

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLÓGICA
ENGENHARIA DE COMPUTAÇÃO**

Leonardo Zanela Breunig

**ESTUDO E IMPLEMENTAÇÃO DE UM SISTEMA DE
COMUNICAÇÃO ÓTICO SEM FIO**

**Santa Maria, RS
2016**

Leonardo Zanela Breunig

**ESTUDO E IMPLEMENTAÇÃO DE UM SISTEMA DE COMUNICAÇÃO ÓTICO
SEM FIO**

Trabalho de Graduação apresentado ao curso de
Engenharia de Computação da Universidade Federal
de Santa Maria (UFSM, RS), como requisito parcial
para a obtenção do grau de
Bacharel em Engenharia de Computação

Orientador: Prof. Dr. Carlos Henrique Barriquello

**Santa Maria, RS
2016**

Leonardo Zanela Breunig

**ESTUDO E IMPLEMENTAÇÃO DE UM SISTEMA DE COMUNICAÇÃO ÓTICO
SEM FIO**

Trabalho de Graduação apresentada ao curso
graduação em Engenharia de Computação da
Universidade Federal de Santa Maria (UFSM,RS),
como requisito parcial para a obtenção do título de
Bacharel em Engenharia de Computação.

Aprovado em 01/07/2016:

Carlos Henrique Barriquello, Dr. (UFSM)
(Presidente/Orientador)

Renato Machado, Dr. (UFSM)
(Coorientador)

Marco Antônio Dalla Costa, Dr. (UFSM)

José Eduardo Baggio, Dr. (UFSM)

Santa Maria, RS
2016

RESUMO

ESTUDO E IMPLEMENTAÇÃO DE UM SISTEMA DE COMUNICAÇÃO ÓTICO SEM FIO

AUTOR: Leonardo Zanela Breunig

ORIENTADOR: Carlos Henrique Barriquello

Em um sistema de comunicação via luz visível, além do uso como iluminação podemos transmitir as informações utilizando a modulação de intensidade na iluminação no transmissor e detecção direta no receptor, esse tipo de sistema possui faixa de livre utilização no espectro eletromagnético, além de ser mais seguro, já que a luz não ultrapassa paredes como os sinais de rádio. A implementação desse sistema utiliza componentes de baixo custo, como um micro controlador utilizado em sistemas embarcados, um LED e um fotodiodo. Com o sistema implementado, é possível verificar o seu funcionamento em diferentes situações, como por exemplo a influência da distância nas transmissões, dos diferentes tamanhos de pacote, da frequência e do ângulo de incidência entre o transmissor e o receptor.

Palavras-chave: Sistemas Embarcados, Comunicação via Luz Visível, Taxa de Erro de Pacote.

ABSTRACT

STUDY AND IMPLEMENTATION OF A SYSTEM OF OPTIC WIRELESS COMMUNICATION

AUTOR: Leonardo Zanela Breunig

ORIENTADOR: Carlos Henrique Barriquello

In a communication system using visible light, and the use as lighting can transmit information using the modulation intensity lighting at the transmitter and direct detection at the receiver, this type of system has free use range in the electromagnetic spectrum, as well as being more safe, since light does not exceed wall as radio signals. The implementation of this system uses low cost components such as a micro controller used in embedded systems, an LED and a photodiode. With the system in place, you can check its operation in different situations, such as the influence of the distance transmissions, the different packet sizes, the frequency and angle of incidence between the transmitter and receiver.

Keywords: Embedded Systems, Communication via Visible Light, Packet Error Rate.

LISTA DE ABREVIATURAS E SIGLAS

AM	Modulação em Amplitude
ASK	Modulação por Chaveamento de Amplitude
BER	Taxa de Erro de Bit
DC	Corrente Contínua
FM	Modulação em Frequência
FSK	Modulação por Chaveamento de Frequência
FSM	Máquina de Estados Finito
IEEE	Instituto de Engenheiros Eletricistas e Eletrônicos
IoL	Internet via Luz
ISM	Bandas de Rádio Industriais, Científicas e Médicas
LAN	Rede de Área Local
LED	Diodo Emissor de Luz
OOK	Chaveamento Liga-Desliga
PER	Taxa de Erro de Pacote
PM	Modulação em Fase
PSK	Modulação por Chaveamento de Fase
RF	Rádio Frequência
VLC	Comunicação via Luz Visível

SUMÁRIO

1	INTRODUÇÃO.....	08
1.1	Motivação.....	08
1.2	Objetivos.....	09
1.3	Organização do trabalho.....	09
2	ELEMENTOS BÁSICOS DE UMA REDE DE COMUNICAÇÃO.....	09
2.1	Modos de transmissão.....	09
2.2	Largura de banda.....	10
3	MODULAÇÃO.....	11
3.1	Definição de símbolo e bit.....	12
3.2	Codificação ou modulação digital.....	13
3.3	Capacidade máxima de um sistema de comunicação.....	14
3.4	VLC e avaliação de erros.....	14
4	DESENVOLVIMENTO.....	16
4.1	Transmissor.....	18
4.2	Receptor.....	19
4.3	Testes.....	20
5	RESULTADOS.....	23
6	CONCLUSÃO.....	36
6.1	Sugestões para trabalhos futuros.....	37
	REFERÊNCIAS.....	38
	APÊNDICES.....	39

1 INTRODUÇÃO

A princípio, podemos considerar as redes de comunicação como sendo os ambientes onde um conjunto de dispositivos, enlaces de comunicação e pacotes de software permitem que pessoas e equipamentos possam trocar informações (DANTAS,2002).

Em um sistema de comunicação ótico sem fio as informações podem ser transmitidas utilizando a luz visível com modulação de intensidade no transmissor e detecção direta no receptor, para isso podem ser utilizados Diodos Emissores de Luz (LEDs), que devido a sua evolução recente apresentam grande potencial para iluminação, pois possuem longa vida útil e maior eficiência luminosa, além de possuírem baixo custo, podem ser usados como um dispositivo de iluminação, por isso podem ser muito uteis em redes domesticas ou sistemas de iluminação inteligentes por exemplo, além de outros cenários com sensibilidade de custo e curta distância de operação.

A Comunicação via Luz Visível (VLC) tem sido muito estudado como uma alternativa as comunicações sem fio para ambientes fechados, como por exemplo, o padrão 802.11 para Redes de Área Local (LANs) sem fio, que operam nas bandas não licenciadas como as Bandas de Rádio Industriais, Cientificas e Medicas (ISM), (por exemplo, 902-928 MHz, 2,4-2,5 GHz, 5,725-5,825 GHz). Todos os dispositivos têm permissão para usar esse espectro, desde que limitem sua potência de transmissão para permitir que diferentes dispositivos coexistam. Naturalmente, isso significa que os rádios 802.11 podem estar competindo com telefones sem fio, aparelhos para abrir portas de garagens e fornos de micro-ondas (TANENBAUM,2011).

Além disso, o VLC é uma alternativa para resolver o problema de trituração de espectro sem fio de comunicações Rádio Frequência (RF) e tem o potencial de permitir a criação de uma nova geração de sistemas de redes (GIUSTIANO; WANG, 2015).

1.1 Motivação

Um sistema de comunicações ótico sem fio possui faixa de livre utilização no espectro eletromagnético, baixo custo, possui baixa complexidade além de poder ser utilizado como um sistema de iluminação eficiente.

A segurança também é uma grande vantagem desta tecnologia, pois a luz não pode penetrar paredes, o que mantém o sinal seguro dentro do ambiente, diferente de ondas de rádio, que além de pouca segurança e eficiência, pois os sinais de rádio necessitam de grande energia e podem ser interceptados, além da possibilidade de causar interferência em outros aparelhos eletrônicos (TED,2011).

Estamos cercados de câmeras e telas, a comunicação de dados pode ser disposta em cima dessas telas codificando informações no padrão em que os LEDs se acendem e apagam, abaixo do limiar da percepção humana. A comunicação via luz visível dessa maneira é inerentemente segura e cria uma rede de baixa velocidade na vizinhança imediata da tela. As luzes piscando nos veículos de emergência podem alertar os sinais de transito e veículos mais próximos para ajudar a limpar um caminho por exemplo (TANENBAUM,2011).

Além disso, o VLC é uma tecnologia emergente que tem atraído a atenção dos pesquisadores para aplicações em interiores, Internet das coisas e a próxima geração de redes celulares de alta velocidade (WANG; DONNO; GIUSTINIANO,2016).

1.2 Objetivos

O objetivo geral deste trabalho consiste em implementar um sistema de comunicação ótico sem fio utilizando luz visível utilizando um LED e um fotodiodo para efetuar a modulação e transmissão de dados.

Neste trabalho, pretende-se implementar o transmissor e o receptor de dados e utilizar um micro controlador para gerar uma modulação com Chaveamento Liga-Desliga (OOK) e uma codificação Manchester e verificar o funcionamento do sistema variando-se a distância entre o receptor e transmissor, a influência da luz no meio de comunicação, além do alinhamento, tamanho do quadro transmitido e frequência de transmissão.

1.3 Organização do Trabalho

O presente trabalho possui a seguinte organização: no capítulo 2 são apresentados os elementos básicos de uma rede de comunicação, os modos de transmissão e largura de banda.

No capítulo 3 são apresentados os conceitos de modulação, diferença entre símbolos e bits, codificação ou modulação digital, capacidade máxima de um sistema de comunicação, VLC e informações de erros sobre o sistema.

2 ELEMENTOS BÁSICOS DE UMA REDE

Uma rede de comunicação é composta por uma fonte geradora de informação, um transmissor de sinal, uma rede de comunicação, um receptor e um destinatário, uma fonte é caracterizada pela geração da informação que se deseja transmitir no sistema de comunicação, a informação gerada na fonte deve ser tratada antes de se utilizar a rede e comunicação, se adequando ao meio para que possa ser transmitida.

O transmissor é o elemento responsável por converter a informação em um sinal que possa ser transmitido, é no transmissor que ocorre a modulação para que ela possa trafegar na rede de comunicação.

A rede de comunicação vem a ser o meio onde a informação trafega até chegar ao receptor, que recebe o sinal da rede de comunicação e faz o tratamento necessário para que o mesmo seja recebido pelo destinatário, como a demodulação e decodificação do sinal. O destinatário é o elemento para o qual a informação da fonte foi endereçada (DANTAS,2002).

2.1 Modos de Transmissão

O modo de transmissão de dados entre dois equipamentos pode ser efetuado considerando-se o sincronismo, ou não, dos relógios dos equipamentos envolvidos na transmissão. No primeiro caso, com sincronismo dizemos que a transmissão é síncrona. No segundo, dizemos que a transmissão é assíncrona.

A transmissão síncrona é caracterizada pela transferência de qualquer bloco de caracteres entre dois equipamentos de uma forma continua. Isto é, o equipamento receptor

garante que o bloco de caracteres enviados pelo remetente será recebido da mesma forma que foi enviado.

Para que a transmissão dos sinais de maneira síncrona seja efetuada com sucesso entre dois equipamentos, podemos utilizar vários níveis de sincronização. Dentre estas técnicas, podem ser destacadas as seguintes:

- O conjunto de bits transmitidos está codificado de uma forma que o receptor pode manter o sincronismo em nível de bit;
- Todos os quadros são precedidos de um ou mais conjunto de caracteres que asseguram que o receptor vai receber a informação enviada pelo remetente corretamente;
- O conteúdo de cada quadro está encapsulado entre um par de caracteres reservados que garante a sincronização.

Na transmissão assíncrona, temos cada caractere tratado de uma forma independente quanto a sincronização de relógio e caractere. O equipamento receptor é responsável pela sincronização a cada caractere recebido. A forma que essa técnica é implementada leva em consideração o uso de caracteres adicionais. Estes são inseridos no início e no final do caractere que se deseja transmitir. Os caracteres de controle empregados são conhecidos como start bit e stop bit. Resumindo, cada caractere (ou byte) transmitido é acompanhado por um par de caracteres especiais. Para diferenciar estes caracteres, geralmente é atribuído um tempo duas vezes maior ao stop bit para que o mesmo não seja confundido com o start bit. Para efetuarmos a transmissão, precisamos modular ou codificar o sinal a ser transmitido (DANTAS,2002).

2.2 Largura de Banda

A largura de banda de um sistema de comunicações é definida como sendo a faixa de frequências que o sistema pode utilizar para transportar informações, é de grande importância por ser diretamente proporcional a capacidade de transmissão de informação do sistema, quanto mais pulsos por segundo puderem ser enviados maior será a capacidade de transporte de informação do sistema, mas antes de transmitirmos os dados, precisamos aplicar uma técnica de modulação.

Sistemas VLC utilizam o espectro de luz visível, como pode ser visto na imagem a seguir.

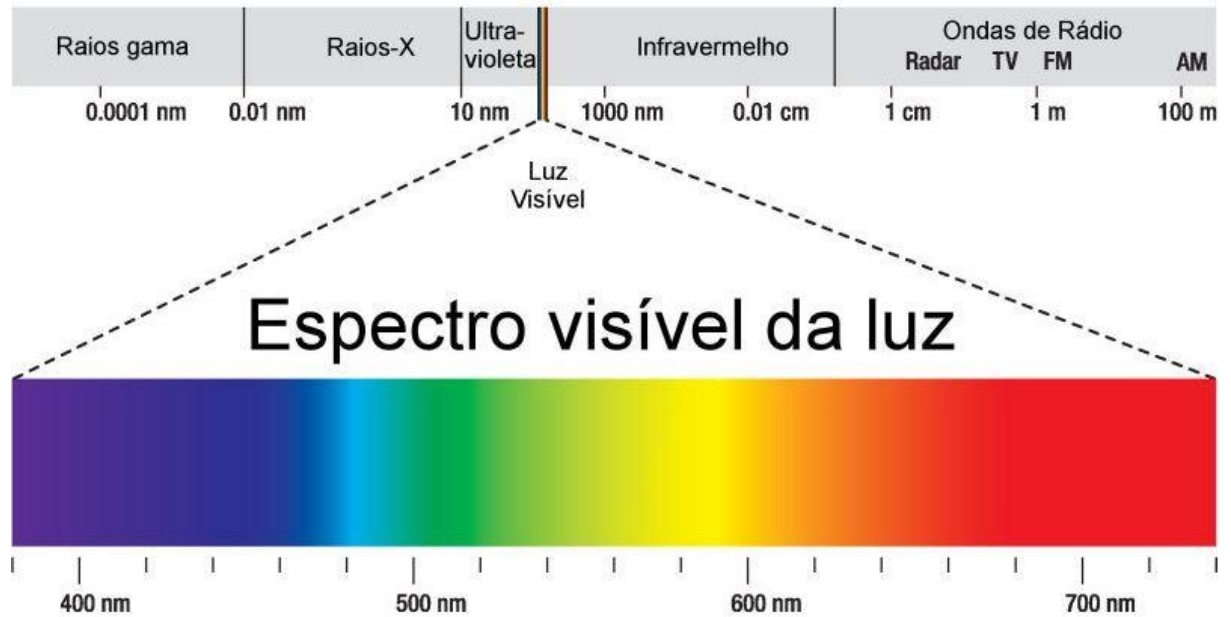


Figura 1-Espectro Visível da Luz

3 MODULAÇÃO

A modulação é um processo no qual certas características de uma onda, denominada de portadora, são modificadas segundo uma função modulante. As funções modulantes são caracterizadas pelas informações de frequência. As três formas genéricas de modulação são Modulação em Amplitude (AM), Modulação em Frequência (FM) e Modulação em Fase (PM), essas três técnicas básicas de modulação funcionam bem para sinais analógicos. Todavia, estas técnicas requerem pelo menos um ciclo de onda portadora para enviar um bit. Podemos ver essas técnicas básicas de modulação na imagem a seguir.

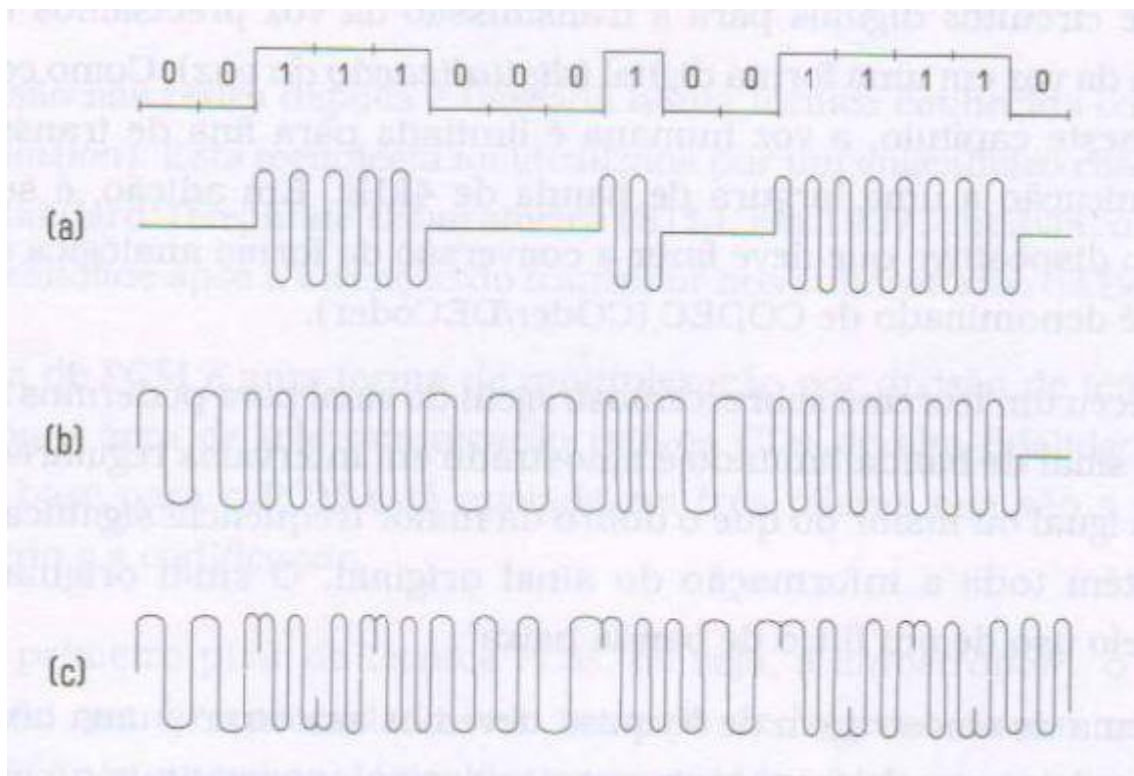


Figura 2-Modulação (a) AM, (b) FM e (c) PM

Visando aumentar o número de bits transmitidos num intervalo de tempo, quando consideramos a transmissão de sinais digitais, algumas das técnicas de modulação são Modulação por Chaveamento de Amplitude (ASK), Modulação por Chaveamento de Frequência (FSK) e Modulação por Chaveamento de Fase (PSK). O motivo de se empregar estas técnicas é que podemos de uma maneira precisa quantificar mais bits em um determinado intervalo de tempo. A modulação ASK consiste em alterar a amplitude utilizando um número finito de sinais distintos para representar dados digitais, utilizando um número finito de amplitudes. A modulação FSK consiste em alterar a frequência do sinal da portadora de acordo com a informação a ser transmitida, nesse caso temos frequências diferentes para a transmissão de 0s e 1s. A modulação PSK consiste em alterar a fase do sinal da portadora de acordo com a informação a ser transmitida, nesse caso temos fases diferentes para a transmissão de 0s e 1s.

A modulação OOK é muito utilizada em sistemas de comunicação via luz visível, ela é a forma mais simples de modulação ASK, onde a presença de uma portadora representa um binário e a ausência desta representa o outro.

3.1 Definição de Símbolo e Bit

Um símbolo está bem separado de bits em conceito, apesar de ambos poderem ser representados por ondas ou funções senoidais. Bit é a unidade de informação, já o símbolo é a unidade de transmissão de energia. É a representação do bit que o meio transmite para enviar a informação. Os bits são como pequenos objetos e os símbolos como caixas nas quais os objetos serão transportados. Pode-se ter um objeto por caixa ou mais. O empacotamento de objetos por caixas é o que o processo de modulação faz (MACHADO).

3.2 Codificação ou Modulação Digital

Já a codificação ou modulação digital é uma mudança na representação da informação de entrada para que esta possa ser transportada num sistema de comunicação digital, independente da informação de entrada ser analógica ou digital, o objetivo da codificação é a transformação de uma informação em um sinal digital adequado as condições de uma rede de comunicação digital, na figura a seguir temos alguns exemplos de codificações.

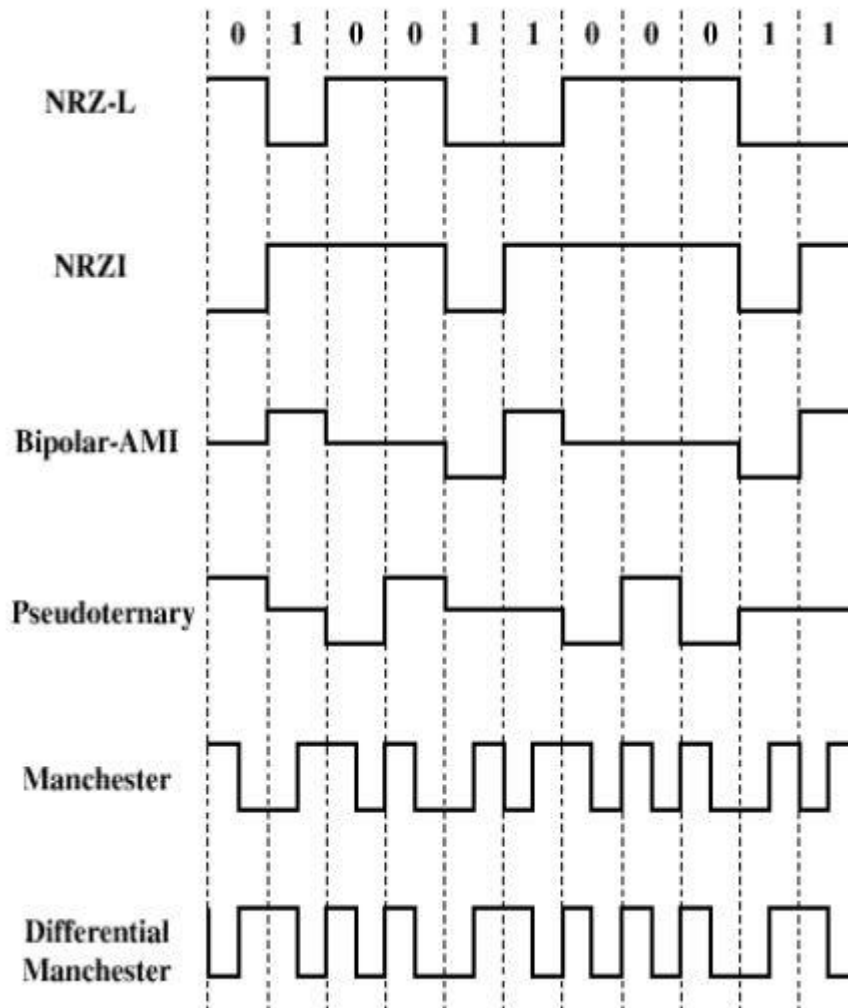


Figura 3-Exemplos de codificações

A forma mais simples de modulação digital é usar uma tensão positiva para representar 1s e uma tensão negativa para representar 0s. Para a comunicação via luz, a presença de luz poderia representar 1s e a ausência de luz 0s. Mas essas codificações podem apresentar longas sequências de 0s, o que se torna um problema se queremos ter uma comunicação via luz visível imperceptível para os olhos humanos, por isso é preferível o uso de codificações com equilíbrio DC, como a Manchester ou a 4b6b.

A codificação Manchester mantém o número de 1s e 0s iguais, já que cada período do bit é dividido em metades complementares. Assim, uma transição de tensão de positiva para negativa no meio do bit indica um número binário '1', enquanto uma transição de negativa para positiva no meio do bit indica um número binário '0'.

A codificação 4b6b mantém no máximo uma sequência de três 0s, e utiliza a seguinte tabela de conversão, como pode ser vista na imagem a seguir.

4-bit	6-bit
0000	001110
0001	001101
0010	010011
0011	010110
0100	010101
0101	100011
0110	100110
0111	100101
1000	011001
1001	011010
1010	011100
1011	110001
1100	110010
1101	101001
1110	101010
1111	101100

Figura 4-Codificação 4B6B

3.3 Capacidade máxima de um sistema de comunicação

A capacidade de um sistema de comunicação está diretamente relacionada a eficiência em largura de banda do sistema, ou seja, é a habilidade de um esquema de modulação acomodar os dados dentro de uma largura de banda limitada. Ou seja,

$$n_b = \frac{R(\text{bits por segundo})}{\text{Largura de Banda(Hz)}}$$

3.4 VLC e Avaliação de Erros

Um sistema de comunicação através da luz visível baseia-se em um sistema de transmissão de dados onde o transmissor engloba tanto a função de transmissão de dados quanto iluminação ambiente. O receptor é capaz de distinguir as variações da iluminação e recuperar os dados transmitidos (SOUZA,2013).

O espectro da banda utilizado no VLC é o da luz visível, que possui comprimento de onda entre 350nm a 750nm, com isso todos os componentes utilizados na transmissão e recepção devem funcionar para estes comprimentos de onda.

Para enviar e receber os dados, podemos utilizar LEDs e fotodiodos, o LED é um componente eletrônico semicondutor, ou seja, um diodo emissor de luz, ele é um componente do tipo bipolar, dependendo como for polarizado permite ou não a passagem de corrente elétrica e também a geração ou não de luz. Já os fotodiodos são dispositivos eletrônicos semicondutores, eles possuem uma junção semicondutora, que tem a propriedade de variar a sua resistência elétrica em função da intensidade da luz nela incidente. Depois de recebido os dados, eles podem ser demodulados em um micro controlador, que é um tipo especial de circuito integrado, de uso geral, pois vem com a possibilidade de ser programado para desempenhar tarefas específicas.

No micro controlador, podemos comparar os dados recebidos com os dados enviados e obter informações do sistema como a Taxa de Erro de Bit (BER), que pode ser calculada através da seguinte equação.

$$BER = \frac{\text{Bits Recebidos com Erro}}{\text{Bits Transmitidos}}$$

Com essa informação podemos verificar o desempenho desse sistema em diferentes condições de uso. Podemos definir a Taxa de Erro de Pacote (PER) também, que pode ser calculada através da seguinte equação.

$$PER = \frac{\text{Pacotes Recebidos com Erro}}{\text{Pacotes Transmitidos}}$$

Podemos adicionar ao pacote de dados um bit de paridade, para verificar erros no receptor ou adotar alguma técnica de correção de erros.

No caso do VLC, a transmissão é feita de modo unidirecional, por isso não possui sinal de sincronismo entre emissor e receptor e a transmissão deve ser assíncrona, isto é, ter um sinal de sincronismo embutido no próprio dado enviado, e devemos ter um cuidado com o efeito flicker, já que de acordo com estudos cerca de 1 em 4.000 pessoas são altamente suscetíveis a luzes que piscam com Ciclismo na gama de 3 a 70 Hz. Tal cintilação pode desencadear doenças tão graves como convulsões epiléticas. Menos conhecido é o facto de a exposição a longo prazo a uma maior frequência (não intencional) de cintilação (na gama de 70-160 Hz) também pode causar mal-estar, dores de cabeça, e perturbações visuais. Os testes mostram que os humanos dificilmente sentem diretamente cintilação da luz a estas frequências mais elevadas, mas investigações indicam que a retina humana é capaz de perceber cintilação de luz de 100 a 150Hz, por isso para LEDs, recomenda-se uma frequência de pelo menos 150Hz (KEEPING,2012).

Outro problema no VLC é a interferência, o sinal recebido pode ser direto ou refletido, o atraso entre um raio luminoso direto e um refletivo pode gerar um alargamento dos impulsos, se eu impulso se estender por mais de um período do símbolo, gera a chamada interferência entre símbolos, limitando a taxa de sinalização (KOMINE: NAKAGAWA, 2004).

Muitos fatores podem afetar o desempenho de canais VLC, a especificação do transmissor e receptor certamente desempenha um papel importante, já que podem ser utilizados diferentes tipos de LEDs para transmissão, enquanto a recepção pode ocorrer com fotodiodos ou LEDs, além disso, como o caso da RF, o VLC também é sensível a parâmetros de nível de protocolo, tais como taxa de símbolo e tamanho do pacote (Heydariaan; Yin; Gnawali; Puccinelli; Giustiniano.2016).

4.DESENVOLVIMENTO

A implementação desse sistema se deu de acordo com o diagrama de blocos a seguir.

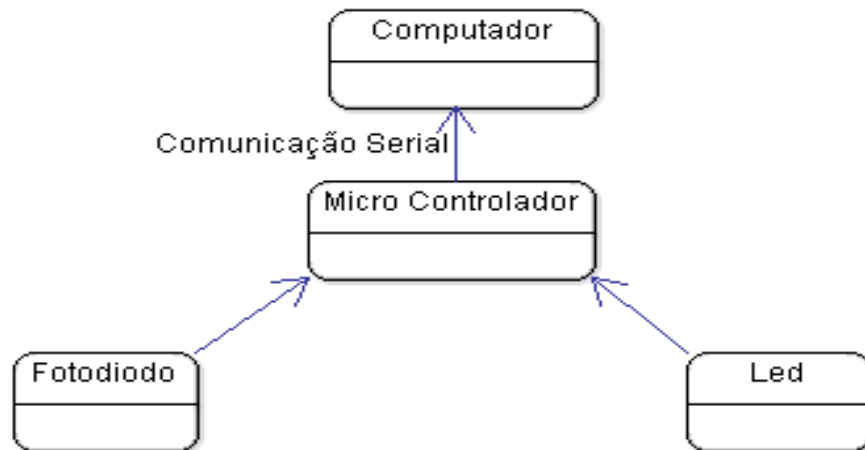


Figura 5-Diagrama de Blocos do Sistema

Para implementar esse sistema, foi 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.

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.

4.1 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, como pode ser visto no diagrama a seguir.

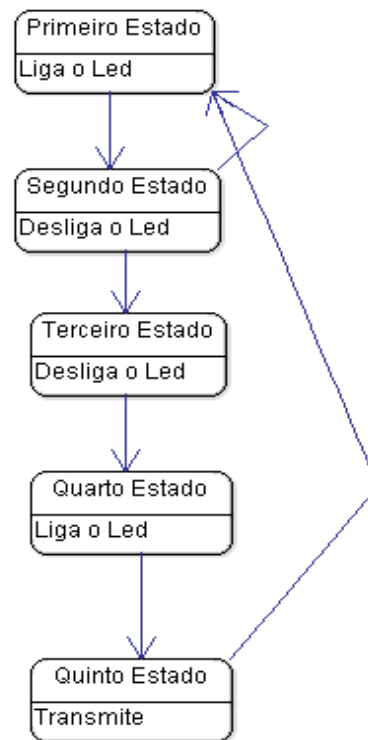


Figura 7-Diagrama de Estados Transmissor

4.2 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, como pode ser visto no diagrama a seguir.

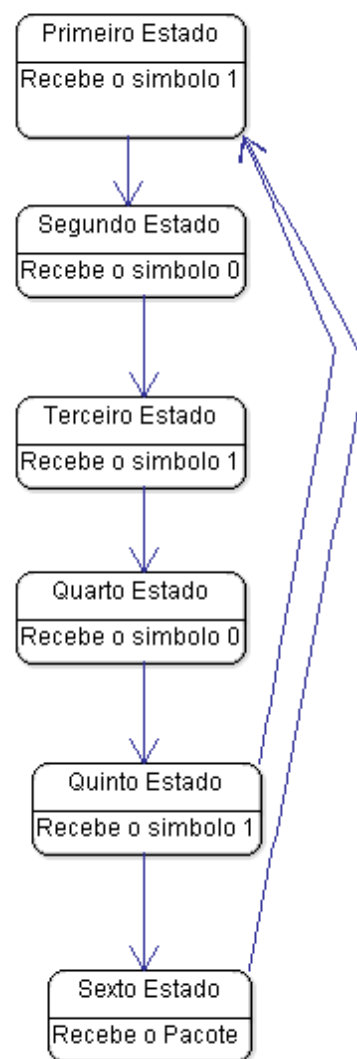


Figura 8- Diagrama de Estados do Receptor

4.3 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. Abaixo podemos ver os símbolos na codificação Manchester que representam esses bits.

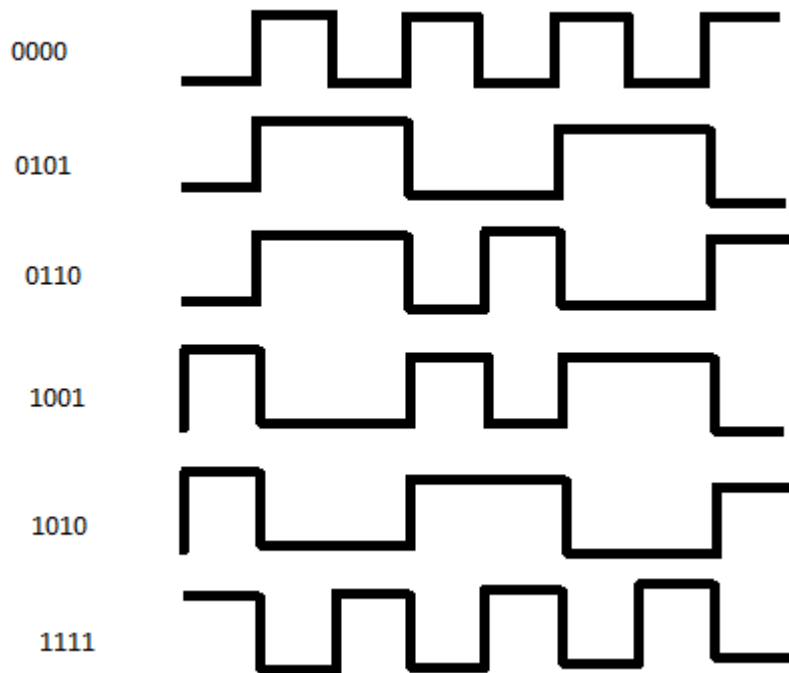


Figura 9- símbolos enviados

Para cada teste foram enviados 1000 pacotes de bits 10 vezes, calculado a média de erro para cada teste, o intervalo de confiança foi definido com 5% de erro para o PER.

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.

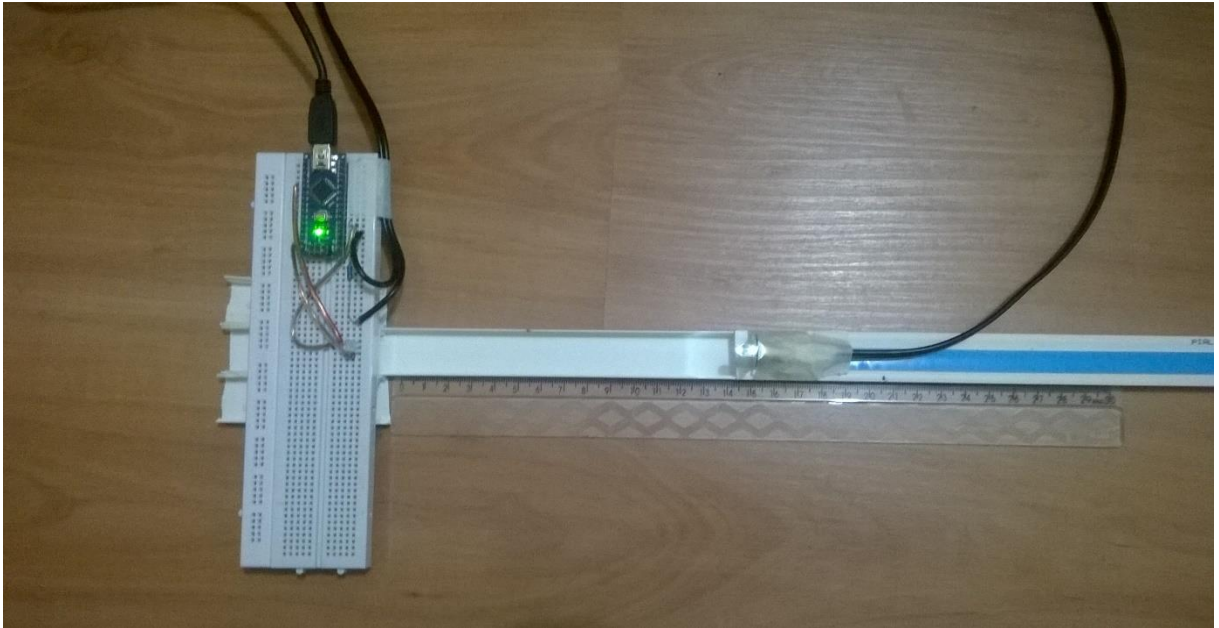


Figura 10 - Teste de Distancia em Ambiente Iluminado

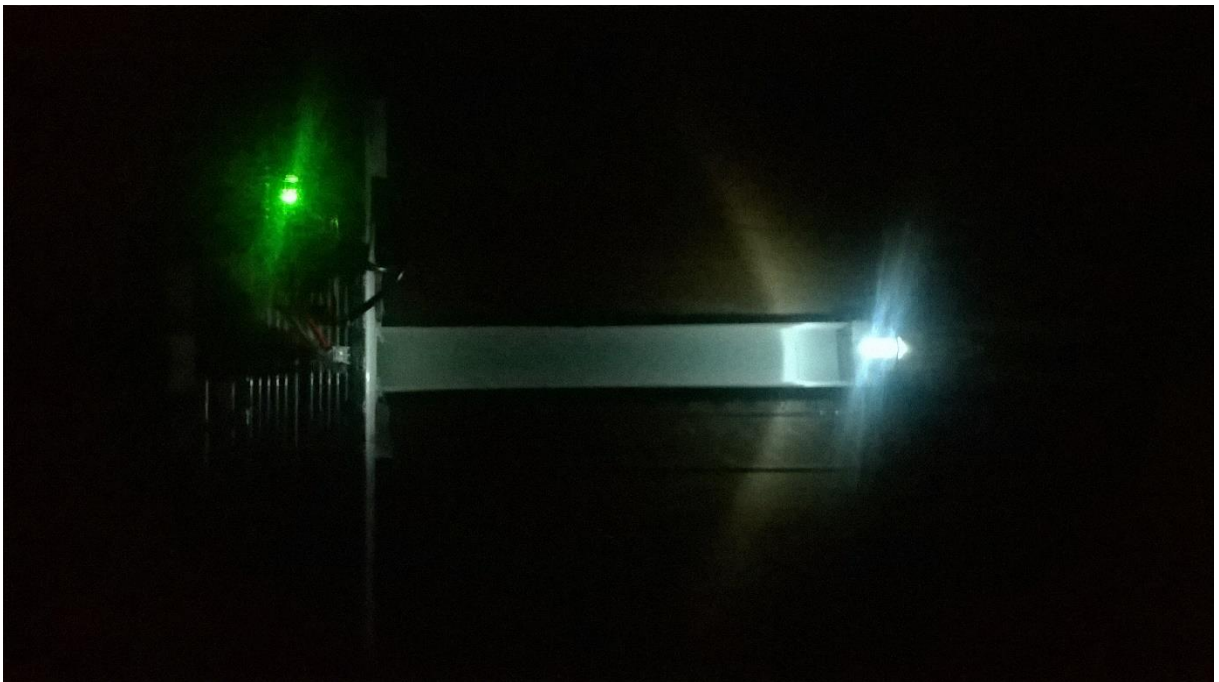


Figura 11 - Teste de Distancia em Ambiente sem Iluminação



Figura 12- Teste de Angulo em Ambiente sem Iluminação

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.

5. RESULTADOS

Este teste só compara a PER 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.

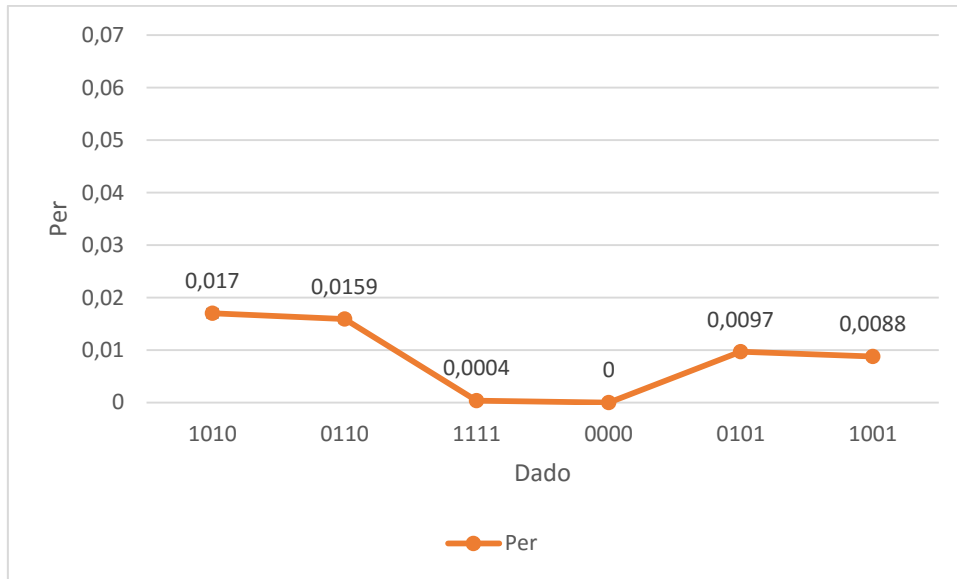


Figura 13- Teste de dados em ambiente iluminado

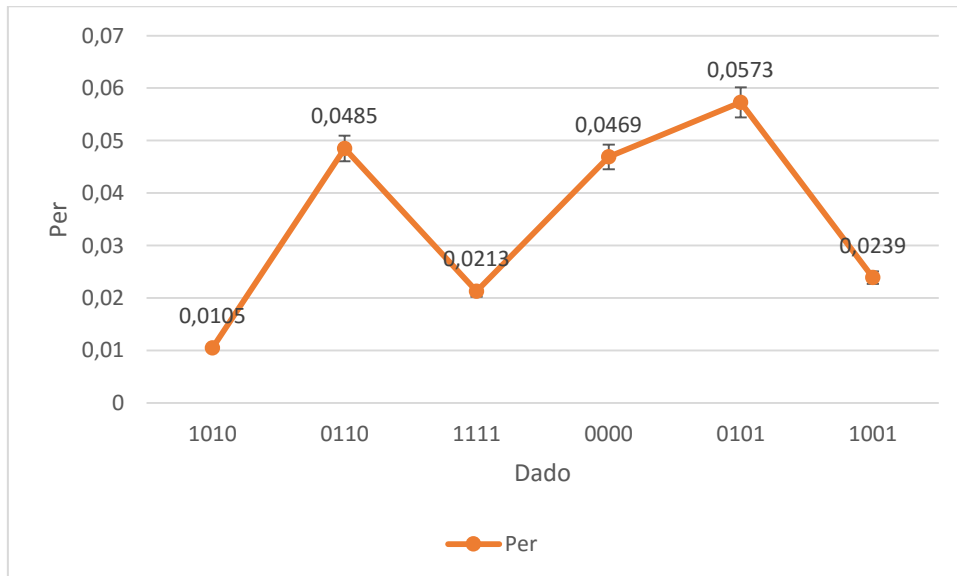


Figura 14 - Teste de dados em ambiente não iluminado

Como o teste para ambiente escuro foi feito com uma margem muito pequena para ruídos comparada com a margem utilizada para o ambiente claro, com isso priorizando o aumento da distância no ambiente escuro, que acabou ficando mais sensível a variações.

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.

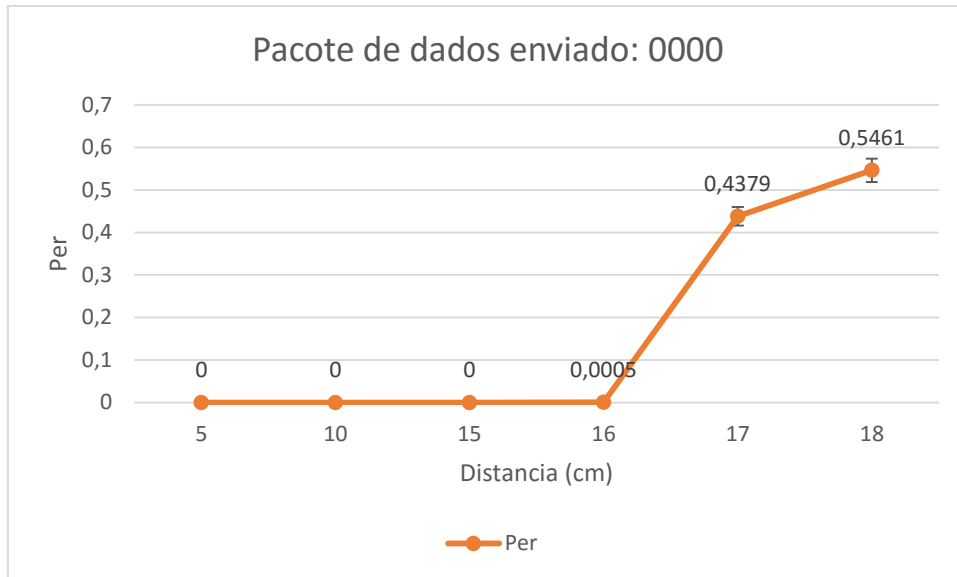


Figura 15- Teste de Distancia em ambiente iluminado

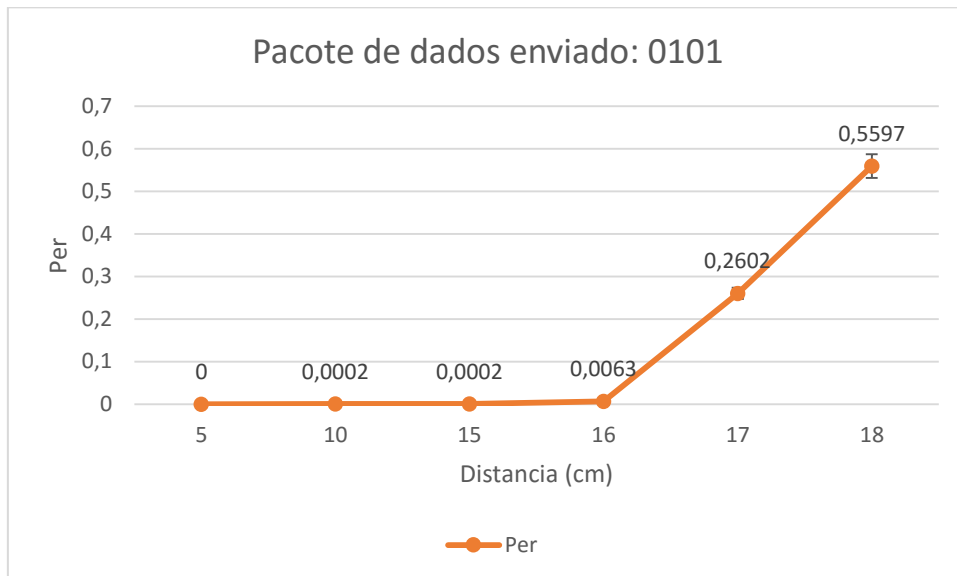


Figura 16- Teste de Distancia em ambiente iluminado

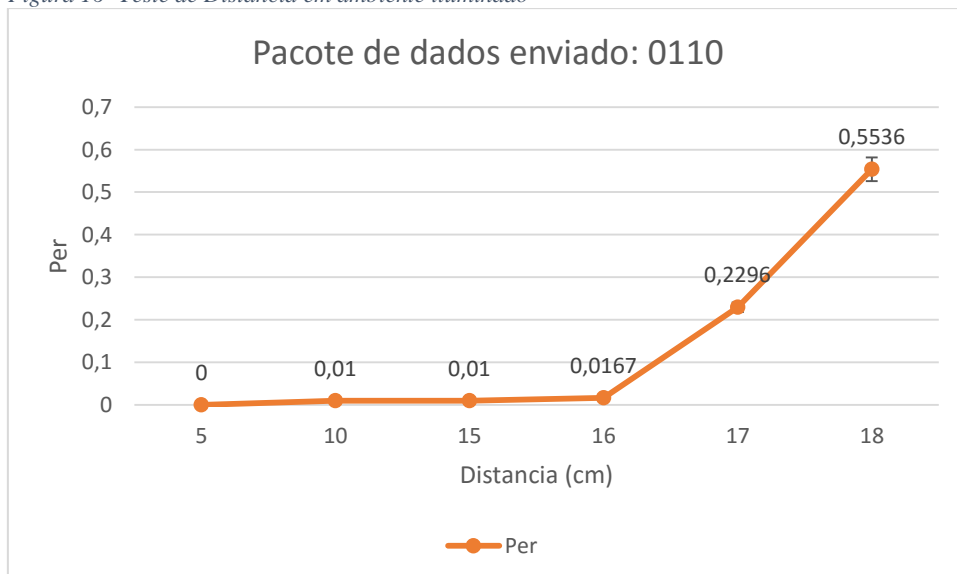


Figura 17-Teste de Distancia em ambiente iluminado

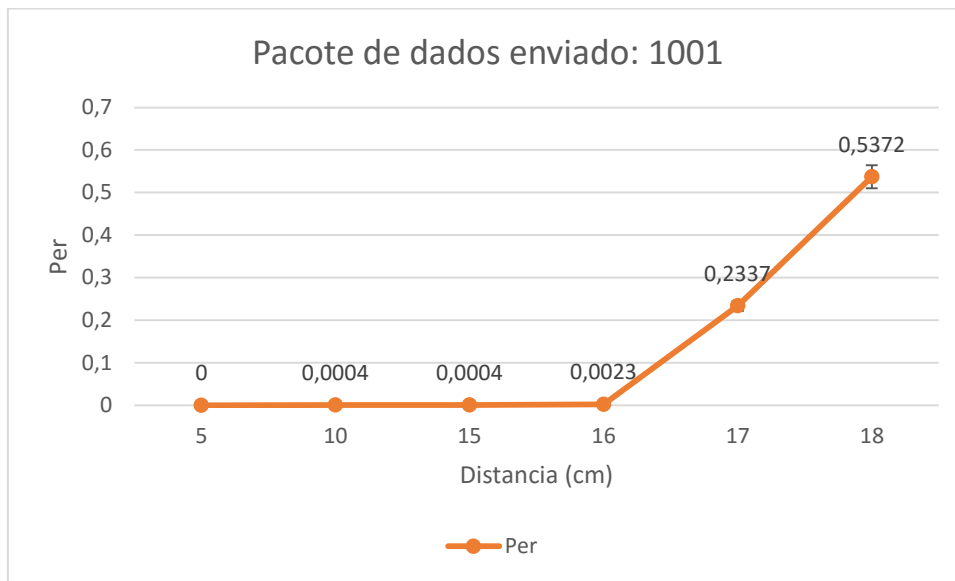


Figura 18-Teste de Distancia em ambiente iluminado

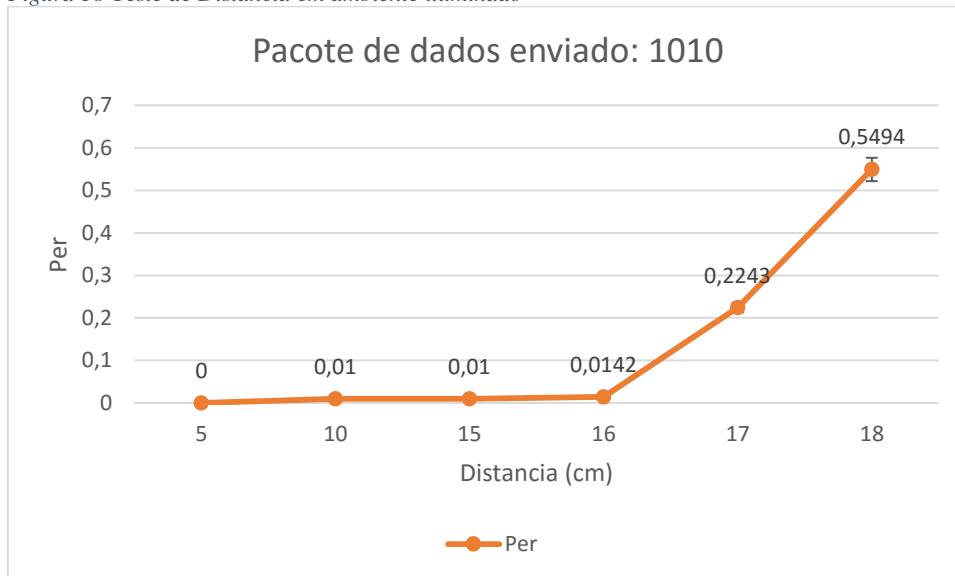


Figura 19-Teste de Distancia em ambiente iluminado

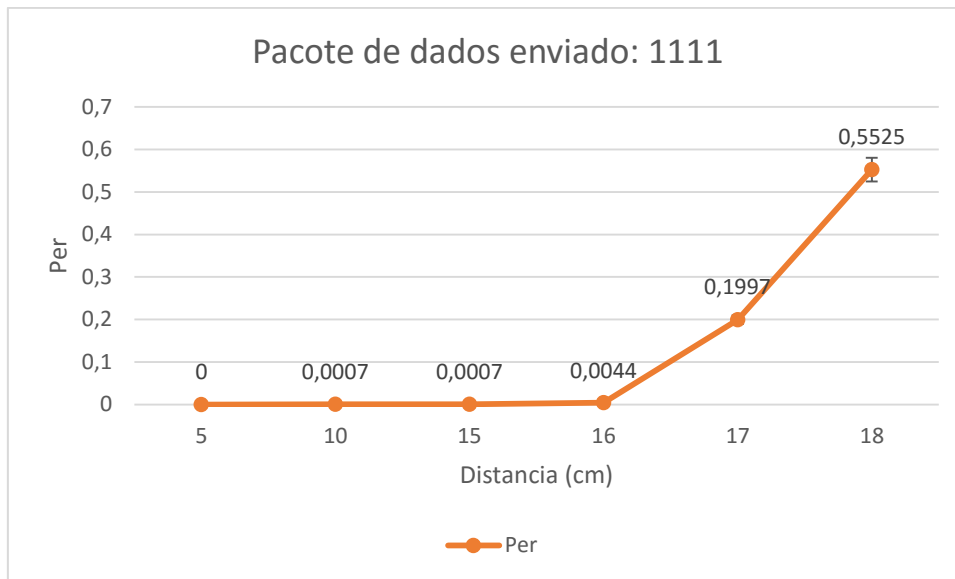


Figura 20-Teste de Distancia em ambiente iluminado

A partir de 18 cm o receptor começou a ter problemas para detectar o início das transmissões no ambiente claro. Já que o que o receptor considera como símbolo 1 não é atingido pelo LED utilizado, que mesmo ligado pela interferência do ambiente, o receptor considera como sendo o símbolo 0.

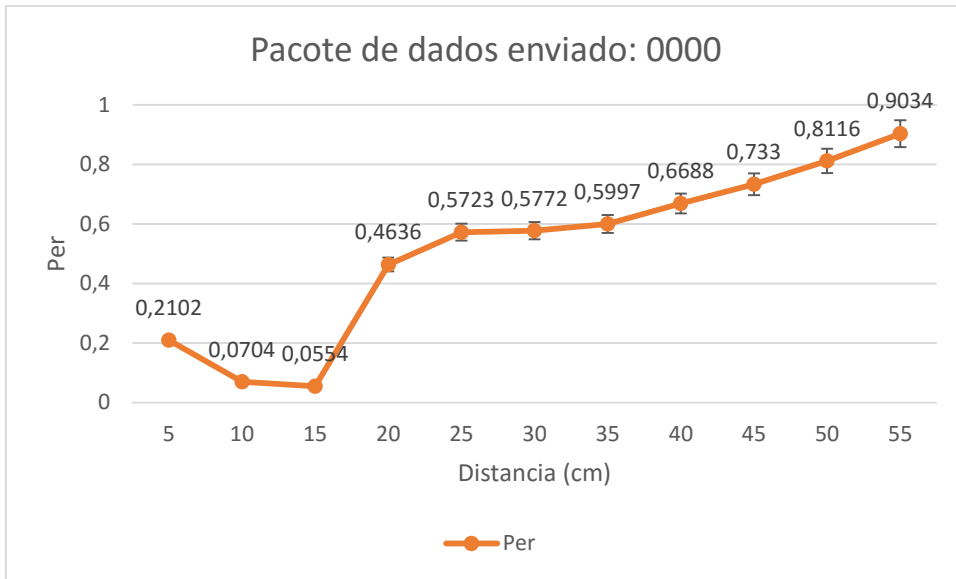


Figura 21-Teste de Distancia em ambiente não iluminado

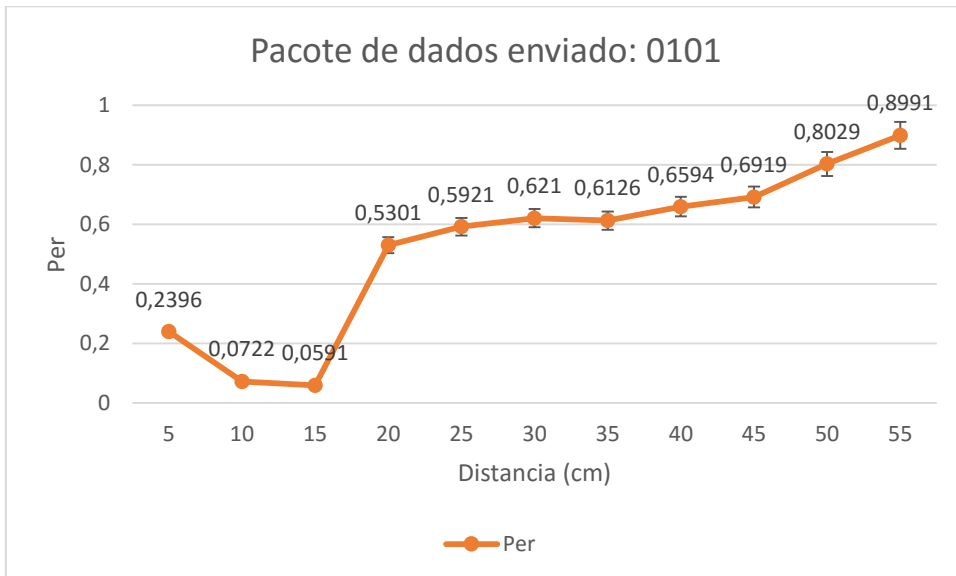


Figura 22-Teste de Distancia em ambiente não iluminado

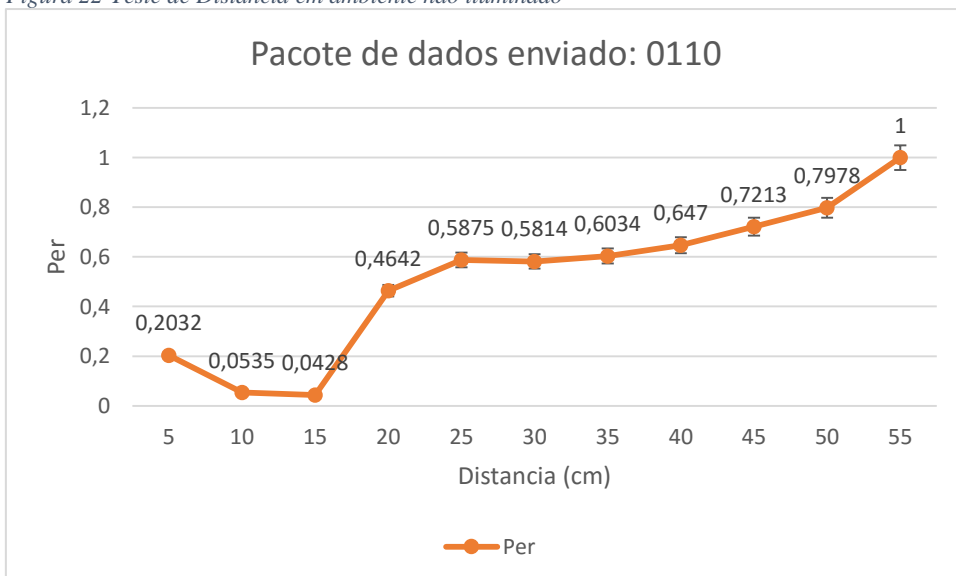


Figura 23-Teste de Distancia em ambiente não iluminado

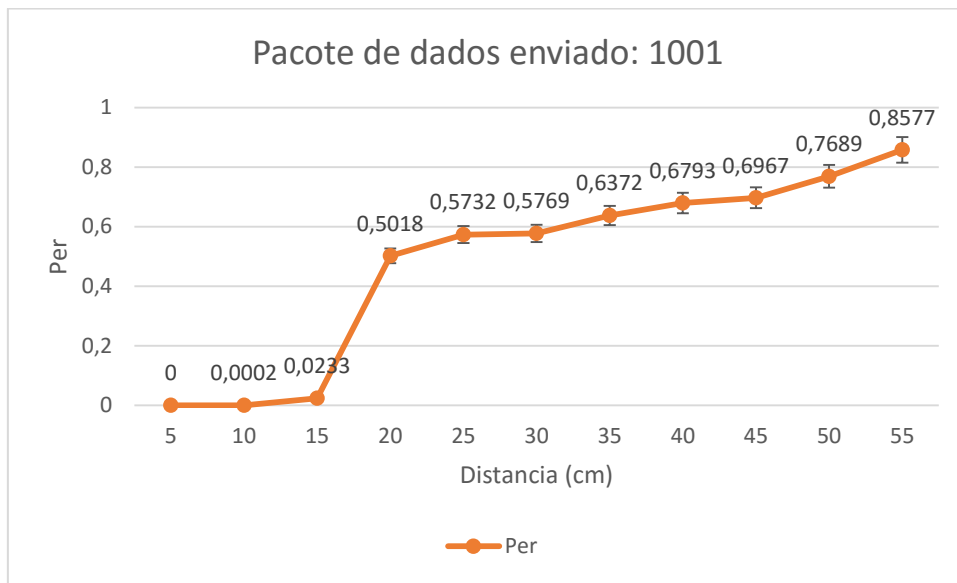


Figura 24-Teste de Distancia em ambiente não iluminado

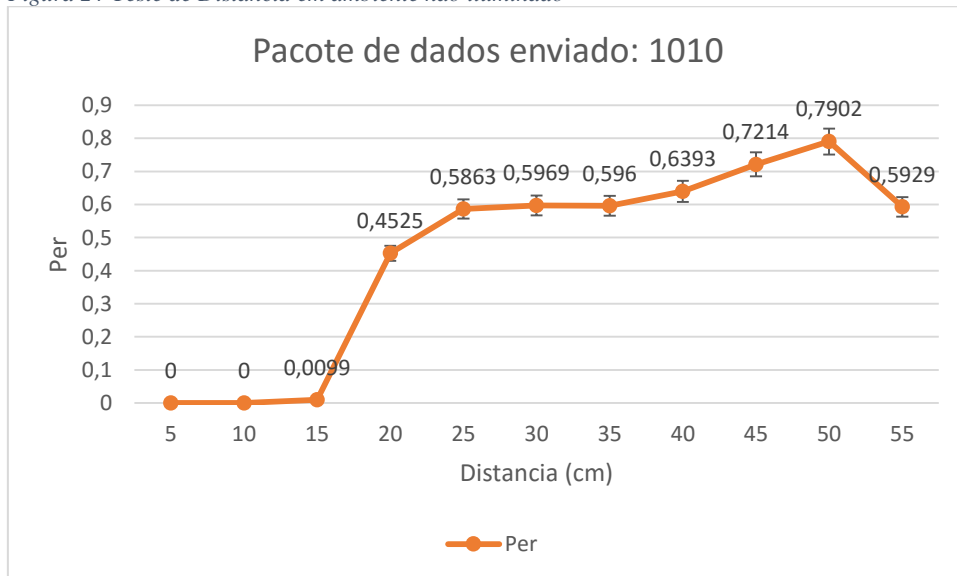


Figura 25-Teste de Distancia em ambiente não iluminado

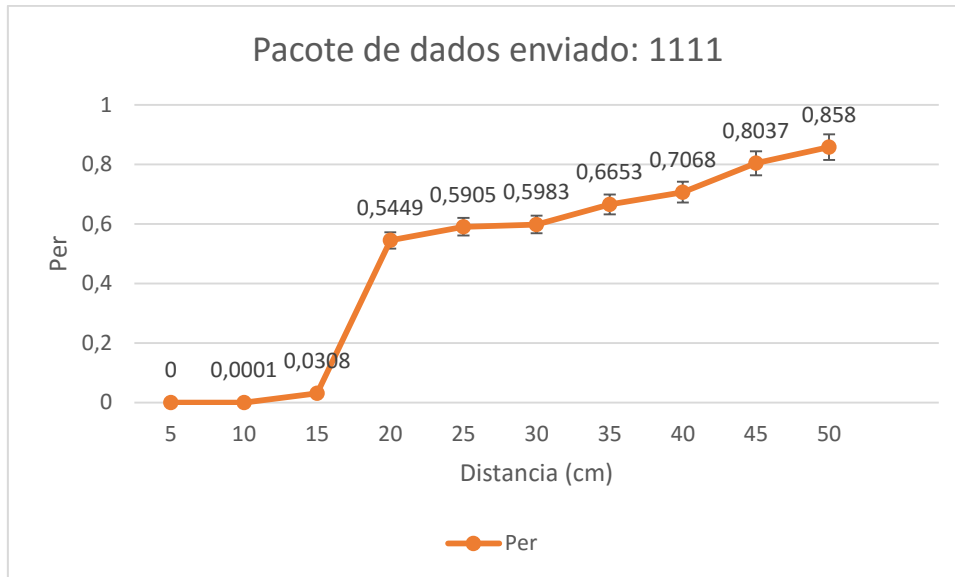


Figura 26-Teste de Distancia em ambiente não iluminado

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.

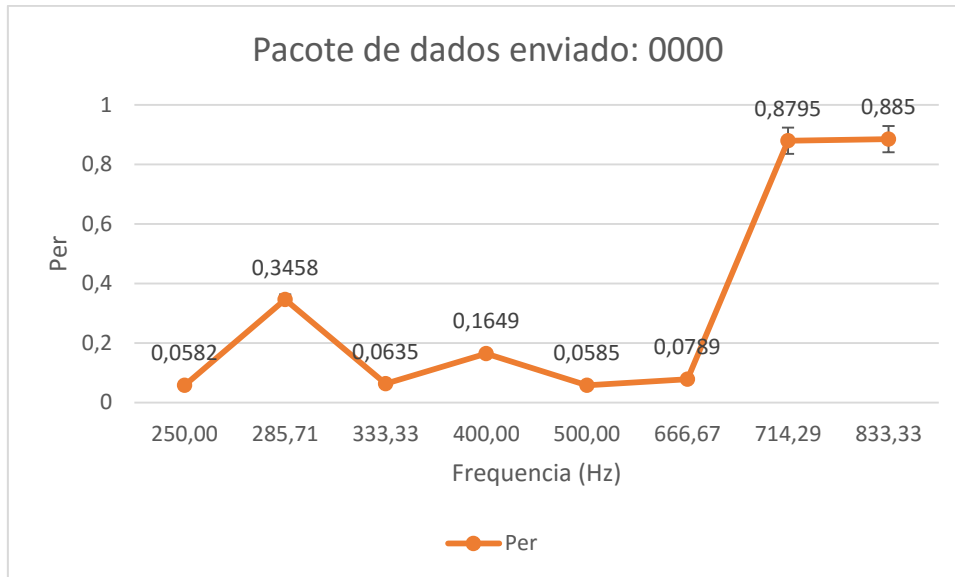


Figura 27-Teste de frequência em ambiente não iluminado

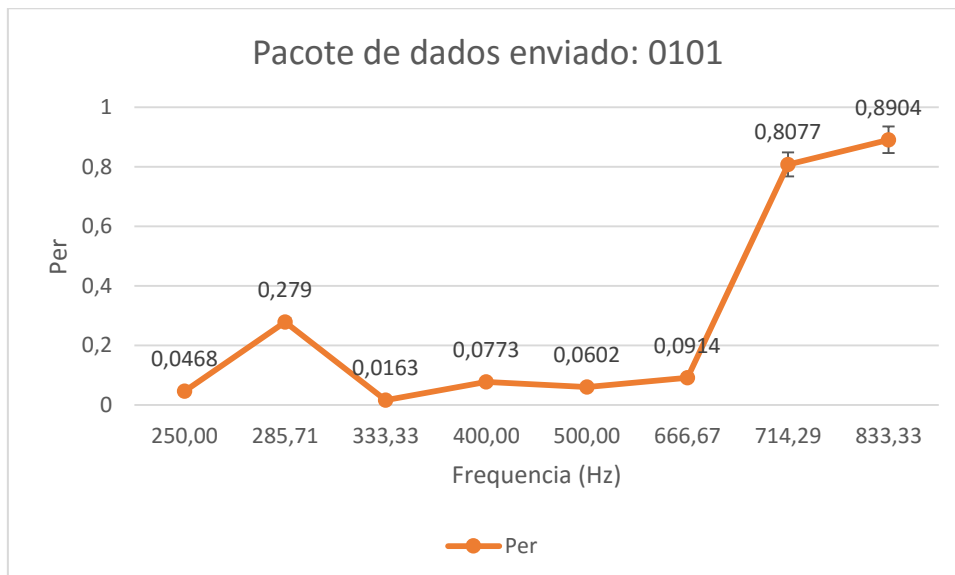


Figura 28-Teste de Frequência em ambiente não iluminado

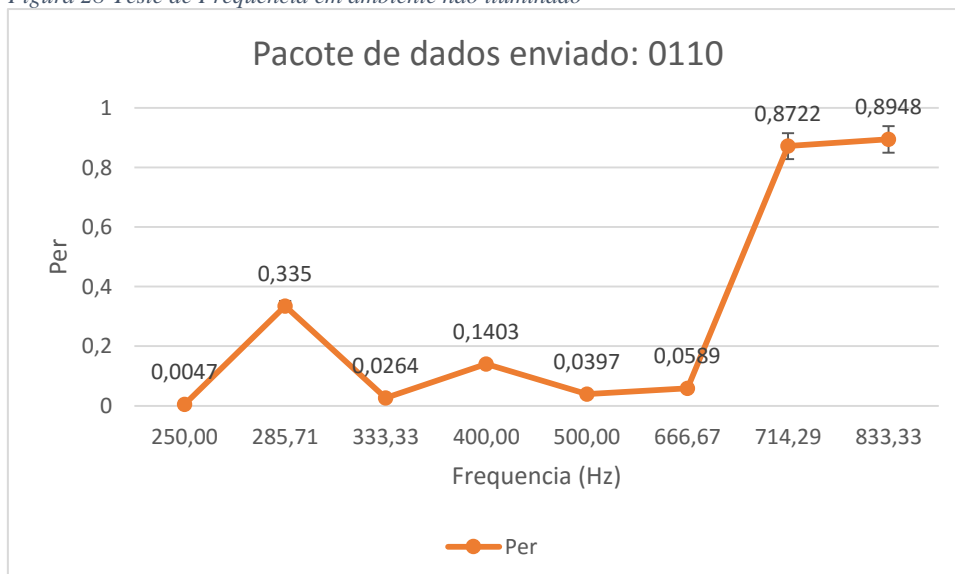


Figura 29-Teste de Frequência em ambiente não iluminado

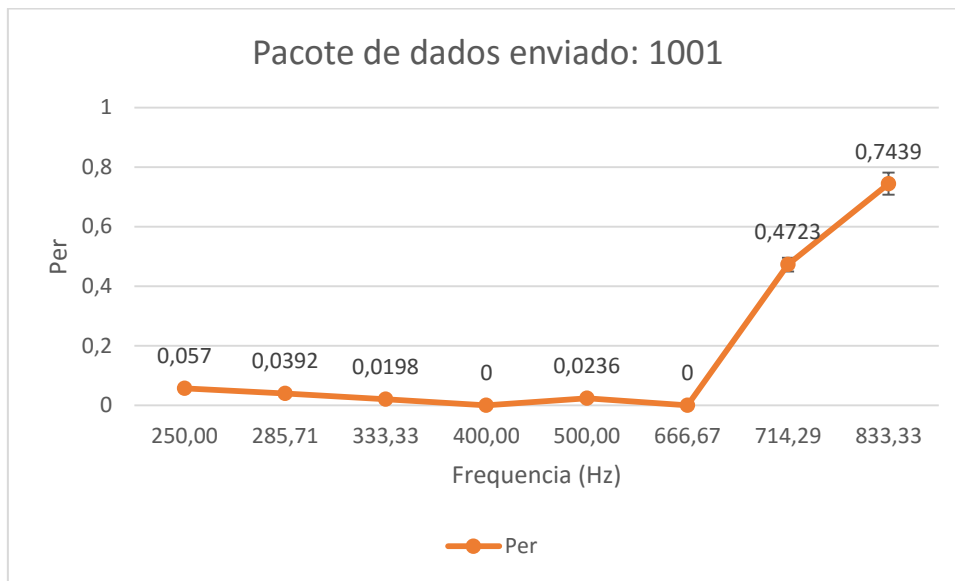


Figura 30-Teste de Frequência em ambiente não iluminado

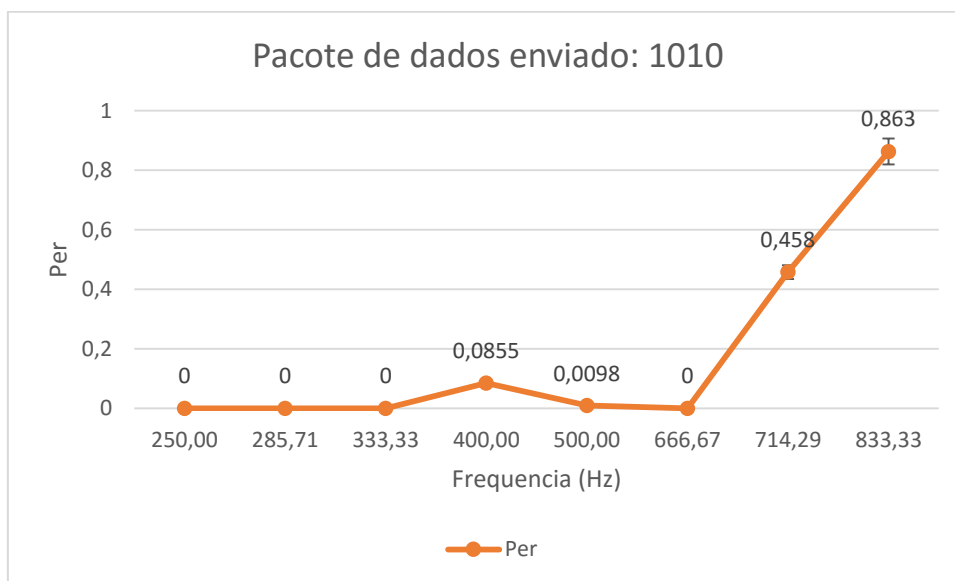


Figura 31-Teste de Frequência em ambiente não iluminado

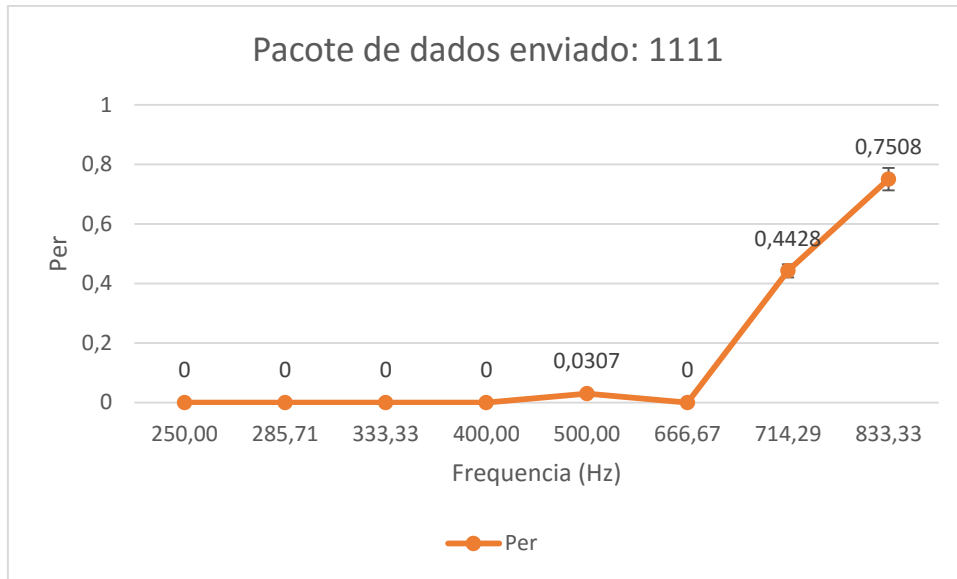


Figura 32-Teste de Frequência em ambiente não iluminado

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.

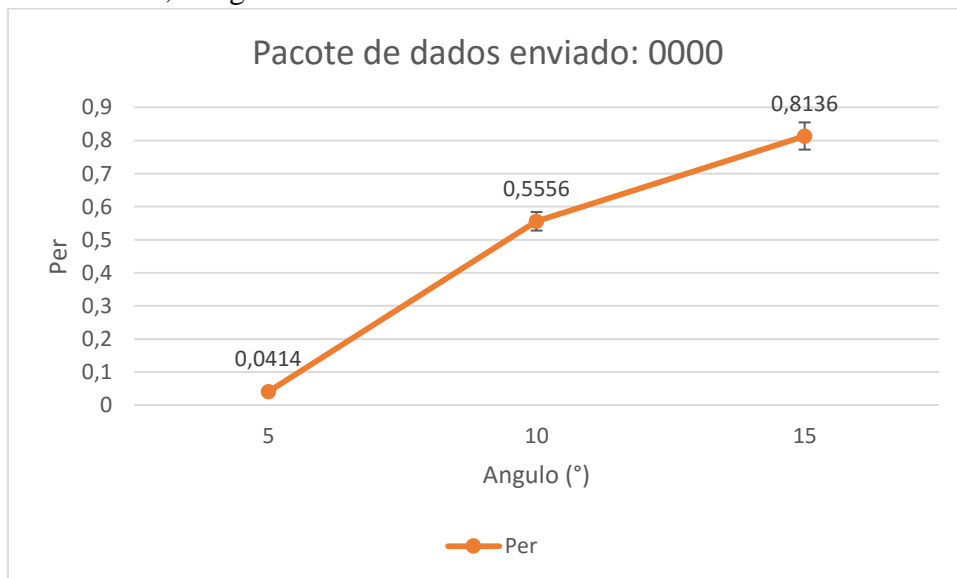


Figura 33-Teste de Angulo de Incidência em ambiente não iluminado

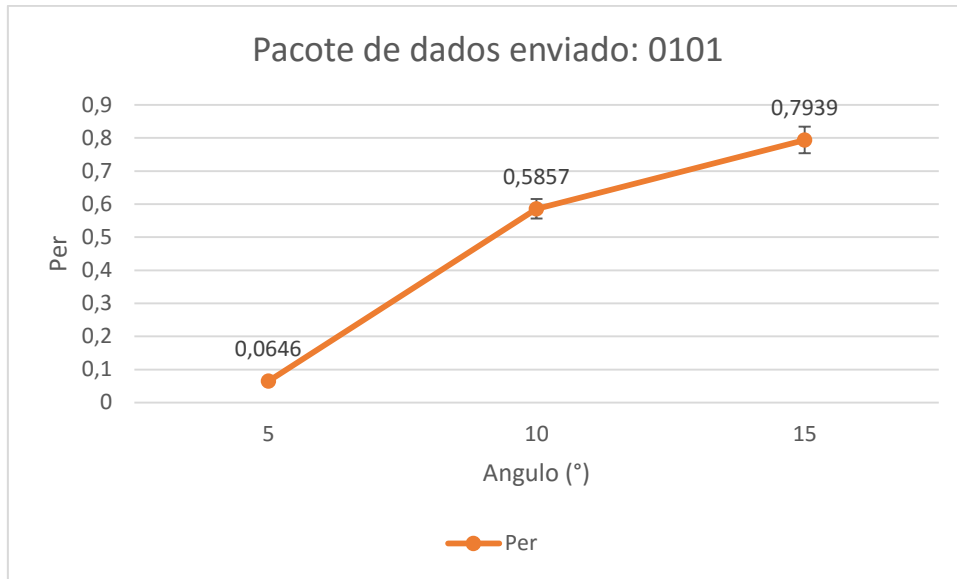


Figura 34-Teste de Angulo de Incidência em ambiente não iluminado

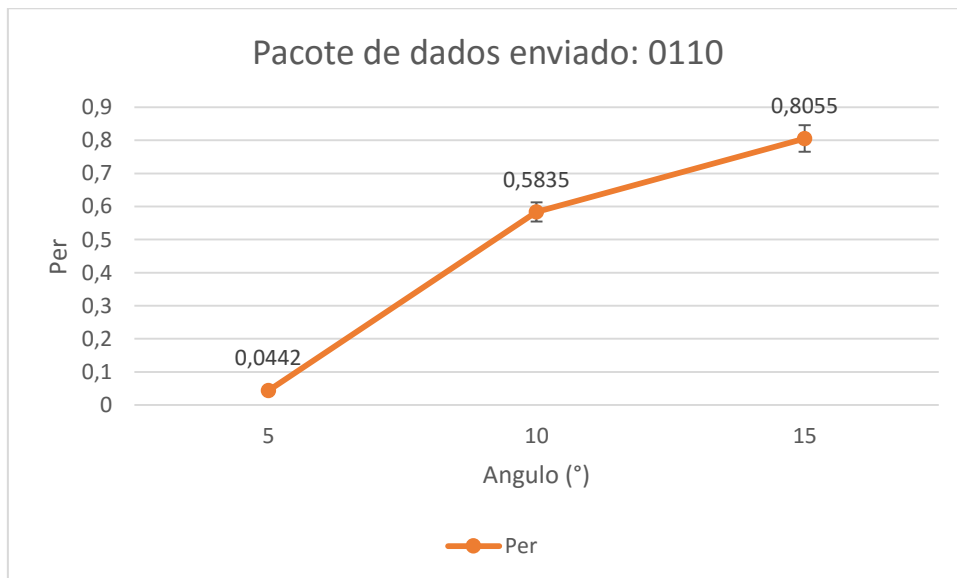


Figura 35-Teste de Angulo de Incidência em ambiente não iluminado

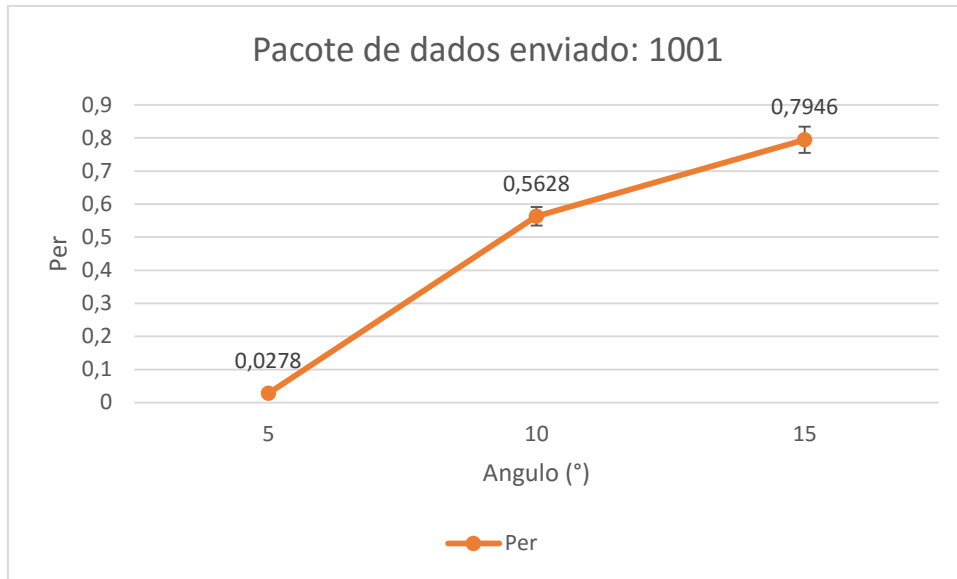


Figura 36-Teste de Angulo de Incidência em ambiente não iluminado

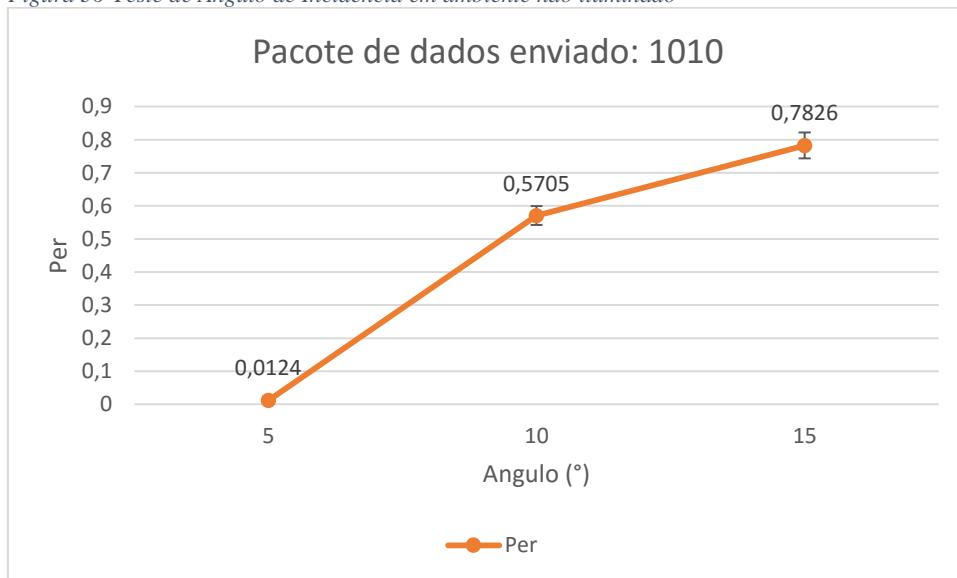


Figura 37-Teste de Angulo de Incidência em ambiente não iluminado

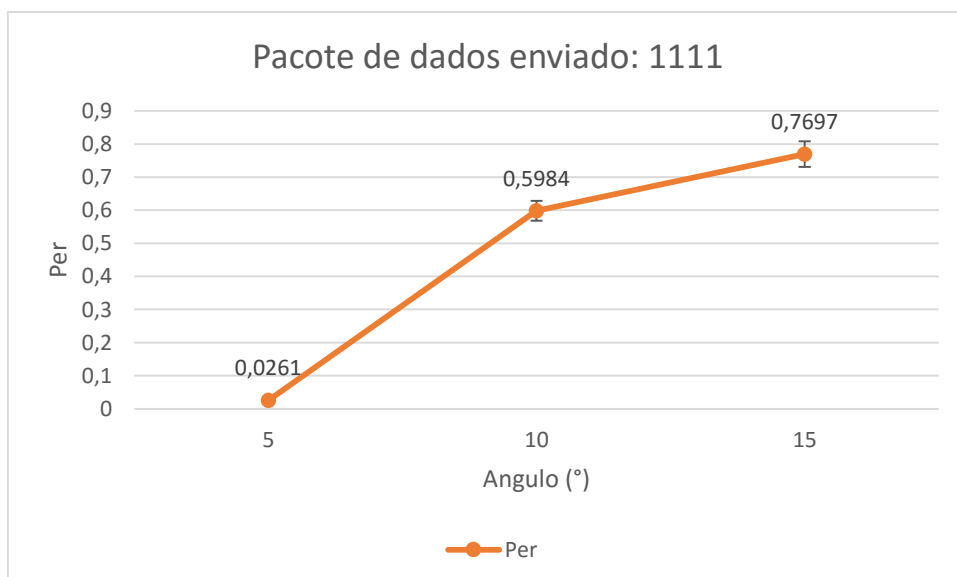


Figura 38-Teste de Angulo de Incidência em ambiente não iluminado

E para o teste de tamanho de pacote, a frequência foi reduzida para 333Hz, pois utilizando a frequência padrão de 500Hz o receptor não conseguia percorrer todas as posições do vetor a tempo e isso gerava erros de sincronismo entre o receptor e o transmissor.

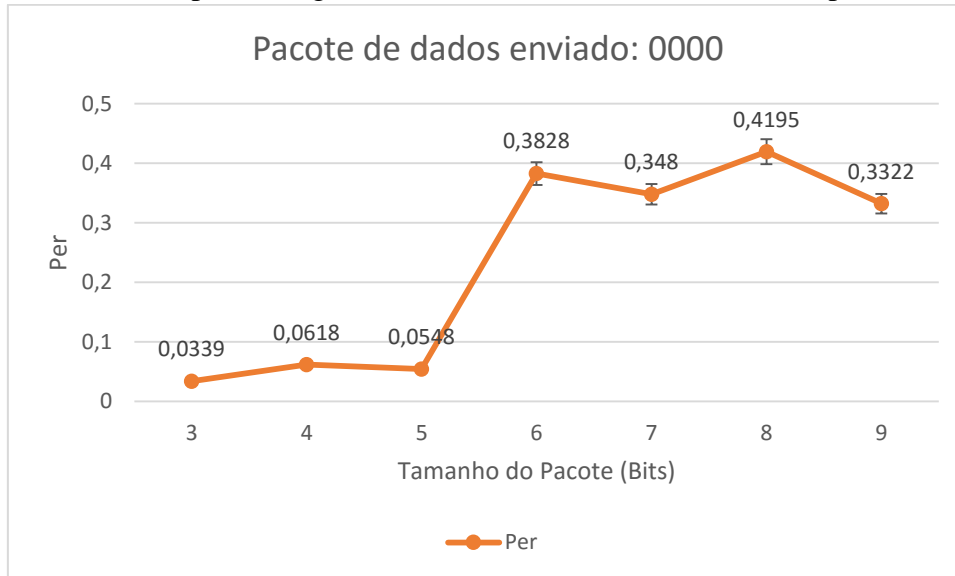


Figura 39-Teste de tamanho do pacote em ambiente não iluminado

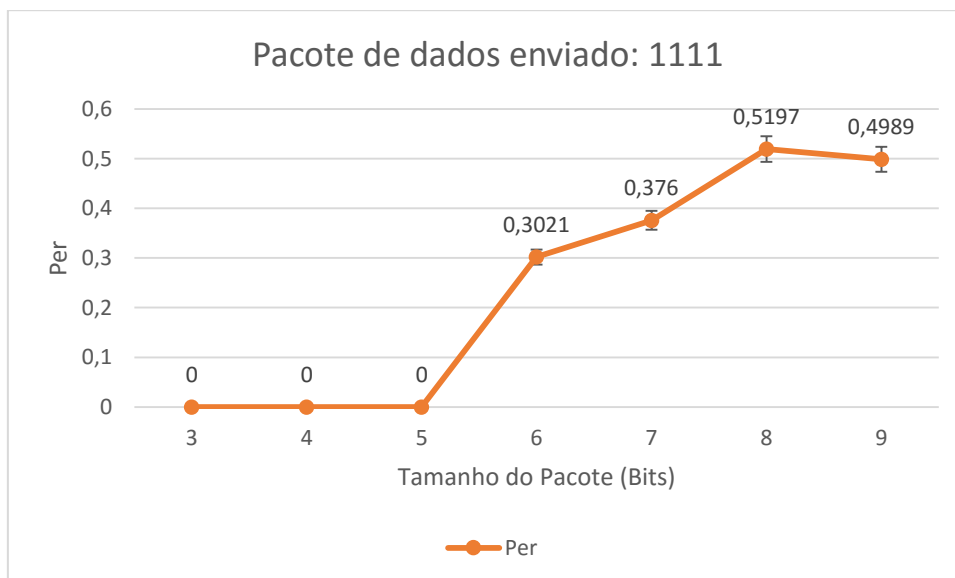


Figura 40-Teste de tamanho do pacote em ambiente não iluminado

Ambos os resultados foram muito próximos, com pequenas exceções como para a frequência de 285Hz que teve um pico dependendo do conteúdo do pacote enviado. E no geral o sistema padrão escolhido previamente foi utilizando uma frequência de 500Hz, um tamanho de pacote de dados de 4 bits e uma distância de 15 cm. Vimos que o sistema seria mais eficiente se utilizasse um tamanho de pacote de 5 bits e uma frequência de 666Hz.

6. CONCLUSÃO

O objetivo desse trabalho era implementar um sistema de comunicação VLC e levantar as características do meio utilizado nesse tipo de comunicação, no geral o trabalho cumpriu com as metas estipuladas.

No entanto, como foi utilizado apenas um LED de baixa potência, não foi possível verificar outros efeitos importantes como a reflexão, e a própria implementação ficou muito dependente do timer do micro controlador, que terá problemas de sincronização para frequências de operação mais elevadas. Além disso a PER não foi a melhor medida de erros, a BER traria informações melhores, como por exemplo se o sistema teria um ganho elevado utilizando um bit de paridade, mas como os pacotes utilizados no teste contem 4 bits, a PER não é uma medida de erro muito distante. No geral foi uma primeira implementação que deve ser aprimorada.

6.1 Sugestões para trabalhos futuros

Trabalhos futuros podem complementar ainda mais o sistema que foi desenvolvido. Aprimorar a função de calibração do nível de luminosidade considerado nível baixo e alto, realizar mais de uma leitura do valor de tensão no fotodiodo em cada ciclo, utilizar alguma técnica para detecção ou correção de erros, e além disso utilizar mais de um LED para tentar alcançar uma distância maior.

REFERENCIAS

- Dantas, M. **Tecnologias de redes de comunicação e computadores**. Tradução . Rio de Janeiro (RJ): Axcel Books, 2002.
- Rappaport, T. **Comunicações sem fio**. Tradução . São Paulo: Pearson Prentice Hall, 2009.
- Tanenbaum, A. Wetherall, D. **Redes de computadores**. Tradução . São Paulo: Pearson Prentice Hall, 2011.
- Boylestad, R. Nashelsky, L. **Dispositivos eletrônicos e teoria de circuitos**. Tradução . São Paulo: Pearson Education do Brasil, 2005.
- SOUZA, Jhonathan Junio de; FONSECA, Zito Palhano da; STEVAN JUNIOR, Sergio Luiz. **Projeto de um sistema de comunicação por luz visível (VLC) baseado em iluminação LED**. [s.i.], 2013.
- Sevincer, Abdullah; Bhattarai, Aashish; Bilgi, Mehmet; Yuksel, Murat; Pala, Nezh. **LIGHTNETs: Smart LIGHTing and Mobile Optical Wireless NETWORKs – A Survey**. [S.I.], 2013.
- Wang, Qing; Gnawali Omprakash; Yin, Shengrong; Giustiniano, Domenico. **Demo: OpenVLC1.0 Platform for Research in Visible Light Communication Networks**, 2015.
- Giustiniano, Domenico; Wang, Qing. **OpenVLC, an Open-Source Platform for Internet of Light**, 2015.
- Wang, Qing; Donno, Danilo de; Giustiniano, Domenico. **Demonstration Abstract: Research Platform for Visible Light Communication and Sensing Systems**, 2016.
- Heydariaan, Milad; Yin, Shengrong; Gnawali, Omprakash; Puccinelli, Daniele; Giustiniano, Domenico. **Embedded Visible Light Communication: Link Measurements and Interpretation**, 2016.
- Machado, Renato. **Notas de Aula**. Acessado em Maio/2016, <http://coral.ufsm.br/gpscom/professores/Renato%20Machado/comunicacaodedados.html>

APÊNDICES

APÊNDICE A – fsm_1_microseconds_final

```
#include "TimerOne.h"

#define LIMITE_LUZ 30 //valor entre 0 e 1023 para separar entre nivel alto e baixa
#define TAMANHO_DADO 4 //tamanho do pacote
#define TAMANHO_PACOTE (2*TAMANHO_DADO)
#define TEMPO 2000 //indica o tempo em us
#define PIN_LED 3

int      limite_luz=30,media=0,repete=0,umBit=0,enviados=0,total=0,certos=0,errados=0;
//CALIBRAÇÃO, adicionei media e limite_luz //contador do laço do transmissor de não enviar
dados
unsigned long t1,t2,t1_inicial,t2_inicial;
volatile int stateT=0,stateR=0;
volatile int flag_envio=1;
volatile int countT,countR;
int pacoteT[TAMANHO_PACOTE];
int dadoT[TAMANHO_DADO]={0,1,1,0};
int pacoteR[TAMANHO_PACOTE];
int dadoR[TAMANHO_DADO];

void registra(){
    int conta=0;
    noInterrupts();
    total++;
    flag_envio=0;
    // interrupts();
    for(int i=0;i<TAMANHO_DADO;i++){
        if(dadoR[i]==dadoT[i]){
            // noInterrupts();
            conta++;
            // interrupts();
        }
    }
    if(conta==TAMANHO_DADO){
        //noInterrupts();
        certos++;
        //interrupts();
    }else if(conta==(TAMANHO_DADO-1)){
        //noInterrupts();
        umBit++;
        //interrupts();
    }else{
        // noInterrupts();
    }
}
```



```

    errados++;
    // interrupts();
}
if((total==1000)or(enviados==1000)){
Serial.print("T ");
Serial.print(enviados);
Serial.print(" R ");
Serial.print(total);
Serial.print(" C ");
Serial.print(certos);
Serial.print(" B ");
Serial.print(umBit);
Serial.print(" E ");
Serial.print(errados);
Serial.print("\n");
enviados=0;
total=0;
certos=0;
umBit=0;
errados=0;
}
//noInterrupts();
flag_envio=1;
interrupts();
}

void manchesterR(){
    int j=0;
    for(int i=0;i<TAMANHO_PACOTE;i=i+2){
        dadoR[j]=pacoteR[i];
        //Serial.print(dadoR[j]);
        j++;
    }
    //Serial.print("\n");
}

void manchesterT(){
    int j=0;
    for(int i=0; i<TAMANHO_PACOTE; i++){
        if(i%2==0){
            pacoteT[i]=dadoT[j];
        }else{
            pacoteT[i]=not(dadoT[j]);
        }
        j++;
    }
}
}

```

```

void transmissor(){
  switch(stateT){
    case 1:          // envia 0 do bit 1, se tiver dado pra enviar vai pro 2, se não volta pro
1(Default que manda 1 do bit 1)
    digitalWrite(PIN_LED,LOW);
    if((repete==1)and(flag_envio)){
      noInterrupts();
      stateT=2;
      repete=0;
      interrupts();
      break;
    }else{
      noInterrupts();
      stateT=0;
      repete++;
      interrupts();
      break;
    }

    case 2:          //envia 0, do bit 0
    digitalWrite(PIN_LED,LOW);
    noInterrupts();
    stateT=3;
    interrupts();
    break;

    case 3:          //envia 1, do bit 0, e codifica o dado
    digitalWrite(PIN_LED,HIGH);
    manchesterT();
    noInterrupts();
    countT=0;
    stateT=4;
    interrupts();
    break;

    case 4:          //envio do dado, apos retorna para o default
    digitalWrite(PIN_LED,pacoteT[countT]);
    noInterrupts();
    countT++;
    interrupts();
    if(countT>=TAMANHO_PACOTE){
      noInterrupts();
      stateT=0;
      enviados++;
      interrupts();
    }
  }
}

```

```

    break;

    default:          //case 0, envia 1 do bit 1
    digitalWrite(PIN_LED,HIGH);
    noInterrupts();
    stateT=1;
    interrupts();
    break;
}
}

int recebeSimbolo(){
    int x;
    digitalWrite(2,HIGH);
    x=analogRead(A0);
    x=x+analogRead(A0);
    // dado=dado+analogRead(A0);
    digitalWrite(2,LOW);
    // dado=dado/2;
    if(x>=2*limite_luz)
        x=1;
    else
        x=0;
    return x;
}

int recebe(){
    int x;
    digitalWrite(2,HIGH);
    x=analogRead(A0);
    x=x+analogRead(A0);
    x=x>>1;
    return x;
}

void setup() {
    Serial.begin(230400);
    pinMode(2,OUTPUT);
    pinMode(PIN_LED,OUTPUT);
    Timer1.initialize(0);
    Timer1.setPeriod(TEMPO);
    Timer1.attachInterrupt(transmissor);
}

void loop() {
    switch(stateR){
        case 1:          //recebe o 0 do bit 1

```

```

// Serial.println("B");
while(1){
    if(recebeSimbolo()==0){
        // limite_luz=(LIMITE_LUZ+((media+recebe())>>1)); //CALIBRAÇÃO, define o limite
        de luz como a media entre o nivel alto no case 0 e nivel baixo no case 1 e soma com o "ganho"
        escolhido
        //Serial.println(limite_luz);
        t1_inicial=micros();
        break;
    }
}
noInterrupts();
stateR=2;
interrupts();
break;

case 2:      //recebe o 1 do bit 1
//Serial.println("C");
while(1){
    if(recebeSimbolo()){
        t1=micros()-t1_inicial;
        break;
    }
}
noInterrupts();
stateR=3;
interrupts();
break;

case 3:      //recebe o 0 do bit 0
//Serial.println("D");
while(1){
    if(recebeSimbolo()==0){
        t2_inicial=micros();
        break;
    }
}
noInterrupts();
stateR=4;
interrupts();
break;

case 4:
//Serial.println("E"); //recebe o 1 do bit 0
while(1){
    if(recebeSimbolo()){
        t2=micros()-t2_inicial;

```

```

        break;
    }
}
if(t2>(1.5*t1)){
    noInterrupts();
    stateR=5;
    countR=0;
    interrupts();
    break;
}else{
    noInterrupts();
    stateR=0;
    interrupts();
    break;
}

case 5:
    //Serial.println("F");
    delayMicroseconds(t1);
    pacoteR[countR]=recebeSimbolo();
    noInterrupts();
    countR++;
    interrupts();
    if(countR>=TAMANHO_PACOTE){
        manchesterR();
        registra();
        noInterrupts();
        stateR=0;
        interrupts();
    }
    break;

default:          //aguarda o simbolo 1 do bit 1
    //Serial.println("A");
    while(1){
        if(recebeSimbolo()){
            // media=recebe(); //CALIBRAÇÃO
            break;
        }
    }
    noInterrupts();
    stateR=1;
    t1=0;
    t2=0;
    interrupts();
    break;
}

```

}