

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

Cleiton Márcio Costa

**Projeto e desenvolvimento de uma máquina identificadora e
separadora de moedas para fornecimento de troco.**

Divinópolis.

2015.

Cleiton Márcio Costa

Projeto e desenvolvimento de uma máquina identificadora e separadora de moedas para fornecimento de troco

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

Áreas de integração: Mecânica, Eletrônica e Computação.

Orientador: Prof. Dr. Luiz Cláudio Oliveira

Coorientador: Prof. Me. Alan Mendes Marotta

Divinópolis.

2015.

Cleiton Márcio Costa

Projeto e desenvolvimento de uma máquina identificadora e separadora de moedas para fornecimento de troco

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

Áreas de integração: Mecânica, Eletrônica e Computação.

Comissão Avaliadora:

Prof. Dr. Luiz Cláudio Oliveira

Centro Federal de Educação Tecnológica de Minas Gerais - CEFET MG

Prof. Dr. Ralney Nogueira de Faria

Centro Federal de Educação Tecnológica de Minas Gerais - CEFET MG

Prof. Adriano Nogueira Drumond

Centro Federal de Educação Tecnológica de Minas Gerais - CEFET MG

Divinópolis.

2015.

Dedico este trabalho aos meus pais,
Sebastião Leopoldo da Costa e Maria
Selma de Jesus Costa.

AGRADECIMENTOS

Agradeço,

A Deus, por ter me concebido a oportunidade e a capacidade de conseguir vencer mais essa etapa na minha vida. Apesar de todos os contratemplos, no final tudo deu certo graças a Ele.

Aos meus pais, Sebastião Leopoldo da Costa e Maria Selma de Jesus Costa, que sempre me deram apoio e me ajudaram nos momentos fáceis e difíceis. Além disso, sou grato a eles por tudo que fizeram por mim, porque desde pequeno, sempre foram preocupados com minha vida acadêmica e são até hoje.

A minha namorada, Kamila Cristina Silva, pelo companheirismo, paciência, por me apoiar e dar força nos momentos de dificuldades e estresse por conta do trabalho.

Ao meu orientador Dr. Luiz Cláudio Oliveira, e ao meu coorientador, Me. Alan Mendes Marotta, onde ambos aceitaram com boa vontade serem meus orientadores neste trabalho.

“Somos o que fazemos para mudar o que fomos”
Guilherme de Sá – Velhos outonos

RESUMO

O presente Trabalho de Conclusão de Curso trata do projeto e desenvolvimento de uma máquina que identifica e separa moedas para fornecer troco. Este projeto tem como foco a redução o erro humano em caixas de comércio. Muitas vezes, se realiza uma compra e o troco não condiz com o valor real. Além disso, foi observada a inexistência de um fabricante de máquinas cuja finalidade seja essa. Para realizar este trabalho, a princípio, foi feita uma revisão bibliográfica, a fim de se encontrar um tipo de construção que atendesse a ideia do projeto. Em seguida, criou-se um algoritmo para tratar o fornecimento de troco no *software* Arduino. Este foi implementado para realizar a leitura do teclado matricial, imprimir as informações em um *display* LCD e enviar sinais para comandar os servomotores. Logo após, desenvolveu-se a estrutura do projeto no *software* Solidworks. Por último, foi construída uma placa de circuito impresso na forma de *shield*, para realizar a comunicação da placa microcontroladora com os sensores, motores, teclado e display. Ao final, conseguiu-se obter um protótipo robusto, que contém pequenos erros de funcionamento na leitura das moedas, mas que consegue fornecer troco de forma satisfatória. Contudo, demorou-se muito para resolver problemas de construção e programação da máquina, o que impediu o desenvolvimento do algoritmo de visão computacional, responsável pela separação das moedas.

Palavras-chave: Fornecimento de troco. Dispositivo mecatrônico. Visão computacional aplicada.

ABSTRACT

This Course Conclusion Assignment deals with the project and development of a machine identifications and separation of coins to provide change. This project focuses on reducing human errors in trade's cashier. Many times, people buy a product and the change does not match the real value. Furthermore, there is not a manufacturer machine whose purpose is this. To carry out this work, at first, a literature review has been made in order to find a type of building that would fit the idea of the project. Then, an algorithm to treat the delivering change was created in the Arduino software. This was implemented to carry out the reading of the matricial keyboard, information display and send signals to command of servomotors. Later, the project structure in SolidWorks software was developed. After it, a printed circuit board was built in the form of shield to perform the communication of the microcontroller board with the sensors, motors, keyboard and display. Finally, it was possible to obtain a robust prototype, which contains small operating errors in reading the coins, but it can give change satisfactorily. However, it has been spent a long time to solve construction problems and machine programming, which precluded the development of computer vision algorithm, responsible for the separation of coins.

Keywords: Change supply. Mechatronic device. Computer vision applied.

LISTA DE FIGURAS

- Figura 1.1 – *Vending machine* de refrigerantes e salgados.
- Figura 1.2 – *Change machine Bill Changer* cm1250.
- Figura 1.3 – Máquina separadora e contadora de moedas CS885 Menno.
- Figura 1.4 – Máquina Cata Moeda.
- Figura 2.1 – Máquina separadora de moedas Klopp SE Basic Series Coin Sorter.
- Figura 2.2 – Limiarização de imagem de um pé de milho.
- Figura 2.3 – Detecção de borda.
- Figura 3.1 – Placa micro controladora Arduino Mega 2560.
- Figura 3.2 – *Display* LCD 16x2.
- Figura 3.3 – Esquema de ligação de um *display* LCD ao Arduino, com ajuste de contraste e brilho.
- Figura 3.4 – Sensor ótico TCRT5000.
- Figura 3.5 – Estrutura de um servo motor.
- Figura 3.6 – Servomotor MG90s.
- Figura 3.7 – Duty cycle e período de PWM do MG90s.
- Figura 3.8 – Relação de linhas e colunas do teclado matricial.
- Figura 3.9 – Teclado matricial.
- Figura 3.10 – Primeira parte do fluxograma da lógica do programa de fornecer troco.
- Figura 3.11 – Segunda parte do fluxograma da lógica do algoritmo de fornecer troco.
- Figura 3.12 – Esquema da montagem de teste para o sistema de troco.
- Figura 3.13 – Projeto da chapa com o rolamento.
- Figura 3.14 – Base para dispensa de moedas.
- Figura 3.15 – Funil para a entrada das moedas nos tubos e para liberação do troco.
- Figura 3.16 – Sistema de fornecimento de troco para um valor de moeda.
- Figura 3.17 – Suporte do sistema de troco.
- Figura 3.18 – Apoio do servomotor do separador e suporte inclinado.
- Figura 3.19 – Bloqueador de moedas do separador.
- Figura 3.20 – Sistema de separação de moedas completo.
- Figura 3.21 – Sistema de comando da máquina.
- Figura 3.22 – Base superior da máquina.

Figura 3.23 – Estrutura inferior da máquina.

Figura 3.24 – Carcaça da máquina.

Figura 3.25 – Limitador acoplado ao sistema de troco.

Figura 3.26 – Suporte para apoio da fonte de computador.

Figura 3.27 – Layout da placa eletrônica.

Figura 3.28 – Esquema de ligação para o sensor óptico.

Figura 3.29 – Polarização direta do LED infravermelho do sensor.

Figura 4.1 – Menu inicial da máquina.

Figura 4.2 – Exibição do montante na máquina.

Figura 4.3 – Quantidade de moedas geral e específica na máquina.

Figura 4.4 – Tela para o usuário digitar e conferir o troco digitado.

Figura 4.5 – Tela exibida ao confirmar o troco.

Figura 4.6 – Tela exibida quando se digita um troco maior do que o montante da máquina.

Figura 4.7 – Máquina de separar moedas para fornece troco. Sistema completo.

Figura 4.8 – Projeto inicial da máquina, que foi aprimorado.

Figura 4.9 – Vista frontal da máquina.

Figura 4.10 – Sistemas de separação de moedas e de troco.

Figura 4.11 – Vista lateral da máquina.

Figura 4.12 – *Shield* com os componentes montados e conectado ao Arduino.

Figura 4.13 – *Layout* da placa de circuito impresso, com os componentes soldados.

LISTA DE TABELAS

Tabela 2.1 – Características das moedas da primeira família do Real.

Tabela 2.2 - Características das moedas da segunda família do Real.

Tabela 3.1 – Funções dos pinos do *display* LCD.

Tabela 3.2 - Cronograma de trabalho para o desenvolvimento do Trabalho de Conclusão de Curso.

Tabela 3.3 – Custos detalhados do projeto da máquina.

Tabela 4.1 – Resultados dos cálculos para a primeira família do Real.

Tabela 4.2 – Resultados dos cálculos para a segunda família do Real.

Tabela 4.3 – Resultados dos testes de contagem das moedas.

Tabela 4.4 – Resultados dos testes de liberação das moedas.

Tabela 4.5 – Resultados da eficiência da leitura dos sensores.

Tabela 4.6 – Resultados da eficiência na dispensa de moedas.

Tabela 4.7 – Resultados dos testes de fornecimento de troco.

Tabela 4.8 – Eficiência de fornecimento de troco.

Tabela 5.1 – Eficiência média do fornecimento de troco

LISTA DE NOTAÇÕES E ACRÔNIMOS

Letras Latinas

g – grama

GND – terra (0V)

Hz – frequência [Hertz]

K - quilo

kgf.cm – unidade de torque [quilograma-força x centímetro]

mA – miliampere

mm – milímetro

m – metro

M - mega

ms – milissegundo

N.m – unidade de torque [*Newton* x metro]

s – segundo

V – tensão [volt]

Vcc – tensão de alimentação [volt]

Letras Gregas

Ω – ohm

Acrônimos

Change machines – Máquinas que realizam troca de dinheiro

LCD – *Liquid Crystal Display* (*display* de cristal líquido)

LED – *Light Emitting Diode* (diodo emissor de luz)

Duty cycle – ciclo ativo de trabalho do PWM

PWM – *Pulse Width Modulation* (modulação por largura de pulso)

Vending machines – Máquinas de vendas

SUMÁRIO

1 – Introdução.....	01
1.1 – Definição do problema.....	01
1.2 – Motivação.....	01
1.3 – Objetivo geral.....	02
1.4 – Objetivos específicos.....	02
1.5 – Estado da arte.....	03
1.6 – Escopo do trabalho.....	06
2 – Revisão da literatura.....	07
2.1 – Contextualização.....	07
2.1.1 – <i>Vending Machines</i>	07
2.1.2 – Separadores e contadores de moedas.....	08
2.2 – Modelos de separadores de moedas.....	08
2.3 – Especificações das moedas da família do Real.....	09
2.4 – Fundamentação teórica.....	11
2.4.1 – Visão computacional.....	11
2.4.2 – Métodos de organização de um sistema de visão computacional.....	11
2.4.3 – OpenCV.....	13
3 – Metodologia.....	14
3.1 – Materiais.....	14
3.1.1 – Arduino Mega 2560.....	14
3.1.2 – <i>Display</i> LCD.....	15
3.1.3 – Sensor óptico.....	17
3.1.4 – Servo motores.....	18
3.1.5 – Teclado matricial.....	21
3.2 – Métodos.....	22
4 – Resultados e discussões.....	46
5 – Conclusões.....	58
6 – Propostas para trabalhos futuros.....	53
7 – Referências bibliográficas.....	64
8 – Apêndices.....	68

1. INTRODUÇÃO

Máquinas de separação e contagem de moedas são de grande importância e utilização em setores comerciais, principalmente por sua praticidade e agilidade em particionar montantes de dinheiro. Porém, não existe no mercado um equipamento que seja capaz de realizar as funções citadas anteriormente e que ainda possa fornecer troco. Geralmente, sistemas que devolvem moedas são de uso restrito, ou seja, fazem parte de outros aparelhos. A aplicação mais usual é em *vending machines*, *change machines* (comum no exterior) e caixas de autoatendimento de bancos.

Como não existem máquinas com a finalidade de separar, identificar e fornecer troco não integradas a outros sistemas, entendeu-se ser viável o desenvolvimento de um protótipo que execute tais funções. Além disso, o fato de automatizar o processo de particionar montantes de dinheiro evita uma situação corriqueira que é o erro humano, o que pode causar uma insatisfação no cliente (OZÓRIO, FIGUEIREDO e ARKADER, 2002).

1.1. DEFINIÇÃO DO PROBLEMA

Um problema comum nos estabelecimentos comerciais é o erro humano ao fornecer um troco, o que pode gerar prejuízo para a loja ou para o cliente. Isso pode trazer uma insatisfação do consumidor e fazê-lo deixar de comprar naquele local.

Para tornar mais eficaz os pagamentos, de forma automatizada, uma máquina de fornecer trocos poderia ser ligada aos computadores, e utilizar o próprio sistema da empresa para enviar os sinais de comando para o aparelho.

1.2. MOTIVAÇÃO

O erro no troco pode prejudicar tanto o funcionário, que terá que arcar com o prejuízo causado por sua falta de atenção, quanto o cliente, que pode ficar insatisfeito com o atendimento devido ao erro na devolução do dinheiro, podendo até deixar de comprar no local ou gerar constrangimentos.

Partindo desse princípio, houve o interesse em se estudar e desenvolver um projeto que visa amenizar este problema. A ideia surgiu através de uma máquina de separação e contagem de moedas. Pensou-se em realizar um aprimoramento, fazendo-a fornecer o troco, de forma a automatizar o processo e ampliar as aplicações desse produto, visto que seu uso nos dias de hoje é limitado basicamente à caixas de banco, *vending machines* e *change machines*.

Além disso, foi pensada a importância acadêmica. Isso porque o projeto abrange uma área de importante aplicação: visão computacional aplicada. Visto a ampla utilização e importância desse método, surgiu o interesse em se estudar essa aplicação em um projeto mecatrônico, procurando adquirir e complementar o conhecimento acadêmico.

1.3. OBJETIVO GERAL

O objetivo geral do presente trabalho é projetar, desenvolver e construir uma máquina que seja capaz de identificar e separar moedas, através da utilização de visão computacional, para fornecer trocos.

1.4. OBJETIVOS ESPECÍFICOS

Os objetivos específicos são:

- Projetar um protótipo da máquina utilizando *softwares* SolidWorks e/ou AutoCAD;
- Construir a máquina de identificação e separação de moedas com fornecimento de troco;
- Analisar a construção física do protótipo;
- Desenvolver dois algoritmos: um para realizar o fornecimento do troco e transmitir informações sobre a máquina, e outro para a separação de moedas por visão computacional;
- Realizar testes e ajustes no projeto, a fim de obter um funcionamento condizente com o proposto.

1.5. ESTADO DA ARTE

Atualmente, não existem muitas aplicações de máquinas que devolvem o troco em moedas. Sua utilização geralmente é em caixas eletrônicas, em *vending machines* e *change machines*. Nas *vending machines*, o cliente deposita o dinheiro (geralmente em cédulas ou com cartões de crédito), escolhe o produto desejado, e caso este seja de menor valor, o troco é devolvido (caso o pagamento tenha sido feito com cédulas).

Estas máquinas são muito utilizadas em aeroportos, rodoviárias e hospitais, e podem vender produtos como alimentos e refrigerantes. Um exemplo de *vending machine* pode ser visualizada na Figura 1.1.



Figura 1.1 – *Vending machine* de refrigerantes e salgados (WD, 2015).

Já as *change machines* são máquinas em que o usuário troca notas, (geralmente de valores mais baixos) por moedas. O processo é quase o mesmo das *vending machines*. Deposita-se as notas na máquina e se escolhe por quais moedas se deseja trocar. A seguir, na Figura 1.2, é possível observar um modelo dessa máquina. No caso desta, é possível trocar apenas notas de \$1 dólar por moedas.



Figura 1.2. – Change machine *Bill Changer cm1250* (cf. <http://amusementunlimited.com/wp-content/uploads/2015/05/bill-changer-change-machine-cm1250.jpg>, 2015).

Quanto à separação e contagem de moedas, existem vários modelos no mercado, os quais variam o modo de funcionamento, podendo ser manual ou automático. As máquinas mais modernas conseguem separar as moedas por família. Abaixo, são apresentadas mais algumas características deste tipo de dispositivo:

- Em média, a capacidade de moedas no reservatório de entrada é próxima a 300;
- Os reservatórios para armazenagem das moedas separadas comportam entre 150 a 200 moedas;
- Separam em média 200 a 250 moedas por minuto.

A seguir, na Figura 1.3, é possível visualizar uma máquina separadora e contadora de moedas. Este modelo é um dos mais acessíveis encontrado à venda no Brasil. Seu preço é de aproximadamente R\$1500,00.



Figura 1.3 – Máquina separadora e contadora de moedas CS885 Menno (Triturare, 2015).

Por último, uma outra aplicação de contadores e separadores de moedas é em uma rede de franquias que surgiu em Florianópolis (Santa Catarina), denominada de Cata Moeda. Tal rede gerencia máquinas, situadas em comércios de diversos setores, como shoppings, supermercados e farmácias.

A franquia disponibiliza máquinas que trocam moedas por cupons, os quais podem ser utilizados dentro das lojas onde os aparelhos estão instalados. Os clientes depositam as moedas em um recipiente, essas são separadas e contadas, e o valor é mostrado em uma tela. Ao final, é emitido um cupom que informa o crédito que a pessoa depositou no Cata Moeda.

Esse tipo de sistema também é adotado em outros países, mas nessas as moedas são trocadas por notas, e não cupons. Na Figura 1.4, a seguir, é visto uma máquina Cata Moeda.



Figura 1.4 – Máquina Cata Moeda (UOL, 2015).

1.6. ESCOPO DO TRABALHO

O trabalho está dividido em seis capítulos. No primeiro, é realizada uma introdução sobre máquinas separadoras e contadoras, além de conter a motivação para a realização do projeto e os objetivos do trabalho. Já no segundo capítulo, é apresentada a revisão de literatura e a fundamentação teórica, a fim de facilitar o entendimento do trabalho, através da definição de conceitos.

No Capítulo 3, é detalhado como foi o processo de pesquisa e desenvolvimento da máquina, tratando sobre os materiais e métodos que foram estudados e aplicados. Em seguida, no Capítulo 4, são discutidos os resultados, descrevendo os processos que obtiveram êxito e quais foram as dificuldades, citando pontos importantes que foram observados.

No Capítulo 5, são apresentadas as conclusões e as considerações finais. Por último, no Capítulo 6, são propostas ideias para implementações futuras no trabalho.

2. REVISÃO DA LITERATURA

Através de uma pesquisa realizada na internet, verificou-se que não existem estudos específicos sobre máquinas que fornecem trocos, quando tratadas como sistemas independentes.

Nas *vending machines*, o custo de manutenção é alto, e o setor sofre com a falta de padronização de cédulas e moedas, que acabam não sendo identificadas corretamente pelas máquinas.

Porém, no caso do real, as cédulas já chegaram a um nível adequado de padronização para que sejam aceitas pelos leitores das máquinas, mas as moedas só devem atingir esta adequação com a terceira família do Real, que deve ser lançada em breve pelo Banco Central (ALMEIDA, 2013).

2.1. CONTEXTUALIZAÇÃO

2.1.1. *Vending Machines*

As *vending machines* são máquinas que têm uma origem muito antiga. Segundo COMPANYY (2012), a primeira máquina conhecida foi inventada por Heron de Alexandria em meados de 215 a.C., um engenheiro e matemático que criou uma máquina que ao receber uma moeda, dispensava água benta. A obra consistiu em uma máquina, que recebia a moeda, esta caía dentro de uma panela ligada a uma alavanca. A alavanca era acionada e liberava um pequeno fluxo de água para fora.

No entanto, foi no início dos anos 1880 que as primeiras máquinas automáticas de venda alimentadas por moedas foram introduzidas. Estas foram instaladas em Londres, para dispensar cartões postais. Ao mesmo tempo, Richard Carlisle, um editor inglês e proprietário de livraria, inventou uma máquina de venda automática que dispensava livros. Nos Estados Unidos a primeira máquina foi introduzida em 1888 pela Companhia Thomas Adams Gum, vendendo chicletes em Nova York, nas plataformas de metrô. Em 1897 a Companhia *Pulver Manufacturing* adicionou figuras animadas para suas máquinas de venda, o que proporcionava entretenimento já que as figuras apresentavam movimento sempre que alguém comprava um chiclete de suas máquinas. Esta

iniciativa deu origem a um novo conceito conhecido como estimuladores de mercado, base para máquinas caça-níqueis e jogos de fliperama.

Atualmente, no Brasil, segundo ABVA (2014), estima-se que o mercado opere atualmente com 45.000 *vending machines*, gerando um faturamento anual aproximado de 250 milhões de reais. O potencial de crescimento é enorme tendo em vista a familiarização do brasileiro com o uso de máquinas nas mais variadas aplicações, e especialmente pela visão de negócios que elas proporcionam.

Visto o crescimento no setor, em maio de 2000 criou-se a ABVA - Associação Brasileira de Vendas Automáticas.

2.1.2. Separadores e contadores de moedas

Atualmente, os caixas eletrônicos dos bancos aceitam depósitos apenas em cédulas. Uma abertura para o depósito de moedas seria interessante para pessoas interessadas em depósitos com valores baixos, ou valores que incluam centavos.

Além disso, um contador de moedas automático seria interessante para empresas de pequeno porte que não possuem condições de comprar máquinas similares no mercado – as quais facilmente custam mais de R\$ 1500,00.

Contadores de moedas acessíveis são importantes para bancos e pequenas empresas pelo baixo valor de compra e manutenção, sem ter uma perda de qualidade (TURATTI *et al.*).

2.2. MODELOS DE SEPARADORES DE MOEDAS

De acordo com uma pesquisa realizada na internet, verificou-se que a maioria das máquinas separadoras realiza o processo mecanicamente. As moedas passam por trilhos, e estes possuem recortes com seus respectivos diâmetros, permitindo que cada valor seja separado em um recipiente, como pode ser visto na Figura 2.1. Além disso, é possível observar como é a entrada do dinheiro. A maioria das máquinas pesquisadas possuem discos, com aletas para mexer as moedas. Possuem também desníveis em suas extremidades, de forma que, pela rotação, as moedas entrem nesses espaços, formando uma fila e fazendo uma moeda por vez entrar no trilho.



Figura 2.1 – Máquina separadora de moedas Klopp SE Basic Series Coin Sorter (cf. <https://drawingthedata.files.wordpress.com/2014/10/1000x1000.jpg>, 2015).

Uma máquina que consegue separar as moedas da família do Real é a Menno CS885, porém somente as da segunda geração. De acordo com seu catálogo (MENNO, 2015), esta máquina possui as seguintes características:

- Velocidade de contagem: 216 moedas por minuto;
- Capacidade de bandeja: de 300 a 500 moedas;
- Capacidade dos receptores: 80 a 150 moedas;
- Capacidade do visor para a contagem: 4 dígitos;
- Tensão: 127 ou 220 V / 50 Hz;
- Dimensões: 355 x 330 x 266 mm;
- Peso líquido: 4,5 kg

É possível ver este modelo de máquina na Figura 1.3, mostrada no Capítulo 1.

2.3. ESPECIFICAÇÕES DAS MOEDAS DA FAMÍLIA DO REAL

Segundo o Banco Central do Brasil (BCB, 2015), as moedas da primeira família do Real possuem as seguintes características, conforme é mostrado na Tabela 2.1.

Tabela 2.1 – Características das moedas da primeira família do Real (BCB, 2015).

Valor Facial (R\$)	Diâmetro (mm)	Peso (g)	Espessura (mm)	Material
0,01	20	2,96	1,2	Aço inoxidável
0,05	21	3,27	1,2	Aço inoxidável
0,1	22	3,59	1,2	Aço inoxidável
0,25	23,5	4,78	1,4	Aço inoxidável
0,5	23	3,92	1,2	Aço inoxidável
1	24	4,27	1,2	Aço inoxidável

Já para a segunda família, as características podem ser vistas na Tabela 2.2.

Tabela 2.2 - Características das moedas da segunda família do Real (BCB, 2015).

Valor Facial (R\$)	Diâmetro (mm)	Peso (g)	Espessura (mm)	Material
0,01	17	2,43	1,65	Aço revestido de cobre
0,05	22	4,1	1,65	Aço revestido de cobre
0,1	20	4,8	2,23	Aço revestido de bronze
0,25	25	7,55	2,25	Aço revestido de bronze
0,50 (1998 à 2001)	23	9,25	2,85	Cuproníquel
0,50 (2002 em diante)	23	7,81	2,85	Aço inoxidável
1,00 (1998 a 2001)	27	7,84	1,95	Cuproníquel (núcleo) e Alpaca (anel)
1,00 (2002 em diante)	27	7	1,95	Aço inoxidável (núcleo) e aço revestido de bronze (anel)

2.4. FUNDAMENTAÇÃO TEÓRICA

2.4.1. Visão computacional

Segundo Shapiro (2001), a visão computacional é um área que engloba métodos para a aquisição, processamento, análise e compreensão de imagens. A análise e o processamento de imagens tem como finalidade a obtenção de informações numéricas ou simbólicas, como por exemplo, na forma de decisões.

Esse tratamento de imagem, segundo Forsyth (2003), pode ser visto como a extração de informações úteis a partir de dados da figura usando modelos construídos com a ajuda da geometria, física, estatística e teoria da aprendizagem.

2.4.2. Métodos de organização de um sistema de visão computacional

Geralmente, nos sistemas de visão computacional existem funções típicas, que são executadas e descritas na seguinte ordem:

Aquisição de imagens: uma imagem digital é produzida por um ou vários sensores de imagem, que incluem vários tipos de câmeras sensíveis à luz. Dependendo do tipo do sensor, o resultado de dados é uma imagem 2D comum, um volume 3D, ou uma sequência de imagens. Os valores de *pixel* normalmente correspondem à intensidade da luz em uma ou várias bandas espectrais (imagens de cinza ou coloridas), mas também podem estar relacionado a várias medidas físicas, tais como a profundidade, absorção, ou reflexão de ondas sonoras, eletromagnéticas, ou ressonância magnética nuclear (DAVIES, 2005).

Pré-processamento: para realizar o processamento de imagens em sistemas de visão computacional, a princípio elas precisam ser pré-processadas. Este pré-processamento assume um papel importante para a extração de certas informações que não serão utilizadas para a análise, que podem influenciar de forma indesejada a decisão a ser tomada. Informações não utilizadas abrangem detalhes como cor, reflexos, e até objetos no fundo da imagem. Ao usar o pré-processamento adequado, estas informações podem ser ignoradas ou até retiradas (ZUECH, 1988). Este procedimento tende a melhorar a tomada de

decisão posterior à análise. De maneira geral, o pré-processamento visa melhorar a qualidade da imagem para facilitar o processamento seguinte.

Extração de características: um método de extração de características é a segmentação. Este é um processo de separação de objetos (cada um com atributos uniformes) do resto da imagem. Esse processo cria partições da imagem baseado em uma variável, por exemplo, a intensidade da cor cinza (ZUECH, 1988). A seguir, é explanado dois dos mais comuns tipos de segmentação.

- **Limiarização:** este processo atribui a cor branca para os *pixels* na imagem que tem seus valores de cinza acima do *threshold*, (ou limiar, que é o valor de cinza de uma imagem monocromática), e atribui a cor preta para os que apresentam um valor abaixo. As áreas mais claras que o limiar tornam-se brancas e as mais escuras tornam-se pretas. A imagem resultante é chamada de imagem binária porque em vez de usar um *byte* para representar cada *pixel*, é usado um *bit* com o valor 0 se a região for preta e 1 se for branca (ZUECH, 1988). A seguir, na Figura 2.2, é possível observar uma limiarização realizada em uma fotografia de um pé de milho.



Figura 2.2 – Limiarização de imagem de um pé de milho (SENA JUNIOR, 2003).

- **Detecção de bordas:** nesse processo são utilizados filtros com métodos matemáticos que identificam pontos na imagem onde o seu brilho muda de forma drástica. Esses pontos são organizados em um conjunto de linhas e curvas. Um exemplo de detecção de borda pode ser visto na Figura 2.3.



Figura 2.3 – Detecção de borda (MELLO, 2015).

Processamento de alto nível: segundo Davies (2005), nesta etapa a entrada é tipicamente um pequeno grupo de dados, por exemplo, um conjunto de pontos ou uma região de imagem que se supõe conter um objetivo específico. Esse tipo de processo garante que os dados satisfaçam pressupostos específicos, como o reconhecimento de imagens classificando um objeto e o detectando em diferentes categorias.

Tomada de decisão: é a execução do processo pretendido. No projeto em questão, é o envio do sinal PWM equivalente para o motor movimentar o trilho de acordo com o valor da moeda.

2.4.3. OpenCV

A OpenCV (*Open Source Computer Vision Library*) é uma biblioteca de visão computacional e *software* de aprendizado de máquina livre ao uso acadêmico e comercial.

A biblioteca tem mais de 2500 algoritmos otimizados, que podem ser usados, por exemplo, para detectar e reconhecer rostos e identificar objetos.

As utilidades do seu uso são vastas. A OpenCV é utilizada no desenvolvimento de diversos aplicativos, que variam desde programas simples de colagem de imagens até programas mais complexos como os de auxílio de navegação robótica. Ele é disponibilizado em versões como C++/C/Python e Java, e pode ser usada em múltiplos sistemas operacionais como Windows, Linux, Android e MacOS (OPENCV, 2015).

3. METODOLOGIA

Este trabalho consiste em uma pesquisa com o objetivo de projetar, desenvolver e construir um protótipo que identifique, separe e conte as moedas para fornecer troco. A etapa inicial para a execução desse projeto foi o estudo dos tipos de separadores de moedas e de como aplicar a visão computacional na execução da separação. Para isso, realizou-se uma revisão bibliográfica, onde ao final, conseguiu-se idealizar um modelo que atendesse a ideia do projeto, bem como determinar os materiais, os quais serão descritos a seguir.

3.1. MATERIAIS

A princípio, através de pesquisas e testes, determinou-se o uso dos seguintes materiais para o desenvolvimento da máquina.

3.1.1. Arduino Mega 2560

Para receber os sinais dos sensores, acionar o *display* LCD, ler o teclado matricial e enviar o sinal de PWM para os servomotores fornecerem o troco, a princípio, pensou-se em utilizar uma placa microcontroladora Arduino Uno R3. Mas, ao analisar seu *datasheet* (ARDUINO UNO, 2015), observou-se que este não atenderia à demanda, devido ao fato de possuir somente 6 saídas de PWM, dentre 13 saídas digitais, ou seja, não possui saídas suficientes para acoplar os sensores, teclado e LCD. Assim, optou-se pela placa Arduino Mega 2560, que possui maior quantidade de portas. Segundo dados fornecidos pelo fabricante (ARDUINO MEGA, 2015), esta plataforma possui as seguintes características:

- Tensão de operação: 5V;
- Tensão de entrada: 7-12V;
- Portas digitais: 54 (15 podem ser usadas como PWM);
- Portas analógicas: 16;
- Corrente Pinos I/O (INPUT/OUTPUT): 40mA;
- Corrente Pinos 3,3V: 50mA;
- Velocidade do *clock*: 16MHz.

A Seguir, na Figura 3.1, é possível visualizar uma placa do Arduino Mega 2560.

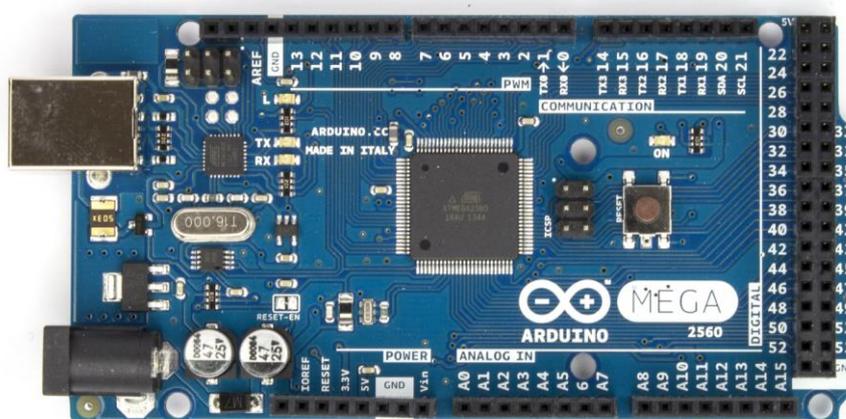


Figura 3.1 – Placa micro controladora Arduino Mega 2560 (ARDUINO, 2015).

3.1.2. *Display* LCD

Para exibir dados como o montante de dinheiro, quantidade de moedas e o valor do troco digitado, optou-se pelo uso de um *display* LCD. Segundo Souza e Lavinia (2010), o *Display* de Cristal Líquido é um periférico que permite o usuário interagir com o equipamento. Os LCDs de caracteres normalmente são compatíveis com o código ASCII, e são capazes de reproduzir letras, números e caracteres especiais, além de caracteres gregos e europeus. Esses dispositivos possuem ainda uma memória RAM interna que permite criar caracteres especiais ou símbolos que podem ser imprescindíveis em determinada aplicação.

Para o projeto, optou-se pelo uso de um LCD 16x2 modelo RT162-7 (32 caracteres alfanuméricos, divididos em duas colunas), o qual pode ser visto na Figura 3.2. Além de ser um dos mais utilizados em equipamentos eletrônicos, seu custo é bastante acessível, e atende bem ao trabalho, visto que será utilizado apenas para exibir uma opção por vez: menu inicial, montante, quantidade de moedas e troco a ser digitado.

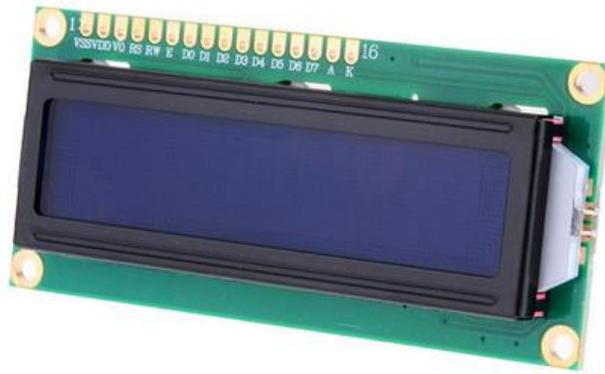


Figura 3.2 – Display LCD 16x2 (FILIPEFLOP, 2015).

O *display* 16x2 possui 16 pinos de acesso com diferentes tipos de funções, conforme pode ser visto na Tabela 3.1.

Tabela 3.1 – Funções dos pinos do *display* LCD (SHENZHEN, 2015).

Pino	Função	Pino	Função
1	GND (0V)	9	DB2
2	VDD (+5V)	10	DB3
3	V ₀	11	DB4
4	RS	12	DB5
5	R/W	13	DB6
6	E	14	DB7
7	DB0	15	A
8	DB1	16	K

O *display* é alimentado com uma fonte CC de 5V (de 4,5 à 5,5V) e sua corrente é entre 1mA à 1,5mA.

O pino R/W muda o estado do LCD entre leitura (*Read*) e escrita (*Write*), e o pino RS (*Register select*) é utilizado para definir o tipo de informação transmitida através da comunicação paralela, se é um dado ou comando. Já o pino E (*Enable*) é utilizado para efetivar a leitura da informação escrita no barramento de dados. Essa leitura é efetuada na borda de descida deste sinal (SOUZA e LAVINIA, 2010).

Os pinos DB0 à DB7 equivalem ao barramento de dados paralelo. Apesar de existirem 8 vias de dados, esses displays também podem operar com 4 vias (DB4 à DB7). A comunicação no modo de 4 *bits* é realizada utilizando apenas as 4 linhas mais significativas de dados (D7 à D4), dividindo o *byte* em 2, e então são

transferidos sempre iniciando pelo mais significativo seguido pelo menos significativo (PEREIRA, 2012).

O pino V_0 , é utilizado para que seja possível ajustar o contraste dos caracteres exibidos no *display*. Para que o ajuste seja variável, é necessário acoplar um potenciômetro. O pino central deverá ser conectado ao V_0 , enquanto os das extremidades sejam ligados um no GND e outro no VDD. A seguir, na Figura 3.3, é possível ver um esquema de ligação, acoplado à uma placa micro controladora Arduino Uno R3.

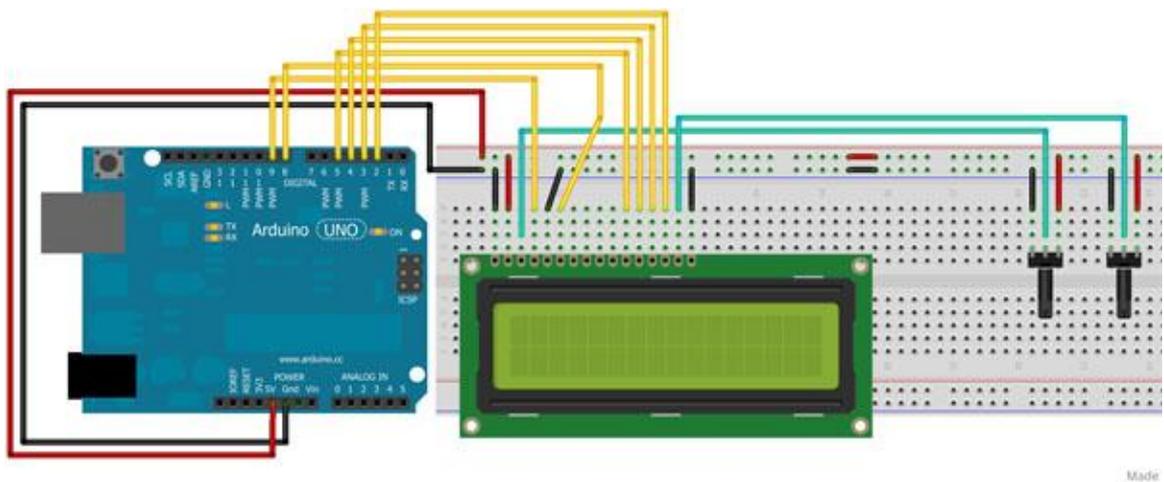


Figura 3.3 – Esquema de ligação de um display LCD ao Arduino, com ajuste de contraste e brilho (cf. <http://www.comofazerascosas.com.br/posts/arquivos-posts/195/esquema-arduino-com-lcd.jpg>, 2015).

Uma situação parecida ocorre com o pino 15. Este é utilizado para ajustar o brilho da tela. Os mesmos procedimentos descritos anteriormente devem ser utilizados para que seja possível controlar sua intensidade. Caso não seja necessário a variação do brilho, o pino 15 pode ser aterrado, ficando no valor máximo. Na Figura 3.3 também é possível verificar este esquema de ligação.

3.1.3. Sensor óptico

Para identificar a entrada das moedas nos tubos, contar o montante e a quantidade de moedas, foi escolhido o uso de sensores ópticos. Estes são sensores cujos elementos de emissão e recepção estão justapostos no mesmo conjunto óptico. Os raios emitidos pelo transmissor refletem na superfície do

objeto detectado e retornam ao elemento receptor. Seu funcionamento baseia-se na emissão de um feixe de luz, o qual é recebido por um elemento fotossensível (WEG, 2015).

Para atender ao projeto, escolheu-se o TCRT5000, devido ao seu baixo custo e sua boa precisão. Segundo o *datasheet* de um fabricante, VISHAY (2015), estes são sensores reflexivos que incluem um emissor de infravermelhos (LED cor azul) e um foto transistor (LED preto), como pode ser visto na Figura 3.4. Tal sensor tem uma capacidade de operação máxima de 950 mm.

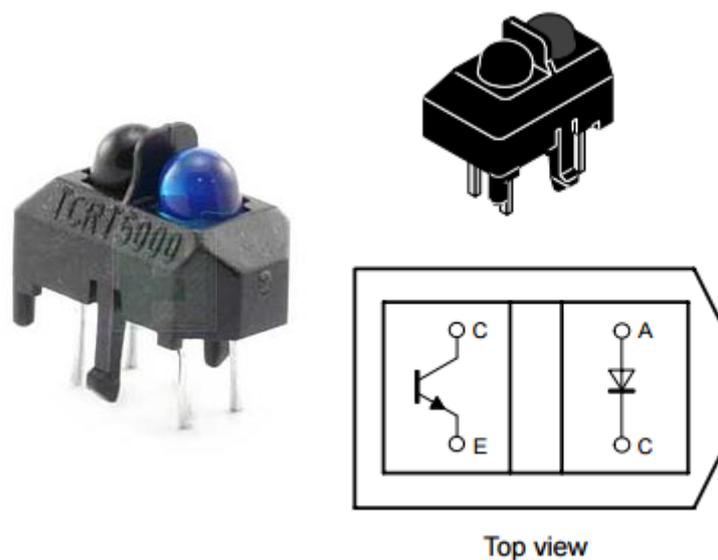


Figura 3.4 – Sensor ótico TCRT5000 (VISHAY, 2015).

3.1.4. Servo motores

Segundo Del Toro (1999) e UNESP (2013), os servo motores são usados em várias aplicações, especialmente naquelas onde se deseja movimentar algo de forma precisa e controlada. Sua característica mais marcante é a sua capacidade de movimentar o seu braço até uma posição e mantê-lo, mesmo quando sofre uma força. Assim, pelo conceito exposto, optou-se por utilizar servo motores para acionar o sistema de fornecimento de troco e o direcionador da rampa por onde a moeda descerá.

Um servo motor é constituído pelas seguintes partes:

- Circuito de Controle: responsável pelo monitoramento do potenciômetro e acionamento do motor visando obter uma posição pré-determinada;

- Potenciômetro: ligado ao eixo de saída do servo, monitora a posição do mesmo, através de sua variação na resistência;
- Motor: movimenta as engrenagens e o eixo principal do servo;
- Engrenagens: reduzem a rotação do motor, transferem mais torque ao eixo principal de saída e movimentam o potenciômetro junto com o eixo;
- Caixa do Servo: caixa para acondicionar as diversas partes do servo.

A seguir, na Figura 3.5, é possível verificar alguns dos mecanismos citados anteriormente.



Figura 3.5 – Estrutura de um servo motor (UNESP, 2013).

Para o projeto, era necessário um servomotor pequeno, que tivesse um torque alto, capaz de movimentar a rampa do separador e liberar as moedas. Assim, através de pesquisas e testes de esforço realizados com alguns motores, determinou-se o uso de um servomotor do tipo MG90s, da Tower Pro, que possui engrenagens de metal e as seguintes especificações, de acordo com seu *datasheet* (TOWER PRO, 2015).

- Peso: 13,4 g;
- Dimensão: 22,5 x 12 x 35,5 mm;
- Torque: 1,8 kgf·cm (0,1765197 N.m) alimentado com 4,8V ou 2,2 kgf·cm (0,2157463 N.m) em 6V;
- Velocidade de operação: 0,1 s/60° (4,8V) ou 0,08 s/60° (6V);

- Tensão de operação: 4,8V – 6V;

A seguir, na Figura 3.6 é possível observar como é o servomotor MG90s, da Tower Pro.



Figura 3.6 – Servomotor MG90s (cf. http://mlb-s1-p.mlstatic.com/micro-servomotor-tower-pro-mg90s-metal-gear-com-acessorios-244101-MLB20281205082_042015-O.jpg, 2015).

Este servomotor possui 3 fios para ligação: marrom, vermelho e amarelo, que são o GND, Vcc e sinal, respectivamente. Além disso, ainda de acordo com o seu *datasheet*, o período deste servomotor é de 20 ms, e seu *Duty Cycle* é de até 2 ms, conforme pode ser visto na Figura 3.7, a seguir.

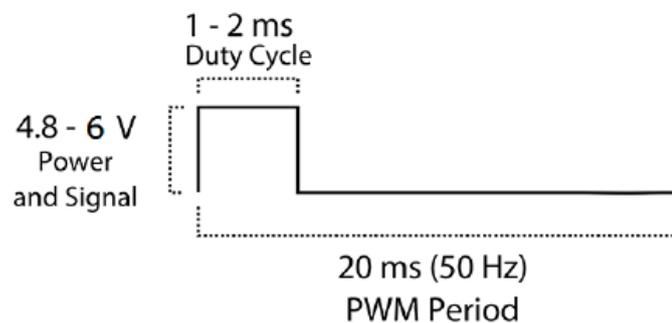


Figura 3.7 – Duty cycle e período de PWM do MG90s (TOWER PRO, 2015).

3.1.5. Teclado matricial

Segundo o *datasheet* de um fabricante (PARALLAX, 2015), o teclado matricial é um componente do Arduino muito utilizado para entrada de dados. Ele possui 16 teclas dispostas em 4 linhas por 4 colunas, e um conector de 8 pinos para ligação. Internamente, são 16 teclas *push-buttons* tipo membrana dispostas na configuração abaixo em um formato *keypad*. Conforme a tecla é pressionada, é feita a conexão entre a linha e a coluna correspondentes. A relação entre linhas, colunas, e suas respectivas ligações podem ser vistas na Figura 3.8.

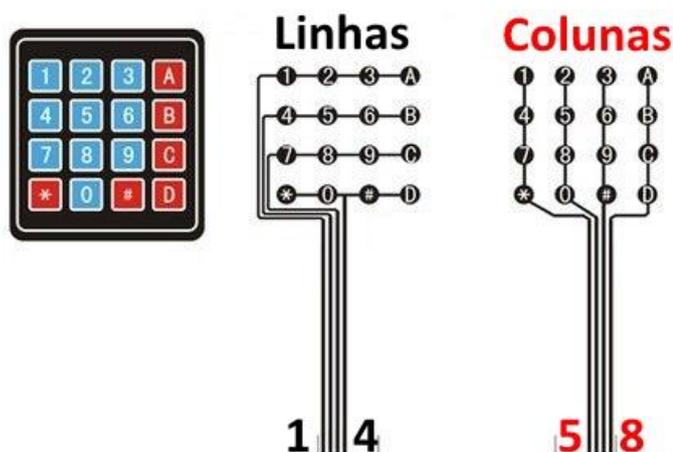


Figura 3.8 – Relação de linhas e colunas do teclado matricial (BLOG FILIPEFLOP, 2015).

Para que se possa digitar o valor do troco, tendo em vista que o projeto não será acoplado a nenhum outro sistema de caixas comerciais (para fins de apresentação), definiu-se a utilização de um modelo de teclado semelhante ao citado anteriormente. Esta escolha deve-se ao seu fácil manuseio e baixo custo.

Esse periférico será utilizado na máquina para que o usuário digite o valor do troco ou chame algumas funções a serem exibidas pelo LCD, as quais possuem teclas pré-determinadas para seus acionamentos. A Figura 3.9 mostra um teclado matricial 4x4, semelhante ao que será utilizado.



Figura 3.9 – Teclado matricial (BLOG FILIPEFLOP, 2015).

3.2. MÉTODOS

Realizada a revisão bibliográfica e definidos os componentes, deu-se início a elaboração de um algoritmo para tratar o problema do fornecimento de troco, utilizando o *software* Dev C++, versão estudantil. O código foi desenvolvido inicialmente neste programa pela facilidade de simulação, já que nele não há necessidade de utilizar a placa microcontroladora.

O programa realiza uma varredura das moedas do valor mais alto para o mais baixo, sequencialmente. Antes de executar a liberação das moedas, é verificado se a máquina possui os valores correspondentes. Se sim, a moeda é liberada. Caso contrário, observa-se a próxima, obedecendo a ordem decrescente dos valores. Por exemplo, deseja-se como troco R\$1,00 e a máquina não possui uma moeda equivalente. Então é verificado se há moedas de R\$0,50 para liberação. Se existir duas, elas são liberadas; se tiver somente uma, esta é liberada e ocorre uma nova verificação, e assim por diante. Caso a máquina não possua moedas, ela não executa nenhuma função, e avisa ao usuário que o troco é inválido.

O valor do troco digitado é comparado inicialmente com a moeda de R\$1,00. Enquanto o valor digitado for maior que o da moeda (e se tiver moedas deste valor na máquina), elas são liberadas, uma a uma. Quando o valor do troco ficar menor que R\$1,00, muda-se o valor de comparação para R\$0,50, e assim até chegar a R\$0,05. Isto foi feito através de *loops*.

Para valores menores que R\$0,05, ocorre outra checagem: se o valor for maior que R\$0,03, é liberada uma moeda de R\$0,05; caso contrário, o troco é finalizado.

Assim, desenvolveu-se uma lógica conforme os fluxogramas a seguir, implementados online no site Gliffy (cf. <https://www.gliffy.com/>), visto nas Figuras 3.10, e 3.11, respectivamente.

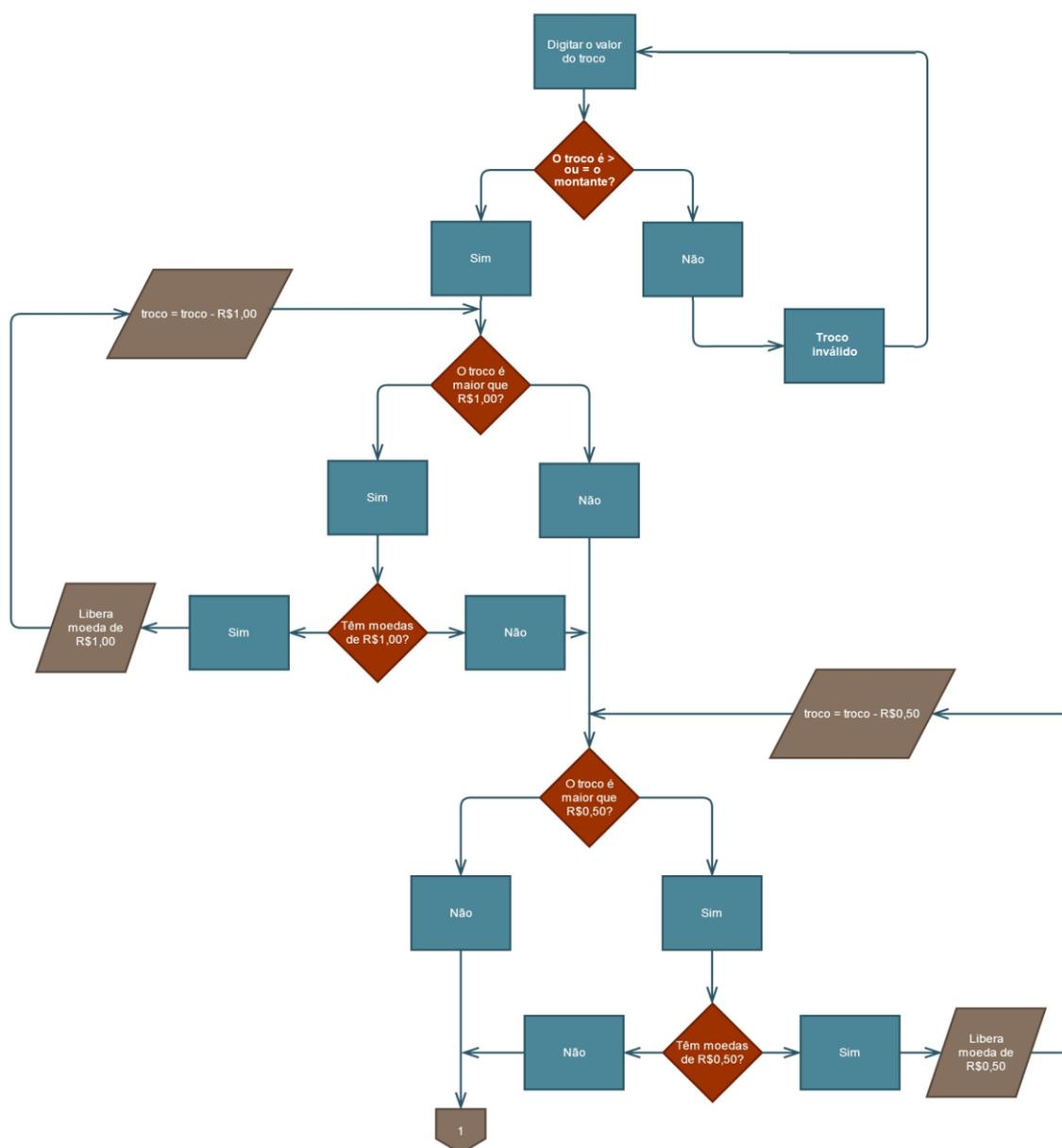


Figura 3.10 – Primeira parte do fluxograma da lógica do programa de fornecer troco.

montagem ficou semelhante ao que é mostrado na Figura 3.12, que foi desenvolvido no *software* Fritzing, disponível gratuitamente na internet.

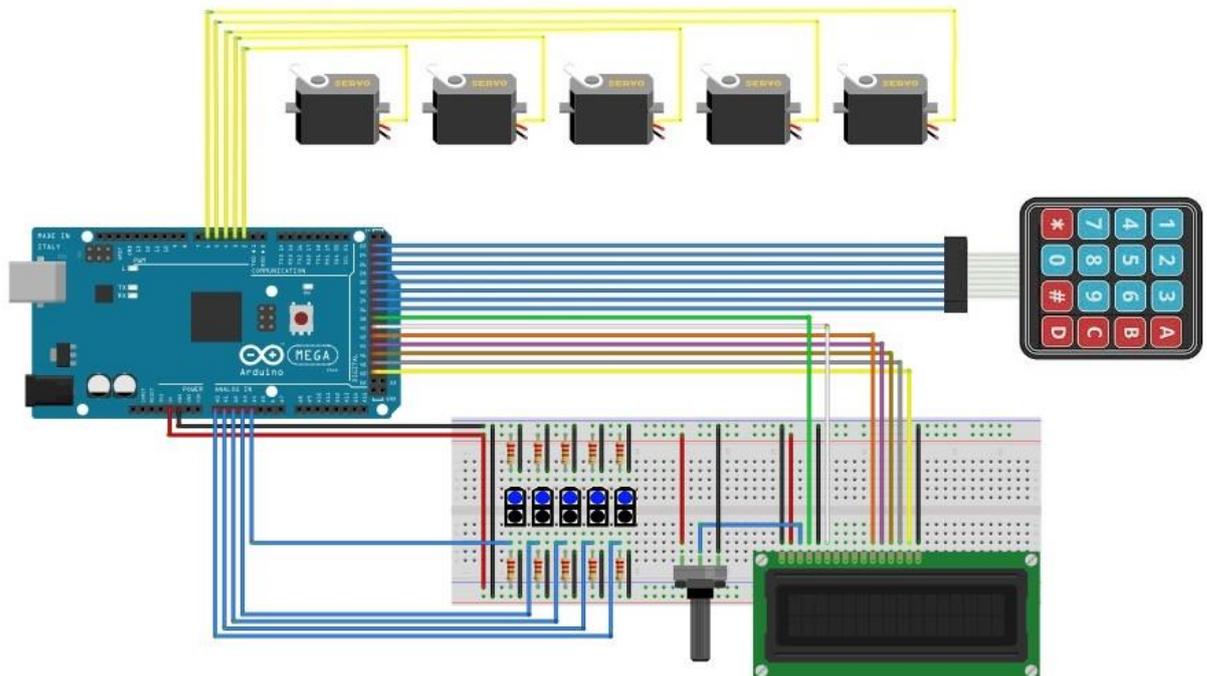


Figura 3.12 – Esquema da montagem de teste para o sistema de troca.

O LCD e o teclado foram ligados nas portas digitais do Arduino. Assim, os pinos foram conectados da seguinte maneira:

Teclado matricial:

- Pinos 22, 24, 26, 28, 30, 32, 34 e 36, do Arduino correspondem, respectivamente aos conectores 1 à 8 do teclado matricial.

Display LCD:

- Pino 38 do Arduino no pino 4 (Rs) do LCD;
- Pino 40 do Arduino no pino 6 (E) do LCD;
- Pino 42 do Arduino no pino 11 (D4) do LCD;
- Pino 44 do Arduino no pino 12 (D5) do LCD;
- Pino 46 do Arduino no pino 13 (D6) do LCD;
- Pino 48 do Arduino no pino 14 (D7) do LCD;

- Pino 50 do Arduino no pino 15 (A) do LCD.

Servomotores:

Os servomotores foram ligados nas portas de PWM do Arduino, da seguinte maneira:

- Pinos 2 ao 6 do Arduino: conectados aos fios de sinais dos servos, correspondentes aos valores de R\$1,00 à R\$0,05, respectivamente.

Sensores ópticos

Para conectar os sensores ópticos foi necessário utilizar as entradas analógicas do Arduino (ANALOG IN), conforme a seguir:

- Pinos 0 ao 4 do Arduino: entradas dos sinais dos sensores, correspondentes aos valores de R\$0,05 à R\$1,00, respectivamente.

Por fim, os servos, *display* e os sensores foram alimentados e aterrados nos pinos “5V” e “GND” do Arduino, respectivamente.

Com a montagem para teste concluída, o algoritmo desenvolvido em linguagem C foi transcrito para o *software* Arduino, responsável por gravar os códigos no microcontrolador da placa. Além de fornecer o troco, foi implementada outras funções no programa, para que se pudesse visualizar as seguintes informações sobre a máquina no *display* LCD:

- Modo “*standby*”, para informar ao usuário as opções e operações que a máquina é capaz de realizar. Esse é o menu inicial da máquina;
- Montante;
- Quantidade de moedas geral e de cada valor;
- Troco que está sendo digitado;
- Quando o troco desejado for confirmado, este valor é exibido;
- Mensagem de advertência caso a quantidade de moedas seja insuficiente para o troco.

Além disso, foi desenvolvida a comunicação do teclado com o restante do sistema utilizando uma biblioteca obtida da internet (Keypad.h), onde as teclas receberam as seguintes funções:

- Teclado numérico: utilizado somente para digitar o troco;
- Tecla A: exibir o montante da máquina;
- Tecla B: exibir a quantidade geral de moedas e de cada valor;
- Tecla C: entrar no modo de digitar o troco. O teclado numérico funciona apenas nesse modo;
- Tecla D: voltar ao menu inicial;
- Tecla *: confirmar o troco digitado. Ao apertar essa tecla, as moedas correspondentes ao troco são liberadas. Essa função funciona somente no modo de digitar o troco;
- Tecla #: corrigir o troco digitado. Essa função funciona apenas quando estiver no modo de digitar o troco.

Por último, as linhas de código correspondentes a leitura dos sensores foram desenvolvidas. Realizou-se testes e ajustes, finalizando o sistema de troco da máquina. O código final está disponível no Apêndice A.

Com o algoritmo finalizado, iniciou-se o desenvolvimento do projeto mecânico da máquina, cujas as peças projetadas podem ser vistas no Apêndice B, com suas vistas em 2D e respectivas dimensões. Para esta etapa, utilizou-se o *software* SolidWorks, versão estudantil.

A primeira parte que foi projetada é o sistema de fornecer trocos. A princípio, determinou-se o tamanho do tubo que iria servir como recipiente para as moedas. Como na sua base deveria ter uma chapa da espessura das moedas, que variam de 1,2 mm à 2,85 mm, pensou-se em não se colocar grandes montantes, acreditando que as chapas poderiam sofrer torções ou que os servomotores não conseguiriam girar e liberar o troco.

Assim, determinou-se como comprimento do tubo 100 mm. Como os sensores seriam acoplados à esses tubos, o limite de altura das moedas caiu para 90 mm. Então, foi calculado a quantidade de moedas de cada valor que estes comportariam, o peso gerado pelo montante, e o valor médio das moedas, em Reais. Os cálculos foram feitos de acordo com as Equações 3.1, 3.2 e 3.3.

$$\text{Quantidade de moedas de R\$}(x) = \frac{90}{\text{espessura da moeda de R\$}(x)} \quad (3.1)$$

onde a espessura é fornecida em (mm);

$$\text{Montante} = \text{Quantidade de moedas de R\$}(x) \times \text{valor da moeda de R\$}(x) \quad (3.2)$$

$$\text{Peso} = \text{quantidade de moedas de R\$}(x) \times \text{peso da moeda de R\$}(x) \quad (3.3)$$

onde o peso da moeda e o total é em (g).

Realizados os cálculos, observou-se que com 90 mm de comprimento, o tubo comportaria uma boa quantidade de moedas, e que não forçaria muito a chapa situada na base do tubo e nem o servomotor. Sendo assim, ficou convencionado 90 mm como altura do reservatório das moedas.

Então, foram escolhidos canos de PVC para servirem como armazenamento das moedas. Apesar de cada valor de moeda possuir um diâmetro, resolveu-se padronizar essa medida para os reservatórios. Assim, foram utilizados canos reaproveitados de 33,30 mm de diâmetro externo e 28 mm interno.

Logo após, foram projetadas no SolidWorks as chapas que serviriam de apoio para as moedas no fundo dos tubos. Além disso, elas são responsáveis por dispensar as moedas, através de rotações de 90°, partindo e retornando a 0°. Tais movimentos são realizados pelos servomotores acoplados às chapas. Para dimensionar as espessuras, foram verificadas as dimensões das moedas. As moedas da primeira família do Real são as que possuem as menores espessuras, como pode ser visto na Tabela 2.1. Seus valores foram usados como referência para pesquisar chapas com essa espessura. Isso porque, se na hora do fornecimento do troco uma moeda da primeira família estiver no momento de ser dispensada, a chapa conseguirá realizar o procedimento. O mesmo ocorrerá com uma moeda da segunda família, já que suas espessuras são maiores.

Se fosse ao contrário, ou seja, se tivesse sido adotado como altura das chapas os valores das moedas da segunda família, o sistema de troco funcionaria

perfeitamente para estas moedas, mas não para as da primeira família. Isso porque se a moeda a ser dispensada possui uma altura menor que a da chapa, e a próxima também, quando a chapa abrir é provável que sairão duas moedas, e não uma, como previsto, visto que algumas moedas da segunda família possuem mais que o dobro da espessura das da primeira família (vide Tabelas 2.1 e 2.2).

Utilizando essas chapas (com as menores espessuras), juntamente com os tubos, os quais foram espaçados de suas bases com uma altura correspondente a maior espessura da moeda de cada valor, o problema citado acima foi resolvido.

Outra preocupação foi em como seria o deslizamento da chapa. A princípio, pensou-se em deixá-la encostada na base. Mas com o tempo de uso, provavelmente elas se desgastariam pelo atrito. Para rotacioná-las de forma a não gerar um esforço nos encaixes dos servomotores, foram utilizados rolamentos. Ao final, a chapa ficou com 42 mm de largura e 73 mm de comprimento. Este último, foi determinado considerando o raio de deslocamento do encaixe do servomotor, que tem 18 mm. Foi levado em conta também 5 mm para o eixo do rolamento, 48 mm para a área do tubo, e 7 mm para o posicionamento do encaixe dos servos, onde foi feito um furo para se ter acesso ao parafuso do encaixe.

Já a largura foi adotada considerando-se o diâmetro dos tubos com um espaçamento para cada lado, deixando o tubo no centro. O projeto do desenho tomou a forma que pode ser vista na Figura 3.13. É possível visualizar também o rolamento com o seu eixo encaixado à chapa e o furo para o acesso ao encaixe dos servos.

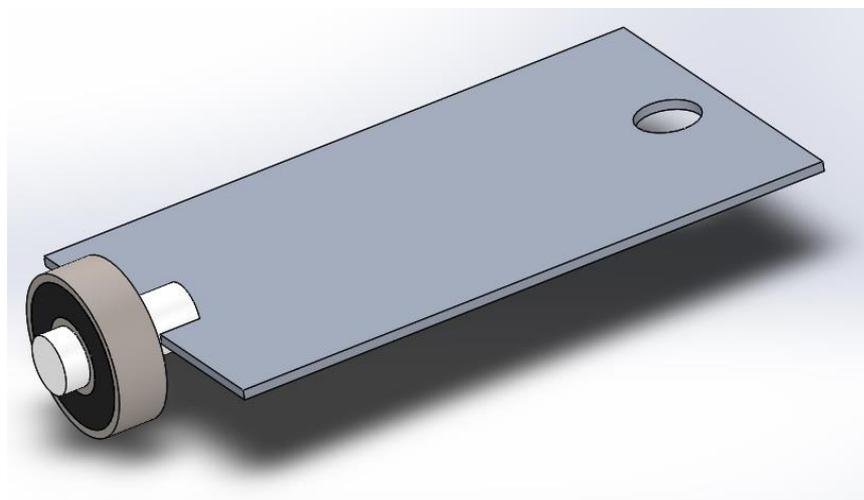


Figura 3.13 – Projeto da chapa com o rolamento.

O eixo do rolamento possui 7 mm de diâmetro, 17 mm de comprimento e um rasgo central para o encaixe da chapa com 5 mm de profundidade e 1,20 mm de espessura. Para a construção destes eixos, utilizou-se uma impressora 3D, empregando ABS como sua matéria prima.

Para a obtenção das chapas, foram utilizadas carcaças de fontes de computadores, de alumínio, que possuem exatamente 1,20 mm, mesmo valor da espessura das moedas da primeira família do Real. Para as moedas de R\$0,25 e R\$1,00, estas chapas também foram utilizadas, visto que a espessura menor não afeta a eficácia do fornecimento de troco.

Como os servomotores têm uma altura entre seus pontos de fixação e a ponta do eixo, foi necessário desenvolver uma base com uma face inclinada, a fim de facilitar a dispensa das moedas para o usuário e de que os motores fossem fixados na base da máquina. Tomando como referência os servos do tópico 3.1.4, a altura do eixo ao ponto de fixação é 16,35 mm. Assim, a base passou a possuir esta espessura, menos 0,2 mm para evitar o contato entre a chapa e a base, ou seja, sua altura passou a ser 16,15 mm.

Já para o deslocamento dos rolamentos, a princípio, verificou-se a disponibilidade no mercado. Os menores encontrados foram do tipo 607, os quais possuem um diâmetro externo de 19 mm e interno de 6mm, com uma largura de 6mm. Então, foi projetado um rasgo na base com profundidade de 8,7 mm.

Para definir essa dimensão, foi considerado o raio do rolamento menos a metade da espessura da chapa mais a sua distância com a base. Ou seja:

$$Espessura = 9,5 - (1,20/2) - 0,2 = 8,70 \text{ mm} \quad (3.4)$$

Para o ângulo de dispensa do troco, foram feitos testes com as moedas. Como as bases foram impressas em ABS, utilizando uma impressora 3D, visou-se reduzir material. Assim, o ângulo projetado foi de 45°. Assim, a base para o troco tomou a forma que pode ser vista na Figura 3.14.

Suas dimensões são: 58,15 mm de comprimento, 60 mm de largura. O local onde ficará os tubos mede (42 mm x 42 mm) e a largura do rasgo é de 18 mm.

Na parte inferior, foram feitos 4 furos para encaixar porcas sextavadas de 7 mm com parafusos de 3 mm, para afixar as bases do troco na base da máquina. O projeto da peça com todas as suas dimensões pode ser visto no Apêndice B.

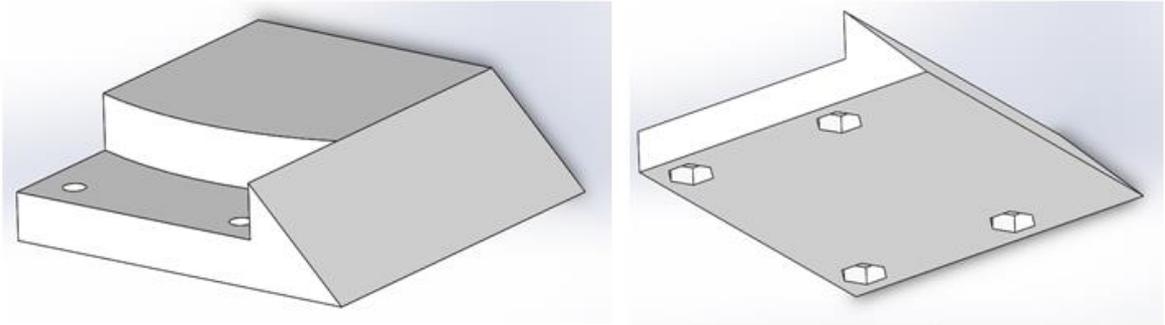


Figura 3.14 – Base para dispensa de moedas.

Em seguida, foi projetado um funil para ampliar a área de entrada das moedas, quando saem do separador. Este funil também serve para direcionar o troco para a saída, juntamente com o limitador, que será explicado adiante. Como seu único objetivo é ampliar as áreas de queda das moedas, pensou-se em desenvolvê-lo com a menor altura possível, também a fim de evitar que o trilho do separador ficasse muito alto da base. Além disso, como os funis seriam impressos em ABS, pensou-se na economia de matéria prima. Assim, chegou-se a uma altura total de 25 mm, onde o funil propriamente dito possui a metade dessa dimensão, e com um ângulo de abertura de 135°. A forma final do desenho pode ser visto a seguir, na Figura 3.15.

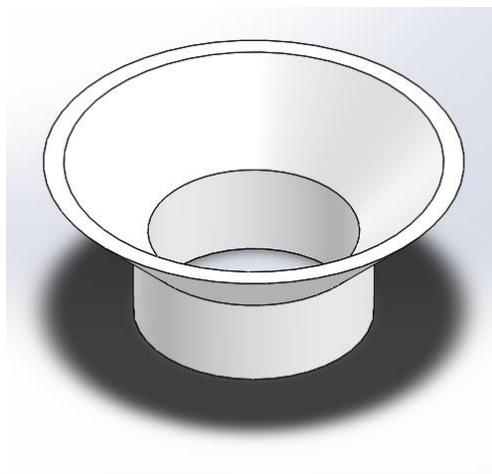


Figura 3.15 – Funil para a entrada das moedas nos tubos e para liberação do troco.

Ao final, para a montagem da máquina no *software*, as peças mostradas nas Figuras 3.13, 3.14 e 3.15 foram unidas, e o sistema de fornecimento de troco tomou a forma que pode ser vista na Figura 3.16.

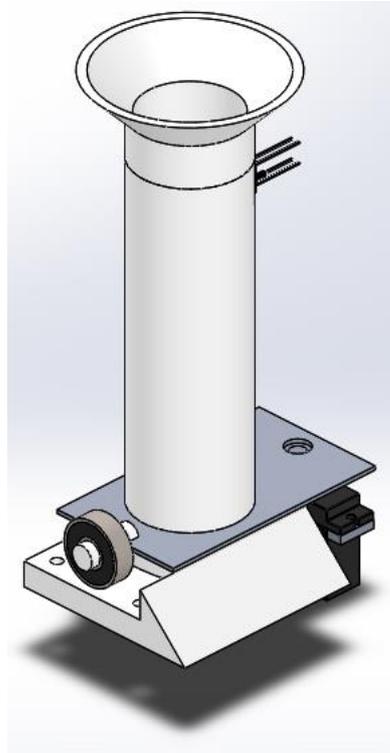


Figura 3.16 – Sistema de fornecimento de troco para um valor de moeda.

Para sustentar os tubos do sistema de troco, foi projetado um suporte. A ideia era que os 5 sistemas de troco, correspondentes a cada valor de moeda, ficassem espaçados em um ângulo de 180° . Cada um ficou então a 36° do outro. O suporte foi desenhado no *software* no formato de um semicírculo, com um raio de 205 mm. A largura da estrutura foi de 10 mm no corpo e 6 mm no acoplamento com os tubos.

Como este suporte fica acima da base, foi necessário projetar seus apoios, acoplados à base. A altura escolhida foi de 70 mm. Sendo assim, o apoio ficou com 82 mm de altura, 88 mm de comprimento do braço e 18 mm de largura.

O material escolhido para a construção de ambos foi MDF de 6 mm de espessura. A montagem do suporte dos tubos do sistema de troco tomou a forma que pode ser vista na Figura 3.17.

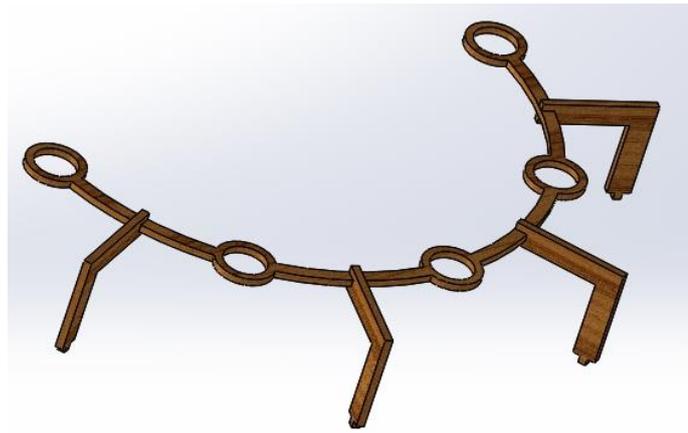


Figura 3.17 – Suporte do sistema de troco.

Com o sistema de fornecimento de troco finalizado, iniciou-se o desenvolvimento do separador de moedas. A ideia do separador era que o usuário depositasse uma moeda por vez, verticalmente. Esta entraria em uma rampa com um barramento, feito por um braço de servomotor. Uma câmera seria a responsável pela identificação da moeda, e ficaria posicionada logo acima deste bloqueio. Assim que fosse identificada, a rampa giraria para o tubo correspondente ao valor verificado pela câmera, o barramento ficaria aberto e a moeda seria liberada.

Através desta ideia, primeiramente foram realizados testes para definir o ângulo da rampa, utilizando um protótipo do trilho do separador produzido na impressora 3D. Assim, chegou-se à um ângulo de 30° . Então, foram desenvolvidas no SolidWorks as peças para formar o separador. A primeira delas foi a peça responsável por inclinar a rampa. Com o ângulo determinado, projetou-se sem forma de triângulo retângulo esta peça, com dois alongamentos nos cantos do cateto adjacente aos 30° e da hipotenusa, para facilitar na montagem, dando encaixe às peças, como um quebra-cabeça. A dimensão do cateto adjacente foi determinada em 40 mm, com um ângulo de 30° , fazendo que o cateto oposto ficasse com 27,09 mm e a hipotenusa com 46,19 mm.

Em seguida, foi projetada a base de apoio do separador com o servomotor responsável por girar a rampa. Foi desenvolvida em forma de retângulo, com um furo no centro para se ter acesso ao parafuso de encaixe da haste do servo, e com 2 rasgos para que a peça projetada se encaixasse nele. Foi dimensionado em 40 mm x 32 mm, levando em consideração a haste do motor, que tem 31,83 mm. A seguir, na Figura 3.18, é possível ver estas duas peças encaixadas na

montagem. O material escolhido para serem fabricadas foi o acrílico de 4 mm, para evitar flambagem e dar maior resistência ao separador. Já as vistas em 2D das peças mencionadas, com suas cotas, podem ser vistas no Apêndice B.

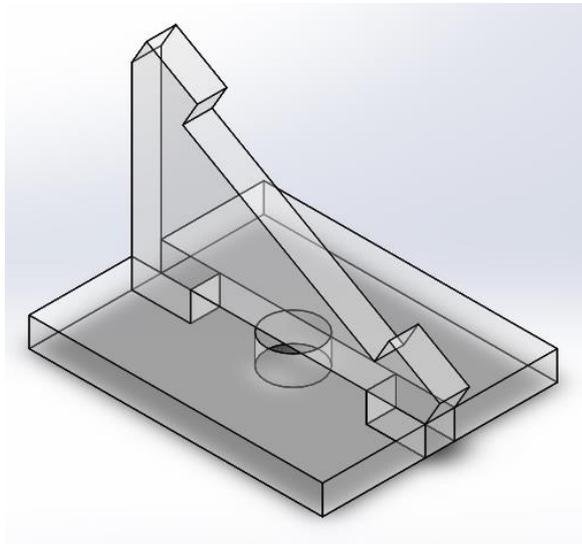


Figura 3.18 – Apoio do servomotor do separador e suporte inclinado.

Com essa base desenvolvida, deu-se início ao projeto da rampa do separador. Através da montagem no *software* da disposição dos separadores, e simulando os movimentos de troca, determinou-se o comprimento desta peça como sendo de 304,64 mm. Já na largura, como as laterais e esta base seriam em acrílico, e com a espessura escolhida de 4 mm, definiu-se como largura 13 mm. Isso porque, a moeda mais larga, de R\$0,50, tem 2,85 mm de espessura. Ao colocar duas laterais na rampa, o espaço para as moedas percorrerem cai para 5 mm, deixando um conforto no deslocamento do dinheiro.

Foram feitos rasgos para se encaixar as laterais e o apoio projetado anteriormente, fazendo que o eixo de rotação dela seja posicionado em 101,62 mm, de forma a provocar um contrapeso na rampa. Também foi feito outro rasgo para se encaixar o suporte do servo responsável por fazer o bloqueio das moedas.

Após a base ter sido projetada, foi desenvolvido o suporte do bloqueador das moedas. Este foi feito em forma de “C”, para se encaixar o servo e fixá-lo nas laterais do separador. Suas dimensões são 25 mm x 34 mm, no retângulo externo, e o retângulo que faz a peça ficar em forma de “C” ficou com 22,50 mm x 16 mm. Para este último, levou-se em consideração as dimensões do servo, de

22,50 mm x 12 mm. Os outros 4 mm são para o encaixe nas laterais. O resultado do projeto desta peça pode ser visto na Figura 3.19.

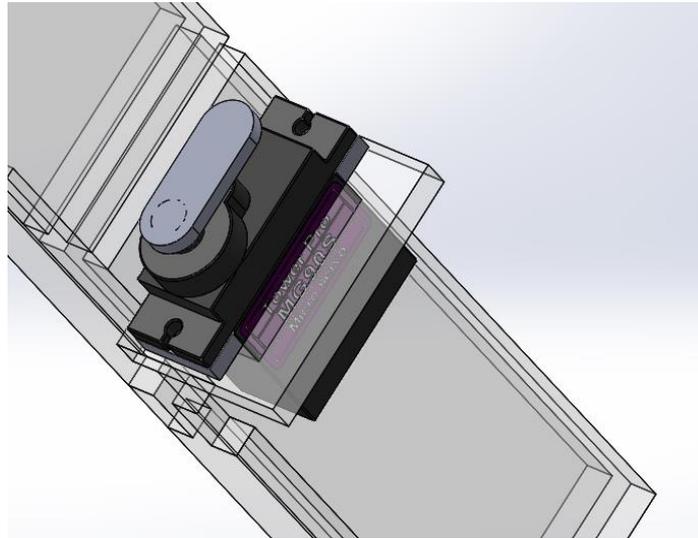


Figura 3.19 – Bloqueador de moedas do separador.

Para finalizar o separador, foram projetadas as laterais. Como a haste do servomotor tem 19 mm, houve a necessidade de se repartir as laterais. A lateral maior foi dimensionada em 196,32 mm por 30 mm de altura, com os alongamentos para encaixe. Já para a menor, suas dimensões foram 104,32 mm por 30 mm. Para esta menor, houve a necessidade de se projetar duas, sendo que uma delas possui o rasgo para encaixar o suporte do servo bloqueador de moedas.

Para suspender o separador e servir de apoio para o servomotor responsável por girá-lo, foi desenvolvido um suporte, em MDF de 6mm. Analisando a montagem no SolidWorks, considerando a altura dos tubos do sistema de troco e a diferença na altura da ponta do separador até o eixo, gerada pelo ângulo de 30°, chegou-se a uma altura de 217,67 mm, com uma largura de 42 mm. Foram feitos também alongamentos para encaixes. Além disso, foi feito o suporte para o servomotor do separador, para se encaixar em cima do apoio citado anteriormente. Esse foi dimensionado tendo 42 mm por 30 mm, e com um furo no centro do tamanho do motor.

Assim, as peças foram unidas em uma montagem, e se finalizou o desenvolvimento do separador, tomando a forma que pode ser vista na Figura 3.20.

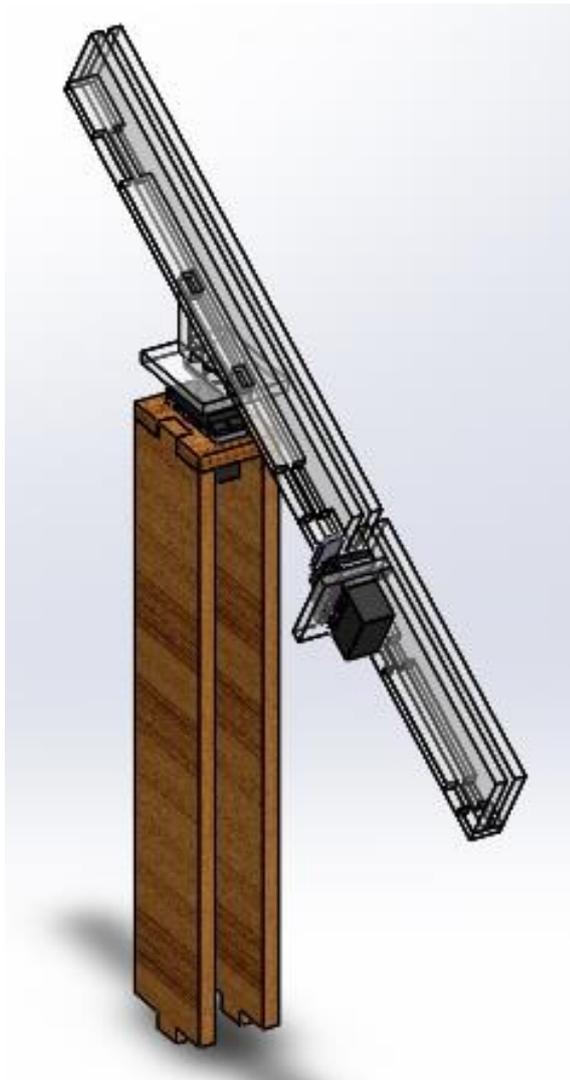


Figura 3.20 – Sistema de separação de moedas completo.

O próximo passo foi a implementação de uma central de comando, responsável por abrigar o teclado matricial e o *display* LCD. Foi obtida a dimensão de ambos. Assim, chegou-se a um valor de 200 mm por 100 mm para o painel frontal, com rasgos para encaixe, e dois furos, sendo um deles para o *display* (71,75 mm x 17 mm) e outro para se passar o cabo flat do teclado (22 mm x 3 mm).

Em seguida, determinou-se que este painel deveria ser inclinado, de forma a dar mais conforto ao usuário, seja para digitar ou visualizar as informações. Assim, convencionou-se em 15° o ângulo de inclinação. Então, foi projetada uma lateral com esse ângulo, e uma tampa traseira, para dar acabamento.

O resultado final do sistema de comando pode ser visto na Figura 3.21.



3.21 – Sistema de comando da máquina.

Com os sistemas de troco, separação e comando desenvolvidos, deu-se início ao projeto da base intermediária da máquina, para sustentar todos estes componentes. Através dessas montagens de teste realizadas no SolidWorks, chegou-se a dimensão de 750 mm por 450 mm para a base. Foram colocados rasgos para o encaixe do restante da carcaça da máquina, furos para o suporte do sistema de troco, separador de moedas, sistema de comando, e parafusos de apoio com a base inferior. Além disso, foram feitos furos para acoplar os funis na saída do sistema de troco de cada moeda.

Assim, a base intermediária da máquina tomou a seguinte forma, vista na Figura 3.22, a seguir.



Figura 3.22 – Base superior da máquina.

Já a base inferior foi projetada para ter a mesma dimensão da base intermediária, mas com furos para o encaixe da rampa, da placa do Arduino e da fonte de computador, utilizada para alimentar os servos e os sensores. A rampa para dispensa do troco foi projetada na forma de um trapézio, com um furo ao centro em forma de meio círculo somado a um retângulo, para que se pudesse passar parafusos de sustentação. Suas dimensões são: 290 mm lado menor, 542 mm lado maior, e 322,40 de laterais. Já o furo mede 228,38 mm de comprimento e 80 mm de largura.

O ângulo da rampa foi obtido de forma experimental, chegando ao valor de 30°. Assim, foi projetado um apoio inclinado para a rampa, com as dimensões: 257 mm de cateto adjacente, 148,38 mm de cateto oposto, e 296,76 mm de hipotenusa.

Além disso, na rampa foram colocadas laterais, a fim de que as moedas não caiam fora dela, e um bloqueio ao seu final, para que as moedas não saiam da máquina de forma indesejada. Assim, a estrutura inferior da máquina tomou a forma que pode ser vista na Figura 3.23, a seguir.

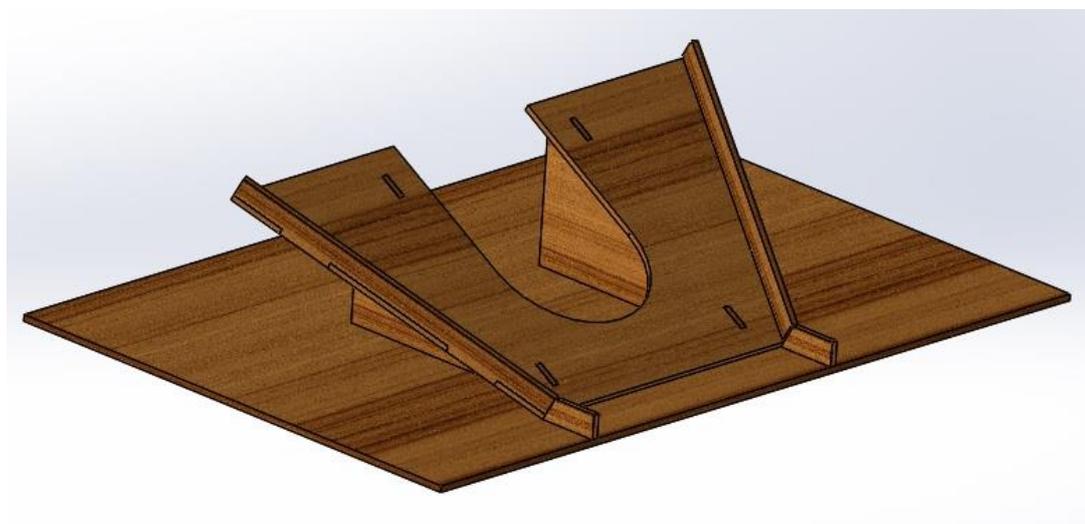


Figura 3.23 – Estrutura inferior da máquina.

Para dar acabamento à máquina, foi projetada uma carcaça, também para servir como a entrada de moedas para o sistema de separação e para acoplar a câmera. As peças separadas e suas dimensões podem ser conferidas no Apêndice B, e a forma da carcaça montada pode ser conferida na Figura 3.24, a seguir.

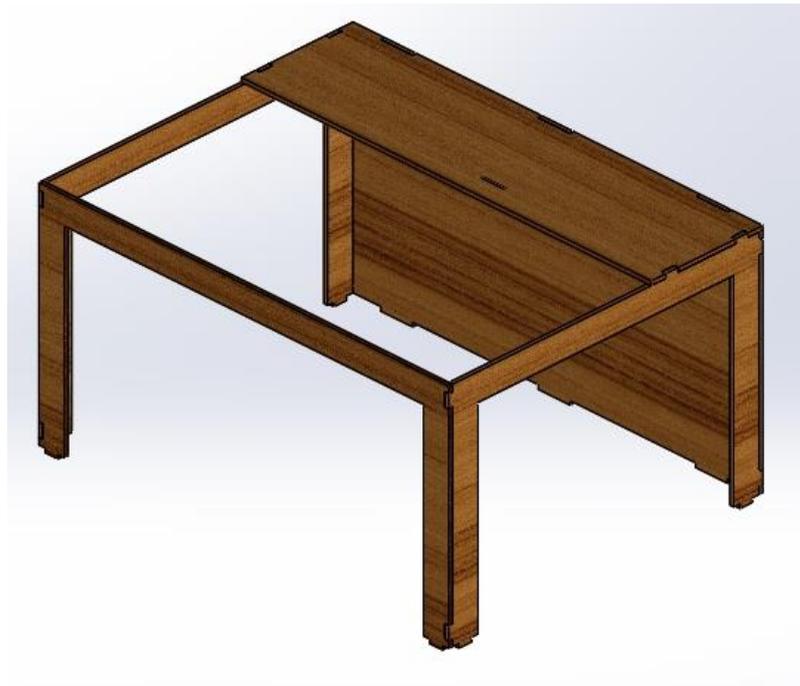


Figura 3.24 – Carcaça da máquina.

Para finalizar o projeto no SolidWorks, foi desenvolvido um limitador para ser acoplado ao lado do sistema de troca, a fim de evitar que as moedas caíssem fora do funil. Além disso, foi feito também um suporte para apoiar a fonte de computador. As dimensões de ambos estão no Apêndice B. O posicionamento do limitador e o suporte podem ser vistos nas Figuras 3.25 e 3.26, respectivamente.

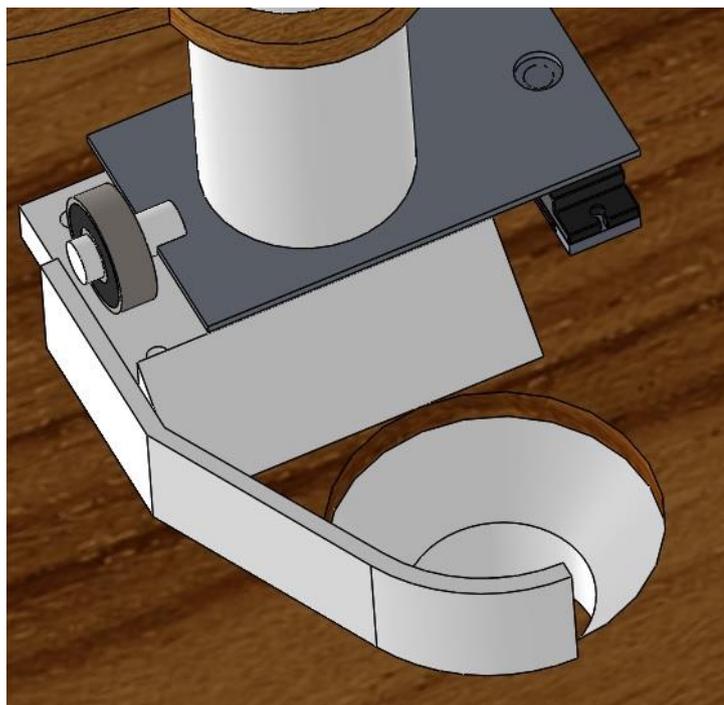


Figura 3.25 – Limitador acoplado ao sistema de troca.

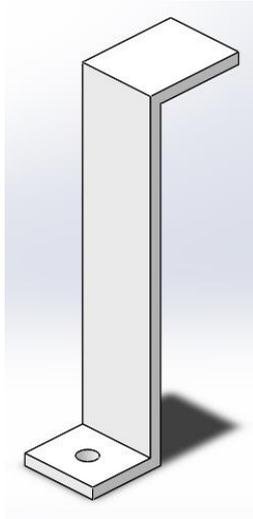


Figura 3.26 – Suporte para apoio da fonte de computador.

A próxima etapa do projeto foi a usinagem das peças e a montagem da estrutura mecânica da máquina. Como foi citado anteriormente, algumas peças foram fabricadas em MDF de 6 mm, acrílico de 4 mm, e impressos na impressora 3D. A seguir, segue a relação de peças com materiais utilizados como suas matérias primas:

ABS:

- Base para dispensa de moedas;
- Funil para a entrada das moedas nos tubos e da dispensa de moedas;
- Limitador de moedas do sistema de troco;
- Suporte para apoio da fonte de computador.

Acrílico de 4 mm de espessura

- Apoio do servomotor do separador;
- Suporte inclinado do separador;
- Laterais e rampa do separador;
- Suporte do servomotor que bloqueia as moedas.

MDF de 6 mm de espessura:

- Suporte do sistema de troco;
- Suporte do separador de moedas;
- Sistema de comando da máquina;
- Base superior da máquina;
- Estrutura inferior da máquina;
- Carcaça total da máquina.

A chapa de liberar o troco, onde se situa o rolamento, foi usinada a partir da carcaça de uma fonte de computador, de alumínio, e com espessura de 1,20 mm.

Terminado o algoritmo de troco, o desenvolvimento do projeto e a construção mecânica, deu-se início a implementação eletrônica. Utilizou-se o *software* Proteus 8.1, versão gratuita, para desenvolver um circuito conforme a Figura 3.12, mostrada anteriormente. Este esquemático pode ser visto no Apêndice C. As distâncias entre as portas do Arduino Mega foram medidas, para se montar a placa em forma de *shield*. Os componentes do esquemático foram posicionados, formando o *layout* da placa, e ao final foi gerado as trilhas do circuito, a ser impresso e passado para a placa, que pode ser visto na Figura 3.27.

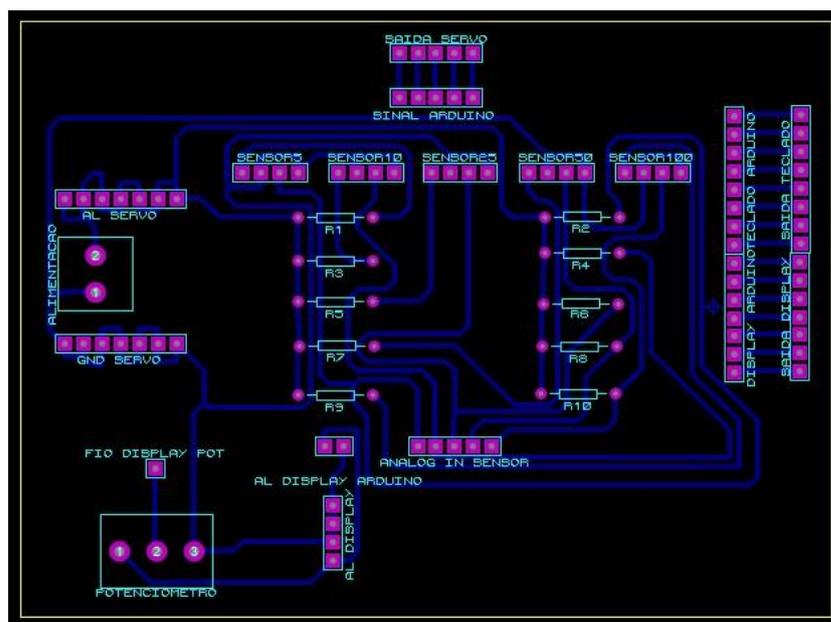


Figura 3.27 – Layout da placa eletrônica.

O layout pronto foi impresso em uma folha de papel couche, e este transferido por aquecimento para uma placa de fenolite. A placa foi corroída com percloroeto de ferro, deixando somente as trilhas necessárias para a placa. Os componentes foram soldados à placa e os fios ligados aos motores, *display*, teclado e sensores, finalizando a parte eletrônica.

Os resistores utilizados foram de $100\ \Omega$ para o LED infravermelho e $10\ \text{k}\Omega$ para o foto transistor. De acordo com o *datasheet* do fabricante do sensor, o sensor deve ser ligado da seguinte maneira, como é mostrado na Figura 3.28.

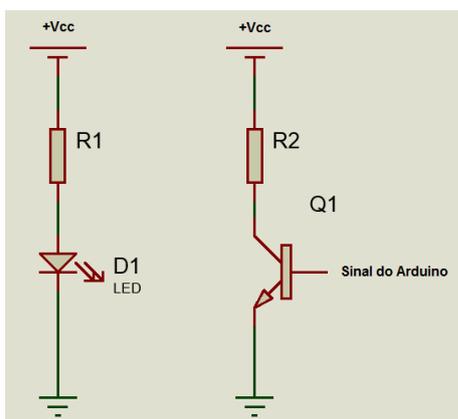


Figura 3.28 – Esquema de ligação para o sensor óptico.

Para determinar o valor de R1, ligado ao LED infravermelho, consultou-se seus gráficos no *datasheet*. Já que se deseja uma maior capacidade de luminosidade, para tornar o sensor mais sensível e preciso, deve-se passar mais corrente no LED. E como sua condução se dá a partir de $0,7\ \text{V}$, foi necessário observar no gráfico de polarização direta, que pode ser visto na Figura 3.29.

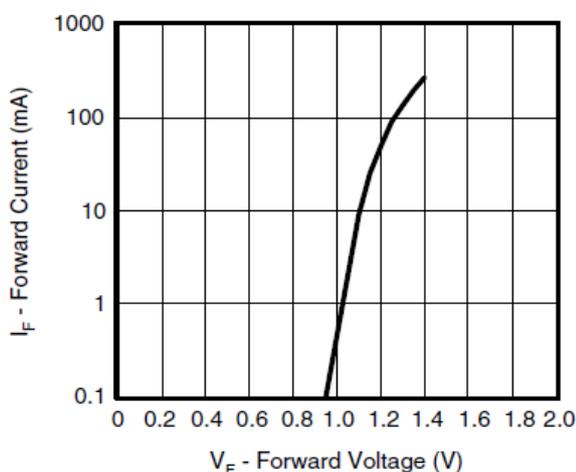


Figura 3.29 – Polarização direta do LED infravermelho do sensor (VISHAY, 2015).

Para uma tensão de aproximadamente 1,1 V, sua corrente é de aproximadamente 40 mA. Assim, determinou-se o valor do resistor pela Equação 3.5. Dado que a tensão de alimentação, $V_{cc} = 5$ V, e para a tensão do LED, $V_{led} = 1,1$ V, a tensão no resistor será $V_R = 3,9$ V. como se escolheu uma corrente $I_{led} = 40$ mA, valor máximo fornecido pelo Arduino, a resistência será:

$$R = \frac{V_R}{I_R} \rightarrow R = \frac{3,9}{0,04} \quad (3.5)$$

$$R = 97,5 \Omega$$

Pesquisando um valor comercial, chegou-se ao valor de 100 Ω .

Como o sensor foi ligado às portas analógicas do Arduino, estas possuem uma resistência interna de 10 M Ω . A sua entrada, que varia de 0 à 5 V é interpretada como um sinal de 0 à 1023, respectivamente. Quando o foto transistor não estiver conduzindo, e para evitar ruídos, a tensão V_{cc} deverá ser quase toda consumida pela resistência interna da placa. Assim, foi colocado o resistor R2 agindo como *pull up*, ou seja, ele serve para desviar a corrente quando o transistor não conduzir.

Assim, escolheu-se um resistor de 10 K Ω e foi calculado o valor da tensão por um divisor de tensão entre R2 e R_{int} , conforme mostrado na Equação 3.6.

$$V_{int} = \frac{R_{int}}{R_2 + R_{int}} \rightarrow V_{int} = 5 \times \frac{10000000}{10000 + 10000000} \quad (3.6)$$

$$V_{int} = 4,95 V$$

Ou seja, o resistor escolhido consegue drenar a tensão para a resistência interna do Arduino quando o foto transistor não está conduzindo. O momento da não-condução irá ocorrer sempre que se passar uma moeda pelo sensor. A variação da resposta analógica (os valores de 0 à 1023), servirá para contar a moeda, acrescentando seu valor à máquina.

Com a máquina construída e a parte eletrônica finalizada, foram feitos testes para determinar a eficiência do sistema, e verificou-se possíveis melhoras.

Todas as etapas citadas ao decorrer desta seção, seguiram o cronograma visto na Tabela 3.2, a seguir. A única etapa que não se conseguiu cumprir foi a do

tópico 9, que é o desenvolvimento do algoritmo do separador de moedas. Isso porque teve-se problema no código do sistema de troco e na comunicação do teclado matricial com o que era visualizado no *display*. Além disso, o projeto estrutural foi totalmente refeito, se comparado com o anterior. Isso demandou um tempo além do planejado, causando um atraso na execução do trabalho. Ambos modelos serão mostrados na seção de Resultados e Discussões.

Tabela 3.2 - Cronograma de trabalho para o desenvolvimento do Trabalho de Conclusão de Curso.

Etapas	Mês								
	Mar	abr	mai	jun	jul	ago	set	out	nov
1 – Revisão da literatura.	X	X							
2 – Elaboração do projeto: simulações e lista de materiais.		X	X						
3 – Realização de orçamentos.			X						
4 – Compra de materiais.			X	X					
5 – Desenvolvimento do algoritmo do sistema de troco					X	X			
6 - Realização de teste do algoritmo com os componentes do sistema						X	X		
7 - Fabricação e usinagem das peças							X		
8 - Montagem do protótipo							X	X	
9 – Desenvolvimento do algoritmo do separador de moedas									X
10 - Realização de testes, análise e discussão dos resultados								X	X
11 - Produção literária									X
12 - Revisão final e preparação para a apresentação do trabalho									X

Quanto aos custos da máquina, é possível verificar na Tabela 3.3, a seguir, os preços detalhados de cada componente que a compõe. No total, o projeto teve um gasto de R\$821,08.

Tabela 3.3 – Custos detalhados do projeto da máquina.

Componentes	Preço unitário (reais)	Quantidade	Total
Sensor ótico TCRT5000 (10 pçs)	3,6	2	7,2
Teclado matricial 4x4	20,65	1	20,65
Display LCD 16x2	18,9	1	18,9
Barra de 40 pinos Header x5 unidades	9,9	1	9,9
Jumpers macho-macho x65	19,9	1	19,9
Frete (Display+Barra+jumpers)	21,08	1	21,08
Arduino Uno R3	20,81	1	20,81
Arduino Mega 2560	30,62	1	30,62
Sensor TCRT5000 (10 pçs) - Mercado Livre	9,9	1	9,9
Frete (Sensor)	15,55	1	15,55
Rolamentos 7mm(Dint) x 19mm(Dext)	8	3	24
500g de ABS 3mm	60	1	60
Frete da Movtech para o ABS ganhado	21	1	21
Parafusos 3mm, porcas e arruelas	3,8	1	3,8
Servo motor Mg90s	20,99	3	62,97
Frete dos 3 servos	19,38	1	19,38
Servo motor Mg90s	21,99	4	87,96
Frete dos 4 servos	24,88	1	24,88
Corte à laser da estrutura em mdf e acrílico	124,5	1	124,5
Placa para circuito impresso	4,8	1	4,8
Pasta para solda	5,8	1	5,8
Cabo flat + frete	22,32	1	22,32
Corte do restante da estrutura	85	1	85
Borne 2 pinos	2,2	1	2,2
1 Barra roscada 7/16" + 24 arruelas + 24 porcas	17,66	1	17,66
Rolamento 607 zz	3	2	6
Webcam	35	1	35
Termo retrátil	1	1,8	1,8
1 Barra roscada 7/16" + 16 arruelas + 16 porcas	15,5	1	15,5
20 m de fios para soldar os sensores	0,4	20	8
Lata de tinta preto fosco	14	1	14
Lata de tinta preto fosco	13	2	26
Total do projeto:			821,08

4. RESULTADOS E DISCUSSÕES

Neste capítulo, serão apresentados os resultados obtidos com as etapas descritas no Capítulo 3.

A princípio, com o algoritmo pronto, deu-se início aos testes para realizar alguns ajustes. Ao verificar o programa sem os sensores, mas com o *display*, teclado e servomotores acoplados ao Arduino, obteve-se êxito em seu funcionamento. Isso porque ainda não se tinha a montagem da máquina, e como os LEDs dos sensores iriam ser colocados um de frente para o outro, sua lógica ficaria invertida ou seja, seria necessário um alinhado de frente ao outro para que o contador ficasse em 0. Caso contrário ele dispararia. Isso foi verificado em uma avaliação inicial, apenas com o sensor, através de um algoritmo para teste. Foi testado um sensor com a carcaça e outro sem.

Então, foi colocado no programa como se a máquina possuísse 10 moedas de cada valor, para o teste, gerando um montante de R\$19,00, visto que esse valor seria obtido com as leituras dos sensores.

Ao acoplar todos os sensores e ligá-los juntamente com o restante dos componentes, viu-se a necessidade de uma fonte de alimentação externa. Isso porque, como visto no *datasheet* do Arduino, sua corrente máxima é de 40 mA. Assim, se calculou a corrente necessária para alimentar todos os sensores, através de um divisor de corrente, visto que todos estão ligados em paralelo. Os cálculos foram realizados com a Equação 4.1, a seguir.

$$i_1 = \frac{\frac{1}{R1}}{\left(\frac{1}{R1} + \frac{1}{R2} + \frac{1}{R3} + \frac{1}{R4} + \frac{1}{R5}\right)} \times i \quad (4.1)$$
$$0,04 = \frac{\frac{1}{100}}{\left(\frac{1}{100} + \frac{1}{100} + \frac{1}{100} + \frac{1}{100} + \frac{1}{100}\right)} \times i$$
$$i = 0,2 A \rightarrow 200 mA$$

Como a corrente foi bem mais alta que a da alimentação do Arduino, resolveu-se utilizar uma fonte de computador, que tem a tensão de entrada de 230V ou 115 V (alternada, da rede), e como tensões de saída 12 V e 5 V

(contínua). A corrente em 5 V é de até 35 A. Assim, decidiu-se alimentar também os servomotores com essa fonte.

Os terras da fonte e do Arduino foram interconectados, para garantir que ficassem no mesmo potencial. Com a fonte alimentando os servos e os sensores, o Arduino ficou responsável apenas por alimentar o *display*. Então, fez-se novamente os testes no algoritmo do sistema de troco. A função Standby, responsável por mostrar o menu inicial da máquina, trazendo instruções de uso, ficou com as seguintes telas, vistas na Figura 4.1.



Figura 4.1 – Menu inicial da máquina.

Como indicado na Figura 4.1, ao se apertar a tecla “A” do teclado matricial, é exibido o montante. Esta tela ficou com a forma que pode ser vista na Figura 4.2. Este modo fica sendo exibido enquanto o usuário não apertar outra tecla alfabética, ou seja, muda de modo apenas com as teclas “A”, “B”, “C” e “D”.



Figura 4.2 – Exibição do montante na máquina.

Outra informação que pode ser vista da máquina é a quantidade de moedas geral e de cada valor, ao apertar a tecla “B”. A cada 3 s, a informação é mudada, e fica neste modo até que o usuário aperte outra tecla alfabética. As telas informativas podem ser vistas na Figura 4.3, a seguir.



Figura 4.3 – Quantidade de moedas geral e específica na máquina.

Já ao apertar a tecla “C”, a máquina entra no modo de troco. Este é o único modo em que o teclado numérico responde, conforme dito na seção 3.2. A tecla pressionada é instantaneamente visualizada na tela. Caso não se pressione nenhum número, a tela fica como pode ser visto na Figura 4.4. Esta tela também permanece em exibição enquanto não se apertar nenhuma outra tecla de opções.



Figura 4.4 – Tela para o usuário digitar e conferir o troco digitado.

Ao confirmar o troco digitado, os servomotores trabalham com um *delay* de 0,5 s, para cada operação de abrir e de fechar o tubo. O troco é liberado do maior para o menor valor, uma moeda por vez. A função *delay* impede que o usuário mude de tela ou aperte outro valor enquanto as moedas estão sendo dispensadas. Este modo da máquina pode ser visto na Figura 4.5, a seguir.



Figura 4.5 – Tela exibida ao confirmar o troco.

Caso o troco digitado seja maior que o valor contido na máquina, uma mensagem de erro é exibida, conforme é mostrado na Figura 4.6.



Figura 4.6 – Tela exibida quando se digita um troco maior do que o montante da máquina.

Com o programa funcionando de forma adequada, o projeto das peças que constituem a máquina foi feito. Os cálculos para determinar se o tubo com 90 mm seria suficiente, conforme foi descrito no tópico 3.2 foram realizados em uma planilha do Microsoft Excel. Foram obtidos os resultados apresentados nas Tabelas 4.1 e 4.2, para a primeira e segunda família do Real, respectivamente.

Tabela 4.1 – Resultados dos cálculos para a primeira família do Real.

Moedas (R\$)	Quantidade de moedas em 90mm	Montante	Peso total (g)
0,01	75	0,75	222
0,05	75	3,75	245,25
0,1	75	7,5	269,25
0,25	64	16	305,92
0,5	75	37,5	294
1	0	0	0
Total	364	65,5	1336,42

É possível notar que para as moedas de R\$1,00 os valores foram 0. Isso porque elas não estão mais em circulação.

Tabela 4.2 – Resultados dos cálculos para a segunda família do Real.

Moedas (R\$)	Quantidade de moedas em 90mm	Montante	Peso total (g)
0,01	54	0,54	131,22
0,05	54	2,7	221,4
0,1	40	4	192
0,25	40	10	302
0,5	31	15,5	242,11
1	46	46	322
Total	265	78,74	1410,73

Na segunda família, as moedas de R\$0,50 e R\$1,00, possuíram outros pesos entre 1998 e 2001, devido a composição de seus materiais. O peso das duas, para a quantidade de moedas obtidas da tabela são 286,75 g e 360,64 g, respectivamente. Os valores calculados na Tabela 3.3 são para as moedas que foram fabricadas de 2002 em diante.

Já o montante total foi de R\$65,50 para a primeira família e R\$78,74 para a segunda.

Ao final, a montagem dessas peças foi realizada no SolidWorks, que pode ser conferida na Figuras 4.7.

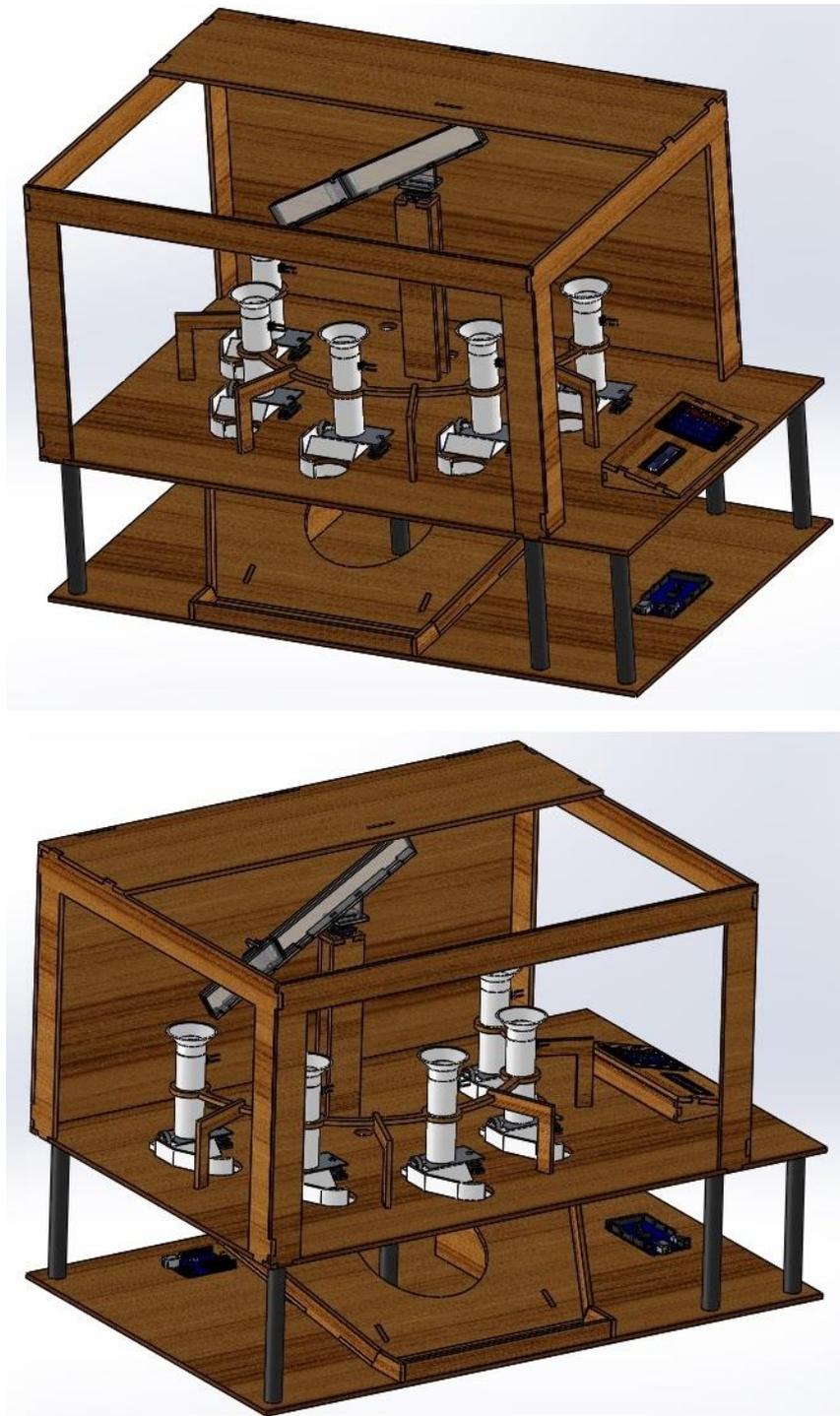


Figura 4.7 – Máquina de separar moedas para fornece troco. Sistema completo.

Como foi dito ao final do Tópico 3.2, gastou-se muito tempo para resolver os problemas de funcionamento do código do sistema de troco e também por

refazer o projeto mecânico. Ao comparar com o projeto inicial, que pode ser visto na Figura 4.8, é fácil notar que houve um replanejamento da máquina, o que acarretou em um atraso no cronograma e impedindo o desenvolvimento do algoritmo da visão computacional, responsável por separar as moedas.

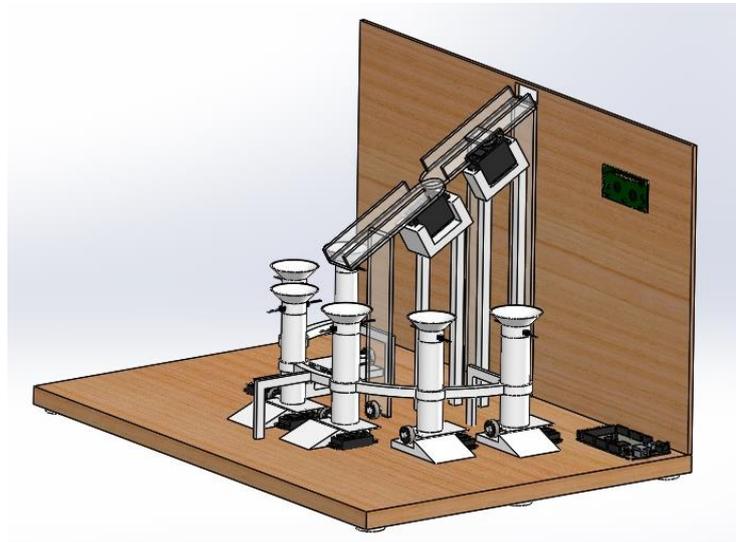


Figura 4.8 – Projeto inicial da máquina, que foi aprimorado.

Após o projeto no *software* ter sido concluído, deu-se início na usinagem das peças e construção da máquina. O resultado da construção pode ser visto nas Figuras 4.9, 4.10 e 4.11, a seguir.



Figura 4.9 – Vista frontal da máquina.

A Figura 4.10 exibe o sistema de troca com os sensores e o separador de moedas.

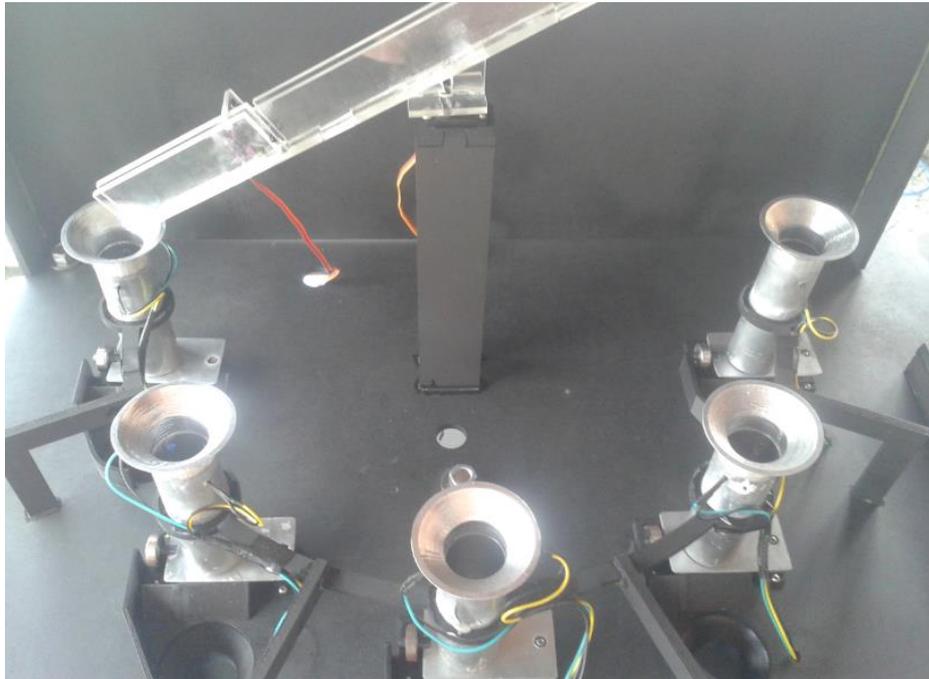


Figura 4.10 – Sistemas de separação de moedas e de troco.

Já a Figura 4.11 exibe a vista lateral da máquina, podendo ser visualizada a central de comando e o *shield* do Arduino.

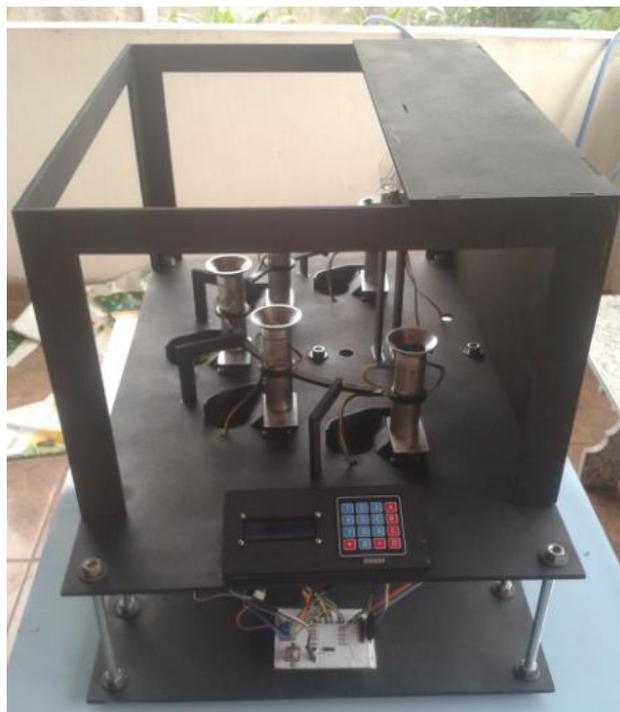


Figura 4.11 – Vista lateral da máquina.

Para finalizar o processo de construção da máquina, a placa de circuito impresso foi construída e ficou conforme pode ser visto na Figura 4.12.

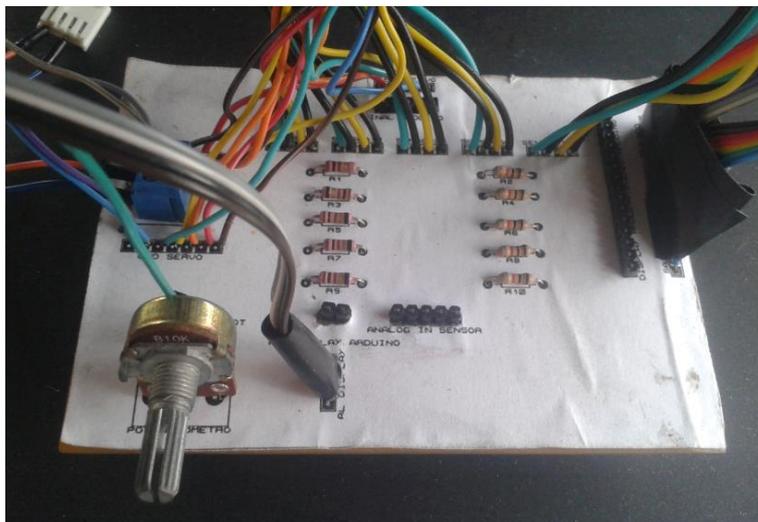


Figura 4.12 – *Shield* com os componentes montados e conectado ao Arduino.

Quanto ao layout da placa, com os pontos de solda, é exibida na Figura 4.13, a seguir. As trilhas e a borda foi estanhada para evitar que o cobre enferrujasse.

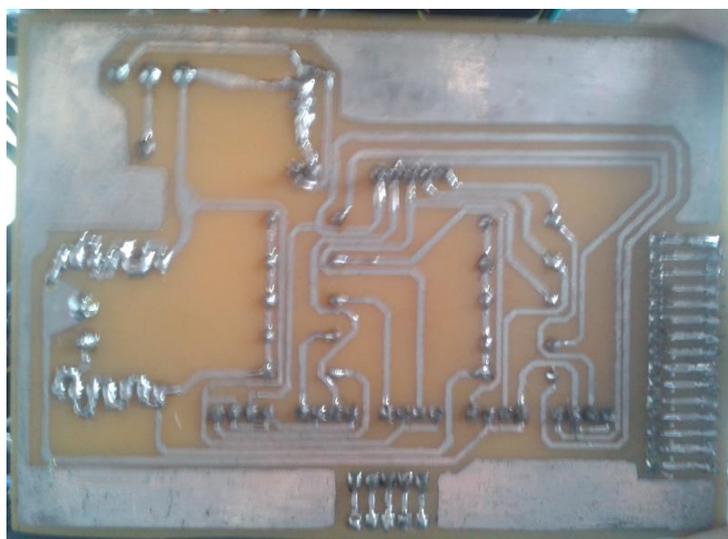


Figura 4.13 – *Layout* da placa de circuito impresso, com os componentes soldados.

Com a máquina montada, fez-se os testes de eficiência. Estes tiveram como finalidade verificar a capacidade do sensor em contar moedas, a eficácia na liberação das moedas e do fornecimento de trocos. Para a realização destes

experimentos, foram utilizadas 10 moedas de cada família e de cada valor, totalizando 90 moedas. Como as moedas de R\$1,00 da primeira geração do real não estão mais em circulação, usou-se apenas a moeda corrente atual.

O primeiro teste a ser realizado foi o de contagem de moedas para a primeira família. As 10 moedas de cada valor (com exceção às de R\$1,00, como explicado anteriormente) foram depositadas uma à uma, em seus respectivos recipientes. Quando estas passaram pelos sensores, posicionados nos topos dos tubos, seus valores foram contabilizados. Para cada valor de moeda, foi conferida a eficiência da contagem.

O segundo teste foi o de liberação das moedas. Como a máquina estava com um montante, utilizado anteriormente para verificar os sensores, bastou-se digitar o valor total de cada recipiente (R\$0,50, R\$1,00, R\$2,50, R\$5,00, correspondente ao total de moedas de R\$0,05, R\$0,10, R\$0,25, R\$0,50, respectivamente). Ao final de cada dispensa, foi verificada a eficiência da liberação das moedas.

Em seguida, os mesmos testes de eficiência na contagem e liberação de moedas foram feitos para a segunda família do Real.

Os resultados da contabilização das moedas pelos sensores podem ser observados na Tabela 4.3, onde o máximo para cada valor é 10. Já a Tabela 4.4 mostra a quantidade de moedas liberadas.

Tabela 4.3 – Resultados dos testes de contagem das moedas.

Processo de contagem de moedas		
Valores	1ª família	2ª família
R\$ 0,05	6	6
R\$ 0,10	10	9
R\$ 0,25	7	6
R\$ 0,50	7	5
R\$ 1,00	0	10

Tabela 4.4 – Resultados dos testes de liberação das moedas.

Liberação de moedas		
Valores	1ª família	2ª família
R\$ 0,05	10	10
R\$ 0,10	10	8
R\$ 0,25	10	10
R\$ 0,50	9	10
R\$ 1,00	0	10

A seguir, nas Tabelas 4.5 e 4.6, foram demonstradas a eficiência em porcentagem para a contagem e dispensa das moedas, para a primeira e segunda família do Real.

Tabela 4.5 – Resultados da eficiência da leitura dos sensores.

Eficiência na contagem de moedas		
Valores	1ª família	2ª família
R\$ 0,05	60%	60%
R\$ 0,10	100%	90%
R\$ 0,25	70%	60%
R\$ 0,50	70%	50%
R\$ 1,00	0%	100%

Tabela 4.6 – Resultados da eficiência na dispensa de moedas.

Eficiência da liberação de moedas		
Valores	1ª família	2ª família
R\$ 0,05	100%	100%
R\$ 0,10	100%	80%
R\$ 0,25	100%	100%
R\$ 0,50	90%	100%
R\$ 1,00	0%	100%

Por último, foi verificada a eficiência no fornecimento de troco para alguns valores específicos. Estes foram determinados considerando combinações de duas, três, quatro e cinco moedas a serem liberadas para formar os trocos. Assim, testou-se os seguintes valores:

- Combinações de duas moedas: R\$0,35, R\$0,30, R\$0,15;
- Combinações de três moedas: R\$1,75, R\$1,15, R\$0,40;
- Combinações de quatro moedas: R\$1,85, R\$1,65, R\$0,90;
- Combinações de cinco moedas: R\$1,90.

Para cada valor, foram feitos 10 testes de fornecimento de troco para a primeira família e 10 para a segunda. A seguir, na tabela 4.7, é mostrado os resultados obtidos para cada troco.

Tabela 4.7 – Resultados dos testes de fornecimento de troco.

Fornecimento de troco		
Valores	1ª família	2ª família
R\$ 1,90	8	7
R\$ 1,85	7	9
R\$ 1,75	7	9
R\$ 1,65	8	7
R\$ 1,50	8	8
R\$ 1,15	6	9
R\$ 0,90	6	8
R\$ 0,40	9	6
R\$ 0,35	10	7
R\$ 0,15	10	6

Já na Tabela 4.8, a seguir, os resultados obtidos foram demonstrados em porcentagem.

Tabela 4.8 – Eficiência de fornecimento de troco.

Eficiência do fornecimento de troco		
Valores	1ª família	2ª família
R\$ 1,90	80%	70%
R\$ 1,85	70%	90%
R\$ 1,75	70%	90%
R\$ 1,65	80%	70%
R\$ 1,50	80%	80%
R\$ 1,15	60%	90%
R\$ 0,90	60%	80%
R\$ 0,40	90%	60%
R\$ 0,35	100%	70%
R\$ 0,15	100%	60%

5. CONCLUSÕES

Este presente trabalho apresentou o estudo para o desenvolvimento de um projeto de uma máquina que seja capaz de identificar, separar e contar moedas para o fornecimento de troco. O protótipo apresentado é uma implementação de modelos já existentes, mas aplicando a visão computacional como método de separação.

Logo de início, algumas dificuldades foram encontradas. A falta de publicações a respeito de separadores ou sistemas de troco foi a principal. Além disso, perdeu-se muito tempo otimizando o projeto, que comparado com o inicial, nota-se que foi desenvolvido um modelo completamente novo. O atraso se deu também por causa de problemas na programação, na parte de comunicação com teclado e visualização do *display*. O tempo gasto para a solução destes problemas impediu que o algoritmo responsável pela separação das moedas não fosse concluído.

Com o protótipo em funcionamento (apenas o sistema de contagem de moedas e fornecimento de troco), fez-se os testes para determinar a eficiência da máquina em contar, dispensar moedas e fornecer o troco.

Para determinar a eficiência do sensor em realizar a contagem de moedas, foram desenvolvidos cálculos estatísticos de média, com seus desvios padrões, para definir sua eficácia para a primeira e segunda família separadamente. Para a primeira, os cálculos foram conforme é mostrado a seguir:

$$Média_{contagem_fam1} = \frac{0,6 + 1 + 0,7 + 0,7}{4} = 0,75 = 75\%$$

$$Variância_{contagem_fam1} = \frac{(0,6 - 0,75)^2 + (1 - 0,75)^2 + (0,7 - 0,75)^2 + (0,7 - 0,75)^2}{4}$$

$$S^2 = Variância_{contagem_fam1} = 0,0225 = 2,25\%$$

$$Desvio\ padrão_{contagem_fam1} = \sqrt{S} = 0,15 = 15\%$$

Já para a segunda família, os cálculos foram:

$$Média_{contagem_fam2} = \frac{0,6 + 0,9 + 0,6 + 0,5 + 1}{5} = 0,72 = 72\%$$

$$Variância_{contagem_fam2} = \frac{(0,6 - 0,72)^2 + (0,9 - 0,72)^2 + (0,6 - 0,72)^2 + (0,5 - 0,72)^2 + (1 - 0,72)^2}{5}$$

$$S^2 = Variância_{contagem_fam2} = 0,0376 = 3,76\%$$

$$Desvio\ padrão_{contagem_fam2} = \sqrt{S} = 0,1939 = 19,39\%$$

Logo após, estipulou-se uma eficiência média entre as duas famílias, conforme pode ser visto a seguir.

$$Média_{contagem} = \frac{0,75 + 0,72}{2} = 0,735 = 73,5\%$$

$$Variância_{contagem} = \frac{(0,75 - 0,735)^2 + (0,72 - 0,735)^2}{2}$$

$$S^2 = Variância_{contagem} = 2,25 \times 10^{-4} = 0,0225\%$$

$$Desvio\ padrão_{contagem} = \sqrt{S} = 0,015 = 1,5\%$$

Para determinar a eficiência dos dispensadores de moedas da máquina, os cálculos estatísticos realizados foram semelhantes aos anteriores. Para a primeira família, os cálculos foram:

$$Média_{dispensa_fam1} = \frac{1 + 1 + 1 + 0,9}{4} = 0,975 = 97,5\%$$

$$Variância_{dispensa_fam1} = \frac{(1 - 0,975)^2 + (1 - 0,975)^2 + (1 - 0,975)^2 + (0,9 - 0,975)^2}{4}$$

$$S^2 = Variância_{dispensa_fam1} = 1,875 \times 10^{-3} = 0,1875\%$$

$$Desvio\ padrão_{dispensa_fam1} = \sqrt{S} = 0,0433 = 4,33\%$$

Já para a segunda família, os resultados foram:

$$Média_{dispensa_fam2} = \frac{1 + 0,8 + 1 + 1 + 1}{5} = 0,96 = 96\%$$

$$Var_{dispensa_fam2} = \frac{(1 - 0,96)^2 + (0,8 - 0,96)^2 + (1 - 0,96)^2 + (1 - 0,96)^2 + (1 - 0,96)^2}{5}$$

$$S^2 = Variância_{dispensa_fam2} = 6,4 \times 10^{-3} = 0,64\%$$

$$Desvio\ padrão_{dispensa_fam2} = \sqrt{S} = 0,08 = 8\%$$

Em seguida, estipulou-se uma eficiência média entre as duas famílias, conforme pode ser visto a seguir.

$$Média_{dispensa} = \frac{0,975 + 0,96}{2} = 0,9675 = 96,75\%$$

$$Variância_{dispensa} = \frac{(0,975 - 0,9675)^2 + (0,96 - 0,9675)^2}{2}$$

$$S^2 = Variância_{dispensa} = 1,125 \times 10^{-4} = 0,01125\%$$

$$Desvio\ padrão_{dispensa} = \sqrt{S} = 0,0106 = 1,06\%$$

Por último, foram realizados os cálculos para se determinar a eficiência da máquina em fornecer troco. Assim como foram desenvolvidos os cálculos anteriores, utilizou-se o *software* Microsoft Excel para a obtenção das médias e desvios padrões. Chegou-se então aos seguintes valores, mostrados na Tabela 5.1.

Tabela 5.1 – Eficiência média do fornecimento de troco

Eficiência média do fornecimento de troco		
Valores	Família 1	Família 2
Média	79%	76%
Variância:	1,89%	1,24%
Desvio Padrão:	13,75%	11,14%

Onde a eficiência média entre as duas famílias foi de 77,5% ± 1,5%.

Com a realização de todos os cálculos descritos acima, concluiu-se que a máquina possui como eficiência para contagem, dispensa e fornecimento de troco, respectivamente:

- Contagem de moedas da primeira família do Real: $75\% \pm 15\%$;
- Contagem de moedas da segunda família do Real: $72\% \pm 19,39\%$;
- Eficiência média da contagem de moedas: $73,5\% \pm 1,5\%$;

- Dispensa de moedas da primeira família: $97,5\% \pm 4,33\%$;
- Dispensa de moedas da segunda família: $96\% \pm 8\%$;
- Eficiência média da dispensa de moedas: $96,75\% \pm 1,06\%$;

- Fornecimento de troco com moedas da primeira família: $79\% \pm 13,75\%$;
- Fornecimento de troco com moedas da segunda família: $76\% \pm 11,14\%$;
- Eficiência média do fornecimento de troco: $77,5\% \pm 1,5\%$.

Assim, pode-se concluir que a máquina conseguiu funcionar razoavelmente, mas que precisa de ajustes e melhorias, tanto na parte de sensoriamento quanto no fornecimento de troco.

O baixo rendimento na contagem de moedas foi devido ao fato delas entrarem nos tubos de armazenamento com ângulos aleatórios. Talvez, se o sistema de separação estivesse funcionando, como os ângulos de posicionamento da rampa seriam fixos, a eficiência poderia ser um pouco maior.

Já o baixo rendimento no fornecimento de troco foi acarretado pelo diâmetro dos tubos, que são padronizados. Viu-se que, para um melhor desempenho, os recipientes devem possuir um diâmetro um pouco maior ou igual ao das moedas. Isso porque, através dos testes, percebeu-se que algumas moedas de R\$0,05 e R\$0,10 da segunda família (que possuem os menores diâmetros) ficavam desorganizadas dentro dos tubos. Quando o dispensador tentava liberá-las, ou a moeda não saía, por estar em uma posição vertical (deveria estar na horizontal, como todas as outras), ou ficava presa próximo ao rolamento, porque o dispensador não conseguia fazê-la chegar ao funil de dispensa.

Outro fator que atrapalhou no fornecimento de troco foram os funis de dispensa. Viu-se que algumas moedas de R\$0,10, R\$0,25, R\$0,50 caiam fora do local de coleta. Isso porque, entre os funis e a rampa não existe nada. Assim as moedas caiam livremente.

Percebeu-se então que pode-se resolver este problema de duas maneiras. A primeira é acoplar pequenos tubos aos funis, obrigando as moedas a percorrerem um caminho linear até chegar a rampa, que por sua vez leva ao local de coleta. A segunda é a substituição das paredes da rampa por outras maiores.

Mais soluções para os problemas descritos acima serão discutidos no Capítulo 6.

6. PROPOSTAS PARA TRABALHOS FUTUROS

Diante do que foi explicado no Capítulo 5, pretende-se desenvolver completamente o algoritmo de visão computacional responsável por separar as moedas. Este programa deverá ser aplicado à máquina, que no momento funciona somente para contar moedas e fornecer troco.

Além disso, mais sensores ópticos poderão ser colocados nos tubos de contagem de moedas, melhorando a precisão na contagem. Os furos para implementação até já existem, mas para instalá-los, será necessário construir outra placa *shield*, visto que os sensores demandam também de resistores.

Outra modificação a ser realizada é o acoplamento de pequenos tubos nos funis de dispensa das moedas, conforme discutido no Capítulo 5, visto que esse procedimento será mais fácil e de baixo custo, comparado com a troca das laterais da rampa. Isso para se obter uma melhor eficiência no fornecimento do troco.

Pretende-se também retirar a fonte de computador e projetar um conversor AC/DC abaixador de tensão, de forma a obter os 5 V necessários para a alimentação dos sensores e servos.

Por último, pretende-se estudar e desenvolver uma estrutura mais compacta, visto que o modelo apresentado pode ser ainda menor em suas dimensões.

7. REFERÊNCIAS BIBLIOGRÁFICAS

ABVA, Associação Brasileira de Vendas Automáticas. **Histórico e desenvolvimento de vending machines**. Disponível em: <<http://www.abva.com.br/template.php?pagina=historico>>. Acesso em: 30 mar. 2015.

ALMEIDA, Marília. **Máquinas de vendas automáticas devem passar a aceitar cartões no Brasil**. 2013. Disponível em: <<http://economia.ig.com.br/empresas/comercioservicos/2013-06-25/maquinas-de-vendas-automaticas-devem-passar-a-aceitar-cartoes-no-brasil.html>>. Acesso em: 25 jun. 2013.

ARDUINO MEGA. **Arduino Mega 2560**. Disponível em: <<http://arduino.cc/en/Main/ArduinoBoardMega2560>>. Acesso em: 25 mar. 2015.

ARDUINO UNO. **Arduino Uno R3**. Disponível em: <<https://www.arduino.cc/en/Main/ArduinoBoardUno>>. Acesso em: 10 nov. 2015.

BCB. BANCO CENTRAL DO BRASIL. **Características da primeira família do Real**. Disponível em: <<http://www.bcb.gov.br/?MOEDAFAM1>>. Acesso em: 11 abr. 2015.

BCB. BANCO CENTRAL DO BRASIL. **Características da segunda família do Real**. Disponível em: <<http://www.bcb.gov.br/?MOEDAFAM1>>. Acesso em: 11 abr. 2015.

BLOG FILIPEFLOP. Blog Filipe Flop. **Como usar o teclado matricial com o Arduino**. Disponível em: <<http://blog.filipeflop.com/arduino/teclado-matricial-4x4-arduino.html>>. Acesso em: 25 abr. 2015.

Coin Sorter Machine. Disponível em: <<https://drawingthedata.files.wordpress.com/2014/10/1000x1000.jpg>>. Acesso em: 23 mar. 2015

COMPANY, Vending Max. **O que é Vending Machines**. Disponível em: <<https://www.vending-max.com/blog/index.php/o-que-e-vending-machines/>>. Acesso em: 30 mar. 2015.

DAVIES, R. (2005). **Machine Vision: Theory, Algorithms, Practicalities**. 1ed., Vol. 1. Morgan Kaufmann.

DEL TORO, Vincent. **Fundamentos de Máquinas Elétricas**. Rio de Janeiro: LTC, 1999.

FILIPEFLOP. Filipe Flop comércio. (Santa Catarina). **Display LCD 16x2**. Disponível em: <<http://www.filipeflop.com/pd-6b7e4-display-lcd-16x2.html?ct=&p=1&s=1>>. Acesso em: 20 maio 2015.

FORSYTH, D. A. (2003). **Computer Vision, A Modern Approach**. (1 ed., Vol. 1). Prentice Hall.

GONZALEZ, R. C. (2000). **Processamento de imagens digitais**. (1 ed). São Paulo: Edgard Blucher.

INPE. (2007). **Processamento Digital de imagens: Realce de Imagens Digitais**. Disponível em: <http://www.dpi.inpe.br/~carlos/Academicos/Cursos/Pdi/pdi_realce.html>. Acesso em: 19 maio 2015.

MELLO, Carlos Alexandre. **Segmentação – Detecção de Bordas**. Pernambuco. 72 slides, color. Disponível em: <http://www.cin.ufpe.br/~cabm/visao/Aula05_EdgeDetection.pdf>. Acesso em: 18 maio 2015.

MENNO. **Separador e contador de moedas CS885**. Disponível em: <<http://www.menno.com.br/produtos/divisao/escritorio/gaveteiros-e-contador-de-cedulas/separador-e-contador-automatico-de-moedas-cs-885->>. Acesso em: 20 maio 2015.

OPENCV. (2015). **OpenCV: About OpenCV**. Disponível em: <<http://opencv.org/about.html>>. Acesso em: 19 maio 2015.

OZORIO, Giovana Benevides; FIGUEIREDO, Kleber Fossati; ARKADER, Rebeca. Estratégias de recuperação de serviço no varejo e seu impacto na fidelização dos clientes. **Revista de administração contemporânea**. Curitiba, v.6, n.3, Dez. 2002. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1415-65552002000300004&lng=en&nrm=iso> Acesso em: 23 mar. 2015.

PARALLAX. **Datasheet teclado matricial 4x4 membrana**. Disponível em: <<https://www.parallax.com/sites/default/files/downloads/27899-4x4-Matrix-Membrane-Keypad-v1.2.pdf>> Acesso em: 10 nov. 2015.

PEREIRA, F. **Microcontroladores PIC: Técnicas avançadas**. 4. ed. São Paulo: Érica Ltda, 2012.

SENA JÚNIOR, Darly G. de *et al.* Avaliação de métodos automáticos de limiarização para imagens de plantas de milho atacadas por *Spodoptera frugiperda*. **Revista Brasileira de Engenharia Agrícola e Ambiental**, Campina Grande, v. 7, n. 2. Maio/ago. 2003. Disponível em: <http://www.scielo.br/scielo.php?pid=S1415-43662003000200030&script=sci_arttext>. Acesso em: 18 maio 2015.

SHAPIRO, L. G. (2001). **Computer Vision**. (1 ed., Vol. 1). Prentice Hall.

SHENZHEN. **Datasheet Display LCD 16x2 RT162-7**. Disponível em: <http://img.filipeflop.com/files/download/Datasheet_Display_16x2.pdf> Acesso em: 10 nov. 2015.

SOUZA, D. J. D.; LAVINIA, N. C. **Conectando o PIC - Recursos avançados**. 3. ed. São Paulo: Érica Ltda, 2010.

TOWER PRO. **Datasheet Servomotor MG90s**. Disponível em: <http://www.electronicoscaldas.com/datasheet/MG90S_Tower-Pro.pdf> Acesso em: 30 ago. 2015.

TRITURARE. **Contador e Separador de Moedas CS 885**. Disponível em: <<http://www.triturare.com.br/loja/contador-e-separador-de-moedas-eu350-automat-ico-preco-a-vista-r.html>>. Acesso em: 30 mar. 2015.

TURATTI, Arlei JosÉ; FRANÇA, Felipe Sales de; FERNANDES, Renan Locatiz. **COOLPACOIN: Contador automático de moedas**. Curitiba, 2010. 49 p.

UNESP. Universidade Estadual Paulista. **Servo Motores**. Disponível em: <<http://www.feis.unesp.br/Home/departamentos/engenhariaeletrica/aula-4---servo-motor-13-03-2013-final.pdf>>. Acesso em: 20 abr. 2015.

UOL. **UOL Economia**. Disponível em: <<http://economia.uol.com.br/empreen-dedorismo/album/2014/02/04/conheca-a-maquina-que-recolhe-moedas-e-da-bonus.htm#fotoNav=2>>. Acesso em: 18 mar. 2015.

VISHAY. **Vishay Semiconductors**. Disponível em: <<http://www.vishay.com/docs/83760/tcrt5000.pdf>>. Acesso em: 26 mar. 2015.

WD. **WD Vending Machine**. Disponível em: <<http://wdvending.com.br/wpcontent/uploads/2013/12/Samba-Top.jpg>>. Acesso em: 19 mar. 2015.

WEG. **Eletromotores WEG SA**. Disponível em: <<http://www.weg.net/br/Produtos-e-Servicos/Controls/Sensores-Industriais/Sensores-Opticos>>. Acesso em: 26 mar. 2015.

ZUECH, Nello. **Applying Machine Vision**. Wiley-interscience, 1988. 280 p.

8. APÊNDICES

Apêndice A: Algoritmo para a o sistema de fornecimento de troco, leitura do teclado, visualização no *display* e sensores.

```
-----  
#include <LiquidCrystal.h>  
#include <Servo.h>  
#include <Keypad.h>  
#define Luz_Fundo 50  
  
LiquidCrystal lcd(38, 40, 42, 44, 46, 48); // Cria um objeto tipo  
LiquidCrystal que se chama "lcd", com numeração dos pinos correspondentes  
  
int m100 = 0, m50 = 0, m25 = 0, m10 = 0, m5 = 0; // Moedas disponíveis  
na máquina.  
int um_real = 0, cinquenta = 0, vinte_cinco = 0, dez = 0, cinco = 0;  
// Quantidade de moedas liberaras.  
long int troco_digitado = 0;  
int modo = 0;  
  
int sensor5 = 0, sensor10 = 1, sensor25 = 2, sensor50 = 3, sensor100 = 4;  
int val5 = 0, val10 = 0, val25 = 0, val50 = 0, val100 = 0;  
long ultmedsensor5 = 0, ultmedsensor10 = 0, ultmedsensor25 = 0,  
ultmedsensor50 = 0, ultmedsensor100 = 0;  
  
Servo servo_5, servo_10, servo_25, servo_50, servo_100;  
  
String inString = "";  
  
long tempoa = 0;  
long tempom = 0;  
int mostra = 0, mostra_moeda = 0;  
  
const byte Linhas = 4; // Definição da constante das 4 linhas.  
const byte Colunas = 4; // Definição da constante das 4 colunas.  
byte Pinos_Linhas[] = {22, 24, 26, 28}; // Pinos do Arduino  
correspondente aos pinos 1,2,3 e 4 do teclado matricial, referente às  
linhas.  
byte Pinos_Colunas[] = { 30, 32, 34, 36}; // Pinos do Arduino  
correspondente aos pinos 5,6,7 e 8 do teclado matricial, referente às  
colunas.  
  
char Teclas [ Linhas ][ Colunas ] =  
{  
  {'1','2','3','A'},  
  {'4','5','6','B'},  
  {'7','8','9','C'},  
  {'*','0','#','D'}  
};  
  
Keypad Teclado1 = Keypad(makeKeymap(Teclas), Pinos_Linhas, Pinos_Colunas,  
Linhas, Colunas); // Cria uma instância do tipo Keypad nomeada  
"Teclado1". É atribuída à matriz "Teclas" os pinos das linhas e colunas  
ligadas ao arduino, bem como suas dimensões.  
  
void setup()  
{
```

```

lcd.begin(16, 2); // Inicia o objeto "lcd" de 2 linhas e 16 colunas.
pinMode(Luz_Fundo, OUTPUT // Define o pino Luz_Fundo como saída.
digitalWrite(Luz_Fundo, HIGH); // Liga a luz do display.
Serial.begin(9600);
servo_5.attach(6);
servo_10.attach(5);
servo_25.attach(4);
servo_50.attach(3);
servo_100.attach(2);
}

void inicia_sensor_5()
{
  if ((millis() - ultmedsensor5) > 50)
  {
    val5 = analogRead(sensor5);
    ultmedsensor5 = millis();
    if(val5 > 28)
    {
      m5++;
    }
  }
}

void inicia_sensor_10()
{
  if ((millis() - ultmedsensor10) > 30)
  {
    val10 = analogRead(sensor10);
    ultmedsensor10 = millis();
    if(val10 > 30)
    {
      m10++;
    }
  }
}

void inicia_sensor_25()
{
  if ((millis() - ultmedsensor25) > 50)
  {
    val25 = analogRead(sensor25);
    ultmedsensor25 = millis();
    if(val25 > 30)
    {
      m25++;
    }
  }
}

void inicia_sensor_50()
{
  if ((millis() - ultmedsensor50) > 50)
  {
    val50 = analogRead(sensor50);
    ultmedsensor50 = millis();
    if(val50 > 28)
    {
      m50++;
    }
  }
}

```

```

void inicia_sensor_100()
{
  if ((millis() - ultmedsensor100) > 50)
  {
    val100 = analogRead(sensor100);
    ultmedsensor100 = millis();
    if(val100 > 28)
    {
      m100++;
    }
  }
}

void Standby()
{
  char pulso = Teclado1.getKey();          // Realiza a leitura do teclado.

  if (millis() - tempoa > 4000)
  {
    mostra ++;
    lcd.clear();
    if (mostra == 8)
    {
      mostra = 0;
    }
    tempoa = millis();
  }

  if (mostra == 0)
  {
    lcd.setCursor(3,0);
    lcd.print ("Maquina de");
    lcd.setCursor(1,1);
    lcd.print ("fornecer troco");
  }

  if (mostra == 1)
  {
    lcd.setCursor(2,0);
    lcd.print ("Escolha uma");
    lcd.setCursor(5,1);
    lcd.print ("opcao:");
  }

  if (mostra == 2)
  {
    lcd.setCursor(3,0);
    lcd.print ("(A) Exibir");
    lcd.setCursor(4,1);
    lcd.print ("montante");
  }

  if (mostra == 3)
  {
    lcd.setCursor(1,0);
    lcd.print ("(B) Quantidade");
    lcd.setCursor(3,1);
    lcd.print ("de moedas");
  }

  if (mostra == 4)
  {
    lcd.setCursor(1,0);

```

```

    lcd.print ("(C) Digitar o");
    lcd.setCursor(1,1);
    lcd.print ("valor do troco");
}

if (mostra == 5)
{
    lcd.setCursor(0,0);
    lcd.print ("(D) Retornar ao");
    lcd.setCursor(2,1);
    lcd.print ("menu inicial");
}

if (mostra == 6)
{
    lcd.setCursor(0,0);
    lcd.print ("(*) em (C) para");
    lcd.setCursor(0,1);
    lcd.print ("liberar o troco");
}

if (mostra == 7)
{
    lcd.setCursor(0,0);
    lcd.print ("(#) em (C) para");
    lcd.setCursor(0,1);
    lcd.print ("corrigir o troco");
}
}

void Tecla_A()
{
    char pulso = Teclado1.getKey();
    lcd.setCursor(4,0);
    lcd.print("Montante:");
    lcd.setCursor(4,1);
    lcd.print("R$ ");
    lcd.print((1*m100)+(0.50*m50)+(0.25*m25)+(0.10*m10)+(0.05*m5));

    if (pulso == 'B')
    {
        lcd.clear();
        modo = 2;
        inString = "";
        tempom = millis();
    }

    if (pulso == 'C')
    {
        lcd.clear();
        modo = 3;
        lcd.setCursor(0,0);
        lcd.print("Digite o troco:");
        lcd.setCursor(4,1);
        lcd.print("R$ ");
    }

    if (pulso == 'D')
    {
        mostra = 0;
        modo = 0;
        lcd.clear();
        inString = "";
    }
}

```

```

        tempoa = millis();
    }
}

void Tecla_B()
{
    char pulso = Teclado1.getKey();
    if (millis() - tempom > 3000)
    {
        mostra_moeda ++;
        lcd.clear();
        if (mostra_moeda == 6)
        {
            mostra_moeda = 0;
        }
        tempom = millis();
    }

    if (mostra_moeda == 0)
    {
        lcd.setCursor(1,0);
        lcd.print("A maquina tem:");
        lcd.setCursor(3,1);
        lcd.print(m5 + m10 + m25 + m50 + m100);
        lcd.print(" moedas");
    }

    if (mostra_moeda == 1)
    {
        lcd.setCursor(4,0);
        lcd.print(m5);
        lcd.print(" moedas");
        lcd.setCursor(3,1);
        lcd.print("de R$ 0,05");
    }

    if (mostra_moeda == 2)
    {
        lcd.setCursor(4,0);
        lcd.print(m10);
        lcd.print(" moedas");
        lcd.setCursor(3,1);
        lcd.print("de R$ 0,10");
    }

    if (mostra_moeda == 3)
    {
        lcd.setCursor(4,0);
        lcd.print(m25);
        lcd.print(" moedas");
        lcd.setCursor(3,1);
        lcd.print("de R$ 0,25");
    }

    if (mostra_moeda == 4)
    {
        lcd.setCursor(4,0);
        lcd.print(m50);
        lcd.print(" moedas");
        lcd.setCursor(3,1);
        lcd.print("de R$ 0,50");
    }
}

```

```

if (mostra_moeda == 5)
{
    lcd.setCursor(4,0);
    lcd.print(m100);
    lcd.print(" moedas");
    lcd.setCursor(3,1);
    lcd.print("de R$ 1,00");
}

if (pulso == 'A')
{
    lcd.clear();
    modo = 1;
    inString = "";
}

if (pulso == 'C')
{
    lcd.clear();
    modo = 3;
    lcd.setCursor(0,0);
    lcd.print("Digite o troco:");
    lcd.setCursor(4,1);
    lcd.print("R$ ");
}

if (pulso == 'D')
{
    mostra = 0;
    modo = 0;
    lcd.clear();
    inString = "";
    tempoa = millis();
}
}

void Tecla_C()
{
    char pulso1 = Teclado1.getKey();

    if (pulso1 == 'A')
    {
        lcd.clear();
        modo = 1;
        inString = "";
    }

    if (pulso1 == 'B')
    {
        lcd.clear();
        modo = 2;
        inString = "";
        tempom = millis();
    }

    if (pulso1 == 'D')
    {
        mostra = 0;
        modo = 0;
        lcd.clear();
        inString = "";
        tempoa = millis();
    }
}

```

```

if (isDigit(pulsol) && (inString.length()<5))
{
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Digite o troco:");
  inString += (char)pulsol;
  lcd.setCursor(4,1);
  lcd.print("R$ ");
  lcd.print((int)(inString.toInt()/100));
  lcd.print(",");
  int fracao = ((int) inString.toInt() % 100);
  if (fracao < 10) lcd.print("0");
  lcd.print( fracao );
}

if (pulsol == '#')
{
  inString = "";
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Digite o troco:");
  lcd.setCursor(4,1);
  lcd.print("R$ ");
  lcd.print(inString);
}

if (pulsol == '*')
{
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Troco digitado:");
  lcd.setCursor(4,1);
  lcd.print("R$");
  lcd.setCursor(7,1);
  troco_digitado = inString.toInt();
  lcd.print((float)troco_digitado/100);
  inString = "";

  if ((float)troco_digitado > ((1*m100)+(0.50*m50)+(0.25*m25)+(0.10*
m10)+(0.05*m5))*100))
  {
    lcd.clear();
    lcd.setCursor(1,0);
    lcd.print("Troco invalido");
    lcd.setCursor(0,1);
    lcd.print("digite novamente");
    inString = "";
  }

else
{
  while (troco_digitado > 0)
  {
    while (troco_digitado >= 100)
    {
      if (m100 > 0)
      {
        troco_digitado = (troco_digitado - 100);
        servo_100.write(40);
        delay(500);
        servo_100.write(134);
        delay(500);
      }
    }
  }
}
}

```

```

        um_real ++;
        m100 = m100 - 1;
    }
    else
    {
        break;
    }
}

while (troco_digitado >= 50)
{
    if (m50 > 0)
    {
        troco_digitado = (troco_digitado - 50);
        servo_50.write(80);
        delay(500);
        servo_50.write(148);
        delay(500);
        cinquenta ++;
        m50 = m50 - 1;
    }
    else
    {
        break;
    }
}

while (troco_digitado >= 25)
{
    if (m25 > 0)
    {
        troco_digitado = (troco_digitado - 25);
        servo_25.write(42);
        delay(500);
        servo_25.write(132);
        delay(500);
        vinte_cinco ++;
        m25 = m25 - 1;
    }
    else
    {
        break;
    }
}

while (troco_digitado >= 10)
{
    if (m10 > 0)
    {
        troco_digitado = (troco_digitado - 10);
        servo_10.write(53);
        delay(500);
        servo_10.write(143);
        delay(500);
        dez ++;
        m10 = m10 - 1;
    }
    else
    {
        break;
    }
}

```

```

while (troco_digitado >= 3)
{
    if (m5 > 0)
    {
        troco_digitado = (troco_digitado - 5);
        servo_5.write(38);
        delay(500);
        servo_5.write(128);
        delay(500);
        cinco ++;
        m5 = m5 - 1;
    }
    else
    {
        break;
    }
}
break;
}inString = "";
}
}
}

```

```

void loop()
{
    servo_5.write(128);
    servo_10.write(143);
    servo_25.write(132);
    servo_50.write(148);
    servo_100.write(134);
    inicia_sensor_5();
    inicia_sensor_10();
    inicia_sensor_25();
    inicia_sensor_50();
    inicia_sensor_100();

    if (modo == 0)
    {
        char pulso = Teclado1.getKey();

        if (pulso == NO_KEY)
        {
            Standby();
        }

        if (pulso == 'A')
        {
            lcd.clear();
            modo = 1;
        }

        if (pulso == 'B')
        {
            lcd.clear();
            modo = 2;
            tempom = millis();
        }

        if (pulso == 'C')
        {
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("Digite o troco:");
        }
    }
}

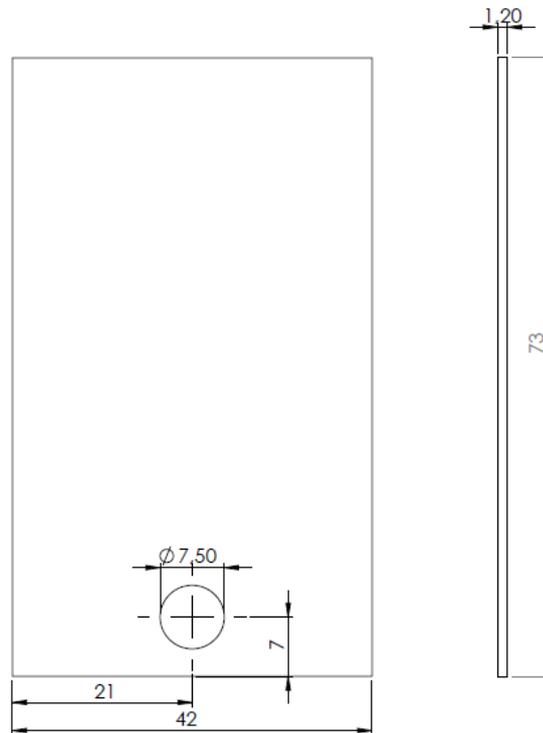
```

```
        lcd.setCursor(4,1);
        lcd.print("R$ ");
        modo = 3;
    }
}

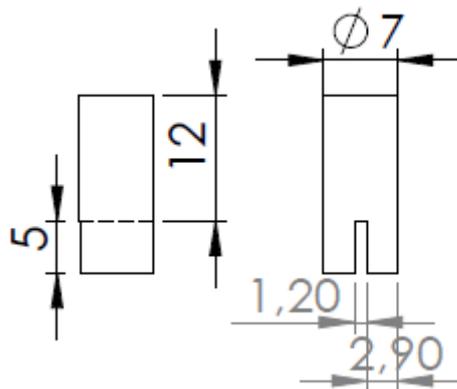
if (modo == 1)
{
    Tecla_A();
}
if (modo == 2)
{
    Tecla_B();
}
if (modo == 3)
{
    Tecla_C();
}
}
```

Apêndice B: Projeto das peças da máquina.

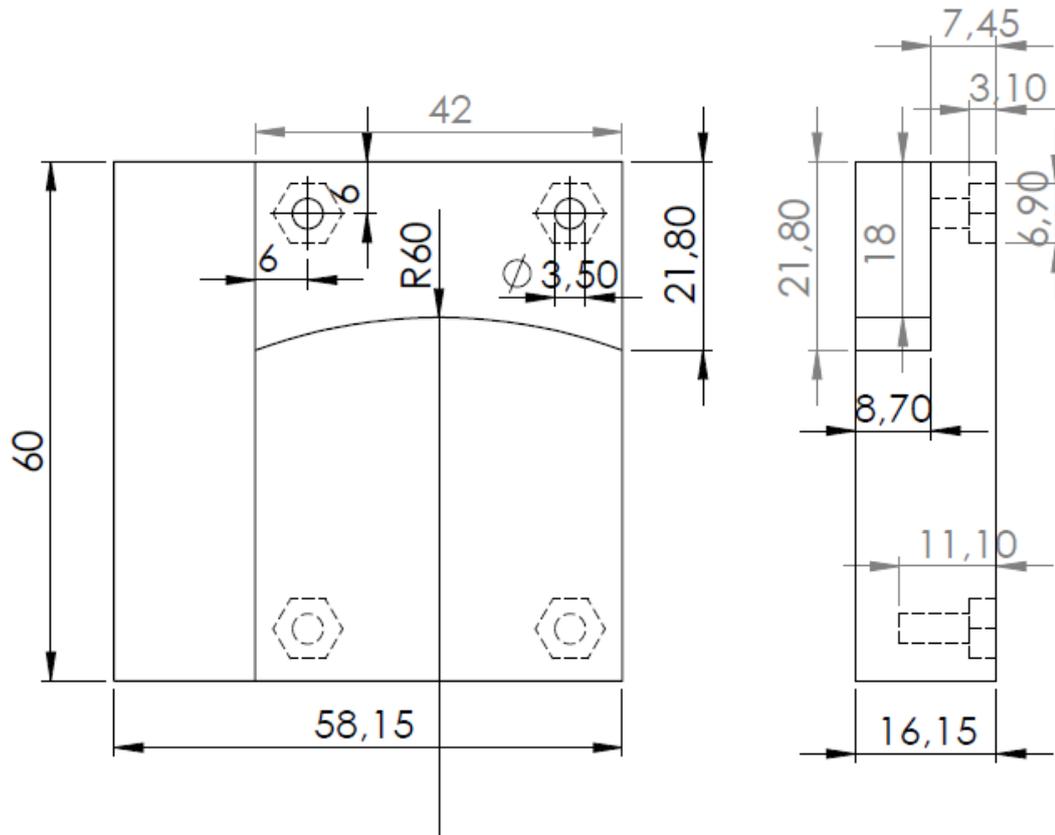
Chapa dispensadora de troco (5 peças):



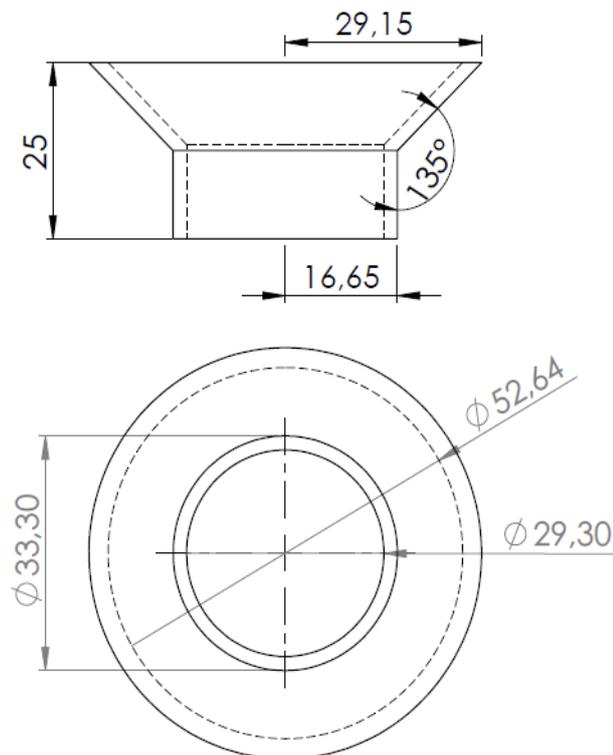
Eixo para os rolamentos (5 peças):



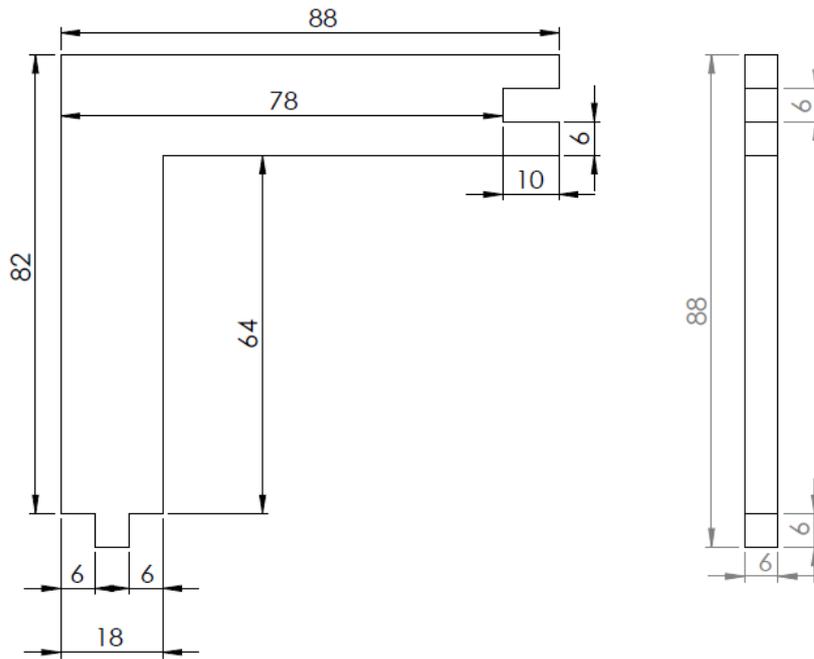
Base para dispensa de moedas (5 peças):



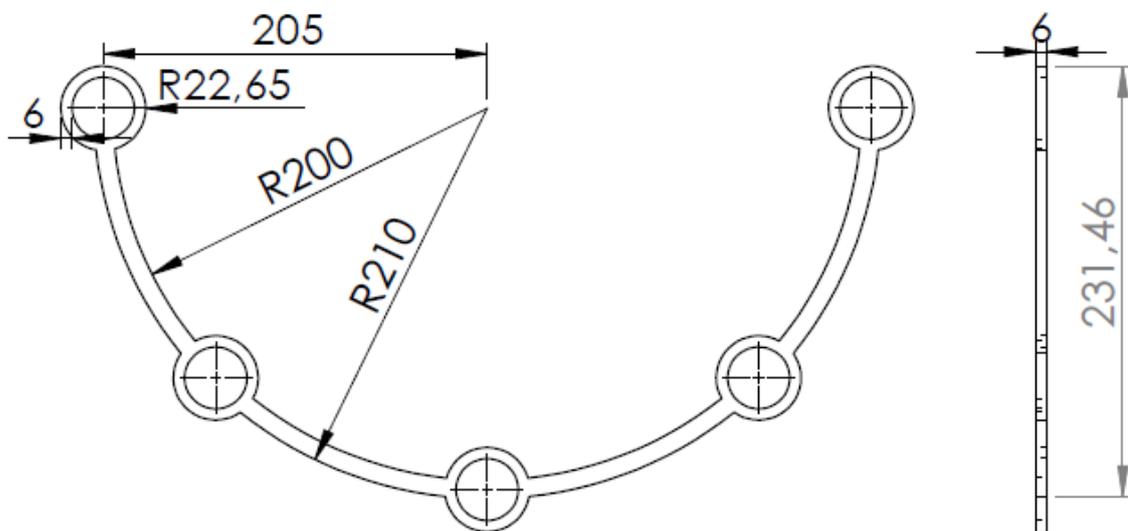
Funil para a entrada das moedas nos tubos e para liberação do troco (10 peças):



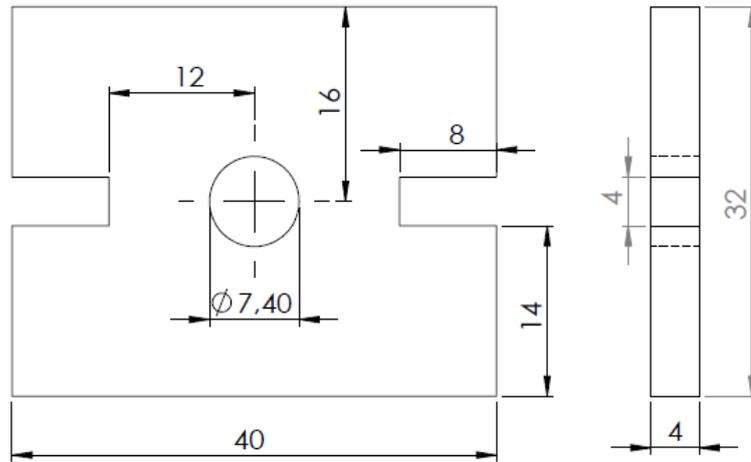
Apoio do suporte dos tubos do sistema de troco (4 peças):



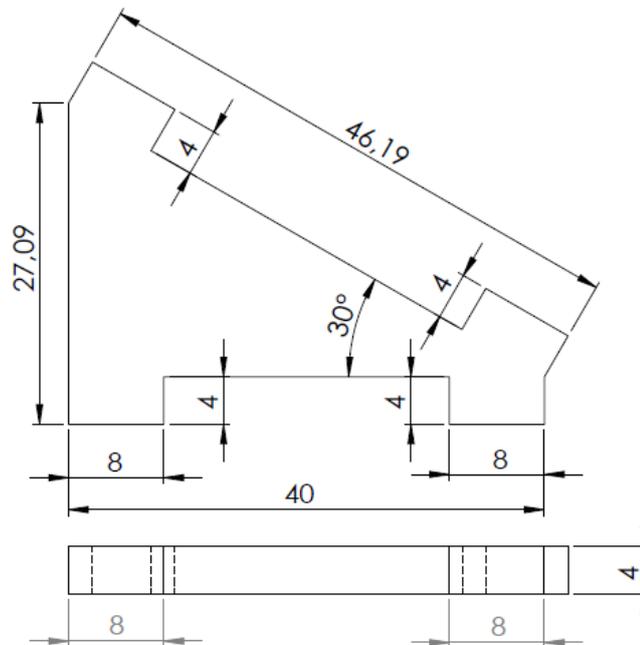
Suporte dos tubos do sistema de troco:



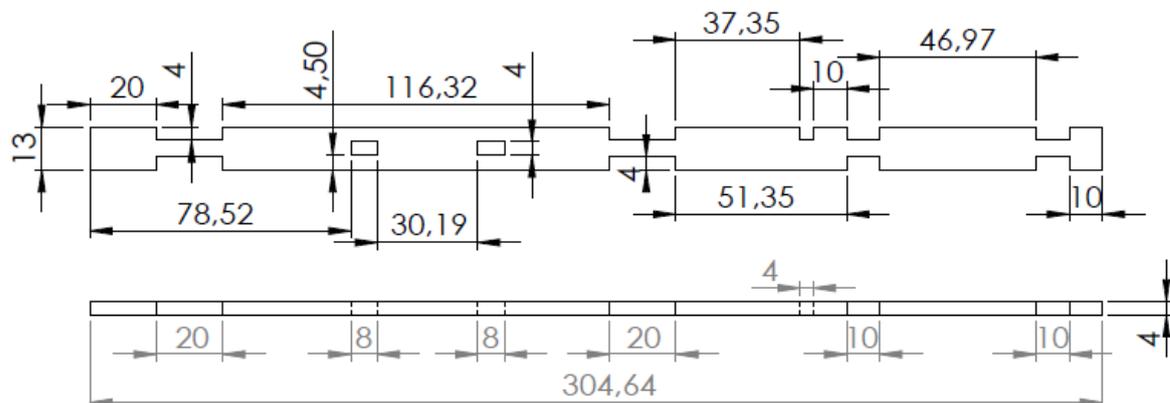
Base do suporte inclinado do separador:



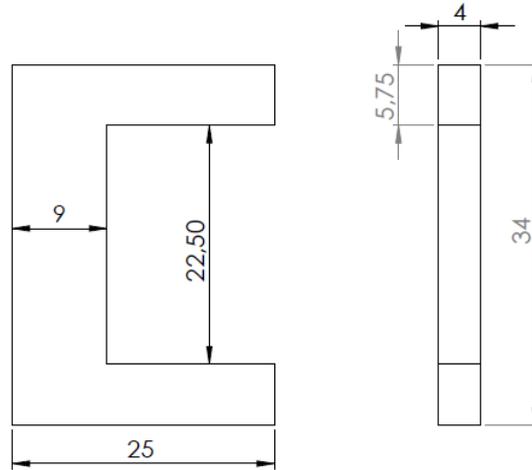
Suporte inclinado da rampa do separador:



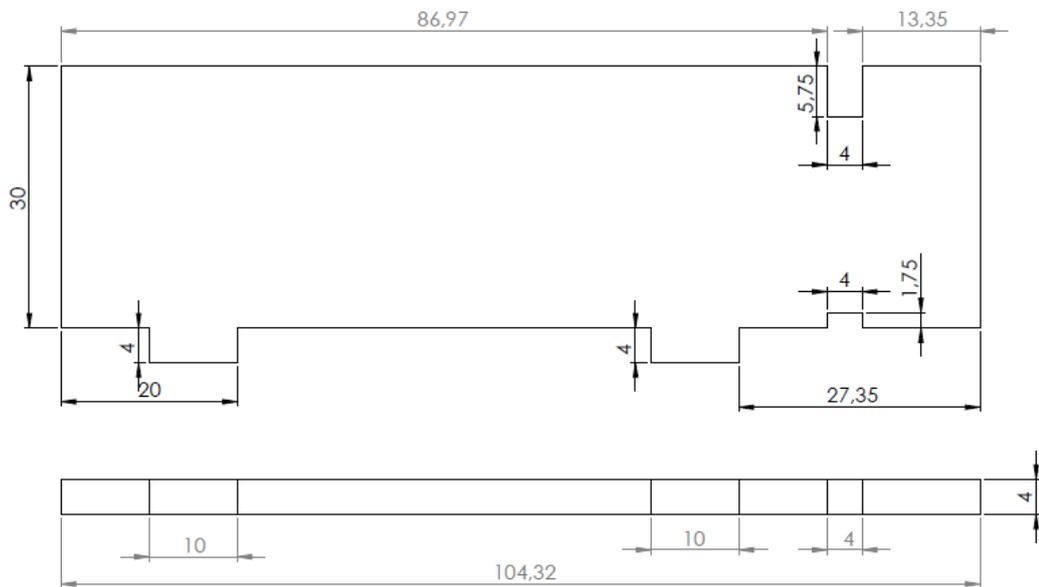
Rampa do separador de moedas:



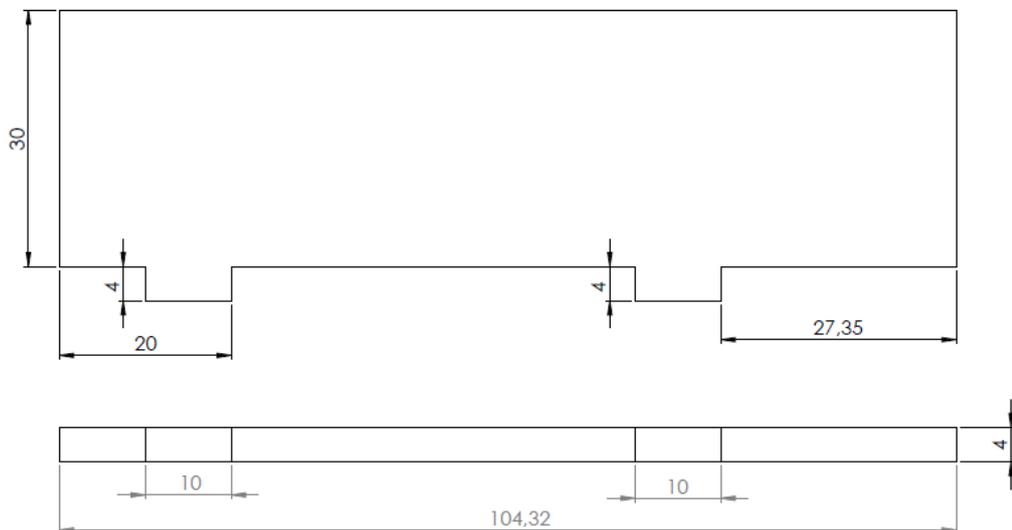
Suporte de fixação do servomotor:



Lateral menor lado do servomotor bloqueador de moedas:



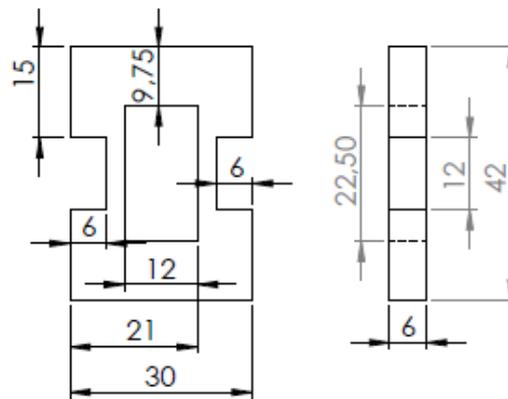
Lateral menor lado oposto ao servomotor:



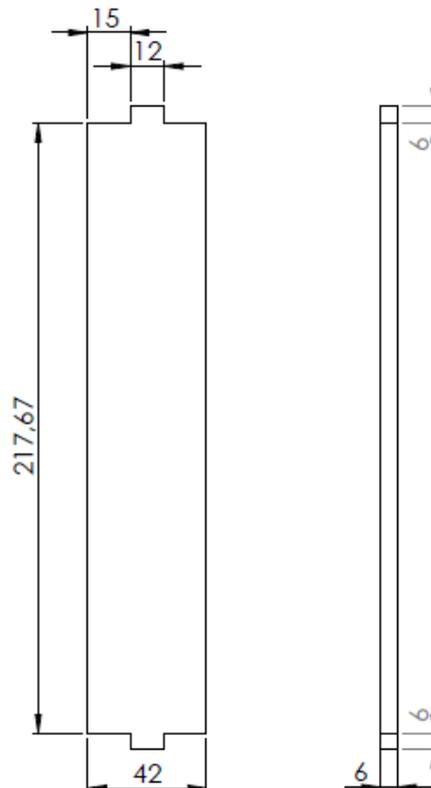
Lateral maior do separador de moedas (2 peças):



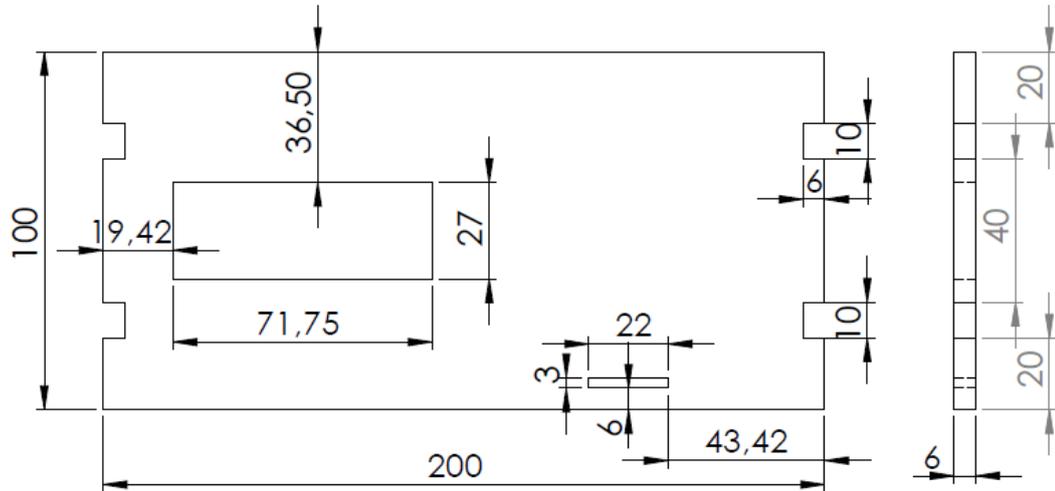
Suporte do servomotor do separador:



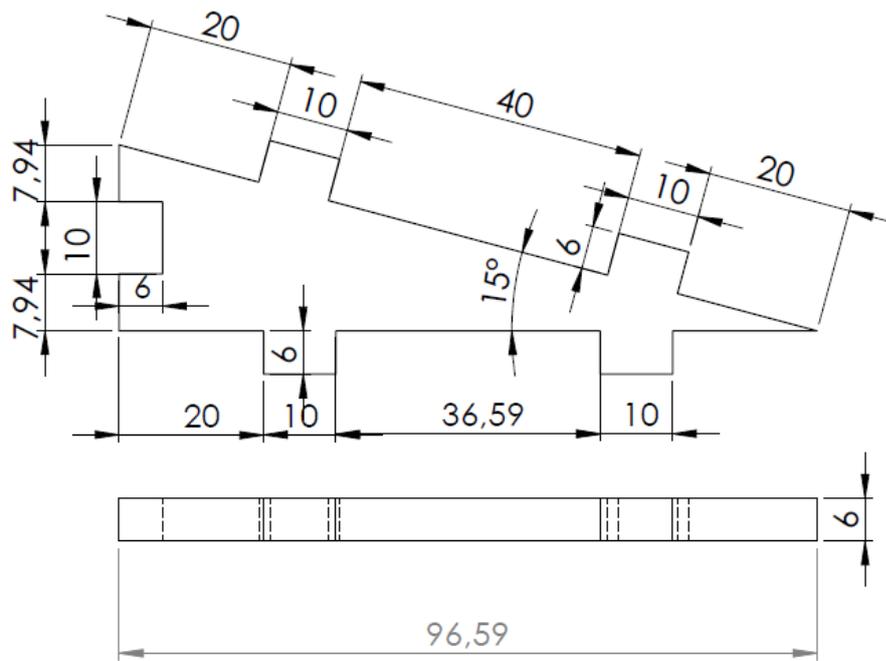
Base do separador de moedas (2 peças):



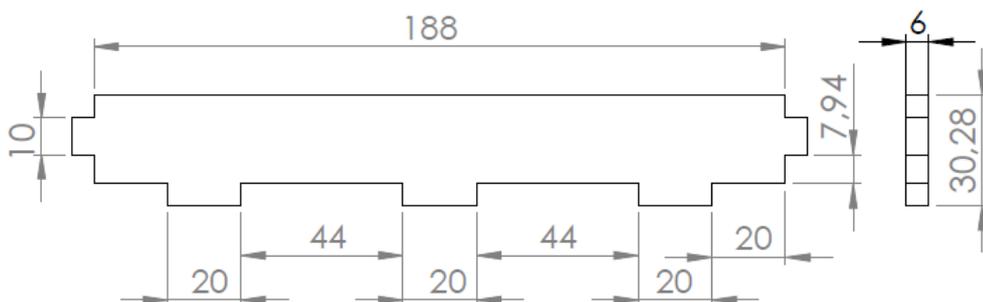
Painel frontal de comando:



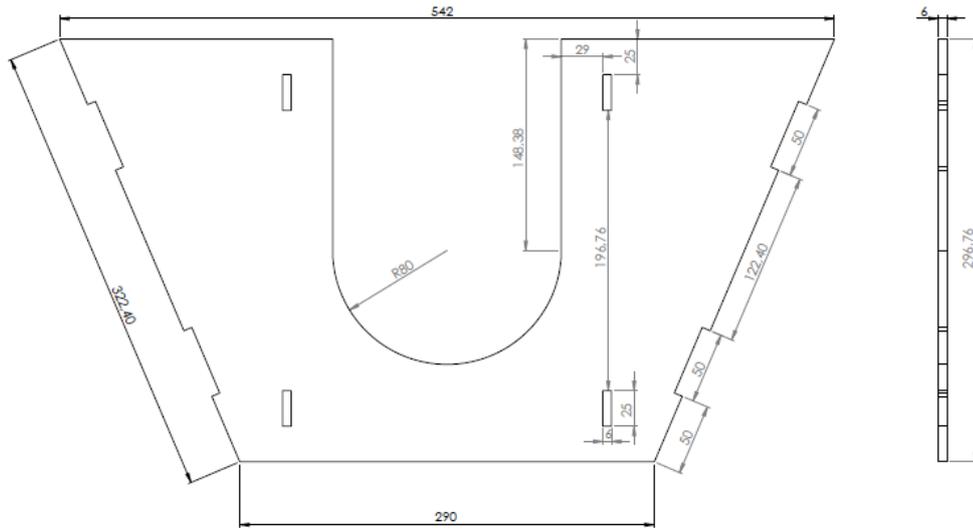
Lateral da central de comando (2 peças):



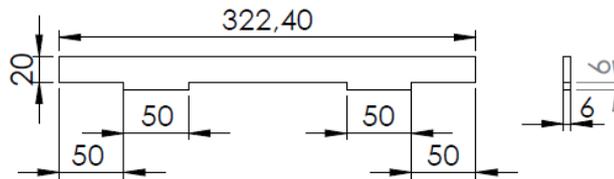
Parte traseira do painel de comando:



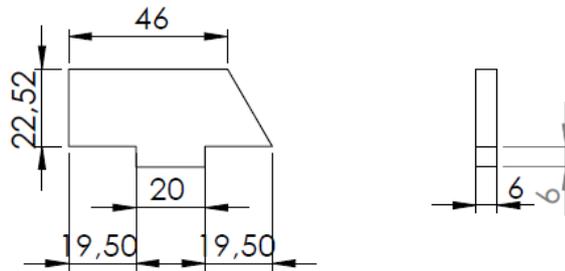
Rampa de dispensa das moedas:



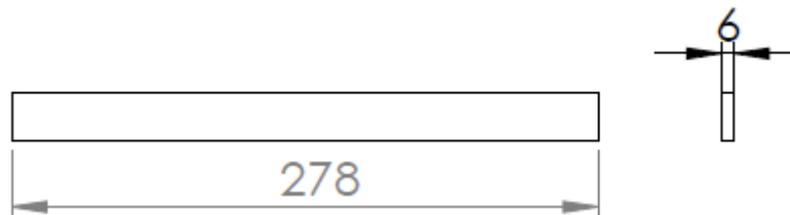
Limitador lateral da rampa (2 peças):



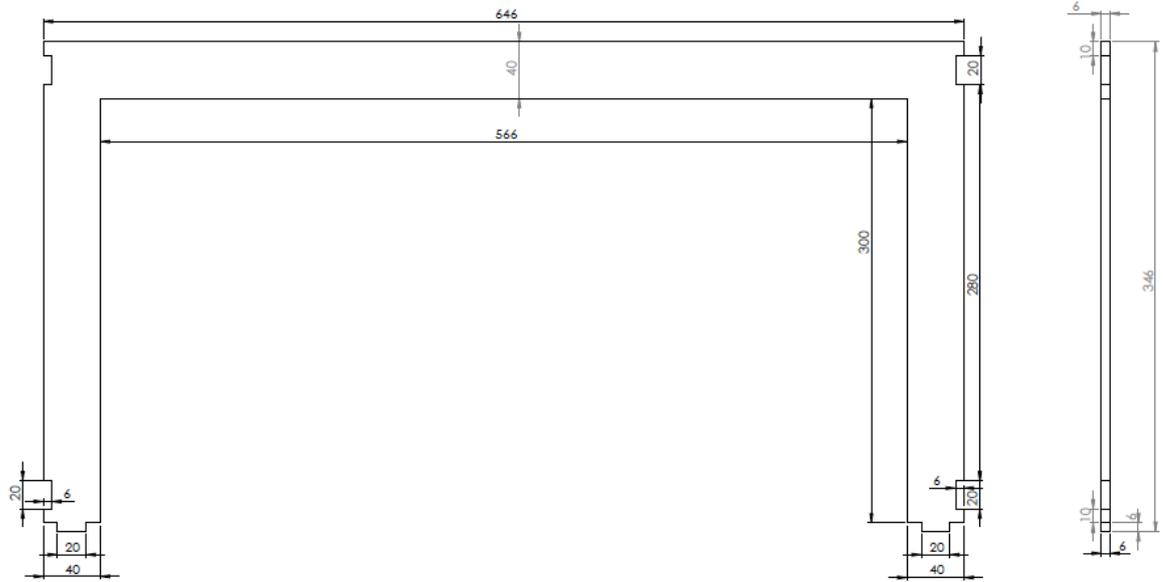
Limitador da extremidade da rampa (2 peças):



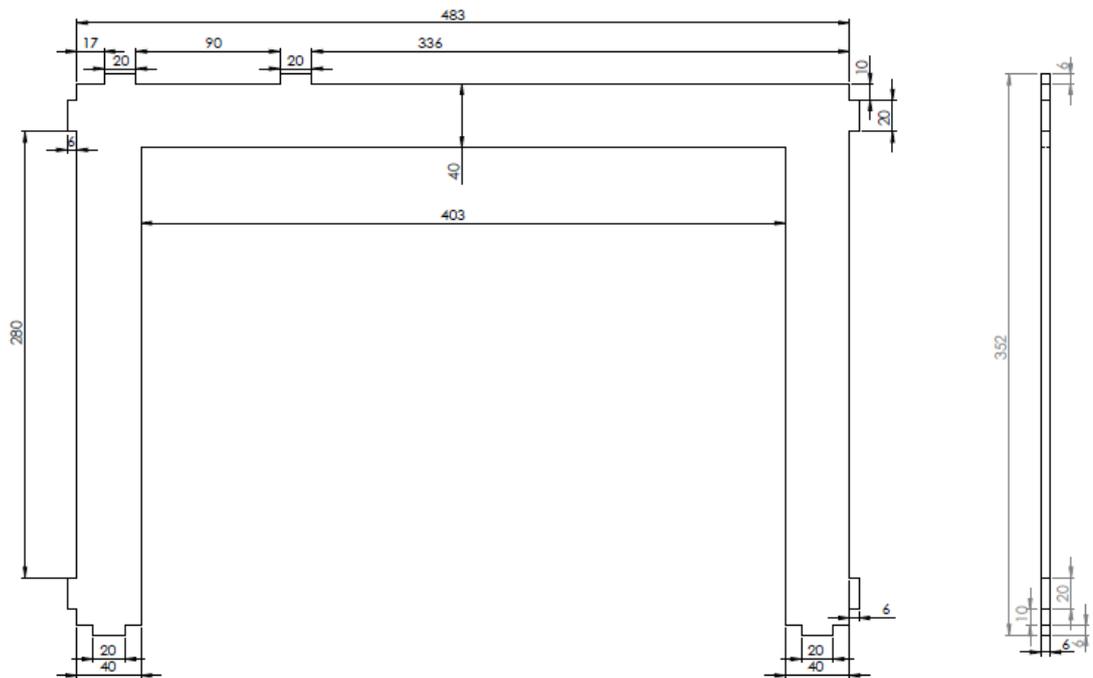
Limitador frontal da rampa:



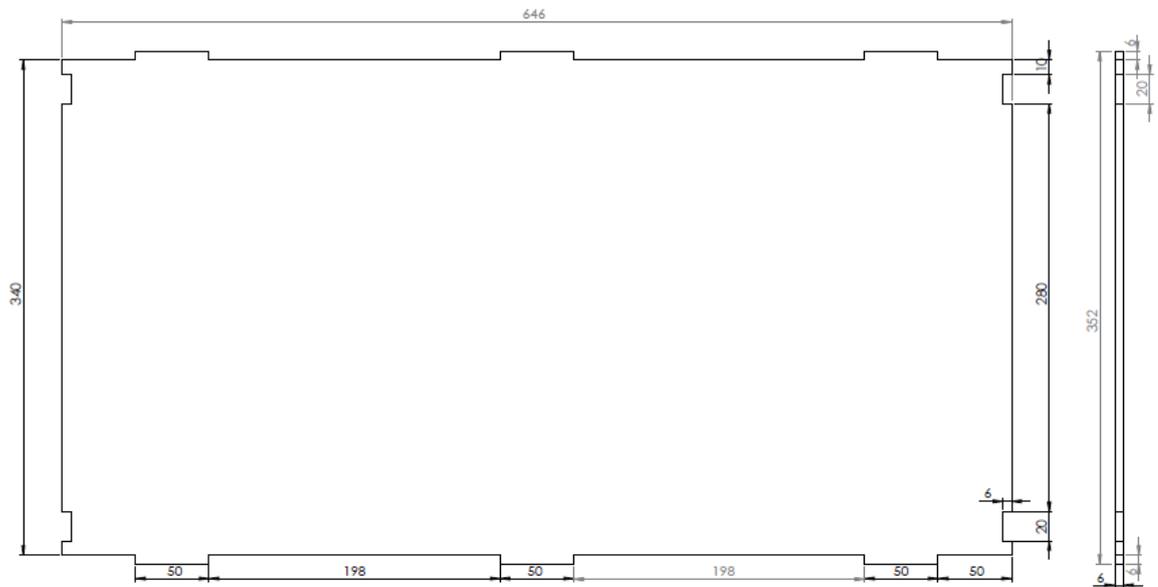
Carça da frente da máquina:



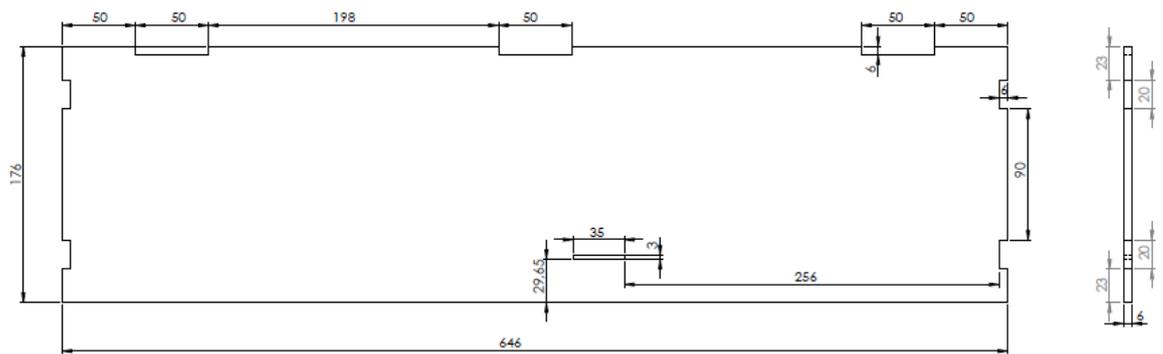
Carça lateral da máquina (2 peças):



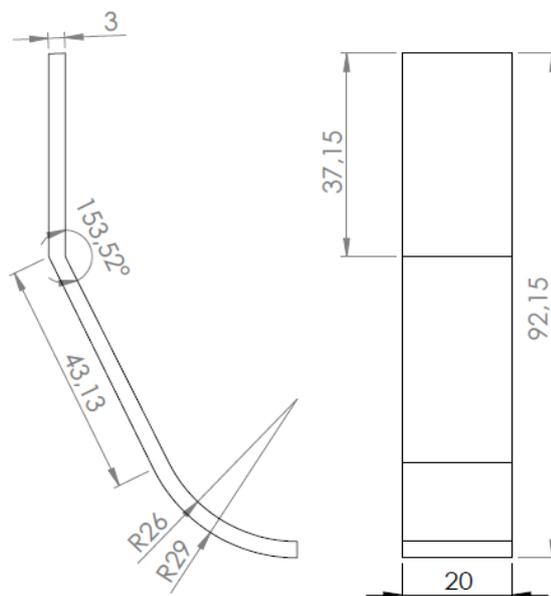
Carcaça traseira da máquina:



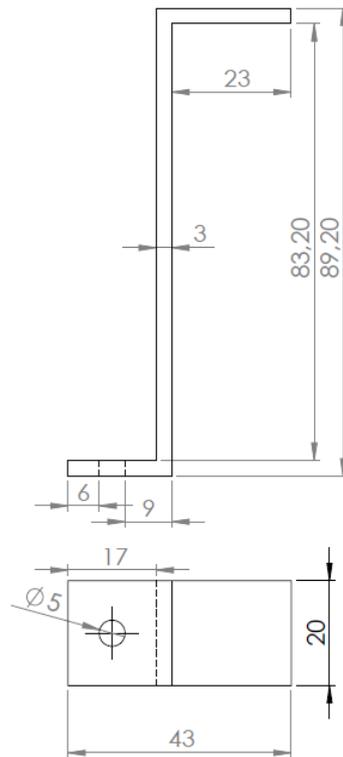
Entrada das moedas do separador:



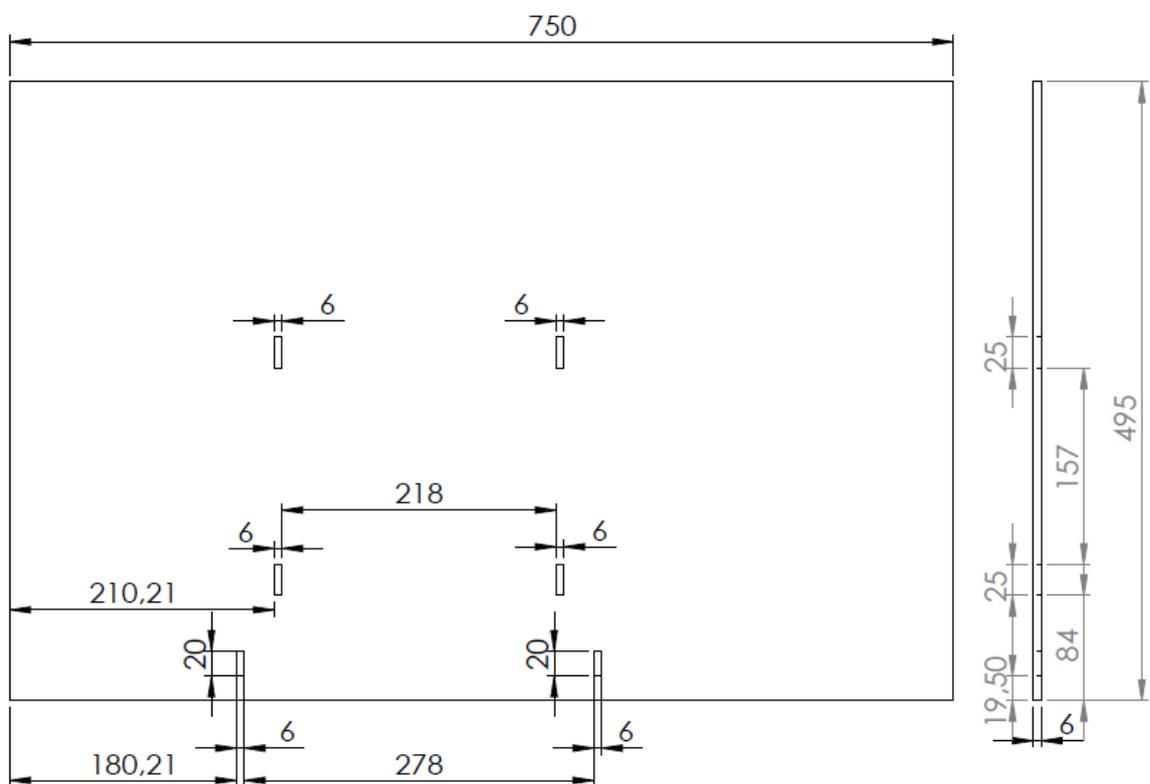
Limitador de moedas do sistema de troco (5 peças):



Apoio da fonte (4 peças):

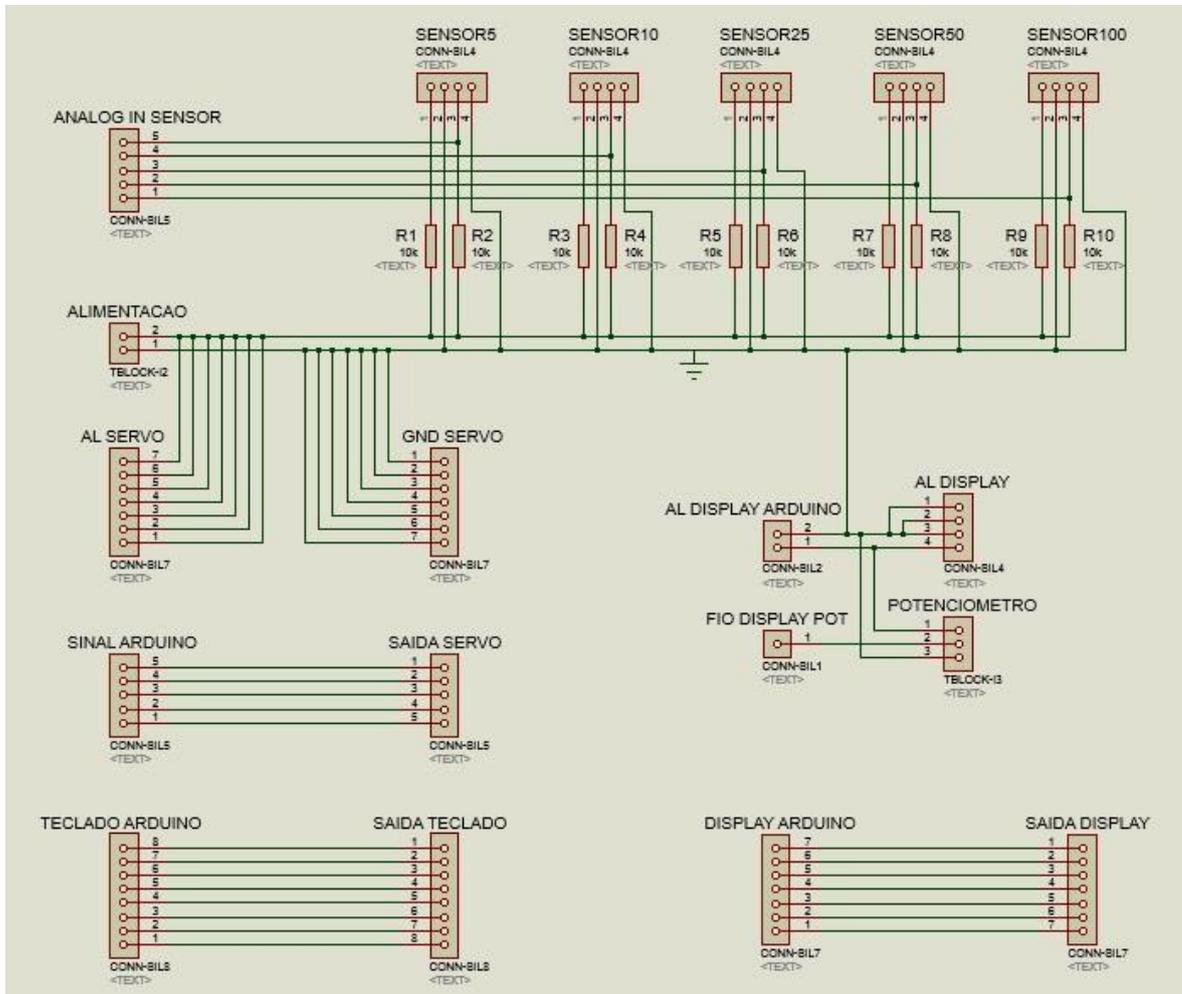


Base inferior da máquina:



Apêndice C: Esquemático da placa eletrônica e *layout* do circuito.

Esquemático do *shield* desenvolvido no software Proteus, para ligar os sensores ópticos, servomotores, *display* e teclado matricial ao Arduino Mega:



Layout do circuito impresso na placa:

