

**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE  
MINAS GERAIS**  
*CAMPUS DIVINÓPOLIS*  
GRADUAÇÃO EM ENGENHARIA MECATRÔNICA

Hugo Araújo Machado

**Impressora para grafia direta em placas de circuito impresso  
utilizando comandos numéricos computadorizados**

Divinópolis.

2015.

Hugo Araújo Machado

## **Impressora para grafia direta em placas de circuito impresso utilizando comandos numéricos computadorizados**

Trabalho de Conclusão de Curso apresentado ao Colegiado de Graduação em Engenharia Mecatrônica como parte dos requisitos exigidos para a obtenção do título de Engenheiro Mecatrônico.

Áreas de integração: Mecânica e Eletrônica.

Orientador: Prof. Me. Claudio H. G. Santos

Co-orientador: Prof. Me. Daniel Alves Costa

Divinópolis.

2015.

Hugo Araújo Machado

## **Impressora para grafia direta em placas de circuito impresso utilizando comandos numéricos computadorizados**

Monografia de Trabalho de Conclusão de Curso apresentada ao Colegiado de Graduação em Engenharia Mecatrônica como parte dos requisitos exigidos para a obtenção do título de Engenheiro Mecatrônico.

Áreas de integração: Mecânica e Eletrônica.

Comissão Avaliadora:

---

Prof. Me. Claudio Henrique Gomes dos Santos  
Centro Federal de Educação Tecnológica de Minas Gerais - CEFET MG

---

Prof. Dr. Valter Junior de Souza Leite  
Centro Federal de Educação Tecnológica de Minas Gerais - CEFET MG

---

Prof. Dr. Marlon Antônio Pinheiro  
Centro Federal de Educação Tecnológica de Minas Gerais - CEFET MG

Divinópolis.

2015.

Esta conquista é de todos que perseveraram ao meu lado.

## **AGRADECIMENTOS**

Agradeço primeiramente a Deus, por ter me dado o dom da perseverança, sabedoria, e dedicação para traspasar todos os obstáculos até aqui enfrentados.

Agradeço de todo o meu coração, toda minha família que, mesmo em momentos difíceis, nunca me abandonou e deixou de dar importância às minhas conquistas. Agradeço de forma especial, a minha namorada, Paola, que sempre esteve ao meu lado me incentivando, apoiando e ajudando nos momentos difíceis. Esta vitória também é sua.

Menções de agradecimento ao professor Dr. Valter Leite pelas oportunidades concedidas, aos meus orientadores Cláudio Santos e Daniel Alves, por me guiarem neste caminho tortuoso e aos laboratoristas Enderson e Mateus, que sempre disponibilizaram de tempo para me ajudar neste processo de formação pessoal.

Por fim, agradeço aos meus companheiros de guerra, que mesmo estando cheios de afazeres, sempre disponibilizaram de um tempo para ajudar, aconselhar e até mesmo conversar sobre qualquer coisa que nos viesse à cabeça. Obrigado a todos.



## RESUMO

Com o aumento do número de cursos técnicos de graduação e pós-graduação, a demanda da produção de placas de circuito impresso vem aumentando significativamente, já que resultados experimentais são vitais no meio acadêmico. Não somente no meio acadêmico, mas também na indústria é demandado a produção de placas como protótipos para desenvolvimento de determinado produto. Em outras palavras, independente se o meio é acadêmico ou industrial, sempre é demandado a produção de uma primeira placa eletrônica a fim de se obter os resultados experimentais sobre o funcionamento de um determinado produto ou método. Empresas especializadas na fabricação de PCIs demandam muito tempo até que a placa chegue ao consumidor, o que é inviável para uma produção com maior urgência. Outra opção é a fabricação da própria PCI a partir de métodos caseiros, como o de transferência por calor e tinta fotossensível. Porém estes métodos não garantem uma placa de qualidade e, por experiência do próprio autor, são muito suscetíveis a falhas, existindo a possibilidade de precisar repetir os processos uma variedade de vezes. Já os métodos de produção de placas a partir de máquinas CNC ainda possuem custo elevado. Este trabalho propôs um método de furação e impressão direta de circuitos em placas de cobre que visa reduzir o tempo de fabricação e a interferência humana no processo. Uma mesa orientada foi construída para o deslocamento dos ferramentais de furação e pintura. Todo o sistema de acionamento foi desenvolvido e fabricado e o sistema respondeu como o desejado. Os suportes das ferramentas e da placa estão em desenvolvimento e o seu resultado será apresentado na versão final deste relatório.

**Palavras-chave:** CNC. PCI. Furação. Pintura. *Layout*.

## ABSTRACT

With the increase of technical courses undergraduate and postgraduate, demand the production of printed circuit boards has increased significantly since experimental results are vital both in academia. Not only in academia but also in the industry it is demanded production boards as prototypes for development of a product. In other words, regardless of whether the medium is the academic or industrial production is always demanded of a first circuit board in order to obtain the experimental results of the operation of a particular product or method. Companies specializing in PCB manufacturing require a long time until the card reaches the consumer, which is unaffordable for a production with greater urgency. Another option is to manufacture the own PCB starting homemade methods, such as heat transfer and photosensitive ink. But these methods do not guarantee a quality plate, and experience of the author himself and acquaintances, are very susceptible to failure, with the possibility of having to repeat the process a variety of times. As for the plates production methods starting from CNC machines still have high cost. This paper proposed a method of drilling and direct printing of circuits on copper plates which aims to reduce the manufacturing time and human interference in the process. A targeted table was built for the displacement of drilling and painting tooling. The entire drive system is designed and manufactured and the system responded as desired. Supporters of the tools and the board are under development and its results will be presented in the final version of this report.

**Keywords:** *NC. PCB. Drilling. Painting. Layout.*

## LISTA DE ILUSTRAÇÕES

Figura 1.1 -	(a) Método (HOLLOWSTATE, 2015). (b) - Resultado da transferência de calor.....	2
Figura 1.2 -	(a) Câmara de luz negra (HANDMADES, 2015). (b) Resultado da transferência por tinta fotossensível.....	3
Figura 1.3 -	(a) Método e (b) Resultado do processo de fresamento de uma PCI (CPRIUS, 2015).....	4
Figura 1.4 -	(a) Método e (b) Resultado do processo de roteamento por laser (HOBBY, 2015).....	5
Figura 1.5 -	(a) Máquina de prototipagem ProtoLaser S. (b) PCI fabricada pela ProtoLaser (LPKF, 2015).....	6
Figura 2.1 -	Diagrama de blocos do CI Allegro A4988 (ALLEGRO, 2015).....	8
Figura 2.2 -	Diagrama do circuito do driver Pololu A4988 (POLOLU, 2015).....	9
Figura 2.3 -	Localização do trimmer no driver (Adaptado de POLOLU, 2015).....	12
Figura 2.4 -	Corrente passante nas bobinas do motor e nos Sense 1 e 2 (ALLEGRO, 2015).....	13
Figura 2.5 -	Plataforma Arduino Uno R3 (Adaptado de PROTONEER, 2015).....	15
Figura 2.6 -	Layout de conexões externas do driver A4988 (POLOLU, 2015).....	16
Figura 2.7 -	(a) Layout da camada inferior da placa. (b) Layout da camada superior da placa.....	16
Figura 2.8 -	(a) Motor de passo unipolar e (b) Motor de passo bipolar (DMT, 2015).....	18
Figura 2.9 -	Motor de passo Nema 17 (DAMENCNC, 2015).....	19
Figura 3.1 -	Projeto da placa desenvolvida no Proteus.....	21
Figura 3.2 -	(a) Camada inferior da placa de suporte. (b) Camada Superior da placa de suporte.....	22
Figura 3.3 -	Placa de interface entre controlador e drivers.....	23
Figura 3.4 -	Projeto da estrutura mecânica.....	23

Figura 3.5 -	Estrutura em alumínio.....	24
Figura 3.6 -	Fresadora CNC Fanuc.....	24
Figura 3.7 -	Fresadora manual Diplomat, localizada no galpão de mecânica.....	25
Figura 3.8 -	Modelo 3D para suporte das barras guias do eixo X .....	26
Figura 3.9 -	Encaixe da peça de suporte à barra guia do eixo X.....	26
Figura 3.10 -	Vista da parte inferior da impressora.....	27
Figura 3.11 -	(a) Tensionador de correia e sua polia. (b) Parafuso para ajuste de tensão da correia.....	27
Figura 3.12 -	Acoplamento de correia.....	28
Figura 3.13 -	Sistema para movimentação do carro ao longo do eixo Y...	28
Figura 3.14 -	Carro que se locomove ao longo do eixo Y e suporte para ferramentas que desloca ao longo do eixo Z.....	29
Figura 3.15 -	Vista geral da estrutura mecânica construída e ligações elétricas.....	30
Figura 3.16 -	Interface do programa Xloader.....	31
Figura 3.17 -	Interface do programa Universal Gcode Sender.....	31
Figura A3.1 -	Janela de geração de código G do Proteus	44
Figura A3.2 -	Interface do FlatCam e mapa de furação da placa.....	45
Figura A3.3 -	(a) Janela Selected e (b) janela Options.....	46

## LISTA DE TABELAS

Tabela 2.1 -	Resolução do passo do motor dada alimentação das portas MSIx ( <i>POLOLU, 2015</i> ).....	10
Tabela 4.1 -	Medidas para conferir alinhamento da estrutura mecânica.	33
Tabela 4.2 -	Máxima excursão de cada eixo.....	33
Tabela 4.3 -	Teste de deslocamento dos eixos coordenados.....	34
Tabela A2.21 -	Lista de gastos com materiais e ferramentas.....	42

# LISTA DE NOTAÇÕES E ACRÔNIMOS

## Letras Latinas

- D - Diâmetro [m]  
 $J_L$  - Inércia da carga [ $Kgf.cm^2$ ]  
 $J_T$  - Inércia total [ $Kgf.cm^2$ ]  
 $J_p$  - Inércia da polia [ $Kgf.cm^2$ ]  
 $J_B$  - Inércia da correia [ $Kgf.cm^2$ ]  
 $J_M$  - Inércia do motor [ $Kgf.cm^2$ ]  
 $K_S$  - Fator de segurança  
 $m_P$  - Massa da polia [Kg]  
 $m_B$  - Massa da correia [Kg]  
 $m_L$  - Massa da carga [Kg]  
 $T_a$  - Torque de aceleração [N.m]  
 $T_L$  - Torque de carga [N.m]  
 $T_M$  - Torque requerido do motor [N.m]

## Letras Gregas

- $\eta$  - Eficiência mecânica  
 $\alpha$  - Ângulo de inclinação  
 $\mu$  - Coeficiente de fricção das superfícies deslizantes

## Acrônimos

- CI - Circuito Integrado  
CNC – Comando Numérico Computadorizado  
PCI - Placa de Circuito Impresso

# SUMÁRIO

1- Introdução.....	1
1.1- Definição do problema.....	2
1.2- Motivação.....	5
1.3- Objetivo geral.....	6
1.4- Objetivos específicos.....	6
1.5- Estado da arte.....	6
1.6- Escopo do trabalho.....	7
2- Revisão da literatura.....	8
3- Metodologia.....	20
4- Resultados e discussões.....	33
5- Conclusões.....	35
6- Propostas para trabalhos futuros.....	36
7- Referências bibliográficas.....	37
8- Apêndice.....	40
9- Glossário.....	47

# 1- INTRODUÇÃO

Com o aumento do desenvolvimento de projetos acadêmicos, extensão e até mesmo de produtos industriais, a demanda por produção de placas eletrônicas para coleta de dados e para o funcionamento de experimentos, acompanha este crescimento. Várias pessoas e empresas terceirizam este processo de fabricação para fabricantes especializados na produção de PCBs. Contudo este método demanda tempo e o custo unitário de placas eletrônicas é elevado, o que é inviável para projetos de testes rápidos e de baixo orçamento.

Como solução mais rápida, muitos partem para a utilização de métodos caseiros, como os de transferência por calor e por tinta fotossensível. Por vivência do próprio autor, estes métodos se mostraram muito suscetíveis a falhas, exigindo que o processo fosse repetido diversas vezes até que se obtenha uma placa funcional, ainda com uma qualidade aquém da desejada. Alguns ainda constroem sua própria máquina de fabricar PCBs, mas a construção da máquina demanda muito tempo, o que também inviabiliza o desenvolvimento de processos rápidos. Então, como fabricar uma placa de circuito impresso de qualidade, a baixo custo e rápido?

Este projeto propõe um método para a fabricação de PCBs que permite aos frequentadores do CEFET, Campus-V, localizado em Divinópolis, a impressão direta do *layout* do circuito desejado sobre o cobre, além de realizar a furação para o posicionamento e soldagem dos componentes. O método proposto consiste em uma estrutura que permite um deslocamento bidimensional das ferramentas e unidimensional na placa de cobre, o que resulta em um movimento global em três dimensões. O acionamento das partes móveis foi dado por três motores de passo, sendo cada um deles responsável por um eixo orientado. Estes motores são acionados por *drivers* Pololu A4988 (POLOLU, 2015), que por sua vez recebem as instruções de movimento de um Arduino Uno R3. O Arduino é carregado com um *firmware* responsável por realizar a interpretação de códigos G em linguagem de máquina. Os programas Proteus, *FlatCam* e *Universal Gcode Sender* são responsáveis por permitir o desenvolvimento do layout de um circuito qualquer, converter a imagem em código G e transferir este código para o Arduino, respectivamente.

## 1.1- DEFINIÇÃO DO PROBLEMA

Existe atualmente uma série de métodos que permitem a fabricação de PCBs em casa e no meio acadêmico sem a necessidade de adquirir maquinário com o custo elevado. Contudo, cada um destes métodos possui ao menos um ponto negativo que decresce drasticamente a qualidade da placa produzida. Nos tópicos subsequentes são mostrados os principais métodos de produção existente no meio acadêmico e amador.

### 1.1.1 Método de transferência por calor

Este método constitui de imprimir a imagem do *layout* do circuito desejado em uma folha de papel *couché* ou transparência, afixá-la sobre a placa de cobre e, com um ferro de passar roupa, ir aquecendo o material até que a tinta seja transferida para a placa.

Este é o método mais barato, de implementação mais simples e funcional, contudo é o mais suscetível a falhas. Observando a Figura 1.1, nota-se um processo de transferência por calor que não saiu como esperado:



Figura 1.1(a): Método (HOLLOWSTATE, 2015) e 1.1(b):Resultado da transferência de calor.

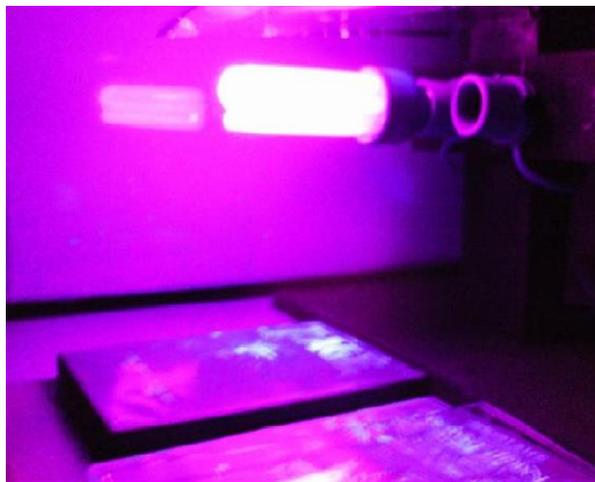
Como é possível observar, nenhuma trilha foi reproduzida devidamente, tornando esta transferência inutilizável. Este processo é muito incerto e pode fazer com que o projetista repita o processo várias vezes até conseguir uma transferência aceitável.

### 1.1.2 Método de transferência por tinta fotossensível

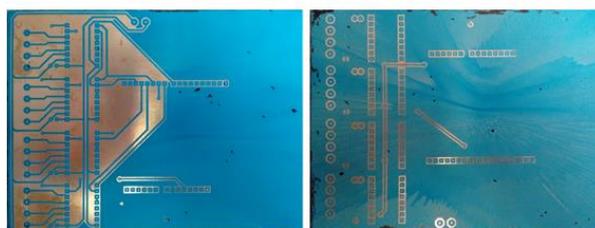
É mais sofisticado que o método anterior e produz resultados mais significativos, ou seja, gera o *layout* do circuito de maneira mais efetiva. Consiste em imprimir uma imagem negativa do *layout* da placa em uma folha transparência, afixá-la sobre uma placa de cobre banhada por uma fina e uniforme camada de tinta fotossensível e expô-la à luz negra.

Para conseguir esta fina camada de tinta, é necessário um dispositivo que gire a placa, que já está banhada de tinta, à aproximadamente 12.000 rpm. Quando a tinta é espalhada, esta deverá ser secada podendo ser utilizado um secador de cabelo comum. Quando estiver seca, deve-se encobrir a placa com a imagem gerada no papel transparência.

Em seguida, basta colocar a placa em uma câmara iluminada por luz negra. A parte que for atingida pela luz, irá aderir-se a placa, ao contrapasso que a parte que não foi atingida não aderirá. Por fim, retira-se a transparência e mergulha a placa em uma solução de bicarbonato de sódio ou removedor de tinta para a limpeza, então corroê-la em uma solução de percloroeto de ferro. O resultado é evidenciado na Figura 1.2.



(a)



(b)

Figura 1.2(a): Câmara de luz negra (HANDMADES, 2015) e 1.2(b): Resultado da transferência por tinta fotossensível.

Mesmo sendo um método que apresenta melhorias muito significativas em relação ao primeiro, ainda apresenta muitas falhas recorrentes do processo, o que é possível observar na Figura 1.2, onde encontra-se várias marcas pretas, que representam os locais que tiveram problemas de transferência da tinta e foi necessária uma correção posterior utilizando pincel.

A maior vantagem deste processo é a capacidade de se fabricar PCI de dupla face com bastante precisão. Sua maior desvantagem é o processo para chegar a tal ponto, pois é difícil de se conseguir uma máquina para espalhar a tinta de maneira correta e que funcione. Além do mais é um processo que demanda tempo, aproximadamente três horas para uma placa como mostrada na Figura 1.2(b).

### 1.1.3 Método de fresamento

É o primeiro método apresentado que possui alguma automação. Este consiste em pegar o código G gerado pelos próprios programas que criam circuitos, transferi-lo para um outro *software* que interprete este código e envie para a máquina.

A máquina, por sua vez, possui uma ferramenta capaz de fresar a superfície da placa de cobre, surgindo o roteamento sobre a mesma. A Figura 1.3 apresenta o resultado do fresamento de uma PCI:

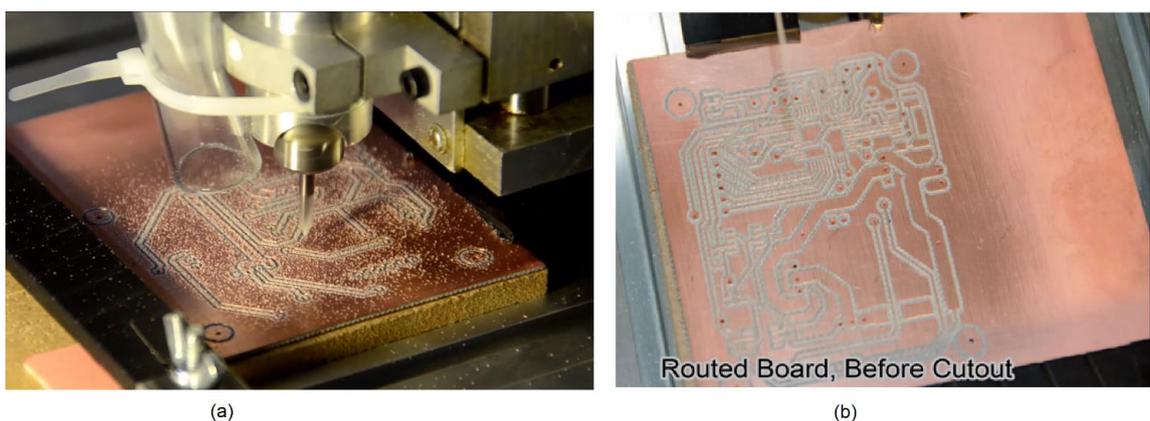


Figura 1.3(a): Método e 1.3(b): Resultado do processo de fresamento de uma PCI (CPRIUS, 2015).

Apesar de criar uma placa em perfeito estado de utilização, este método gera muita sujeira e cria um acabamento muito ruim na placa. Outro ponto negativo é

que, se por alguma razão, o *layout* da placa estiver errado, a placa de fenolite não poderá ser reutilizada, já que este não é um processo reversível.

#### 1.1.4 Método do roteamento por *laser*

Este método consiste em aplicar um feixe de *laser* sobre a camada de cobre que se deseja retirar e em seguida banhar a placa em água. Também se utiliza do código G exportado dos *softwares* de prototipagem de circuitos. A Figura 1.4 mostra o resultado de uma CNC *laser* fabricada artesanalmente:

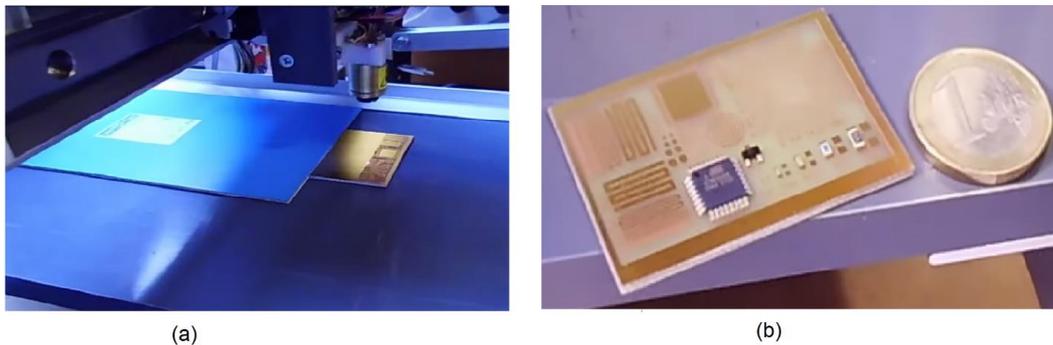


Figura 1.4(a): Método e 1.4(b): Resultado do processo de roteamento por *laser* (HOBBY, 2015).

Apesar deste processo possuir alta qualidade, o tempo gasto para produzir uma placa com tamanho aproximado de duas moedas de 1 Euro foi de aproximadamente doze minutos, sendo que o projeto proposto deve imprimir placas com três vezes este tamanho, levando apenas a metade do tempo.

## 1.2- MOTIVAÇÃO

A maior motivação para desenvolver um projeto de tal magnitude e especificidade é incentivar, cada vez mais, alunos a produzirem suas placas de circuito impresso com mais qualidade, retomando a motivação para criar circuitos cada vez mais complexos e com aparência de produtos industrializados.

Isto pode ser uma razão a mais para se penetrar em uma área de conhecimento técnico e tecnológico, já que PCIs de má qualidade são impecílios para melhor desempenho de qualquer projeto, seja ele um manipulador, um carro guiado por controle remoto, ou até mesmo uma fresadora, enfim, qualquer situação que envolva fabricação de sistemas eletrônicos.

Com o advento deste projeto, mais alunos se interessarão em desenvolver seus próprios trabalhos e fazê-los de maneira mais profissional, preparando-os, em auxílio ao curso de Engenharia Mecatrônica, para o mercado de trabalho.

### 1.3- OBJETIVO GERAL

Desenvolver uma mesa orientada que seja capaz de locomover o ferramental que será utilizado para a criação direta do *layout* de uma PCI qualquer através de deposição de tinta, seguida pela furação da mesma a fim de poder instalar e soldar componentes *through hole*.

### 1.4- OBJETIVOS ESPECÍFICOS

Este projeto visa desenvolver um método que seja simples e intuitivo de ser utilizado, exigindo o mínimo possível da interferência do usuário, em outras palavras, este deverá apenas inserir a placa sobre a máquina, projetar o *layout* e mandá-lo para imprimir, retirando uma placa apenas para a corrosão após o processo.

### 1.5- ESTADO DA ARTE

O que há de mais moderno para a fabricação de PCI em menor escala é a ProtoLaser S (LPKF, 2015). Esta máquina produz placas em altíssima qualidade como mostra a Figura 1.5:

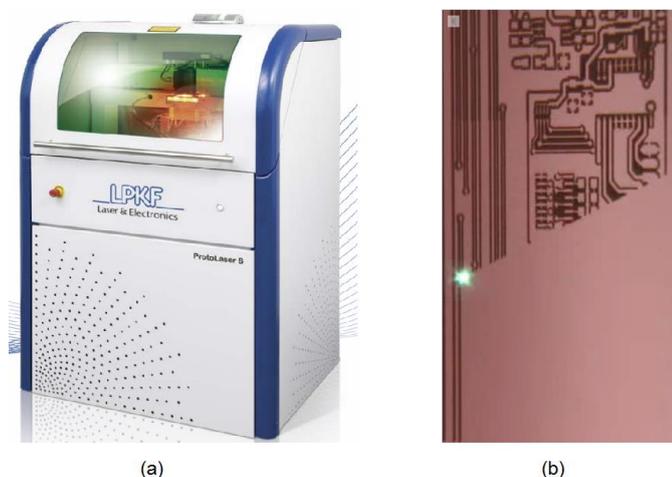


Figura 1.5(a): Máquina de prototipagem ProtoLaser S. Figura 1.5(b): PCI fabricada pela ProtoLaser (LPKF, 2015).

Com este método é possível a fabricação de placas com trilhas muito finas. Esta característica de criar trilhas bem finas é de grande utilidade para reduzir o tamanho das placas e para a utilização de componentes SMD com tamanho reduzido, com trilhas de 50 $\mu$ m e espaçamento entre trilhas de 25 $\mu$ m (LPKF, 2015). Este processo é igual ao citado na seção 1.1.4, contudo, por ser um aparelho industrial e com alto investimento em pesquisa e desenvolvimento, a *ProtoLaser* imprime circuitos em alta velocidade (LPKF,2015).

## **1.6- ESCOPO DO TRABALHO**

Este trabalho mostra o estudo e o processo para a construção de uma máquina do tipo mesa orientada. Seus capítulos foram divididos em Revisão de Literatura, no qual procura-se oferecer o embasamento necessário para melhor entender os conceitos de funcionamento destas máquinas. No capítulo Metodologia, foi abordada a forma com que a parte de eletrônica e mecânica foram construídas, além de mostrar o processo para a fabricação de uma PCI neste protótipo no subtópico Comunicação computador-máquina. O capítulo Resultados e Discussões debateu os dados oriundos deste projeto. Por fim, no capítulo Conclusões, procurou expor as principais contribuições e características do trabalho, culminando em propostas para trabalhos futuros, sugeridos no capítulo subsequente.

## 2- REVISÃO DA LITERATURA

### 2.1 Parte elétrica

Para que haja uma comunicação entre o computador e a máquina, uma placa eletrônica que seja capaz de receber os comandos do computador, interpretá-los e enviá-los em forma de sinais que a máquina consiga converter em movimento de suas partes deve ser fabricada.

Para o desenvolvimento de uma placa eletrônica é necessário ter o conhecimento básico de funcionamento dos componentes a serem utilizados e qual sua função no circuito. Nos tópicos a seguir, uma breve explanação dos componentes utilizados para a construção de uma placa para o controle de motores de passo foi feita.

#### 2.1.1 Driver

A movimentação dos eixos da mesa orientada foi dada através de motores de passo acoplados à barras roscadas. Mas, para que pudessem funcionar corretamente, foi necessário o uso de *drivers* a fim de controlar seu posicionamento e velocidade. A Figura 2.1 mostra o diagrama de blocos do driver para motor de passo A4988 da Allegro (ALLEGRO,2015).

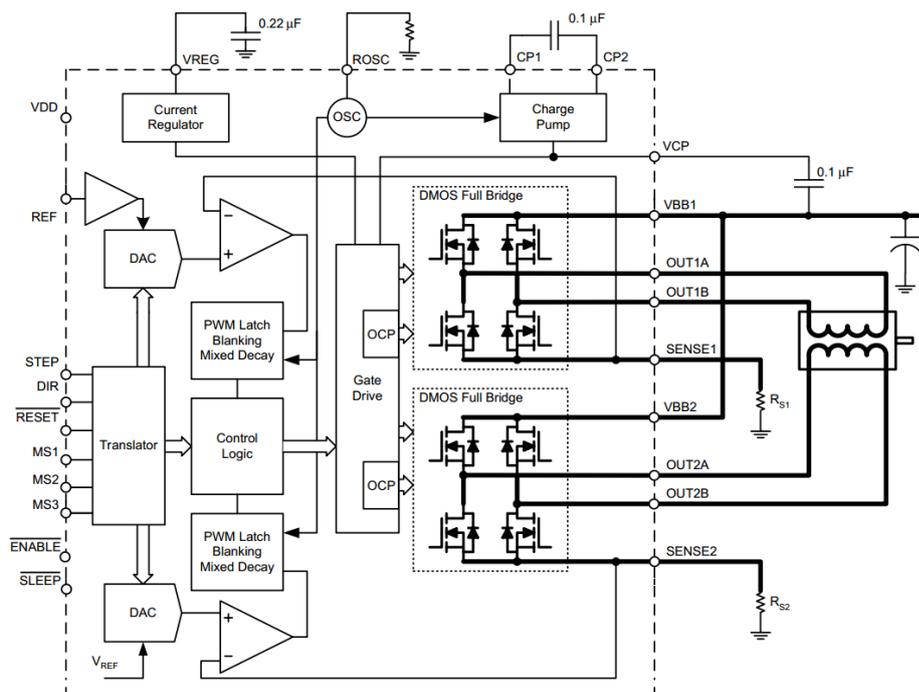


Figura 2.1: Diagrama de blocos do CI Allegro A4988 (ALLEGRO, 2015).

A estrutura interna do CI mostrado na Figura 2.1 possui todas as características necessárias para o controle de um motor de passo, que vai desde entradas digitais e analógicas, utilizadas para caracterizar sentido de giro, velocidade e número de passos que devem ser realizados pelo motor, até pontes H que são as conexões diretas para as bobinas do motor.

Contudo, mesmo possuindo todas as características necessárias para comandar um motor de passo, para que possa funcionar corretamente e de modo mais intuitivo ao usuário, são necessárias algumas conexões externas, como mostra o esquemático da Figura 2.2:

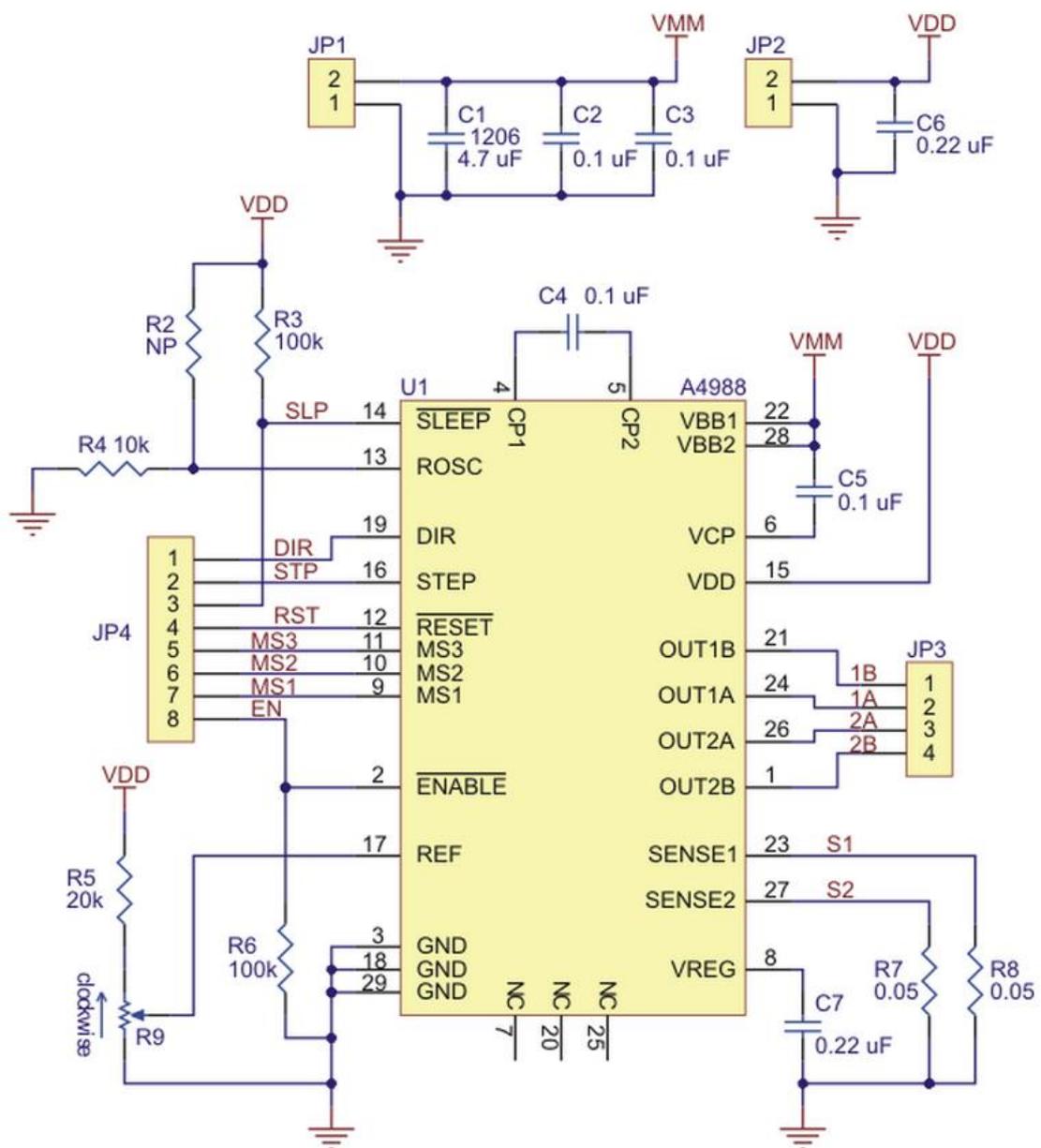


Figura 2.2: Diagrama do circuito do driver Pololu A4988 (POLOLU, 2015).

### 2.1.1.1 Bloco JP4

Os sinais de entrada para o driver são dados pelo bloco denominado 'JP4'. Cada uma destas oito entradas é responsável por receber um comando específico.

A primeira porta de entrada é o pino 'DIR', o qual é responsável por selecionar a direção de giro do motor. O sentido do giro só muda quando detecta uma borda de subida do comando *step* (STP), que corresponde ao pino 2 deste mesmo bloco. O sinal na entrada STP é interpretado pelo bloco '*Translator*' (Figura 6), e a cada transição de nível baixo para alto aplicado neste é traduzido e transformado em um micropasso para o motor.

O terceiro pino é o de ' $\overline{\text{Sleep}}$ ' e é utilizado para reduzir o consumo de energia pelo circuito. Isto ocorre da seguinte maneira: se não houver tensão suficiente aplicada em sua entrada, seu modo *sleep* é ativado e todo o circuito, incluindo as saídas da ponte H (Figura 2.1) são desligadas. Quando sua entrada está alimentada, o circuito é ativado e permite a operação normal do mesmo.

A quarta entrada é o ' $\overline{\text{Reset}}$ '. Quando em nível lógico baixo, gera no bloco '*Translator*' (Figura 2.1) um estado de *Home*, que desliga todas as saídas das pontes H. Somente quando sua entrada está em nível lógico alto é que os comandos de *step* são repassados para o circuito.

Os pinos cinco, seis e sete representam as configurações de passo 'MS3', 'MS2' e 'MS1', respectivamente. Estes pinos são responsáveis por reduzir o comprimento do passo do motor de acordo com a configuração de suas alimentações. A Tabela 2.1 mostra qual a redução de comprimento de passo é obtida dada alimentação a nível lógico nas portas 'MSx' (POLOLU, 2015).

Tabela 2.1: Resolução do passo do motor dada alimentação das portas MSx (POLOLU, 2015).

MS1	MS2	MS3	Resolução do Micropasso
Baixo	Baixo	Baixo	Passo completo
Alto	Baixo	Baixo	$\frac{1}{2}$ passo
Baixo	Alto	Baixo	$\frac{1}{4}$ de passo
Alto	Alto	Baixo	$\frac{1}{8}$ de passo
Alto	Alto	Alto	$\frac{1}{16}$ de passo

Na porta oito, temos o comando ' $\overline{Enable}$ ' que habilita/desabilita as saídas da ponte H (Figura 2.1) que alimentam as bobinas do motor de passo. Em nível lógico baixo, esta porta habilita as saídas das pontes H, e em nível lógico alto, desabilita as mesmas.

O bloco denominado 'JP3' da Figura 2.2 representa as saídas para o acionamento do motor de passo. As saídas 1A e 1B vão para a primeira bobina do motor de passo, enquanto as saídas 2A e 2B vão para a segunda bobina.

### 2.1.1.2 Limitação de corrente

Para que não haja sobrecorrente nas bobinas de um motor de passo, o driver possui um limitador de corrente. Isto é bem útil, pois a entrada do driver referente à alimentação do motor deve ser suprida com tensões que variam de 8 à 35V. Isto significa que, se fosse utilizar um motor que possui alimentação inferior à fornecida, sem o limitador de corrente este iria queimar.

Sendo assim, basta saber a corrente nominal suportada pelo motor e realizar alguns cálculos para a regulação desta no driver. A maneira mais simples de calcular a tensão que o regulador de corrente necessita para fornecer a corrente máxima desejada é dada por (POLOLU, 2015):

$$V_{ref} = \frac{I_{max}}{2.5} \quad (2.1)$$

Onde  $I_{max}$  é a corrente máxima desejada. Por uma característica construtiva do *driver*, a corrente nominal que chega aos terminais do motor é apenas 70% da corrente máxima que é desejada. Por isto é necessário que, para que a corrente nominal seja igual à corrente máxima, se divida a corrente máxima por uma proporção de 0,7:

$$I_{nom} = \frac{I_{max}}{0.7} \quad (2.2)$$

Em que  $I_{nom}$  é a corrente nominal. Isto foi verificado experimentalmente, porém em Pololu (2015) consta a mesma informação. Ao substituir a corrente nominal fornecida em (2.2) pela corrente máxima de (2.1), temos:

$$V_{ref} = \frac{I_{nom}}{2.5} \quad (2.3)$$

Para exemplificar, vamos considerar um motor de passo que suporte até 2A em suas bobinas. Este valor é sua corrente máxima. Portanto, aplicando este valor em (2.2):

$$I_{nom} = \frac{I_{max}}{0.7} = \frac{2}{0.7} = 2.857$$

Aplicando este valor em (2.3):

$$V_{ref} = \frac{I_{nom}}{2.5} = \frac{2.857}{2.5} = 1.143V$$

A tensão  $V_{ref}$  é a tensão de referência a ser regulada no *trimmer* do *driver*. A Figura 2.3 mostra a localização do *trimmer*, que está circulado de vermelho (Pololu, 2015):

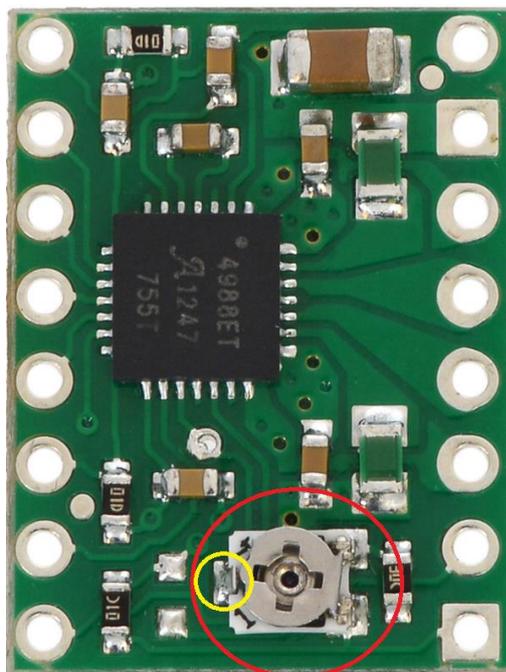


Figura 2.3: Localização do trimmer no driver (Adaptado de POLOLU, 2015).

Já a parte circulado em amarelo, é o ponto de tensão  $V_{ref}$ . Isto significa que, ao colocar uma ponta de prova de multímetro neste ponto e a outra sobre o *ground* do *driver* que se localiza no pino mais à direita e abaixo é esperado que haja uma tensão de 1,143V, referentes à um motor de 2A.

### 2.1.1.3 Senses 1 e 2

Cada uma das pontes H é acionada por um circuito de PWM controlado por corrente. Cada par de DMOS, quando ligado, permite a passagem de corrente, através dos resistores  $R_{s1}$  e  $R_{s2}$  via enrolamentos da bobina do motor. A tensão medida sobre os resistores é comparada com a tensão de referência de saída do bloco 'DAC' (Figura 2.2).

Quando as tensões comparadas são iguais, a entrada de sinal do PWM é resetada e inicia-se modo decaimento de decaimento da corrente durante determinado intervalo de tempo. A Figura 2.4 mostra a corrente passando por cada uma das bobinas do motor e, conseqüentemente, pelos resistores  $R_{s1}$  e  $R_{s2}$ :

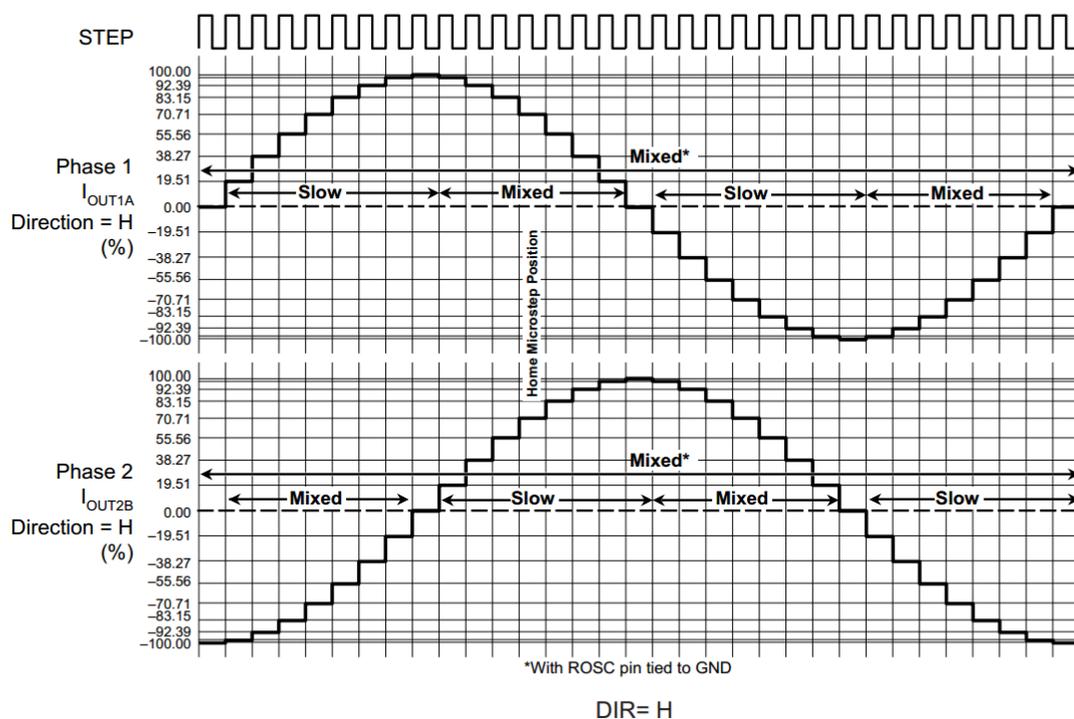


Figura 2.4: Corrente passante nas bobinas do motor e nos Sense 1 e 2 (ALLEGRO, 2015).

Como é possível observar na Figura 9, a onda senoidal da corrente é quantizada em amplitude. Isto é devido ao controle de corrente por histerese, ou seja, ela trabalha em um intervalo de valores e não em um valor predefinido fixo. Como a corrente é controlada por PWM, seus valores dão pequenos saltos discretos no tempo. O tamanho destes degraus depende da velocidade de chaveamento do PWM.

Outro fato a ser observado é a defasagem de  $45^\circ$  entre as formas de onda. Esta é de extrema importância para que haja a sincronia entre as correntes que passam pelas bobinas, para que assim o motor possa girar de maneira correta e suave. Se não houvesse a defasagem, o motor iria apenas travar seu rotor ao invés de girar. Caso a defasagem fosse muito grande, o motor daria passos muito grandes e perderia a suavidade em seu movimento.

Outro detalhe a respeito dos *Sense 1* e *Sense 2* é que se um comando de *step* for enviado sem que o motor esteja conectado aos terminais do *driver*, a tensão sobre  $R_{s1}$  e  $R_{s2}$  sempre será igual a zero. Isto significa que quando esta for comparada com a tensão de referência, a diferença entre as duas sempre será máxima, o que fará com que o *driver* tente aumentar a corrente indefinidamente, o que acarreta na queima do mesmo devido à sobrecorrente.

### 2.1.2 Placa de *Interface*

Ainda que haja o *driver* apropriado para comandar os motores de passo, este não realiza os comandos por si só, ou seja, é necessário um controlador para enviar as informações de velocidade, posicionamento e sentido de rotação ao motor. Por razões de simplicidade de programação e por possuir *softwares* de domínio público que realizam a interpretação de código G, foi escolhida uma plataforma de projetos Arduino com um controlador Atmel 328.

A Figura 2.5 mostra um Arduino modelo Uno R3 e suas portas que serão utilizadas para comandar os *drivers*:

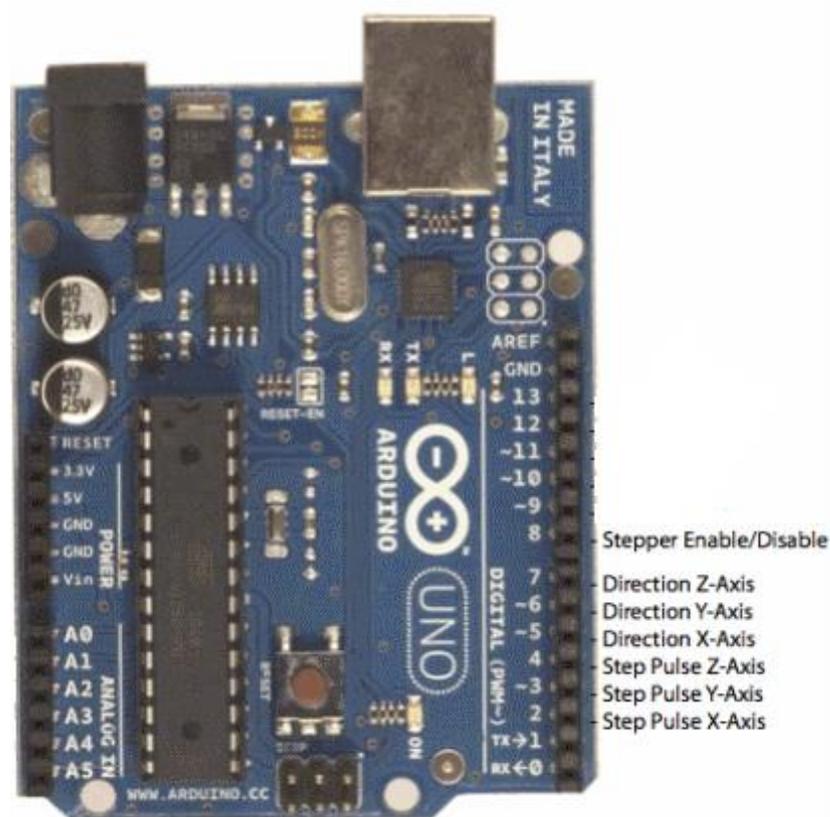


Figura 2.5: Plataforma Arduino Uno R3 (Adaptado de PROTONEER, 2015).

Como é possível observar, há dois sinais que devem ser enviados para o motor de cada eixo, um para direção e outro para incrementar passos à este, além de possuir um terceiro sinal que habilita/desabilita o *driver*. Este último pode ser comum à todos os *drivers*, mas os outros deverão ser exclusivos de cada um. A Figura 2.6 mostra a localização dos pinos a serem conectados com o controlador, além de evidenciar outras conexões externas (POLOLU, 2015):

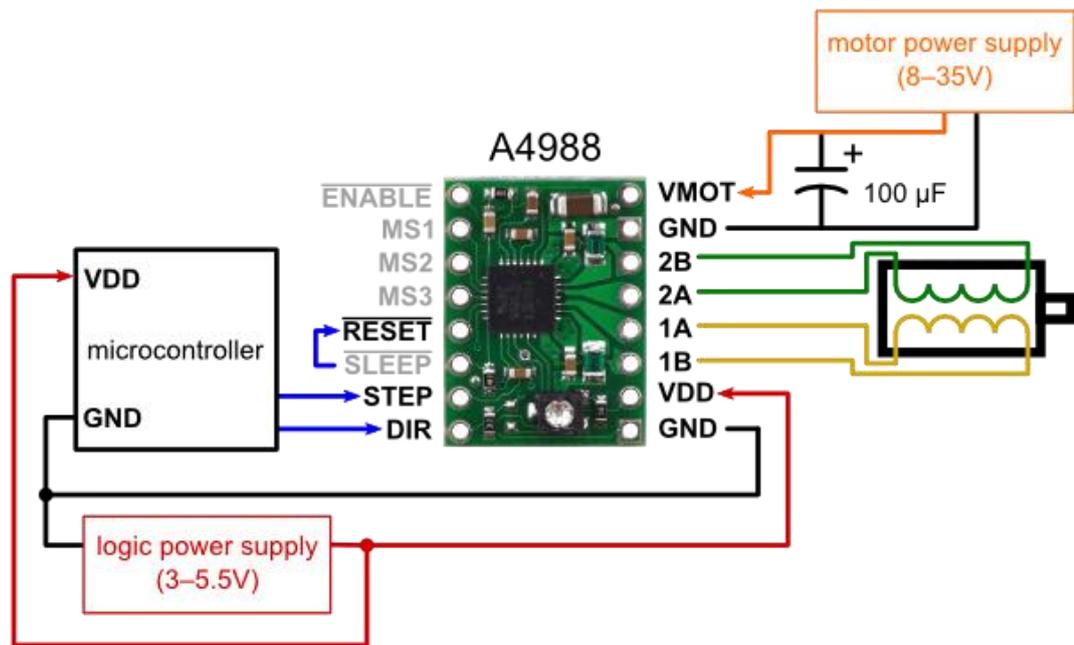


Figura 2.6: Layout de conexões externas do driver A4988 (POLOLU, 2015).

### 2.1.2.1 Plano de terra

Na fabricação de uma placa para a interconexão de uma plataforma Arduino e múltiplos *drivers* A4988 para controle de quatro motores de passo, foi necessário a criação de um plano de terra, como mostrado na Figura 2.7:

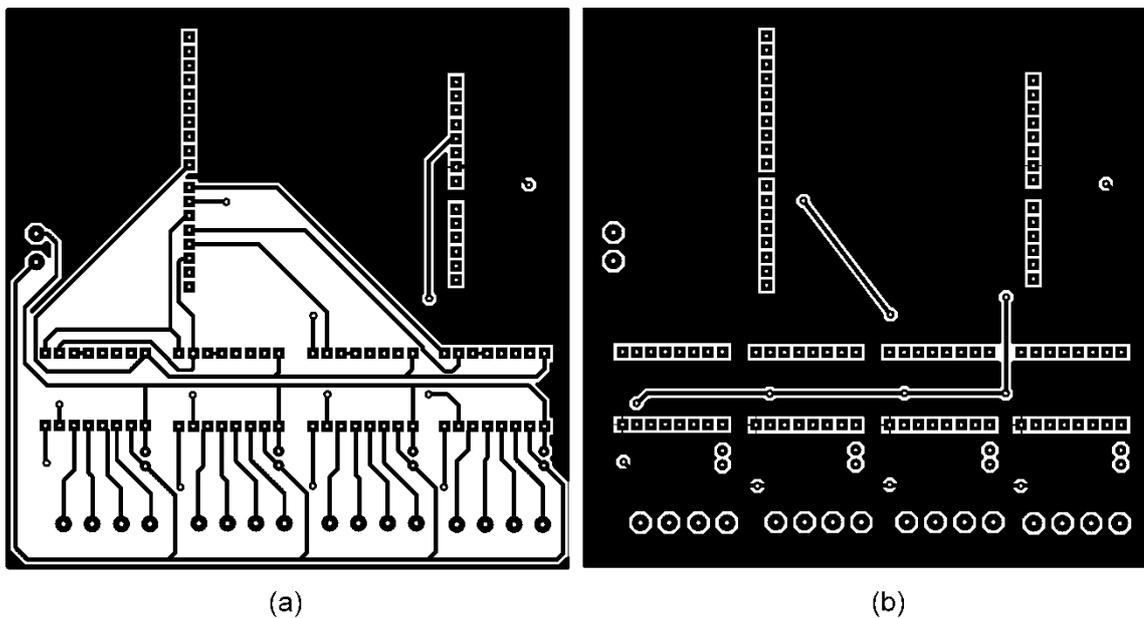


Figura 2.7(a): Layout da camada inferior da placa. Figura 2.7(b): Layout da camada superior da placa.

Como é possível observar na Figura 2.7, existe uma grande superfície em preto sobre ambas camadas e representam onde há superfície de cobre. Elas são interconectadas e denominadas 'Plano de terra'. De acordo com Paul (2006) o plano de terra é importante para reduzir ao máximo a indutância do caminho de retorno (terra) da corrente para a fonte.

Além do mais, este plano oferece inúmeros pontos de retorno para todo o circuito, fazendo com que as correntes de retorno não necessitem percorrer longas trilhas até chegar à fonte de alimentação, o que aumentaria a indutância do caminho e por consequência o retorno da corrente através das trilhas de alimentação (PAUL, 2006).

Uma outra grande vantagem de se criar plano de terra é reduzir o tamanho dos *loops* criados pelas trilhas, pois estes são extremamente suscetíveis a interferência eletromagnética de outras partes do circuito ou até mesmo de outros dispositivos elétricos/eletrônicos que estejam próximos (PAUL, 2006).

### 2.1.2.2 Capacitores de desacoplamento:

Como é possível notar na Figura 2.6, existe um capacitor de  $100 \mu F$  entre a tensão de alimentação do motor de passo (VMOT) e o terra do mesmo. Este capacitor em questão é conhecido como 'Capacitor de desacoplamento' e sua função é de minimizar ao máximo o ripple de tensão na entrada do pino de alimentação durante a distribuição da potência. Assim como no caminho de retorno, o caminho de potência também possui indutância em suas trilhas e, conseqüentemente, uma queda de tensão dada por:

$$V(t) = L \frac{di}{dt}$$

Esta queda de tensão nas trilhas de potência devido à indutância também é conhecida como "*Power rail collapse*" ou "Colapso da trilha de potência" (PAUL, 2006). O capacitor de desacoplamento corrige os efeitos da queda de tensão causada pelos motivos supracitados, reduzindo o *ripple* e permitindo que a tensão entregue seja a mais próxima possível da nominal.

### 2.1.2.3 Quinas de 90° em trilhas

Quinas de 90° em trilhas de placa de circuito impresso caracterizam uma descontinuidade da resistência do condutor para o fluxo de corrente. Esta descontinuidade pode degradar consideravelmente o desempenho do circuito (EHOW, 2015). Além do mais, esta descontinuidade também faz com que a trilha se comporte como uma antena.

Por estes motivos, quando for necessário que uma trilha mude de curso, ao invés de criar quinas com 90°, utiliza-se de desvios mais suaves, de 45°, fazendo com que a 'curva' seja feita em duas etapas. Este método foi aplicado e pode ser visto na Figura 2.7.

## 2.2 Motor de passo e relação *step/mm*

Motores de passo são amplamente utilizados no meio industrial (PAWLAK, 2007). Sua aplicação vai de mecanismos de alimentação de combustível em motores à controle de posicionamento em máquinas programadas por CNC. Os motores de passo podem ser classificados como unipolares ou bipolares, de acordo com suas características construtivas, como mostrado na Figura 2.8. O motor unipolar da Figura 2.8(a) recebe este nome pois possui um fio em comum e são ligados no meio de cada uma das bobinas avulsas. O motor bipolar, Figura 2.8(b), não possui nenhuma ligação comum, gerando dois polos separados, o que dá origem a seu nome.

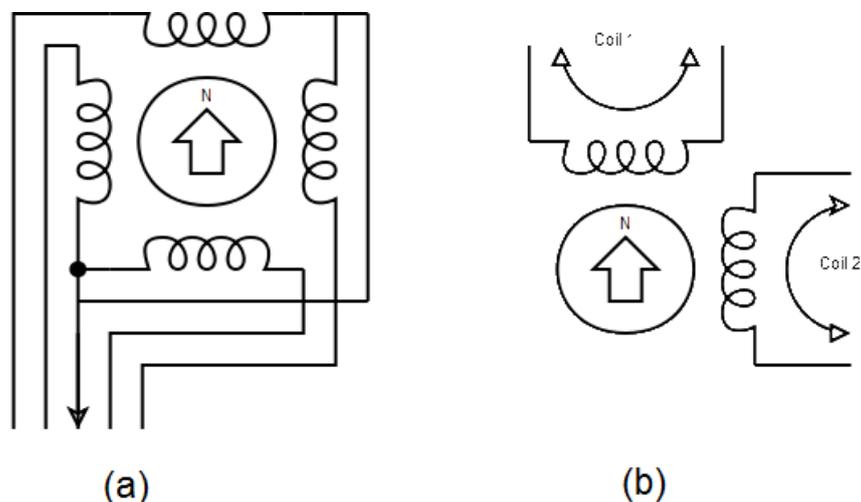


Figura 2.8(a): Motor de passo unipolar e Figura 2.8(b): Motor de passo bipolar (DMT, 2015).

O motor de passo utilizado neste projeto foi um Nema 17 bipolar, como mostra a Figura 2.9. Este motor possui duzentos passos por volta, ou seja, a cada duzentos pulsos de *step* enviados pelo controlador ao motor, este terá respondido com um giro de 360°.

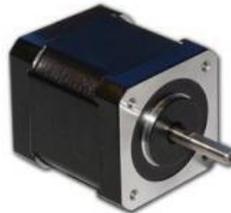


Figura 2.9: Motor de passo Nema 17 (DAMENCNC, 2015).

A relação *step/mm* indica quantos passos um motor deverá dar para que o sistema se desloque por um milímetro. Esta relação depende de quantos passos por volta o motor possui, qual a relação *step/mm* do sistema de tração utilizado e se a função *microstep* foi utilizada (Tabela 2.1). Para este projeto foram utilizados dois sistemas de tração diferentes. O primeiro sistema é o de polia e correia, como mostra a Figura 3.10. Este possui uma relação  $R_{pc} = 5 \text{ step/mm}$  mas, com a função *microstep* de  $1/16$ , esta relação deve ser dezesseis vezes maior, ou seja:

$$R_{pc} = 5 \times 16 = 80 \text{ steps/mm}$$

O segundo sistema utilizado foi o de barra de fuso como mostra a Figura 3.14. A barra utilizada possui uma rosca M8, com uma relação de 1,2 milímetros por volta. Como o motor de passo possui duzentos passos por volta, significa que a barra de fuso acoplada à ele gera um deslocamento linear de 1,2 milímetros a cada duzentos passos, ou uma relação de sistema de barra de fuso  $R_{bf} = 160 \text{ steps/mm}$ . Como este sistema possui a função *microstep* de  $1/4$ , sua relação também deve ser dezesseis vezes maior:

$$R_{bf} = 4 \times 160 = 640 \text{ steps/mm}$$

## 3- METODOLOGIA

Para a construção deste projeto, partiu-se do pressuposto que a máquina deve fabricar placas de até 250x250mm. A partir desta informação, construiu-se uma máquina que atendesse esta especificação e que ocupasse o menos espaço possível.

### 3.1 Recursos físicos utilizados

Para a realização deste projeto, a disponibilização de laboratórios e ferramentais foi de suma importância. Para a fabricação de uma placa eletrônica para comandar o acionamento dos motores de passo a partir de um computador qualquer, foi necessário o uso do laboratório 310, localizado no CEFET-MG, Campus-V, Divinópolis, prédio 3. Este laboratório possuía todos os aparatos que permitiram a fabricação, teste e montagem da placa, tais como espalhador de tinta fotossensível, câmara de luz negra, multímetro de bancada e fonte corrente contínua.

A estrutura mecânica foi fabricada em dois locais diferentes. O primeiro local foi o laboratório de CNC, sala 311, também localizado no Campus-V prédio 3 e disponibilizava de uma fresadora CNC Denford. O segundo local foi o galpão de mecânica, uma estrutura locada pelo CEFET-MG e que se localiza fora das dependências físicas do Campus-V e ofertava maquinário de maior porte, como a serra fita e uma fresadora manual. A maior parte dos ferramentais e vários treinamentos requeridos para a construção deste trabalho de conclusão de curso foi disponibilizada pelos laboratoristas do Campus.

### 3.2 Parte eletrônica

O desenvolvimento da placa eletrônica que realiza a comunicação entre o computador e a máquina começou com a análise de quais terminais do Arduino (Figura 2.5) se conectariam aos terminais do motor de passo (Figura 2.6). A partir destas conexões necessárias, partiu-se para o projeto do *layout* da placa. O programa utilizado para tal processo foi o Proteus e o *layout* criado é mostrado na Figura 3.1.

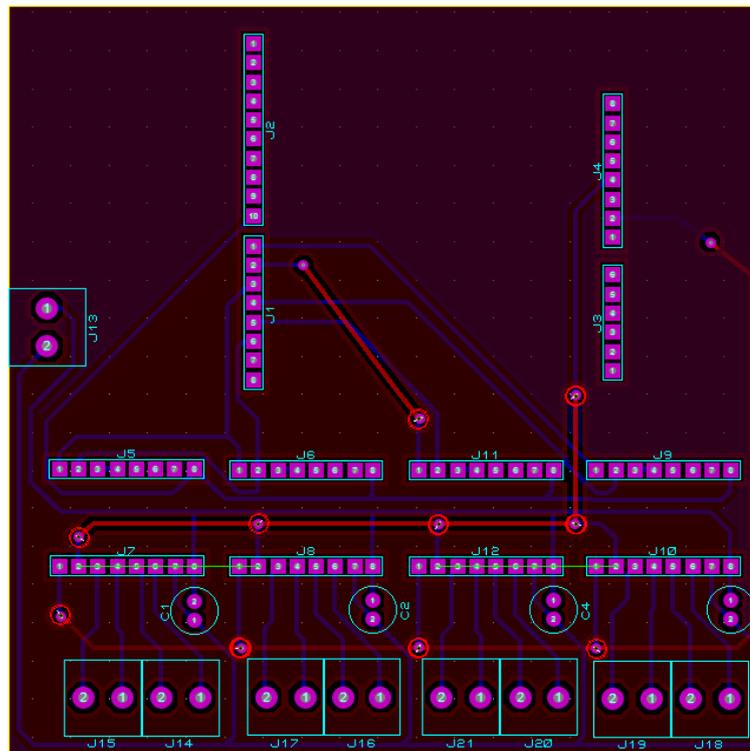


Figura 3.1: Projeto da placa desenvolvida no Proteus.

A placa mostrada na Figura 3.1 possui duas camadas com a finalidade de otimizar o tamanho da placa e reduzir os loops criados caso esta possuísse apenas uma camada. As duas camadas são mostradas separadamente na Figura 2.7, em a parte preta representa onde há deposição de cobre e em branco há a retirada do cobre.

Com o projeto feito e revisado, o método de transferência por tinta fotossensível, mostrada na Seção 1.1.2, foi utilizado para gerar a imagem sobre a placa de cobre. O primeiro passo foi imprimir a imagem de cada uma das camadas do circuito em folha de transparência avulsas. Vale a pena ressaltar que é necessário que se imprima o negativo da imagem do circuito, pois a parte que não recebe luz é que tem a camada de cobre retirada.

Em seguida, aplicou-se uma camada de tinta fotossensível sobre a placa de cobre e espalhou-a com o espalhador de tinta até que esta ficasse bem fina e uniforme. Após aplicar e espalhar a tinta, foi preciso secá-la utilizando um secador de cabelos.

Com a tinta seca, foi possível alinhar e prender as impressões do circuito sobre a placa pintada e levá-la para a câmara de luz negra similar à mostrada na Figura 1.2(a). Após cinco minutos de incidência de luz negra, a placa foi retirada e banhada em uma solução de bicarbonato de sódio até que a camada de tinta que

não recebeu incidência da luz saísse. O resultado é mostrado na Figura 1.2(b). Como a parte de cobre que deve ser removida está exposta e a parte que não deve ser removida está protegida pela camada de tinta, o processo seguinte foi de mergulhar a placa em uma solução de Percloroeto de Ferro até que o excesso de cobre seja retirado. Após esta etapa, lavou-se a placa em água e palha de aço, resultando no circuito da Figura 3.2.

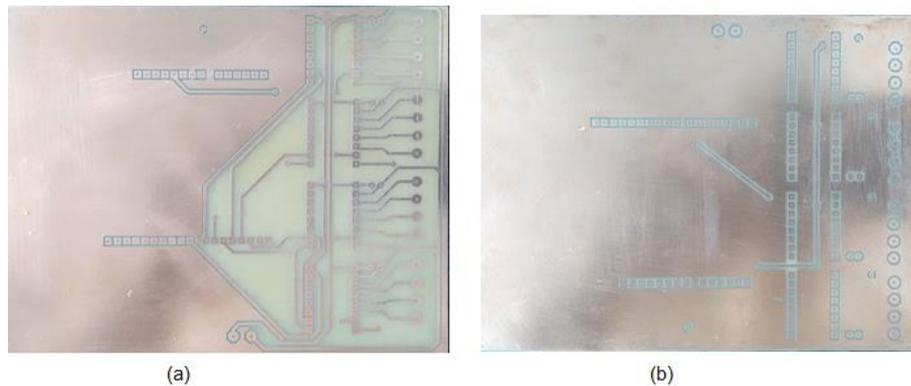


Figura 3.2(a): Camada inferior da placa de suporte. Figura 3.2(b): Camada Superior da placa de suporte.

Antes de terminar o processo de fabricação, testes de continuidade foram realizados para assegurar que não houvessem trilhas rompidas nem curto circuitos. Por fim, após conferir a continuidade da placa, foram realizados furos para que os componentes pudessem ser encaixados e soldados. A Figura 3.3 exibe o resultado de todo o processo de fabricação da placa.

Existem três tipos de encaixes diferentes na placa da Figura 3.3. As barras de pinos à esquerda são para encaixar o Arduino, as barras de furo no centro são para encaixar os *drivers* Pololu A4988, e os bornes à direita são para alimentar as bobinas dos motores de passo.

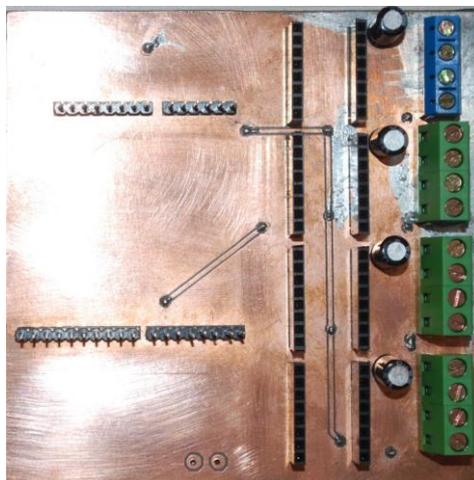


Figura 3.3: Placa de interface entre controlador e drivers.

### 3.3 Parte mecânica

A fim de se ter um ponto de partida para a construção da estrutura mecânica deste projeto, foi feito em Autocad, um desenho que contivesse as características desejadas. Estas características foram de, poder fabricar placas com dimensões de até 250 x 250 milímetros ocupando o menor espaço possível, economizando material. Outra característica importante foi que o deslocamento ao longo do eixo Z, poderia ser pequeno, já que a função de aproximar e retirar as ferramentas não exigia grandes cursos. Após alguns desenhos e modificações, o projeto que melhor atendeu aos requerimentos foi o mostrado na Figura 3.4.

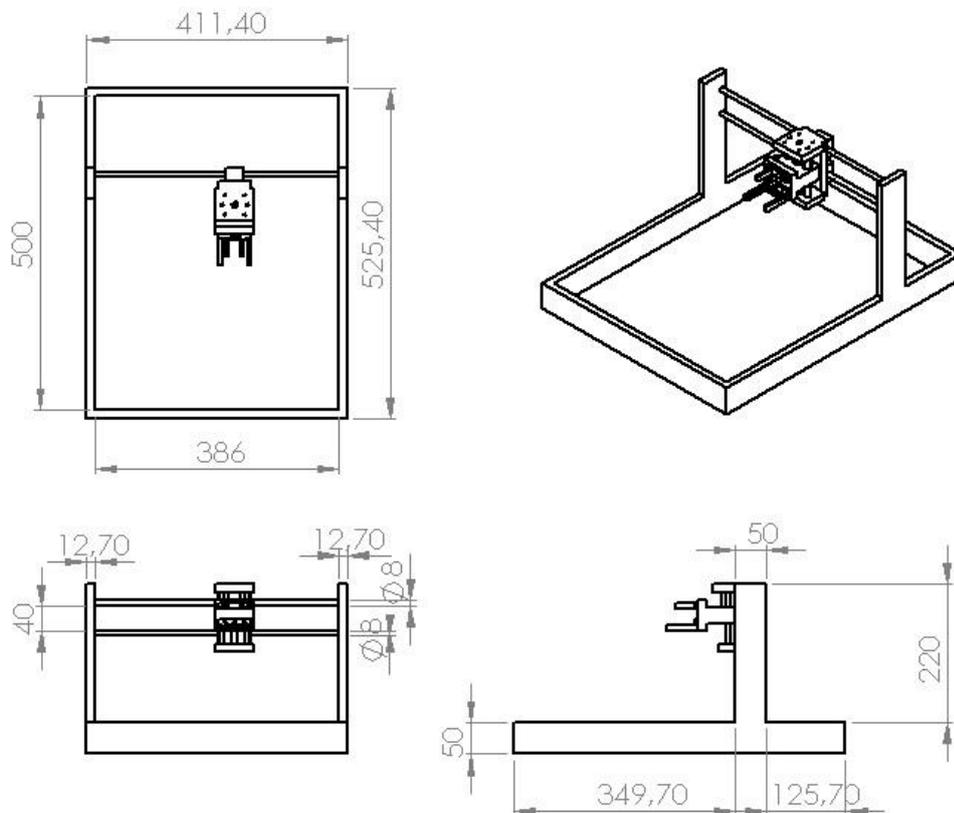


Figura 3.4: Projeto da estrutura mecânica.

A Figura 3.5 mostra uma estrutura em alumínio utilizada como base para o projeto e possui as mesmas características presentes no projeto mostrado na Figura 3.4. As barras horizontais e verticais que se encaixam, foram usinadas em sua área de contato para que uma peça adentrasse a outra e ficassem sempre alinhadas.

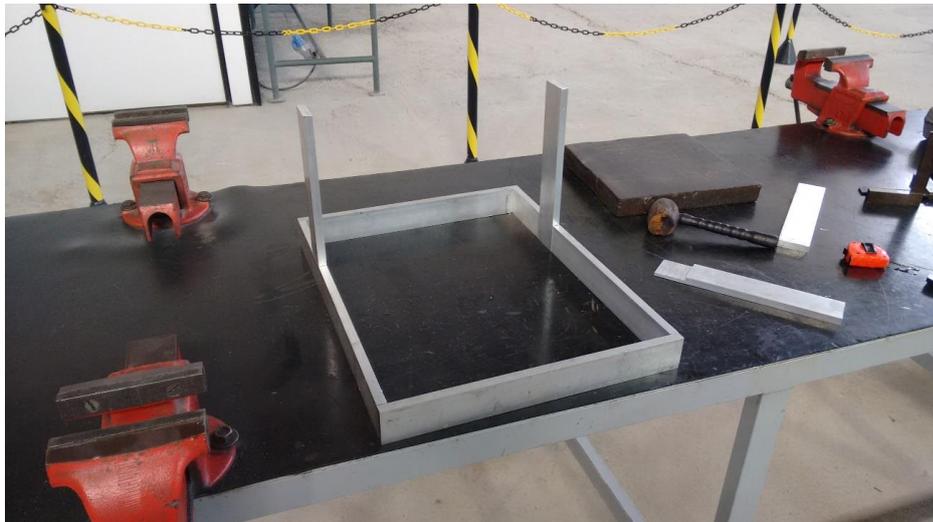


Figura 3.5: Estrutura em alumínio.

Por questões de disponibilidade de uso, a usinagem do alumínio ocorreu em duas etapas. Na primeira, foi utilizada uma fresadora CNC Denford, mostrada na Figura 3.6, localizada no laboratório de CNC sala 311. Nesta etapa, foi feito o fresamento das barras verticais, pois estas possuíam dimensões menores (290mm), compatíveis com esta máquina. Foi utilizado um código G, de linguagem Fanuc (DENFORDATA, 2015), para o desbaste por camadas até atingir as dimensões exigidas.



Figura 3.6: Fresadora CNC Denford.

Na segunda etapa, foi utilizada uma fresadora manual Diplomat presente no galpão de mecânica, aqui mostrada na Figura 3.7. Esta é uma máquina mais robusta e possui a capacidade de usinar peças de maiores dimensões. Por este motivo, algumas barras horizontais, que possuíam 500mm de comprimento, foram trabalhadas nesta máquina. As duas barras menores (411,4mm) não necessitam serem usinadas.



Figura 3.7: Fresadora manual Diplomat, localizada no galpão de mecânica.

A Figura 3.8 mostra o desenho da peça utilizada para suportar as barras guia que permitem a locomoção de uma mesa ao longo do eixo X. Foi fabricada em uma impressora 3D a partir de um filamento de ABS e possui quatro furos. O maior furo serve para o encaixe da barra guia, e os três restantes para aparafusá-la à estrutura de alumínio. Quatro deste mesmo modelo foram produzidos, duas para cada uma das barras e devem ser acopladas na guia como na Figura 3.9.

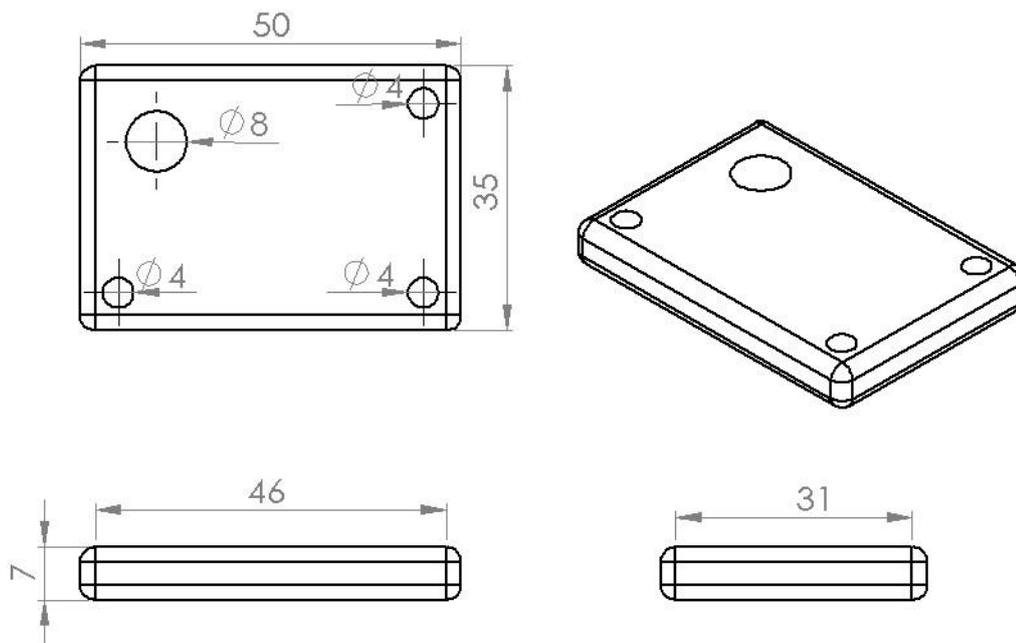


Figura 3.8: Modelo 3D para suporte das barras guias do eixo X.

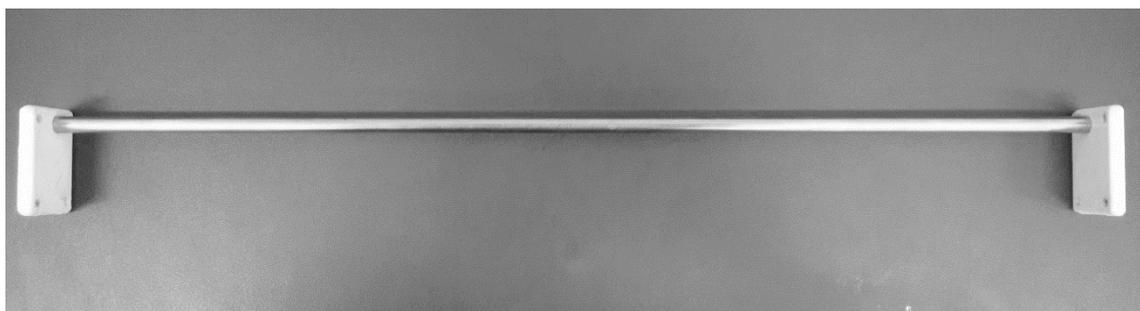


Figura 3.9: Encaixe da peça de suporte à barra guia do eixo X.

A mesa móvel que percorre o eixo X foi feita inicialmente, de MDF, por questões de redução de peso e custos, além de atender plenamente às necessidades exigidas. Esta mesa possui dimensões de 380 x 250mm e, para que pudesse se deslocar, foram aparafusados quatro *pillow blocks* em cada uma de suas extremidades, como mostrado na Figura 3.10. Os *pillow blocks* são estruturas metálicas com um rolamento linear em seu interior e servem para acoplar a mesa as barras guia, e permite deslize na direção X. As barras guia, com dimensões de 500mm de comprimento e 8mm de diâmetro, traspassam os rolamentos lineares internos aos *pillow blocks* e as peças de suporte são encaixadas em suas extremidades, seguindo o mesmo raciocínio mostrado na Figura 3.9.

A estrutura da mesa foi encaixada a estrutura base como mostrado na Figura 3.10. Pode-se notar que o sistema de tração da mesa, composta por quatro partes: Motor de passo, correia dentada, tensionador de correia e acoplamento de

correia, respectivamente. O motor de passo se localiza a direita e é responsável por gerar o torque, enquanto a correia transfere este torque para a mesa.

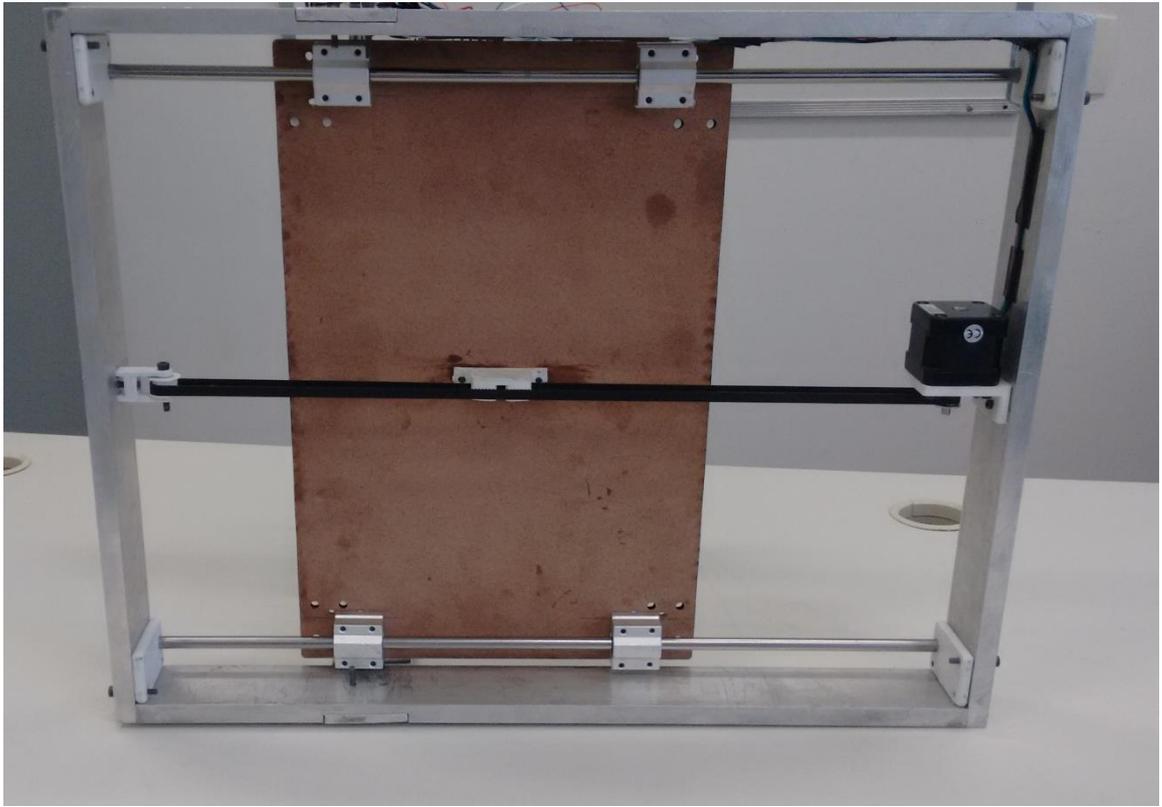


Figura 3.10: Vista da parte inferior da impressora.

Já o tensionador de correia, à esquerda (Veja detalhe na Figura 3.11(a)), possui uma polia por onde passa a correia, mantendo-a esticada, além de possuir um parafuso por onde se ajustar a tensão aplicada nesta correia (Figura 3.11(b)). Por último, o acoplamento de correia (Figura 3.11), localizado junto à mesa, serve para unir as duas extremidades da correia e fixá-las a mesa, gerando o movimento linear no eixo X.

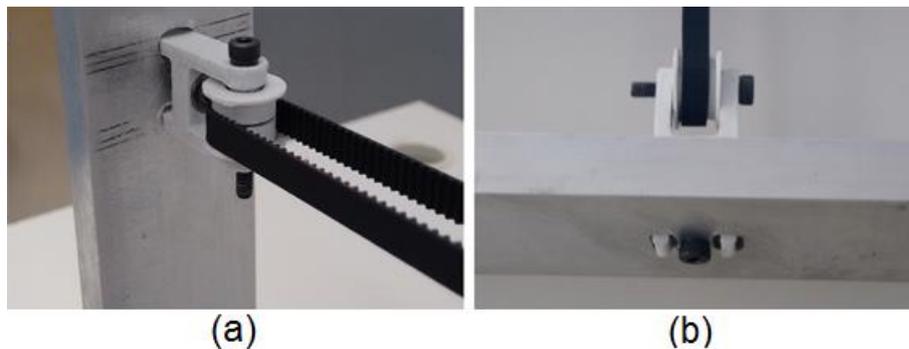


Figura 3.11(a): Tensionador de correia e sua polia. Figura 3.11(b): Parafuso para ajuste de tensão da correia.



Figura 3.12: Acoplamento de correia.

A estrutura para movimentação no eixo Y, mostradas na Figura 3.13, é composta por duas barras guias de 400mm de comprimento acopladas às barras de alumínio verticais dispostas. A distância entre as barras guia é de 48mm. A necessidade de se utilizar duas barras é permitir que o carro possua um movimento linear ao longo do eixo Y sem que este carro balance como um pêndulo. Por não haver *pillow block*, o carro móvel do eixo Y foi conectado diretamente a três rolamentos lineares, como mostrado na Figura 3.13. A estrutura do carro foi feita em uma impressora 3D para simplificar o processo de fabricação e reduzir o peso, já que se esta peça fosse feita em metal, as barras guia do eixo Y sofreriam flambagem, retirando a precisão da mesa. A maneira com que foram dispostos os três rolamentos lineares, além de permitir o deslocamento linear do carro com maior suavidade, evita a ação de folgas que seriam presentes caso tivesse sido utilizado apenas dois rolamentos. Semelhante ao do eixo X o eixo Y possui quatro partes: Motor de passo, correia dentada, tensionador de correia e acoplamento de correia.

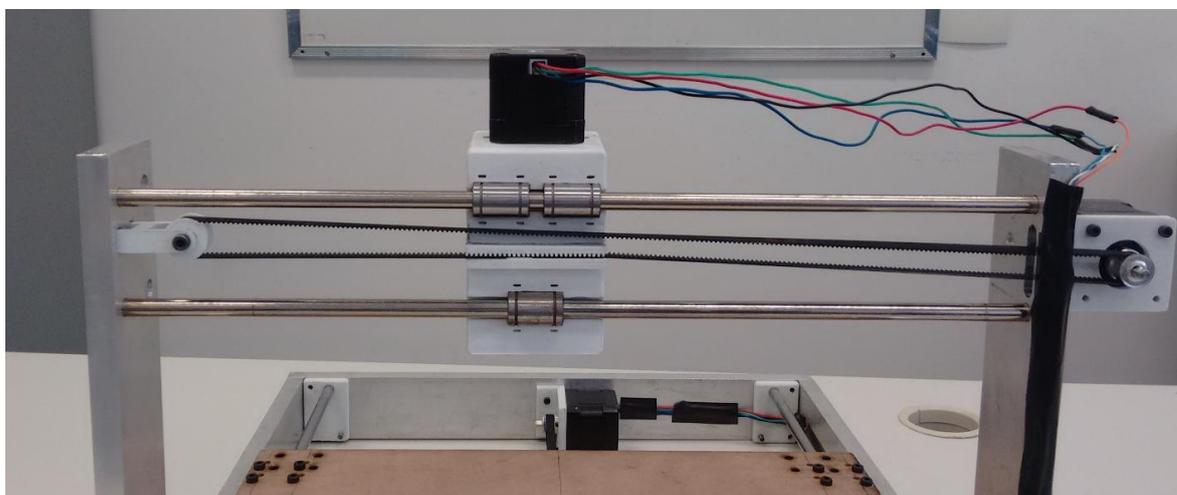
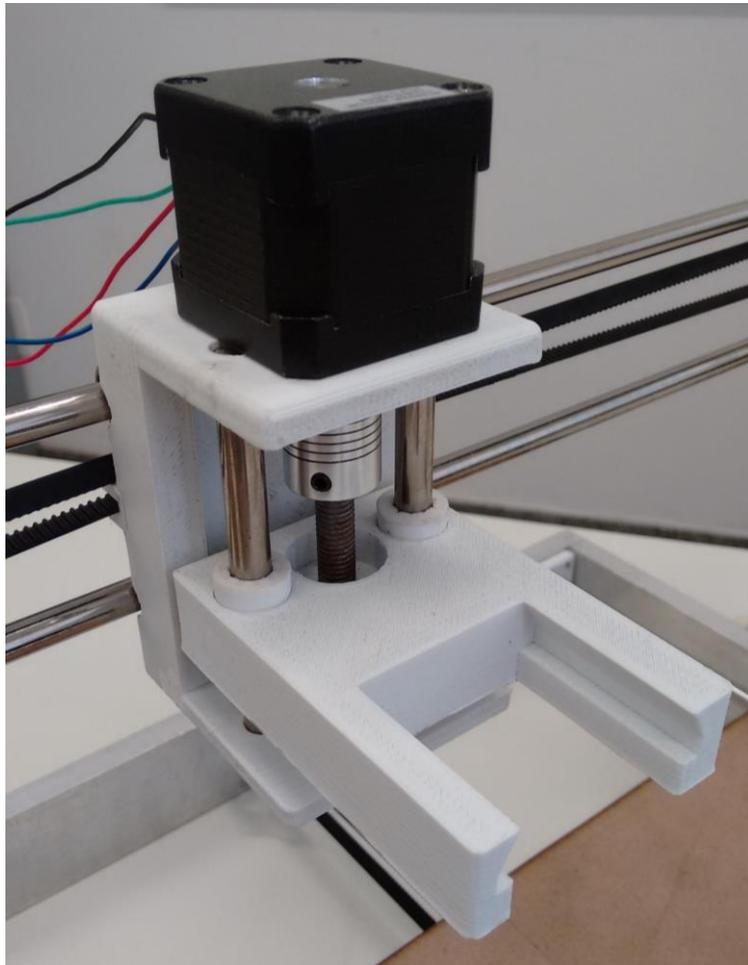


Figura 3.13: Sistema para movimentação do carro ao longo do eixo Y.

O carro que se desloca ao longo de Y, possui um suporte para um motor de passo, para a ferramenta (caneta ou furador de placa) e para o sistema de deslocamento vertical (Eixo Z) do mesmo como visto na Figura 3.14.



*Figura 3.14: Carro que se locomove ao longo do eixo Y e suporte para ferramentas que desloca ao longo do eixo Z.*

O deslocamento ao longo do eixo Z se dá através de um eixo roscado acoplada ao motor de passo. Comparado ao sistema de correia, o sistema de barra roscada possui um movimento mais lento, porém com precisão e torque maiores.

Na Figura 3.14 é mostrado suporte para o eixo Z. Como pode ser observado, há duas barras guias paralelas a barra roscada, cujo eixo é conectado ao motor de passo. Estas estruturas são responsáveis por segurar a ferramenta de pintura e furação. Foram projetadas para encaixar um suporte que sustenta as ferramentas, de maneira que seja simples a troca e que ao mesmo tempo não permita que estas se soltem ou balancem no decorrer do processo de fabricação de PCBs.

Por fim, a Figura 3.15 mostra uma vista da estrutura mecânica completa construída, com os motores de passo acoplados devidamente para tração de

seus respectivos eixos. Em resumo, o eixo X é considerado o responsável pelo deslocamento da mesa em MDF. O eixo Y se refere ao deslocamento do carro que transporta as ferramentas. Já o eixo Z, localizado no carro de transporte de ferramentas é o responsável pelo movimento vertical das mesmas.

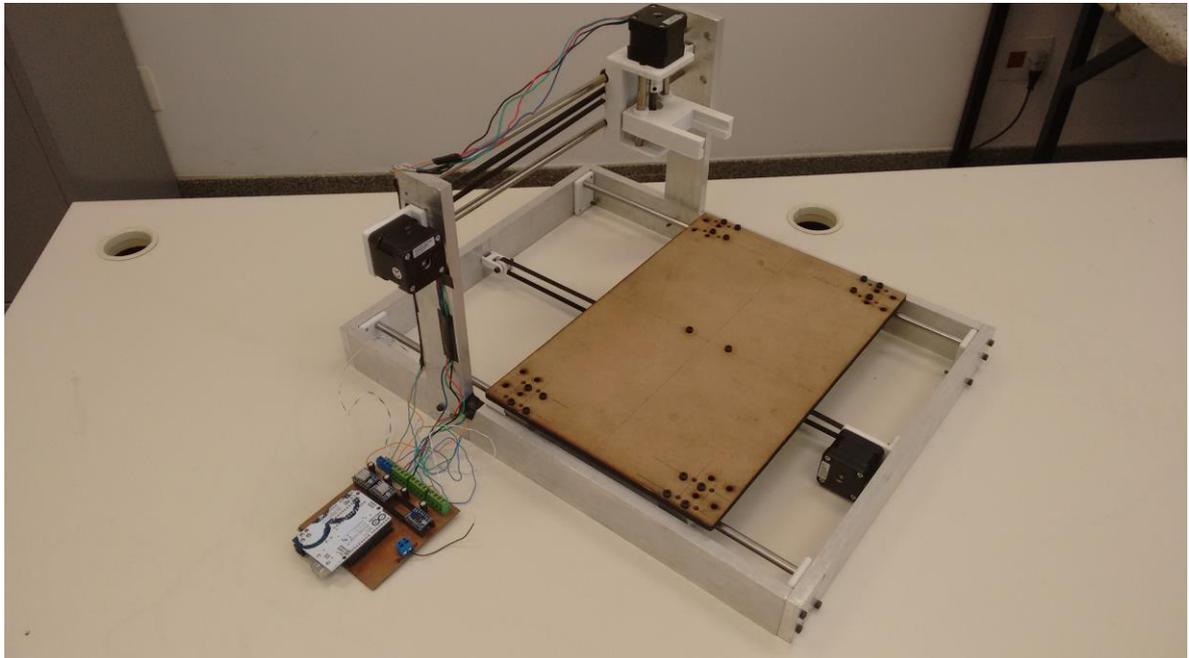


Figura 3.15: Vista geral da estrutura mecânica construída e ligações elétricas.

### 3.4 Comunicação computador-máquina

A comunicação computador-máquina é dada através de uma placa Arduino que possui um *firmware* em sua memória. Segundo Psafe (2015):

“*Firmware* é o conjunto de instruções programadas diretamente no *hardware* de um equipamento eletrônico. É ele que contém as informações de inicialização que permitem o correto funcionamento do aparelho.”

Ou seja, para que os motores de passo recebam os comando de maneira que este possa funcionar corretamente, um *firmware* apropriado deve estar programado do equipamento eletrônico, neste caso, o Arduino. O *firmware* utilizado para o funcionamento correto do motor de passo possui o nome `grbl_v0_8c_atmega328p_16mhz_9600` (GITHUB, 2015) por ser compatível com a plataforma Arduino e ser *open source*. Sua extensão é `.hex` e para gravá-lo na memória do dispositivo o *software Xloader* (RUSSEMOTTO 2015), mostrado na Figura 3.16, foi utilizado.

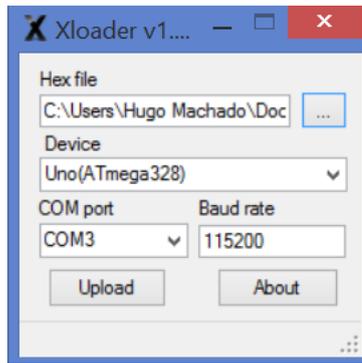


Figura 3.16: Interface do programa Xloader.

Para o funcionamento correto do programa mostrado na Figura 3.16 foi preciso seguir alguns passos. O primeiro passo foi conectar a placa Arduino em qualquer uma das portas USB do computador, em seguida abriu-se o programa *Xloader* e, clicando na janela com três pontos escritos, selecionou o arquivo .hex supracitado. Na janela abaixo, a plataforma utilizada foi selecionada, neste caso um Arduino Uno, com um microcontrolador Atmega 328. Em seguida, selecionou-se a porta USB que está comunicando com o Arduino, geralmente aparece somente uma opção para se selecionar, portanto não será necessário ter que descobrir qual porta foi utilizada. Por último, clicou-se na janela de *upload* e esperou com que o arquivo fosse devidamente carregado na placa. Após este processo, o uso do *Xloader* não é mais requerido.

A interface homem-computador é dado pelo programa *Universal Gcode Sender* (UGCS, 2015), mostrado na Figura 3.17.

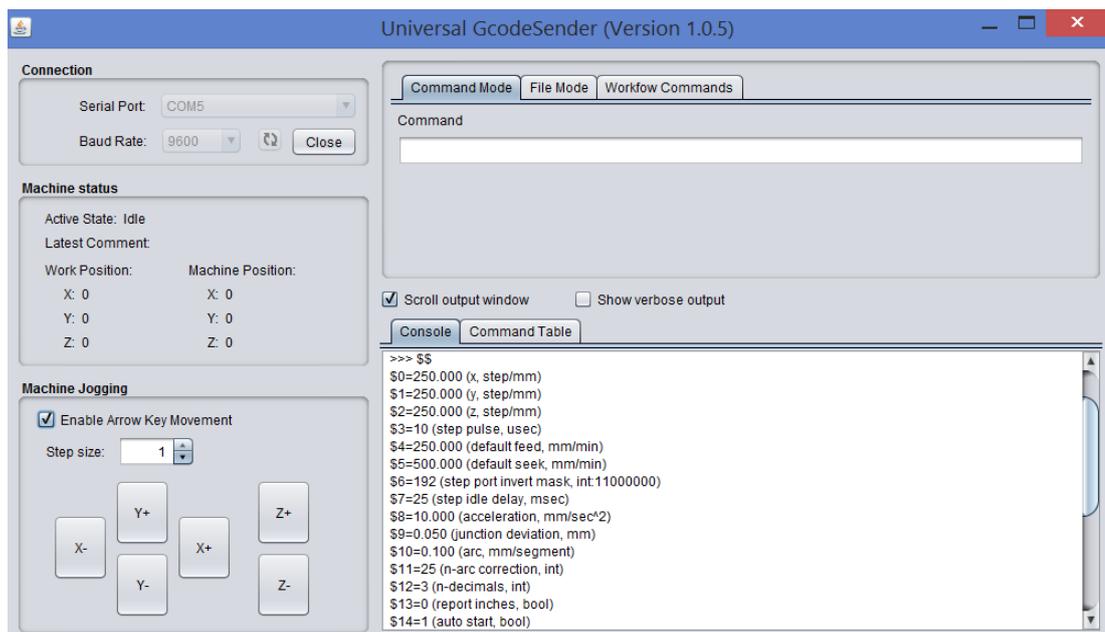


Figura 3.17: Interface do programa Universal Gcode Sender.

Com o *Universal Gcode Sender* (Figura 3.17) já é possível realizar comandos de movimentos na máquina, contudo, para que a máquina possa realizar procedimentos referentes a impressão e furação de uma PCI, são necessários alguns passos para se conseguir o código G utilizado no programa. Estes passos são explicados detalhadamente no Apêndice 3.

## 4- RESULTADOS E DISCUSSÕES

Ao terminar a construção da parte mecânica, algumas medidas foram tiradas, a fim de conferir se a estrutura estava alinhada. Este processo consistiu em medir as diagonais e a distância entre as barras de 500 milímetros em ambas as extremidades da estrutura retangular utilizada como base (Figura 3.15). Os resultados das medições podem ser conferidos na Tabela 4.1.

Tabela 4.1: Medidas para conferir alinhamento da estrutura mecânica.

Local medido	Medição (mm)
Diagonal 1	666±1
Diagonal 2	666±1
Extremidade 1	411±1
Extremidade 2	411±1

As medidas foram tiradas utilizando uma trena, que possui uma precisão de 1 milímetro. Dentro desta precisão, foi medido o mesmo valor nas diagonais, o que representa que a estrutura está no esquadro. O mesmo ocorreu com a medição das duas extremidades, significando que as barras estão devidamente paralelas. Isto resulta em um deslocamento linear e suave do carro do eixo X e garante a perpendicularidade deste com o eixo Y.

Com a estrutura devidamente montada como mostrado na Figura 3.15 e conectada a um computador, a partir do *software Universal Gcode Sender* (Figura 3.17), alguns testes de funcionamento foram realizados. O primeiro teste realizado foi o de deslocamento máximo de cada um dos eixos e ocorreu da seguinte forma: Encostou-se as partes móveis de cada um dos eixos em uma extremidade e, a partir do *Universal Gcode Sender*, foi-se incrementando o deslocamento de cada um dos eixos até que as partes esbarrassem na outra extremidade. A máxima excursão de cada eixo é mostrada na Tabela 4.2.

Tabela 4.2: Máxima excursão de cada eixo.

Eixo	Máxima excursão (mm)
X	285±1
Y	280±1
Z	35±1

Como esperado, a excursão dos eixos X e Y são superiores a 250mm, o que possibilita a fabricação de placas com as dimensões máximas especificadas na seção 3.3. Em relação ao Eixo Z, sua excursão de 35mm é o suficiente para que as ferramentas atuem corretamente no processo de pintura e furação da placa de cobre.

Um último teste foi realizado a fim de conferir o funcionamento correto da máquina. Este se resumia em enviar comandos de posicionamento de cada um dos eixos através do *Universal Gcode Sender* e, utilizando um paquímetro, que possui uma precisão de 0,05 milímetro, medir o deslocamento e analisar se confere com o valor enviado. Três diferentes posições foram enviadas para cada um dos eixos. Os resultados podem ser observados na Tabela 4.3.

Tabela 4.3: Teste de deslocamento dos eixos coordenados.

Eixo	Comando (mm)	Valor medido (mm)
X	10.00±0.05	10.00±0.05
	15.00±0.05	15.00±0.05
	20.00±0.05	20.00±0.05
Y	10.00±0.05	10.00±0.05
	15.00±0.05	15.00±0.05
	20.00±0.05	20.00±0.05
Z	10.00±0.05	10.00±0.05
	15.00±0.05	15.00±0.05
	20.00±0.05	20.00±0.05

Observando a Tabela 4.3, cada um dos três eixos respondeu corretamente ao comando de deslocamento requerido. Com isto é possível notar que o sistema não possui folgas nem falhas elétricas que afete a precisão da impressora.

A parte de suporte de ferramentas e de placa ainda está em processo de desenvolvimento, para que até o dia da apresentação deste trabalho de conclusão de curso, a máquina esteja produzindo corretamente as placas de circuito impresso. Outra parte que ainda se encontra em fase de melhoria é o de conversão de imagem para código G e, para isto, estuda-se a substituição do *software Proteus* para o *software Eagle*.

## 5- CONCLUSÕES

A partir de todos os experimentos realizados no decorrer deste trabalho de conclusão de curso, nota-se as seguintes conclusões:

- 1) A estrutura física foi construída de forma bem alinhada, para que todo o sistema se deslocasse de maneira fluida;
- 2) Tanto o sistema de polia e correia, quanto o de barra de fuso funcionaram em perfeitas condições;
- 3) A parte mecânica não possui folga ou desbalanceamento que afete a precisão dos movimentos da máquina;
- 4) A placa de controle conseguiu executar os comandos para controlar os motores de passo sem qualquer interferência que prejudicasse seu desempenho;
- 5) O *firmware* instalado no Arduino e o *Universal Gcode Sender* conseguiram desempenhar suas funções de capturar e transformar o código em movimento para os motores de passo;
- 6) Os comandos enviados pelo usuário foram reproduzidos de forma fiel em forma de movimento pela máquina;
- 7) O método utilizado da geração de imagem até a transformação de código G ainda não é intuitivo.

## 6- PROPOSTAS PARA TRABALHOS FUTUROS

Os tópicos a seguir, são sugestões de possíveis melhorias para este projeto em questão:

- 1) Instalação de sensoriamento de fim de curso, para evitar que comandos de movimento que ultrapassem a excursão da mesa resulte em avarias no projeto;
- 2) Criação de um *software* que seja mais intuitiva na transformação do *layout* em comandos numéricos para a máquina;
- 3) Substituição da caneta como ferramenta de impressão por um outro sistema melhor de ejeção de tinta;
- 4) Desenvolvimento de um sistema de cabeamento para alimentação dos motores que fique mais protegido e visualmente agradável.

## 7- REFERÊNCIAS BIBLIOGRÁFICAS

AKIYAMA. **Treinamento de motores de passo, *drivers* e introdução a servo acionamentos.** Disponível em [file:///C:/Users/Hugo%20Machado/Downloads/Apostila\\_Treinamento%20Motores%20de%20Passo\\_Drivers\\_Rev01.pdf](file:///C:/Users/Hugo%20Machado/Downloads/Apostila_Treinamento%20Motores%20de%20Passo_Drivers_Rev01.pdf)  
Acesso em :08/06/2015.

ALLEGRO. ***DMOS Microstepping Driver with Translator And Overcurrent Protection.*** Disponível para download em [https://www.pololu.com/file/download/A4988.pdf?file\\_id=0J450](https://www.pololu.com/file/download/A4988.pdf?file_id=0J450) >  
Acesso em: 26/02/2015.

CEEMS. **Atrito estático e cinético.** Disponível em < [http://profs.ccems.pt/PauloPortugal/CFQ/Atrito\\_histria\\_cincia/Atrito\\_hist\\_cincia.html](http://profs.ccems.pt/PauloPortugal/CFQ/Atrito_histria_cincia/Atrito_hist_cincia.html) >  
Acesso em: 08/06/2015.

CPRIUS. ***CNC isolation routing PCB through hole plating.*** Disponível em <https://www.youtube.com/watch?v=yqjKohZX2t8> >  
Acesso em: 07/03/2015.

DAMENCNC. ***Stepper motors.*** Disponível em < <https://www.damencnc.com/en/components/motors-and-drivers/stepper-motors> >  
Acesso em: 08/06/2015.

DENFORDATA. **Fanuc OT (*Offline turning*) *Programming manual.*** Disponível para download em < <http://www.denfordata.com/bb/download/file.php?id=1235> >  
Acesso em: 02/05/2015.

DMT. ***Bipolar stepper motor control with Picaxe and L293D chips.*** Disponível em <<https://dmt195.wordpress.com/2009/01/28/bipolar-stepper-motor-control-with-picaxe-and-l293d-chips/>>  
Acesso em: 08/06/2015.

EHOW. **Problemas comuns de integridade de sinal em uma PCI.** Disponível em <[http://www.ehow.com.br/problemas-comuns-integridade-sinal-pci-info\\_112561/](http://www.ehow.com.br/problemas-comuns-integridade-sinal-pci-info_112561/)>

Acesso em: 6/03/2015.

GITHUB. **GRBL.** Disponível para download em <<https://github.com/grbl/grbl/downloads>>

Acesso em: 07/05/2015.

HANDMADES. **Lâmpada U.V (Luz negra) – cuidados.** Disponível em <<http://www.handmades.com.br/forum/index.php?topic=4411.0>>

Acesso em: 09/03/2015.

HOBBY, Eletronik. **PCB Laserprinter.** Disponível em <<https://www.youtube.com/watch?v=4SNkzoOvoD8>>

Acesso em: 03/03/2015.

HOLLOWSTATE. **Como fazer uma placa de circuito impresso em casa (PCB por transferência de toner).** Disponível em <<http://hollowstate.blogspot.com.br/2015/02/como-fazer-uma-placa-de-circuito.html>>

Acesso em: 09/03/2015.

LPKF. **LPKF Laser & Eletronics.** Disponível em <<http://www.lpkf.com/products/rapid-pcb-prototyping/laser-circuit-structuring/laser-structuring-printed-circuit-boards.htm>>

Acesso em: 07/03/2015

PAUL, Clayton R. **Introduction to Electromagnetic Compatibility.** 2ed. Estados Unidos: *Jonh Wiley and Sons*: 2006.

PBCLINEAR. **Stepper motor Nema 17.** Disponível em <<http://www.pbclinear.com/Download/DataSheet/Stepper-Motor-Support-Documents.pdf>> Acesso em: 05/04/2015.

POLOLU. **A4988 stepper driver carrier.** Disponível em  
<<https://www.pololu.com/product/1182> >  
Acesso em: 23/02/2015.

PROTONEER. **Arduino CNC Shield – 100% GRBL compatible.** Disponível em  
<<http://blog.protoneer.co.nz/arduino-cnc-shield/> >  
Acesso em: 23/02/2015.

PSAFE. **Saiba o que é firmware e como ele pode ser infectado por vírus.**  
Disponível em < <http://www.psafe.com/blog/firmware-como-pode-infectar-virus/> >  
Acesso em: Acesso em 07/06/2015.

RUSSEMOTTO. **Xloader.**  
Disponível em <<http://xloader.russemotto.com/>>  
Acesso em 23/02/2015

SIGNIFICADOS. **Significado de software.** Disponível em  
<<http://www.significados.com.br/software/>>  
Acesso em: 09/06/2015

## APÊNDICE 1

Para confirmar que o torque de 4 N.m do motor de passo Nema 17 suporta a carga da impressora, os seguintes cálculos foram realizados (AKIYAMA, 2015):

Passo 1- Calcular a inércia a partir das fórmulas:

$$J_L = \frac{1}{4} m_L D^2 \quad (\text{A.1})$$

$$J_P = \frac{1}{8} m_P D^2 \cdot 2 \quad (\text{A.2})$$

$$J_B = \frac{1}{4} m_B D^2 \quad (\text{A.3})$$

$$J_T = J_L + J_P + J_B + J_M \quad (\text{A.4})$$

Passo 2- Calcular o torque:

$$T_a = J_T \cdot a \quad (\text{A.5})$$

$$T_L = \frac{m_L \cdot g \cdot D (\sin \alpha + \mu \cos \alpha)}{2\eta} \quad (\text{A.6})$$

$$T_T = T_L + T_a \quad (\text{A.7})$$

$$J_M = K_S + T_T \quad (\text{A.8})$$

São dados os seguintes valores para as variáveis:

$$m_L = 1,5 \text{ Kg}$$

$$m_P = 0,02 \text{ Kg}$$

$$m_B = 0,05 \text{ Kg}$$

$$J_M = 5,7 \cdot 10^{-6} \text{ (DAMENCNC, 2015)}$$

$$D = 12 \cdot 10^{-3}$$

$$\mu = 0,7 \text{ (CCEMS, 2015)}$$

$$\alpha = 0^\circ$$

$$K_S = 2$$

Ao substituir todos os valores nas equações acima, encontra-se o seguinte torque para o motor:

$$T_M = 139,294.10^{-3} \text{ N.m}$$

## Apêndice 2

Para a realização deste projeto, uma gama de materiais e ferramentas foram compradas. A Tabela A2.1 mostra a lista de materiais comprados e gasto total:

Tabela A2.1: Lista de gastos com materiais e ferramentas.

<b>Produto</b>	<b>Quantidade</b>	<b>Unidade</b>	<b>Valor</b>
Acoplador para motor de passo	4	Uni	R\$ 76,00
Arduino	2	Uni	R\$ 40,00
Barra Alumínio	1	Uni	R\$ 220,00
Barra Roscada	2	m	R\$ 8,80
Broca + serra	1	Uni	R\$ 21,00
Driver Motor de Passo	10	Uni	R\$ 31,17
Driver Motor de Passo	2	Uni	R\$ 32,00
Ferramentas de rosca	-	-	R\$ 94,73
Ferramentas de rosca	1	Uni	R\$ 35,00
Motor de passo	5	Uni	R\$ 175,50
Motor para furação e ferramenta	1	Uni	R\$ 13,55
Parafusos e porcas	-	Uni	R\$ 10,00
Pillow Block	4	Uni	R\$ 82,00
Prancha madeira	1	Uni	R\$ 10,00
Filamento para impressora 3D	1	Kg	R\$ 123,00
Kit correia e polias	1	Uni	R\$ 20,00
Equipamentos eletrônicos	1	Uni	R\$ 42,00
Pincel de transpaência especial	1	Uni	R\$ 7,00
<b>Total Gasto</b>			<b>R\$ 1.041,75</b>

## Apêndice 3

Para que se possa ter acesso às funções do *Universal Gcode Sender*, primeiro é preciso conectar o Arduino ao computador, como no *Xloader*, e clicar na janela *open*. Ao digitar a instrução '\$\$' na janela *Command* do programa, uma gama de opções de configurações aparecem na janela *Console*, como na Figura 3.17. Para que os motores de passo gerem o mesmo deslocamento que o usuário requereu no comando, é preciso que as configurações '\$0', '\$1' e '\$2' estejam com as mesmas relações de passo/mm calculada na seção 2.2. Como os comandos '\$0' e '\$1' são responsáveis pelas relações dos eixos X e Y, respectivamente, e ambos possuem o mesmo sistema de tração utilizando polias e correias GT2 com micropasso de  $\frac{1}{16}$ , a relação é de 80 passos/mm. Para colocar esta informação no programa basta digitar as seguintes linhas de comando, uma de cada vez, na janela *command*:

- \$0=80
- \$1=80

Já o comando '\$2' é responsável pela relação do eixo Z, que por utilizar um sistema de tração por barra roscada M8, possui uma relação de redução de 2560 ao utilizar o micropasso de  $\frac{1}{4}$ . A linha de comando é a seguinte:

- \$2=640

Vale a pena ressaltar que, para que o comando realmente seja informado para o programa é preciso precionar a tecla *enter* do computador. Com estas tres configurações já é possível a movimentação correta dos motores, mas, para um desempenho diferente do padrão, pode-se alterar a taxa de alimentação (Configuração o'\$5') e a aceleração (Configuraçã o'\$8')

Os comandos necessários para o deslocamento dos eixos se resumem a G0 e G1 (DENFOR DATA, 2015). Com estes dois comandos já é possível a realização de avanço rápido e lento, respectivamente, das partes móveis de cada um dos eixos

da mesa. Por exemplo se quiser criar um movimento rápido de 15mm no eixo X, 19mm no eixo Y e -3mm no eixo Z, basta digitar na janela *command* a instrução:

- G0 X15 Y19 Z-3

Já para o mesmo movimento, porém com velocidade reduzida, a instrução fica:

- G1 X15 Y19 Z-3

Com estas informações já é possível a criação de programas que realizem alguma tarefa requerida pelo usuário, contudo, seria inviável a programação manual de um código que realizasse a impressão de uma PCI. Por isto, um método de conseguir o código G pronto para gerar o *layout* sobre a placa de cobre foi desenvolvido. Utilizando o Proteus, cria-se o *layout* desejado para uma PCI (foi utilizado o *layout* mostrado na Figura 3.1). Em seguida, na opção *output*, localizada na parte superior esquerda da janela do programa, seleciona-se a opção 'Generate Gerber/Excellon Files'. Uma janela se abre, como mostra a Figura 30. As opções para a geração de código G para as duas camadas da placa e suas furações são marcadas e salvas, ainda de acordo com a Figura A3.1.

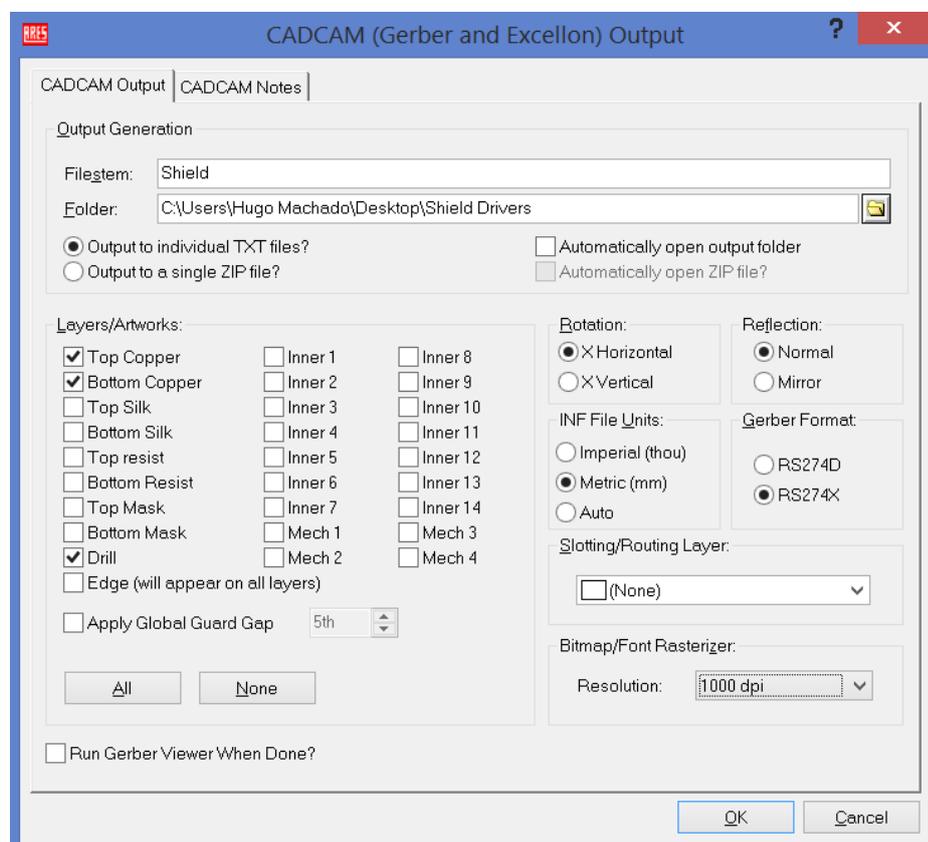


Figura A 3.1: Janela de geração de código G do Proteus.

Em seguida, utilizando o programa *FlatCam* (FLATCAM, 2015), mostrado na Figura A3.2, importa-se os arquivos gerados no Proteus. Para fim de ilustração, apenas o arquivo de furação foi importado. Indo na opção *File*, na extremidade superior esquerda do programa, selecionou-se a opção '*Open Ecellon*', abra o arquivo gerado que possua o escrito '*Drill*' ao fim de seu nome. A imagem de furação é gerada na área de trabalho do *FlatCam* (BQ DIY, 2015), como mostra a Figura 3.19.

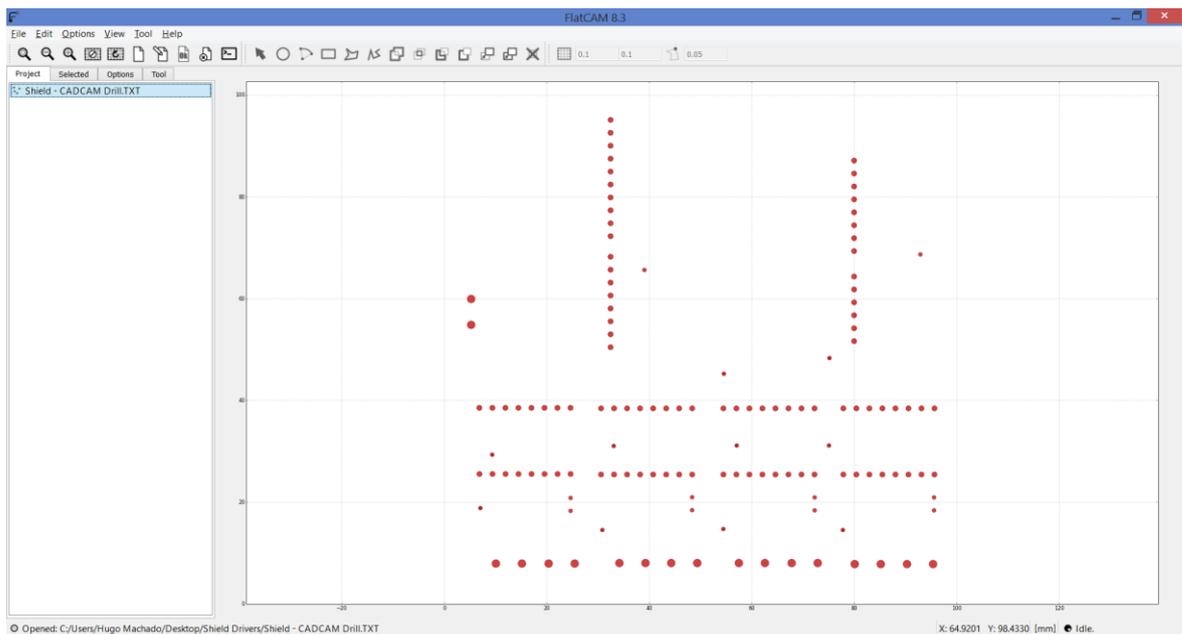


Figura A3.2: Interface do FlatCam e mapa de furação da placa.

Ao dar duplo clique sobre o arquivo que gerado na área de trabalho da esquerda, há o redirecionamento para uma segunda área de trabalho, na qual é possível escolher a penetração da ferramenta na placa, espessura da ferramenta, taxa de alimentação e *offset* da imagem. A configuração utilizada para este exemplo é mostrada na Figura A3.3(a). Já em *Options*, Figura A3.3(b), é possível escolher a opção de escolher a unidade de medida. Neste caso, foi utilizado em milímetros.

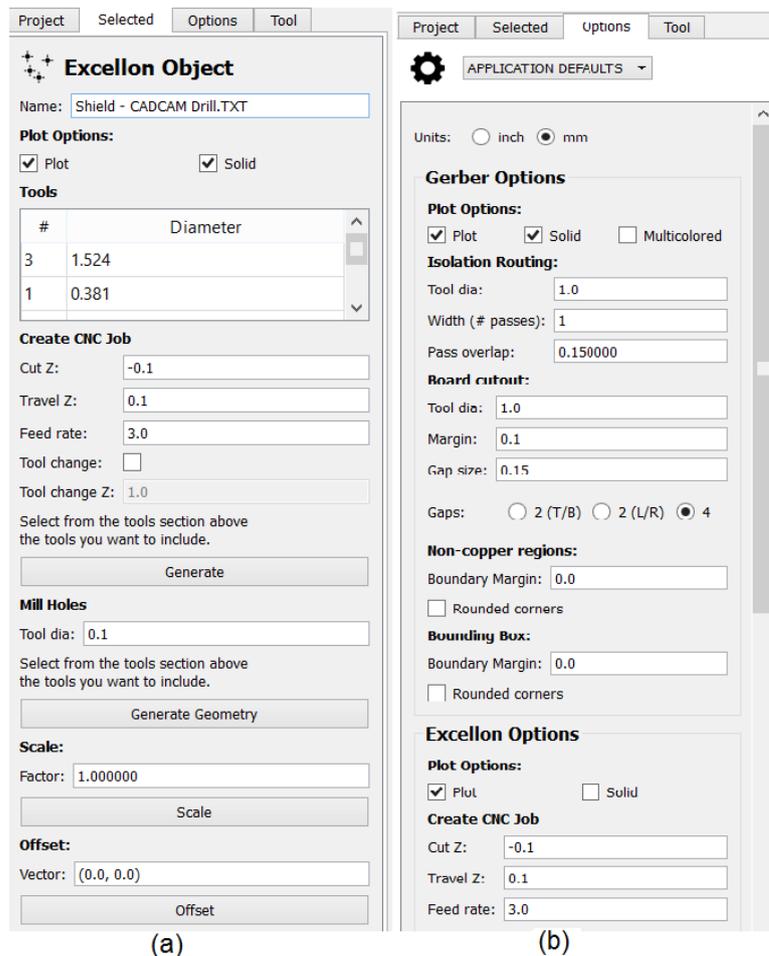


Figura A3.3(a): Janela Selected e A3.3(b) janela Options.

Voltando na janela *selected* da Figura A3.3(a), clica-se na opção *Generate* e em seguida *Generate Geometry*. Um segundo arquivo aparece na janela *Project*, basta abrí-lo com duplo clique e, na janela que abrir, clicar na opção *Export G-Code* (BQ DIY, 2015).

Por fim, abrir o arquivo gerado em extensão *.txt* e importá-la para o *Universal Gcode Sender*, a partir da janela *File Mode*, presente na Figura 3.17, e executar o processo.

## 9- GLOSSÁRIO

**Interface** - Conexão entre dois dispositivos em um sistema de computação. Também usado para definir o modo de comunicação entre o computador e o usuário.

**Layout** - Traçado. Resultado da criação de trilhas para a intercomunicação de componentes eletrônicos em projetos de PCI.

**Percloroato de Ferro** - Meio ácido utilizado para a retirada da camada de cobre indesejada de uma PCI.

**Ripple** – Ondulação. É a componente de corrente alternada (Vca) que se sobrepõe ao valor médio da tensão de uma fonte de corrente contínua.

**Software** – “É uma sequência de instruções escritas para serem interpretadas por um computador com o objetivo de executar tarefas específicas. Também pode ser definido como os programas que comandam o funcionamento de um computador” (SIGNIFICADOS, 2015).