 tem o dobro do tempo, mas considere uma margem, por isso 1.5 vezes.

Isso quer dizer que uma transmissão irá começar, então ele vai para o sexto estado, se não ele retorna para o primeiro estado, por esse modo que foi implementado o receptor que são necessários 2 bits 1 antes de enviar o pacote de dados.

No sexto estado, ele aguarda o tempo de um símbolo e armazena o valor recebido após esse tempo, ele continua fazendo isso até receber todo o pacote, por isso o tamanho do pacote deve ser definido no receptor também. Após receber todo o pacote ele executa a função que descodifica os dados e a função que verifica os erros, após isso retorna ao estado inicial.

Testes

Foram realizados testes em um ambiente com iluminação externa variando-se a distância e 6 diferentes pacotes de bits, e para um ambiente escuro variando se a distância, 6 diferentes pacotes de bits, frequência, ângulo e tamanho do pacote. Os testes de reflexão não foram realizados pois o Led utilizado era muito fraco.

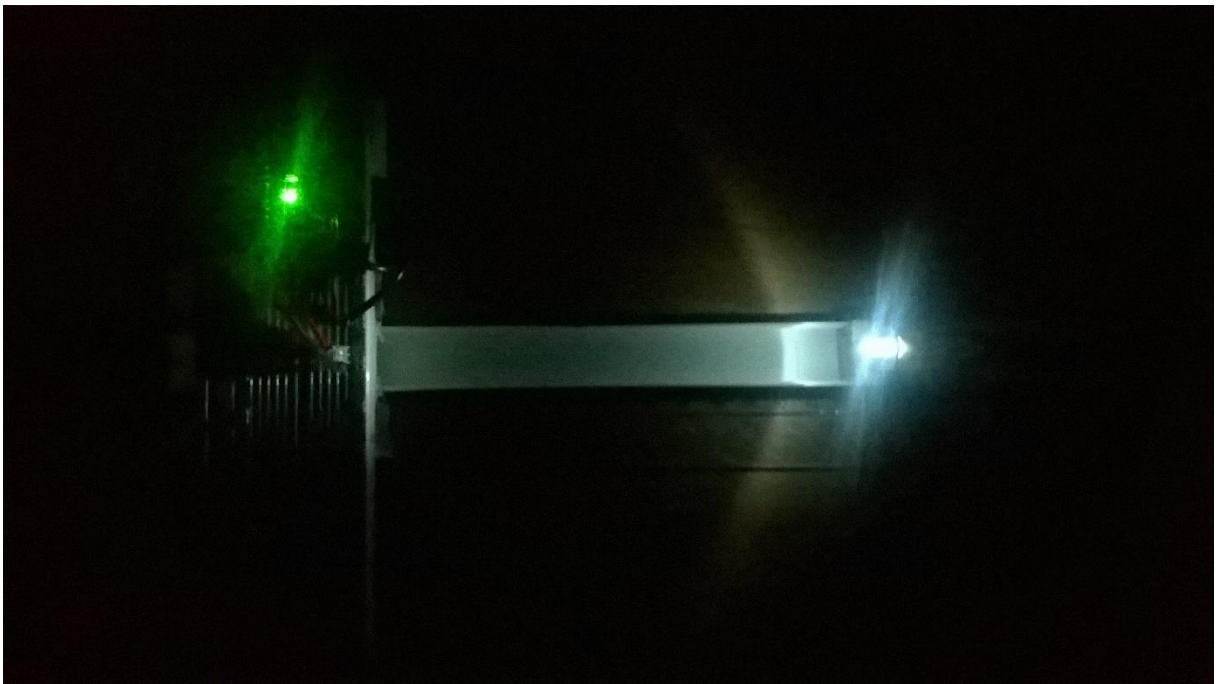
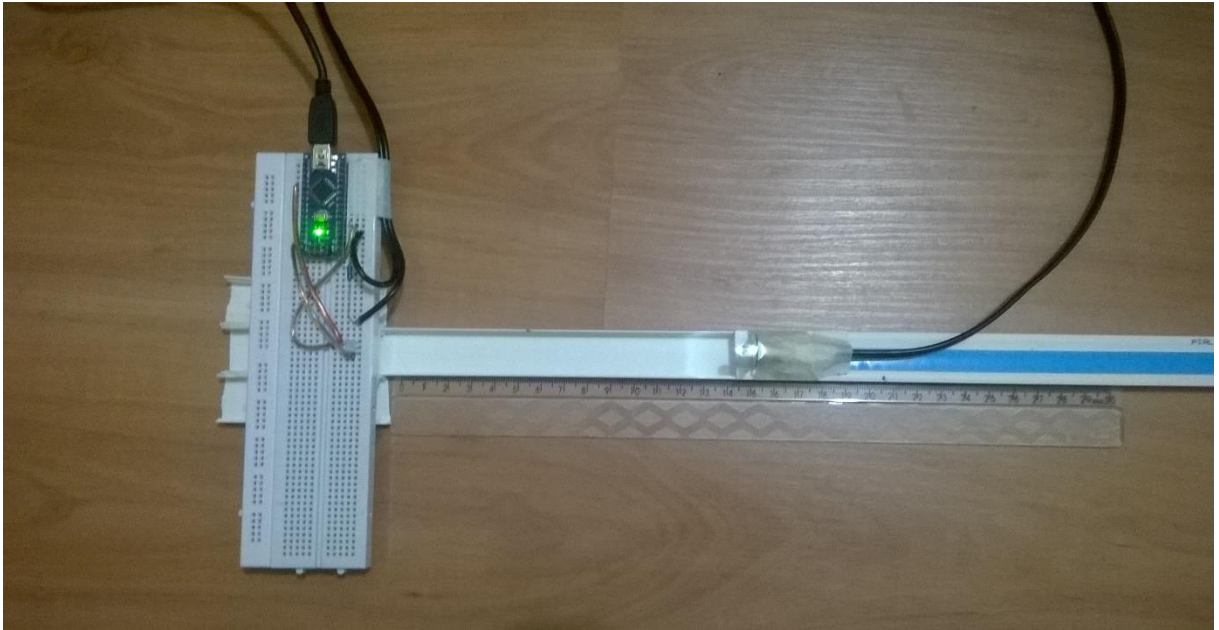
As combinações de pacotes de bits escolhidas para os testes foram 0000, 0101, 0110, 1001, 1010, 1111.

IMAGEM DOS SIMBOLOS DESSES BITS

Para cada teste foram enviados 1000 pacotes de bits 10 vezes, calculado a média e através dos valores máximos e mínimos de cada teste definidos o intervalo de confiança, o intervalo de confiança foi definido para cada tipo de teste, para cada conjunto de 10 valores, foi calculado o valor médio, o valor máximo e o valor mínimo, esses valores máximo e mínimo foram utilizados para determinar o erro positivo e negativo para cada conjunto de valores, e o intervalo de confiança utilizado foi o máximo desses valores dentro de cada teste, já para o BER foi utilizado um intervalo de confiança de 5%.

A imagem a seguir mostra a realização dos testes no ambiente com iluminação externa, o ambiente estava sendo iluminado por uma lâmpada fluorescente compacta de 25W. a protoboard foi fixada sobre uma canaleta e o Led também, deste modo a distância entre o fotodiodo e o Led pode ser alterada sem alterar o ângulo.

A segunda imagem mostra a realização desses mesmos testes, só que agora sem a iluminação externa, e a terceira imagem mostra os testes de ângulo, onde utilizei uma folha com os ângulos desenhados como orientação e agora tanto o Led quanto a protoboard foram removidas da canaleta para continuarem da mesma altura.

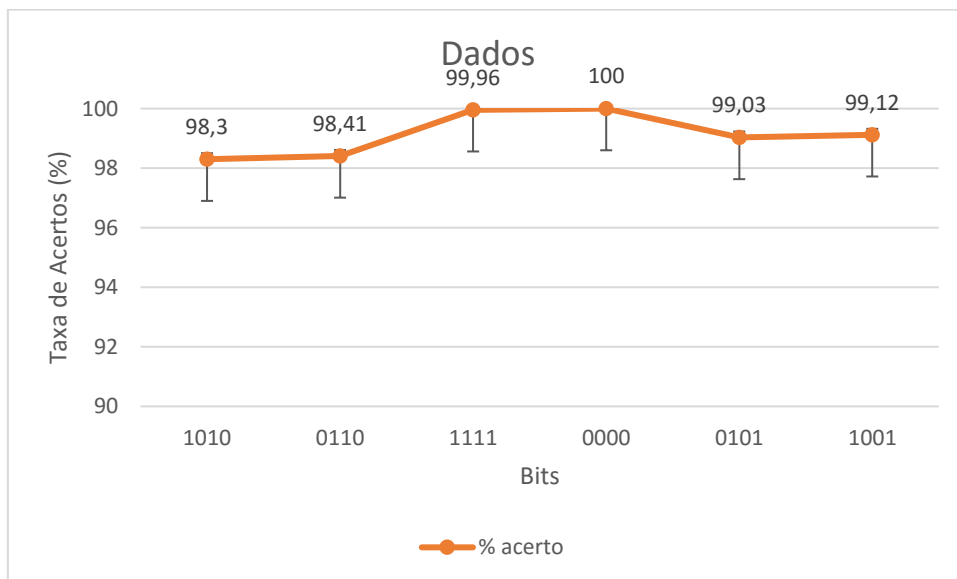


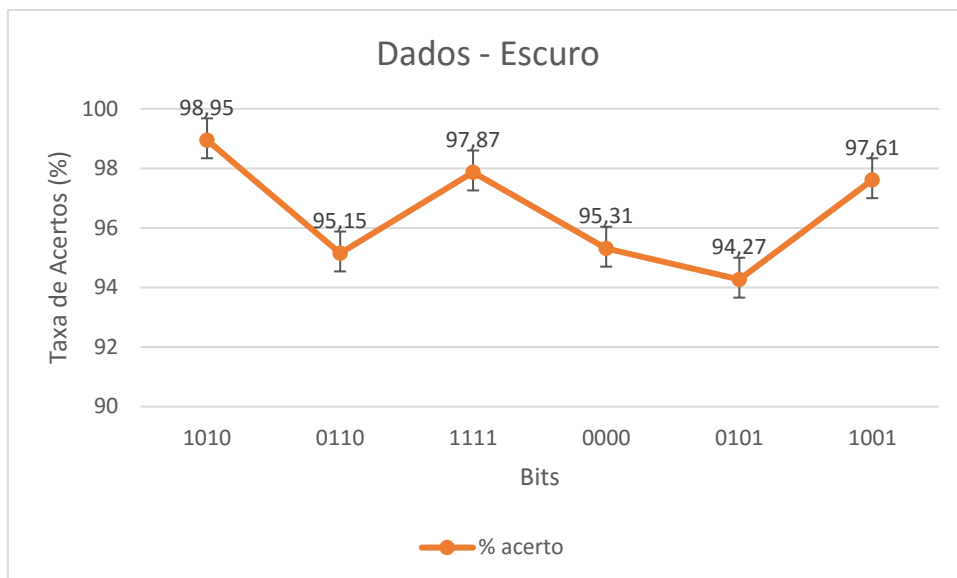


Após todos os testes foram observados a taxa de acertos para cada caso e a BER

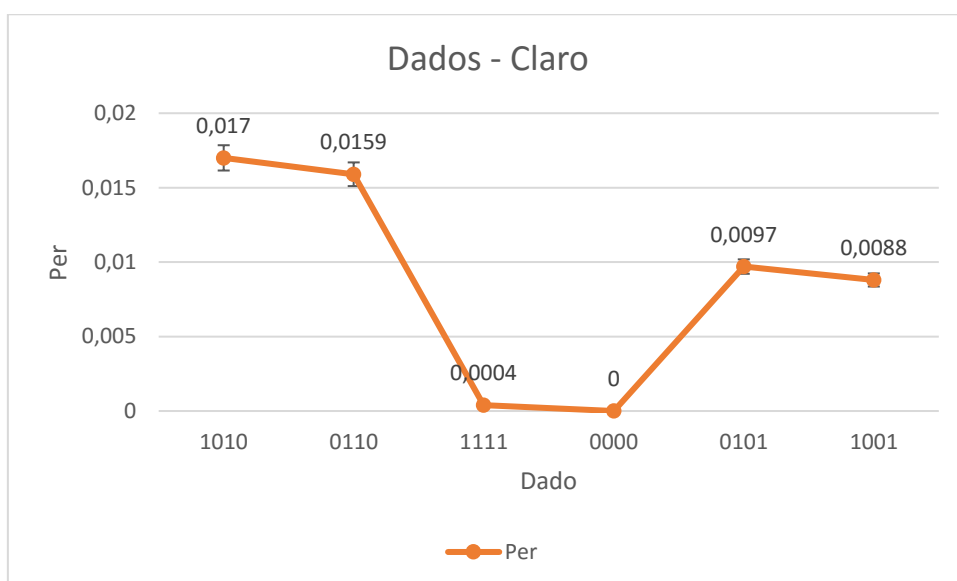
Resultados

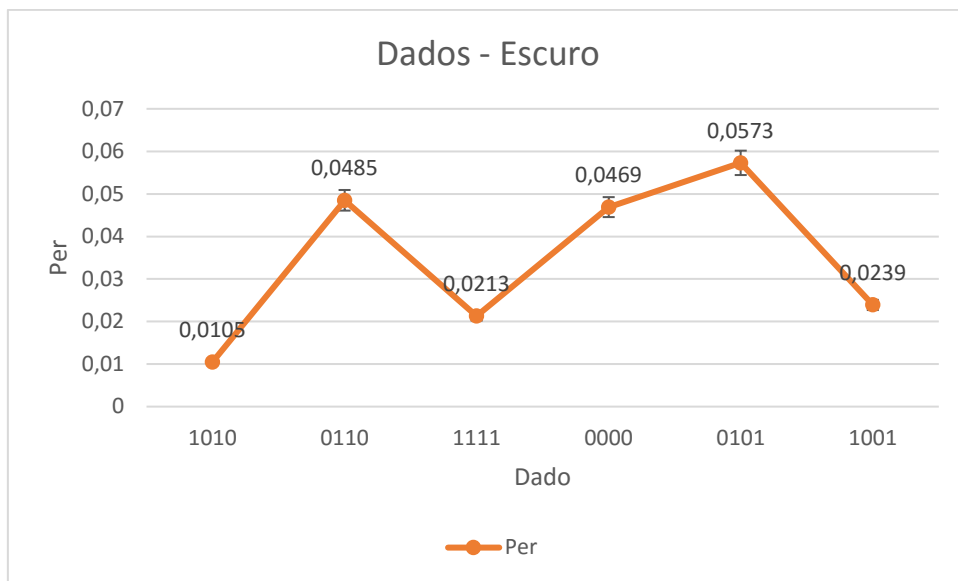
Este teste só compara a taxa de acertos de cada pacote de dados, dentre esses 6 escolhidos, a uma distância de 15 cm, inicialmente para o ambiente com iluminação e após sem iluminação externa.



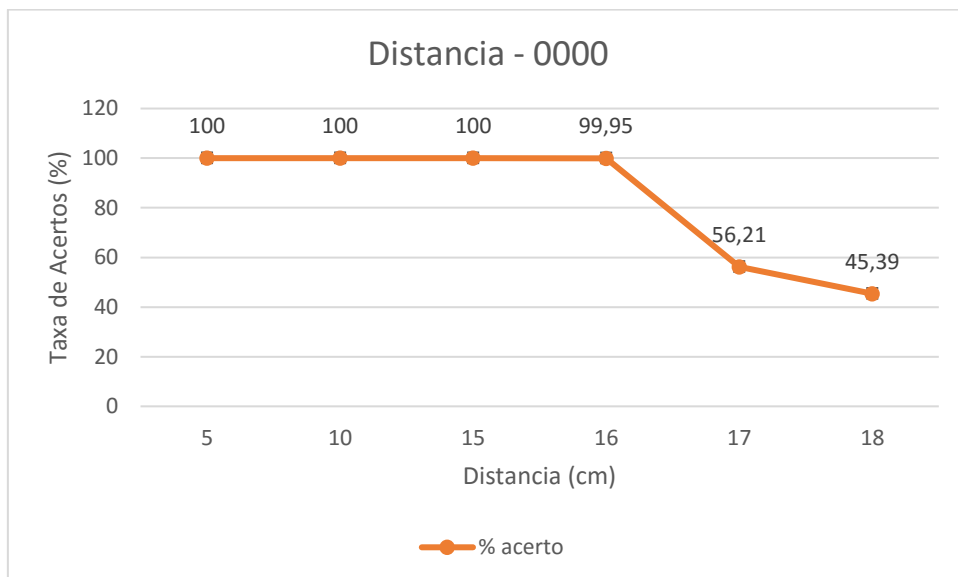


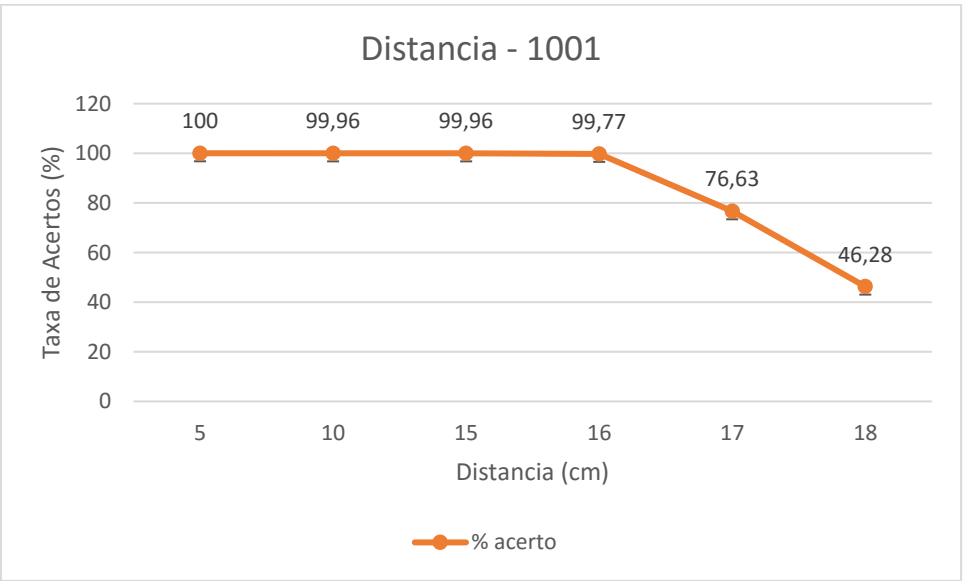
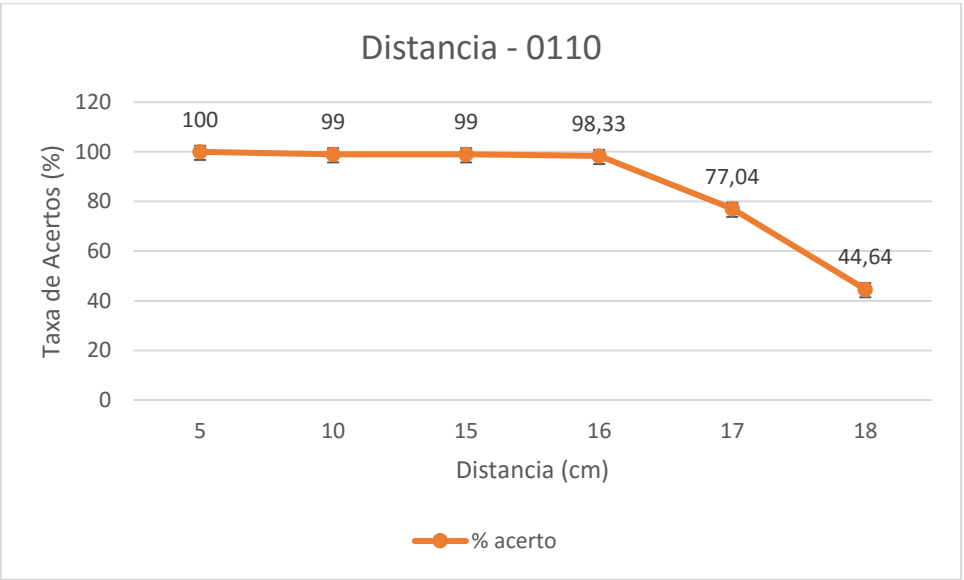
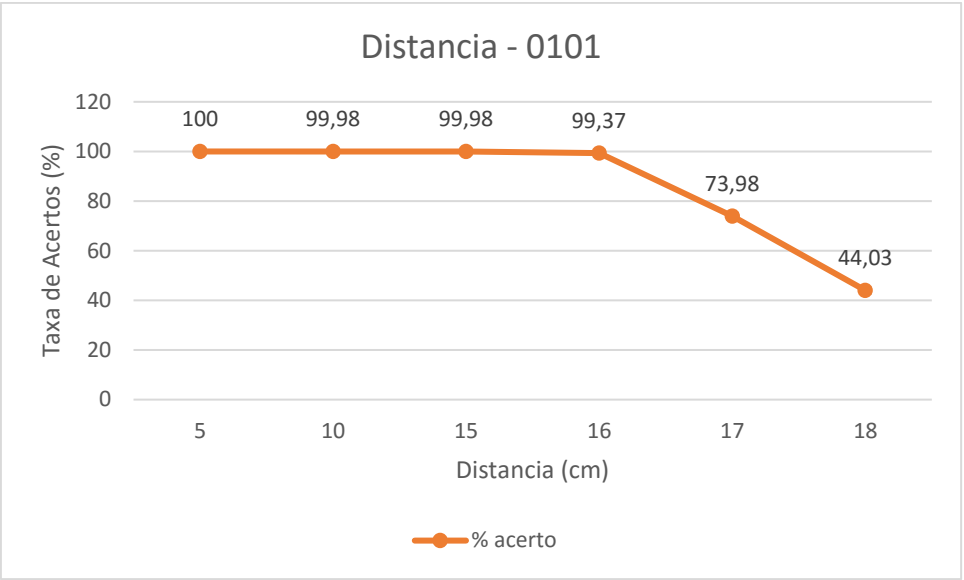
O sistema foi configurado de 2 maneiras em todos os testes, uma para o modo claro e outra para o modo escuro, o desempenho inferior no escuro pode ser atribuído a essa configuração, que foi configurado com uma tolerância muito menor a ruídos para o escuro, a seguir temos a PER para esses dois casos.

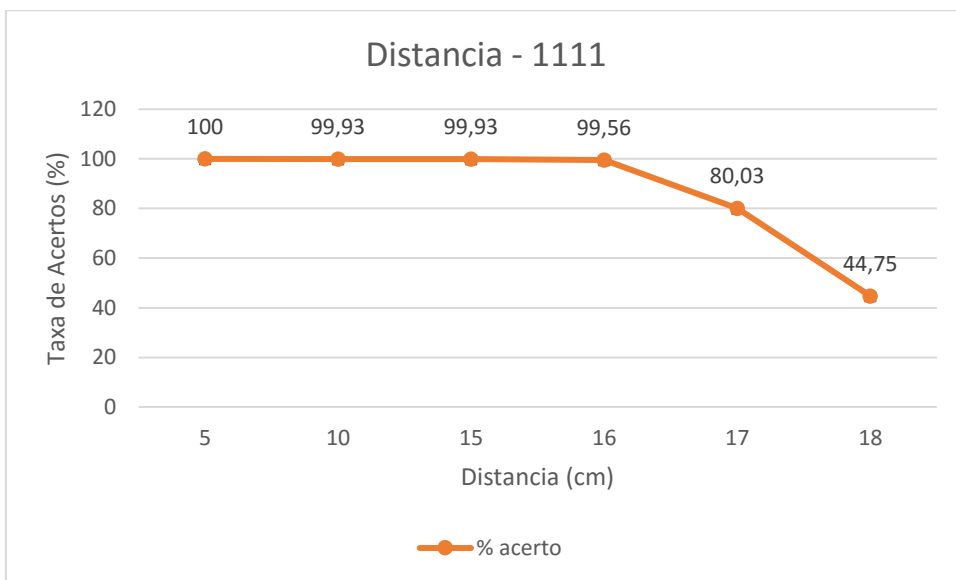
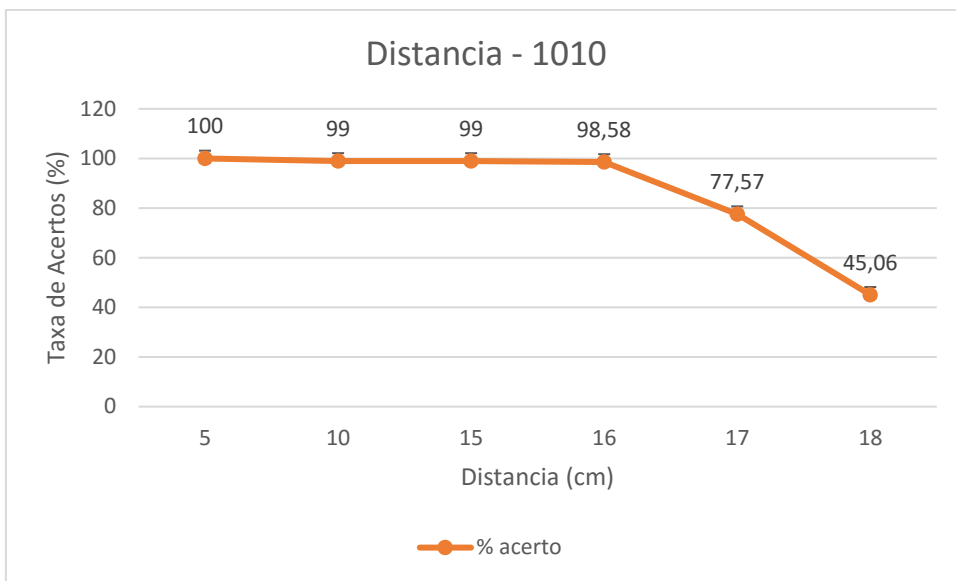




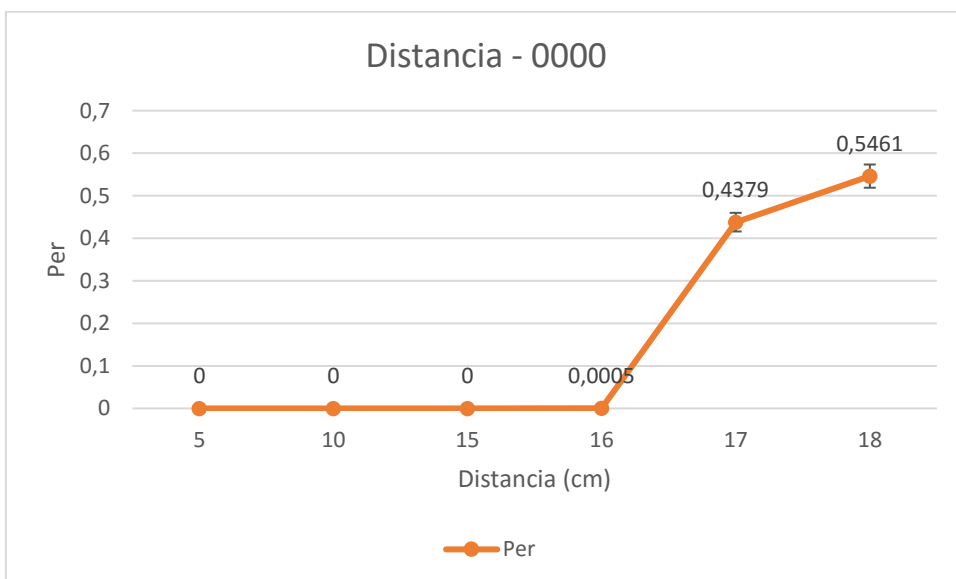
A seguir estão os testes de distância no ambiente com iluminação externa para cada uma das 6 combinações de bits, para o ambiente escuro foi utilizado um limite de luz de 2, para o claro de 30, para ambos os casos foi utilizada uma frequência de 500Hz e pacotes de dados de 4 bits.

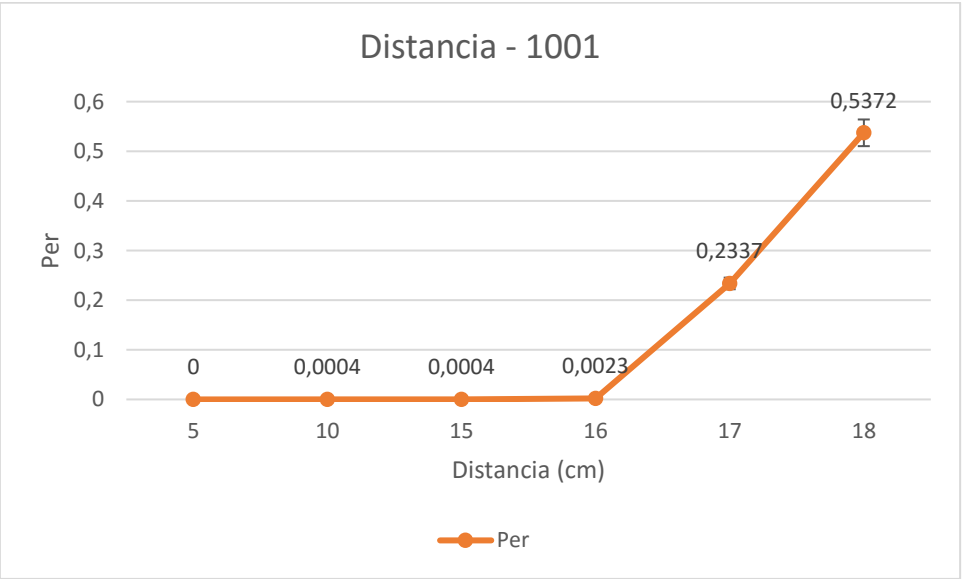
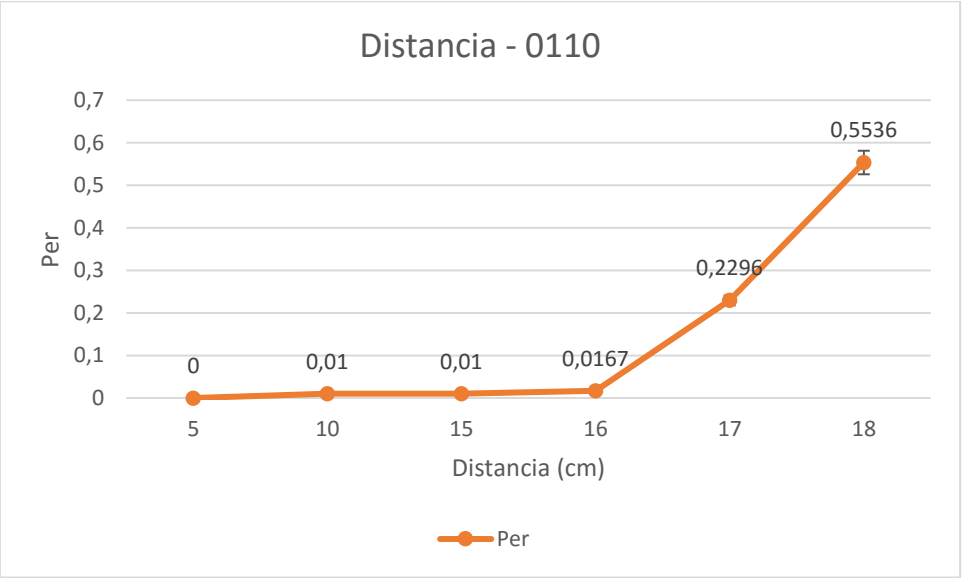
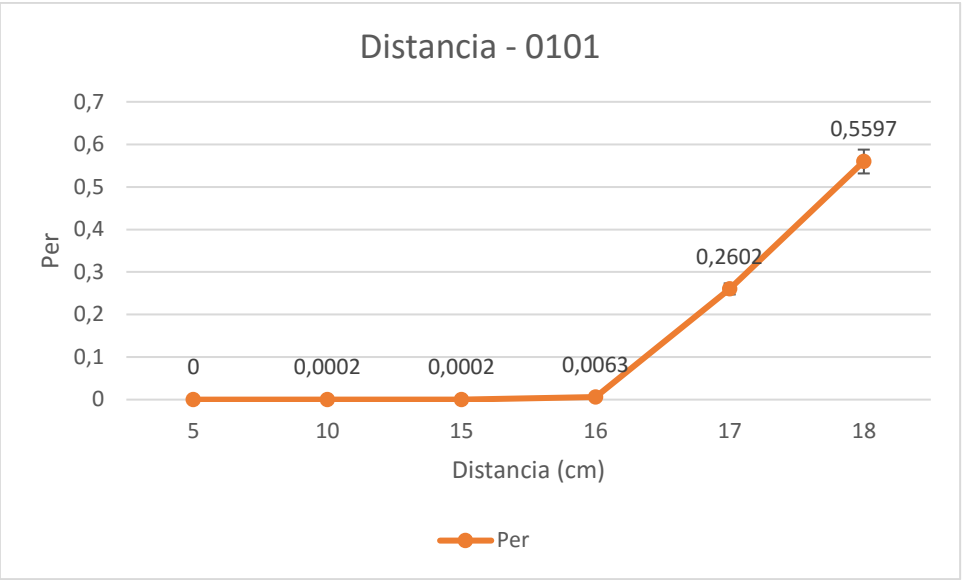


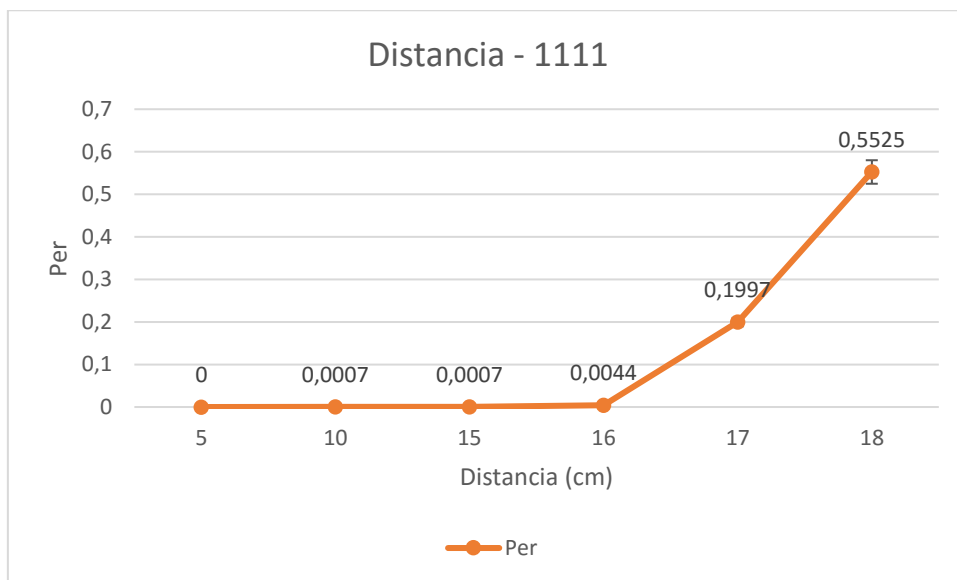
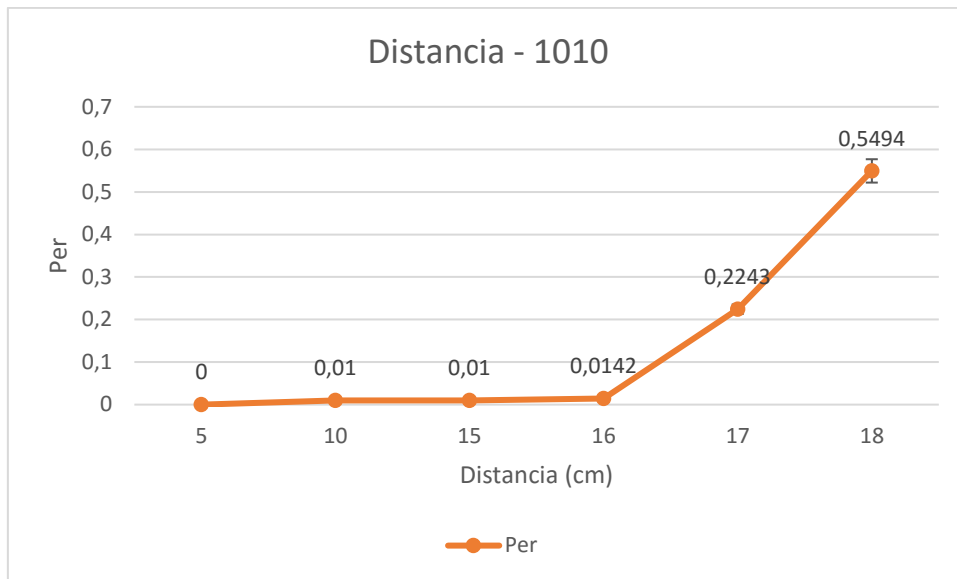




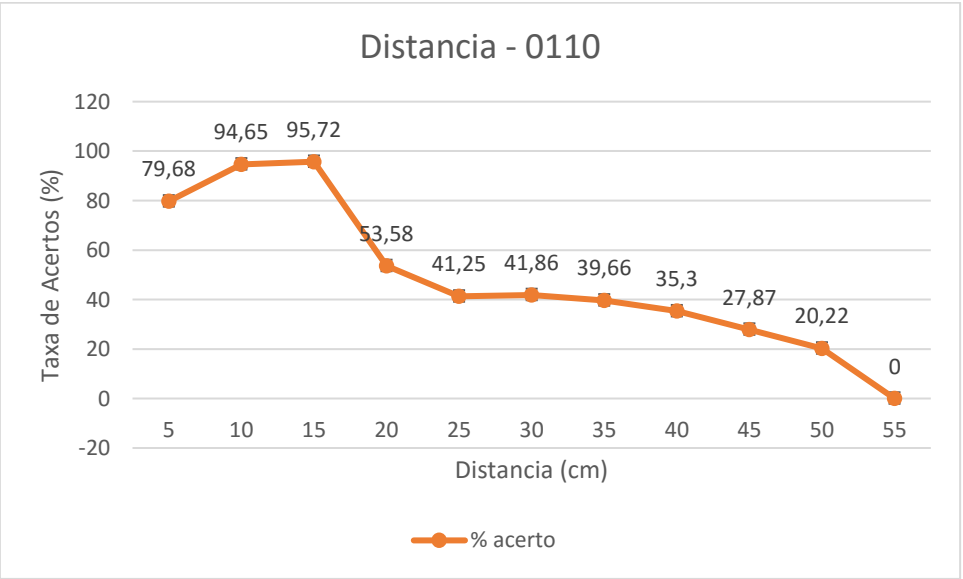
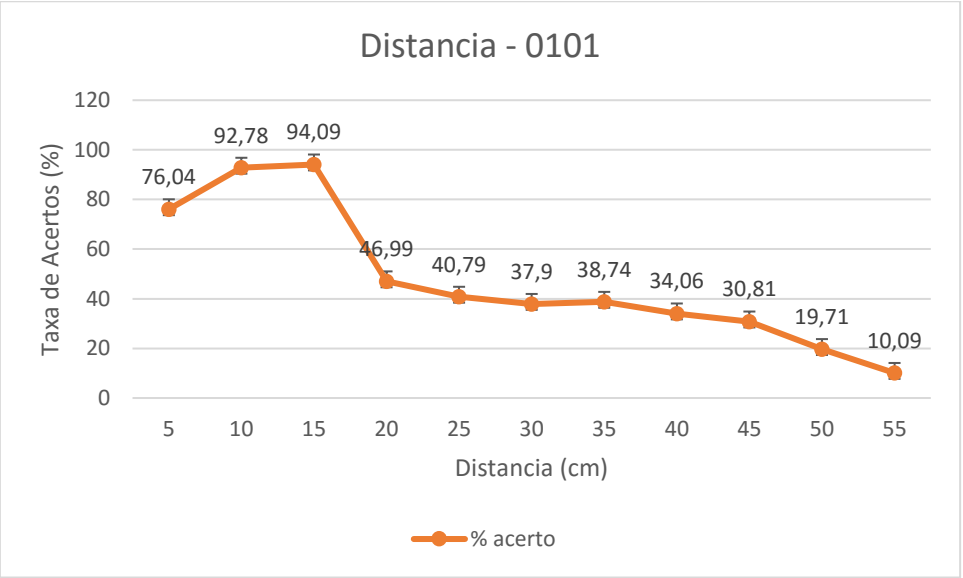
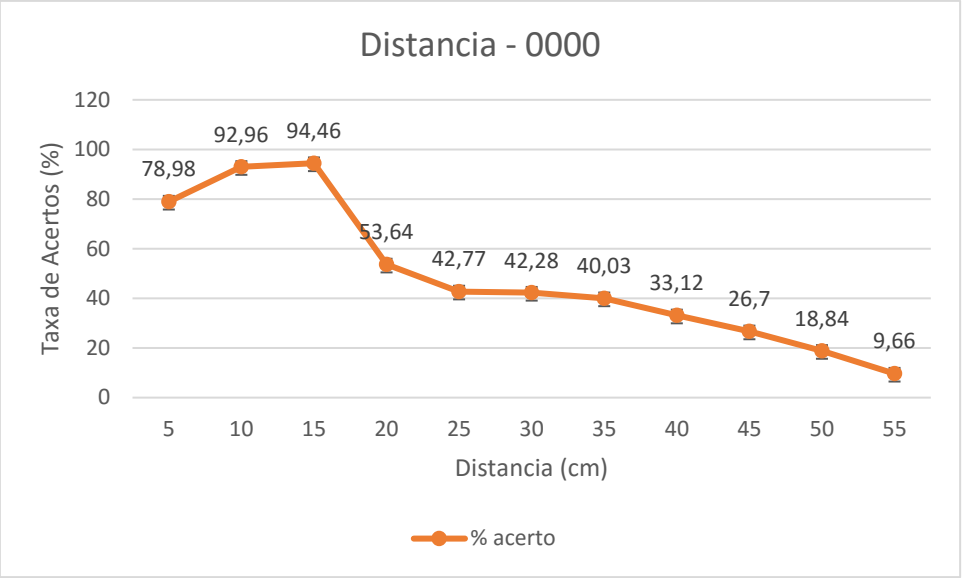
Ambos estão muito próximos, a seguir a PER para o teste de distância no escuro.

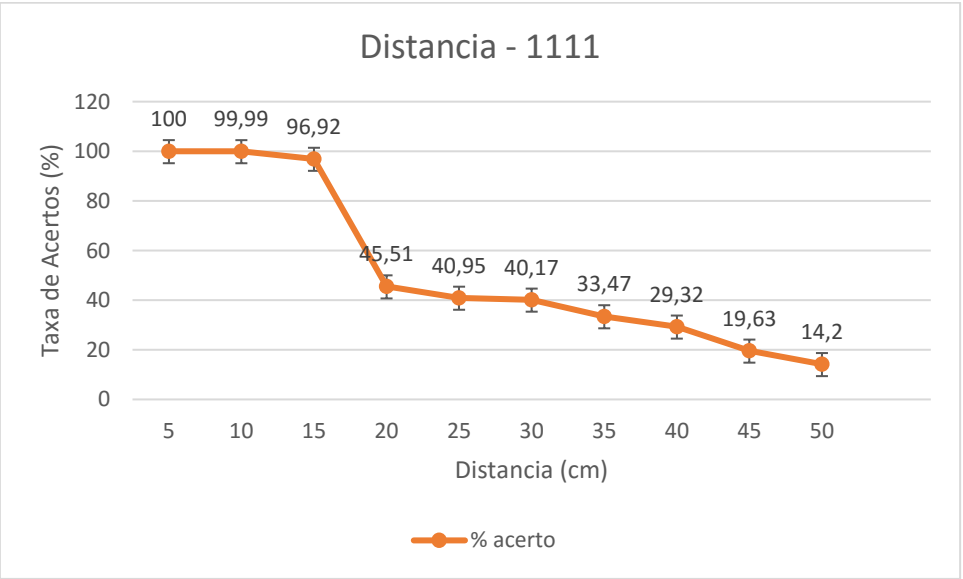
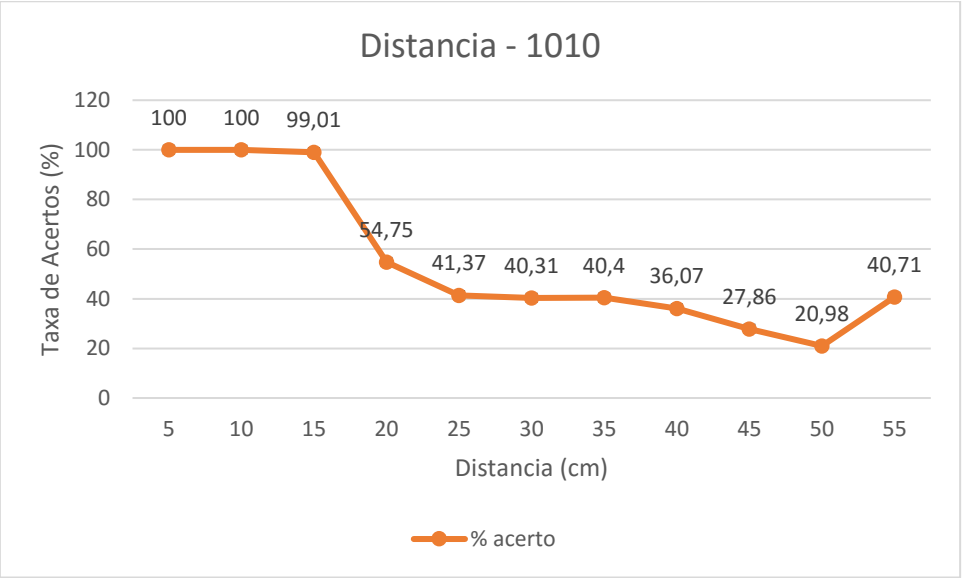
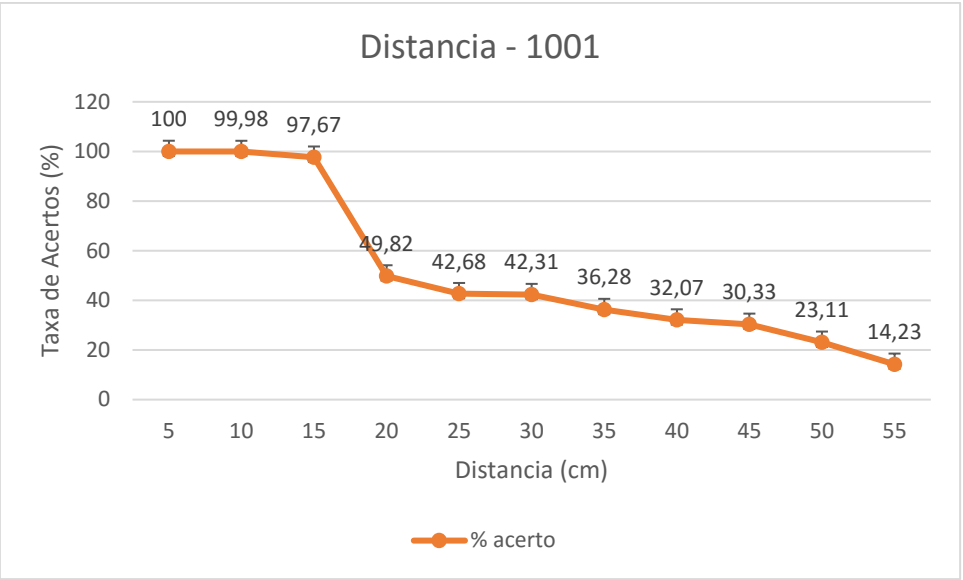


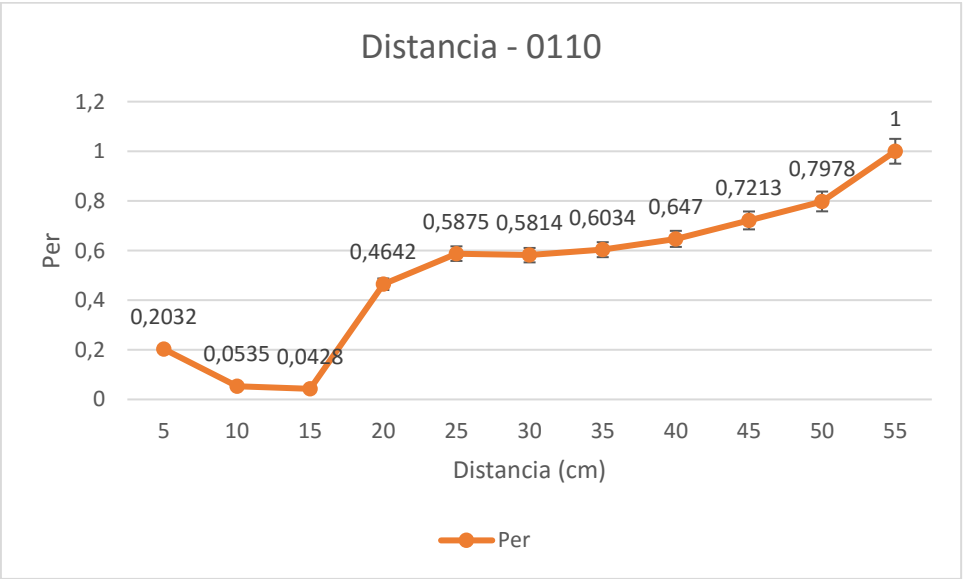
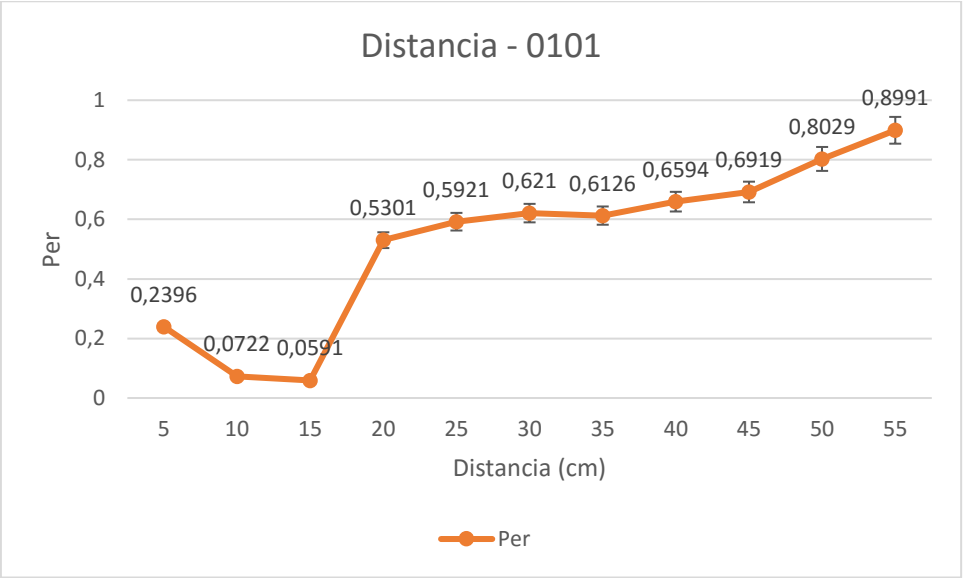
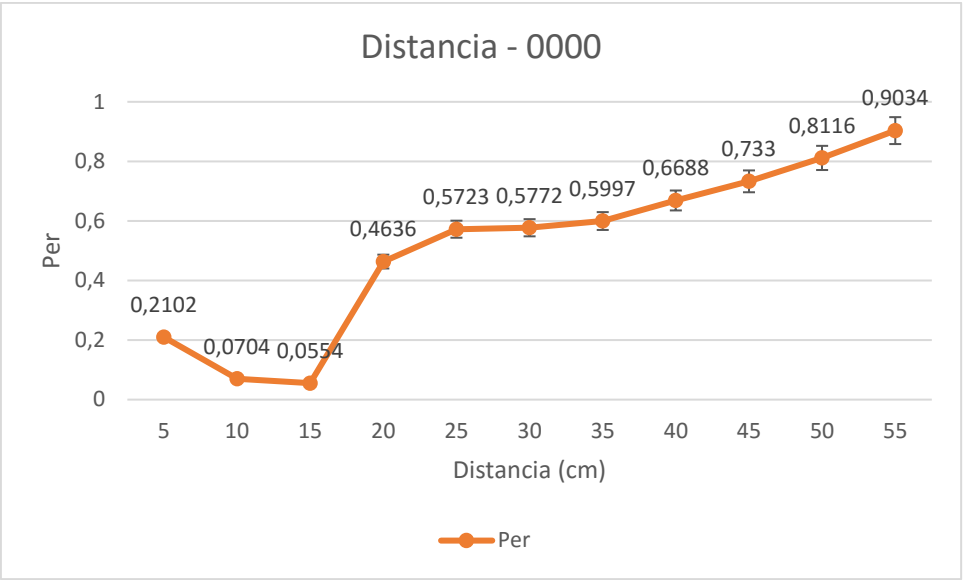


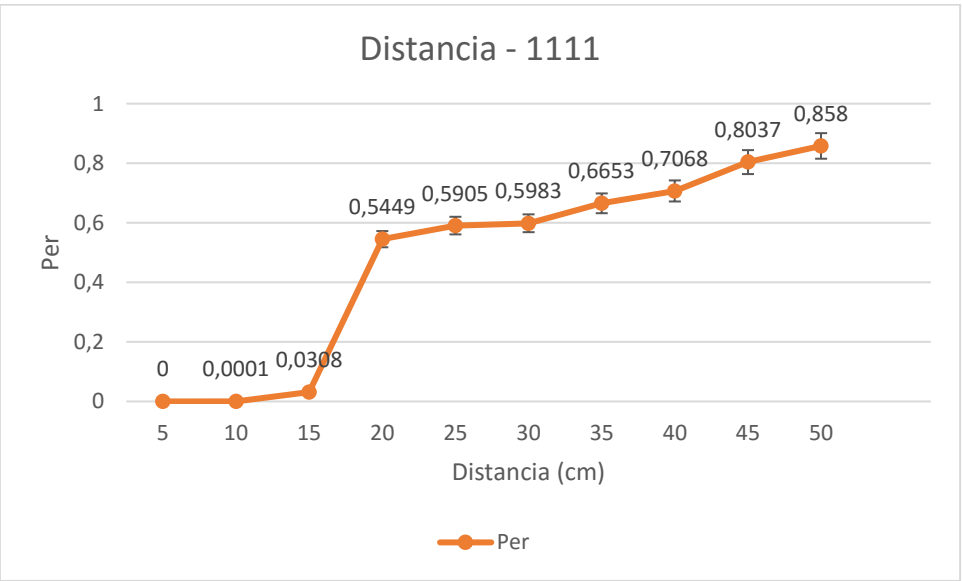
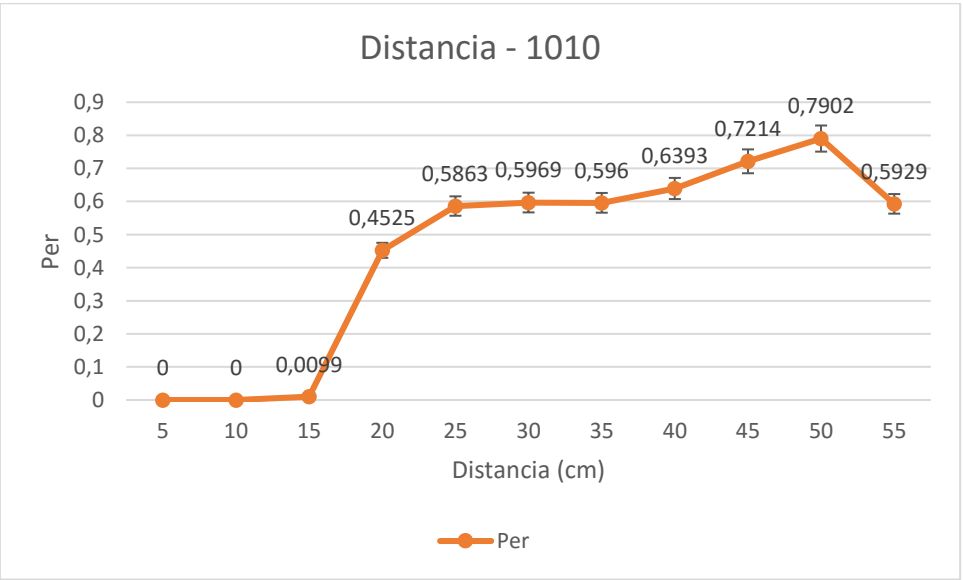
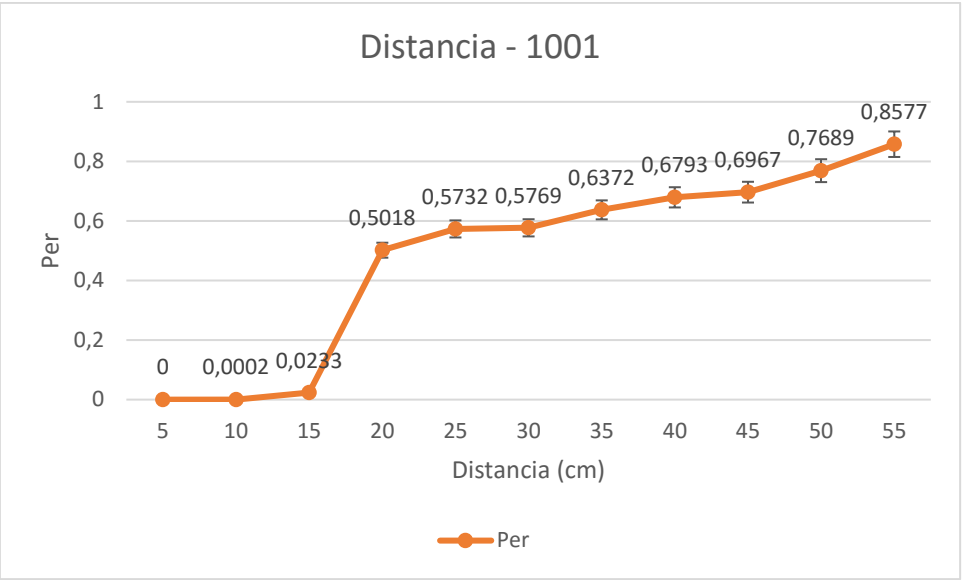


A partir de 18 cm o receptor começou a ter problemas para detectar o início das transmissões no ambiente claro, já no ambiente escuro essa distância foi aumentada para 50 cm, até 55 cm para algumas combinações, mas a partir de 50 começaram a ter problemas de sincronismo entre o transmissor e o receptor para alguns conjuntos de dados.

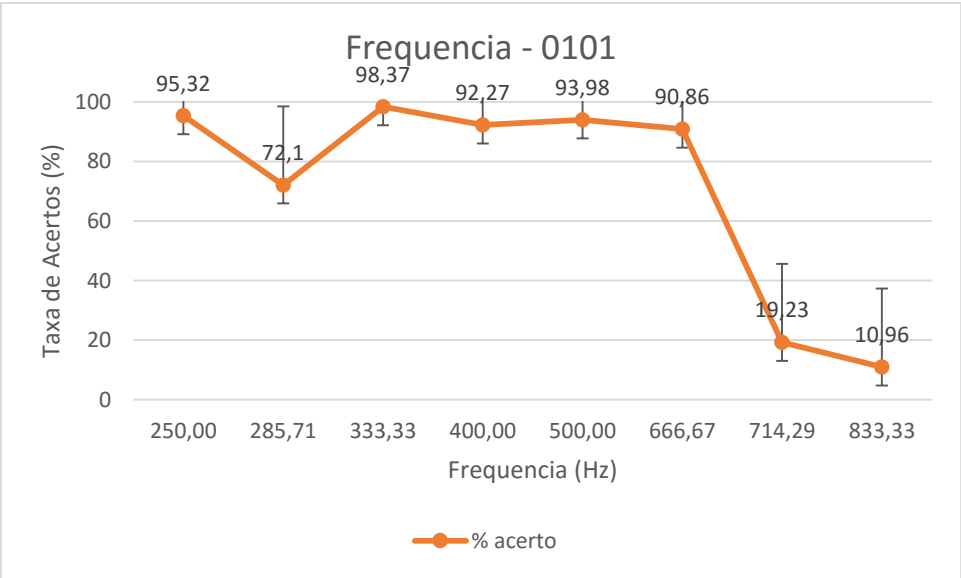
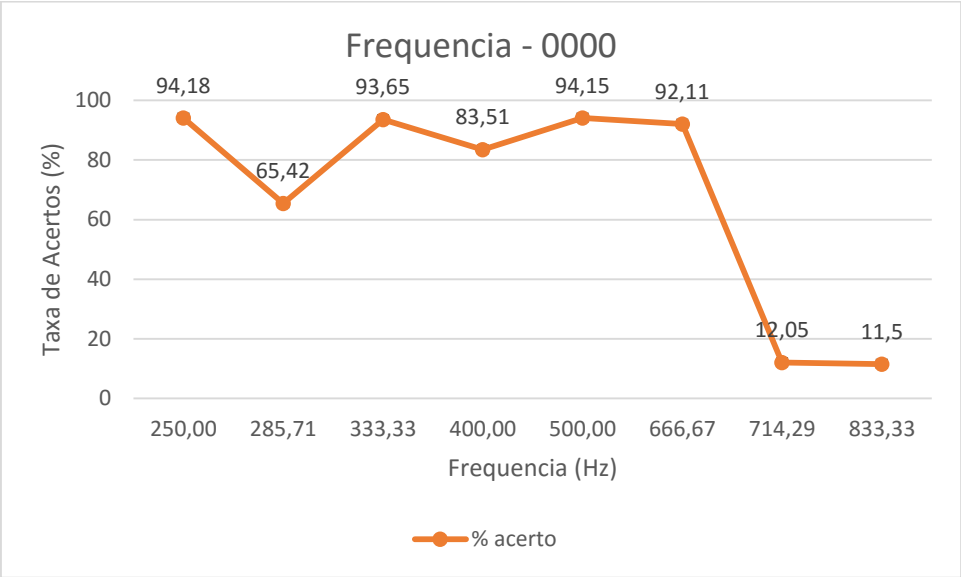


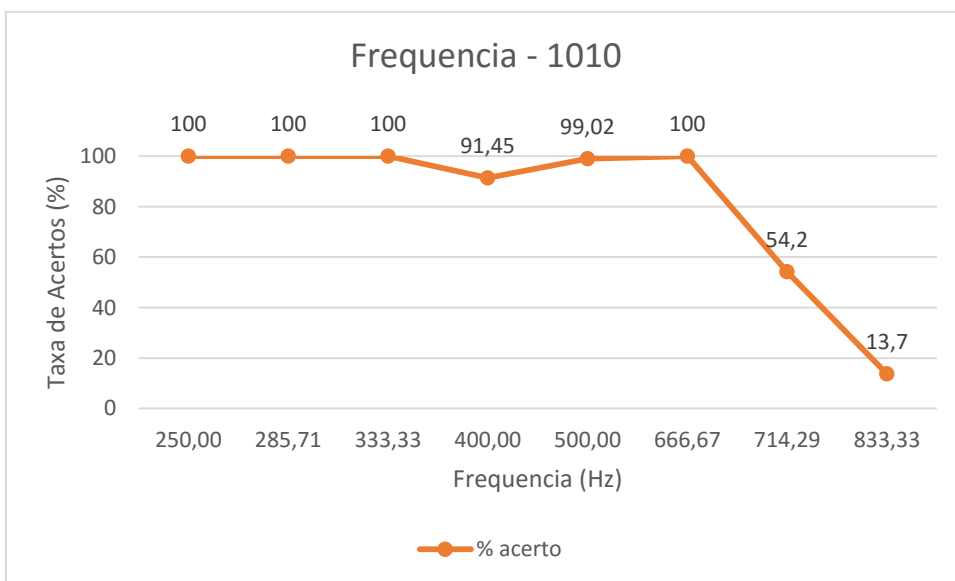
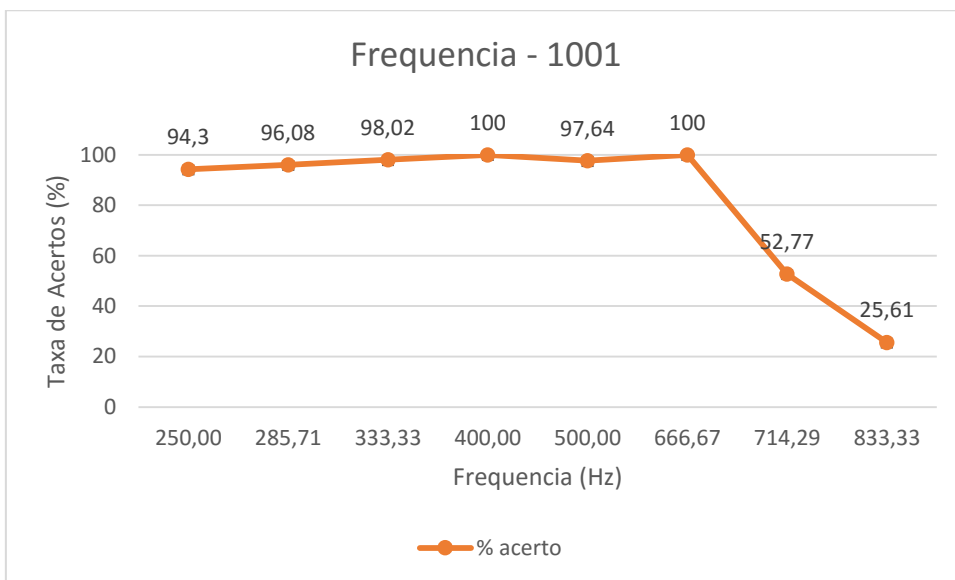
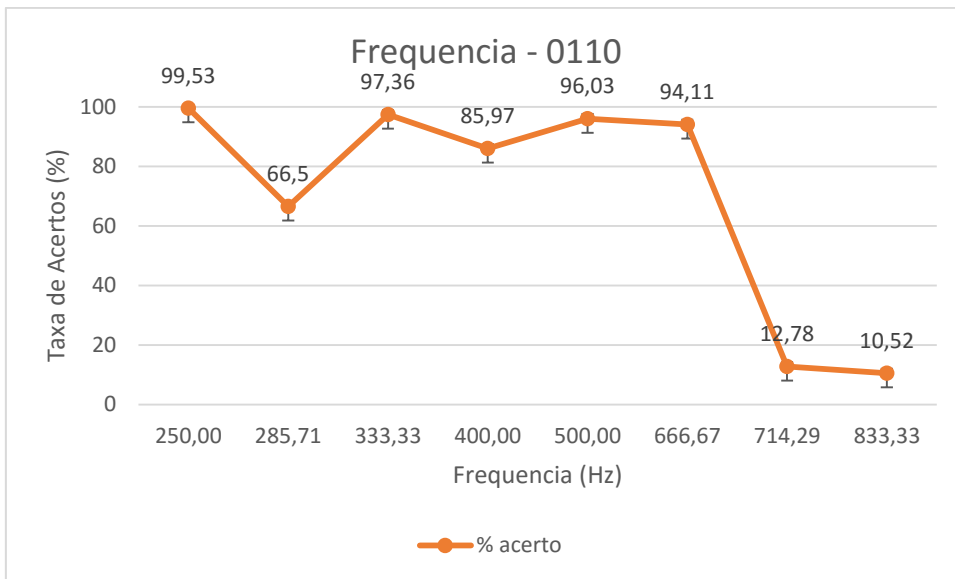


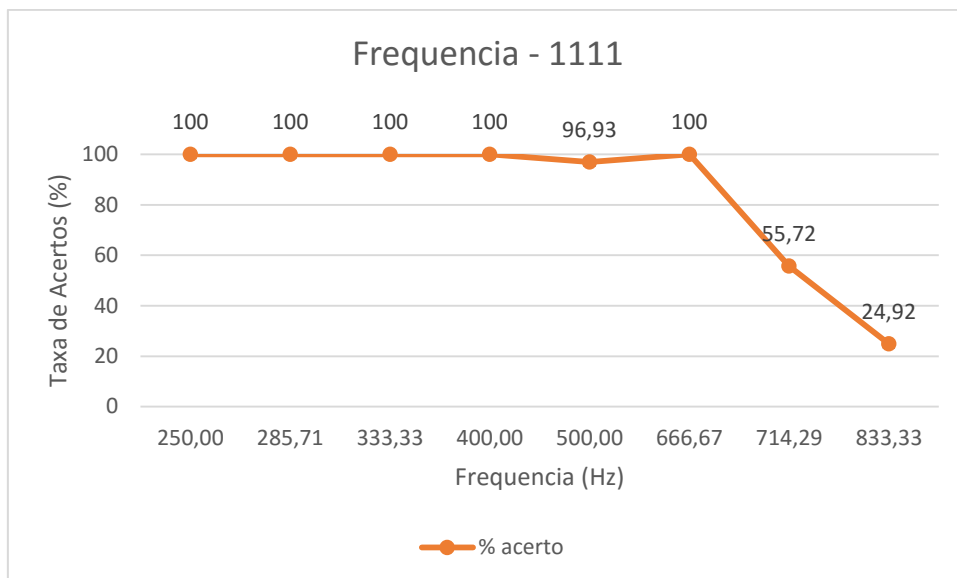




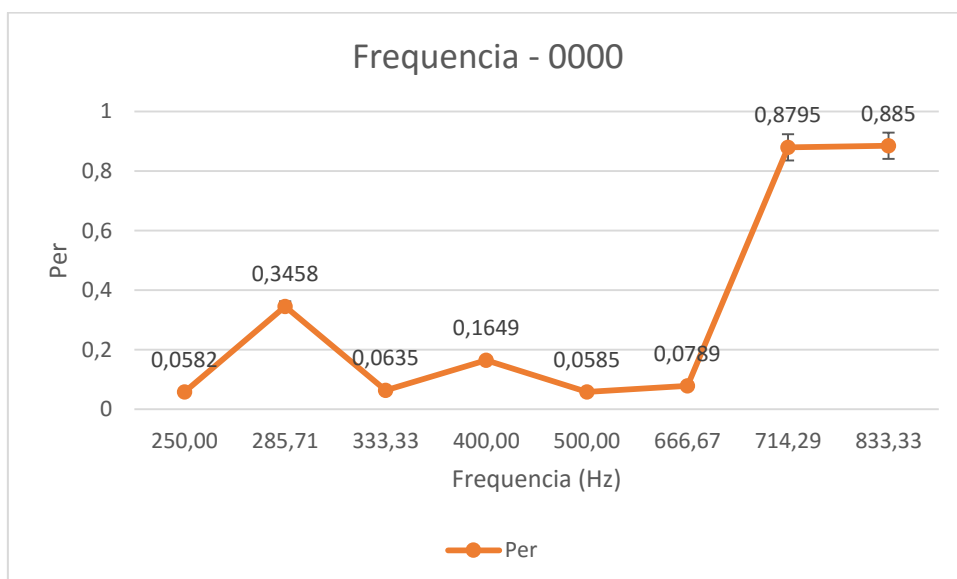
Agora para o teste de frequência, a distância é fixa em 15 cm, os testes são realizados no ambiente escuro com limite de luz definido em 2, e frequência de 500Hz.

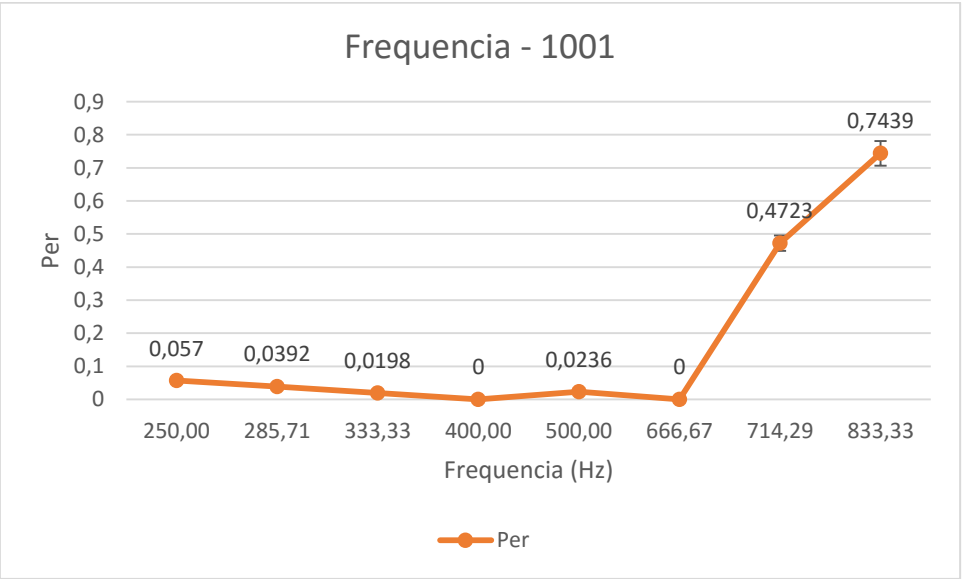
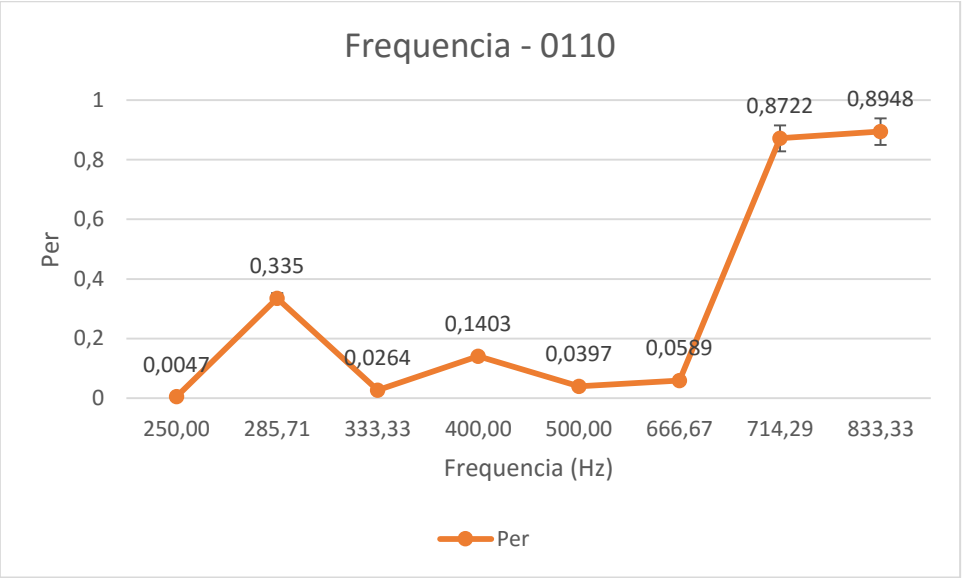
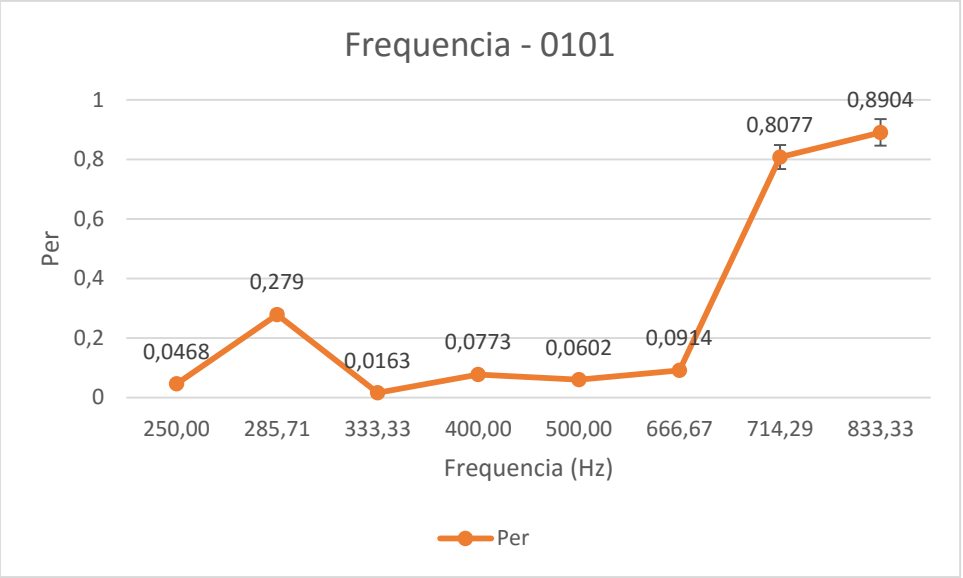


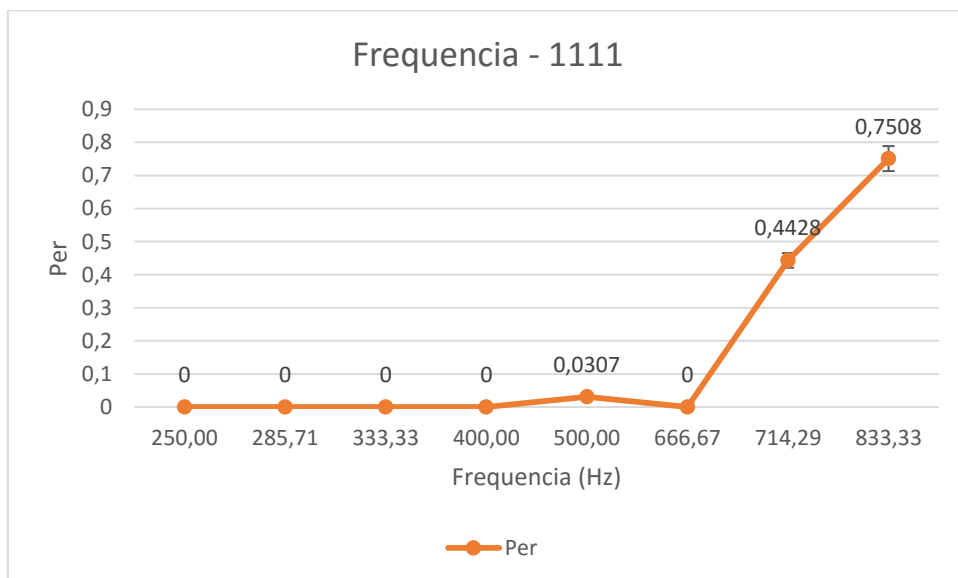
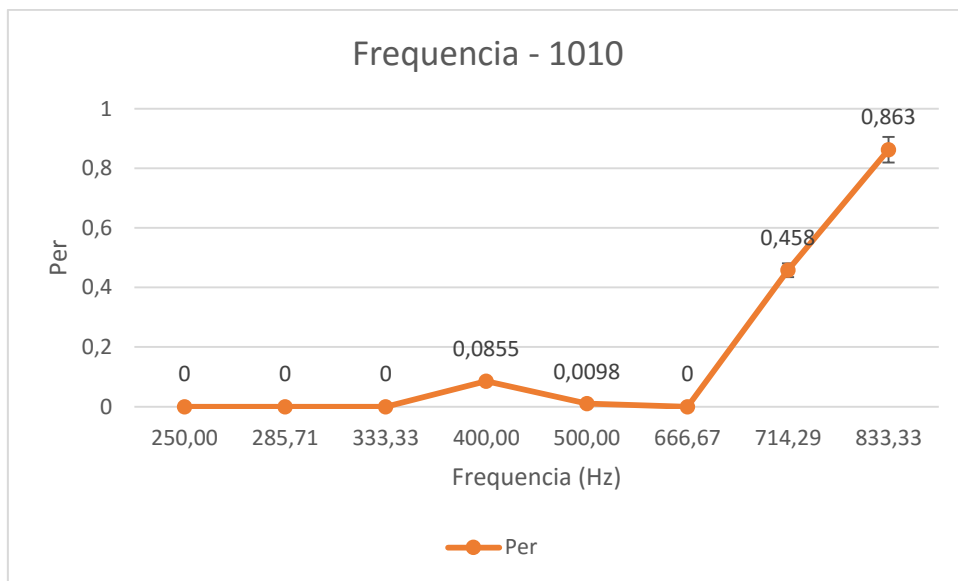




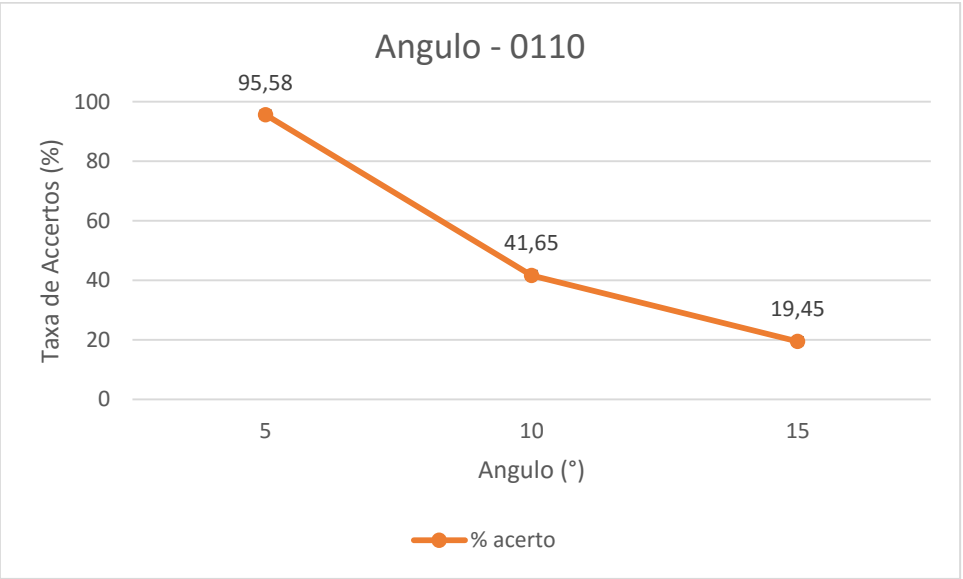
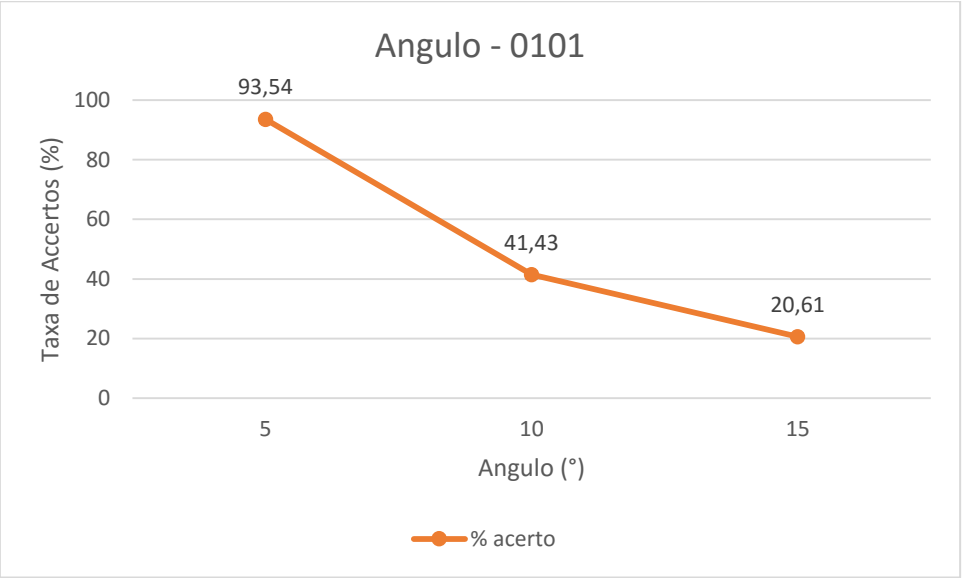
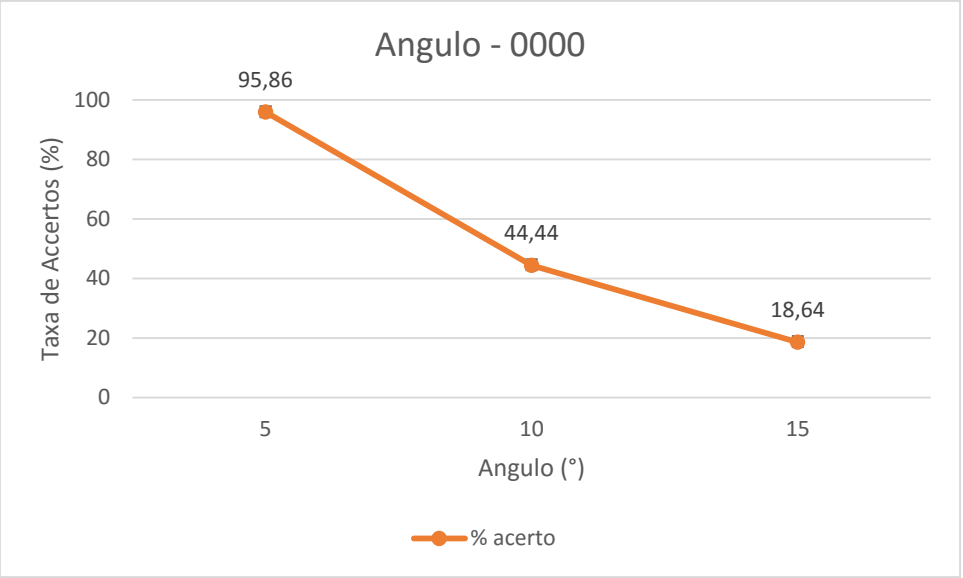
Em todos os testes, não somente para os testes de frequência, antes da comunicação serial começar a transmitir as informações sobre os erros, é definido que não existe nenhum dado a ser enviado para a comunicação serial não influenciar na frequência do sistema, a seguir temos a PER.

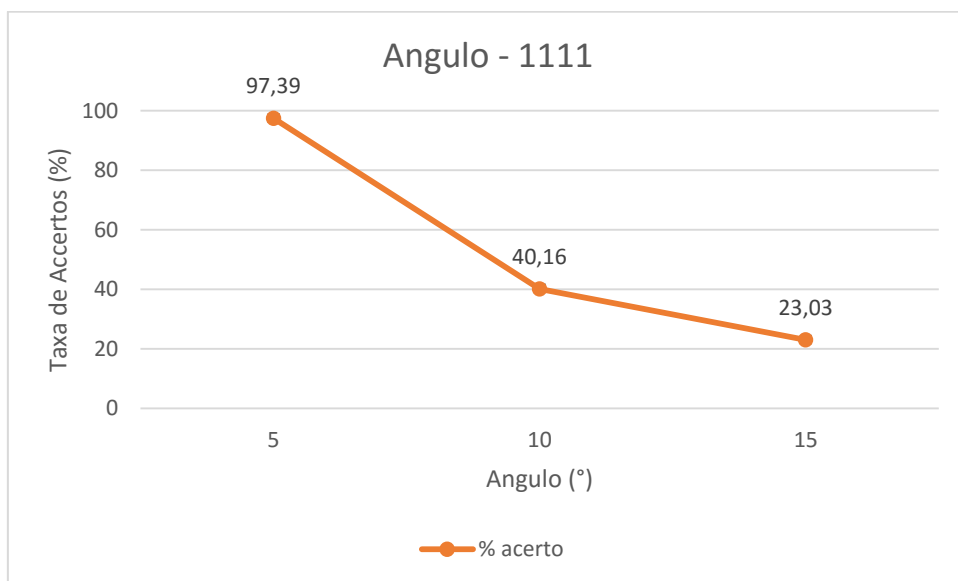
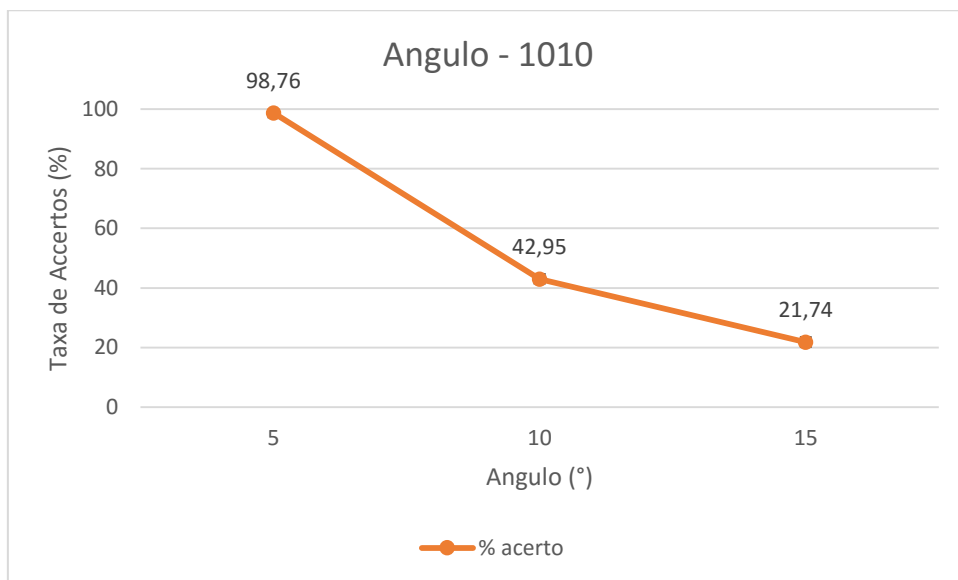
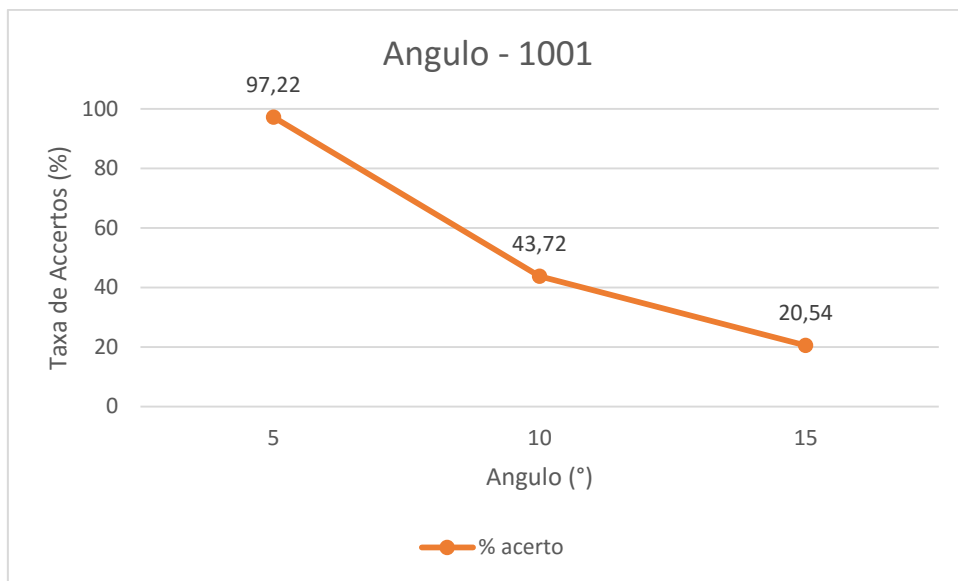




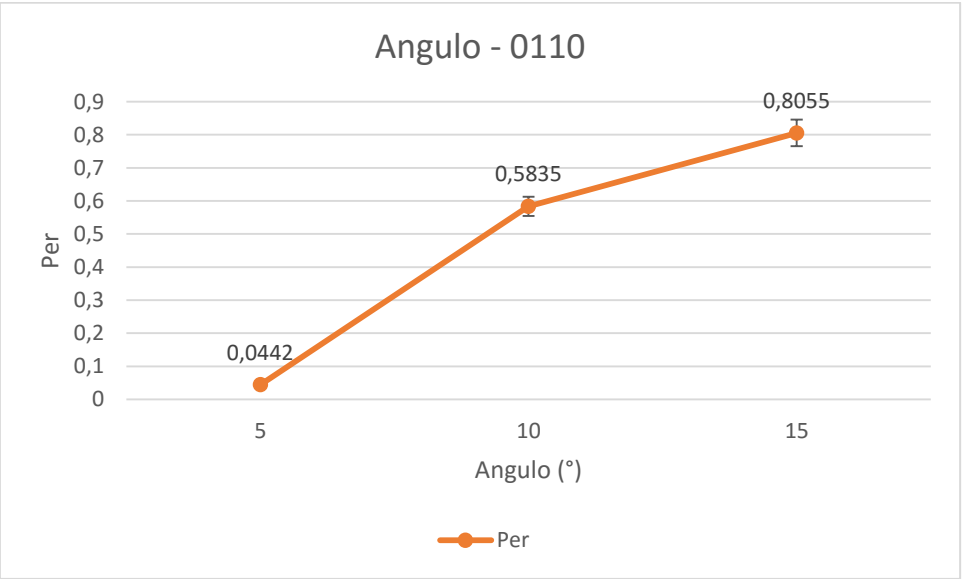
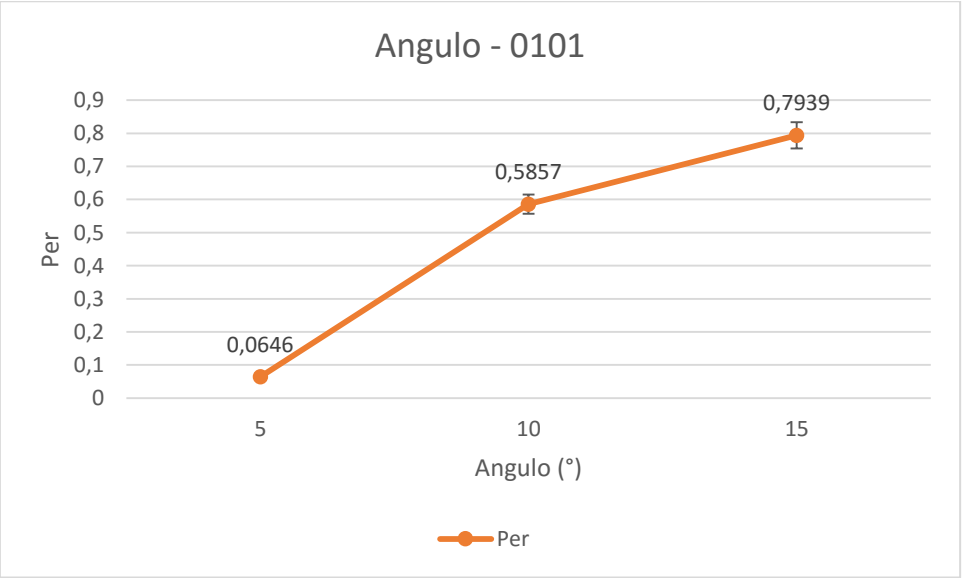
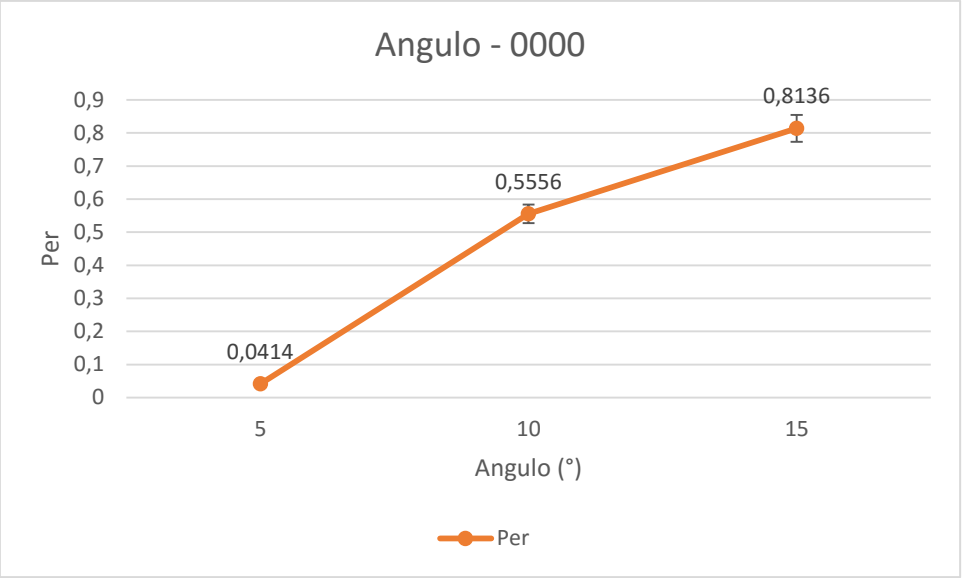


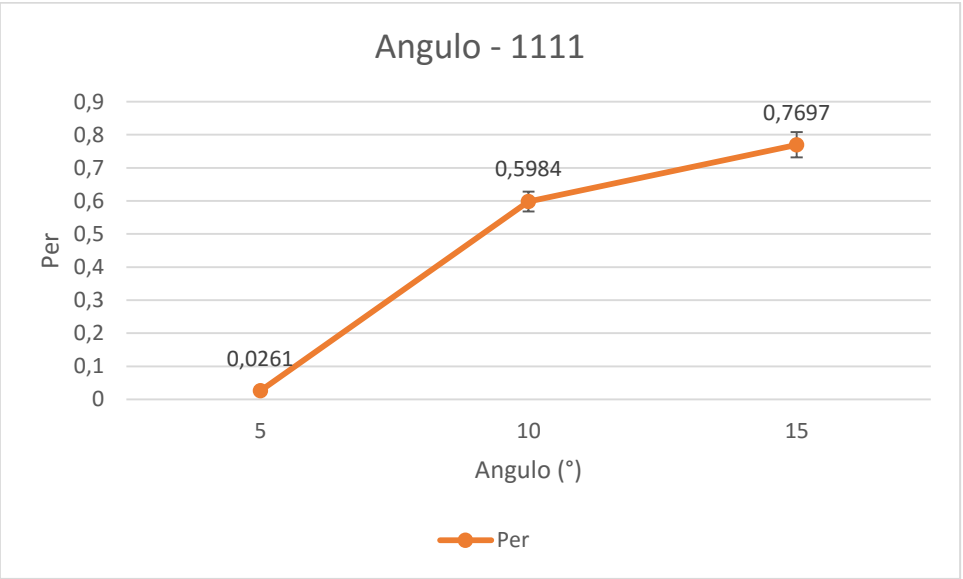
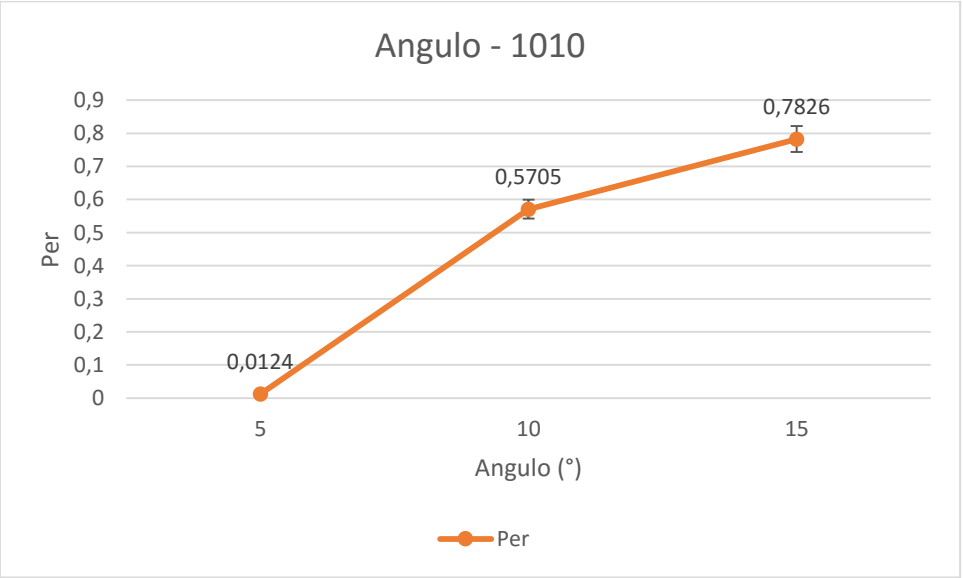
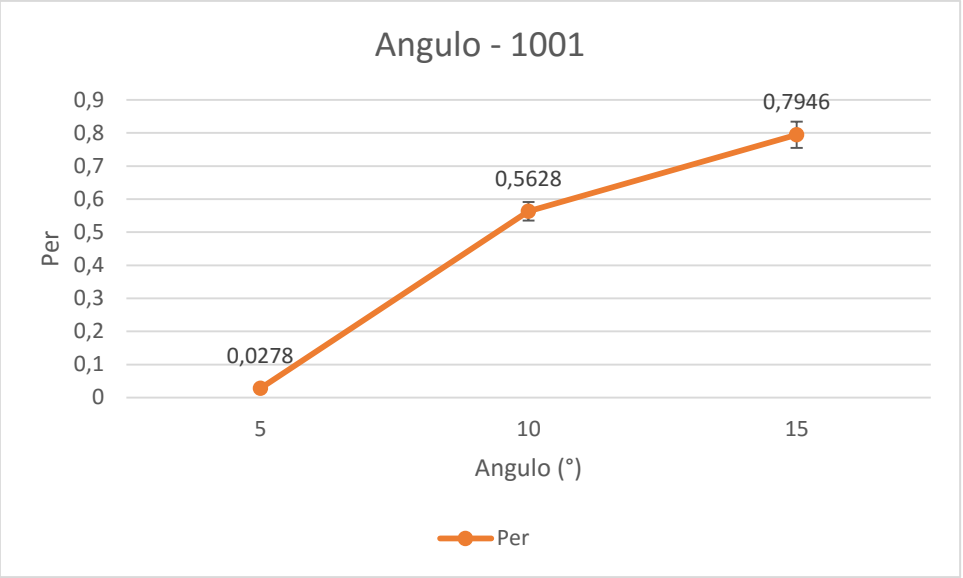
Para os testes de ângulo, a distância continua fixa em 15 cm, o pacote de dados com 4 bits e a frequência de 500 Hz.



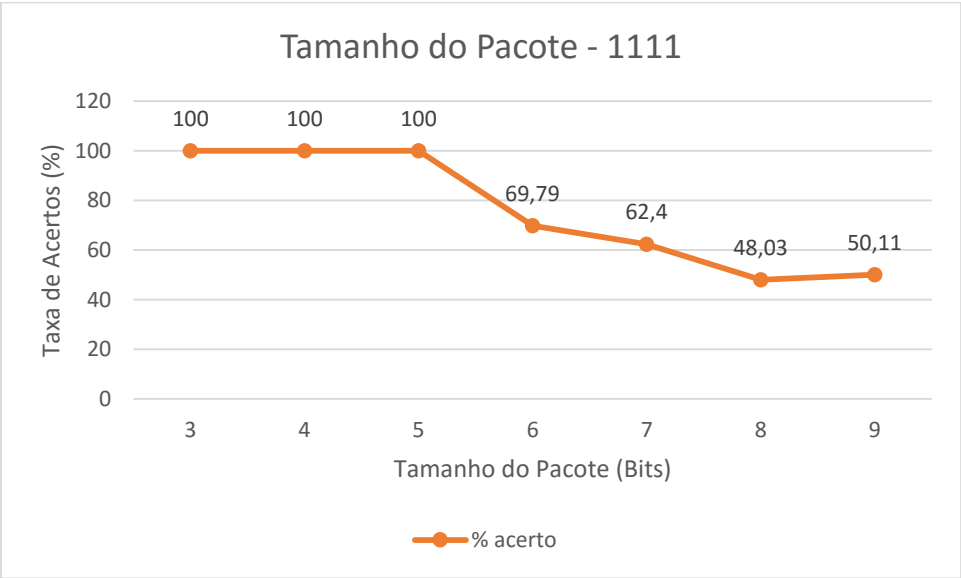
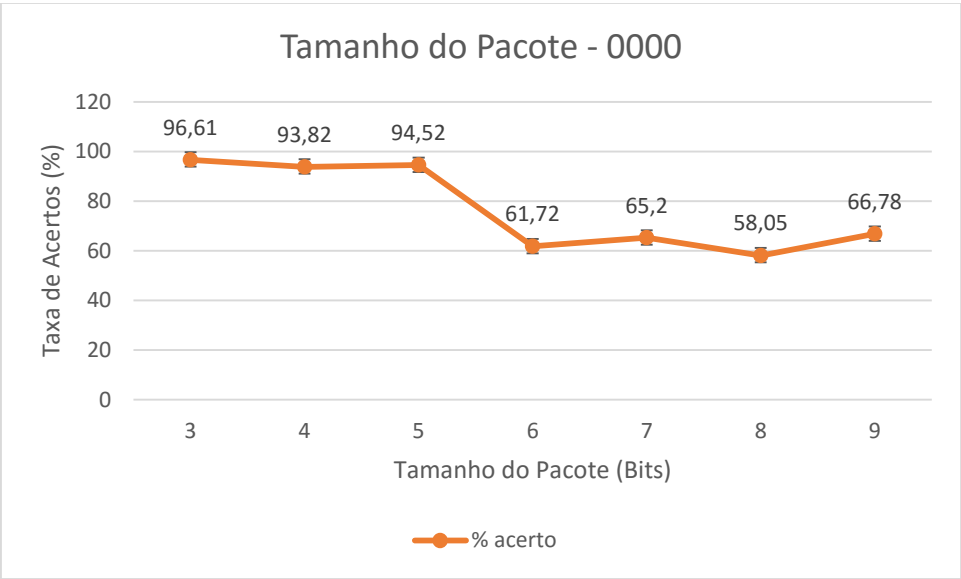


A partir de 15° o receptor começou a ter problemas de sincronismo, a seguir a PER.





E para o teste de tamanho de pacote, a frequência foi reduzida para 333Hz.



E a PER

