

**FUNDAÇÃO OSWALDO ARANHA
CENTRO UNIVERSITÁRIO DE VOLTA REDONDA
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE MESTRADO PROFISSIONAL EM MATERIAIS**

ALOANO REGIO DE ALMEIDA PEREIRA

**DESENVOLVIMENTO DE UMA IMPRESSORA 3D PARA FINS
ACADÊMICOS E ANÁLISE DE SUA INFLUÊNCIA NAS PROPRIEDADES
DO PLA**

VOLTA REDONDA, RJ

2024

**FUNDAÇÃO OSWALDO ARANHA
CENTRO UNIVERSITÁRIO DE VOLTA REDONDA
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE MESTRADO PROFISSIONAL EM MATERIAIS**

**DESENVOLVIMENTO DE UMA IMPRESSORA 3D PARA FINS
ACADÊMICOS E ANÁLISE DE SUA INFLUÊNCIA NAS PROPRIEDADES
DO PLA**

Dissertação apresentada ao Mestrado Profissional em Materiais do Centro Universitário de Volta Redonda – UniFOA, como requisito obrigatório para obtenção do título de Mestre em Materiais, sob a orientação do Prof. Dr. Alexandre Alvarenga Palmeira, na área de concentração de processamento e caracterização de materiais, linha de pesquisa de materiais poliméricos.

Aluno: Aloano Regio de Almeida Pereira

Orientador: Professor Dr. Alexandre Alvarenga
Palmeira

VOLTA REDONDA, RJ

2024

FICHA CATALOGRÁFICA

Bibliotecária: Alice Tacão Wagner - CRB 7/RJ 4316

P436d Pereira, Aloano Régio de Almeida

Desenvolvimento de impressora 3D de baixo custo para fins acadêmicos e análise de sua influência nas propriedades do PLA. / Aloano Régio de Almeida Pereira. - Volta Redonda: UniFOA, 2024. 186 p.: Il

Orientador (a): Prof. Dr. Alexandre Alvarenga Palmeira

Dissertação (Mestrado) – UniFOA / Mestrado Profissional em Materiais, 2024

1. Materiais - dissertação. 2. Impressora 3D. 3. Prototipagem. 4. Indústria 4.0. I. Palmeira, Alexandre Alvarenga. II. Centro Universitário de Volta Redonda. III. Título.

CDD – 620.1



FUNDAÇÃO OSWALDO ARANHA
CENTRO UNIVERSITÁRIO DE VOLTA REDONDA
PRO-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
MESTRADO PROFISSIONAL EM MATERIAIS



ALOANO RÉGIO DE ALMEIDA PEREIRA

DESENVOLVIMENTO DE IMPRESSORA 3D DE BAIXO CUSTO PARA FINS
ACADÊMICOS E ANÁLISE DE SUA INFLUÊNCIA NAS PROPRIEDADES DO
PLA

Orientador: Prof. Dr. Alexandre Alvarenga Palmeira

ESTA DISSERTAÇÃO FOI JULGADA ADEQUADA PARA OBTENÇÃO DO
TÍTULO DE
"MESTRE EM MATERIAIS"

APROVADO EM SUA FORMA FINAL PELO PROGRAMA DE MESTRADO
PROFISSIONAL EM MATERIAIS

Roberto de Oliveira Magnago

Prof. Dr. Roberto de Oliveira Magnago
Coordenador

BANCA EXAMINADORA:

Alexandre Alvarenga Palmeira

Prof. Dr. Alexandre Alvarenga Palmeira
Presidente / UniFOA

Roberto de Oliveira Magnago

Prof. Dr. Roberto de Oliveira Magnago
Examinador interno / UniFOA

Zoroastro Torres Vilar

Prof. Dr. Zoroastro Torres Vilar

Examinador externo / UFERSA

Abril de 2024.

AGRADECIMENTOS

Nesta dissertação de mestrado, é crucial salientar o apoio fundamental de diversas pessoas. Em primeiro lugar, expresso minha gratidão a Deus pela vida e pela capacidade de superar os obstáculos encontrados ao longo deste trabalho. Em segundo lugar, agradeço imensamente à minha esposa e filhos pelo apoio incondicional, pelo incentivo nos momentos difíceis e pela compreensão em relação à minha ausência enquanto eu me dedicava a este projeto. Em terceiro lugar, desejo expressar a minha sincera gratidão ao meu orientador, Professor Dr Alexandre Alvarenga Palmeira, pela paciência, empenho e orientação prática oferecida ao longo deste trabalho. Agradeço também ao Professor Dr. Roberto de Oliveira Magnago, coordenador do curso de mestrado, pelo constante incentivo e apoio. Sou grato aos Professores Me Edson Carvalho e Dra. Izabel de Oliveira Motta por seu estímulo ao longo deste trabalho de mestrado. Por fim, destaco a importância do CENTRO UNIVERSITÁRIO DE VOLTA REDONDA em minha formação profissional, agradecendo pela dedicação e pelo conhecimento adquirido ao longo dos anos de estudo.

RESUMO

O objetivo desse trabalho desenvolver uma impressora de baixo custo para fins acadêmico, inspirada nos conceitos de software livre (*Open Source*), RepRap (*Replicating Rapid-prototyper*) e IOT e efetuar uma previa análise de sua influência nas propriedades do PLA. A motivação desse trabalho visa difundir conceitos tanto como da parte mecânica, eletrônica, software e calibração e ensaios dos materiais que venha auxiliar pessoas que pretendem efetuar a construção e entender do processo de manufatura aditiva, possibilitando também aos alunos inserção ao conceito indústria 4.0. Vários conceitos são necessários e importantes nesse processo para conseguir uma qualidade de impressão como a montagem correta da estrutura mecânica, da instalação da estrutura eletrônica como sensores, atuadores, do firmware do microcontrolador (Arduino, Raspberry Pi), da calibração dos passos dos motores e da extrusão e do monitoramento remoto e como verificar as propriedades dos materiais impressos. Inicialmente foi feito um estudo bibliográfico do processo de manufatura aditiva, onde uma vez esse conhecimento melhor sedimentado iniciou-se o processo de montagem da impressora 3D. Para executar esse projeto foi seguida as seguintes etapas: 1 – Pesquisa sobre manufatura aditiva, empregando a impressão 3D; 2 – Funcionamento da impressão 3D; 3 - Desenvolvimento, construção da impressora 3D; 4 - Comissionamento e calibração da impressora; 5 – Instalação do Raspberry Pi, testes e validação; 6 – Ensaio das propriedades dos materiais impressos para validação da impressora, bem como uma prévia análise nas influencias nos materiais impressos, no caso PLA. O equipamento irá contribuir muito aos cursos acadêmicos, como áreas da mecânica, elétrica, eletrônica, informática e materiais, assim como para os profissionais e/ou pessoas que buscam conhecimentos, tanto fins didáticos, quanto para fins industriais. Pois além de mostrar o conceito e funcionamento de uma impressora 3D é possível identificar as principais divergências entre a teoria e a prática e com isso explorar o estado da arte relativamente da prototipagem 3D, assim como identificar de que forma uma Impressora 3D com recursos IoT influenciará o mundo em que vivemos e as indústrias.

Palavras-chave: Impressora 3D; Prototipagem; Industria 4.0; Arduino; Raspberry

ABSTRACT

The objective of this work is to develop a low-cost printer for academic purposes, inspired by the concepts of free software (Open Source), RepRap (Replicating Rapid-prototyper) and IOT and to carry out a preliminary analysis of their influence on the properties of PLA. The motivation for this work aims to disseminate concepts such as mechanics, electronics, software and calibration and testing of materials that will help people who intend to build and understand the additive manufacturing process, also enabling students to introduce the industry 4.0 concept. Several concepts are necessary and important in this process to achieve print quality, such as the correct assembly of the mechanical structure, the installation of the electronic structure such as sensors, actuators, the microcontroller firmware (Arduino, Raspberry Pi), the calibration of the motor steps and extrusion and remote monitoring and how to check the properties of printed materials. Initially, a bibliographical study of the additive manufacturing process was carried out, where once this knowledge was better consolidated, the process of assembling the 3D printer began. To execute this project, the following steps were followed: 1 – Research on additive manufacturing, using 3D printing; 2 – How 3D printing works; 3 - Development, construction of the 3D printer; 4 - Commissioning and calibration of the printer; 5 – Raspberry Pi installation, testing and validation; 6 – Tests of the properties of printed materials to validate the printer, as well as a prior analysis of the influences on printed materials, in this case PLA. The equipment will greatly contribute to academic courses, such as mechanics, electrical, electronics, IT and materials, as well as to professionals and/or people seeking knowledge, both for teaching and industrial purposes. In addition to showing the concept and functioning of a 3D printer, it is possible to identify the main divergences between theory and practice and thus explore the state of the art regarding 3D prototyping, as well as identify how a 3D Printer with IoT resources will influence the world we live in and the industries.

Keywords: 3D printer; Prototyping; Industry 4.0; Arduino; Raspberry.

SUMÁRIO

1	INTRODUÇÃO	14
1.1	OBJETIVO GERAL	17
1.2	OBJETIVOS ESPECÍFICO	17
2	REFERENCIAL TEORICO	18
2.1	INDUSTRIA 4.0	18
2.2	MANUFATURA ADITIVA NO MEIO ACADÊMICO: INOVAÇÃO E DESAFIOS	19
2.3	MICROCONTROLADORES	20
2.3.1	Microcontroladores na Impressão 3D	21
2.3.2	Microcontroladores e Placas de Controle em Impressoras 3D:	21
2.4	IMPRESSORAS 3D	22
2.4.1	Processo de prototipagem e engenharia reversa	28
2.4.2	Principais métodos de prototipagem 3D	28
2.4.2.1	Processo de Fusão e Deposição de Material (FDM)	28
2.4.2.2	Processo de Sinterização Seletiva a Laser (SLS)	29
2.4.2.3	Processo de Stereolitografia (SLA)	30
2.5	POLÍMEROS	32
2.5.1	Tipos de Polímeros	32
2.5.2	Comportamento Térmico dos Polímeros	33
2.5.3	Propriedades Mecânicas dos Polímeros	34
2.6	CONSTRUÇÃO DA IMPRESSORA	37
2.6.1	Microcontrolador Arduino	38
2.6.2	Placa Shield Ramps	39
2.6.3	Drive dos motores de passo	41
2.6.4	Motores de passo	43
2.6.5	Extrusor e hotend	47
2.6.5.1	Tipos de sistemas de extrusão para impressoras 3D	50
2.6.5.1.1	Extrusão tipo Bowden	50
2.6.5.1.2	Extrusão direta para impressora 3D	51
2.6.5.2	Bico do hotend	53
2.6.6	Sensor fim de curso – Endstop	54
2.6.7	Sensores de temperatura – Termistor	57
2.6.8	Mesa aquecida	60
3	METODOLOGIA	61
4	DESENVOLVIMENTO	63
4.1	ESTRUTURA MECÂNICA	64
4.2	ESTRUTURA ELETRÔNICA	66
4.2.1	Componente de controle e gerenciamento – estrutura eletrônica	66
4.3	INSTALAÇÃO E PARAMETRIZAÇÃO DO FIRMWARE	75
4.4	CALIBRAÇÃO E VALIDAÇÃO	92
4.5	INSTALAÇÃO DO ACESSO AO IoT – OCTOPRINT	99
5	RESULTADOS	110
5.1	ENSAIO DE TRAÇÃO	116
5.2	ENSAIO DE DUREZA	123

6	CONCLUSÃO	125
7	PROPOSTA PARA TRABALHOS FUTUROS	127
	REFERÊNCIAS.....	128
	ANEXOS	134
	ANEXO A – ESTRUTURA (FRAME) IMPRESSORA 3D MDF	134
	ANEXO B – LISTA DE PEÇAS	136
	ANEXO C – DADOS DO ARDUINO MEGA.....	138
	ANEXO D – ESPECIFICAÇÕES TÉCNICAS RAMPS.....	139
	ANEXO E – G-CODES E M-CODES.....	140
	ANEXO F – ESPECIFICAÇÕES TÉCNICAS DRV8825	153
	ANEXO G – DATASHEET MOTOR DE PASSO 17HS4401	154
	APÊNDICES	155
	APÊNDICE A – PROGRAMA MARLIN CONFIGURE.H	155

LISTA DE QUADROS

Quadro 1 - Propriedades do poliácido láctico (PLA ou ácido poliláctico)	36
Quadro 2 - Medidas de dureza Shore D.....	124

LISTA DE FIGURAS

Figura 1 - Tipos de impressoras 3D	24
Figura 2 - Kit para impressoras 3D.....	24
Figura 3 - Principais etapas da impressão 3D.....	25
Figura 4 - Fatiamento de um modelo	26
Figura 5 - Tecnologias de impressão 3D.....	28
Figura 6 - Esquemática geral do processo FDM	29
Figura 7 - Esquemática geral do processo SLS.	30
Figura 8 - Esquemática geral do processo SLA.	31
Figura 9 - Representação das principais etapas do processo de impressão camada por camada.	32
Figura 10 - Tipos de filamentos para impressão 3D.....	35
Figura 11 - Comparativo resistência térmicas entre ABS, PLA, PLA HT Impact	37
Figura 12 - Comparativo resistência ao impacto ABS, PLA, PLA HT Impact	37
Figura 13 - Placa Arduino.....	39
Figura 14 - Placa Ramps 1.6.....	41
Figura 15 - Driver DVR8825.....	43
Figura 16 - Motor de passo	45
Figura 17 - Tipos de bobinas motor de passo	46
Figura 18 - Micropassos de um motor de passo	46
Figura 19 - Extrusor vs Hotend.....	48
Figura 20 - Componentes da cabeça de impressão.....	49
Figura 21 - Sistema Bowden	51
Figura 22 - Sistema Direct Drive	52
Figura 23 - Vista cortada do bico.....	54
Figura 24 - Extrusora Direct Drive	54
Figura 25 - Sensor endstop mecânico.....	55
Figura 26 - Diagrama elétrico do sensor Endstop mecânico.....	56
Figura 27 - Coordenada Home sensor endstop	57
Figura 28 - Mesa Aquecida MK2B.....	61
Figura 29 - Frame MDF	64
Figura 30 - Carro X com a extrusora	65
Figura 31 - Eixos retificado, barras roscadas e rolamentos.	65
Figura 32 - Fluxo de dados arduino / Ramps	66
Figura 33 - Diagrama de ligações Ramps	67
Figura 34 - Ligação motores passo na placa Ramps	68
Figura 35 - Disposição motores de passo na estrutura.....	68
Figura 36 - Posição dos drives na placa Ramps	69
Figura 37 - Ligação dos endstops	69
Figura 38 - Ligação mesa /Ramps	70
Figura 39 - Ligação Hotend na placa Ramps	71
Figura 40 - Interligação Fonte.	72
Figura 41 - Rsense e trimpt de ajuste dos Drivers DVR8825 e A4988.....	73
Figura 42 - Medições e ponto de ajuste de tensão dos motores de passo	73
Figura 43 - Micro passos dirve DVR8825.....	74
Figura 44 - Exemplo de Pontos de seleção de micro passos.....	75

Figura 45 - Interface IDE do Arduino.....	76
Figura 46 - Firmware Marlin aberto no IDE	77
Figura 47 - Programação Porta USB e velocidade.....	79
Figura 48 - Variável Ramps.....	80
Figura 49 - Tipos de Ramps que o Marlin pode controlar	80
Figura 50 - Variável número de extrusores e diâmetro do filamento	81
Figura 51 - Variável configuração da fonte de alimentação	82
Figura 52 - Tipos de sensores de temperatura no Marlin	82
Figura 53 - Variáveis sensor de temperatura ativado.....	83
Figura 54 - Variáveis Mínimo e Máximo do sensor de temperatura	84
Figura 55- Variável proteção temperatura.....	85
Figura 56 - Variável tempo e histerese proteção de temperatura.....	85
Figura 57 - Variáveis posição mínima dos endstops	86
Figura 58 - Posicionamento ponto 0 ou ponto Home	87
Figura 59 - Variáveis sentido ponto zero, volume de operação dos endstops	88
Figura 60 - Variável tipos contato do endstop	88
Figura 61 - Variáveis seleção drive DVR8825.....	89
Figura 62 - Variável Resolução Steps/mm	90
Figura 63 - Calculadora passos por milímetros para correias	90
Figura 64 - Calculadora passos por milímetros fusos	91
Figura 65 - Variável seleção do tipo de idioma.....	92
Figura 66 - Tipo do display.....	92
Figura 67 - Software Prontface.....	93
Figura 68 - Resultado calibração eixo X.....	94
Figura 69 - Resultado calibração eixo Y.....	94
Figura 70 - Resultado calibração do eixo Z.....	95
Figura 71 - Resultado calibração extrusora.....	95
Figura 72 - Medição da temperatura no Hotend.....	96
Figura 73 - Medição da temperatura na mesa.....	96
Figura 74 - Modelos para calibração e validação	97
Figura 75 - Verificação parede do cubo (taxa de extrusão).....	98
Figura 76 - Verificação dimensional do cubo X, Y e Z.....	98
Figura 77 - Tela de instalação do S.O do Raspberry e Octopi	100
Figura 78 - Conexão Raspberry PI3 com arduino e rede internet	100
Figura 79 - Tela de configuração inicial do Octoprint	101
Figura 80 - Tela de configuração de senha no Octoprint	102
Figura 81 - Tela envio de informações de Bugs	103
Figura 82 - Tela de checagem de conexão	104
Figura 83 - Tela de configuração de instalação de plugin	104
Figura 84 - Tela do nome da impressora	105
Figura 85 - Tela de configuração do volume de impressão da impressora.	106
Figura 86 - Tela de configuração do bico	106
Figura 87 - Tela do octoprint - Aba temperatura.....	107
Figura 88 - Tela octoprint - Imagem remota da área de impressão.....	108
Figura 89 - Tela de acompanhamento das Layers do produto	108
Figura 90 - Tela de gravação da impressão.....	109
Figura 91 -Modelo Case para o display 128x64.....	110

Figura 92 - Case impresso - utilizando filamento Amarelo	110
Figura 93 - Suporte e o guia do filamento	111
Figura 94 - Tela de monitoramento da impressão	112
Figura 95 - G-Code Viewer - Layer de impressão	113
Figura 96 - Monitoramento através da imagem.....	113
Figura 97 - Tela de monitoramento através do celular	114
Figura 98 - Especificações referente aos corpos de prova ASTM638-10	115
Figura 99 - Modelo 3D corpo de prova ASTM-D638-10	115
Figura 100 - Comparativo de teste de tração nos polimeros PLA, ABS e PETG	118
Figura 101 - Gráfico tração x deformação PLA espessura de camada 0,4mm	119
Figura 102 - Gráfico tensão x deformação PLA p 20%, espessura de camada de 0,4mm	119
Figura 103 - Gráfico tensão x deformação PLA com tratamento p 20% espessura de camada de 0,4mm.....	120
Figura 104 - Gráfico tensão x deformação PLA, p 100% espessura de camada 0,4mm	121
Figura 105 - Gráfico tensão x deformação PLA com tratamento térmico p 100% espessura de camada de 0,4mm	122
Figura 106 - Propriedades dos materiais PLA, ABS e PETG	123
Figura 107 - Media das durezas PLA, ABS e PETG Shore D	124
Figura 108 - Media dureza shore D.....	125

LISTA DE SIGLAS

3D Três Dimensões
ABS Acrilonitrila Butadieno Estireno
AM Additive Manufacturing
CAD Computer Aided Design
CAM Computer Aided Manufacturing
CNC Comando Numérico Computadorizado
DIY Do it Yourself
FDM Fused Deposition Modeling
FFF Fusion Filament Fabrication
GNUGPL General Public License
IDE Integrated Development Environment
IOT Internet das coisas
LCD Liquid Crystal Display
MDF Medium Density Fiberboard
PCI Placas de Circuito Impresso
PLA Ácido Poliláctico
SD Secure Digital
SLA StereoLithoGraphy
STL Standard Template Library
TDP Three Dimensional Printing
USB Universal Serial Bus
UV Radiação Ultravioleta

1 INTRODUÇÃO

Nos últimos anos, testemunhou-se um avanço tecnológico notável no campo da impressão 3D. Essa tecnologia inovadora tem o potencial de revolucionar a forma como produzimos e fabricamos objetos, possibilitando a criação de produtos personalizados, complexos e de alta qualidade. Nesta dissertação, será explorado os princípios e as aplicações da impressora 3D, bem como seu impacto nas indústrias e na sociedade em geral.

(Da Costa, 2017) descreve o modelo industrial como estando em constante evolução, isto é causado pelo desenvolvimento e utilização de novas tecnologias, o que levou a uma nova era na indústria, que é conhecida como indústria 4.0.

Esse cenário envolve um impacto significativo no cotidiano da sociedade e na progressão da tecnologia, pois os avanços tecnológicos estão diretamente relacionados ao desenvolvimento humano (AGOSTINI; FILIPPINI et al., 2019).

Além disso, o processo produtivo está se tornando cada vez mais dinâmico, inteligente, ágil e flexível. Fabricar produtos inteligentes e utilizar equipamentos inteligentes tornam a produção mais inteligente e autônoma. (HUBA e KOZÁK et al., 2016).

A prototipagem 3D tem grande potencial para alterar os modelos de produção atuais como alternativa à criação de dispositivos e objetos personalizados com diferentes geometrias e difíceis de fabricar utilizando técnicas de fabricação tradicionais. A tecnologia está em constante aperfeiçoamento e uma ampla gama de materiais pode ser impressa, sejam eles funcionais, simples ou compostos. (KALSOOM et al., 2018).

(VOLPATO et al., 2007) sugere que a prototipagem rápida permite a criação de técnicas de desenvolvimento de produtos divididas em várias etapas e incentiva a colaboração de vários departamentos, como planejamento, desenvolvimento e execução. O objetivo principal é produzir peças funcionais, protótipos e modelos de objetos em um curto espaço de tempo, respeitando as especificações necessárias para o produto final. Na análise de Pipes (2010), a prototipagem rápida traz uma vantagem significativa: a capacidade de visualizar produtos, proporcionando uma compreensão tridimensional dos modelos. Esses protótipos podem ser feitos em pequena escala, tornando a impressão 3D uma técnica hoje amplamente utilizada na fase de projeto. Este

método permite que produtos sejam produzidos diretamente a partir de um modelo computacional, eliminando a necessidade de ferramentas tradicionais e reduzindo tempo e custos de produção.

Existem atualmente no campo de manufatura aditiva, várias tecnologias de impressão 3D, sendo estas:

- Extrusão de materiais (FDM/FFF), que realiza o depósito de um filamento ou material fundido;
- Sinterização seletiva a laser (SLS), que realiza a solidificação e união controlada de pequenas partículas, areias e plástico;
- Fotopolimerização (SLA/ DLP) utilizam o princípio de fotopolimerização, ou seja, utilizam uma resina como matéria-prima e é curada através de raios ultravioleta;
- Sinterização seletiva a laser a metais (DMLS/SLM), que realiza a solidificação e união de materiais metálicos;

Lemke et al. (2016) afirmaram que no meio acadêmico a impressão 3D tornou-se uma opção viável para a produção de objetos físicos que podem ser utilizados como objetos de aprendizagem em diversas áreas e em diferentes níveis educacionais.

Nesse sentido, a manufatura aditiva pode proporcionar aos estudantes de graduação uma experiência pedagógica que certamente contribuirá para a formação de profissionais alinhados com as novas tecnologias, iniciando os alunos na cultura da indústria 4.0 (SANTOS et al., 2018).

Diversos materiais estão disponíveis para o FDM, incluindo ABS, Nylon, Policarbonato, PLA, entre outros. Para avaliar como cada material se comporta em determinadas situações, é necessário compará-los usando parâmetros predefinidos (Monteiro et al., 2012).

Conforme Carneiro et al. (2019), esses parâmetros têm um impacto direto no resultado final da peça, influenciando seu dimensionamento e propriedades mecânicas. Bolelli e Lebão (2019) destacam a importância de considerar uma variedade de parâmetros de impressão, pois eles determinam as características do protótipo na fase final.

Alguns exemplos de parâmetros que podem ser ajustados incluem a geometria interna do modelo, densidade de preenchimento, orientação durante o processo de deposição de filamentos e velocidade de produção.

O estudo da engenharia é complexo e apresenta grandes desafios para os alunos na aprendizagem. As tecnologias disponíveis sempre foram consideradas ferramentas didáticas importantes. Nesse contexto, a utilização da manufatura aditiva no ambiente acadêmico pode concretizar a teoria em objetos físicos, estimulando o incentivo, a criatividade e a inovação entre os futuros profissionais (MORAES, 2018).

Ao lado do desenvolvimento contínuo da manufatura aditiva, outra tecnologia que vem ganhando destaque tanto nas indústrias quanto no meio acadêmico é a IoT (Internet das Coisas). Segundo Santos et al. (2016), a IoT surgiu da evolução de várias tecnologias, incluindo sistemas embarcados, microeletrônica, sensoriamento e comunicação.

De acordo com Freitas (2017), em sua obra, a IoT é uma tendência irreversível devido à sua capacidade de conectar e simplificar tarefas cotidianas, além de possibilitar que as indústrias usem essa tecnologia para analisar e compartilhar informações cruciais dos produtos pela internet. Nesse sentido, a capacidade de monitoramento e análise em tempo real pela internet torna-se atraente para diversos setores, guiando os novos processos industriais.

Para execução desse trabalho foi seguindo o conceito RepRap (Replicating Rapid-prototyper), e posteriormente configurada através do firmware conforme a estrutura utilizada e para a calibração foram impressos modelos específicos para validação alterando alguns parâmetros, além da proposta de poder efetuar variações de diversos parâmetro para validação de materiais dentro do meio acadêmico possibilitando vários estudos. No contexto da aplicação do conceito IoT, uma interface web Octoprint foi incorporada para permitir o gerenciamento, controle e monitoramento remotos das operações. Além disso, a operação pode ser monitorada remotamente por meio de imagens. Podemos ressaltar que a impressora terá como base um Arduino que fará o controle de todos o processo acoplado a uma interface eletrônica (Ramps). As etapas seguidas para a montagem da impressora foram divididas em mecânica, eletrônica, firmware e software de gerenciamento. Para todos os processos que envolvem firmware

e software, foram utilizados sistemas livres (Open Source) para que fosse possível efetuar uma customização e tornar o projeto de baixo custo, entre eles podemos citar: Marlin, Octoprint, e fatiadores como Cura.

Conforme apresentado a impressora 3d poderá ser aplicada ou inserida em vários meios, entre eles o meio industrial e o acadêmico e buscando contribuir na evolução do estudo e no aperfeiçoamento do profissional ou aluno, propôs-se efetuar a construção de uma impressora 3D para fins didáticos (da teoria à prática) de forma a possibilitar ao aluno a analisar todo o processo dessa montagem desde a mecânica até a aplicação no conceito da indústria 4.0. Outro ponto a ser verificado é que com esse tipo de tecnologia de manufatura aditiva aplicada ao meio acadêmico é possível modelar e estudar as estruturas dos materiais aplicados e verificar se houve influência nas propriedades do PLA em materiais confeccionados pela impressora construída.

1.1 Objetivo geral

O objetivo deste trabalho é realizar a construção de uma impressora 3D de baixo custo, utilizando firmware e software open source e gratuitos, passo a passo para fins acadêmicos, mostrando da teoria à prática e integrá-la ao conceito indústria 4.0 e efetuar a validação através de ensaios das propriedades mecânica dos materiais impressos, efetuando uma previa análise da influência nas propriedades do PLA em materiais confeccionados por essa impressora.

1.2 Objetivos específico

Para esse trabalho pretende-se:

- Conceituar Impressão 3D, suas aplicações no mercado, estrutura eletrônica, estrutura mecânica e software.
- Conceituar Polímeros (filamentos).
- Desenvolver uma impressora 3D, capaz de imprimir qualquer tipo de material por extrusão.
- Inserir a impressora no conceito indústria 4.0 aplicando o IoT;

- Imprimir, verificar e caracterizar um filamento de 2 fabricantes, para verificar a influência ocorrida em suas propriedades em materiais confeccionados por essa impressora.

2 REFERENCIAL TEORICO

Neste tópico, serão abordados os itens que servem de base para o estudo, referente a manufatura aditiva e ao funcionamento de um equipamento já bem divulgado no mercado que é a impressora 3D, da teoria à prática e a introdução no conceito indústria 4.0, usando a tecnologia IoT.

2.1 Industria 4.0

Na última década, testemunhou-se uma revolução industrial que está redefinindo o panorama econômico global: a Indústria 4.0. Este termo, originado na Alemanha, refere-se à quarta revolução industrial, que se caracterizou pela integração de tecnologias digitais avançadas nas operações industriais (Schwab, 2017). Ao contrário de suas predecessoras, a Indústria 4.0 não é apenas sobre a automação, mas também sobre a interconexão e análise de dados em tempo real para otimizar a produção, melhorar a eficiência e impulsionar a inovação.

Neste contexto, tecnologias como inteligência artificial, Internet das Coisas (IoT), big data, realidade aumentada e impressão 3D desempenham papéis cruciais. Pode-se dizer que o IoT conecta máquinas e dispositivos, permitindo a coleta instantânea de dados, enquanto a inteligência artificial possibilita a análise preditiva e a automação inteligente. O big data fornece insights valiosos a partir de grandes conjuntos de dados, enquanto a realidade aumentada transforma a forma como os trabalhadores interagem com o ambiente de produção. Além disso, a impressão 3D está revolucionando o design e a fabricação de produtos (Kagermann et al., 2013).

Esta revolução industrial não apenas transforma a maneira como os produtos são fabricados, mas também cria novas oportunidades de negócios e modelos econômicos. Empresas que adotam a Indústria 4.0 podem personalizar a produção em massa, reduzir

o desperdício, melhorar a qualidade dos produtos e acelerar o tempo de colocação no mercado. Além disso, essa transformação impulsiona a criação de empregos especializados, fomentando o desenvolvimento de habilidades relevantes para o futuro (Manyika et al., 2015).

2.2 Manufatura Aditiva no Meio Acadêmico: Inovação e Desafios

Nos círculos acadêmicos, a Manufatura Aditiva (MA), também conhecida como impressão 3D, emergiu como uma das tecnologias mais promissoras e transformadoras dos tempos modernos (Gibson, Rosen & Stucker, 2014). Originada nas décadas de 1980 e 1990, a Manufatura Aditiva (MA) possibilita a criação de objetos tridimensionais, construídos camada por camada, utilizando uma variedade de materiais, tais como polímeros, metais e cerâmicas. Esta técnica inovadora não apenas revoluciona a forma como os produtos são fabricados, mas também catalisa avanços significativos na pesquisa acadêmica, estimulando estudos interdisciplinares e colaborações inovadoras.

A aplicação da Manufatura Aditiva no meio acadêmico oferece um terreno fértil para a exploração científica e tecnológica. Pesquisadores em engenharia, ciência dos materiais, design industrial e outras disciplinas encontram na MA um campo vasto para investigações, levando a descobertas que vão desde novos materiais com propriedades excepcionais até métodos avançados de design e otimização estrutural (Gibson, Rosen & Stucker, 2014). A capacidade de produzir geometrias complexas e personalizadas impulsiona estudos inovadores em campos tão diversos como medicina, arquitetura e ciências biológicas.

No entanto, o meio acadêmico também enfrenta desafios significativos ao lidar com a Manufatura Aditiva. Questões relacionadas à padronização, durabilidade e segurança dos produtos fabricados ainda são áreas de pesquisa intensiva (Bourell et al., 2017).

Além disso, a otimização dos processos de impressão, a redução de custos e a sustentabilidade ambiental são temas cruciais que demandam a atenção dos acadêmicos e pesquisadores.

Neste cenário dinâmico, este artigo explora a aplicação da Manufatura Aditiva no meio acadêmico, destacando não apenas as inovações empolgantes que surgiram, mas

também os desafios persistentes que continuam a inspirar pesquisas intensivas (Attaran, 2017).

Através da análise crítica dessas questões, busca-se não apenas ampliar o conhecimento existente, mas também fornece orientações valiosas para futuras pesquisas, promovendo um ambiente acadêmico ainda mais robusto e colaborativo.

2.3 Microcontroladores

Na era digital, onde a automação e a interconectividade são essenciais, os microcontroladores surgem como os heróis silenciosos, capacitando uma variedade impressionante de dispositivos eletrônicos que permeiam nossas vidas diárias. Essas minúsculas maravilhas tecnológicas integram em um único chip um processador, memória e interfaces de entrada/saída, tornando-se os cérebros por trás de inúmeros sistemas eletrônicos, desde eletrodomésticos até veículos inteligentes.

Os microcontroladores desempenham um papel crucial na evolução tecnológica, permitindo a criação de dispositivos inteligentes e autônomos. Sua capacidade de processamento e flexibilidade os torna fundamentais em diversas aplicações, incluindo robótica, sistemas embarcados, e controle de dispositivos eletrônicos (Ayala, 2006).

Uma das características mais notáveis dos microcontroladores é sua versatilidade. Eles podem ser programados para se adaptarem a uma ampla gama de tarefas, desde simples operações de controle até cálculos complexos em tempo real. Além disso, a capacidade de personalização dos microcontroladores permite que os engenheiros os configurem para atenderem às necessidades específicas de cada aplicação (Valvano, 2012).

No contexto da Indústria 4.0, onde a automação inteligente e a conectividade são imperativos, os microcontroladores desempenham um papel crucial. Eles não apenas impulsionam a eficiência operacional, mas também facilitam a inovação, abrindo caminho para novos produtos e serviços inovadores que transformam setores inteiros da economia (Ullah *et al.*, 2019).

2.3.1 Microcontroladores na Impressão 3D

Na revolução da Impressão 3D, os microcontroladores desempenham um papel central, funcionando como o cérebro por trás da precisão e eficiência do processo. Esses dispositivos inteligentes, essenciais para o funcionamento de impressoras 3D, coordenam cada movimento, controlam temperaturas e garantem que os objetos sejam impressos com detalhes impressionantes.

Os microcontroladores são responsáveis por transformar comandos digitais em ações físicas. Na Impressão 3D, eles controlam motores, sensores e aquecedores, garantindo que o processo ocorra de forma precisa e eficiente. Além disso, eles coordenam o movimento do cabeçote de impressão e a deposição de material camada por camada, resultando em objetos tridimensionais precisos (Huang, Zhang, & Zhang, 2014).

Cada impressora 3D tem requisitos específicos, e os microcontroladores são frequentemente customizados para atender a essas necessidades. Isso envolve programação intrincada para garantir que a impressora funcione de acordo com as especificações desejadas, o que pode incluir variáveis como velocidade de impressão, temperatura do bico e adesão à superfície de impressão (Cruz, Carretero, & Úbeda, 2019).

Com o avanço da tecnologia, os microcontroladores utilizados na Impressão 3D estão em constante evolução. Novos dispositivos são desenvolvidos para lidar com desafios específicos, como a redução do tempo de impressão, a melhoria da qualidade do objeto final e a implementação de sensores avançados para monitoramento em tempo real (Cruz, Carretero, & Úbeda, 2019).

2.3.2 Microcontroladores e Placas de Controle em Impressoras 3D:

As impressoras 3D, contam com uma variedade de microcontroladores e placas de controle para operar de maneira precisa e eficiente. A escolha do dispositivo adequado pode significar a diferença entre uma impressão de qualidade e uma experiência insatisfatória. Entre as opções disponíveis, alguns dos microcontroladores notáveis incluem:

- **Arduino Mega:** O Arduino Mega é uma escolha confiável e popular entre os entusiastas de impressão 3D. Sua flexibilidade e facilidade de uso o tornam uma opção versátil para uma variedade de impressoras 3D personalizadas (Arduino, s.d.).
- **RAMPS (RepRap Arduino Mega Pololu Shield):** A RAMPS é uma placa de expansão que se integra perfeitamente com o Arduino Mega. Ela oferece suporte para até cinco motores de passo, sensores de temperatura e outros periféricos essenciais para o funcionamento adequado de uma impressora 3D (Reprap.org, [s.d.]).
- **Smoothieboard:** A Smoothieboard, baseada no ARM Cortex-M3, é conhecida por sua velocidade de processamento e precisão. Oferece um ambiente de controle confiável para impressoras 3D que exigem alta qualidade em impressões complexas (Smoothieware, [s.d.]).
- **Duet 2:** A Duet 2 é uma placa de controle avançada equipada com um microcontrolador ARM de 32 bits. Ela oferece recursos avançados, incluindo conectividade Wi-Fi integrada, tornando-a uma escolha popular para impressoras 3D que demandam funcionalidades de rede (Duet3D, [s.d.]).
- **Raspberry Pi:** Embora tradicionalmente conhecido como um minicomputador, o Raspberry Pi também é usado como uma interface de controle para impressoras 3D. Com a ajuda de distribuições específicas, como OctoPrint, o Raspberry Pi permite controle remoto e monitoramento das impressões por meio de uma interface web (OctoPrint, [s.d.]).

Essa diversidade de microcontroladores e placas de controle oferece aos entusiastas de impressão 3D uma gama de opções para atender a diversas necessidades e orçamentos. Cada uma dessas escolhas reflete a contínua evolução da tecnologia que impulsiona o mundo da impressão 3D.

2.4 Impressoras 3D

A impressão 3D é uma das tecnologias mais revolucionárias do século XXI. Essa inovação tem o potencial de transformar o mundo da manufatura, abrindo novas possibilidades e desafiando os limites da criatividade humana.

A impressora 3D tem despertado o interesse de empresas, pesquisadores e entusiastas em todo o mundo devido à sua capacidade de criar objetos tridimensionais utilizando uma variedade de materiais.

A tecnologia da impressão 3D, também conhecida como fabricação aditiva, funciona por meio de um processo camada por camada, no qual o objeto é construído a partir de um modelo digital. Utilizando diferentes técnicas, como deposição de material fundido (FDM), estereolitografia (SLA) e sinterização seletiva a laser (SLS), as impressoras 3D têm a capacidade de criar uma ampla variedade de objetos, desde protótipos complexos até peças funcionais e até mesmo órgãos humanos.

Uma das maiores vantagens da impressão 3D é a liberdade de design que oferece. Ao contrário dos métodos de fabricação tradicionais, que muitas vezes são limitados por formas e geometrias específicas, a impressão 3D permite a criação de formas intrincadas e personalizadas. Isso abre um novo mundo de possibilidades para designers, arquitetos e engenheiros, que agora podem materializar suas ideias de forma rápida e eficiente.

Segundo (KIETZMANN et al., 2015), as primeiras pesquisas sobre impressoras 3D datam de 1970. A diferença entre os métodos novos utilizados e os que existiam na época era que eles imprimiam camada por camada, tornando-se produção aditiva, enquanto o método existente era subtrativo, ou seja, retirava o material que formava o modelo final.

Segundo (Hausman e Horne (2014)), a impressão 3D foi desenvolvida depois que o pesquisador britânico Adrian Bowyer criou o sistema RepRap, o primeiro sistema de prototipagem rápida auto-replicante, reutilizando motores de sua loja de ferragens e materiais, capaz de imprimir um grande número de componentes para fabricar novas impressoras 3D ou melhorar as existentes.

Segundo a pesquisa, o engenheiro mecânico Adrian Bowyer publicou on-line o primeiro documento discutindo o projeto RepRap em 2004, e montou a primeira impressora 3D produzida pelo método em 2007, espalhando o conhecimento para outros interessados. muito superior ao das gráficas comerciais da época (JONES et al., 2011).

Há diversos tipos de modelos de impressoras 3D no mercado, sendo impressoras de custo mais elevados para âmbitos industriais e modelos com custo mais baixos para

âmbitos residenciais e pequenas empresas. Pode-se verificar na Figura 1 alguns modelos.

Figura 1 - Tipos de impressoras 3D



Fonte: <https://www.printit3d.com.br/post/tipos-de-impressora-3d> (2023)

Existem projetos de impressoras 3D em sites utilizando a metodologia de código aberto (Open source) e software gratuito, que são de fabricação de baixo custo e utilizam-se da cultura Maker. Essa cultura se baseia no sistema RepRap citada anteriormente por Hausman e Horne (2014) que são impressoras autorreplicáveis, utilizando-se de peças reutilizáveis de outros equipamentos ou até mesmo adquirindo kits de baixo custo para montagem dessas impressoras, conforme mostra a Figura 2.

Figura 2 - Kit para impressoras 3D



Fonte: Reprap (2023)

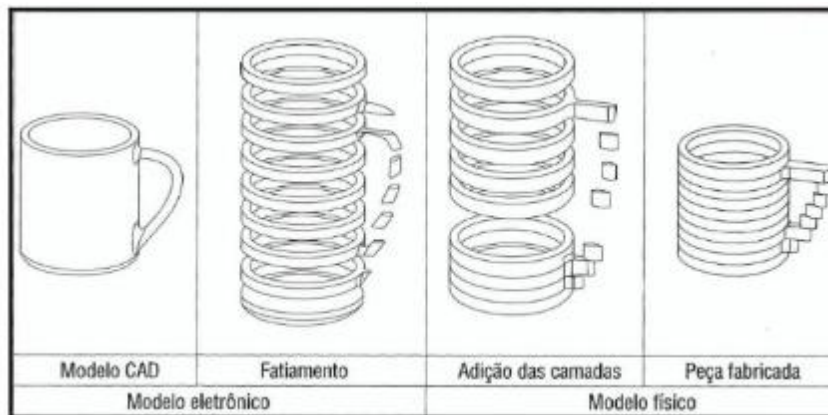
Para possibilitar o processo de impressão em 3D, softwares especializados são responsáveis por dividir uma imagem tridimensional em inúmeras camadas. Essas camadas são então impressas sequencialmente. Esses programas realizam a divisão

dos arquivos no formato "STL" (Stereolithography, também conhecido como Standard Tessellation Language). Esse formato descreve as superfícies de um objeto através dos vértices que compõem cada uma de suas camadas (AZEVEDO et al., 2013). A segmentação do objeto é programada em um arquivo eletrônico que indica à impressora o caminho necessário para cada camada durante a impressão.

Uma unidade de controle recebe esses comandos e controla a posição do bico extrusor usando motores de passo e sistemas de transmissão, garantindo que todas as camadas se sobreponham corretamente até que o objeto seja totalmente fabricado (ALCAIDE; WILTGEN, 2018).

Na Figura 3, pode ser visto as principais etapas da impressão 3D.

Figura 3 - Principais etapas da impressão 3D



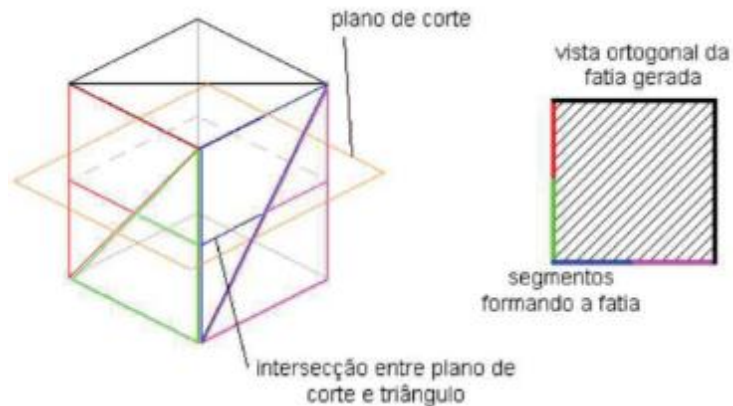
Fonte: Volpato et al. (2006)

Nos dias atuais, a maioria dos sistemas CAD tem a capacidade de realizar o processamento matemático para criar um arquivo STL. Para o usuário, o processo geralmente é tão simples quanto selecionar as opções "Arquivo", "Salvar como" e "STL".

Após a obtenção do arquivo digital modelado, o próximo passo é fatiar esse modelo. Nesse processo, o modelo é orientado em relação a uma superfície que representa a mesa de impressão e é dividido em várias camadas horizontais. A espessura dessas camadas é crucial para garantir uma boa qualidade de impressão, sendo que camadas mais finas resultam em melhor resolução.

Na Figura 4 pode ser analisado a vista superior de uma das camadas geradas a partir de um modelo STL.

Figura 4 - Fatiamento de um modelo



Fonte: Badotti et al. (2003)

Uma vez concluída a operação matemática de fatiamento, o software de fatiamento gera um código de máquina que a impressora 3D compreende e executa. Esse código é conhecido como G-Code. Esta linguagem é bastante variada, com comandos e implementações muito específicos, diferenciando-se de outras linguagens de programação pelo fato de seus comandos serem compactados e representados por uma letra e um número. Ter um entendimento básico dessa linguagem permite ao usuário avaliar o arquivo gerado pelo software de fatiamento (G-Code) e corrigir possíveis falhas ou defeitos que possam surgir durante o processo de impressão. De acordo com SAMPAIO (2017), a sintaxe do código G-Code pode ser representada da seguinte forma:

Comando: parâmetro do comando

Os comandos são palavras-chave compostas por letras e números que definem qual operação a impressora realizará. Os mais comuns são "G" e "M", relacionados à movimentação e extrusão da máquina. Os parâmetros também são dados por letras e números, com os tipos mais comuns sendo:

- "S": usado para designar um parâmetro genérico para distância ou temperatura, empregado em comandos de retração e ajuste de hotend e mesa;

- "F": usado para designar velocidade;
- "X", "Y" e "Z": coordenadas e distâncias;
- "E": quantidade de filamento do extrusor;
- "P": parâmetro para comandos variados.

Os comentários servem apenas como orientação para os desenvolvedores e não influenciam nos comandos da impressora. Pode ser verificado abaixo um exemplo dos comandos e seus parâmetros:

G28: Este comando executa o procedimento de "home" da impressora, posicionando todos os eixos no ponto inicial.

G90: Ativa o modo de movimento absoluto no interpretador, garantindo que todas as coordenadas fornecidas no código G sejam interpretadas como posições específicas no espaço lógico das coordenadas.

G0 e G1: Estes comandos indicam movimento linear e geralmente são seguidos por coordenadas de destino e velocidade.

M119: Recebe sinais dos sensores de fim de curso, indicando se determinadas posições foram alcançadas pelos eixos da impressora.

G1 X20 Y20 F180: Move o bico para a posição X 20, Y 20 a uma velocidade de 180 unidades por minuto, sem extrusão de filamento.

G0 X30 Y30 Z30 F180 E0.8: Move o bico para as posições X 30, Y 30, Z 30 a uma velocidade de 180 unidades por minuto, enquanto extrude 0,8 unidades de comprimento de filamento ao longo dessa trajetória.

M665: Permite a modificação dos parâmetros de uma impressora do tipo Delta, ajustando suas configurações específicas.

M92 E100: Define a taxa de avanço do extrusor, indicando o quanto de filamento deve ser extrudado em relação ao movimento dos eixos da impressora.

Alguns G-Codes e M-Codes estão no anexo E.

2.4.1 Processo de prototipagem e engenharia reversa

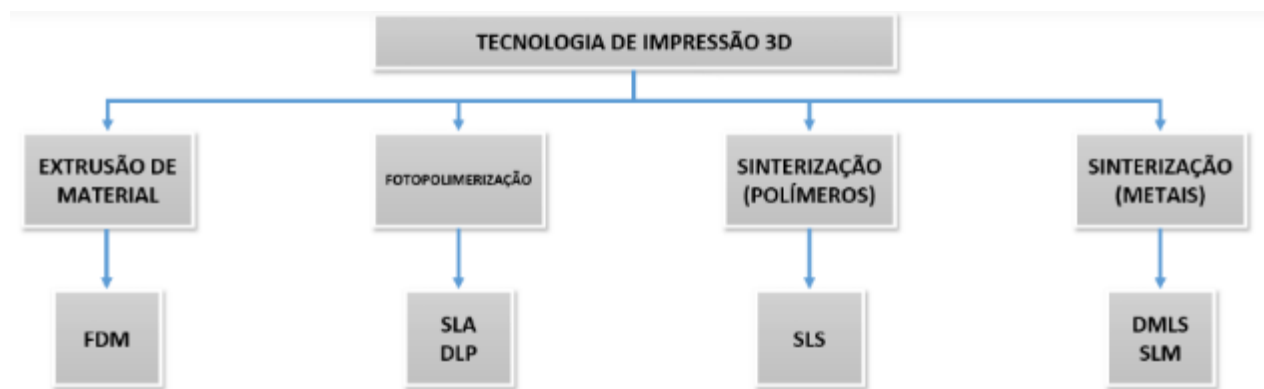
Conforme (JUNIOR, 2005), a engenharia reversa se trata da tentativa de entender o funcionamento de um produto, possibilitando o desenvolvimento de um produto similar com dimensões aproximadas, apontando algumas empresas, como a IBM-PC, que utilizam dessa técnica de engenharia em seus meios de produção.

Afirma-se (VIANNA, 2012), que o processo de engenharia reversa está ligado a diversas áreas da engenharia como, projetos de máquinas e prototipagem, apresentando uma ligação muito forte com a tecnologia de prototipagem rápida. No livro Design Thinking, inovações em negócios, define a prototipagem como sendo: “Prototipar é tangibilizar uma ideia, a passagem do abstrato para o físico de forma a representar a realidade – mesmo que simplificada – proporcionar validações”.

2.4.2 Principais métodos de prototipagem 3D

No processo de impressão 3D há diversos tipos de modelos que podem ser aplicados ao processo por impressão aditiva conforme mostrado na Figura 5 alguns tipos.

Figura 5 - Tecnologias de impressão 3D



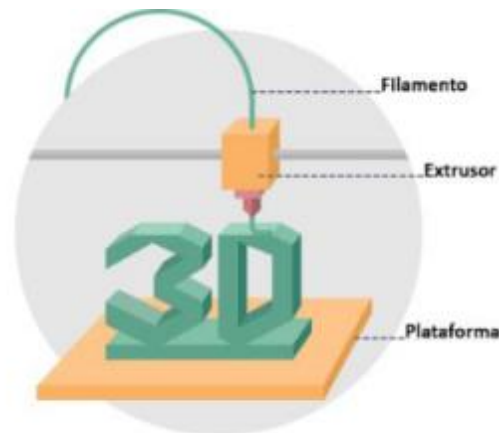
Fonte: Portela [s.d.]

2.4.2.1 Processo de Fusão e Deposição de Material (FDM)

O processo FDM (Modelagem por Deposição Fundida) é o mais popular entre os processos de impressão 3D, amplamente utilizado em diversos setores econômicos e

educacionais devido à sua versatilidade e praticidade. Neste processo, um filamento plástico é depositado, camada por camada, e fundido pelo bico injetor para formar produtos de plástico sobre uma superfície plana (BAGLIOTTI et al., 2017). A Figura 6 representa o esquema para a produção de um produto com a utilização do processo FDM, no qual, basicamente, ocorre a movimentação do bico injetor nos eixos x e y e da mesa no eixo z, que, através de um controlador, deposita material nas regiões demandadas.

Figura 6 - Esquemática geral do processo FDM



Fonte: THRE3D, 2014

Para iniciar o processo de fabricação, é necessário primeiro obter um modelo virtual da peça desejada. Após desenvolver o modelo virtual, o arquivo é geralmente exportado em formatos "STL" e enviado para um software de fatiamento. Esse software divide a peça em camadas e gera um arquivo Gcode, que contém os comandos necessários para a impressora 3D. O Gcode orienta o bico injetor a se movimentar sobre a superfície de trabalho, realizando a fabricação do modelo camada por camada.

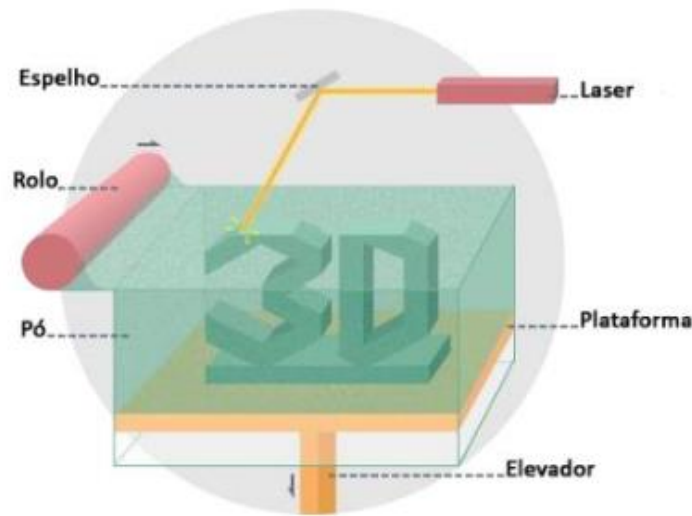
2.4.2.2 Processo de Sinterização Seletiva a Laser (SLS)

Inventada em 1980 por pesquisadores da Universidade do Texas, a técnica direciona um feixe de laser de alta potência sobre uma superfície depositada com pó, que é sinterizado formando uma camada sólida. Após a criação de uma camada, um rolo com

material deposita novo pó e assim sucessivamente até completar a peça (BAGLIOTTI et al., 2017).

Na Figura 7 representa o esquema para a produção de um produto com a utilização do processo SLS, onde um laser funde as partículas de um pó de polímero, construindo a peça camada por camada.

Figura 7 - Esquemática geral do processo SLS.



Fonte: THRE3D, 2014

Muitas das máquinas SLS para metais ainda não são adequadas para o uso doméstico, pois precisam de uma câmara selada e preenchida com nitrogênio para que o oxigênio não reaja na fusão do material. Como este é um processo quente, na fabricação peças grandes podem ser necessárias esperar até um dia para que se resfrie e possa ser retirada do seu interior e, além disso, alguns materiais oferecem o risco de explodir se trabalhados incorretamente (LIPSON e KURMAN, 2013).

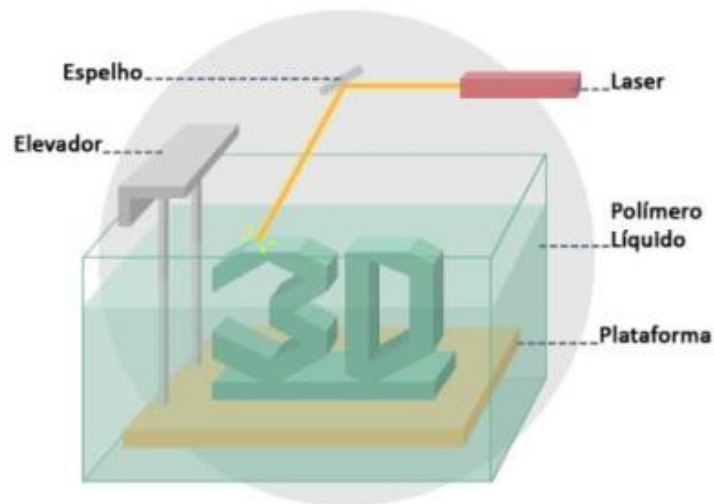
2.4.2.3 Processo de Stereolitografia (SLA)

Neste sistema, um feixe de laser passa através da superfície de um fotopolímero líquido sensível aos UV, que endurece quando exposto à luz UV à medida que o feixe de laser percorre um caminho definido e endurece o polímero. Após a conclusão de uma camada, a plataforma que sustenta a peça é abaixada alguns décimos de milímetro para

que o líquido no tanque cubra toda a superfície recém-criada e o processo recomeça (MANYIKA et al., 2013).

A Figura 8 demonstra a esquemática para a produção de um produto com a utilização do processo SLA, no qual, basicamente, ocorre a movimentação do laser nos eixos x e y e da mesa no eixo z, que, através de um controlador, solidificando o material nas regiões demandadas.

Figura 8 - Esquemática geral do processo SLA.



Fonte: THRE3D, 2014

Porém, a tecnologia só consegue imprimir cada peça com um único tipo de material, e os fotorpolímeros utilizados não são tão duráveis quanto os utilizados industrialmente na injeção de plástico. Além disso, o custo e a complexidade de manutenção de tais máquinas continuam muito importantes para os usuários domésticos, fator que abre oportunidades para o desenvolvimento de mercado de modelos de baixo custo utilizando lasers UV (LIPSON e KURMAN, 2013).

As técnicas mais comuns usadas na impressão 3D são extrusão plástica ou modelagem por difusão (FDM). O método envolve o aquecimento de um material (filamento plástico) até a temperatura de fusão do material, convertendo o material em um fluido que solidifica ao entrar em contato com uma superfície de temperatura mais baixa. Esse processo consiste em aplicar camada sobre camada até que a peça seja finalizada, conforme Figura 9 (REIS, 2016).

Figura 9 - Representação das principais etapas do processo de impressão camada por camada.



Fonte: ACRONSOFT. (2019)

2.5 Polímeros

O termo "polímero" deriva do grego, onde "poli" significa muitos e "meros" refere-se a partes. Um polímero é, essencialmente, uma estrutura macromolecular composta por uma sequência repetitiva de unidades monoméricas. A maneira como essas unidades são dispostas e interagem confere aos polímeros uma variedade de propriedades, desde plásticos flexíveis até fibras resistentes. Esse arranjo único permite a adaptação dos polímeros para satisfazer uma ampla gama de necessidades industriais e de consumo (O'dian, 2004).

2.5.1 Tipos de Polímeros

Os polímeros, estruturas moleculares macromoleculares, podem ser classificados em três categorias principais com base em seu comportamento térmico e propriedades mecânicas: termoplásticos, termofixos (ou termorrígidos) e elastômeros. Esta exploração detalhada dos tipos de polímeros destaca suas estruturas distintas e aplicações variadas, respaldadas por pesquisas acadêmicas significativas no campo da ciência dos materiais.

- **Termoplásticos: Flexibilidade e Reciclabilidade:** Os termoplásticos são polímeros que podem ser derretidos repetidamente quando aquecidos e solidificados quando resfriados, mantendo sua estrutura química. Exemplos incluem polietileno (PE) e polipropileno (PP). Eles são amplamente utilizados

devido à sua versatilidade, facilidade de moldagem e capacidade de reciclagem. A natureza linear ou ramificada desses polímeros permite uma reorganização molecular reversível durante o aquecimento, tornando-os ideais para processos de fabricação como moldagem por injeção (Callister Jr., 2006).

- **Termofixos (ou termorrígidos): Durabilidade e Estabilidade Térmica:** Os termofixos são polímeros que, uma vez moldados e solidificados, não podem ser derretidos ou remodelados pelo calor sem degradação química. Exemplos incluem resinas epóxi e poliuretanos termofixos. Sua estrutura reticulada proporciona alta resistência mecânica, estabilidade térmica e resistência química, tornando-os ideais para aplicações que exigem durabilidade e rigidez, como compósitos de fibra de carbono em indústrias aeroespaciais e automotivas (Gao, Ma, & Fang, 2019).
- **Elastômeros: Flexibilidade e Resiliência:** Elastômeros são polímeros com estrutura molecular reticulada, proporcionando flexibilidade e resiliência excepcionais. Eles podem ser esticados e retornam à sua forma original quando liberados. Borracha natural e polibutadieno são exemplos de elastômeros. Sua capacidade de retornar à forma original após deformações significativas faz deles ideais para aplicações que requerem vedação, amortecimento e flexibilidade, como pneus e vedantes industriais (Mark, 2007).

A compreensão das características distintas desses tipos de polímeros é essencial para selecionar o material adequado para diversas aplicações, desde embalagens até componentes de alta tecnologia.

2.5.2 Comportamento Térmico dos Polímeros

O comportamento térmico dos polímeros é fundamental para entender suas propriedades e aplicações. Três transições de fase importantes são comumente estudadas: a temperatura de transição vítrea (T_g), a temperatura de fusão cristalina (T_m) e a temperatura de cristalização (T_c). Estas transições de fase são indicativas de mudanças significativas na estrutura molecular dos polímeros e têm implicações diretas nas propriedades mecânicas e térmicas dos materiais poliméricos.

- **Temperatura de Transição Vítrea (Tg):** A Tg é a temperatura abaixo da qual um polímero amorfo se torna rígido e frágil, perdendo sua capacidade de fluir. Esta transição é crucial para determinar a faixa de temperatura na qual um polímero pode ser utilizado com sucesso. Polímeros com Tg mais altas, como o polimetilmetacrilato (PMMA), são usados em aplicações de alta temperatura, enquanto polímeros com Tg mais baixas, como o poliestireno (PS), são mais adequados para temperaturas ambiente (Paul, 2014).
- **Temperatura de Fusão Cristalina (Tm) e Temperatura de Cristalização (Tc):** A Tm é a temperatura à qual um polímero cristalino funde e passa do estado sólido para o líquido. A Tc, por outro lado, é a temperatura à qual um polímero amorfo começa a cristalizar. Ambas as transições estão relacionadas à organização molecular dos polímeros. Polímeros com alta Tm, como o polietileno de alta densidade (PEAD), são usados em aplicações de alta resistência, enquanto polímeros com Tc bem definidas, como o polietileno de baixa densidade (PEBD), são importantes para filmes plásticos e embalagens (Cowie & Arrighi, 2008).

2.5.3 Propriedades Mecânicas dos Polímeros

As propriedades mecânicas dos polímeros desempenham um papel vital na sua seleção e aplicação em várias indústrias, desde engenharia até medicina, conforme citado.

- **Resistência:** A resistência mecânica dos polímeros é uma medida de sua capacidade de suportar cargas sem deformação permanente ou ruptura. É essencial entender a resistência à tração, à compressão e à flexão para determinar a adequação de um polímero em diversas aplicações, Karger-Kocsis e Mahmoodi (2008).
- **Tenacidade:** A tenacidade dos polímeros está relacionada à quantidade de energia que um material pode absorver antes de falhar. Polímeros tenazes, como o polipropileno modificado com elastômero (EPDM), são amplamente

usados em aplicações que requerem resistência ao impacto, como para-choques de carros, Thomason (2002).

- **Dureza e Elasticidade:** A dureza e elasticidade dos polímeros são características cruciais em aplicações que envolvem contato com superfícies ou materiais abrasivos. Polímeros elastoméricos, como a borracha de silicone, possuem excelente elasticidade e dureza, tornando-os ideais para vedação e isolamento, Lake e Thomas (1967). Com base em pesquisas científicas contínuas, as propriedades mecânicas dos polímeros estão em constante evolução. A compreensão aprofundada dessas propriedades é fundamental para a inovação em materiais poliméricos, impactando uma variedade de setores industriais.

Os polímeros utilizados em impressoras 3D, são chamados de filamento.

Segundo (BAGLIOTTI E GASPAROTTO, 2017), o filamento, insumo básico utilizado na impressão 3D, é um material termoplástico capaz de criar formas complexas quando fundido. Os materiais estão disponíveis no mercado nos diâmetros de 1,75mm e 3,00mm, em inúmeras cores e tipos, conforme mostra a Figura 10 (BAGLIOTTI E GASPAROTTO, 2017).

Figura 10 - Tipos de filamentos para impressão 3D.



Fonte: ABSPLASTIC (2016)

Segundo (Bagliotti e Gasparotto, 2017), os principais filamentos são o ácido polilático (PLA), um material termoplástico para impressoras 3D com custo

comparativamente baixo, versatilidade e algumas limitações devido às suas baixas propriedades mecânicas, é derivado do amido de milho, o que torna biodegradável e o filamento de acrilonitrila butadieno estireno (ABS) é um material termoplástico para impressoras 3D feito de um plástico durável que pode ser usado para fabricar peças com características funcionais e é derivado do petróleo.

Um dos filamentos utilizados nesse trabalho será o PLA HT Impact preto que tem nucleadores modificadores térmicos e modificador de impactos e o PLA amarelo da 3Dlab sem modificadores, para efeito de comparação, verificando-se a influência da impressora construída nesse trabalho irá influenciar nas características desses 2 materiais.

No quadro 1 pode ser observado algumas características do PLA.

Quadro 1 - Propriedades do poliácido láctico (PLA ou ácido poliláctico)

Tipo de polimerização	Policondensação
Tipo de cadeia	Linear
Densidade g/cm ³	1,24
Cristalinidade %	até 40
T _m , °C	130-180
T _g , °C	50-80
Índice de fluidez (190°C), g/10min	10

Fonte: Wikipédia

Nas Figuras 11 e 12 segue algumas comparações entre o ABS e os PLAs do fabricante 3D prime.

Figura 11 - Comparativo resistência térmicas entre ABS, PLA, PLA HT Impact



Fonte: 3D prime (2023)

Figura 12 - Comparativo resistência ao impacto ABS, PLA, PLA HT Impact



Fonte: 3D prime (2023)

2.6 Construção da impressora

Para a construção da impressora dividiu-se em três partes, sendo elas a parte eletrônica, mecânica e o os softwares de controle.

Será descrito no referencial teórico somente os componentes principais da eletrônica de controle da impressora e as demais etapas como mecânica e software será descrito no desenvolvimento.

Para o funcionamento da impressora é necessário que seja controlada eletronicamente, pois é necessário ter o correto controle de temperatura e o correto movimento para manter as coordenadas X, Y e Z.

A principal função da impressora é fundir um fio na peça a ser construída, para isso será necessário extrudar um fio plástico (abs, pla etc..) para adicionar esse material a peça a ser construída, aquecer o fio para que o material seja aplicado em um formato pastoso (líquido) para que ocorra a aderência na peça. Mas para que isso aconteça é necessário um controle eletrônico. Esses componentes serão descritos abaixo.

2.6.1 Microcontrolador Arduino

O Arduino, uma plataforma de prototipagem eletrônica de código aberto, encontrou um caminho na impressão 3D. Ele pode controlar motores de passo, sensores de temperatura, extrusores e outras partes cruciais da impressora, coordenando seu funcionamento para criar objetos precisos e complexos. Sua acessibilidade e flexibilidade o tornam uma escolha popular tanto para entusiastas quanto para fabricantes profissionais (Vig, Vashishtha, & Maheshwari, 2019).

Além de controlar os aspectos mecânicos da impressora 3D, o Arduino permite a implementação de funcionalidades avançadas, como auto nivelamento e monitoramento remoto. A comunidade acadêmica tem explorado essas possibilidades, criando algoritmos inteligentes que melhoram a eficiência e a precisão da impressão. Além disso, o Arduino é frequentemente utilizado em pesquisas interdisciplinares, impulsionando a inovação na interface entre eletrônica e impressão 3D (Krishna, Kumar, & Narayana, 2020).

O Arduino foi criado em 2005 com o objetivo de criar uma espécie de computador que fosse barato, funcional, prático e simples de se programar, isso para que fosse acessível para estudantes e projetistas amadores. Com esse objetivo em mente seus criadores resolveram aderir ao conceito de hardware livre, ou seja, qualquer um pode montar, modificar, melhorar e adaptar a placa partindo da mesma base.

A partir desse conceito foi criada uma placa que possui um funcionamento praticamente igual ao de um computador de pequeno porte, com um microcontrolador Atmel, circuitos I/O que podem facilmente ser conectados a um outro computador e programados pela IDE (em linguagem C/C++), tudo isso apenas com um cabo USB.

Ao longo dos anos, vários modelos com tamanhos, especificações e preços (que variam de acordo com a sua capacidade de armazenamento, quantidade de entradas e saídas) diferentes de Arduino foram sendo desenvolvidos e apresentados ao mercado.

Para esse trabalho foi utilizado o microcontrolador Arduino Mega de forma a rodar as configurações e fazer toda a interface entre os outros componentes e shields da máquina.

A placa Arduino Mega, que foi utilizada no projeto, pode ser alimentada pelo mesmo cabo que realiza sua comunicação com o computador, pelo cabo USB, mas conta também com a opção de ser alimentada por uma fonte externa, conforme a Figura 13.

Figura 13 - Placa Arduino



Fonte: github- guia-maker-da-impresao-3d-tecnologiafff. Adoc (2023).

2.6.2 Placa Shield Ramps

A placa shield RAMPS (RepRap Arduino Mega Pololu Shield) se destaca como um pilar de estabilidade e precisão. Integrando-se harmoniosamente com o Arduino Mega,

essa placa oferece um controle meticuloso sobre motores e sensores, permitindo a criação de objetos 3D com detalhes impressionantes e complexidade surpreendente.

A RAMPS desempenha um papel crucial na impressão 3D, assegurando movimentos suaves dos motores de passo e monitoramento preciso das temperaturas. Ao traduzir comandos digitais em movimentos físicos, ela possibilita a criação de objetos tridimensionais com uma precisão milimétrica (Reprap.org, [s.d.]).

Uma das características notáveis da RAMPS é sua adaptabilidade. Ela oferece um alto grau de personalização, permitindo que os entusiastas da impressão 3D ajustem parâmetros específicos para atender às necessidades exclusivas de seus projetos. Seja ajustando a velocidade dos motores ou controlando a temperatura com precisão, a RAMPS coloca o controle nas mãos dos usuários (Reprap.org, [s.d.]).

A RAMPS é uma parte integrante do movimento RepRap, uma comunidade dedicada à ideia de impressoras 3D de código aberto e autoeplicáveis. Sua presença constante em projetos RepRap atesta sua confiabilidade e eficácia, refletindo a confiança da comunidade em sua capacidade de fornecer resultados consistentes (Reprap.org, s.d.).

Ao proporcionar precisão e controle inigualáveis, a placa shield RAMPS continua a catalisar a inovação na impressão 3D. Em um mundo onde a complexidade dos designs é um padrão, a RAMPS oferece uma base sólida sobre a qual os criadores podem construir suas visões tridimensionais.

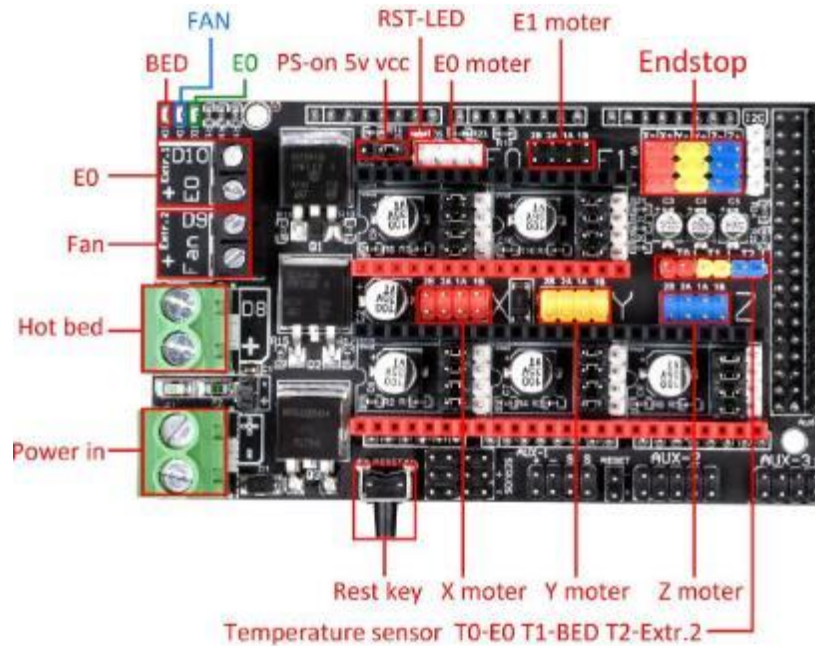
Os Shield são responsáveis por comunicar com o microcontrolador (Arduino) e executar as funções necessárias para os projetos e isso que torna os projetos com microcontroladores mais versáteis, sendo que há vários tipos de shield no mercado e com várias funções. Conforme comentado, há vários shields no mercado e com várias funções podendo ser aplicado ao Arduino conforme o projeto a ser montado.

Nesse trabalho foi utilizado o Shield Ramps 1.6, que é um Shield responsável pelo controle de potência dos servomotores, resistências aquecedoras e termistores.

Há outras versões com 1.2, 1.3, 1.4, etc... a versão que foi utilizada no trabalho é a 1.6, mas que tem a mesma pinagem da 1.4.

A Figura 14 mostra a placa e a sua pinagem.

Figura 14 - Placa Ramps 1.6



Fonte: github- guia-maker-da-impressao-3d-tecnologiaiff. Adoc (2023).

Como pode-se observar na Figura acima, a placa (Shield) ramps faz a interface entre o Arduino, fim de curso e potência da impressora 3D. A alimentação da placa também faz a alimentação do Arduino não sendo necessário outra fonte de tensão.

Pode-se citar algumas características da placa:

- Suporta 5 step drivers;
- Seus mosfets podem ser controlados por PWM;
- LED indicador de status do aquecedor;
- Controla 2 motores de passo para eixo Z em paralelo;
- Alimentação de 12V;

Para acionar os motores de passo é necessário estar acoplado a essa placa os drives dos motores, que será descrito posteriormente.

Mais detalhes da placa ramps encontra-se no anexo D.

2.6.3 Drive dos motores de passo

No cerne da revolução da impressão 3D, os drives para motor de passo desempenham um papel central na garantia de precisão e qualidade nas criações tridimensionais. Estes dispositivos eletrônicos, que traduzem comandos digitais em

movimentos físicos milimetricamente calculados, são fundamentais para o sucesso das impressoras 3D modernas. Esta introdução explora o mundo dinâmico dos drives para motor de passo na impressão 3D, destacando sua importância e inovação contínua, apoiadas por pesquisas acadêmicas fundamentais.

Os drives para motor de passo são a espinha dorsal dos movimentos precisos na impressão 3D. Controlam meticulosamente os motores de passo, assegurando que cada camada do objeto seja depositada com precisão e detalhes impressionantes. Essa precisão é essencial para criar desde protótipos complexos até peças finais altamente detalhadas (Brown, 2017).

A diversidade de drives para motor de passo oferece uma gama de opções, desde drives de alta velocidade até aqueles otimizados para fornecer torque robusto. A personalização dos parâmetros, como corrente e micro passos, é essencial para adaptar o processo de impressão a diferentes materiais e requisitos de detalhamento (Ahmed, Ibrahim, & Fahmy, 2018).

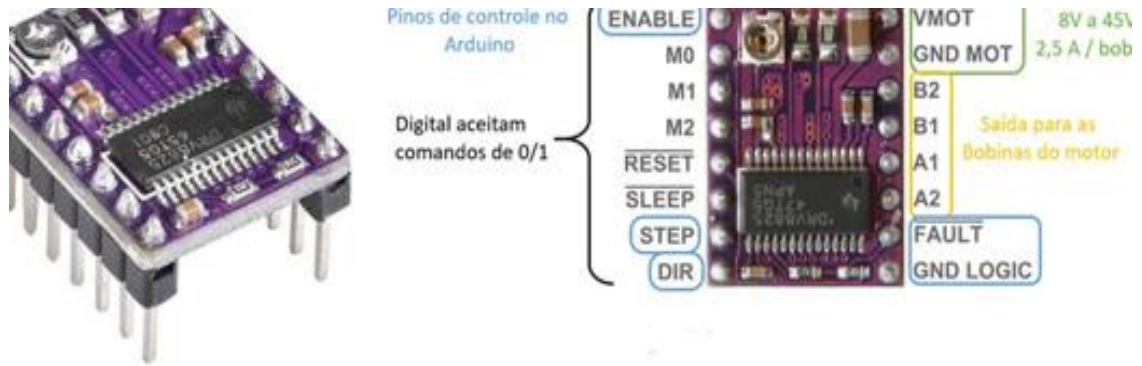
As inovações recentes nos drives para motor de passo têm sido impulsionadas por pesquisas acadêmicas rigorosas. Sensores integrados, algoritmos avançados e técnicas de controle otimizadas são resultados diretos de estudos científicos, garantindo que os motores funcionem de maneira mais eficiente e precisa durante o processo de impressão (Smith, Johnson, & Anderson, 2019).

Ao entender o papel central desses dispositivos e as contribuições significativas dos estudos científicos, os entusiastas e profissionais podem explorar todo o potencial da impressão 3D, criando objetos complexos e extremamente detalhados com confiança.

Para que seja necessário o acionamento dos motores de passos, motores dos eixos X, Y e Z, é necessário um drive. Esse drive é responsável por enviar pulsos de forma a chavear e fazer o motor de passo funcionar, girando conforme os Steps do motor.

O drive utilizado no projeto é o drive DRV8825 e o mesmo é acoplado na placa ramps de onde vem os comandos de rotação. O drive DRV8825 e sua pinagem pode ser visto na Figura 15.

Figura 15 - Driver DVR8825



Fonte: Crescer Engenharia (2023)

Segue alguns dados do drive:

- Tensão Lógica recomendada: 2,8V a 5,1V
- Tensão para motores: 8,2V a 45V
- Corrente de saída: 2,5A máximo, ajustável (Acima de 1,5A utilizar dissipador. Acompanha o produto)
- Proteção térmica interna $>160^{\circ}\text{C}$
- Proteção contra sobrecorrente $>3\text{A}$
- Proteção contra subtensão $< 8,2\text{V}$
- Frequência máxima de pulsos 250 kHz
- Duração mínima de pulso $1,9\mu\text{s}$

Mais dados encontra-se no anexo F.

Para esse projeto foi escolhido esse drive devido a corrente máxima fornecida por ele (2,5 A), sendo que para o motor de passo utilizado pode consumir uma corrente máxima de 1,7 A.

2.6.4 Motores de passo

Na engenharia intrincada da impressão 3D, os motores de passo emergem como os artífices silenciosos que transformam códigos em objetos tangíveis. Estes dispositivos elétricos, fundamentais para os movimentos precisos em impressoras 3D, são a espinha

dorsal da precisão cinemática. Esta introdução explora o papel essencial dos motores de passo na impressão 3D, destacando sua importância crucial e os avanços sustentados por pesquisas acadêmicas inovadoras.

Os motores de passo são a força motriz por trás dos movimentos precisos em todas as direções nas impressoras 3D. Controlando a rotação de cada eixo milimetricamente, esses motores asseguram que cada camada do objeto seja depositada com precisão, resultando em detalhes impressionantes e contornos nítidos. Sua capacidade de dividir o movimento em passos discretos é fundamental para a criação de objetos complexos com exatidão (Zhang, Zhao, Liu, & Wang, 2018).

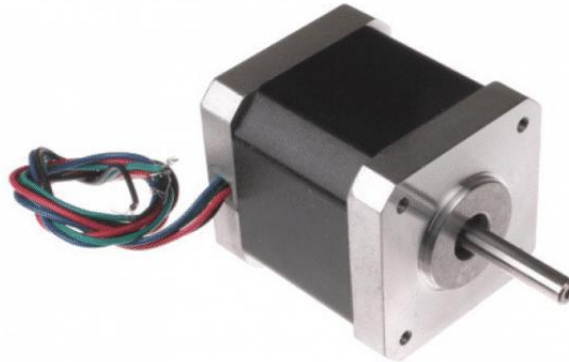
Os motores de passo são oferecidos em várias categorias, desde motores unipolares até bipolares, cada um com suas próprias vantagens. Além disso, sua capacidade de fornecer torque consistente em baixas velocidades torna-os ideais para movimentos controlados e detalhados, essenciais para a impressão 3D (Ahmed, Zafar, & Iqbal, 2019).

Estudos recentes exploram novos materiais e geometrias para otimizar o desempenho, garantindo que esses motores atendam às crescentes demandas da impressão 3D de alta precisão (Chen, Xu, Wu, & Zhu, 2020).

O motor de passo ou stepper motors é um tipo especial de motor que converte pulsos elétricos em movimento angular discreto. Ao contrário dos motores convencionais, que giram continuamente, os motores de passo se movem em incrementos precisos, conhecidos como "passos", daí o nome.

Conforme mencionado (Cao e Schwartz, 1999), os motores de passo são atuadores eletromecânicos incrementais não lineares, permitem controle preciso de velocidade e posição e seu baixo custo as torna a escolha ideal para muitas aplicações. Esses motores são aplicados em utilizações de controle e metrologia e podem ser encontrados em impressoras jato de tinta, bombas de deslocamento positivo e máquinas de controle numérico computadorizado (CNC), entre outras utilizações. O motor de passo pode ser visto na Figura 16.

Figura 16 - Motor de passo



Fonte: Saravana Electronics- SE (2018)

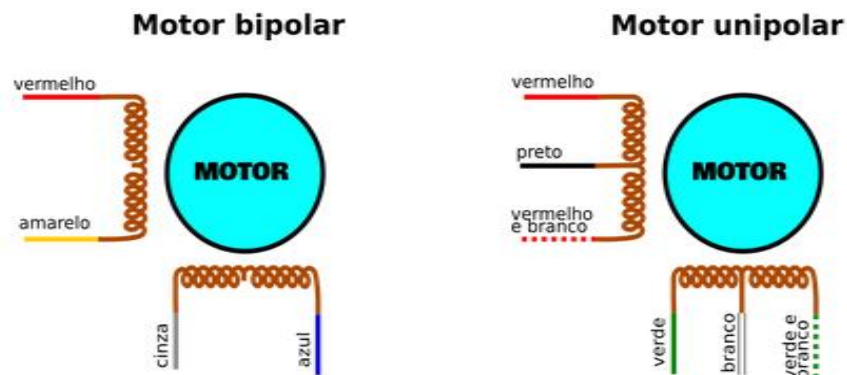
O funcionamento básico de um motor de passo envolve a interação entre o rotor (parte móvel) e o estator (parte fixa). O rotor é composto por um conjunto de ímãs permanentes, enquanto o estator possui bobinas eletromagnéticas. As bobinas do estator são organizadas em um padrão específico, geralmente em duas ou quatro bobinas, dependendo do tipo de motor de passo.

Quando uma corrente elétrica é aplicada a uma bobina específica do estator, ela cria um campo magnético temporário. Esse campo magnético interage com os ímãs do rotor, atraindo ou repelindo-os, dependendo da polaridade. Isso causa o movimento angular do rotor em direção à próxima posição magnética estável.

Para controlar com precisão o movimento do motor de passo, os pulsos elétricos são fornecidos às bobinas do estator em sequência. Esses pulsos são gerados por um controlador, como um microcontrolador, que determina a velocidade e a direção do movimento do motor. A sequência dos pulsos elétricos determina o número de passos e a velocidade de rotação do motor.

Existem diferentes tipos de motores de passo, como motores de passo unipolares e bipolares, que possuem características e circuitos de acionamento distintos. Os motores de passo unipolares possuem bobinas com um ponto central em comum, enquanto os bipolares têm bobinas separadas, conforme Figura 17. Isso afeta o circuito de controle necessário para alimentar os motores.

Figura 17 - Tipos de bobinas motor de passo



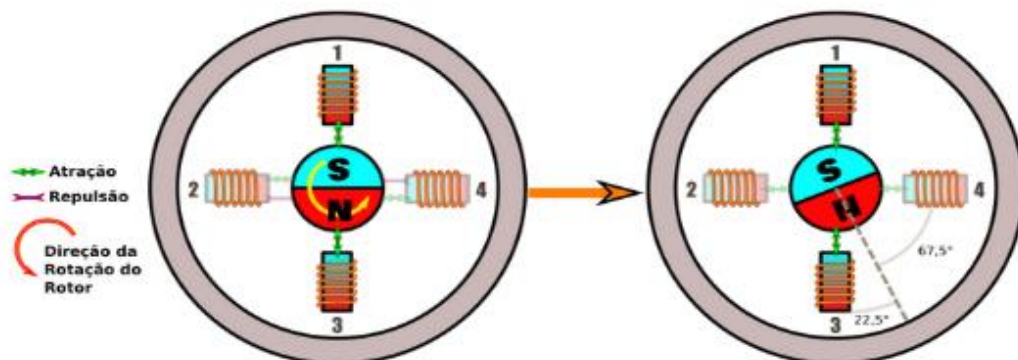
Fonte: <https://github.com/Patola/ebook/blob/master/guia-maker-da-impressao-3d-tecnologiafff.adoc>

Um dos principais benefícios dos motores de passo é a sua precisão de posicionamento. Eles podem ser controlados de maneira muito precisa, permitindo a realização de movimentos repetíveis e previsíveis. Isso os torna ideais para aplicações que requerem controle preciso de movimento, como em robótica, máquinas CNC (Controle Numérico Computadorizado), impressoras 3D e equipamentos de automação industrial.

Em resumo, o motor de passo é um tipo especial de motor que se move em incrementos discretos chamados de passos ou micropassos, devido à interação entre as bobinas eletromagnéticas do estator e os ímãs do rotor. Através do controle de pulsos elétricos, é possível controlar com precisão o movimento angular do motor, permitindo aplicações que exigem posicionamento preciso e repetível.

Na Figura 18 mostra o motor a $\frac{1}{4}$ de passos.

Figura 18 - Micropassos de um motor de passo



Fonte: github- guia-maker-da-impressao-3d-tecnologiafff. Adoc (2023).

O motor escolhido para esse projeto foi o motor de passo nema 17HS4401, devido ao seu torque e a suas dimensões. Segue os dados do motor de passo:

- Ângulo do Passo (°): 1.8
- Corrente (A): 1.7 A
- Resistência da Fase: (ohm): 1.5
- Holding Torque (g.cm): 4200
- Qtde Fios (NO.): 4
- Peso do Motor: 255g

Mais dados encontra-se no anexo G.

2.6.5 Extrusor e hotend

Extrusores, são mecanismos responsáveis por alimentar filamentos na impressora, e hotends, é onde a fusão térmica acontece, formam o cerne da impressão 3D. O hotend é onde o filamento é aquecido e depositado em camadas precisas, enquanto o extrusor controla o fluxo do material. A integração perfeita dessas partes é essencial para a precisão e qualidade das impressões (Kreiger, Mulvaney, & Pearce, 2017).

Extrusores vêm em várias formas, incluindo modelos de acionamento direto e Bowden, cada um projetado para atender a diferentes requisitos de impressão. Os hotends são construídos com materiais de alta condutividade térmica, e seu design otimizado facilita a fusão uniforme do material, independentemente da complexidade do objeto (Husseini, Anwer, & El-Sayed, 2018).

O extrusor é uma parte fundamental de uma impressora 3D que desempenha um papel essencial no processo de fabricação aditiva. Também conhecido como cabeça de impressão, o extrusor é responsável por depositar o material de impressão em camadas sucessivas, formando o objeto desejado.

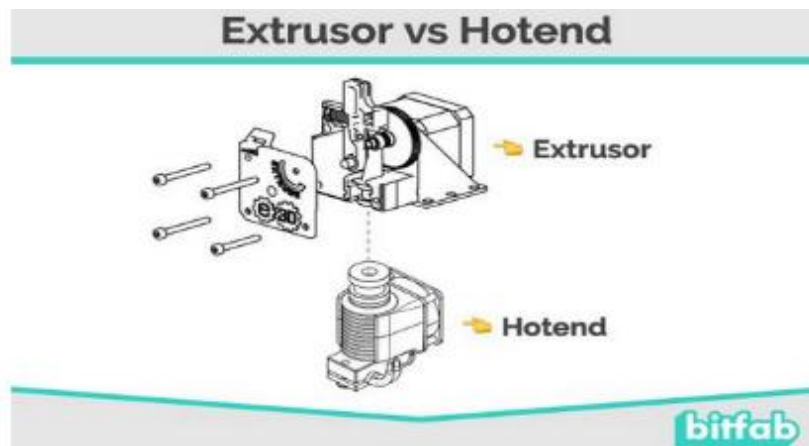
Vale ressaltar que a cabeça de impressão é composta de extrusor e hotend e são dois componentes distintos, apesar de falarmos como extrusor todo o conjunto.

O sistema de extrusão tem 2 componentes principais. É aqui onde podemos fazer a distinção entre extrusora e hotend.

- **O extrusor propriamente dito**, também chamado de *feeder drive* é o conjunto que empurra o filamento, normalmente está formado por um motor, roda dentada, um rolamento para pressionar o filamento contra a roda etc.
- **O fusor ou hotend** é o componente da impressora onde o material é derretido e se expulsa através da ponta para depositá-lo, e geralmente tem partes como o *nozzle* ou bico, o dissipador, o bloco aquecedor etc.

Na Figura 19 pode ser visto os 2 componentes.

Figura 19 - Extrusor vs Hotend



Fonte: <https://bitfab.io/pt-pt/blog/extrusor-impressora-3d/> (2023)

Conforme comentado anteriormente, chamamos de extrusor todo o conjunto, mas será descrito todos os seus componentes.

O extrusor consiste em vários componentes principais:

- **Hotend (bico aquecido):** É a parte do extrusor que aquece e derrete o material de impressão, geralmente filamento de plástico. O hotend é composto por um elemento aquecedor, um termistor para medir a temperatura e um bico por onde o material derretido é depositado.
- **Motor de extrusão:** O motor de extrusão é responsável por empurrar o filamento de material através do extrusor. Ele puxa o filamento da bobina e o alimenta para o hotend. O motor é controlado eletronicamente para ajustar a quantidade de material que é extrudado.
- **Engrenagens de alimentação:** As engrenagens de alimentação são conectadas ao motor de extrusão e têm a função de tracionar e guiar o filamento

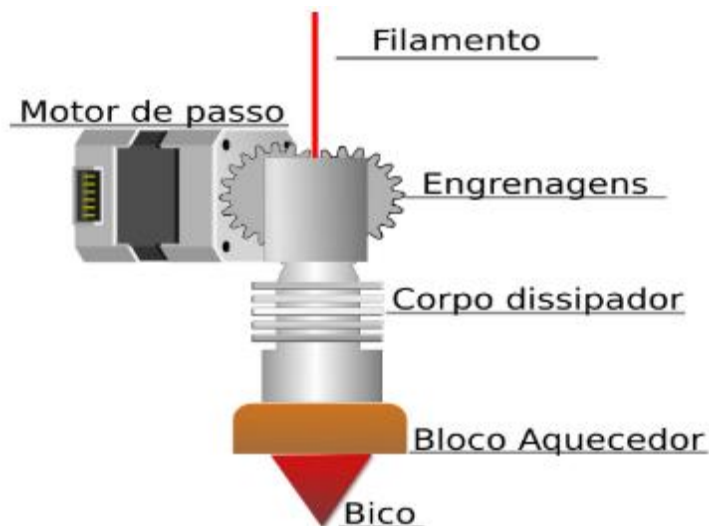
até o hotend. Elas garantem que o material seja alimentado de forma consistente e precisa.

- Sistema de resfriamento: Para evitar o superaquecimento do material de impressão, o extrusor possui um sistema de resfriamento. Geralmente, isso inclui um ventilador que direciona o fluxo de ar para resfriar o filamento logo após ele ser depositado.

- Mecanismo de ajuste de altura: O extrusor também pode ter um mecanismo de ajuste de altura que permite controlar a distância entre o bico do hotend e a plataforma de impressão. Isso é importante para garantir uma aderência adequada da primeira camada do objeto.

Na Figura 20 pode ser verificado um extrusor com hotend.

Figura 20 - Componentes da cabeça de impressão



Fonte: Maker-hero (2023).

Durante o processo de impressão 3D, o extrusor recebe comandos do software de controle, que determina o caminho e a quantidade de material que deve ser extrudado em cada camada. O filamento é puxado pelo motor de extrusão, aquecido no hotend até atingir o estado de fusão e então depositado em camadas sucessivas, formando o objeto desejado.

A qualidade e o desempenho do extrusor são cruciais para uma impressão 3D bem-sucedida. Um extrusor bem projetado e calibrado adequadamente garante a deposição precisa do material, evitando problemas como entupimentos, sub-extrusão ou super-extrusão. Por isso, é importante escolher um extrusor de qualidade e realizar a manutenção regularmente para garantir resultados consistentes e de alta qualidade em suas impressões 3D.

O extrusor é a parte mais complexa e sensível para o trabalho. Pois é ele que fará a deposição do material e sem ele todos os outros componentes, microcontrolador, drives e motores para executar os movimentos de nada serviria.

É muito importante a escolha de um extrusor, mas para isso é de suma importância saber qual o tipo de material que se pretende utilizar, por isso seria necessário além de outros pontos conhecer a temperatura do material a se trabalhar.

2.6.5.1 Tipos de sistemas de extrusão para impressoras 3D

Podem ser encontrados no mercado diferentes tipos de extrusoras podendo ser do tipo Bowden: quando o hotend e a extrusora estão separados e conectados por um tubo de PTFE ou do tipo direct: quando o hotend e a extrusora estão integrados na mesma cabeça, mais detalhes serão descritos abaixo.

2.6.5.1.1 Extrusão tipo Bowden

Nas impressoras 3D do tipo Bowden, a extrusora é montada separadamente, podendo estar localizada na parte superior da impressora. Esta extrusora possui um motor de passo que move o filamento através de um tubo de PTFE (com um diâmetro ligeiramente maior do que o filamento) para guiá-lo até o hotend.

Como resultado, há uma distância maior percorrida pelo filamento, o que pode tornar a impressão mais difícil.

No sistema Bowden, a cabeça de impressão (hotend) é consideravelmente mais leve, o que resulta em movimentos mais suaves e permite uma impressão mais rápida. No entanto, é importante ter cautela ao imprimir em alta velocidade, pois a peça final

pode perder precisão devido à distância da extrusora, que pode causar tensões na saída do hotend, afetando a qualidade do resultado.

Este sistema requer uma retração mais longa e rápida para evitar tensões.

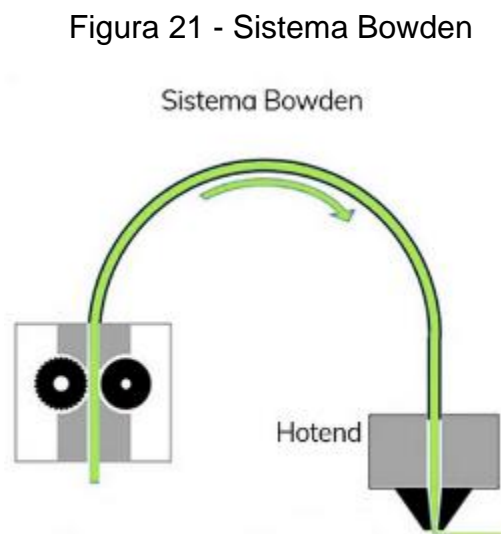
Sobre valores padrões utilizados seria:

- Velocidade de retração: 45mm / s
- Distância de redução: 3.2 - 3.8

Algumas características da extrusora do tipo Bowden:

- Cabeça mais leve
- Possibilita movimentos mais rápidos da cabeça
- Permite retratações mais longas
- Pode levar a problemas de stringing
- Requer um tubo Bowden de alta qualidade

Na Figura 21 pode ser visto o sistema tipo bowden.



2.6.5.1.2 Extrusão direta para impressora 3D

Com a extrusora localizada junto ao hotend, há uma menor pressão ao empurrar o filamento para o bocal da impressora 3D, o que reduz os problemas de gotejamento de fios e filamentos. Devido à curta distância entre a extrusora e o bico (hotend), o filamento

pode ser alimentado rapidamente, resultando em retratações mais ágeis. Além disso, não é necessário um motor tão potente para impulsionar o filamento.

No entanto, é importante notar que o calor gerado pelo motor da extrusora, devido à sua proximidade com o hotend, pode afetá-lo, levando ao superaquecimento da zona quente e impactando a qualidade das impressões.

Sobre valores padrão para retrações de extrusão direta seriam:

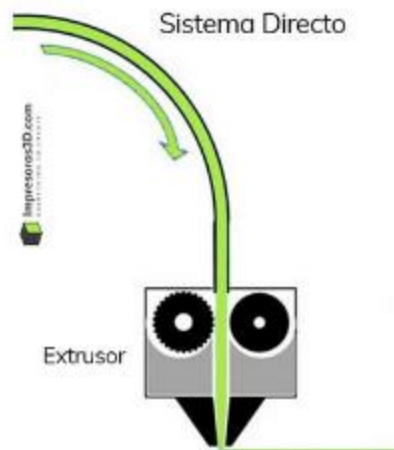
- Velocidade de retração: 35mm / s
- Distância de redução: 0.8 - 1.2

Outro ponto forte da extrusão direta é a maior facilidade para imprimir filamentos flexíveis, conforme Figura 22.

Algumas características da extrusora direta:

- A extrusora (o sistema push) está integrada ao fusor.
- Exige menor pressão durante o avanço do filamento.
- Apresenta menor problema relacionados a fios.
- Necessita de uma velocidade de retração mais lenta.
- É mais adequada para imprimir filamentos flexíveis.

Figura 22 - Sistema Direct Drive



Fonte: Impresoras3D.com

2.6.5.2 Bico do hotend

O bico é a parte onde o hotend termina, e você provavelmente está se perguntando sobre o material desse componente, que geralmente tem uma tonalidade amarelada, quase dourada, conhecida como latão, que é uma liga metálica não ferromagnética composta por cobre e zinco, destacando-se por seu baixo coeficiente de atrito, alta condutividade térmica e grande durabilidade. Algumas variações de latão incluem a adição de chumbo (aproximadamente 2%), o que levanta preocupações sobre sua utilização para contato com alimentos.

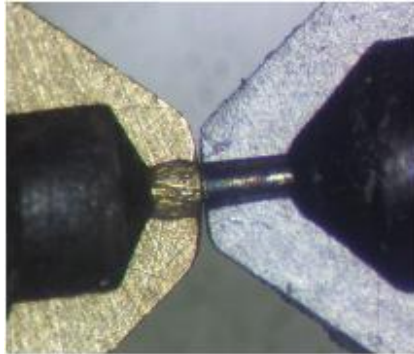
Os bicos de impressoras 3D nem sempre são feitos de latão, especialmente com a introdução de novos materiais abrasivos, como ABS com fibra de carbono, que estão sendo cada vez mais usados em impressoras 3D convencionais. Embora não haja um padrão obrigatório para as dimensões dos bicos, a maioria dos bicos "genéricos" utiliza uma rosca M6 (6mm de diâmetro) para se conectar ao bloco aquecedor de alumínio e seguem certas medidas.

A medida de saída do orifício de um bico varia pouco. As impressoras 3D do tipo Makerbot geralmente possuem bicos com 0,35mm de diâmetro, enquanto as repraps têm bicos de 0,4mm de diâmetro.

O diâmetro do orifício é um equilíbrio entre a pressão interna necessária para extrudar o filamento e o diâmetro máximo desejável do filete de material, ou seja, reduzir o orifício para 0,2mm dobra a pressão interna necessária e aumenta a chance de entupimentos, enquanto aumentar para 0,8mm reduz pela metade a pressão, mas também exige um bloco e cartucho aquecedores com maior capacidade de energia térmica e maior potência.

Na Figura 23 pode ser visto o bico.

Figura 23 - Vista cortada do bico



Fonte: Github. guia-maker-da-impresao-3d-tecnologiaff. adoc (2023)

Nesse trabalho optou-se por utilizar a extrusora direct drive, conforme mostra a Figura 24.

Figura 24 - Extrusora Direct Drive



Fonte: Autor (2023)

2.6.6 Sensor fim de curso – Endstop

Os sensores endstop são dispositivos de detecção cruciais em impressoras 3D. Eles determinam os pontos exatos onde o movimento deve começar ou parar assegurando que cada camada do objeto seja depositada no lugar certo. Sua capacidade

de detectar e indicar as posições com precisão milimétrica é fundamental para a qualidade da impressão (Rodríguez, Cabanes, & Díaz, 2018).

Esses sensores vêm em várias formas, desde sensores mecânicos simples até os mais avançados, como os ópticos. Eles são estrategicamente colocados em diferentes partes da impressora 3D para marcar pontos cruciais no movimento dos eixos X, Y e Z.

Os sensores de fim de curso, também conhecidos como Endstops, são basicamente interruptores comutadores acionados por uma força mínima, que agem sobre o circuito correspondente. Estes dispositivos têm uma vida útil bastante longa, podendo chegar a 1 milhão de ciclos, dependendo da aplicação, especialmente em situações de uso pesado. Quando a haste do sensor é acionada, os terminais do sensor fecham o circuito. Isso faz com que o sensor perceba que foi acionado e, em seguida, envie um sinal elétrico para o motor, indicando que o mesmo deve parar ou inverter a rotação. As chaves de fim de curso têm a capacidade de determinar a presença ou ausência de um objeto, indicar a passagem, o posicionamento e o término do curso do objeto em questão. É por isso que esses dispositivos são conhecidos como "chaves de fim de curso".

Essas chaves de fim de curso utilizadas em impressora 3D ou CNC, podem determinar o limite máximo que os motores podem atingir sem danificar ou forçar motores e peças da impressora 3D ou CNC.

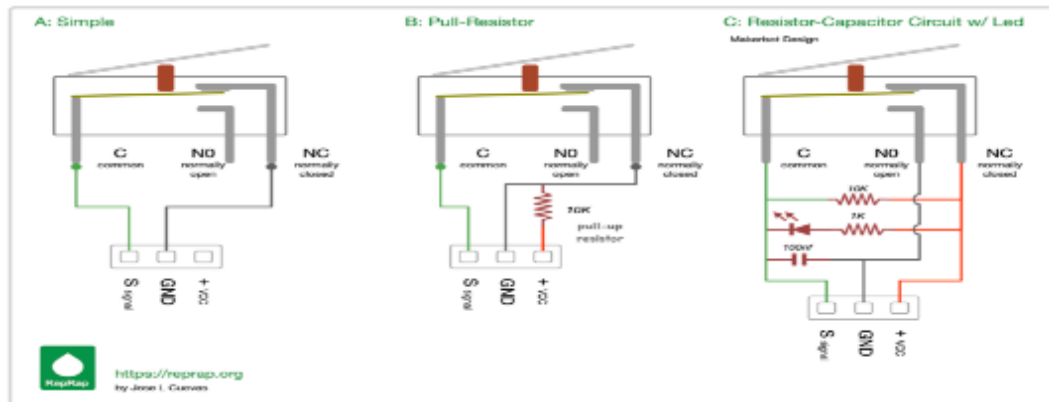
Geralmente, em impressoras 3D e CNCs, são empregados 3 ou 4 sensores de fim de curso para determinar o ponto zero da máquina (Home). Estes sensores são utilizados para marcar o ponto zero dos eixos x, y e z, bem como para detectar o fim do filamento. As Figuras 25 e 26 mostram o sensor e o seu diagrama elétrico.

Figura 25 - Sensor endstop mecânico



Fonte: <https://reprap.org> (2023)

Figura 26 - Diagrama elétrico do sensor Endstop mecânico



Fonte: <https://reprap.org> (2023)

Nesse trabalho os Endstops (fim de curso) foram utilizados para verificar a posição inicial dos eixos X, Y e Z. Esses interruptores são montados nos limites de movimento da plataforma de impressão ou do carro do extrusor e ajudam a garantir que a impressora 3D esteja corretamente calibrada e que os movimentos ocorram dentro dos limites desejados.

Existem dois tipos principais de endstops usados em impressoras 3D: endstops mecânicos e endstops ópticos.

- **Endstops mecânicos:** Esses endstops consistem em interruptores mecânicos que são ativados quando ocorre um movimento físico. Eles geralmente são compostos por um botão ou alavanca que é pressionado quando a plataforma de impressão ou o carro do extrusor se move até o limite. O pressionamento do interruptor envia um sinal para o sistema de controle da impressora 3D, indicando que o limite foi alcançado.
- **Endstops ópticos:** Esses endstops utilizam feixes de luz para detectar a posição dos eixos. Eles são compostos por um emissor de luz, como um LED, e um receptor de luz. Quando o feixe de luz é interrompido, por exemplo, pelo movimento da plataforma de impressão ou do carro do extrusor, o receptor detecta a interrupção e envia um sinal para o sistema de controle.

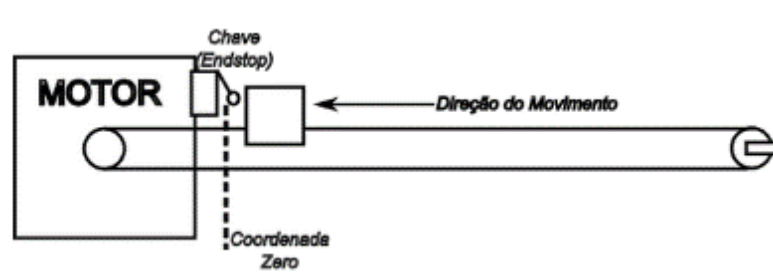
Os endstops são conectados ao sistema de controle da impressora 3D, que pode ser um microcontrolador ou um controlador específico. Esses dispositivos de controle interpretam os sinais dos endstops e utilizam essa informação para determinar a posição inicial e final dos eixos.

Os endstops são usados em vários estágios do processo de impressão 3D. Durante a inicialização da impressora, os endstops são ativados para definir a posição inicial de referência dos eixos. Durante a impressão, os endstops podem ser usados para verificar se a plataforma de impressão está nivelada corretamente, garantindo uma aderência adequada da primeira camada. Além disso, se ocorrer um desalinhamento ou falha no movimento dos eixos durante a impressão, os endstops podem interromper o processo e evitar danos à impressora.

Os endstops são componentes importantes para garantir a precisão e a segurança em uma impressora 3D. Eles desempenham um papel fundamental na calibração inicial, no controle de movimento e no monitoramento dos limites de movimento. Ao fornecer informações precisas sobre a posição dos eixos, os endstops ajudam a garantir uma impressão 3D precisa e confiável.

Na Figura 27 pode ser visto a sua atuação como posição zero.

Figura 27 - Coordenada Home sensor endstop



Fonte: Github, guia-maker-da-impressao-3d-tecnologiafff. Adoc (2023)

2.6.7 Sensores de temperatura – Termistor

No processo de impressão 3D, a temperatura é uma variável crítica, moldando a qualidade e a durabilidade das criações tridimensionais. Os sensores de temperatura emergem como os guardiões termométricos, garantindo que cada camada seja solidificada com precisão.

Os sensores de temperatura são os olhos eletrônicos da impressão 3D, monitorando e regulando as temperaturas com precisão. Em impressoras 3D, esses sensores garantem que o material de impressão seja aquecido na temperatura ideal para uma deposição uniforme e livre de distorções, garantindo a integridade estrutural dos objetos impressos (Smith, Brown, & Johnson, 2018).

Uma variedade de sensores de temperatura está disponível, desde termistores até termopares, cada um projetado para condições específicas de temperatura e ambientes. A capacidade de calibrar e ajustar esses sensores conforme necessário é essencial para garantir uma leitura precisa e, por sua vez, um controle térmico eficaz durante o processo de impressão (Kim, Park, & Lee, 2019).

Novos materiais sensíveis ao calor e técnicas avançadas de medição estão sendo desenvolvidos para melhorar a precisão e a confiabilidade desses sensores. Além disso, algoritmos inteligentes são explorados para interpretar dados térmicos em tempo real, permitindo ajustes dinâmicos e refinados nas configurações de impressão (Li, Wang, & Zhang, 2020).

O sensor utilizado nesse trabalho foi do tipo Termistor. Esse sensor é um componente essencial nas impressoras 3D, especialmente nos hotends dos extrusores, pois desempenha um papel fundamental no controle da temperatura do bico de impressão. Ele é responsável por medir a temperatura do hotend e fornecer feedback para o sistema de controle da impressora. Também é utilizado um termistor na mesa aquecida, para garantir a correta temperatura da mesa onde será depositado o material, mas iremos descrever nesse tópico somente a aplicação no hotend.

O termistor é um tipo de sensor de temperatura que varia sua resistência elétrica de acordo com a temperatura ambiente. No caso das impressoras 3D, um tipo comum de termistor utilizado é o NTC (Negative Temperature Coefficient), que significa "coeficiente de temperatura negativo". Isso significa que a resistência do termistor diminui à medida que a temperatura aumenta.

A forma como o termistor funciona na impressora 3D é relativamente simples. O termistor é colocado no hotend, em contato direto com o bloco de aquecimento ou próximo a ele. À medida que o bloco de aquecimento aquece, a temperatura do termistor aumenta e sua resistência diminui. Essa variação de resistência é medida pelo circuito eletrônico da impressora 3D.

A partir da leitura da resistência do termistor, o controlador da impressora 3D pode calcular a temperatura do hotend com base em uma curva de calibração pré-definida. Essa curva de calibração relaciona a resistência medida pelo termistor à temperatura correspondente. O controlador utiliza essa informação para regular a potência fornecida

ao elemento aquecedor do hotend, ajustando-a para atingir e manter a temperatura desejada.

O controlador da impressora 3D monitora constantemente a leitura do termistor e faz ajustes na potência do aquecedor para garantir que a temperatura do hotend seja mantida dentro de uma faixa precisa. Isso é essencial para obter uma fusão adequada do filamento e garantir uma impressão 3D de alta qualidade.

Ao monitorar a temperatura do hotend, o sistema de controle pode ajustar a potência do elemento aquecedor para manter a temperatura desejada. Isso é especialmente importante para materiais de impressão que requerem temperaturas específicas para uma extrusão adequada, como ABS, PLA, PETG, entre outros. O controle preciso da temperatura é fundamental para evitar problemas como sub-extrusão, falhas de aderência e deformação do objeto impresso.

É importante destacar que a calibração correta do termistor é fundamental para garantir a precisão da leitura de temperatura. Cada termistor pode ter características de resistência específicas, portanto, é necessário configurar corretamente as configurações de temperatura no firmware da impressora 3D para que a leitura do termistor corresponda à temperatura real do hotend.

Em resumo, o termistor na impressora 3D é responsável por medir a temperatura do hotend, fornecendo feedback ao sistema de controle para regular a potência do aquecedor. Isso permite que a impressora mantenha a temperatura desejada durante o processo de impressão 3D, resultando em uma fusão adequada do filamento e em impressões de qualidade.

O termistor utilizado foi o do tipo NTC 100k com cabo que é um sensor de temperatura resistivo com coeficiente de variação de resistência com a temperatura negativa, ou seja, sua resistência diminui com o aumento da temperatura. Possui cabo de isolamento térmica de 1m. Pode ser usado em impressoras 3D RepRap ou mesmo em projetos com microcontroladores. Segue algumas especificações:

- Termistor NTC;
- Resistência: 100K ohms;
- Sensor de temperatura;
- Faixa de detecção: -40° C a 270° C;

- Comprimento do cabo: Aproximadamente 1 m.

2.6.8 Mesa aquecida

As mesas aquecidas são peças vitais do quebra-cabeça, fornecendo uma superfície estável e aquecida para a adesão e a solidificação dos materiais de impressão.

As mesas aquecidas são essenciais para garantir que o material de impressão adira firmemente durante o processo de impressão. Elas criam uma superfície quente e uniforme sobre a qual a primeira camada do objeto é depositada. Isso não apenas promove uma melhor adesão, mas também previne problemas como o empenamento (Le, Tran, & Pham, 2018).

Estas mesas são fabricadas em diferentes materiais, como alumínio e vidro, cada um com propriedades térmicas específicas. Além disso, algumas mesas aquecidas são dotadas de sensores de temperatura embutidos para garantir um controle térmico preciso. A capacidade de ajustar a temperatura conforme o tipo de material de impressão utilizado é crucial para assegurar a qualidade da impressão (Bártolo, Alves, & Mendes, 2019).

Estudos recentes têm se concentrado em técnicas de aquecimento mais eficientes, materiais de mesa com melhor condutividade térmica e algoritmos de controle que garantem uma distribuição de calor uniforme. Estas inovações são fundamentais para manter a estabilidade térmica e, conseqüentemente, a qualidade das impressões 3D (Wang, Zhang, & Li, 2020).

A mesa utilizada nesse trabalho foi a mesa MK2, que nada mais é que uma base com uma resistência em formato de PCB para o aquecimento e com a possibilidade de se colocar um sensor termistor para se efetuar a medição da temperatura e manter o seu controle. A mesa possui as seguintes características:

- Modelo: MK2B;
- Cor: Vermelho;
- Furo: M3;
- Entrada de alimentação: 12V 10A ou 24V 5A;
- Temperatura de trabalho: ~120°C;

- Distância entre o centro dos furos: ~206mm;
- Espessura: 1,6mm;
- Gravura: 35 µm de cobre;
- Laminado: FR4 de 1,6 mm;
- Resistência entre 1.0-1.4 ohms (12V) ou 3-5 ohms (24V);
- Área aquecida (CxL): 200x200mm;
- Dimensões (CxL): 214x214mm;

Pode ser visto na Figura 28 a mesa utilizada.

Figura 28 - Mesa Aquecida MK2B



Fonte: Autor (2023)

3 METODOLOGIA

O estudo tem como suporte documentações sobre o processo de manufatura aditiva, seus componentes e construção e que buscam a responder às questões como funcionamento e aplicações em conceito de indústria 4.0. Para isso é necessário conhecer o processo de funcionamento, construção das diversas tecnologias de manufatura aditiva e uma das necessidades das indústrias 4.0 que é a conexão das impressoras 3D, mesmo que antigas, a uma rede consolidando o conceito de IoT.

Este estudo está dividido em etapas para facilitar o desenvolvimento desse equipamento, como citado abaixo:

1. Pesquisa sobre manufatura aditiva;

Nessa etapa foi pesquisado o funcionamento da manufatura aditiva, suas aplicações em vários meios entre eles o acadêmico e é explicado o funcionamento da manufatura aditiva, os tipos de aplicações e as tecnologias aplicadas para esse processo, como pode-se citar: Extrusão de materiais (FDM/FFF); Sinterização seletiva a laser (SLS); Fotopolimerização (SLA/ DLP); Sinterização seletiva a laser a metais (DMLS/SLM);

2. Funcionamento de uma impressora 3D

Nessa etapa foi discutido o princípio de funcionamento de uma impressora 3D, quais os tipos, funcionamento da parte eletrônica, da parte mecânica e da parte de gestão da impressora como os firmware e os softwares empregados.

Um dos temas discutidos nessa etapa é o termo RepRap que é a abreviação de Replicating Rapid-prototyper, que quer dizer “Replicação Rápidas de Protótipos”. É mantida pelo projeto comunitário Reprap.org (RepRap, 2020), criado por Adrian Bowyer, que busca disponibilizar a impressão 3D e outras tecnologias que podem ser replicadas com baixo custo para o benefício de todos que tem como fundamento a técnica de fabricação de uma impressora 3D auto-replicante capaz de imprimir objetos poliméricos seguindo os princípios do Movimento de Software Livre, sem nenhum custo sob licença de código aberto GNU (*General Public License*).

3. Desenvolvimento, construção e calibração da impressora 3D

Esta etapa foi dividida em sub-etapas, como:

3.1. Construção da parte mecânica

Nessa etapa foi estudado toda fixação da estrutura, como suporte dos motores, fixação dos eixos e carros X, Y e Z e suas limitações.

3.2. Construção parte eletrônica

Nessa etapa foi feita toda a interligação dos componentes eletrônicos, como motores de passo, display, endstops, fontes, resistências de aquecimento, conforme a sua interface e controladores, para garantir o correto funcionamento do equipamento.

3.3. Instalação dos firmwares

Nessa etapa foi discutido a instalação do firmware no Arduino e dos principais parâmetros a ser alterados para o funcionamento conforme a impressora (arquitetura) montada, como posicionamento dos eixos X, Y e Z, a parametrização dos passos de deslocamentos, auto nivelamento, etc.

4. Comissionamento e calibração da impressora

Nessa etapa foram feitos todos os ajustes para o correto funcionamento da impressora 3D, como a verificação da precisão do movimento dos eixos, a taxa de extrusão de forma a garantir a dimensão do material e a qualidade do produto. A calibração da impressora foi feita utilizando modelos de calibração.

5. Instalação do raspberry, testes e validação

Nessa etapa foi feito a instalação e a configuração do raspberry de forma a inserir a impressora no conceito IoT. Para o funcionamento desse conceito foi instalado e configurado o software Octoprint que consiste em uma aplicação para um gerenciador de impressão, podendo funcionar tanto em redes com cabo ou sem fio. Vale ressaltar que a impressora poderá funcionar com ou sem raspberry.

6. Validação mecânica do material impresso

Nessa etapa será impresso 5 corpos de prova de filamento de 2 fabricantes diferentes e verificar a influência da impressora nesses 2 materiais, através do ensaio de tração, seguindo a norma ASTM D – 638 e ensaio de dureza seguindo a norma ASTM D -2240.

4 DESENVOLVIMENTO

Após obter o conhecimento das principais partes da manufatura aditiva, como a definição, tipos de tecnologias aplicadas e o princípio de funcionamento de uma impressora 3D, pode se efetuar o desenvolvimento da impressora.

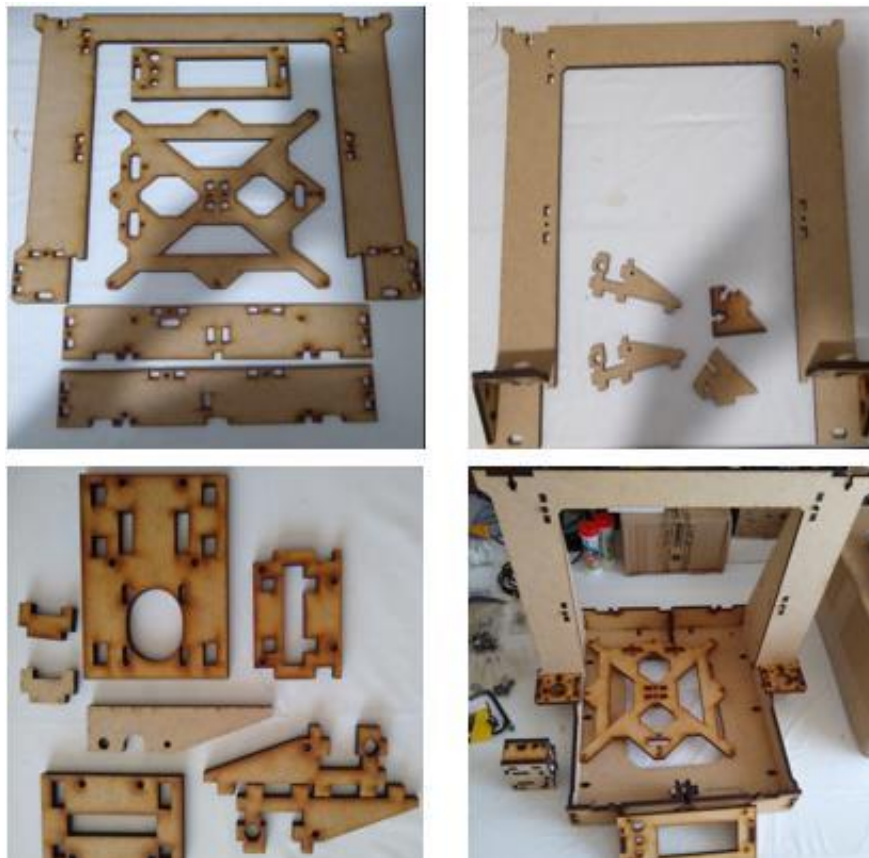
O desenvolvimento iniciou-se pelas seguintes etapas:

4.1 Estrutura mecânica

Também conhecida como frame, optou-se por um frame em MDF já existente no mercado, como intuito de reduzir o custo.

O Frame utilizado foi do sistema Grabber3 em MDF de 3mm cortado a laser. A Figura 29 mostra partes do frame e da montagem final e a Figura 30 mostra o carro X onde é fixada a extrusora e o hotend.

Figura 29 - Frame MDF



Fonte: Autor (2023)

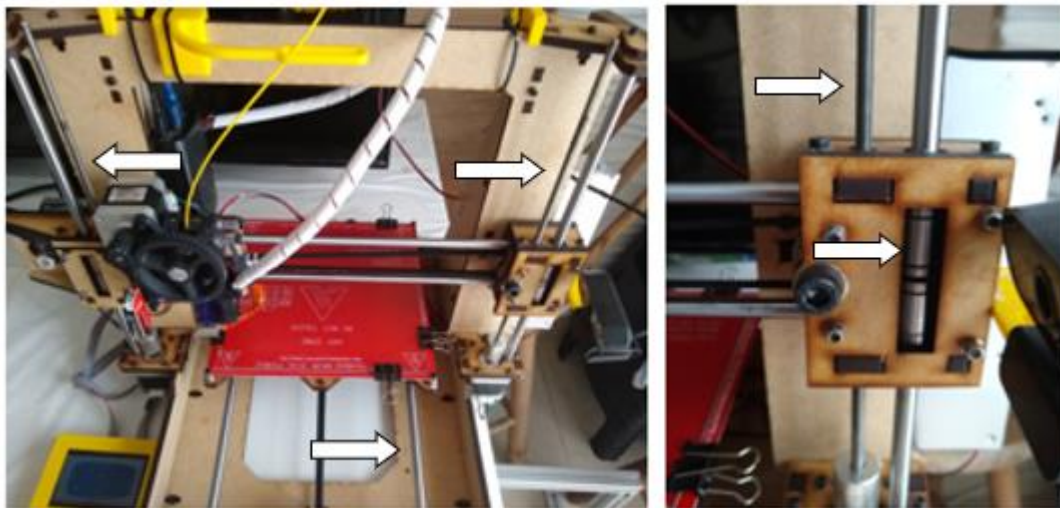
Figura 30 - Carro X com a extrusora



Fonte: Autor (2023)

Além da estrutura de MDF foram utilizados eixos retificados 8mm, rolamentos e barras roscadas M5 para a movimentação dos eixos pode ser visto na Figura 31.

Figura 31 - Eixos retificado, barras roscadas e rolamentos.



Fonte: Autor (2023)

Para a fixação da estrutura foram utilizados parafusos allen com porcas parlock. No anexo A encontra-se o frame de mdf.

4.2 Estrutura Eletrônica

Na montagem eletrônica, foram utilizados todos os componentes responsáveis pelos movimentos e gerenciamento da impressora e será descrito abaixo.

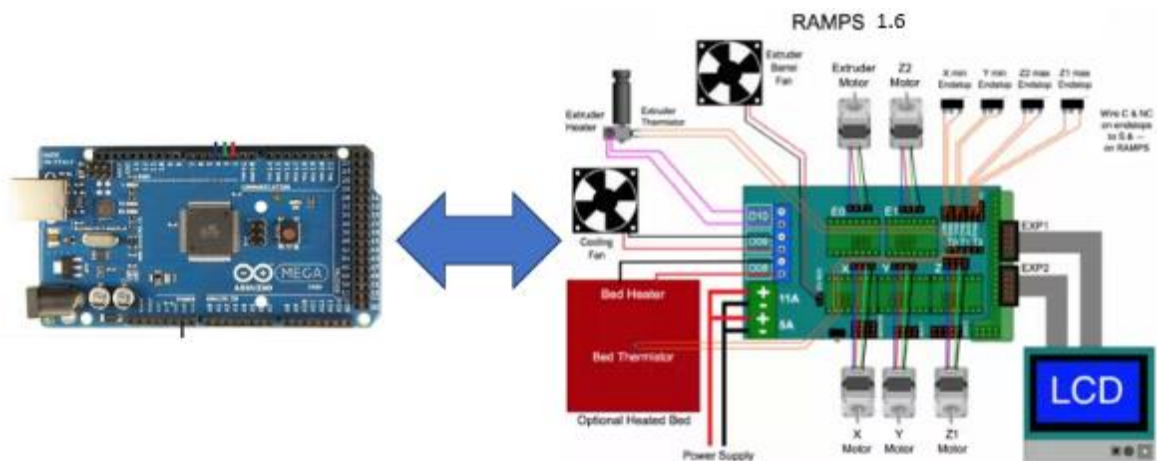
Um dos principais componentes utilizados foi o Arduino e foi dividido em etapas:

4.2.1 Componente de controle e gerenciamento – estrutura eletrônica

O componente utilizado foi o Arduino para efetuar o controle e gerenciamento da impressora a ele foi inserido uma placa de interface (Shield) chamada de Ramps, que tem como função interligar os sensores e atuadores ao microcontrolador Arduino.

Para o funcionamento do Arduino foi instalado o firmware Marlin, que é um firmware responsável por toda o funcionamento e gerenciamento dos sensores e atuadores. Pode ser visto na Figura 32 um o fluxo do Arduino e a interface ramps.

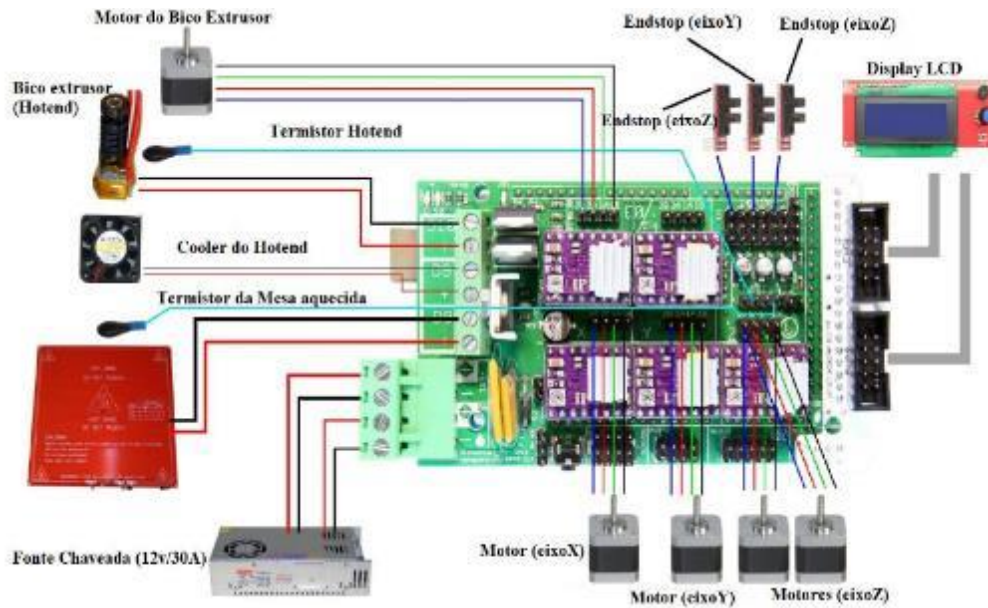
Figura 32 - Fluxo de dados arduino / Ramps



Fonte: Autor (2023)

Para a estruturação das ligações da impressora foi seguido o seguinte diagrama, conforme mostra a Figura 33.

Figura 33 - Diagrama de ligações Ramps



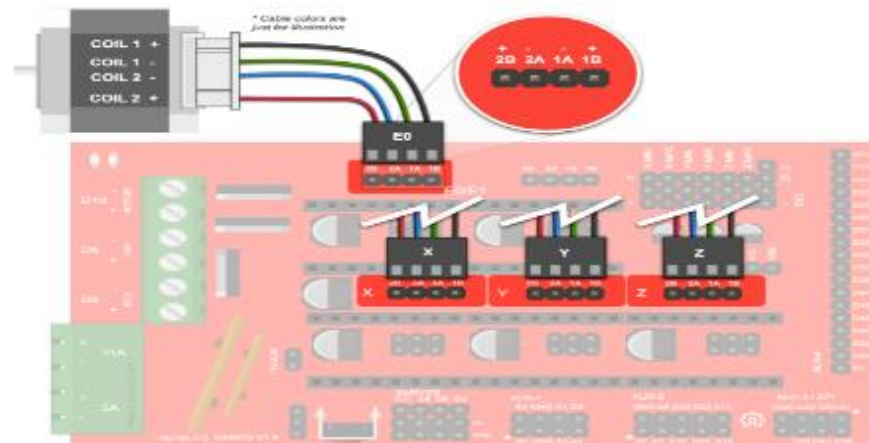
Fonte: Autor (2023)

Os componentes utilizados para essa foram citados na revisão e alguns pontos são essenciais para essa montagem, conforme será descrito.

Nessa etapa foi iniciada com a fixação dos motores de passo na estrutura da impressora com seus respectivos cabos, tendo o 1 motor para o movimento em X, 1 motor para em movimento Y, 2 motores para movimento em Z e 1 motor para movimento da extrusora. Todos esses motores serão interligados a placa Ramps. Um ponto que merece atenção é que os motores do eixo Z serão interligados juntos e serão controlados pelo mesmo drive.

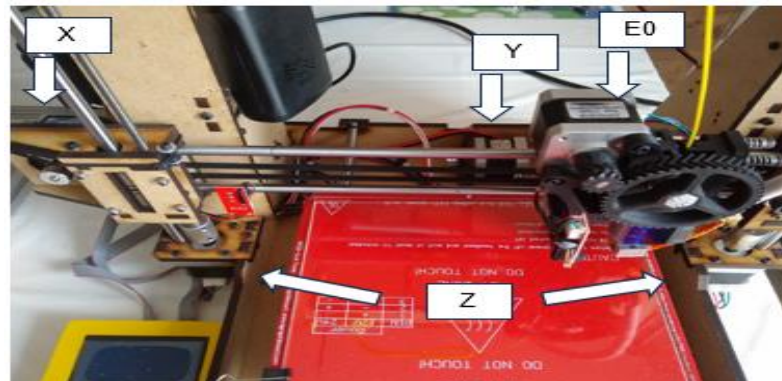
Na Figura 34 pode ser visto a ligação dos motores de passo na ramps e na Figura 35 pode ser visto a disposições dos motores junto a estrutura mecânica.

Figura 34 - Ligação motores passo na placa Ramps



Fonte: <https://reprap.org> (2023)

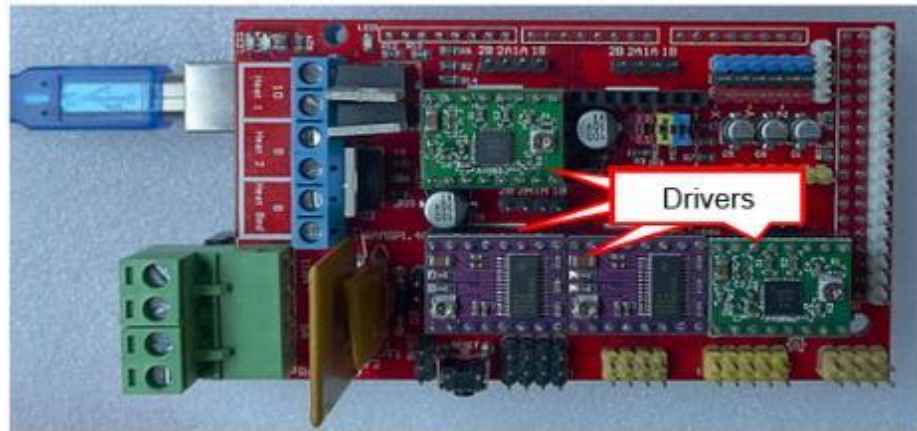
Figura 35 - Disposição motores de passo na estrutura



Fonte: Autor (2023)

Para acionamento dos motores de passo, foram utilizados os drives DVR8825, conforme explicado acima. Esses drives são fixados diretamente na ramps através de seus pinos e nesse drive há um trimpot (resistor variável) que é responsável por ajustar a corrente nos motores. Na Figura 36 mostra os drives conectados a ramps.

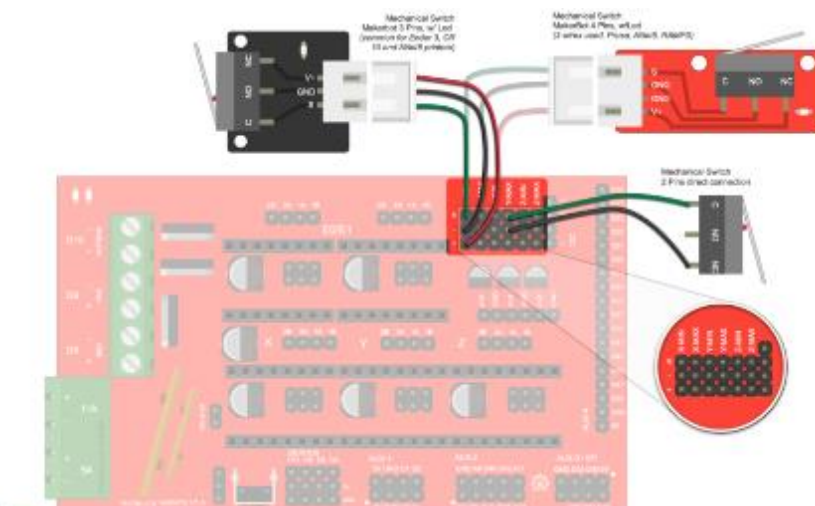
Figura 36 - Posição dos drives na placa Ramps



Fonte: <https://reprap.org> , adaptado pelo Autor (2023)

Próxima montagem será os sensores endstop que são responsáveis pelo referenciamento (ponto zero) da impressora. Conforme citado acima a impressora pode ter até 4 sensores, sendo 1 para o referenciamento x, outro para o Y, outro para o Z e caso necessário outro para o fim de filamento. Nesse trabalho iremos utilizar somente 3 sensores, não será utilizado o sensor de fim de filamento. Grande parte das impressoras utilizam sensores 2 fios, que funcionam perfeitamente, mas iremos utilizar os sensores de 3 fios que tem uma indicação visual através de um led que mostrará quando o mesmo está ativado, conforme pode ser visto na Figura 37.

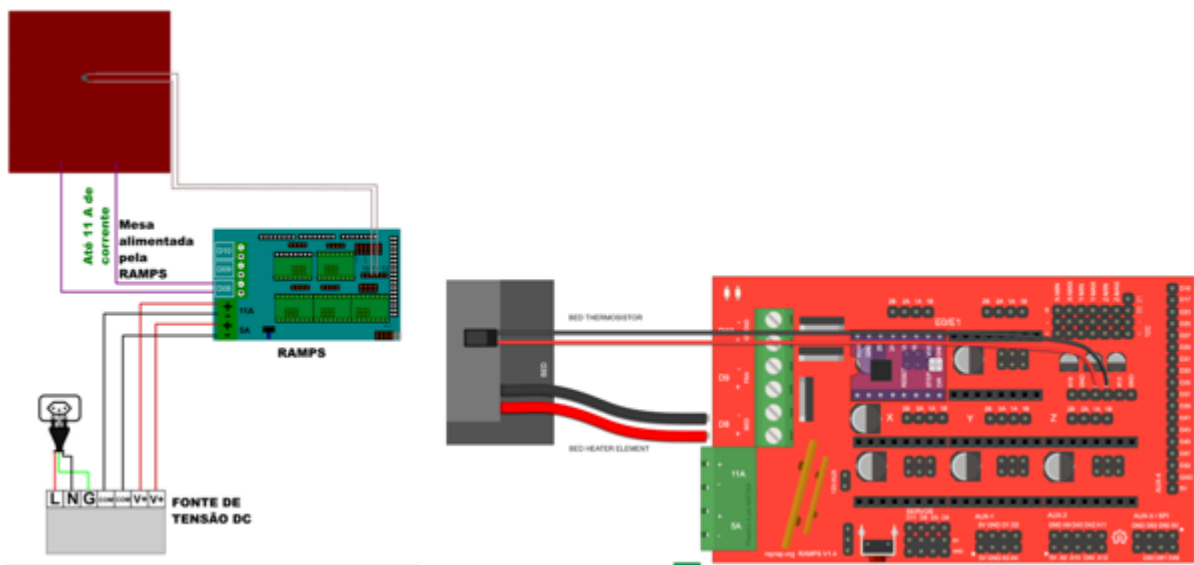
Figura 37 - Ligação dos endstops



Fonte: <https://reprap.org> (2023).

Na próxima montagem será feito a interligação da mesa aquecida que conforme mencionado acima é uma mesa onde será feito a impressão, mas que tem uma resistência para que a mesma aqueça e ajude na fixação da peça a ser impressa na mesa garantindo uma aderência e não venha se soltar durante a impressão. A ligação da mesa com a ramps será feito através de modulo de potência. Esse módulo é responsável por diminuir a corrente nos terminais da ramps, que como fornece potência para outros componentes da impressora e a corrente é elevada esses terminais podem-se danificar e prejudicar o funcionamento da impressora. Na mesa também foi inserido um termistor que tem a função de efetuar a medição da temperatura na mesa e informar ao Arduino através da ramps para o correto gerenciamento de temperatura. Na Figura 38 mostra a mesa com a sua interligação.

Figura 38 - Ligação mesa /Ramps

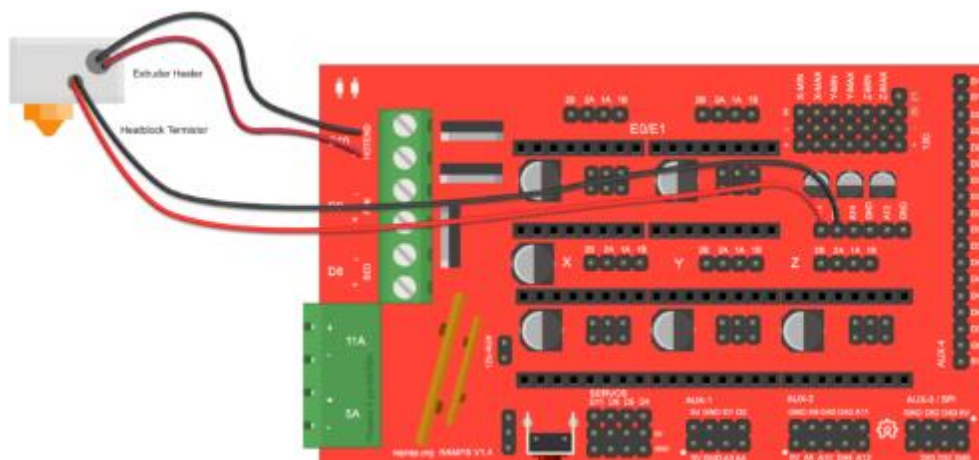


Fonte: Github. guia-maker-da-impressao-3d-tecnologiaaff.adoc adaptado pelo Autor (2023).

A próxima interligação será o hotend. Conforme citado acima o hotend é um componente que aquece o filamento para efetuar a impressão, para isso o mesmo dependerá de uma resistência de aquecimento para que venha fundir o filamento para ser aplicado. As características da resistência utilizada no hotend foram de 12V, 40W podendo chegar a uma temperatura de 280°C.

Para garantir a temperatura solicitada o mesmo tem um sensor (termistor) que efetuará a medição da temperatura e enviará o sinal para o Arduino através da ramps para efetuar esse controle. No corpo do hotend há um cooler responsável pela refrigeração de seu corpo chamado de heatbreak, garantindo que o filamento não venha se fundir nessa região e venha obstruir a passagem do filamento. Na Figura 39 mostra a ligação entre o hotend e a ramps.

Figura 39 - Ligação Hotend na placa Ramps

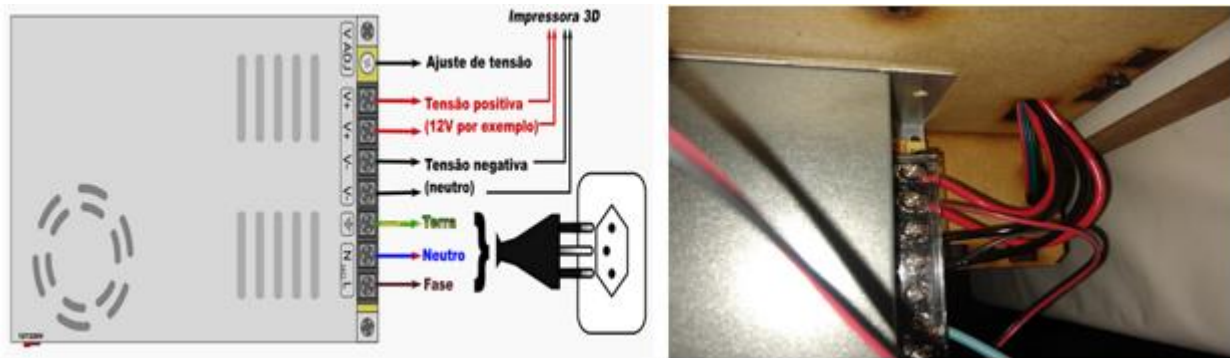


Fonte: <https://reprap.org> (2023).

Por último e com grande importância é a fonte para alimentação dos circuitos e componentes, a fonte utilizada foi uma fonte tipo colmeia de 12V, 30 A. A corrente elevada é para garantir todos os componentes funcionando ao mesmo tempo como motores de passo, resistências de aquecimento e cooler. Foi escolhido a colocação da fonte separadamente para que não fique desorganizado a montagem.

Na Figura 40 pode ser visto a fonte e as suas interligações.

Figura 40 - Interligação Fonte.



Fonte: Autor (2023)

Após a interligação de todos os componentes, é necessário alimentar a impressora e efetuar alguns ajustes para o seu bom funcionamento, antes de instalar o firmware, efetuar a calibração e efetuar os testes finais.

Esse ajuste é garantir o correto funcionamento dos motores de passo, pois conforme mencionando acima é necessário a corrente correta nos motores. Esse ajuste é feito medindo-se a tensão no tripot e ajustando-se o mesmo. Para isso foi utilizado uma equação conforme o datashet do drive DVR8825, conforme mostrado abaixo:

$$I_{FS} (A) = \frac{xVREF (V)}{A_v \times R_{SENSE} (\Omega)} = \frac{xVREF (V)}{5 \times R_{SENSE} (\Omega)} \quad (eq1)$$

Onde:

IFS = Corrente full scale – corrente do motor (A)

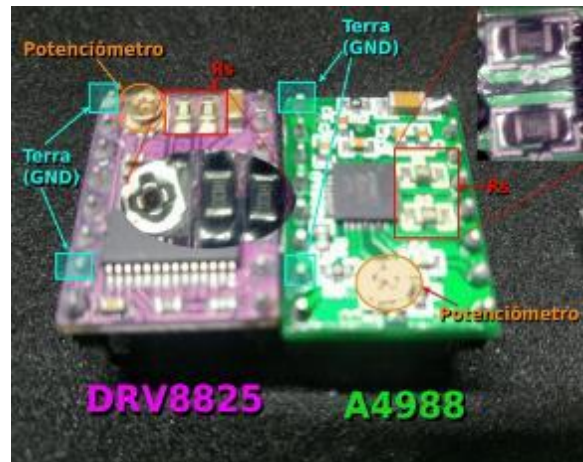
Vref = Tensão de referência (V)

Av = Ganho

Rsense = Resistor sensor de corrente. (Ω)

Para determinar a corrente dos motores, utilizaremos 70% da corrente máxima do motor, isso para garantir que o motor não venha apresentar muito aquecimento e garantir a vida útil do motor. Verificando também qual é o resistor sensor (Rsense) do drive conforme Figura 41, e aplicando a equação 1 consegue-se determinar qual a tensão Vref necessária para garantir 70% da corrente do motor.

Figura 41 - Rsense e trimpot de ajuste dos Drivers DVR8825 e A4988

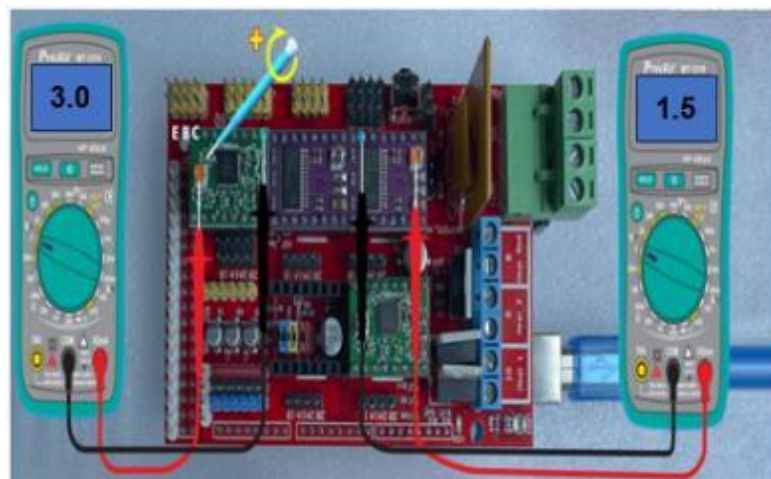


Fonte: <https://reprap.org> (2023).

No trabalho foi utilizado motores de passo de 1,7 A, conforme citado anteriormente, então para 70% utiliza-se 1,2 A, e os drives utilizados possuem Rsense de 0,25K Ω , inserindo na equação precisaremos de uma tensão Vref no trimpot de 1,5V, isso para os motores do eixo X, Y e motor da Extrusora.

Para o eixo Z, teremos 2 motores em paralelo então seria necessário dobrar a tensão, pois com motores em paralelo o drive iria fornecer 1,2 A mas cada motor terá 0,6 A, então para isso viu-se necessário ajustar para um Vref de 3,0V, deforma a garantir 1,2 A em cada motor. Na Figura 42 mostra como faz a medição dessas tensões e o ponto de ajuste.

Figura 42 - Medições e ponto de ajuste de tensão dos motores de passo



Fonte: guia-maker-da-impresao-3d-tecnologiaff.adoc(2023).

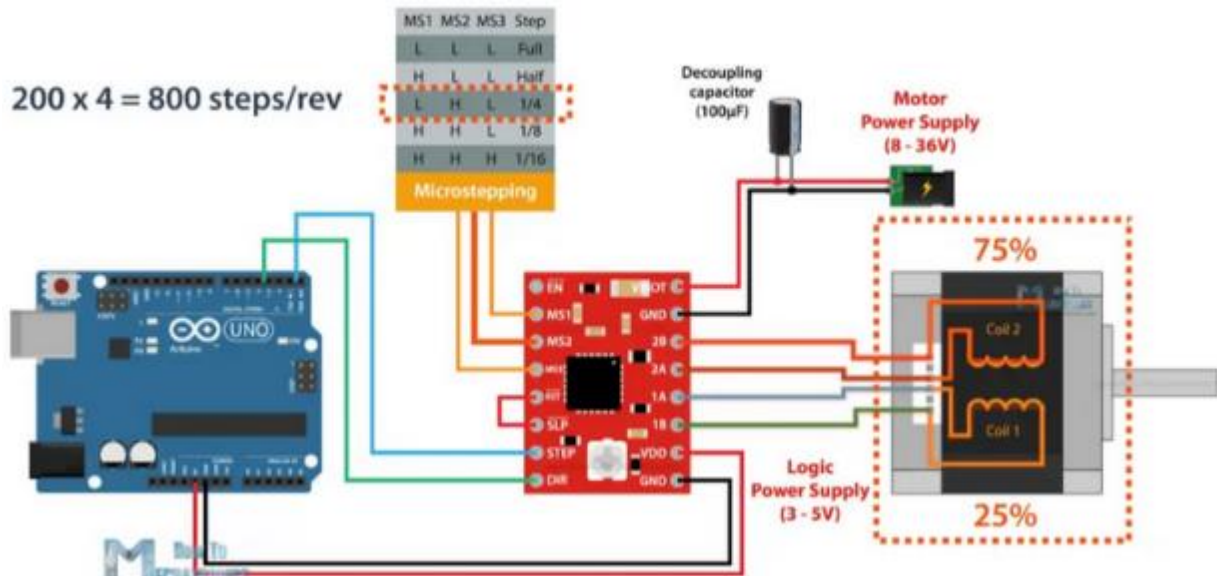
Outra observação referente aos drive para sua configuração são os micro-passos, quanto maior o micro-passo melhor a precisão da impressora, ou seja, a cada passo do motor ele se desloca em um ângulo correspondente, então se pegarmos e dividir em micro passo, pode-se garantir uma melhor precisão no seu deslocamento, diminuindo o ângulo de deslocamento, mas tem se um ponto ruim, quanto mais dividir em micropasso, menor será o torque do motor. Então podemos dizer que é necessário fazer uma verificação de corrente e micropasso para garantir que os motores consigam se movimentar corretamente sem travar e garantir boa precisão. Essa configuração de micropassos é feita na ramps através de jumper. Nesse trabalho foi configurado para 1/32 passos. Nas Figuras 43 e 44 podem ser vistos a os micros passos do DVR8825 e a ligação e os pontos de seleção dos micros passos no drive.

Figura 43 - Micro passos dirve DVR8825

MODE2	MODE1	MODE0	STEP MODE
0	0	0	Full step (2-phase excitation) with 71% current
0	0	1	1/2 step (1-2 phase excitation)
0	1	0	1/4 step (W1-2 phase excitation)
0	1	1	8 microsteps/step
1	0	0	16 microsteps/step
1	0	1	32 microsteps/step
1	1	0	32 microsteps/step
1	1	1	32 microsteps/step

Fonte: Texas Instruments (2023)

Figura 44 - Exemplo de Pontos de seleção de micro passos



Fonte: Howtomechatronic (2023)

Feito esses ajustes pode-se avançar para as próximas etapas que seria a instalação do firmware, calibração, testes e validações.

4.3 Instalação e parametrização do firmware

Atualmente, existem vários firmwares disponíveis para instalação em impressoras 3D, sendo dois deles o Marlin e o Klipper. O Marlin é compatível com microcontroladores de 8 bits até 32 bits, enquanto o Klipper é adequado para microcontroladores acima de 32 bits. No contexto deste trabalho, o microcontrolador utilizado é o Arduino, que opera em 8 bits, tornando necessário o uso do firmware Marlin.

O Marlin é um software de código aberto e pode ser baixado no site oficial do Marlin¹. Ele desempenha um papel vital ao coordenar aquecedores, motores de passo, sensores, luzes, display LCD, botões e todos os componentes envolvidos no processo de impressão 3D. O software "interpreta" cada linha de código e envia os sinais elétricos apropriados para os motores, seja por meio de cabo serial, USB, cartão SD ou conexão sem fio (MARLIN, 2023). Neste projeto, utilizou-se a versão 1.8.15 do Marlin, embora versões mais atualizadas estejam disponíveis.

¹ <https://marlinfw.org/meta/download/>

Para instalar o Marlin no Arduino, é necessário possuir a Interface de Desenvolvimento Integrada (IDE) do Arduino, que pode ser obtida gratuitamente no site oficial do Arduino². A Figura 45 mostra a interface do software de programação do Arduino, que é a ferramenta fundamental para carregar o firmware na placa Arduino. Esse processo é essencial para personalizar e otimizar o desempenho da impressora 3D, permitindo uma impressão precisa e confiável.

Figura 45 - Interface IDE do Arduino



```
Marlin | Arduino 1.8.15
Arquivo  Editar  Sketch  Ferramentas  Ajuda
Marlin  Conditionals.h  Conditionals_LCD.h  Conditionals_post.h  Configuration.h  Configuration_adv.h
1  #
2  -----
3
4  Marlin Firmware
5
6  (c) 2011-2018 MarlinFirmware
7  Portions of Marlin are (c) by their respective authors.
8  All code complies with GPLv2 and/or GPLv3
9  -----
10
11
12  Greetings! Thank you for choosing Marlin as your 3D printer firmware.
13
14  To configure Marlin you must edit Configuration.h and Configuration_adv.h
15  located in the root 'Marlin' folder. Check the example_configurations folder to
16  see if there's a more suitable starting-point for your specific hardware.
17
18  Before diving in, we recommend the following essential links:
19
20  Marlin Firmware Official Website
21
22  - http://marlinfw.org/
23  The official Marlin Firmware website contains the most up-to-date
24  documentation. Contributions are always welcome!
25
26  Configuration
27
28  - https://www.youtube.com/watch?v=3gwWVFtdg-4
29  -----
30
31  Arduino Mega or Mega 2560, ATmega2560 (Mega 2560) em COM5
```

Fonte: Autor (2023)

Para realizar as configurações essenciais no firmware, é necessário abrir o Marlin no ambiente de programação do Arduino e efetuar as seguintes personalizações:

- Configurar o Arduino, selecionando a porta serial e a velocidade de comunicação.
- Especificar os tipos de sensores de fim de curso (endstop) utilizados na impressora.
- Determinar os deslocamentos em relação aos passos dos motores.
- Definir o offset Z para o nivelamento correto da mesa de impressão.
- Estabelecer o sentido de movimentação dos eixos da impressora.

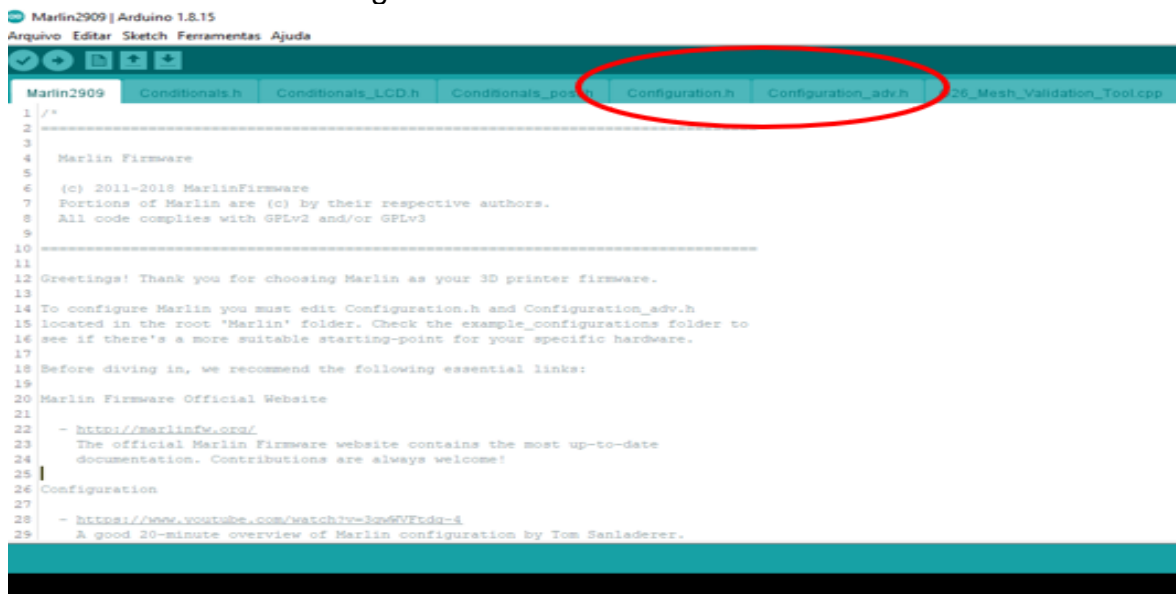
² <https://www.arduino.cc/en/software>

- Escolher o tipo de display a ser utilizado.
- Identificar o tipo de impressora e suas dimensões, incluindo altura, largura e comprimento da área de impressão.
- Implementar proteções contra superaquecimento nos componentes da impressora.

O processo completo de parametrização da impressora será detalhado no apêndice A do projeto. É importante observar que para efetuar essas parametrizações, é fundamental ter ajustado previamente a corrente dos motores, conforme explicado anteriormente. Essas configurações precisas são cruciais para garantir o funcionamento adequado e a precisão das impressões.

Basicamente, a configuração do firmware envolve dois arquivos distintos. O primeiro é denominado `configuration.h`, que contém parâmetros relacionados ao movimento e funcionalidades básicas. O segundo arquivo é o `configuration.adv`, no qual são especificadas configurações mais avançadas e detalhadas. Ambos os arquivos são essenciais para parametrizar a impressora 3D de acordo com as necessidades específicas do usuário, como ilustrado na Figura 46. Através dessas configurações é possível otimizar a performance da impressora e assegurar resultados de alta qualidade durante o processo de impressão.

Figura 46 - Firmware Marlin aberto no IDE



```

1 /*
2
3 -----
4 Marlin Firmware
5
6 (c) 2011-2018 MarlinFirmware
7 Portions of Marlin are (c) by their respective authors.
8 All code complies with GPLv2 and/or GPLv3
9
10 -----
11
12 Greetings! Thank you for choosing Marlin as your 3D printer firmware.
13
14 To configure Marlin you must edit Configuration.h and Configuration_adv.h
15 located in the root 'Marlin' folder. Check the example_configurations folder to
16 see if there's a more suitable starting-point for your specific hardware.
17
18 Before diving in, we recommend the following essential links:
19
20 Marlin Firmware Official Website
21
22 - http://marlinfw.org/
23   The official Marlin Firmware website contains the most up-to-date
24   documentation. Contributions are always welcome!
25
26 Configuration
27
28 - https://www.youtube.com/watch?v=3pe6VFtdg-4
29   A good 20-minute overview of Marlin configuration by Tom Sacladerer.

```

Fonte: Autor (2023)

Na aba `conFIGuration.h` podem ser feitas as seguintes configurações:

- Porta serial do Arduino e suas configurações
- Tipo de eletrônica a ser utilizada
- Quantidade de extrusores
- Tipo de fonte
- Tipo de sensor de temperatura
- Tipo geométrico da impressora
- Configuração dos endstops
- Tipo de LCd

Abrindo o firmware Marlin pelo ide do Arduino, será parametrizado a variável referente a porta serial que o Arduino irá se comunicar com o seu computador e a sua velocidade de comunicação.

Na Figura 47 pode ser visto essa variável que é importante deixar com parâmetro 0, para que toda vez que for conectado ao seu computador ele reconhecerá qualquer porta USB, caso coloque outro valor, o Arduino irá se comunicar somente com a porta parametrizada nesse parâmetro, e com a velocidade de 115200, que seria a velocidade que o Arduino trabalha sem problemas. Então foi inserido o valor 0 e velocidade de 115200.

Figura 47 - Programação Porta USB e velocidade

```
106 // @section machine
107
108 /**
109  * Select the serial port on the board to use for communication with the host.
110  * This allows the connection of wireless adapters (for instance) to non-default port pins.
111  * Serial port 0 is always used by the Arduino bootloader regardless of this setting.
112  *
113  * :[0, 1, 2, 3, 4, 5, 6, 7]
114  */
115 #define SERIAL_PORT 0
116
117 /**
118  * This setting determines the communication speed of the printer.
119  *
120  * 250000 works in most cases, but you might try a lower speed if
121  * you commonly experience drop-outs during host printing.
122  * You may try up to 1000000 to speed up SD file transfer.
123  *
124  * :[2400, 9600, 19200, 38400, 57600, 115200, 250000, 500000, 1000000]
125  */
126 #define BAUDRATE 115200
```

Fonte: Autor (2023)

Na próxima etapa de configuração, é necessário determinar o tipo de eletrônica a ser utilizada na impressora. Neste projeto, optou-se por trabalhar com a Ramps 1.6, que serve como a interface entre o Arduino, os sensores e os atuadores da impressora. É importante mencionar que existem várias versões disponíveis para essa eletrônica. A escolha foi pela Ramps 1.6 devido à sua compatibilidade com o projeto, mas outras versões estão disponíveis no mercado.

A Figura 48 ilustra essa parametrização. É fundamental fazer a escolha correta, pois as pinagens variam entre as diferentes versões. Se a escolha for diferente da configurada, as pinagens não serão correspondentes, resultando em um funcionamento inadequado da impressora. Na variável, optou-se pela 1.4 EFB, que possui o mesmo pinout da 1.6, garantindo a correta interligação dos componentes da impressora. Essa escolha é vital para o correto funcionamento de todos os aspectos da impressora 3D.

Figura 48 - Variável Ramps

```

131 // The following define selects which electronics board you have.
132 // Please choose the name from boards.h that matches your setup
133 #ifndef MOTHERBOARD
134 #define MOTHERBOARD BOARD_RAMPS_14_EFB
135 #endif

```

Fonte: Autor (2023)

Para verificar qual as placa e versões que o Marlin pode trabalhar, pode ser acessado a aba Board e efetuar a verificações dos tipos, conforme mostrado na Figura 49.

Conforme mencionado foi escolhida a ramps 1.4 EFB, que determina as saídas de potência: E (resistência de aquecimento do hotend), F (controle do ventilador do hotend), B (resistência de aquecimento da mesa de impressão).

Figura 49 - Tipos de Ramps que o Marlin pode controlar

```

32 #define BOARD_RAMPS_OLD      3      // MEGA/RAMPS up to 1.2
33
34 #define BOARD_RAMPS_13_EFB  33     // RAMPS 1.3 (Power outputs: Hotend, Fan, Bed)
35 #define BOARD_RAMPS_13_EEB  34     // RAMPS 1.3 (Power outputs: Hotend0, Hotendl, Bed)
36 #define BOARD_RAMPS_13_EFF  35     // RAMPS 1.3 (Power outputs: Hotend, Fan0, Fan1)
37 #define BOARD_RAMPS_13_EEF  36     // RAMPS 1.3 (Power outputs: Hotend0, Hotendl, Fan)
38 #define BOARD_RAMPS_13_SF   38     // RAMPS 1.3 (Power outputs: Spindle, Controller Fan)
39
40 #define BOARD_RAMPS_14_EFB  43     // RAMPS 1.4 (Power outputs: Hotend, Fan, Bed)
41 #define BOARD_RAMPS_14_EEB  44     // RAMPS 1.4 (Power outputs: Hotend0, Hotendl, Bed)
42 #define BOARD_RAMPS_14_EFF  45     // RAMPS 1.4 (Power outputs: Hotend, Fan0, Fan1)
43 #define BOARD_RAMPS_14_EEF  46     // RAMPS 1.4 (Power outputs: Hotend0, Hotendl, Fan)
44 #define BOARD_RAMPS_14_SF   48     // RAMPS 1.4 (Power outputs: Spindle, Controller Fan)
45
46 #define BOARD_RAMPS_PLUS_EFB 143    // RAMPS Plus 3DYM (Power outputs: Hotend, Fan, Bed)
47 #define BOARD_RAMPS_PLUS_EEB 144    // RAMPS Plus 3DYM (Power outputs: Hotend0, Hotendl, Bed)
48 #define BOARD_RAMPS_PLUS_EFF 145    // RAMPS Plus 3DYM (Power outputs: Hotend, Fan0, Fan1)
49 #define BOARD_RAMPS_PLUS_EEF 146    // RAMPS Plus 3DYM (Power outputs: Hotend0, Hotendl, Fan)
50 #define BOARD_RAMPS_PLUS_SF  148    // RAMPS Plus 3DYM (Power outputs: Spindle, Controller Fan)

```

Fonte: Autor (2023)

A próxima etapa de configuração refere-se ao número de extrusoras a serem utilizadas e ao diâmetro do filamento, de acordo com o extrusor adquirido. Neste projeto, foi utilizado apenas um extrusor com entrada para filamento de 1,75 mm. No entanto,

caso haja a necessidade de trabalhar com múltiplos extrusores em uma única máquina, é possível ajustar esses parâmetros para a quantidade desejada e modificar o diâmetro conforme necessário. A Figura 50 ilustra essa variável.

Essa configuração flexível permite adaptar a impressora às necessidades específicas do usuário, seja para projetos simples com uma extrusora ou para aplicações mais complexas que requerem várias extrusoras e diferentes diâmetros de filamento. Essa versatilidade é fundamental para a personalização da impressora 3D de acordo com os requisitos de cada projeto.

Figura 50 - Variável número de extrusores e diâmetro do filamento

```
145 // @section extruder
146
147 // This defines the number of extruders
148 // :[1, 2, 3, 4, 5]
149 #define EXTRUDERS 1
150
151 // Generally expected filament diameter (1.75, 2.85, 3.0, ...). Used for Volumetric, Filament Width Sensor, etc.
152 #define DEFAULT_NOMINAL_FILAMENT_DIA 1.75
```

Fonte: Autor (2023)

O diâmetro do filamento deve ser informado corretamente, pois é por meio desse parâmetro que o volume de aplicação correto será calculado. Isso é essencial para garantir a qualidade e o funcionamento adequado da impressora 3D. A precisão na especificação do diâmetro do filamento é fundamental para a consistência das impressões e a precisão dos detalhes.

Na próxima configuração, foi selecionado o tipo de fonte de alimentação a ser utilizada. Este parâmetro é importante não apenas para fornecer energia à impressora, mas também para possibilitar o desligamento ou religamento remoto da mesma quando não está em operação. Nesse trabalho, a opção escolhida foi a opção 1, que utiliza uma fonte de alimentação chaveada sem função de desligamento. A Figura 51 mostra essa configuração.

A escolha correta do tipo de fonte de alimentação é essencial para garantir a estabilidade do sistema e permitir um controle eficaz da impressora, mesmo à distância.

Figura 51 - Variável configuração da fonte de alimentação

```

230 // @section machine
231
232 /**
233  * Select your power supply here. Use 0 if you haven't connected the PS_ON_PIN
234  *
235  * 0 = No Power Switch
236  * 1 = ATX
237  * 2 = X-Box 360 203Watts (the blue wire connected to PS_ON and the red wire to VCC)
238  *
239  * :{ 0:'No power switch', 1:'ATX', 2:'X-Box 360' }
240  */
241 #define POWER_SUPPLY 1

```

Fonte: Autor (2023)

Na próxima etapa de configuração, é realizada a parametrização do sensor de temperatura utilizado na impressora 3D. Existe uma ampla variedade de sensores disponíveis no mercado, e é importante configurar corretamente o sensor específico que está sendo utilizado. Para verificar se o sensor em uso é compatível com o Marlin, basta consultar a lista disponível na própria programação do software. A Figura 52 ilustra alguns dos sensores monitorados pelo Marlin.

A escolha e configuração adequadas do sensor de temperatura são essenciais para garantir a precisão na medição e no controle da temperatura durante o processo de impressão. Uma correspondência precisa entre o sensor e o firmware assegura um monitoramento confiável e preciso, garantindo a qualidade das impressões 3D.

Figura 52 - Tipos de sensores de temperatura no Marlin

```

258 // @section temperature
259
260 //=====
261 //===== Thermal Settings =====
262 //=====
263
264 /**
265  * --NORMAL IS 4.7kohm PULLUP!-- 1kohm pullup can be used on hotend sensor, using correct resistor and table
266  *
267  * Temperature sensors available:
268  *
269  *   -4 : thermocouple with AD8495
270  *   -3 : thermocouple with MAX31855 (only for sensor 0)
271  *   -2 : thermocouple with MAX6675 (only for sensor 0)
272  *   -1 : thermocouple with AD595
273  *   0 : not used
274  *   1 : 100k thermistor - best choice for EPCOS 100k (4.7k pullup)
275  *   2 : 200k thermistor - ATC Semitec 204GT-2 (4.7k pullup)
276  *   3 : Mendel-parts thermistor (4.7k pullup)
277  *   4 : 10k thermistor !! do not use it for a hotend. It gives bad resolution at high temp. !!

```

Fonte: Autor (2023)

Neste projeto específico, foi utilizado o sensor identificado pelo código 1, conforme indicado na Figura 52. No entanto, a escolha do sensor pode variar com base nas necessidades e requisitos do usuário.

Na Figura 53, é possível visualizar a programação referente ao tipo de sensor empregado na impressora 3D. Durante essa parametrização, é importante configurar dois sensores: um para o hotend e outro para a mesa de impressão. É relevante observar que, em algumas configurações, pode haver mais de um hotend. Neste trabalho, a parametrização foi realizada para um único hotend e uma única mesa, com as demais variáveis ajustadas para 0, indicando que não estão sendo utilizados. Esse cuidado na configuração dos sensores é essencial para garantir a precisão na medição de temperatura, fator crítico para as impressões 3D.

Figura 53 - Variáveis sensor de temperatura ativado

```

311 | * :{ '0': "Not used", '1':"100k / 4.7k - EPCOS",
312 | */
313 | #define TEMP_SENSOR_0 1
314 | #define TEMP_SENSOR_1 0
315 | #define TEMP_SENSOR_2 0
316 | #define TEMP_SENSOR_3 0
317 | #define TEMP_SENSOR_4 0
318 | #define TEMP_SENSOR_BED 1
319 | #define TEMP_SENSOR_CHAMBER 0

```

Fonte: Autor (2023)

Para que tenha um bom controle de funcionamento, é importante parametrizar as temperaturas mínimas e máximas de operação do sensor, sabendo-se o sensor tem um range de 5 a 300°C, é necessário informar esses valores no firmware, conforme mostrado na Figura 54.

Com isso irá garantir que abaixo da temperatura mínima parametrizada a impressora não irá ligar a potência para aquecimento, e quando ultrapassar o valor parametrizado irá desligar o sistema de aquecimento. Isso garante que caso o cabo do sensor se danifique a impressora não fique com a potência de aquecimento ligado todo o tempo.

Figura 54 - Variáveis Mínimo e Máximo do sensor de temperatura

```
340 // The minimal temperature defines the temperature below which the heater will not be enabled It is used
341 // to check that the wiring to the thermistor is not broken.
342 // Otherwise this would lead to the heater being powered on all the time.
343 #define HEATER_0_MINTEMP 5
344 #define HEATER_1_MINTEMP 5
345 #define HEATER_2_MINTEMP 5
346 #define HEATER_3_MINTEMP 5
347 #define HEATER_4_MINTEMP 5
348 #define BED_MINTEMP 5
349
350 // When temperature exceeds max temp, your heater will be switched off.
351 // This feature exists to protect your hotend from overheating accidentally, but *NOT* from thermistor short/failure!
352 // You should use MINTEMP for thermistor short/failure protection.
353 #define HEATER_0_MAXTEMP 265
354 #define HEATER_1_MAXTEMP 275
355 #define HEATER_2_MAXTEMP 275
356 #define HEATER_3_MAXTEMP 275
357 #define HEATER_4_MAXTEMP 275
358 #define BED_MAXTEMP 150
```

Fonte: Autor (2023)

Foi definido nesse trabalho uma temperatura mínima para o hotend e mesa de 5°C e uma temperatura máxima de 265°C para o hotend e 150°C para a mesa, mas poderia ser inserido qualquer valor dentro do range do sensor, ou da temperatura de trabalho do hotend.

Outra configuração que deve ser feita é a de proteção térmica, caso venha ter algum problema no sistema de aquecimento tanto do hotend quanto da mesa a impressora poderá pegar fogo, então é necessário habilitar essa proteção, conforme Figura 55.

Figura 55- Variável proteção temperatura

```

466 //=====
467 //===== Thermal Runaway Protection =====
468 //=====
469
470 /**
471 * Thermal Protection provides additional protection to your printer from damage
472 * and fire. Marlin always includes safe min and max temperature ranges which
473 * protect against a broken or disconnected thermistor wire.
474 *
475 * The issue: If a thermistor falls out, it will report the much lower
476 * temperature of the air in the room, and the the firmware will keep
477 * the heater on.
478 *
479 * If you get "Thermal Runaway" or "Heating failed" errors the
480 * details can be tuned in Configuration_adv.h
481 */
482
483 #define THERMAL_PROTECTION_HOTENDS // Enable thermal protection for all extruders
484 #define THERMAL_PROTECTION_BED // Enable thermal protection for the heated bed

```

Fonte: Autor (2023)

O tempo de monitoramento para essa proteção foi configurada para 40s e incremento a cada 4°C, isso garante que se a impressora não atingir a temperatura parametrizada máxima em 40s em um incremento cada 4°C a mesma irá desligar o sistema de potência, garantindo a segurança, os parâmetros podem ser vistos na Figura 56.

Figura 56 - Variável tempo e histerese proteção de temperatura

```

74 * If you get false positives for "Thermal Runaway", increase
75 * THERMAL_PROTECTION_HYSTERESIS and/or THERMAL_PROTECTION_PERIOD
76 */
77 #if ENABLED(THERMAL_PROTECTION_HOTENDS)
78 #define THERMAL_PROTECTION_PERIOD 40 // Seconds
79 #define THERMAL_PROTECTION_HYSTERESIS 4 // Degrees Celsius

```

Fonte: Autor (2023)

A próxima configuração é a dos fins de curso (endstop), essencial para estabelecer os pontos zero da mesa ou o ponto de origem (home) e os pontos máximos da mesa.

Essas configurações são fundamentais para garantir as coordenadas precisas de impressão.

Neste projeto, foi configurado que o ponto de origem (home) da impressora está localizado nos pontos mínimos dos eixos X, Y e Z. Ao definir os valores mínimos, automaticamente os valores máximos são determinados, já que o sistema possui uma profundidade predeterminada para os eixos X, Y e Z.

A Figura 57 mostra a parametrização do sensor para identificação dos pontos mínimos, assegurando o correto alinhamento da máquina ao iniciar o processo de impressão.

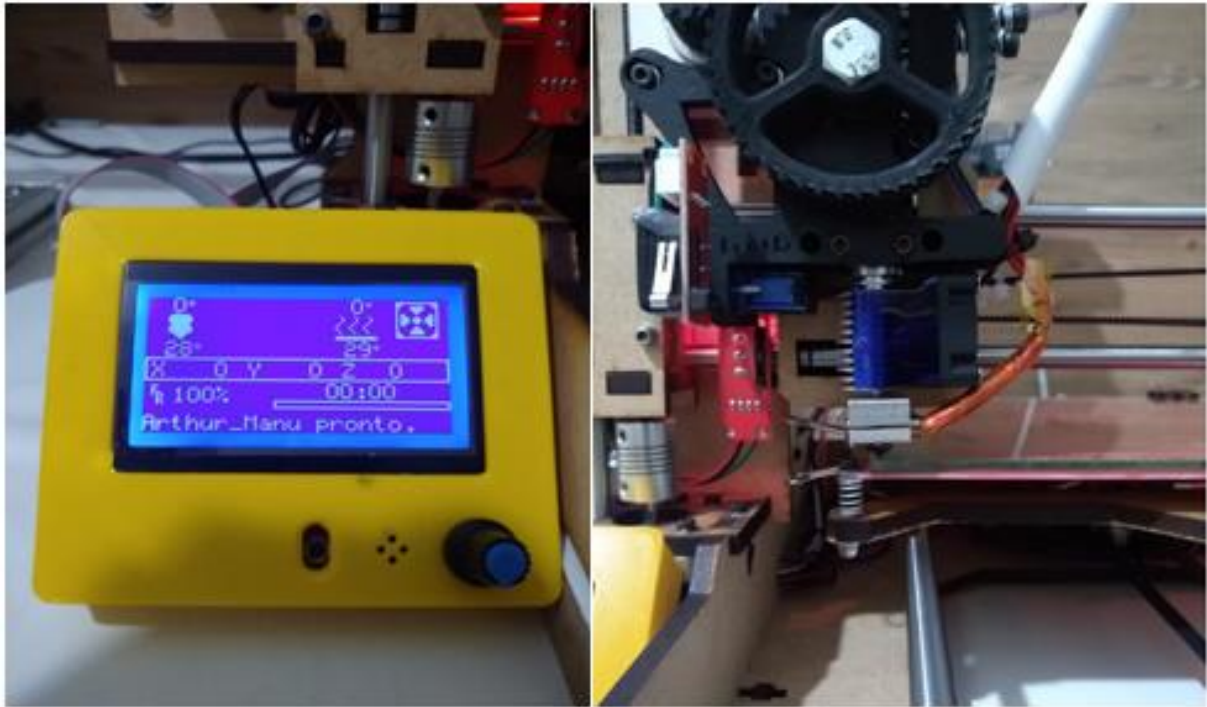
Figura 57 - Variáveis posição mínima dos endstops

```
501 //=====
502 //===== Endstop Settings =====
503 //=====
504
505 // @section homing
506
507 // Specify here all the endstop connectors that are connected to any endstop or probe.
508 // Almost all printers will be using one per axis. Probes will use one or more of the
509 // extra connectors. Leave undefined any used for non-endstop and non-probe purposes.
510 #define USE_XMIN_PLUG
511 #define USE_YMIN_PLUG
512 #define USE_ZMIN_PLUG
513 //#define USE_XMAX_PLUG
514 //#define USE_YMAX_PLUG
515 //#define USE_ZMAX_PLUG
```

Fonte: Autor (2023)

Com isso será garantido que o bico da impressora terá o seu ponto zero à esquerda, na parte da frente da mesa e com o bico encostado na mesa, conforme mostra a Figura 58.

Figura 58 - Posicionamento ponto 0 ou ponto Home



Fonte: Autor (2023)

Para garantir que ao iniciar a impressora e que venha garantir que o motor se movimento para o lado dos endstops (lado minimo) é necessário efetuar a seguinte parametrização inserindo na variável HOME DIR o valor -1 para que a mesma se movimento para o lado do endstop a fim de garantir o seu ponto zero, através desse ponto e impressora saberá qual o valor máximo da mesa de trabalho que nesse trabalho foi determinado 200mm x 200mm, através da variável X e Y_BED_SIZE conforme mostrado na Figura 59.

Figura 59 - Variáveis sentido ponto zero, volume de operação dos endstops

```

870 // Direction of endstops when homing; 1=MAX, -1=MIN
871 // :[-1,1]
872 #define X_HOME_DIR -1
873 #define Y_HOME_DIR -1
874 #define Z_HOME_DIR -1
875
876 // @section machine
877
878 // The size of the print bed
879 #define X_BED_SIZE 200
880 #define Y_BED_SIZE 200
881
882 // Travel limits (mm) after homing, corresponding to endstop positions.
883 #define X_MIN_POS 0
884 #define Y_MIN_POS 0
885 #define Z_MIN_POS 0
886 #define X_MAX_POS X_BED_SIZE
887 #define Y_MAX_POS Y_BED_SIZE
888 #define Z_MAX_POS 200

```

Fonte: Autor (2023)

Na Figura 59, observa-se que os valores mínimos de X, Y e Z foram definidos como 0 nas variáveis X_MIN_POS, Y_MIN_POS e Z_MIN_POS, enquanto os valores máximos foram estabelecidos como 200 mm em X, Y e Z. Com base nessas informações, basta identificar os pontos mínimos ou ponto de origem (home) da impressora. A partir desses pontos e dos valores inseridos, a impressora determinará automaticamente seus limites máximos, garantindo uma área de impressão precisa.

Além disso, na configuração dos fins de curso, é fundamental especificar se os sensores são do tipo Nc (normalmente fechado) ou NO (normalmente aberto). Sem essa definição, o Arduino não será capaz de identificar se o endstop foi ativado, como ilustrado na Figura 60.

Figura 60 - Variável tipos contato do endstop

```

530 // Mechanical endstop with COM to ground and NC to Signal uses "false" here (most common setup).
531 #define X_MIN_ENDSTOP_INVERTING true // set to true to invert the logic of the endstop.
532 #define Y_MIN_ENDSTOP_INVERTING true // set to true to invert the logic of the endstop.
533 #define Z_MIN_ENDSTOP_INVERTING true // set to true to invert the logic of the endstop.
534 #define X_MAX_ENDSTOP_INVERTING true // set to true to invert the logic of the endstop.
535 #define Y_MAX_ENDSTOP_INVERTING true // set to true to invert the logic of the endstop.
536 #define Z_MAX_ENDSTOP_INVERTING true // set to true to invert the logic of the endstop.
537 #define Z_MIN_PROBE_ENDSTOP_INVERTING true // set to true to invert the logic of the probe.

```

Fonte: Autor (2023)

Como os sensores utilizados são normalmente abertos foi necessário colocar o parâmetro true em X, Y e Z, considerando também o Z probe, que será discutido posteriormente.

Na próxima etapa de parametrização, é necessário selecionar o driver do motor de passo para garantir o movimento preciso dos motores. Como os motores de passo operam com um sinal PWM (Pulse Width Modulation), é necessário que o Arduino envie a informação correta para os motores, assegurando um controle preciso.

Neste projeto, foi utilizado o driver DRV8825, e é essencial informar o driver correto na programação para cada eixo, incluindo o motor da extrusora, como mostra a Figura 61. A escolha adequada do driver é fundamental para garantir o funcionamento suave e preciso dos motores de passo, resultando em movimentos precisos e impressões de alta qualidade na impressora 3D.

Figura 61 - Variáveis seleção drive DVR8825

```

540 * Stepper Drivers
541 *
542 * These settings allow Marlin to tune stepper driver timing and enable advanced options for
543 * stepper drivers that support them. You may also override timing options in Configuration_adv.h.
544 *
545 * A4988 is assumed for unspecified drivers.
546 *
547 * Options: A4988, DRV8825, LV8729, L6470, TB6560, TB6600, TMC2100,
548 *         TMC2130, TMC2130_STANDALONE, TMC2208, TMC2208_STANDALONE,
549 *         TMC26X, TMC26X_STANDALONE, TMC2660, TMC2660_STANDALONE,
550 *         TMC5130, TMC5130_STANDALONE
551 * :['A4988', 'DRV8825', 'LV8729', 'L6470', 'TB6560', 'TB6600', 'TMC2100', 'TMC2130', 'TMC2130_STANDALONE',
552 */
553 #define X_DRIVER_TYPE  DRV8825
554 #define Y_DRIVER_TYPE  DRV8825
555 #define Z_DRIVER_TYPE  DRV8825
556 //#define X2_DRIVER_TYPE A4988
557 //#define Y2_DRIVER_TYPE A4988
558 //#define Z2_DRIVER_TYPE A4988
559 #define E0_DRIVER_TYPE DRV8825
560 //#define E1_DRIVER_TYPE A4988
561 //#define E2_DRIVER_TYPE A4988
562 //#define E3_DRIVER_TYPE A4988

```

Fonte: Autor (2023)

Feito a escolha correta do drive, será necessário a configuração de passos da máquina por milímetro, parâmetro que vai garantir as dimensões correta do objeto a ser

impresso e a qualidade do mesmo. Essa configuração é feita conforme Figura 62 e deve ser efetuado a medição em cada eixo.

Figura 62 - Variável Resolução Steps/mm

```
607 * Default Axis Steps Per Unit (steps/mm)
608 * Override with M92
609 *           X, Y, Z, E0 [, E1[, E2[, E3[, E4]]]]
610 */
611 #define DEFAULT_AXIS_STEPS_PER_UNIT { 160, 160, 4000, 1368.43 }
```

Fonte: Autor (2023)

Para determinar esses valores foi utilizado uma calculadora de passos que pode ser encontrado em diversos sites, utilizou-se o site Atividade Maker³, conforme Figura 63.

Figura 63 - Calculadora passos por milímetros para correias

CORREIAS

Se a sua máquina se movimenta através de correias, utilize esse formulário!

Qual a Revolução do seu Motor

1.8 (200 passos por revolução)

Como você configurou seus Micropassos

1/32 Passo

Pitch da Correia (mm)

2

Número de Dentes da polia

20

CALCULAR

Passos por milímetro = **160 Steps/mm**

Resolução da máquina = **0.00625 mm**

Fonte: Atividade Maker (2023)

³ <https://atividademaker.com.br/calculadora-de-passos>

Para isso é necessário ter a característica angulo do motor de passo que é de 1.8° , ter a configuração dos micropassos conforme explicado acima que foi configurado para 1/32 micropassos, saber qual o pitch da correia que nesse caso é 2mm e o número de dentes da polia que vai ao eixo do motor de passo. Nesse caso para os eixos X e Y tem-se 160 steps/mm e uma resolução de 0,00625mm. Em relação ao eixo z foi utilizado uma barra roscada de M5. Segue na Figura 64 o valor do eixo Z.

Figura 64 - Calculadora passos por milímetros fusos

FUSOS EM MILÍMETROS

Se a sua máquina se movimenta através de fusos em milímetros, utilize esse formulário!

Qual a Revolução do seu Motor

1.8 (200 passos por revolução)

Como você configurou seus Micropassos

1/32 Passo

Passos do fuso (mm/revolução)

1.6

CALCULAR

Passos por milímetro = **4000 Steps/mm**

Resolução da máquina = **0.00025 mm**

Fonte: Atividade Maker (2023)

Então tem-se um valor 4000 steps/mm como uma resolução de 0,00025mm.

O valor de passos da extrusora será definido na parte da calibração, esses valores podem sofrer alterações, pois devido aos componentes mecânicos pode ser que seja necessário estar alterando esses valores.

Para a última etapa de configuração do firmware é a parte do LCD que é necessário para estar definindo o idioma e o tipo de display que será utilizado. Nas Figuras 65 e 66 mostra a configuração do idioma, e do tipo de display, foi utilizado nesse projeto o display gráfico 128 x 64 e idioma português.

Figura 65 - Variável seleção do tipo de idioma

```

1388 * LCD LANGUAGE
1389 *
1390 * Select the language to display on the LCD. These languages are available:
1391 *
1392 *   en, an, bg, ca, cn, cz, cz_utf8, de, el, el-gr, es, es_utf8, eu,
1393 *   fi, fr, fr_utf8, gl, hr, it, kana, kana_utf8, ko_KR, nl, pl, pt,
1394 *   pt_utf8, pt-br, pt-br_utf8, ru, sk_utf8, tr, uk, zh_CN, zh_TW, test
1395 *
1396 * :{ 'en':'English', 'an':'Aragonese', 'bg':'Bulgarian', 'ca':'Catalan', 'cn':'Chinese',
1397 */
1398 #define LCD LANGUAGE pt-br

```

Fonte: Autor (2023)

Figura 66 - Tipo do display

```

1651 // CONTROLLER TYPE: Graphical 128x64 (DOGM)
1652 //
1653 // IMPORTANT: The U8glib library is required for Graphical Display!
1654 //           https://github.com/olikraus/U8glib\_Arduino
1655 //
1656
1657 //
1658 // RepRapDiscount FULL GRAPHIC Smart Controller
1659 // http://reprap.org/wiki/RepRapDiscount\_Full\_Graphic\_Smart\_Controller
1660 //
1661 #define REPRAP_DISCOUNT_FULL_GRAPHIC_SMART_CONTROLLER

```

Fonte: Autor (2023)

Terminado essas configurações básicas, é necessário descarregar o código no Arduino e terminar o processo de configuração e calibração.

Feito o processo de parametrização será necessário iniciar o processo de calibração e validação da impressora.

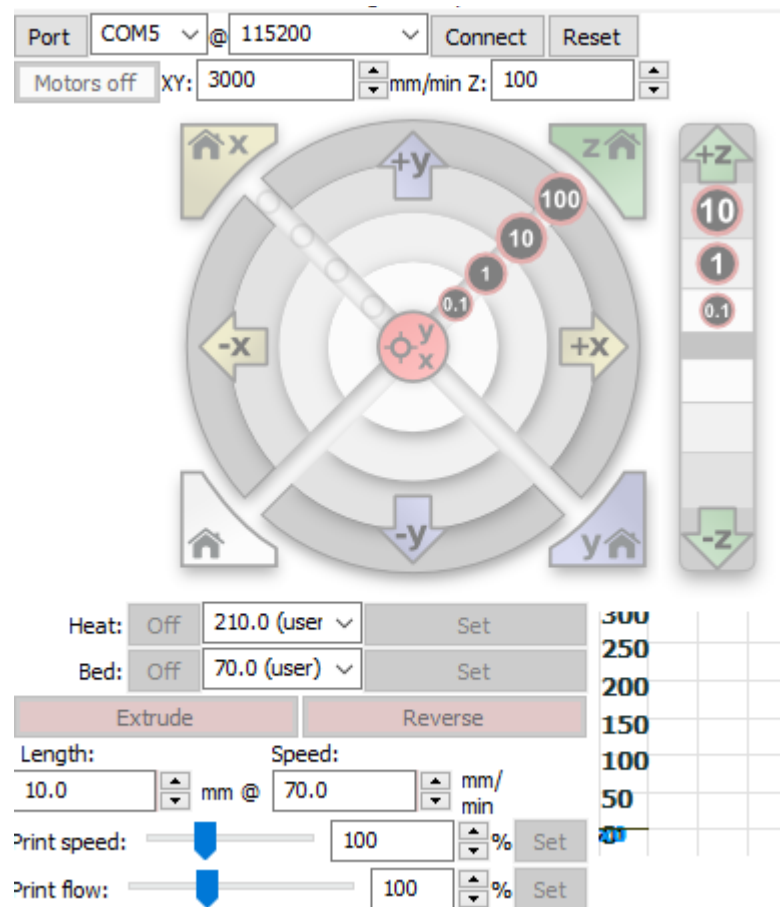
4.4 Calibração e Validação

Esse processo é importante pois através dele é que será garantido a precisão e a qualidade da impressora, e para iniciar iremos primeiramente verificar o deslocamento dos eixos e para verificar a precisão do mesmo. Para facilitar esse processo foi utilizado

o software Prontface que também é um software gratuito que irá facilitar os movimentos de cada eixo.

Segue a tela do software na Figura 67.

Figura 67 - Software Prontface



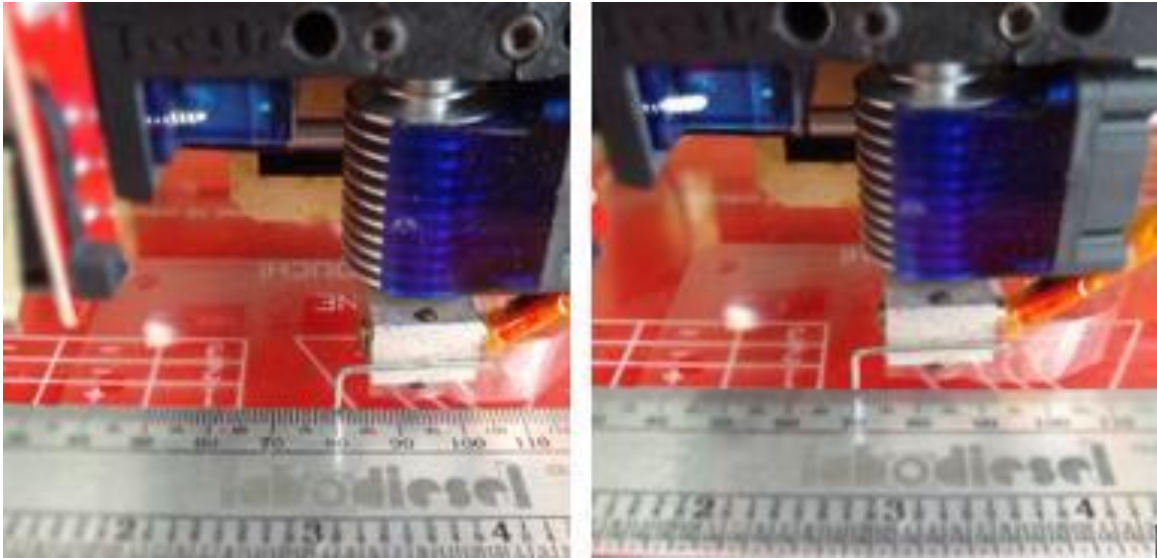
Fonte: Prontface (2023)

Como pode ser observado na Figura acima, há um alvo com movimentos de 0,1 mm, 1 mm, 10 mm e 100 mm. Ao conectar o Prontface, foi selecionado um deslocamento de 10 mm para cada eixo, com o objetivo de verificar se o deslocamento coincide com o solicitado. Caso não coincida, é necessário retornar à seção de configuração de passos por milímetro para realizar ajustes precisos.

Os resultados obtidos em relação ao que foi inserido no firmware estão apresentados nas Figuras 68, 69, 70 e 71. Esses testes são essenciais para garantir que

a impressora 3D esteja calibrada corretamente, assegurando movimentos precisos e consistentes em todos os eixos.

Figura 68 - Resultado calibração eixo X



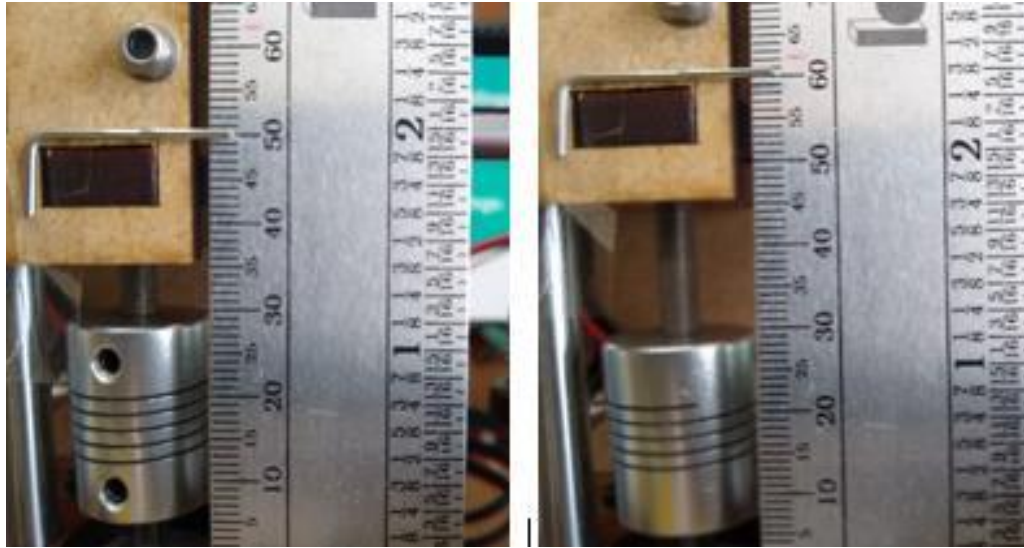
Fonte: Autor (2023)

Figura 69 - Resultado calibração eixo Y



Fonte: Autor (2023)

Figura 70 - Resultado calibração do eixo Z



Fonte: Autor (2023)

Figura 71 - Resultado calibração extrusora



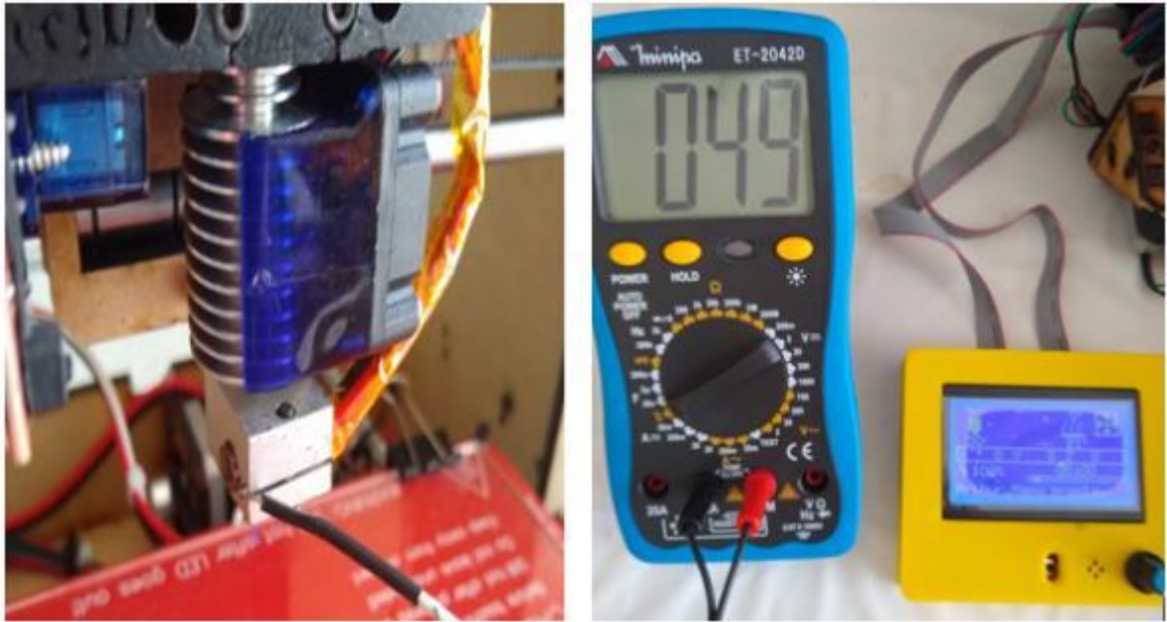
Fonte: Autor (2023)

Com esses resultados pode-se dizer que os valores parametrizados na configuração passos/mm se encontram corretos, mas caso seja necessário alterar os valores é somente aplicar uma regra de três para achar os novos valores. Esse seria o ponto inicial para a calibração e a validação, pois a partir dessa etapa já pode-se colocar a impressora para imprimir os modelos para validação.

Outros pontos também foram aferidos como a temperatura na mesa e no bico, conforme pode ser visto nas Figuras 72 e 73.

A temperatura foi aferida com o auxílio de um multímetro com um termopar tipo K, foi escolhida uma temperatura de 50°C para o hotend e 60°C para a mesa.

Figura 72 - Medição da temperatura no Hotend



Fonte: Autor (2023)

Figura 73 - Medição da temperatura na mesa

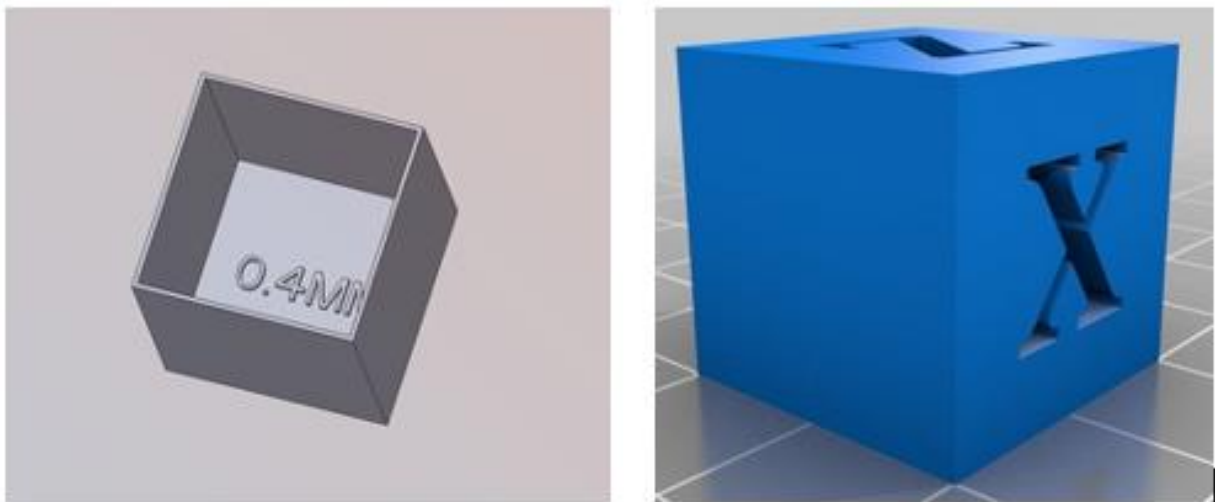


Fonte: Autor (2023)

Para validar as dimensões dos produtos fabricados, foram baixados modelos de calibração específicos para essa verificação. Esses modelos incluem um cubo oco de 20 x 20 mm, utilizado para verificar o volume e a taxa de extrusão, e um cubo denso de 20 x 20 mm, empregado para verificar a calibração dos eixos X, Y e Z.

Ambos os modelos podem ser visualizados na Figura 74 e foram obtidos no site Thingiverse⁴, uma plataforma online popular para compartilhamento de designs de impressão 3D. Esses modelos de calibração são essenciais para garantir a precisão das dimensões das peças impressas.

Figura 74 - Modelos para calibração e validação



Fonte: Autor (2023)

Para impressão foi utilizado o software Cura que é um fatiador gratuito. Após a impressão foram analisados o dimensional das peças conforme Figuras 75 e 76.

Para a Figura 75, pode-se analisar a taxa de extrusão aplicada, onde foi efetuado a medição com auxílio de um paquímetro nos 4 lados tendo como resultado final a média de 0,8mm, considerando que na configuração do material foi solicitado 2 filetes por parede e que o bico é de 0,4mm o resultado será de 0,8mm, com isso garante que a configuração dos Steps/mm foi configurada corretamente.

⁴ <https://www.thingiverse.com/>

Figura 75 - Verificação parede do cubo (taxa de extrusão)



Fonte: Autor (2023)

Para verificar dimensional conforme o cubo 20x20mm, feito a impressão foi efetuado a aferição com auxílio de um paquímetro. A Figura 76 mostra os resultados.

Figura 76 - Verificação dimensional do cubo X, Y e Z



Fonte: Autor (2023)

Pode se verificar que os valores estão dentro da especificação para os 3 eixos 20mm e como observação essas impressões foram realizadas com a mesa nivelada manualmente.

O próximo passo para finalização da impressora, será inserção ao IoT que hoje atualmente é uma utilização tanto em ambiente industriais quanto em ambientes escolares.

4.5 Instalação do acesso ao IoT – Octoprint

Para aprimorar as capacidades da impressora, foi integrado um microcontrolador Raspberry Pi 3B, conhecido por seu alto poder de processamento. A presença desse componente possibilita o acesso remoto à impressora por meio da internet, tanto via computador quanto por dispositivos móveis. Isso permite o envio de comandos, o carregamento de arquivos para impressão e a visualização do progresso do trabalho por meio de imagens. Além disso, o usuário pode monitorar possíveis erros e receber notificações quando a impressão é concluída. O Raspberry Pi foi instalado com o software Octoprint, uma solução de servidor de impressão de código aberto. Uma de suas principais vantagens é a capacidade de acessar remotamente até quatro impressoras 3D, independentemente de estarem equipadas com placas Arduino ou não.

Esse software pode ser baixado no site Octoprint⁵ ou pelo próprio site do raspberry foundation ⁶, que pode ser baixado a imagem para a instalação no raspberry.

Optou-se por baixar a imagem do Octoprint diretamente no site da raspberry.

Pode ser visto na Figura 77 a tela para instalar o sistema no cartão de memória SD.

⁵ <https://octoprint.org/>

⁶ <https://www.raspberrypi.com/software/>

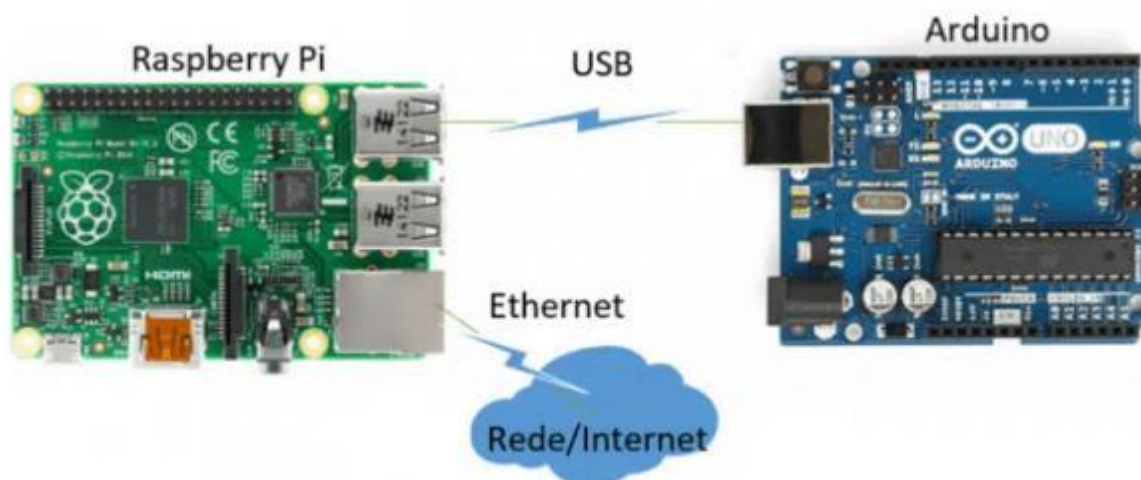
Figura 77 - Tela de instalação do S.O do Raspberry e Octopi



Fonte: Raspberry Pi

Após baixar a imagem, esta foi transferida para um cartão de memória SD e, posteriormente, inserida no Raspberry Pi para iniciar o sistema. A integração entre o Raspberry Pi e o Arduino será estabelecida por meio de uma conexão USB, enquanto a conectividade com a internet poderá ser realizada tanto por meio de uma rede cabeada quanto por uma rede sem fio (Wi-Fi), conforme ilustrado na Figura 78.

Figura 78 - Conexão Raspberry PI3 com arduino e rede internet



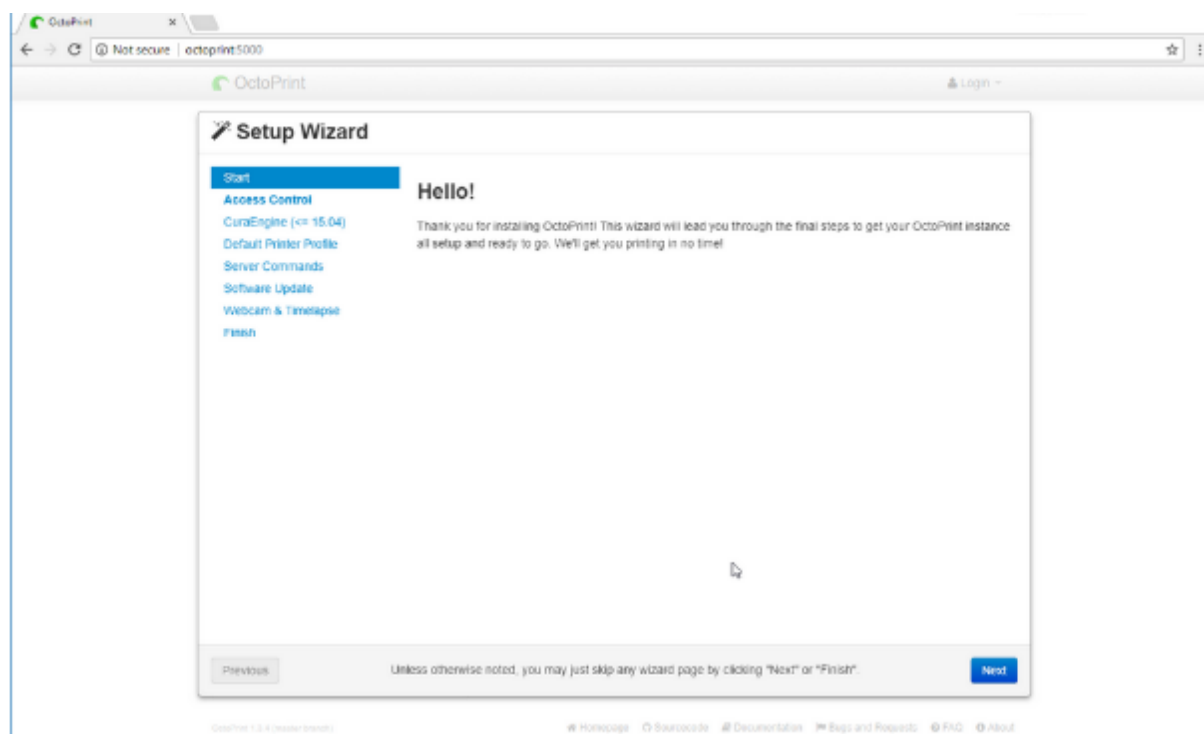
Fonte: [Embarcados.com.br/sistema-web-com-raspberry-pi-e-arduino/](https://embarcados.com.br/sistema-web-com-raspberry-pi-e-arduino/) (2023)

Após a interconexão do Raspberry com o Arduino, torna-se necessário alimentar ambos os dispositivos e permitir que o sistema seja iniciado automaticamente.

Para configurar o Octoprint e acessar a impressora remotamente, é essencial conhecer o endereço IP do Raspberry Pi. Existem duas maneiras de obtê-lo: conectando um monitor à entrada HDMI do Raspberry e assim visualizar o número IP durante o processo de inicialização, ou utilizando um aplicativo para identificar dispositivos conectados à rede, como o Fing⁷, por exemplo.

Com o endereço IP do Raspberry em mãos, basta digitá-lo no navegador do seu computador para acessar e configurar o sistema, como mostrado na Figura 79. Esse passo é crucial para garantir o controle e monitoramento eficaz da impressora 3D à distância.

Figura 79 - Tela de configuração inicial do Octoprint



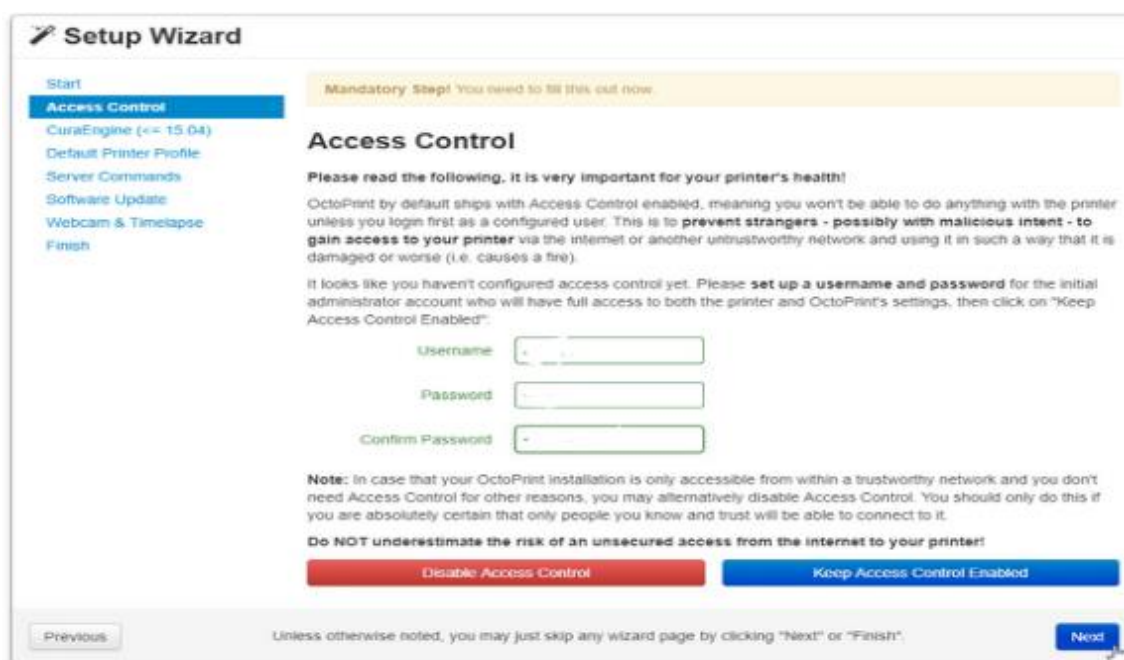
Fonte: Autor (2023)

⁷ <https://www.fing.com/> (2023)

Esta é a interface inicial (Figura 80) de configuração do Octoprint, onde todas as personalizações essenciais para o funcionamento remoto correto são realizadas.

A primeira etapa envolve a criação de uma senha de acesso, uma medida fundamental para evitar acessos não autorizados à sua impressora. Sem essa senha, qualquer pessoa poderia acessar indevidamente com o sistema. Você pode visualizar essa configuração na Figura 80. Estabelecer uma senha robusta é crucial para garantir a segurança e a privacidade das operações de impressão à distância.

Figura 80 - Tela de configuração de senha no Octoprint



Setup Wizard

Start

Access Control

CuraEngine (<= 15.04)

Default Printer Profile

Server Commands

Software Update

Webcam & Timelapse

Finish

Mandatory Step! You need to fill this out now.

Access Control

Please read the following, it is very important for your printer's health!

OctoPrint by default ships with Access Control enabled, meaning you won't be able to do anything with the printer unless you login first as a configured user. This is to **prevent strangers - possibly with malicious intent - to gain access to your printer** via the internet or another untrustworthy network and using it in such a way that it is damaged or worse (i.e. causes a fire).

It looks like you haven't configured access control yet. Please **set up a username and password** for the initial administrator account who will have full access to both the printer and OctoPrint's settings, then click on "Keep Access Control Enabled":

Username:

Password:

Confirm Password:

Note: In case that your OctoPrint installation is only accessible from within a trustworthy network and you don't need Access Control for other reasons, you may alternatively disable Access Control. You should only do this if you are absolutely certain that only people you know and trust will be able to connect to it.

Do NOT underestimate the risk of an unsecured access from the internet to your printer!

Previous Next

Unless otherwise noted, you may just skip any wizard page by clicking "Next" or "Finish".

Fonte: Autor (2023)

Na etapa seguinte da configuração, existe a opção de enviar informações anônimas para a correção de eventuais falhas no sistema. Neste caso específico, essa opção foi habilitada, como pode ser visto na Figura 81. Esse recurso permite contribuir para o aprimoramento contínuo do software ao compartilhar dados de maneira anônima para identificação e resolução de possíveis bugs.

Figura 81 - Tela envio de informações de Bugs



Fonte: Autor (2023)

Na próxima etapa de configuração, é avaliada a necessidade de realizar um teste de conectividade com a internet. Neste caso específico, essa opção foi desabilitada. A razão para essa escolha reside no fato de que, uma vez que o arquivo tenha sido carregado na impressora via internet, se a conexão cair, a impressão continuará sem ser interrompida. Pode-se visualizar essa configuração na Figura 82. Essa decisão visa garantir a continuidade do processo de impressão, independentemente de eventuais interrupções na rede.

Figura 82 - Tela de checagem de conexão



Fonte: Autor (2023)

Na etapa seguinte da configuração, foi ativado um bloqueio para evitar a instalação de plugins indesejáveis ao abrir o Octoprint. Essa medida impede que plugins não autorizados sejam instalados sem a permissão do usuário. Você pode ver essa configuração na Figura 83. Essa precaução ajuda a manter a integridade e a segurança do sistema, garantindo que apenas os plugins aprovados e seguros sejam utilizados.

Figura 83 - Tela de configuração de instalação de plugin

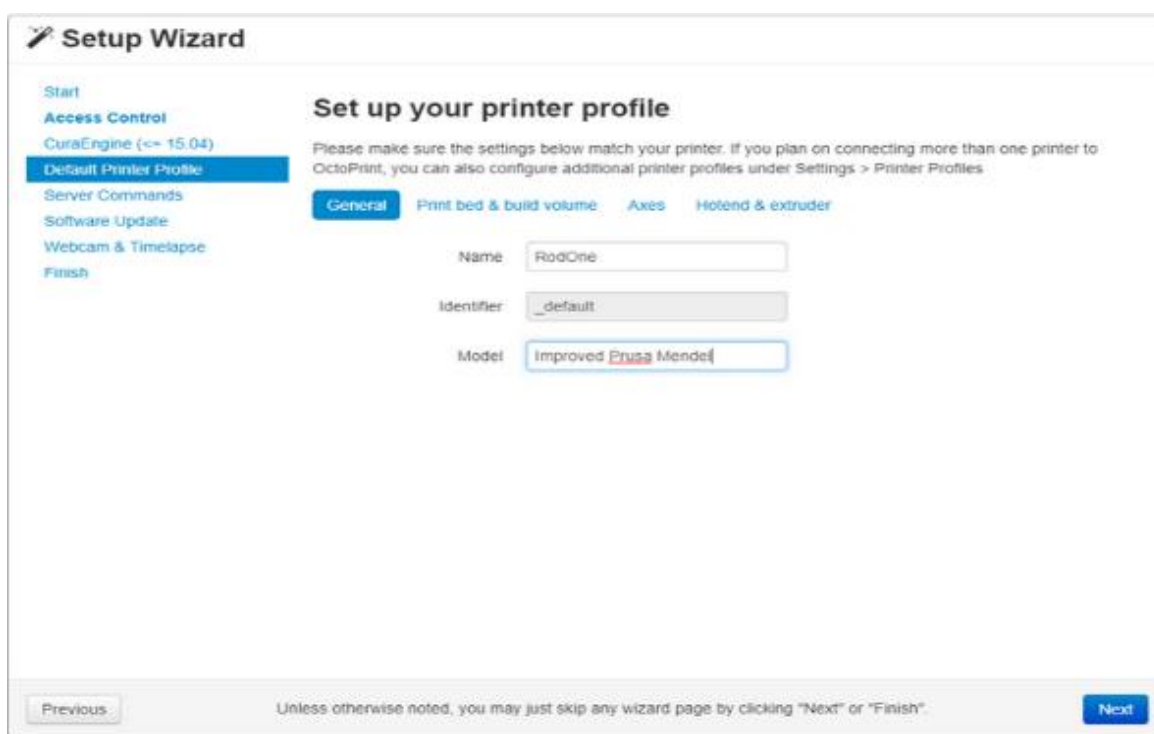


Fonte: Autor (2023)

Finalmente, na última etapa da configuração, são inseridos os dados da impressora, incluindo o nome da impressora e, no caso de múltiplas impressoras, uma identificação individual para cada uma. Essas informações são cruciais para a identificação na rede. Além disso, são configuradas as dimensões da mesa de impressão e o número de hotends utilizados, bem como o diâmetro da saída do bico do hotend.

Esses parâmetros são fundamentais para garantir que a impressora funcione de maneira precisa e eficiente. Essas configurações podem ser visualizadas nas Figuras 84, 85 e 86, respectivamente. Esses detalhes são essenciais para personalizar o funcionamento da impressora de acordo com as necessidades específicas do usuário.

Figura 84 - Tela do nome da impressora



The screenshot shows the 'Setup Wizard' interface for OctoPrint. The main heading is 'Set up your printer profile'. Below this, there is a navigation menu on the left with options: Start, Access Control, CuraEngine (<- 15.04), Default Printer Profile (highlighted), Server Commands, Software Update, Webcam & Timelapse, and Finish. The main content area has a sub-heading 'Set up your printer profile' and a note: 'Please make sure the settings below match your printer. If you plan on connecting more than one printer to OctoPrint, you can also configure additional printer profiles under Settings > Printer Profiles'. There are four tabs: 'General' (selected), 'Print bed & build volume', 'Axes', and 'Hotend & extruder'. The 'General' tab contains three input fields: 'Name' with the value 'RodOne', 'Identifier' with the value '_default', and 'Model' with the value 'Improved Prusa Mendel'. At the bottom, there is a 'Previous' button on the left, a 'Next' button on the right, and a footer note: 'Unless otherwise noted, you may just skip any wizard page by clicking "Next" or "Finish".'

Fonte: Autor (2023)

Figura 85 - Tela de configuração do volume de impressão da impressora.

Setup Wizard

- Start
- Access Control
- CuraEngine (<= 15.04)
- Default Printer Profile**
- Server Commands
- Software Update
- Webcam & Timelapse
- Finish

Set up your printer profile

Please make sure the settings below match your printer. If you plan on connecting more than one printer to OctoPrint, you can also configure additional printer profiles under Settings > Printer Profiles

General **Print bed & build volume** Axes Hotend & extruder

Form Factor Rectangular Circular

Origin

Heated Bed

Please define the print volume.

Width (X) mm

Depth (Y) mm

Height (Z) mm

If your printer's print head may move slightly outside the print volume (e.g. for nozzle-cleaning routines) you can define a custom safe bounding box for its movements below.

Custom bounding box

Fonte: Autor (2023)

Figura 86 - Tela de configuração do bico

Setup Wizard

- Start
- Access Control
- CuraEngine (<= 15.04)
- Default Printer Profile**
- Server Commands
- Software Update
- Webcam & Timelapse
- Finish

Set up your printer profile

Please make sure the settings below match your printer. If you plan on connecting more than one printer to OctoPrint, you can also configure additional printer profiles under Settings > Printer Profiles

General **Print bed & build volume** Axes **Hotend & extruder**

Nozzle Diameter mm

Number of Extruders

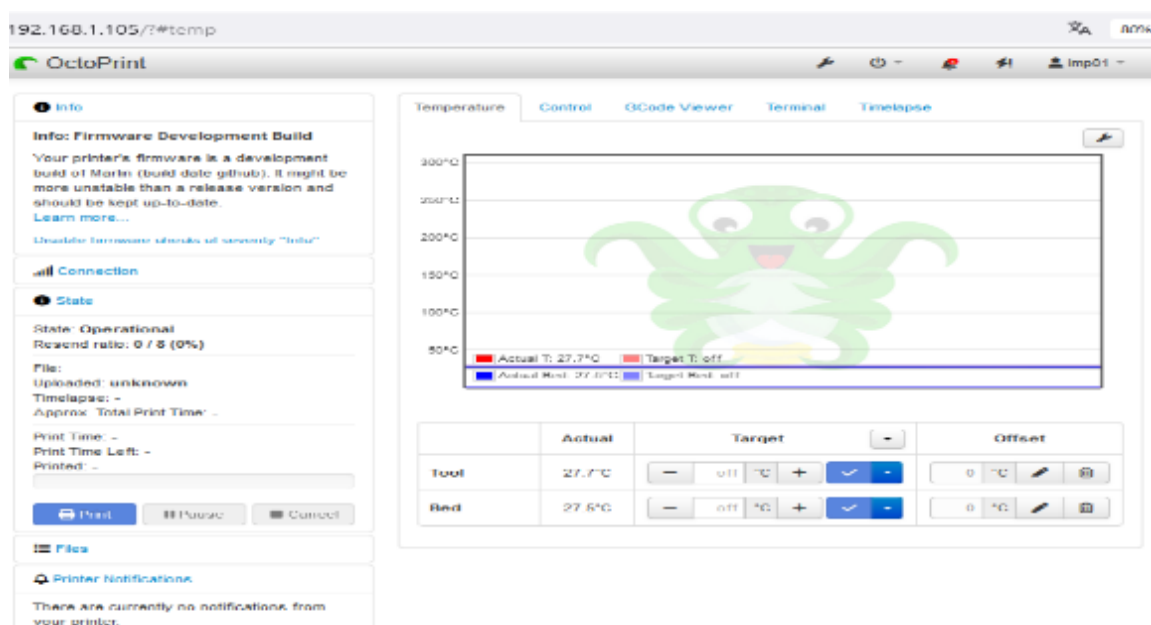
This information is used for the graph and controls available in the "Temperature" tab, the GCODE viewer and when slicing from within OctoPrint. It does NOT influence already sliced files that you upload to OctoPrint!

Previous Unless otherwise noted, you may just skip any wizard page by clicking "Next" or "Finish". **Next**

Fonte: Autor (2023)

Após concluir a configuração do Octoprint, a tela inicial apresentará o layout conforme mostrado na Figura 87. Nessa interface, é possível monitorar e controlar a impressora remotamente. Funcionalidades como o envio de arquivos para impressão e ajuste de temperatura podem ser realizadas de forma conveniente e eficaz por meio dessa plataforma. Essa interface intuitiva oferece ao usuário total controle sobre o processo de impressão, proporcionando uma experiência prática e simplificada.

Figura 87 - Tela do octoprint - Aba temperatura

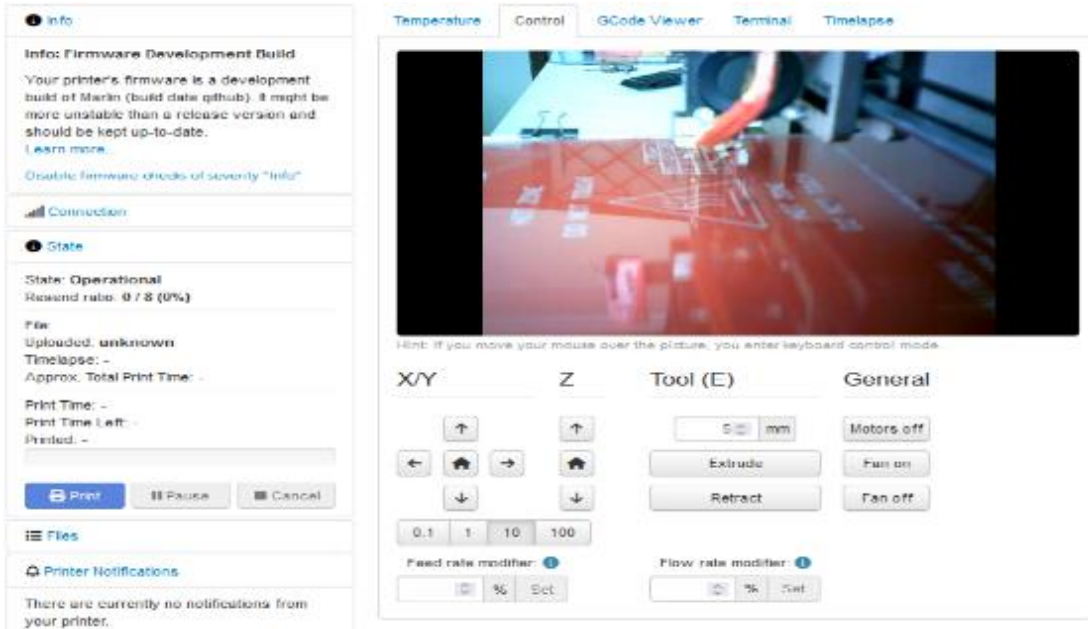


Fonte: Autor (2023)

Nesta tela, é possível monitorar a temperatura por meio de gráficos ou definir uma temperatura específica. Além disso, é possível identificar falhas, verificar qual arquivo está sendo impresso ou qual está na lista de impressões, e pausar ou cancelar uma impressão remotamente.

Outra opção oferecida é a capacidade de verificar e controlar a impressora em tempo real. Isso inclui a visualização em tempo real do que está ocorrendo durante o processo de impressão. Se necessário, é possível movimentar os eixos da impressora remotamente para verificar possíveis falhas ou ajustar o posicionamento, como ilustrado na Figura 88. Esse controle em tempo real proporciona uma supervisão detalhada, permitindo intervenções imediatas caso ocorram problemas durante a impressão, para o trabalho foi implementada uma webcam UBS para efetuar essa visualização.

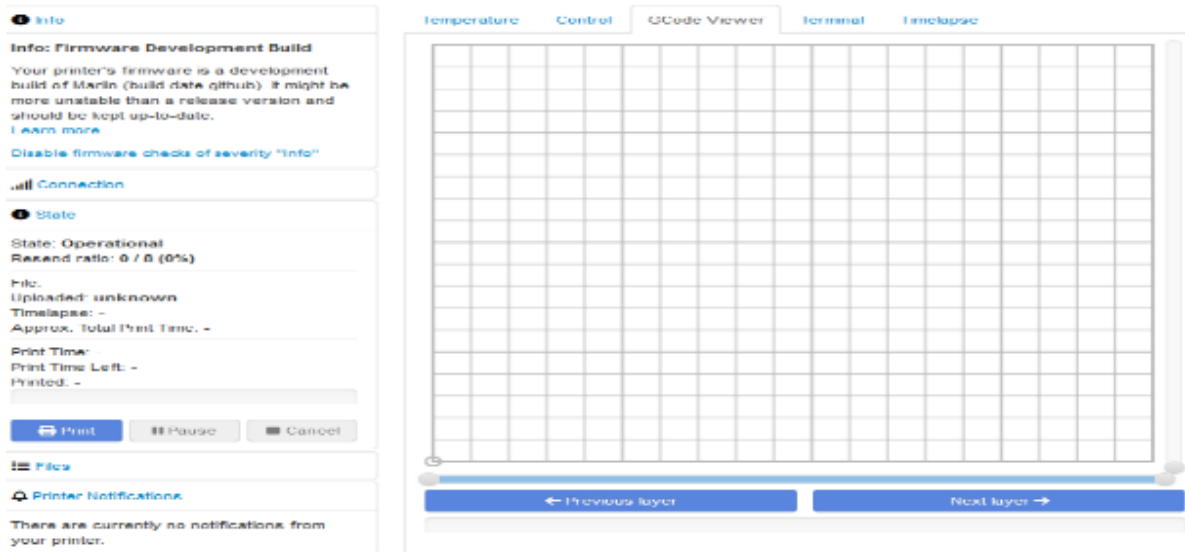
Figura 88 - Tela octoprint - Imagem remota da área de impressão.



Fonte Autor (2023)

Conforme mostrado na Figura 89, é possível acessar o G-code, permitindo visualizar em qual camada (layer) a impressão se encontra. Essa funcionalidade fornece informações detalhadas sobre o progresso da impressão, permitindo um acompanhamento preciso do processo em diferentes estágios.

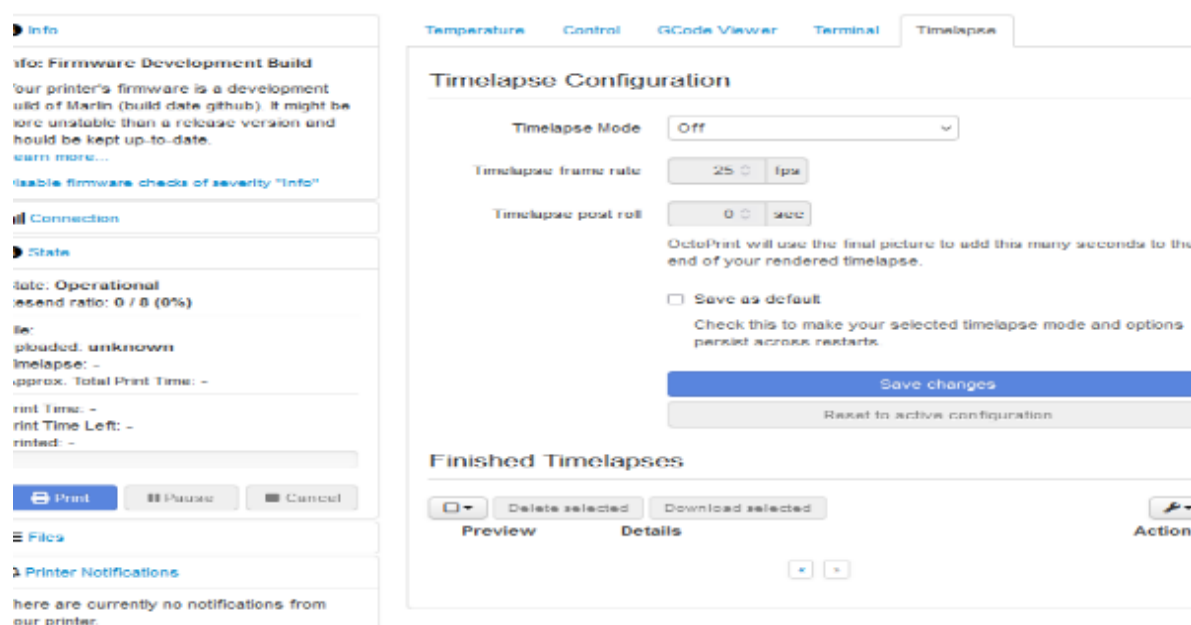
Figura 89 - Tela de acompanhamento das Layers do produto



Fonte: Autor (2023)

O Octoprint oferece outra opção valiosa: a capacidade de gravar suas impressões. Isso permite registrar o processo de impressão para uma revisão posterior em caso de falhas ou para apresentar ao cliente, conforme ilustrado na tela da Figura 90. Essa funcionalidade proporciona uma maneira eficaz de documentar e revisar cada impressão, facilitando a análise de eventuais problemas e garantindo a qualidade do resultado final.

Figura 90 - Tela de gravação da impressão.



Fonte: Autor (2023)

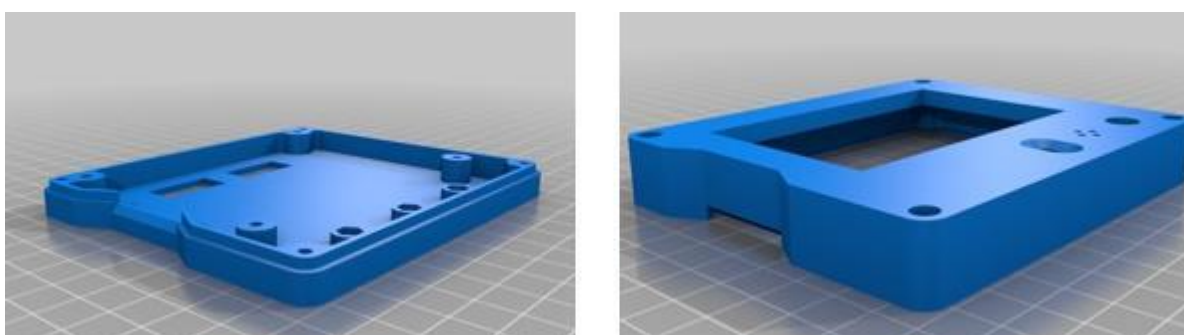
5 RESULTADOS

Após a conclusão dos processos de calibração e validação, foram produzidos diversos objetos como prova do funcionamento adequado da impressora 3D.

Para avaliar a precisão da impressora, foram impressos vários modelos obtidos no site da Thingiverse. Esses modelos serviram como referência para a avaliação da montabilidade e estética das peças impressas. Entre os modelos impressos, destaca-se o case projetado para acomodar o display, conforme ilustrado na Figura 91.

Esses testes práticos foram feitos para validar não apenas a funcionalidade da impressora 3D, mas também para assegurar que as peças impressas atendam a qualidade desejada.

Figura 91 -Modelo Case para o display 128x64



Fonte: amacedo on Thingiverse: <https://www.thingiverse.com/thing:2724633> (2023)

Na Figura 92 pode ser visto o modelo impresso.

Figura 92 - Case impresso - utilizando filamento Amarelo



Fonte: Autor (2023)

Outros objetos também foram impressos como o suporte de apoio do filamento, guia do filamento, esses materiais foram utilizados na impressora, por isso o conceito RepRAp.

Na Figura 93 pode ser visto essas impressões.

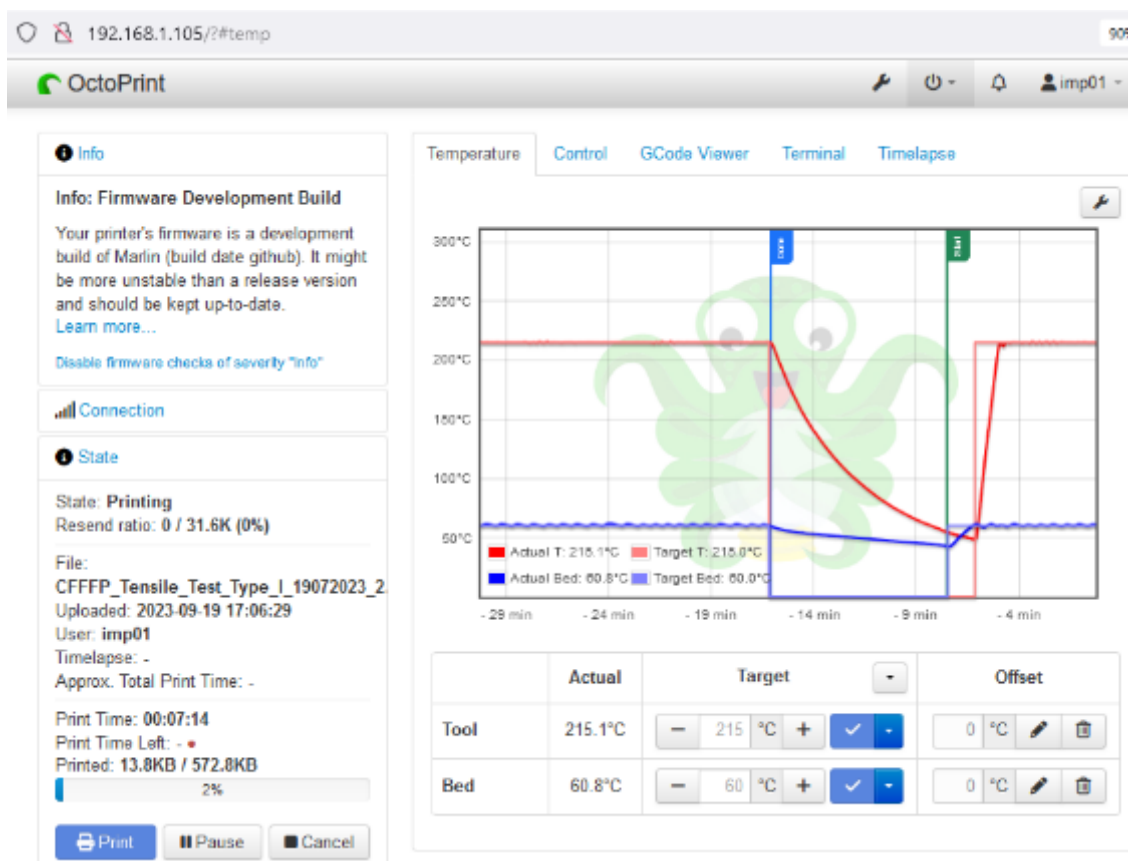
Figura 93 - Suporte e o guia do filamento



Fonte: Autor (2023)

Para realizar os testes de conectividade, foi utilizado o software OctoPrint. Através dessa interface, foi possível enviar o código para a impressora remotamente, monitorar o progresso da impressão usando o visualizador G-Code Viewer e observar o comportamento da peça na mesa por meio de uma imagem, tudo de forma remota. Nas Figuras 94, 95 e 96 pode-se visualizar um exemplo de impressão sendo realizada.

Figura 94 - Tela de monitoramento da impressão

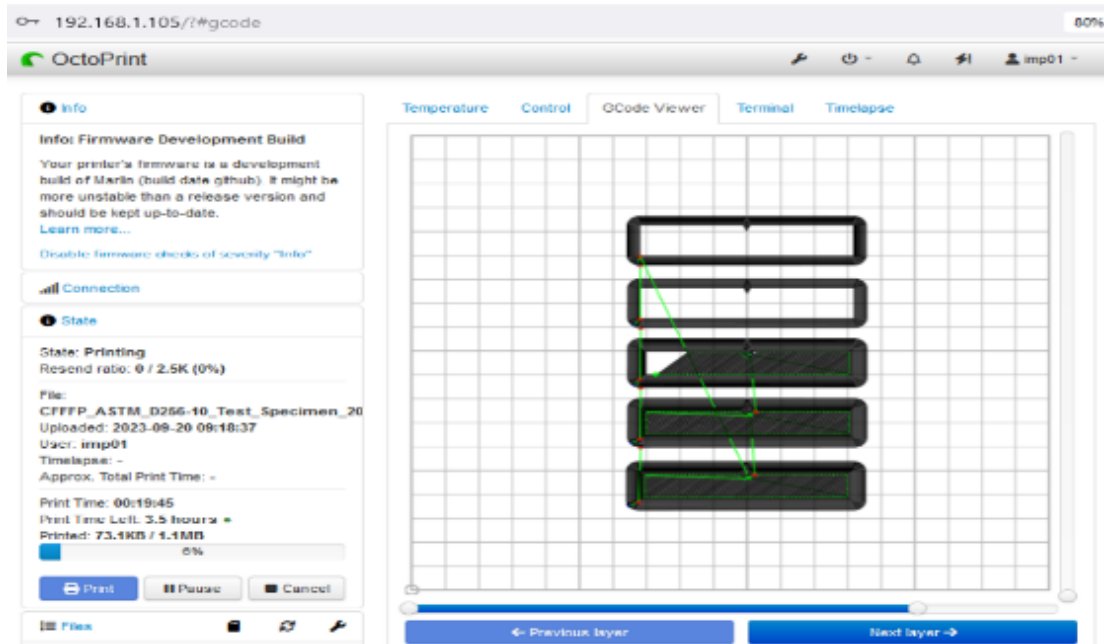


Fonte: Autor (2023)

Na Figura 94, pode ser visto todos os dados durante a impressão, como a temperatura da mesa e do hotend, através de gráfico, é possível inclusive alterar se for necessário. Outra verificação que pode ser visto é o tempo de impressão e o percentual para término da impressão e o nome do produto que está sendo impresso, isso de forma totalmente remota como mostra o endereço IP na parte superior da imagem “192.168.1.105”.

Na Figura 95 pode ser verificada o Gcode Viewer, que é possível verificar em qual layer está o objeto que está sendo impresso.

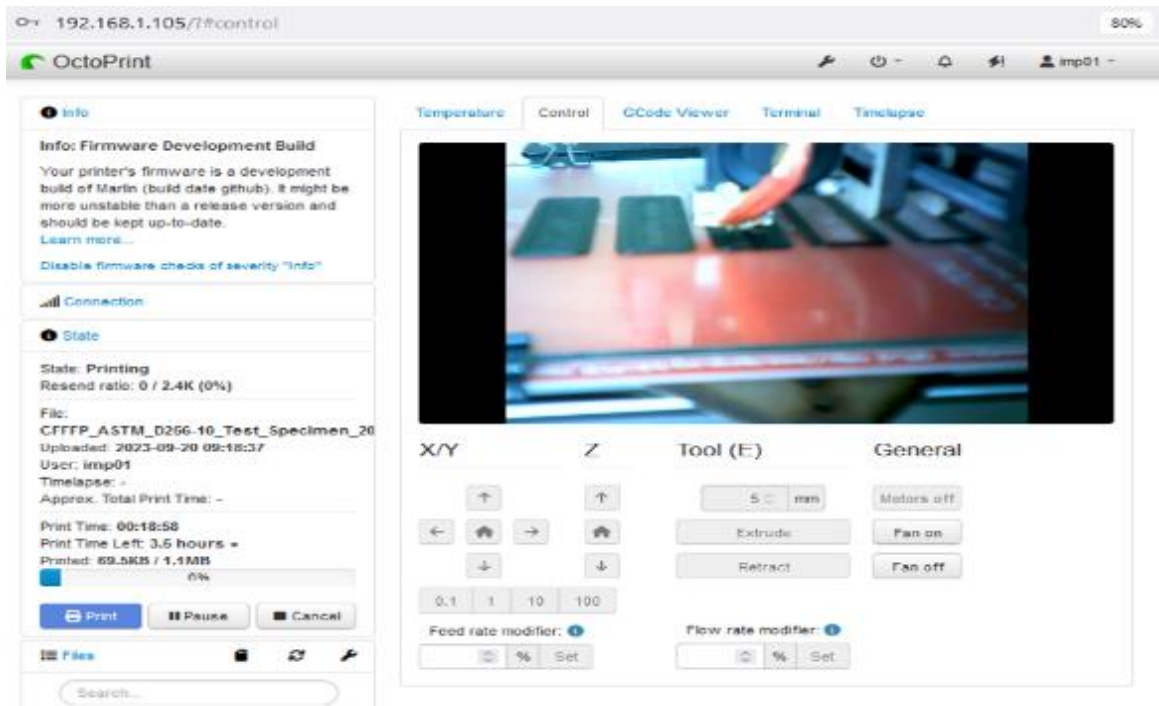
Figura 95 - G-Code Viewer - Layer de impressão



Fonte: Autor (2023)

Na Figura 96, pode ser visto a imagem mostrando o comportamento do objeto na mesa.

Figura 96 - Monitoramento através da imagem

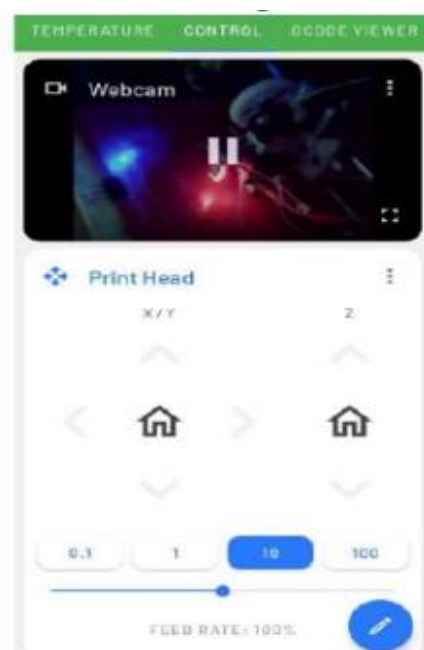


Fonte: Autor (2023)

Esse sistema permite também o monitoramento através do celular, podendo também ser enviado o código a partir dele.

Na Figura 97, pode ser visto o monitoramento pelo celular, através do octoremote.

Figura 97 - Tela de monitoramento através do celular



Fonte: Autor (2023)

Através dessas verificações pode ser visto que os testes com a conexão remota funcionam perfeitamente e com isso pode ser visto que é possível estar empregando o conceito do IoT.

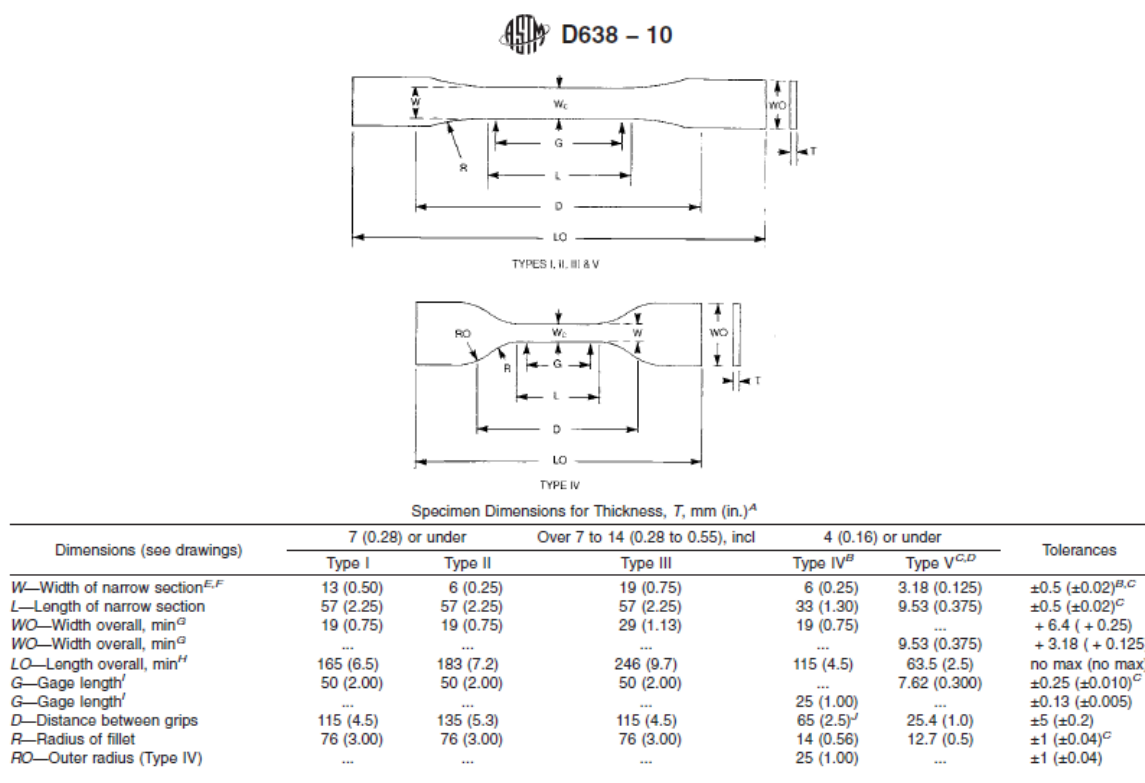
Para o próximo resultado foram impressos 20 corpos de prova para ensaios de tração e 3 corpos de prova para ensaios de dureza, para verificar se houve qualquer influência no material que foi utilizado como insumo (filamento). As análises feitas foram em cima de padrões específicos a fim de verificar a alteração nos materiais, sendo o PLA normal do fabricante 3Dfila e o PLA Ht impact do fabricante 3dPrime. Os corpos de prova que foram impressos para ensaio, se baseiam nas normas ASTM638 e ASTM D 2240.

A norma ASTM638 descreve o ensaio de tração aplicado a materiais plásticos. Durante esse ensaio, são medidas diversas características de tração, incluindo o módulo de tração, a tensão de escoamento, a deformação no ponto de escoamento, a resistência à tração e a deformação na ruptura. O ensaio de tração é amplamente utilizado na

caracterização mecânica dos polímeros. Consiste em aplicar uma carga uniaxial crescente à amostra, ao mesmo tempo em que são registradas as variações em suas dimensões.

O padrão utilizado para esse teste foi o Padrão 1, conforme mostra a Figura 98. A norma recomenda no mínimo 5 corpos de prova e foram disponibilizados 20 corpos de prova para essa verificação com variações de impressão.

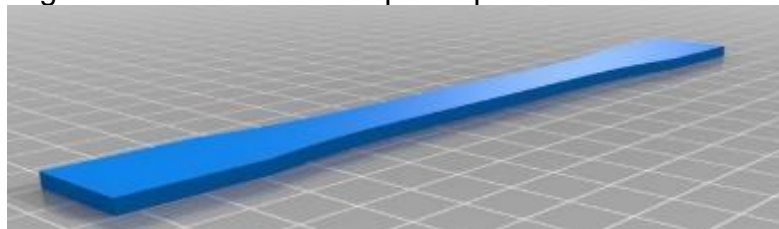
Figura 98 - Especificações referente aos corpos de prova ASTM638-10



Fonte: Norma ASTM – Designação 638-10

O modelo para esse corpo de prova foi baixado no site thinkverse, conforme mostrado na Figura 99.

Figura 99 - Modelo 3D corpo de prova ASTM-D638-10



Fonte: Znanzhu on Thingiverse: <https://www.thingiverse.com/thing:2332080>

Foi utilizado o ensaio de Dureza Shore para medir a resistência à penetração. Após a aplicação de uma carga localizada, é realizada uma análise da resistência à indentação do polímero em questão. Para avaliar a dureza em polímeros, recomenda-se o ensaio de dureza Shore, conforme estipulado pela norma ASTM D 2240 – *Standard Test Method for Rubber Property - Durometer Hardness*. Esta norma apresenta variações específicas para cada tipo de material, sendo que, para termoplásticos, borrachas e plásticos mais rígidos, a escala D é amplamente empregada (Carneiro et al., 2019).

De acordo com a norma ASTM D 2240, a dureza Shore D pode variar de 0 a 100, sendo os valores mais elevados associados a maiores durezas. As amostras devem ser planas, com espessura mínima de 6 mm. Quanto ao ponto de medição, deve estar a pelo menos 6 mm das bordas da amostra, com uma distância de 6 mm entre cada ponto de medição (recomenda-se um mínimo de 5 medições). A carga aplicada é de 5 kgf.

Os corpos de prova, em formato quadrado, foram impressos com dimensões de 40 x 40mm e 6 mm de espessura, sendo produzidos três corpos de prova. Foram conduzidas cinco medições em cada amostra de PLA, foi utilizado somente o PLA sem tratamento térmico considerando 20% de preenchimento.

5.1 Ensaio de tração

Para os ensaios de tração, os corpos de prova (CP) foram impressos horizontalmente para facilitar a fabricação, pois essa orientação proporciona uma maior área de contato com a mesa de impressão.

Uma abordagem clássica para representar os resultados do ensaio de tração é por meio de um gráfico ou curva de tensão versus deformação. Essa curva é obtida através da medição dos valores da força aplicada e da deformação sofrida pelo material em cada momento, até a ruptura da amostra. Em outras palavras, é uma representação gráfica do comportamento do material durante todo o ensaio. As análises apresentadas baseiam-se em diagramas de tensão versus deformação e gráficos de colunas, com os dados expressos em MPa (tensão) e em porcentagem de deformação.

Os corpos para ensaios de tração foram divididos da seguinte maneira:

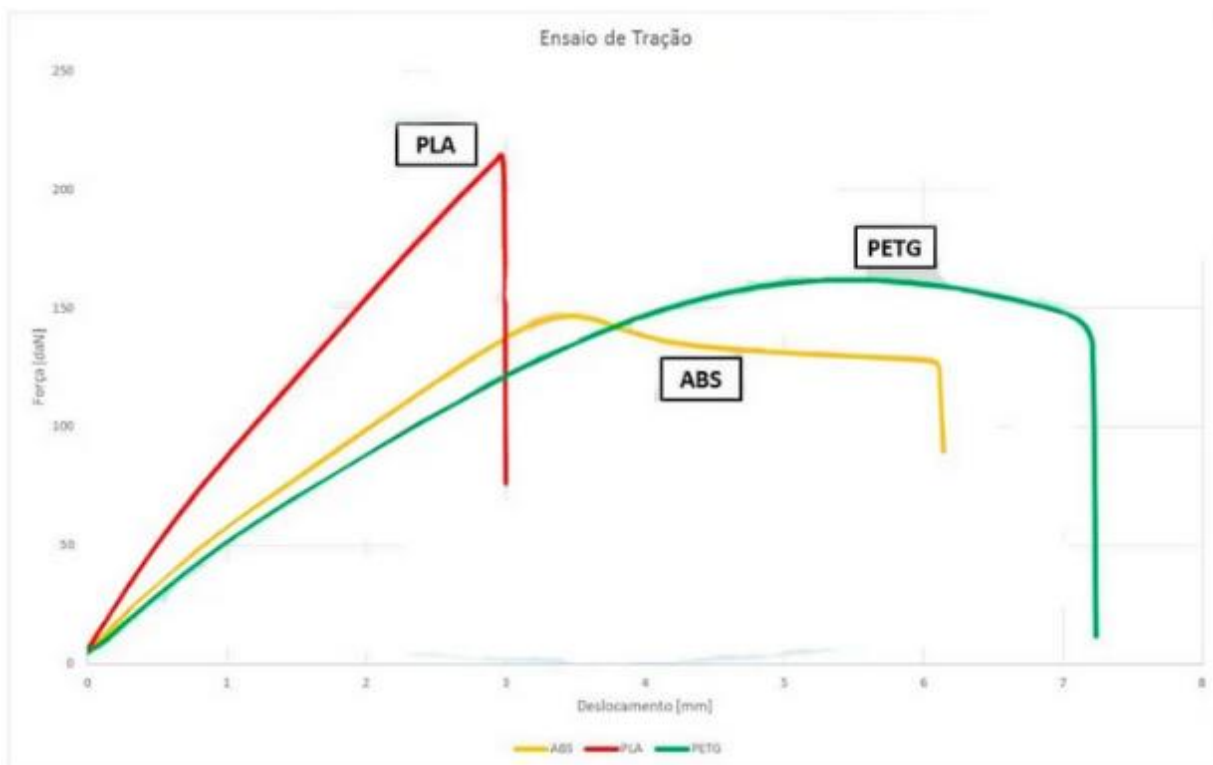
- 5 corpos com preenchimento 20% orientação de impressão 0/90°
- 5 corpos com preenchimento 20% orientação de impressão 0/90° com tratamento térmico (recozimento).
- 5 corpos com preenchimento 100% orientação de impressão 0/90°
- 5 corpos com preenchimento 100% orientação de impressão 0/90° com tratamento térmico (recozimento).

Foram também conduzidos testes com corpos de prova submetidos a tratamento térmico (recozimento), nos quais os corpos foram colocados em um forno e mantidos a uma temperatura de 100°C por 30 minutos, com o intuito de avaliar a influência no teste de tração.

Após a realização dos testes, os resultados foram analisados e comparados com dados provenientes de artigos e dissertações já existentes. Posteriormente, será estabelecida uma correlação entre os testes realizados e as análises de artigos e dissertações disponíveis.

Conforme mencionado anteriormente, estes testes têm como objetivo realizar uma comparação da influência no insumo (PLA) no contexto da construção de uma impressora para fins acadêmicos. Na Figura 100 apresenta como orientação o gráfico tensão-deformação de diversos polímeros fornecido pelo fabricante de filamento PLA, 3dlab.

Figura 100 - Comparativo de teste de tração nos polímeros PLA, ABS e PETG

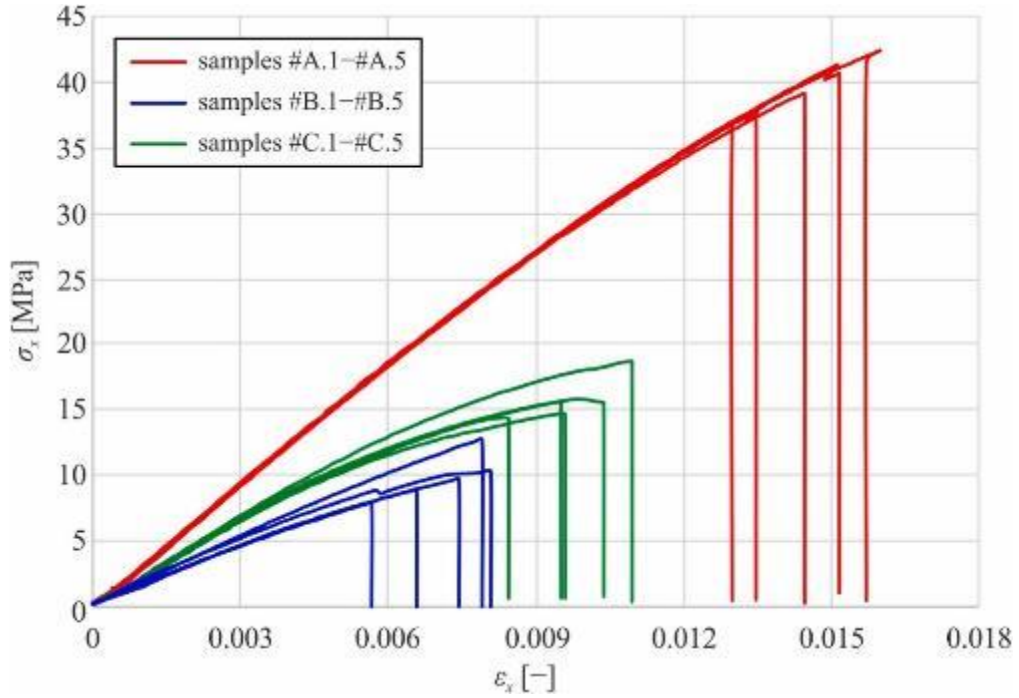


Fonte: <https://3dlab.com.br/propriedades-dos-materiais-para-impressora-3d/>, 2017

Nas Figuras 101 e 102, foi apresentada uma comparação entre um teste conduzido e documentado no artigo (Sabik, A et al, 2022) e o teste realizado com os corpos de prova produzidos na impressora construída. Os procedimentos dos testes realizados seguiram como comparação, ou seja, a nível de semelhança entre as curvas dos corpos de prova feitos na impressa. As comparações foram feitas com as amostras impressas com preenchimento em 20% e em 100%, mantendo a mesma direção de preenchimento 0/90°.

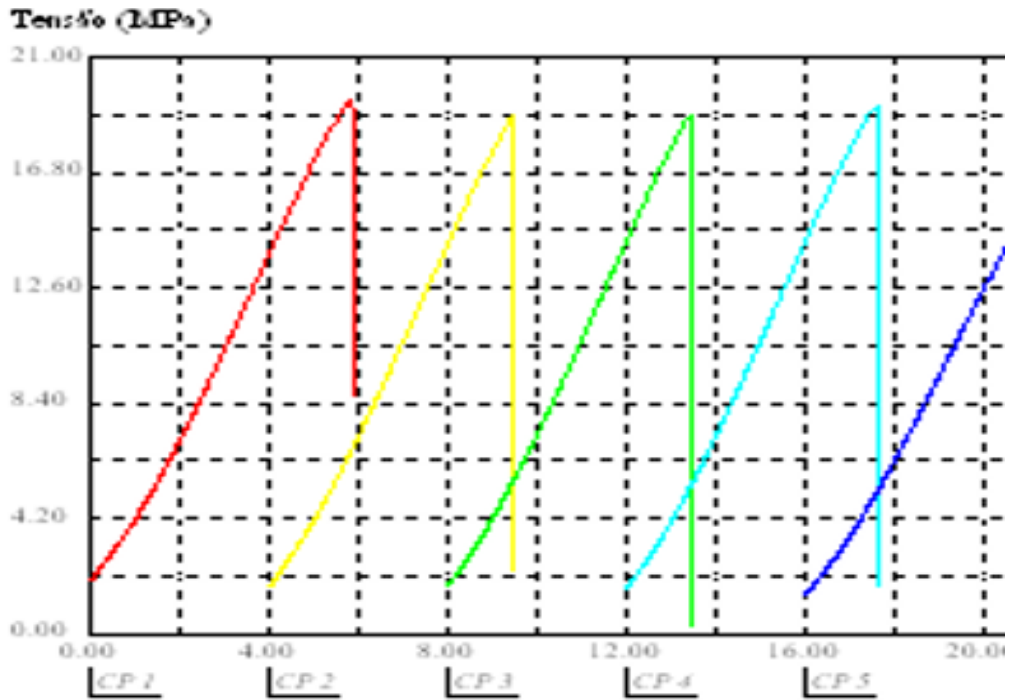
Um aspecto adicional incluído em nosso teste é a utilização de PLA com tratamento, tanto com 20% quanto com 100% de preenchimento, permitindo uma comparação entre os materiais.

Figura 101 - Gráfico tração x deformação PLA espessura de camada 0,4mm



Fonte: Sabik, A et al, 2022).

Figura 102 - Gráfico tensão x deformação PLA p 20%, espessura de camada de 0,4mm

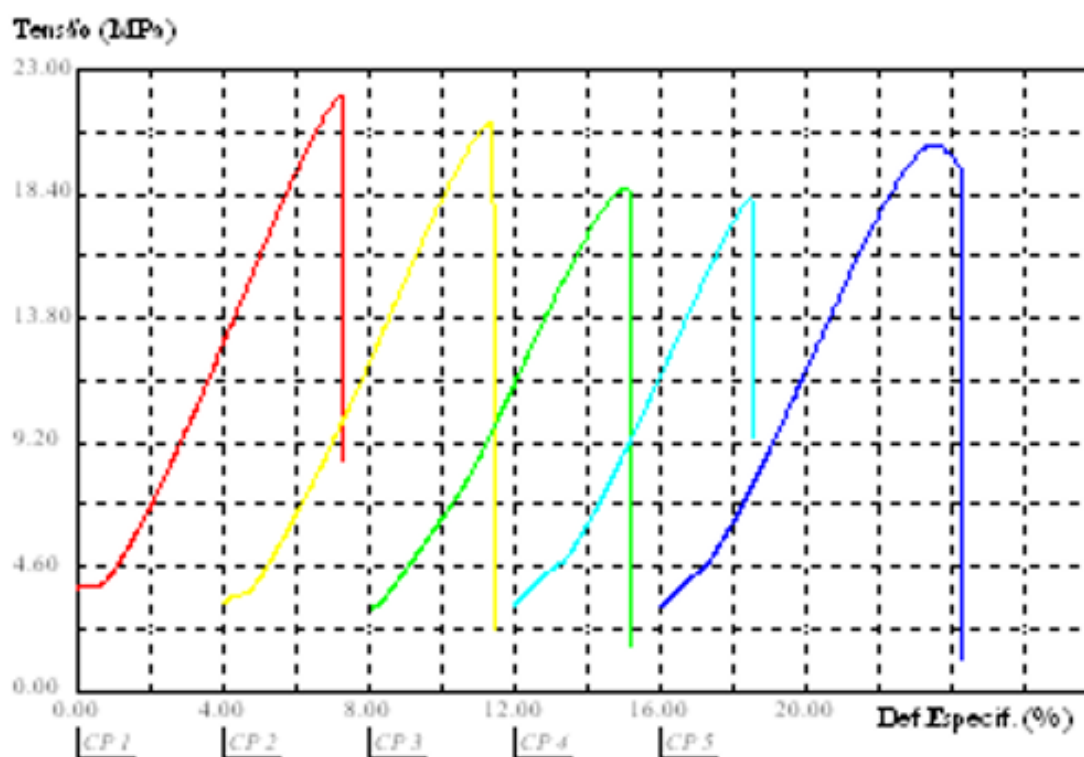


Fonte: Autor (2023)

Observa-se uma semelhança no comportamento do material entre a curva e os valores representados no gráfico, onde a tensão máxima suportada é de 20MPa para uma deformação média de 6,0%, considerando os 5 corpos de prova e apresentando um desvio padrão entre as amostras de 1,105MPa. Em comparação ao gráfico do artigo, pode ser visto que este resultado está próximo ao comparado com as amostras C1 a C5, que se encontra com um valor de 19MPa.

Na Figura 103, encontra-se outro corpo de prova com características de impressão idênticas, contudo, foi submetido a um tratamento térmico (recozimento) em forno, mantendo uma temperatura de 100°C por 30 minutos. Este teste foi realizado para verificar e comparar dois materiais produzidos pela mesma impressora, analisando o comportamento do PLA com esse tratamento, exclusivamente para fins de comparação.

Figura 103 - Gráfico tensão x deformação PLA com tratamento p 20% espessura de camada de 0,4mm



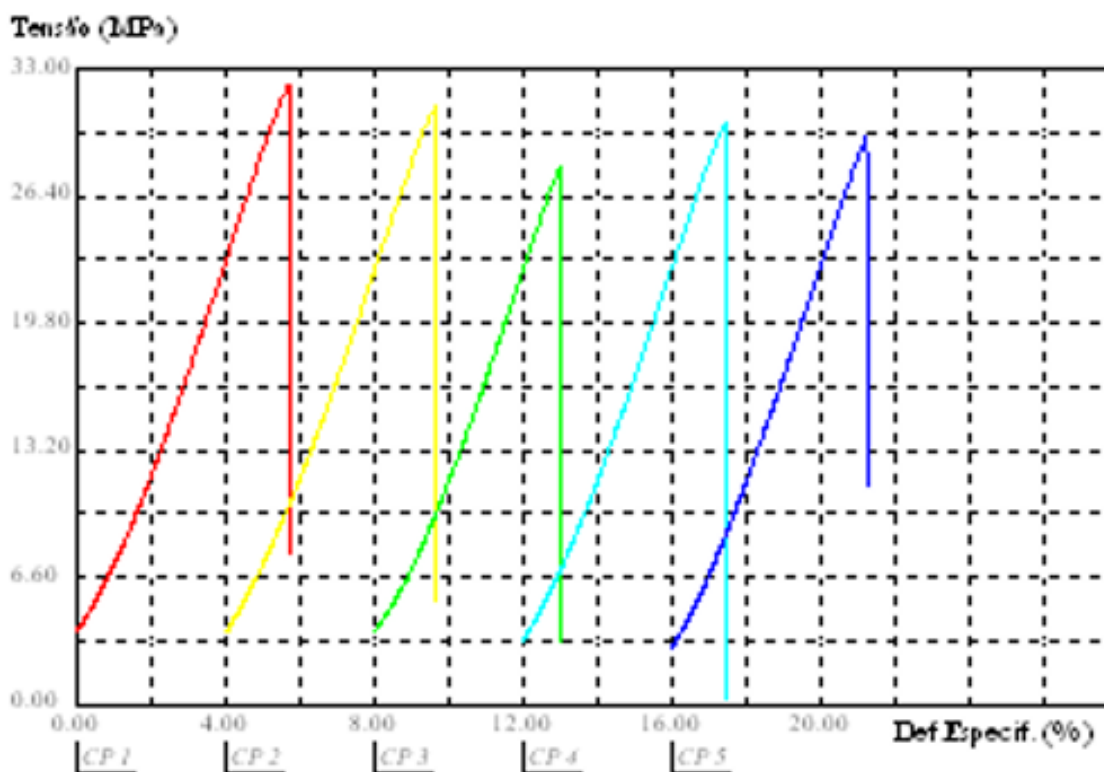
Fonte: Autor (2023)

Os filamentos são alterados pela incorporação de aditivos, visando a adaptação a diferentes condições, como o aumento da resistência mecânica e a elevação da

temperatura de trabalho. O tratamento do PLA com aditivos tem como objetivo aprimorar a resistência mecânica a impactos e permitir que suporte temperaturas de trabalho mais elevadas. Conforme demonstrado acima, observa-se um aumento na tensão e na deformação, resultando em uma máxima suportada de 22 MPa e uma deformação de 6,5% apresentando um desvio padrão de 1,632MPa.

O próximo ensaio de tração será realizado com um preenchimento de 100%, sendo comparado com um gráfico do mesmo artigo (Figura 101) e com um gráfico do material impresso na impressora construída, conforme ilustrado na Figura 104.

Figura 104 - Gráfico tensão x deformação PLA, p 100% espessura de camada 0,4mm



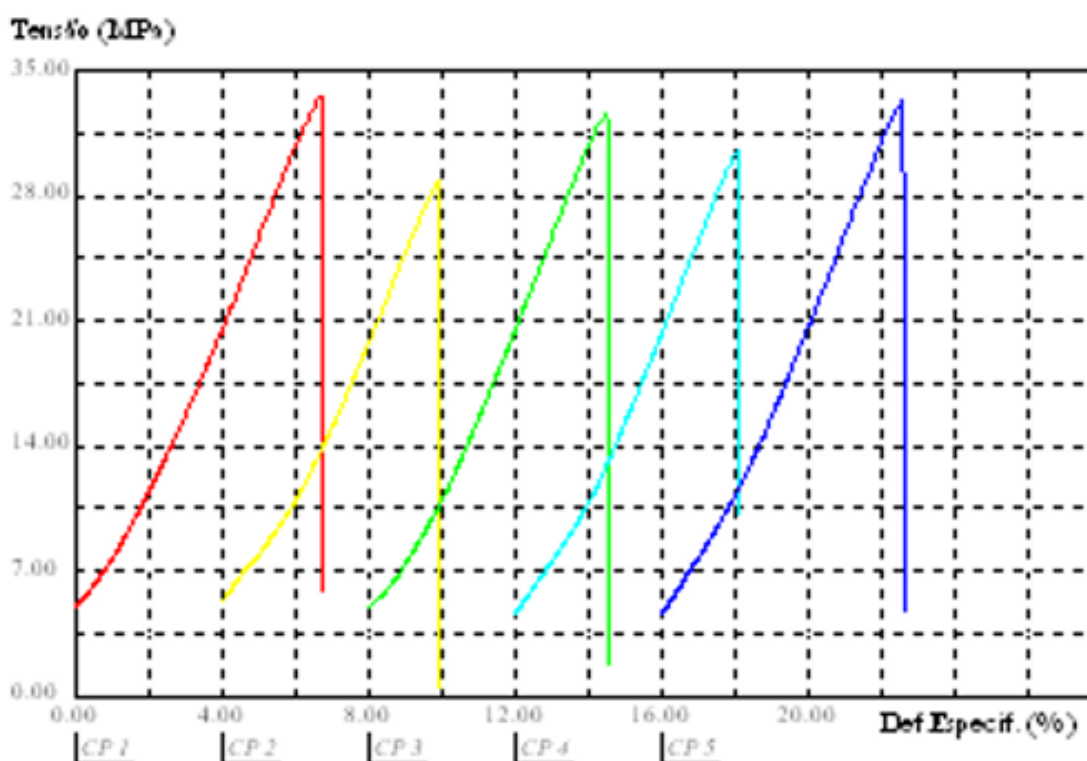
Fonte: Autor (2023)

É possível observar uma semelhança notável entre as curvas de tensão versus deformação das curvas em comparação as amostras A1 e A5, destacando ainda um aumento na resistência à tração do material decorrente do incremento na densidade

(preenchimento) do material. A tensão máxima suportada alcançou 32 MPa, resultando em uma deformação média de 6,0% e um desvio padrão de 1,629MPa.

Para fins de comparação, o mesmo ensaio foi conduzido, com a diferença de que, ao final, foi aplicado o tratamento térmico (recozimento), mantendo as características anteriores inalteradas, ou seja, a uma temperatura de 100°C por 30 minutos. Os resultados são apresentados na Figura 105.

Figura 105 - Gráfico tensão x deformação PLA com tratamento térmico p 100% espessura de camada de 0,4mm



Fonte: Autor (2023)

Conforme observado, houve um incremento tanto na tensão máxima suportada quanto na deformação, com uma máxima suportada de 34 Mpa, uma deformação de 6,5% e apresentando um desvio padrão de 2,046MPa.

Como nota adicional, é relevante destacar que os testes realizados desempenham um papel crucial na avaliação não apenas da estrutura e do comportamento do material PLA, mas também no impacto comparativo entre uma impressora previamente utilizada em outro projeto e a impressora construída neste trabalho, voltado para fins acadêmicos.

Vale ressaltar também que o preenchimento na produção do material também mostra grande influência na resistência do material, conforme mostrado acima.

5.2 Ensaio de dureza

As peças destinadas ao ensaio de dureza Shore D foram impressas conforme os parâmetros predefinidos, resultando na produção de 3 corpos de prova que foram posteriormente comparados com a literatura, mais especificamente, com uma dissertação.

Quanto às dimensões, as peças possuem formato de 40 x 40 mm e 6 mm de espessura, seguindo uma espessura de camada de 0,4 mm e um preenchimento de 20%.

Realizou-se um total de 5 medições em cada peça, sempre na superfície oposta àquela em contato com a plataforma da impressora. A carga aplicada foi de 5 kg. Os valores obtidos foram registrados nas extremidades (esquerda e direita) e no centro da amostra. A Figura 106 exibe os dados fornecidos pelo fabricante.

Figura 106 - Propriedades dos materiais PLA, ABS e PETG

Propriedades Materia Prima 3D LAB - Grãos			
Propriedades	PLA	ABS Premium	PETG
Densidade	1,24 [g/cm ³]	1,04 [g/cm ³]	1,27 [g/cm ³]
Temp. Fusão	185 [°C]	220 [°C]	240 [°C]
Tg	60 [°C]	100 [°C]	85 [°C]
Tensão de Escoamento	66 [Mpa]	38 [Mpa]	51 [Mpa]
Resistencia a Flexão	130 [Mpa]	66 [Mpa]	72 [Mpa]
Modulo de Elasticidade	4350 [Mpa]	2200 [Mpa]	2120[Mpa]
Resultado ensaio de Tração segundo a Norma ASTM D 638 - Corpo de provas Impresso			
Propriedades	PLA	ABS Premium	PETG
Tensão de Escoamento	24,8 [Mpa]	14,7[Mpa]	18,6 [Mpa]
Modulo de Elasticidade	1896,0 [Mpa]	1335,9 [Mpa]	1067,9 [Mpa]
Tensão de Ruptura	46 [Mpa]	29 [Mpa]	32,6 [Mpa]
Alongamento	3,69 [%]	7,08 [%]	7,74 [%]
Resultado ensaio de Dureza segundo a Norma ASTM D 2240 - Corpo de provas Impresso			
Dureza Shore D	85 [Shore D]	74 [Shore D]	75 [Shore D]
Resultado ensaio HTD segundo a Norma ISO 75 - Corpo de provas Impresso			
Temperatura HDT	55,11°C	86,13°C	67,3 °C

Fonte: 3D lab (2021), adaptado pelo Autor.

Os resultados encontrados no ensaio de dureza estão descritos no quadro 2.

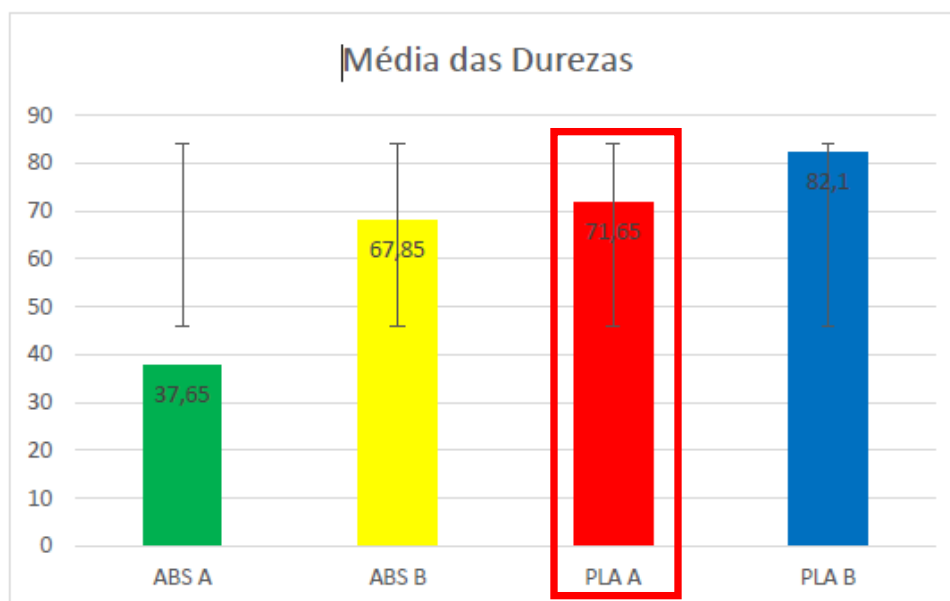
Quadro 2 - Medidas de dureza Shore D

PLA			
	CP 1	CP2	CP3
Medição 1	70,7	70,7	70,7
Medição 2	70,4	70,4	70,7
Medição 3	70,6	70,6	70,8
Medição 4	70,6	70,6	70,5
Medição 5	70,6	70,7	70,6
Média	70,58	70,6	70,66
Desvio Padrão	0,110	0,122	0,114

Fonte: Autor (2023)

Na Figura 107, segue o resultado do experimento realizado com PLA em uma literatura (Silva, 2022). A impressora utilizada foi a Impressora MakerBot Replicator 2X.

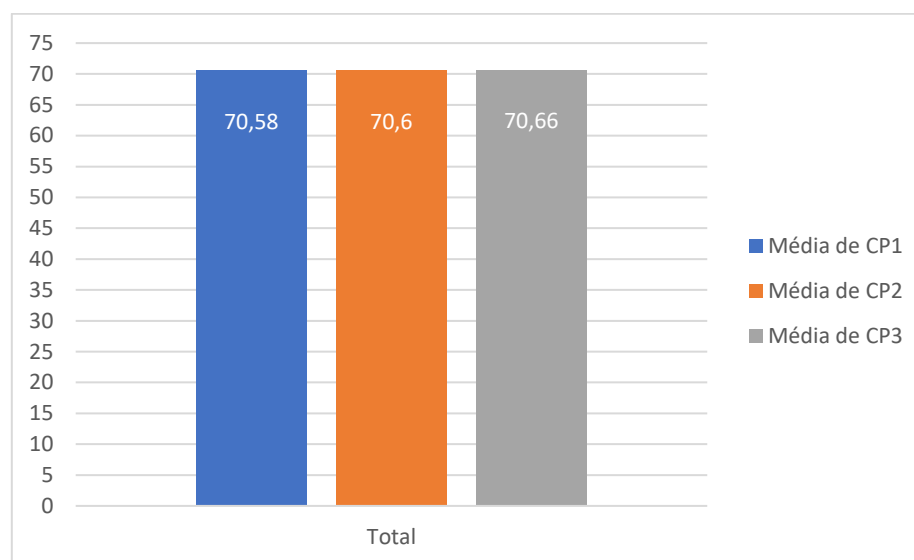
Figura 107 - Media das durezas PLA, ABS e PETG Shore D



Fonte: Silva (2022), adaptado pelo autor.

Em comparação segue na Figura 108 o gráfico referente os dados de dureza do material impresso pela impressora 3D construída, considerando preenchimento de 20% sem tratamento térmico.

Figura 108 - Media dureza shore D



Fonte: Autor (2023)

Observa-se que os valores obtidos estão em proximidade com os valores impressos pela impressora, o que assegura que a impressora construída está alinhada com outras impressoras disponíveis no mercado.

6 CONCLUSÃO

Nos últimos anos, houve um aumento significativo no uso de impressoras 3D, com destaque para a técnica abordada neste trabalho, a FDM - Modelagem por Fusão e Deposição. O que inicialmente era exclusivamente destinado à prototipagem rápida evoluiu para a produção efetiva de componentes e peças funcionalmente relevantes. Nesse contexto, a análise dos parâmetros revela-se fundamental e essencial para compreender o desempenho de qualquer item impresso quando exposto a forças externas.

Em linhas gerais, fica evidente a importância da comparação com estudos teóricos, desde a montagem até a conclusão com a realização de ensaios nos materiais

produzidos pela impressora, tanto para obter a aprovação quanto para a certificação do equipamento. Os resultados apresentados, tanto na montagem da estrutura e calibração, quanto nos ensaios finais dos materiais, indicaram que a impressora construída manualmente para fins acadêmicos obteve resultados satisfatórios em comparação com as impressoras industriais e que ter o conhecimento das partes estruturais como partes mecânicas e partes eletrônicas é crucial para o desenvolvimento e aplicações para a manufatura aditiva pelo processo FDM, conforme foi mostrado nesse trabalho.

No que diz respeito aos resultados finais dos ensaios nos corpos de prova, a impressora apresentou desempenho positivo. As tensões máximas para os corpos de prova com 20% de preenchimento, tanto com recozimento quanto sem, e para os com 100% de preenchimento, também com e sem recozimento, foram registradas em 19MPa, 20MPa, 30MPa e 32MPa, apresentando um desvio padrão para os corpos de prova com 20% de preenchimento de 1,105MPa e 1,632MPa e para os corpos de prova com preenchimento de 100% de 1,629MPa e 2,049MPa respectivamente. É notável que esses valores se aproximam das práticas de mercado, mantendo as características de impressão inalteradas e comprovam também que o preenchimento também influencia na resistência do material. Quanto aos testes de dureza, a média dos valores para os corpos com 20% de preenchimento foi de 70,6 Shore D e desvio padrão médio de 0,115 Shore D, demonstrando proximidade com as práticas adotadas no mercado.

Analisando a literatura existente sobre PLA, propriedades mecânicas e parâmetros de impressão, destaca-se a lacuna em pesquisas que abordem mais de um material e mais de um parâmetro variável. Com base nessas análises, fica claro que é viável construir uma impressora 3D de baixo custo para aplicação no meio acadêmico, visando a análise do conceito de montagem e o estudo da caracterização de materiais. Além disso, esse conceito pode ser aplicado no meio industrial. Observa-se também que foi possível aplicar integrar a impressora ao IOT e que é possível integrar qualquer impressora, desde as mais antigas até as mais avançadas, no contexto da Indústria 4.0 e do conceito IoT, graças aos microcontroladores disponíveis no mercado, conforme evidenciado no estudo. Essa aplicação pode ir além, incluindo o controle e monitoramento remoto do equipamento, facilitando o processo de impressão de materiais.

7 PROPOSTA PARA TRABALHOS FUTUROS

Como perspectivas para trabalhos futuros, surgem diversas opções à medida que o tema de estudo é aprofundado:

1. Avaliar a influência de outros parâmetros de impressão e de material;
2. Avaliar o comportamento do PLA impresso a 60°C e 80°C após o tratamento térmico;
3. Explorar os mesmos parâmetros investigados neste estudo, variando os ranges para cada um deles;
4. Analisar o desempenho de diferentes materiais e parâmetros de impressão em condições de desgaste cíclico;
5. Investigar os materiais com o auxílio do MEV (Microscópio Eletrônico de Varredura).

REFERÊNCIAS

AGOSTINI, L.; FILIPPINI, R. **Organizational and managerial challenges in the path toward Industry 4.0**. *European Journal of Innovation Management*, 2019.

AHMED, A. A., IBRAHIM, E. M., & FAHMY, M. A. (2018). ***Design and Implementation of a Closed-Loop Control System for a 3D Printer Stepper Motor***. *International Journal of Electrical, Computer, and Systems Engineering*, 12(3), 243-250.

AHMED, M., ZAFAR, S., & IQBAL, A. (2019). ***A Review on Stepper Motor: Advantages, Types, and Applications***. *International Journal of Automation and Smart Technology*, 9(1), 1-12.

ALCAIDE, E.; WILTGEN, F. **Estudo das tecnologias em prototipagem rápida: Passado, presente e futuro**. *Revista Ciências Exatas*, v. 24, n. 2, p. 12–20, 2018.

Arduino. (s.d.). **Arduino Mega**. Retrieved from <https://www.arduino.cc/en/Main/ArduinoBoardMega>

ATTARAN, M. (2017). **The rise of 3-D printing: The advantages of additive manufacturing over traditional manufacturing**. *Business Horizons*, 60(5), 677-688.

AYALA, J. L. (2006). ***The 8051 Microcontroller: Architecture, Programming and Applications***. Cengage Learning.

AZEVEDO, F. M. d. et al. **Estudo e projeto de melhoria em máquina de impressão 3D**. Tese (Doutorado) — Universidade de São Paulo, 2013.

BADOTTI, A. V. B. et al. **Avaliação do processo de metalização superficial aplicado a peças obtidas por estereolitografia**. Florianópolis, SC, 2003

BÁRTOLO, P., ALVES, N., & MENDES, A. (2019). ***Optimization of the Heating Process of a 3D Printer's Heated Bed***. *Procedia Manufacturing*, 41, 541-548.

BOURELL, D. L., KRUTH, J. P., LEU, M. C., LEVY, G. N., & ROSEN, D. W. (2017). ***Roadmap for additive manufacturing: Identifying the future of freeform processing***. *Journal of Materials Processing Technology*, 249, 743-755.

BROWN, Z. (2017). ***STEPPER MOTOR DRIVES AND THEIR INDUSTRIAL APPLICATIONS***. INDUSTRIAL ELECTRONICS SOCIETY, IECON 2017 - 43rd Annual Conference of the IEEE (pp. 2352-2356).

CALLISTER JR., W. D. (2006). **Ciência e Engenharia de Materiais: Uma Introdução**. John Wiley & Sons.

CARNEIRO, L.; et al. Influência da temperatura da mesa de impressão 3d na dureza e dimensões geométricas de amostras em abs. In: XXVI Congresso Nacional de Estudantes de Engenharia Mecânica, 2019

CHEN, W., XU, J., WU, J., & ZHU, Q. (2020). **Research on Key Technologies of Intelligent 3D Printing based on Stepper Motor**. Journal of Manufacturing Systems, 58, 472-481.

COWIE, J. M. G., & ARRIGHI, V. (2008). **Polymers: Chemistry and Physics of Modern Materials**. CRC Press.

CRUZ, P., CARRETERO, J., & ÚBEDA, R. (2019). **3D printing as a key technology for the open-source materials platform**. *Polymers*, 11(11), 1829.

DA COSTA, C. Indústria 4.0: o futuro da indústria nacional. **POSGERE-Pós-Graduação em Revista/IFSP-Campus São Paulo**, v. 1, n. 4, p. 5-14, 2017.

DEJAN. **Stepper Motors and Arduino - The Ultimate Guide**. Disponível em: <<https://howtomechatronics.com/tutorials/arduino/stepper-motors-and-arduino-the-ultimate-guide/>>.

DUET3D. (s.d.). *Duet 2 Wifi*. Retrieved from <https://www.duet3d.com/Duet2Wifi>

ebook/guia-maker-da-impressao-3d-tecnologiafff.adoc at master · Patola/ebook. Disponível em: <<https://github.com/Patola/ebook/blob/master/guia-maker-da-impressao-3d-tecnologiafff.adoc>>. Acesso em: 15 out. 2023.

FARAHANI, R. D., DUBÉ, M., & THERRIAULT, D. (2016). **Three-dimensional printing of multifunctional nanocomposites: manufacturing techniques and applications**. *ACS Applied Materials & Interfaces*, 8(6), 3334-3349.

FREITAS, A. A. **A internet das coisas e seus efeitos na indústria 4.0**. Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Computação) – Universidade Federal Fluminense, Rio de Janeiro, 2017.

GAO, W., ZHANG, Y., RAMANUJAN, D., RAMANI, K., CHEN, Y., WILLIAMS, C. B., ... & MANI, M. (2015). **The status, challenges, and future of additive manufacturing in engineering**. *Computer-Aided Design*, 69, 65-89.

GAO, X., MA, P., & FANG, Z. (2019). **A New Reactive Flame Retardant for Epoxy Resin Based on Phosphorus-Containing Polybenzoxazine**. *Journal of Applied Polymer Science*, 136(21), 47559.

GIBSON, I., ROSEN, D. W., & STUCKER, B. (2014). **Additive Manufacturing Technologies: 3D Printing, Rapid Prototyping, and Direct Digital Manufacturing**. Springer.

HUANG, Y., ZHANG, L., & ZHANG, Y. (2014). **Fused deposition modeling-based 3D printing technologies: materials, research trends, applications and challenges.** *Assembly Automation*, 34(4), 310-321.

HUBA, M.; KOZÁK, Š. From. E-learning. to Industry. 4.0. **In: Emerging eLearning Technologies and Applications. (ICETA), 2016 International Conference on.** IEEE. p. 103-108, 2016.

HUSSEIN, A., ANWER, N., & EL-SAYED, A. (2018). **Modeling and Analysis of the Fused Filament Fabrication Process.** *Procedia Manufacturing*, 21, 394-404.

IMPRESORAS3D.COM. **Guia de extrusoras para impressora 3D.** Disponível em: <<https://www.impresoras3d.com/pt/la-guia-definitiva-extrusores-para-impresoras-3d/>>. Acesso em: 15 out. 2023.

JESUS, Marco Sousa. Desenvolvimento de um compósito polímero-metal à base de resina epóxi para aplicações em moldes rápidos. Dissertação (Mestrado em Engenharia de Materiais). Florianópolis. 2005. 102p. Universidade Federal de Santa Catarina.

KAGERMANN, H., LUKAS, W., & WAHLSTER, W. (EDS.). (2013). **Industrie 4.0: Mit dem Internet der Dinge auf dem Weg zur 4. industriellen Revolution.** VDI.

KALSOOM, U.; NESTERENKO, P. N.; PAULL, B. **Current and future impact of 3D printing on the separation sciences.** *Trac Trends in Analytical Chemistry*, 2018.

KARGER-KOCSIS, J., & MAHMOODI, P. (2008). **Toughening of Polymers with Micro- and Nano-sized Particles.** *Progress in Polymer Science*, 33(8), 874-901.

KHORASANI, A. M., GIBSON, I., & GOLDBERG, M. (2017). **A review of powder additive manufacturing processes for metallic biomaterials.** *Biomaterials*, 135, 28-40.

KIM, Y., PARK, J., & LEE, S. (2019). **Development of a High-Temperature Thermistor for 3D Printing Applications.** *Sensors*, 19(24), 5468.

KLIPPER. (S.d.). *Klipper*. Retrieved from <https://www.klipper3d.org/>. Acesso em 22 de jun. 2023.

KREIGER, M., MULVANEY, J., & PEARCE, J. M. (2017). **Tensile Strength of Commercial Polymer Materials for Fused Filament Fabrication 3D Printing.** *Additive Manufacturing*, 15, 40-47.

KRISHNA, S. B., KUMAR, V., & NARAYANA, K. L. (2020). **A Review on 3D Printing Technology Using Arduino Microcontroller.** *IOP Conference Series: Materials Science and Engineering*, 827, 012036.

LAKE, G. J., & THOMAS, A. G. (1967). **The Strength of Highly Elastic Materials.** *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 300(1461), 108-119.

LE, Q. V., TRAN, T. K., & PHAM, D. T. (2018). **A Review on Heated Bed Technologies for Fused Deposition Modelling (FDM) 3D Printing Machines**. *International Journal of Advanced Manufacturing Technology*, 94(9-12), 3239-3256.

LEE, J., LAPIRA, E., BAGHERI, B., & KAO, H. A. (2013). **Industry 4.0: A vision for sustainable growth**. *Procedia CIRP*, 16, 12-17.

LEMKE, R.; SIPLE, I. Z.; FIGUEIREDO, E. B. **OAs para o ensino de cálculo: potencialidades de tecnologias 3D**. In: *Revista Novas Tecnologias na Educação (RENOTE)*. CINTED-UFRGS. Porto Alegre, v.14, n.1, julho, 2016.

LI, C., WANG, L., & ZHANG, Y. (2020). **Intelligent Temperature Control for 3D Printing Based on Real-time Thermal Imaging**. *Robotics and Computer-Integrated Manufacturing*, 61, 101837.

MANYIKA, J., CHUI, M., BISSON, P., WOETZEL, J., DOBBS, R., BUGHIN, J., & AHARON, D. (2015). **Unlocking the Potential of the Internet of Things**. McKinsey & Company.

MARK, J. E. (2007). **Physical Properties of Polymers Handbook**. Springer Science & Business Media.

Marlin 3D Printer Firmware, 2023. Disponível em: <http://marlinfw.org/docs/basics/introduction.html#how-marlin-works>. Acesso em: 22 jun. 2023.

MORAES, E. C.; BONATELLI, M. L. **Manufatura aditiva: primeiras impressões 3D e o futuro da produção camada por camada**, 2018. Disponível em: <<https://bityli.com/wj9WZ>>. Acesso em: 28 jun. 2023

OctoPrint. (s.d.). *OctoPrint*. Retrieved from <https://octoprint.org/>

ODIAN, G. (2004). **Principles of Polymerization**. John Wiley & Sons.

PAUL, D. R. (2014). **Introduction to Polymer Blends**. In *Polymer Blends* (pp. 1-34). John Wiley & Sons.

PIPES, A. **Desenho para designers**. São Paulo: Blucher, 2010.

PORTELA, S. **Impressora 3D de metal: conheça a máquina que transformará o mercado**. 3DLAB. Disponível em: < <https://bityli.com/8zWSo>>. Acesso em: 22 jun. 2023.

RAZGRIZ, G. **Conjunto de extrusor para impressão 3D: te contamos tudo sobre ele!** Disponível em: <<https://www.makehero.com/blog/conjunto-de-extrusor-para-impressao-3d/>>. Acesso em: 15 out. 2023.

REPRAP.ORG. (s.d.). *RAMPS 1.4*. Retrieved from https://reprap.org/wiki/RAMPS_1.4

RODRÍGUEZ, J. C., CABANES, I., & DÍAZ, J. (2018). ***Towards More Precise and Faster 3D Printers: A Delta Printer Case.*** Robotics and Computer-Integrated Manufacturing, 49, 292-300.

Sabik, A.; Rucka, M.; Andrzejewska, A.; Wojtczak, E. **Tensile Failure Study of 3D Printed PLA Using DIC Technique and FEM Analysis.** Mech. Mater. **2022**, 175, 104506.

SAMPAIO, C. **Guia Maker da Impressão 3D - Teoria e Prática Consolidadas.** [s.n.], 2017. Disponível em: <<http://www.makerlinux.com.br/drupal/content/ebook>>.

SANTOS, B. P. *et al.* **Internet das coisas: da teoria à prática.** Minicursos SBRC-Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, 2016.

SANTOS, B. P. *et al.* **Indústria 4.0: Desafios e Oportunidades.** Revista Produção e Desenvolvimento, v. 4, n. 1, p. 111–124, 2018. Acesso em: 28 jun. 2023.

SCHWAB, K. (2017). ***The Fourth Industrial Revolution.*** Currency.

SILVA, Milena Fontoura, **Relação dos parâmetros de impressão com propriedades mecânicas de peças poliméricas obtidas por impressão 3D pela técnica FDM.** Dissertação (Mestrado em Engenharia de Materiais). Rio Grande do Sul (2022). Pontifícia Universidade Católica do Rio Grande do Sul.

SMITH, A., BROWN, Z., & JOHNSON, R. (2018). ***Temperature Sensing and Control in 3D Printing Using Infrared Thermography.*** Journal of Manufacturing Science and Engineering, 140(10), 101005.

SMITH, P., JOHNSON, R., & ANDERSON, K. (2019). ***Advanced Control Techniques for Stepper Motors in 3D Printing Applications.*** IEEE Transactions on Automation Science and Engineering, 16(3), 1454-1461.

SMOOTHIEWARE. (s.d.). ***Smoothieboard.*** Retrieved from <http://smoothieware.org/smoothieboard>

THINGIVERSE. **Thingiverse Featured. 2019.** Disponível em: <https://www.thingiverse.com/search?q=+hook&dwh=835c81a2e7ad72f>. Acesso em: 28 de ago. 2023.

THOMASON, J. L. (2002). ***The Influence of Fibre Length and Concentration on the Properties of Glass Fiber Reinforced Polypropylene: 5. Injection Moulded Long and Short Fiber PP.*** Composites Part A: Applied Science and Manufacturing, 33(12), 1641-1652.

THRE3D - ***3D Printing, Simplified., 2014.*** Disponível em:<<https://thre3d.com>>. Acesso em: 22 jun. 2023.

TRAPERO, D. **Guia para escolher o extrusor da minha impressora 3D**. Disponível em: <<https://bitfab.io/pt-pt/blog/extrusor-impressora-3d>>. Acesso em: 23 jun. 2023.

Tudo sobre o driver DRV8825 para motor de passo. Disponível em: <<https://www.crescerengenharia.com/post/tudo-sobre-drv8825>>.

ULLAH, Z., NAEEM, M., ANWAR, S. M., & LEE, S. (2019). **Internet of Things (IoT) in 5G wireless communications**. In *Internet of Things (IoT) in 5G Mobile Technologies* (pp. 45-67). Springer.

VAEZI, M., & SEITZ, H. (2016). **Highly filled biodegradable PLA/wood-derived char bio-nanocomposites with improved flame retardancy, dimensional stability, and strength**. *Journal of Applied Polymer Science*, 133(12), 43274.

VALVANO, J. (2012). **Embedded Systems: Introduction to Arm® Cortex™-M Microcontrollers**. Cengage Learning.

VIG, A., VASHISHTHA, S., & MAHESHWARI, S. (2019). **3D PRINTING WITH ARDUINO**. *PROCEDIA Computer Science*, 152, 90-98.

VOLPATO, N. (2007). **Prototipagem rápida: Tecnologias e aplicações**. São Paulo: Edgard Blucher.

VOLPATO, N. et al. **Prototipagem rápida: tecnologias e aplicações**. – São Paulo: Blücher, 2006.

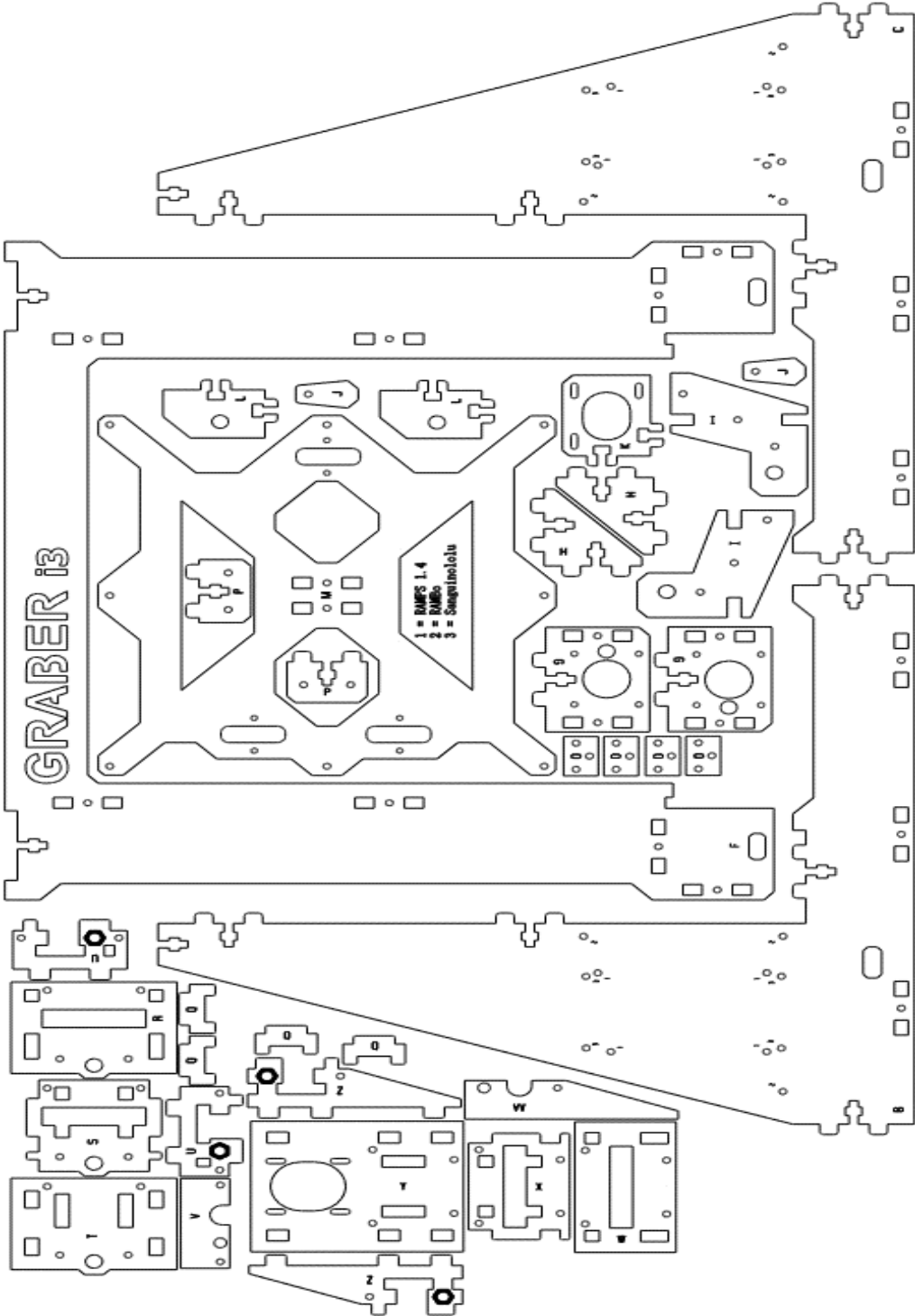
WANG, H., ZHANG, Y., & LI, J. (2020). **Heat Distribution Control of Heated Bed in 3D Printing Based on Multi-Objective Optimization**. *International Journal of Advanced Manufacturing Technology*, 107, 3063-3077.

WITTBRODT, B. T., & PEARCE, J. M. (2015). **The effects of PLA color on material properties of 3-D printed components**. *Additive Manufacturing*, 8, 110-116.

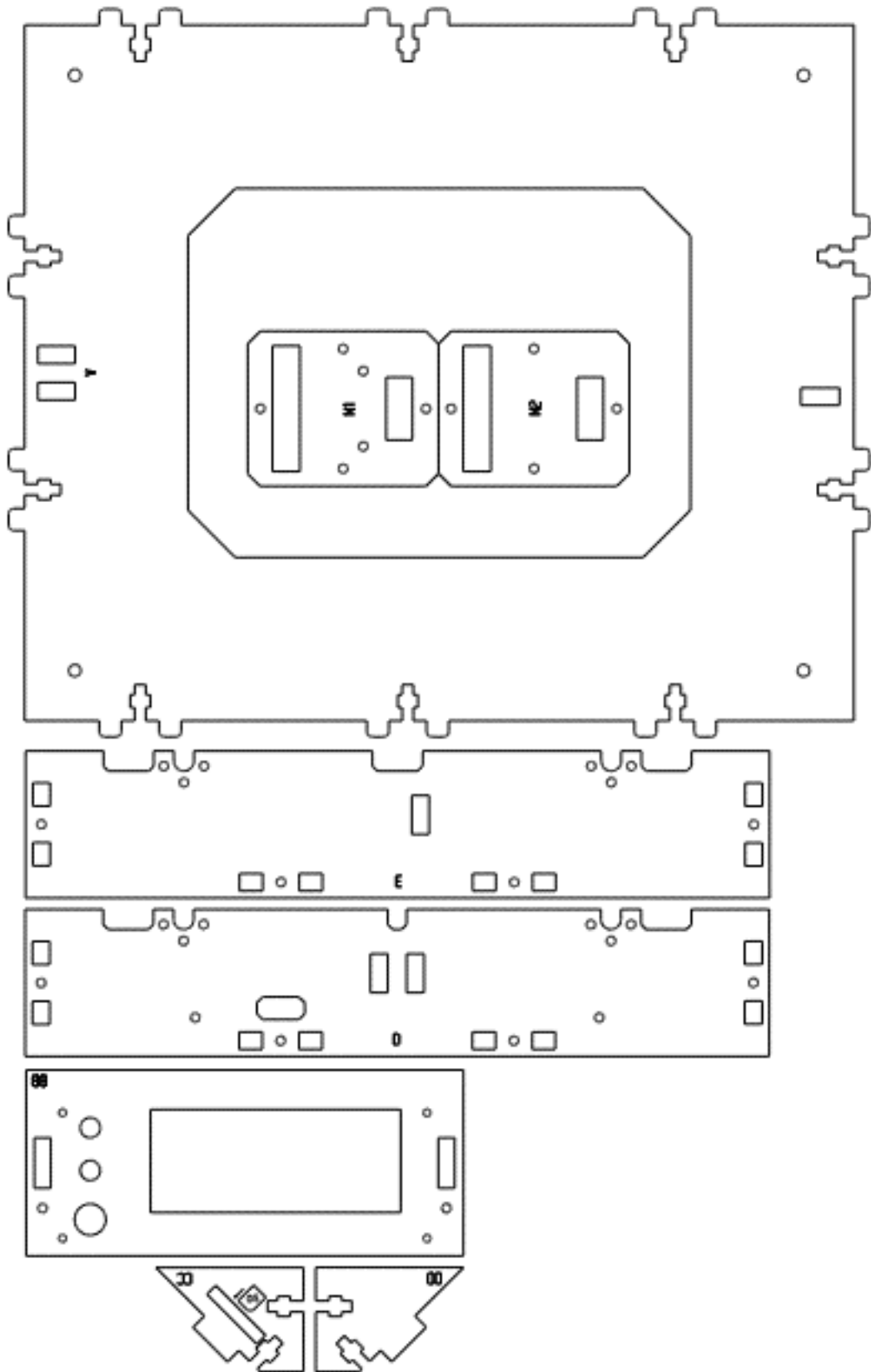
ZHANG, Y., ZHAO, Z., LIU, H., & WANG, W. (2018). **Stepper Motor Control System in 3D Printing Based on ARM**. *Journal of Physics: Conference Series*, 1065, 052015.

ANEXOS

Anexo A – Estrutura (Frame) impressora 3D MDF



Fonte: <https://github.com/sgraber/Graber>



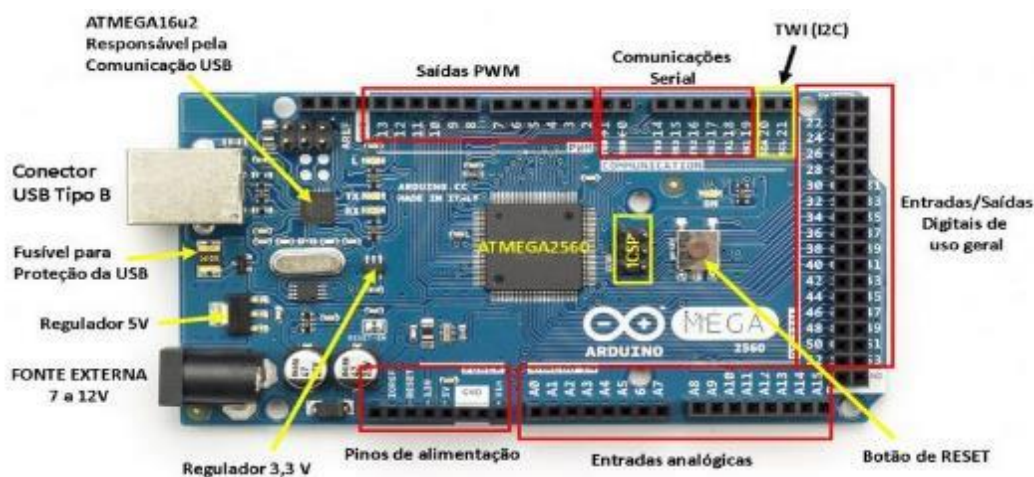
Anexo B – Lista de peças

Lista de materiais impressora 3d				
Item	Peças	Qntd	Link	Preço (\$)
1	Estrutura MDF	1 m ²	https://produto.mercadolivre.com.br/MLB-1087008605-estrutura-impressora-3d-graber-i3-mdf-promoco-relampago-JM#position=53&search_layout=stack&type=item&tracking_id=8f3cb73e-1c5b-4763-955d-4aa5626e4d79	\$ 16,06
2	kit 5 Motores Nema 17	1	https://br.banggood.com/BR-DirectTWO TREES-5pcs-17HS4401S-V5-V6-4-lead-Nema17-Stepper-Motor-42-motor-Nema-17-42BYGH-1_5A-for-CNC-XYZ-For-3D-Printer-Parts-p-1996119.html?gmcCountry=BR&currency=BRL&cur_warehouse=BR&createTmp=1&ID=564750	\$ 60,04
3	Kit Arduino/Eletrônica com ramps e display e mesa MK2	1	https://pt.aliexpress.com/item/32958935172.html?spm=a2g0o.detail.1000023.6.16534BOL4BOLQK&gatewayAdapt=glo2bra	\$ 31,12
4	Barra Roscada M5	2	http://www.itcterm.com.br/produto/conjunto-barra-roscada-m5-inox-prusa-i3/	\$ 4,02
5	Hotend Direct drive 0,4/1,75mm com termistor	1	https://www.robobuilders.com.br/produto/hotend-e3d-v6-direct-allmetal-175mm-c-ventilador.html?utm_source=Site&utm_medium=GoogleMerchant&utm_campaign=GoogleMerchant	\$ 11,24
6	Extrusor direct drive completa	1	https://www.gtmax3d.com.br/componentes/extrusora-direct-gtmax3d	\$ 30,12
7	Barras Lisas	6	https://beecost.com.br/kit-eixos-lineares-barras-lisas-graber-i3-impressora-3d-diy-p.1__11906267381__413680054	\$ 22,09
8	Termistor 100k mesa	1	https://www.baudaeletronica.com.br/produto/termistor-ntc-100k-com-cabo-1m-e-conector.html?utm_source=Site&utm_medium=GoogleMerchant&utm_campaign=GoogleMerchant&qad_source=1&qclid=EAlaQobChMIrfGri-rphAMVSFIAB1fpQXuEAQYByABEgLD1_D_BwE	\$ 1,79
9	Servo 9g, Chave fim de curso e fios	1	https://www.a2robotics.com.br/micro-servo-9g-sg90-towerpro?utm_source=Site&utm_medium=GoogleShopping&utm_campaign=GooglePMax&qad_source=4&qclid=EAlaQobChMItu2gtOjphAMVb2FIAB1M4QIU EAQYASABEglnwfd_BwE	\$ 2,01
10	Fonte 12V 30A	1	https://produto.mercadolivre.com.br/MLB-3584922527-fonte-chaveada-bivolt-30a-12v-360w-versatil-facil-instalaco-JM?matt_tool=40343894&matt_word=&matt_source=google&matt_campaign_id=14303413655&matt_ad_group_id=133855953276&matt_match_type=&matt_network=g&matt_device=c&matt_creative=584156655519&matt_keyword=&matt_ad_position=&matt_ad_type=pla&matt_merchant_id=5332409896&matt_product_id=MLB3584922527&matt_product_partition_id=2268053647590&matt_target_id=aud-1966503641507:pla-2268053647590&cq_src=google_ads&cq_cmp=14303413655&cq_net=g&cq_plt=gp&cq_med=pla&qad_source=4&qclid=EAlaQobChMIINbLlOjphAMV_19IAB31eQZAE AQYBiABEgKyLvD_BwE	\$ 8,03

11	Parafusos e porcas		https://pt.aliexpress.com/item/1005004605678220.html?src=google&src=google&albch=shopping&acnt=768-202-3196&slnk=&plac=&mtctp=&albbt=Google_7_shopping&isSmbAutoCall=false&needSmbHouyi=false&albcpr=17364769157&albag=&trgt=&crea=pt1005004605678220&netw=x&device=c&albpq=&albpd=pt1005004605678220&gad_source=1&gclid=EAlaQobChMIr-SSzejphAMVDEVIAB26bgeJEAQYASABEgLSfD_BwE&gclidsrc=aw.ds&aff_fcid=45daa262389d4e2ea361d7345a061e58-1710077906817-00141-UnaMJZVf&aff_fsk=UnaMJZVf&aff_platform=aaf&sk=UnaMJZVf&aff_trace_key=45daa262389d4e2ea361d7345a061e58-1710077906817-00141-UnaMJZVf&terminal_id=fceec372564549c1a0c111baf553e203&afSmartRedirect=y	\$	17,67
12	Cooler radial extra	1	https://www.eletrogate.com/cooler-ventoinha-dc-12v-?utm_source=Site&utm_medium=GoogleMerchant&utm_campaign=GoogleMerchant&utm_source=google&utm_medium=cpc&utm_campaign=[MC4]_[G]_[PMax]_Categorias&utm_content=&utm_term=&gad_source=4&gclid=EAlaQobChMI4OqCj-nphAMV0I9IAB3hdgBdEAQYASABEgLDpPD_BwE	\$	2,01
13	Kit Movimento Impressora Reprap Prusa Graber I3 3d	1	https://www.mssantos.com.br/MLB-1214058142-kit-movimento-impressora-reprap-prusa-graber-i3-3d- JM	\$	21,89
				Total	\$ 228,09

Fonte: Autor, (2023)

Anexo C – Dados do Arduino Mega

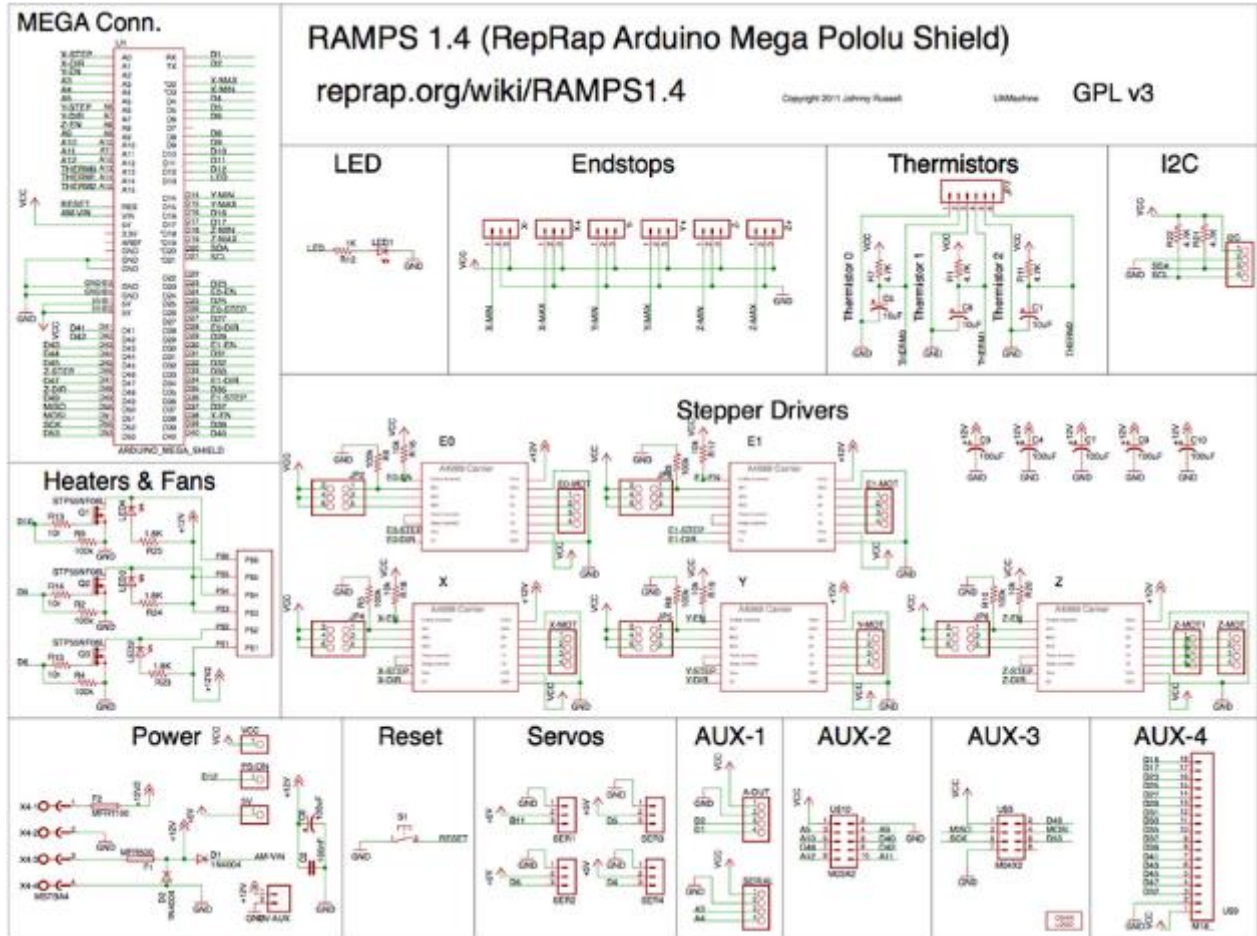


Fonte: <https://embarcados.com.br/arduino-mega-2560/>

Board	Name	Arduino® Mega 2560 Rev3
	SKU	A000067
Microcontroller	ATmega2560	
USB connector	USB-B	
Pins	Built-in LED Pin	13
	Digital I/O Pins	54
	Analog input pins	16
	PWM pins	15
Communication	UART	Yes, 4
	I2C	Yes
	SPI	Yes
Power	I/O Voltage	5V
	Input voltage (nominal)	7-12V
	DC Current per I/O Pin	20 mA
	Supported battery	9V battery
	Power Supply Connector	Barrel Plug
Clock speed	Main Processor	ATmega2560 16 MHz
	USB-Serial Processor	ATmega16U2 16 MHz
Memory	ATmega2560	8KB SRAM, 256KB FLASH, 4KB EEPROM
Dimensions	Weight	37 g
	Width	53.3 mm
	Length	101.5 mm

Fonte: <https://docs.arduino.cc/hardware/mega-2560>

Anexo D – Especificações técnicas Ramps



Fonte: https://reprap.org/wiki/RAMPS_1.4

Anexo E – G-codes e M-codes

11 G-commands
 11.1 G0 & G1: Move
 11.2 G2 & G3: Controlled Arc Move
 11.3 G4: Dwell
 11.4 G5: Direct Stopper Move
 11.5 G10: Set tool, Offset and/or workpiece coordinates and/or tool temperatures
 11.6 G11: Retract
 11.7 G11: Unretract
 11.8 G12: Clear Tool
 11.9 G17,18: Plane Selection (CNC specific)
 11.10 G20: Set Units to Inches
 11.11 G21: Set Units to Millimeters
 11.12 G22: Firmware Release
 11.13 G23: Firmware Restore
 11.14 G26: Mesh Validation Pattern
 11.15 G27: Park Toolhead
 11.16 G28: Move to Origin (Home)
 11.17 G29: Detailed Z-Probe
 11.17.1 G29 Auto Bed Leveling (Marlin - MK4duo)
 11.17.2 G29 Unified Bed Leveling (Marlin - MK4duo)
 11.17.3 G29 Manual Bed Leveling (Marlin - MK4duo)
 11.17.4 G29 Auto Bed Leveling (Repetier-Firmware)
 11.17.5 G29 Mesh Bed Compensation (Repetier-Firmware)
 11.18 G29.1: Set Z probe mesh offset
 11.19 G29.2: Set Z probe mesh offset calculated from touch-off position
 11.20 G30: Single Z-Probe
 11.21 G31: Set to Report Current Probe status
 11.22 G31: Dock Z Probe sled
 11.23 G32: Probe Z and calculate Z plane
 11.23.1 Probe and calculate in Repetier firmware
 11.23.2 Probe and calculate in Repetier firmware
 11.24 G33: Undock Z Probe sled
 11.25 G33: Firmware dependent
 11.25.1 G33: MeasureListAdjust: Distortion Matrix (Repetier - Redeem)
 11.25.2 G33: Delta Auto Calibration (Marlin 1.1.x - MK4duo)
 11.26 G34: Z Stopper Auto-Align
 11.27 G34: Calculate Delta Height from toolhead position (DELTA)
 11.28 G35.x Straight Probe (CNC specific)
 11.28.1 G35.1 probe toward workpiece, stop on contact, signal error if failure
 11.28.2 G35.2 probe toward workpiece, stop on contact
 11.28.3 G35.4 probe away from workpiece, stop on loss of contact, signal error if failure
 11.28.4 G35.5 probe away from workpiece, stop on loss of contact

- 11.29 G40: Compensation Off (CNC specific)
- 11.30 G42: Move to Grid Point
- 11.31 G53..59: Coordinate System Select (CNC specific)
- 11.32 G60: Save current position to slot, Restore and Delete
- 11.33 G68: Coordinate rotation
- 11.34 G69: Cancel coordinate rotation
- 11.35 G75: Print temperature interpolation
- 11.36 G76: PINDA probe temperature calibration
- 11.37 G80: Cancel Canned Cycle (CNC specific)
- 11.38 G80: Mesh-based Z probe
- 11.39 G81: Mesh bed leveling status
- 11.40 G82: Single Z probe at current location
- 11.41 G83: Babystep in Z and store to EEPROM
- 11.42 G84: UNDO Babystep Z (move Z axis back)
- 11.43 G85: Pick best babystep
- 11.44 G86: Disable babystep correction after home
- 11.45 G87: Enable babystep correction after home
- 11.46 G88: Reserved
- 11.47 G90: Set to Absolute Positioning
- 11.48 G91: Set to Relative Positioning
- 11.49 G92: Set Position
- 11.49.1 G92.x: Reset Coordinate System Offsets (CNC specific)
- 11.50 G93: Feed Rate Mode (Inverse Time Mode) (CNC specific)
- 11.51 G94: Feed Rate Mode (Units per Minute) (CNC specific)
- 11.52 G98: Activate farm mode
- 11.53 G99: Deactivate farm mode
- 11.54 G100: Calibrate floor or rod radius
- 11.55 G130: Set digital potentiometer value
- 11.56 G131: Remove offset
- 11.57 G132: Calibrate endstop offsets
- 11.58 G133: Measure steps to top
- 11.59 G161: Home axes to minimum
- 11.60 G162: Home axes to maximum
- 11.61 G425: Perform auto-calibration with calibration cube

12 M-commands

- 12.1 M0: Stop or Unconditional stop
- 12.2 M1: Sleep or Conditional stop
- 12.3 M2: Program End
- 12.4 M3: Spindle On, Clockwise (CNC specific)
- 12.5 M4: Spindle On, Counter-Clockwise (CNC specific)
- 12.6 M5: Spindle Off (CNC specific)
- 12.7 M6: Tool change
- 12.8 M7: Mist Coolant On (CNC specific)
- 12.9 M8: Flood Coolant On (CNC specific)
- 12.10 M9: Coolant Off (CNC specific)

12.11 M10: Vacuum On (CNC specific)
12.12 M11: Vacuum Off (CNC specific)
12.13 M13: Spindle on (clockwise rotation) and coolant on (flood)
12.14 M16: Expected Printer Check
12.15 M17: Enable/Power all stepper motors
12.16 M18: Disable all stepper motors
12.17 M20: List SD card
12.18 M21: Initialize SD card
12.19 M22: Release SD card
12.20 M23: Select SD file
12.21 M24: Start/resume SD print
12.22 M25: Pause SD print
12.23 M26: Set SD position
12.24 M27: Report SD print status
12.25 M28: Begin write to SD card
12.26 M29: Stop writing to SD card
12.27 M30: Delete a file on the SD card
12.27.1 M30: Program Stop
12.28 M31: Output time since last M109 or SD card start to serial
12.29 M32: Select file and start SD print
12.30 M33: Get the long name for an SD card file or folder
12.31 M33: Stop and Close File and save restart.gcode
12.32 M34: Set SD file sorting options
12.33 M35: Upload firmware NEXTION from SD
12.34 M36: Return file information
12.35 M36.1: Return embedded thumbnail data
12.36 M37: Simulation mode
12.37 M38 Compute SHA1 hash of target file
12.38 M39 Report SD card information
12.39 M40: Eject
12.40 M41: Loop
12.41 M42: Switch I/O pin
12.42 M43: Stand by on material exhausted
12.43 M43: Pin report and debug
12.44 M44: Codes debug - report codes available
12.45 M44: Reset the bed skew and offset calibration
12.46 M45: Bed skew and offset with manual Z up
12.47 M46: Show the assigned IP address
12.48 M47: Show end stops dialog on the display
12.49 M48: Measure Z-Probe repeatability
12.50 M49: Set G26 debug flag
12.51 M70: Display message
12.52 M72: Play a tone or song
12.53 M73: Set/Get build percentage
12.54 M74: Set weight on print bed
12.55 M75: Start the print job timer

12.56 M76: Pause the print job timer
12.57 M77: Stop the print job timer
12.58 M78: Show statistical information about the print jobs
12.59 M80: ATX Power On
12.60 M81: ATX Power Off
12.61 M82: Set extruder to absolute mode
12.62 M83: Set extruder to relative mode
12.63 M84: Stop idle hold
12.64 M85: Set Inactivity Shutdown Timer
12.65 M86: Set Safety Timeout
12.66 M87: Cancel Safety Timer
12.67 M92: Set axis_steps_per_unit
12.68 M93: Send axis_steps_per_unit
12.69 M98: Call Macro/Subprogram
12.70 M99: Return from Macro/Subprogram
12.71 M101: Turn extruder 1 on (Forward), Undo Retraction
12.72 M102: Turn extruder 1 on (Reverse)
12.73 M102: Configure Distance Sensor
12.74 M103: Turn all extruders off, Extruder Retraction
12.75 M104: Set Extruder Temperature
12.75.1 M104 in Marlin Firmware
12.75.2 M104 in Teacup Firmware
12.75.3 M104 in RepRapFirmware and Klipper
12.76 M105: Get Extruder Temperature
12.77 M106: Fan On
12.77.1 M106 in RepRapFirmware
12.77.2 M106 in Teacup Firmware
12.78 M107: Fan Off
12.79 M108: Cancel Heating
12.80 M108: Set Extruder Speed (BFB)
12.81 M109: Set Extruder Temperature and Wait
12.81.1 M109 in Teacup
12.81.2 M109 in Marlin, MK4duo, Sprinter (ATmega port), RepRapFirmware, Prusa
12.81.3 M109 in Sprinter (4pi port)
12.81.4 M109 in MakerBot
12.81.5 M109 in Klipper
12.82 M110: Set Current Line Number
12.83 M111: Set Debug Level
12.83.1 M111 in RepRapFirmware
12.83.2 M111 in Repetier
12.84 M112: Full (Emergency) Stop
12.85 M113: Set Extruder PWM
12.86 M113: Host Keepalive
12.87 M114: Get Current Position
12.88 M115: Get Firmware Version and Capabilities
12.89 M116: Wait

- 12.90 M117: Get Zero Position
- 12.91 M117: Display Message
- 12.92 M118: Echo message on host
- 12.93 M118: Negotiate Features
- 12.94 M119: Get Endstop Status
- 12.95 M120: Push
- 12.96 M121: Pop
- 12.97 M120: Enable endstop detection
- 12.98 M121: Disable endstop detection
- 12.99 M122: Firmware dependent
 - 12.99.1 M122: Diagnose (RepRapFirmware)
 - 12.99.2 M122: Set Software Endstop (MK4duo)
 - 12.99.3 M122: Debug Stepper drivers (Marlin)
- 12.100 M123: Firmware dependent
 - 12.100.1 M123: Tachometer value (RepRap, Prusa & Marlin)
 - 12.100.2 M123: Endstop Logic (MK4duo)
- 12.101 M124: Firmware dependent
 - 12.101.1 M124: Immediate motor stop
 - 12.101.2 M124: Set Endstop Pullup
- 12.102 M126: Open Valve
 - 12.102.1 M126 in MakerBot
- 12.103 M127: Close Valve
 - 12.103.1 M127 in MakerBot
- 12.104 M128: Extruder Pressure PWM
- 12.105 M129: Extruder pressure off
- 12.106 M130: Set PID P value
- 12.107 M131: Set PID I value
- 12.108 M132: Set PID D value
 - 12.108.1 M132 in MakerBot
- 12.109 M133: Set PID I limit value
 - 12.109.1 M133 in MakerBot
- 12.110 M134: Write PID values to EEPROM
 - 12.110.1 M134 in MakerBot
- 12.111 M135: Set PID sample interval
 - 12.111.1 M135 in MakerBot
- 12.112 M136: Print PID settings to host
- 12.113 M140: Set Bed Temperature (Fast)
- 12.114 M141: Set Chamber Temperature (Fast)
- 12.115 M142: Firmware dependent
 - 12.115.1 M142: Holding Pressure
 - 12.115.2 M142: Set Cooler Temperature (Fast)
- 12.116 M143: Maximum heater temperature
- 12.117 M144: Bed Standby
- 12.118 M146: Set Chamber Humidity
- 12.119 M149: Set temperature units
- 12.120 M150: Set LED color

- 12.121 M154: Auto Report Position
- 12.122 M155: Automatically send temperatures
- 12.123 M160: Number of mixed materials
- 12.124 M163: Set weight of mixed material
- 12.125 M164: Store weights
- 12.126 M165: Set multiple mix weights
- 12.127 M190: Wait for bed temperature to reach target temp
- 12.128 M191: Wait for chamber temperature to reach target temp
- 12.129 M192: Wait for Probe Temperature
- 12.130 M200: Set filament diameter
- 12.131 M201: Set max acceleration
- 12.132 M201.1: Set reduced acceleration for special move types
- 12.133 M202: Set max travel acceleration
- 12.134 M203: Firmware dependent
- 12.134.1 M203: Set maximum feedrate
- 12.134.2 M203 (Repetier): Set temperature monitor
- 12.135 M204: Firmware dependent
- 12.135.1 M204: Set default acceleration
- 12.135.2 M204 (Repetier): Set PID values
- 12.136 M205: Firmware dependent
- 12.136.1 M205: Advanced settings
- 12.136.2 M205 (Repetier): EEPROM Report
- 12.137 M206: Firmware dependent
- 12.137.1 M206: Offset axes
- 12.137.2 M206 (Repetier): Set EEPROM value
- 12.138 M207: Firmware dependent
- 12.138.1 M207: Set retract length
- 12.138.2 M207 Calibrate Z axis with Z max endstop
- 12.138.3 M207 (Repetier): Set jerk without saving to EEPROM
- 12.139 M208: Firmware dependent
- 12.139.1 M208: Set unretract length
- 12.139.2 M208 (RepRapFirmware): Set axis max travel
- 12.140 M209: Enable automatic retract
- 12.141 M210: Set homing feedrates
- 12.142 M211: Disable/Enable software endstops
- 12.143 M212: Set Bed Level Sensor Offset
- 12.144 M214: Set Arc configuration values
- 12.145 M217: Toolchange Parameters
- 12.146 M218: Set Hotend Offset
- 12.147 M220: Set speed factor override percentage
- 12.148 M221: Set extrude factor override percentage
- 12.149 M220: Turn off AUX V1.0.5
- 12.150 M221: Turn on AUX V1.0.5
- 12.151 M222: Set speed of fast XY moves
- 12.152 M223: Set speed of fast Z moves
- 12.153 M224: Enable extruder during fast moves

- 12.154 M225: Disable on extruder during fast moves
- 12.155 M226: G-code Initiated Pause
- 12.156 M226: Wait for pin state
- 12.157 M227: Enable Automatic Reverse and Prime
- 12.158 M228: Disable Automatic Reverse and Prime
- 12.159 M229: Enable Automatic Reverse and Prime
- 12.160 M230: Disable / Enable Wait for Temperature Change
- 12.161 M231: Set OPS parameter
- 12.162 M232: Read and reset max. advance values
- 12.163 M240: Trigger camera
- 12.164 M240: Start conveyor belt motor / Echo off
- 12.165 M241: Stop conveyor belt motor / echo on
- 12.166 M245: Start cooler
- 12.167 M246: Stop cooler
- 12.168 M250: Set LCD contrast
- 12.169 M256: Set LCD brightness
- 12.170 M251: Measure Z steps from homing stop (Delta printers)
- 12.171 M260: i2c Send Data
- 12.172 M261: i2c Request Data
- 12.173 M280: Set servo position
- 12.174 M281: Set Servo Angles
- 12.175 M282: Detach Servo
- 12.176 M290: Babystepping
- 12.177 M291: Display message and optionally wait for response
- 12.178 M292: Acknowledge message
- 12.179 M293: Babystep Z+
- 12.180 M294: Babystep Z-
- 12.181 M300: Play beep sound
- 12.182 M301: Set PID parameters
 - 12.182.1 MK4duo
 - 12.182.2 Marlin
 - 12.182.3 RepRapFirmware 1.15 onwards
 - 12.182.4 RepRapFirmware 1.09 to 1.14 inclusive
 - 12.182.5 Smoothie
 - 12.182.6 Other implementations
 - 12.182.7 Teacup
- 12.183 M302: Allow cold extrudes
- 12.184 M303: Run PID tuning
- 12.185 M304: Set PID parameters - Bed
 - 12.185.1 M304 in RepRapPro version of Marlin: Set thermistor values
- 12.186 M305: Set thermistor and ADC parameters
- 12.187 M306: Set home offset calculated from toolhead position
- 12.188 M307: Set or report heating process parameters
- 12.189 M308: Set or report sensor parameters
- 12.190 M309: Set or report heater feedforward
- 12.191 M310: Temperature model settings

12.192 M320: Activate autolevel (Repetier)
12.193 M321: Deactivate autolevel (Repetier)
12.194 M322: Reset autolevel matrix (Repetier)
12.195 M323: Distortion correction on/off (Repetier)
12.196 M340: Control the servos
12.197 M350: Set microstepping mode
12.198 M351: Toggle MS1 MS2 pins directly
12.199 M355: Turn case lights on/off
12.200 M360: Report firmware configuration
12.201 SCARA calibration codes (Morgan)
12.202 M360: Move to Theta 0 degree position
12.203 M361: Move to Theta 90 degree position
12.204 M362: Move to Psi 0 degree position
12.205 M363: Move to Psi 90 degree position
12.206 M364: Move to Psi + Theta 90 degree position
12.207 M365: SCARA scaling factor
12.208 M366: SCARA convert trim
12.209 M370: Morgan manual bed level - clear map
12.210 M371: Move to next calibration position
12.211 M372: Record calibration value, and move to next position
12.212 M373: End bed level calibration mode
12.213 M374: Save calibration grid
12.214 M375: Display matrix / Load Matrix
12.215 M376: Set bed compensation taper
12.216 M380: Activate solenoid
12.217 M381: Disable all solenoids
12.218 M400: Wait for current moves to finish
12.219 M401: Deploy Z Probe
12.220 M402: Stow Z Probe
12.221 M403: Set filament type (material) for particular extruder and notify the MMU
12.222 M404: Filament diameter
12.223 M405: Filament Sensor on
12.224 M406: Filament Sensor off
12.225 M407: Display filament diameter
12.226 M408: Report JSON-style response
12.227 M409: Query object model
12.228 M410: Quick-Stop
12.229 M412: Disable Filament Runout Detection
12.230 M413: Power-Loss Recovery
12.231 M415: Host Rescue
12.232 M416: Power loss
12.233 M420: Set RGB Colors as PWM (MachineKit)
12.234 M420: Leveling On/Off/Fade (Marlin)
12.235 M421: Set a Mesh Bed Leveling Z coordinate
12.236 M422: Set a G34 Point
12.237 M423: X-Axis Twist Compensation

- 12.238 M424: Global Z Offset
- 12.239 M425: Backlash Correction
- 12.240 M450: Report Printer Mode
- 12.241 M451: Select FFF Printer Mode
- 12.242 M452: Select Laser Printer Mode
- 12.243 M453: Select CNC Printer Mode
- 12.244 M460: Define temperature range for thermistor-controlled fan
- 12.245 M470: Create Directory on SD-Card
- 12.246 M471: Rename File/Directory on SD-Card
- 12.247 M472: Delete File/Directory on SD-Card
- 12.248 M486: Cancel Object
- 12.249 M493: Fixed-Time Motion Control
- 12.250 M500: Store parameters in non-volatile storage
- 12.251 M501: Read parameters from EEPROM
- 12.252 M502: Restore Default Settings
- 12.253 M503: Report Current Settings
- 12.254 M504: Validate EEPROM
- 12.255 M505: Firmware dependent
- 12.255.1 M505: Clear EEPROM and RESET Printer
- 12.255.2 M505: Set configuration file folder
- 12.255.3 M505: Set a named EEPROM value
- 12.256 M509: Force language selection
- 12.257 M510: Lock Machine
- 12.258 M511: Unlock Machine with Passcode
- 12.259 M512: Set Passcode
- 12.260 M513: Remove Password
- 12.261 M524: Abort SD Printing
- 12.262 M530: Enable printing mode
- 12.263 M531: Set print name
- 12.264 M532: Set print progress
- 12.265 M540: Set MAC address
- 12.266 M540 in Marlin: Enable/Disable "Stop SD Print on Endstop Hit"
- 12.267 M544: Gcode Parser Options
- 12.268 M550: Set Name
- 12.269 M551: Set Password
- 12.270 M552: Set IP address, enable/disable network interface
- 12.271 M553: Set Netmask
- 12.272 M554: Set Gateway and/or DNS server
- 12.273 M555: Set compatibility
- 12.274 M556: Axis compensation
- 12.275 M557: Set Z probe point or define probing grid
- 12.276 M558: Set Z probe type
- 12.277 M558.1: Calibrate height vs. reading for analog Z probe
- 12.278 M558.2: Set, report or calibrate drive current for analog Z probe
- 12.279 M559: Upload configuration file
- 12.280 M560: Upload web page file

- 12.281 M561: Set Identity Transform
- 12.282 M562: Reset temperature fault
- 12.283 M563: Define or remove a tool
- 12.284 M564: Limit axes
- 12.285 M565: Set Z probe offset
- 12.286 M566: Set allowable instantaneous speed change
- 12.287 M567: Set tool mix ratios
- 12.288 M568: Tool settings
- 12.289 M568: Turn off/on tool mix ratios (obsolete meaning in old RepRapFirmware versions)
- 12.290 M569: Stepper driver control
- 12.291 M569.1: Stepper driver closed loop configuration
- 12.292 M569.2: Read or write any stepper driver register
- 12.293 M569.3: Read Motor Driver Encoder
- 12.294 M569.4: Set Motor Driver Torque Mode
- 12.295 M569.5: Collect Data from Closed-loop Driver
- 12.296 M569.6: Execute Closed-loop Driver Tuning Move
- 12.297 M569.7: Configure motor brake port
- 12.298 M569.8: Read Axis Force
- 12.299 M569.9: Sets the driver sense resistor and maximum current
- 12.300 M570: Configure heater fault detection
- 12.301 M571: Set output on extrude
- 12.302 M572: Set or report extruder pressure advance
- 12.303 M573: Report heater PWM
- 12.304 M574: Set endstop configuration
- 12.305 M575: Set serial comms parameters
- 12.306 M576: Set SPI comms parameters
- 12.307 M577: Wait until endstop is triggered
- 12.308 M578: Fire inkjet bits
- 12.309 M579: Scale Cartesian axes
- 12.310 M580: Select Roland
- 12.311 M581: Configure external trigger
- 12.312 M582: Check external trigger
- 12.313 M584: Set drive mapping
- 12.314 M585: Probe Tool
- 12.315 M586: Configure network protocols
- 12.316 M586.4: Configure MQTT server
- 12.317 M587: Store WiFi host network in list, or list stored networks
- 12.318 M588: Forget WiFi host network
- 12.319 M589: Configure access point parameters
- 12.320 M590: Report current tool type and index
- 12.321 M591: Configure filament monitoring
- 12.322 M592: Configure nonlinear extrusion
- 12.323 M593: Configure Input Shaping
- 12.324 M594: Enter/Leave Height Following mode
- 12.325 M595: Set movement queue length
- 12.326 M596: Select movement queue number

- 12.327 M597: Collision avoidance
- 12.328 M598: Sync motion systems
- 12.329 M599: Define keepout zone
- 12.330 M600: Set line cross section
- 12.331 M600: Filament change pause
- 12.332 M601: Pause print
- 12.333 M602: Resume print
- 12.334 M603: Stop print (Prusa i3)
- 12.335 M603: Configure Filament Change
- 12.336 M605: Set dual x-carriage movement mode
- 12.337 M650: Set peel move parameters
- 12.338 M651: Execute peel move
- 12.339 M665: Set delta configuration
- 12.340 M666: Set delta endstop adjustment
- 12.341 M667: Select CoreXY mode
- 12.342 M668: Set Z-offset compensations polynomial
- 12.343 M669: Set kinematics type and kinematics parameters
- 12.344 M670: Set IO port bit mapping
- 12.345 M671: Define positions of Z leadscrews or bed leveling screws
- 12.346 M672: Program Z probe
- 12.347 M673: Align plane on rotary axis
- 12.348 M674: Set Z to center point
- 12.349 M675: Find center of cavity
- 12.350 M700: Level plate
- 12.351 M701: Load filament
- 12.352 M702: Unload filament
- 12.353 M703: Configure Filament
- 12.354 M704: Preload_to_MMU
- 12.355 M705: Eject filament
- 12.356 M706: Cut filament
- 12.357 M707: Read from MMU register
- 12.358 M708: Write to MMU register
- 12.359 M709: MMU reset
- 12.360 M710: Firmware dependent
- 12.360.1 M710: Controller Fan settings
- 12.360.2 M710: Erase the EEPROM and reset the board
- 12.361 M750: Enable 3D scanner extension
- 12.362 M751: Register 3D scanner extension over USB
- 12.363 M752: Start 3D scan
- 12.364 M753: Cancel current 3D scanner action
- 12.365 M754: Calibrate 3D scanner
- 12.366 M755: Set alignment mode for 3D scanner
- 12.367 M756: Shutdown 3D scanner
- 12.368 M800: Fire start print procedure
- 12.369 M801: Fire end print procedure
- 12.370 M808: Set or Goto Repeat Marker

12.370.1 M808 in Marlin 2.0.8
12.371 M810-M819: G-code macros stored in memory or flash not filename
12.372 M851: Set Z-Probe Offset
12.372.1 M851 in Marlin 1.0.2
12.372.2 M851 in Marlin 1.1.0
12.372.3 M851 in Marlin 2.0.0
12.372.4 M851 in MK4duo 4.3.25
12.372.5 M851 in RepRapFirmware 2.02 and later
12.373 M860 Wait for Probe Temperature
12.374 M861 Set Probe Thermal Compensation
12.375 M862: Print checking
12.375.1 M862.1: Check nozzle diameter
12.375.2 M862.2: Check model code
12.375.3 M862.3: Model name
12.375.4 M862.4: Firmware version
12.375.5 M862.5: Gcode level
12.375.6 M862.6: Firmware features
12.376 M871: PTC Configuration
12.377 M876: Dialog handling
12.378 M890 Run User Gcode
12.379 M900 Set Linear Advance Scaling Factors
12.380 M905: Set local date and time
12.381 M906: Set motor currents
12.382 M907: Set digital trimpot motor
12.383 M908: Control digital trimpot directly
12.384 M909: Set microstepping
12.385 M910: Set decay mode
12.386 M910: TMC2130 init
12.387 M911: Configure auto save on loss of power ("power panic")
12.388 M911: Set TMC2130 holding currents
12.389 M912: Set electronics temperature monitor adjustment
12.390 M912: Set TMC2130 running currents
12.391 M913: Set motor percentage of normal current
12.392 M913: Print TMC2130 currents
12.393 M914: Set/Get Expansion Voltage Level Translator
12.394 M914: Set TMC2130 normal mode
12.395 M915: Configure motor stall detection
12.396 M915: Set TMC2130 silent mode
12.397 M916: Resume print after power failure
12.398 M916: Set TMC2130 Stallguard sensitivity threshold
12.399 M917: Set motor standstill current reduction
12.400 M917: Set TMC2130 PWM amplitude offset (pwm_ampl)
12.401 M918: Configure direct-connect display
12.402 M918: Set TMC2130 PWM amplitude gradient (pwm_grad)
12.403 M928: Start SD logging
12.404 M929: Start/stop event logging to SD card

- 12.405 M950: Create heater, fan or GPIO/servo device
- 12.406 M951: Set height following mode parameters
- 12.407 M952: Set CAN expansion board address and/or normal data rate
- 12.408 M953: Set CAN-FD bus fast data rate
- 12.409 M954: Configure as CAN expansion board
- 12.410 M955: Configure Accelerometer
- 12.411 M956: Collect accelerometer data and write to file
- 12.412 M957: Raise event
- 12.413 M958: Excite harmonic vibration
- 12.414 M995: Calibrate Touch Screen
- 12.415 M997: Perform in-application firmware update
- 12.416 M998: Request resend of line
- 12.417 M999: Restart after being stopped by error

Anexo G – Datasheet Motor de Passo 17HS4401



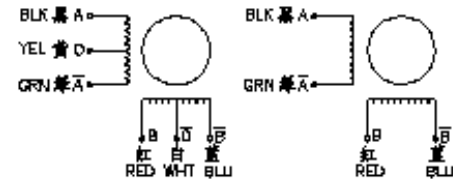
HB Stepper Motor Catalog

MotionKing (China) Motor Industry Co., Ltd.

**2 Phase Hybrid Stepper Motor
17H2A series-Size 42mm(1.8 degree)**



Wiring Diagram:



UNI-POLAR(6 LEADS)

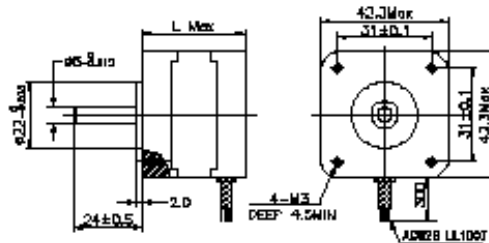
BI-POLAR(4LEADS)

Electrical Specifications:

Series Model	Old P/N	Motor Length (mm)	Rated Current (A)	Phase Resistance (ohm)	Phase Inductance (mH)	Holding Torque (N.cm)	Detent Torque (N.cm Max)	Rotor Inertia (g.cm ²)	Lead Wire (No.)	Motor Weight (g)
17H2A2406	17HS2408	28	0.6	8	10	12	1.6	34	4	150
17H2A3413	17HS3401	34	1.3	2.4	2.8	28	1.6	34	4	220
17H2A3417	17HS3410	34	1.7	1.2	1.8	28	1.6	34	4	220
17H2A3404	17HS3430	34	0.4	30	35	28	1.6	34	4	220
17H2A3604	17HS3630	34	0.4	30	18	21	1.6	34	6	220
17H2A4417	17HS4401	40	1.7	1.5	2.8	40	2.2	54	4	280
17H2A4413	17HS4402	40	1.3	2.5	5.0	40	2.2	54	4	280
17H2A4612	17HS4612	40	1.2	4.1	3.5	28	2.2	54	6	280
17H2A4604	17HS4630	40	0.4	30	28	28	2.2	54	6	280
17H2A8417	17HS8401	48	1.7	1.8	3.2	52	2.6	68	4	350
17H2A8413	17HS8402	48	1.3	3.2	5.5	52	2.6	68	4	350
17H2A8423	17HS8403	48	2.3	1.2	1.6	52	2.6	68	4	350
17H2A8604	17HS8630	48	0.4	30	38	34	2.6	68	6	350
17H2A9612	17HS9612	58	1.2	4.5	5.5	54	3.2	75	6	450
17H2A9425	17HS9425	58	2.5	1.7	2.7	76	3.2	75	4	450

*Note: We can manufacture products according to customer's requirements.

Dimensions: unit=mm



Motor Length:

Model	Length
17HS2XXX	28 mm
17HS3XXX	34 mm
17HS4XXX	40 mm
17HS8XXX	48 mm

APENDICES

Apêndice A – Programa Marlin Configure.H

```

/**
 * Marlin 3D Printer Firmware
 * Copyright (C) 2016 MarlinFirmware [https://github.com/MarlinFirmware/Marlin]
 *
 * Based on Sprinter and grbl.
 * Copyright (C) 2011 Camiel Gubbels / Erik van der Zalm
 *
 * This program is free software: you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation, either version 3 of the License, or
 * (at your option) any later version.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with this program. If not, see <http://www.gnu.org/licenses/>.
 */

/**
 * Configuration.h
 *
 * Basic settings such as:
 *
 * - Type of electronics
 * - Type of temperature sensor
 * - Printer geometry
 * - Endstop configuration
 * - LCD controller
 * - Extra features
 *
 * Advanced settings can be found in Configuration_adv.h
 */
#ifndef CONFIGURATION_H
#define CONFIGURATION_H
#define CONFIGURATION_H_VERSION 010109

//=====
//===== Getting Started =====
//=====

/**
 * Here are some standard links for getting your machine calibrated:
 *
 * http://reprap.org/wiki/Calibration
 * http://youtu.be/wAL9d7Fglnk
 * http://calculator.josefprusa.cz
 * http://reprap.org/wiki/Triffid_Hunter%27s_Calibration_Guide
 * http://www.thingiverse.com/thing:5573
 * https://sites.google.com/site/repraplogphase/calibration-of-your-reprap
 * http://www.thingiverse.com/thing:298812
 */

```

```

//=====
//===== DELTA Printer =====
//=====
// For a Delta printer start with one of the configuration files in the
// example_configurations/delta directory and customize for your machine.
//

//=====
//===== SCARA Printer =====
//=====
// For a SCARA printer start with the configuration files in
// example_configurations/SCARA and customize for your machine.
//

//=====
//===== HANGPRINTER =====
//=====
// For a Hangprinter start with the configuration file in the
// example_configurations/hangprinter directory and customize for your machine.
//

// @section info

// User-specified version info of this build to display in [Pronterface, etc] terminal window during
// startup. Implementation of an idea by Prof Braino to inform user that any changes made to this
// build by the user have been successfully uploaded into firmware.
#define STRING_CONFIG_H_AUTHOR "Arthur, Manu" // Who made the changes.
#define SHOW_BOOTSCREEN
#define STRING_SPLASH_LINE1 SHORT_BUILD_VERSION // will be shown during bootup in line 1
#define STRING_SPLASH_LINE2 WEBSITE_URL // will be shown during bootup in line 2

/**
 * *** VENDORS PLEASE READ ***
 *
 * Marlin allows you to add a custom boot image for Graphical LCDs.
 * With this option Marlin will first show your custom screen followed
 * by the standard Marlin logo with version number and web URL.
 *
 * We encourage you to take advantage of this new feature and we also
 * respectfully request that you retain the unmodified Marlin boot screen.
 */

// Enable to show the bitmap in Marlin/_Bootscreen.h on startup.
//#define SHOW_CUSTOM_BOOTSCREEN

// Enable to show the bitmap in Marlin/_Statusscreen.h on the status screen.
//#define CUSTOM_STATUS_SCREEN_IMAGE

// @section machine

/**
 * Select the serial port on the board to use for communication with the host.
 * This allows the connection of wireless adapters (for instance) to non-default port pins.
 * Serial port 0 is always used by the Arduino bootloader regardless of this setting.
 *
 * :[0, 1, 2, 3, 4, 5, 6, 7]
 */
#define SERIAL_PORT 0

/**
 * This setting determines the communication speed of the printer.
 *

```

```

* 250000 works in most cases, but you might try a lower speed if
* you commonly experience drop-outs during host printing.
* You may try up to 1000000 to speed up SD file transfer.
*
* :[2400, 9600, 19200, 38400, 57600, 115200, 250000, 500000, 1000000]
*/
#define BAUDRATE 115200

// Enable the Bluetooth serial interface on AT90USB devices
//#define BLUETOOTH

// The following define selects which electronics board you have.
// Please choose the name from boards.h that matches your setup
#ifndef MOTHERBOARD
  #define MOTHERBOARD BOARD_RAMPS_14_EFB
#endif

// Optional custom name for your RepStrap or other custom machine
// Displayed in the LCD "Ready" message
#define CUSTOM_MACHINE_NAME "Arthur_Manu"

// Define this to set a unique identifier for this printer, (Used by some programs to differentiate between machines)
// You can use an online service to generate a random UUID. (eg http://www.uuidgenerator.net/version4)
//#define MACHINE_UUID "00000000-0000-0000-0000-000000000000"

// @section extruder

// This defines the number of extruders
// :[1, 2, 3, 4, 5]
#define EXTRUDERS 1

// Generally expected filament diameter (1.75, 2.85, 3.0, ...). Used for Volumetric, Filament Width Sensor, etc.
#define DEFAULT_NOMINAL_FILAMENT_DIA 1.75

// For Cyclops or any "multi-extruder" that shares a single nozzle.
//#define SINGLENOZZLE

/**
 * Průša MK2 Single Nozzle Multi-Material Multiplexer, and variants.
 *
 * This device allows one stepper driver on a control board to drive
 * two to eight stepper motors, one at a time, in a manner suitable
 * for extruders.
 *
 * This option only allows the multiplexer to switch on tool-change.
 * Additional options to configure custom E moves are pending.
 */
//#define MK2_MULTIPLEXER
#if ENABLED(MK2_MULTIPLEXER)
  // Override the default DIO selector pins here, if needed.
  // Some pins files may provide defaults for these pins.
  #define E_MUX0_PIN 40 // Always Required
  #define E_MUX1_PIN 42 // Needed for 3 to 8 steppers
  #define E_MUX2_PIN 44 // Needed for 5 to 8 steppers
#endif

// A dual extruder that uses a single stepper motor
//#define SWITCHING_EXTRUDER
#if ENABLED(SWITCHING_EXTRUDER)
  #define SWITCHING_EXTRUDER_SERVO_NR 0
  #define SWITCHING_EXTRUDER_SERVO_ANGLES { 0, 90 } // Angles for E0, E1[, E2, E3]
  #if EXTRUDERS > 3

```

```

#define SWITCHING_EXTRUDER_E23_SERVO_NR 1
#endif
#endif

// A dual-nozzle that uses a servomotor to raise/lower one of the nozzles
// #define SWITCHING_NOZZLE
// #if ENABLED(SWITCHING_NOZZLE)
//   #define SWITCHING_NOZZLE_SERVO_NR 0
//   #define SWITCHING_NOZZLE_SERVO_ANGLES { 0, 90 } // Angles for E0, E1
//   #define HOTEND_OFFSET_Z { 0.0, 0.0 }
// #endif

/**
 * Two separate X-carriages with extruders that connect to a moving part
 * via a magnetic docking mechanism. Requires SOL1_PIN and SOL2_PIN.
 */
// #define PARKING_EXTRUDER
// #if ENABLED(PARKING_EXTRUDER)
//   #define PARKING_EXTRUDER_SOLENOIDS_INVERT // If enabled, the solenoid is NOT magnetized with
//   applied voltage
//   #define PARKING_EXTRUDER_SOLENOIDS_PINS_ACTIVE LOW // LOW or HIGH pin signal energizes the coil
//   #define PARKING_EXTRUDER_SOLENOIDS_DELAY 250 // Delay (ms) for magnetic field. No delay if 0 or not
//   defined.
//   #define PARKING_EXTRUDER_PARKING_X { -78, 184 } // X positions for parking the extruders
//   #define PARKING_EXTRUDER_GRAB_DISTANCE 1 // mm to move beyond the parking point to grab the
//   extruder
//   #define PARKING_EXTRUDER_SECURITY_RAISE 5 // Z-raise before parking
//   #define HOTEND_OFFSET_Z { 0.0, 1.3 } // Z-offsets of the two hotends. The first must be 0.
// #endif

/**
 * "Mixing Extruder"
 * - Adds G-codes M163 and M164 to set and "commit" the current mix factors.
 * - Extends the stepping routines to move multiple steppers in proportion to the mix.
 * - Optional support for Repetier Firmware's 'M164 S<index>' supporting virtual tools.
 * - This implementation supports up to two mixing extruders.
 * - Enable DIRECT_MIXING_IN_G1 for M165 and mixing in G1 (from Pia Taubert's reference implementation).
 */
// #define MIXING_EXTRUDER
// #if ENABLED(MIXING_EXTRUDER)
//   #define MIXING_STEPPERS 2 // Number of steppers in your mixing extruder
//   #define MIXING_VIRTUAL_TOOLS 16 // Use the Virtual Tool method with M163 and M164
//   #define DIRECT_MIXING_IN_G1 // Allow ABCDHI mix factors in G1 movement commands
// #endif

// Offset of the extruders (uncomment if using more than one and relying on firmware to position when changing).
// The offset has to be X=0, Y=0 for the extruder 0 hotend (default extruder).
// For the other hotends it is their distance from the extruder 0 hotend.
// #define HOTEND_OFFSET_X { 0.0, 20.00 } // (in mm) for each extruder, offset of the hotend on the X axis
// #define HOTEND_OFFSET_Y { 0.0, 5.00 } // (in mm) for each extruder, offset of the hotend on the Y axis

// @section machine

/**
 * Select your power supply here. Use 0 if you haven't connected the PS_ON_PIN
 *
 * 0 = No Power Switch
 * 1 = ATX
 * 2 = X-Box 360 203Watts (the blue wire connected to PS_ON and the red wire to VCC)
 *
 * :{ 0:'No power switch', 1:'ATX', 2:'X-Box 360' }
 */

```

```

#define POWER_SUPPLY 1

#if POWER_SUPPLY > 0
// Enable this option to leave the PSU off at startup.
// Power to steppers and heaters will need to be turned on with M80.
//#define PS_DEFAULT_OFF

//#define AUTO_POWER_CONTROL // Enable automatic control of the PS_ON pin
#if ENABLED(AUTO_POWER_CONTROL)
#define AUTO_POWER_FANS // Turn on PSU if fans need power
#define AUTO_POWER_E_FANS
#define AUTO_POWER_CONTROLLERFAN
#define POWER_TIMEOUT 30
#endif

#endif

// @section temperature

//=====
//----- Thermal Settings -----
//=====

/**
 * -NORMAL IS 4.7kohm PULLUP!-- 1kohm pullup can be used on hotend sensor, using correct resistor and table
 *
 * Temperature sensors available:
 *
 * -4 : thermocouple with AD8495
 * -3 : thermocouple with MAX31855 (only for sensor 0)
 * -2 : thermocouple with MAX6675 (only for sensor 0)
 * -1 : thermocouple with AD595
 * 0 : not used
 * 1 : 100k thermistor - best choice for EPCOS 100k (4.7k pullup)
 * 2 : 200k thermistor - ATC Semitec 204GT-2 (4.7k pullup)
 * 3 : Mendel-parts thermistor (4.7k pullup)
 * 4 : 10k thermistor !! do not use it for a hotend. It gives bad resolution at high temp. !!
 * 5 : 100K thermistor - ATC Semitec 104GT-2/104NT-4-R025H42G (Used in ParCan & J-Head) (4.7k pullup)
 * 501 : 100K Zonestar (Tronxy X3A) Thermistor
 * 6 : 100k EPCOS - Not as accurate as table 1 (created using a fluke thermocouple) (4.7k pullup)
 * 7 : 100k Honeywell thermistor 135-104LAG-J01 (4.7k pullup)
 * 71 : 100k Honeywell thermistor 135-104LAF-J01 (4.7k pullup)
 * 8 : 100k 0603 SMD Vishay NTCS0603E3104FXT (4.7k pullup)
 * 9 : 100k GE Sensing AL03006-58.2K-97-G1 (4.7k pullup)
 * 10 : 100k RS thermistor 198-961 (4.7k pullup)
 * 11 : 100k beta 3950 1% thermistor (4.7k pullup)
 * 12 : 100k 0603 SMD Vishay NTCS0603E3104FXT (4.7k pullup) (calibrated for Makibox hot bed)
 * 13 : 100k Hisens 3950 1% up to 300°C for hotend "Simple ONE" & "Hotend "All In ONE"
 * 15 : 100k thermistor calibration for JGAurora A5 hotend
 * 20 : the PT100 circuit found in the Ultimainboard V2.x
 * 60 : 100k Maker's Tool Works Kapton Bed Thermistor beta=3950
 * 66 : 4.7M High Temperature thermistor from Dyze Design
 * 70 : the 100K thermistor found in the bq Hephestos 2
 * 75 : 100k Generic Silicon Heat Pad with NTC 100K MGB18-104F39050L32 thermistor
 *
 * 1k ohm pullup tables - This is atypical, and requires changing out the 4.7k pullup for 1k.
 * (but gives greater accuracy and more stable PID)
 * 51 : 100k thermistor - EPCOS (1k pullup)
 * 52 : 200k thermistor - ATC Semitec 204GT-2 (1k pullup)
 * 55 : 100k thermistor - ATC Semitec 104GT-2 (Used in ParCan & J-Head) (1k pullup)
 *
 * 1047 : Pt1000 with 4k7 pullup

```

```

* 1010 : Pt1000 with 1k pullup (non standard)
* 147 : Pt100 with 4k7 pullup
* 110 : Pt100 with 1k pullup (non standard)
*
* Use these for Testing or Development purposes. NEVER for production machine.
* 998 : Dummy Table that ALWAYS reads 25°C or the temperature defined below.
* 999 : Dummy Table that ALWAYS reads 100°C or the temperature defined below.
*
* { '0': "Not used", '1':"100k / 4.7k - EPCOS", '2':"200k / 4.7k - ATC Semitec 204GT-2", '3':"Mendel-parts / 4.7k",
'4':"10k !! do not use for a hotend. Bad resolution at high temp. !!", '5':"100K / 4.7k - ATC Semitec 104GT-2 (Used in
ParCan & J-Head)", '501':"100K Zonestar (Tronxy X3A)", '6':"100k / 4.7k EPCOS - Not as accurate as Table 1",
'7':"100k / 4.7k Honeywell 135-104LAG-J01", '8':"100k / 4.7k 0603 SMD Vishay NTCS0603E3104FXT", '9':"100k /
4.7k GE Sensing AL03006-58.2K-97-G1", '10':"100k / 4.7k RS 198-961", '11':"100k / 4.7k beta 3950 1%", '12':"100k /
4.7k 0603 SMD Vishay NTCS0603E3104FXT (calibrated for Makibox hot bed)", '13':"100k Hisens 3950 1% up to
300°C for hotend 'Simple ONE ' & hotend 'All In ONE'", '20':"PT100 (Ultimainboard V2.x)", '51':"100k / 1k - EPCOS",
'52':"200k / 1k - ATC Semitec 204GT-2", '55':"100k / 1k - ATC Semitec 104GT-2 (Used in ParCan & J-Head)",
'60':"100k Maker's Tool Works Kapton Bed Thermistor beta=3950", '66':"Dyze Design 4.7M High Temperature
thermistor", '70':"the 100K thermistor found in the bq Hephestos 2", '71':"100k / 4.7k Honeywell 135-104LAF-J01",
'147':"Pt100 / 4.7k", '1047':"Pt1000 / 4.7k", '110':"Pt100 / 1k (non-standard)", '1010':"Pt1000 / 1k (non standard)",
'-4':"Thermocouple + AD8495", '-3':"Thermocouple + MAX31855 (only for sensor 0)", '-2':"Thermocouple + MAX6675
(only for sensor 0)", '-1':"Thermocouple + AD595", '998':"Dummy 1", '999':"Dummy 2" }
*/
#define TEMP_SENSOR_0 1
#define TEMP_SENSOR_1 0
#define TEMP_SENSOR_2 0
#define TEMP_SENSOR_3 0
#define TEMP_SENSOR_4 0
#define TEMP_SENSOR_BED 1
#define TEMP_SENSOR_CHAMBER 0

// Dummy thermistor constant temperature readings, for use with 998 and 999
#define DUMMY_THERMISTOR_998_VALUE 25
#define DUMMY_THERMISTOR_999_VALUE 100

// Use temp sensor 1 as a redundant sensor with sensor 0. If the readings
// from the two sensors differ too much the print will be aborted.
//#define TEMP_SENSOR_1_AS_REDUNDANT
#define MAX_REDUNDANT_TEMP_SENSOR_DIFF 10

// Extruder temperature must be close to target for this long before M109 returns success
#define TEMP_RESIDENCY_TIME 10 // (seconds)
#define TEMP_HYSTERESIS 3 // (degC) range of +/- temperatures considered "close" to the target one
#define TEMP_WINDOW 1 // (degC) Window around target to start the residency timer x degC early.

// Bed temperature must be close to target for this long before M190 returns success
#define TEMP_BED_RESIDENCY_TIME 10 // (seconds)
#define TEMP_BED_HYSTERESIS 3 // (degC) range of +/- temperatures considered "close" to the target one
#define TEMP_BED_WINDOW 1 // (degC) Window around target to start the residency timer x degC early.

// The minimal temperature defines the temperature below which the heater will not be enabled It is used
// to check that the wiring to the thermistor is not broken.
// Otherwise this would lead to the heater being powered on all the time.
#define HEATER_0_MINTEMP 5
#define HEATER_1_MINTEMP 5
#define HEATER_2_MINTEMP 5
#define HEATER_3_MINTEMP 5
#define HEATER_4_MINTEMP 5
#define BED_MINTEMP 5

// When temperature exceeds max temp, your heater will be switched off.
// This feature exists to protect your hotend from overheating accidentally, but *NOT* from thermistor short/failure!
// You should use MINTEMP for thermistor short/failure protection.

```

```

#define HEATER_0_MAXTEMP 265
#define HEATER_1_MAXTEMP 275
#define HEATER_2_MAXTEMP 275
#define HEATER_3_MAXTEMP 275
#define HEATER_4_MAXTEMP 275
#define BED_MAXTEMP 150

//=====
//----- PID Settings -----
//=====
// PID Tuning Guide here: http://reprap.org/wiki/PID\_Tuning

// Comment the following line to disable PID and enable bang-bang.
#define PIDTEMP
#define BANG_MAX 255 // Limits current to nozzle while in bang-bang mode; 255=full current
#define PID_MAX BANG_MAX // Limits current to nozzle while PID is active (see PID_FUNCTIONAL_RANGE
below); 255=full current
#define PID_K1 0.95 // Smoothing factor within any PID loop
#if ENABLED(PIDTEMP)
  //#define PID_AUTOTUNE_MENU // Add PID Autotune to the LCD "Temperature" menu to run M303 and apply the
  result.
  //#define PID_DEBUG // Sends debug data to the serial port.
  //#define PID_OPENLOOP 1 // Puts PID in open loop. M104/M140 sets the output power from 0 to PID_MAX
  //#define SLOW_PWM_HEATERS // PWM with very low frequency (roughly 0.125Hz=8s) and minimum state time of
  approximately 1s useful for heaters driven by a relay
  //#define PID_PARAMS_PER_HOTEND // Uses separate PID parameters for each extruder (useful for mismatched
  extruders)
  // Set/get with gcode: M301 E[extruder number, 0-2]
  #define PID_FUNCTIONAL_RANGE 10 // If the temperature difference between the target temperature and the
  actual temperature
  // is more than PID_FUNCTIONAL_RANGE then the PID will be shut off and the heater will be
  set to min/max.

  // If you are using a pre-configured hotend then you can use one of the value sets by uncommenting it

  // Ultimaker
  #define DEFAULT_Kp 22.2
  #define DEFAULT_Ki 1.08
  #define DEFAULT_Kd 114

  // MakerGear
  //#define DEFAULT_Kp 7.0
  //#define DEFAULT_Ki 0.1
  //#define DEFAULT_Kd 12

  // Mendel Parts V9 on 12V
  //#define DEFAULT_Kp 63.0
  //#define DEFAULT_Ki 2.25
  //#define DEFAULT_Kd 440
#endif // PIDTEMP

//=====
//----- PID > Bed Temperature Control -----
//=====

/**
 * PID Bed Heating
 *
 * If this option is enabled set PID constants below.
 * If this option is disabled, bang-bang will be used and BED_LIMIT_SWITCHING will enable hysteresis.
 *

```

```

* The PID frequency will be the same as the extruder PWM.
* If PID_dT is the default, and correct for the hardware/configuration, that means 7.689Hz,
* which is fine for driving a square wave into a resistive load and does not significantly
* impact FET heating. This also works fine on a Fetek SSR-10DA Solid State Relay into a 250W
* heater. If your configuration is significantly different than this and you don't understand
* the issues involved, don't use bed PID until someone else verifies that your hardware works.
*/
##define PIDTEMPBED

##define BED_LIMIT_SWITCHING

/**
 * Max Bed Power
 * Applies to all forms of bed control (PID, bang-bang, and bang-bang with hysteresis).
 * When set to any value below 255, enables a form of PWM to the bed that acts like a divider
 * so don't use it unless you are OK with PWM on your bed. (See the comment on enabling PIDTEMPBED)
 */
#define MAX_BED_POWER 255 // limits duty cycle to bed; 255=full current

#if ENABLED(PIDTEMPBED)

  ##define PID_BED_DEBUG // Sends debug data to the serial port.

  //120V 250W silicone heater into 4mm borosilicate (MendelMax 1.5+)
  //from FOPDT model - kp=.39 Tp=405 Tdead=66, Tc set to 79.2, aggressive factor of .15 (vs .1, 1, 10)
  #define DEFAULT_bedKp 10.00
  #define DEFAULT_bedKi .023
  #define DEFAULT_bedKd 305.4

  //120V 250W silicone heater into 4mm borosilicate (MendelMax 1.5+)
  //from pidautotune
  ##define DEFAULT_bedKp 97.1
  ##define DEFAULT_bedKi 1.41
  ##define DEFAULT_bedKd 1675.16

  // FIND YOUR OWN: "M303 E-1 C8 S90" to run autotune on the bed at 90 degreesC for 8 cycles.
#endif // PIDTEMPBED

// @section extruder

/**
 * Prevent extrusion if the temperature is below EXTRUDE_MINTEMP.
 * Add M302 to set the minimum extrusion temperature and/or turn
 * cold extrusion prevention on and off.
 *
 * *** IT IS HIGHLY RECOMMENDED TO LEAVE THIS OPTION ENABLED! ***
 */
#define PREVENT_COLD_EXTRUSION
#define EXTRUDE_MINTEMP 170

/**
 * Prevent a single extrusion longer than EXTRUDE_MAXLENGTH.
 * Note: For Bowden Extruders make this large enough to allow load/unload.
 */
#define PREVENT_LENGTHY_EXTRUDE
#define EXTRUDE_MAXLENGTH 200

//===== Thermal Runaway Protection =====
//
/**

```

* Thermal Protection provides additional protection to your printer from damage
 * and fire. Marlin always includes safe min and max temperature ranges which
 * protect against a broken or disconnected thermistor wire.
 *

* The issue: If a thermistor falls out, it will report the much lower
 * temperature of the air in the room, and the the firmware will keep
 * the heater on.
 *

* If you get "Thermal Runaway" or "Heating failed" errors the
 * details can be tuned in Configuration_adv.h
 */

```
#define THERMAL_PROTECTION_HOTENDS // Enable thermal protection for all extruders
#define THERMAL_PROTECTION_BED // Enable thermal protection for the heated bed
```

```
//=====
//----- Mechanical Settings -----
//=====
```

```
// @section machine
```

```
// Uncomment one of these options to enable CoreXY, CoreXZ, or CoreYZ kinematics
// either in the usual order or reversed
```

```
##define COREXY
##define COREXZ
##define COREYZ
##define COREYX
##define COREZX
##define COREZY
```

```
//=====
//----- Endstop Settings -----
//=====
```

```
// @section homing
```

```
// Specify here all the endstop connectors that are connected to any endstop or probe.
// Almost all printers will be using one per axis. Probes will use one or more of the
// extra connectors. Leave undefined any used for non-endstop and non-probe purposes.
```

```
#define USE_XMIN_PLUG
#define USE_YMIN_PLUG
#define USE_ZMIN_PLUG
##define USE_XMAX_PLUG
##define USE_YMAX_PLUG
##define USE_ZMAX_PLUG
```

```
// Enable pullup for all endstops to prevent a floating state
```

```
#define ENDSTOPPULLUPS
#if DISABLED(ENDSTOPPULLUPS)
  // Disable ENDSTOPPULLUPS to set pullups individually
  ##define ENDSTOPPULLUP_XMAX
  ##define ENDSTOPPULLUP_YMAX
  ##define ENDSTOPPULLUP_ZMAX
  ##define ENDSTOPPULLUP_XMIN
  ##define ENDSTOPPULLUP_YMIN
  ##define ENDSTOPPULLUP_ZMIN
  ##define ENDSTOPPULLUP_ZMIN_PROBE
#endif
```

```
// Mechanical endstop with COM to ground and NC to Signal uses "false" here (most common setup).
```

```
#define X_MIN_ENDSTOP_INVERTING true // set to true to invert the logic of the endstop.
#define Y_MIN_ENDSTOP_INVERTING true // set to true to invert the logic of the endstop.
```

```

#define Z_MIN_ENDSTOP_INVERTING true // set to true to invert the logic of the endstop.
#define X_MAX_ENDSTOP_INVERTING true // set to true to invert the logic of the endstop.
#define Y_MAX_ENDSTOP_INVERTING true // set to true to invert the logic of the endstop.
#define Z_MAX_ENDSTOP_INVERTING true // set to true to invert the logic of the endstop.
#define Z_MIN_PROBE_ENDSTOP_INVERTING true // set to true to invert the logic of the probe.

/**
 * Stepper Drivers
 *
 * These settings allow Marlin to tune stepper driver timing and enable advanced options for
 * stepper drivers that support them. You may also override timing options in Configuration_adv.h.
 *
 * A4988 is assumed for unspecified drivers.
 *
 * Options: A4988, DRV8825, LV8729, L6470, TB6560, TB6600, TMC2100,
 *          TMC2130, TMC2130_STANDALONE, TMC2208, TMC2208_STANDALONE,
 *          TMC26X, TMC26X_STANDALONE, TMC2660, TMC2660_STANDALONE,
 *          TMC5130, TMC5130_STANDALONE
 * ;['A4988', 'DRV8825', 'LV8729', 'L6470', 'TB6560', 'TB6600', 'TMC2100', 'TMC2130', 'TMC2130_STANDALONE',
 * 'TMC2208', 'TMC2208_STANDALONE', 'TMC26X', 'TMC26X_STANDALONE', 'TMC2660',
 * 'TMC2660_STANDALONE', 'TMC5130', 'TMC5130_STANDALONE']
 */
#define X_DRIVER_TYPE DRV8825
#define Y_DRIVER_TYPE DRV8825
#define Z_DRIVER_TYPE DRV8825
//#define X2_DRIVER_TYPE A4988
//#define Y2_DRIVER_TYPE A4988
//#define Z2_DRIVER_TYPE A4988
#define E0_DRIVER_TYPE DRV8825
//#define E1_DRIVER_TYPE A4988
//#define E2_DRIVER_TYPE A4988
//#define E3_DRIVER_TYPE A4988
//#define E4_DRIVER_TYPE A4988

// Enable this feature if all enabled endstop pins are interrupt-capable.
// This will remove the need to poll the interrupt pins, saving many CPU cycles.
//#define ENDSTOP_INTERRUPTS_FEATURE

/**
 * Endstop Noise Filter
 *
 * Enable this option if endstops falsely trigger due to noise.
 * NOTE: Enabling this feature means adds an error of +/-0.2mm, so homing
 * will end up at a slightly different position on each G28. This will also
 * reduce accuracy of some bed probes.
 * For mechanical switches, the better approach to reduce noise is to install
 * a 100 nanofarads ceramic capacitor in parallel with the switch, making it
 * essentially noise-proof without sacrificing accuracy.
 * This option also increases MCU load when endstops or the probe are enabled.
 * So this is not recommended. USE AT YOUR OWN RISK.
 * (This feature is not required for common micro-switches mounted on PCBs
 * based on the Makerbot design, since they already include the 100nF capacitor.)
 */
#define ENDSTOP_NOISE_FILTER

//=====
//===== Movement Settings =====
//=====
// @section motion

/**
 * Default Settings

```

```

*
* These settings can be reset by M502
*
* Note that if EEPROM is enabled, saved values will override these.
*/

/**
* With this option each E stepper can have its own factors for the
* following movement settings. If fewer factors are given than the
* total number of extruders, the last value applies to the rest.
*/
#define DISTINCT_E_FACTORS

/**
* Default Axis Steps Per Unit (steps/mm)
* Override with M92
*
* X, Y, Z, E0 [, E1[, E2[, E3[, E4]]]]
*/
#define DEFAULT_AXIS_STEPS_PER_UNIT { 160, 160, 4000, 1368.43 }

/**
* Default Max Feed Rate (mm/s)
* Override with M203
*
* X, Y, Z, E0 [, E1[, E2[, E3[, E4]]]]
*/
#define DEFAULT_MAX_FEEDRATE { 300, 300, 2.5, 25 }
#define DEFAULT_MAX_FEEDRATE { 300, 300, 5, 25 }

/**
* Default Max Acceleration (change/s) change = mm/s
* (Maximum start speed for accelerated moves)
* Override with M201
*
* X, Y, Z, E0 [, E1[, E2[, E3[, E4]]]]
*/
#define DEFAULT_MAX_ACCELERATION { 3000, 3000, 100, 10000 }

/**
* Default Acceleration (change/s) change = mm/s
* Override with M204
*
* M204 P Acceleration
* M204 R Retract Acceleration
* M204 T Travel Acceleration
*/
#define DEFAULT_ACCELERATION 3000 // X, Y, Z and E acceleration for printing moves
#define DEFAULT_RETRACT_ACCELERATION 3000 // E acceleration for retracts
#define DEFAULT_TRAVEL_ACCELERATION 3000 // X, Y, Z acceleration for travel (non printing) moves

/**
* Default Jerk (mm/s)
* Override with M205 X Y Z E
*
* "Jerk" specifies the minimum speed change that requires acceleration.
* When changing speed and direction, if the difference is less than the
* value set here, it may happen instantaneously.
*/
#define DEFAULT_XJERK 10.0
#define DEFAULT_YJERK 10.0
#define DEFAULT_ZJERK 0.3
#define DEFAULT_EJERK 5.0

/**
* S-Curve Acceleration

```

```

*
* This option eliminates vibration during printing by fitting a Bézier
* curve to move acceleration, producing much smoother direction changes.
*
* See https://github.com/synthetos/TinyG/wiki/Jerk-Controlled-Motion-Explained
*/
#define S_CURVE_ACCELERATION

//=====
//----- Z Probe Options -----
//=====
// @section probes

//
// See http://marlinfw.org/docs/configuration/probes.html
//

/**
 * Z_MIN_PROBE_USES_Z_MIN_ENDSTOP_PIN
 *
 * Enable this option for a probe connected to the Z Min endstop pin.
 */
#define Z_MIN_PROBE_USES_Z_MIN_ENDSTOP_PIN

/**
 * Z_MIN_PROBE_ENDSTOP
 *
 * Enable this option for a probe connected to any pin except Z-Min.
 * (By default Marlin assumes the Z-Max endstop pin.)
 * To use a custom Z Probe pin, set Z_MIN_PROBE_PIN below.
 *
 * - The simplest option is to use a free endstop connector.
 * - Use 5V for powered (usually inductive) sensors.
 *
 * - RAMPS 1.3/1.4 boards may use the 5V, GND, and Aux4->D32 pin:
 * - For simple switches connect...
 * - normally-closed switches to GND and D32.
 * - normally-open switches to 5V and D32.
 *
 * WARNING: Setting the wrong pin may have unexpected and potentially
 * disastrous consequences. Use with caution and do your homework.
 */
#define Z_MIN_PROBE_ENDSTOP

/**
 * Probe Type
 *
 * Allen Key Probes, Servo Probes, Z-Sled Probes, FIX_MOUNTED_PROBE, etc.
 * Activate one of these to use Auto Bed Leveling below.
 */

/**
 * The "Manual Probe" provides a means to do "Auto" Bed Leveling without a probe.
 * Use G29 repeatedly, adjusting the Z height at each point with movement commands
 * or (with LCD_BED_LEVELING) the LCD controller.
 */
#define PROBE_MANUALLY
#define MANUAL_PROBE_START_Z 0.2

/**
 * A Fix-Mounted Probe either doesn't deploy or needs manual deployment.

```

```

* (e.g., an inductive probe or a nozzle-based probe-switch.)
*/
#define FIX_MOUNTED_PROBE

/**
 * Z Servo Probe, such as an endstop switch on a rotating arm.
 */
#define Z_PROBE_SERVO_NR 0 // Defaults to SERVO 0 connector.
#define Z_SERVO_ANGLES {60,150} // Z Servo Deploy and Stow angles

/**
 * The BLTouch probe uses a Hall effect sensor and emulates a servo.
 */
#define BLTOUCH

/**
 * Enable one or more of the following if probing seems unreliable.
 * Heaters and/or fans can be disabled during probing to minimize electrical
 * noise. A delay can also be added to allow noise and vibration to settle.
 * These options are most useful for the BLTouch probe, but may also improve
 * readings with inductive probes and piezo sensors.
 */
#define PROBING_HEATERS_OFF // Turn heaters off when probing
#if ENABLED(PROBING_HEATERS_OFF)
  #define WAIT_FOR_BED_HEATER // Wait for bed to heat back up between probes (to improve accuracy)
#endif
#define PROBING_FANS_OFF // Turn fans off when probing
#define DELAY_BEFORE_PROBING 200 // (ms) To prevent vibrations from triggering piezo sensors

// A probe that is deployed and stowed with a solenoid pin (SOL1_PIN)
#define SOLENOID_PROBE

// A sled-mounted probe like those designed by Charles Bell.
#define Z_PROBE_SLED
#define SLED_DOCKING_OFFSET 5 // The extra distance the X axis must travel to pickup the sled. 0 should be
fine but you can push it further if you'd like.

//
// For Z_PROBE_ALLEN_KEY see the Delta example configurations.
//

/**
 * Z Probe to nozzle (X,Y) offset, relative to (0, 0).
 * X and Y offsets must be integers.
 *
 * In the following example the X and Y offsets are both positive:
 * #define X_PROBE_OFFSET_FROM_EXTRUDER 10
 * #define Y_PROBE_OFFSET_FROM_EXTRUDER 10
 *
 * +- BACK +-
 * |         |
 * L| (+) P | R <- probe (20,20)
 * E|         |
 * F| (-) N (+) | G <- nozzle (10,10)
 * T|         |
 * | (-) | T
 * |         |
 * O-- FRONT --+
 * (0,0)
 */
#define X_PROBE_OFFSET_FROM_EXTRUDER 10 // X offset: -left +right [of the nozzle]
#define Y_PROBE_OFFSET_FROM_EXTRUDER 10 // Y offset: -front +behind [the nozzle]

```

```

#define Z_PROBE_OFFSET_FROM_EXTRUDER -7.6 // Z offset: -below +above [the nozzle]

// Certain types of probes need to stay away from edges
#define MIN_PROBE_EDGE 10

// X and Y axis travel speed (mm/m) between probes
#define XY_PROBE_SPEED 4000

// Feedrate (mm/m) for the first approach when double-probing (MULTIPLE_PROBING == 2)
#define Z_PROBE_SPEED_FAST HOMING_FEEDRATE_Z

// Feedrate (mm/m) for the "accurate" probe of each point
#define Z_PROBE_SPEED_SLOW (Z_PROBE_SPEED_FAST / 2)

// The number of probes to perform at each point.
// Set to 2 for a fast/slow probe, using the second probe result.
// Set to 3 or more for slow probes, averaging the results.
// #define MULTIPLE_PROBING 2

/**
 * Z probes require clearance when deploying, stowing, and moving between
 * probe points to avoid hitting the bed and other hardware.
 * Servo-mounted probes require extra space for the arm to rotate.
 * Inductive probes need space to keep from triggering early.
 *
 * Use these settings to specify the distance (mm) to raise the probe (or
 * lower the bed). The values set here apply over and above any (negative)
 * probe Z Offset set with Z_PROBE_OFFSET_FROM_EXTRUDER, M851, or the LCD.
 * Only integer values >= 1 are valid here.
 *
 * Example: 'M851 Z-5' with a CLEARANCE of 4 => 9mm from bed to nozzle.
 * But: 'M851 Z+1' with a CLEARANCE of 2 => 2mm from bed to nozzle.
 */
#define Z_CLEARANCE_DEPLOY_PROBE 10 // Z Clearance for Deploy/Stow
#define Z_CLEARANCE_BETWEEN_PROBES 5 // Z Clearance between probe points
#define Z_CLEARANCE_MULTI_PROBE 5 // Z Clearance between multiple probes
// #define Z_AFTER_PROBING 5 // Z position after probing is done

#define Z_PROBE_LOW_POINT -2 // Farthest distance below the trigger-point to go before stopping

// For M851 give a range for adjusting the Z probe offset
#define Z_PROBE_OFFSET_RANGE_MIN -20
#define Z_PROBE_OFFSET_RANGE_MAX 20

// Enable the M48 repeatability test to test probe accuracy
// #define Z_MIN_PROBE_REPEATABILITY_TEST

// For Inverting Stepper Enable Pins (Active Low) use 0, Non Inverting (Active High) use 1
// { 0:'Low', 1:'High' }
#define X_ENABLE_ON 0
#define Y_ENABLE_ON 0
#define Z_ENABLE_ON 0
#define E_ENABLE_ON 0 // For all extruders

// Disables axis stepper immediately when it's not being used.
// WARNING: When motors turn off there is a chance of losing position accuracy!
#define DISABLE_X false
#define DISABLE_Y false
#define DISABLE_Z false
// Warn on display about possibly reduced accuracy
// #define DISABLE_REDUCE_ACCURACY_WARNING

```

```

// @section extruder

#define DISABLE_E false // For all extruders
#define DISABLE_INACTIVE_EXTRUDER true // Keep only the active extruder enabled.

// @section machine

// Invert the stepper direction. Change (or reverse the motor connector) if an axis goes the wrong way.
#define INVERT_X_DIR false
#define INVERT_Y_DIR false
#define INVERT_Z_DIR false

// @section extruder

// For direct drive extruder v9 set to true, for geared extruder set to false.
#define INVERT_E0_DIR false
#define INVERT_E1_DIR false
#define INVERT_E2_DIR false
#define INVERT_E3_DIR false
#define INVERT_E4_DIR false

// @section homing

#define NO_MOTION_BEFORE_HOMING // Inhibit movement until all axes have been homed

#define UNKNOWN_Z_NO_RAISE // Don't raise Z (lower the bed) if Z is "unknown." For beds that fall when Z is
powered off.

#define Z_HOMING_HEIGHT 4 // (in mm) Minimal z height before homing (G28) for Z clearance above the bed,
clamps, ...
// Be sure you have this distance over your Z_MAX_POS in case.

// Direction of endstops when homing; 1=MAX, -1=MIN
// [-1,1]
#define X_HOME_DIR -1
#define Y_HOME_DIR -1
#define Z_HOME_DIR -1

// @section machine

// The size of the print bed
#define X_BED_SIZE 200
#define Y_BED_SIZE 200

// Travel limits (mm) after homing, corresponding to endstop positions.
#define X_MIN_POS 0
#define Y_MIN_POS 0
#define Z_MIN_POS 0
#define X_MAX_POS X_BED_SIZE
#define Y_MAX_POS Y_BED_SIZE
#define Z_MAX_POS 200

/**
 * Software Endstops
 *
 * - Prevent moves outside the set machine bounds.
 * - Individual axes can be disabled, if desired.
 * - X and Y only apply to Cartesian robots.
 * - Use 'M211' to set software endstops on/off or report current state
 */

// Min software endstops constrain movement within minimum coordinate bounds

```

```

#define MIN_SOFTWARE_ENDSTOPS
#if ENABLED(MIN_SOFTWARE_ENDSTOPS)
  #define MIN_SOFTWARE_ENDSTOP_X
  #define MIN_SOFTWARE_ENDSTOP_Y
  #define MIN_SOFTWARE_ENDSTOP_Z
#endif

// Max software endstops constrain movement within maximum coordinate bounds
#define MAX_SOFTWARE_ENDSTOPS
#if ENABLED(MAX_SOFTWARE_ENDSTOPS)
  #define MAX_SOFTWARE_ENDSTOP_X
  #define MAX_SOFTWARE_ENDSTOP_Y
  #define MAX_SOFTWARE_ENDSTOP_Z
#endif

#if ENABLED(MIN_SOFTWARE_ENDSTOPS) || ENABLED(MAX_SOFTWARE_ENDSTOPS)
  // #define SOFT_ENDSTOPS_MENU_ITEM // Enable/Disable software endstops from the LCD
#endif

/**
 * Filament Runout Sensors
 * Mechanical or opto endstops are used to check for the presence of filament.
 *
 * RAMPS-based boards use SERVO3_PIN for the first runout sensor.
 * For other boards you may need to define FIL_RUNOUT_PIN, FIL_RUNOUT2_PIN, etc.
 * By default the firmware assumes HIGH=FILAMENT PRESENT.
 */
// #define FILAMENT_RUNOUT_SENSOR
#if ENABLED(FILAMENT_RUNOUT_SENSOR)
  #define NUM_RUNOUT_SENSORS 1 // Number of sensors, up to one per extruder. Define a
  FIL_RUNOUT#_PIN for each.
  #define FIL_RUNOUT_INVERTING false // set to true to invert the logic of the sensor.
  #define FIL_RUNOUT_PULLUP // Use internal pullup for filament runout pins.
  #define FILAMENT_RUNOUT_SCRIPT "M600"
#endif

//=====
//===== Bed Leveling =====
//=====
// @section calibrate

/**
 * Choose one of the options below to enable G29 Bed Leveling. The parameters
 * and behavior of G29 will change depending on your selection.
 *
 * If using a Probe for Z Homing, enable Z_SAFE_HOMING also!
 *
 * - AUTO_BED_LEVELING_3POINT
 * Probe 3 arbitrary points on the bed (that aren't collinear)
 * You specify the XY coordinates of all 3 points.
 * The result is a single tilted plane. Best for a flat bed.
 *
 * - AUTO_BED_LEVELING_LINEAR
 * Probe several points in a grid.
 * You specify the rectangle and the density of sample points.
 * The result is a single tilted plane. Best for a flat bed.
 *
 * - AUTO_BED_LEVELING_BILINEAR
 * Probe several points in a grid.
 * You specify the rectangle and the density of sample points.
 * The result is a mesh, best for large or uneven beds.
 */

```

```

* - AUTO_BED_LEVELING_UBL (Unified Bed Leveling)
* A comprehensive bed leveling system combining the features and benefits
* of other systems. UBL also includes integrated Mesh Generation, Mesh
* Validation and Mesh Editing systems.
*
* - MESH_BED_LEVELING
* Probe a grid manually
* The result is a mesh, suitable for large or uneven beds. (See BILINEAR.)
* For machines without a probe, Mesh Bed Leveling provides a method to perform
* leveling in steps so you can manually adjust the Z height at each grid-point.
* With an LCD controller the process is guided step-by-step.
*/
#define AUTO_BED_LEVELING_3POINT
#define AUTO_BED_LEVELING_LINEAR
#define AUTO_BED_LEVELING_BILINEAR
#define AUTO_BED_LEVELING_UBL
#define MESH_BED_LEVELING

/**
 * Normally G28 leaves leveling disabled on completion. Enable
 * this option to have G28 restore the prior leveling state.
 */
#define RESTORE_LEVELING_AFTER_G28

/**
 * Enable detailed logging of G28, G29, M48, etc.
 * Turn on with the command 'M111 S32'.
 * NOTE: Requires a lot of PROGMEM!
 */
#define DEBUG_LEVELING_FEATURE

#if ENABLED(MESH_BED_LEVELING) || ENABLED(AUTO_BED_LEVELING_BILINEAR) ||
    ENABLED(AUTO_BED_LEVELING_UBL)
  // Gradually reduce leveling correction until a set height is reached,
  // at which point movement will be level to the machine's XY plane.
  // The height can be set with M420 Z<height>
  #define ENABLE_LEVELING_FADE_HEIGHT

  // For Cartesian machines, instead of dividing moves on mesh boundaries,
  // split up moves into short segments like a Delta. This follows the
  // contours of the bed more closely than edge-to-edge straight moves.
  #define SEGMENT_LEVELLED_MOVES
  #define LEVELLED_SEGMENT_LENGTH 5.0 // (mm) Length of all segments (except the last one)

  /**
   * Enable the G26 Mesh Validation Pattern tool.
   */
  #define G26_MESH_VALIDATION
  #if ENABLED(G26_MESH_VALIDATION)
    #define MESH_TEST_NOZZLE_SIZE 0.4 // (mm) Diameter of primary nozzle.
    #define MESH_TEST_LAYER_HEIGHT 0.2 // (mm) Default layer height for the G26 Mesh Validation Tool.
    #define MESH_TEST_HOTEND_TEMP 205.0 // (°C) Default nozzle temperature for the G26 Mesh Validation
    Tool.
    #define MESH_TEST_BED_TEMP 60.0 // (°C) Default bed temperature for the G26 Mesh Validation Tool.
  #endif

#endif

// Set the number of grid points per dimension.
#define GRID_MAX_POINTS_X 3

```

```

#define GRID_MAX_POINTS_Y GRID_MAX_POINTS_X

// Set the boundaries for probing (where the probe can reach).
//#define LEFT_PROBE_BED_POSITION MIN_PROBE_EDGE
//#define RIGHT_PROBE_BED_POSITION (X_BED_SIZE - MIN_PROBE_EDGE)
//#define FRONT_PROBE_BED_POSITION MIN_PROBE_EDGE
//#define BACK_PROBE_BED_POSITION (Y_BED_SIZE - MIN_PROBE_EDGE)

#define LEFT_PROBE_BED_POSITION 60
#define RIGHT_PROBE_BED_POSITION 185
#define FRONT_PROBE_BED_POSITION 60
#define BACK_PROBE_BED_POSITION 185
// Probe along the Y axis, advancing X after each column
//#define PROBE_Y_FIRST

#if ENABLED(AUTO_BED_LEVELING_BILINEAR)

  // Beyond the probed grid, continue the implied tilt?
  // Default is to maintain the height of the nearest edge.
  //#define EXTRAPOLATE_BEYOND_GRID

  //
  // Experimental Subdivision of the grid by Catmull-Rom method.
  // Synthesizes intermediate points to produce a more detailed mesh.
  //
  //#define ABL_BILINEAR_SUBDIVISION
  #if ENABLED(ABL_BILINEAR_SUBDIVISION)
    // Number of subdivisions between probe points
    #define BILINEAR_SUBDIVISIONS 3
  #endif

#endif

#elif ENABLED(AUTO_BED_LEVELING_UBL)

  //=====
  //----- Unified Bed Leveling -----
  //=====

  //#define MESH_EDIT_GFX_OVERLAY // Display a graphics overlay while editing the mesh

  #define MESH_INSET 1 // Set Mesh bounds as an inset region of the bed
  #define GRID_MAX_POINTS_X 10 // Don't use more than 15 points per axis, implementation limited.
  #define GRID_MAX_POINTS_Y GRID_MAX_POINTS_X

  #define UBL_MESH_EDIT_MOVES_Z // Sophisticated users prefer no movement of nozzle
  #define UBL_SAVE_ACTIVE_ON_M500 // Save the currently active mesh in the current slot on M500

  //#define UBL_Z_RAISE_WHEN_OFF_MESH 2.5 // When the nozzle is off the mesh, this value is used
  // as the Z-Height correction value.

#elif ENABLED(MESH_BED_LEVELING)

  //=====
  //----- Mesh -----
  //=====

  #define MESH_INSET 10 // Set Mesh bounds as an inset region of the bed
  #define GRID_MAX_POINTS_X 3 // Don't use more than 7 points per axis, implementation limited.
  #define GRID_MAX_POINTS_Y GRID_MAX_POINTS_X

  //#define MESH_G28_REST_ORIGIN // After homing all axes ('G28' or 'G28 XYZ') rest Z at Z_MIN_POS

```

```

#endif // BED_LEVELING

/**
 * Points to probe for all 3-point Leveling procedures.
 * Override if the automatically selected points are inadequate.
 */
#if ENABLED(AUTO_BED_LEVELING_3POINT) || ENABLED(AUTO_BED_LEVELING_UBL)
  #define PROBE_PT_1_X 15
  #define PROBE_PT_1_Y 180
  #define PROBE_PT_2_X 15
  #define PROBE_PT_2_Y 20
  #define PROBE_PT_3_X 170
  #define PROBE_PT_3_Y 20
#endif

/**
 * Add a bed leveling sub-menu for ABL or MBL.
 * Include a guided procedure if manual probing is enabled.
 */
#define LCD_BED_LEVELING

#if ENABLED(LCD_BED_LEVELING)
  #define MBL_Z_STEP 0.025 // Step size while manually probing Z axis.
  #define LCD_PROBE_Z_RANGE 4 // Z Range centered on Z_MIN_POS for LCD Z adjustment
#endif

// Add a menu item to move between bed corners for manual bed adjustment
#define LEVEL_BED_CORNERS

#if ENABLED(LEVEL_BED_CORNERS)
  #define LEVEL_CORNERS_INSET 30 // (mm) An inset for corner leveling
  #define LEVEL_CORNERS_Z_HOP 4.0 // (mm) Move nozzle up before moving between corners
  #define LEVEL_CENTER_TOO // Move to the center after the last corner
#endif

/**
 * Commands to execute at the end of G29 probing.
 * Useful to retract or move the Z probe out of the way.
 */
#define Z_PROBE_END_SCRIPT "G1 Z10 F12000\nG1 X15 Y330\nG1 Z0.5\nG1 Z10"

// @section homing

// The center of the bed is at (X=0, Y=0)
#define BED_CENTER_AT_0_0

// Manually set the home position. Leave these undefined for automatic settings.
// For DELTA this is the top-center of the Cartesian print volume.
#define MANUAL_X_HOME_POS 0
#define MANUAL_Y_HOME_POS 0
#define MANUAL_Z_HOME_POS 0

// Use "Z Safe Homing" to avoid homing with a Z probe outside the bed area.
//
// With this feature enabled:
//
// - Allow Z homing only after X and Y homing AND stepper drivers still enabled.
// - If stepper drivers time out, it will need X and Y homing again before Z homing.
// - Move the Z probe (or nozzle) to a defined XY point before Z Homing when homing all axes (G28).
// - Prevent Z homing when the Z probe is outside bed area.

```

```

//
#define Z_SAFE_HOMING

#if ENABLED(Z_SAFE_HOMING)
  #define Z_SAFE_HOMING_X_POINT ((X_BED_SIZE) / 2) // X point for Z homing when homing all axes (G28).
  #define Z_SAFE_HOMING_Y_POINT ((Y_BED_SIZE) / 2) // Y point for Z homing when homing all axes (G28).
#endif

// Homing speeds (mm/m)
#define HOMING_FEEDRATE_XY (50*60)
#define HOMING_FEEDRATE_Z (3*60)

// @section calibrate

/**
 * Bed Skew Compensation
 *
 * This feature corrects for misalignment in the XYZ axes.
 *
 * Take the following steps to get the bed skew in the XY plane:
 * 1. Print a test square (e.g., https://www.thingiverse.com/thing:2563185)
 * 2. For XY_DIAG_AC measure the diagonal A to C
 * 3. For XY_DIAG_BD measure the diagonal B to D
 * 4. For XY_SIDE_AD measure the edge A to D
 *
 * Marlin automatically computes skew factors from these measurements.
 * Skew factors may also be computed and set manually:
 *
 * - Compute AB : SQRT(2*AC*AC+2*BD*BD-4*AD*AD)/2
 * - XY_SKEW_FACTOR : TAN(PI/2-ACOS((AC*AC-AB*AB-AD*AD)/(2*AB*AD)))
 *
 * If desired, follow the same procedure for XZ and YZ.
 * Use these diagrams for reference:
 *
 * Y      Z      Z
 * ^      ^      ^
 * B-----C   B-----C   B-----C
 * | // //    | // //    | // //
 * | // //    | // //    | // //
 * | A-----D | A-----D | A-----D
 * +-----+X +-----+X +-----+Y
 * XY_SKEW_FACTOR  XZ_SKEW_FACTOR  YZ_SKEW_FACTOR
 */
#define SKEW_CORRECTION

#if ENABLED(SKEW_CORRECTION)
  // Input all length measurements here:
  #define XY_DIAG_AC 282.8427124746
  #define XY_DIAG_BD 282.8427124746
  #define XY_SIDE_AD 200

  // Or, set the default skew factors directly here
  // to override the above measurements:
  #define XY_SKEW_FACTOR 0.0

  //#define SKEW_CORRECTION_FOR_Z
  #if ENABLED(SKEW_CORRECTION_FOR_Z)
    #define XZ_DIAG_AC 282.8427124746
    #define XZ_DIAG_BD 282.8427124746
    #define YZ_DIAG_AC 282.8427124746
    #define YZ_DIAG_BD 282.8427124746
    #define YZ_SIDE_AD 200
    #define XZ_SKEW_FACTOR 0.0
  #endif

```

```

#define YZ_SKEW_FACTOR 0.0
#endif

// Enable this option for M852 to set skew at runtime
//#define SKEW_CORRECTION_GCODE
#endif

//=====
//----- Additional Features -----
//=====

// @section extras

//
// EEPROM
//
// The microcontroller can store settings in the EEPROM, e.g. max velocity...
// M500 - stores parameters in EEPROM
// M501 - reads parameters from EEPROM (if you need reset them after you changed them temporarily).
// M502 - reverts to the default "factory settings". You still need to store them in EEPROM afterwards if you want to.
//
#define EEPROM_SETTINGS // Enable for M500 and M501 commands
#define DISABLE_M503 // Saves ~2700 bytes of PROGMEM. Disable for release!
#define EEPROM_CHITCHAT // Give feedback on EEPROM commands. Disable to save PROGMEM.

//
// Host Keepalive
//
// When enabled Marlin will send a busy status message to the host
// every couple of seconds when it can't accept commands.
//
#define HOST_KEEPALIVE_FEATURE // Disable this if your host doesn't like keepalive messages
#define DEFAULT_KEEPALIVE_INTERVAL 2 // Number of seconds between "busy" messages. Set with M113.
#define BUSY_WHILE_HEATING // Some hosts require "busy" messages even during heating

//
// M100 Free Memory Watcher
//
//#define M100_FREE_MEMORY_WATCHER // Add M100 (Free Memory Watcher) to debug memory usage

//
// G20/G21 Inch mode support
//
//#define INCH_MODE_SUPPORT

//
// M149 Set temperature units support
//
#define TEMPERATURE_UNITS_SUPPORT

// @section temperature

// Preheat Constants
#define PREHEAT_1_TEMP_HOTEND 180
#define PREHEAT_1_TEMP_BED 70
#define PREHEAT_1_FAN_SPEED 0 // Value from 0 to 255

#define PREHEAT_2_TEMP_HOTEND 240
#define PREHEAT_2_TEMP_BED 110
#define PREHEAT_2_FAN_SPEED 0 // Value from 0 to 255

//**

```



```

// Default number of triangles
#define NOZZLE_CLEAN_TRIANGLES 3

// Specify positions as { X, Y, Z }
#define NOZZLE_CLEAN_START_POINT { 30, 30, (Z_MIN_POS + 1)}
#define NOZZLE_CLEAN_END_POINT {100, 60, (Z_MIN_POS + 1)}

// Circular pattern radius
#define NOZZLE_CLEAN_CIRCLE_RADIUS 6.5
// Circular pattern circle fragments number
#define NOZZLE_CLEAN_CIRCLE_FN 10
// Middle point of circle
#define NOZZLE_CLEAN_CIRCLE_MIDDLE NOZZLE_CLEAN_START_POINT

// Moves the nozzle to the initial position
#define NOZZLE_CLEAN_GOBACK
#endif

/**
 * Print Job Timer
 *
 * Automatically start and stop the print job timer on M104/M109/M190.
 *
 * M104 (hotend, no wait) - high temp = none, low temp = stop timer
 * M109 (hotend, wait) - high temp = start timer, low temp = stop timer
 * M190 (bed, wait) - high temp = start timer, low temp = none
 *
 * The timer can also be controlled with the following commands:
 *
 * M75 - Start the print job timer
 * M76 - Pause the print job timer
 * M77 - Stop the print job timer
 */
#define PRINTJOB_TIMER_AUTOSTART

/**
 * Print Counter
 *
 * Track statistical data such as:
 *
 * - Total print jobs
 * - Total successful print jobs
 * - Total failed print jobs
 * - Total time printing
 *
 * View the current statistics with M78.
 */
// #define PRINTCOUNTER

//=====
//===== LCD and SD support =====
//=====

// @section lcd

/**
 * LCD LANGUAGE
 *
 * Select the language to display on the LCD. These languages are available:
 *
 * en, an, bg, ca, cn, cz, cz_utf8, de, el, el-gr, es, es_utf8, eu,

```

```

* fi, fr, fr_utf8, gl, hr, it, kana, kana_utf8, ko_KR, nl, pl, pt,
* pt_utf8, pt-br, pt-br_utf8, ru, sk_utf8, tr, uk, zh_CN, zh_TW, test
*
* :{ 'en':English, 'an':Aragonese, 'bg':Bulgarian, 'ca':Catalan, 'cn':Chinese, 'cz':Czech, 'cz_utf8':Czech (UTF8),
'de':German, 'el':Greek, 'el-gr':Greek (Greece), 'es':Spanish, 'es_utf8':Spanish (UTF8), 'eu':Basque-Euskera,
'fi':Finnish, 'fr':French, 'fr_utf8':French (UTF8), 'gl':Galician, 'hr':Croatian, 'it':Italian, 'kana':Japanese,
'kana_utf8':Japanese (UTF8), 'ko_KR':Korean, 'nl':Dutch, 'pl':Polish, 'pt':Portuguese, 'pt-br':Portuguese
(Brazilian), 'pt-br_utf8':Portuguese (Brazilian UTF8), 'pt_utf8':Portuguese (UTF8), 'ru':Russian, 'sk_utf8':Slovak
(UTF8), 'tr':Turkish, 'uk':Ukrainian, 'zh_CN':Chinese (Simplified), 'zh_TW':Chinese (Taiwan), 'test':TEST }
*/
#define LCD_LANGUAGE pt-br

/**
* LCD Character Set
*
* Note: This option is NOT applicable to Graphical Displays.
*
* All character-based LCDs provide ASCII plus one of these
* language extensions:
*
* - JAPANESE ... the most common
* - WESTERN ... with more accented characters
* - CYRILLIC ... for the Russian language
*
* To determine the language extension installed on your controller:
*
* - Compile and upload with LCD_LANGUAGE set to 'test'
* - Click the controller to view the LCD menu
* - The LCD will display Japanese, Western, or Cyrillic text
*
* See http://marlinfw.org/docs/development/lcd\_language.html
*
* :{JAPANESE, 'WESTERN', 'CYRILLIC'}
*/
#define DISPLAY_CHARSET_HD44780 JAPANESE

/**
* SD CARD
*
* SD Card support is disabled by default. If your controller has an SD slot,
* you must uncomment the following option or it won't work.
*
*/
#define SDSUPPORT

/**
* SD CARD: SPI SPEED
*
* Enable one of the following items for a slower SPI transfer speed.
* This may be required to resolve "volume init" errors.
*/
#define SPI_SPEED SPI_HALF_SPEED
//#define SPI_SPEED SPI_QUARTER_SPEED
//#define SPI_SPEED SPI_EIGHTH_SPEED

/**
* SD CARD: ENABLE CRC
*
* Use CRC checks and retries on the SD communication.
*/
//#define SD_CHECK_AND_RETRY

```

```

/**
 * LCD Menu Items
 *
 * Disable all menus and only display the Status Screen, or
 * just remove some extraneous menu items to recover space.
 */
#define NO_LCD_MENUS
#define SLIM_LCD_MENUS

//
// ENCODER SETTINGS
//
// This option overrides the default number of encoder pulses needed to
// produce one step. Should be increased for high-resolution encoders.
//
#define ENCODER_PULSES_PER_STEP 4

//
// Use this option to override the number of step signals required to
// move between next/prev menu items.
//
#define ENCODER_STEPS_PER_MENU_ITEM 1

/**
 * Encoder Direction Options
 *
 * Test your encoder's behavior first with both options disabled.
 *
 * Reversed Value Edit and Menu Nav? Enable REVERSE_ENCODER_DIRECTION.
 * Reversed Menu Navigation only? Enable REVERSE_MENU_DIRECTION.
 * Reversed Value Editing only? Enable BOTH options.
 */

//
// This option reverses the encoder direction everywhere.
//
// Set this option if CLOCKWISE causes values to DECREASE
//
#define REVERSE_ENCODER_DIRECTION

//
// This option reverses the encoder direction for navigating LCD menus.
//
// If CLOCKWISE normally moves DOWN this makes it go UP.
// If CLOCKWISE normally moves UP this makes it go DOWN.
//
#define REVERSE_MENU_DIRECTION

//
// Individual Axis Homing
//
// Add individual axis homing items (Home X, Home Y, and Home Z) to the LCD menu.
//
#define INDIVIDUAL_AXIS_HOMING_MENU

//
// SPEAKER/BUZZER
//
// If you have a speaker that can produce tones, enable it here.
// By default Marlin assumes you have a buzzer with a fixed frequency.
//
#define SPEAKER

```

```

//
// The duration and frequency for the UI feedback sound.
// Set these to 0 to disable audio feedback in the LCD menus.
//
// Note: Test audio output with the G-Code:
// M300 S<frequency Hz> P<duration ms>
//
// #define LCD_FEEDBACK_FREQUENCY_DURATION_MS 2
// #define LCD_FEEDBACK_FREQUENCY_HZ 5000

//=====
//===== LCD / Controller Selection =====
//===== (Character-based LCDs) =====
//=====

//
// RepRapDiscount Smart Controller.
// http://reprap.org/wiki/RepRapDiscount_Smart_Controller
//
// Note: Usually sold with a white PCB.
//
// #define REPRAP_DISCOUNT_SMART_CONTROLLER

//
// ULTIMAKER Controller.
//
// #define ULTIMAKERCONTROLLER

//
// ULTIPANEL as seen on Thingiverse.
//
// #define ULTIPANEL

//
// PanelOne from T3P3 (via RAMPS 1.4 AUX2/AUX3)
// http://reprap.org/wiki/PanelOne
//
// #define PANEL_ONE

//
// GADGETS3D G3D LCD/SD Controller
// http://reprap.org/wiki/RAMPS_1.3/1.4_GADGETS3D_Shield_with_Panel
//
// Note: Usually sold with a blue PCB.
//
// #define G3D_PANEL

//
// RigidBot Panel V1.0
// http://www.inventapart.com/
//
// #define RIGIDBOT_PANEL

//
// Makeboard 3D Printer Parts 3D Printer Mini Display 1602 Mini Controller
// https://www.aliexpress.com/item/Micromake-Makeboard-3D-Printer-Parts-3D-Printer-Mini-Display-1602-Mini-Controller-Compatible-with-Ramps-1/32765887917.html
//
// #define MAKEBOARD_MINI_2_LINE_DISPLAY_1602

//

```

```

// ANET and Tronxy 20x4 Controller
//
// #define ZONESTAR_LCD // Requires ADC_KEYPAD_PIN to be assigned to an analog pin.
// This LCD is known to be susceptible to electrical interference
// which scrambles the display. Pressing any button clears it up.
// This is a LCD2004 display with 5 analog buttons.

//
// Generic 16x2, 16x4, 20x2, or 20x4 character-based LCD.
//
// #define ULTRA_LCD

//=====
//===== LCD / Controller Selection =====
//===== (I2C and Shift-Register LCDs) =====
//=====

//
// CONTROLLER TYPE: I2C
//
// Note: These controllers require the installation of Arduino's LiquidCrystal_I2C
// library. For more info: https://github.com/kiyoshigawa/LiquidCrystal_I2C
//

//
// Elefu RA Board Control Panel
// http://www.elefu.com/index.php?route=product/product&product_id=53
//
// #define RA_CONTROL_PANEL

//
// Sainsmart (YwRobot) LCD Displays
//
// These require F.Malpartida's LiquidCrystal_I2C library
// https://bitbucket.org/fmalpartida/new-liquidcrystal/wiki/Home
//
// #define LCD_SAINSMART_I2C_1602
// #define LCD_SAINSMART_I2C_2004

//
// Generic LCM1602 LCD adapter
//
// #define LCM1602

//
// PANELOLU2 LCD with status LEDs,
// separate encoder and click inputs.
//
// Note: This controller requires Arduino's LiquidTWI2 library v1.2.3 or later.
// For more info: https://github.com/lincomatic/LiquidTWI2
//
// Note: The PANELOLU2 encoder click input can either be directly connected to
// a pin (if BTN_ENC defined to != -1) or read through I2C (when BTN_ENC == -1).
//
// #define LCD_I2C_PANELOLU2

//
// Panucatt VIKI LCD with status LEDs,
// integrated click & L/R/U/D buttons, separate encoder inputs.
//
// #define LCD_I2C_VIKI

```

```

//
// CONTROLLER TYPE: Shift register panels
//

//
// 2 wire Non-latching LCD SR from https://goo.gl/aJJ4sH
// LCD configuration: http://reprap.org/wiki/SAV_3D_LCD
//
//#define SAV_3DLCD

//=====
//===== LCD / Controller Selection =====
//===== (Graphical LCDs) =====
//=====

//
// CONTROLLER TYPE: Graphical 128x64 (DOGM)
//
// IMPORTANT: The U8glib library is required for Graphical Display!
//           https://github.com/olikraus/U8glib_Arduino
//

//
// RepRapDiscount FULL GRAPHIC Smart Controller
// http://reprap.org/wiki/RepRapDiscount_Full_Graphic_Smart_Controller
//
#define REPRAP_DISCOUNT_FULL_GRAPHIC_SMART_CONTROLLER

//
// ReprapWorld Graphical LCD
// https://reprapworld.com/?products_details&products_id/1218
//
//#define REPRAPWORLD_GRAPHICAL_LCD

//
// Activate one of these if you have a Panucatt Devices
// Viki 2.0 or mini Viki with Graphic LCD
// http://panucatt.com
//
//#define VIKI2
//#define miniVIKI

//
// MakerLab Mini Panel with graphic
// controller and SD support - http://reprap.org/wiki/Mini_panel
//
//#define MINIPANEL

//
// MaKr3d Makr-Panel with graphic controller and SD support.
// http://reprap.org/wiki/MaKr3d_MaKrPanel
//
//#define MAKRPANEL

//
// Adafruit ST7565 Full Graphic Controller.
// https://github.com/eboston/Adafruit-ST7565-Full-Graphic-Controller/
//
//#define ELB_FULL_GRAPHIC_CONTROLLER

//
// BQ LCD Smart Controller shipped by

```

```

// default with the BQ Hephestos 2 and Witbox 2.
//
//#define BQ_LCD_SMART_CONTROLLER

//
// Cartesio UI
// http://mauk.cc/webshop/cartesio-shop/electronics/user-interface
//
//#define CARTESIO_UI

//
// LCD for Melzi Card with Graphical LCD
//
//#define LCD_FOR_MELZI

//
// SSD1306 OLED full graphics generic display
//
//#define U8GLIB_SSD1306

//
// SAV Oled LCD module support using either SSD1306 or SH1106 based LCD modules
//
//#define SAV_3DGLCD
#if ENABLED(SAV_3DGLCD)
  //#define U8GLIB_SSD1306
  #define U8GLIB_SH1106
#endif

//
// Original Ulticontroller from Ultimaker 2 printer with SSD1309 I2C display and encoder
// https://github.com/Ultimaker/Ultimaker2/tree/master/1249_Ulticontroller_Board_(x1)
//
//#define ULTI_CONTROLLER

//
// TinyBoy2 128x64 OLED / Encoder Panel
//
//#define OLED_PANEL_TINYBOY2

//
// MKS MINI12864 with graphic controller and SD support
// http://reprap.org/wiki/MKS_MINI_12864
//
//#define MKS_MINI_12864

//
// Factory display for Creality CR-10
// https://www.aliexpress.com/item/Universal-LCD-12864-3D-Printer-Display-Screen-With-Encoder-For-CR-10-CR-7-Model/32833148327.html
//
// This is RAMPS-compatible using a single 10-pin connector.
// (For CR-10 owners who want to replace the Melzi Creality board but retain the display)
//
//#define CR10_STOCKDISPLAY

//
// ANET and Tronxy Graphical Controller
//
//#define ANET_FULL_GRAPHICS_LCD // Anet 128x64 full graphics lcd with rotary encoder as used on Anet A6
// A clone of the RepRapDiscount full graphics display but with
// different pins/wiring (see pins_ANET_10.h).

```

```

//
// MKS OLED 1.3" 128 x 64 FULL GRAPHICS CONTROLLER
// http://reprap.org/wiki/MKS_12864OLED
//
// Tiny, but very sharp OLED display
//
// #define MKS_12864OLED // Uses the SH1106 controller (default)
// #define MKS_12864OLED_SSD1306 // Uses the SSD1306 controller

//
// Silvergate GLCD controller
// http://github.com/android444/Silvergate
//
// #define SILVER_GATE_GLCD_CONTROLLER

//=====
//----- Other Controllers -----
//=====

//
// CONTROLLER TYPE: Standalone / Serial
//

//
// LCD for Malyan M200 printers.
// This requires SDSUPPORT to be enabled
//
// #define MALYAN_LCD

//
// CONTROLLER TYPE: Keypad / Add-on
//

//
// RepRapWorld REPRAPWORLD_KEYPAD v1.1
// http://reprapworld.com/?products_details&products_id=202&cPath=1591_1626
//
// REPRAPWORLD_KEYPAD_MOVE_STEP sets how much should the robot move when a key
// is pressed, a value of 10.0 means 10mm per click.
//
// #define REPRAPWORLD_KEYPAD
// #define REPRAPWORLD_KEYPAD_MOVE_STEP 10.0

//=====
//----- Extra Features -----
//=====

// @section extras

// Increase the FAN PWM frequency. Removes the PWM noise but increases heating in the FET/Arduino
// #define FAST_PWM_FAN

// Use software PWM to drive the fan, as for the heaters. This uses a very low frequency
// which is not as annoying as with the hardware PWM. On the other hand, if this frequency
// is too low, you should also increment SOFT_PWM_SCALE.
// #define FAN_SOFT_PWM

// Incrementing this by 1 will double the software PWM frequency,
// affecting heaters, and the fan if FAN_SOFT_PWM is enabled.
// However, control resolution will be halved for each increment;
// at zero value, there are 128 effective control positions.

```

```

#define SOFT_PWM_SCALE 0
// If SOFT_PWM_SCALE is set to a value higher than 0, dithering can
// be used to mitigate the associated resolution loss. If enabled,
// some of the PWM cycles are stretched so on average the desired
// duty cycle is attained.
//#define SOFT_PWM_DITHER

// Temperature status LEDs that display the hotend and bed temperature.
// If all hotends, bed temperature, and target temperature are under 54C
// then the BLUE led is on. Otherwise the RED led is on. (1C hysteresis)
//#define TEMP_STAT_LEDS

// M240 Triggers a camera by emulating a Canon RC-1 Remote
// Data from: http://www.doc-diy.net/photo/rc-1_hacked/
//#define PHOTOGRAPH_PIN 23

// SkeinForge sends the wrong arc g-codes when using Arc Point as fillet procedure
//#define SF_ARC_FIX

// Support for the BariCUDA Paste Extruder
//#define BARICUDA

// Support for BlinkM/CyzRgb
//#define BLINKM

// Support for PCA9632 PWM LED driver
//#define PCA9632

/**
 * RGB LED / LED Strip Control
 *
 * Enable support for an RGB LED connected to 5V digital pins, or
 * an RGB Strip connected to MOSFETs controlled by digital pins.
 *
 * Adds the M150 command to set the LED (or LED strip) color.
 * If pins are PWM capable (e.g., 4, 5, 6, 11) then a range of
 * luminance values can be set from 0 to 255.
 * For Neopixel LED an overall brightness parameter is also available.
 *
 * *** CAUTION ***
 * LED Strips require a MOSFET Chip between PWM lines and LEDs,
 * as the Arduino cannot handle the current the LEDs will require.
 * Failure to follow this precaution can destroy your Arduino!
 * NOTE: A separate 5V power supply is required! The Neopixel LED needs
 * more current than the Arduino 5V linear regulator can produce.
 * *** CAUTION ***
 *
 * LED Type. Enable only one of the following two options.
 */
#define RGB_LED
#define RGBW_LED

#if ENABLED(RGB_LED) || ENABLED(RGBW_LED)
  #define RGB_LED_R_PIN 34
  #define RGB_LED_G_PIN 43
  #define RGB_LED_B_PIN 35
  #define RGB_LED_W_PIN -1
#endif

// Support for Adafruit Neopixel LED driver
#define NEOPIXEL_LED

```

```

#if ENABLED(NEOPIXEL_LED)
  #define NEOPIXEL_TYPE NEO_GRBW // NEO_GRBW / NEO_GRB - four/three channel driver type (defined in
Adafruit_NeoPixel.h)
  #define NEOPIXEL_PIN 4 // LED driving pin on motherboard 4 => D4 (EXP2-5 on Printboard) / 30 => PC7
(EXP3-13 on Rumba)
  #define NEOPIXEL_PIXELS 30 // Number of LEDs in the strip
  #define NEOPIXEL_JS_SEQUENTIAL // Sequential display for temperature change - LED by LED. Disable to
change all LEDs at once.
  #define NEOPIXEL_BRIGHTNESS 127 // Initial brightness (0-255)
  //#define NEOPIXEL_STARTUP_TEST // Cycle through colors at startup
#endif

/**
 * Printer Event LEDs
 *
 * During printing, the LEDs will reflect the printer status:
 *
 * - Gradually change from blue to violet as the heated bed gets to target temp
 * - Gradually change from violet to red as the hotend gets to temperature
 * - Change to white to illuminate work surface
 * - Change to green once print has finished
 * - Turn off after the print has finished and the user has pushed a button
 */
#if ENABLED(BLINKM) || ENABLED(RGB_LED) || ENABLED(RGBW_LED) || ENABLED(PCA9632) ||
ENABLED(NEOPIXEL_LED)
  #define PRINTER_EVENT_LEDS
#endif

/**
 * R/C SERVO support
 * Sponsored by TrinityLabs, Reworked by codexmas
 */

/**
 * Number of servos
 *
 * For some servo-related options NUM_SERVOS will be set automatically.
 * Set this manually if there are extra servos needing manual control.
 * Leave undefined or set to 0 to entirely disable the servo subsystem.
 */
#define NUM_SERVOS 1 // Servo index starts with 0 for M280 command

// Delay (in milliseconds) before the next move will start, to give the servo time to reach its target angle.
// 300ms is a good value but you can try less delay.
// If the servo can't reach the requested position, increase it.
#define SERVO_DELAY { 300 }

// Only power servos during movement, otherwise leave off to prevent jitter
//#define DEACTIVATE_SERVOS_AFTER_MOVE

#endif // CONFIGURATION_H

```