



MEC-SETEC
INSTITUTO FEDERAL MINAS GERAIS – *Campus Formiga*
Curso de Engenharia Elétrica

DESENVOLVIMENTO DE UMA MÁQUINA CNC DE BAIXO CUSTO COM *SOFTWARE* E *HARDWARE* ABERTOS

Fernando Henrique Polastrini

Orientador: Prof. Dr. Niltom Vieira Junior

FORMIGA – MG
2016

FERNANDO HENRIQUE POLASTRINI

**DESENVOLVIMENTO DE UMA MÁQUINA CNC DE BAIXO
CUSTO COM *SOFTWARE* E *HARDWARE* ABERTOS**

Trabalho de Conclusão de Curso apresentado ao Instituto Federal de Minas Gerais – *Campus* Formiga, como requisito parcial para a obtenção do título de Bacharel em Engenharia elétrica.

Orientador: Prof. Dr. Niltom Vieira Junior

**FORMIGA – MG
2016**

FERNANDO HENRIQUE POLASTRINI

**DESENVOLVIMENTO DE UMA MÁQUINA CNC DE BAIXO CUSTO
COM *SOFTWARE* E *HARDWARE* ABERTOS**

Trabalho de Conclusão de Curso apresentado ao
Curso de Engenharia Elétrica do Instituto Federal
de Minas Gerais como requisito para obtenção do
título de Bacharel em Engenharia Elétrica.

Avaliado em: _____ de _____ de _____.

Nota: _____

BANCA EXAMINADORA

Prof. Dr. Nilton Vieira Junior
Instituto Federal de Minas Gerais – *Campus* Arcos
Orientador

Prof. Dr. Carlos Bernardes Rosa Júnior
Instituto Federal de Minas Gerais – *Campus* Formiga

Profa. Dra. Maria Elizabeth de Gouvêa
Instituto Federal de Minas Gerais – *Campus* Formiga

Formiga, dia 9 de dezembro de 2016.

“O pessimista vê dificuldade em cada oportunidade; o otimista vê oportunidade em cada dificuldade.”

Winston Churchill.

RESUMO

Com o crescimento do movimento DIY (faça você mesmo), estimulado principalmente pela difusão de componentes e ferramentas como Arduino e impressoras 3D, cada vez mais diversas tecnologias se tornam acessíveis a usuários comuns. Motivado por resultados satisfatórios e precisos, este conceito tem chegado também aos laboratórios de pesquisa para fins de prototipagem. De acordo com Pearce (2012), esta possibilidade dá início à era das máquinas livres, onde tanto *software* quanto *hardware* são disponibilizados de maneira gratuita. Baseado neste conceito, o presente trabalho desenvolveu uma máquina CNC (Comando Numérico Computadorizado) de *software* e *hardware* abertos, visando o baixo custo e fácil reprodução.

Palavras chave: Máquinas CNC, Grbl, Tecnologias livres, Arduino, G-code, usinagem e gravação.

ABSTRACT

With the growth of the DIY (Do It Yourself) movement, stimulated mainly by the diffusion of components and tools like Arduino and 3D printers, more and more diverse technologies become accessible to ordinary users. Motivated by satisfactory and accurate results, this concept has also reached the research laboratories for prototyping purposes. According to Pearce (2012), this possibility opens the age of free machines, where both software and hardware are available free of charge. Based on this concept, the present work developed a Computer Numerical Control (CNC) machine of open software and hardware, aiming at low cost and easy reproduction.

Keywords: CNC machines, Grbl, Free technologies, Arduino, G-code, machining and engraving.

LISTA DE FIGURAS

Figura 1 - 1959: Fresa CNC Milwaukee-Matic-II, primeira máquina com um trocador de ferramentas automático.....	15
Figura 2 - Impressora 3D objet1000.	16
Figura 3 - Etapas do processo de fabricação de uma peça com o auxílio de uma máquina CNC.	17
Figura 4 - Sistema de eixos de deslocamento.	18
Figura 5 - Exemplo de uma máquina CNC com 7 eixos.	19
Figura 6 - Exemplo de uma <i>router</i> CNC com seus 3 eixos de coordenadas sinalizados onde a ferramenta (<i>spindle</i>) desloca-se sobre o material.	20
Figura 7 - Fresadora CNC FC1000.....	20
Figura 8 - <i>Router</i> CNC modelo I Servo de fabricação nacional.	21
Figura 9 - Esquema de deslocamento de uma <i>router</i> CNC de 3 eixos.	22
Figura 10 - Torno CNC ROMI Centur 30D.	23
Figura 11 - Esquema de deslocamento de um torno CNC de 3 eixos.	23
Figura 12 - Esquema de deslocamento de uma <i>laser</i> CNC de 2 eixos.	24
Figura 13 - Exemplo de um desenho e seu G-code.	26
Figura 14 - Grandes desenvolvedores que fornecem suportem ao Grbl CNC.	27
Figura 15 - Diagrama de conexão do <i>firmware</i> Grbl v0.9 em um Arduino UNO.....	28
Figura 16 - Exemplo de projeto desenvolvido em um programa CAD.	29
Figura 17 - Silhueta formada pelo trajeto que a ferramenta irá percorrer contidos no G-code de uma peça.	30
Figura 18 - Tela de comando do Universal GcodeSender.	31
Figura 19 - Interface de visualização do processo em tempo real do Universal GcodeSender.	32
Figura 20 - Interface do programa CAM bCNC.....	33
Figura 21 - Interface da plataforma <i>web</i> Easel.	34
Figura 22 - Alguns objetos disponíveis na biblioteca online que podem ser feitos com o Easel.	34
Figura 23 - Etapas de um projeto desenvolvido com o Easel.	35
Figura 24 - X-carve, Shapeoko e máquina desenvolvida no projeto.	36
Figura 25 - Perfil de alumínio estrutural 20x40mm.	37

Figura 26 - Placas de ACM e compensado de madeira.	38
Figura 27 - Peças planas do projeto.	38
Figura 28 - Sistema trilhos e rolamentos.	39
Figura 29 - Disposição dos perfis de alumínio estrutural que compõem o sistema trilho- estrutura.	40
Figura 30 - Rolamentos, 627z, 608zz e 608zz revestido.	40
Figura 31 - Exemplo de conjunto de rolamentos utilizados no projeto.	41
Figura 32 - Encaixe dos rolamentos sobre os trilhos.	41
Figura 33 - Eixo fuso, cremalheira e correia dentada.	42
Figura 34 - Sistema de tração do eixo Z.	43
Figura 35 - Arduino UNO rev3.	45
Figura 36 - Diagrama de portas usadas pelo firmware no Arduino UNO.	45
Figura 37 - Esquema de conexão do Arduino UNO diretamente aos <i>drivers</i> de potência.	46
Figura 38 - Arduino CNC shield v3.	47
Figura 39 - A4988 <i>Stepper Motor Driver Carrier</i>	48
Figura 40 - Esquema de ligação do microcontrolador, driver de potência, motor de passo e fonte de alimentação.	48
Figura 41 - Montagem dos componentes eletrônicos.	49
Figura 42 - Conjunto estator e rotor de um motor de passo de 4 fases.	50
Figura 43 - Motores utilizados no projeto.	51
Figura 44 - Fonte de alimentação utilizada no projeto.	52
Figura 45 - Partes de um motor <i>spindle</i>	53
Figura 46 - Micro retifica Western usada como <i>spindle</i> da máquina CNC.	53
Figura 47 - Tipos de fresas.	54
Figura 48 - Micro retifica usinando uma peça de alumínio.	54
Figura 49 - Vista explodida do projeto.	57
Figura 50 - <i>Preview</i> do modelo 3D já disponibilizado na internet.	56
Figura 51 - Etapas do projeto de fabricação.	58
Figura 52 - Peças do projeto sendo usinadas.	58
Figura 53 - Peças já usinadas e prontas para a montagem.	59
Figura 54 - Diagrama de conexão dos motores e Alimentação.	59
Figura 55 - Montagens mecânica e eletrônica completa.	60
Figura 56 - Interface de comunicação para conexão e configuração do <i>firmware</i>	61
Figura 57 - Configurações pré-estabelecidas no <i>firmware</i>	62

Figura 58 - Configurando os parâmetros da máquina.	62
Figura 59 - Subdivisão da área de corte em quadrados de 250x250mm.	63
Figura 60 - Peças para usinagem em madeira compensada visualizadas no <i>software</i> Easel.	64
Figura 61 - Peça para usinagem em alumínio visualizada no <i>software</i> Easel.	65
Figura 62 - Peças teste usinadas em madeira compensada.	65
Figura 63 - Comparação entre uma peça usinada em madeira compensada e seu desenho.	66
Figura 64 - Exemplo de peças usinadas em madeira compensada.	67
Figura 65 - Comparação ente peça usinada em alumínio e seu desenho.	67

LISTA DE QUADROS

Quadro 1 - Vantagens e desvantagens dos sistemas de tração.	42
Quadro 2 - Consumo de energia dos dispositivos.	51
Quadro 3 - Lista de componentes.	55
Quadro 4 - Custos das peças da máquina CNC.	68
Quadro 5 - Comparação de preços ente maquinas CNC equivalentes.	70
Quadro 6 - Tempo de montagem da máquina CNC.	71

LISTA DE SIGLAS

ACM	Aluminium Composite Material
CAD	Computer Aided Designe
CAM	Computer Aided Manufacturing
CNC	Controle Numérico Computadorizado
DIY	Do It Yourself
NC	Controle Numérico
PWM	Pulse Width Modulation
SMD	Surface Mount Device
USB	Universal Serial Bus

SUMÁRIO

1	INTRODUÇÃO	13
1.1	Justificativa	13
1.2	Objetivo	13
1.3	Estrutura do Trabalho	14
2	FUNDAMENTAÇÃO TEÓRICA	15
2.1	Tecnologia CNC	15
2.2	Eixos de Deslocamento	18
2.3	Tipos de máquinas CNC	19
2.3.1	<i>Fresadora CNC</i>	20
2.3.2	<i>Router CNC</i>	21
2.3.3	<i>Torno CNC</i>	22
2.3.4	<i>Laser CNC</i>	24
3	SISTEMA PROPOSTO	25
4	LINGUAGENS, <i>FIRMWARE</i> E PROGRAMAS.....	26
4.1	G-code	26
4.2	Grbl	27
4.3	Programas CAD e CAM	29
4.4	Terminal Serial	30
4.4.1	<i>Universal GcodeSender</i>	31
4.4.2	<i>bCNC</i>	32
4.4.3	<i>Easel</i>	33
5	MECÂNICA	36
5.1	Estrutura	36
5.1.1	Alumínio Estrutural	37
5.1.2	Componentes base e de fixação	37
5.2	Sistema de descolamento	39
5.2.1	Trilhos.....	39
5.2.2	Rolamentos	40

5.2.3	Sistema de tração	42
6	ELETRÔNICA	44
6.1	Placa controladora	44
6.1.1	Arduino UNO	44
6.1.2	Arduino CNC <i>Shield</i>	46
6.1.3	<i>Drivers</i> de potência.....	47
6.2	Atuadores.....	49
6.3	Fonte de alimentação	51
7	FERRAMENTA.....	53
8	FABRICAÇÃO E MONTAGEM.....	55
8.1	Configuração eletrônica	60
8.2	Testes.....	63
8.2.1	Análises de peças usinadas.....	65
9	RESULTADOS E DISCUSSÕES.....	68
9.1	Custos	68
9.2	Facilidade de Reprodução	70
9.3	Qualidade.....	71
9.4	Possíveis aplicações	71
10	CONSIDERAÇÕES FINAIS	73
	REFERÊNCIAS.....	75
	APÊNDICE A - PARÂMETROS DE CONFIGURAÇÃO DA MÁQUINA CNC	77
	ANEXO 1 - APOSTILA PARA CONFIGURAÇÃO DO <i>FIRMWARE</i> GRBL (CONRADO, 2016)	78

1 INTRODUÇÃO

Máquinas CNC (Comando Numérico Computadorizado) são estruturas capazes de reproduzir peças elaboradas computacionalmente a partir de mecanismos automatizados de usinagem. O presente trabalho tem foco em desenvolver uma máquina CNC de baixo custo, feita com componentes encontrados no mercado nacional e de forma a demandar a mínima mão de obra possível, assim como pouco conhecimento técnico eletroeletrônico e mecânico do usuário para sua construção e montagem.

Visa-se tornar o projeto uma ferramenta versátil e de qualidade, mantendo-a com *hardware* e *software* abertos, de modo a permitir seu uso nas mais diversas aplicações em educação, pesquisa e pequenas produções industriais.

A motivação para o projeto foi a experiência observada durante a graduação de engenharia, onde os alunos são submetidos a trabalhos acadêmicos que complementam a sua formação. Geralmente, estes trabalhos de cunho prático demandam o uso de métodos complexos e laboriosos para confecção de seus componentes. Esta situação faz com que grande parte do tempo de desenvolvimento do projeto seja gasto em trabalhos manuais, já que, tecnologias que automatizam o processo se mostram muitas vezes inacessíveis financeiramente à realidade acadêmica.

1.1 Justificativa

Métodos de confecção automática de peças e produtos, quando aplicados na academia, podem permitir que o estudante se dedique mais ao desenvolvimento conceitual do projeto do que na manufatura propriamente. Este tempo extra no desenvolvimento garante um maior envolvimento com o tema, possibilitando assim um ganho no aprendizado, além de permitir a implementação de projetos não antes possíveis devido à ausência de ferramentas. Além disso, do ponto de vista industrial, em especial para empresas de pequeno porte, tem-se o benefício de um equipamento de baixo custo, vez que a aplicação de ferramentas deste gênero possibilita expressivo ganho de qualidade e rapidez na linha de produção, possibilitando ainda a confecção de protótipos para fins de pesquisa e desenvolvimento. Com isso, empreendedores e estudantes de posse destas ferramentas se tornam profissionais mais competitivos no mercado de trabalho.

1.2 Objetivo

O presente trabalho se propõe a desenvolver uma máquina CNC, de baixo custo e fácil reprodução onde, a partir de um desenho feito no computador, este equipamento possa realizar cortes em madeiras, polímeros e metais macios como ligas de alumínio, bronze ou cobre, além de gravação em superfícies planas de forma automatizada.

Visando impulsionar o desenvolvimento tecnológico nacional o presente trabalho também propõe que o *hardware* e *software* sejam totalmente livres, onde o usuário poderá reproduzir ou modificar o projeto para qualquer finalidade, inclusive comercial. Desta forma, a máquina aqui desenvolvida poderá não só produzir peças para novos projetos, mas sim ser a plataforma de novos projetos.

Além disso, a estrutura desta CNC pode inclusive ser utilizada como base para outras máquinas de acordo com o desejo do usuário como, por exemplo, ser implementada para impressão em 3D, corte e gravação a *laser* entre outros. Deste modo, uma CNC se torna uma poderosa ferramenta para a criação de novos projetos, confecção de peças mecânicas, impressão de circuitos e desenvolvimento de outras máquinas.

1.3 Estrutura do Trabalho

Os elementos textuais que compõem este trabalho encontram-se divididos em 10 capítulos. Neste Capítulo, encontra-se a introdução aqui apresentada. O Capítulo 2 apresenta a fundamentação teórica na qual o trabalho foi embasado. O Capítulo 3 apresenta o sistema proposto, assim como os parâmetros que se desejou atender no projeto. O Capítulo 4 expõe as demandas computacionais necessárias para o funcionamento e desenvolvimento do projeto, assim como os *softwares* utilizados para a operação do mesmo quando finalizado. Os Capítulos 5 e 6 tratam, respectivamente, dos componentes mecânicos e eletrônicos utilizados no projeto e o Capítulo 7 apresenta a ferramenta proposta para a máquina operatriz. O Capítulo 8 mostra o processo de fabricação e montagem do trabalho proposto.

Por fim, os Capítulos 9 e 10 apresentam respectivamente as considerações finais do presente trabalho e algumas possibilidades de trabalhos futuros que podem ser desenvolvidos a partir do projeto aqui apresentado.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta um breve referencial teórico do projeto que foi desenvolvido, passando por uma explicação do que é a tecnologia, como funciona e os tipos de máquinas mais comuns que empregam a tecnologia CNC.

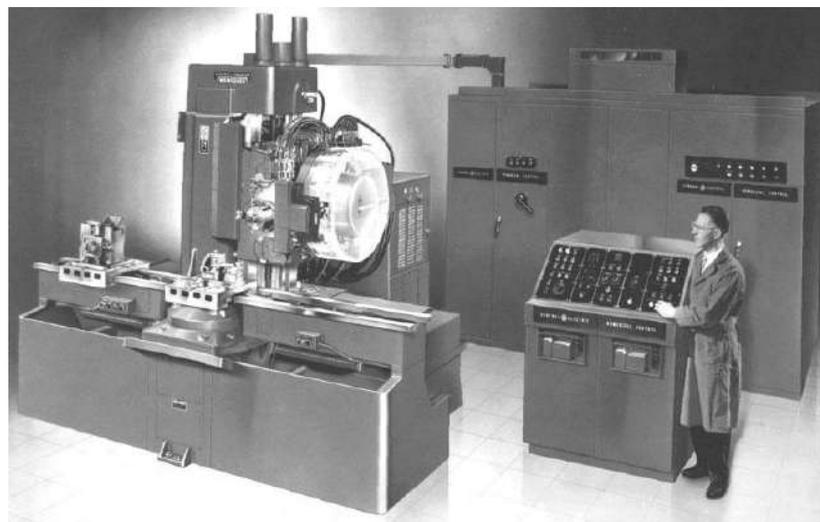
2.1 Tecnologia CNC

O comando numérico computadorizado (CNC) é uma tecnologia que permite o controle de máquinas a partir de interfaces computadorizadas. Em 1947 a *Parsons Corporation*, uma pequena fabricante rotores e hélices de helicópteros, experimentou adaptar uma máquina de usinagem convencional para ser controlada numericamente por um computador que lia as informações de controle em um cartão perfurado. Surgia assim, o início das máquinas de comando numérico computadorizado (SEAMES, 2001, p.2).

Desde então, a tecnologia CNC vem evoluindo e se adaptando juntamente com o avanço tecnológico, automatizando ainda mais os processos de fabricação. Já em 1959 era apresentada a primeira fresadora CNC capaz de trocar automaticamente sua ferramenta de corte (GENG, 2015, p.10-2), dessa forma a máquina demandava menos interferência humana no processo (

Figura 1).

Figura 1 - 1959: Fresa CNC Milwaukee-Matic-II, primeira máquina com um trocador de ferramentas automático.



Fonte: GENG (2015, p.10-2).

Hoje em dia, as máquinas CNC já são capazes de criar peças através da extrusão de matéria prima para a construção de peças complexas por deposição de camadas, as impressoras 3D, como visto na Figura 2.

Figura 2 - Impressora 3D objet1000.



Fonte: SMG3D (2016).

As máquinas de NC (controle numérico) e posteriormente as CNC são máquinas-ferramentas que automatizam o processo de produção sem a necessidade de supervisão constante de um operador, além de permitir um ganho de produtividade. Controle numérico pode ser definido como uma operação de máquinas-ferramentas por meio de instruções especificadas em código para o sistema de controle de máquina (SMID, 2003, p.1).

As instruções recebidas pela máquina seguem um padrão linguagem de programação comumente conhecido como G-code.

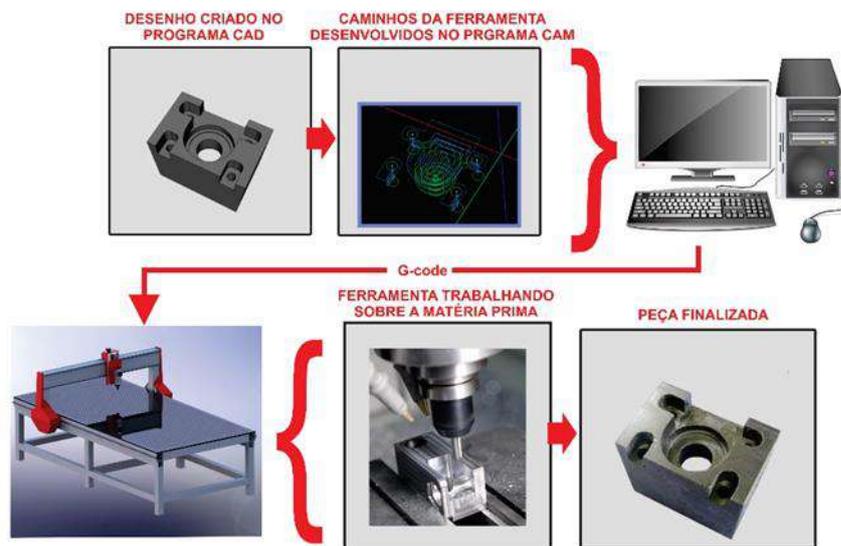
G-code é uma notação abreviada para um conjunto de funções da máquina, que regulam o movimento das várias partes da máquina (BELL; CHARLES, 2014, p.14). Nos próximos capítulos esta linguagem será melhor apresentada.

O processo de operação de uma máquina CNC ocorre da seguinte forma: a peça é desenhada em um programa CAD (Desenho Auxiliado por Computador) e, em seguida, o arquivo é enviado a um programa CAM (Manufatura Auxiliada por Computador), usado para criar o arquivo G-code que contém as instruções e os parâmetros das ferramentas escolhidas para o processo, seguindo as formas da peça do arquivo CAD. As instruções em G-code, são

enviadas à máquina CNC que interpreta linha por linha e executa as funções nele contida. Com este processo, uma vez criado o desenho da peça e convertido para a linguagem da máquina, pode se repetir a produção utilizando o mesmo arquivo. Além disso, o desenho CAD pode ser replicado no programa CAM de forma que o mesmo faça a leitura de diversas peças e as reproduza em um único arquivo G-code, permitindo assim a produção em escala apenas com uma única programação.

O processo de fabricação de uma peça segue o fluxo proposto na Figura 3.

Figura 3 - Etapas do processo de fabricação de uma peça com o auxílio de uma máquina CNC.



Fonte: o próprio autor.

Pode-se descrever o processo visto na Figura 3 nas seguintes etapas:

- A peça é desenhada em um *software* CAD de acordo com as dimensões e formas especificadas no projeto;
- O arquivo CAD é lido por um *software* CAM que interpreta as dimensões do desenho e as transformam em G-code;
- O G-code é enviado linha por linha para a unidade controladora da máquina CNC;
- A controladora interpreta os sinais e os envia para os *drivers* de potência dos atuadores;
- Os *drivers* de potência recebem os sinais digitais e os amplificam, gerando potência suficiente para os atuadores;
- Os atuadores transformam o sinal em movimento mecânico;
- Os movimentos mecânicos causam o deslocamento da ferramenta de corte;

- A ferramenta de corte trabalha sobre a matéria prima reproduzindo o desenho criado computacionalmente.

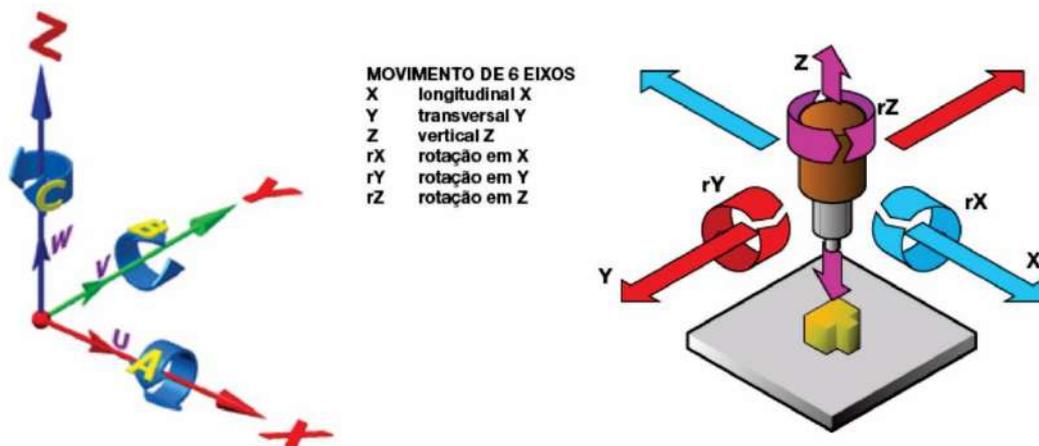
Assim, frente ao processo exposto, tem-se fabricada uma peça com o uso da tecnologia CNC. A seguir, será apresentado o sistema de eixos de deslocamento de uma máquina CNC.

2.2 Eixos de Deslocamento

O eixo de uma máquina CNC refere-se à capacidade de deslocamento em uma das direções do plano espacial. Quando se diz que uma máquina CNC tem 3 eixos, significa que a máquina tem a capacidade de deslocamento em 3 direções simultâneas.

De acordo com Madison (2006, p.11-17), o sistema de coordenadas cartesianas com seus três eixos lineares (X, Y e Z) é adotado como eixos lineares primários de deslocamento da máquina, porém a capacidade de rotação em torno destes eixos gera mais 3 eixos ditos angulares. A rotação em torno do eixo X recebe o nome de eixo rX ou eixo A; já a capacidade de rotação em torno do eixo Y é chamada de eixo rY ou eixo B; e no eixo Z é chamado de eixo rZ ou eixo C.

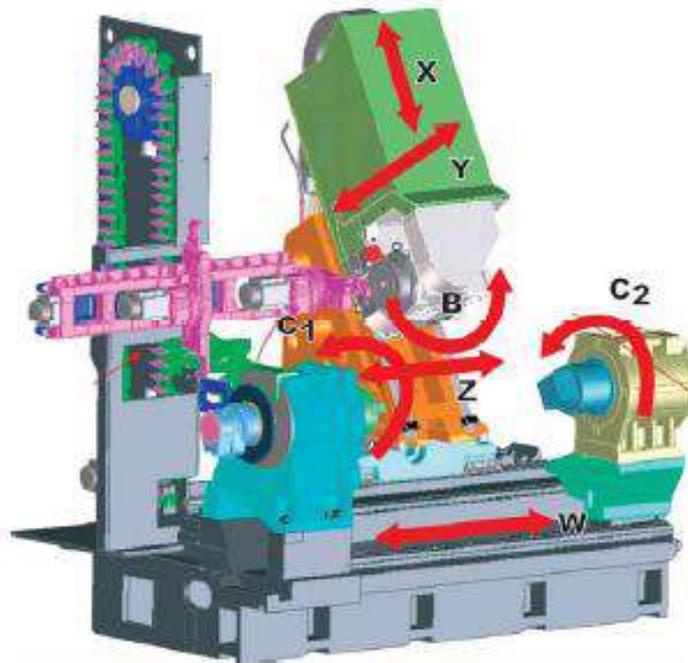
Figura 4 - Sistema de eixos de deslocamento.



Fonte: Adaptado de <https://goo.gl/EMsnxQ> e <https://goo.gl/RkYDGw> (Acesso em: 20 nov. 2016).

Ainda segundo Madsion (2006, p.11-17), existem máquinas CNC de 7 eixos ou mais, que são utilizadas para confecção de peças com elevado grau de complexidade. Para isso são depositos outros três eixos lineares e paralelos aos eixos primários. São os eixos U, V e W, paralelos aos eixos X, Y e Z, respectivamente, e estes eixos são chamados de eixos lineares secundários (Figura 5).

Figura 5 - Exemplo de uma máquina CNC com 7 eixos.

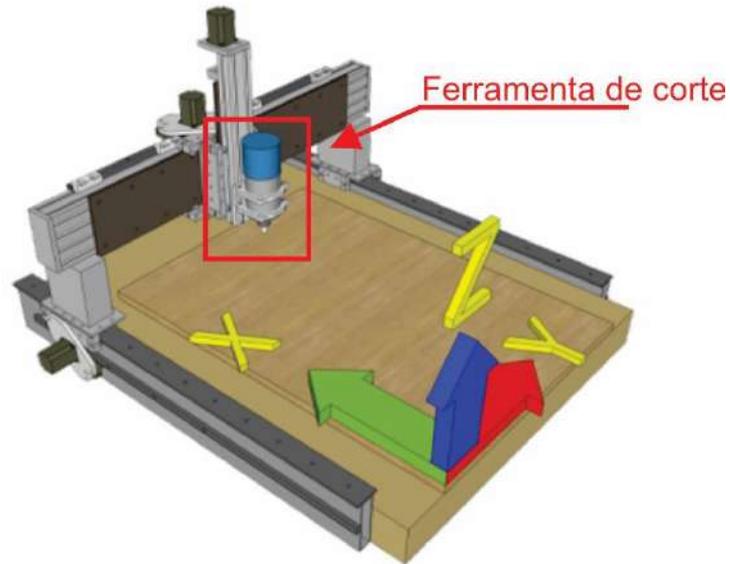


Fonte: POST-PRO (2016).

2.3 Tipos de máquinas CNC

Basicamente, uma máquina CNC é um equipamento eletrônico que desloca ferramentas, ou o material a ser trabalhado, ao longo de trilhos em um plano de coordenadas cartesianas. Este movimento é a interpretação do G-code pela máquina que pode ser, por exemplo, fresadora, *router*, máquina de corte a *laser* e corte a plasma, torno, dobradoras automáticas, gravadora de circuitos eletrônicos, insersora de componentes eletrônicos SMD (*Surface Mount Device*) e muitas outras. A Figura 6 apresenta o exemplo de uma máquina *router* CNC com três eixos de deslocamento.

Figura 6 - Exemplo de uma *router* CNC com seus 3 eixos de coordenadas sinalizados onde a ferramenta (*spindle*) desloca-se sobre o material.



Fonte: Adaptado de <https://goo.gl/O3u9zk> (Acesso em: 20 nov. 2016) .

2.3.1 Fresadora CNC

A fresadora CNC é uma máquina desenvolvida para usinar materiais desde alumínio, aço e até mesmo titânio. Devido ao material de trabalho, sua construção prioriza a força de deslocamento e não a velocidade (Figura 7).

Figura 7 - Fresadora CNC FC1000.



Fonte: Soluções industriais (200-).

O material é fixado em uma base e esta se desloca nos eixos X e Y do plano cartesiano, já o motor que gira a ferramenta de corte se desloca ao longo do eixo Z.

O custo de uma fresadora CNC é maior do que o de uma *router* CNC, de igual área de corte, devido aos materiais e equipamentos utilizados nas mesmas.

2.3.2 Router CNC

Pode se dizer que uma *router* CNC é uma fresadora CNC projetada para usinar materiais mais leves como madeiras, polímeros e ligas de alumínio. Sua construção pode priorizar tanto força de deslocamento quanto velocidade de deslocamento, variando com a especificação do projeto e pretensão de material que a mesma irá usinar (Figura 8).

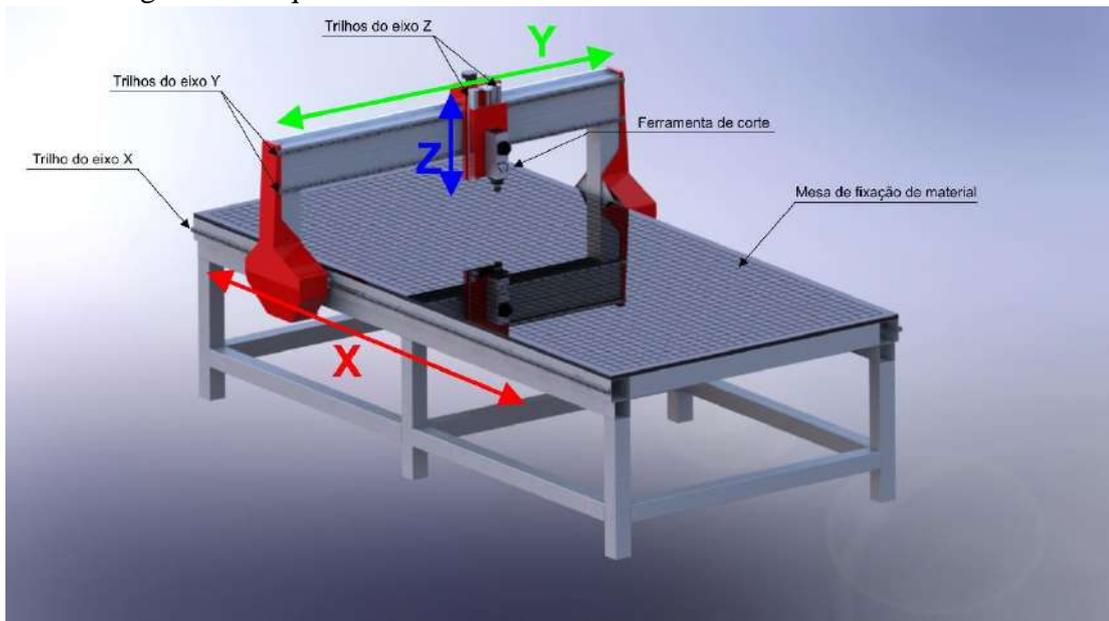
Figura 8 - Router CNC modelo I Servo de fabricação nacional.



Fonte: DS4 (200-).

O material é fixado em uma mesa onde trilhos, em suas laterais, garantem a fixação e mobilidade do conjunto de deslocamento do eixo X. Existe um outro segmento de trilhos perpendicular ao eixo X onde o conjunto de deslocamento Y desliza, e neste último, trilhos verticais garantem o deslocamento do conjunto de ferramentas que corre no eixo Z. A Figura 9 descreve melhor esta estrutura.

Figura 9 - Esquema de deslocamento de uma *router* CNC de 3 eixos.



Fonte: o próprio autor.

Este arranjo permite a mobilidade da ferramenta de corte ao longo da área da mesa, inclusive em profundidades variáveis, sendo assim pode-se criar peças 3D complexas.

2.3.3 Torno CNC

É um equipamento semelhante ao torno mecânico, porém, contando com um conjunto de motores e controladoras que automatizam o processo de usinagem de peças de revolução ou cilíndricas. Ele possui duas bases sobre as quais se deslocam o eixo X (seu deslocamento determina o diâmetro da peça) e outro o eixo Z (seu deslocamento determina o comprimento da peça), o material (geralmente um tarugo circular) é acoplado em uma castanha, que por sua vez é fixada em uma placa acoplada ao eixo central da máquina (chamado de eixo árvore) (Figura 11).

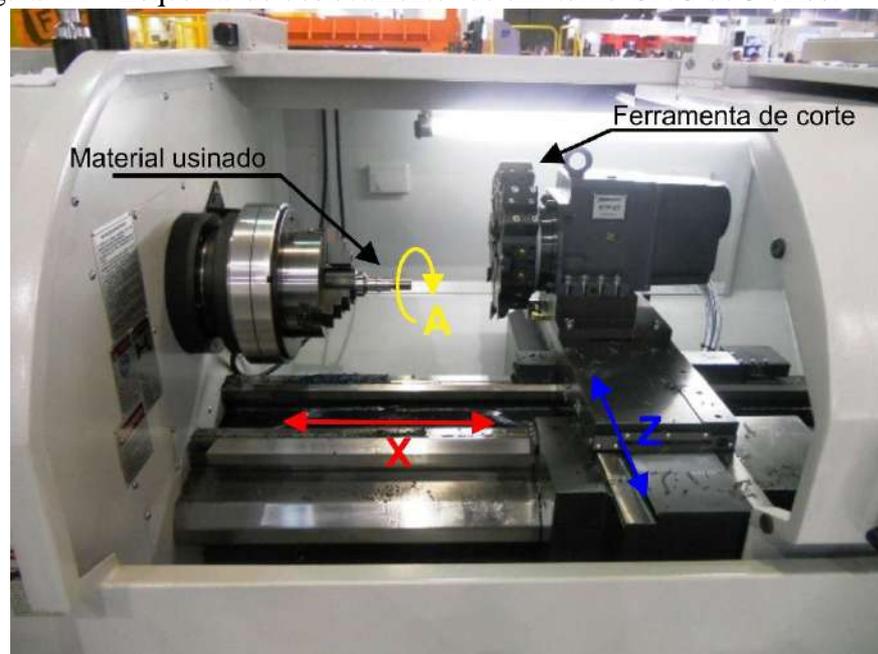
Figura 10 - Torno CNC ROMI Centur 30D.



Fonte: ROMI (200-).

Diferentemente de outras máquinas CNC, onde a ferramenta possui um movimento rotacional para perfurar e desbastar o material, o eixo árvore faz com que o material acoplado a ele gire ao invés das ferramentas, as ferramentas se deslocam ao longo dos eixos X e Z e assim trabalham no material dando a sua forma projetada no desenho CAD. A Figura 11 apresenta um torno CNC com seus eixos de deslocamento em destaque.

Figura 11 - Esquema de deslocamento de um torno CNC de 3 eixos.



Fonte: Adaptado de <https://goo.gl/LeROm1> (Acesso em: 20 nov. 2016).

2.3.4 Laser CNC

O funcionamento de uma *laser* CNC é igual ao de uma *router* CNC, diferindo apenas na ausência do eixo de deslocamento Z, onde o mesmo é substituído por um raio *laser* com potência ajustável que incide sobre o material. A potência configurada define se o material será cortado ou terá sua superfície gravada pelo raio *laser*. Na Figura 12 pode-se observar que a *laser* CNC apresenta apenas 2 eixos de deslocamento, que são suficientes para o processo.

Figura 12 - Esquema de deslocamento de uma *laser* CNC de 2 eixos.



Fonte: Adaptado de <https://goo.gl/nFQ3S8> (Acesso em: 20 nov. 2016).

A construção de uma máquina *Laser* CNC prioriza a velocidade de deslocamento, já que a ferramenta não tem contato com o material usinado.

3 SISTEMA PROPOSTO

Como previamente relatado, o trabalho propõe a criação de uma máquina CNC de pequeno porte com área de trabalho útil de 500x500mm e que possa usinar materiais como madeiras, polímeros e metais macios como alumínio. Com eletrônica baseada em Arduino, melhor detalhado no item 6.1.1, *firmware* Grbl item 4.2 e estrutura física em alumínio estrutural, propõe-se a criação uma máquina de baixo custo e versátil.

A máquina proposta visa a mínima intervenção necessária do usuário na sua construção e configuração, assim como, capacidade de expansão para oferecer suporte a novos projetos. Tendo isso em mente, prioriza-se neste projeto componentes prontos que demandam apenas montagem ou conexão o mais simples possível, visando ainda a redução do custo final para o seu desenvolvimento.

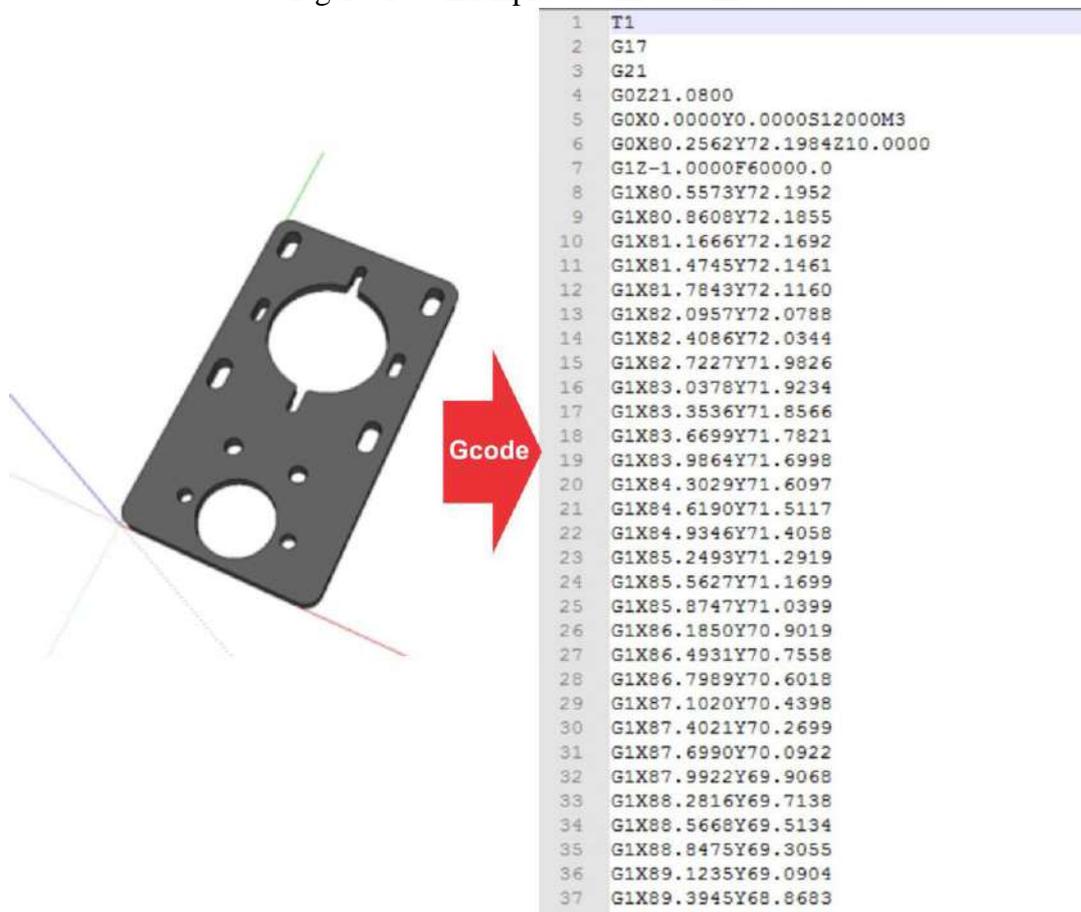
Existem outros trabalhos na literatura que se propuseram a desenvolver equipamentos semelhantes, como os de LYRA, 2010, SILVA; ZANIN, 2011 e FACHIM, 2013, porém o agora proposto apresenta como vantagens uma estrutura simplificada, de maior facilidade de reprodução e garante uma maior precisão e durabilidade. Além disso, utiliza uma eletrônica mais avançada e compatível com *softwares* mais recentes, comparado aos seus precursores.

4 LINGUAGENS, *FIRMWARE* E PROGRAMAS

4.1 G-code

G-code é o nome genérico para a linguagem de programação utilizada para controle de máquinas-ferramentas. Quando um desenho é feito no computador, o mesmo contém informações que devem ser convertidas para uma linguagem que a máquina irá entender e o G-code é a linguagem responsável por essa interpretação (uma ilustração do seu código pode ser vista na Figura 13).

Figura 13 - Exemplo de um desenho e seu G-code.



Fonte: o próprio autor.

Quando o usuário aciona um botão para movimentar uma máquina CNC ou quando carrega um arquivo para iniciar a criação de uma peça, linhas de comandos em G-code são enviados para a máquina onde um controlador as interpreta e transforma os comandos em movimentos ou aciona ferramentas.

4.2 Grbl

O Grbl (pronuncia-se “gerbil”) é um *firmware* desenvolvido e otimizado para ser gravado em microcontroladores atmega328p (sendo estes presentes em placas Arduino UNO compatíveis), com a finalidade de controlar os movimentos de máquinas operatrizes (GITHUB, 2015).

Trata-se de um *software* livre, lançado sob a licença GPLv3 e desenvolvido em C, que requer um *hardware* simples e opera seguindo os padrões industriais. Por estes motivos, o Grbl se tornou um fenômeno entre as plataformas de código aberto, sendo base para máquinas CNC, impressoras 3D entre outros. A Figura 14 apresenta os desenvolvedores que apoiam o projeto Grbl.

Figura 14 - Grandes desenvolvedores que fornecem suporte ao Grbl CNC.

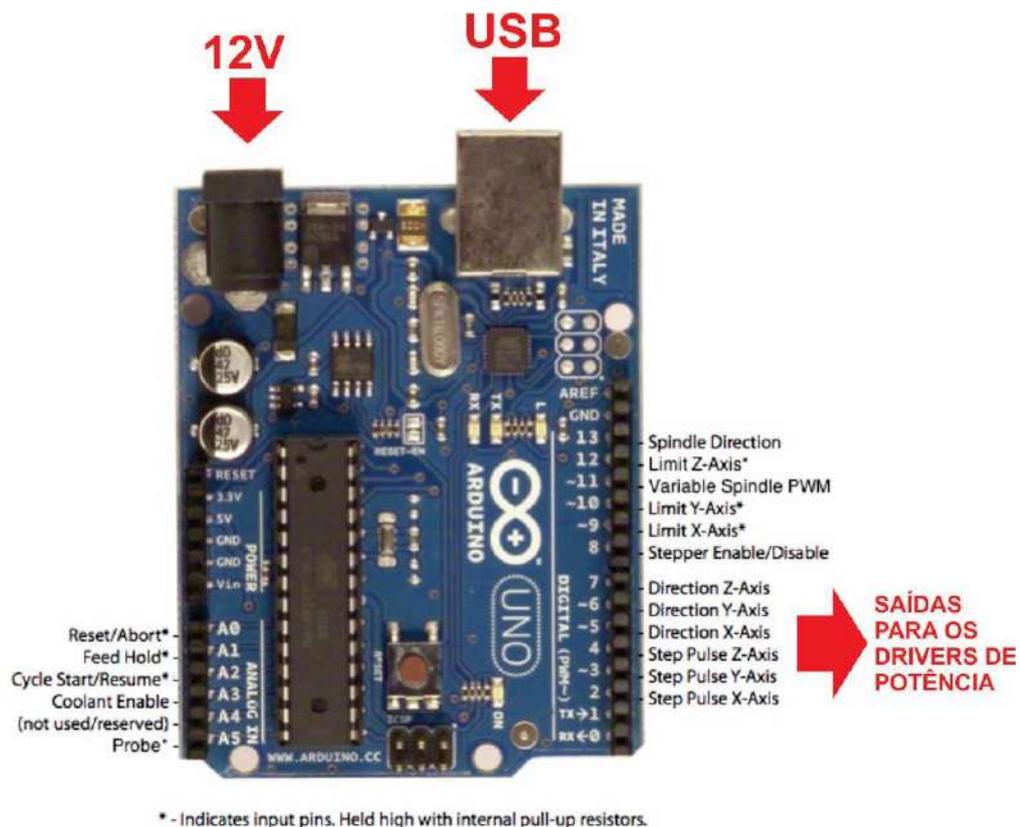


Fonte: GITHUB (2015).

Existem tutoriais de instalação e configuração do Grbl disponibilizados no próprio *site* dos desenvolvedores¹. Porém, o escolhido e disponibilizado no Anexo 1 deste trabalho foi o desenvolvido por Conrado (2016). O tutorial é autoexplicativo e muito bem detalhado, por isto, o corpo deste trabalho apresentará apenas o funcionamento do *firmware*.

A função do Grbl, instalado em um Arduino UNO, é receber via USB e interpretar o conjunto de instruções enviados em G-code pelo computador e, a partir destes, transmitir os comandados para os *drivers* de potência e atuadores que transformam os sinais em movimentos. Além disso, o Grbl também suporta comandos de operação de ferramentas como liga/desliga, velocidade e direção de rotação e interações como pausa e interrupção durante o processo (Figura 15).

Figura 15 - Diagrama de conexão do *firmware* Grbl v0.9 em um Arduino UNO.



Fonte: Adaptado de <https://goo.gl/Myaael> (Acesso em: 20 nov. 2016).

A versão utilizada do Grbl é a v0.9j, versão mais atual do *firmware* até a data de criação deste trabalho.

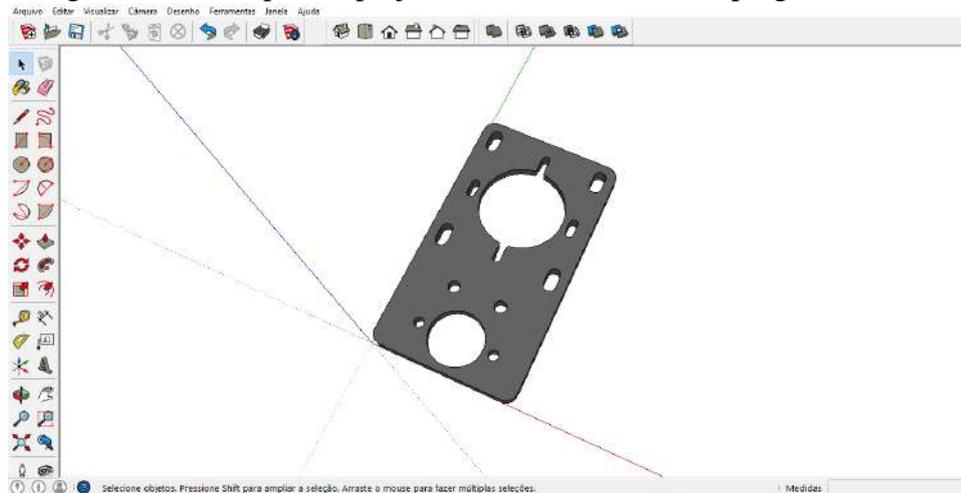
¹ <https://github.com/grbl/grbl/wiki/Flashing-Grbl-to-an-Arduino> (Acesso em: 24 nov. 2016).

4.3 Programas CAD e CAM

Computer-Aided (assistido por computador) é o significado das duas primeiras letras de ambas as siglas (CAD e CAM), Design e Manufacturing (desenho e manufatura) são o significado das letras subsequentes. Ou seja, desenho assistido por computador e manufatura assistida por computador. Cada um tem parte importante no desenvolvimento de peças fabricadas com o auxílio de uma máquina CNC.

O CAD é utilizado durante o processo de desenvolvimento e dimensionamento da peça, o usuário através de uma interface gráfica cria um desenho bidimensional ou tridimensional do projeto ou parte dele. Este desenho fornece informações como largura, altura e espessura, além de particularidades como furos e rebaxos que o objeto possa ter (Figura 16).

Figura 16 - Exemplo de projeto desenvolvido em um programa CAD.



Fonte: o próprio autor.

Alguns exemplos de programa CAD são o AutoCAD®, o SolidWorks® e o Sketchup. O programa CAD, utilizado tanto na concepção do projeto da máquina CNC quanto na criação de peças que a mesma irá usinar, foi o *software* Sketchup da Trimble, por se tratar de uma ferramenta gratuita, intuitiva, de fácil aprendizado e operação.

Já um programa CAM é empregado após a criação do desenho no *software* CAD. O CAM converte as dimensões do desenho em coordenadas que a ferramenta terá que percorrer para fabricar o objeto. Pode se dizer que o CAM cria o processo de produção, calculando e ajustando os percursos de deslocamento de acordo com as dimensões do objeto e da ferramenta que será utilizada na manufatura. O *software* CAM utilizado neste trabalho foi o Easel da

Inventables, também por ser uma ferramenta gratuita e intuitiva. A Figura 17 apresenta uma peça já em G-code, visualizada na ferramenta Universal Gcode Sender – mais informações a seu respeito serão tratadas nos tópicos a seguir.

Figura 17 - Silhueta formada pelo trajeto que a ferramenta irá percorrer contidos no G-code de uma peça.



Fonte: o próprio autor.

Pode-se dizer que o *software* CAM exige as geometrias geradas pelo *software* CAD, sendo assim estas duas ferramentas são usadas em conjunto no processo de operação de uma máquina CNC.

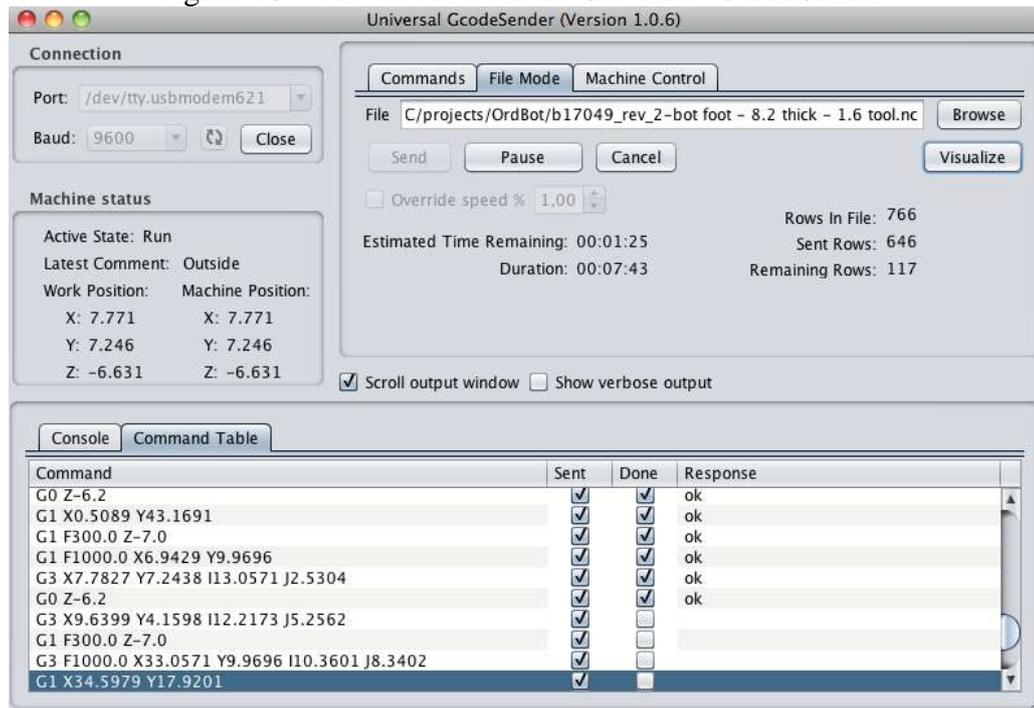
4.4 Terminal Serial

Para a comunicação entre computador e máquina, é necessário um programa que envie o conjunto de instruções que o *firmware* deverá executar. Existem diversos programas que o usuário pode escolher para executar essa função (compatíveis com o Grbl), dentre eles se destacam o Universal G-code Sender, bCNC e Easel (GITHUB, 2016). E estes foram testados durante a execução deste trabalho. Estes programas compõem os programas CAM utilizados neste trabalho e o usuário pode escolher, de acordo com sua preferência ou afinidade, qual dos programas operar.

4.4.1 Universal GcodeSender

O mais simples e também utilizado na configuração da *firmware* máquina, no momento de sua montagem, o Universal GcodeSender trabalha fazendo a leitura do G-code e o enviando para a máquina (Figura 18).

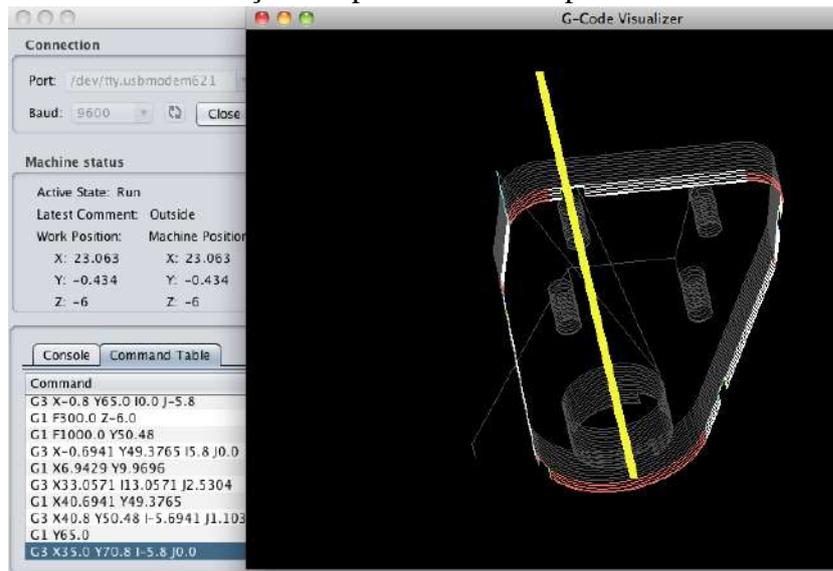
Figura 18 - Tela de comando do Universal GcodeSender.



Fonte: <https://goo.gl/Ot8xCM> (Acesso em: 20 nov. 2016).

Desenvolvido em Java, o mesmo pode ser executado em sistemas operacionais, Windows®, Linux, OSX e Raspberry Pi. Um empecilho observado é que o mesmo apenas envia G-codes, não os cria e nem fornece suporte a edição, mesmo assim o projeto apresenta grande estabilidade e é desenvolvido em colaboração com o Grbl sendo, por ele próprio, recomendado. A Figura 19 apresenta a interface de *visualizer* do Universal GcodeSender, nela pode-se acompanhar o processo de fabricação de uma peça enquanto a mesma está sendo usinada.

Figura 19 - Interface de visualização do processo em tempo real do Universal GcodeSender.



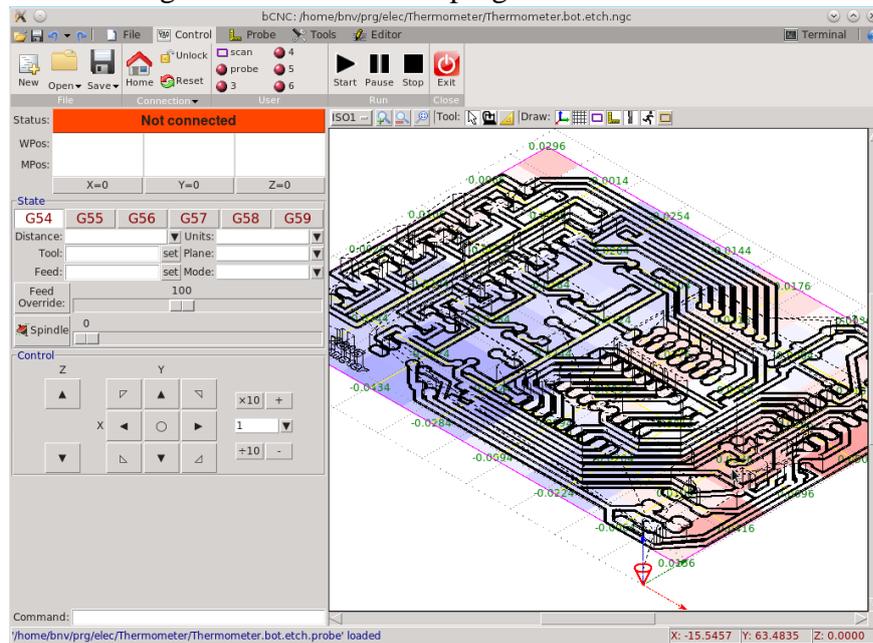
Fonte: <https://goo.gl/ZkQjO1> (Acesso em: 20 nov. 2016).

O Universal GcodeSender é uma plataforma livre, lançado sob a licença GPLv3 e a versão utilizada neste trabalho foi a v1.0.9.

4.4.2 bCNC

Também multi-plataforma, o bCNC pode trabalhar em sistemas Windows, Linux, OSX e Raspberry Pi. Possui uma interface um pouco mais complexa do que o Universal G-code Sender, mesmo assim é fácil e intuitiva, sendo dotado de mais ferramentas e fornece suporte a edição e criação de G-codes. Isso permite importar arquivos desenvolvidos em programas CAD, transforma-los em G-code e envia-los diretamente para a máquina CNC (Figura 20).

Figura 20 - Interface do programa CAM bCNC.



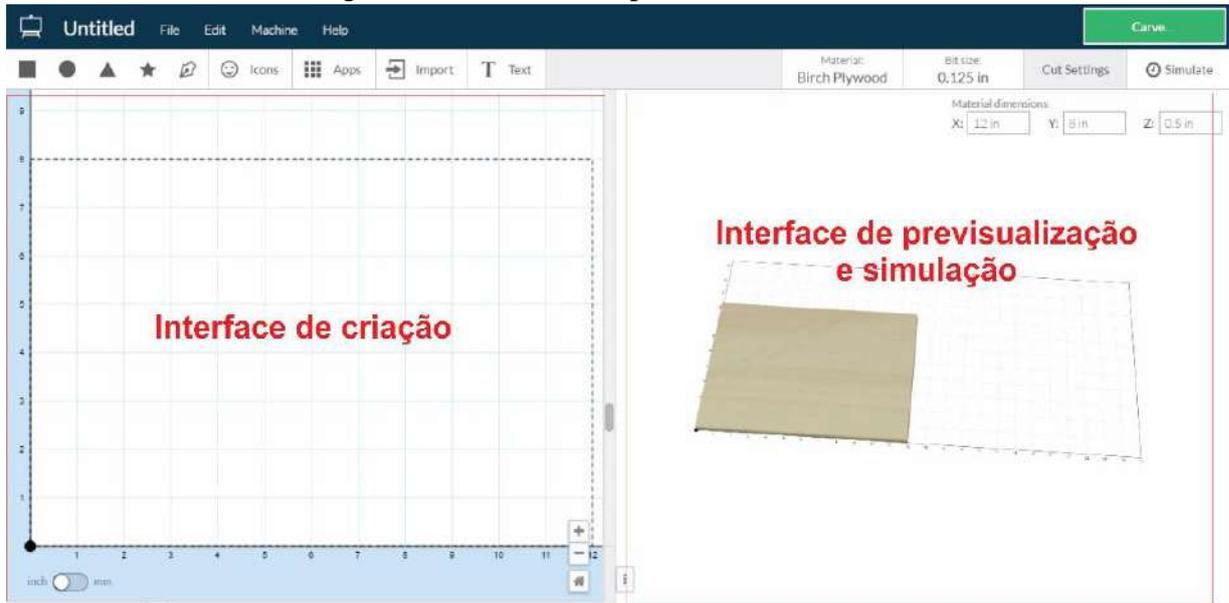
Fonte: <https://goo.gl/ZkQjO1> (Acesso em: 20 nov. 2016).

Uma das ferramentas do bCNC é o sistema de auto-nivelamento por área do eixo Z. O usuário pode ajustar a distância da ferramenta em relação a mesa de corte, para compensar as possíveis elevações e rebaixos que a superfície da área de trabalho da máquina possa ter.

Disponibilizado livremente sobre licença GPL 2.0, a versão utilizada neste trabalho foi a v0.9.6.

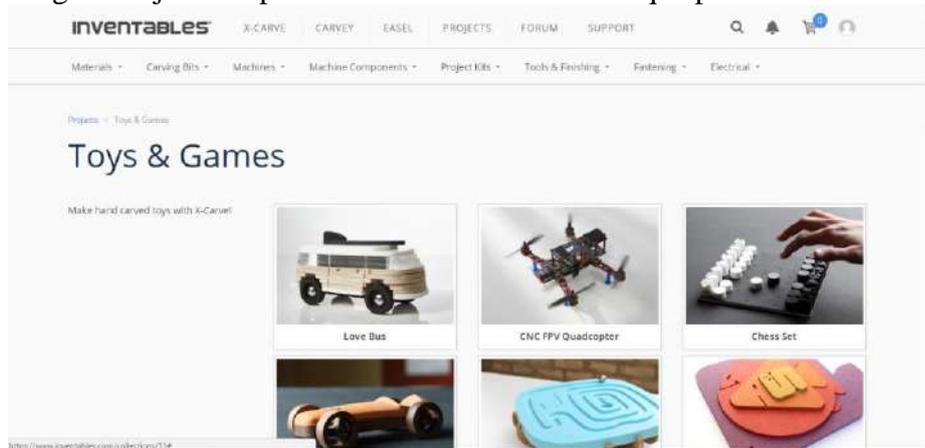
4.4.3 Easel

Easel é uma plataforma *web* disponibilizada gratuitamente pela Inventables, que funciona diretamente no navegador. É a plataforma mais amigável, intuitiva e prática para iniciantes. Ele foi desenvolvido para trabalhar em conjuntos com máquinas CNC da Inventables, Carvey e Xcarve, que também funcionam com o *firmware* Grbl e por isso a máquina criada neste projeto é diretamente compatível com a plataforma (Figura 21).

Figura 21 - Interface da plataforma *web* Easel.

Fonte: Adaptado de: <http://easel.inventables.com/> (Acesso em: 20 nov. 2016).

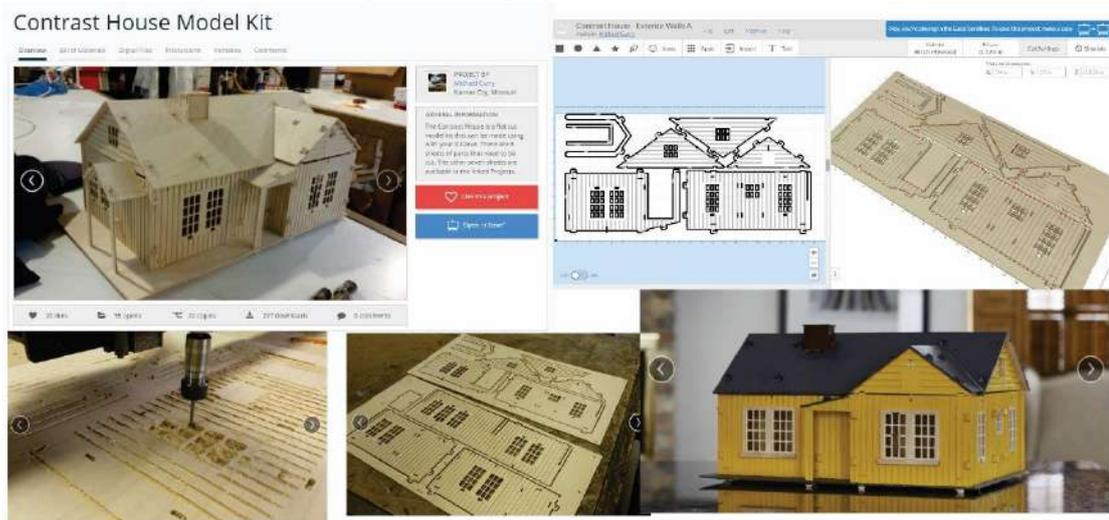
Com o propósito de ser *all-in-one*, o Easel é uma ferramenta CAD e CAM que também dá suporte a desenvolvedores que queiram criar aplicativos para serem implementados na plataforma. Além disso, o usuário pode importar, desenhar, compartilhar ou escolher objetos em uma biblioteca *online* e criá-los na sua própria máquina. A Figura 22 ilustra alguns objetos disponíveis na biblioteca *online* que podem ser importados para o Easel.

Figura 22 - Alguns objetos disponíveis na biblioteca *online* que podem ser feitos com o Easel.

Fonte: <https://goo.gl/NVvNHR> (Acesso em: 20 nov. 2016).

O usuário precisa apenas se cadastrar no site do desenvolvedor² para ter acesso ao programa. Durante os primeiros projetos, uma interface de ajuda é apresentada para guiar no uso das principais ferramentas da plataforma. A Figura 23 apresenta um objeto na biblioteca *online*, que foi importado para o Easel e fabricado por uma *router* CNC.

Figura 23 - Etapas de um projeto desenvolvido com o Easel.



Fonte: o próprio autor.

O Easel requer apenas a instalação de um programa chamado *Easel Driver*, que se comunica diretamente com o *firmware* Grbl da máquina compatível. Na plataforma, o usuário ajusta o tamanho da sua área de corte e tipo de ferramenta. Após isso, o Easel está pronto para operar a máquina CNC.

² <http://easel.inventables.com/> (Acesso em: 24 nov. 2016).

5 MECÂNICA

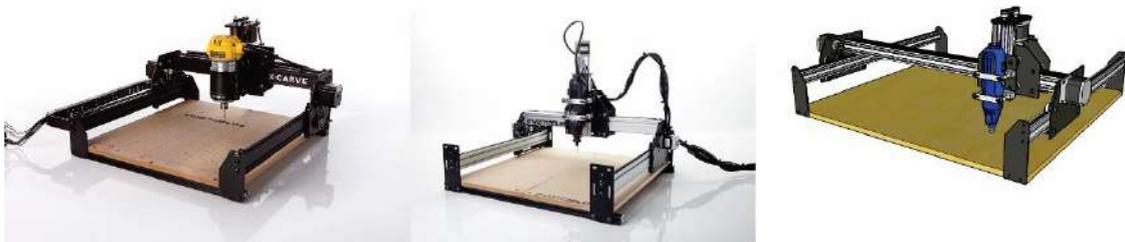
Neste capítulo serão tratados a estrutura e componentes mecânicos desenvolvidos, porém com a intenção de facilitar o acesso a usuários inexperientes, o projeto da máquina não considera cálculos complexos de resistência e equilíbrios para sua base, fato ainda reforçado por se tratar de um equipamento de pequena escala.

5.1 Estrutura

Para a construção de uma máquina CNC deve-se determinar a área de trabalho útil que se deseja obter, assim como o tipo de material que será usinado e o número de eixos que a máquina vai possuir. Estes parâmetros determinam suas dimensões finais, assim como o material empregado em sua estrutura, torque dos motores utilizados e grau de complexidade das peças que a máquina poderá usinar.

Baseado nos modelos de *hardware* abertos encontrados no mercado internacional, X-Carve e Shapeoko (Figura 24), desenvolveu-se uma estrutura leve e prática, adaptada para os materiais encontrados no Brasil, já que componentes como trilhos e rolamentos destas máquinas não estão disponíveis nacionalmente. Prezou-se também pelo baixo preço dos componentes, sem se ter uma perda na qualidade dos mesmos.

Figura 24 - X-carve, Shapeoko e maquina desenvolvida no projeto.



Fonte: o próprio autor.

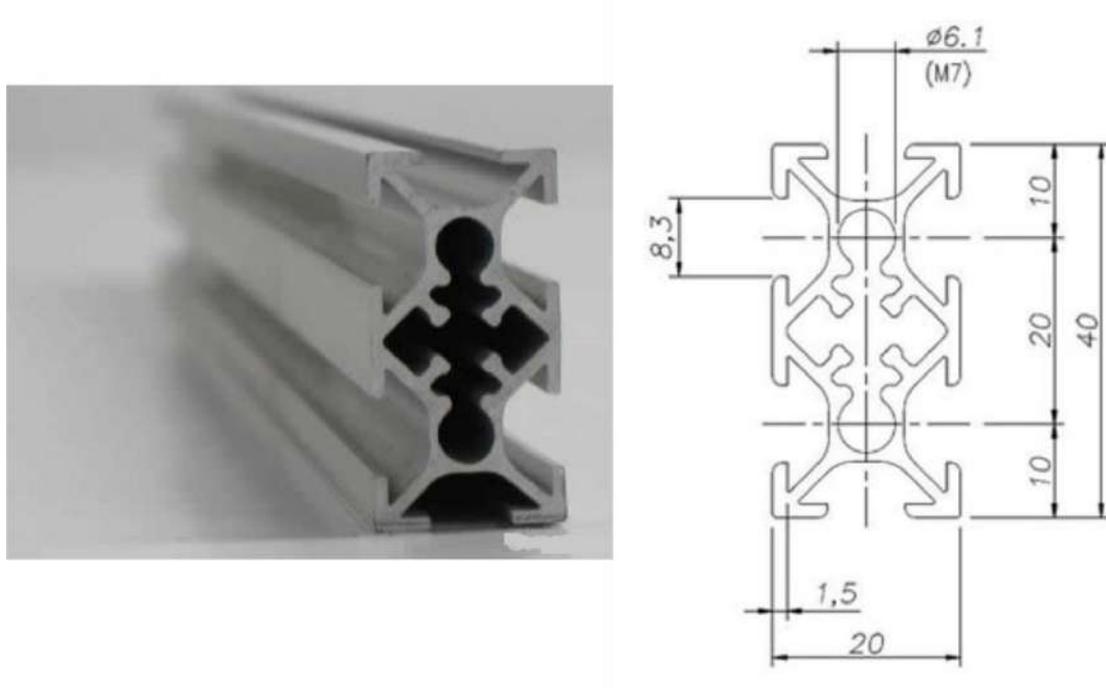
Com o objetivo de construir uma máquina de baixo custo, fácil reprodução e montagem, optou-se pela criação de um projeto que mantenha os aspectos das estruturas de máquinas CNC de código aberto mais divulgadas internacionalmente. Esta estratégia foi tomada para propiciar a facilidade de difusão do modelo criado, além de aproveitar o suporte já existente em fóruns e

comunidades na *internet* dos demais modelos, garantindo assim o não isolamento dos usuários deste projeto e criando uma base sólida para a sua continuidade.

5.1.1 Alumínio Estrutural

Perfis de alumínio estrutural são uma opção econômica para criação de estruturas, por permitir a fácil junção sem solda e nem requerer acabamento (Figura 25). Além disso, pode-se facilmente recuperar o material usado em um projeto e aplicá-lo em outro.

Figura 25 - Perfil de alumínio estrutural 20x40mm.



Fonte: o próprio autor.

5.1.2 Componentes base e de fixação

O projeto requer chapas planas, parafusos e porcas para o acoplamento de suas partes fixas e móveis. Para este propósito. Em fase de prototipagem, trabalhou-se com chapas de compensado de madeira, para testar, em um material de baixo custo, possíveis erros de projeto. No produto final utilizou-se chapas de ACM (Aluminium Composite Material), que é uma chapa formada por duas lâminas de alumínio com um núcleo de polietileno (Figura 26).

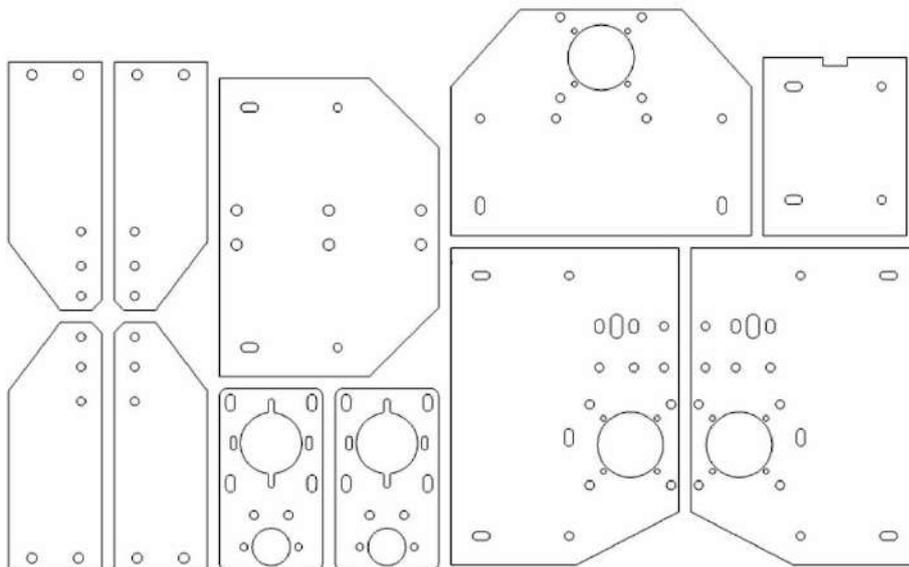
Figura 26 - Placas de ACM e compensado de madeira.



Fonte: o próprio autor.

Além de ser mais resistente, tal material resulta em um melhor acabamento ao projeto. Podem ser utilizados no projeto chapas de alumínio, madeira ou acrílico, cabe ao usuário determinar o material de mais fácil acesso para seu projeto na construção das partes necessárias à máquina (Figura 27).

Figura 27 - Peças planas do projeto.



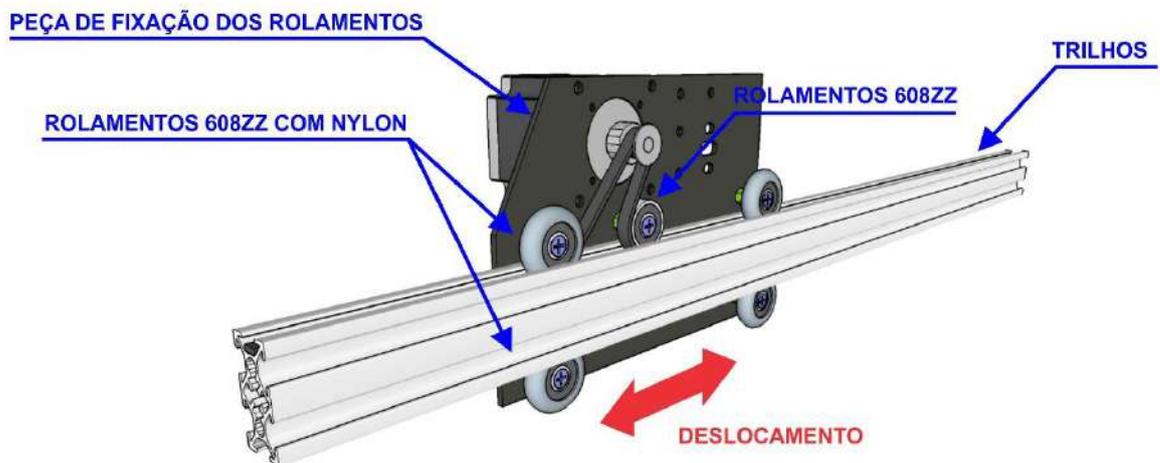
Fonte: o próprio autor.

As peças demonstradas na Figura 27 são as bases de fixação de motores, eixos e rolamentos. Estas peças são presas à estrutura por parafusos e tem o propósito de conferir estabilidade, assim como suportar os movimentos e cargas de trabalho da máquina.

5.2 Sistema de descolamento

O deslocamento da ferramenta sobre a matéria prima produz a peça projetada e o que garante a precisão no deslocamento e estabilidade da ferramenta é, justamente, o sistema composto por trilhos e rolamentos – partes estas de extrema importância na construção do projeto de uma máquina CNC (Figura 28). Um trilho com imperfeições ao longo de seu corpo, ou um rolamento com folga, pode transmitir defeitos para a peça criada.

Figura 28 - Sistema trilhos e rolamentos.



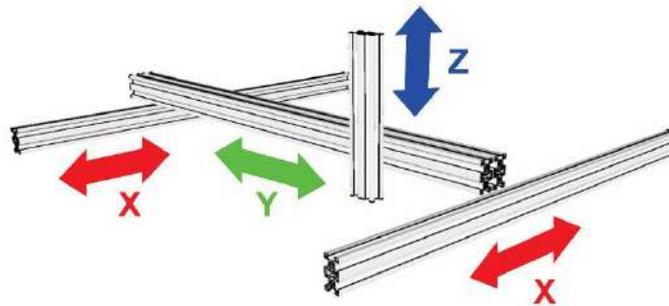
Fonte: o próprio autor.

O deslocamento acontece quando rolamentos acoplados a uma parte da máquina são dispostos sobre os trilhos que compõem outra parte da mesma. Esta junção permite que a parte onde os rolamentos se encontram acoplados tenha liberdade de movimentação ao longo do trilho.

5.2.1 Trilhos

Construídos em perfis retangulares de alumínio estrutural 20x40mm, os trilhos também têm a função de compor a estrutura da máquina e assim, geram economia por executar duas funções. Os trilhos lineares correspondentes aos eixos X, Y e Z garantem precisão e leveza por sua composição e processo de fabricação (Figura 29).

Figura 29 - Disposição dos perfis de alumínio estrutural que compõem o sistema trilho-estrutura.



Fonte: o próprio autor.

5.2.2 Rolamentos

A partir dos trilhos definidos, necessitou-se definir um conjunto de rolamentos adequados para os mesmos. As paredes laterais do trilho possuem um vão com 8,3 mm e 6mm de profundidade. As opções com dimensões mais próximas das requisitadas foram os rolamentos 627z (com 7mm de altura e 22 mm de diâmetro externo), rolamentos 608zz (com 7mm de altura e 22 mm de diâmetro externo) e rolamentos 608zz revestido de nylon (com 9mm de altura e 29 mm de diâmetro externo) (Figura 30).

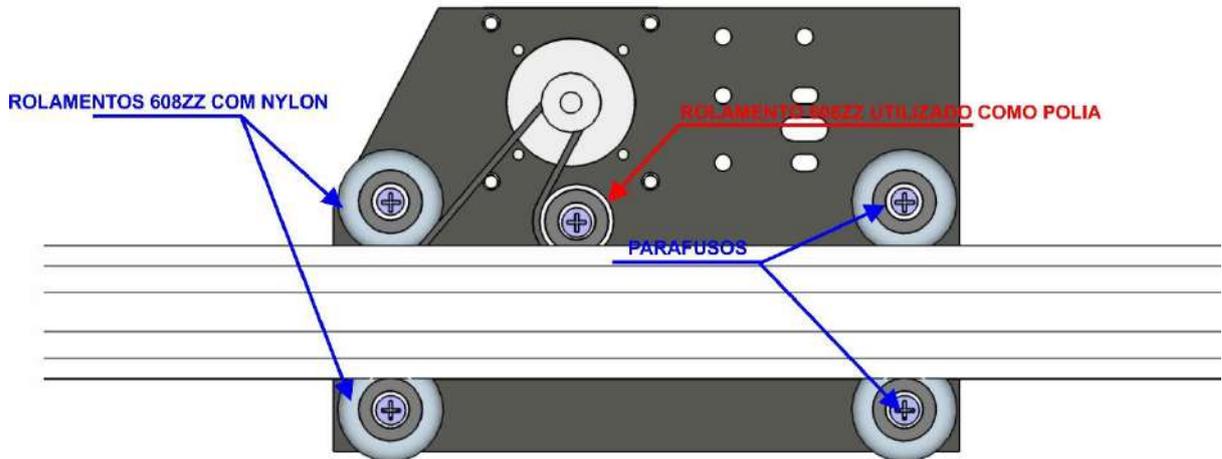
Figura 30 - Rolamentos, 627z, 608zz e 608zz revestido.



Fonte: o próprio autor.

Os dois primeiros rolamentos apresentados na Figura 30, se mostraram eficientes para funcionarem como polias tensionadoras de correias. O escolhido foi o 608zz por ser o mais encontrado no mercado. A Figura 31 apresenta um exemplo de aplicação dos dois tipos de rolamentos usados no projeto.

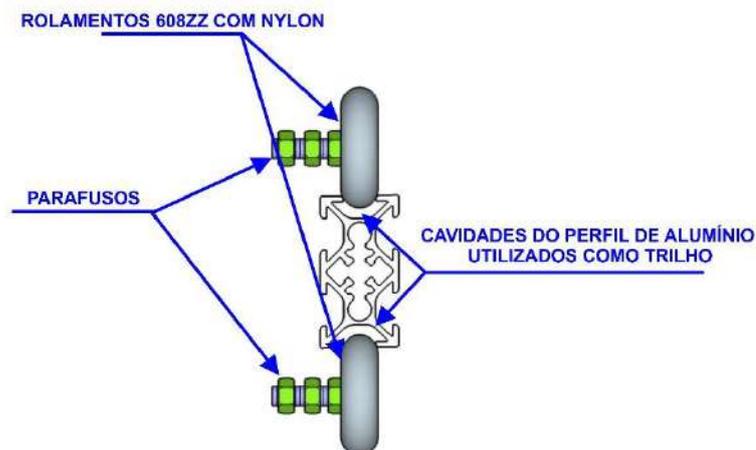
Figura 31 - Exemplo de conjunto de rolamentos utilizados no projeto.



Fonte: o próprio autor.

Para o deslocamento sobre os trilhos, o escolhido foi o 608zz revestido de nylon, por seu perfil arredondado com altura maior que o vão entre as paredes dos trilhos. Esta propriedade permite um apoio sem folga e seu diâmetro permite uma distância maior do centro do rolamento até os trilhos. Como vantagem adicional, por ser revestido em nylon, o rolamento absorve melhor as vibrações geradas pela ferramenta e pelo movimento sobre os trilhos (Figura 32).

Figura 32 - Encaixe dos rolamentos sobre os trilhos.



Fonte: o próprio autor.

O perfil com cavidades que o alumínio estrutural possui, permitiu que fossem utilizados como trilhos que guiam o deslocamento dos rolamentos.

5.2.3 Sistema de tração

Diversos sistemas de tração podem ser adotados como, eixos fusos, cremalheiras e correias dentadas. Cada um desses sistemas tem suas vantagens e desvantagens e algumas podem ser observadas no Quadro 1.

Quadro 1 - Vantagens e desvantagens dos sistemas de tração.

Sistema	Vantagem	Desvantagem
Fusos	<ul style="list-style-type: none"> • Possibilidade de usar barra roscada e Porcas de baixo custo. • Suporta alto torque. 	<ul style="list-style-type: none"> • Sistema de alto atrito. • Quanto maior, maior sua possibilidade de empenamento. • Requer pontos de apoios em sua extremidade o que aumenta o custo de implantação. • Vibração com velocidades elevadas. • Vida útil baixa
Cremalheiras	<ul style="list-style-type: none"> • Suporta alto torque. • Fácil implantação no sistema. • Vida útil alta 	<ul style="list-style-type: none"> • Acrescenta peso ao projeto. • Alto custo.
Correia dentada	<ul style="list-style-type: none"> • Opções industriais de baixo custo. • Fácil implantação no sistema • Suporta altas velocidades 	<ul style="list-style-type: none"> • Suporta torque moderado • Vida útil moderada

Fonte: o próprio autor.

A Figura 33 apresenta um exemplo dos sistemas de transmissão de eixo fuso, cremalheira e correia dentada.

Figura 33 - Eixo fuso, cremalheira e correia dentada.

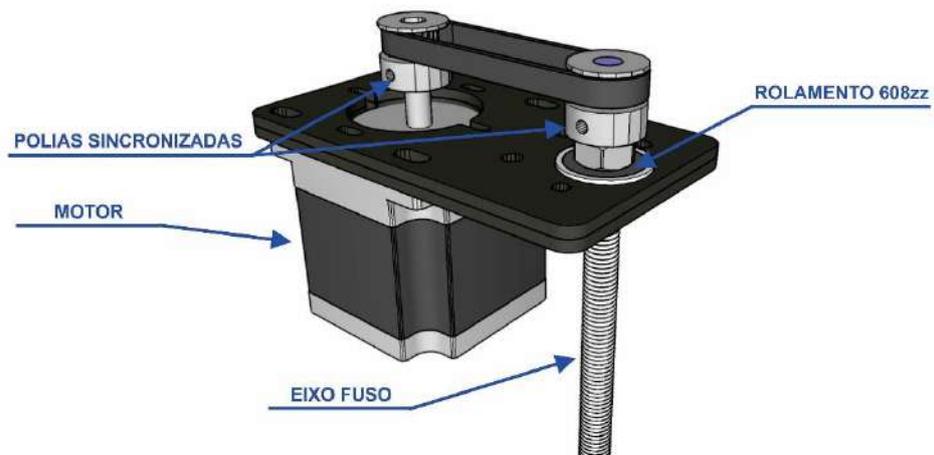


Fonte: o próprio autor.

Neste projeto adotou-se os sistemas de correia dentada para os eixos X e Y e fuso metálico para o eixo Z, este último justifica o seu uso diferenciado por sofrer com a força gravitacional no sentido paralelo ao seu movimento. Dessa forma, o esforço causado pelo peso que o sistema iria suportar incidiria diretamente nas correias de transmissão (caso o mesmo as tivesse). Já um sistema de eixo fuso metálico tem capacidade de suportar a carga do sistema sem dilatação e desgaste prematuro, sendo assim, o mais indicado para o projeto.

Vale ressaltar que eixo Z é disposto verticalmente e tem um deslocamento total de 100mm, o que permitiu a fácil implementação de um eixo fuso em seu sistema de transmissão (Figura 34). Já os eixos X e Y tem um deslocamento de 500mm e são dispostos horizontalmente, fatores estes que necessitariam de maiores cuidados por risco de empenamento e o uso de mancais de apoio em suas extremidades, além de gerar um projeto mais complexo por ter mais peças moveis.

Figura 34 - Sistema de tração do eixo Z.



Fonte: o próprio autor.

O eixo fuso do sistema de tração do eixo Z é fixado por um rolamento 608zz em sua base e conectado ao motor por um sistema de correia dentada fechada e polias sincronizadas. Isso garante que nenhum esforço perpendicular seja transmitido ao eixo do motor.

6 ELETRÔNICA

O sistema eletroeletrônico de uma máquina CNC pode ser dividido em módulos para melhor entendimento: placa controladora, *drivers* de potência, atuadores, fonte de energia e ferramenta.

É importante lembrar que um dos objetivos do projeto é permitir o fácil acesso e uso de componentes e equipamentos disponíveis no mercado nacional. Sendo assim optou-se pela não criação de dispositivos e sim pelo uso dos já existentes que serão apresentados a seguir.

6.1 Placa controladora

Responsável por receber e interpretar os dados enviados pelo computador, a placa controladora é quem coordena os movimentos executados pela máquina, como sentido, velocidade e qual atuador deve ser acionado. Além disso, a controladora pode receber comandos externos como parada, pause e retorno de trabalho.

6.1.1 Arduino UNO

Como dito anteriormente, o *firmware* Grbl foi desenvolvido para os microcontroladores Atmega328 que integram a placa Arduino UNO. E optou-se por utilizar nesse projeto a placa de desenvolvimento Arduino UNO (Figura 35) devido ao amplo suporte do desenvolvedor e comunidade, USB integrado, placas de circuito que podem ser conectadas facilmente e baixo custo.

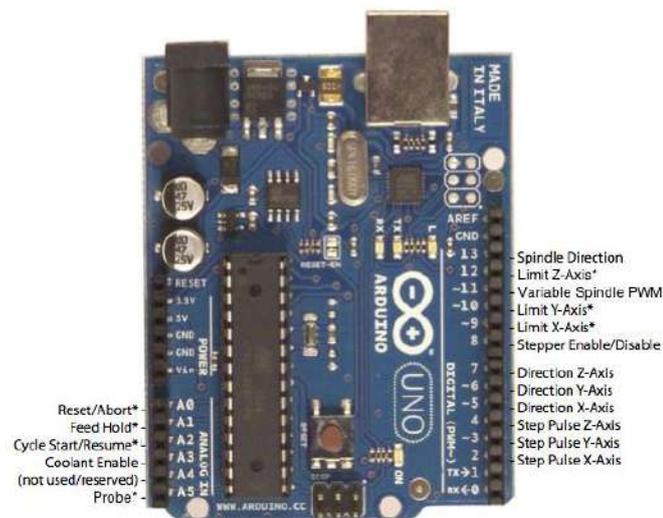
Figura 35 - Arduino UNO rev3.



Fonte: <https://goo.gl/LUBeIG> (Acesso em: 20 nov. 2016).

O Arduino UNO possui quatorze portas digitais de entrada e saída, sendo seis providas de PWM (*Pulse Width Modulation*), seis entradas analógicas e 32KB de memória *flash*. Ele recebe via USB o G-code para realizar o processo operação da máquina CNC e a partir da leitura do G-code o mesmo aciona seis de suas portas digitais para a movimentação dos atuadores, três para limite de curso e duas para controle da ferramenta. As portas analógicas recebem informações de interação do usuário com a máquina como pausar o processo ou cancelá-lo (Figura 36).

Figura 36 - Diagrama de portas usadas pelo firmware no Arduino UNO.



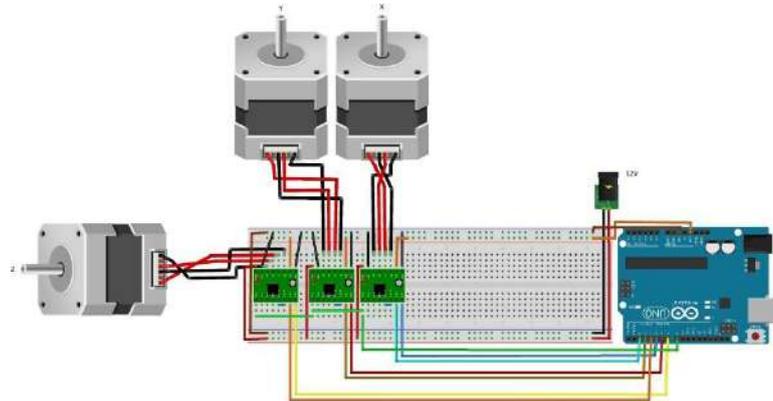
Fonte: <https://goo.gl/Myaael> (Acesso em: 20 nov. 2016).

A configuração de qual porta deve ser usada assim como o seu uso, é pré-configurado pelo Grbl. Desenvolvedores avançados podem editar o *firmware* e reconfigurar as portas de acordo com suas especificações.

6.1.2 Arduino CNC Shield

O Arduino UNO pode ser ligado através de fios diretamente aos periféricos eletrônicos da máquina, como os *drivers* de potência ou, por exemplo, botoeiras, porém, esta ligação seria demasiada complexa, susceptível a erros e mal contatos (Figura 37).

Figura 37 - Esquema de conexão do Arduino UNO diretamente aos *drivers* de potência.

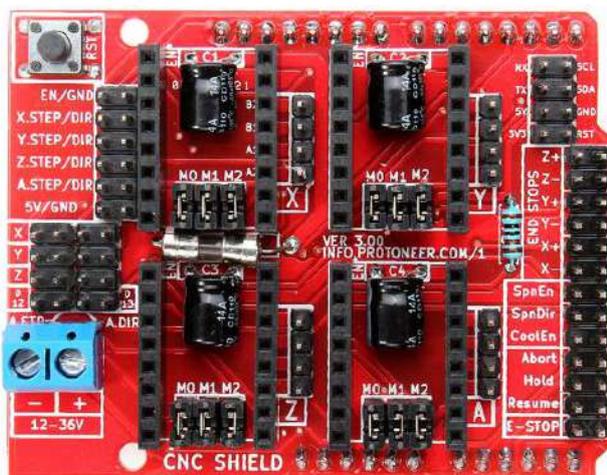


Fonte: <https://goo.gl/6oNtNR> (Acesso em: 20 nov. 2016).

Para organizar e promover uma melhor operabilidade, manutenção e evitar falhas optou-se por utilizar uma *shield* juntamente com o Arduino UNO. *Shields* (do inglês escudos), são placas de circuitos que podem ser facilmente conectadas às placas Arduino ou até mesmo a outras *shields*. Ligadas a um Arduino, elas expandem a capacidade ou facilitam conexões com *displays*, leitores de cartões de memória, módulos *bluetooth*, entre outros.

No caso deste projeto, utilizou-se o Arduino CNC Shield v3 (Figura 38), uma placa que faz a conexão dos *drivers* de potência, fonte de energia e demais periféricos com o Arduino UNO.

Figura 38 - Arduino CNC shield v3.



Fonte: <https://goo.gl/51s2Iz> (Acesso em: 20 nov. 2016).

Este *shield* simplifica a instalação eletrônica e possui encaixes para o acoplamento direto de até quatro *drivers* de potência, com circuito de alimentação de energia tanto para o Arduino quanto para os *drivers*.

6.1.3 Drivers de potência

Drivers de potência são utilizados para transformar sinais elétricos de baixa potência, recebidos em sua entrada, em pulsos elétricos de maior potência em sua saída. Seu uso é necessário quando as especificações de alimentação (corrente e tensão) de um dispositivo é maior do que as do sistema que o controla.

No projeto, os *drivers* de potência são necessários para o funcionamento dos atuadores. O *driver* compatível e suficiente para os acionamentos dos atuadores utilizados é o pololu A4988 *Stepper Motor Driver* (Figura 39).

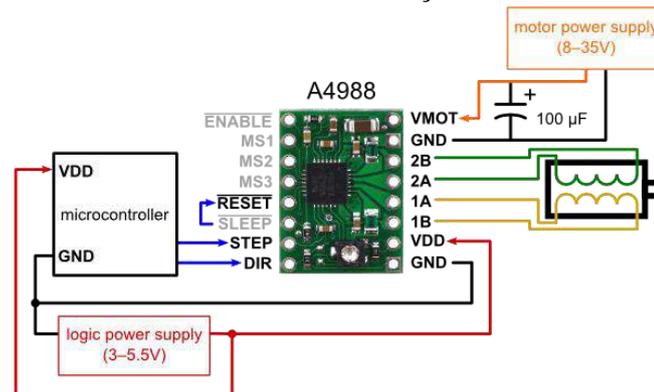
Figura 39 - A4988 Stepper Motor Driver Carrier.



Fonte: <https://goo.gl/e3I5YY> (Acesso em: 20 nov. 2016).

O *driver* de potência é utilizado para controlar motores de passo e pode operar com tensões entre 8 e 35 volts e entregar até 2 ampères por bobina.

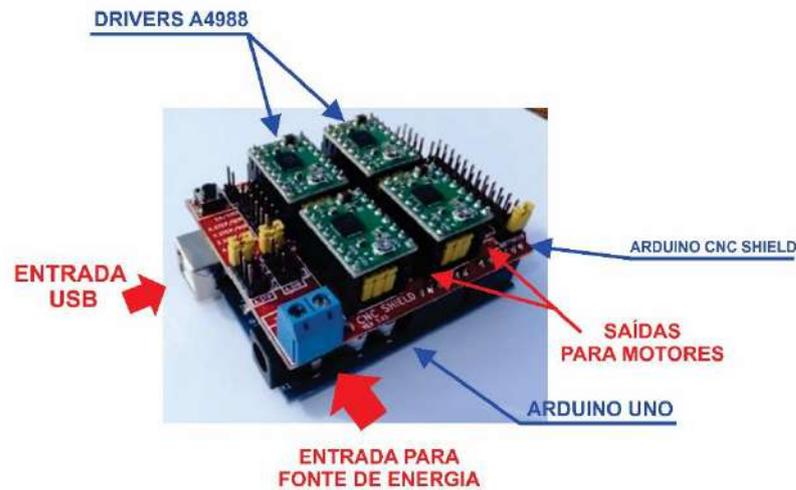
Figura 40 - Esquema de ligação do microcontrolador, *driver* de potência, motor de passo e fonte de alimentação.



Fonte: <https://goo.gl/e3I5YY> (Acesso em: 20 nov. 2016).

A conexão eletrônica esboçada na Figura 40 é feita pelo Arduino CNC *shied*, sendo necessário apenas que o usuário junte os componentes eletrônicos e forneça a alimentação adequada como demonstrado na Figura 41.

Figura 41 - Montagem dos componentes eletrônicos.



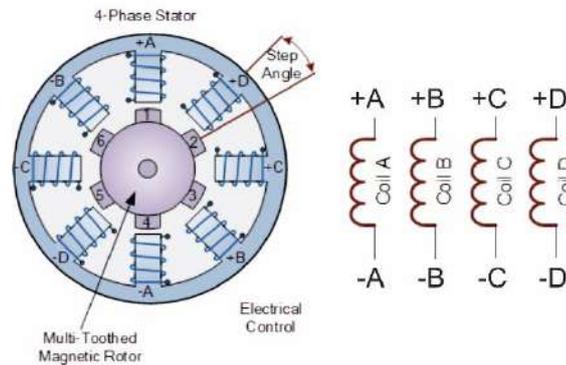
Fonte: o próprio autor.

6.2 Atuadores

Atuadores são elementos que convertem um dado tipo de energia em movimento. Os utilizados nesse trabalho foram motores de passo, que convertem pulsos elétricos recebidos em seus terminais em movimento. Seu nome se dá justamente pela forma que o mesmo gera movimento.

Basicamente o motor de passo é composto por um conjunto rotor e estator, sendo este último dotado de pares de bobinas de cobre interligadas entre si e posicionadas em sentidos opostos, estando suas extremidades ligadas aos terminais do motor. Cada conjunto de bobinas é chama de fase e, quando energizadas, as interações eletromagnéticas no interior do motor fazem o rotor girar. O motor é alimentado por pulsos elétricos ordenados em suas fases, onde cada pulso gera um movimento angular do rotor e este movimento é chamado de passo. A sequência de acionamento das fases determina o sentido de giro do rotor (Figura 42).

Figura 42 - Conjunto estator e rotor de um motor de passo de 4 fases.



Fonte: Adaptado de: <https://goo.gl/SHpG46> (Acesso em: 20 nov. 2016).

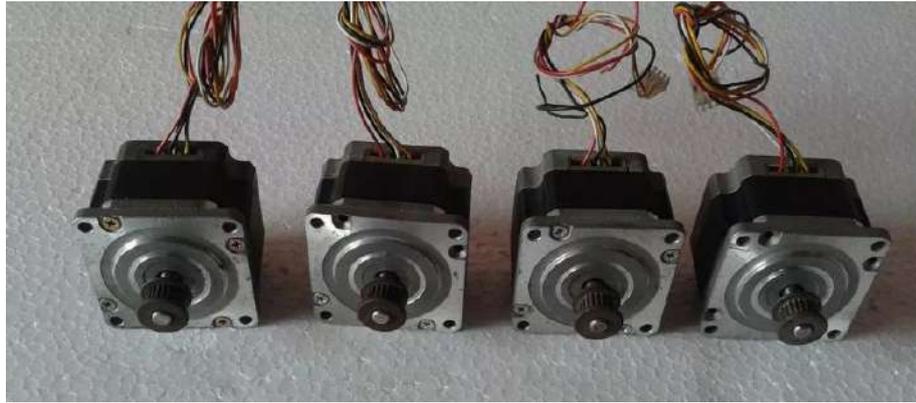
É possível subdividir o passo de um motor em micro-passos, fornecendo simultaneamente corrente elétrica a duas fases. Fornecer o mesmo valor de corrente a duas fases, gera um campo magnético igual em dois conjuntos de bobinas e o movimento angular resultante será equivalente à metade do caminho de um passo original do motor, ou seja, meio passo. Com valores de corrente diferentes aplicado a duas fases, a posição do rotor tenderá a se deslocar em direção proporcional ao campo eletromagnético mais forte. *Drivers* de potência têm a capacidade de controlar o fluxo de energia em cada fase do motor e, com isso, têm a capacidade de subdividir o passo do motor em valores proporcionais às correntes nas duas fases acionadas por vez.

Utilizar a função micro-passo permite diminuir a distância mínima de deslocamento do motor de passo e isso proporciona uma maior precisão, porém ocasiona a perda de torque nominal do motor. Por esta razão, deve-se analisar qual será a aplicação do projeto afim de descobrir qual será a melhor configuração do mesmo.

Pela possibilidade de controle de movimento, contabilizando os pulsos enviados, a aplicação de motores de passo em máquinas CNC é extremamente difundida e proporciona resultados satisfatórios com baixo investimento. Tendo isso em conta, optou-se por utilizar esta categoria de motores neste projeto.

Os motores de passo utilizados são de dimensões que seguem a padronização Nema 23 (SCHNEIDER, 200-), possuem 5 kgf.cm de torque, passo de 1.8° e corrente máxima de 2 Ampères por fase (Figura 43).

Figura 43 - Motores utilizados no projeto.



Fonte: o próprio autor.

6.3 Fonte de alimentação

A fonte de alimentação é o dispositivo projetado para suprir toda demanda de energia necessária para o pleno funcionamento de um projeto. Antes de sua escolha, deve se determinar a demanda energética de todos os dispositivos que serão alimentados pela fonte, evitando assim falhas de projetos que podem ser causados por uma alimentação insuficiente e danos oriundos da queima de componentes alimentados por uma tensão acima de suas limitações. Para isso elaborou se o Quadro 2, demonstrando a demanda energética de cada dispositivo.

Quadro 2 - Consumo de energia dos dispositivos.

Componente	Quantidade	Tensão de operação	Consumo	Consumo total
Arduino UNO	1	7 a 12 V	800mA	800mA
Arduino CNC <i>Shield</i>	1	12 a 36V	--	--
<i>Driver</i> de potencia	4	8 a 35 V	50 mA	200mA
Motor de passo	4	5 a 36 V	2 A	8 A
Demanda energética compatível		12 V		9A

Fonte: o próprio autor.

Observa-se no Quadro 2, que a placa Arduino UNO tem recomendação de tensão de operação de 7 a 12 volts e os *drivers* de potência operam na faixa de tensão de 8 a 35 volts. Optou-se, então, por utilizar uma fonte de energia que forneça tensão nominal de 12 volts em sua saída.

Outro fator importante é a corrente mínima necessária para o funcionamento do projeto, então, com cada motor de passo consumindo 2 Ampères (tendo o projeto utilizado 4 motores) tem-se uma corrente mínima de 8 Ampères necessários, caso todos os motores sejam acionados simultaneamente e o acréscimo de 1 Ampère para o funcionamento dos demais componentes eletrônicos. Assim, tem-se 9 Ampères como corrente mínima de projeto. Por decisão de projeto, se estabeleceu uma margem de segurança adicional de 30% na corrente, então, considerou-se uma corrente mínima necessária de 11,7 Ampères.

A partir destas especificações, pode-se determinar que a fonte de alimentação deve fornecer 12 Volts e 11,7 Ampères.

A fonte de alimentação escolhida foi a fonte padrão ATX utilizada em computadores, com 264 Watts de potência e saídas de tensão de +3.3V, +5V, +12V e -12V. A saída de +12V fornece correntes de até 14 Ampères, fatores que justificam a escolha da mesma (Figura 44).

Figura 44 - Fonte de alimentação utilizada no projeto.

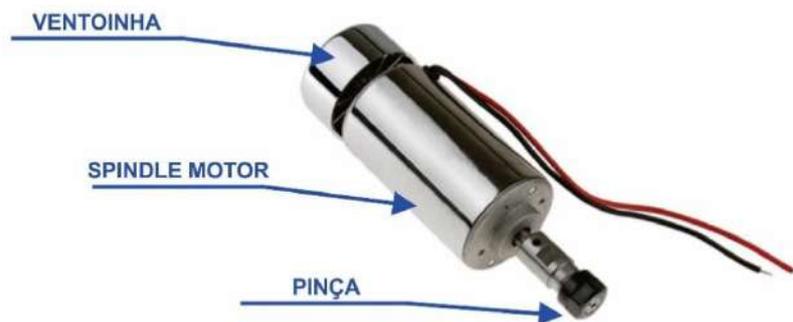


Fonte: o próprio autor.

7 FERRAMENTA

A ferramenta é a parte da máquina CNC que faz o corte da matéria prima. Em uma *router* CNC ela é basicamente um motor de alta rotação que possui em seu eixo uma haste de fixação (pinça) da ferramenta cortante chamada de fresa.

Figura 45 - Partes de um motor *spindle*.



Fonte: Adaptado de <https://goo.gl/RhszWy> (Acesso em: 20 nov. 2016).

O *spindle* usado neste trabalho foi uma micro retifica Western com velocidade variável de 8.000 a 30.000 RPM (rotações por minuto) e 130 Watts de potência, 127 Volts (Figura 46). Esta é uma opção barata e que traz resultados satisfatórios para trabalhos com madeiras leves e polímeros. Caso o usuário tenha um ciclo de trabalho mais elevado na máquina, o mesmo pode instalar um motor mais potente ou até mesmo outro tipo de ferramenta, como um canhão *laser*.

Figura 46 - Micro retifica Western usada como *spindle* da máquina CNC.



Fonte: Adaptado de <https://goo.gl/Aqrra7> (Acesso em: 20 nov. 2016).

A fresa é a peça rotativa que entra em contato com a matéria prima. Feitas para usinar o material, são construídas com uma série de dentes e gumes que desbastam a matéria prima no sentido de movimento da máquina CNC, dando o formato desejado da peça. Existem diversos tipos de fresas, como demonstrando na Figura 47, para os mais diversos usos e tipos de material, desde desbaste, furação, acabamento, gravação entre outros.

Figura 47 - Tipos de fresas.



Fonte: <https://goo.gl/oeITaf> (Acesso em: 20 nov. 2016).

Pode-se observar na Figura 48, uma fresa de topo reto (a primeira da esquerda para direita na Figura 47), usinando uma chapa de alumínio.

Figura 48 - Micro retifica usando uma peça de alumínio.



Fonte: o próprio autor.

8 FABRICAÇÃO E MONTAGEM

Depois de estabelecidos todos os parâmetros e componentes da máquina, elaborou-se a lista de componentes apresentada no Quadro 3.

Quadro 3 - Lista de componentes.

Item nº	Nome	Quantidade
01	Perfil alumínio estrutural 20x40 - 700mm	2
02	Perfil alumínio estrutural 20x40 - 750mm	2
03	Perfil alumínio estrutural 20x40 - 200mm	1
04	Chapa ACM 3mm - 500x500mm	1
05	Rolamentos 608ZZ	5
06	Rolamentos 808ZZ revestido com nylon	20
07	Correia dentada GT2 fechada 80 dentes	1
08	Correia dentada GT2 - 800mm	2
09	Correia dentada GT2 - 850mm	1
10	Barra roscada 8mm - 70mm	4
11	Barra roscada 8mm - 150mm	1
12	Porcas 8mm	35
13	Parafusos allen M4 - 25mm	8
14	Porcas M4	24
15	Parafusos allen M5 - 25mm	20
16	Parafusos allen M5 - 35mm	4
17	Porcas M5	72
18	Parafusos allen M6 - 25mm	16
19	Parafusos allen M6 - 35mm	12
20	Chapa em MDF 15mm para a mesa - 700x750mm	1
21	Arduino UNO compatível	1
22	Arduino CNC <i>Shield</i>	1
23	<i>Driver</i> de potência A4988	4
24	Motor de passo Nema 23 - 5 Kgf.cm	4
25	Fonte de alimentação padrão ATX	1
26	Micro retífica	1

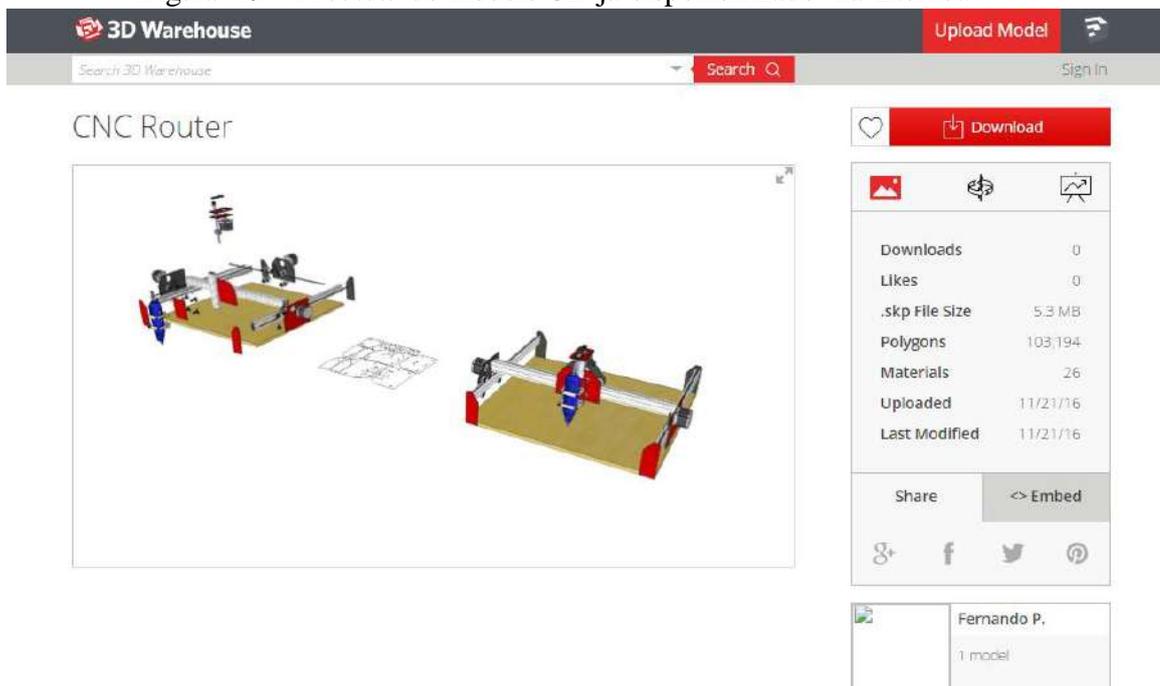
Quadro - Lista de componentes (continuação).

Item nº	Nome	Quantidade
27	Cabo manga 4 vias - 1m	1
28	Cabo manga 4 vias - 2m	3

Fonte: o próprio autor.

O diagrama de montagem foi construído a partir do próprio projeto mecânico e está disponibilizado em 3D através da biblioteca virtual *3D Warehouse* (Figura 49). O usuário pode acessar a *internet*³, efetuar o *download* do projeto gratuitamente e visualiza-lo através do *software* Sketchup. Desta forma o usuário tem um manual 3D da máquina em escala real e pode, por ele, se orientar na montagem e na construção das peças necessárias (Figura 50).

Figura 49 - Preview do modelo 3D já disponibilizado na internet.

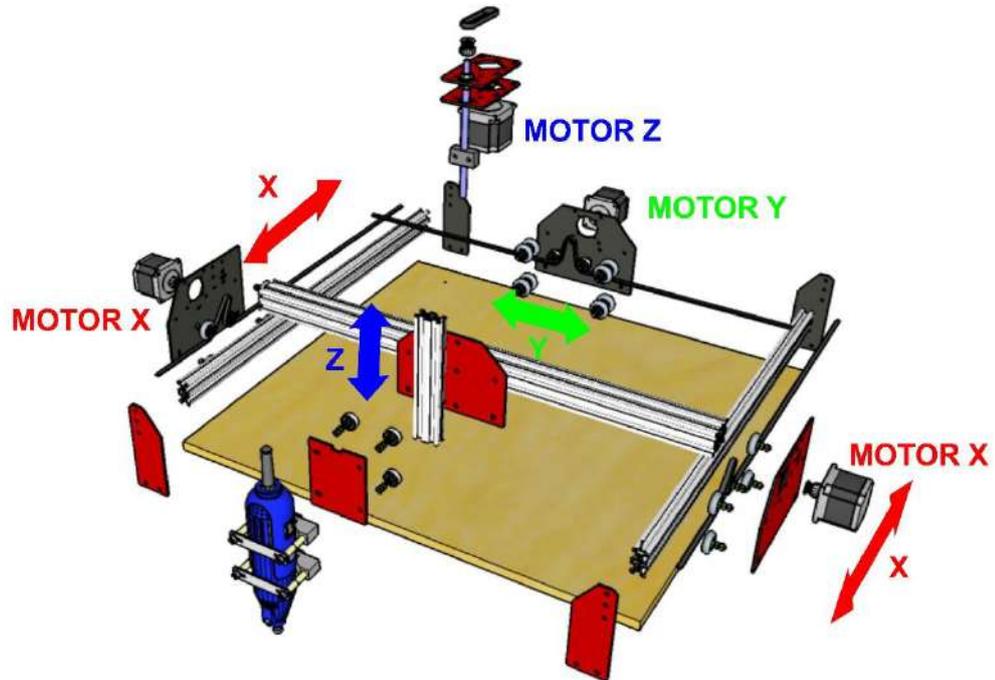


Fonte: <https://goo.gl/0yo2L0> (Acesso em: 21 nov. 2016).

A Figura 50 demonstra a vista explodida do projeto, que pode ser encontrada no arquivo disponibilizado na *web*.

³ Modelo 3D disponibilizado em <https://goo.gl/0yo2L0> (Acesso em: 24 nov. 2011) (Figura 50).

Figura 50 - Vista explodida do projeto.



Fonte: o próprio autor.

Vale observar que no projeto utilizam-se dois motores para o eixo X, enquanto nos demais eixos utiliza-se somente um motor por eixo. Isto se deve ao fato do eixo X deslocar todos os conjuntos de demais eixos e, por isso, ter seus dois trilhos de deslocamento distanciados em 750mm. A utilização de apenas um motor em um dos trilhos causaria o deslocamento inclinado do eixo.

Todos os componentes podem ser encontrados no mercado nacional e as únicas partes fabricadas foram as peças planas que servem para a fixação de motores, rolamentos e eixos. Estas peças foram primeiramente produzidas em madeira compensada a fim de verificar possíveis erros e se o projeto virtual poderia ser reproduzido fisicamente sem complicações e, posteriormente, finalizadas em chapas ACM. A Figura 51 demonstra um exemplo do processo.

Figura 51 - Etapas do projeto de fabricação.



Fonte: o próprio autor.

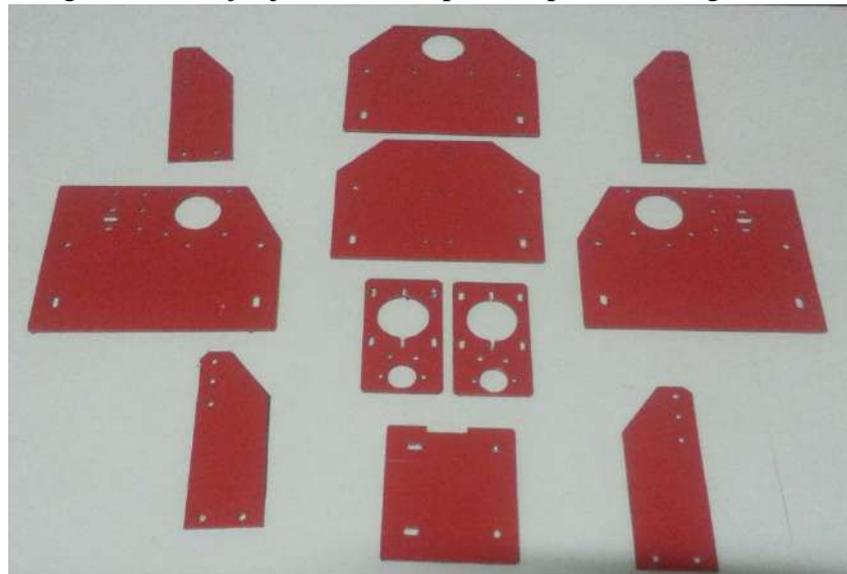
Para a confecção das peças planas, o usuário pode fabricá-las manualmente com ajuda de ferramentas como furadeiras de bancadas e serras de mesa ou, a partir das dimensões contidas no arquivo digital já disponibilizado, usiná-las em uma máquina CNC ou, como espera-se, por o projeto ser *open hardware*, encontrar outros usuários que disponibilizem para venda as peças já prontas. Neste projeto, utilizou-se a máquina *router* CNC 3DTrasnform oferecida gratuitamente pela empresa Agnus Aviação sediada em Lagoa da Prata – Minas Gerais. O trabalho de corte das peças pode ser visto na Figura 52. Demorou cerca de 20 minutos e, após o processo, as peças já estavam prontas para a montagem (Figura 53).

Figura 52 - Peças do projeto sendo usinadas.



Fonte: o próprio autor.

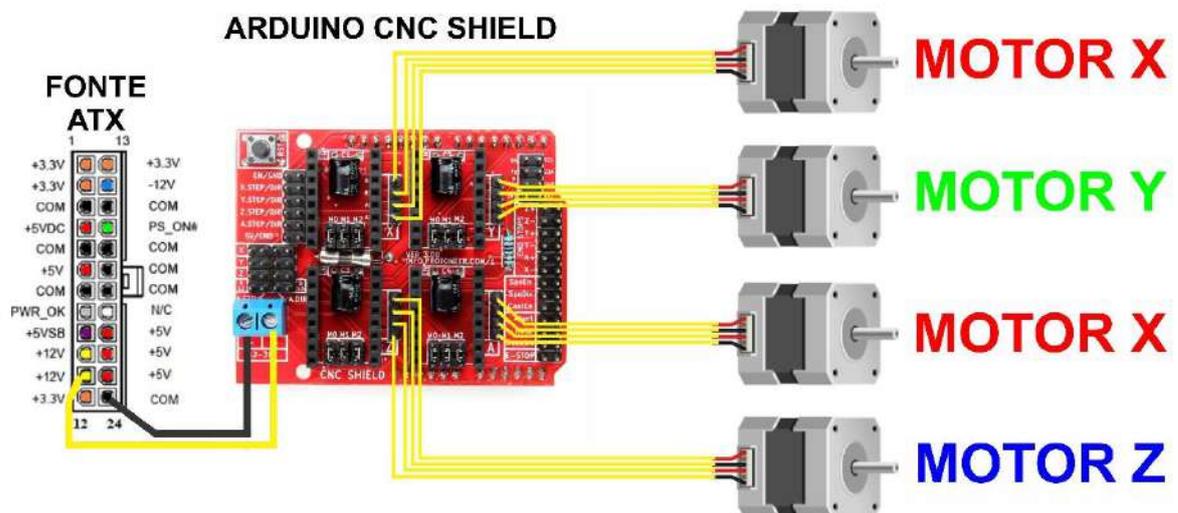
Figura 53 - Peças já usinadas e prontas para a montagem.



Fonte: o próprio autor.

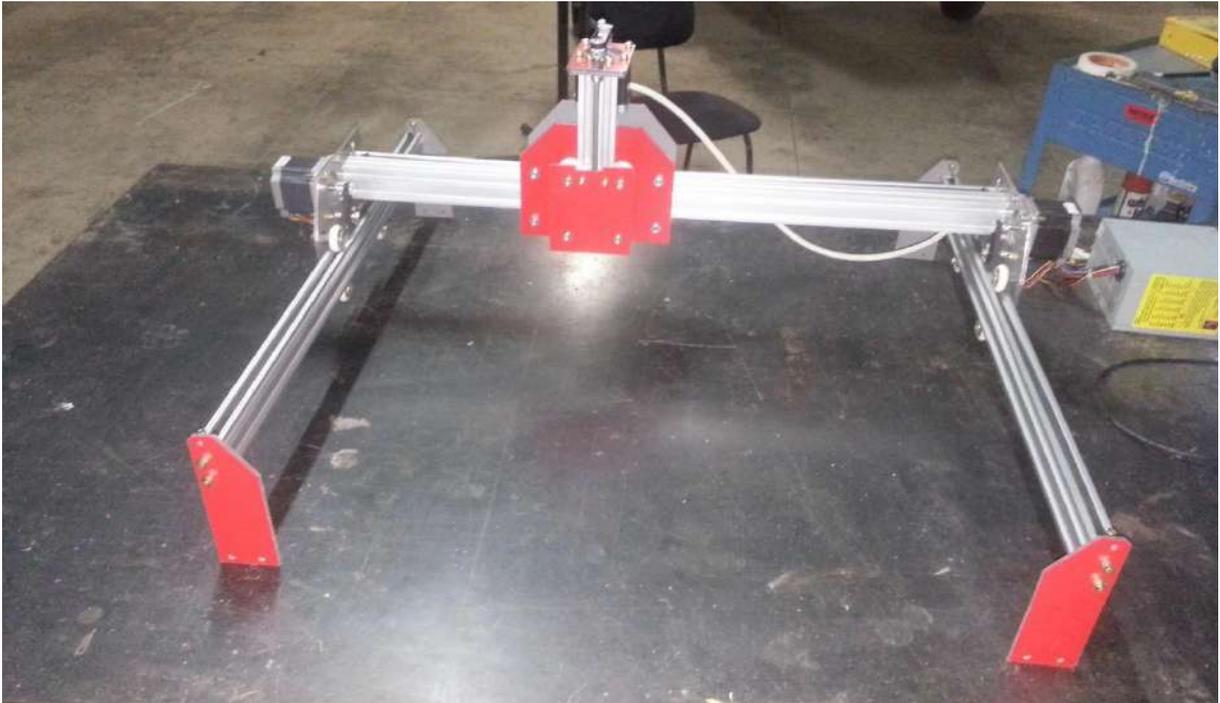
Com todos componentes em mãos, executou-se a montagem mecânica e, posteriormente, a montagem eletrônica exposta via diagrama de conexão na Figura 54 e, logo em seguida, na Figura 55, a montagem eletrônica e mecânica completa.

Figura 54 - Diagrama de conexão dos motores e Alimentação.



Fonte: o próprio autor.

Figura 55 - Montagens mecânica e eletrônica completa.



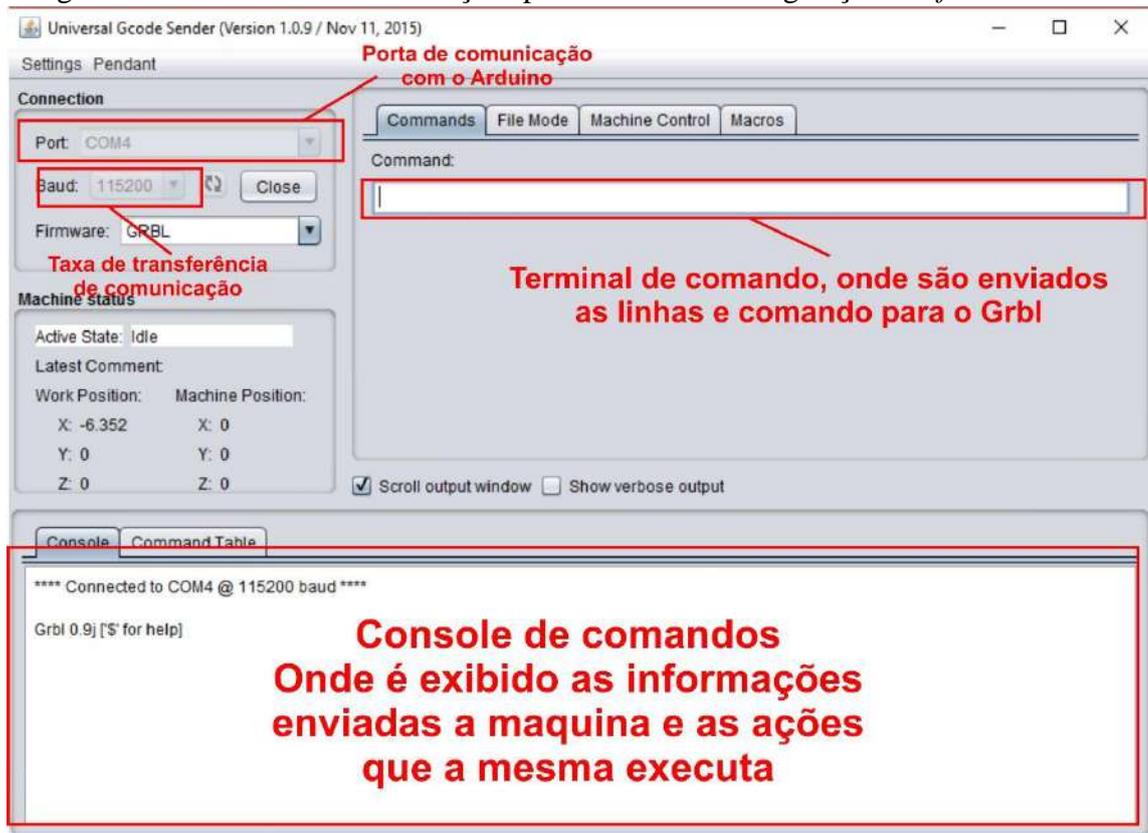
Fonte: o próprio autor.

Todos os componentes foram conectados e com a máquina finalizada, partiu-se para a próxima etapa, a configuração eletrônica e calibração.

8.1 Configuração eletrônica

Segundo o tutorial disponibilizado por Conrado (2016), instalou-se o *firmware* Grbl na placa controladora Arduino UNO. Após a instalação, iniciou-se no computador o programa *Universal GcodeSender* e estabeleceu-se conexão com a placa Arduino UNO. A Figura 56 demonstra a interface de comunicação para configuração do Grbl.

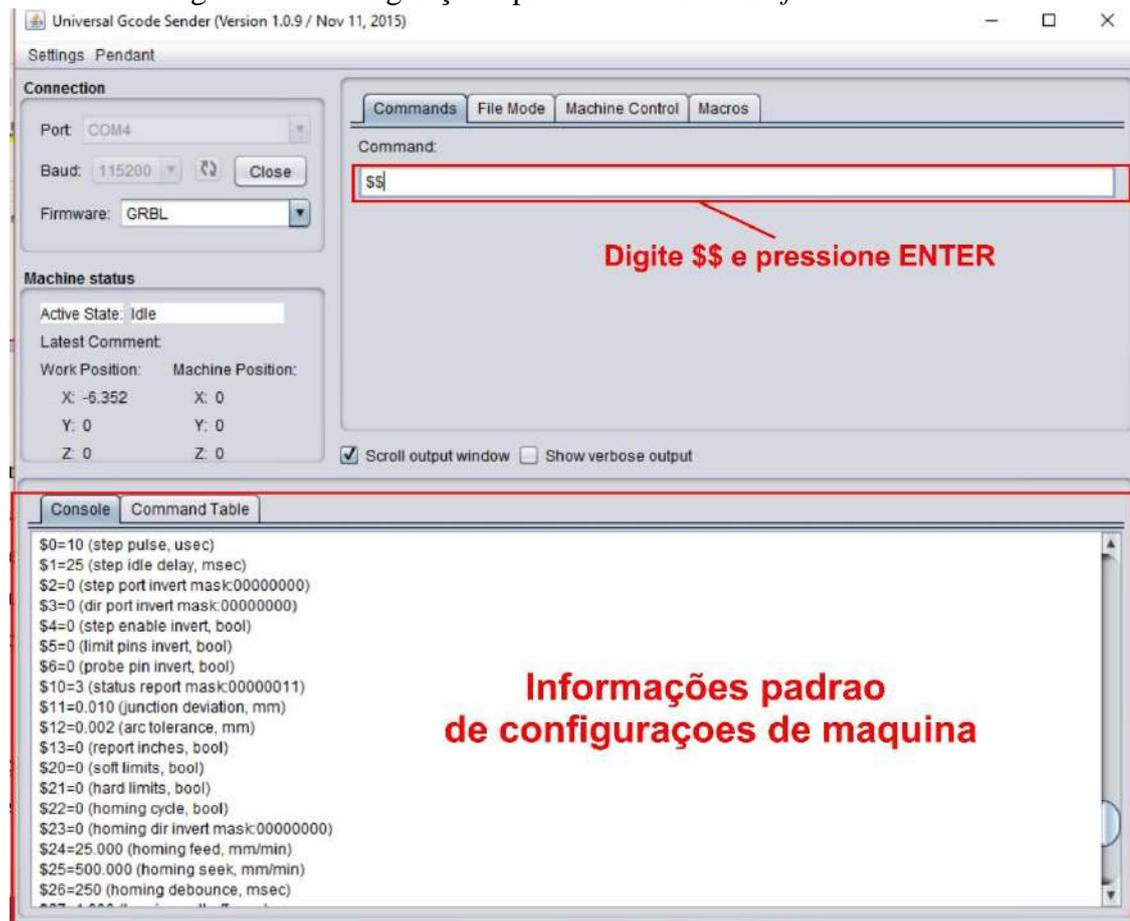
Figura 56 - Interface de comunicação para conexão e configuração do *firmware*.



Fonte: o próprio autor.

Cada máquina tem suas configurações particulares, como quantidade de passos que o motor deve dar para deslocar 1mm ou sentido do deslocamento. Estas informações devem ser gravadas no *firmware* para a correta operação da máquina. Por padrão, o Grbl já possui uma pré configuração que pode ser visualizada digitando o comando \$\$ no terminal de comando e pressionado a tecla ENTER. O padrão pode ser visualizado na Figura 57.

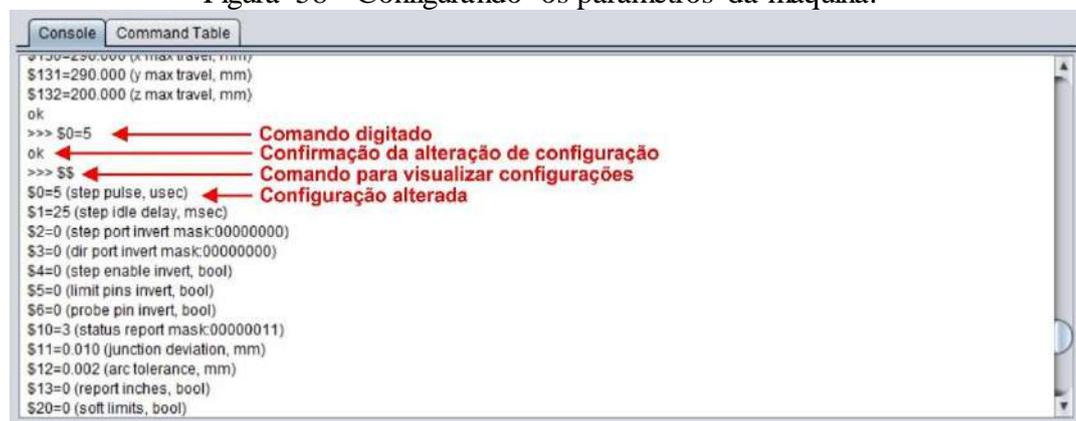
Figura 57 - Configurações pré-estabelecidas no *firmware*.



Fonte: o próprio autor.

Para a entrada de dados de configuração da máquina deve-se digitar no terminal de comando o \$ e o número correspondente a configuração desejada, por exemplo, se o usuário deseja alterar o tempo de passo para 5 microssegundos, ele deve digitar no terminal de comando: \$0=5 e pressionar ENTER. A Figura 58 mostra esta ação de configuração.

Figura 58 - Configurando os parâmetros da máquina.



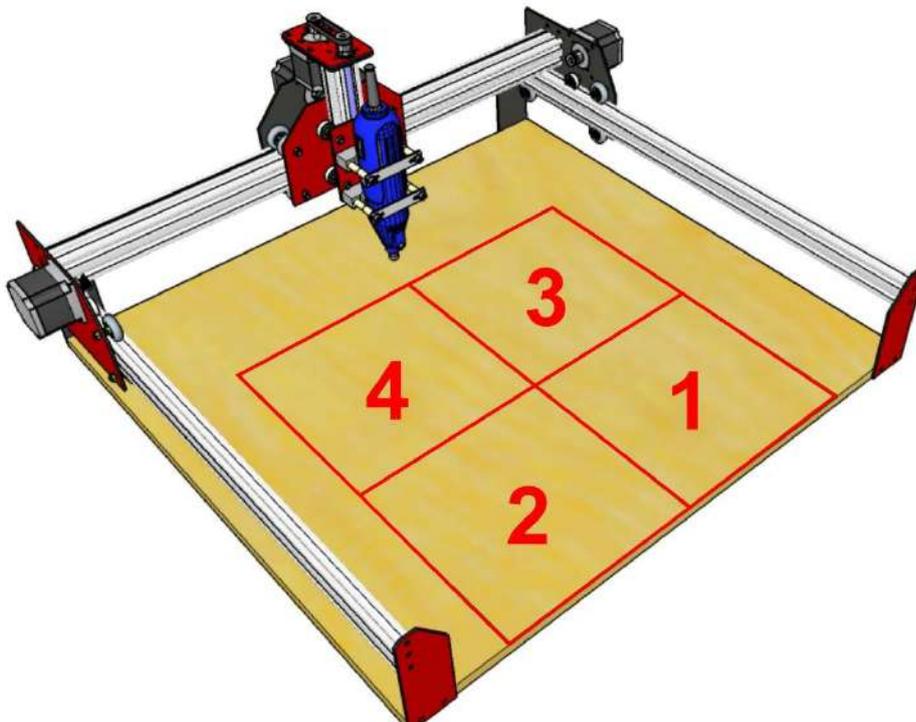
Fonte: o próprio autor

E assim, cada configuração pode ser ajustada para a máquina deste projeto. As configurações deste projeto estão dispostas no APÊNDICE A deste trabalho.

8.2 Testes

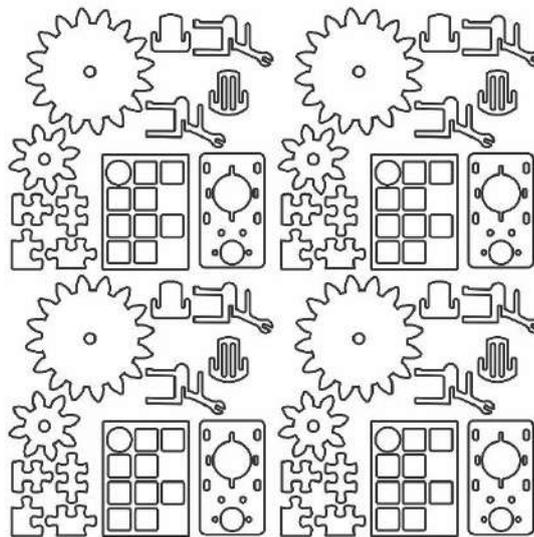
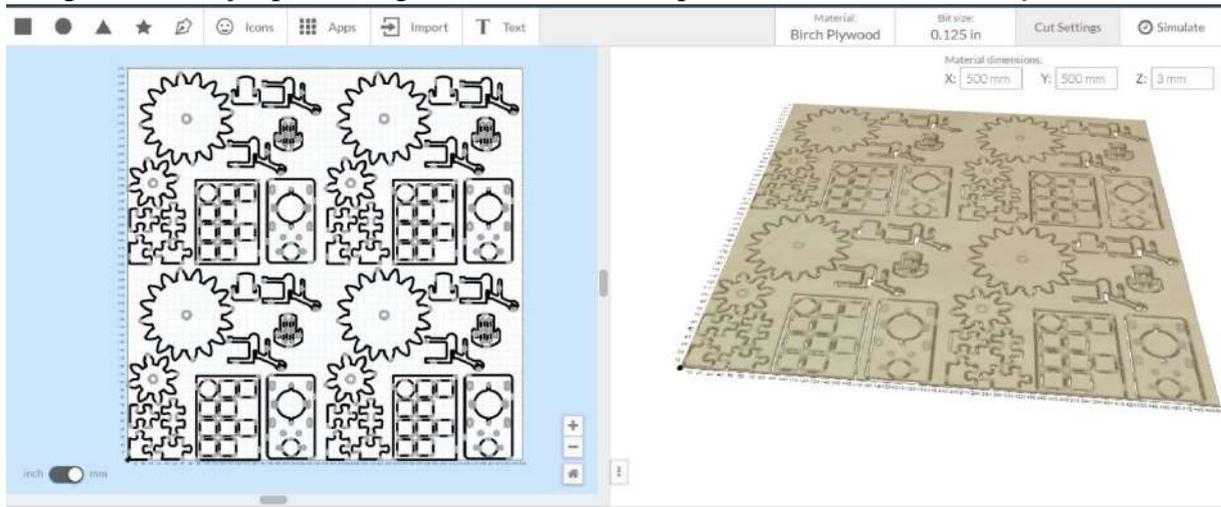
Para a validação, verificação de estabilidade, acurácia e precisão, submeteu-se a máquina CNC a testes de usinagens em madeira compensada e alumínio. Criou-se um arquivo de testes para ser usinado em madeira compensada com figuras geométricas de diferentes tamanhos e complexidade, em uma área de 250x250mm, e subdividiu-se a área de corte da máquina (500x500mm) em quatro quadrantes. Cada quadrante recebeu uma cópia do arquivo de teste. A complexidade e tamanho das peças serviram para analisar o comportamento da máquina ao desenvolver percursos retos, curvos e arredondados. Já a execução do mesmo arquivo, em quatro subáreas da máquina, serviu para observar possíveis comportamentos discrepantes entre as áreas estudadas. A Figura 59 apresenta a subdivisão da área de corte da máquina CNC e a Figura 60 apresenta o arquivo criado para o teste.

Figura 59 - Subdivisão da área de corte em quadrados de 250x250mm.



Fonte: o próprio autor.

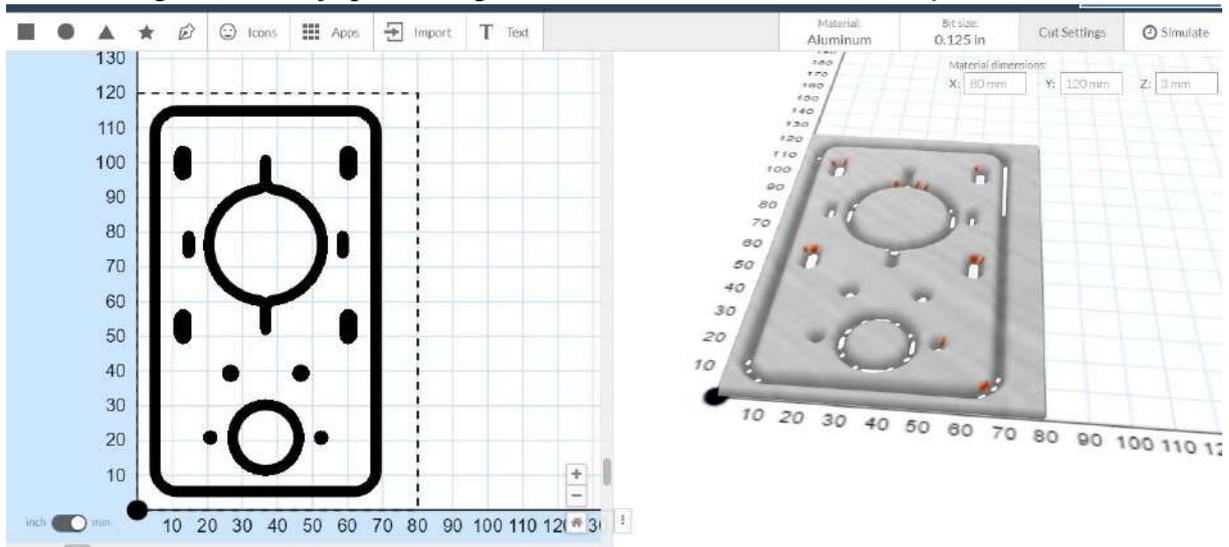
Figura 60 - Peças para usinagem em madeira compensada visualizadas no *software* Easel.



Fonte: o próprio autor.

Após a usinagem das peças do arquivo de testes, elaborou-se outro arquivo para usinagem em alumínio, a peça escolhida foi a própria base do eixo Z da máquina por conter furos e cortes de diversos tamanhos, sendo assim uma peça complexa e ideal para análise (a Figura 61 apresenta a peça escolhida).

Figura 61 - Peça para usinagem em alumínio visualizada no *software* Easel.

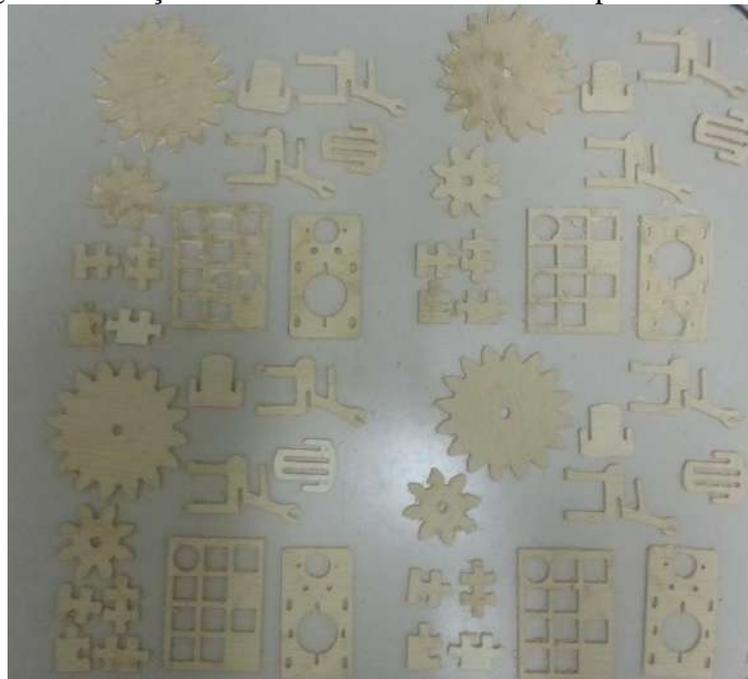


Fonte: o próprio autor.

8.2.1 Análises de peças usinadas

A Figura 62 apresenta as peças usinadas, algumas rebarbas podem ser observadas nas peças, porém são em função das características do material usinado e podem ser facilmente retiradas manualmente.

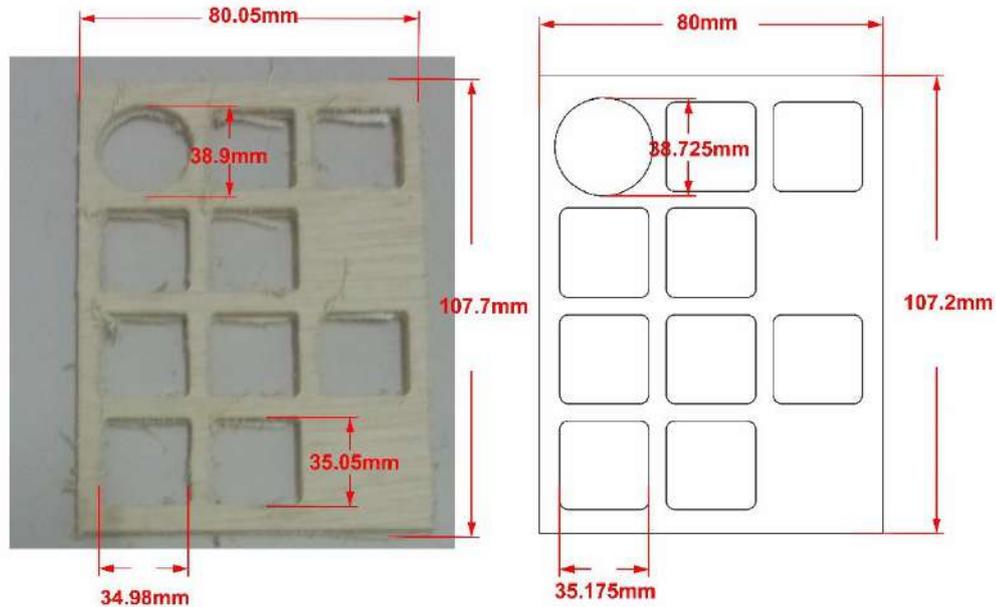
Figura 62 - Peças teste usinadas em madeira compensada.



Fonte: o próprio autor.

Não se observou outras discrepâncias entre peças de diferentes quadrantes, o que indicou conformidade em toda área útil de corte. A Figura 63 apresenta a comparação entre uma peça usinada e seu arquivo virtual.

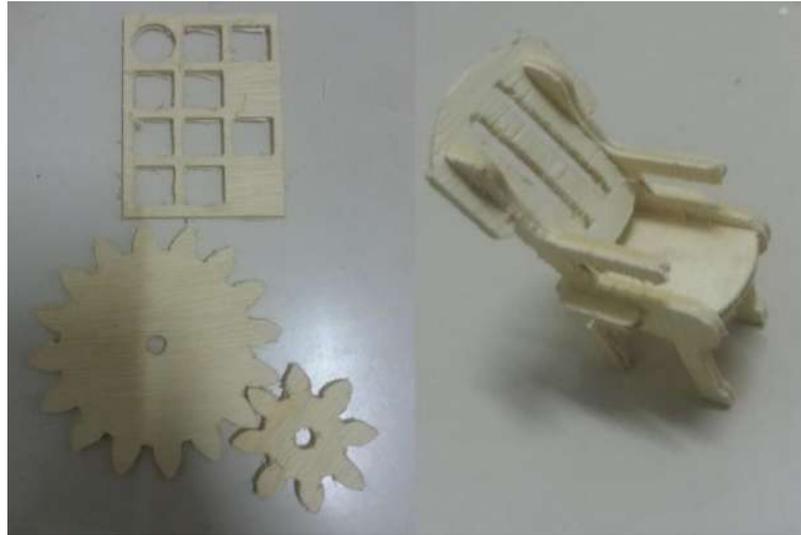
Figura 63 - Comparação entre uma peça usinada em madeira compensada e seu desenho.



Fonte: o próprio autor.

Com o teste pôde-se observar que a máquina apresentou uma variação máxima de 0,2mm entre peças iguais e variação máxima de 0,5mm entre a peça e o arquivo virtual. Valores que foram considerados aceitáveis e que não influenciaram no produto final, como pode-se observar na Figura 64.

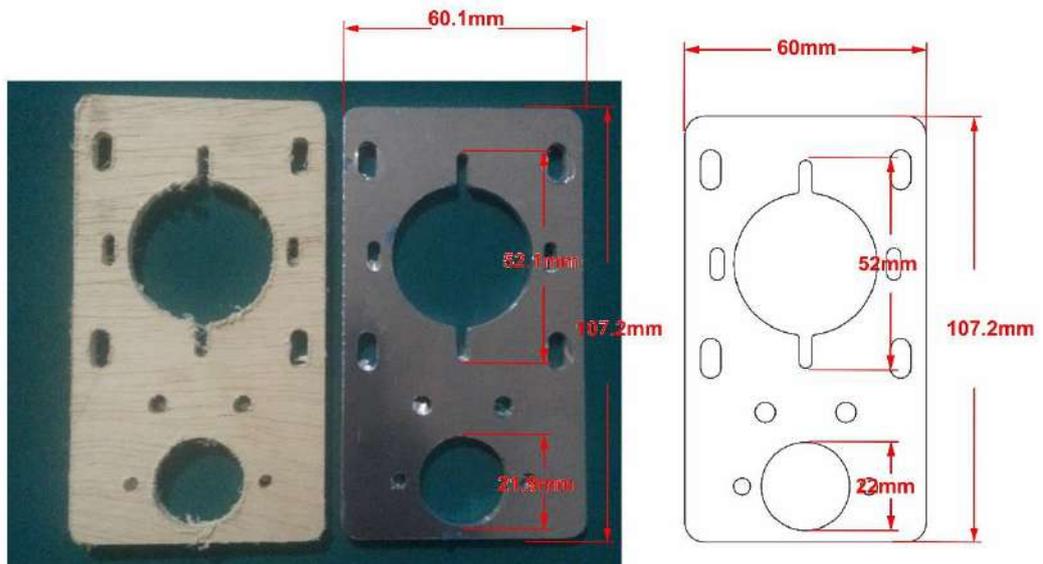
Figura 64 - Exemplo de peças usinadas em madeira compensada.



Fonte: o próprio autor.

A peça usinada em alumínio seguiu as mesmas variações observadas nas peças iguais, porém, usinadas em madeira compensada. Apresentada na Figura 65, esta peça teve variação máxima de 0,1mm quando comparada ao seu desenho virtual.

Figura 65 - Comparação ente peça usinada em alumínio e seu desenho.



Fonte: o próprio autor.

9 RESULTADOS E DISCUSSÕES

Neste capítulo são apresentados os resultados obtidos, planilha de custo e as possíveis aplicações da CNC proposta. Inicialmente, a seguir, são apresentados os custos do projeto.

9.1 Custos

As peças para a construção da máquina CNC foram adquiridas por conta própria e os valores podem ser observados no Quadro 1Quadro 4.

Quadro 4 - Custos das peças da máquina CNC.

Item nº	Nome	Quantidade	Valor Unitário (R\$)	Valor total Parcial (R\$)
01	Perfil alumínio estrutural 20x40 - 700mm	2	28,00	56,00
02	Perfil alumínio estrutural 20x40 - 750mm	2	30,00	60,00
03	Perfil alumínio estrutural 20x40 - 200mm	1	8,00	8,00
04	Chapa ACM 3mm - 500x500mm	1	60,00	60,00
05	Rolamentos 608ZZ	5	0,75	3,75
06	Rolamentos 808ZZ revestido com nylon	20	2,30	46,00
07	Correia dentada GT2 fechada 80 dentes	1	10,00	10,00
08	Correia dentada GT2 - 800mm	2	8,00	16,00
09	Correia dentada GT2 - 850mm	1	8,50	8,50
10	Barra roscada 8mm - 70mm	4	0,18	0,72
11	Barra roscada 8mm - 150mm	1	0,38	0,38
12	Porcas 8mm	35	0,25	8,75
13	Parafusos allen M4 - 25mm	8	0,17	1,36
14	Porcas M4	24	0,05	1,20
15	Parafusos allen M5 - 25mm	20	0,23	4,60

Quadro 4 - Custos das peças da máquina CNC (Continuação).

Item nº	Nome	Quantidade	Valor Unitário (R\$)	Valor total Parcial (R\$)
16	Parafusos allen M5 - 35mm	4	0,29	1,16
17	Porcas M5	72	0,08	5,76
18	Parafusos allen M6 - 25mm	16	0,27	4,32
19	Parafusos allen M6 - 35mm	12	0,33	3,96
20	Chapa em MDF 15mm para a mesa - 700x750mm	1	15,00	15,00
21	Arduino UNO compatível	1	30,00	30,00
22	Arduino CNC <i>Shield</i>	1	40,00	40,00
23	<i>Driver</i> de potência A4988	4	10,00	40,00
24	Motor de passo Nema 23 - 5 Kgf.cm	4	50,00	200,00
25	Fonte de alimentação padrão ATX	1	40,00	40,00
26	Micro retifica	1	150,00	150,00
27	Cabo manga 4 vias - 1m	1	1,00	1,00
28	Cabo manga 4 vias - 2m	3	2,00	6,00
Total				822,46

Fonte: o próprio autor.

Baseando-se no decreto 8.618 (BRASIL, 2015), que define o valor do salário mínimo nacional⁴, pode-se observar que o projeto é uma opção economicamente acessível, pelo custo equiparado ao de um salário mínimo vigente e também pela possibilidade do usuário poder fracionar a aquisição de peças de acordo com seu orçamento.

O Quadro 5 apresenta uma comparação de custos deste projeto com os valores de máquinas CNC equivalentes produzidas internacionalmente. As máquinas escolhidas possuem áreas uteis semelhantes, potência e aplicações equivalentes.

⁴ Que à época de execução desse projeto era de R\$ 880,00.

Quadro 5 - Comparação de preços ente máquinas CNC equivalentes.

Máquina	Especificações	Valor (R\$)
Desenvolvida	Área útil: 500mm (x), 500mm (y), 100mm (z). Motores de passo: Nema 23	822,46
Shapeoko 3 kit ⁵	Área útil: 406,4mm (x), 406,4mm (y), 76,2mm (z). Motores de passo: Nema 23	3.780,56 ⁶ (US\$ 1099,00) ⁷
X carve ⁸	Área útil: 500mm (x), 500mm (y), 65mm (z). Motores de passo: Nema 23	4.790,206 ⁶ (US\$1392,50) ⁷

Fonte: o próprio autor.

Pode-se observar no Quadro 5 que os custos para a produção deste projeto são inferiores aos custos de máquinas equivalentes disponíveis no mercado, reforçando que o projeto aqui desenvolvido mostra-se uma opção economicamente viável.

9.2 Facilidade de reprodução

O projeto requer a confecção mínima de componentes por parte do usuário e tem o foco no fácil acesso no mercado nacional de todos os componentes requeridos, além dos custos serem relativamente baixos. Também se preocupou com que todos os programas e componentes utilizados fossem de licença gratuita e multiplataforma, o que facilita ainda mais o acesso, tanto na produção quanto no uso.

O tempo de montagem e configuração, até que a máquina CNC fique pronta para trabalho, é demonstrado no Quadro 6.

⁵ <https://shop.carbide3d.com/collections/machines/products/shapeoko3?variant=11517165318> (Acesso em: 27 nov. 2016)

⁶ Valor sem impostos, baseado no dólar comercial cotado a R\$3,44 do dia 14 de nov. 2016.

⁷ Valor informado no *site* do fabricante sem considerar taxas de envio.

⁸ <https://www.inventables.com/technologies/x-carve/customize#750mm> (Acesso em: 27 nov. 2016).

Quadro 6 - Tempo de montagem da máquina CNC⁹.

Estagio	Tempo em horas
Montagem mecânica	3,5
Montagem eletrônica e configuração	2
Calibração	0,5
Testes	1
Ajustes pós testes e checagem	0,5
TEMPO TOTAL	7,5

Fonte: o próprio autor.

O projeto pode ser montado em menos de 8 horas devido à pouca variedade de peças e ao acesso ao projeto virtual em 3D, o que facilita a visualização da posição e localização de uma peça que o usuário tenha dúvida. Como os dados de configuração da máquina já estão acessíveis no APÊNDICE A, cabe ao usuário apenas digita-las na tela de configuração, desta forma pode-se reduzir ainda mais o tempo de produção e demanda menos conhecimento teórico para produção.

9.3 Qualidade

As peças reproduzidas pela máquina apresentaram diferenças de seus desenhos computacionais de no máximo 0,5mm, este valor se justifica por possíveis folgas entre os eixos e vibrações geradas pelos motores. Porém, pelo o baixo custo do projeto estes valores são altamente satisfatórios.

9.4 Possíveis aplicações

A máquina desenvolvida é uma opção acessível a estudantes e pesquisadores (nas universidades e/ou empresas). Ela permite a produção de placas de circuitos impressos, peças mecânicas e estruturais, além de brinquedos pedagógicos ou *kits* didáticos que podem ser construídos a partir desta CNC.

⁹ Tempo médio considerando montador de experiência mediana no manuseio de ferramentas comuns como chaves de fenda, allen e de boca.

Por ter uma estrutura robusta, eletrônica comum e grande área útil de trabalho, o projeto pode ser usado como base para uma impressora 3D, máquina inserora de componentes SMD ou máquina de gravação a *laser*.

10 CONSIDERAÇÕES FINAIS

O projeto atendeu às especificações desejadas utilizando apenas ferramentas livres, tanto na sua concepção, construção ou operação. Isto demonstra a qualidade e capacidade de ferramentas gratuitas como Arduino, Grbl e Sketchup, mostrando que é possível realizar projetos de impacto relevante com baixos orçamentos.

Com os resultados satisfatórios do projeto, visa-se como projeto futuro, desenvolver mais aplicações. Para a próxima etapa pretende-se criar uma máquina *laser* CNC para cortes de polímeros finos e papel, com possibilidade de gravação em madeira e superfícies metálicas.

Outra aplicação futura é expandir a área de trabalho útil da máquina. Esta atualização de projeto pode ser realizada com poucos ajustes, visto que utilizou-se perfis de alumínio estrutural, material que pode ser adquirido em dimensões de acordo com a demanda do projetista – e as peças planas fabricadas ainda poderão ser utilizadas sem necessidade de alteração de projeto. Tendo estas vantagens em vista, pretende-se utilizar motores de passo e *spindle* com maior torque e elaborar uma máquina com área útil de 1,2m por 1m. Estes ajustes permitirão que a máquina trabalhe com uma gama maior de materiais, propiciando a construção de peças maiores. Tais alterações permitirão que a máquina seja usada para confecção de moveis em marcenarias de pequeno porte.

Para continuar a proposta de acessibilidade, prevê-se o desenvolvimento de vídeo aulas a serem disponibilizadas gratuitamente na *web*, ensinando o passo a passo para a montagem da máquina CNC, além do desenvolvimento de um manual de operação para a mesma.

Além disso propõe-se implementações como a troca automática de ferramentas e a criação de uma interface microcontrolada que permita a leitura e envio do G-code através de *pen drives* e cartões de memórias diretamente para a máquina sem o uso de um computador conectado a mesma durante o processo de usinagem.

Ainda como trabalho futuro, prevê-se também que este projeto dê início a concepção de um Fab Lab (laboratório de fabricação), em espaço físico a ser viabilizado juntamente com o IFMG – *Campus Arcos*¹⁰. Um Fab Lab é, segundo Ginesi (2015), “um espaço em que pessoas de diversas áreas se reúnem para realizar projetos de fabricação digital de forma colaborativa”. Tal iniciativa visa disseminar regionalmente a ideia de projetos livres, contribuindo para avanços tecnológicos cada vez mais acessíveis à sociedade como um todo.

¹⁰ Mediante convênio de cooperação a ser futuramente proposto.

Por fim, reforça-se a motivação original de que esta proposta surgiu da dificuldade ao acesso de tecnologias que simplificassem a produção de novos projetos. Tendo este problema em vista, elaborou-se um dispositivo funcional e de baixo custo que suprisse a demanda de estudantes e pesquisadores. Adotando o conceito de *software* e *hardware* abertos, qualquer usuário pode modificar, melhorar ou personalizar a máquina CNC de acordo com suas necessidades. Desta forma, este projeto visa ser uma ferramenta de ideias e espera-se, com este trabalho, incentivar o desenvolvimento de novas tecnologias e difundir a ideia de que uma tecnologia somente é suficientemente inovadora quando é acessível a todos.

REFERÊNCIAS

BELL, C. **Maintaining and Troubleshooting Your 3D Printer**. New York: Apress, 2014.

BRASIL. Decreto nº 8.618, de 29 de dezembro de 2015. Regulamenta a Lei nº 13.152, de 29 de julho de 2015, que dispõe sobre o valor do salário mínimo e a sua política de valorização de longo prazo. Diário Oficial, Brasília, DF, 30 dez. 2015. Seção 1, p. 5.

CONRADO, R. Grbl v0.9j O que é? Para que serve? Como configurar?, **Atividade Maker**, [S.l.], v. online, 2016. Disponível em: < <http://atividademaker.com.br/grbl-v09j> > Acesso em: 20 nov. 2016

DS4. **I Servo**. [S.l: s.n., 200-]. Disponível em: < <http://ds4.com.br/fresadora-router/i-servo/>> Acesso em: 20 nov. 2016.

FACHIM, A. **Projeto de fresadora CNC com plataforma livre Arduino**. Relatório (Graduação em Engenharia de Controle e Automação) – Universidade Federal do Rio Grande do Sul. Escola de Engenharia. Curso de Engenharia de Controle e Automação, 2013

GENG, H. **Manufacturing Engineering Handbook, Second Edition**. New York: McGraw Hill Education, 2015.

GINESI, C. Saiba como funcionam os laboratórios de inovação – fab labs – espalhados pelo Brasil. E como você pode usá-los. **Projeto Draft**, [S.l.], v. online, abr. 2015. Disponível em: <<http://projetodraft.com/saiba-como-funcionam-os-laboratorios-de-inovacao-fab-labs-espalhados-pelo-brasil-e-como-voce-pode-usa-los/>> Acesso em: 20 nov. 2016.

GITHUB. About Grbl. **GitHub wiki**, [S.l.], v. online, ago. 2015. Disponível em: <<https://github.com/grbl/grbl/wiki>> Acesso em: 20 nov. 2016.

GITHUB. Using Grbl. **GitHub wiki**, [S.l.], v. online, ago. 2016. Disponível em: <<https://github.com/grbl/grbl/wiki/Using-Grbl>> Acesso em: 20 nov. 2016.

LYRA, P, V, A. **Desenvolvimento de uma Máquina Fresadora CNC Didática**. Relatório (Graduação em Engenharia de Controle e Automação) – Faculdade de Tecnologia, Universidade de Brasília, Brasília, 2010.

MADISON, J. **CNC Machining Handbook: Basic Theory, Production Data, and Machining Procedures**. Norwalk: Industrial Press Inc., 1996.

PEARCE, J. M. Building Research Equipment with Free, Open-Source Hardware. **Science Magazine**, v. 337, n. 6100, p. 1303–1304, 2012.

POST-PRO. **What does Post-Pro provide?** [S.l: s.n., 200-]. Disponível em: < <http://www.post-pro.com/Pages/default.aspx> > Acesso em: 20 nov. 2016.

ROMI. **CNC ROMI Centur 30D**. [S.l: s.n., 200-]. Disponível em: <<http://www.romi.com/produtos/linha-centur/>> Acesso em: 20 nov. 2016.

SCHNEIDER. **NEMA size 23 1.8° 2-phase stepper motor**. [S.l: s.n., 200-]. Disponível em: < <http://motion.schneider-electric.com/downloads/quickreference/NEMA23.pdf> > Acesso em: 20 nov. 2016.

SEAMES, W, S. **Computer Numerical Control: Concepts and Programming**. Stanford: Cengage Learning, 2001.

SILVA, L, A; ZANIN, M, J. **Montagem de fresa CNC**. Relatório (Graduação em Tecnologia em Manutenção Industrial) – Faculdade de Tecnologia, Universidade Tecnológica Federal do Paraná Campus Medianeira, Medianeira, 2011.

SMG3D. **Objet1000 3D Printer**. [S.l: s.n., 200-]. Disponível em: <http://www.smg3D.co.uk/entertainment/objet1000_3D_printer > Acesso em: 20 nov. 2016.

SMID, P. **CNC Programming Handbook: A Comprehensive Guide to Practical CNC Programming**. Norwalk, Industrial Press Inc., 2003.

SOLUÇÕES INDUSTRIAIS. **Descrição do produto - Fresadora CNC FC1000**. [S.l: s.n., 200-]. Disponível em: < <http://www.solucoesindustriais.com.br/empresa/metal-mecanica-e-usinagem/fobrasa-comercio-de-maquinas-ltda/produtos/maquinas-ferramenta/fresadora-cnc-fc1000> > Acesso em: 20 nov. 2016.

APÊNDICE A - PARÂMETROS DE CONFIGURAÇÃO DA MÁQUINA CNC

\$0=10

\$1=25

\$2=0

\$3=6

\$4=0

\$5=0

\$6=0

\$10=3

\$11=0.020

\$12=0.002

\$13=0

\$20=0

\$21=0

\$22=0

\$23=1

\$24=50.000

\$25=635.000

\$26=250

\$27=1.000

\$100=80

\$101=80

\$102=320

\$110=8000.000

\$111=8000.000

\$112=500.000

\$120=50.000

\$121=50.000

\$122=50.000

\$130=500.000

\$131=500.000

\$132=100.000

ANEXO 1 - APOSTILA PARA CONFIGURAÇÃO DO *FIRMWARE GRBL*
(CONRADO, 2016)



grbl

v0.9j

O que é? Para que serve? Como configurar?

```
$0=10 (step pulse, usec)
$1=25 (step idle delay, msec)
$2=0 (step port invert mask:00000000)
$3=6 (dir port invert mask:00000110)
$4=0 (step enable invert, bool)
$5=0 (limit pins invert, bool)
$6=0 (probe pin invert, bool)
$10=3 (status report mask:00000011)
$11=0.020 (junction deviation, mm)
$12=0.002 (arc tolerance, mm)
$13=0 (report inches, bool)
$20=0 (soft limits, bool)
$21=0 (hard limits, bool)
$22=0 (homing cycle, bool)
$23=1 (homing dir invert mask:00000001)
$24=50.000 (homing feed, mm/min)
$25=635.000 (homing seek, mm/min)
$26=250 (homing debounce, msec)
$27=1.000 (homing pull-off, mm)
$100=314.961 (x, step/mm)
$101=314.961 (y, step/mm)
```



RODRIGO CONRADO

Quem somos

Atividade Maker é um projeto com o propósito de divulgar o estilo de vida "faça você mesmo" (DIY – Do It Yourself), aqui vamos divulgar projetos de Marcenaria, Eletrônica, Robótica, Arduino, Impressão 3D, CNC entre outros, Somos apaixonados pelo estilo Faça Você Mesmo.

Atividade Maker - Construa o Seu Mundo, Você Pode!

Como entrar em contato?

Você pode entrar em contato pelos seguintes meios:



www.atividademaker.com.br



www.facebook.com/atividademaker



www.youtube.com/c/AtividadeMakerOficial



contato@atividademaker.com.br



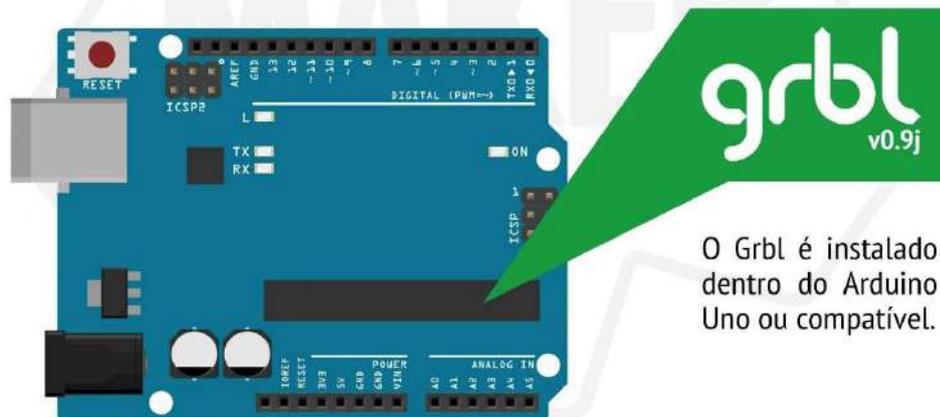


Sobre o Grbl

O Grbl é grátis, open source (código aberto), software de alto desempenho para controlar o movimento de máquinas que se movem, que fazem as coisas, ou que fazem as coisas se moverem, e será executado diretamente em um Arduino.

Muitas impressoras 3D de código aberto usam Grbl em seus núcleos, sendo adaptado para uso em centenas de projetos, incluindo máquinas de corte a laser, CNC entre outros. Devido ao seu desempenho, simplicidade e requisitos de hardware, o Grbl tornou-se um fenômeno de código aberto.

Em 2009, Simen Svale Skogsrud (<http://bengler.no/grbl>) agradeceu a comunidade de código aberto, escrevendo e liberando as primeiras versões do Grbl a todos (inspiradas no Arduino Gcode Interpreter por Mike Ellery). Desde 2011, Grbl está a avançar como um projeto de código aberto voltada para a comunidade sob a liderança pragmática da Sungeun K. Jeon Ph.D. (@chamnit).



O Grbl é instalado dentro do Arduino Uno ou compatível.

Placa Arduino Uno ou Compatível



grbl

v0.9j

Primeiros Passos - Arduino

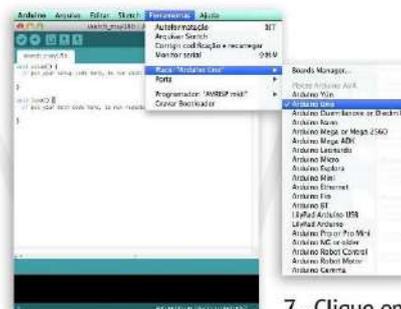
Vamos instalar o Grbl no Arduino através da IDE do Arduino, para isso você precisa acessar o site www.arduino.cc e efetuar o download da IDE para o seu sistema operacional.



- 1 - Acessar www.arduino.cc;
- 2 - Clicar em Download;
- 3 - Clicar no seu sistema operacional para efetuar o download;
- 4 - Faça a instalação conforme seu sistema;



- 5 - Abra a IDE Arduino;
- 6 - Clique em **Ferramentas > Porta** e **selecione a porta serial**.



- 7 - Clique em **Ferramentas > Placa** e **selecione Arduino Uno**;





Primeiros Passos - Grbl

Após a instalação da IDE Arduino e suas configurações você vai precisar efetuar o download do Grbl para que possa ser enviado para a memória do Arduino.



1 - Acesse: <https://github.com/grbl/grbl> e faça o download do Grbl clicando no botão **Download ZIP**;

2 - Descompactar o arquivo em um local de sua preferência;

3 - Abra o IDE Arduino;

4 - Clique em **Sketch > Include Library > add .Zip Library** e selecione a pasta que você descompactou o Grbl;

5 - Conecte o cabo USB no seu computador e na sua placa Arduino;

6 - Clique em **Arquivos > Exemplos > e escolha o grbl > grblUpload**;

7 - Depois clique para transferir o código para o Arduino.





Primeiros Passos - Terminal Serial

Pronto agora você já tem o Grbl instalado dentro do Arduino, você já pode fechar a IDE do Arduino pois não iremos mais utilizá-la.

Agora é preciso escolher um **Terminal Serial** para que possamos utilizar o Grbl.

Você encontra uma lista com várias opções no seguinte endereço:

<https://github.com/grbl/grbl/wiki/Using-Grbl>

Iremos utilizar em nosso tutorial o terminal serial, **Universal Gcode Sender** na versão **1.0.9**.

Você pode efetuar o download no seguinte endereço;

<https://github.com/winder/Universal-G-Code-Sender>



- 1 - Conecte o cabo USB no seu computador e em seu Arduino Uno;
- 2 - Clique em **Port** e selecione a porta serial;
- 3 - Clique em **Baud** e selecione 115200;
- 4 - Clique em Open;
- 5 - Se tudo ocorreu bem você deve ver a seguinte mensagem:

Grbl 0.9j ['\$' for help]





Iniciando o Grbl v0.9j

Primeiro, conecte-se ao Grbl usando o terminal serial de sua escolha, em seguida selecione a porta serial onde seu arduino está instalado e defina a taxa de transmissão (Baud) para 115200, clique em Open para se conectar.

Uma vez conectado você deverá visualizar o aviso, que se parece com isso:

```
Grbl 0.9j ['$' for help]
```

Configurando o Grbl v0.9j

Para visualizar as configurações, digite na linha de comando **\$\$** e pressione Enter.

O Grbl deve responder com uma lista das configurações atuais do sistema.

Todas essas configurações são persistentes e mantidas na memória EEPROM, então se você desligar seu Arduino os dados serão mantidos na memória e na próxima vez que você ligar eles estarão do mesmo jeito que você deixou.





Configurações do Sistema

Você pode alterar qualquer valor de configuração digitando na linha de comando o \$ mais o número do parâmetro que você quer alterar e informar o valor a ser inserido.

\$X = valor (Salva o valor no Grbl)

O \$x=valor salva ou altera um valor de configuração no Grbl, o que pode ser feito manualmente, enviando este comando quando conectado ao Grbl através de um programa terminal serial.

Para alterar manualmente, por exemplo, a opção \$0 para 10us você deve digitar:

\$0 = 10 (depois dar um enter)

Se tudo correu bem, o Grbl vai responder com um 'ok' e essa configuração é armazenada na memória EEPROM e ficará lá para sempre ou até que seja alterada novamente. Você pode verificar se o Grbl recebeu e armazenou sua nova configuração corretamente, digitando \$\$ para ver as configurações do sistema novamente.

```

$0=10 (step pulse, usec)
$1=25 (step idle delay, msec)
$2=0 (step port invert mask:00000000)
$3=6 (dir port invert mask:00000110)
$4=0 (step enable invert, bool)
$5=0 (limit pins invert, bool)
$6=0 (probe pin invert, bool)
$10=3 (status report mask:00000011)
$11=0.020 (junction deviation, mm)
$12=0.002 (arc tolerance, mm)
$13=0 (report inches, bool)
$20=0 (soft limits, bool)
$21=0 (hard limits, bool)
$22=0 (homing cycle, bool)
$23=1 (homing dir invert mask:00000001)
$24=50.000 (homing feed, mm/min)
$25=635.000 (homing seek, mm/min)
$26=250 (homing debounce, msec)
$27=1.000 (homing pull-off, mm)
$100=314.961 (x, step/mm)
$101=314.961 (y, step/mm)
$102=314.961 (z, step/mm)
$110=635.000 (x max rate, mm/min)
$111=635.000 (y max rate, mm/min)
$112=635.000 (z max rate, mm/min)
$120=50.000 (x accel, mm/sec^2)
$121=50.000 (y accel, mm/sec^2)
$122=50.000 (z accel, mm/sec^2)
$130=225.000 (x max travel, mm)
$131=125.000 (y max travel, mm)
$132=170.000 (z max travel, mm)

```





Parâmetros do Grbl v0.9j e seus significados

NOTA: A numeração de Configurações mudou desde a v0.8c.

\$0 = Tamanho do Pulso em microsegundos

Os drivers controladores de motor são projetados para entender o passo com um determinado tamanho de pulso. Verifique o data sheet ou apenas faça testes com alguns valores. Você precisa de pulsos curtos mas que sejam reconhecidos pelo seu driver. Se os pulsos forem muito longos você pode ter problemas ao executar o sistema em altas taxas de alimentação de pulsos, pois os pulsos de passos podem começar a se sobreporem uns aos outros. Recomendamos algo em torno de 10 microsegundos, que é o valor padrão.

\$1 = Atraso da Inatividade do Motor em milisegundos

Toda vez que seus motores completam um movimento e chegam a uma parada, o Grbl vai atrasar o desativamento dos motores por este valor. Ou você pode sempre manter seus eixos habilitados (alimentados de forma a manter a posição), definindo esse valor para o máximo de 255 milisegundos. Mais uma vez apenas para fixar você pode manter todos os eixos sempre habilitados pela configuração \$1=255.

O atraso da inatividade do motor é o tempo de duração que o Grbl irá manter os motores travados antes de desativar. Dependendo do sistema, você pode definir isso para zero e desativá-lo. Em outros, você pode precisar de 25 - 50 milisegundos para garantir que seu eixo chegue a uma parada completa antes de desativá-lo.





\$2 = Máscara para Inversão do Sinal dos Passos

Este item faz a inversão do sinal de pulso. Por padrão, um sinal começa de nível baixo e vai a alto, depois de um tempo definido em **\$0**, o valor volta para baixo, até o próximo evento do pulso. Quando invertido, o comportamento do pulso muda de alto para baixo depois de um tempo definido em **\$0** volta para alto.

A maioria dos usuários não precisará usar essa configuração, mas isso pode ser útil para determinados controladores CNC que têm exigências peculiares.

Esta máscara para inversão dos passos é um valor que armazena os eixos para serem invertidos com bit. Você realmente não precisa entender completamente como funciona. Você só precisa digitar o valor de configurações para os eixos que deseja inverter. Por exemplo, se você quiser inverter os eixos X e Z, você irá enviar \$2=5.

Valor	Máscara	Inverter X	Inverter Y	Inverter Z
0	00000000	Não	Não	Não
1	00000001	Sim	Não	Não
2	00000010	Não	Sim	Não
3	00000011	Sim	Sim	Não
4	00000100	Não	Não	Sim
5	00000101	Sim	Não	Sim
6	00000110	Não	Sim	Sim
7	00000111	Sim	Sim	Sim





\$3 = Máscara para Inversão da Direção

Esta configuração inverte o sinal de sentido para cada eixo. Por padrão, o Grbl assume que os eixos vão se mover em uma direção positiva quando o sinal de sentido é baixo, e uma direção negativa quando o sinal é alto. Muitas vezes, os eixos não se movem desta forma em algumas máquinas.

Esta definição irá inverter o sinal de sentido dos eixos.

Esta máscara para inversão da direção funciona exatamente como a máscara para inversão dos passos. Para configurar essa opção, você só precisa enviar o valor para os eixos que deseja inverter. Use a tabela anterior. Por exemplo, se quiser inverter apenas a direção do eixo Y, você envia \$3=2

\$4 = Inverter o Enable do Controlador

Por padrão, o enable desativa os motores com nível alto e ativa com nível baixo.

Se você precisar inverter essa condição apenas digite \$4=1 se quiser deixar como padrão deixe \$4=0 (Pode ser necessário desligar a alimentação do arduino e ligar novamente para carregar as mudanças).





\$5 = Inverter os Limites

Por padrão, os limites estão normalmente em alta com resistência de pull-up interno do Arduino. Quando um limite é baixo, o Grbl interpreta isso como disparado. Para o comportamento oposto, basta inverter os limites digitando \$5=1. Desativar com \$5=0. (Pode ser necessário desligar a alimentação do arduino e ligar novamente para carregar as mudanças).

NOTA: Se inverter seu limite, você precisará de um resistor pull-down externo para todos os limites para evitar a sobrecarga e fritá-los.

\$6 = Inverter o Probe

Por padrão, o probe é realizado normalmente em alta com resistência de pull-up interno do Arduino. Quando o probe é baixo, o Grbl interpreta isso como disparado. Para o comportamento oposto, basta inverter o probe pela digitação de \$6=1. Desativar com \$6=0. (Pode ser necessário desligar a alimentação do arduino e ligar novamente para carregar as mudanças).

NOTA: Se você inverter o probe, você precisará de um resistor pull-down externo com o probe para evitar a sobrecarga e fritá-lo.





\$10 = Relatório de Status

Esta configuração determina que o Grbl retorne em tempo real um relatório de status. Por padrão o Grbl irá enviar de volta o seu estado de execução (não pode ser desligado), a posição da máquina, e do trabalho (posição da máquina com coordenadas as compensações e outras compensações aplicadas). Três recursos de relatórios adicionais estão disponíveis, que são úteis para interfaces ou usuários que criam suas máquinas, que incluem o buffer serial RX, o uso do bloco do buffer, e os estados limites (como alto ou baixo, mostrado na ordem ZYX).

Para configurá-los, use a tabela abaixo para determinar quais os dados que você gostaria que o Grbl enviasse de volta. Some os valores dos tipos que você gostaria de ver nos relatórios de status. Este é o valor que você usa para enviar para o Grbl. Por exemplo, se você precisa de posição de máquina e de trabalho, some os valores 1 e 2 e envie ao Grbl \$10=3. Ou, se você precisa da posição da máquina somente e estado dos limites, some os valores 1 e 16 e envie ao Grbl \$10=17.

Em geral, é melhor utilizar essa configuração somente se necessário pois eleva a utilização de recursos desnecessariamente.

Tipo de Relatório	Valor
Posição da Máquina	1
Posição de Trabalho	2
Buffer	4
RX Buffer	8
Limites	16





\$11 = Desvio de Junções

O desvio de junção é usado pelo sistema que controla a aceleração para determinar quão rápido ele pode mover-se através de junções de segmentos gerados pelo programa g-code. Por exemplo, se o seu programa g-code tem uma curva acentuada de 10 graus chegando e a máquina está se movendo na velocidade máxima, esta definição ajuda a determinar o quanto a máquina precisa desacelerar para ir com segurança sem perder passos.

Calcular isso é um pouco complicado, mas em geral, valores mais elevados dão movimentos mais rápidos, enquanto aumenta o risco de perder passos e posicionamento. Os valores mais baixos fazem com que o sistema que controla a aceleração seja mais cuidadoso levando a máquina a se mover mais lentamente. Então se você quer que sua máquina faça as curvas mais rapidamente você deve aumentar esse valor para acelerar o movimento.

\$12 = Tolerância de Arco em milímetros

O Grbl faz círculos G2 e G3, subdividindo-os em linhas menores, tal que a precisão do traçado nunca seja inferior a este valor. Você provavelmente nunca terá de ajustar esta definição, uma vez que 0,002 milímetros está bem abaixo da precisão da maioria das máquinas CNC.

Mas se você achar que seus círculos são realizados lentamente, ajuste essa configuração. Os valores mais baixos vão dar maior precisão, mas pode levar a problemas de desempenho, os valores mais elevados levam a uma precisão inferior, mas pode acelerar o desempenho.





\$13 = Relatório em Polegadas

O Grbl tem um recurso de relatório de posicionamento em tempo real para fornecer um feedback ao usuário de onde a máquina está exatamente nesse momento, bem como, os parâmetros para coordenar as compensações e probe. Por padrão, ele é definido para relatar em mm, mas enviando um comando $\$13=1$, o relatório será exibido em polegadas. $\$13=0$ para voltar para mm.

\$20 = Limites Através do Software

Os limites através do software são um recurso de segurança para ajudar a prevenir que sua máquina tente se deslocar além dos limites disponíveis de curso, fazendo com que bata ou quebre alguma coisa. Ele funciona por saber os limites máximos de cada eixo e sabendo a posição atual de trabalho da máquina, quando um novo código g-code é enviado para o Grbl, ele vai verificar se é possível efetuar o movimento sem causar acidentes excedendo os limites da máquina. Configurando os limites de software o Grbl emitirá um bloqueio imediato de avanço desligando todos os movimentos e exibindo um alarme indicando o problema.

NOTA: Limites através do software requer que homing seja habilitado e as configurações máximas de cada eixo também, porque o Grbl precisa saber onde ele está. $\$20=1$ para habilitar, e $\$20=0$ para desabilitar.





\$21 = Limites de Hardware

Limites de hardware tem basicamente a mesma função que limites de software, mas usam chaves fim de curso para controlar de forma física os movimentos. Basicamente você conecta algumas chaves (mecânicas, magnéticas ou ópticas) perto do final do curso de cada um dos eixos, ou onde quer que você ache necessário que o movimento seja interrompido. Quando o interruptor dispara, ele irá parar imediatamente todo o movimento e entrar em modo de alarme, o que o obriga a você verificar a sua máquina.

Para utilizar os limites de hardware com o Grbl, os limites são mantidos em nível alto com um resistor pull-up interno, então tudo que você tem a fazer é colocar um interruptor normalmente aberto com o terra e definir limite de hardware com $\$21=1$. (Desativar com $\$21=0$) Aconselhamos a tomar medidas de prevenção de interferência elétrica. Se você quiser um limite para ambas as extremidades do curso do eixo, ligue os fios em paralelo assim qualquer um deles irá acionar o limite.

Tenha em mente que um evento de limite de hardware é considerado um evento crítico, onde os motores param imediatamente. O Grbl não terá nenhum feedback sobre onde e qual o posicionamento da sua máquina e vai entrar em um modo de loop infinito ALARME, dando-lhe a oportunidade de verificar sua máquina e forçá-lo a redefinir o Grbl. Lembre-se que é puramente um recurso de segurança.





\$22 = Ciclo de Homing

Para aqueles que estão iniciados no mundo do CNC, o ciclo de homing é usado para localizar uma posição conhecida e consistente em uma máquina cada vez que você iniciar o seu Grbl entre as sessões. Em outras palavras, você sabe exatamente onde você está em um determinado momento, de cada vez. Digamos que você começa a usinagem de algo ou está prestes a iniciar a próxima etapa em um trabalho e você precisa parar, quando reiniciar o Grbl ele não terá idéia de onde ele estava. Se você tem homing, você tem sempre o ponto de referência zero da máquina, a partir daí o que você tem a fazer é executar o ciclo de homing e retomar de onde parou.

Para configurar o ciclo de homing para o Grbl, você precisa ter interruptores de limite em uma posição fixa, ou então o seu ponto de referência fica confuso. Normalmente, eles são configurados no ponto mais distante em X+, Y+, e Z+ de cada um dos eixos. Por padrão, no ciclo de homing o Grbl move o eixo Z positivo primeiro para limpar a área de trabalho e, em seguida, move os eixos X e Y ao mesmo tempo no sentido positivo.

Quando o homing está habilitado o Grbl vai bloquear todos os comandos g-code até que você execute um ciclo de homing. Nenhum eixos se movimentará, a menos que o bloqueio seja desativado com (\$X).

\$22=1 habilita o homing e \$22=0 volta a desativar.

\$x = desabilita mensagem de alerta e \$h = Procura o homing

NOTA: Confira o config.h para mais opções de homing para usuários avançados. Você pode desativar o bloqueio de homing na inicialização, configurar quais os eixos que se movem pela primeira vez durante um ciclo de homing e em que ordem, e muito mais.





\$23 = Máscara para Inverter a Direção do Homing

Por padrão, o Grbl assume que os seus interruptores de limite estão no sentido positivo, o primeiro movimento será no eixo Z positivo, então os próximos movimentos serão simultaneamente nos eixos X e Y positivos até localizar seus limites de curso. Se a sua máquina tem um interruptor de limite na direção negativa, a máscara para inverter a direção do homing pode inverter a direção dos eixos. Ele funciona exatamente como a inversão do passo e da direção, onde tudo que você tem a fazer é enviar o valor correspondente da tabela para indicar quais os eixos que deseja inverter e procurar na direção oposta.

\$23 = Usar a tabela de mascaras

\$24 = Homing Feed em mm/min

Ao iniciar o ciclo de homing sua máquina irá procurar o interruptor de fim de curso em uma velocidade mais alta, depois de encontrá-los, move-se a uma velocidade mais baixa até chegar ao ponto zero da máquina. Defina essa opção para um valor que seja possível voltar ao ponto zero sempre com precisão e repetibilidade.

\$25 = Homing Seek em mm/min

É a velocidade em que sua máquina irá procurar as chaves de fim de curso. Ajuste para a maior velocidade possível mas que garanta que a máquina não colida muito fortemente com os interruptores.





\$26 = Homing Debounce em milisegundos

Sempre que um interruptor de fim de curso for acionado é necessário verificar se o acionamento foi de fato pela máquina ter encontrado o fim de curso ou se o acionamento foi dado através de ruído elétrico ou mecânico, ou seja, bounce é o sinal alto e baixo em milisegundos antes que ele se estabilize.

Para resolver isso você precisa estabilizar o sinal que poder ser por hardware com algum tipo de condicionador de sinal ou por software com um pequeno atraso da chegada do sinal.

Defina este valor de atraso para que o seu sistema possa obter homing repetível. Na maioria dos casos, de 5-25 milisegundos é bom.

\$27 = Homing Pull-Off em milímetros

Quando sua máquina completar o homing ela poderá voltar um determinado valor em milímetros para impedir que o acionamento dos fim de cursos sejam acionados acidentalmente caso eles estejam configurados em ambos os lados do seu eixo.





\$100, \$101 e \$102 = X,Y e Z Passos por Milímetros

O Grbl precisa saber quantos passos são necessários para mover sua ferramenta. Para calcular os passos/mm para um eixo da sua máquina você precisa saber:

- Quantos **mm** a sua máquina se moveu após uma revolução.
- Quantos passos para seu motor fazer uma revolução (normalmente 200)
- Quantos micropassos você configurou em seu controlador (tipicamente 1, 2, 4, 8 ou 16).

Dica: Utilizar valores altos de micropassos (por exemplo, 16) pode reduzir o torque do motor de passo, então use o menor que lhe dá a resolução do eixo desejado e as propriedades de funcionamento confortáveis.

Os passos/mm pode ser calculados da seguinte forma:

$$\text{passos_por_mm} = \frac{(\text{passos_por_revolução} * \text{micropassos})}{\text{mm_percorridos_revolução}}$$

Calcule este valor para cada eixo e escreva essas configurações para o Grbl.





\$110, \$111 e \$112 = X,Y e Z Taxa Máxima de mm/min

Isso define a taxa máxima que cada eixo pode se mover. Sempre que o Grbl executa um movimento, ele verifica antes se o movimento a ser executado irá ultrapassar a taxa máxima. Isto significa que cada um dos eixos tem a sua própria velocidade independente, que é extremamente útil para a limitação do eixo Z tipicamente mais lentos.

A maneira mais simples para determinar esses valores é testar cada eixo um de cada vez, aumentando lentamente as configurações de taxa e movê-lo. Por exemplo, para testar o eixo X, enviar ao Grbl algo como G0 X50 com suficiente distância de deslocamento de modo a que o eixo acelera a sua velocidade máxima.

Você saberá que você bateu o limiar de taxa máxima quando seus motores pararem de funcionar. Ele vai fazer um pouco de barulho, mas não irá estragar seu motor. Introduza um ajuste de 10-20% abaixo deste valor, então você pode testar novamente até chegar a uma taxa adequada. Em seguida, repita para os outros eixos.

NOTA: Esta definição de taxa de máxima também define a velocidade do G0.





\$120, \$121 e \$122 = X,Y e Z Aceleração em mm/s²

Isso define os parâmetros de aceleração dos eixos em mm/segundo ao quadrado.

Com um valor mais baixo o Grbl fará movimentos mais lentos, enquanto um valor maior produz movimentos mais rápidos. Muito parecido com o ajuste da taxa máxima, cada eixo tem o seu próprio valor de aceleração e são independentes uns dos outros. Isto significa que um movimento multi-eixo só irá acelerar levando em consideração o valor do menor eixo.

A maneira mais simples para determinar os valores para esta configuração é testar individualmente cada eixo aumentando lentamente valores até atingir o limite do motor. Em seguida, finalizar a sua configuração de aceleração com um valor de 10-20% abaixo deste valor máximo absoluto.

É altamente recomendável que você teste com alguns programas g-code pois as vezes a carga em sua máquina é diferente quando se deslocam em todos os eixos juntos.

\$130, \$131 e \$132 = X,Y e Z Tamanho da Área Útil em mm

Isso define o curso máximo de ponta a ponta para cada eixo em milímetros. Isto só é útil se você tem Limites de software (e homing) ativado, pois isso só é usado pelo recurso de limites de software do Grbl para verificar se você excedeu seus limites da máquina com um comando de movimento.



Obrigado

Espero que esse material tenha sido útil para que você aprimore seu conhecimento e faça a sua CNC funcionar da melhor forma possível.

Caso queira mais informações assista o canal Atividade Maker no youtube, lá você vai encontrar mais informações sobre o Grbl v0.9j.

Como entrar em contato?

Você pode entrar em contato pelos seguintes meios:



www.atividademaker.com.br



www.facebook.com/atividademaker



www.youtube.com/c/AtividadeMakerOficial



contato@atividademaker.com.br

