

Willy Ferraz de Oliveira

**Desenvolvimento de comunicação
para sistemas de iluminação
distribuídos baseado no protocolo
DMX-512A**

Trabalho de Conclusão de Curso
apresentado à Escola de Engenharia
de São Carlos, da Universidade de
São Paulo

Curso de Engenharia Elétrica com
ênfase em eletrônica

ORIENTADORA: Prof.^a Luiza Maria Romeiro Codá

São Carlos

2011

AUTORIZO A REPRODUÇÃO E DIVULGAÇÃO TOTAL OU PARCIAL DESTES TRABALHOS, POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica preparada pela Seção de Tratamento
da Informação do Serviço de Biblioteca – EESC/USP

Oliveira, Willy Ferraz de

048d

Desenvolvimento de comunicação para sistemas de iluminação distribuídos baseado no protocolo DMX-512A. / Willy Ferraz de Oliveira ; orientador Luiza Maria Romeiro Codá -- São Carlos, 2011.

Monografia (Graduação em Engenharia Elétrica com ênfase em Eletrônica) -- Escola de Engenharia de São Carlos da Universidade de São Paulo, 2011.

1. DMX512. 2. LED. 3. Iluminação. 4. Pista de dança. 5. PWM. I. Título.

FOLHA DE APROVAÇÃO

Nome: Willy Ferraz de Oliveira

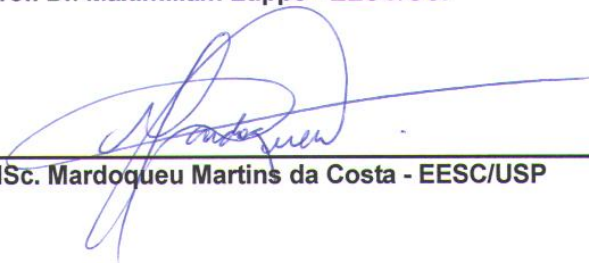
Título: “Desenvolvimento de Comunicação para Sistemas de Iluminação Distribuídos Baseado no Protocolo DMX-512A”

Trabalho de Conclusão de Curso defendido e aprovado
em 30/11/2011,

com NOTA 8,8 (oito, oito), pela comissão julgadora:



Prof. Dr. Maximilian Luppe - EESC/USP



MSc. Mardoqueu Martins da Costa - EESC/USP



Prof. Associado Homero Schiabel
Coordenador da CoC-Engenharia Elétrica
EESC/USP

Dedicatória

Dedico esse trabalho aos meus pais que me forneceram condições para que pudesse me desenvolver e atingir esse objetivo. Dedico-o, também, à Dayse Barreiros pelo apoio nos momentos mais difíceis.

Agradecimentos

Agradeço a Deus que nunca me desamparou;

Agradeço a minha família que é a razão do meu existir;

Agradeço à Dayse Barreiros que me motiva a seguir em frente;

Agradeço aos meus amigos da engenharia elétrica e irmãos de República Gato Preto;

Agradeço ao meu amigo Flávio Pascoal Vieira pela grande ajuda e por acreditar nesse projeto;

Agradeço aos meus professores que tanto se importaram em ensinar, em especial agradeço a professora Luiza Maria Romeiro Codá por ser minha orientadora.

Agradeço ao Mardoqueu Martins da Costa por ter acreditado em mim e pela oportunidade oferecida.

Sumário

Lista de figuras	X
Lista de tabelas.....	XII
Lista de siglas.....	XIII
Resumo	XIV
<i>Abstract</i>	XV
1. Introdução.....	1
1.1 - Sistemas de iluminação cênica.....	1
1.2 - Controle de sistemas de iluminação cênica.....	2
1.3 - Motivação do projeto	3
2. Objetivo	6
3. Fundamentação teórica	7
4.1 - Visão geral do DMX512A	7
4.2 - Topologia de Rede.....	7
4.3 - Características Elétricas e conectores	9
4.4 – Protocolo	11
4.5 - Outras Considerações – DMX512A.....	13
4. Especificações, materiais e desenvolvimento	15
5.1 - Especificações do sistema	15
5.2 - Protocolo específico	17
5.3 - Diagrama de Blocos.....	20
5.4 - <i>Transceiver</i> - SN75176a	22
5.5 - Microcontrolador – PIC 18F24K22	22
5.6 - Controlador PWM – PCA9626B	23
5.7 - Prova de conceito	25
5.8 - Desenvolvimento do protótipo DMX24PWM.....	33
5.9 - Desenvolvimento da interface com computador.....	35
5. Resultados.....	37
5.1 - DMX512A.....	37
5.2 – PCA9626.....	38
5.3 - Prova Conceito.....	39
5.4 - Protótipo DMX24PWM	40
5.5 - Protótipo reduzido da pista de dança	41
5.6 - <i>Software</i> de controle	41
5.7 - Resultado Final	42

5.8 - Discussão dos resultados	44
6. Conclusão.....	45
Referências Bibliográficas.....	47
Apêndice.....	49
Apêndice A – Descrição da pinagem do PIC 18F24K22.....	49
Apêndice B – Registradores de controle do PCA9626.	51
Apêndice C – Software do DMX24PWM.	52
Apêndice D – Driver para acesso do PCA9626	54
Apêndice E – Diagrama Elétrico.....	58

Lista de figuras

Figura 1 - Equipamento de iluminação cênica: a) Canhão de Led b) <i>Moving Head</i> c) <i>Laser</i> d) Ribalta de Leds e) <i>Strobo</i> f) <i>Jarag</i>	1
Figura 2 - Comparação entre 0-10V e DMX512A para um sistema com 30 sinais de controle.....	3
Figura 3 - Pista de dança com iluminação de piso dos anos 70. Cena retirada do filme “ <i>Saturday Night Fever</i> ”, 1977, <i>Paramount Pictures</i>	4
Figura 4 - Pista de Dança moderna iluminada por Leds.....	5
Figura 5 – Diagrama de posicionamento do DMX24PWM no segmento DMX512.	6
Figura 6 - Topologia <i>Dayse chain</i> [7]	8
Figura 7 - Operação diferencial utilizada no RS-485 [7]......	9
Figura 8- a) conector XLR-5. b) conector XLR-3.	10
Figura 9 - a) pinos XLR-5. b) pinos XLR-3.	11
Figura 10 - <i>Frame</i> do protocolo DMX512A [8]......	12
Figura 11 - <i>Moving Head</i> , dispositivo adequado ao padrão DMX512A.	13
Figura 12 – Exemplo de endereçamento no protocolo DMX512A.	14
Figura 13 - Proposta física de um módulo da pista de dança, vista superior.	16
Figura 14 - Tabela conversão de cores do protocolo específico.	18
Figura 15 - Tabela de conversão da intensidade de brilho para o vermelho.	20
Figura 16 - Diagrama de blocos do DMX24PWM.....	20
Figura 17 - SN75176a, responsável por digitalizar o sinal analógico do protocolo DMX512A.	22
Figura 18 - PIC 18F24K22, microcontrolador responsável por interpretar o protocolo DMX512 e comandar as etapas de saída [9].	22
Figura 19 - PCA 9626B [10].....	24
Figura 20 - <i>Frame</i> da comunicação I ² C [10].....	24
Figura 21 - Mesa DMX OPERATOR III.....	26
Figura 22 - Analisador Lógico Saleae logic.	27
Figura 23 - Fluxograma simplificado do processo de recepção dos dados da rede.	28
Figura 24 - Segmento de fita de LEDs RGB.	29
Figura 25 - Fluxograma simplificado da conversão do protocolo específico.....	31
Figura 26 – Visão geral da utilização da prova de conceito.....	32
Figura 27 – Composição dos blocos do projeto.	32
Figura 28 – <i>Print</i> preliminar da placa de circuito impresso.	33
Figura 29 – Projeto mecânico do protótipo da pista de dança.....	34
Figura 30 – Construção física do protótipo reduzido de um módulo da pista de dança.	35
Figura 31 - Desenvolvimento do modelo do protótipo para utilização em <i>software</i>	36
Figura 32 - Conversor USB para DMX512.	36
Figura 33 - <i>Frame</i> de 192 Canais da Mesa DMX OPERATOR III.	37
Figura 34 - Sete primeiros canais da mesa DMX OPERATOR III	37
Figura 35 - Linha I2C de comunicação entre o microcontrolador e o PCA 9626.	38
Figura 36 - Cores obtidas utilizando a prova de conceito.....	39
Figura 37 - 4 amostras de intensidade obtidas utilizando a prova de conceito.	39
Figura 38 - Placa de circuito impresso do DMX24PWM.....	40
Figura 39 - Resultado final do DMX24PWM.....	40
Figura 40 - Módulo da pista de dança em escala reduzida.	41

Figura 41 - Tela do <i>software Freestyler 512</i> aplicado ao controle da pista de dança. .	42
Figura 42 - Resultado final das cores geradas.	43
Figura 43 - Amostras de intensidades obtidas.	43
Figura 44 - Visão Geral de funcionamento do sistema.	43
Figura 45 - Diagrama Elétrico - Parte A.	59
Figura 46 - Diagrama Elétrico - Parte B.	60
Figura 47 - Diagrama Elétrico - Parte C.	61

Lista de tabelas

Tabela 1 - Demais características RS-485.....	9
Tabela 2 - Conexões XLR-5 e XLR-3.....	11
Tabela 3 - Temporização do protocolo DMX512A.....	13
Tabela 4 - Alocação do DMX24PWM na rede DMX512 utilizando o protocolo específico para controle da pista de dança.	17
Tabela 5 - Intensidade RGB para formação das cores compostas pela pista de dança.	19
Tabela 6 - Comparação de temporização do protocolo DMX512	38

Lista de siglas

DMX: *Digital multiplex.*

LED: *Light-emitting diode.*

PWM: *Pulse width modulation.*

RGB: *Red, green and blue.*

PCB: *Printed Circuit Board.*

I²C: *Inter-Integrated Circuit.*

PIC: *Programmable Interface Controller.*

MIPS: *Milhões de Instruções Por Segundo.*

USB: *Universal Serial Bus.*

Resumo

Desenvolvimento de comunicação para sistemas de iluminação distribuídos baseado no protocolo DMX-512A

Essa monografia apresenta um protótipo chamado de DMX24PWM, desenvolvido para controle de intensidade de cores e luminosidade, com 24 canais de saída PWM (*Pulse width modulation*) aplicados a LEDs RGB, responsáveis pela iluminação inferior de uma pista de dança utilizando o protocolo de comunicação DMX512A, amplamente difundido em sistemas de iluminação distribuídos. Através de um protocolo específico desenvolvido, o DMX24PWM tem capacidade de controlar a intensidade e cor de um segmento de 1m² de pista da dança pelas especificações do projeto. Foi implementada a comunicação entre uma mesa de controle DMX512 comercial e o protótipo desenvolvido. Foi construído um protótipo reduzido da pista de dança para verificação da viabilidade do projeto. Para integração a todas as tendências de utilização do mercado, o sistema foi adaptado para que fosse controlado por um microcomputador através de uma interface USB. O *hardware* montado permite, com simples atualizações de *firmware* a criação de soluções customizadas, atendendo à demandas específicas, podendo ser utilizado em projetos de iluminação decorativa de ambientes.

Palavras Chave: DMX512, LED, iluminação, pista de dança, PWM.

Abstract

Development of communication for distributed lighting systems based on the DMX-512A.

This monograph presents a prototype called the DMX24PWM developed for control of color intensity and brightness, with 24 channels of PWM (Pulse width modulation) applied to RGB LEDs, responsible for lighting bottom of a dance floor using the communication protocol DMX512A , widely distributed in lighting systems. Through a specific protocol developed, DMX24PWM has the ability to control the intensity and color of a segment of 1m2 dance track by the project specifications. Was implemented communication between a DMX512 commercial desk control and the prototype. It was built a small prototype of the dance floor to check the feasibility of the project. For integration to all usage trends of the market, the system was adapted to be controlled by a microcomputer via a USB interface. The mounted hardware allows, by simple firmware upgrades to create customized solutions, given the specific demands and can be used in the design of decorative lighting environments.

Keywords: DMX512, LED, lighting, dance floor, PWM.

1. Introdução

1.1 - Sistemas de iluminação cênica

A iluminação constitui um dos principais fatores para a criação de um ambiente cênico diferenciado. Através de seus recursos é possível criar diferentes situações e transmitir sensações aos espectadores ou utilizadores de um determinado ambiente. Esse setor tem recebido atenção especial dos produtores de eventos por permitir uma interação visual com o público; e o grande interesse por esse ramo leva em consideração que a capacidade de inferir o sentimento de satisfação está intimamente ligada ao sentido da visão humana [1]. A composição cênica ainda utiliza outros recursos físicos para realçar os efeitos ópticos como o uso de máquinas de fumaça. Atendendo a essa demanda diversos dispositivos estão disponíveis no mercado para comporem as mais variadas cenas, entre os mais comuns dispositivos de iluminação cênica podemos citar: Canhões de Led, *Moving Heads*, *Lasers*, Ribalta de Leds, *Strobo*, *Jarag*. Os equipamentos são mostrados na figura 1.



Figura 1 - Equipamento de iluminação cênica: a) Canhão de Led b) *Moving Head* c) *Laser* d) Ribalta de Leds e) *Strobo* f) *Jarag*

Existem outros dispositivos à disposição no mercado, muitos deles, variações dos apresentados na figura 1 que proporcionam amplas opções de montagens e

possibilidades de criação de cenas complexas fazendo uso de vários componentes simultaneamente. Essa gama de opções também remete a complexidade de controlar todo esse sistema de maneira conjunta, eficiente e com sincronismo.

1.2 - Controle de sistemas de iluminação cênica

Por muitos anos o controle de sistemas de iluminação cênica foi atuado de maneira precária, apenas com a evolução dos *dimmers* houve um ganho significativo em relação ao histórico desse segmento. Com desenvolvimento da eletrônica, principalmente os controladores, muitos métodos surgiram para suprir a carência de algum protocolo eficiente. Dentre eles o que se tornou mais popular foi o “0-10V”, mesmo assim ainda não houve padronização eficiente do método, o que causava dificuldades na popularização maciça e travamento no fluxo do mercado de equipamentos de iluminação. O método “0-10V” é fundamentado em um sinal de controle em tensão CC que varia de 0 a 10V no qual, por exemplo, 0V pode significar uma lâmpada apagada e 10V uma lâmpada com intensidade máxima. O funcionamento simplificado do sistema era um ponto positivo a seu favor, entretanto, não foi suficiente para compensar os problemas de volume excessivo de cabos, visto que cada canal de um determinado equipamento precisava de uma conexão direta e exclusiva, o que encarecia demasiadamente o sistema e exigia cabos com muitas vias para a interligação dos dispositivos. Além de limitar as distâncias devido às quedas de tensões do sinal de controle, a inserção de ruídos e interferências também influenciavam negativamente o método [2].

Com base na ineficiência dos métodos utilizados e na falta de padronização dos fabricantes a *Unites States Institute of Theatre Technology (USITT)* criou, em 1986, um padrão de transmissão de dados entre controladores de equipamentos de iluminação e dispositivos de atuação, com características elétricas baseadas na norma EIA/TIA-485 e chamou o método de DMX512 que é proveniente de “*Digital Multiplex with 512 individual pieces of information*”. Em 1990 a USITT fez revisões no método e lançou uma nova versão chamada DMX512/1990. Em 1998 a USITT transferiu a manutenção do padrão para a *Entertainment Services and Technology Association (ESTA)* que realizou novas revisões no padrão, e este foi aprovado pela American National Standards Institute (ANSI) em 2004. O padrão, cuja nomenclatura é E.1.11, passou a se chamar DMX512-A - *Asynchronous Serial Digital Data Transmission Standard for Controlling Lighting Equipment and Accessories (DMX512-A – Transmissão serial de dados digitais assíncrona para controle de equipamentos de*

iluminação), ou simplesmente DMX512A [3]. Esse padrão tornou-se dominante no mercado devido a sua relativa simplicidade e por suportar até 512 canais de comunicação em uma única rede, simplificando as ligações e reduzindo drasticamente a quantidade de cabos. A sua última revisão aconteceu em 2008, sendo a sua versão mais atual a E1.11 - 2008, USITT DMX512-A. A figura 2 mostra um comparativo resumido entre as tecnologias “0-10V” e DMX512A. O protocolo de comunicação e as características do padrão DMX512-A serão apresentadas no capítulo 3.

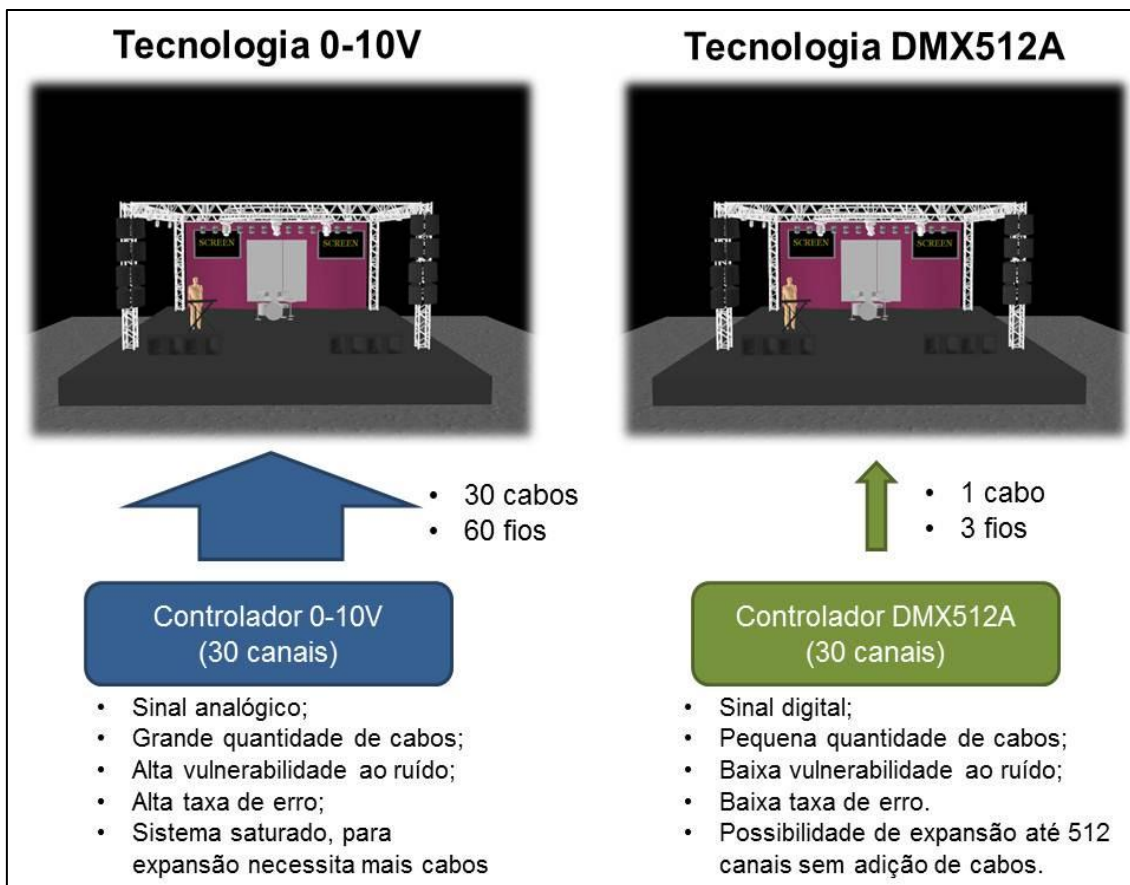


Figura 2 - Comparação entre 0-10V e DMX512A para um sistema com 30 sinais de controle.

1.3 - Motivação do projeto

Com a popularização das discotecas, crescente nos anos 70, a iluminação cênica, que antes era praticamente dedicada a teatros, passou a ser aplicada em ambientes voltados à música e dança. Frequentada por jovens, em sua maioria, essas casas tornaram-se um fenômeno mundial e conseqüentemente um mercado rentável. A figura 3 mostra a cena de um filme que fez muito sucesso à época. Esse crescimento fez com que fossem desenvolvidos equipamentos específicos para esse segmento de mercado. Um dos equipamentos mais inovadores à época eram as

pistas de dança com iluminação de piso, semelhantes às pistas convencionais, porém possuíam lâmpadas coloridas abaixo do piso translúcido. A baixa durabilidade desse tipo de lâmpadas e elevada potência do sistema completo tornavam esse tipo pista extremamente caras, encontradas apenas em casas noturnas mais sofisticadas.



**Figura 3 - Pista de dança com iluminação de piso dos anos 70.
Cena retirada do filme “Saturday Night Fever”, 1977, Paramount Pictures.**

O desenvolvimento e popularização dos diodos emissores de luz, LED, permitiu que fossem desenvolvidos novos modelos de pistas de dança utilizando essa tecnologia e com as seguintes vantagens:

- Baixo consumo de energia;
- Alta eficiência energética;
- Baixo custo;
- Facilidade de controle.

A popularização do protocolo DMX512 tornou atrativo esse sistema de controle para essas novas pistas de dança, fazendo com que esse equipamento voltasse ao mercado. A figura 4 mostra uma pista de dança, atualmente disponível no mercado, com tecnologia LED.



Figura 4 - Pista de Dança moderna iluminada por Leds.

O Brasil pode ser considerado um mercado em expansão nesse segmento, já que poucos estabelecimentos possuem esse tipo de equipamento, e a maioria do material existente é proveniente de importação [4]. Atualmente é possível encontrar no mercado brasileiro segmentos de 1m² de pista de dança iluminada por leds ao preço próximo de R\$4400,00 ¹.

Com a motivação da possibilidade de espaço no mercado nacional promoveu-se a elaboração do projeto descrito no decorrer desse trabalho como um estágio inicial para construção de uma pista de dança iluminada por leds.

¹ Preço referente ao modelo Pista de LED American Pro LED21010 – R\$ 4.462,00 em < <http://www.lumixpro.com.br> > acessado em 3 de outubro de 2011.

2. Objetivo

Levando em conta a demanda de mercado citada no Capítulo 1, esse trabalho tem como principal objetivo desenvolver um protótipo, que foi batizado de **DMX24PWM**. Esse protótipo deve ser capaz de comunicar-se com mesas e *softwares* comerciais de controle de iluminação e acionar uma pista de dança controlando sua cor e intensidade. Essa comunicação deve ser feita através do protocolo dominante no mercado de iluminação distribuída, regulamentado pela ANSI E1.11 DMX512A e deve funcionar de maneira eficiente para que o DMX24PWM possa cumprir o propósito de se tornar um equipamento comercial e devidamente adequado ao padrão.

O DMX24PWM foi desenvolvido para ser um *hardware* genérico, que com simples modificações no *firmware* possa atender a situações além da sua concepção original de gerir a interface entre um controlador comercial e uma pista de dança através de 24 saídas PWM. Relacionando os objetivos centrais do DMX24PWM a figura 5 mostra o posicionamento do dispositivo desenvolvido no segmento DMX512.

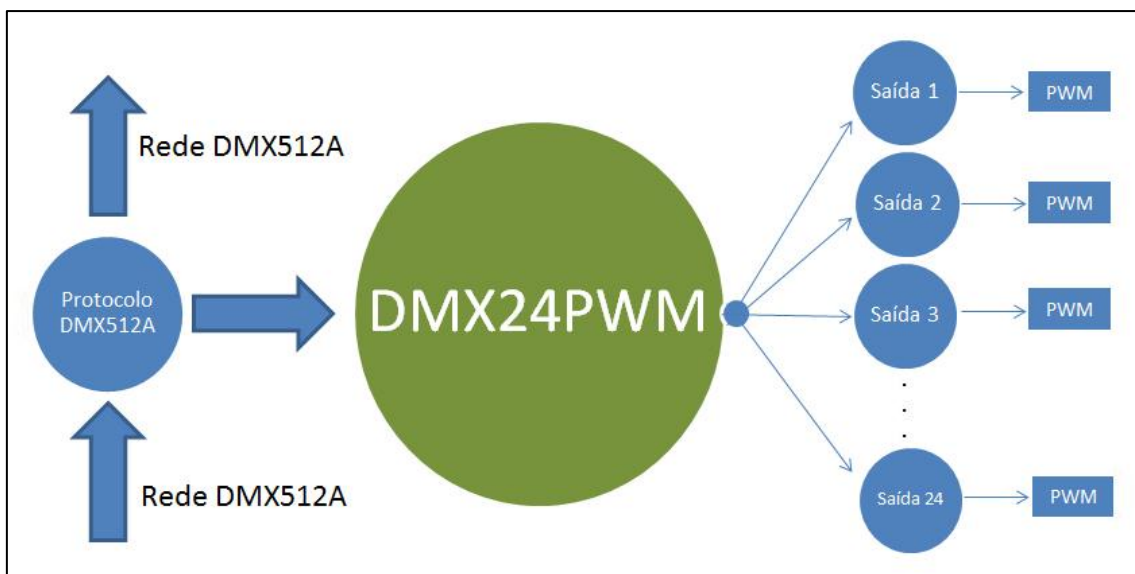


Figura 5 – Diagrama de posicionamento do DMX24PWM no segmento DMX512.

3. Fundamentação teórica

4.1 - Visão geral do DMX512A

Como descrito na introdução, o DMX512 surgiu como protocolo forte em um mercado que carecia de padronização e eficiência. Com a aceitação pela ANSI em 2004 tornou-se praticamente exclusivo para sistemas de iluminação distribuídos. O DMX512A utiliza uma comunicação unidirecional, e por não haver detecção de erros inerentes, não há total garantia de que os pacotes foram entregues corretamente. Por esse motivo não pode ser utilizado para acionar cargas perigosas [5]. O acionamento de cargas pirotécnicas ou movimentação de palcos, por exemplo, são proibidos pela norma.

Sua camada física é gerida pelo padrão TIA/EIA-485, também conhecido como RS-485 [6].

O sinal gerado por um transmissor DMX512A traz, em seu pacote, informações referentes a 512 canais, cada canal contendo um byte, ou seja, oito (8) bits de informação.

Os frames são transmitidos de maneira serial e assíncrona a uma taxa de 250kbps.

Os conceitos citados nesse capítulo são referenciados [6] pelo padrão ANSI E1.11 - 2008 - *Entertainment Technology - USITT DMX512-A, Asynchronous Serial Digital Data Transmission Standard for Controlling Lighting Equipment and Accessories*.

4.2 - Topologia de Rede

Os dispositivos que utilizam o padrão DMX512A são ligados utilizando uma topologia comumente chamada de “*Dayse chain*”[7] , descrita pela figura 6.

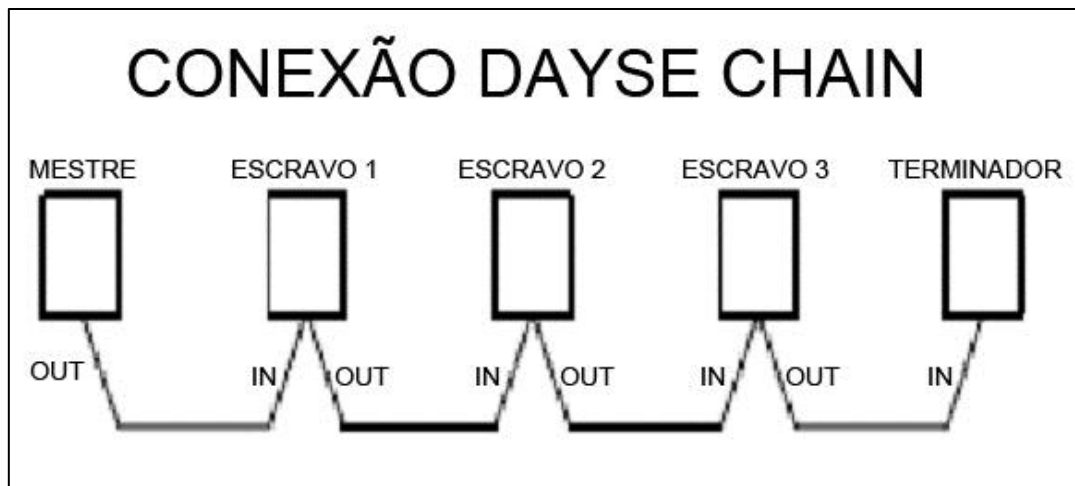


Figura 6 - Topologia *Dayse chain* [7]

A rede é formada por um controlador DMX512A, considerado o mestre e o demais dispositivos receptores são “escravos”. Cada um desses dispositivos possui um conector de entrada “*IN*”, através dele os dados do pacote são assimilados e continuam o tráfego na rede pela saída “*OUT*”. A norma USITT E1.11 DMX512A [6] recomenda que as conexões *IN* sejam formadas conectores “machos” e as conexões *OUT* por conectores “fêmeas”. Esses conectores serão abordados posteriormente. A norma também recomenda que a cada, no máximo, 32 dispositivos escravos seja adicionado à cadeia um repetidor, comumente chamado de *splitter*.

Cada equipamento escravo possui um endereço na rede que pode variar de 1 a 512, esse endereço é fundamental para que o equipamento entenda qual pacote de informação lhe diz respeito. Até 512 equipamentos diferentes podem ser controlados, entretanto se mais de um equipamento for configurado com o mesmo endereço ambos serão controlados igualmente. Essa possibilidade torna-se interessante quando é necessário controlar dispositivos idênticos.

Outra característica desse tipo de ligação é a utilização de um terminador ao final de cadeia, a norma define um resistor de 120Ω de pelo menos $\frac{1}{4}$ de watt [6]. Faz-se necessário para evitar reflexão na linha de comunicação RS-485. Essa possível reflexão gera erros na decodificação do sinal e esse recurso deve ser utilizado mesmo que haja apenas um dispositivo escravo na cadeia.

4.3 - Características Elétricas e conectores

O padrão DMX512A utiliza as especificações elétricas da norma TIA/EIA-485 [6]. Sua transmissão serial e assíncrona consiste em um par trançado, blindado, com operação diferencial. A figura 7 exemplifica esse tipo de transmissão.

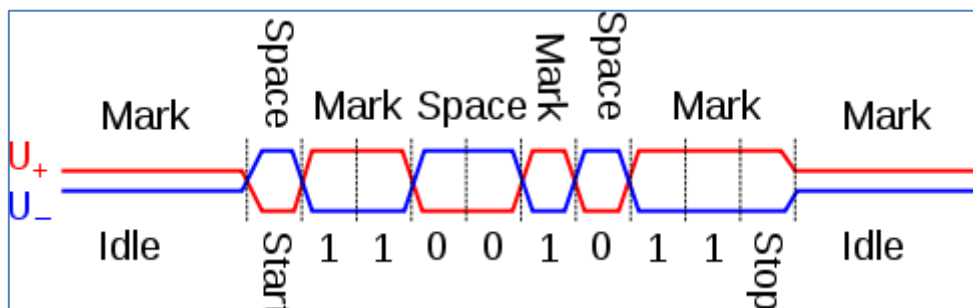


Figura 7 - Operação diferencial utilizada no RS-485 [7].

O nível lógico alto (1 ou *mark*) é reconhecido quando U_+ tem tensão maior que U_- e nível lógico baixo (0 ou *space*) é reconhecido quando U_- tem tensão maior que U_+ . Esse funcionamento apresenta grande vantagem quando há ruídos inseridos ao sistema, pois o ruído afeta U_+ e U_- de maneira igual e como sistema responde baseado na diferença entre eles há pouco prejuízo no saída do comparador. Esse sinal é obtido através de um *transceiver* diferencial.

A tabela 1 mostra outras características elétricas dessa comunicação [7]:

Tabela 1 - Demais características RS-485.

Tensão máxima	+12V
Tensão Mínima	-7V
Diferença Mínima para ALTO entre U_+ e U_- .	200mV
Diferença Mínima para BAIXO entre U_+ e U_- .	-200mV

A taxa mínima da RS-485 é calculada através da equação (1) e leva em consideração a taxa de transmissão (T), distância máxima (D) entre os pontos de comunicação.

$$D < \frac{10^8}{T} \quad (1)$$

Como a taxa mínima da comunicação RS-485 é de 100kbps a distância máxima D é de 1000m; como o DMX512A tem taxa de 250kbps a distância real máxima é de 400m.

Desde a revisão da norma DMX512/1990 os conectores XLR-5, mostrado na figura 8a, são os conectores definidos pelo padrão. Constituído de um pino para o aterramento, um par diferencial para dados, um segundo par para dados secundários que também pode ser usado como uma extensão da rede. Entretanto, alguns fabricantes utilizam os conectores XLR-3, mostrado na figura 8b eliminando o par secundário de dados, visto que poucos dispositivos utilizam esse recurso. Esse tipo de conector é bem aceito por possuir robustez e uma trava de segurança, qualidades que agregam valores em conexões que exigem montagens e desmontagens sucessivas.



Figura 8- a) conector XLR-5. b) conector XLR-3.

A tabela 2 traduz os pinos de ambos conectores e a figura 9 o arranjo físico do conector.

Tabela 2 - Conexões XLR-5 e XLR-3.

XLR-5			XLR-3		
Pino	Conexão	Sinal	Pino	Conector	Sinal
1	Blindagem	Terra 0V	1	Blindagem	Terra 0V
2	Preto	Dado -	2	Preto	Dado -
3	Branco	Dado +	3	Branco	Dado +
4	Verde	Dado Secundário +			
5	Vermelho	Dado Secundário -			

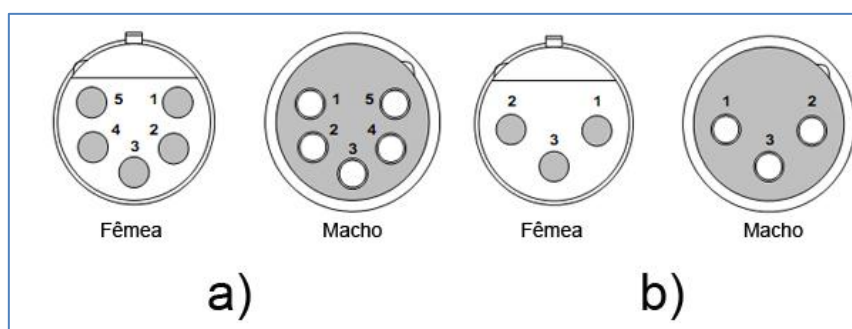


Figura 9 - a) pinos XLR-5. b) pinos XLR-3.

As normas de cabeamento foram removidas do padrão E1.11 - 2008, USITT DMX512-A e lançadas em outro padrão exclusivo para tratar da questão. A ANSI E1.27-1 exige que seja utilizado cabos com par trançando para os dados, blindado, com impedância entre 110Ω e 120Ω , capacitância entre os condutores menor que 25pF e entre o condutor e a blindagem menor que 40pF . O cabo condutor precisa ser robusto suficientemente para suportar as condições em que são comumente utilizados: ambiente sujeito a impactos, forças externas e intempéries.

4.4 – Protocolo

O protocolo DMX512A utiliza uma transmissão de dados seriais e assíncronos a 250 kbaud, repetitiva, contendo até 512 bytes de informação e seus elementos de controle para parada e identificação de um novo ciclo. O *frame* carrega conteúdo para os possíveis 512 canais suportados pelo protocolo, cada canal recebe um byte de informação, ou seja, um valor que pode variar de 0 a 255. O DMX512A é um protocolo simples, sem identificador de erro, mas com considerada eficiência. A figura 10 mostra o frame do protocolo DMX512A.

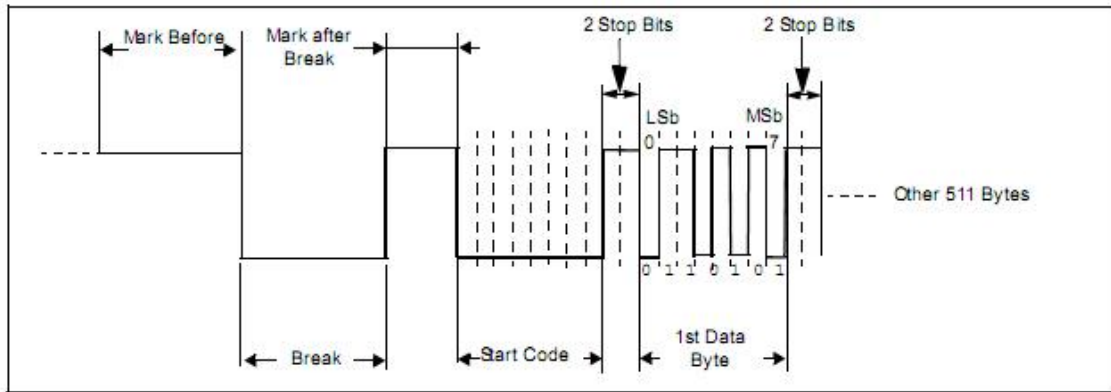


Figura 10 - *Frame* do protocolo DMX512A [8].

O padrão E1.11 - 2008, USITT DMX512-A [6] estabelece que cada bit do protocolo tenha largura típica de $4\mu\text{s}$. Cada ciclo de *frame* do protocolo é reconhecido por um período de *BREAK* em estado baixo seguido de um período em alto (*MARK AFTER BREAK*). Em seguida é iniciado o *START CODE*, definido para o DMX512A como 8 bits em nível baixo. Dois *STOP BITS* precedem o byte do primeiro canal. Sucessivamente são apresentados os 511 bytes restantes entrepostos com dois bits de *STOP BITS*. Após finalizar os 512 canais o sistema fica em alto no *MARK BEFORE BREAK* até que o novo *BREAK* comece nova atualização do frame.

- **Break:** O início do pacote DMX512 é iniciado pelo *Break*, caracterizado por um intervalo de, pelo menos, $92\mu\text{s}$. Como cada bit tem largura típica de $4\mu\text{s}$, o receptor precisa reconhecer, no mínimo, 23 bits em nível baixo para identificar o período de break.
- **Mark After Break:** Imediatamente após o *Break*, o *Mark After Break* (MAB) é caracterizado por um sinal alto de, no mínimo, 3 bits ou $12\mu\text{s}$.
- **Start Code:** É o primeiro pacote no formato de byte, a norma E1.11 - 2008, USITT DMX512-A estabelece que para aplicações de iluminação distribuída o byte *Start Code* deve ser nulo. Todos os bytes são seguidos, pelo menos, 2 stop bits em nível alto.
- **Stop Bit:** Os *stop bits* intercalam os bytes que contém a informação, possui largura mínima de 2 bits, ou seja, $8\mu\text{s}$.
- **Data Byte:** Após o stop bit iniciam-se os bytes de dados, que podem ser em até 512 dados, entretanto, um sistema pode não utilizar toda a extensão do protocolo. Os dados do *Data Byte* podem representar até 256 níveis diferente que serão decodificados pelo receptores e atuarão de acordo com os protocolos internos dos equipamentos.

- **Mark Before Break:** O *Mark Before Break* (MBF) é caracterizado por uma sequência de bits em nível alto que deve complementar o *frame* do protocolo para assegurar que ele não seja menor que 1204µs.

A largura do pacote DMX512A não pode durar mais do que 1 segundo, e vale a pena ressaltar que o *frame* não carrega informação sobre a numeração do pacote, essa identificação é feita através de um contador de pacotes no receptor. Ao contar a quantidade de bytes de informação recebidos identifica o canal e conseqüentemente avalia se corresponde um dado que lhe diz respeito.

A tabela 3 mostra os valores de temporização exigidos pela E1.11 - 2008, USITT DMX512-A.

Tabela 3 - Temporização do protocolo DMX512A.

Descrição	Mínimo	Típico	Máximo	Unidade
Break	92	176	-	µs
Mark After Break	12	-	< 1000000	µs
Bit	3,92	4	4,02	µs
Pacote DMX512A	1204	-	< 1000000	µs

Muitos equipamentos DMX512 obedecem a normas anteriores à atualização de 2008, conseqüentemente, algumas temporizações são diferentes. As maiores divergências são o tempo mínimo de *Break* (88µs) e *Mark After Break* (8µs).

4.5 - Outras Considerações – DMX512A

Cada canal DMX512 consegue representar até 256 níveis que são decodificados pelos protocolos internos dos dispositivos de iluminação, entretanto, alguns dispositivos precisam de mais resolução para traduzir sua aplicação física. Podemos citar, como exemplo, um *Moving Head* típico mostrado na figura 11.



Figura 11 - *Moving Head*, dispositivo adequado ao padrão DMX512A.

O *Moving Head* precisa controlar o posicionamento do feixe de luz na posição x-y, a intensidade da luz, cor, desenho do gobo e muitas outras variáveis de seu protocolo interno. Para isso é necessário que o aparelho utilize mais de um canal DMX512A, ou seja, um canal dedicado para controlar cada uma das necessidades citadas. Como consequência dessa alocação, o endereçamento do próximo dispositivo deve levar em conta a quantidade de canais utilizados pelo seu antecessor.

Baseado nas informações desse capítulo pode-se fazer uma analogia do funcionamento do DMX512A com um sistema de entrega de um carteiro responsável por uma rua que pode conter até 512 residências. Os dispositivos podem ser as casas ou até prédios, visto que alguns utilizam apenas um e outros uma sequência de endereços. O carteiro leva as correspondências à todas as casas mas apenas a carta destinada para aquele endereço específico é aberta.

A figura 12 exemplifica o funcionamento do protocolo DMX512A com dispositivos que demandam mais de um canal.

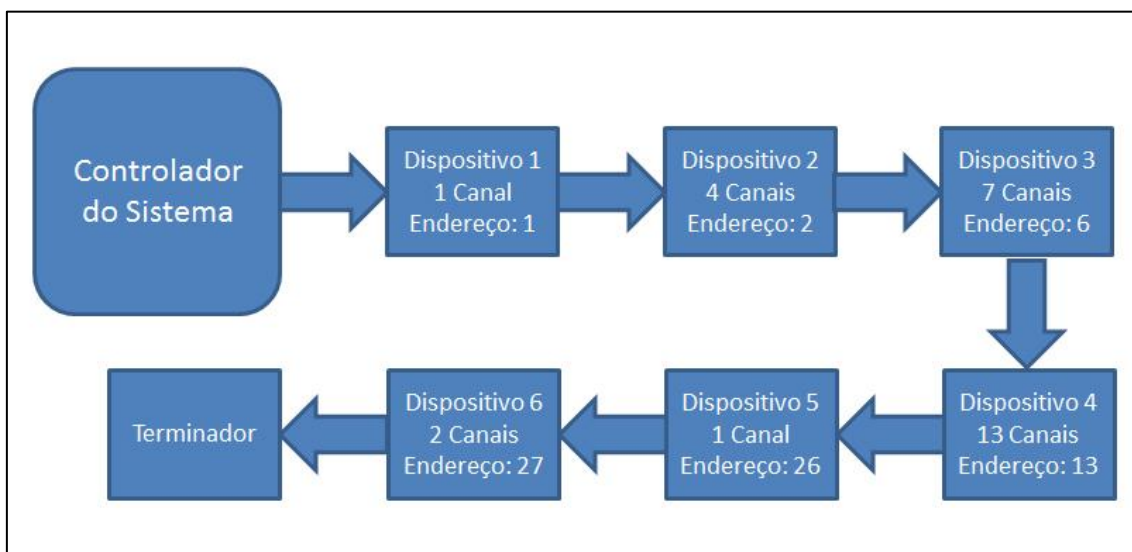


Figura 12 – Exemplo de endereçamento no protocolo DMX512A.

4. Especificações, materiais e desenvolvimento

5.1 - Especificações do sistema

Os objetivos centrais do projeto, descritos no capítulo 2, podem ser resumidas da seguinte maneira:

- Adequação ao padrão E1.11 - 2008, USITT DMX512-A;
- *Hardware* genérico, porém, capaz de controlar uma pista de dança.

E para tornar prático o desenvolvimento do DMX24PWM, protótipo resultado desse trabalho, foi necessário impor algumas diretrizes para nortear o projeto.

Diretrizes do *Hardware* Genérico:

- Possuir 24 saídas PWM incorporadas à placa do DMX24PWM;
- Cada PWM deve controlar um estágio de saída capaz de fornecer até 2A de corrente contínua;
- Possuir um conjunto de “*jumpers*” para seleção de *firmware* adequado à aplicação desejada;
- Possuir uma derivação anterior ao estágio de saída, proporcionando 24 saídas PWM que podem ser ligadas a uma segunda placa com driver de maior potência para aplicações variadas, ou acionamento de cargas com outras características, como, por exemplo, cargas CA.
- Possuir um botão de endereçamento para permitir vários DMX24PWM ligado à rede DMX512A.

Diretrizes aplicadas à Pista de Dança:

- Pista de dança com módulos de 1m².
- Junção de 32 módulos, sendo assim o tamanho máximo da pista seria 32m². Esse tamanho foi escolhido para que não houvesse necessidade de utilização de um *Splitter*, por ser considerado um tamanho grande o suficiente para o tipo de mercado desejado, e pela definição do protocolo específico do DMX24PWM definido posteriormente. Uma das possibilidades de extensão do tamanho seria a operação em modo espelho, com mais de um módulo no mesmo endereço da rede DMX512A.

- Cada módulo possui 8 segmentos triangulares, como mostra a figura13.

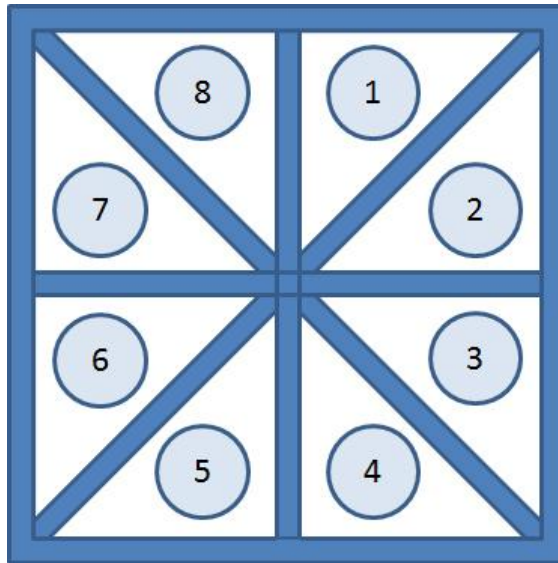


Figura 13 - Proposta física de um módulo da pista de dança, vista superior.

- Os segmentos triangulares possuem 16 leds RGB em cada uma das suas 3 laterais internas.
- Cada segmento deve ser controlado independentemente em cor e intensidade de brilho.

O trabalho descrito nessa monografia foi dividido, basicamente, em 6 etapas.

- Pesquisas iniciais focadas na concepção e necessidades do projeto bem como sua adequação às normas estabelecidas no mercado.
- Montagem em prova de conceito de sistema microprocessado para comunicação com dispositivos comerciais DMX512A.
- Pesquisa e estudo de componentes dedicados à geração de vários sinais de PWM.
- Desenvolvimento de protocolo específico para controle dos sistemas de iluminação distribuídos.
- Geração de resultados utilizando a prova de conceito e projeto de *hardware* do protótipo final.
- Construção de modelo de aplicação reduzido e propostas de novas implementações.

5.2 - Protocolo específico

O protocolo específico desenvolvido tem como função estabelecer a ponte entre o padrão DMX512A e a aplicação do projeto. O protocolo específico aplicado à pista de dança e contido no DMX24PWM utiliza dois canais DMX512 para cada segmento de um módulo de 1m² de pista de dança, logo para o controle de cor e intensidade de todo o módulo são necessários 16 canais. O primeiro canal é responsável pelo controle da cor do segmento, e o segundo canal é responsável pela intensidade dos LEDs. Esta sequência (1-Cor, 2-Intensidade) se repete até que o final dos 16 canais lidos por cada DMX24PWM. Se um dispositivo DMX24PWM for endereçado no canal 1 da rede DMX o próximo dispositivo deve estar endereçado no canal 17, pois o DMX24PWM utilizou os 16 primeiros canais para a tradução do seu protocolo específico.

Utilizando o conceito dos 512 canais sequenciais do protocolo DMX512A, podemos tornar genérica a integração do protocolo específico à rede DMX da maneira descrita pela tabela 4. Toma-se, por exemplo, um dispositivo alocado em um endereço N, que deve ser menor que 497, já que precisa estar contido nos 512 canais do protocolo.

Tabela 4 - Alocação do DMX24PWM na rede DMX512 utilizando o protocolo específico para controle da pista de dança.

Endereço do DMX24PWM na rede DMX512: N	
Canal	Conteúdo
N	Cor do segmento 1
N+1	Intensidade do segmento 1
N+2	Cor do segmento 2
N+3	Intensidade do segmento 2
N+4	Cor do segmento 3
N+5	Intensidade do segmento 3
N+6	Cor do segmento 4
N+7	Intensidade do segmento 4
N+8	Cor do segmento 5
N+9	Intensidade do segmento 5
N+10	Cor do segmento 6
N+11	Intensidade do segmento 6
N+12	Cor do segmento 7
N+13	Intensidade do segmento 7
N+14	Cor do segmento 8
N+15	Intensidade do segmento 8

Como cada canal tem 1 byte de informação é possível quantificar até 256 cores diferentes e 256 níveis de intensidade, entretanto, por considerar que haveria pouco ganho na aplicação específica da pista de dança e para simplificar o controle de um operador de mesa comercial distribui-se 16 cores nos 265 níveis possíveis. Ou seja, com um dado de 0-15 no byte do canal da cor o DMX24PWM responde ligando o segmento da pista em vermelho, de 16-31 responde em verde e assim sucessivamente como mostra a figura 14.

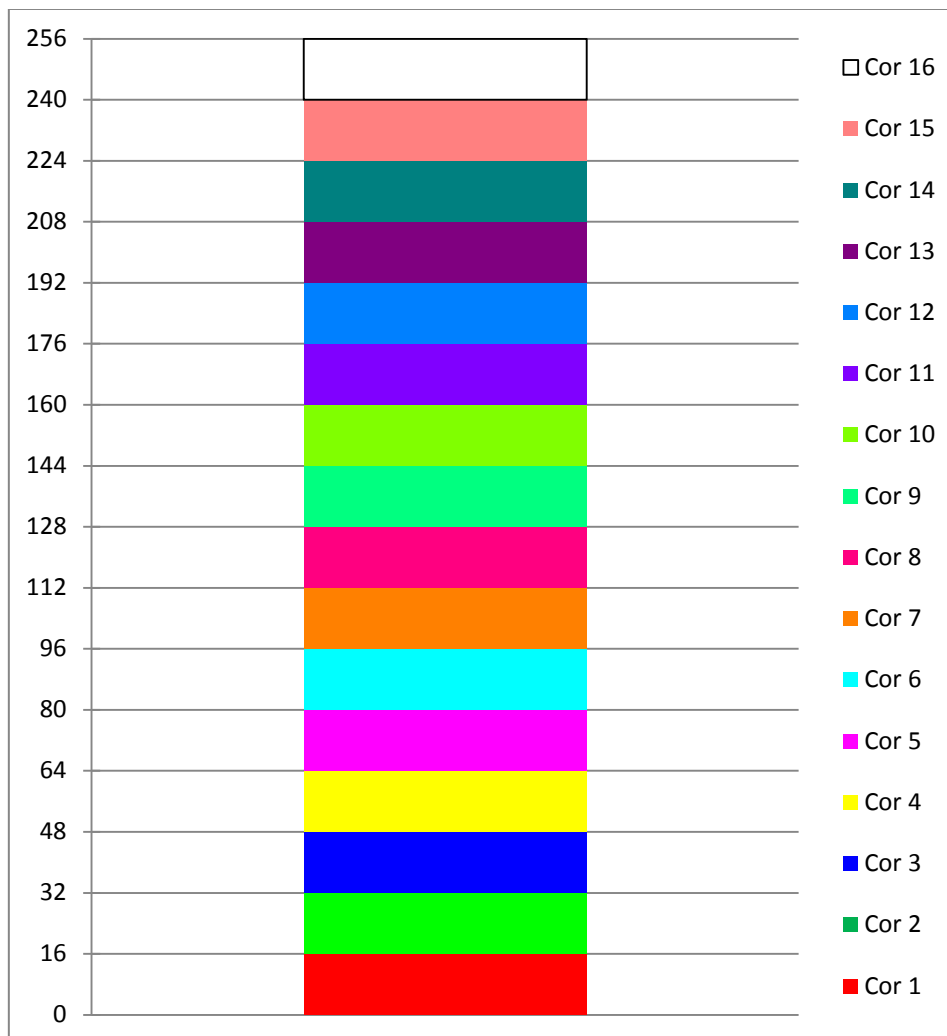


Figura 14 - Tabela conversão de cores do protocolo específico.

As cores são obtidas pelas composições em diferentes intensidades dos leds vermelhos, verdes e azuis, através de misturas exatas ou com porcentagens proporcionais. Cada segmento do módulo requer três PWM, uma para cada componente do LED RGB.

A tabela 5 mostra a formação das cores aplicadas ao projeto no padrão RGB, ou seja, o valor da intensidade de cada componente para a obtenção da cor desejada.

Tabela 5 - Intensidade RGB para formação das cores compostas pela pista de dança.

	R	G	B
Cor 1	255	0	0
Cor 2	0	255	0
Cor 3	0	0	255
Cor 4	255	255	0
Cor 5	255	0	255
Cor 6	0	255	255
Cor 7	255	128	0
Cor 8	255	0	128
Cor 9	0	255	128
Cor 10	128	255	0
Cor 11	128	0	255
Cor 12	0	128	255
Cor 13	128	0	128
Cor 14	0	128	128
Cor 15	255	128	128
Cor 16	255	255	255

O procedimento para a intensidade do brilho foi o mesmo, a informação é obtida através dos dados do segundo canal DMX512A. Entretanto, a quantificação da intensidade utilizou 7 bits, permitindo 128 níveis de intensidade diferente. Como não seria possível dividir igualmente os espaços do byte optou-se por perder o brilho máximo de 100% e manter um nível com 0% de intensidade, ou seja, desligado, então o máximo brilho da pista será de 99%. A figura 15 mostra a tabela de conversão de intensidade de brilho na para a cor vermelha.

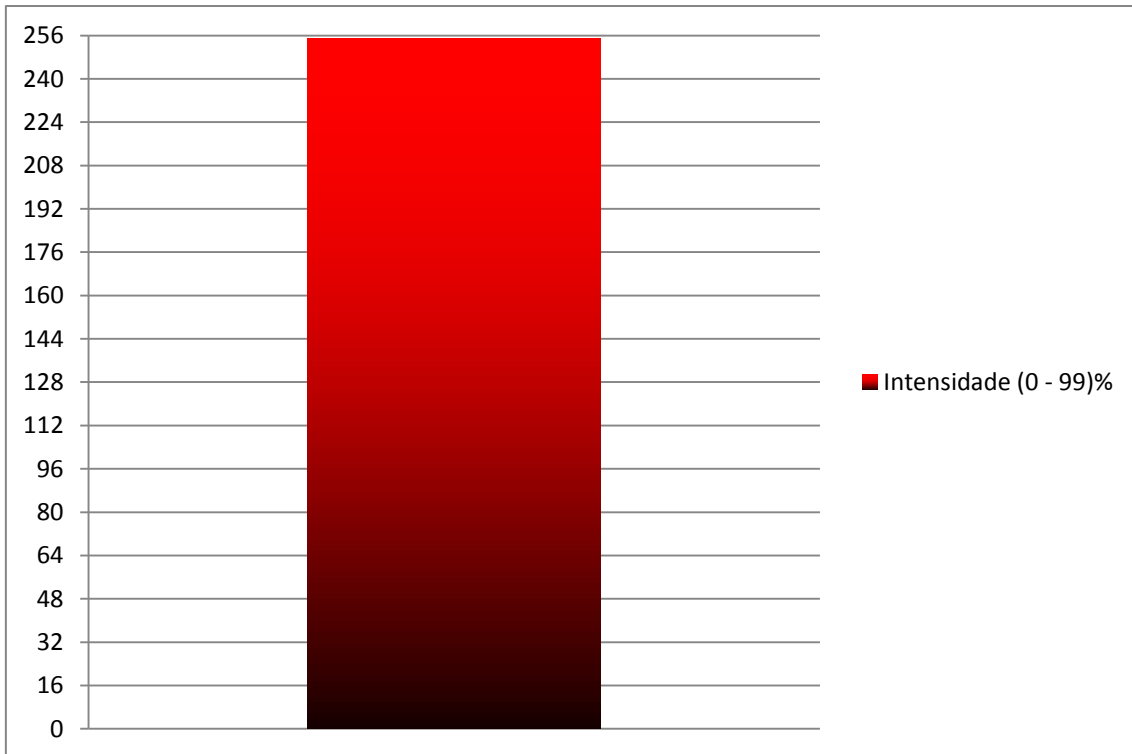


Figura 15 - Tabela de conversão da intensidade de brilho para o vermelho.

5.3 - Diagrama de Blocos

Esse tópico apresenta um resumo aplicado sobre os blocos do DMX24PWM, bem como a suas interações com o usuário e interfaces de saída baseado em suas especificações. A figura 16 introduz o diagrama de blocos do dispositivo.

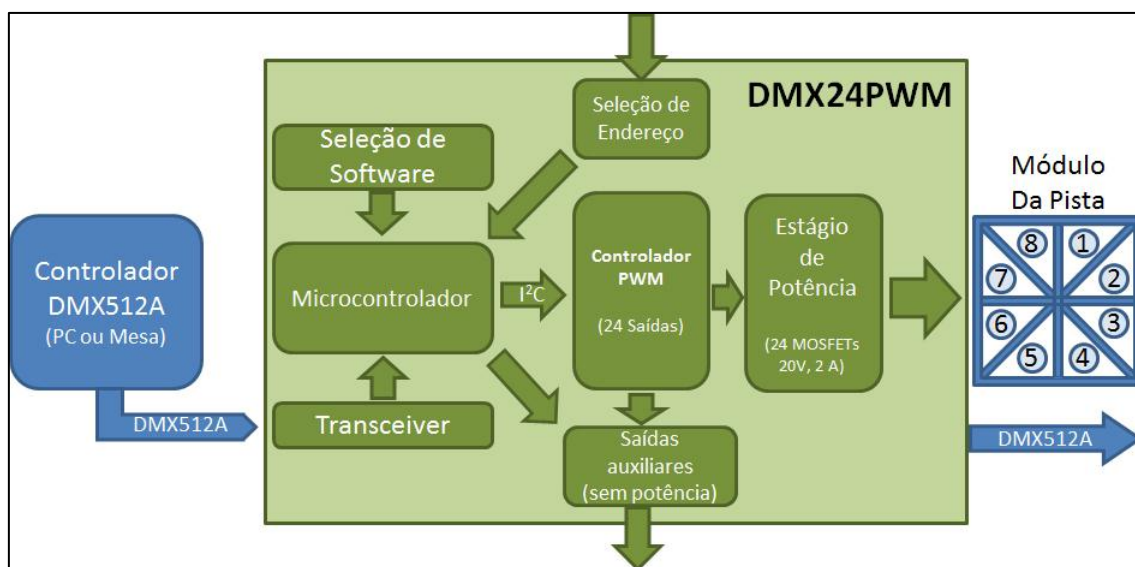


Figura 16 - Diagrama de blocos do DMX24PWM.

Transceiver: Responsável pela digitalização do sinal analógico da rede DMX512A que utiliza em sua camada física o padrão EIA/TIA-485. É composto de um *transceiver* diferencial.

Microntrolador: É o cérebro do DMX24PWM, mantém a constante leitura dos pacotes da rede DMX512A e identifica os destinados ao endereço do dispositivo, obtém os dados dos bytes referentes aos canais utilizados. Traduz, através do protocolo específico, as informações necessárias e transmite via protocolo serial I²C para o controlador PWM. Permite gravação *in-circuit*.

Seleção de software: Consiste em um conjunto de três bits de jumpers para seleção de modo de operação. É possível escolher entre até oito programas diferentes. É proveniente da compreensão de que o DMX24PWM deve ser um *hardware* genérico e ágil na adequação de sistemas de iluminação distribuídos controlados por DMX512A.

Seleção de endereços: É uma chave 9 bits responsável por identificar o endereço do DMX24PWM na rede DMX512A, o controlador lê essa informação e a utiliza para identificar o byte de dados que corresponde ao endereço do dispositivo.

Controlador PWM: O controlador de PWM é um banco de registradores com saídas PWM que recebe, via I²C, do microcontrolador um frame característico desse protocolo de comunicação contendo um endereço, um registrador, um dado, e sinal de controle. Essas informações são enviadas, pelo microcontrolador, após terem sido traduzidas do DMX512A para I²C pelo protocolo específico. O chip identifica o dado e gera 24 sinais PWM para o estágio de potência.

Saídas Auxiliares: É uma alternativa do sistema, que disponibiliza as 24 saídas do controlador PWM em um conector para utilização em sistemas conjugados, que podem ser outros *drivers* de saída anexos e adequados a outras soluções em iluminação. Um módulo que utilize *triacs* pode ser um exemplo de aplicação, mesmo em uma placa conjugada pode ser controlado pelo DMX24PWM. Nesse conector também estão disponíveis as portas de entrada ou saída do microcontrolador que não foram utilizadas para implementações futuras.

Estágio de Potência: São 24 MOSFET's de potência acionados pelo controlador de PWM, com capacidade de gerar a saída, 20V e 2A por MOSFET.

5.4 - Transceiver - SN75176a

O chip SN75176a, mostrado pela figura 16, é responsável por receber os sinais DMX512A, analógicos, e pela análise diferencial digitalizar o sinal e enviar para o microcontrolador o conteúdo digital regenerado.

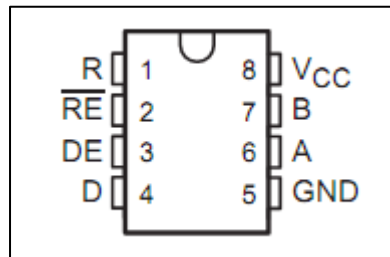


Figura 17 - SN75176a, responsável por digitalizar o sinal analógico do protocolo DMX512A.

O *transceiver* SN75176a é adequado ao padrão da camada física do protocolo. A sensibilidade diferencial entre as entradas A e B é de $\pm 200\text{mV}$, como exigida pelo padrão RS-485. As tensões máximas e mínimas nos terminais também são adequadas, ou seja, +12V e -7V. O pino R (1) envia para uma entrada do microcontrolador um sinal digital em alto com amplitude mínima de 2,7V e em nível baixo com amplitude máxima de 0,45V.

5.5 - Microcontrolador – PIC 18F24K22

O microcontrolador escolhido para gerenciar o DMX24PWM foi o PIC 18F24K22 [9], mostrado pela figura 18. É um microcontrolador da Microchip, com 16 KB de memória de programa.

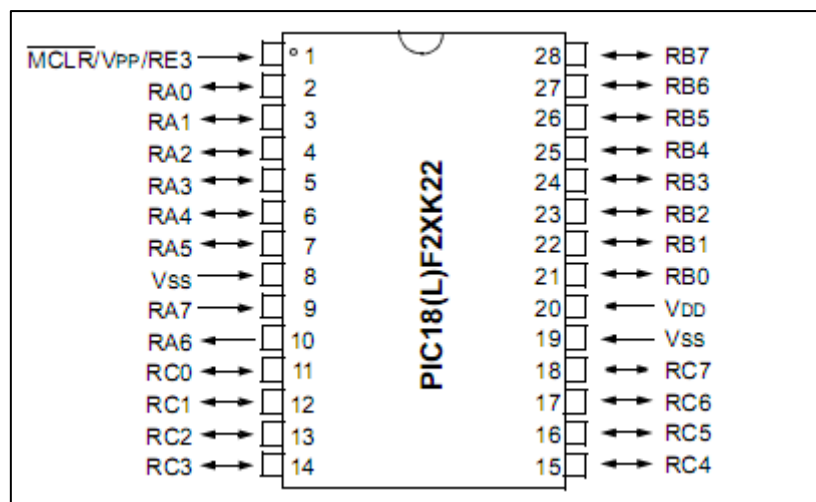


Figura 18 - PIC 18F24K22, microcontrolador responsável por interpretar o protocolo DMX512 e comandar as etapas de saída [9].

No início do desenvolvimento foi utilizado o PIC 18F2480, posteriormente foi substituído pelo 18F24K22. Os principais fatores que levaram à mudança do PIC foram:

- Maior velocidade de processamento (16 MIPS contra 10MIPS do seu antecessor)
- O PIC 18F2480 possuía alguns elementos desnecessários ao projeto, como, por exemplo, rede CAN.
- Microcontrolador mais barato que seu antecessor, tendo em vista a vertente comercial desse projeto.

O desenvolvimento do DMX24PWM na plataforma de microcontroladores PIC da “*Microchip*” também foi influenciada pela experiência adquirida com esse sistema em outros projetos. Existem outros microcontroladores com os recursos utilizados nesse projeto que poderiam ter sido utilizados.

A tabela contendo a relação entre as portas do microcontrolador encontra-se do apêndice A

5.6 - Controlador PWM – PCA9626B

O PCA9626 [10], mostrado na figura 19, é um chip baseado em registradores com 8bits de resolução e dedicado para geração de sinais PWM. Possui 24 saídas que podem ser ajustadas em ligado, desligado e ajustável com *duty cycle* de 0% a 99,6%.

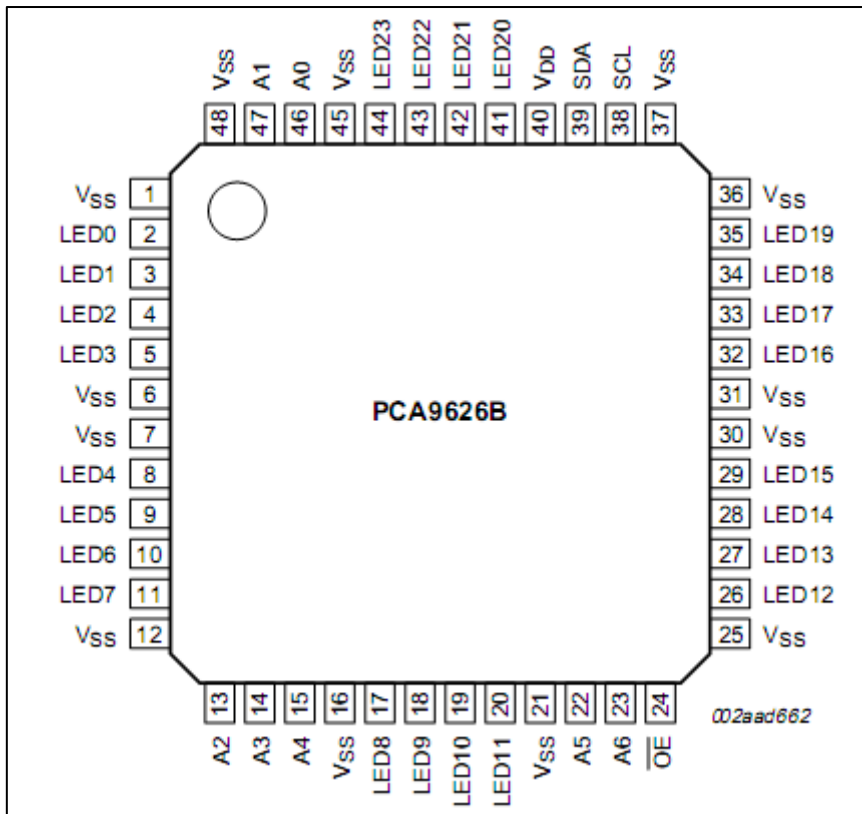


Figura 19 - PCA 9626B [10]

O PCA tem capacidade de fornecer até 100mA de corrente em cada saída, totalizando 2400mA.

A comunicação entre o PCA9626 e o microcontrolador é feita através do protocolo serial I²C. O PCA9626 funciona como escravo na comunicação, embora possua 7 bits(A7-A0), o que permitiria até 128 dispositivos na rede, é o único dispositivo escravo no DMX24PWM e por isso é configurado no endereço 00h. A escolha do protocolo I²C não foi uma característica opcional do projeto, é a única comunicação aceita pelo PCA9626. O frame da comunicação I²C é mostrado pela figura 20.

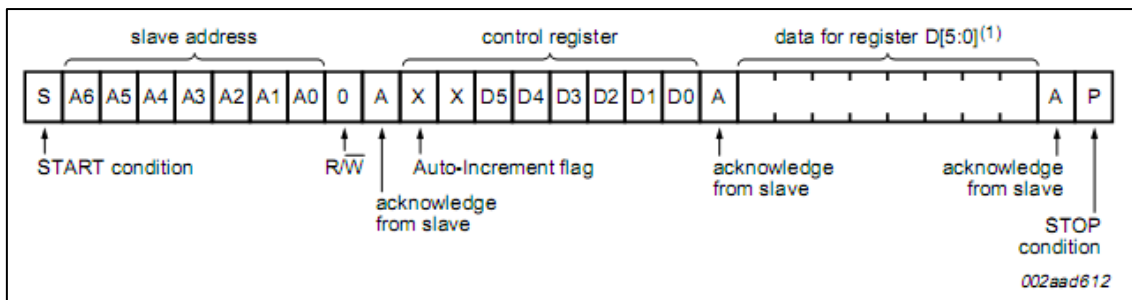


Figura 20 - Frame da comunicação I²C [10]

O PCA8626 recebe do mestre (microcontrolador) a informação de qual registrador será escrito por um dado de 8bits enviado pela rede I²C. O dado subsequente contém a informação a ser aplicada a esse registrador, dessa maneira o PCA8626 atua no sistema compondo as cores e as intensidades dos LEDs RGB.

O chip ainda conta com configurações pré-definidas de auto incremento no registrador diminuindo a quantidade de informações trafegadas na rede.

A tabela dos registradores encontra-se no apêndice B.

5.7 - Prova de conceito

Após a escolha dos componentes principais buscou-se a implementação do modelo desenvolvido, descrito pelo diagrama da figura 16, com objetivo de fazer uma prova de conceito do projeto e mostrar que o dispositivo desenvolvido nesse trabalho tem condições de comunicar-se com uma rede DMX512A e também, através do seu protocolo específico, é capaz de acionar um sistema de iluminação distribuído mediante controle de mesa e *softwares* comerciais. Utilizou-se, também, a prova de conceito para obtenção das formas de onda do protocolo DMX512A para verificação do conteúdo apresentado na fundamentação teórica.

A primeira montagem envolveu a implementação do circuito *transceiver*, utilizando o componente, citado nesse capítulo, SN75176.

Utilizou-se para ser o mestre da rede DMX512A uma mesa comercial DMX512A do fabricante EXELL e modelo DMX OPERATOR III, a figura 21 mostra a mesa utilizada no trabalho.



Figura 21 - Mesa DMX OPERATOR III

O modelo DMX OPERATOR III tem capacidade de controlar até 192 canais DMX512A além de armazenar configuração dos canais em 30 bancos de 8 cenas cada. A escolha dessa mesa para primeiro controle do sistema ocorreu pela disponibilidade de empréstimo para o desenvolvimento do trabalho. Apesar de sua capacidade máxima seja inferior à mínima requerida para controle da pista completa de 32m², a mesa foi suficiente para a proposta de testar um protótipo da pista de dança em escala reduzida que será apresentado posteriormente.

O sinal de saída do transceiver foi obtido através de um analisador lógico Saleae. Esse recurso foi utilizado na obtenção dos sinais digitais desse trabalho por possuir 8 canais, o *software* de gerenciamento apresentar tradução das sequências de bits e facilidade na definição de *trigger*. A figura 22 mostra uma imagem do analisador lógico Saleae Logic.



Figura 22 - Analisador Lógico Saleae logic.

Os resultados obtidos serão apresentados no capítulo 5.

Na sequência do desenvolvimento o microcontrolador recebeu o *frame* digital oriundo do transceiver de recebimento da rede DMX512A. Foi desenvolvido um *software* capaz de identificar o início do protocolo e seus subsequentes dados, também foi necessário um contador de bytes para que o sistema fosse capaz de reconhecer o pacote destinado ao seu endereçamento na rede DMX512A. O sistema foi proposto para estar adequado a diferenças de temporização entre versão atual e antiga do protocolo.

O método desenvolvido para a sincronização e decodificação do protocolo DMX512 para implementação no DMX24PWM é mostrado pelo fluxograma mostrado na figura 23

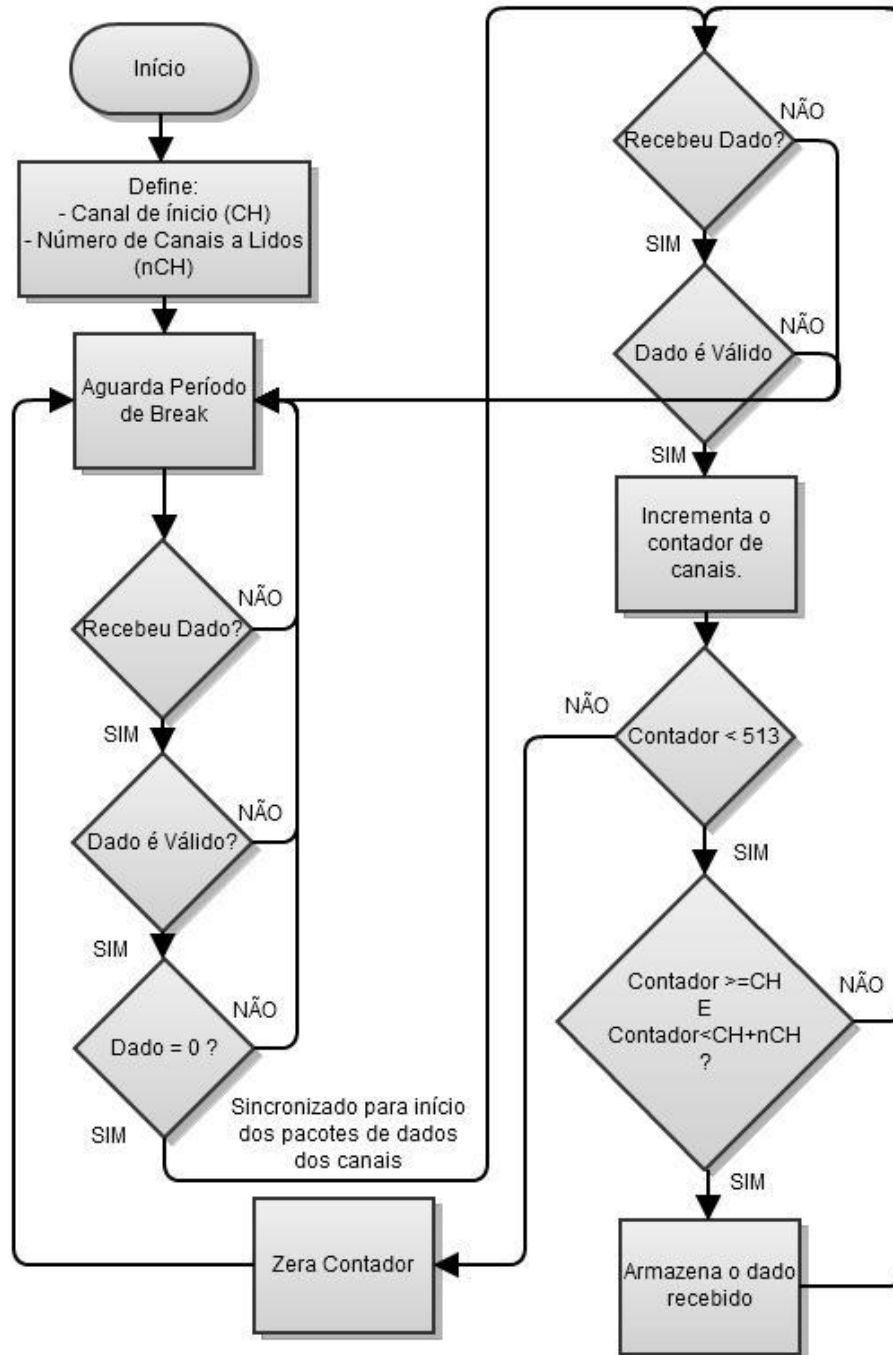


Figura 23 - Fluxograma simplificado do processo de recepção dos dados da rede.

A verificação de reconhecimento do protocolo DMX512 pelo microcontrolador foi aferida utilizando o dado do byte do canal 1 como valor que controlava o *duty cycle* da saída PWM do próprio microcontrolador. Dessa forma controlou-se a intensidade da corrente de um segmento de fita de LEDs RGB. A fita possuía densidade de 60 LEDs RGB por metro. A figura 24 mostra o segmento, contendo 3 LEDs RGB, utilizado no desenvolvimento.



Figura 24 - Segmento de fita de LEDs RGB.

Foi possível impor o funcionamento do DMX24PWM em toda a extensão do protocolo, ou seja, responder em qualquer um dos 512 canais, dependendo apenas do endereçamento do dispositivo escravo projetado.

Buscando atender as especificações sistêmicas do DMX24PWM foi desenvolvido um código para conversão do protocolo específico em resultado aplicado à pista de dança. Primeiramente fez-se a comunicação entre o PIC18F24K22 e o PCA9626 através do protocolo I²C. O microcontrolador armazenava os dados dos 16 canais requeridos pelo dispositivo, como mostrado na tabela 4. As informações contidas nesses 16 canais controlam as cores e intensidades dos 8 segmentos da 1 módulo da pista de dança. O microcontrolador interpreta os dados dos canais de cor e

compara com uma tabela, contida no programa, que referencia a intensidade de cada componente vermelho, verde e azul necessário para formação da cor definida no projeto. Através da comunicação I²C implementada entre o microcontrolador e o PCA9626, são enviados valores para os registradores que controlam a intensidade de cada saída PWM do PCA9626. Dessa forma, cada segmento da pista de dança é capaz de reproduzir as 16 cores propostas no projeto, e pode ser controlado independentemente um do outro. Seguindo a correspondência da tabela 4, os canais responsáveis pela intensidade do segmento são lidos pelo programa de conversão e transformados em um fator de escala. Dessa maneira, é mantida a mesma proporção entre os PWM referentes ao vermelho, verde e azul de cada segmento, entretanto, multiplicado por um fator de escala, proporcional e menor ao valor típico gerando uma representação da cor escolhida em intensidade luminosa menor.

A figura 25 mostra o fluxograma da conversão do protocolo específico.

Foi realizada também a implementação da comunicação com auto incremento nos registradores do PCA9626, reduzindo o fluxo de informação pela linha I²C.

Os códigos do programa contido no microcontrolador são mostrados no apêndice C.

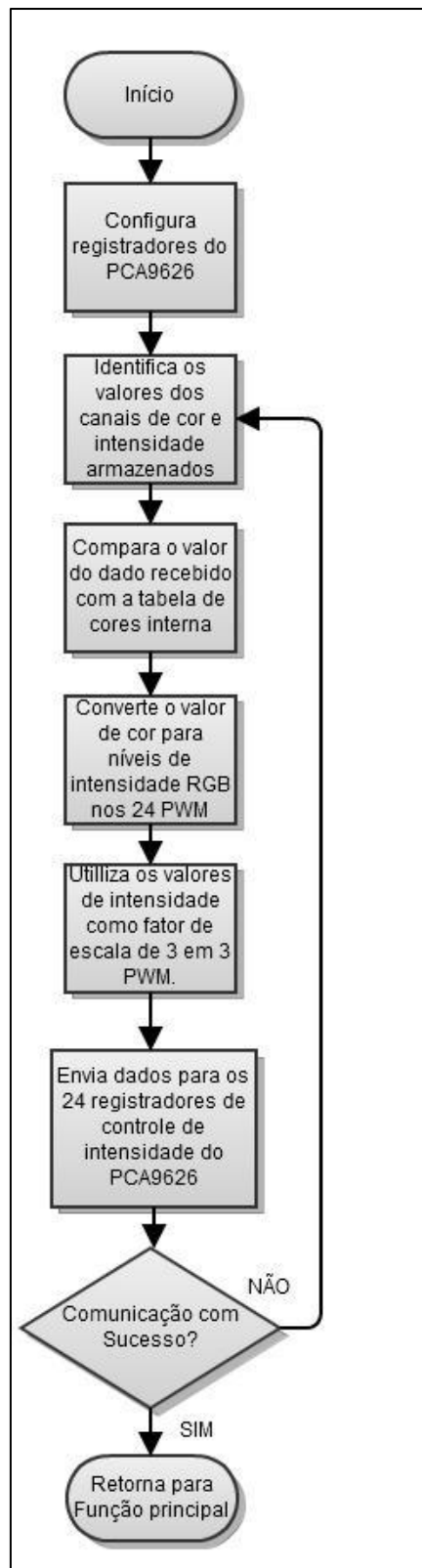


Figura 25 - Fluxograma simplificado da conversão do protocolo específico.

A figura 26 e 27 mostram imagens da prova de conceito utilizada no projeto.

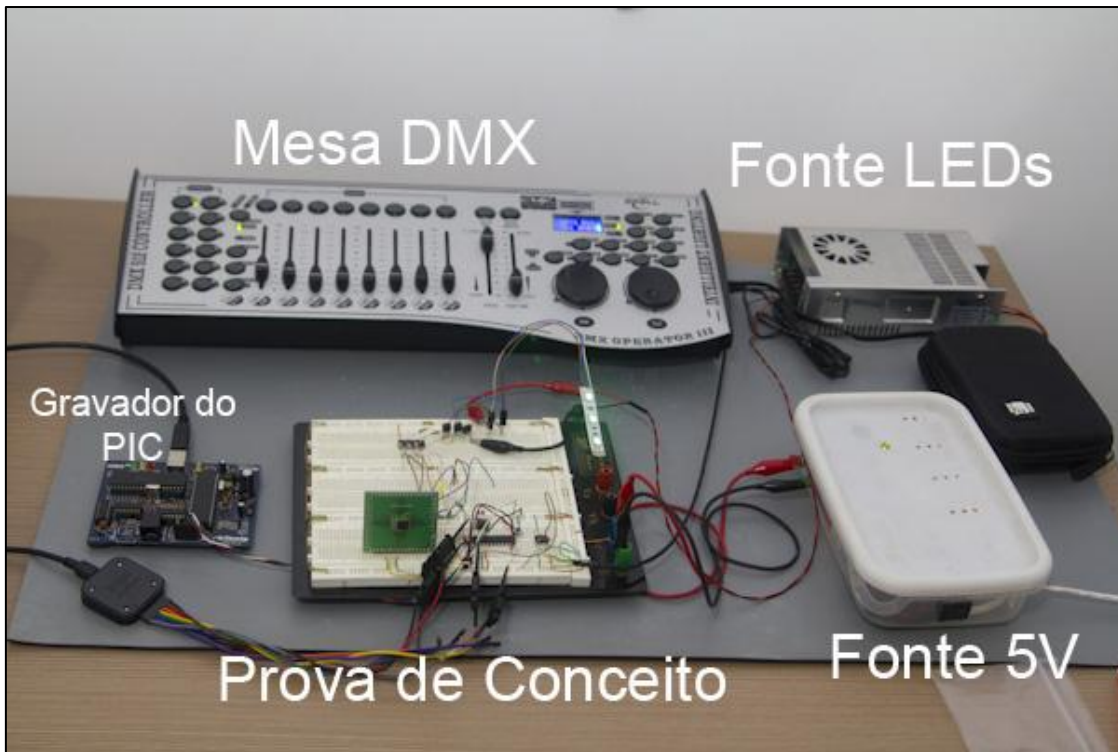


Figura 26 – Visão geral da utilização da prova de conceito.

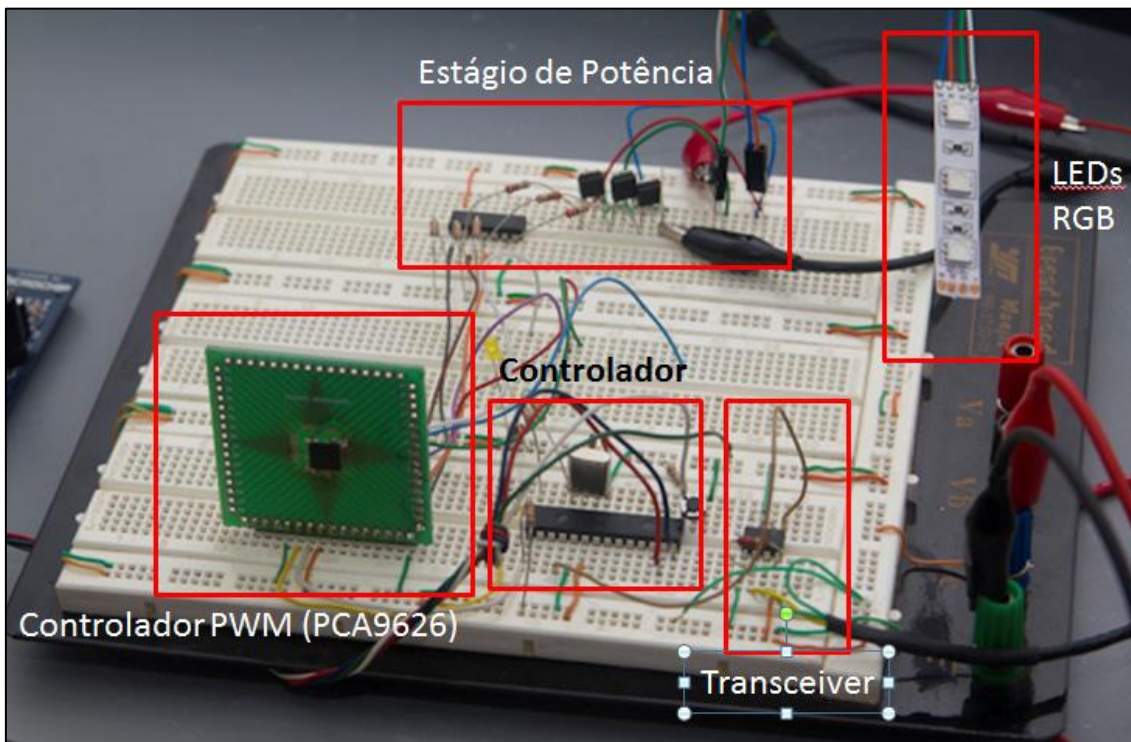


Figura 27 – Composição dos blocos do projeto.

5.8 - Desenvolvimento do protótipo DMX24PWM

Tendo em vista a vertente prática desse trabalho e a possibilidade do projeto desenvolvido tornar-se um produto comercial pela sua compatibilidade com equipamentos DMX512A do mercado, foi considerado válido o desenvolvimento de um protótipo em circuito impresso desse trabalho. A PCB, baseada na prova de conceito, foi desenvolvida no software de design eletrônico ALTIUM. As imagens do projeto são mostradas na figura 28.

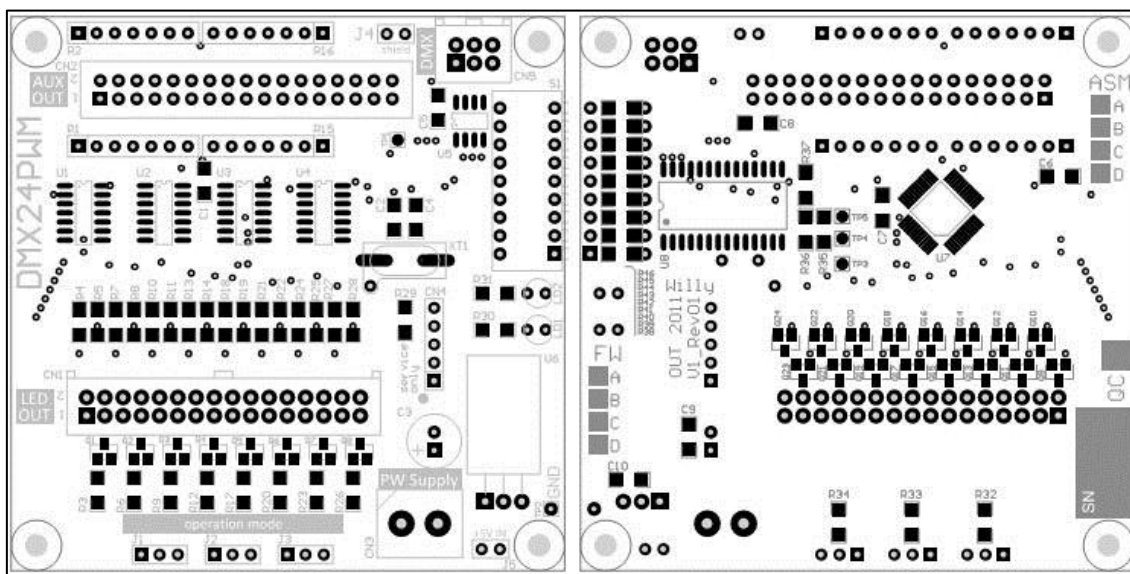


Figura 28 – Print preliminar da placa de circuito impresso.

O esquema elétrico do DMX24PWM, utilizado para construção da placa de circuito impresso, é mostrado no apêndice E.

Para testar a eficiência e aplicabilidade do projeto desenvolveu-se um protótipo em escala reduzida de uma pista de dança. O protótipo possui 8 segmentos independentes como a proposta original, porém, possui apenas 17 centímetros de lado e a área total é de 0,0289m².

O projeto do protótipo reduzido foi feito através do *software* de design 3D CAD *Solidworks*, foram criadas dois tipos de peças que associadas fossem capazes de reproduzir o formato proposto nas especificações. A figura 29 mostra imagens do desenvolvimento em *software*.

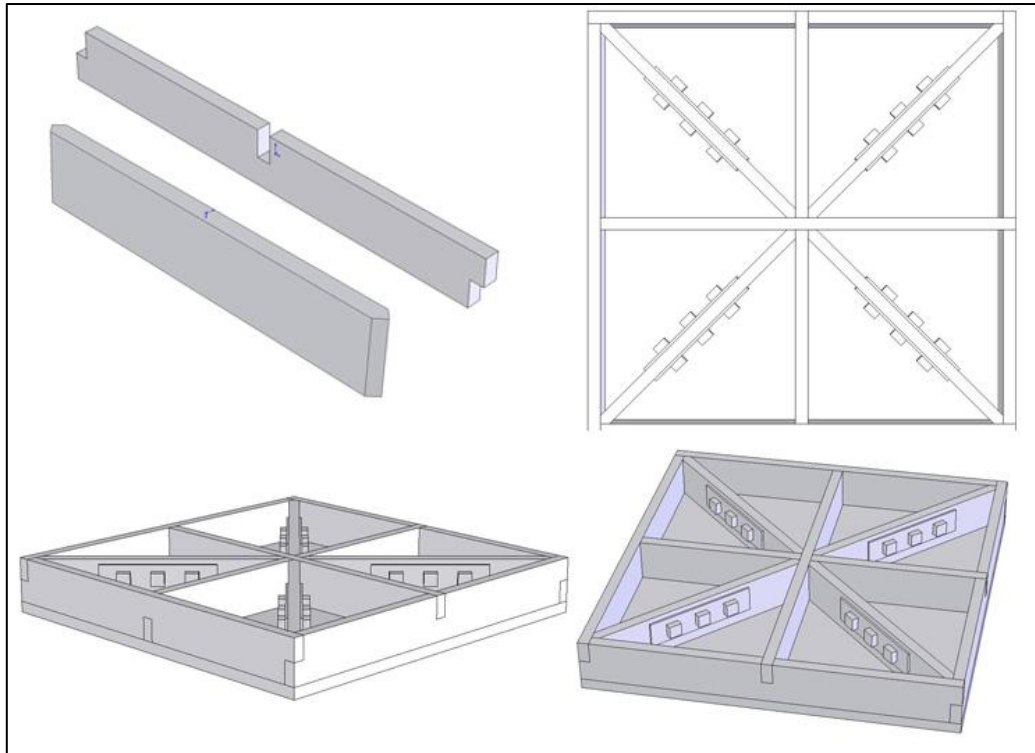


Figura 29 – Projeto mecânico do protótipo da pista de dança

Utilizou-se papel pluma para a construção física do protótipo. A Resistência, bom acabamento externo, leveza e facilidade de manipulação foram determinantes para escolha desse material. O papel pluma é comumente utilizado na construção de maquetes arquitetônicas. Para compor a iluminação do protótipo utilizou-se 8 segmentos de fita de LEDs RGB, idênticos ao mostrado na figura 24. A figura 30 mostra fotos da construção do protótipo.

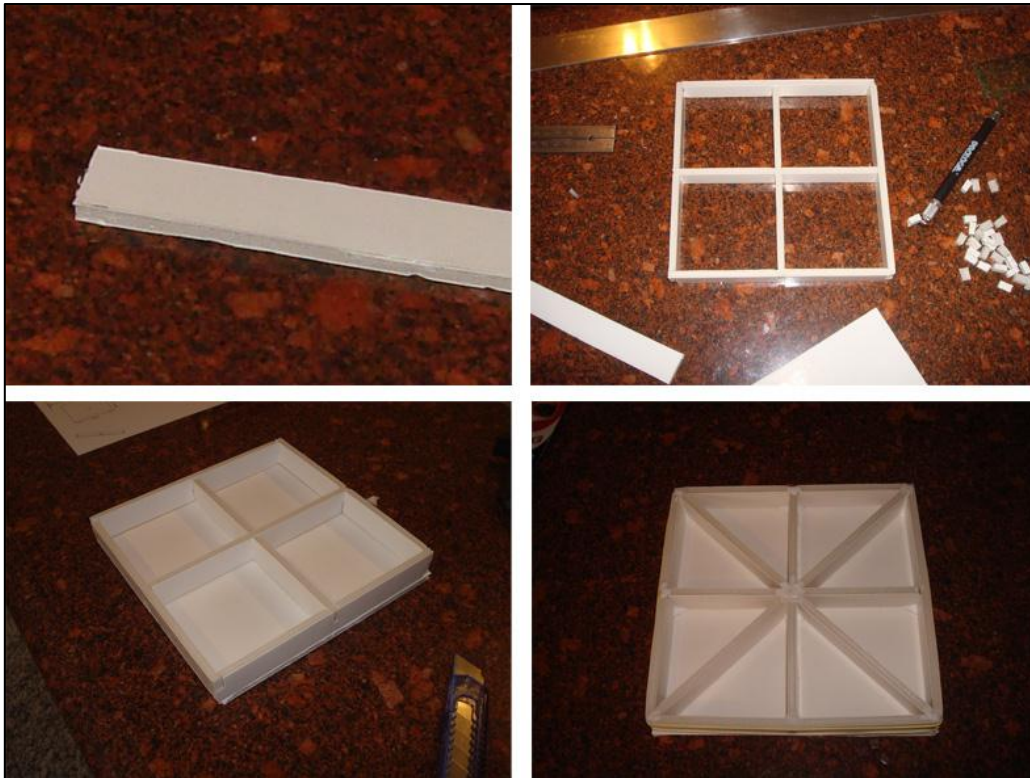


Figura 30 – Construção física do protótipo reduzido de um módulo da pista de dança.

5.9 - Desenvolvimento da interface com computador

Visando alcançar todos os requisitos propostos nesse trabalho, o controle via software foi realizada utilizando o Freestyler DMX 3.4.4, um *software* muito difundido no controle da rede DMX512. O controle via software uma rede DMX possibilita mais dinamismo ao usuário, facilidade em gravar cenas previamente estabelecidas e as representações gráficas das ações da pista de dança como segmento selecionado, cor e intensidade geram mais velocidade e eficiência na interface homem-máquina. A transformação da aplicação destinada à pista de dança para um recurso utilizável no *Freestyler DMX* foi realizada utilizando o *Fixture Creator 6.1*, programa destinado a criação de interfaces homem-máquina para dispositivos DMX. A figura 31 mostra esse desenvolvimento.

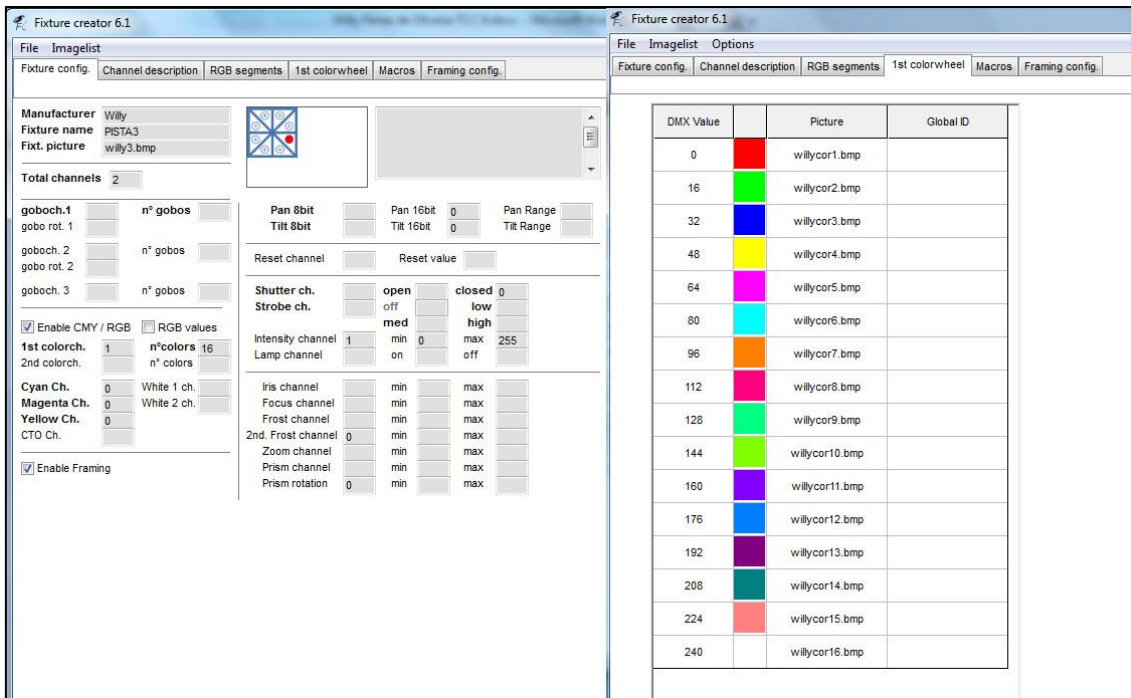


Figura 31 - Desenvolvimento do modelo do protótipo para utilização em *software*.

Para comunicar-se com o DMX24PWM foi utilizado um conversor USB-DMX512A mostrado na figura 32. Os resultados obtidos e integração entre o *Freestyler 512* e a aplicação serão mostrados posteriormente.



Figura 32 - Conversor USB para DMX512.

5. Resultados

Esse capítulo mostra os resultados obtidos durante todo o processo de desenvolvimento do DMX24PWM e protótipo de teste.

5.1 - DMX512A

A figura 33 mostra um frame obtido através da mesa DMX OPERATOR III e identificado pelo *software* contido no DMX24PWM. O frame mostra as características descritas na exposição do protocolo e foi devidamente identificado pelo microcontrolador.

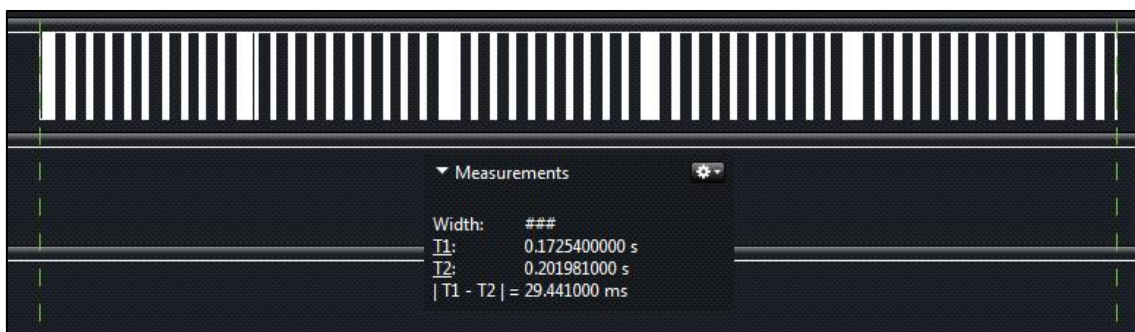


Figura 33 - *Frame* de 192 Canais da Mesa DMX OPERATOR III.

A figura 34 mostra os 7 primeiros dos 192 canais gerados pela mesa.

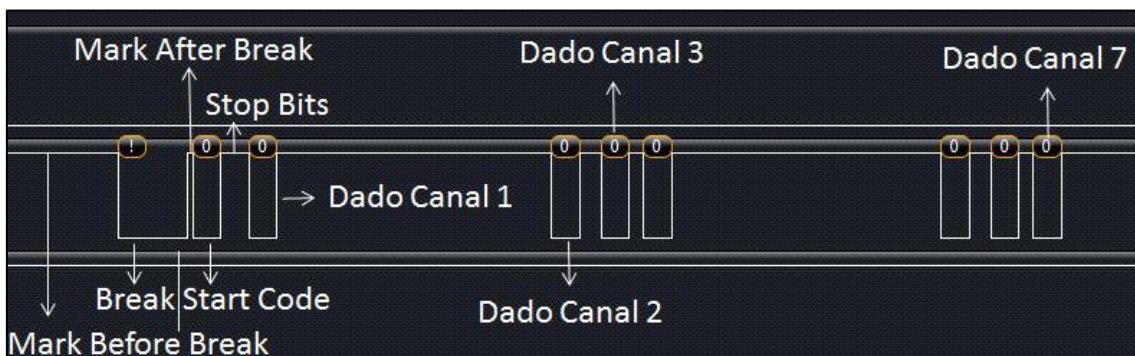


Figura 34 - Sete primeiros canais da mesa DMX OPERATOR III

A mesa de controle utilizada nos testes apresentou temporizações relativas ao protocolo DMX512A anterior a atualização mais recente. A tabela 6 mostra um comparativo entre os resultados obtidos através da mesa de controle e o protocolo atual.

Tabela 6 - Comparação de temporização do protocolo DMX512

Descrição	Tempo Verificado (μ s)	ANSI E1.11 2008 (μ s)
Break (mínimo)	88,5	92
Mark After Break (mínimo)	9,5	12
Bit	4	4
Stop bits (mínimo)	32	8
Pacote DMX512 (máximo)	80000 (estimado)	1000000

O frame verificado apresentou diferenças nos tempos mínimos de *Break* e *Mark After Break*. Essa diferença foi considerada benéfica pois trouxe ao projeto a necessidade de adequação às diferenças existentes no padrão, tornando-o apto a interagir com dispositivos de controle mais antigos. Como a mesa utilizada gera um *frame* de apenas 192 canais estimou-se que o pacote DMX512 tenha 80ms visto que a largura do *frame* com 192 canais foi de quase 30ms.

5.2 – PCA9626

A implementação da comunicação serial I²C velocidade maior que a tava de atualização do protocolo DMX512A e por isso não comprometeu o desempenho da aplicação. A utilização do auto-incremento promoveu a diminuição na quantidade de dados necessários para controlar as 24 saídas PWM do PCA9626 e, conseqüentemente, trouxe ganho significativo na velocidade de comunicação na linha I²C. A figura 35 mostra a comparação dos pacotes I²C com e sem a configuração de auto-incrementos dos registradores.



Figura 35 - Linha I2C de comunicação entre o microcontrolador e o PCA 9626.

A velocidade da comunicação sem auto-incremento transmitiu informação para o controle dos 24 PWM em 7,01ms. Já a utilização do auto-incremento reduziu a

quantidade de dados transmitidos e o tempo para 2,23ms. Ambas condições são mais rápidas que os 30ms para atualização do *frame* DMX.

5.3 - Prova Conceito

O resultado obtido com a integração de todos os componentes do projeto na prova de conceito é mostrado pelas figuras 36 e 37. A figura 36 mostra a tradução feita pelo protocolo específico do comando da mesa de controle e a luz emitida pelos LEDs. A figura 37 mostra a tradução da intensidade de luz dos LEDs irradiando o vermelho.



Figura 36 - Cores obtidas utilizando a prova de conceito

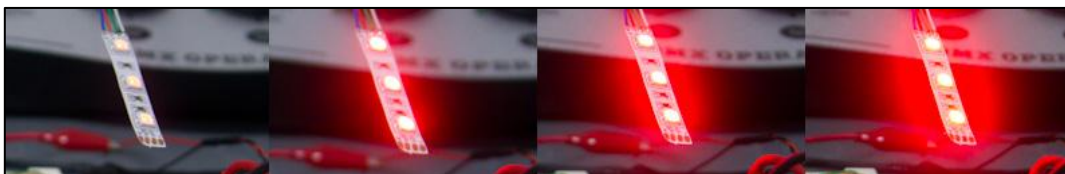


Figura 37 - 4 amostras de intensidade obtidas utilizando a prova de conceito.

5.4 - Protótipo DMX24PWM

A placa de circuito impresso do DMX24PWM é mostrada pela figura 38. À esquerda pode-se observar a parte superior (*top*) da placa, à direita a parte inferior (*bottom*). A figura 39 mostra o resultado final do protótipo após soldagem dos componentes.

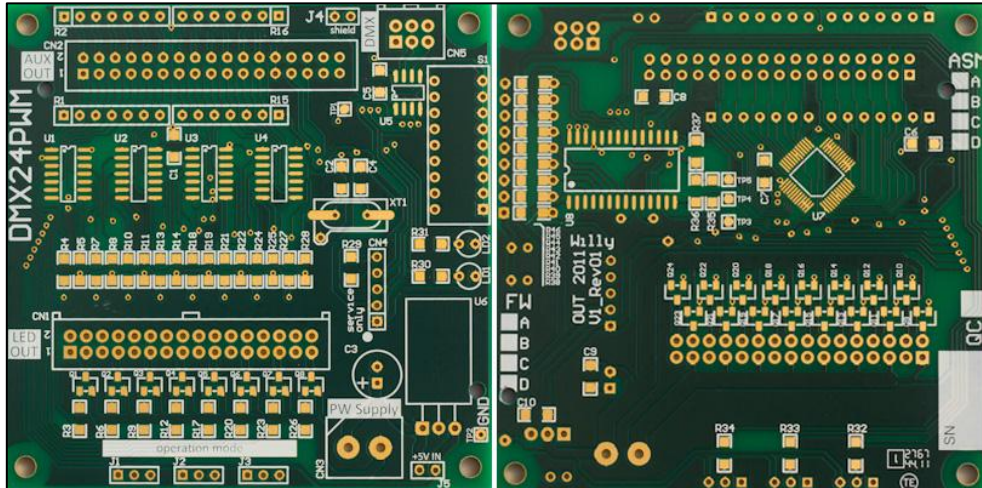


Figura 38 - Placa de circuito impresso do DMX24PWM

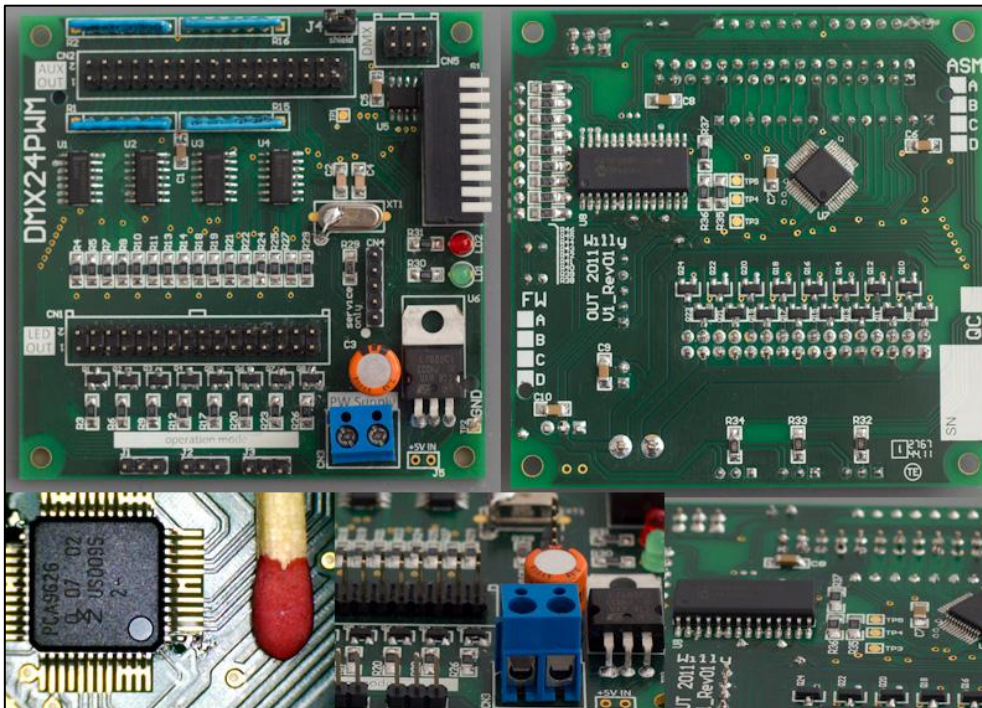


Figura 39 - Resultado final do DMX24PWM.

5.5 - Protótipo reduzido da pista de dança

O resultado final do protótipo reduzido de um módulo da pista de dança é mostrado na figura 40.

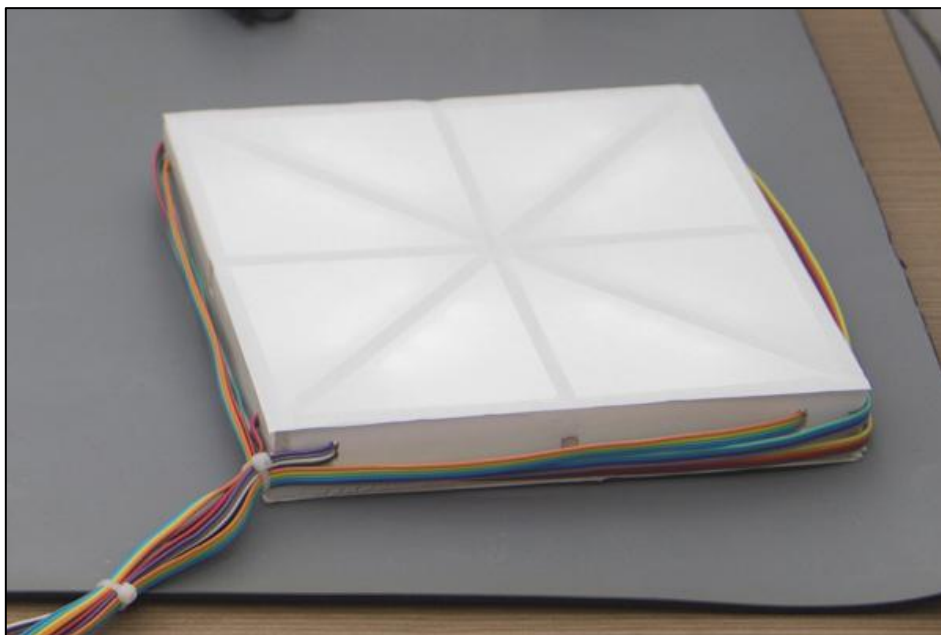


Figura 40 - Módulo da pista de dança em escala reduzida.

5.6 - Software de controle

A adequação do *software* de controle à aplicação de gerência da pista de dança através da interface DMX24PWM foi realizada. O *Freestyler* forneceu à rede DMX512A valores condizentes ao estabelecido pelas figuras 14 e 15 nas especificações de cor e intensidade. A figura 41 mostra a tela do Freestyler 512 com 8 segmentos que formam 1 módulo da pista, as 16 opções de cores, o nível de intensidade e o conteúdo dos canais 1 ao 16 da rede DMX512A. Como o dispositivo está endereçado no canal 1, todos os canais ímpares mostram informações relativas a cor, já os canais pares mostram os valores das intensidades.

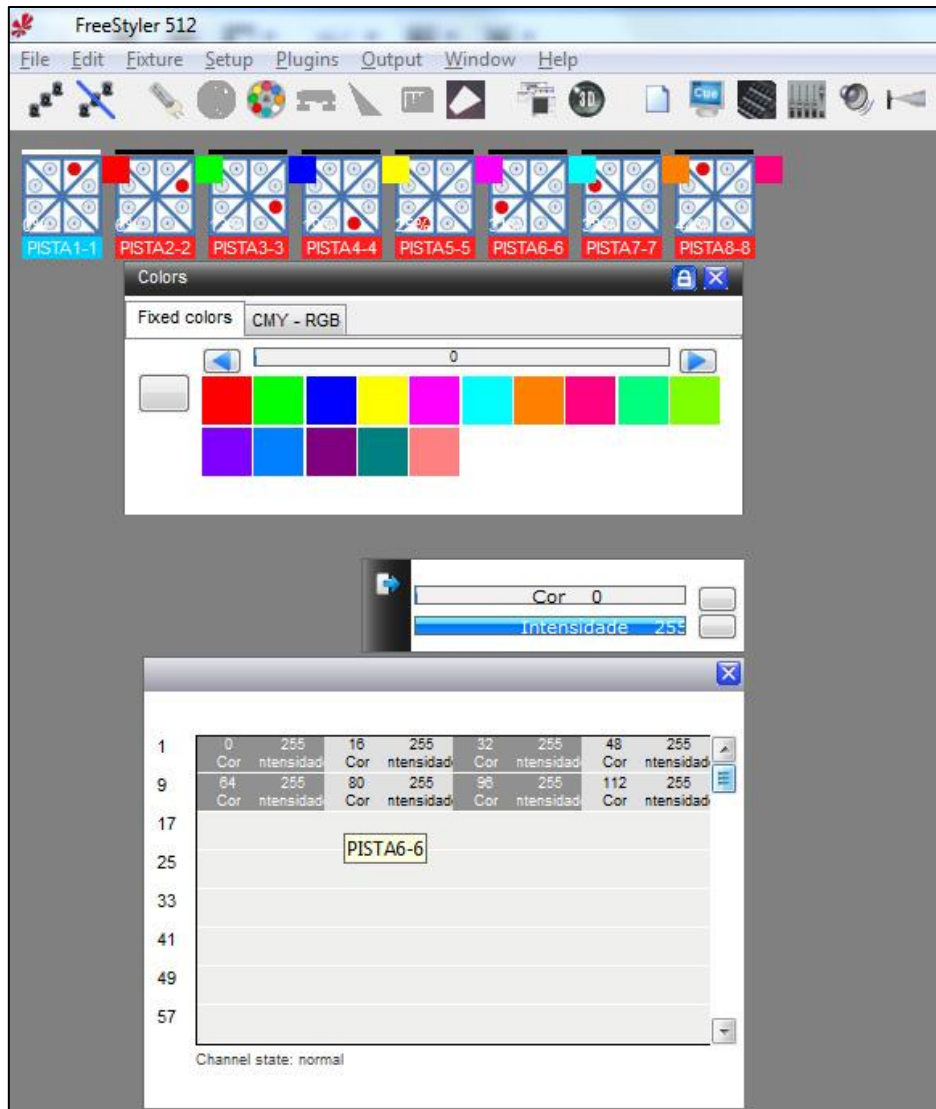


Figura 41 - Tela do *software Freestyler 512* aplicado ao controle da pista de dança.

5.7 - Resultado Final

O resultado da integração de todos os componentes do sistema é mostrado pela figura 42 e 43. A figura 42 mostra as 16 cores geradas pelo sistema e a figura 43 6 níveis de intensidade.

O custo da fabricação da placa de circuito impresso e componentes do DMX24PWM é de aproximadamente R\$120,00.



Figura 42 - Resultado final das cores geradas.



Figura 43 - Amostras de intensidades obtidas.

A figura 44 mostra o funcionamento, em conjunto, de todos os elementos do projeto sendo controlado pela mesa comercial e o *software*.



Figura 44 - Visão Geral de funcionamento do sistema.

5.8 - Discussão dos resultados

Os resultados obtidos na identificação do protocolo DMX512, gerados pela mesa comercial utilizada no projeto, mostraram um *frame* adequado ao padrão antigo da norma, comprovado pela tabela 6. A implementação do código de reconhecimento do protocolo (DMX512) foi suficiente para que fosse possível a comunicação entre o DMX24PWM e controladores DMX512 referenciados pela norma antiga e atual.

A velocidade de comunicação interna (I²C) do DMX24PWM foi mais rápida que a atualização de informações do protocolo DMX512. A utilização do auto-incremento reduziu o tempo de comunicação interna entre o microcontrolador e o PCA9626. A vantagem discutida é a diminuição da quantidade de dados enviados pelo microcontrolador, reduzindo o esforço computacional do dispositivo. É relevante considerar que a utilização do auto-incremento não é fundamental para o funcionamento do projeto visto que o tempo de comunicação interna é sempre menor que a atualização externa do protocolo DMX512.

As cores utilizadas no projeto, obtidas através da composição do padrão RGB, foram comparadas por observação visual devido à falta de equipamento específico para reconhecimento de cores. Entretanto a inspeção visual entre a fita de LEDs da prova de conceito ou o protótipo reduzido da pista de dança e a tabela de cores proposta no trabalho mostrou-se condizente e satisfatória. Mesmo assim, analisando comparativamente, a representação da pista de dança em escala reduzida mostrou resultado mais detalhado na definição de cores por concentrar a luz em um segmento específico.

Estima-se que o custo da construção de um módulo de 1m² de pista em tamanho real utilizando o DMX24PWM não ultrapasse os R\$1200,00. Deste modo, esse projeto mantém a concepção de tornar-se um produto comercial, entretanto necessita de um estudo a viabilidade econômica mais aprofundado, levando em conta fatores não mencionados nessa monografia.

6. Conclusão

Os objetivos propostos e descritos no capítulo 2 desta monografia foram alcançados, os resultados utilizando a prova de conceito comprovaram que o desenvolvimento do *software* e escolha dos componentes foram suficientes para controle dos níveis de cores e intensidade dos conjuntos de LED. Esses resultados forneceram respaldo para o desenvolvimento do protótipo DMX24PWM que comunicou satisfatoriamente com as plataformas de controle DMX512 mais comuns do mercado. O protótipo ainda mostrou-se eficaz no funcionamento com controladores dirigidos por normas não atualizadas. O DMX24PWM foi construído de forma robusta e funcional para a aplicação de acionamento de um módulo de pista de dança iluminada por LEDs. O sistema apresentou velocidade de processamento e comunicação interna compatível com as normas do protocolo DMX512A, e o *hardware* pode ser considerado genérico suficiente para ser aplicado em outras soluções de iluminação distribuída.

Os testes utilizando o protótipo final apresentaram resultados satisfatórios nos seguintes aspectos:

- Representação das cores e níveis de intensidade estabelecidos no projeto;
- Comunicação DMX512 compatível com a maioria dos controladores e equipamentos desse segmento;
- Controle simplificado, utilizando 16 canais DMX512 e adequação do módulo da pista de dança desenvolvido ao software Freestyler 512;

O protótipo, em escala reduzida, do módulo da pista de dança somado ao desempenho do DMX24PWM trouxeram a expectativa de desenvolvimento de um produto comercial capaz de concorrer no mercado nacional com qualidade e custo competitivo no segmento de pistas de dança iluminadas por LEDs.

Academicamente o projeto utilizou conhecimento adquirido durante toda a graduação e envolveu a rotina completa de um projeto de engenharia desde a sua concepção, embasamento, implementação e desenvolvimento de protótipo sofisticado.

Para melhoria do projeto propõe-se utilizar o recurso incorporado à placa de seleção externa de *firmware*. Implementando-se uma simplificação no protocolo específico, o controle de cor e intensidade pode ser feito utilizando-se apenas 1 canal, para isso, seria necessário reduzir a resolução do controle de intensidade para 16 níveis em cada cor. Dessa maneira, reduzindo a quantidade de canais utilizados por cada segmento, o tamanho máximo da pista poderia ser de até 64m². O objetivo é que o DMX24PWM tenha 8 opções dos programas que possam ser selecionados pelo usuário. Algumas opções de utilização do projeto deste trabalho podem ser:

- Utilizar 24 canais para controlar, através dos PWM, *dimmers* de lâmpadas;
- Controlar cenas de iluminação decorativa interna e externa como teatros, restaurantes, bares, fachadas, jardins, residências.
- Utilizar as entradas e saídas auxiliares contidas no DMX24PWM, para conectar um módulo auxiliar de comunicação wireless do protocolo DMX512A, previsto pela revisão mais moderna da norma [6].

Referências Bibliográficas.

[1] **CAMARGO, Roberto Gill** - Livros técnicos sobre iluminação cênica. Disponível em < <http://www.iar.unicamp.br/lab/luz/>>. Acesso em: 9 de maio de 2011.

[2] **0-10V Control Systems**. Disponível em <<http://www.davmark.co.uk/group/services/lighting/10v.html>>. Acesso em: 9 de maio de 2011.

[3] **DMX512 Standard** – USITT. Disponível em <<http://www.usitt.org/Resources/Standards2/DMX512>>. Acesso em: 9 de maio de 2011.

[4] **Setor de casas noturnas, shows e eventos ganha feira especializada** – Pequenas Empresas & Grandes Negócios. Disponível em <<http://revistapegn.globo.com/Revista/Common/0,,EMI132950-17180,00-SETOR+DE+CASAS+NOTURNAS+SHOWS+E+EVENTOS+GANHA+FEIRA+ESPECIALIZADA.html>>. Acesso em: 5 de setembro de 2011.

[5] **DMX512 FAQ** – USITT. Disponível em <<http://www.usitt.org/Resources/Standards2/DMX512/DMX512FAQ>>. Acesso em: : 9 de maio de 2011.

[6] **ANSI E1.11 - 2008** - Entertainment Technology - USITT DMX512-A, Asynchronous Serial Digital Data Transmission Standard for Controlling Lighting Equipment and Accessories.

[7] **Leão, Vitor** – 2007 - RS-485 Especificação e Utilização. Disponível em <<http://pt.scribd.com/doc/19000292/rs485-especificacao-e-utilizacao>>. Acesso em 15 de agosto de 2011.

[8] **Using a PIC Microcontroller for DMX512 Communication** - *Microchip Application Note AN1076*. Disponível em <<http://www.microchip.com/>>. Acesso em: 2 de maio de 2011.

[9] **PIC18F44K22 Datasheet**. Disponível em <<http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en547758>>. Acesso em: 7 de junho de 2011.

[10] **PCA 9626 Datasheet**. Disponível em < <http://www.nxp.com>>. Acesso em: 22 de agosto de 2011.

Apêndice.

Apêndice A – Descrição da pinagem do PIC 18F24K22.

PIC18(L)F2X/4XK22

TABLE 1: PIC18(L)F2XK22 PIN SUMMARY

28-SSOP, SOIC 28-SPDIP	28-QFN, UQFN	I/O	Analog	Comparator	CTMU	SR Latch	Reference	(E)CCP	EUSART	MSSP	Timers	Interrupts	Pull-up	Basic
2	27	RA0	AN0	C12IN0-										
3	28	RA1	AN1	C12IN1-										
4	1	RA2	AN2	C2IN+			VREF-/ DACOUT							
5	2	RA3	AN3	C1IN+			VREF+							
6	3	RA4		C1OUT		SRQ		CCP5			T0CKI			
7	4	RA5	AN4	C2OUT		SRNQ	HLVDIN			SS1				
10	7	RA6												OSC2/ CLKO
9	6	RA7												OSC1/ CLKI
21	18	RB0	AN12			SRI		CCP4 FLT0		SS2		INT0	Y	
22	19	RB1	AN10	C12IN3-				P1C		SCK2/ SCL2		INT1	Y	
23	20	RB2	AN8		CTED1			P1B		SD2/ SDA		INT2	Y	
24	21	RB3	AN9	C12IN2-	CTED2			CCP2/ P2A ⁽¹⁾		SDO2			Y	
25	22	RB4	AN11					P1D			T5G	I0C	Y	
26	23	RB5	AN13					CCP3/ P3A ⁽⁴⁾ P2B ⁽³⁾			T1G T3CK ⁽²⁾	I0C	Y	
27	24	RB6							TX2/CK2			I0C	Y	PGC
28	25	RB7							RX2/DT2			I0C	Y	PGD
11	8	RC0						P2B ⁽³⁾			SOSCO/ T1CKI T3CK ⁽²⁾ T3G			
12	9	RC1						CCP2/ P2A ⁽¹⁾			SOSCI			
13	10	RC2	AN14		CTPLS			CCP1/ P1A			T5CKI			
14	11	RC3	AN15							SCK1/ SCL1				
15	12	RC4	AN16							SDI1/ SDA1				
16	13	RC5	AN17							SDO1				
17	14	RC6	AN18					CCP3/ P3A ⁽⁴⁾	TX1/CK1					
18	15	RC7	AN19					P3B	RX1/DT1					
1	26	RE3												MCLR/ VPP
8	5													VSS
19	16													VSS
20	17													VDD

Note 1: CCP2/P2A multiplexed in fuses.
 2: T3CKI multiplexed in fuses.
 3: P2B multiplexed in fuses.
 4: CCP3/P3A multiplexed in fuses.

Apêndice B – Registradores de controle do PCA9626.

Table 4. Register summary^{[1][2]}

Register number (hex)	D5	D4	D3	D2	D1	D0	Name	Type	Function
00	0	0	0	0	0	0	MODE1	read/write	Mode register 1
01	0	0	0	0	0	1	MODE2	read/write	Mode register 2
02	0	0	0	0	1	0	PWM0	read/write	brightness control LED0
03	0	0	0	0	1	1	PWM1	read/write	brightness control LED1
04	0	0	0	1	0	0	PWM2	read/write	brightness control LED2
05	0	0	0	1	0	1	PWM3	read/write	brightness control LED3
06	0	0	0	1	1	0	PWM4	read/write	brightness control LED4
07	0	0	0	1	1	1	PWM5	read/write	brightness control LED5
08	0	0	1	0	0	0	PWM6	read/write	brightness control LED6
09	0	0	1	0	0	1	PWM7	read/write	brightness control LED7
0A	0	0	1	0	1	0	PWM8	read/write	brightness control LED8
0B	0	0	1	0	1	1	PWM9	read/write	brightness control LED9
0C	0	0	1	1	0	0	PWM10	read/write	brightness control LED10
0D	0	0	1	1	0	1	PWM11	read/write	brightness control LED11
0E	0	0	1	1	1	0	PWM12	read/write	brightness control LED12
0F	0	0	1	1	1	1	PWM13	read/write	brightness control LED13
10	0	1	0	0	0	0	PWM14	read/write	brightness control LED14
11	0	1	0	0	0	1	PWM15	read/write	brightness control LED15
12	0	1	0	0	1	0	PWM16	read/write	brightness control LED16
13	0	1	0	0	1	1	PWM17	read/write	brightness control LED17

Table 4. Register summary^{[1][2]} ...continued

Register number (hex)	D5	D4	D3	D2	D1	D0	Name	Type	Function
14	0	1	0	1	0	0	PWM18	read/write	brightness control LED18
15	0	1	0	1	0	1	PWM19	read/write	brightness control LED19
16	0	1	0	1	1	0	PWM20	read/write	brightness control LED20
17	0	1	0	1	1	1	PWM21	read/write	brightness control LED21
18	0	1	1	0	0	0	PWM22	read/write	brightness control LED22
19	0	1	1	0	0	1	PWM23	read/write	brightness control LED23
1A	0	1	1	0	1	0	GRPPWM	read/write	group duty cycle control
1B	0	1	1	0	1	1	GRPFREQ	read/write	group frequency
1C	0	1	1	1	0	0	CHASE	read/write	chase control
1D	0	1	1	1	0	1	LEDOUT0	read/write	LED output state 0
1E	0	1	1	1	1	0	LEDOUT1	read/write	LED output state 1
1F	0	1	1	1	1	1	LEDOUT2	read/write	LED output state 2
20	1	0	0	0	0	0	LEDOUT3	read/write	LED output state 3
21	1	0	0	0	0	1	LEDOUT4	read/write	LED output state 4
22	1	0	0	0	1	0	LEDOUT5	read/write	LED output state 5
23	1	0	0	0	1	1	SUBADR1	read/write	I ² C-bus subaddress 1
24	1	0	0	1	0	0	SUBADR2	read/write	I ² C-bus subaddress 2
25	1	0	0	1	0	1	SUBADR3	read/write	I ² C-bus subaddress 3
26	1	0	0	1	1	0	ALLCALLADR	read/write	LED All Call I ² C-bus address

Apêndice C – Software do DMX24PWM.

```

/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*
*                               Função Principal                               *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
//-----

void main ()
{
unsigned char Dado;
BYTE DataRcv[nCH];
unsigned long Contador;
BOOL erro;
BOOL complete;
BYTE indice;

    // Definicao e inicializacao das portas
    ConfiguraPortas();
    //BlinkLeds(1);
    StartPCA();
    SetupSerial();
    INTCON = 0x00;             // Desabilita todas as interrupções
    INTCON2 = 0x00;
    INTCON3 = 0x00;
    T0CON = 0x00;
    T1CON = 0x00;
    T2CON = 0x00;
    T3CON = 0x00;
    CCP1CON = 0x00;

    //inicialização das variáveis
    for(Contador=0;Contador<nCH;Contador++){
        DataRcv[Contador] = 0;
    }
    Contador = 0;

    // Loop infinito
    do {

        do{ // loop que aguarda o break e o dado 0 no início
            erro = FALSE;

            do{ // aguarda ocorrer um erro

                if(PIR1bits.RCIF){ // chegou um dado?

                    if (RCSTAbits.FERR){ // é um dado inválido?

```

```

        erro = TRUE;
    }
    Dado = RCREG; // limpa o buffer
}
} while(!erro);
complete = FALSE;
do{ // aguarda um dado válido
    if(PIR1bits.RCIF){ //chegou um dado?
        if (!RCSTAbits.FERR){ // é um dado válido?
            complete = TRUE;
        }
    }
} while (!complete);
Dado = RCREG;
} while (Dado != 0); // em caso de erro volta ao começo
//neste ponto está sincronizado com o início

Contador = 0;
erro = FALSE;
complete = FALSE;
Dado = 0;
indice = 0;
while((Contador<505) && (!erro) && (!complete)){
    if (PIR1bits.RCIF){ //recebeu um novo dado ?
        if (!RCSTAbits.FERR){ //o dado não está corrompido
            //dado válido
            Dado = RCREG;
            Contador ++;
            if((Contador >= CH_DESEJADO) && (Contador < CH_DESEJADO+nCH)){
                DataRcv[indice] = Dado;
                indice++;
                if (indice==nCH){complete = TRUE;}
            }
        }else{ //para o caso do dado estar corrompido
            erro = TRUE;
        }
    }
}
}
StopSerial();
LED1 = !LED1;
AplicaNovoDado(DataRcv);

```

```

        StartSerial();

    } while(TRUE);

}

// Fim main *****

```

Apêndice D – Driver para acesso do PCA9626

//LookUpTable que determina as cores

```

unsigned char LUT [64][3] = {

//      R          G          B
    1,          0,          0,          //0 - 15
    0,          1,          0,          //16 - 31
    0,          0,          1,          //32 - 47
    1,          1,          0,          //48 - 65
    1,          0,          1,          //64 - 79
    0,          1,          1,          //80 - 95
    1,          3,          0,          //96 - 111
    1,          0,          3,          //112 - 127
    0,          1,          3,          //128 - 143
    3,          1,          0,          //144 - 159
    3,          0,          1,          //160 - 175
    0,          3,          1,          //176 - 195
    3,          0,          3,          //192 - 207
    0,          3,          3,          //208 - 223
    1,          3,          3,          //224 - 239
    1,          1,          1 }; //240 - 255

```

```

#include <i2c.h>

#include <p18F2480.h>

#include "PCA9626Driver.h"

#include <delays.h> //Delay library functions

#include "main.h"

#include "IO.h"

//-----

void StartPCA(void)

{

    OpenI2C(MASTER, SLEW_ON); // inicia o HW I2C

    SSPADD = 0x6F; // configura velocidade de comunicação

    //Registadores de MODO

    PCASend(ADDR, MODE1, 0x00);

    PCASend(ADDR, MODE2, 0x00);

```

```

//Configura o modo das saídas
PCASend(ADDR, LEDOUT0, 0xAA);
PCASend(ADDR, LEDOUT1, 0xAA);
PCASend(ADDR, LEDOUT2, 0xAA);
PCASend(ADDR, LEDOUT3, 0xAA);
PCASend(ADDR, LEDOUT4, 0xAA);
PCASend(ADDR, LEDOUT5, 0xAA);
//configura para auto incremento
PCASend(ADDR, MODE1, 0b10100000);
}

// Fim StartPCA *****

//-----

void AplicaNovoDado(BYTE *DataRcv)
{
unsigned char cor, intensidade;
unsigned char fator;
unsigned char i, k;
unsigned char valor[24];
    for(i=0;i<8;i++){
        //identificação da COR
        cor = DataRcv[2*i] >> 4;    //seleciona a parte da cor
        cor &= 0x0F;    //só para garantir a validade do dado

        // identifica a intensidade
        intensidade = DataRcv[2*i+1];

        for(k=0;k<3;k++){
            fator = LUT[cor][k]; //lê o valor na tabela
            switch(fator){
                case 0:
                    valor[3*i+k] = 0;
                    break;
                case 1:
                    valor[3*i+k] = intensidade;
                    break;
                default:
                    valor[3*i+k] = intensidade >> (fator-1);    //função de divisão por 2
                    break;
            }
        }
    }
}

```

```

        }
    }
}

EnviaTodosOsPWM(valor);
}

// Fim AplicaNovoDado *****

int EnviaTodosOsPWM(BYTE *valor)
{
    unsigned char i;
//melhorar aqui e colocar a função de envio em sequencia
    IdleI2C();           // ensure module is idle
    StartI2C();         // initiate START condition
    while ( SSPCON2bits.SEN ); // wait until start condition is over
    if ( PIR2bits.BCLIF ) // test for bus collision
    {
        return ( -1 ); // return with Bus Collision error
    }
    else // start condition successful
    {
        if ( WriteI2C( ADDR ) ) // write byte - R/W bit should be 0
        {
            StopI2C();
            return ( -3 ); // set error for write collision
        }

        if ( !SSPCON2bits.ACKSTAT ) // test for ACK condition received
        {
            if ( WriteI2C( 0b10000010 ) ) // write word address for EEPROM
            {
                StopI2C();
                return ( -3 ); // set error for write collision
            }

            for(i=0;i<23;i++){
                if ( !SSPCON2bits.ACKSTAT ) // test for ACK condition received
                {
                    if ( WriteI2C( valor[i] ) ) // data byte for EEPROM
                    {
                        StopI2C();

```

```

        return ( -3 );          // set error for write collision
    }
}
else
{
    StopI2C();
    return ( -2 );            // return with Not Ack error condition
}
}

}
else
{
    StopI2C();
    return ( -2 );          // return with Not Ack error condition
}
}
StopI2C();                  // send STOP condition
while ( SSPCON2bits.PEN );  // wait until stop condition is over
if ( PIR2bits.BCLIF )      // test for bus collision
{
    return ( -1 );          // return with Bus Collision error
}
return ( 0 );              // return with no error

```


Apêndice E – Diagrama Elétrico

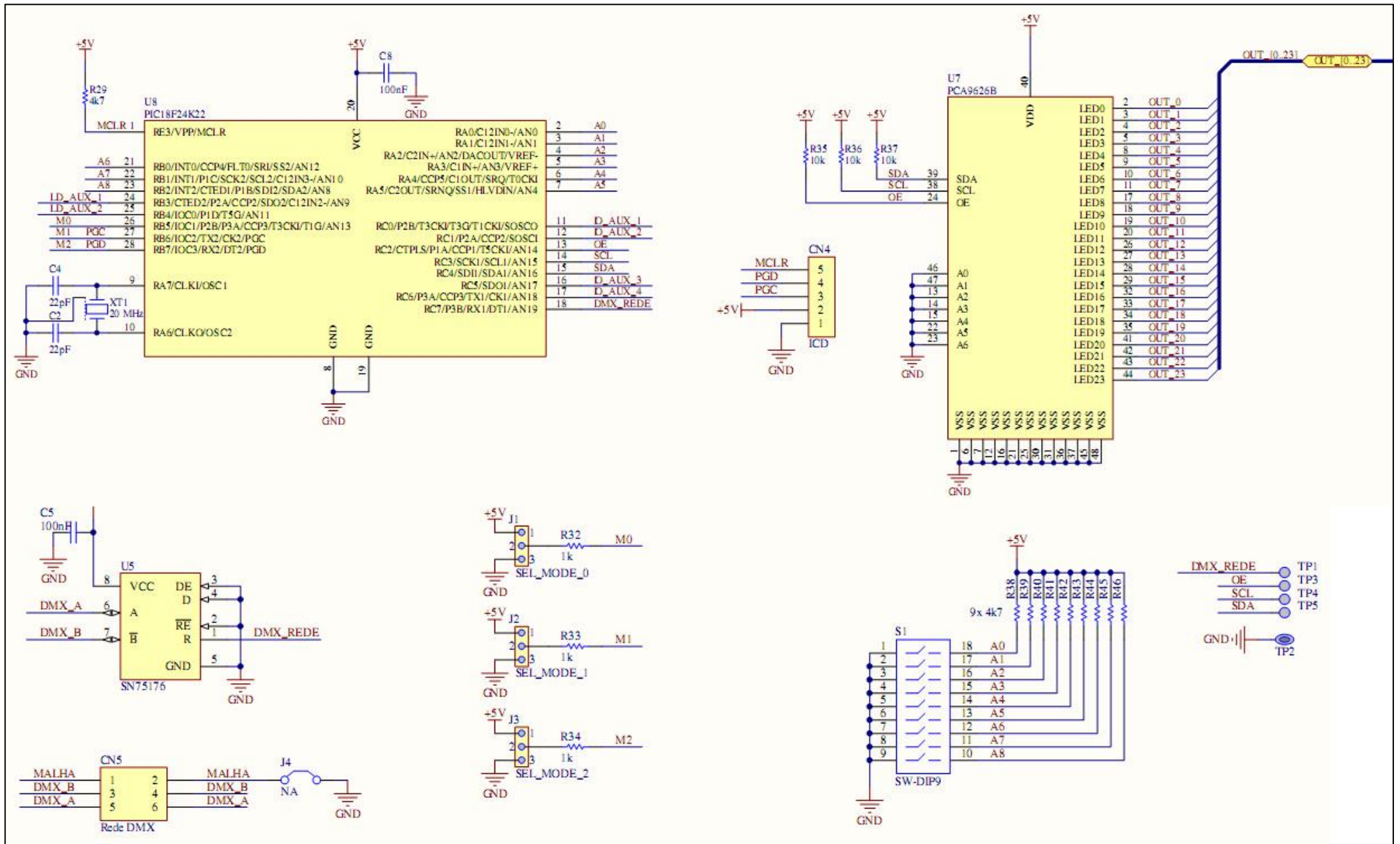


Figura 45 - Diagrama Eléctrico - Parte A.

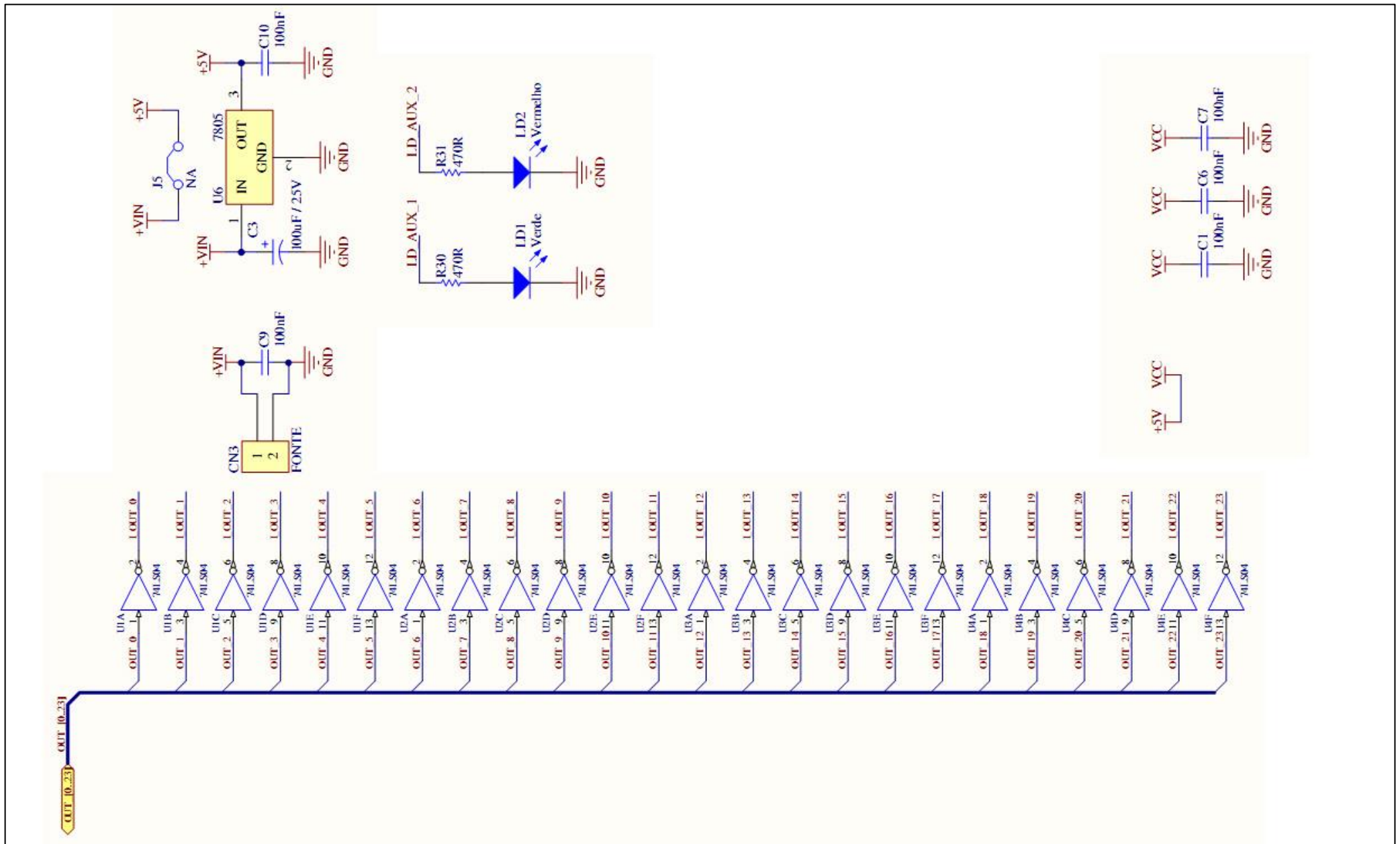


Figura 46 - Diagrama Elétrico - Parte B.

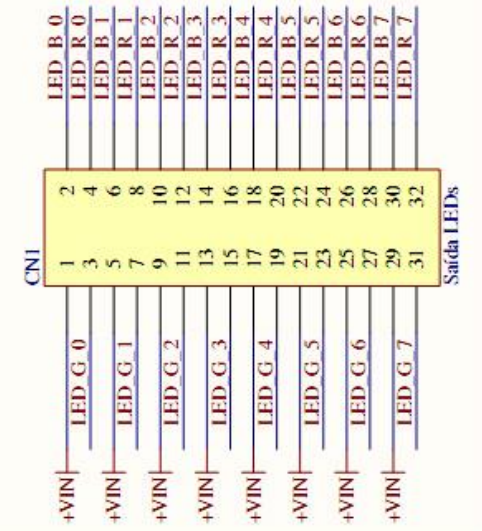
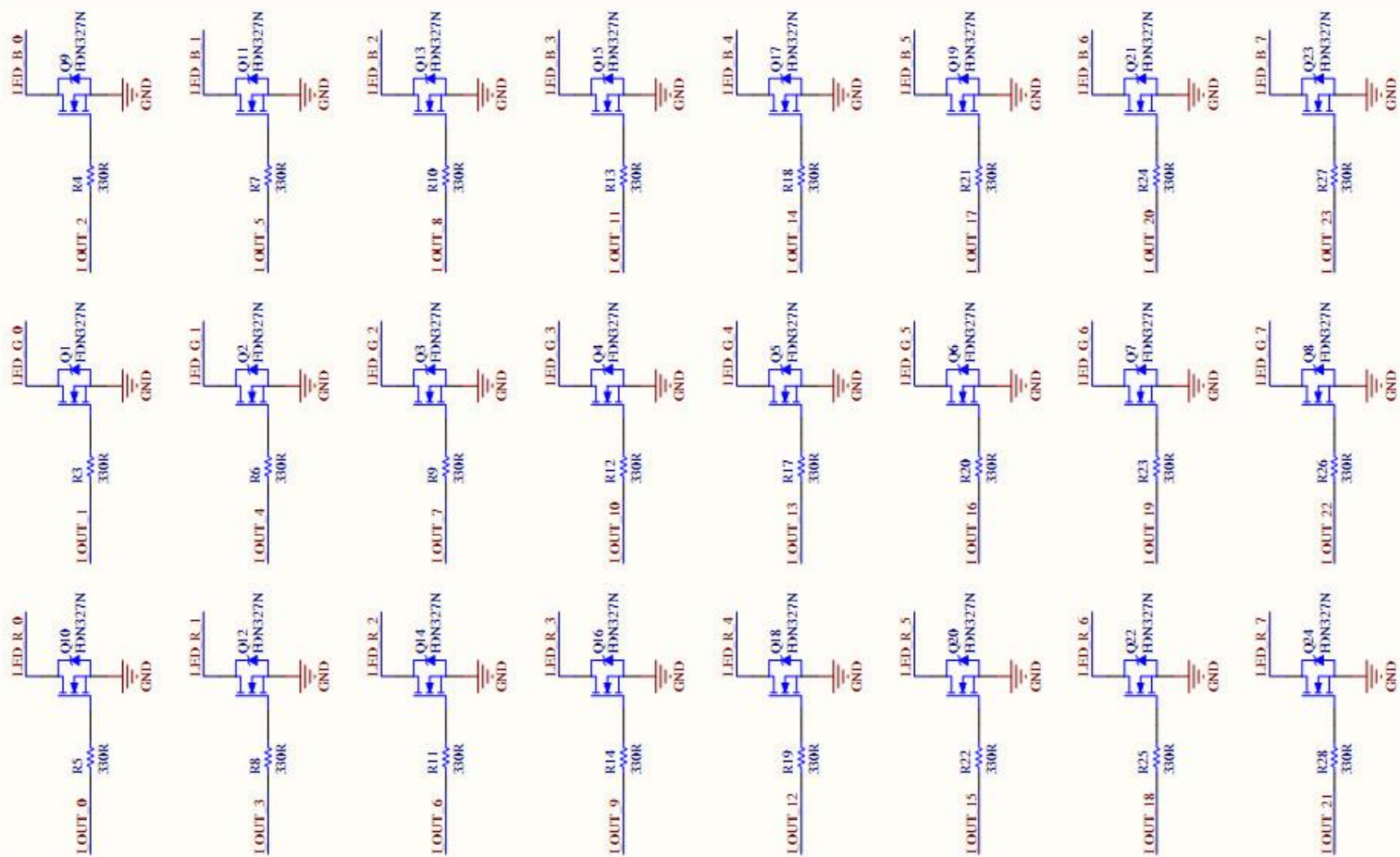


Figura 47 - Diagrama Elétrico - Parte C