UFSM

JESTA
Journal of Exact Sciences and Technological Applications

ACESSO ABERTO

JESTA, Cachoeira do Sul, v. 2, p. 01-17, e74253, 2023
Submissão: 30/03/2023 • Aprovação: 14/09/2023 • Publicação: 12/12/2023

Aplicações tecnológicas

Desenvolvimento de Moedeiro Microcontrolado para auxílio no ensino de Lógica e Programação

Development of a Microcontrolled Coin Maker to assist in the teaching of Logic and Programming

César Teixeira Pacheco[I], Gustavo Lenhardt Steffen[I], Arthur Cordeiro Andrade[I], Diogo Ribeiro Vargas[I]

[I] Universidade Federal de Santa Maria, Cachoeira do Sul, RS, Brasil

RESUMO

O presente trabalho busca desenvolver um contador de moedas, no qual, a partir da inserção de moedas em uma caixa, serão informados em um display LCD (16x2) o valor da moeda inserida, o número de moedas na caixa e o total de dinheiro acumulado na mesma.

Palavras-chave: Contador; Moedeiro; Desenvolver

ABSTRACT

The present work seeks to develop a coin counter, in which, by inserting coins in to a box, an LCD display (16x2) will show the value of the inserted coin, the number of coins in the box, and the total amount of money accumulated in the box.

Keywords: Counter; Grinder; Develop

Artigo publicado por JESTA sob uma licença CC BY-NC 4.0.

INTRODUÇÃO

Na eletrônica existem recursos disponíveis para manipular e controlar ações por meio da comunicação com componentes. O tipo de controle mais utilizado em automação, ou até mesmo em circuitos essenciais, tem como principal componente eletrônico o microcontrolador.

A popularização do uso de microcontroladores no ensino, em todos os seus níveis é justificada pela criação das placas de prototipagem eletrônica, entre as quais pode-se citar o Arduino, o Raspberry, STM32, que possuem todos os recursos necessários para a criação de protótipos de forma fácil e rápida.

Nas placas de prototipagem eletrônica, o microcontrolador é o “cérebro” do processo, tomando decisões de acordo com os valores das entradas, executando tarefas com circuitos auxiliares como, osciladores, reguladores de tensão e circuitos de transferência de dados, dessa forma, a aprendizagem é facilitada sendo possível a execução de uma infinidade de projetos a partir de uma placa de prototipagem eletrônica. A placa de prototipagem mais utilizada atualmente é o Arduino. O mesmo é uma plataforma eletrônica open source composta por *hardware* e *software*.

O *hardware* consiste em uma placa com um microcontrolador, já o *software* é a IDE (ambiente integrado de desenvolvimento em tradução livre) utilizada para o desenvolvimento das aplicações. Uma curiosidadecom relação a criação da palavra Arduino, é que ela tem origem germânica, nativa do latim Teutoni, que significa “firmeza e resistência”. Outra curiosidade é que a placa não possuía um nome, até que os criadores frequentaram um pub chamado Bar di Re Arduino, em homenagem ao rei d’Ivrea, que governou o país entre 1002 a 1014, marcado na história por seus projetos inovadores [1].

Como o Arduino é uma ferramenta acessível, a mesma é utilizada a fim de solucionar problemas do nosso cotidiano das mais variadas complexidades, como por exemplo, o tempo desprendido para se realizar a simples contagem de moedas.

O presente trabalho buscar desenvolver um contador de moedas, onde, a partir da inserção de moedas em uma caixa, serão informados em um display LCD (16x2) o valor da moeda inserida, o número de moedas na caixa e o total de dinheiro acumulado na mesma. O protótipo a ser desenvolvido tem o objetivo de identificar os cinco principais modelos de moeda utilizados: 5 centavos na cor cobre, 10 centavos na cor dourada, 25 centavos na cor dourada, 50 centavos na cor prata e a moeda de 1 real prata e dourada. Portanto, as entradas e saídas do protótipo a ser desenvolvido podem ser resumidas de acordo com a tabela 1.

Tabela 1. Parâmetros do Protótipo

TIPO DE VARIÁVEL	PARÂMETRO	UNIDADE
Entrada	Moeda	Valor absoluto
Saída	Número de moedas	Valor absoluto
	Valor da última moeda	R$
	Valor acumulado	R$

Com o objetivo de validar a eficácia do protótipo, serão conduzidos testes nos quais diferentes moedas serão inseridas, visando verificar a eficiência do sistema. Como objetivo, será buscada uma taxa de assertividade na leitura das moedas acima de 85%.

DESENVOLVIMENTO

Para a construção de um protótipo, que seja capaz de identificar o valor de uma moeda, quantificando o número de moedas e o total acumulado, serão utilizados um Arduino Uno R3, 5 sensores de obstáculos reflexivo infravermelho, 1 display LCD 16x2 integrado com módulo i2c, 1 push botton, cabos e conexões, 1 caixa de papelão, chapa de Madeira MDF e papel contact

Referencial Teórico

Neste tópico serão abordados os principais equipamentos eletrônicos utilizados para a confecção do protótipo.

Arduino Uno R3

O Arduino Uno R3 é uma placa que utiliza o microcontrolador ATmega328 como base. Possui 14 pinos de entrada/saída digital, dos quais 6 podem ser configurados como saídas PWM. Além disso, possui 6 entradas analógicas, um cristal oscilador de 16MHz, uma conexão USB, uma entrada de alimentação, uma conexão ICSP e um botão de reset. O microcontrolador pode ser alimentado tanto pela conexão USB quanto por uma fonte de alimentação externa [3]. Cada um dos 14 pinos digitais do Uno R3 pode ser usado como entrada ou saída utilizando as funções pinMode(), digitalWrite() e digitalRead(). Eles operam a uma tensão de 5V e cada pino pode fornecer ou receber no máximo 40mA. O Arduino Uno também possui 6 entradas analógicas, identificadas de A0 a A5, com uma resolução de 10 bits (ou seja, 1024 valores diferentes). Por padrão, essas entradas medem de 0 a 5V. Na Figura 1 a seguir, estão destacadas as principais portas disponíveis no Arduino Uno R3 [4].

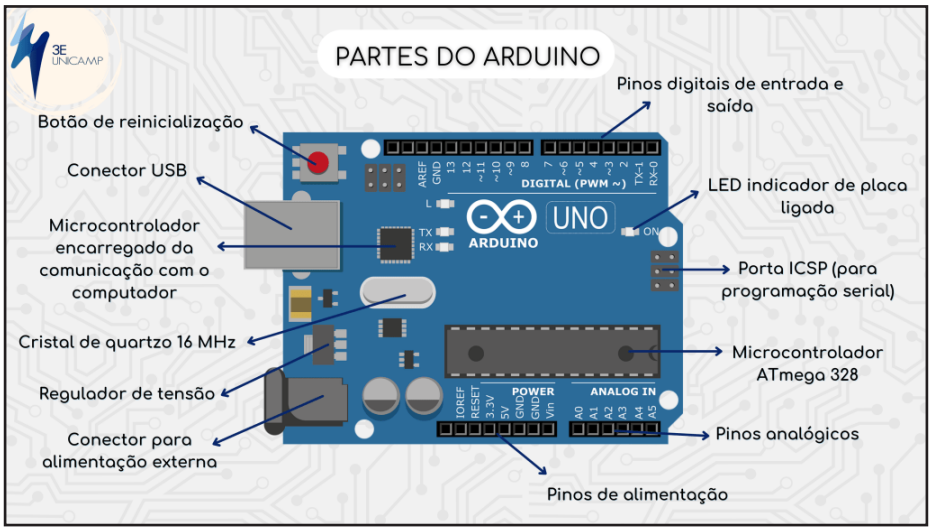

Figura 1. Principais portas do Arduino UNO R3.

Sensor de Obstáculos Reflexivo Infravermelho

O Módulo Sensor de Obstáculo Reflexivo Infravermelho possui um LED emissor de infravermelho e um fotodiodo, quando algum obstáculo/objeto passa no ângulo de reflexão dentro da distância ajustada, o sensor indica tal situação colocando a saída em nível lógico “LOW”. Esse módulo possui um trimpot para ajuste e regulagem da distância de detecção que fica entre 2cm a 30cm [6].

Na Figura 2 é possível visualizar as principais partes de um sensor de obstáculo infravermelho.

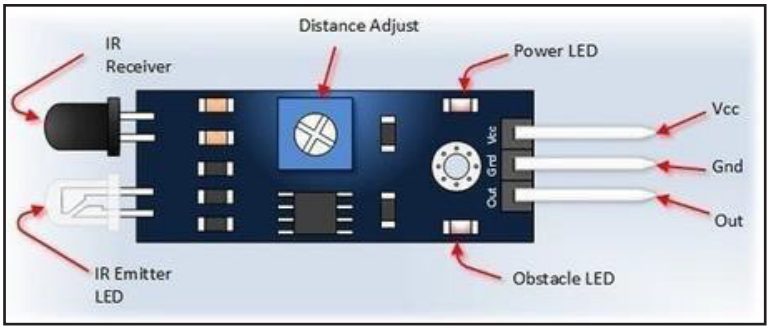

Figura 2. Partes de um Sensor de Obstáculo Reflexivo Infravermelho.

Caso algum objeto seja detectado pelo receptor o pino de saída do módulo terá valor igual a 0, caso contrário, seu valor será 5V. Dessa forma ele pode ser conectado diretamente ao Arduino.

A escolha da utilização deste sensor no protótipo se deve ao fato do seu baixo custo, aliado à sua alta disponibilidade, simples aplicação e ainda pelo fato do mesmo possuir um trimpot de ajuste da distância de detecção. No trabalho serão utilizados 5 sensores, cada um com o objetivo de detectar um tipo de moeda.

Display LCD 16x2 integrado com módulo I2C

Os Dislpays LCD’s são a forma mais simples de informar ao usuário valores obtidos através de medições e/ou cálculos realizados com o Arduino. Usualmente eles possuem 16 colunas e 2 linhas, com backlight (luz de fundo) azul e letras na cor branca.

Para sua conexão são 16 pinos, dos quais usamos 12 para uma conexão básica. Um inconveniente ao utilizar o display diretamente com o Arduino é o seu elevado número de conexões, e como no Arduino UNO o número de entradas é bastante limitado, uma alternativa bastante comum e eficaz é a utilização do módulo de comunicação I2C [3].

O protocolo I2C é largamente utilizado no desenvolvimento embarcado para realizar comunicações entre plataformas iguais ou diferentes utilizando apenas duas linhas seriais, como a de dados (SDA) e a linha de clock (SCL). Dessa forma, ao invés de se utilizar 6 entradas analógicas do Arduino na conexão básica, é possível a utilização de apenas duas portas analógicas, sem que o Display perca qualquer função. Para utilização tanto do display, quanto do módulo I2C junto ao Arduino é necessária a inclusão de 2 bibliotecas no sketch [5]:

- Wire.h;
- LiquidCrystal_I2C.h.

Na Figura 3 é possível visualizar tanto o display LCD (16x2) quanto o módulo I2C.

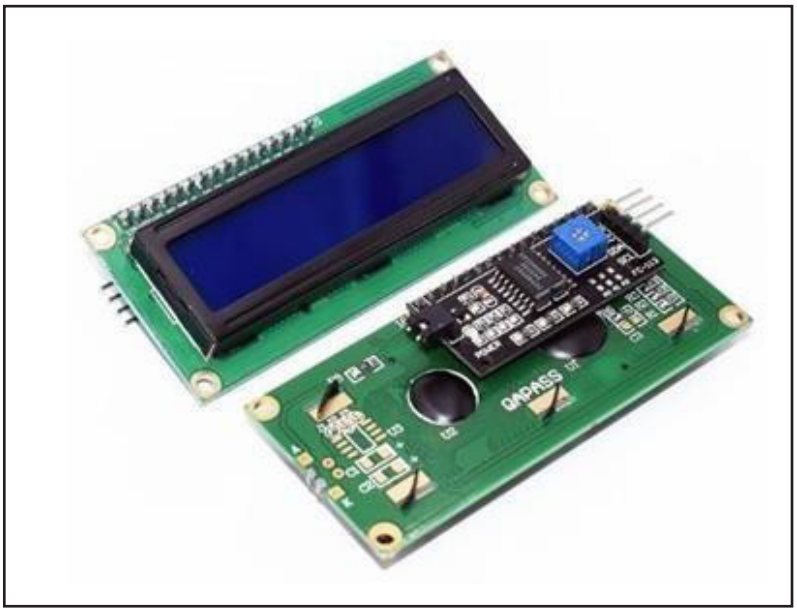

Figura 3. Display LCD (16x2) e módulo I2C.

Protótipo e código

Com base no objetivo proposto, foi desenvolvido o protótipo, conforme a Figura 4.

Figura 4. Protótipo confeccionado.

Seu princípio de funcionamento segue 5 etapas principais para atuação, descritas através da Figura 5.

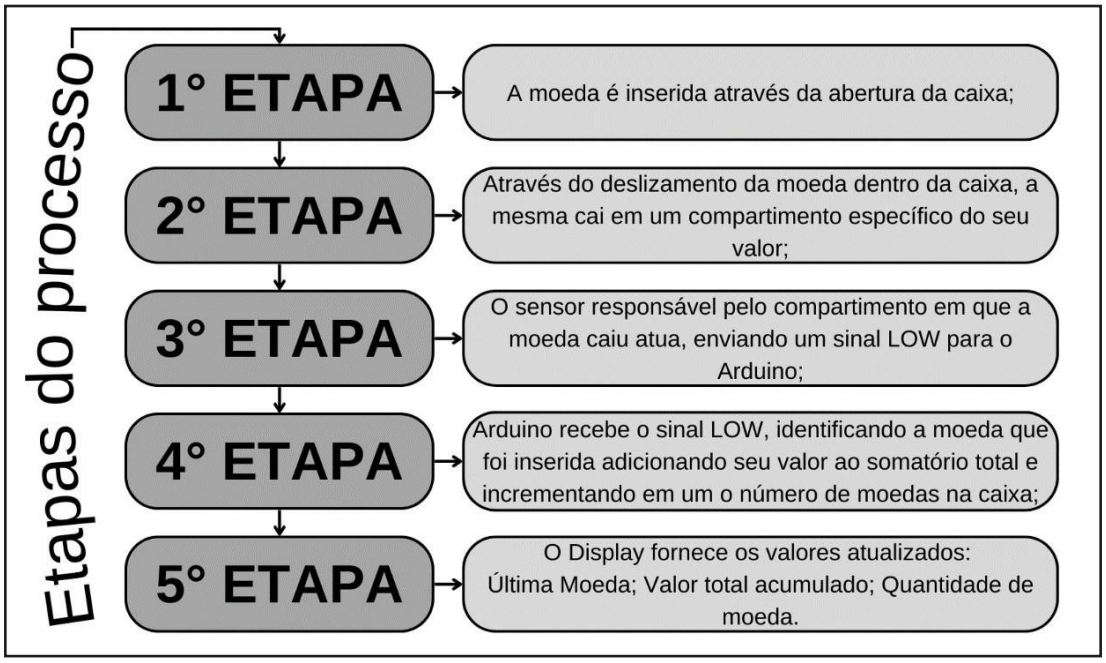

Figura 5. Etapas do processo do protótipo.

A abordagem mais simples para a construção envolveu a separação das moedas na parte mecânica. Por outro lado, o circuito eletrônico, ilustrado na Figura 6, foi projetado para alcançar os objetivos estabelecidos.

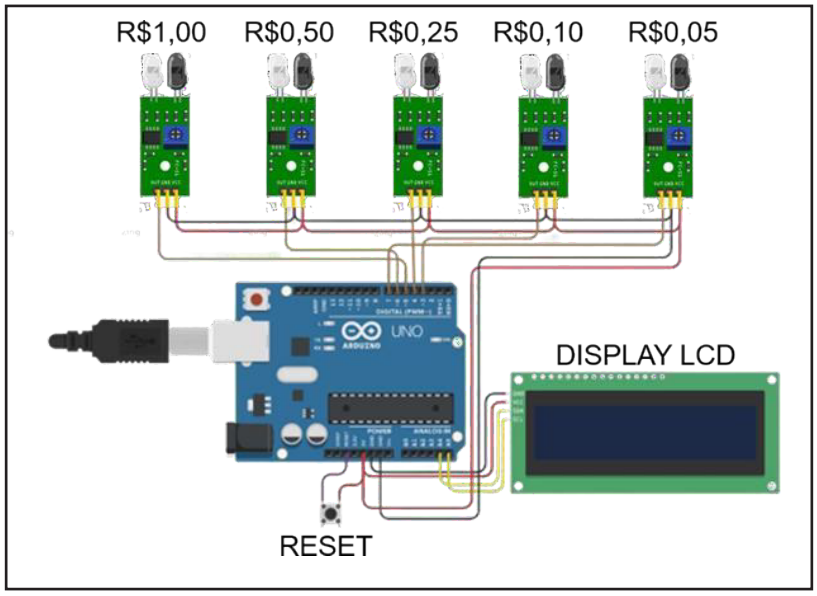

Figura 6. Esquemático da parte eletrônica do protótipo.

O código base para o Arduino será desmembrado a seguir, a fim de se realizar as devidas constatações sobre o mesmo. Na Figura 7 é possível visualizar o diagrama de blocos das etapas de funcionamento do código.

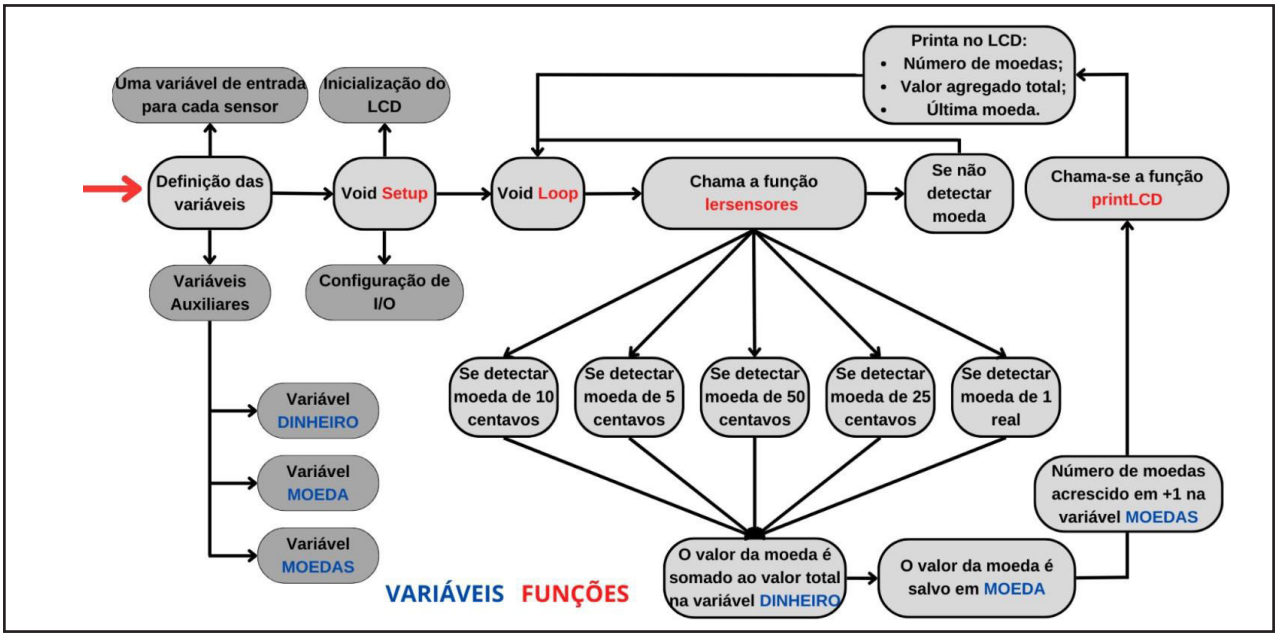

Figura 7. Lógica do código desenvolvido.

Para descrever o funcionamento do código o mesmo será subdividido em 5 partes, sendo a parte 1 demonstrado na Figura 8.

```
1   #include <Wire.h>
2   #include <LiquidCrystal_I2C.h>
3   LiquidCrystal_I2C LCD(0x27,16,2);
4
5   #define S100 5
6   #define S50  6
7   #define S25  4
8   #define S10  3
9   #define S5   7
10
11  float DINHEIRO = 0.0;
12  float MOEDA = 0.0;
13  int MOEDAS = 0;
```

Figura 8. Parte 1 do Código Base.

Nas linhas 1 e 2 são incluídas as bibliotecas necessárias para o funcionamento, sendo a “Wire.h” devido ao protocolo de comunicação I2C, e “LiquidCrystal_I2C.h” para a utilização do display.

Da linha 5 até a linha 9, tem-se a definição dos pinos com relação ao valor que eles representam. Já nas linhas 11, 12 e 13 tem-se as variáveis auxiliares, representando, respectivamente, “DINHEIRO”, o valor total acumulado, “MOEDA”, o valor da última moeda lida, e “MOEDAS”, o número total de moedas, conforme demonstrado na Figura 9.

```
14
15  void setup(){
16
17    LCD.init();
18    LCD.backlight();
19
20    pinMode(S100,INPUT);
21    pinMode(S50,INPUT);
22    pinMode(S25,INPUT);
23    pinMode(S10,INPUT);
24    pinMode(S5,INPUT);
25  }
```

Figura 9. Parte 2 do Código Base.

A função “void setup” será executada uma vez sempre que o Arduino for conectado a uma fonte de alimentação. Nas linhas 17 e 18 ocorre a inicialização do

display LCD. Entre as linhas 20 até a linha 24 tem-se a configuração dos pinos como entradas, sendo que cada pino corresponde a uma saída de cada sensor.

Na Figura 10 é apresentada a terceira parte do código.

```
26
27  void loop(){
28      lerSensores();
29      delay(250);
30  }
```

Figura10. Parte 3 do Código Base

```
31
32  void lerSensores(){
33    if(digitalRead(S100)==LOW){
34      MOEDA = 1.0;
35      DINHEIRO += MOEDA;
36      MOEDAS += 1;
37      novaMoeda();
38    }
39    else if(digitalRead(S50)==LOW){
40      MOEDA = 0.5;
41      DINHEIRO += MOEDA;
42      MOEDAS += 1;
43      novaMoeda();
44    }
45    else if(digitalRead(S25)==LOW){
46      MOEDA = 0.25;
47      DINHEIRO += MOEDA;
48      MOEDAS += 1;
49      novaMoeda();
50    }
51    else if(digitalRead(S10)==LOW){
52      MOEDA = 0.1;
53      DINHEIRO += MOEDA;
54      MOEDAS += 1;
55      novaMoeda();
56    }
57    else if(digitalRead(S5)==LOW){
58      MOEDA = 0.05;
59      DINHEIRO += MOEDA;
60      MOEDAS += 1;
61      novaMoeda();
62    }
63  }
```

Figura 11. Parte 4 do Código Base.

A função “void loop” irá se repetir durante todo o funcionamento do código, e na linha 28, tem-se a função “lerSensores”, sendo que após a conclusão dessa função

há um delay de 250ms e a função “lerSensores” se repete, reiniciando o processo.

A quarta parte do processo, é demonstrada pela Figura 11 e seu principal objetivo é realizar a alteração do valor das variáveis de acordo com o sensor acionado.

A partir do acionamento de algum dos sensores, será enviado um sinal “LOW” em sua respectiva entrada, e esse valor será detectado na função “if/else if” do valor correspondente. Caso uma moeda de 10 centavos seja detectada, a condição da função “else if” da linha 51 será verdadeira e, neste caso, será atribuído o valor 0.1 à variável “MOEDA”, de acordo com a linha 52. O valor da variável “DINHEIRO”, linha 53, sofrerá um incremento de 0.1, pois este é o valor da variável moeda, e a variável “MOEDAS” sofrerá um incremento de 1, linha 54.

Por fim, será executada a função “printLCD”. O processo descrito anteriormente é o mesmo variando apenas o valor da variável “MOEDA”, dependente da moeda detectada. A quinta e última parte do código corresponde a função “printLCD”, demonstrada na Figura 12.

```
64
65  void printLCD(){
66    LCD.clear();
67    LCD.setCursor(0,0);
68    LCD.print("Caixa: R$");
69    LCD.print(DINHEIRO);
70    LCD.setCursor(0,1);
71    LCD.print(MOEDAS);
72    if(MOEDAS/10<1){LCD.print("    ");}
73    else if(MOEDAS/100<1){LCD.print("   ");}
74    else if(MOEDAS/1000<1){LCD.print("  ");}
75    else{LCD.print(" ");}
76    LCD.print(": R$");
77    LCD.print(MOEDA);
78  }
```

Figura 12. Parte 5 do Código Base.

Assim que a função é executada, a linha 66, “LCD.clear”, tem como objetivo limpar a tela do display, e alinha seguinte posiciona o cursor na primeira linha da primeira coluna do display.

As linhas 68 e 69 escrevem no display o valor total acumulado, da seguinte

forma: “Caixa: R$ DINHEIRO”.

A linha 70 move o cursor para a segunda linha da primeira coluna do display, e da linha 71 até a linha 75 tem-se a declaração da variável “MOEDAS” no LCD, sendo que as linhas 72, 73, 74 e 75 são de configuração do número de espaços da variável “MOEDA”. Por fim, as linhas 76 e 77 printa no LCD o valor da última moeda lida pelos sensores.

Resumidamente este é o funcionamento geral do código a ser utilizado. Levando em consideração que, caso o botão de reset seja apertado em qualquer instante, haverá o reset do Arduino, zerando o valor de todas as variáveis.

TESTES E RESULTADOS

A partir da montagem do protótipo e da parte lógica do projeto, foi possível testar e coletar dados a fim de se mensurar a eficiência do sistema proposto. Foram realizadas 3 baterias de testes, onde cada bateria teve 3 moedas de cada valor, os valores coletados no teste 1 são demonstrados através da Tabela 2.

Tabela 2. Parâmetros do Protótipo.

TIPO DE MOEDA	ACERTOS(N°)	ERROS(N°)	EFICIÊNCIA(%)
R$0,05	3	0	100
R$0,10	3	0	100
R$0,25	3	0	100
R$0,50	3	0	100
R$1,00	3	0	100
Total	15	0	100

Na Figura 13 é possível visualizar o resultado do teste 1 tanto no display LCD quanto nos compartimentos das moedas. A sequência escolhida para o protótipo foi, da direita para a esquerda, a moeda de menor diâmetro para a de maior, seguindo a sequência: R$ 0,10, R$ 0,05, R$ 0,50, R$ 0,25, R$ 1,00.

Figura 13. Resultado do Teste 1.

Os valores coletados no teste 2 são demonstrados através da Tabela 3.

Tabela 3. Dados Coletados Teste 2

TIPO DE MOEDA	ACERTOS	ERROS	EFICIÊNCIA(%)
R$0,05	2	1	66,6
R$0,10	3	0	100
R$0,25	3	0	100
R$0,50	3	0	100
R$1,00	3	0	100
Total	14	1	93,33

Na Figura 14 é possível visualizar o resultado do teste 2 tanto no display LCD quanto nos compartimentos das moedas.

Figura 14. Resultado do Teste 2.

Neste teste houve um erro com uma moeda de R$ 0,05, sendo que a mesma não deslizou corretamente sobre a rampa, ficando presa e desta forma não chegou até o seu compartimento.

Os valores coletados no teste 3 são demonstrados através da Tabela 4.

Tabela 4. Dados Coletados Teste 3

TIPO DE MOEDA	ACERTOS	ERROS	EFICIÊNCIA(%)
R$ 0,05	3	0	100
R$ 0,10	2	1	66,66
R$ 0,25	3	0	100
R$ 0,50	3	0	100
R$ 1,00	2	1	66,6
Total	13	2	86,66

Na Figura 15 é possível visualizar o resultado do teste 3 tanto no display LCD quanto nos compartimentos das moedas.

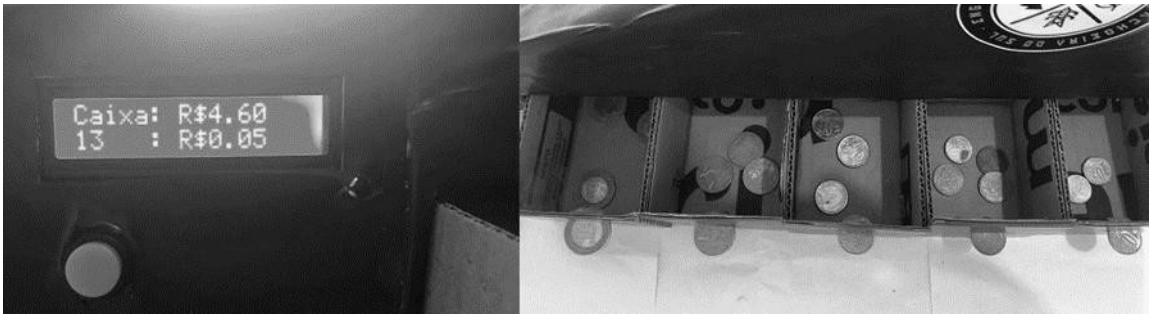

Figura 15. Resultado do Teste 3

O erro que aconteceu com a moeda de R$ 0,10 foi que a mesma caiu no compartimento da moeda de R$ 0,05 e o sensor não atuou, já com a moeda de R$ 1,00, apesar da mesma ter chegado corretamente até o seu compartimento, não houve a atuação do sensor.

O aproveitamento final dos três testes resultou em uma eficiência de 93,33%, com 42 acertos de 45 possíveis. O resultado foi satisfatório, atendendo o objetivo desejado,

observando que a maior problemática do projeto é com relação a parte mecânica, já que algumas moedas não deslizaram corretamente sobre a rampa, principalmente moedas de R$ 0,05, que são as mais leves dentre as cinco moedas utilizadas, dessa forma, ela encontra maior dificuldade para chegar até o seu compartimento.

Quanto a parte elétrica do circuito, a mesma teve um bom funcionamento, ocorrendo apenas um erro, devido ao não acionamento de um sensor. Uma limitação do trabalho quanto a leitura de moedas é que o sistema proposto não possui a capacidade de leitura de todas as moedas do sistema brasileiro, visto que existam algumas moedas que possuem tamanhos idênticos, como por exemplo a moeda de R$ 0,05 e de R$ 0,10 prateadas, como é possível visualizar na Figura 16.

Figura 16. Moedas Prateadas R$ 0,10 X R$ 0,05.

Ao total, atualmente, estão em circulação no Brasil 10 tipos diferentes de moedas, sendo que o trabalho proposto consegue identificar 5 modelos distintos. Outra limitação é com relação a alocação de moedas que deve ser realizada uma de cada vez.

CONCLUSÃO

Dessa forma o projeto cumpriu com os objetivos definidos inicialmente, pois sua eficiência foi superior a proposta na introdução, de 85%. Em contrapartida o projeto teve uma eficiência de, aproximadamente, 93% de acerto, sendo esse resultado devido a problemas mecânicos do circuito, visto que a separação é realizada devido as características construtivas das moedas e do protótipo.

Por fim, é sugerido para trabalhos futuros encontrar outros métodos para detecção de moedas semelhantes, além da criação de sistemas onde possa ser inserida mais de uma moeda por vez.

REFERÊNCIAS

[1] 3E Unicamp. (2023) Arduino: entenda mais sobre essa versátil plataforma. Disponível em:https://3eunicamp.com/arduino-entenda-mais-sobre-essa-versatil-plataforma/. Acesso em: 23 maio 2023.

[2] Thomsen, A. (2022) O que é Arduino, para que serve e primeiros passos. Disponível em:https://www.filipeflop.com/blog/o-que-e-arduino/. Acesso em: 9 jul. 2022.

[3] Unicamp. (2022) Arduino: entenda mais sobre essa versátil plataforma–.Disponível em: https://3eunicamp.com/arduino-entenda-mais-sobre-essa- versatil-plataforma/>. Acesso em: 9 jul. 2022.

[4] Equibancada. (2022) Arduino UNO. Disponível em: https://equibancada.com.br item/ Arduino- UNO.html. Acesso em: 9 jul. 2022.

[5] Embarcados. (2023) Comunicação I2C: o que é e como funciona. Disponível em: https:// embarcados.com.br/comunicacao-i2c/. Acesso em: 23 maio 2023.

[6] Silva, Lucas de H. (2022) Sensor Infravermelho de obstáculo com LM393. Disponível em: https://arduinoeasy.wordpress.com/aplicacoes/sensor-infravermelho-de-obstaculo-com-lm393/. Acesso em: 9 jul. 2022.

[7] Shopee. (2022) Display Lcd 16x2 + Módulo Adaptador I2c Arduino (soldado). Disponível em:https://shopee.com.br/Display-Lcd-16x2-M%C3%B3dulo-Adaptador-I2c-Arduino-%28soldado%29- i.291912630.8065288673. Acesso em: 9 jul. 2022.

Sobre os autores:

César Teixeira Pacheco

Graduando em Engenharia Elétrica, Universidade Federal de Santa Maria, Cachoeira do Sul, RS, Brasil
cesartp22@gmail.com - https://orcid.org/0000-0002-1295-1036

Gustavo Lenhardt Steffen

Graduando em Engenharia Elétrica, Universidade Federal de Santa Maria, Cachoeira do Sul, RS, Brasil
gustavo97_steffen@hotmail.com - https://orcid.org/0000-0001-7964-0948

Arthur Cordeiro Andrade

Graduando em Engenharia Elétrica, Universidade Federal de Santa Maria, Cachoeira do Sul, RS, Brasil
arthurcoand@gmail.com

Diogo Ribeiro Vargas

Professor Doutor do curso de Engenharia Elétrica, Universidade Federal de Santa Maria, Cachoeira do Sul, RS, Brasil
diogo.vargas@ufsm.br - https://orcid.org/0000-0002-0705-8112

Como citar este artigo

Pacheco, C. T., Steffen, G. L., Andrade, A. C., & Vargas, D. R. Moedeiro microcontrolado para ensino de disciplina de lógica e programação. *JESTA,* Cachoeira do Sul, (2) e74253, Disponível em: https://periodicos.ufsm.br/JESTA/article/view/74253 Acessado em: dia mês abreviado. ano. Disponível em: