

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

JOÃO PAULO AMARAL DUARTE

**DESENVOLVIMENTO DE UMA PLATAFORMA DE IMPRESSORA 3D
ACOPLADA A UM SCANNER 3D**

UBERLÂNDIA

2017

JOÃO PAULO AMARAL DUARTE

**DESENVOLVIMENTO DE UMA PLATAFORMA DE IMPRESSORA 3D
ACOPLADA A UM SCANNER 3D**

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia Elétrica da Universidade Federal de Uberlândia como requisito à obtenção do título de Bacharelado em Engenharia Elétrica

Orientador: Prof. Dr. Carlos Augusto Bissochi Junior

UBERLÂNDIA

2017

AGRADECIMENTOS

Agradeço, primeiramente, a todos os que contribuíram direta ou indiretamente para o meu sucesso acadêmico, especialmente, a meu orientador, Prof. Dr. Carlos Augusto Bissochi Junior, pelo acompanhamento, orientação e amizade. Aos meus familiares e amigos que foram grandes incentivadores e que sempre acreditaram no meu potencial, especialmente, aos meus colegas de laboratório Daniel, Fernando, Matheus e Thiago pelas contribuições e sugestões desde o início do projeto. E por fim, a administração da Universidade Federal de Uberlândia e da Faculdade de Engenharia Elétrica por todo o apoio recebido durante todos esses anos de graduação.

RESUMO

O presente trabalho detalha a fabricação de um equipamento, denominado Impressora 3D capaz de produzir objetos tridimensionais. A intenção com esse projeto é aprimorar e continuar na universidade pesquisas de prototipagem rápida tornando-a pioneira nessa área no Brasil, visto que no país ainda não há muitos estudos. Os principais focos serão o aprimoramento técnico do hardware da impressora, com novos recursos como a impressão de até três materiais e uma área útil de impressão maior. Outro foco será a construção de um scanner 3D de baixo custo e sua integração na estrutura da impressora já desenvolvida formando uma multifuncional 3D.

Palavras-Chave: Impressora 3D. Scanner 3D. Engenharia Elétrica. Controle e Automação. Universidade Federal de Uberlândia.

ABSTRACT

The present work details the fabrication of an equipment, called 3D Printer capable of producing three-dimensional objects. The intention with this project is to improve and continue in the university rapid prototyping research making it a pioneer in this area in Brazil, since in the country there are still many studies. The main focus will be the technical improvement of the printer hardware, with new features such as printing up to three materials and a larger print area. Another focus will be the construction of a low-cost 3D scanner and its integration into the already developed printer frame forming a multifunctional 3D.

Key words: 3D Printer. 3D Scanner. Electrical Engineering. Control and Automation. Federal University of Uberlândia.

LISTA DE ILUSTRAÇÕES

Figura 1 - CNC e Impressora 3D.....	12
Figura 2 - Modelos de impressoras do tipo RepRap	14
Figura 3 - Scanners profissionais de contato (braço articulado) e sem contato (óptico).....	15
Figura 4 - Projeto FabScan	16
Figura 5 - Processo de Reconstrução Tridimensional	17
Figura 6 - Maquete e robôs impressos em 3D	19
Figura 7 - Próteses e modelos dentários impressos em 3D	19
Figura 8 - Impressora 3D desenvolvida na UFU	21
Figura 9 - Robô cartesiano.....	23
Figura 10 - Perfil 20x20 e modelo da estrutura da impressora desenhado em software 3D	24
Figura 11 – Vista Superior e Frontal para ilustração dos eixos X (azul), Y (vermelho) e Z (amarelo).....	25
Figura 12 - Rolamentos utilizados nas juntas prismáticas (<i>pillow block</i> e linear longo)	26
Figura 13 - Sistema de Transmissão de Potência (Fuso e Correias)	27
Figura 14 – Localização das extrusora do tipo <i>bowden</i>	27
Figura 15 - Hotend comum simples, comum triplo e diamante triplo.....	28
Figura 16 - Mesa de alumínio e parafuso de ajuste do nivelamento	29
Figura 17 - Estrutura final com três carretéis de filamentos.....	29
Figura 18 - Arduino (azul), RAMPS (vermelho) e 5 drivers	31
Figura 19 – Motor de passo Nema 17 bipolar	32
Figura 20 - Esquema de ligação do driver (a) e sequência de pulsos enviados pelo Arduino (b)	34
Figura 21 - Módulo complementar para sexto driver	35
Figura 22 - Sensor indutivo e mecânico	35
Figura 23 - Cartucho aquecedor e manta térmica.....	36
Figura 24 - Termistor (a) e módulo de controle da mesa com PT100 (b)	37
Figura 25 - Fonte, módulo de força e botão de emergência	37
Figura 26 - Interface com o usuário (LCD).....	38
Figura 27 - Esquema de ligação da eletrônica	39
Figura 28 - Softwares CAD e CAM	40
Figura 29 - Interface RepertierHost.....	41

Figura 30 - Filamentos em cores variadas	42
Figura 31 - Estrutura scanner com e sem MDF	44
Figura 32 - Perfil 15x15 e modelo da estrutura do scanner desenhado em software 3D	45
Figura 33 - Arduino UNO, placa de extensão CNC e driver.....	46
Figura 34 - Laser	46
Figura 35 - Webcam C270 (a) e adaptação para ajuste de foco (b).....	47
Figura 36 - Suportes impressos em preto e branco para laser, motor e câmera	47
Figura 37 - Interface Simple3DScan	48
Figura 38 - Pontos lidos em um escaneamento e tratamento	49

LISTA DE TABELAS

Tabela 1 - Tamanho Perfis Estruturais 20x20	24
Tabela 2 - Tamanho Guias Lineares 8mm.....	26
Tabela 3 - Características do Motor de Passo Adotado.....	33
Tabela 4 - Tamanho Perfis Estruturais 15x15	45
Tabela 5 - Custo Mecânica Impressora 3D	50
Tabela 6 - Custo Mecânica Scanner 3D	51
Tabela 7 - Custo Eletrônica Impressora 3D.....	52
Tabela 8 - Custo Eletrônica Scanner 3D	52
Tabela 9 - Custo Material de Consumo (Filamentos).....	53
Tabela 10 - Custo Final Impressora e Scanner 3D	54

LISTA DE ABREVIATURAS E SIGLAS

2D	Duas dimensões
3D	Três dimensões
ABS	<i>Acrylonitrile Butadiene Styrene</i>
AC	<i>Alternating Current</i> ou Corrente Alternada
CAD	<i>Computer-Aided Design</i>
CAM	<i>Computer-Aided Manufacturing</i>
CI	Circuito Integrado
CNC	Controle Numérico Computadorizado
DC	<i>Direct Current</i> ou Corrente Contínua
FDM	<i>Fused Deposition Modeling</i>
FEELT	Faculdade de Engenharia Elétrica
GNU	<i>General Public Licence</i>
IDE	<i>Integrated Development Environment</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
I/O	<i>Input/Output</i>
LCD	Liquid-Crystal Display ou Tela de Cristal Líquido
LED	<i>Light-Emitting Diode</i> ou Diodo Emissor de Luz
NC	Numerical Control
NEMA	<i>Nation Electrical Manufactures Association</i>
MDF	Medium Density Fiberboard
MPR	Milímetros Por Rotação
PC	<i>Personal Computer</i>
PID	<i>Proportional-Integral-Derivative</i>

PLA	Ácido Polilático
PLC	<i>Programmable Logic Controller</i>
PPM	Passos por milímetro
PTFE	Politetrafluoretileno
PWM	<i>Pulse-Width Modulation</i> ou Modulação por Largura de Pulso
RAMPS	<i>RepRap Shield Arduino</i>
RGB	<i>Red, Green, Blue</i>
SLA	<i>Stereo Lithography Apparatus</i>
SLS	<i>Selective Laser Sintering</i> ou Sinterização Seletiva a Laser
STL	<i>STeroLithography</i>
UART	<i>Universal Asynchronous Receiver-Transmitter</i> ou Porta Serial de Hardware
UFU	Universidade Federal de Uberlândia
USB	<i>Universal Serial Bus</i> ou Porta Serial Universal

SUMÁRIO

1. INTRODUÇÃO	11
1.1. ANTECEDENTES	11
1.2. APLICAÇÕES	16
1.3. MOTIVAÇÃO	20
2. DESENVOLVIMENTO	22
2.1. IMPRESSORA 3D	22
2.1.1. MECÂNICA	23
2.1.2. ELETRÔNICA	30
2.1.3. SOFTWARE	39
2.1.4. MATERIAL DE CONSUMO	41
2.2. SCANNER 3D	43
2.3. ORÇAMENTO	50
2.3.1. MECÂNICA	50
2.3.2. ELETRÔNICA	51
2.3.3. SOFTWARES	53
2.3.4. MATERIAL DE CONSUMO	53
2.3.5. CUSTO FINAL	54
3. CONCLUSÃO	55
REFERÊNCIAS BIBLIOGRÁFICAS	57
ANEXO A – Código de Parâmetros para o Firmware da Impressora 3D (config.h)	59
ANEXO B – Código de Parâmetros para o Firmware do Scanner 3D	68

1. INTRODUÇÃO

A pesquisa é uma atividade cotidiana, uma atitude, um “questionamento sistemático crítico e criativo, mais a intervenção competente na realidade, ou o diálogo crítico permanente com a realidade em sentido teórico e prático” (DEMO, 1996). Inicia com um problema que não se tem informações para solucionar, onde um conjunto de ações e propostas é empregado para encontrar as respostas deste problema, que têm por base procedimentos racionais e sistemáticos (SILVA, 2001).

Debates e contribuições recentes têm enfatizado a relevância do papel do conhecimento para que o país se desenvolva economicamente. No Brasil, reconhecer a importância da inovação trouxe grandes mudanças no modo de atuar de empresas, entidades representativas, instituições de ensino e de pesquisa e organizações do setor público (Filho & Nogueira, 2006).

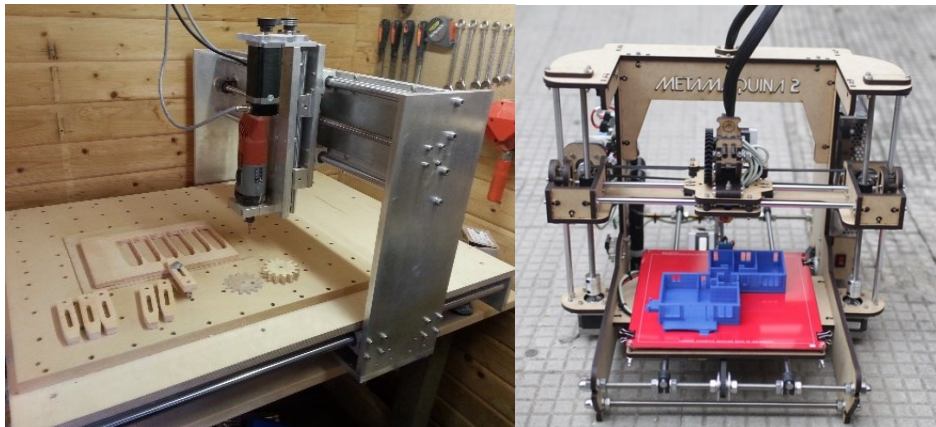
1.1. ANTECEDENTES

A história da impressora remonta ao século XV, com a invenção de Johannes Gutenberg de uma máquina móvel para a produção em massa de livros impressos, com a utilização de tinta a base de óleo e uma prensa de madeira, surgindo assim, a impressora e, consequentemente, a imprensa. Com o decorrer do tempo, com a invenção do computador, essa impressora foi sendo aprimorada, chegando à impressora convencional que conhecemos atualmente, capaz de imprimir colorido direto do computador e que possui diversas tecnologias embutidas em uma máquina compacta e acessível, o que a popularizou.

Inicialmente, para entender a diferença entre os tipos de máquinas, é necessário entender a respeito de dimensões, as quais as mais usuais são duas (2D) e três dimensões (3D). Um exemplo de sistema de duas dimensões é uma impressora convencional de papel que imprime apenas em um plano. Já o espaço tridimensional é aquele que pode ser definido como tendo três dimensões, que além de comprimento e largura, possui altura ou profundidade, o que na prática indica relevo, podendo uma máquina 3D produzir objetos mais complexos.

A prototipagem rápida é uma tecnologia que permite a construção de modelos físicos a partir de um modelo *Computer Aided Design* (CAD) 3D, surgindo para o desenvolvimento de produtos da indústria de bens de consumo, sendo capaz de reproduzir fisicamente um modelo virtual em vários tipos de materiais (Nishida, 2011). Hoje em dia, existem dois métodos, conforme Figura 1, que são mais utilizados para produção de objetos em três dimensões: a Prototipagem Rápida Aditiva, que utiliza técnicas de deposição de materiais, e seu principal exemplo é a Impressora 3D, e a Prototipagem Rápida Subtrativa, que funciona removendo material, de forma a esculpir o objeto, e como modelo temos a máquina de Comando Numérico Computadorizado, conhecida por CNC.

Figura 1 - CNC e Impressora 3D



Fonte: Internet

A Impressora 3D surgiu a partir dos anos 80, onde diversas propostas e patentes foram feitas, além de um grande esforço de pesquisa e desenvolvimento foi gasto para fazer com que as impressoras 3D fossem o que elas representam nos dias de hoje (Cunico, 2015). Em 1984, Chuck Hull, engenheiro norte-americano, criou uma impressora baseada no método da Estereolitografia (SLA - *Stereo Lithography Apparatus*) que é a solidificação de uma resina fotocurável, e, em 1989, Steven Scott Crump, patenteou o Método de Fusão e Deposição (FDM - *Fused Deposition Modeling*), o que implicou na fundação, por cada um, das duas principais empresas nesse segmento, a 3D System e a Stratasys respectivamente. A partir de então, outros métodos também foram inventados, como o de sinterização seletiva por laser (SLS - *Selective Laser Sintering*) e o de modelagem de objetos laminados.

Para podermos entender a utilização da prototipagem rápida aditiva no processo de desenvolvimento de produtos é necessário entendermos quais as diferentes tecnologias,

sistemas e processos de prototipagem rápidas disponíveis hoje ao mercado. Todos estes equipamentos têm em comum a possibilidade de construção de geometrias complexas, sem a necessidade da utilização de nenhum tipo de ferramenta de construção do tipo moldes, ou seja, constroem-se os objetos de forma livre.

Ao longo dos anos o número de variantes, configurações e aplicações destas tecnologias vem aumentando progressivamente. Contudo, SLA, FDM, e SLS podem ser destacadas como as tecnologias pioneiras de uma possível nova era industrial (Cunico, 2015).

A estereolitografia, ou SLA, utiliza como matéria prima uma resina que se polimeriza sob a luz ultravioleta de alta potência que é projetada sobre esse material, fazendo com que a mesma endureça nas regiões desejadas. A solidificação é feita camada a camada, e este processo permite a obtenção de peças com incrível definição de detalhes, alta precisão e acabamento, fino e liso das peças. A desvantagem, é o alto custo desse tipo de impressora, e produção, apenas, de peças de dimensões menores.

A Sinterização Seletiva a Laser, ou SLS, é semelhante a SLA. Porém, ao invés de resina líquida, utiliza materiais na forma de pó, que é processado em um ambiente inerte e termicamente controlado no interior de uma câmara que é sinterizado por ação de um laser de gás carbônico. Depois que uma camada é produzida, uma nova é depositada, e assim sucessivamente, até finalizar a construção da peça. Sua grande vantagem é a variedade de materiais que podem ser usados, incluindo metais, e produzir peças mais resistentes a esforços mecânicos, impactos e a temperaturas mais altas. A desvantagem é peças com superfície áspera e baixa quantidade de detalhes, alto custo da máquina, além de que essa impressora exige um trabalho de pós-processamento para melhorar o acabamento final.

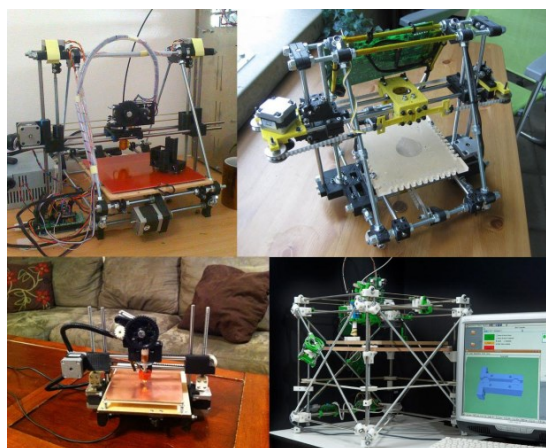
O Método por Fusão e Deposição, ou FDM, faz uso de filamentos de resina termoplásticas aquecidas e perfiladas a partir de uma matriz em forma de furo circular, com a liberdade de movimento nos eixos X, Y e Z (Garcia, 2010). A cabeça injetora traça perímetros e os preenche construindo, assim, camada por camada. Uma das principais vantagens desse processo de fabricação, que se processa de forma relativamente simples e eficiente, está no custo, possibilitando a produção com preços mais acessíveis, em relação às outras tecnologias. Pode-se afirmar que este instrumento produz peças resistentes, duráveis e funcionais, podendo receber um pós-tratamento para melhorar seu aspecto visual.

Atualmente, diversos equipamentos têm-se desenvolvido e aprimorados por meio da ideia do código livre, ou *open source*. Nesse tipo de iniciativa os desenvolvedores de

hardware e software disponibilizam seus projetos à comunidade de desenvolvedores por todo o mundo para que, de um modo colaborativo o projeto seja mantido, aperfeiçoado e popularizado. Por trás dos projetos *open source* existe uma filosofia que defende a liberdade para qualquer pessoa modificar aperfeiçoar, compartilhar e, em alguns casos, até mesmo comercializar. Projetos de hardware livre também estão indo na mesma direção, componentes eletrônicos, circuitos, máquinas e equipamentos, têm sido disponibilizados para que a comunidade faça suas evoluções (MOREIRA, 2016). Seguindo essa linha, surgiu na Inglaterra em 2004, uma comunidade voltada para o desenvolvimento de impressoras 3D, chamadas de RepRap, que, por sinal, atualmente, são as mais utilizadas no mundo, por serem de construção simples, custo mais baixo e permitirem que os próprios usuários as personalizem, como mostra a Figura 2..

A RepRap é uma Impressora 3D de código livre, capaz de se auto replicar, através da impressão de seus próprios componentes estruturais, tratando-se de um dispositivo que qualquer um pode desenvolver. Foi uma das primeiras impressoras 3D de baixo custo, e o seu projeto começou a revolucionar o segmento de impressoras 3D de código livre, *Open Source GNU (General Public Licence)* (RepRap, 2017). Hoje em dia, diversos modelos, por serem de código livre, já foram montados e testados por usuários em todo o mundo e cada um tem suas características e designs diferentes, algumas, por exemplo, focam na rigidez estrutural, no design ou no tamanho da área de trabalho.

Figura 2 - Modelos de impressoras do tipo RepRap



Fonte: (RepRap, 2017)

A modelagem tridimensional (3D) é muito requerida atualmente, tanto no entretenimento, com jogos e filmes, quanto na indústria, substituindo metodologias já consolidadas por um custo menor de produção e uma reutilização constante. Produzir um modelo tridimensional atualmente é algo bem mais simples, pois há uma infinidade de softwares que auxiliam nessa tarefa (MOREIRA, 2016).

Para se fazer esse processo de digitalização e obtenção de imagens tridimensionais usa-se equipamentos, denominados, scanners 3D. No mercado há uma infinidade desse tipo de produto, cada um com método e precisão diferentes para desempenhar esta função, no entanto o custo dos mesmos é elevado, tornando-os inacessíveis. As principais técnicas se resumem a basicamente duas, as de “contato” e “sem contato”, conforme a Figura 3. Neste primeiro tipo, a superfície é tocada usando uma sonda mecânica na extremidade de um braço articulado ou uma máquina de medição de coordenadas. Já os “sem contato”, que, geralmente, baseiam-se em métodos reflexivos acústico, como radares e sonares, ou óticos, no qual utilizam câmeras de alta capacidade e emissores de luz (incluindo laser) para obtenção das formas de objetos analisando as imagens capturadas e submetendo-as a um algoritmo específico.

Figura 3 - Scanners profissionais de contato (braço articulado) e sem contato (óptico)

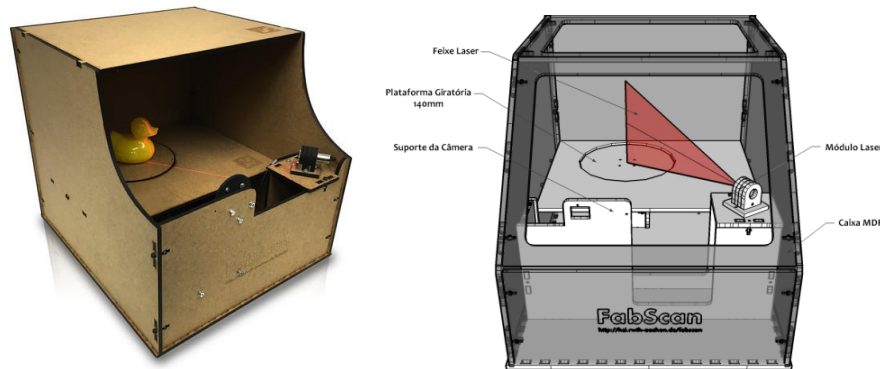


Fonte: Internet

Similarmente ao projeto RepRap, de código livre, existe um para Scanner 3D também. O projeto FabScan, ilustrado na Figura 4, surgiu a partir de um trabalho executado por Francis Engelmann, na Alemanha em 2011. O propósito do trabalho segundo Engelmann (2011) foi produzir um scanner 3D de baixo custo, portátil, que fizesse captura do objeto em 360 graus e que fosse de fácil construção. É elaborado com o método de triangulação utilizando um

emissor laser, uma câmera e uma base giratória movida por um motor de passo controlado por Arduino, unidos no interior de uma câmara de MDF (MOREIRA, 2016).

Figura 4 - Projeto FabScan



Fonte: (The FabScan Project, 2017)

1.2. APLICAÇÕES

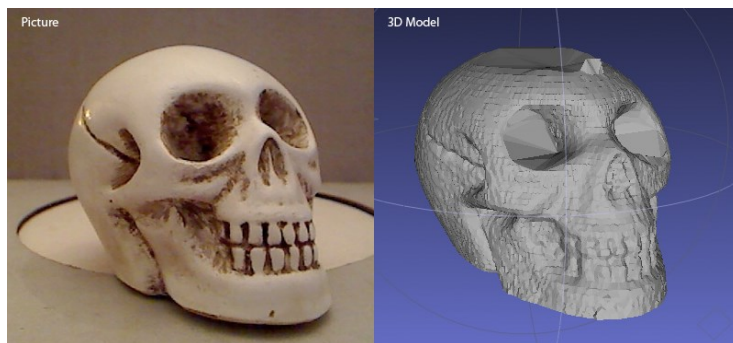
Um dos maiores avanços em nossa sociedade industrial foi a obtenção da maioria dos produtos industrializados de forma rápida e fácil, podendo-se definir essa fase da nossa história como a era de diversificação de nosso desejo. As pressões de tempo, qualidade e custos junto com o aumento da variedade de produtos e o processo de competição mundial entre as empresas têm tornado a tecnologia de Prototipagem Rápida parte integrante dos processos de negócios (Friedel & Liedtka, 2007).

A integração e a rapidez entre as várias fases do desenvolvimento de produtos são sem dúvida, elementos chave na competitividade entre as empresas. Assim, a passagem rápida do conceito do produto à produção é um incremento de competitividade, que a prototipagem por meio de sistema CAD oferece as empresas como meio determinante para o sucesso do desenvolvimento de novos produtos (Ferreira, Madureira, & Castro, 2006).

De acordo com Azevedo e Tavares (2009), o processo de reconstrução tridimensional tem se tornado relevante nas últimas décadas especialmente na área médica e indústria cinematográfica, no entanto temos acompanhado um vasto crescimento de sua aplicação nos campos de engenharia, geologia, astronomia, biologia, etc. Isso tem se tornado mais intenso

porque o volume de imagens vem aumentando exponencialmente; equipamentos como tomógrafos, sonares, telescópios, câmeras digitais, além de uma infinidade de sensores tem se tornado cada vez mais acessíveis e presentes no nosso dia-a-dia.

Figura 5 - Processo de Reconstrução Tridimensional



Fonte: (The FabScan Project, 2017)

Uma aplicação para essa tecnologia tridimensional nos últimos anos foi a “engenharia reversa”, técnica da qual se parte do objeto final para se obter um modelo 3D, exemplificado na Figura 5, permitindo assim sua modificação e produção. O processo de engenharia reversa envolve uma leitura do objeto real produzindo sua nuvem de pontos nos eixos “x, y, z” (largura, altura e profundidade) e, através de software específico, geração de sua malha geométrica para reconstrução tridimensional (MOREIRA, 2016). E a partir de então, com o objeto modelado é possível imprimi-lo, da forma que foi digitalizado ou com modificações.

As impressões 3D atuais são tão realísticas quanto outro método de fabricação. É possível imprimir peças coloridas, móveis, de diferentes materiais, entre outras tecnologias. O diferencial da impressão 3D é a possibilidade de a peça ser testada em um modelo físico, prototipado, podendo ser reajustado, diferente de somente ter a visualização da peça no computador, além de ter as dimensões e design mais fiéis à realidade, de modo rápido e econômico comparado à produção em massa.

No Brasil, a aplicação dessas técnicas ainda é restrita, por ser uma tecnologia recente. Isso se deve a dois fatores: um econômico e outro de ordem social. O econômico está relacionado aos altos custos dos equipamentos e insumos, que são em sua maioria importados, embora, recentemente, algumas empresas nacionais estão a investir no desenvolvimento e produção de maquinário para fabricação digital. Existe também uma grande limitação social

ao emprego das técnicas impressão 3D, pois ainda não há grande quantidade de mão de obra especializada nessa área.

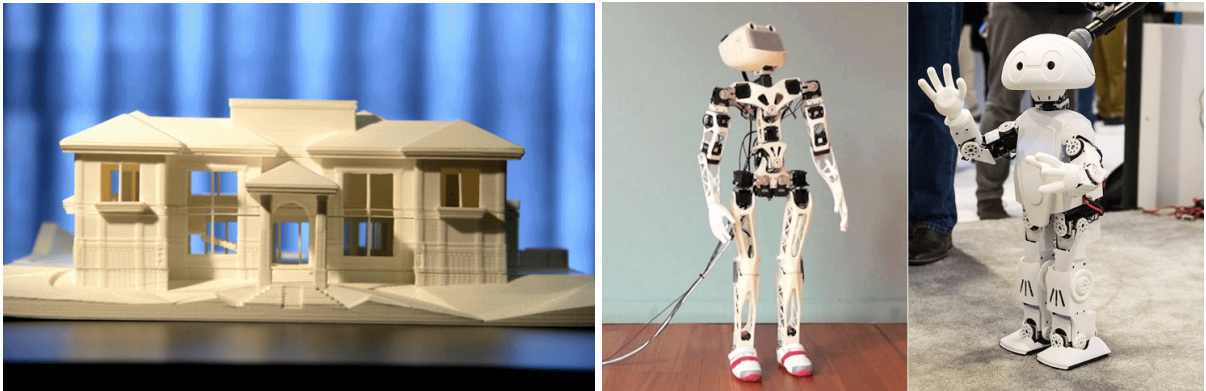
A viabilização desses tipos de equipamentos está se tornando, cada vez maior devido à redução do custo, tanto da impressora em si, quanto do scanner e do filamento de consumo, além da produtividade, pois reduz o tempo de produção e, por meio de protótipos, se torna possível determinar erros e possíveis modificações antes de iniciar a produção em massa. Ainda é possível se produzir scanner e impressoras de códigos abertos, utilizando ferramentas e equipamentos fáceis de encontrar no mercado nacional reduzindo seu custo, e permitindo uma difusão dessa tecnologia, tanto para estudo quanto para criação de empresas.

As áreas de aplicação das impressões 3D crescem cada dia mais, e uma das principais áreas é a industrial. Não há nada melhor em protótipos, do que confeccioná-los no material em que serão produzidos em escala, permitindo testes reais de material com alta precisão dimensional nos protótipos, pois o objetivo é minimizar erros e acelerar a chegada do produto ao mercado, sem perder qualidade no processo desenvolvimento do projeto (Junior, Junior, & Neto, 2007).

Com a grande difusão nos últimos anos do uso de softwares *Computer Aided Design* (CAD) no processo de projeto de novos produtos e de softwares *Computer Aided Manufacture* (CAM) na integração desses projetos à manufatura, muitos engenheiros e estudantes de engenharia passaram a ter a necessidade de máquinas capazes de gerar formas tridimensionais reais a partir dos modelos criados no computador (Nishida, 2011).

A área da engenharia civil e da arquitetura têm adotado a impressão 3D como forma de produzir modelos e maquetes 3D de construções civis que podem ser facilmente criados economizando tempo e dinheiro, além de ser mais fiel do que maquetes produzidas artesanalmente. Na engenharia mecânica permite-se a criação de moldes, protótipos em tamanho real, ou até mesmo, a própria peça diretamente na impressora, tendo uma precisão muito maior do que usando tornos e ferramentas manuais. Já na engenharia mecatrônica e na de controle e automação a produção de peças para robôs, processos de automação e protótipos também são uma realidade. Para produção de réplicas de aeronaves, carros, motos, ou outro qualquer objeto em menor escala para serem utilizados em estudos iniciais de aerodinâmica, físicos, estéticos, para avaliações de projeto ou para apresentações finais. A Figura 6 mostra alguns modelos de maquetes e robôs impressos em 3D.

Figura 6 - Maquete e robôs impressos em 3D



Fonte: Internet

Na engenharia biomédica a impressão em 3D poderá contribuir na produção de peças para fabricação de próteses, permitindo que partes humanas sejam substituídas por próteses com características semelhantes às trocadas, permitindo todos os movimentos, fazendo com que se criem peças altamente complexas, com muitas articulações, como uma réplica de um joelho ou de uma mão (Figura 7). Além de reproduzir a geometria de partes do corpo humano, a prototipagem rápida tem como uma de suas aplicações à confecção de biomodelos para planejamento cirúrgico (Volpato, 2007). Já está em desenvolvimento também a utilização de células para a fabricação de órgãos, ou seja, seria possível imprimir um osso provisório para que o machucado possa se recuperar, além de ser útil em procedimentos odontológicos (Figura 7) e no tratamento de doenças como a osteoporose.

Figura 7 - Próteses e modelos dentários impressos em 3D



Fonte: Internet

Outros estão utilizando a impressora para reproduzir cópias fieis de fósseis de dinossauros, visando o estudo antropológico, de espécies extintas, e de historiadores,

abrangendo a área das humanas. Museus e espaços históricos serão capazes de construir réplicas sofisticadas de seus materiais que são restritos aos visitantes, preservando assim os originais.

No entanto, ainda hoje, uma série de empresas auxiliada por artistas plásticos, cria seus produtos baseados em esculturas modeladas em argila, isopor ou outro material apropriado. Há também aquelas que possuem uma série de peças antigas, produzidas por métodos convencionais e que necessitam fazer pequenas alterações para produzir um novo ferramental. A dificuldade surge no momento em que os modelos são gerados apenas fisicamente, o que impede sua manipulação e qualquer tipo de alteração. A solução a ser aplicada para que se possa trabalhar com esses modelos é digitalizá-los e portá-los para um sistema CAD eficiente, e é a partir daí que se aplica o Scanner 3D.

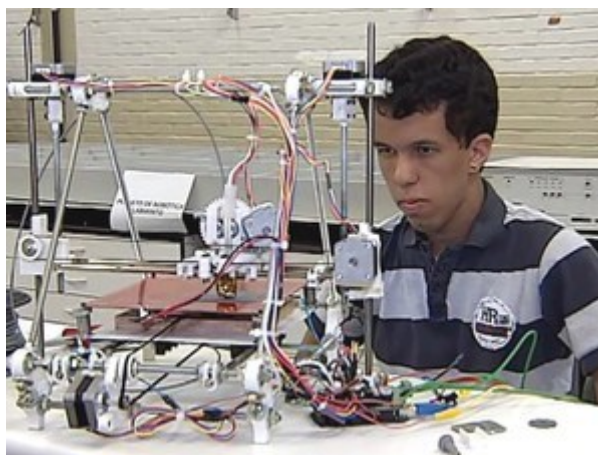
1.3. MOTIVAÇÃO

Tendo em vista o grande número de aplicações e o crescente interesse do público em geral por máquinas capazes de criar objetos personalizados, as chamadas Impressoras 3D, notou-se a demanda por um equipamento, em 2013, capaz de executar essa mesma tarefa, mas que tivesse um custo menor e fosse, assim, mais acessível a todos, além de que a construção de um equipamento desse porte, na área acadêmica, possibilita o estudo em diversas áreas, onde já é aplicada, permitindo o avanço e aprimoramento de técnicas, além de pesquisar novas utilidades para essa tecnologia 3D.

Com isso, em 2013, foi aprovado um projeto com bolsa, para que na Universidade Federal de Uberlândia (UFU), fosse construída, uma Impressora 3D do tipo RepRap e, assim, em 2014, foi concluída, a primeira máquina desse tipo da região, que está em pleno funcionamento até os dias de hoje nas dependências da Faculdade de Engenharia Elétrica (FEELT). Ela foi construída utilizando da lógica de reaproveitamento, visto que 30% de suas peças foram reaproveitadas de impressoras de papel estragadas. Isso, combinado a outros fatores, como a aquisição otimizada de peças, tornou a montagem total do equipamento 40% mais barata do preço normal de mercado. Esse projeto considerado inovador por ser o primeiro da cidade trouxe para a Universidade diversos benefícios, além do reconhecimento pela mídia de Uberlândia e região, onde foi exibido matérias em telejornais e jornais

eletrônicos, divulgando o nome da UFU na área de pesquisas inovadoras, como mostra a Figura 8, exibida em matéria no portal de notícias da globo (G1) em 2014

Figura 8 - Impressora 3D desenvolvida na UFU



Fonte: (G1, 2014)

Já foram feitas, diversas construções de peças em 3D demonstrando a eficiência do projeto concluído, e as mesmas foram utilizadas para aprimorar a estrutura física da própria máquina, melhorando questões de ajustes, estabilidade e qualidade, além de projetos para a área acadêmica, como maquetes, protótipos, próteses, cases, robôs, trabalhos escolares, dentre outros. No entanto, a RepRap caseira desenvolvida possui limitações, como área de trabalho pequena e impressão de peças em um só material ou cor, dificuldades de calibração, tornando-a imprecisa, além de não ser operável por qualquer pessoa e sua estrutura apresentar necessidade de manutenção corretiva constantemente. Por isso, notou-se a necessidade de o desenvolvimento de uma nova plataforma RepRap profissional, com novos recursos, menos propensas a erros e, conseqüentemente, menos manutenção e mais precisa, além de ser acessível a qualquer usuário que a adquirir.

A partir do ano, de 2016, começou a ser desenvolvida, uma nova versão que será detalhada nesse trabalho, e, implementadas algumas melhorias, como a possibilidade de imprimir com até três tipos de materiais ou cores diferentes, uma área de impressão maior, e, um módulo Scanner 3D com o objetivo de digitalizar uma peça, tratá-la e, se desejar, imprimir a mesma. Isso tudo seguirá a lógica, de tornar o equipamento de fácil manuseio, operação e manutenção ao público em geral, além de ser um equipamento mais barato, comparado aos similares a venda no mercado.

2. DESENVOLVIMENTO

A Impressora 3D proposta neste trabalho pode ser dividida em quatro partes principais: mecânica, eletrônica, software e material de consumo. A primeira contém toda a estrutura física e a segunda a parte eletrônica. Já a terceira, é responsável pelo firmware e o programa de interface da máquina com o usuário. E por fim, o material de consumo consiste nos tipos de filamentos que são usados como matéria prima para se imprimir os objetos. Inicialmente, abordarei sobre os detalhes construtivos da Impressora 3D, seguida dos aspectos do Scanner 3D e, posteriormente, farei uma análise de como foi feito a aquisição das peças e do custo de ambos.

2.1. IMPRESSORA 3D

Após um grande estudo, optou-se por trabalhar com a impressora RepRap, porém com uma estrutura física, totalmente diferente das convencionais, e com modificações na parte eletrônica. A parte de software sofrerá mudanças na programação e nos parâmetros inseridos devido a alteração de toda a mecânica, porém a maioria do corpo permanece igual aos outros modelos, inclusive o programa da interface, devido a grande documentação existente no que se refere ao mesmo.

A parte mecânica da Impressora 3D é formado por uma estrutura física de suporte e sustentação que permite a movimentação de um robô cartesiano por meio de motores precisos regidos por um micro controlador. O projeto eletrônico da máquina consiste, basicamente, de motores de passos e seus drivers de potência, da ferramenta de extrusão, da plataforma de aquecimento, de sensores (posição e temperatura), da tela de cristal líquido (LCD), responsável pela interface com o operador, de um módulo de energia, além do micro controlador principal responsável por gerenciar tudo.

2.1.1. MECÂNICA

A estrutura física da Impressora 3D é formada por perfis estruturais, guias lineares, rolamentos, fusos, juntas, suportes, três extrusoras, além de parafusos, arruelas e porcas. Todas as peças foram adquiridas no mercado nacional, e com o intuito de deixar o produto final com o máximo de profissionalismo. A Impressora 3D a ser construída nada mais é que um robô, ou seja, uma máquina manipuladora com vários graus de liberdade controlada automaticamente, reprogramável, multifuncional, que pode ter base fixa ou móvel para utilização em aplicações de automação (International Organization for Standardization, 10218).

O robô cartesiano é formado por três trilhos que articulam em ângulos retos um com o outro, sendo que um deles corre verticalmente para controlar a altura e os dois outros correm horizontalmente (um para controlar o movimento esquerda/direita e o outro o movimento avanço/recuo). Robôs cartesianos são altamente precisos, bem lentos e podem possuir elevada capacidade de carga (Teixeira, 2001). É o robô de configuração mais simples, desloca as três juntas, chamadas de prismáticas ou lineares por mover em linha reta e ser compostas de duas hastes que deslizam entre si. Este robô opera dentro de um envoltório de trabalho cúbico, e tem a configuração XY-Z, que significa que a ferramenta se move no plano XY e a mesa no eixo Z, de acordo com a Figura 9..

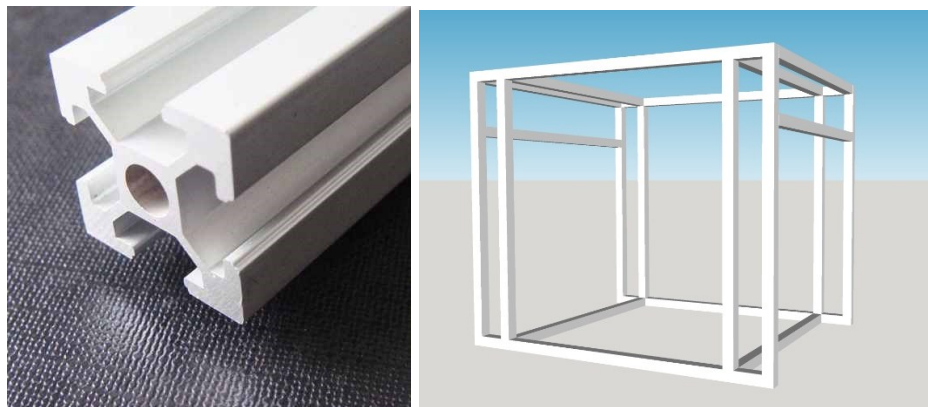
Figura 9 - Robô cartesiano



Fonte: Internet

Inicialmente, foi projetado o envoltório de trabalho, foi formado por perfil estrutural em alumínio do tipo 20x20, devido a sua leveza e praticidade. Para dimensionar a estrutura, foi elaborado no software 3D Sketchup como seria montado a impressora, demonstrado na Figura 10, de forma a abrigar, em seu interior, o robô cartesiano e todos os demais componentes eletrônicos.

Figura 10 - Perfil 20x20 e modelo da estrutura da impressora desenhado em software 3D



Fonte: Próprio autor

Essa estrutura composta por 20 barras de dimensões relatadas na Tabela 1 arranjadas com cantoneiras de 90 graus, permite uma elevada rigidez estrutural aos esforços de flexão e torção necessária para que não haja problemas de vibração no decorrer das impressões, baixo custo e extrema versatilidade, permitindo personalizar facilmente as estruturas da forma que forem mais convenientes, além de proporcionar uma aparência mais profissional comparado as impressoras RepRap convencionais, formadas de barras roscadas e peças plásticas.

Tabela 1 - Tamanho Perfis Estruturais 20x20

Quantidade	Tamanho
8	460 mm
4	540 mm
4	500 mm
4	600 mm

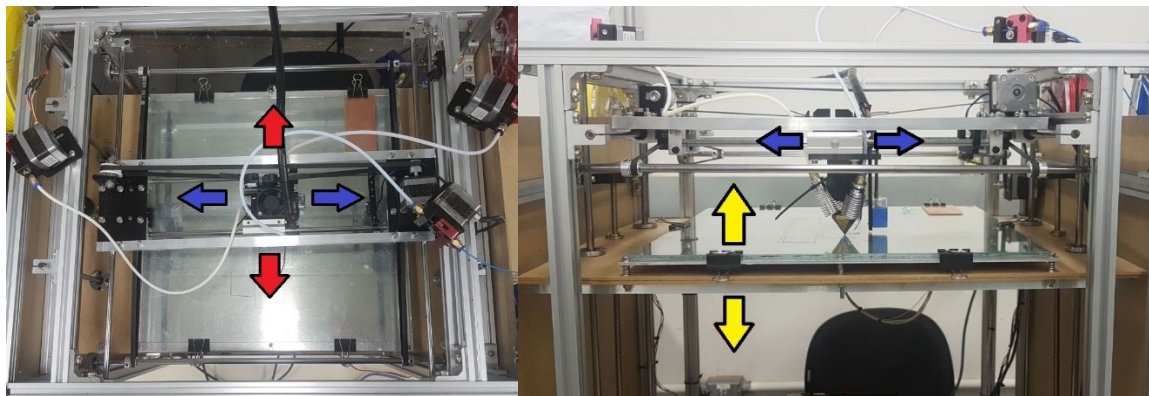
Fonte: Próprio autor

De forma a projetar uma área de impressão maior que as convencionais e, de forma a, também não deixar a impressora muito grande, tornando-a pesada e de difícil transporte, foi feita uma com área útil de impressão de 220mm de largura (L_x) por 290mm de profundidade (L_y) por 320mm de altura (L_z). Sendo, assim, o cálculo do volume de trabalho segue a modelação matemática abaixo, na qual permite que o robô imprimir um volume máximo de $0,2 \text{ m}^3$ em sua área útil de trabalho.

$$\text{Volume} = L_x * L_y * L_z = 220 * 290 * 320 = 20.416.000 \text{ mm}^3 = 0,02 \text{ m}^3$$

O número de graus de liberdade que um robô apresenta é o número de variáveis independentes de posição que precisam ser especificadas para se definir a localização de todas as partes do mecanismo, de forma inequívoca. O robô é normalmente uma combinação de elos e juntas em forma de cadeia cinemática aberta. Portanto, o número de juntas equivale ao número de graus de liberdade.

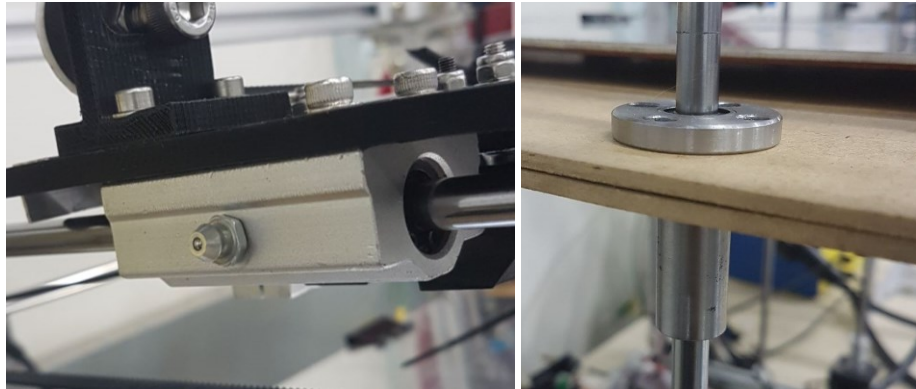
Figura 11 – Vista Superior e Frontal para ilustração dos eixos X (azul), Y (vermelho) e Z (amarelo)



Fonte: Próprio autor

O robô cartesiano possui três graus de liberdade, que são definidos pelos movimentos lineares nos eixos X, Y e Z, que de acordo com a Figura 11, estão ilustrados, respectivamente, azul, vermelho e amarelo, sendo todos formados por juntas prismática. Para que essas articulações se desloquem ao longo das guias lineares no espaço de trabalho, são usados rolamentos lineares, de dois tipos, o *pillow block* e o linear longo, mostrados na Figura 12.

Figura 12 - Rolamentos utilizados nas juntas prismáticas (*pillow block* e linear longo)



Fonte: Próprio Autor

O sistema de deslocamento da Impressora 3D utiliza seis barras lineares de 8mm em aço tratado (duas por eixo), nas dimensões da Tabela 2, quatro *pillow block* de 8mm longo para os eixos X e Y e quatro rolamentos lineares de 8mm longos para o eixo Z.

Tabela 2 - Tamanho Guias Lineares 8mm

Quantidade	Eixo	Tamanho
2	X	350 mm
2	Y	500 mm
4	Z	540 mm
2	Auxiliares	500 mm

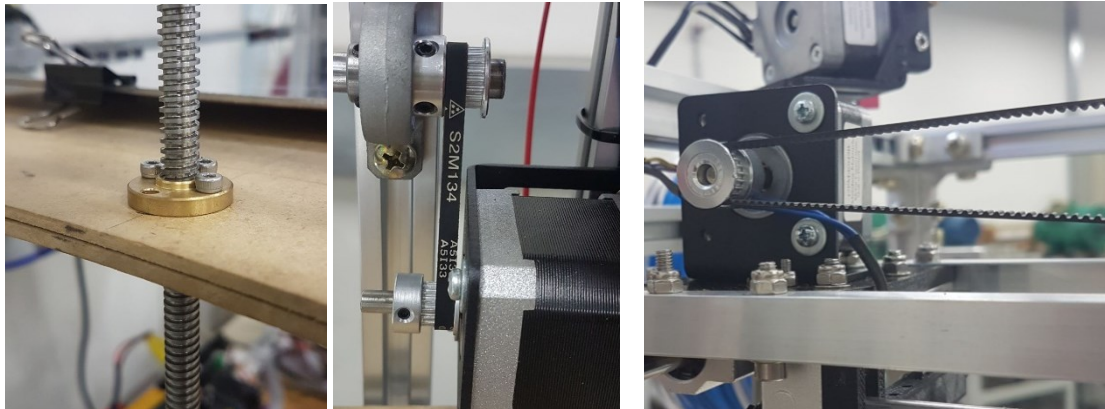
Fonte: Próprio autor

Um sistema de transmissão de potência mecânica permite transformar a rotação do eixo do motor em movimento linear, por meio de dois métodos principais: por correia ou por parafuso sem fim (fusos), que são componentes mecânicos para transmitir potência dos motores para que os eixos se movimentem. Os sistemas utilizados, combinados com a precisão dos motores e drivers, proporcionam uma precisão de frações de milímetros na movimentação dos eixos.

As correias têm as vantagens de oferecer capacidade para velocidades altas, movimentação precisa e baixo custo. No entanto, não é capaz de suportar grandes cargas e com o tempo a ressonância aumenta, devido a tensão não se manter constante, o que pode levar a correia a soltar-se. Já os parafusos sem-fim são umas das soluções mais usadas para a movimentação linear. Como a razão de conversão de rotação é de 1:1, ou seja, o eixo do

motor está acoplado diretamente no fuso, tem-se um bom aproveitamento do movimento do motor, e se consegue movimentar cargas linearmente mais expressivas e com boa precisão. Todavia, em relação à velocidade fica aquém do método anterior e o desgaste pode resultar no aumento de ressonância, levando-o a vibrar (Gonçalves, 2014).

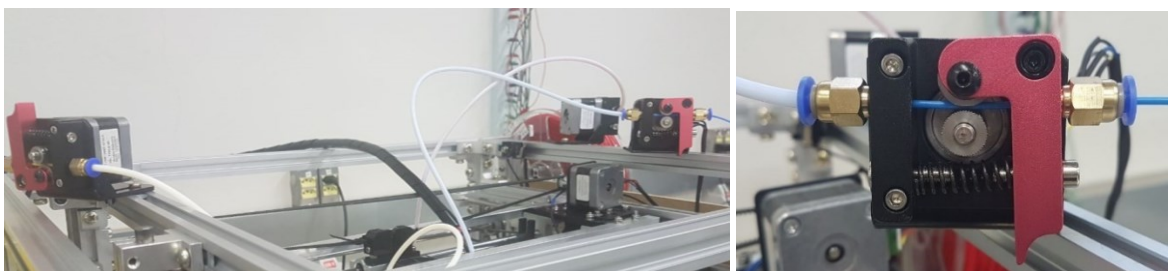
Figura 13 - Sistema de Transmissão de Potência (Fuso e Correias)



Fonte: Próprio autor

Todo objeto construído pela Impressora 3D é formado por um filamento enrolado em um carretel que será o material de consumo da mesma. Esse rolo é preso nas laterais da estrutura de forma a poder girar livremente a medida que o polímero vai sendo consumido. Para puxar o esse material do carretel e o empurrar para dentro do bico aquecido é necessário uma extrusora, e nessa impressora optou-se por utilizar as do tipo *bowden*, demonstradas na Figura 14, ao invés das do tipo *direct*, ligadas diretamente ao bico, pois além de o carro central da máquina ficar mais leve, e conseguindo, assim, atingir velocidades mais altas, a necessidade de três extrusoras para um único bico inviabiliza o uso da *direct*, devido a exigência de três bicos separados. Para conduzir o filamento da extrusora até o bico se utiliza um tubo flexível de material resistente a alta temperatura.

Figura 14 – Localização das extrusora do tipo *bowden*

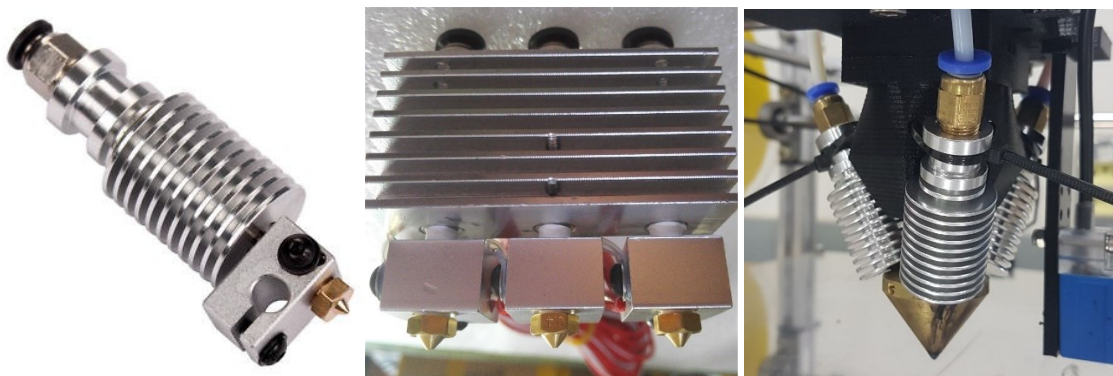


Fonte: Próprio autor

Nessa impressora, devido ao melhor custo-benefício, será utilizado o método de impressão FDM, no qual constrói objetos por extrusão de filamentos através de um dispositivo acoplado ao final robô cartesiano chamado de *hotend*, ou bico aquecido, com a função de derreter o plástico, e por meio do movimento no plano XY imprimir um desenho pré-estabelecido. Ao final do mesmo, é iniciada uma nova camada ao deslocar o Z para baixo em uma distância igual à espessura configurada, formando camadas superpostas de filamento até formar o objeto pretendido.

O bico aquecido, também chamado de *hotend*, é a ferramenta do robô cartesiano, responsável por esquentar e depositar o material de modo a converter a peça de um simples modelo computacional num objeto real e palpável. Todas as suas peças são metálicas, pois as temperaturas ficam por horas na casa dos 230°C. Consiste basicamente de um bloco de alumínio para dissipar o calor e um cabeçote que contém um bico circular com um diâmetro de 0,4 milímetros. O *hotend* utilizado nessa impressora, também chamado de diamante triplo, é um diferencial, pois possui a capacidade de entrada de até três tipos de filamentos diferentes em um único bico, podendo esses serem de materiais ou cores distintas. A diferença dos três tipos de bicos mais utilizados está mostrada na Figura 15, sendo o diamante triplo o último.

Figura 15 - Hotend comum simples, comum triplo e diamante triplo

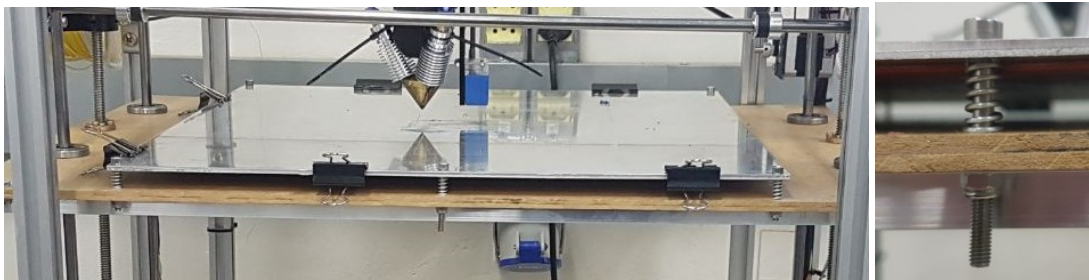


Fonte: Próprio autor

A área de trabalho, é composta de uma chapa de alumínio de 3mm de 42x42 cm, que fica afixada por meio de parafusos ajustáveis a uma superfície de MDF, responsável pelo deslocamento do eixo Z, visto que é nela que se encontram as castanhas e rolamentos lineares que se ligam aos fusos e guias lineares respectivamente. Acima do alumínio é colocado um

vidro de forma que o bico não entre em contato direto com o alumínio, não o danificando. Esse vidro é fixado por seis presilhas metálicas e é onde as peças ficam aderidas durante a impressão. Para fazer o nivelamento da mesa são usados seis conjuntos de parafusos e molas, detalhados na Figura 16, que ao serem ajustados proporcionam a elevação ou redução da altura da mesa de impressão.

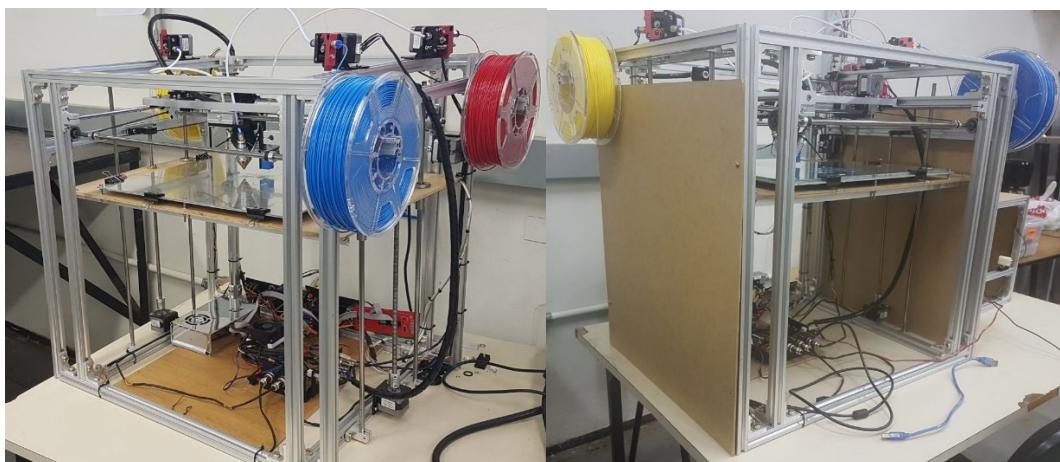
Figura 16 - Mesa de alumínio e parafuso de ajuste do nivelamento



Fonte: Próprio autor

E por fim, a estrutura projetada permite que seja feito a colocação de chapas de acrílico, vidro, MDF ou outro tipo de material de forma a vedá-la, criando uma estufa no seu interior, mantendo a temperatura ambiente constante, proporcionando uma qualidade de impressão maior, e criando um design diferenciado, semelhante as impressoras profissionais no mercado. Nesse modelo foram colocadas chapas em MDF devido ao baixo custos das mesmas. Há também, na estrutura, três suportes para os carretéis de filamento usados durante a impressão, mostrados na Figura 17.

Figura 17 - Estrutura final com três carretéis de filamentos



Fonte: Próprio autor

2.1.2. ELETRÔNICA

A eletrônica da Impressora 3D, é basicamente composta por um micro controlador, que é o cérebro da máquina, por sensores, capazes de coletar informações como posição e temperatura, por atuadores, capazes de transmitir comandos de deslocamento e de aquecimento e, por último, de dispositivo de interfaces com o usuário, como LCD, Diodos Emissores de Luz (LEDs), visores e, conexão via Porta Universal Serial (USB) para computador.

Optou-se por utilizar, como micro controlador principal, o Arduino, que permite a compatibilidade com várias outras plataformas complementares que facilitam conexão com a internet, bluetooth, diversas interfaces, sensores e motores, entre várias outras eletrônicas complementares.

O projeto Arduino, nascido na Itália em 2005, constitui uma plataforma de hardware e de software com o objetivo de possibilitar que pessoas não especialistas em programação e/ou em eletrônica possam desenvolver aplicações de objetos e ambientes interativos. Para isso, a proposta do projeto visa tanto a criação de um hardware fácil de manusear e com os recursos necessários para trabalhar com os "mundos" digital e analógico, quanto um software de desenvolvimento acessível para a programação dos projetos interativos. (Alves, Silva, Pinto, Sampaio, & Elia, 2012).

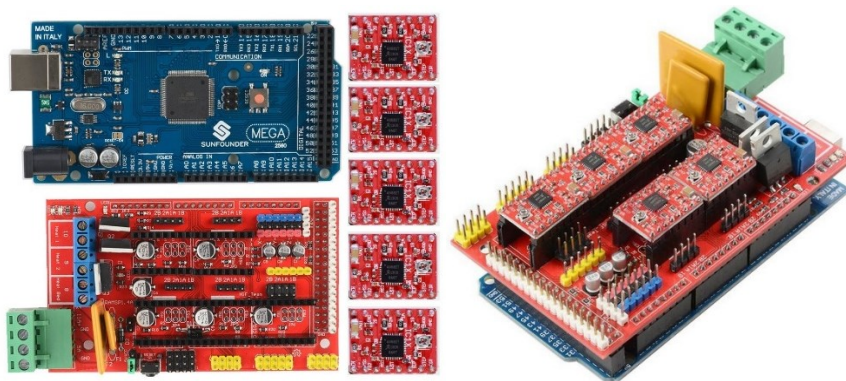
O Arduino Mega é uma placa de micro controlador baseada no ATmega2560. Ele tem 54 pinos de entrada e/ou saída digitais (dos quais 14 podem ser usados como saídas de PWM), 16 entradas analógicas, quatro portas seriais de hardware (UART's), um cristal oscilador de 16 MHz, uma conexão USB, um conector de alimentação e um botão de reset. Ele contém tudo o necessário para suportar o micro controlador, basta conectá-lo a um computador com um cabo USB ou ligá-lo a uma bateria ou uma fonte que converte corrente alternada (AC) para corrente contínua (DC).

Juntamente com a plataforma Arduino é acoplada uma placa de extensão (*shield*) de baixo custo, responsável por dividir entradas e saídas do micro controlador, simplificando as ligações da toda a eletrônica da Impressora 3D como os drivers dos motores de passo, os elementos de aquecimento, além de dos sensores, quer seja de fim de curso ou de temperatura. O design modular permite um trabalho eficiente, além de tornar fácil a substituição de qualquer peça e ter uma capacidade de expansão de componentes. Esse *shield* é chamado de

RAMPS, é a versão dela utilizada na Impressora 3D é a 1.4, a mais recente desenvolvida até o momento.

Junto a RAMPS, são conectados drivers de potência que recebe do Arduino os sinais de direção e passo e os traduzem em sinal de potência para os atuadores, no caso, os motores. Foram escolhidos os drivers Pololu A4988, por atenderem prontamente os requisitos de projeto e serem capazes de fazer o posicionamento do motor por meio da contagem de passos. Alimentam os motores bipolares em passo completo, meio passo e micro passo, com 35V e uma corrente de até 2A por enrolamento, além controlada por um potenciômetro presente em cada driver. O conjunto dos mesmos com a RAMPS e o Arduino é montado como mostra a Figura 18.

Figura 18 - Arduino (azul), RAMPS (vermelho) e 5 drivers



Fonte: Internet

Acionamentos elétricos podem ser utilizados em juntas prismáticas, com a transformação do movimento rotativo do motor em movimento linear por meio de um sistema de transmissão. Os acionadores elétricos mais comuns em uso nos robôs são: motor de corrente contínua ou DC, servo-motor e motor de passo (Carrara, 2010). Quando é conectado a sua fonte de energia, o eixo do motor responde com um movimento de rotação. A carga fixada no eixo do motor é movida pela rotação do eixo. O motor mais apropriado para uso na Impressora 3D é o de passo, igual da Figura 19, um tipo de motor elétrico que pode ser controlado por sinais digitais, mas com a possibilidade de se controlar sua velocidade, direção e ângulo, podendo girá-los num ângulo determinado com extrema precisão, obtendo ajustes finos de posicionamento.

Os motores de passo apresentam uma gama de rotação muito ampla que pode variar de 0 até 5400 rotações por minuto ou mais. Além de apresentar boa relação peso/potência, permitem ainda a inversão de rotação em pleno funcionamento. Algumas das vantagens do motor de passo em relação aos outros são a estabilidade e a precisão dos seus movimentos (Teixeira, 2001).

Os motores de passo podem ser bipolares ou unipolares. Em ambos os casos as fontes utilizadas são de tensão contínua e requerem um circuito digital que produza as sequências de sinais para que o motor funcione corretamente. Motores de passo podem ser acionados de diversas formas, sendo as mais comuns, o passo completo (pleno), meio passo e micro passo.

Figura 19 – Motor de passo Nema 17 bipolar



Fonte: Internet

Tipicamente, os motores de passo realizam movimentos discretos, cujos ângulos entre um passo e outro são da ordem de 90° ; 15° ; $7,5^\circ$; $3,6^\circ$; $1,8^\circ$; $0,9^\circ$ e $0,72^\circ$ graus. Um motor com ângulo de passo de $1,8^\circ$ por exemplo, executa 200 passos por volta completa do rotor. Os motores de passo possuem precisão de posicionamento de ângulo de passo da ordem de 5%. Mas para aplicações onde se necessita de movimentos incrementais muito pequenos, todas as resoluções no posicionamento citadas podem se mostrar inadequadas. Esse é o caso da Impressora 3D, onde se exigem movimentos discretos de maior precisão, o que requer técnicas para redução dos deslocamentos angulares a cada passo, chamada de micro passos.

O acionamento de um motor de passo pela técnica de fracionamento eletrônico do ângulo de passo, sendo possível dividir o ângulo de passo natural do motor em um ou mais subpassos, sem a necessidade de redutores mecânicos acoplados ao eixo. Este aspecto torna o motor de passo ainda mais atrativo para aplicações onde se deseja resolução e precisão de posicionamento (Freitas, 1998). Essa técnica, associada ao sistema de transmissão com fusos

e correia, faz com que a impressora execute movimentos exatos e com alta definição, e como exemplo, foi elaborado uma análise matemática.

Inicialmente, foi escolhido o motor Nema 17 da marca Minebea Matsushita e modelo 17PM-K049B com as características da Tabela 3, extraída do seu *datasheet*.

Tabela 3 - Características do Motor de Passo Adotado

Modelo	Ângulo de Passo (graus)	Sequência de condução	Corrente nominal (A)	Resistência (Ohms)	Torque de Retenção (mNm)	Inercia do rotor (g·cm ²)	Torque Residual (mNm)	Massa (g)
17PM-K049B	1.8	BI-POLAR	0.7	6.4	270	37	11.0	200

Fonte: <http://www.eminebea.com/>

Os motores de passo selecionados apresentam um ângulo de passo de 1.8°. E como já foi analisado, os drivers propostos funcionam em micro passo (1/16). Logo, o número de passos para esse motor girar uma volta completa é obtido através da equação:

$$\text{Num. de passos} = \frac{\text{Ângulo de rotação}}{\text{Ângulo de passo}} * \text{Num. de divisões entre os rolamentos}$$

$$\text{Num. de passos} = \frac{360}{1,8} * 16 = 3200$$

Por sua vez, os fusos T8 adotados apresentam uma distância entre os dentes, ou *pitch*, de 2mm. Deste modo é obtido quantos milímetros ele sobe em uma revolução completa do fuso, ou seja, o MPR (milímetro por revolução):

$$\text{MPR} = \frac{1}{\text{Pitch}} = \frac{1}{2} = 0,5$$

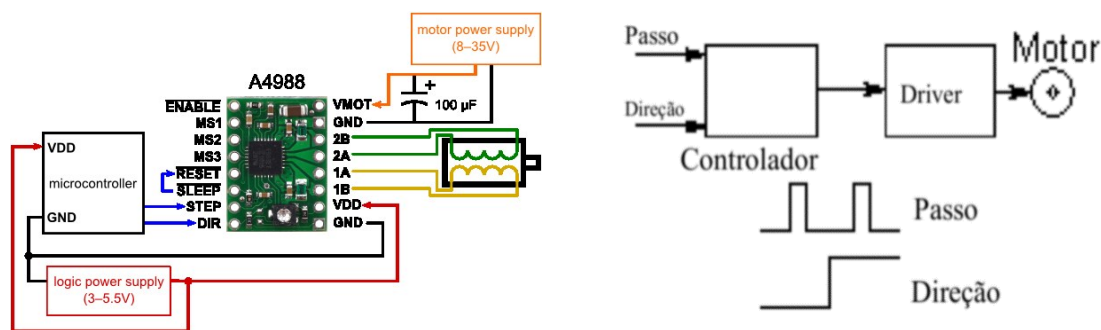
Em seguida é calculado o produto entre o número de passos obtidos na equação e o valor do MPR, e descoberto quantos passos são necessários para que seja deslocado 1mm linearmente, ou seja, PPM (passos por milímetro):

$$PPM = \text{Num. de passos} * MPR = 3200 * 0,5 = 1600$$

Dado a razão de conversão do fuso ser 1:1, pois há um acoplamento direto do mesmo com o eixo do motor, o que significa que o número de passos de ambos se mantém. Assim, o valor final é de 1600 passos por milímetro, ou seja, para movimentar 1mm é necessário que o motor execute 1600 passos. Resumindo, cada passo move linearmente apenas $6,25 * 10^{-6}$ mm. Por correia e polia, mantendo o *pitch* de 2mm, e como a relação não é mais de 1:1, e por isso, possui uma redução maior ainda, se obtém uma resolução mais precisa que a por fusos.

Os motores são ligados a um circuito excitador, constituído por um circuito sequencial (micro controlador) e um estágio amplificador de saída (driver de potência). O micro controlador é responsável por gerar a sequência de pulsos responsáveis por acionar os motores em um deslocamento contínuo e enviá-los para o driver, ligado de acordo com a Figura 20a, que amplifica as formas de onda e fornece corrente e tensão necessária para alimentá-los, segundo a Figura 20b.

Figura 20 - Esquema de ligação do driver (a) e sequência de pulsos enviados pelo Arduino (b)

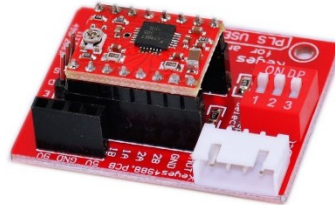


Fonte: Internet

Na Impressora 3D são utilizados seis motores de passos iguais os da Tabela 3, sendo um para movimentação do eixo X, outro para o Y, dois para os fusos do eixo Z e três para acionamento das extrusoras de filamentos. Porém, nota-se que a placa RAMPS 1.4 possui apenas a entrada de cinco drivers, e por isso seria necessário fazer a adição de uma plataforma

complementar para o sexto. Para fazer isso, foi adquirido um módulo da Figura 21, e conectado por meio de jumpers nas portas de saída disponíveis na RAMPS (64,59 e 44).

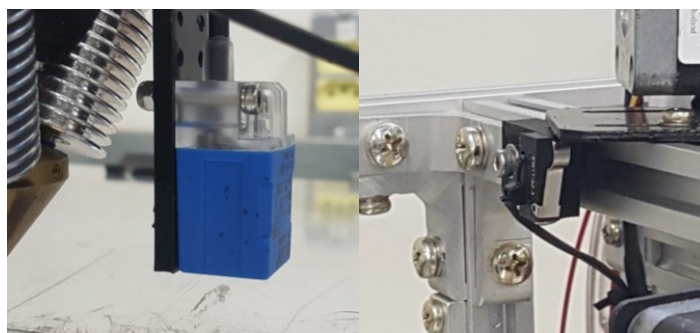
Figura 21 - Módulo complementar para sexto driver



Fonte: Internet

Nessa impressora são utilizados sensores para determinar a posição referência no deslocamento de cada eixo sendo, uma das formas mais tradicionais de controlar o curso dos eixos do robô cartesiano a chave de fim de curso, que posicionada nas extremidades dos mesmos, enviam um sinal assim que acionadas. Só assim o robô cartesiano tem um ponto de referência para cada grau de liberdade, caso contrário, perde a capacidade precisa de deslocamento das suas juntas, e por esse motivo é necessário que os sensores e os eixos estejam alinhados corretamente. Existem uma infinidade de tipos de chaves, porém nessa impressora adotou-se as de acionamento mecânico, para o X e Y e a indutiva para detectar a mesa de impressão no Z, demonstrados na Figura 22.

Figura 22 - Sensor indutivo e mecânico



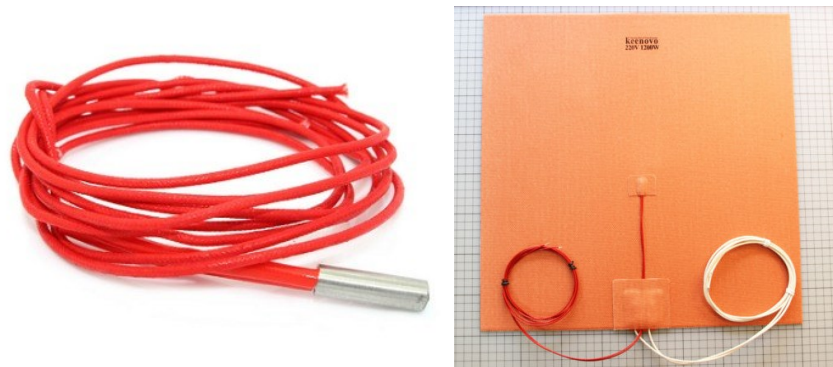
Fonte: Próprio autor

No método FDM, que consiste em sobrepor camadas de material sucessivamente, até que o objeto tridimensional seja formado, um cabeçote contendo um termoplástico, conhecido por *hotend*, jorra o respectivo material na bandeja do equipamento, que também é aquecida

por uma manta térmica, construindo, assim, a peça desejada. Para fazer o acionamento dos aquecedores do bico aquecido e da manta térmica são necessários comandos provenientes do Arduino.

Para fazer o aquecimento do bico e da bandeja de trabalho é utilizado, respectivamente, um cartucho resistor de 40W, ligado em 12V e capaz de chegar a temperaturas de até 350 graus, e uma manta térmica de silicone de 1200W ligado em 220V capaz de chegar a 232 graus. Ambos são mostrados na Figura 23.

Figura 23 - Cartucho aquecedor e manta térmica



Fonte: Internet

Para ambos os elementos de aquecimento, existe um sensor de temperatura, também chamado de termistores NTC de 100k Ω , que são dispositivos elétricos que mudam a sua resistência termicamente, ou seja, o valor da resistência é alterado conforme a temperatura a que estão sujeitos. Assim, através da alteração dos valores da resistência, e consequentemente, da corrente aos seus terminais, é possível determinar a temperatura na superfície desejada do cilindro.

Para o bico é usado um termistor de 100k (Figura 24a) ligado ao Arduino, e para a manta térmica um sensor PT100 que informa a um módulo de controle próprio o valor lido. Ao contrário de outros tipos de impressora que fazem o ajuste da temperatura da mesa no próprio Arduino com 12V, usou-se um outro controlador (Figura 24b) externo com rele de estado sólido, por estar trabalhando com uma tensão maior (220V) e para não sobrecarregar a fonte de alimentação. No entanto, o Arduino é informado por um termistor em tempo real da temperatura da mesa aquecida, só não é capaz de fazer o controle da mesma.

Figura 24 - Termistor (a) e módulo de controle da mesa com PT100 (b)



Fonte: Próprio autor

A entrada de energia alternada é feita por um cabo tripolar que se liga a um conector próprio no módulo de força da impressora, onde também está a chave liga-desliga geral, um porta fusível e uma saída de tensão alternada. Depois, essa energia passa por um botão de emergência, que corta toda a eletricidade caso acionado. A partir daí, é alimentado o módulo de controle da temperatura da mesa aquecida e a fonte chaveada de 360W, que fornece 12V em corrente continua de até 30A para os demais itens da Impressora 3D, como o Arduino, os dois coolers de resfriamento (um para a eletrônica e outro para as aletas do *hotend*) e o sensor indutivo. Todos os itens responsáveis pela alimentação são visualizados na Figura 25.

Figura 25 - Fonte, módulo de força e botão de emergência



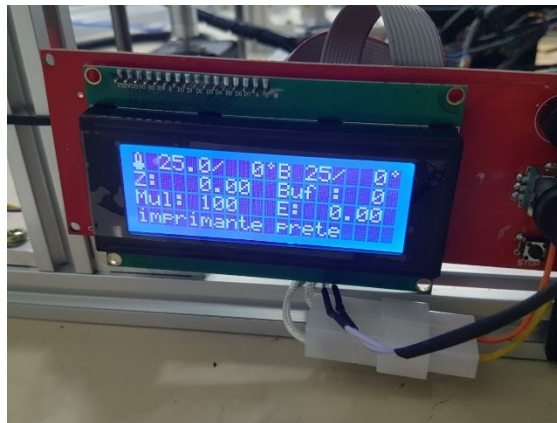
Fonte: Próprio autor

Para fazer o cálculo de demanda e dimensionar a fonte, foi suposto que cada motor trabalhasse na capacidade máxima do driver que é 2A, o cartucho aquecedor permanentemente ligado consumindo 3,5A e o Arduino e demais itens da eletrônica (coolers e LEDs) 1A. Somando de acordo com a fórmula abaixo, a fonte instalada supre toda a necessidade da eletrônica da Impressora.

$$I_{\max} = (6 * 2A) + 3,5A + 1A = 16,5A < 30A$$

Por fim, a Impressora 3D possui uma interface na qual permite o usuário visualizar e fazer o ajuste de vários parâmetros, além de iniciar e monitorar impressões, não necessitando de um computador ligado ao Arduino constantemente. Esse módulo de comunicação com o usuário é composto de uma tela de cristal líquido (LCD), um potenciômetro de ajuste e um buzzer, para alerta sonoro. Juntamente a esse módulo da Figura 26, possui uma leitora de cartão de memória que permite que o usuário imprima peças sem a necessidade da presença de um computador.

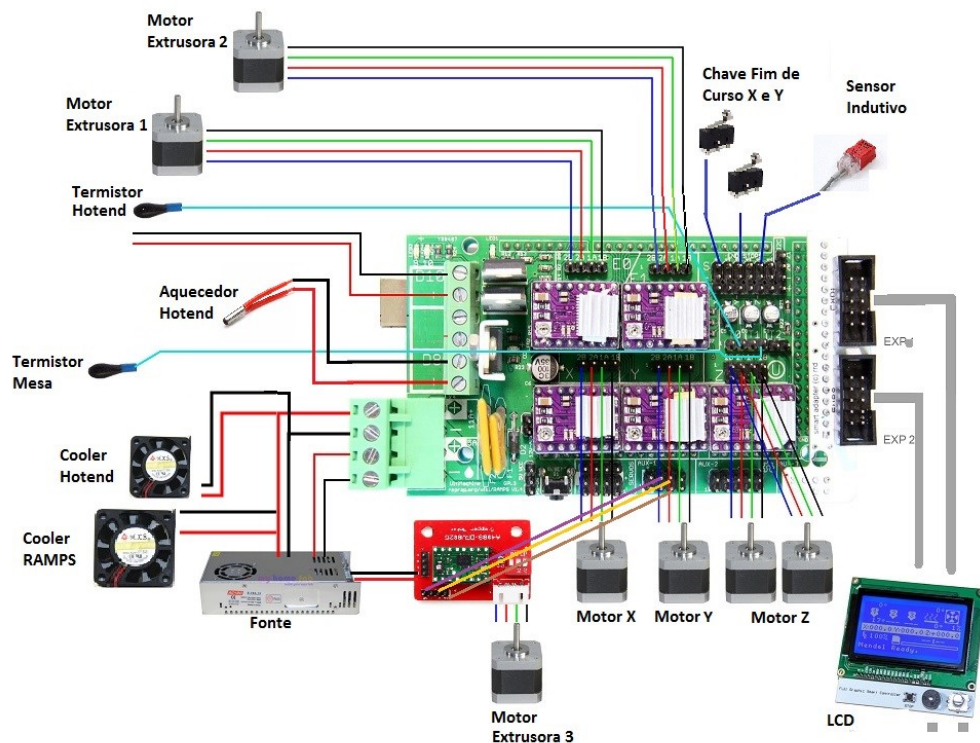
Figura 26 - Interface com o usuário (LCD)



Fonte: Próprio autor

Portanto, todos os itens da eletrônica que trabalham com 12V são ligados de acordo com o esquema da Figura 27. E para que as ligações dos atuadores/sensores com o Arduino permitam a desconexão fácil, facilitando a manutenção futuramente, optou-se por trabalhar com conectores metálicos, proporcionando uma conexão versátil e segura.

Figura 27 - Esquema de ligação da eletrônica



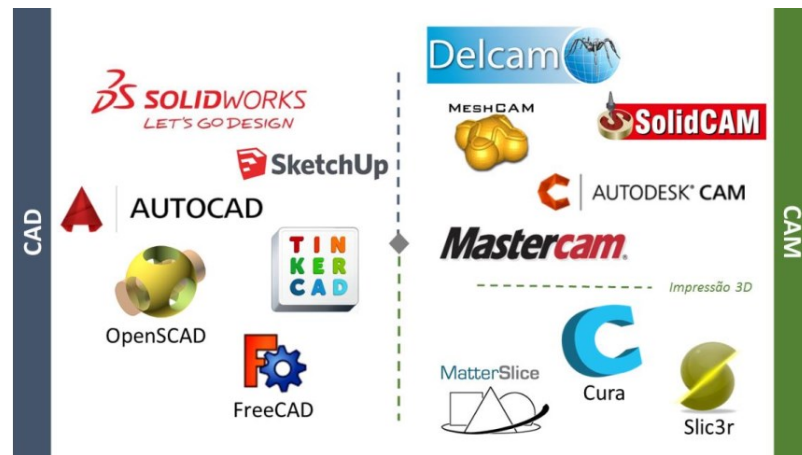
Fonte: Próprio autor

2.1.3. SOFTWARE

A Impressora 3D se comunica com o computador via USB, onde os softwares enviam os comandos necessários para que ela execute a impressão. O conjunto de ferramentas responsáveis pela criação, conversão para código e interpretação do objeto a imprimir é dividido em quatro partes: software CAD, CAM, de interface e *firmware* da eletrônica, que é a programação da máquina.

Os programas de CAD e CAM, exemplificados na Figura 28, fazem parte do topo da pirâmide e estão associados aos primeiros passos da prototipagem, responsáveis pela criação do objeto e conversão deste em código a ser interpretado pela máquina, respectivamente. Tipicamente o formato de arquivo exportado pelo sistema CAD tem extensão STL (STereoLithography), formato mais utilizado pelos programas de CAM para impressão 3D. A função deste conjunto é possibilitar desenvolver a partir da geometria do modelo do objeto o código conhecido como G-Code, a linguagem padrão para máquinas de prototipagem rápida.

Figura 28 - Softwares CAD e CAM



Fonte: (MOREIRA, 2016)

O G-Code descreve cada superfície horizontal através de comandos utilizados em máquinas de controle numérico, tais como: caminhe a uma velocidade de X m/s das coordenadas A até B com o laser cabeça de impressão ligada. Isso fará com que a Impressora 3D imprima uma linha sólida de espessura definida pela tecnologia, no plano definido pelas coordenadas A e B. O arquivo G-Code descreve todas as superfícies fatiadas em um único arquivo que é interpretado pela maioria das impressoras 3D (Takagaki, 2012).

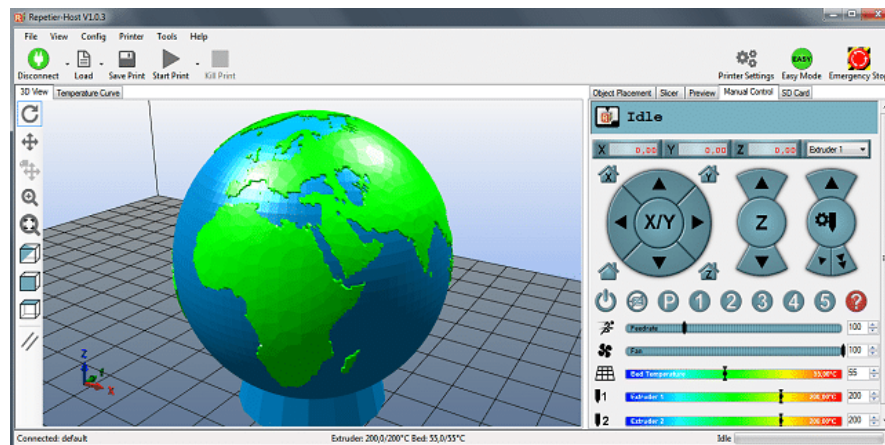
Os softwares CAM para impressoras 3D são denominados de “*Slicers*”, pois o modelo elaborado em CAD é cortado em fatias, constituindo as camadas que formam o objeto e geram, a partir de um arquivo STL proveniente de um CAD, um código com instruções codificadas para o firmware, que envia para a impressora os comandos para se construir a peça. Para cada objeto com parâmetros distinto é gerado um código diferente

O software de controle da impressora, chamado de *Host* estabelecerá a conexão entre o computador e a eletrônica da impressora e será o cérebro de todo o sistema, além de funcionar como interface com o usuário. Ele é o responsável por interpretar o código vindo do sistema CAM, contendo as operações necessárias para imprimir um objeto, e enviar para o mundo físico os sinais elétricos correspondentes. Tipicamente estes sinais são de passo, que faz o motor movimentar, e de direção, que diz qual o sentido do movimento, além de acionar o aquecimento do bico e da mesa.

O software de controle utilizado nessa impressora é o RepetierHost 1.6.2, mostrado na Figura 29, e tem como vantagem o fato do programa trazer integrado softwares CAM, o

que permite fazer upload direto do arquivo STL proveniente do CAD sem necessidade de uma ferramenta intermediária. Por ser de código livre, vem sendo aprimorado constantemente, e permite ao usuário adaptá-lo de acordo com a Impressora 3D a ser utilizada.

Figura 29 - Interface RepetierHost



Fonte: (Repetier, 2017)

O *Firmware* exerce a função de "Sistema operacional" da máquina, visto que é o conjunto de instruções operacionais que faz a impressora funcionar, responsável por colocar os parâmetros na mesma e fica gravado diretamente na eletrônica, ou seja, no micro controlador Arduino. É basicamente, a programação da máquina, que recebe os comandos do software de controle e os faz serem executados no equipamento e informa os dados lidos no mesmo pelos sensores ao programa. É desenvolvido em linguagem C, no ambiente de desenvolvimento do Arduino, chamado de IDE, que é de uso livre.

Existem diversos firmwares disponíveis na internet, cada um com suas vantagens e desvantagens. Após uma análise dos mesmos, e por um grande tempo já trabalhando no Firmware Repetier na primeira impressora construída, optou-se por continuar com ele nessa nova máquina com a versão 0.92.9. O mesmo está disponível no site www.repetier.com/firmware/v092/, e a parte do código onde são inseridos os parâmetros para os ajustes dessa Impressora 3D está no Anexo A desse trabalho.

2.1.4. MATERIAL DE CONSUMO

Existem diversos tipos de materiais e cores, como mostra a Figura 30, disponíveis para a impressão 3D, no qual alguns serão adquiridos buscando o seu estudo para analisar o que mais condiz com o projeto em questão, sendo os mais comuns o ABS (Acrilonitrila Butadieno Estireno) e o PLA (Ácido Polilático). É importante saber que existem outros tipos de filamentos bastante utilizados em aplicações mais específicas, tais como Nylon, PVA (*Polyvinyl Alcohol*), PETG (*Politereftalato de Etileno Glicol*) e Policarbonato, e, constantemente, está surgindo novos materiais.

Figura 30 - Filamentos em cores variadas



Fonte: Próprio autor

O ABS é o plástico mais comum para impressoras 3D utilizadas atualmente. Suas propriedades caracterizam por ser rígido e leve, com acabamento brilhante, o ponto de fusão varia de 215°C a 235°C. É bastante comum na indústria, muito utilizado na fabricação de produtos moldados para usos diversos, reciclável, derivado do petróleo, é um plástico que apresenta um ótimo resultado em resistência e flexibilidade no processo de impressão 3D, além de ser isolante elétrico. O ABS tem uma característica interessante, quando se imprime com cores diferentes, que é a capacidade de se misturar e gerar uma nova cor, como por exemplo, ao juntar filamento azul com amarelo, ele se transforma em verde.

O filamento plástico PLA, é um poliéster termoplástico feito com ácido láctico, biodegradável, é produzido a partir de fontes renováveis como milho, mandioca, cana de açúcar, entre outros. Nas impressoras 3D, é tão comum quanto o ABS, diferenciando pela maior facilidade de impressão, pois não requer mesa aquecida, pela temperatura de fusão ser

sensivelmente menor, entre 170 e 200°C e o apelo ecológico. Além de que o PLA, é mais rígido que o ABS, menos flexível, menos resistente a temperaturas muito altas e obtém um excelente acabamento superficial nas peças impressas.

2.2. SCANNER 3D

O princípio utilizado para o Scanner 3D fazer a reconstrução tridimensional é por meio de triangulação, através do método reflexivo ótico, no qual, se resume, segundo Gazziro (2011), a projeção de um feixe de linha vertical produzida por um emissor laser sobre a superfície do objeto a ser capturado; essa projeção é então capturada pela câmera e a distância até o objeto se dá através de cálculo geométrico.

Para cada posição do feixe de laser, uma câmera fixa focada no objeto em questão, capturará uma imagem. A imagem capturada será tratada por um software de filtragem, de modo a eliminar todas as cores da imagem que diferem do padrão do laser. O resultado dessa filtragem é uma linha vermelha que representa o contorno do objeto para aquela posição do laser. Assim, repete-se este processo, para cada giro da plataforma digitalizadora, criando-se um buffer que contém os contornos do objeto para cada posição na respectiva varredura. Dependendo da complexidade do mesmo uma única varredura pode não ser suficiente para capturar todas as faces e, por isso, o número de varreduras necessárias para digitalização será variável de acordo com o grau de detalhes do objeto e será definido pelo usuário no processo de calibração do scanner.

A etapa posterior a esta é transformar cada ponto bidimensional do contorno em um ponto tridimensional. Para este fim, será aplicada ao conjunto de pontos que compõem o contorno uma matriz de transformação. Uma vez calculados os pontos tridimensionais que representam o objeto, o próximo passo é tratar esta nuvem de pontos. A partir daí, tornasse necessário a aplicação de um algoritmo que conecte cada um dos pontos da nuvem a fim de produzir uma estrutura na forma de malha poligonal.

Construída a malha poligonal a próxima etapa da reconstrução tridimensional é aplicar a ela um algoritmo de renderização, para somente então, projetar o objeto digitalizado no display de saída, aplicando é claro, algoritmos de tratamento de superfícies escondidas. Cabe ressaltar aqui que o software permitirá que o usuário salve a nuvem de pontos obtida no

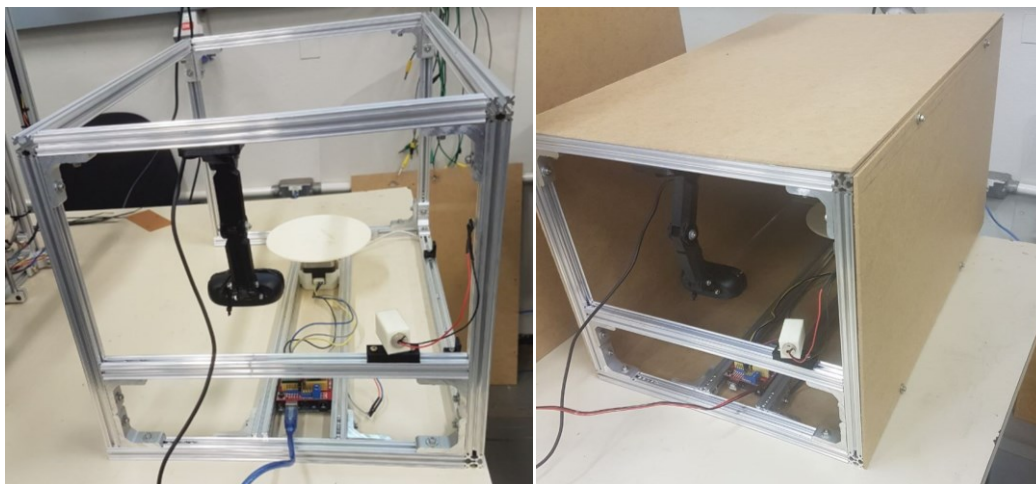
processo de digitalização no formato de arquivos com as extensões 3DS (3D Studio Max), DXF (AutoCAD) e até STL, para poder reimprimir o objeto caso desejado.

No desenvolvimento do Scanner 3D foi analisado a parte mecânica, como, a melhor estrutura física a ser utilizada, os mecanismos de tração para os elementos móveis da plataforma, a parte eletrônica, como o tipo de micro controlador, câmera, sensores e motores a serem usados, e por fim, o software que inclui a interface gráfica com o usuário, os mecanismos de programação e o programa de renderização e tratamento.

Devido a complexidade da construção do zero de um Scanner 3D, e a grande similaridade com um projeto já desenvolvido e robusto já mencionado, que é o FabScan, além de que esse digitalizador tridimensional não é o foco principal desse trabalho e sim, um item adicional a plataforma da Impressora 3D, optou-se por pegar o projeto como modelo e a partir daí adotou-se algumas mudanças pontuais, principalmente na parte mecânica.

A parte estrutural do Scanner é bem menos complexa do que a da Impressora, sendo composta de perfis estruturais de 15x15, cantoneiras de 90 graus e chapas em MDF, para fazer o fechamento da estrutura, mostrados na Figura 31.

Figura 31 - Estrutura scanner com e sem MDF

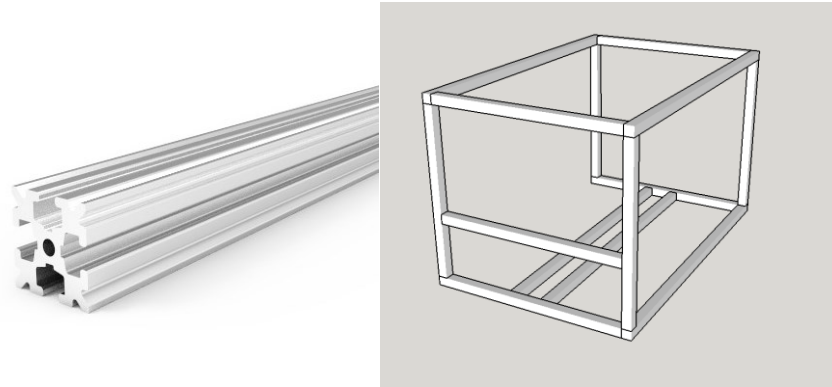


Fonte: Próprio autor

Adotou-se uma estrutura cúbica de tamanho suficiente para acomodar todos os componentes eletrônicos do scanner em seu interior, inclusive o objeto a ser digitalizado. Foram utilizados perfis estruturais, de tamanho relatado na Tabela 4, devido a facilidade de montagem, robustez e versatilidade, mesmos princípios adotados para a Impressora 3D,

porém optou-se por um de menor dimensão, 15x15 cm, devido a ausência de esforços físicos e movimentações em seu interior. O perfil e o projeto feito em Sketchup está na Figura 32.

Figura 32 - Perfil 15x15 e modelo da estrutura do scanner desenhado em software 3D



Fonte: Próprio autor

Tabela 4 - Tamanho Perfis Estruturais 15x15

Quantidade	Tamanho
3	500 mm
2	470 mm
2	316,5 mm
4	276 mm
3	301 mm

Fonte: Próprio autor

Já a parte eletrônica do Scanner 3D é composta basicamente por um micro controlador, um sistema de leitura ótica, que inclui uma câmera e um laser, e um único atuador, que é um motor para movimentação da plataforma digitalizadora do objeto. A alimentação da eletrônica é proveniente da fonte instalada na Impressora 3D. E nessa máquina não há presença de dispositivo de interface próprio, pois como é necessário que se conecte um computador, o mesmo serve também de interlocução com o usuário.

O micro controlador utilizado é o Arduino Uno R3, uma versão mais compacta do que o Mega, utilizado na Impressora 3D, devido a ausência de necessidade de tantas portas, e baixa capacidade de processamento exigido pela eletrônica do Scanner. Para controlar o motor de passo e acionar o módulo laser, foi utilizado uma placa de extensão SHIELD CNC v3.0 e

um driver de passo A4988. O conjunto de controle é demonstrado na Figura 33. Já para alimentar essa eletrônica foi adquirida uma fonte externa de 12V.

Figura 33 - Arduino UNO, placa de extensão CNC e driver



Fonte: Internet

A plataforma digitalizadora será dotada de um grau de rotação, objetivando assim, capturar detalhes do objeto em varreduras sucessivas. A rotação do disco será executada por um motor de passo e será controlada por software. Conforme já dito, para que os sinais de comandos do Arduino sejam enviados para o motor, há a presença de um driver que os converte em sinais elétricos, e aqui no scanner também será usado a técnica de micro passos, o que garantirá maior precisão no deslocamento. Já o feixe linear de laser será produzido por uma caneta ótica, do modelo semelhante a Figura 34, de 650nm e 5mW com tensão de 3-5V com traçado em linha de cor vermelha e foco ajustável, o que permite regular a espessura do feixe linear.

Figura 34 - Laser



Fonte: Internet

A câmera utilizada no projeto, responsável por capturar as imagens do objeto colocado sobre a base giratória foi o modelo C270 da fabricante Logitech, igual a Figura 35a.

É uma webcam de fácil aquisição no mercado, tem resolução de 3 megapixels e captura vídeos em alta definição (HD). Para uma melhor qualidade da imagem capturada pela câmera, foi inserido uma adaptação impressa em 3D para ajuste da distância focal, por meio da regulagem da objetiva manualmente, mostrado na Figura 35b.

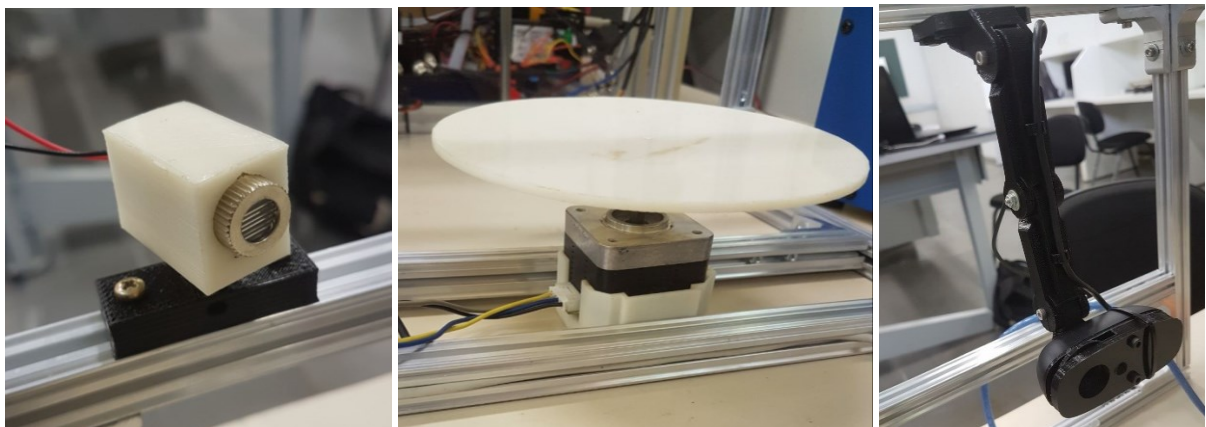
Figura 35 - Webcam C270 (a) e adaptação para ajuste de foco (b)



Fonte: Próprio autor

Para se integrar esses itens da eletrônica na estrutura mecânica, se imprimiu algumas peças, representadas na Figura 36, para servirem de suporte e possibilitarem o ajuste de posição para o laser, a plataforma digitalizadora e a câmera, que especialmente tem um braço articulado, permitindo ajuste da posição de acordo com o objeto a ser digitalizado.

Figura 36 - Suportes impressos em preto e branco para laser, motor e câmera



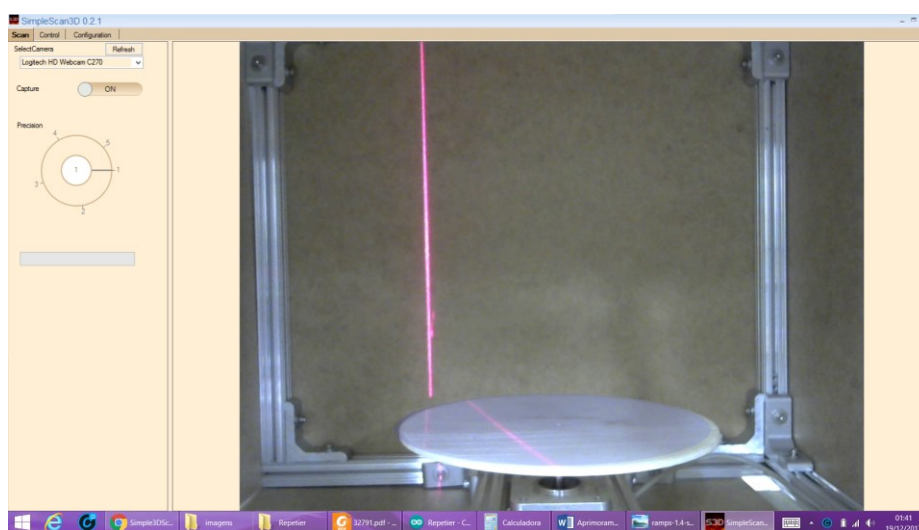
Fonte: Próprio autor

O processo de geração do modelo tridimensional obedece a duas etapas, uma sendo visão computacional que é a geração da nuvem de pontos que define a malha poligonal do

objeto em estudo a partir de imagens bidimensionais e da matriz de transformação, já outra fase compreende a computação gráfica que tem renderização da malha poligonal e sua respectiva visualização no dispositivo de saída.

Para fazer a digitalização do objeto tridimensional e gerar o arquivo com a nuvem de pontos será usado o software Simple3DScan que também permite gerenciar toda a eletrônica do scanner, como rotação da plataforma, o acionamento do feixe de laser, além de visualizar a imagem da câmera. Esse software fará uma filtragem para cada imagem capturada e gerará uma estrutura de dados com todos os pontos do contorno do objeto. Estas informações serão alocadas em um buffer circular para que possam ser utilizadas pela etapa de processamento, fazendo parte dos parâmetros desta etapa as estruturas de dados que contém as coordenadas correspondentes. Nota-se na Figura 37 a presença da plataforma digitalizadora e do laser ativado.

Figura 37 - Interface Simple3DScan



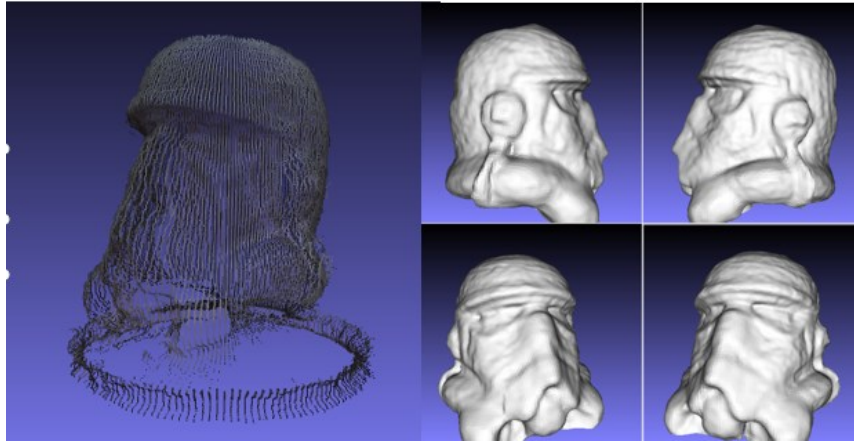
Fonte: Próprio autor

O Simple3DScan é a interface com o usuário e responsável por enviar os comandos diretamente para o *firmware* (Anexo B), que da mesma maneira da Impressora 3D está inserido dentro da memória do Arduino Uno e é o cérebro da máquina, sendo responsável por controlar todo o hardware, exceto a câmera que é ligada diretamente via USB no computador.

O processamento da nuvem de pontos é feito por um outro software, específico, chamado MeshLab, no qual o usuário pode trabalhar com várias ferramentas disponíveis, que

vão desde a reconstrução da malha do objeto a partir dos pontos recebidos, formando uma espécie de esqueleto, com as arestas e os polígonos, até a aplicação de algoritmos de preenchimento, permitindo a formação de superfícies e uma visualização mais coerente como mostra a Figura 38.

Figura 38 - Pontos lidos em um escaneamento e tratamento



Fonte: Próprio autor

2.3. ORÇAMENTO

2.3.1. MECÂNICA

A parte mecânica contempla toda a estrutura física da Impressora e do Scanner 3D, incluindo a parte de acabamento. Para tornar o produto de fácil acesso, todas as peças, relacionadas na Tabela 5, foram adquiridas no Brasil, seja via internet ou em lojas de Uberlândia. Uma pequena parte foi impressa em 3D no próprio laboratório, para suprir algumas peças que não se encontrava no mercado, devido as otimizações e inovações feitas nesse modelo de impressora, como o sistema de suporte para o bico triplo na impressora e o sistema de suporte para a câmera no scanner.

Tabela 5 - Custo Mecânica Impressora 3D

Material	Valor Unitário	Quantidade	Subtotal
Perfil estrutural de alumínio 20x20 mm (metro)	R\$ 17,00	10	R\$ 170,00
Guia linear aço tratado (metro)	R\$ 20,00	5	R\$ 100,00
Cantoneira ½ polegada alumínio (metro)	R\$ 2,00	6	R\$ 12,00
Chapa alumínio 3mm - 42x 42 cm	R\$ 70,00	1	R\$ 70,00
Vidro comum 4mm - 41x 39 cm	R\$ 20,00	1	R\$ 20,00
MDF 2mm (m^2)	R\$ 15,00	4	R\$ 60,00
Cantoneira 90 graus	R\$ 2,00	40	R\$ 80,00
Fuso T8 40cm	R\$ 60,00	2	R\$ 120,00
Castanha trapezoidal T8	R\$ 15,00	2	R\$ 30,00
Acoplamento flexível em Alumínio 5x8mm	R\$ 20,00	2	R\$ 40,00
Correia 6mm (metro)	R\$ 20,00	4	R\$ 80,00
Polia em alumínio de 20 Dentes	R\$ 20,00	7	R\$ 140,00
Rolamento 608	R\$ 8,00	3	R\$ 24,00
<i>Pillow block</i> longo 8mm	R\$ 40,00	4	R\$ 160,00
Rolamento linear longo 8mm	R\$ 25,00	4	R\$ 100,00
Mancal eixo 8mm	R\$ 18,00	4	R\$ 72,00
Suporte alumínio para eixo de 8mm	R\$ 15,00	16	R\$ 240,00
Extrusora MK8 completa em alumínio	R\$ 50,00	3	R\$ 150,00
Tubo PTFE (metro)	R\$ 20,00	1	R\$ 20,00
Outros	R\$ 100,00	1	R\$ 100,00
Total			R\$ 2988,00

Fonte: Próprio autor

Já para o Scanner 3D, a parte mecânica é menor, se resumindo a três itens, relacionados na Tabela 6, e mais algumas partes impressas.

Tabela 6 - Custo Mecânica Scanner 3D

Material	Valor Unitário	Quantidade	Subtotal
Perfil estrutural de alumínio 15x15 mm (metro)	R\$ 11,00	6	R\$ 66,00
Cantoneira alumínio 90 graus	R\$ 2,00	20	R\$ 40,00
MDF 2mm (m^2)	R\$ 15,00	1,5	R\$ 22,00
Outros	R\$ 40,00	1	R\$ 40,00
Total			R\$ 168,00

Fonte: Próprio autor

Os materiais da categoria outros inclui algumas ferragens que completam a parte mecânica de ambas as máquinas, como parafusos, porcas, arruelas, presilhas, molas e abraçadeiras de nylon. Esses itens foram todos comprados em estabelecimentos comerciais de Uberlândia.

2.3.2. ELETRÔNICA

A parte eletrônica da Impressora e do Scanner 3D inclui toda os componentes eletrônicos e a alimentação de ambas as máquinas de acordo com as Tabelas 7 e 8. A maioria das peças foram adquiridas via internet, e dessas, somente duas foram compradas no exterior, devido à falta no mercado nacional, que foi o bico triplo e a manta térmica para a mesa de alumínio nas dimensões de 40 por 40 centímetros. A eletrônica do Scanner, por ser mais simples, é também mais barata, e por ser um módulo da Impressora 3D, é aproveitado a fonte de alimentação da mesma, visto que é usada para alimentar apenas um motor de 12V.

Tabela 7 - Custo Eletrônica Impressora 3D

Material Permanente	Valor Unitário	Quantidade	Subtotal
Arduino Mega 2560 R3 com cabo USB	R\$ 70,00	1	R\$ 70,00
Placa de extensão RAMPS 1.4 e para sexto driver	R\$ 60,00	1	R\$ 60,00
Driver motor de passo Pololu com dissipador	R\$ 15,00	6	R\$ 90,00
Coolers	R\$ 15,00	2	R\$ 30,00
Motor Nema 17	R\$ 70,00	7	R\$ 490,00
Fonte chaveada 360W	R\$ 80,00	1	R\$ 80,00
Bico <i>hotend</i> triplo com aquecedor e termistor	R\$ 350,00	1	R\$ 350,00
Manta térmica - 40x40cm com termistor	R\$ 500,00	1	R\$ 500,00
Chave fim de curso mecânica	R\$ 3,00	2	R\$ 6,00
Sensor indutivo	R\$ 35,00	1	R\$ 35,00
Interface LCD com leitor de cartão de memória	R\$ 150,00	1	R\$ 150,00
Módulo de controle de temperatura da mesa com relê	R\$ 150,00	1	R\$ 150,00
Módulo de alimentação com cabo	R\$ 25,00	1	R\$ 25,00
Botão de emergência	R\$ 30,00	1	R\$ 30,00
Outros	R\$ 80,00	1	R\$ 80,00
Total			R\$ 2.146,00

Fonte: Próprio autor

Tabela 8 - Custo Eletrônica Scanner 3D

Material Permanente	Valor Unitário	Quantidade	Subtotal
Plataforma Arduino Uno	R\$ 50,00	1	R\$ 50,00
Placa de Extensão Arduino CNC V3	R\$ 40,00	1	R\$ 40,00
Driver motor de passo Pololu com dissipador	R\$ 15,00	1	R\$ 15,00
Laser	R\$ 30,00	1	R\$ 30,00
Motor Nema 17	R\$ 70,00	1	R\$ 70,00
Câmera Logitech C270	R\$ 180,00	1	R\$ 180,00
Cabo USB	R\$ 10,00	1	R\$ 10,00
Outros	R\$ 20,00	1	20,00
Total			R\$ 415,00

Fonte: Próprio autor

Além dos itens relatados nas Tabelas 7 e 8, foram adquiridos materiais para montagem da eletrônica, como conectores, terminais, jumpers, fiação de diversas bitolas, borracha termo retrátil, fitas de alta temperatura, organizador de fios, botões, LEDs e estanho. Porém isso não teve um custo alto, chegando tudo ao custo máximo de R\$ 100,00.

2.3.3. SOFTWARES

Todos os programas e firmwares utilizados são gratuitos, tanto da Impressora 3D quanto do Scanner 3D e estão disponíveis na rede, visto que ambos os equipamentos são de código livre, sendo assim, nessa parte não tivemos custos. No entanto, foi adquirido um aplicativo para celular, que monitora e informa o status da impressão em tempo real que teve um custo de R\$ 15,00.

2.3.4. MATERIAL DE CONSUMO

Esses materiais serão adquiridos em forma de fio, com diâmetro de 1,75 mm, enrolados em carretel. Foram adquiridos dois carretéis de 1kg do ABS nas cores branco e preto, e, quatro rolos do PLA nas tonalidades preto, amarelo, azul e vermelho, conforme descrito na Tabela 9.

Tabela 9 - Custo Material de Consumo (Filamentos)

Material de Consumo	Valor Unitário	Quantidade	Subtotal
Carretel PLA 1,75mm	R\$ 130,00	4	R\$ 520,00
Carretel ABS 1,75mm	R\$ 90,00	2	R\$ 180,00
Total			R\$ 700,00

Fonte: Próprio autor

2.3.5. CUSTO FINAL

O desenvolvimento do projeto foi realizado nas dependências do Laboratório de Máquinas Especiais da Faculdade de Engenharia Elétrica da Universidade Federal de Uberlândia. Portanto, o custo final da Impressora e do Scanner construídos foram, respectivamente, de R\$ 5.134,00 e R\$ 583,00, de acordo com a Tabela 10, bancados exclusivamente com recursos financeiros próprios do aluno.

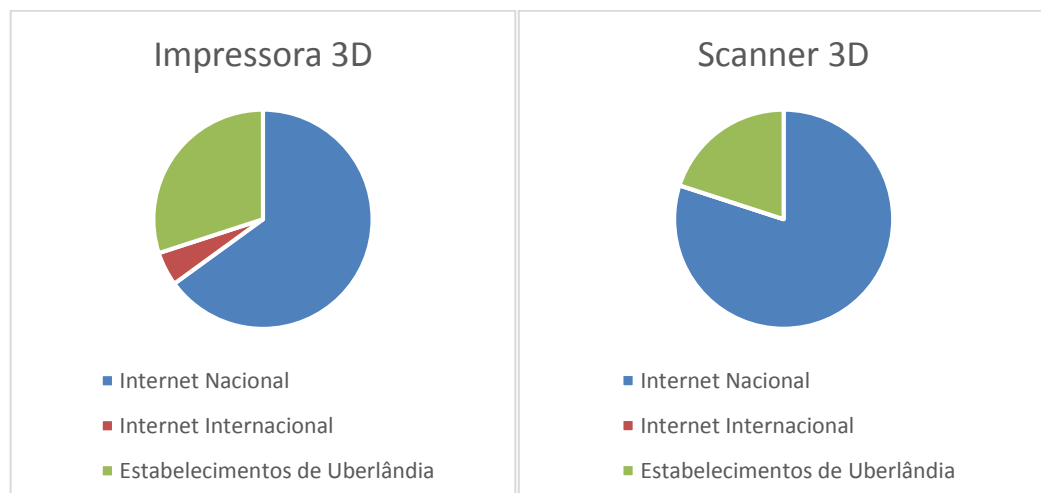
Tabela 10 - Custo Final Impressora e Scanner 3D

Máquina	Divisão	Custo	Subtotal
Impressora 3D	Mecânica	R\$ 2.988,00	R\$ 5.134,00
	Eletrônica	R\$ 2.146,00	
Scanner 3D	Mecânica	R\$ 168,00	R\$ 583,00
	Eletrônica	R\$ 415,00	
Material de Consumo	Filamentos	R\$ 700,00	R\$ 700,00
Total			R\$ 6.417,00

Fonte: Próprio autor

A aquisição de todos os itens resume ao Gráfico 1 com o percentual de aquisição do material, separado em categorias das quais foram compradas no mercado nacional ou internacional.

Gráfico 1 - Aquisição de peças no mercado



Fonte: Próprio autor

3. CONCLUSÃO

Atualmente a informação, o conhecimento e a capacidade de resposta às necessidades do mercado, são ativos valiosos que as empresas utilizam para estarem competitivas em um mercado que exige uma grande variedade de produtos de qualidade com tempo e custo reduzido, para isso as empresas estão utilizando as novas tecnologias de informação para integrar vários setores das organizações por meio do conhecimento dos indivíduos e de tecnologias e equipamentos, capazes de promover e interpretar os resultados desta integração (Friedel & Liedtka, 2007).

A partir de sites conceituados percebe-se que a compra de uma impressora 3D pronta, atualmente, varia de R\$1.500,00 (RepRap simples) a R\$100.000,00 (modelo SLS profissional). Uma máquina similar no mercado a criada neste trabalho, com capacidade de impressão de até três materiais e área útil de impressão equivalente, custa cerca de R\$15.000,00, enquanto a construída teve custo de R\$ 5.134,00, ou seja, 65% mais barata, mostrando a viabilidade de montá-la ao invés de comprar uma pronta. Até por que, mesmo as profissionais no mercado apresentam muitas limitações, como filamentos exclusivos e de preço elevado, por serem aceitos somente os do próprio fabricante, e engessamento do software e do hardware, sem possibilidade de mudanças, ao contrário da construída que por ser *open source*, permite melhorias contínuas tanto na parte de software, quanto na de hardware, além de aceitarem qualquer filamento comercializado.

Já o scanner desenvolvido que teve um custo de R\$ 583,00, e se comparado aos existentes no mercado de precisão semelhante, os quais variam o valor de R\$ 2.000,00 a R\$ 5.000,00, mostra que foi atingido o objetivo de montar um de baixo custo, possibilitando, assim, a popularização dessa tecnologia de digitalização tridimensional no país. Esse equipamento pode ser aprimorado, visto que a versão atual, ainda tem uma precisão mediana, muito aquém da resolução que impressora 3D possui, além de que a área de escaneamento é reduzida perante a de impressão.

E devido à falta de tempo e recursos financeiros, algumas melhorias projetadas na impressora e no scanner não foram implementadas, como a substituição das chapas em MDF por acrílico, a criação de um sistema para limpeza do bico de impressão, um aproveitamento melhor da área útil de impressão, a criação de uma central de controle externa, a integração do

hardware e do software de ambas as máquinas, dentre outras melhorias, que poderão ser executadas no futuro.

A máquina projetada se destina tanto a projetistas, engenheiros, hobbistas, estudantes de engenharia ou cursos técnicos, como para crianças, adolescentes e o público leigo em geral. Para o público técnico, a máquina tem como seu principal uso, a criação de peças projetadas a partir de softwares de CAD no conceito conhecido como prototipagem rápida aditiva. Com isso estes terão uma poderosa ferramenta para criar rapidamente pequenos protótipos funcionais e testar peças, encurtando e melhorando o ciclo de desenvolvimento de novos produtos. Por outro lado, o público em geral terá em mãos um equipamento capaz de criar objetos personalizados. Crianças e adolescentes poderão criar brinquedos, cartões, brindes, gravuras em alto e baixo relevo entre outros. As aplicações para o scanner são as mais diversas em vários setores da indústria e educação.

Por fim, foram realizados diversos testes em ambos os equipamentos construídos e constatou-se que apresentaram resultados de acordo com os desejados. A maior dificuldade encontrada, é a etapa de calibração de ambas as estruturas que demanda um tempo demasiado até se achar os diversos parâmetros necessários dos softwares, visto que são estruturas inéditas, porém que executam tarefas executadas por uma infinidade de máquinas, que é a impressão e o escaneamento em 3D.

REFERÊNCIAS BIBLIOGRÁFICAS

(s.d.). Acesso em 30 de Março de 2013, disponível em Arduino:

<http://arduino.cc/en/Main/arduinoBoardMega>

Additive/Subtractive manufacturing research and development in Europe. (2004). Acesso em 8 de Outubro de 2011, disponível em WORLD TECHNOLOGY EVALUATION CENTER, INC. – WTEC: <http://www.wtec.org/additive/report/additive-report.pdf>

Alves, R. M., Silva, A. L., Pinto, M. d., Sampaio, F. F., & Elia, M. d. (2012). Uso do Hardware Livre Arduino em Ambientes. Jornada de Atualização em Informática na Educação - JAIE.

Barros, M. (s.d.). *Apostila de Robótica*. Acesso em 29 de Março de 2013, disponível em Site da Zeugma Informática: <http://www.zeugma.com.br/marcosbarros/rob%C3%B3tica.pdf>

Carrara, V. (2010). *Apostila de Robótica*. Universidade Braz Cubas.

Cravo, G. L., Lima, L. E., & Andreão, R. V. (2009). Controle de um Robô Cartesiano para Produção Gráfica por Comandos de Voz. *COBENGE*. Recife - PE.

Cunico, M. W. (2015). *Impressoras 3D: o novo meio produtivo*. Curitiba: Concep3d Pesquisas Científicas.

Doro, M. M. (12 de Junho de 2004). Sistemática para Implantação da Garantia da Qualidade em Empresas Montadoras de Placas de Circuito Impresso. Florianópolis, SC, Brasil: Universidade Federal de Santa Catarina.

Ferreira, J. C., Madureira, H., & Castro, J. (2006). Integration of VP/RP/RT/RE/RM for rapid product and process development. *Rapid Prototyping Journal*, Vol. 12, pp. 18-25.

Filho, F. d., & Nogueira, J. M. (2006). Sistemas de inovação e promoção tecnológica regional e local no Brasil. *Interações - Revista Internacional de Desenvolvimento Local*, v.8, nº 13, 107-117.

Foggiatto, J. A. (2006). O Uso da prototipagem rápida na área médico-odontológica. *Revista Tecnologia & Humanismo*, v. 20, n. 30, 60-68.

Freitas, M. A. (Abril de 1998). ACIONAMENTO DO MOTOR DE PASSO NO MODO DE FRACIONAMENTO ELETRÔNICO DO ÂNGULO DE PASSO. Uberlândia, MG, Brasil: UNIVERSIDADE FEDERAL DE UBERLÂNDIA.

Friedel, R., & Liedtka, J. (2007). Possibility thinking: lessons from breakthrough engineering. *Journal of Bussines Strategy*, pp. 30-7.

- G1. (02 de Dezembro de 2014). Fonte: Globo.com: <http://g1.globo.com/minas-gerais/triangulo-mineiro/noticia/2014/03/estudante-de-engenharia-eletrica-da-ufu-cria-impressora-3d.html>
- Garcia, L. H. (2010). Desenvolvimento e fabricação de uma mini-impressora 3D para cerâmicas. São Carlos, SP, Brasil: Universidade de São Paulo.
- Gazziro, M. (2011). *Scanner 3D Construção e Aplicações*. biblioteca 24 horas.
- Gonçalves, J. F. (Outubro de 2014). Impressora 3D com LinuxCNC. Porto.
- Grimm, T. (2005). Choosing the Right RP System. A Study of Seven RP System.
- International Organization for Standardization. (10218).
- Junior, A. S. (Outubro de 2008). Análise Comparativa entre os Processos de Prototipagem Rápida na Concepção de Novos Produtos: Um Estudo de Caso para Determinação do Processo mais Indicado. Curitiba, PR, Brasil: Pontifícia Universidade Católica do Paraná.
- Junior, A. S., Junior, O. C., & Neto, A. I. (2007). ANÁLISE COMPARATIVA ENTRE OS PROCESSOS DE PROTOTIPAGEM RÁPIDA POR DEPOSIÇÃO OU REMOÇÃO DE MATERIAL NA CONCEPÇÃO DE NOVOS PRODUTOS. *XXVII Encontro Nacional de Engenharia de Produção*. Curitiba, PR, Brasil: Pontifícia Universidade Católica do Paraná.
- MOREIRA, S. S. (2016). PROJETO E CONSTRUÇÃO DE UM SCANNER TRIDIMENSIONAL. Ilha Solteira.
- Nishida, J. C. (Novembro de 2011). IMPRESSORA 3D DE BAIXO. *Trabalhos de Conclusão de Curso*. São Paulo, SP, Brasil: Escola Politécnica da Universidade de São Paulo.
- Repertier. (25 de Novembro de 2017). Fonte: Site Oficial Repertier: <https://www.repertier.com/documentation/repertier-host/>
- RepRap, S. o. (10 de 06 de 2017). Fonte: <http://reprap.org>
- Souza, A. F., & Ulbrich, C. B. (2009). *Engenharia Integrada por Computadores e Sistemas Cad / Cam / Cnc - Princípios e Aplicações*. São Paulo: Artliber.
- Takagaki, L. K. (2012). Tecnologia de Impressão 3D. *Revista Inovação Tecnológica*, 28-40.
- Teixeira, E. N. (2001). A0A0 – Um Protótipo de Robô Autônomo. Uberlândia, MG, Brasil: Monografia.
- The FabScan Project*. (10 de Novembro de 2017). Fonte: <https://hci.rwth-aachen.de/fabscan>
- Volpato, N. (2007). *Prototipagem Rápida: Tecnologias e Aplicações*. São Paulo: Edgard .

ANEXO A – Código de Parâmetros para o Firmware da Impressora 3D (config.h)

```

#define CONFIGURATION_H
#define CONFIGURATION_H

#define NUM_EXTRUDER 3
#define MOTHERBOARD 33
#include "pins.h"

#undef FAN_BOARD_PIN
#define FAN_BOARD_PIN -1
#define BOARD_FAN_SPEED 255
#define FAN_THERMO_PIN -1
#define FAN_THERMO_MIN_PWM 128
#define FAN_THERMO_MAX_PWM 255
#define FAN_THERMO_MIN_TEMP 45
#define FAN_THERMO_MAX_TEMP 60
#define FAN_THERMO_THERMISTOR_PIN -1
#define FAN_THERMO_THERMISTOR_TYPE 1

#define BLUETOOTH_SERIAL -1
#define BLUETOOTH_BAUD 115200
#define MIXING_EXTRUDER 1

#define DRIVE_SYSTEM 0
#define XAXIS_STEPS_PER_MM 80
#define YAXIS_STEPS_PER_MM 80
#define ZAXIS_STEPS_PER_MM 1600
#define EXTRUDER_FAN_COOL_TEMP 50
#define PDM_FOR_EXTRUDER 1
#define PDM_FOR_COOLER 1
#define
DECOUPLING_TEST_MAX_HOLD_VARIANCE
20

#define DECOUPLING_TEST_MIN_TEMP_RISE 1
#define KILL_IF_SENSOR_DEFECT 0
#define RETRACT_ON_PAUSE 5
#define PAUSE_START_COMMANDS ""
#define PAUSE_END_COMMANDS ""

#define EXT0_X_OFFSET 0
#define EXT0_Y_OFFSET 0
#define EXT0_Z_OFFSET 0
#define EXT0_STEPS_PER_MM 96
#define EXT0_TEMPSSENSOR_TYPE 1
#define EXT0_TEMPSSENSOR_PIN TEMP_0_PIN
#define EXT0_HEATER_PIN HEATER_0_PIN
#define EXT0_STEP_PIN ORIG_E0_STEP_PIN
#define EXT0_DIR_PIN ORIG_E0_DIR_PIN
#define EXT0_INVERSE 0
#define EXT0_ENABLE_PIN
ORIG_E0_ENABLE_PIN
#define EXT0_ENABLE_ON 0
#define EXT0_MAX_FEEDRATE 50
#define EXT0_MAX_START_FEEDRATE 0
#define EXT0_MAX_ACCELERATION 5000
#define EXT0_HEAT_MANAGER 0
#define EXT0_WATCHPERIOD 1
#define EXT0_PID_INTEGRAL_DRIVE_MAX 230
#define EXT0_PID_INTEGRAL_DRIVE_MIN 40
#define EXT0_PID_PGAIN_OR_DEAD_TIME 7
#define EXT0_PID_I 2
#define EXT0_PID_D 40
#define EXT0_PID_MAX 255
#define EXT0_ADVANCE_K 0
#define EXT0_ADVANCE_L 0
#define EXT0_ADVANCE_BACKLASH_STEPS 0
#define EXT0_WAIT_RETRACT_TEMP 150

```

```

#define EXT0_WAIT_RETRACT_UNITS 0
#define EXT0_SELECT_COMMANDS ""
#define EXT0_DESELECT_COMMANDS ""
#define EXT0_EXTRUDER_COOLER_PIN -1
#define EXT0_EXTRUDER_COOLER_SPEED 255
#define EXT0_DECOUPLE_TEST_PERIOD
1200000
#define EXT0_JAM_PIN -1
#define EXT0_JAM_PULLUP 0

#define EXT1_X_OFFSET 0
#define EXT1_Y_OFFSET 0
#define EXT1_Z_OFFSET 0
#define EXT1_STEPS_PER_MM 96
#define EXT1_TEMPSENSOR_TYPE 1
#define EXT1_TEMPSENSOR_PIN TEMP_0_PIN
#define EXT1_HEATER_PIN 11
#define EXT1_STEP_PIN ORIG_E1_STEP_PIN
#define EXT1_DIR_PIN ORIG_E1_DIR_PIN
#define EXT1_INVERSE 0
#define EXT1_ENABLE_PIN
ORIG_E1_ENABLE_PIN
#define EXT1_ENABLE_ON 0
#define EXT1_MAX_FEEDRATE 50
#define EXT1_MAX_START_FEEDRATE 0
#define EXT1_MAX_ACCELERATION 5000
#define EXT1_HEAT_MANAGER 0
#define EXT1_WATCHPERIOD 1
#define EXT1_PID_INTEGRAL_DRIVE_MAX 230
#define EXT1_PID_INTEGRAL_DRIVE_MIN 40
#define EXT1_PID_PGAIN_OR_DEAD_TIME 7
#define EXT1_PID_I 2
#define EXT1_PID_D 40
#define EXT1_PID_MAX 255
#define EXT1_ADVANCE_K 0
#define EXT1_ADVANCE_L 0
#define EXT1_ADVANCE_BACKLASH_STEPS 0

```

```

#define EXT1_WAIT_RETRACT_TEMP 150
#define EXT1_WAIT_RETRACT_UNITS 0
#define EXT1_SELECT_COMMANDS ""
#define EXT1_DESELECT_COMMANDS ""
#define EXT1_EXTRUDER_COOLER_PIN -1
#define EXT1_EXTRUDER_COOLER_SPEED 255
#define EXT1_DECOUPLE_TEST_PERIOD
1200000
#define EXT1_JAM_PIN -1
#define EXT1_JAM_PULLUP 0

#define EXT2_X_OFFSET 0
#define EXT2_Y_OFFSET 0
#define EXT2_Z_OFFSET 0
#define EXT2_STEPS_PER_MM 96
#define EXT2_TEMPSENSOR_TYPE 1
#define EXT2_TEMPSENSOR_PIN TEMP_0_PIN
#define EXT2_HEATER_PIN 66
#define EXT2_STEP_PIN ORIG_E2_STEP_PIN
#define EXT2_DIR_PIN ORIG_E2_DIR_PIN
#define EXT2_INVERSE 0
#define EXT2_ENABLE_PIN
ORIG_E2_ENABLE_PIN
#define EXT2_ENABLE_ON 0
#define EXT2_MAX_FEEDRATE 50
#define EXT2_MAX_START_FEEDRATE 0
#define EXT2_MAX_ACCELERATION 5000
#define EXT2_HEAT_MANAGER 0
#define EXT2_WATCHPERIOD 1
#define EXT2_PID_INTEGRAL_DRIVE_MAX 230
#define EXT2_PID_INTEGRAL_DRIVE_MIN 40
#define EXT2_PID_PGAIN_OR_DEAD_TIME 7
#define EXT2_PID_I 2
#define EXT2_PID_D 40
#define EXT2_PID_MAX 255
#define EXT2_ADVANCE_K 0
#define EXT2_ADVANCE_L 0

```

```

#define EXT2_ADVANCE_BACKLASH_STEPS 0
#define EXT2_WAIT_RETRACT_TEMP 150
#define EXT2_WAIT_RETRACT_UNITS 0
#define EXT2_SELECT_COMMANDS ""
#define EXT2_DESELECT_COMMANDS ""
#define EXT2_EXTRUDER_COOLER_PIN -1
#define EXT2_EXTRUDER_COOLER_SPEED 255
#define EXT2_DECOUPLE_TEST_PERIOD
1200000
#define EXT2_JAM_PIN -1
#define EXT2_JAM_PULLUP 0

#define EXT3_X_OFFSET 0
#define EXT3_Y_OFFSET 0
#define EXT3_Z_OFFSET 0
#define EXT3_STEPS_PER_MM 96
#define EXT3_TEMPSENSOR_TYPE 1
#define EXT3_TEMPSENSOR_PIN TEMP_0_PIN
#define EXT3_HEATER_PIN 65
#define EXT3_STEP_PIN ORIG_E3_STEP_PIN
#define EXT3_DIR_PIN ORIG_E3_DIR_PIN
#define EXT3_INVERSE 0
#define EXT3_ENABLE_PIN
ORIG_E3_ENABLE_PIN
#define EXT3_ENABLE_ON 0
#define EXT3_MAX_FEEDRATE 50
#define EXT3_MAX_START_FEEDRATE 20
#define EXT3_MAX_ACCELERATION 5000
#define EXT3_HEAT_MANAGER 0
#define EXT3_WATCHPERIOD 1
#define EXT3_PID_INTEGRAL_DRIVE_MAX 230
#define EXT3_PID_INTEGRAL_DRIVE_MIN 40
#define EXT3_PID_PGAIN_OR_DEAD_TIME 7
#define EXT3_PID_I 2
#define EXT3_PID_D 40
#define EXT3_PID_MAX 255
#define EXT3_ADVANCE_K 0

```

```

#define EXT3_ADVANCE_L 0
#define EXT3_ADVANCE_BACKLASH_STEPS 0
#define EXT3_WAIT_RETRACT_TEMP 150
#define EXT3_WAIT_RETRACT_UNITS 0
#define EXT3_SELECT_COMMANDS ""
#define EXT3_DESELECT_COMMANDS ""
#define EXT3_EXTRUDER_COOLER_PIN -1
#define EXT3_EXTRUDER_COOLER_SPEED 255
#define EXT3_DECOUPLE_TEST_PERIOD
1200000
#define EXT3_JAM_PIN -1
#define EXT3_JAM_PULLUP 0

#define EXT4_X_OFFSET 0
#define EXT4_Y_OFFSET 0
#define EXT4_Z_OFFSET 0
#define EXT4_STEPS_PER_MM 96
#define EXT4_TEMPSENSOR_TYPE 1
#define EXT4_TEMPSENSOR_PIN TEMP_0_PIN
#define EXT4_HEATER_PIN 0
#define EXT4_STEP_PIN ORIG_E4_STEP_PIN
#define EXT4_DIR_PIN ORIG_E4_DIR_PIN
#define EXT4_INVERSE 0
#define EXT4_ENABLE_PIN
ORIG_E4_ENABLE_PIN
#define EXT4_ENABLE_ON 0
#define EXT4_MAX_FEEDRATE 50
#define EXT4_MAX_START_FEEDRATE 20
#define EXT4_MAX_ACCELERATION 5000
#define EXT4_HEAT_MANAGER 0
#define EXT4_WATCHPERIOD 1
#define EXT4_PID_INTEGRAL_DRIVE_MAX 230
#define EXT4_PID_INTEGRAL_DRIVE_MIN 40
#define EXT4_PID_PGAIN_OR_DEAD_TIME 7
#define EXT4_PID_I 2
#define EXT4_PID_D 40
#define EXT4_PID_MAX 255

```

```

#define EXT4_ADVANCE_K 0
#define EXT4_ADVANCE_L 0
#define EXT4_ADVANCE_BACKLASH_STEPS 0
#define EXT4_WAIT_RETRACT_TEMP 150
#define EXT4_WAIT_RETRACT_UNITS 0
#define EXT4_SELECT_COMMANDS ""
#define EXT4_DESELECT_COMMANDS ""
#define EXT4_EXTRUDER_COOLER_PIN -1
#define EXT4_EXTRUDER_COOLER_SPEED 255
#define EXT4_DECOUPLE_TEST_PERIOD 120000
#define EXT4_JAM_PIN -1
#define EXT4_JAM_PULLUP 0

#define FEATURE_RETRACTION 1
#define AUTORETRACT_ENABLED 0
#define RETRACTION_LENGTH 5
#define RETRACTION_LONG_LENGTH 5
#define RETRACTION_SPEED 50
#define RETRACTION_Z_LIFT 0.3
#define RETRACTION_UNDO_EXTRA_LENGTH 0
#define RETRACTION_UNDO_EXTRA_LONG_LENGTH 1
#define RETRACTION_UNDO_SPEED 20
#define FILAMENTCHANGE_X_POS 0
#define FILAMENTCHANGE_Y_POS 0
#define FILAMENTCHANGE_Z_ADD 2
#define FILAMENTCHANGE_REHOME 1
#define FILAMENTCHANGE_SHORTRETRACT 5
#define FILAMENTCHANGE_LONGRETRACT 50
#define JAM_STEPS 220
#define JAM_SLOWDOWN_STEPS 320
#define JAM_SLOWDOWN_TO 70
#define JAM_ERROR_STEPS 500
#define JAM_MIN_STEPS 10
#define JAM_ACTION 1

```

```

#define RETRACT_DURING_HEATUP true
#define PID_CONTROL_RANGE 20
#define SKIP_M109_IF_WITHIN 2
#define SCALE_PID_TO_MAX 0
#define TEMP_HYSTERESIS 0
#define EXTRUDE_MAXLENGTH 160
#define NUM_TEMPS_USERTHERMISTOR0 0
#define USER_THERMISTORTABLE0 {}
#define NUM_TEMPS_USERTHERMISTOR1 0
#define USER_THERMISTORTABLE1 {}
#define NUM_TEMPS_USERTHERMISTOR2 0
#define USER_THERMISTORTABLE2 {}
#define GENERIC_THERM_VREF 5
#define GENERIC_THERM_NUM_ENTRIES 33
#define HEATER_PWM_SPEED 2

// ##### Heated bed configuration
#####

#define HAVE_HEATED_BED 1
#define HEATED_BED_MAX_TEMP 300
#define SKIP_M190_IF_WITHIN 3
#define HEATED_BED_SENSOR_TYPE 1
#define HEATED_BED_SENSOR_PIN TEMP_1_PIN
#define HEATED_BED_HEATER_PIN HEATER_1_PIN
#define HEATED_BED_SET_INTERVAL 5000
#define HEATED_BED_HEAT_MANAGER 0
#define HEATED_BED_PID_INTEGRAL_DRIVE_MAX 255
#define HEATED_BED_PID_INTEGRAL_DRIVE_MIN 80
#define HEATED_BED_PID_PGAIN_OR_DEAD_TIME 196
#define HEATED_BED_PID_IGAIN 33

```

```

#define HEATED_BED_PID_DGAIN 290
#define HEATED_BED_PID_MAX 255
#define
HEATED_BED_DECOUPLE_TEST_PERIOD
3000000
#define MIN_EXTRUDER_TEMP 0
#define MAXTEMP 275
#define MIN_DEFECT_TEMPERATURE -10
#define MAX_DEFECT_TEMPERATURE 300

// ##### Endstop configuration
#####

#define ENDSTOP_PULLUP_X_MIN true
#define ENDSTOP_X_MIN_INVERTING true
#define MIN_HARDWARE_ENDSTOP_X true
#define ENDSTOP_PULLUP_Y_MIN true
#define ENDSTOP_Y_MIN_INVERTING true
#define MIN_HARDWARE_ENDSTOP_Y true
#define ENDSTOP_PULLUP_Z_MIN true
#define ENDSTOP_Z_MIN_INVERTING true
#define MIN_HARDWARE_ENDSTOP_Z true
#define ENDSTOP_PULLUP_X_MAX true
#define ENDSTOP_X_MAX_INVERTING false
#define MAX_HARDWARE_ENDSTOP_X false
#define ENDSTOP_PULLUP_Y_MAX true
#define ENDSTOP_Y_MAX_INVERTING false
#define MAX_HARDWARE_ENDSTOP_Y false
#define ENDSTOP_PULLUP_Z_MAX true
#define ENDSTOP_Z_MAX_INVERTING false
#define MAX_HARDWARE_ENDSTOP_Z false
#define max_software_endstop_r true

#define min_software_endstop_x false
#define min_software_endstop_y false
#define min_software_endstop_z false
#define max_software_endstop_x true

```

```

#define max_software_endstop_y true
#define max_software_endstop_z true
#define ENDSTOP_X_BACK_MOVE 5
#define ENDSTOP_Y_BACK_MOVE 5
#define ENDSTOP_Z_BACK_MOVE 10
#define
ENDSTOP_X_RETEST_REDUCTION_FACTOR 3
#define
ENDSTOP_Y_RETEST_REDUCTION_FACTOR 3
#define
ENDSTOP_Z_RETEST_REDUCTION_FACTOR 5
#define ENDSTOP_X_BACK_ON_HOME 1
#define ENDSTOP_Y_BACK_ON_HOME 1
#define ENDSTOP_Z_BACK_ON_HOME 0
#define ALWAYS_CHECK_ENDSTOPS 1

// ##### XYZ movements
#####

#define X_ENABLE_ON 0
#define Y_ENABLE_ON 0
#define Z_ENABLE_ON 0
#define DISABLE_X 0
#define DISABLE_Y 0
#define DISABLE_Z 0
#define DISABLE_E 0
#define INVERT_X_DIR 1
#define INVERT_Y_DIR 1
#define INVERT_Z_DIR 1
#define X_HOME_DIR -1
#define Y_HOME_DIR -1
#define Z_HOME_DIR -1
#define X_MAX_LENGTH 210
#define Y_MAX_LENGTH 250
#define Z_MAX_LENGTH 300
#define X_MIN_POS 0
#define Y_MIN_POS 0
#define Z_MIN_POS 0

```

```

#define DISTORTION_CORRECTION 0//1
#define DISTORTION_CORRECTION_POINTS 0
//5
#define DISTORTION_CORRECTION_R 100
#define DISTORTION_PERMANENT 0 //1
#define DISTORTION_UPDATE_FREQUENCY 15
#define DISTORTION_START_DEGRADE 0.5
#define DISTORTION_END_HEIGHT 1
#define
DISTORTION_EXTRAPOLATE_CORNERS 0
#define DISTORTION_XMIN 10
#define DISTORTION_YMIN 10
#define DISTORTION_XMAX 190
#define DISTORTION_YMAX 190

#define FEATURE_BABYSTEPPING 1
#define BABYSTEP_MULTIPLICATOR 1

#define
DELTA_SEGMENTS_PER_SECOND_PRINT 180
// Move accurate setting for print moves
#define
DELTA_SEGMENTS_PER_SECOND_MOVE 70 //
Less accurate setting for other moves
#define EXACT_DELTA_MOVES 1

// Delta settings
#define DELTA_HOME_ON_POWER 0

#define DELTASEGMENTS_PER_PRINTLINE 24
#define STEPPER_INACTIVE_TIME 360L
#define MAX_INACTIVE_TIME 0L
#define MAX_FEEDRATE_X 200
#define MAX_FEEDRATE_Y 200
#define MAX_FEEDRATE_Z 2
#define HOMING_FEEDRATE_X 40
#define HOMING_FEEDRATE_Y 40
#define HOMING_FEEDRATE_Z 40 //20

```

```

#define HOMING_ORDER HOME_ORDER_YXZ
#define ZHOME_MIN_TEMPERATURE 0
#define ZHOME_HEAT_ALL 0 // 1
#define ZHOME_HEAT_HEIGHT 0
#define ZHOME_X_POS 0 //999999
#define ZHOME_Y_POS 0 //999999
#define ENABLE_BACKLASH_COMPENSATION
0
#define X_BACKLASH 0
#define Y_BACKLASH 0
#define Z_BACKLASH 0
#define RAMP_ACCELERATION 1
#define STEPPER_HIGH_DELAY 0
#define DIRECTION_DELAY 0
#define STEP_DOUBLER_FREQUENCY 12000
#define ALLOW_QUADSTEPPING 1
#define DOUBLE_STEP_DELAY 0 // time in
microseconds
#define
MAX_ACCELERATION_UNITS_PER_SQ_SECO
ND_X 2000
#define
MAX_ACCELERATION_UNITS_PER_SQ_SECO
ND_Y 2000
#define
MAX_ACCELERATION_UNITS_PER_SQ_SECO
ND_Z 10
#define
MAX_TRAVEL_ACCELERATION_UNITS_PER_
SQ_SECOND_X 2000
#define
MAX_TRAVEL_ACCELERATION_UNITS_PER_
SQ_SECOND_Y 2000
#define
MAX_TRAVEL_ACCELERATION_UNITS_PER_
SQ_SECOND_Z 150
#define
INTERPOLATE_ACCELERATION_WITH_Z 0
#define ACCELERATION_FACTOR_TOP 100
#define MAX_JERK 20
#define MAX_ZJERK 0// 0.3

```

```

#define PRINTLINE_CACHE_SIZE 16
#define MOVE_CACHE_LOW 10
#define LOW_TICKS_PER_MOVE 250000
#define EXTRUDER_SWITCH_XY_SPEED 100
#define DUAL_X_AXIS 0
#define FEATURE_TWO_XSTEPPER 0
#define X2_STEP_PIN ORIG_E1_STEP_PIN
#define X2_DIR_PIN ORIG_E1_DIR_PIN
#define X2_ENABLE_PIN
ORIG_E1_ENABLE_PIN
#define FEATURE_TWO_YSTEPPER 0
#define Y2_STEP_PIN ORIG_E1_STEP_PIN
#define Y2_DIR_PIN ORIG_E1_DIR_PIN
#define Y2_ENABLE_PIN
ORIG_E1_ENABLE_PIN
#define FEATURE_TWO_ZSTEPPER 0
#define Z2_STEP_PIN ORIG_E1_STEP_PIN
#define Z2_DIR_PIN ORIG_E1_DIR_PIN
#define Z2_ENABLE_PIN
ORIG_E1_ENABLE_PIN
#define FEATURE_THREE_ZSTEPPER 0
#define Z3_STEP_PIN ORIG_E2_STEP_PIN
#define Z3_DIR_PIN ORIG_E2_DIR_PIN
#define Z3_ENABLE_PIN
ORIG_E2_ENABLE_PIN
#define FEATURE_DITTO_PRINTING 0
#define USE_ADVANCE 0
#define ENABLE_QUADRATIC_ADVANCE 0

// ##### Misc. settings
#####

#define BAUDRATE 115200
#define ENABLE_POWER_ON_STARTUP 1
#define POWER_INVERTING 0
#define KILL_METHOD 1
#define ACK_WITH_LINENUMBER 0

```

```

#define ECHO_ON_EXECUTE 1
#define EEPROM_MODE 3
#undef PS_ON_PIN
#define PS_ON_PIN -1
#define JSON_OUTPUT 0

/* ===== Servos =====
#define FEATURE_SERVO 0
#define SERVO0_PIN 11
#define SERVO1_PIN -1
#define SERVO2_PIN -1
#define SERVO3_PIN -1
#define SERVO0_NEUTRAL_POS -1
#define SERVO1_NEUTRAL_POS -1
#define SERVO2_NEUTRAL_POS -1
#define SERVO3_NEUTRAL_POS -1
#define UI_SERVO_CONTROL 0
#define FAN_KICKSTART_TIME 200
#define FEATURE_WATCHDOG 0

// ##### Z-Probing
#####

#define Z_PROBE_Z_OFFSET 0
#define Z_PROBE_Z_OFFSET_MODE 0
#define UI_BED_COATING 1
#define FEATURE_Z_PROBE 1 // 1
#define Z_PROBE_BED_DISTANCE 4
#define Z_PROBE_PIN ORIG_Z_MIN_PIN
#define Z_PROBE_PULLUP 1
#define Z_PROBE_ON_HIGH 0
#define Z_PROBE_X_OFFSET 0
#define Z_PROBE_Y_OFFSET 0
#define Z_PROBE_WAIT_BEFORE_TEST 0
#define Z_PROBE_SPEED 2
#define Z_PROBE_XY_SPEED 100
#define Z_PROBE_SWITCHING_DISTANCE 0.2

```

```

#define Z_PROBE_REPETITIONS 1
#define Z_PROBE_HEIGHT 0
#define Z_PROBE_START_SCRIPT "Auto
Nivelando... Aguarde..."
#define Z_PROBE_FINISHED_SCRIPT
"Nivelamento Concluido"
#define Z_PROBE_REQUIRES_HEATING 0
#define Z_PROBE_MIN_TEMPERATURE 0
#define FEATURE_AUTOLEVEL 1 // 1
#define Z_PROBE_X1 10
#define Z_PROBE_Y1 10
#define Z_PROBE_X2 210
#define Z_PROBE_Y2 10
#define Z_PROBE_X3 10
#define Z_PROBE_Y3 270
#define BED_LEVELING_METHOD 1
#define BED_CORRECTION_METHOD 0
#define BED_LEVELING_GRID_SIZE 5
#define BED_LEVELING_REPETITIONS 5
#define BED_MOTOR_1_X 0
#define BED_MOTOR_1_Y 0
#define BED_MOTOR_2_X 200
#define BED_MOTOR_2_Y 0
#define BED_MOTOR_3_X 100
#define BED_MOTOR_3_Y 200
#define BENDING_CORRECTION_A 0
#define BENDING_CORRECTION_B 0
#define BENDING_CORRECTION_C 0
#define FEATURE_AXISCOMP 1
#define AXISCOMP_TANXY 0
#define AXISCOMP_TANYZ 0
#define AXISCOMP_TANXZ 0

#ifndef SDSUPPORT
#define SDSUPPORT 1
#undef SDCARDDetect
#define SDCARDDetect -1

```

```

#define SDCARDDetectINVERTED 0
#endif
#define SD_EXTENDED_DIR 1 /** Show extended
directory including file length. Don't use this with
Pronterface! */
#define SD_RUN_ON_STOP ""
#define
SD_STOP_HEATER_AND_MOTORS_ON_STOP
1
#define ARC_SUPPORT 1
#define FEATURE_MEMORY_POSITION 1
#define FEATURE_CHECKSUM_FORCED 0
#define FEATURE_FAN_CONTROL 1
#define FEATURE_FAN2_CONTROL 0
#define FEATURE_CONTROLLER 2
#define ADC_KEYPAD_PIN -1
#define LANGUAGE_EN_ACTIVE 0
#define LANGUAGE_DE_ACTIVE 0
#define LANGUAGE_NL_ACTIVE 0
#define LANGUAGE_PT_ACTIVE 1
#define LANGUAGE_IT_ACTIVE 0
#define LANGUAGE_ES_ACTIVE 0
#define LANGUAGE_FI_ACTIVE 0
#define LANGUAGE_SE_ACTIVE 0
#define LANGUAGE_FR_ACTIVE 0
#define LANGUAGE_CZ_ACTIVE 0
#define LANGUAGE_PL_ACTIVE 0
#define LANGUAGE_TR_ACTIVE 0
#define UI_PRINTER_NAME "JPRINT 3.0"
#define UI_PRINTER_COMPANY "JP"
#define UI_PAGES_DURATION 4000
#define UI_ANIMATION 0
#define UI_SPEEDDEPENDENT_POSITIONING 0
#define UI_DISABLE_AUTO_PAGESWITCH 1
#define UI_AUTORETURN_TO_MENU_AFTER
30000
#define FEATURE_UI_KEYS 0
#define UI_ENCODER_SPEED 1

```

```

#define UI_REVERSE_ENCODER 0
#define UI_KEY_BOUNCETIME 10
#define UI_KEY_FIRST_REPEAT 500
#define UI_KEY_REDUCE_REPEAT 50
#define UI_KEY_MIN_REPEAT 50
#define FEATURE_BEEPER 0
#define CASE_LIGHTS_PIN -1
#define CASE_LIGHT_DEFAULT_ON 1
#define UI_START_SCREEN_DELAY 1000
#define UI_DYNAMIC_ENCODER_SPEED 1

#define BEEPER_SHORT_SEQUENCE 2,2
#define BEEPER_LONG_SEQUENCE 8,8

#define
UI_SET_PRESET_HEATED_BED_TEMP_PLA 60

#define
UI_SET_PRESET_EXTRUDER_TEMP_PLA 200

#define
UI_SET_PRESET_HEATED_BED_TEMP_ABS
110

#define
UI_SET_PRESET_EXTRUDER_TEMP_ABS 240

#define UI_SET_MIN_HEATED_BED_TEMP 30

```

```

#define UI_SET_MAX_HEATED_BED_TEMP 120
#define UI_SET_MIN_EXTRUDER_TEMP 170
#define UI_SET_MAX_EXTRUDER_TEMP 250
#define UI_SET_EXTRUDER_FEEDRATE 1

#define
UI_SET_EXTRUDER_RETRACT_DISTANCE 5

#define NUM_MOTOR_DRIVERS 0

#endif

```

ANEXO B – Código de Parâmetros para o Firmware do Scanner 3D

```

// define PINS

//Table Stepper
#define TableStepperEnablePinPin    8
#define TableStepperStepPinPin     2
#define TableStepperDirectionPin5

//Laser Stepper
#define LaserStepperEnablePin    8
#define LaserStepperStepPin     3
#define LaserStepperDirectionPin 6

//Unused third stepper
#define ThirdStepperEnablePin  11
#define ThirdStepperStepPin    12
#define ThirdStepperDirectionPin  13

// unused fourth stepper
#define FourthStepperEnablePin    A0
#define FourthStepperStepPin      A1
#define FourthStepperDirectionPin A2

// others
#define LightPin          A3
#define LaserPin          4
#define MicroSteppingPin  A5

//define command
#define DisableLaser      100
#define EnableLaser       101

#define RotateLaserCw      110
#define RotateLaserCcw     111

#define RotateTurnTableCw  120
#define RotateTurnTableCcw 121

#define EnableLight        130
#define DisableLight       131

#define Ping                201
#define ACK                 202
#define NACK                203
#define Error               204

#define EnableMicroStepping 141
#define DisableMicroStepping 142

//define constants
#define TableStepper      1
#define LaserStepper      2

#define ClockWise         1
#define CounterClockWise 2

void setup()
{
    Serial.begin(9600);
    pinMode(LaserPin, OUTPUT);
    pinMode(LightPin, OUTPUT);
    pinMode(MicroSteppingPin, OUTPUT);
    digitalWrite(MicroSteppingPin, HIGH);
    pinMode(TableStepperEnablePinPin, OUTPUT);
    pinMode(TableStepperDirectionPin, OUTPUT);
    pinMode(TableStepperStepPinPin, OUTPUT);
    pinMode(LaserStepperEnablePin, OUTPUT);
    pinMode(LaserStepperDirectionPin, OUTPUT);
    pinMode(LaserStepperStepPin, OUTPUT);
    pinMode(ThirdStepperEnablePin, OUTPUT);
    pinMode(ThirdStepperDirectionPin, OUTPUT);
    pinMode(ThirdStepperStepPin, OUTPUT);
    pinMode(FourthStepperEnablePin, OUTPUT);
    pinMode(FourthStepperDirectionPin, OUTPUT);

```

```

pinMode(FourthStepperStepPin, OUTPUT);

disableStepper(TableStepperEnablePinPin);
disableStepper(LaserStepperEnablePin);
disableStepper(ThirdStepperEnablePin);
disableStepper(FourthStepperEnablePin);

disableLight();
disableLaser();
Serial.write(ACK);
}

void loop()
{
if(Serial.available() > 0)
{

int incomingByte = Serial.read();
int numberOfSteps = 0;
int result = ACK;

    if(incomingByte == RotateLaserCw ||
incomingByte == RotateLaserCcw || incomingByte ==
== RotateTurnTableCw || incomingByte ==
RotateTurnTableCcw || incomingByte ==
EnableLight)

        delay(3); //otherwise second byte is read
to fast = 0

switch(incomingByte)
{
case DisableLaser:
disableLaser();
break;
case EnableLaser:
enableLaser();
break;
case RotateLaserCw:

```

```

numberOfSteps = Serial.read();
performSteps(LaserStepper, ClockWise,
numberOfSteps);
break;
case RotateLaserCcw:
numberOfSteps = Serial.read();
performSteps(LaserStepper, CounterClockWise,
numberOfSteps);
break;
case RotateTurnTableCw:
numberOfSteps = Serial.read();
performSteps(TableStepper, ClockWise,
numberOfSteps);
break;
case RotateTurnTableCcw:
numberOfSteps = Serial.read();
performSteps(TableStepper, CounterClockWise,
numberOfSteps);
break;
case EnableLight:
incomingByte = Serial.read();
enableLight(incomingByte);
break;
case DisableLight:
disableLight();
break;
case EnableMicroStepping:
enableMicroStepping();
break;
case DisableMicroStepping:
disableMicroStepping();
break;
case Ping:
delay(1);
break;
default:
result = NACK;
break;

```

```
}
```

```
Serial.write(result);
```

```
}
```

```
}
```

```
void performSteps(int stepper, int dir, int  
numberOfSteps)
```

```
{
```

```
int stepperEnablePin = 0;
```

```
int stepperDirectionPin = 0;
```

```
int stepperStepPin = 0;
```

```
switch(stepper)
```

```
{
```

```
case TableStepper:
```

```
stepperEnablePin = TableStepperEnablePinPin;
```

```
stepperDirectionPin = TableStepperDirectionPin;
```

```
stepperStepPin = TableStepperStepPinPin;
```

```
break;
```

```
case LaserStepper:
```

```
stepperEnablePin = LaserStepperEnablePin;
```

```
stepperDirectionPin = LaserStepperDirectionPin;
```

```
stepperStepPin = LaserStepperStepPin;
```

```
break;
```

```
}
```

```
enableStepper(stepperEnablePin);
```

```
setStepperDirectionPin(stepperDirectionPin, dir);
```

```
for(int i=0; i<numberOfSteps; i++)
```

```
{
```

```
performStep(stepperStepPin);
```

```
}
```

```
}
```

```
disableStepper(stepperEnablePin);
```

```
}
```

```
void performStep(int stepperPin)
```

```
{
```

```
//turn stepper on, sleep, turn stepper off
```

```
digitalWrite(stepperPin, LOW);
```

```
delay(3);
```

```
digitalWrite(stepperPin, HIGH);
```

```
delay(3);
```

```
}
```

```
void setStepperDirectionPin(int stepperPin, int dir)
```

```
{
```

```
switch(dir)
```

```
{
```

```
case ClockWise:
```

```
digitalWrite(stepperPin, HIGH);
```

```
break;
```

```
case CounterClockWise:
```

```
digitalWrite(stepperPin, LOW);
```

```
break;
```

```
}
```

```
}
```

```
void enableStepper(int stepperPin)
```

```
{
```

```
digitalWrite(stepperPin, LOW);
```

```
}
```

```
void disableStepper(int stepperPin)
```

```
{
```

```
digitalWrite(stepperPin, HIGH);
```

```
}
```

```
void disableLaser()
```

```
{  
digitalWrite(LaserPin, LOW);  
}  
  
void enableLaser()  
{  
digitalWrite(LaserPin, HIGH);  
}  
  
void disableLight()  
{  
digitalWrite(LightPin, LOW);  
}  
  
void enableLight(int hue)  
{  
analogWrite(LightPin, hue);  
}  
  
void enableMicroStepping()  
{  
digitalWrite(MicroSteppingPin, HIGH);  
}  
  
void disableMicroStepping()  
{  
digitalWrite(MicroSteppingPin, LOW);  
} {  
    digitalWrite(MicroSteppingPin, LOW);  
}
```