



UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE TECNOLOGIA E GEOCIÊNCIAS
DEPARTAMENTO DE ENGENHARIA MECÂNICA
CURSO DE ENGENHARIA MECÂNICA

REBECA MARIA COUTO MOTA

**DESENVOLVIMENTO DE MONITORAMENTO DE PESAGEM COM A
UTILIZAÇÃO DE UM SISTEMA EMBARCADO NA INDÚSTRIA CALÇADISTA**

Recife
2024

REBECA MARIA COUTO MOTA

**DESENVOLVIMENTO DE MONITORAMENTO DE PESAGEM COM A
UTILIZAÇÃO DE UM SISTEMA EMBARCADO NA INDÚSTRIA CALÇADISTA**

Trabalho de Conclusão de Curso
apresentado ao Curso de Engenharia
Mecânica da Universidade Federal de
Pernambuco como requisito parcial para
obtenção do título de Engenheira
Mecânica

Orientador: Prof. Dr. José Rodrigues de Oliveira Neto

Recife

2024

Ficha de identificação da obra elaborada pelo autor,
através do programa de geração automática do SIB/UFPE

Mota, Rebeca Maria Couto.

Desenvolvimento de monitoramento de pesagem com a utilização de um sistema embarcado na indústria calçadista / Rebeca Maria Couto Mota. - Recife, 2024.

64 : il., tab.

Orientador(a): José Rodrigues de Oliveira Neto

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de Pernambuco, Centro de Tecnologia e Geociências, Engenharia Mecânica - Bacharelado, 2024.

Inclui referências, apêndices.

1. Sistema embarcado. 2. Monitoramento de matéria prima. 3. Sistema de aquisição de dados. 4. Indústria. 5. Plataforma arduino. I. Oliveira Neto, José Rodrigues de . (Orientação). II. Título.

620 CDD (22.ed.)

REBECA MARIA COUTO MOTA

**DESENVOLVIMENTO DE MONITORAMENTO DE PESAGEM COM A
UTILIZAÇÃO DE UM SISTEMA EMBARCADO NA INDÚSTRIA CALÇADISTA**

Trabalho de Conclusão de Curso
apresentado ao Curso de Engenharia
Mecânica da Universidade Federal de
Pernambuco, como requisito parcial para
obtenção do título de Engenheira
Mecânica.

Aprovado em: 15/08/2024

BANCA EXAMINADORA

José Rodrigues de Oliveira Neto (Orientador)
Universidade Federal de Pernambuco

João Paulo Cerquinho Cajueiro (Examinador Interno)
Universidade Federal de Pernambuco

Andrea Maria Nogueira Cavalcanti Ribeiro (Examinadora Externa)
Universidade Federal de Pernambuco

AGRADECIMENTOS

Gostaria de iniciar este agradecimento com uma menção honrosa ao meu avô, Manoel Couto. Mesmo não tendo a oportunidade de viver minha vida ao seu lado, sua presença e seu espírito de engenheiro sempre estiveram presentes em minha vida, inspirando minhas escolhas e conquistas.

À minha família, expresso toda minha gratidão pelo apoio constante nos meus estudos e pelo incentivo incessante para que eu sempre fizesse o meu melhor. Agradeço à minha mãe e minha avó, Fabiana Couto e Vilma Couto, que são minha base para tudo. A dedicação, o amor incondicional e a tranquilidade que vocês me deram durante essa jornada foram fundamentais para eu chegar até aqui.

Aos meus amigos, agradeço imensamente pela parceria e amizade ao longo desses anos de graduação. Desde o início, compartilhamos a tensão do ABI, o desespero pelas provas, mas também muitas comemorações e alegrias. Cada momento vivido com vocês tornou esta caminhada mais leve e memorável.

Minhas grandes amigas Raquel Castro e Manuela Nader que são fundamentais na minha vida e sempre estiveram ao meu lado em todas as situações. Sua amizade e apoio constante são inestimáveis.

Aos meus parceiros de curso, em especial Lucas Arnaud e Heitor Lins, que estiveram comigo durante toda a graduação, agradeço por tornarem essa jornada incrível, o companheirismo de vocês foi essencial.

Agradeço também a todas as empresas nas quais pude trabalhar e impactar, em especial a EIXO Consultoria, que foi a base do meu contato com a indústria e o mercado de trabalho. A experiência adquirida nesse período foi valiosa para minha formação profissional e pessoal.

E por fim, ao meu professor orientador, José Rodrigues de Oliveira Neto, que ao longo da minha graduação foi fundamental na minha escolha de aprofundamento na área de automação e me ajudou no desenvolvimento deste trabalho de forma magnífica. Sua orientação e apoio foram cruciais para o sucesso deste projeto.

Todos foram cruciais na minha jornada profissional e pessoal, deixo aqui meu singelo agradecimento por tudo que fizeram e fazem por mim.

RESUMO

Até os anos 1980, o desperdício na indústria era visto como inevitável e irrelevante para a gestão, sendo negligenciado nos resultados. No entanto, hoje é reconhecido como um problema que deve ser controlado e eliminado, começando pela compreensão de sua natureza. A gestão precisa da matéria-prima é crucial para reduzir desperdícios, equilibrando volumes e garantindo a qualidade do produto final. Para tal, são necessárias informações precisas, mas a coleta pode ser dificultada por protocolos de comunicação variados entre fabricantes de equipamentos. Os avanços em sistemas embarcados têm facilitado a integração e o fluxo de informações nos processos industriais. Logo este trabalho traz uma análise de uma indústria calçadista, na qual há uma divergência entre o consumo real e o teórico de matéria-prima e não existia uma coleta de informação precisa desses dados. Para resolver este problema foi desenvolvido um sistema de monitoramento de pesagem de matéria prima utilizando uma placa de desenvolvimento, Arduino, para isso foi necessário inicialmente entender o comportamento de um sistema de medição de pesagem, em seguida entender o funcionamento e características de um sistema embarcado e por fim como realizar a comunicação entre esses dois sistemas citados.

Por conseguinte, foram definidas as funcionalidades que precisavam ser atendidas para o processo produtivo. Os componentes eletrônicos foram selecionados pelo custo-benefício e testados individualmente, depois integrados e programados via Arduino IDE. Após o desenvolvimento da interface para exibir informações coletadas, uma caixa foi modelada em software CAD para fabricação. A etapa final envolveu a validação com pesos padrão e simulação em ambiente produtivo, garantindo a confiabilidade e aquisição de dados necessárias para as análises de desperdício.

Palavras-chave: Sistema embarcado; monitoramento de matéria prima; sistema de aquisição de dados; indústria; plataforma arduino.

ABSTRACT

Until the 1980s, waste in the industry has been seen as inevitable and irrelevant to management, thus being neglected in the results. However, today, it is recognized as a problem that must be controlled and eliminated, beginning with understanding its nature. Precise management of raw materials is crucial to reduce waste, balance volumes, and ensure the quality of the final product. For this, precise information is necessary, but its collection can be hindered by varying communication protocols among equipment manufacturers. Advances in embedded systems have facilitated the integration and flow of information in industrial processes. In this context, this study analyzed the scenario of a footwear industry where there is a discrepancy between the actual and theoretical consumption of raw materials, and there needed to be precise data collection for these figures.

Consequently, a monitoring system for raw material weighing was developed using Arduino. This development requires initially understanding the behavior of a weighing measurement system, then comprehending the operation and characteristics of an embedded system, and finally, how to achieve communication between these two systems. Therefore, the functionalities required for the production process were defined. The electronic components were selected based on cost-benefit analysis, tested individually, and then integrated and programmed via the Arduino IDE. After developing the interface to display the collected information, a case was modeled in CAD software for manufacturing. The final stage involved validation with standard weights and simulation in a productive environment, ensuring the reliability and data acquisition necessary for waste analysis.

Keywords: Embedded system; raw material monitoring; data acquisition system; industry; Arduino platform.

LISTA DE FIGURAS

Figura 1 - Representação ilustrativa do processo produtivo.	15
Figura 2 - Representação ilustrativa do sistema.	18
Figura 3 – Representação da parte que compõe de um sistema embarcado.	19
Figura 4 – Representação da relação da memória com os comandos.	20
Figura 5 – Representação do sistema de armazenamento de memória.	21
Figura 6 - Placa do Arduino UNO.	22
Figura 7 - IDE do Arduino.	23
Figura 8 – Funcionamento do protocolo I ² C.	24
Figura 9 – Conexões do protocolo UART.	25
Figura 10 – Conector DE-9.	25
Figura 11 – Configuração protocolo SPI com o mestre e os escravos.	26
Figura 12 – Balança B180 Lider.	28
Figura 13 – Display LD2051 Lider.	28
Figura 14 – Detalhamento do Protocolo de Comunicação Lider 1.	29
Figura 15 – Visualização dos dados no aplicativo Hercules.	30
Figura 16 – Simulação física do projeto.	31
Figura 17 – Arduino Mega 2560.	32
Figura 18 – Detalhamento das entradas do Arduino Mega 2560.	32
Figura 19 – Tensões e temperatura aceitáveis para o Arduino Mega.	32
Figura 20 – Módulo RTC DS 1307.	33
Figura 21 – Bateria CR2032.	33
Figura 22 – Pinos do modulo RTC DS1307.	33
Figura 23 – Display LCD 16x2 backlight azul.	34
Figura 24 – Pinos do Display 16x2.	35
Figura 25 – Módulo MicroSD.	36
Figura 26 – Pinos do modulo Micro SD Card.	36
Figura 27 – Fonte DC chaveada 9V.	37
Figura 28 – Representação de conexão do módulo Micro SD.	38
Figura 29 – Execução do teste realizado no módulo Micro SD.	39
Figura 30 – Representação de conexão do módulo RTC.	39
Figura 31 – Execução do teste realizado no módulo RTC.	40
Figura 32 – Representação de conexão do módulo LCD.	41

Figura 33 – Execução do teste realizado no módulo RTC.....	42
Figura 34 – Representação de conexão da serial da balança com arduino.....	43
Figura 35 – Representação de conexão dos botões	45
Figura 36 – Representação de todas as conexões do projeto.....	45
Figura 37 – Caixa vazia a esquerda e com os componentes a direita.	47
Figura 38 – Modelo físico do sistema completo.....	47
Figura 39 – Passos para conversão de TXT para Google Planilhas.....	48
Figura 40 – Planilha com junção dos dados.	48
Figura 41 – Planilha com dados e parâmetros completo.	49
Figura 42 – Modelo do Dashboard.	49
Figura 43 – Exemplificação de aprofundamento dos dados.	50
Figura 44 – Pesos padrões.	51
Figura 45 – Pesagem realizadas com cada peso padrão.	52
Figura 46 – Dados mostrado no display para cada peso padrão.....	52
Figura 47 – Arquivo gerado pelas pesagens com o peso padrão.	53
Figura 48 – Um dos registros de pesagem hora a hora.....	54
Figura 49 – Arquivo gerado pelas pesagens hora a hora durante dois dias	54
Figura 50 – Parâmetros definidos na câmera termográficas.	55
Figura 51 – Registro da temperatura inicial do sistema.	55
Figura 52 – Registro de balança sem energia.	56
Figura 53 – Registro do religamento do sistema	56
Figura 54 – Temperatura registradas no início e no final do teste.	57
Figura 55 – Arquivo gerado pelas pesagens hora a hora durante dois dias	57
Figura 56 – Ficha Técnica da caixa.....	64

LISTA DE FLUXOGRAMAS

Fluxograma 1 – Representação das etapas do sistema.....	18
Fluxograma 2 – Representação do código exemplo ReadWrite.....	38
Fluxograma 3 – Representação da adaptação do código exemplo ds1307.....	40
Fluxograma 4 – Representação do código exemplo HelloWorld	41
Fluxograma 5 – Representação do teste de comunicação.....	43
Fluxograma 6 – Representação da logica adaptada para a comunicação	44
Fluxograma 7 – Representação das etapas do sistema de botões	44

LISTA DE TABELAS

Tabela 1 – Detalhamento dos pinos do conector DE-9	25
Tabela 2 – Detalhamento das configurações padrões da balança.	29
Tabela 3 – Exemplificações de visualização do protocolo de comunicação	29
Tabela 4 – Exemplificações de visualização do protocolo de comunicação	30
Tabela 5 – Detalhamento dos pinos do módulo RTC	33
Tabela 6 – Detalhamento dos pinos do módulo LCD	35
Tabela 7 – Detalhamento dos pinos do modulo SD Card	36
Tabela 8 – Detalhamento das pesagens realizadas	51

SUMÁRIO

1	INTRODUÇÃO	14
1.1	OBJETIVO GERAL E ESPECÍFICOS	16
1.2	METODOLOGIA	16
1.2.1	Definição do Funcionamento do Projeto	17
2	FUNDAMENTAÇÃO TEÓRICA	19
2.1	SISTEMA EMBARCADO.....	19
2.1.1	Microcontrolador.....	21
2.1.2	Arduino	22
2.2	SISTEMA DE COMUNICAÇÃO.....	23
2.2.1	Protocolo I ² C	23
2.2.2	Protocolo UART.....	24
2.2.3	Protocolo RS 232.....	25
2.2.4	Protocolo SPI.....	26
3	DESENVOLVIMENTO DO PROJETO	28
3.1	INSTRUMENTO DE MEDIÇÃO.....	28
3.1.1	Configurações da comunicação do sistema de medição.....	29
3.2	VALIDAÇÕES INICIAIS	30
3.2.1	Comunicação com a balança	30
3.2.2	Simulação Computacional.....	31
3.3	SELEÇÃO DOS COMPONENTES.....	31
3.3.1	Arduino Mega 2560	32
3.3.2	Módulo RCT	33
3.3.3	Módulo LCD	34
3.3.4	Módulo Micro SD.....	35
3.3.5	Fonte de Energia	36
3.4	TESTES UNITÁRIOS E INTEGRAÇÃO DO PROTÓTIPO	37
3.4.1	Teste do Modulo Micro SD	37
3.4.2	Teste do Modulo RTC	39
3.4.3	Teste do LCD	40
3.4.4	Teste da Conexão Serial	42
3.4.5	Teste de Integração.....	44
3.5	CAIXA DE ALOCAÇÃO DO SISTEMA.....	46

3.5.1	Modelagem	46
3.5.2	Fabricação	47
3.6	DASHBOARD	47
3.6.1	Banco de Dados	47
3.6.2	Parâmetros	48
3.6.3	Disposição e Visualização dos Dados	49
4	RESULTADOS E DISCUSSÕES	51
4.1	VALIDAÇÃO DAS COMUNICAÇÕES DO SISTEMA	51
4.2	SIMULAÇÃO DO AMBIENTE PRODUTIVO	54
5	CONCLUSÕES	58
5.1	TRABALHOS FUTUROS	58
	REFERÊNCIAS	60
	APÊNDICE A – FICHAS TÉCNICAS DA CAIXA	64

1 INTRODUÇÃO

Ao longo dos anos da história industrial, o desperdício era frequentemente encarado como inevitável ou até mesmo como uma parte normal do processo produtivo, escapando ao controle dos gestores ou sendo considerado irrelevante para a gestão (MARTINS, 2013).

Contudo o desperdício, antes percebido durante os processos produtivos, mas negligenciado nos resultados, agora é reconhecido como um problema a ser controlado e reduzido, se não totalmente eliminado. Logo, compreender a natureza do desperdício é o primeiro passo para sua identificação e eliminação eficaz (OHNO, 1997).

A gestão da matéria-prima pode ser um aliado na busca da redução de desperdícios, visto que essa gestão envolve a necessidade de equilibrar os volumes ou quantidades desses materiais de forma precisa. Isso é crucial para garantir a uniformidade e a exatidão no consumo de matéria-prima, possibilitando o planejamento preciso da produção e seus custos, além de assegurar um padrão elevado de qualidade no produto final (ALVES, 2016).

Contudo para a gestão da matéria-prima e análise de desperdício é necessário informações para tomada de decisões. A coleta dessas informações nem sempre requer uma tarefa simples, os equipamentos de controle das máquinas apresentam fabricantes diferentes e com isso protocolos de comunicação diferenciados (COELHO, 2010).

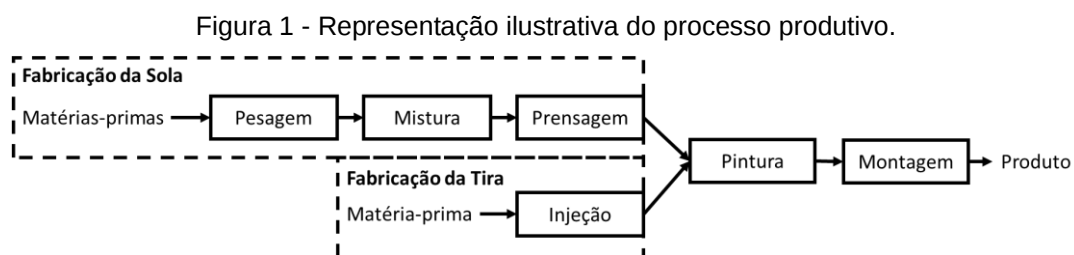
Os avanços tecnológicos com a Indústria 4.0 introduz novos conceitos que estão transformando a atual organização dos processos produtivos. As coletas de informações, anteriormente isolados ou até inexistente, estão se reconfigurando para formar um banco de dados integrado. Essa integração resulta em um incremento na eficiência, possibilitando um fluxo de insumos e produtos de maneira mais inteligente, garantindo uma análise mais qualificada do desperdício (CARDOSO, 2016).

Diante disso, os sistemas embarcados vêm possibilitando a inserção desses sistemas nos processos industriais da indústria 4.0, em processos que requerem determinada capacidade de processamento, bem como integrar padrões de comunicação de dados que tem como objetivo facilitar o fluxo das informações nos processos (RODRIGUES, PEDÓ, TEDESCO, 2013).

Frente a essa situação, foi verificado que na indústria calçadista, tendo como

objeto da análise a fábrica em que este trabalho foi desenvolvido, existe uma grande divergência entre o consumo teórico de matéria prima e o consumo que está realmente sendo utilizado no processo, evidenciando que existe desperdício ao longo do processo que apenas é identificado mensalmente, dificultando uma atuação mais rápida na causa de um possível erro operacional ou de desperdício no processo.

Para contextualizar onde o problema está inserido no processo industrial da indústria calçadista na Figura 1 será mostrado o processo produtivo para a fabricação de chinelos.



Fonte: Autor (2024).

Com isso é possível identificar que o problema se encontra no início do processo produtivo, impactando toda a linha produtiva em sequência. Adentrando ao processo de pesagem ele é subdividido em 2 partes: a pesagem de componentes químicos e pesagem de borracha. Esse trabalho será desenvolvido para a divisão da pesagem de borracha, visto que ao comparar as divergências de pesagem, entre o real e o teórico, o custo relacionado a pesagem de borracha foi maior.

Diante disso, para o processo de pesagem de borracha existem três variações de matéria prima sendo elas intituladas de Borracha 1502, Borracha Dyne e a Resina que tem como faixa de trabalho a ser pesada variando de 3 a 5kg a depender da formulação determinada pelo setor de qualidade.

A realização da pesagem dessas matérias primas é realizada por um operador em uma balança de forma continua durante o dia todo, não havendo uma ordem definida de qual matéria prima será pesada, a demanda de pesagem é de acordo com as variações da formulação que está sendo utilizada no processo, além disso em nenhum momento durante essa operação é registrado o valor pesado, com isso os dados referentes ao consumo das matéria prima só são registrados pelo processo seguinte ao realizar o recebimento do kit com todas as matérias primas necessárias para a mistura.

Logo, diante dessas informações foi visto como necessário ter no processo de pesagem de borracha uma aquisição de dados e melhor gestão dessa informação

para assim ser identificado o desperdício de matéria de forma mais rápida e eficiente. Como uma solução possível para esse problema, este trabalho propõe o desenvolvimento de um procedimento de pesagem de matéria prima usando um sistema embarcado no meio industrial.

1.1 OBJETIVO GERAL E ESPECÍFICOS

O objetivo geral deste trabalho é o desenvolvimento de um procedimento de coleta e monitoramento de pesagem de matéria prima na indústria calçadista. Esse procedimento atende a necessidade de monitorar pesagem no sistema produtivo e fornecer dados para análise de desperdícios. Para isso, o sistema deve ser capaz de se comunicar com o equipamento de medição e armazenar os dados de pesagem. Com isso, os dados coletados serão monitorados por uma plataforma visual para tomadas de decisão. Nesse sentido, os seguintes objetivos específicos deverão ser alcançados:

Como objetivos específicos podem-se citar:

- Desenvolver um módulo de comunicação entre o sistema de medição de peso e o sistema embarcado;
- Desenvolver o protocolo de registo dos dados provenientes da balança;
- Desenvolver o protótipo para instalação no meio produtivo;
- Criar uma interface para acompanhamento das pesagens coletadas.

1.2 METODOLOGIA

A pesquisa em questão classifica-se como aplicada, quanto à sua natureza, uma vez que está associada à resolução de um problema real. Do ponto de vista de seus objetivos, a pesquisa apresenta caráter exploratório, visto que se dedica à análise aprofundada de um tema específico para ampliar a compreensão em relação ao problema e conduzir uma investigação sobre o fenômeno de interesse. Inicialmente, a pesquisa foca na análise das características funcionais do objeto de estudo e nos processos de automação empregados, visando propor uma solução viável para o referido problema em questão. (NASCIMENTO; SOUSA, 2016)

No que diz respeito à abordagem metodológica, a pesquisa configura-se como combinada, em que o pesquisador emprega elementos das abordagens qualitativa e quantitativa em todas ou em algumas fases do processo de pesquisa. (NASCIMENTO;

SOUSA, 2016)

No que concerne aos procedimentos adotados, esta pesquisa caracteriza-se pelo emprego da bibliografia como fonte de conhecimento e aprofundamento das técnicas utilizadas no projeto, bem como pela condução de testes e desenvolvimento de protótipos para a solução do problema proposto. Logo, o foco deste trabalho reside na concepção de um sistema de monitoramento de pesagem de matéria-prima por meio da implementação de um sistema embarcado. (NASCIMENTO; SOUSA, 2016)

1.2.1 Definição do Funcionamento do Projeto

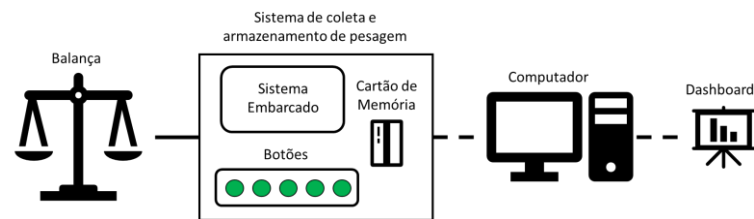
É necessário definir quais função são desempenhadas para o projeto. Com base nos objetivos específicos delineados, os métodos desenvolvidos para assegurar a plena realização das premissas são detalhados da seguinte maneira:

- A. Para desenvolver um método de comunicação entre o sistema de pesagem, a balança, e o sistema embarcado, arduino, foi utilizado um protocolo de comunicação, o RS 232;
- B. Para desenvolver uma interface entre homem máquina para a coleta de dados foi realizado um sistema com três botões para a seleção do tipo de matéria prima que está sendo pesado, um botão para envio das informações e mais um botão para ligar e desligar o sistema. Além disso também foi utilizado um display para comunicação com o usuário ao ser apertado os botões.
- C. Para desenvolver um método de armazenamento e transmissão dos dados, por no local da implementação existir uma limitação na propagação de sinais sem fio, foi utilizado um módulo de cartão SD para armazenamento foi utilizado cartões de memórias que salvam os dados em txt;
- D. Para obter informações sobre data e hora de coleta foi implementado um módulo RTC para aquisição desses dados;
- E. Para utilização do sistema em ambiente industrial foi criado uma caixa na qual contém armazenamento de todos os componentes eletrônicos e é resistente a condições externas de poeira e umidade que será inserido no processo produtivo;
- F. A criação da interface para acompanhamento das pesagens coletadas foi realizada por meio de *dashboard*, um painel de visualização, interativo por meio de planilhas do google e Looker Studio na qual os dados coletos no cartão de memória são transferidos para o banco de dados no drive;

G. Para a identificação de desvios de pesagem foi criado no dashboard uma condição com as condições de contorno aceitas no processo, considerando uma margem de 3% para mais e para menos, definindo assim automaticamente quando uma pesagem for realizada fora das especificações.

Com o detalhamento definido, é possível fazer uma representação demonstrando como está interligado fisicamente o projeto, Figura 2, que contém uma balança, diretamente conectada ao sistema embarcado, este possui a interface com operador através dos botões e display, além da interação com o cartão de memória para registo dos dados, que é conectado ao computador para armazenar as informações e serem colocados no *dashboard*.

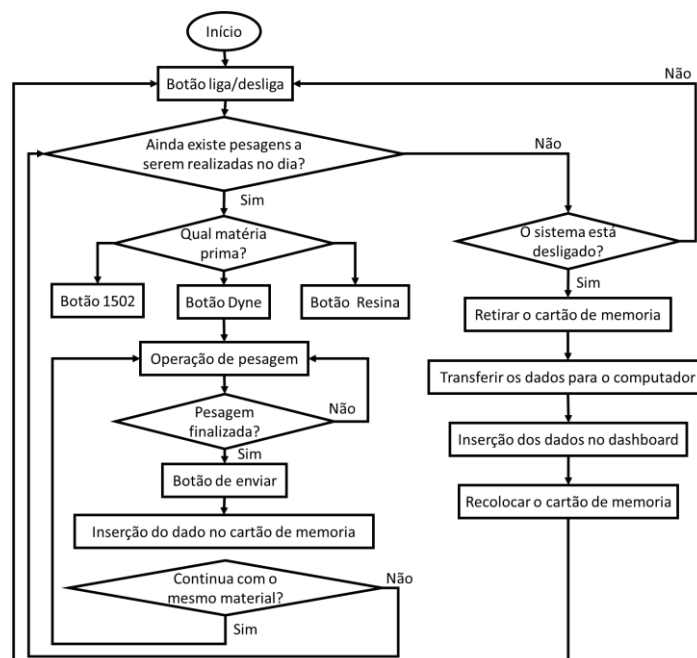
Figura 2 - Representação ilustrativa do sistema.



Fonte: Autor (2024).

Além da representação física do sistema é possível mostrar o princípio de funcionamento que será realizado pelo operador, Fluxograma 1, iniciando com a pesagem, as coletas de dados até a transferência dos dados para o computador e inserção no *dashboard*.

Fluxograma 1 - Representação das etapas do sistema.



Fonte: Autor (2024).

2 FUNDAMENTAÇÃO TEÓRICA

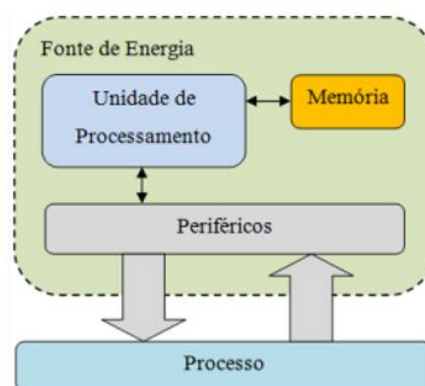
Neste capítulo é abordado as características do sistema embarcado, sua definição e característica, e a apresentado também as definições dos sistemas de comunicação necessário à solução proposta.

2.1 SISTEMA EMBARCADO

Um sistema embarcado é um sistema eletrônico e baseado em um microprocessador que é construído para controlar uma determinada atividade específica, diferentemente do PC (*personal computer*) na qual o usuário final tem a opção de modificação da funcionalidade do sistema. Em resumo tem-se que o sistema embarcado é direcionado a um objetivo pré-definido, podendo assim ele ser simples como o controle de TV ou complexo como o controle de funções de um carro (HEATH, 2003).

Com isso, o sistema embarcado tem como finalidade monitorar e/ou controlar processos. Para isso são necessários quatro elementos de hardware: a unidade de processamento, as memórias, os periféricos e uma fonte de energia para alimentar o sistema. A configuração desse sistema pode ser vista na Figura 3, além da parte de software envolvida na determinação do comportamento do sistema (MIYADAIRA, 2009).

Figura 3 – Representação da parte que compõe de um sistema embarcado.



Fonte: Fernandes (2017).

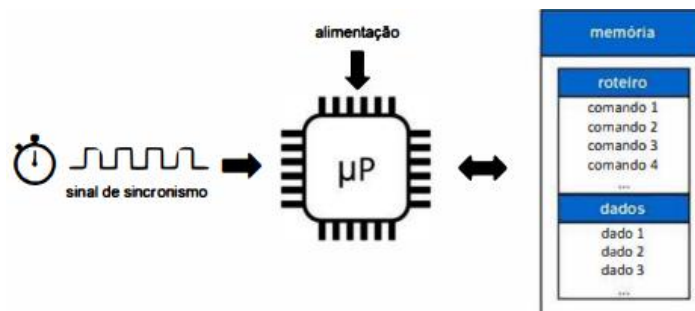
Como mostrado, na Figura 3, a Unidade de Processamento é um dos elementos do sistema embarcado, sendo ele o responsável por fornecer todo o processamento necessário para a execução das atividades do sistema. Um ponto a ser observado em relação ao processador ao escolhê-lo para um projeto é o tamanho e complexidade das atividades que serão realizadas pelo sistema, pois pode ocorrer

incompatibilidade em relação a capacidade de processamento se esses pontos não forem verificados (HEATH, 2003).

O processador (unidade de processamento), que coordena, executa e gerencia todos os recursos disponíveis, é um circuito digital sequencial, caracterizado pela dependência do estado atual e dos dados de entrada para determinar o próximo estado do circuito. Essa transição entre estados é controlada por um circuito de sincronismo, conhecido como *clock* e o intervalo de tempo entre dois estados consecutivos é denominado ciclo (ALMEIDA, 2017).

O roteiro, com as atividades que devem ser realizadas, é apresentado em um conjunto de comandos, onde cada comando é denominado instrução. Cada instrução pode estar associada a múltiplas operações a serem executadas. Alguns comandos têm a capacidade de realizar operações com valores armazenados na memória, sendo esses valores também referidos como dados, esse processo está representado na Figura 4 (ALMEIDA, 2017).

Figura 4 – Representação da relação da memória com os comandos.

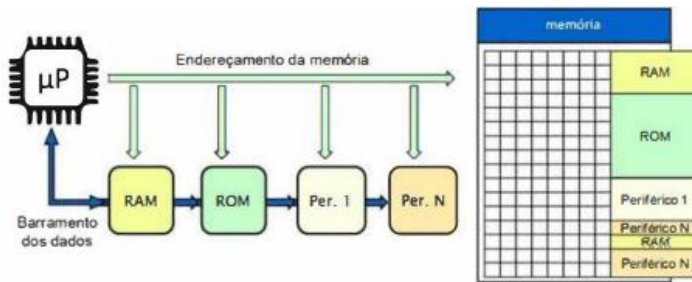


Fonte: Almeida (2017).

O segundo elemento do sistema embarcado são as memórias, sendo uma memória apenas de leitura (ROM, do inglês *Read-Only Memory*) e uma para o armazenamento de dados, esses podendo ser variáveis de programa, informação de status ou resultados parciais que são gerados durante o processamento do programa, sendo essa uma memória de gravação de leitura e escrita (RAM do inglês *Random Access Memory*) (HEATH, 2003).

As memórias do sistema podem ser divididas em vários sistemas de armazenamento possuindo uma variedade de tipos de memória e de ligação com dados de periféricos, como mostrado na Figura 5. Logo para identificação da posição da memória é necessário de um endereço, que se tratando de sistemas digitais é codificado em binário (ALMEIDA, 2017).

Figura 5 – Representação do sistema de armazenamento de memória.



Fonte: Almeida (2017).

Os periféricos são as interfaces com o mundo externo das unidades de processamento, trazendo ou enviando informações, através de sensores, atuadores e módulos. Existem diversos tipos de periféricos os de saída binária, que usam sinais de desligado/ligado, os de saídas seriais, na qual enviam e recebem dados usando pinos em modo serial, os de valores analógicos, que ao interagirem com o processador são convertidos para digitais, os displays, podendo ser LEDs a painéis alfanuméricos, e as saídas derivadas do tempo, sendo esses os contadores e temporizadores (HEATH, 2003).

Por último tem-se as fontes de energia, atualmente as mais usadas em sistemas embarcado são as fontes de alimentação que convertem a tensão alternada em contínua e as baterias (FERNANDES, 2017).

Um projeto com uma fonte inadequada pode resultar na introdução de ruídos indesejáveis no sistema, ocasionando reinicializações do dispositivo ou, em casos mais graves, danos permanentes. Portanto, a consideração cuidadosa e o dimensionamento preciso da alimentação são imperativos para garantir a integridade e estabilidade operacional do sistema embarcado (MIYADAIRA, 2009).

2.1.1 Microcontrolador

Os microcontroladores podem ser definidos sistemas digitais que possuem ao menos um núcleo de processamento, memórias de programa e de dados, e periféricos. De modo que, em muitos casos, tudo o que é necessário para usá-los em um sistema embarcado é adicionar firmware (HEATH, 2003).

Dito isso, o microcontrolador é um circuito integrado que incorpora sistemas eletrônicos com circuitos especializados e funções específicas em sua construção. Reconhecido como um dispositivo versátil, destaca-se por sua programação acessível, custo reduzido e dimensões compactas, sendo amplamente empregado em

diversas aplicações industriais, residenciais, entre outras (GREFIM, SILVA E CRUZ, 2018).

Um microcontrolador bastante utilizado e fabricado pela ATMEL, atualmente pertencente a empresa Microchip Technology, é o ATmega2560, com um circuito integrado CMOS (*Complementary metal-oxide semiconductor* do inglês metal-óxido-semicondutor complementar) de 8 bits e de baixa potência, que utiliza a arquitetura AVR, esta estrutura apresenta um conjunto simples e pequeno de instruções que para serem executados necessitam aproximadamente a mesma quantidade de tempo. O microcontrolador executa os comandos em um único ciclo de clock (ATMEL, 2014).

2.1.2 Arduino

O Arduino é um framework de prototipação de sistemas embarcados desenvolvida pela empresa Arduino, caracterizada pelo uso de hardware e software de código aberto. Sua origem remonta ao Instituto de Design de Interações de Ivrea no início dos anos 2000. Como exemplo de uma placa de desenvolvimento Arduino, tem-se o Arduino UNO mostrado na Figura 6 (ARDUINO, 2020).

Figura 6 - Placa do Arduino UNO.



Fonte: Arduino (2020).

A plataforma Arduino disponibiliza um software com uma interface de programação que facilita a compilação e o envio de código para qualquer placa compatível. Este software é compatível com sistemas operacionais Windows, Mac OS X e Linux. A sua IDE (*Integrated Development Environment* - Ambiente de Desenvolvimento Integrado), apresentada na Figura 7, oferece uma programação acessível e permite a escrita de código em uma linguagem própria, fortemente baseada em C/C++ (ARDUINO, 2020).

Figura 7 - IDE do Arduino.



Fonte: Autor (2024).

2.2 SISTEMA DE COMUNICAÇÃO

Devido à abrangente difusão da comunicação entre sistemas digitais que vai desde automação residencial até industrial, observa-se uma ampla variedade de padrões e protocolos para comunicação serial. A diversidade desses protocolos é necessária para garantir o desempenho ótimo em diferentes aplicações.

REFERENCIA

Um protocolo de comunicação consiste em um conjunto de normas que possibilita a transmissão de informações entre duas ou mais entidades dentro de um sistema de comunicação. Este protocolo estabelece as regras, sintaxe, semântica e sincronização da comunicação e a sua implementação pode ocorrer por meio de hardware, software ou por uma combinação de ambos (DAWOUD, 2022).

A comunicação serial se configura como uma solução relevante para sistemas embarcados, possibilitando a transmissão sequencial de dados, um bit por vez, por meio de um barramento ou canal de comunicação. Nos tópicos a seguir serão detalhados alguns desses protocolos (EXXACTA, 2022).

2.2.1 Protocolo I²C

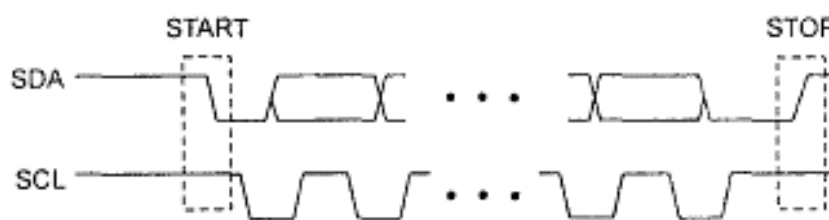
O protocolo I²C (*Inter-Integrated Circuit*) é um sistema de comunicação síncrona mestre-escravo (*master-slave*) desenvolvido pela PHILIPS, visando simplificar a interação entre dispositivos eletrônicos. Atualmente, esse protocolo é amplamente difundido e adotado por diversos fabricantes em uma variedade de produtos, incluindo drivers de LCD, memórias seriais, geradores de tom, conversores A/D e D/A, RTC (*real-time clock*), microcontroladores, entre outros (MIYADAIRA, 2009).

O barramento I²C consiste em duas linhas de comunicação: uma linha de dados (SDA) e uma linha de *clock* (SCL). A linha SCL é controlada pelo dispositivo configurado como mestre, sendo responsável pelo envio do sinal de *clock* ao dispositivo escravo (DAWOUD, 2022).

Este barramento adota a topologia *multi-master*, permitindo que mais de um dispositivo inicie uma comunicação (enviando o sinal de *clock*), desde que todos os outros dispositivos estejam configurados como mestre durante esse processo. Os dispositivos geralmente configurados como mestre são microcontroladores ou microprocessadores, dada a necessidade de capacidade de processamento necessária (MIYADAIRA, 2009).

O funcionamento do protocolo I²C é caracterizado pelo emprego de duas condições essenciais para marcar o início e o encerramento de uma comunicação. Essas condições são denominadas *START* e *STOP*, como demonstrado na Figura 8. A condição de *START* é determinada pela transição da linha de dados (SDA) do nível lógico '1' para o nível '0', ao passo que a linha de *clock* (SCL) mantém-se em nível '1'. Por outro lado, a condição de *STOP* é reconhecida quando a linha SDA transita do nível '0' para o nível '1', enquanto a linha SCL permanece em nível lógico '1' (MIYADAIRA, 2009).

Figura 8 – Funcionamento do protocolo I²C.



Fonte: Miyadaira (2009).

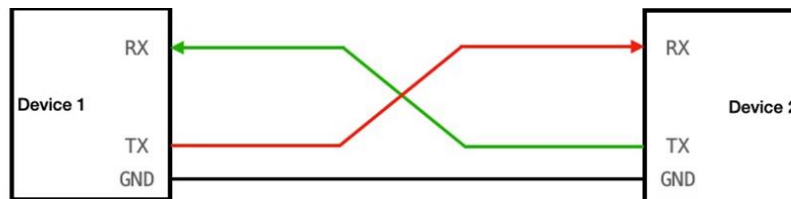
2.2.2 Protocolo UART

O protocolo UART, sigla para *universal asynchronous reception and transmission*, tem como função primordial a conversão de dados entre formatos paralelo e serial. Dispositivos que adotam esse protocolo são comumente empregados em padrões de comunicação serial, notadamente o RS-232C, RS-422 e RS-485 (ALVES, 2023).

O protocolo caracteriza-se por um sistema que não emprega *clock* ou sincronização na comunicação. Portanto, para estabelecer a comunicação utilizando

UART é imperativo que ambos os dispositivos estejam sincronizados na mesma velocidade, conhecida como *baud rate* e conectados de acordo com a Figura 9 (DAWOUD, 2022).

Figura 9 – Conexões do protocolo UART.



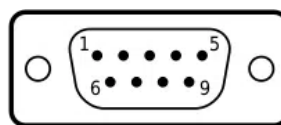
Fonte: Alves (2023).

2.2.3 Protocolo RS 232

O padrão RS232 é uma norma de comunicação serial empregada para a transmissão de dados por meio de uma linha específica. Desenvolvido pela *Electronic Industries Association* (EIA) nos Estados Unidos durante a década de 1960, esse padrão estabelece as diretrizes para a comunicação serial entre dispositivos (ALVES, 2023).

Os conectores empregados no padrão RS-232 são reconhecidos como conectores DE-9 ou DB-9, cujo mostra a Figura 10 e a Tabela 1. Esses conectores são habitualmente utilizados para a conexão do cabo RS-232 aos dispositivos de entrada/saída (I/O). Esse conector é caracterizado por possuir nove pinos e é empregado na conexão do cabo RS-232 a dispositivos diversos (DAWOUD, 2022).

Figura 10 – Conector DE-9.



Fonte: Souza (2022).

Tabela 1 – Detalhamento dos pinos do conector DE-9

Pino	Descrição	Função
Pino 1	Data Carrier Detedt (DCD)	Detecção de portadora de dados
Pino 2	Receive Data (RXD)	Recepção de dados serial
Pino 3	Transmit Data (TXD)	Transmissão de dados serial
Pino 4	Data Terminal Ready (DTR)	Terminal de dados
Pino 5	Signal Ground (GND)	Neutro/Terra
Pino 6	Data Set Ready (DSR)	Conjunto de dados
Pino 7	Request To Send (RTS)	Solicitação de envio
Pino 8	Clear To Send (CTS)	Limpa o envio
Pino 9	Ring Indicator (RI)	Indicador de anel

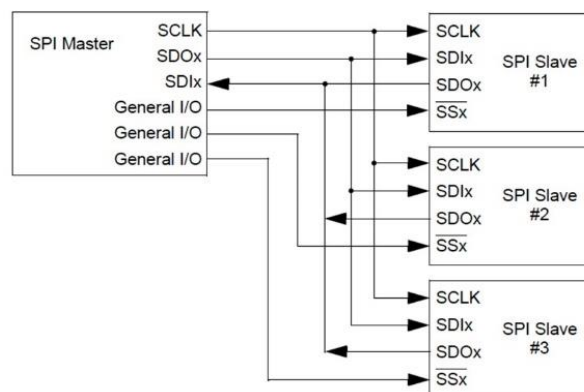
Fonte: Adaptado de Dawoud (2022).

2.2.4 Protocolo SPI

A interface SPI (*Serial Peripheral Interface*) foi concebida pela Motorola, configurando-se como um sistema de comunicação síncrona operando em *modo full-duplex*, modo no qual o transmissor e receptor podem atuar de maneira simultânea e em ambos os sentidos. Essa estrutura suporta um dispositivo mestre e um ou mais escravos conectados ao barramento. Atualmente, é amplamente adotada em diversos dispositivos devido à sua eficácia na transmissão de dados e à facilidade de implementação, encontrando aplicação em microcontroladores, sensores de temperatura, potenciômetros digitais, entre outros (MIYADAIRA, 2009).

A interface SPI compreende no mínimo quatro sinais distintos: MOSI (*Master Out Slave In*), utilizado pelo dispositivo mestre para transmitir dados aos escravos, MISO (*Master In Slave Out*), destinado à comunicação do escravo para o mestre, SCLK (*Serial Clock*), responsável pela sincronização das informações na comunicação serial e o SS (*Slave Select*), canal de seleção de escravo. O canal SS, no contexto do controle do dispositivo mestre, requer uma saída dedicada para cada escravo, e cada uma dessas saídas deve ser equipada com um resistor de *pull-up* para possibilitar a comunicação com os escravos, evitando conflitos por meio de sinais de nível baixo enviados pelo dispositivo controlador, como demonstrado na Figura 11 (DAWOUD, 2022).

Figura 11 – Configuração protocolo SPI com o mestre e os escravos.



Fonte: Alves (2023).

No funcionamento do protocolo SPI, os dispositivos configurados como escravo são selecionados pelo pino SS, em vez de possuírem um endereço. Essa característica, aliada à capacidade de operar em modo *full-duplex*, permite que a SPI alcance taxas de comunicação superiores à interface I²C. Contudo, devido à ausência de padronização, a seleção do dispositivo escravo pode ser realizada tanto em nível

lógico alto ('1') quanto em nível lógico baixo ('0'), sendo esta última a opção mais comum (MIYADAIRA, 2009).

3 DESENVOLVIMENTO DO PROJETO

Neste capítulo é apresentado as definições do funcionamento do projeto, assim como as simulações realizadas para validação, o protótipo e o modelo de *dashboard* para visualização das informações.

3.1 INSTRUMENTO DE MEDIÇÃO

Para a medição da pesagem foi utilizada a balança de modelo B180 da marca Lider¹, Figura 12, dentre as balanças existentes na fábrica a sua escolha foi baseada nas suas configurações de comunicação que são fundamentais para o desenvolvimento do projeto.

Figura 12 – Balança B180 Lider.



Fonte: B180 Balança -Descrição Técnica.

A balança selecionada é um equipamento de fabricação nacional construído com célula de carga central, com capacidade de 20 kg, fabricado em aço carbono, equipada com pés ajustáveis, display de LED vermelho com alimentação universal.

O indicador digital associado a balança de modelo LD1050², mostrado na Figura 13, também é da marca Lider e possui canal de comunicação baseado no padrão RS232 e pode ser interconectado a uma impressora.

Figura 13 – Display LD2051 Lider.



Fonte: LD1050 Terminal -Descrição Técnica.

¹ Datasheet B180: <https://www.liderbalancas.com.br/wp-content/uploads/produtos/files/1651278494.pdf>

² Datasheet LD1050: <https://www.liderbalancas.com.br/wp-content/uploads/produtos/files/1450359761.pdf>

3.1.1 Configurações da comunicação do sistema de medição

Em relação a comunicação serial da balança segundo o seu manual existe alguns parâmetros pré determinados como padrão que podem ser modificados se necessário, nesse projeto foi utilizada a configuração padrão da balança B180-L1050 que possui as seguintes configurações vista na Tabela 2.

Tabela 2 – Detalhamento das configurações padrões da balança.

Função	Parâmetro Padrão
Protocolo de comunicação	Líder 1
Baud Rate (Serial para computador)	9600
Endereço (Apenas com o protocolo Modbus)	1 a 255
Frequência de transmissão	Continua
Intervalo de transmissão	1000ms
Checksum	Protocolo sem Checksum
Paridade e Bits de Transmissão	Paridade None, Data Bits 8
Stop Bits de Transmissão	1 Stop Bit
Nível da Bateria	Protocolo sem o nível da bateria

Fonte: Adaptado de B180-LD1050.

Como pode ser visto na Tabela 2, na função de protocolo de comunicação está determinado o parâmetro Líder 1, esse protocolo é descrito no manual da balança e na Figura 14, é mostrado todo o detalhamento desse protocolo.

Figura 14 – Detalhamento do Protocolo de Comunicação Líder 1.

Início	Peso Líquido\ Qnt. De Peças		Status	Final
0x01	7 x ASCII	0x20	E	0x02

Início	0x01 (hex) Byte indica início de transmissão
Peso Líquido	Peso líquido em ASCII incluindo ponto decimal ou indicador de peso negativo.
Separador	0x20 (hex) Byte de separação de dados.
Status	E – Peso Estável I – Peso Instável P – Valor de Pico (F110=1) M – Peso médio calculado (F110=2) G – Imprime
Final	0x02 (hex) Byte indica final da transmissão.

Fonte: Adaptado de B180-LD1050.

Além disso o manual mostra algumas exemplificações de como será mostrado ao usuário a informação em casos de pesagem estável, excesso de peso positivo e negativo, demonstrados na Tabela 3.

Tabela 3 – Exemplificações de visualização do protocolo de comunicação.

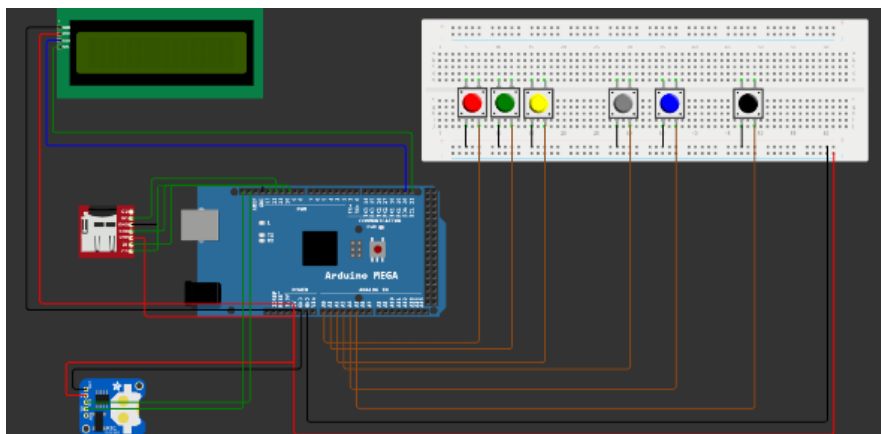
Situação	Visualização de Leitura
Peso Estável de 10kg (ASCII)	<SOH> 010.000 E<STX>
Excesso de peso positivo	<SOH> SSSSSS E<STX>
Excesso de peso negativo	<SOH> NNNNNN E<STX>

Fonte: Adaptado de B180-LD1050.

3.2.2 Simulação Computacional

Para a validação do projeto, antes da realização de um protótipo, foi realizado por meio computacional uma simulação no site Wokwi⁴. Nessa simulação foi verificado quais módulos seriam necessários e qual placa do Arduino que melhor atenderia a demanda solicitada, na Figura 16 mostra a esquematização realizada no site.

Figura 16 – Simulação física do projeto.



Fonte: Autor (2024).

Na Figura 16 pode-se observar que está sendo usado um display, para verificação e validações dos dados, além de ter como objetivo uma interação como o usuário. O módulo da cor vermelha se qualifica como um módulo de cartão de memória na qual serão alocados os dados do sistema, o módulo em azul refere-se ao módulo de marcação de data e hora usado para rastreabilidade das pesagens que serão registradas. Por fim seis botões para interação do usuário, três deles para as matérias primas, um de envio e outro de reset e o ultimo de ativação do sistema.

Além da validação estrutural do projeto, no site também foi realizado o primeiro esboço do código do projeto para validação da ideia.

3.3 SELEÇÃO DOS COMPONENTES

Nessa sessão será descrito os componentes que foram utilizados para desenvolvimento do projeto, com as especificações necessárias para atender as funcionalidades do sistema que será desenvolvido.

⁴ Disponível em: <https://wokwi.com>

3.3.1 Arduino Mega 2560

Para o projeto foi selecionado a placa do Arduino Mega 2560, mostrado na Figura 17, devido a quantidade e variedade de entradas que são necessárias para o devido funcionamento do projeto.

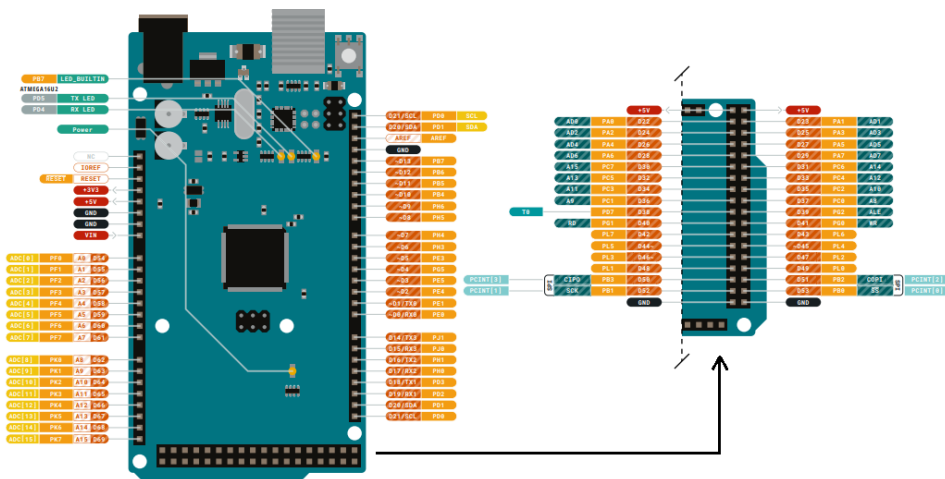
Figura 17 – Arduino Mega 2560.



Fonte: Arduino (2020).

Com origem no microcontrolador ATmega2560, o dispositivo é caracterizado por 54 pinos designados para funções de entrada e saída digital, dos quais 15 são disponíveis para operação como saídas PWM. Além disso, apresenta 16 entradas analógicas e 4 portas dedicadas à comunicação serial, como detalhado na Figura 18.

Figura 18 – Detalhamento das entradas do Arduino Mega 2560.



Fonte: Adaptado de Arduino (2020).

Além disso para seu devido funcionamento as tensões e temperatura máximas e mínimas que ele pode operar devem ser seguidas de acordo com as especificações do datasheet para nenhum tipo de risco aos componentes, Figura 19.

Figura 19 – Tensões e temperatura aceitáveis para o Arduino Mega

Symbol	Description	Min	Typ	Max	Unit
V _{IN}	Input voltage from VIN pad / DC Jack	7	7.0	12	V
V _{USB}	Input voltage from USB connector	4.8	5.0	5.5	V
T _{OP}	Operating Temperature	-40	25	85	°C

Fonte: Arduino (2020)

3.3.2 Módulo RCT

O Módulo relógio de tempo real (RTC, do inglês *Real-Time Clock*) DS1307⁵, Figura 20, é um dispositivo de relógio em tempo real dotado de um calendário, se comunicando através do protocolo I²C.

Figura 20 – Módulo RTC DS 1307.



Fonte: Recicomp (2023).

Para seu correto funcionamento, além da alimentação principal da placa, é imprescindível a utilização de uma bateria CR2032, Figura 21, tal configuração visa assegurar a retenção dos dados armazenados mesmo em situações de falta de energia.

Figura 21 – Bateria CR2032.



Fonte: Recicomp (2023).

O modulo DS1307, representado na Figura 22, possui 12 pinos, porém para esse projeto foram utilizados apenas quatro deles, descritos na Tabela 5.

Figura 22 – Pinos do modulo RTC DS1307.



Fonte: Adaptado de Oliveira (2020).

Tabela 5 – Detalhamento dos pinos do módulo RTC.

Pino	Descrição	Função
SCL	Serial Clock	Sincronização dos dados
SDA	Serial Data	Transmissão e recepção dos dados
VCC	Tensão de Corrente Contínua	3.3V ou 5V
GND	Terra/Neutro	-

Fonte: Adaptado de DS1307 Datasheet (2022).

⁵ Datasheet DS1307: <https://semiconductors.es/DS1307.html>

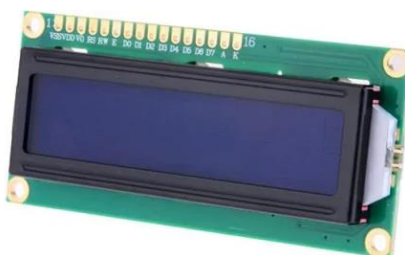
Para operar devidamente, o referido módulo, além das interconexões de hardware, torna-se imprescindível implementar a instalação da biblioteca RTCLib.h, durante a programação na IDE do Arduino.

Como dito anteriormente, este módulo fornece dados precisos relativos a horas, minutos, segundos, dia, data, mês e ano, realizando ajustes automáticos no término de meses com menos de 31 dias, bem como correções associadas à anos bissextos e versatilidade quanto ao formato de horário de 12 ou 24 horas. Sendo estas as características fundamentais para o projeto em questão.

3.3.3 Módulo LCD

O módulo Display de Cristal Líquido (LCD, do inglês *Liquid Crystal Display*), mostrado na Figura 23, constitui-se como um dispositivo destinado à apresentação de caracteres alfanuméricos. Possui um cristal líquido de matriz de pontos, operando sob o controle de um microprocessador de 4 ou 8 bits e para sua utilização é necessária uma fonte de alimentação de 2,7V a 5,5V.

Figura 23 – Display LCD 16x2 backlight azul.

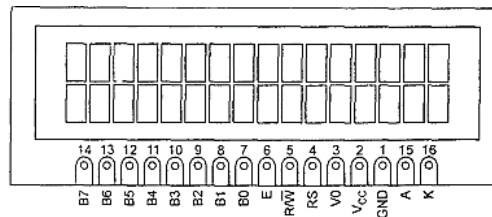


Fonte: Recicomp (2023).

O módulo LCD está disponível em várias dimensões, sendo escolhida a configuração de 16x2, o HD44780⁶, para este projeto específico, uma vez que esta escolha atende a funcionalidade do projeto. Logo, é crucial compreender a disposição e a conexão dos 16 pinos no módulo LCD a fim de assegurar a adequada integração no contexto do presente trabalho, que pode ser visto na Figura 24 e detalhado na Tabela 6.

⁶ Datasheet HD44780: <https://blogmasterwalkershop.com.br/arquivos/datasheet/Datasheet%20HD44780.pdf>

Figura 24 – Pinos do Display 16x2.



Fonte: Miyadaira (2009).

Tabela 6 – Detalhamento dos pinos do módulo LCD

Pino	Descrição	Função
A	Anodo	Anodo da luz de fundo
K	Cadoto	Catodo da luz de fundo
VCC	Tensão de Corrente Contínua	3.3V ou 5V
GND	Terra/Neutro	-
V0	-	Contraste
RS	-	Seleção de comando
R/W	Leitura/Escrita	Leitura e escrita
E	-	Ativação/desativação do display
B0 a B7	-	Barramento de dados

Fonte: Adaptado de Datasheet LCD [s.d.].

O módulo demanda a integração da biblioteca LiquidCrystal.h durante a etapa de programação na IDE do Arduino. Esta biblioteca, universalmente compatível com todas as placas Arduino.

3.3.4 Módulo Micro SD

O módulo Micro Cartão de Dados de Segurança (*Micro SD Card*, do inglês *Security Data Micro Card*) destina-se à realização de operações de leitura e escrita em cartões SD, este dispositivo é aplicável na preservação de históricos de valores provenientes de diversas fontes, tais como entradas de dados, registros de módulos GPS, sensores de temperatura, distância, entre outros. Devido essa capacidade de armazenamento de dados e a não disponibilidade de internet estável no ambiente do projeto do módulo MOD100717⁷, Figura 25, foi escolhido.

⁷ Datasheet MOD100717: <https://cdn.awsli.com.br/945/945993/arquivos/Datasheet-MicroSD-Module.pdf>

Figura 25 – Módulo MicroSD.



Fonte: Recicomp (2023).

Os pinos de saída do Módulo Micro SD Card apresentam funções distintas, como mostrado na Figura 26 e detalhado na Tabela 7.

Figura 26 – Pinos do módulo Micro SD Card.



Fonte: STRAUB (2018).

Tabela 7 – Detalhamento dos pinos do modulo SD Card.

Pino	Descrição	Função
VCC	Tensão de Corrente Contínua	3.3V ou 5V
GND	Terra/Neutro	-
CS	-	Habilita/Desabilita dispositivos específicos no barramento SPI
MOSI	<i>Master OUT Slave IN</i>	Entrada SPI do Módulo
SCK	<i>Serial Clock</i>	Sincronização dos dados
MISO	<i>Master IN Slave OUT</i>	Saída SPI do Módulo

Fonte: Adaptado de Datasheet MicroSD.

As especificações do módulo são definidas por uma comunicação SPI e realiza operações de leitura e escrita em sistemas de alocação de arquivos.

Por fim, para o adequado funcionamento do módulo em questão requer, além das conexões físicas de *hardware*, a integração à programação na IDE do arduino das bibliotecas Sd.h e SPI.h.

3.3.5 Fonte de Energia

Neste projeto é utilizada a Fonte de Alimentação Chaveada DC de 9V, acompanhada de um conector P4⁸, como mostra a Figura 27. Essa fonte opera em modo bivolt, adaptando-se automaticamente à energia de entrada, proporcionando

⁸ Disponível em: <https://www.makerhero.com/produto/fonte-dc-chaveada-9v-1a-plug-p4/>

versatilidade para operar em ambientes com tensões de 127V ou 220V e sua corrente de saída nominal é de 1A.

Figura 27 – Fonte DC chaveada 9V.



Fonte: RECICOMP.

3.4 TESTES UNITÁRIOS E INTEGRAÇÃO DO PROTÓTIPO

Nesta fase, todos os componentes físicos necessários para a concretização do protótipo estavam disponíveis. Para avaliação de desempenho, cada módulo foi submetido a testes individuais utilizando códigos exemplos das bibliotecas de cada módulo. Posteriormente, ocorreu a integração dos módulos, culminando na montagem e codificação final do projeto.

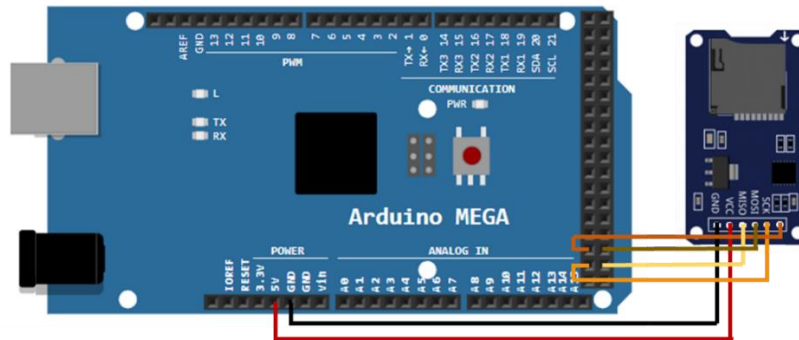
3.4.1 Teste do Módulo Micro SD

O primeiro passo é a montagem física com os componentes, para este teste foram necessários os seguintes componentes:

- Arduino Mega 2560 e seu cabo de conexão;
- Módulo Cartão SD;
- Cartão Micro SD e seu adaptador;
- Jumpers de conexão fêmea-macho;

Com isso, os componentes foram conectados em seus respectivos pinos de acordo com o descrito no datasheet de cada componentes como pode ser visto na Figura 28.

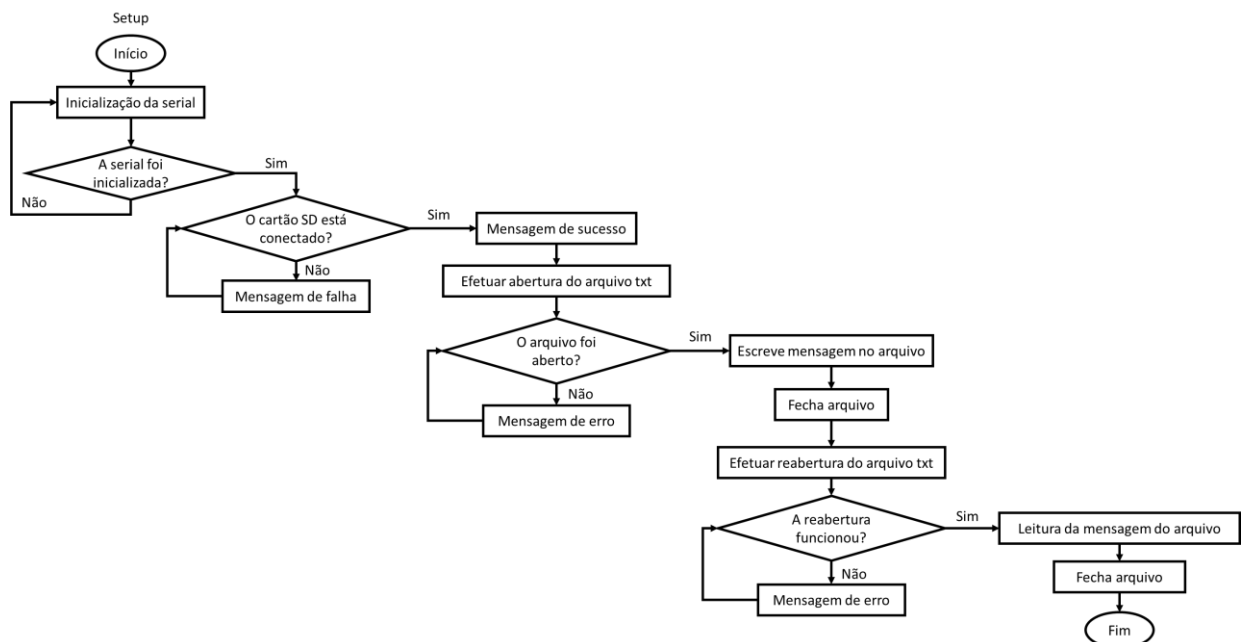
Figura 28 – Representação de conexão do módulo Micro SD.



Fonte: Autor (2024).

Em seguida, na IDE do arduino foi colocado um código já disponível na IDE ao instalar a biblioteca SD.h para a testagem do módulo SD, podendo ser acessado indo em Arquivos->Exemplos->SD e seleccionando o código de teste/exemplo ReadWrite, representado pelo Fluxograma 2.

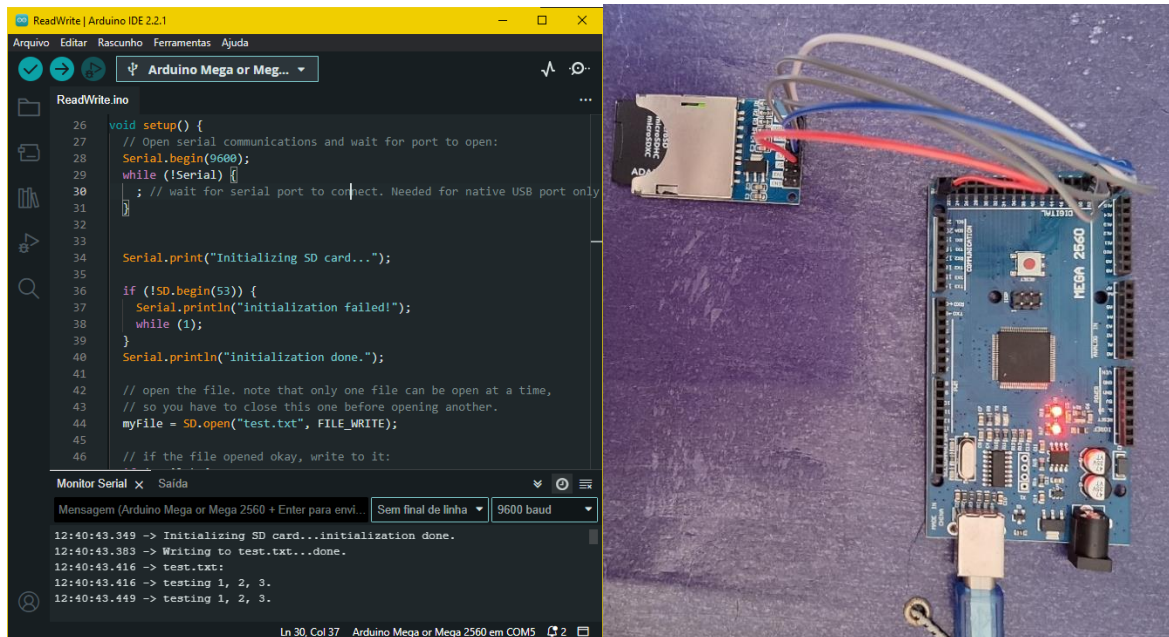
Fluxograma 2 – Representação do código exemplo ReadWrite.



Fonte: Autor (2024).

Por fim, foi gravado o código na placa Arduino Mega e verificado seu correto funcionamento, como mostra a Figura 29.

Figura 29 – Execução do teste realizado no módulo Micro SD.



Fonte: Autor (2024).

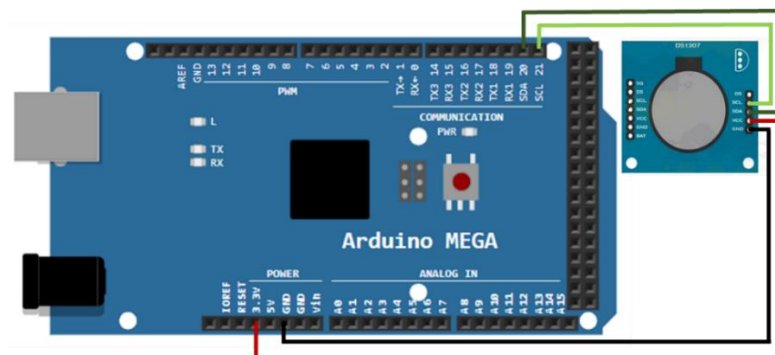
3.4.2 Teste do Modulo RTC

Da mesma maneira do módulo Micro SD, o primeiro passo é a montagem física com os componentes, para essa testagem foram necessários os seguintes componentes:

- Arduino Mega 2560 e seu cabo de conexão;
- Módulo RTC DS1307;
- Bateria CR2032;
- Jumpers de conexão fêmea-macho;

Os componentes foram conectados em seus respectivos pinos de acordo com o descrito no datasheet de cada componentes, como pode ser visto na Figura 30.

Figura 30 – Representação de conexão do módulo RTC.

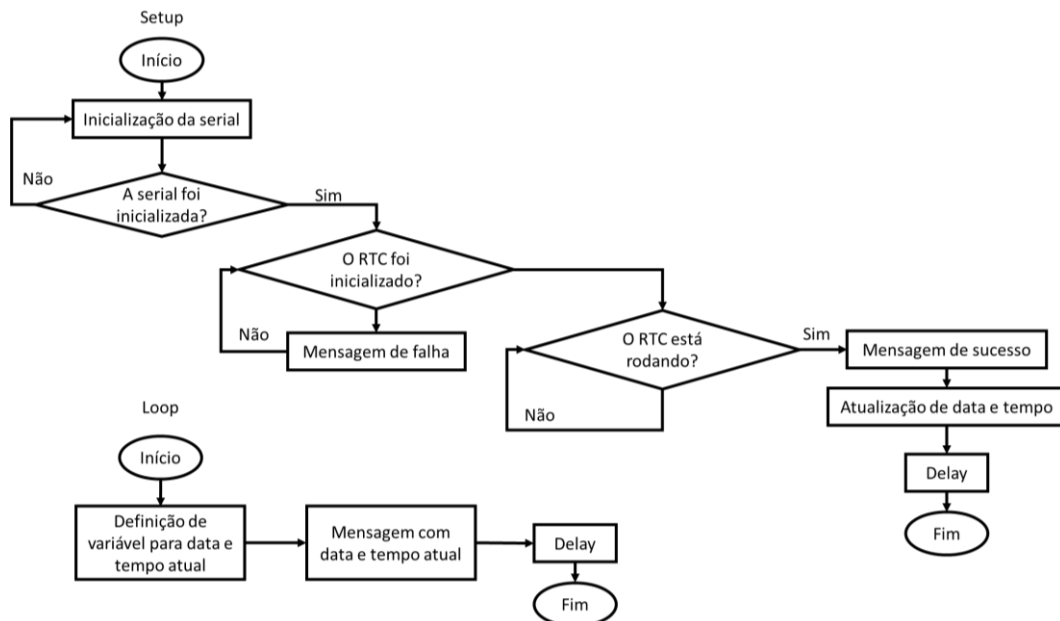


Fonte: Autor (2024).

Em seguida, na IDE do Arduino instalado a biblioteca RTCLib.h e utilizado uma

adaptação do código exemplo ds1307, representado pelo Fluxograma 3.

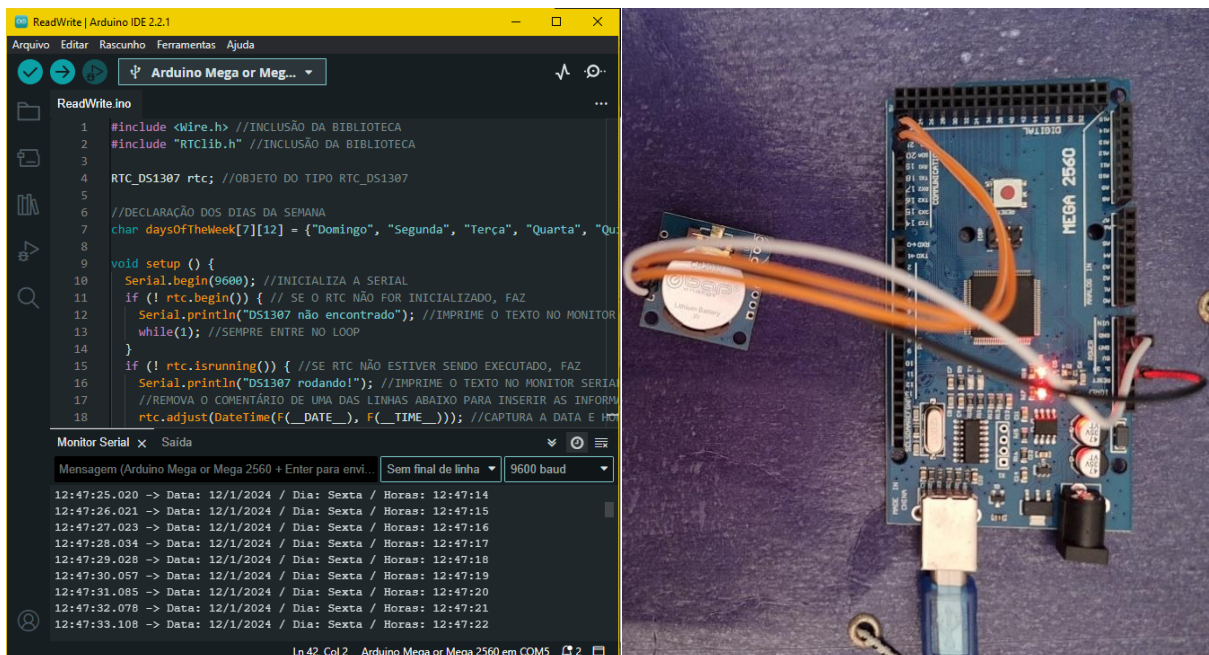
Fluxograma 3 – Representação da adaptação do código exemplo ds1307.



Fonte: Autor (2024).

Por fim, foi gravado o código na placa Arduino Mega o arduino e verificado seu correto funcionamento, na Figura 31 é mostrado esse momento.

Figura 31 – Execução do teste realizado no módulo RTC.



Fonte: Autor (2024).

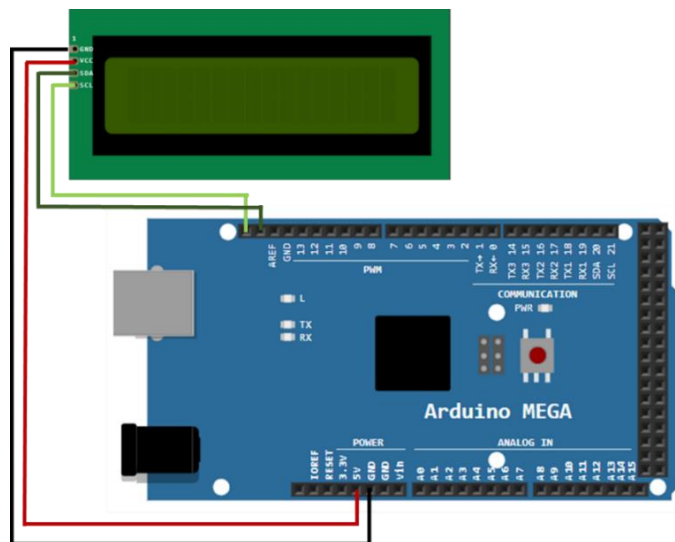
3.4.3 Teste do LCD

Seguindo a mesma metodologia, inicialmente é realizada a montagem do sistema físico, para essa montagem foram necessários os componentes abaixo:

- Arduino Mega 2560 e seu cabo de conexão
- Módulo LTC 16x2;
- Módulo I²C;
- Jumpers de conexão fêmea-macho;

Para esse conjunto foi necessário o módulo I²C para o protocolo de comunicação com o display, neste caso foram comprados os módulos separados e soldados para utilização. Em seguida, componentes foram conectados em seus respectivos pinos de acordo com o descrito no datasheet de cada componentes, como pode ser visto na Figura 32.

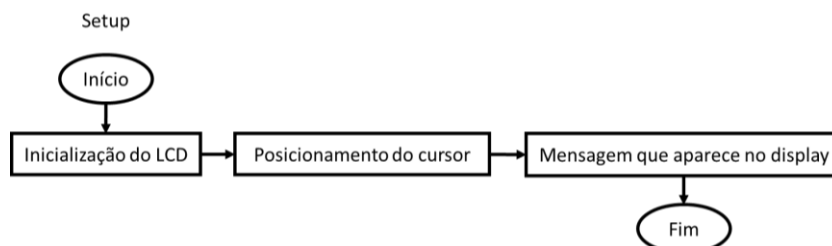
Figura 32 – Representação de conexão do módulo LCD.



Fonte: Autor (2024).

Por fim, na IDE do Arduino foi colocado um código-base já disponível na IDE ao instalar a biblioteca LiquidCrystal_I2C.h para a testagem do módulo SD, podendo ser acessado indo em Arquivos->Exemplos->LiquidCrystal I2C e selecionar o código de testagem HelloWorld, representado no Fluxograma 4.

Fluxograma 4 – Representação do código exemplo HelloWorld

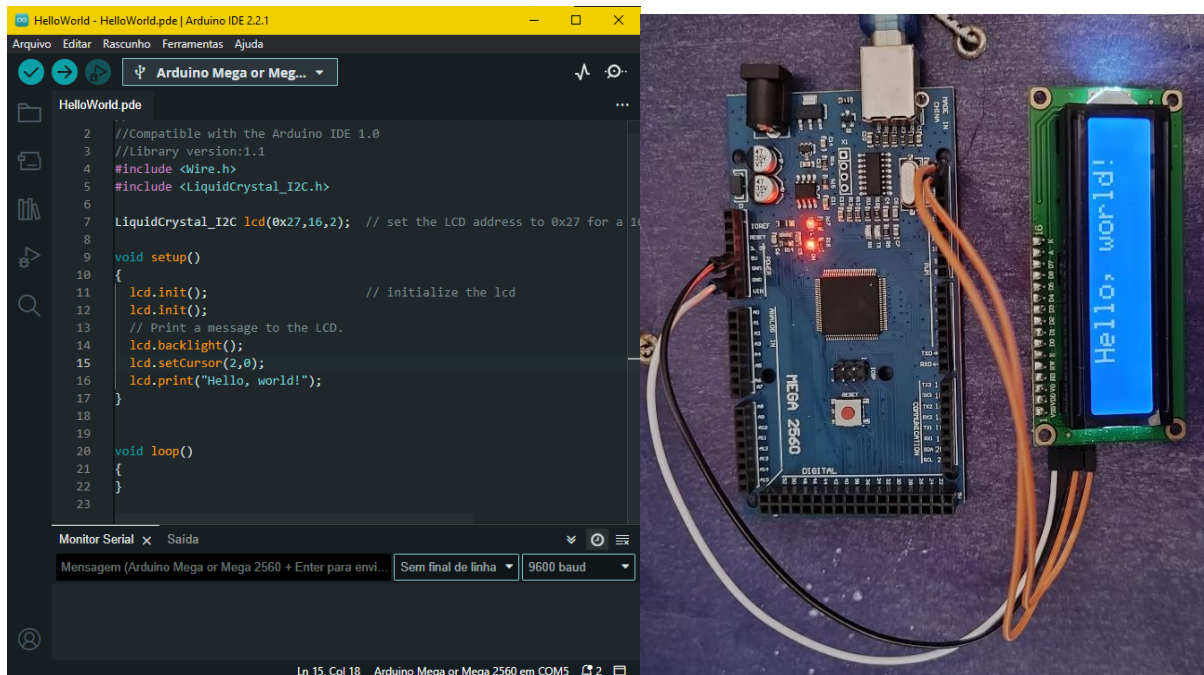


Fonte: Autor (2024).

Um ponto importante a ser tratado é que no módulo I²C existe um potenciômetro e ele deve ser ajustado para melhor visualização das informações no display, na Figura 33 mostra o teste realizado já com o potenciômetro ajustado

corretamente.

Figura 33 – Execução do teste realizado no módulo RTC.



Fonte: Autor (2024).

3.4.4 Teste da Conexão Serial

Para a testagem da comunicação a primeira verificação necessária é saber se a tensão de saída da balança está entre 5V e 9V, que é o faixa de tensão permitida no arduino para não causar dano ao componente.

Com isso foi utilizado um osciloscópio e testado as saídas na balança, na qual foi encontrado que a tensão de saída da balança é 5V, valor dentro do permitido, e também foi descoberto que a conexão serial da balança o pino 2, TX, e o pino 3, RX, se encontravam invertidos do que descrito no datasheet, como consta na sessão 2.3.3 (Protocolo RS 232) desse trabalho.

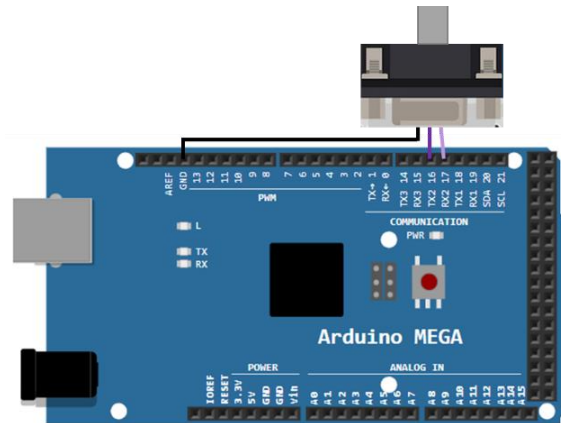
Seguindo a mesma metodologia dos tópicos anteriores foi realizada a montagem do sistema físico, para essa montagem foram necessários os componentes abaixo:

- Arduino Mega 2560 e seu cabo de conexão
- Jumpers de conexão fêmea-macho
- Balança

Os jumpers foram conectados diretamente dos pinos de numeração 2,3 e 5 da entrada DB9 da balança para os pinos TX, RX e GND do Arduino de acordo com o

teste feito com o osciloscópio, como pode ser visto na Figura 34.

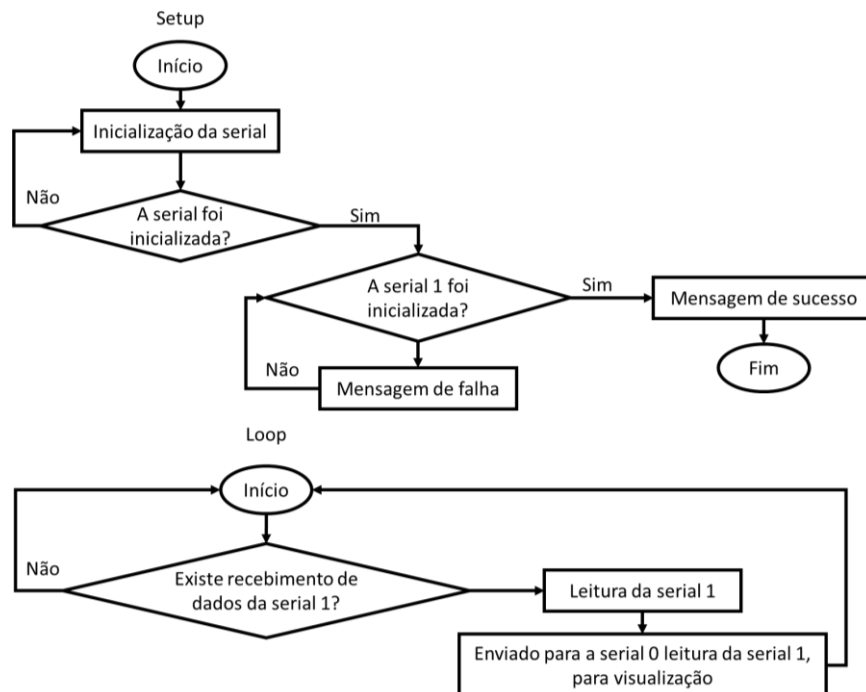
Figura 34 – Representação de conexão da serial da balança com arduino.



Fonte: Autor (2024).

Para a testagem da comunicação, não se tem um código base, então a estratégia foi realizar um código simples apenas para essa testagem na IDE do Arduino, que segue a lógica no Fluxograma 5.

Fluxograma 5 – Representação do teste de comunicação.



Fonte: Autor (2024).

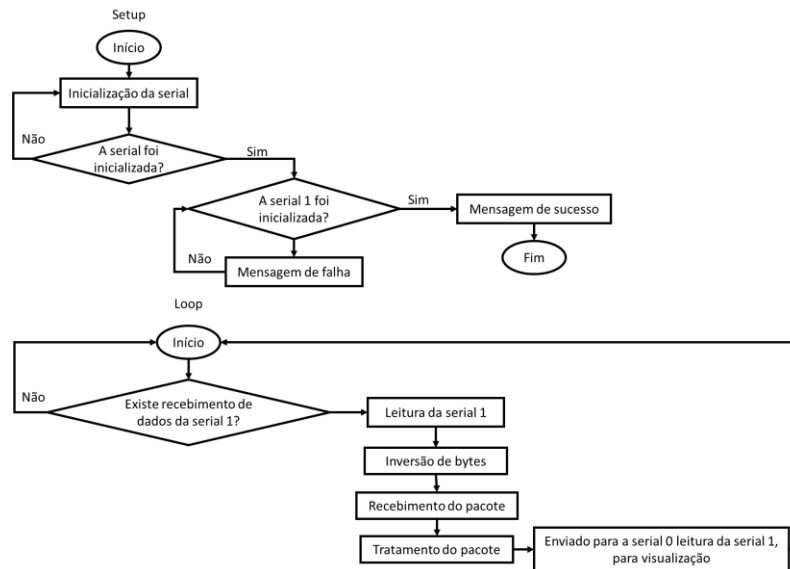
Porém nessa leitura foi verificado que os dados que estavam sendo enviados seus bits estavam invertidos, foi desenvolvido no código uma lógica para corrigir esse problema e testado novamente.

A partir disso metade dos dados foram corrigidos e a outra parte por meio de testagens com diversas pesagens foi verificado que existia um padrão de deslocamento de bits no range de 0 kg a 9kg, e outro padrão a partir de pesos maiores

de que 10kg.

Então novamente foi desenvolvido no código uma lógica que abrangesse todas essas situações encontradas na comunicação considerando que cada pesagem gera um pacote de dados, ficando o código final com a lógica do Fluxograma 6, que foi passado para o arduino, testado e obtido o sucesso na leitura das pesagens.

Fluxograma 6 – Representação da logica adaptada para a comunicação.

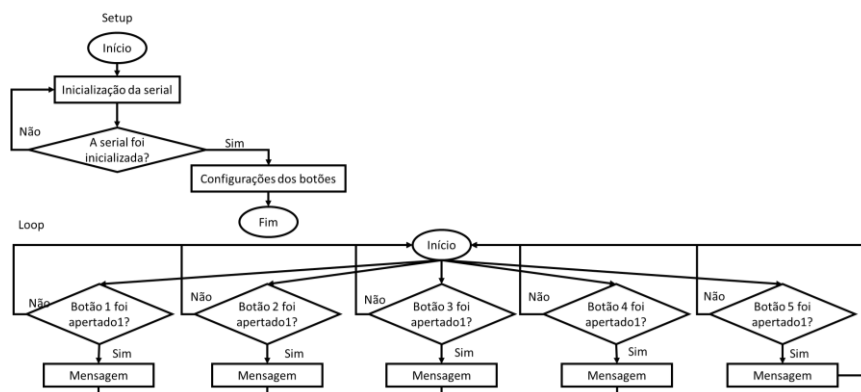


Fonte: Autor (2024).

3.4.5 Teste de Integração

Com todos os módulos testados, a próxima etapa é a realização da junção dos códigos de testes deles, e a montagem final do protótipo, que consiste em todos os módulos já citados além de um sistema com botões para a coleta dos dados enviados pela balança, para o teste dos botões foi realizado um código para confirmação que eles funcionam. No Fluxograma 7 é mostrado a lógica para esse teste.

Fluxograma 7 – Representação das etapas do sistema de botões.



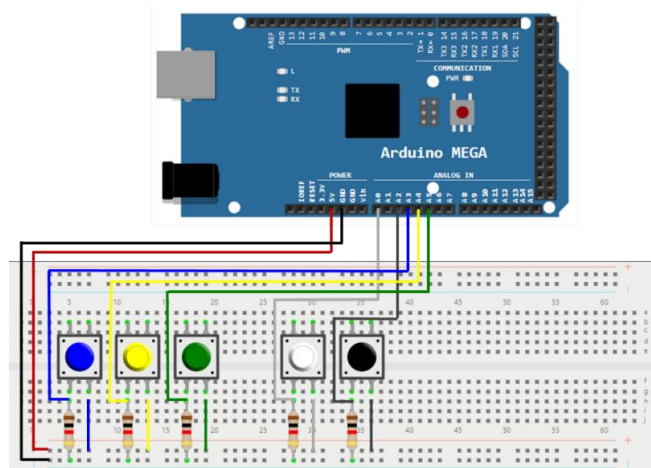
Fonte: Autor (2024).

Então, iniciando pelas montagens dos botões, são necessários os seguintes componentes:

- Protoboard;
- Botões (x5);
- Resistências;
- Jumpers fêmea-macho e macho-macho;

Logo tem-se a seguinte configuração física do protótipo, Figura 35.

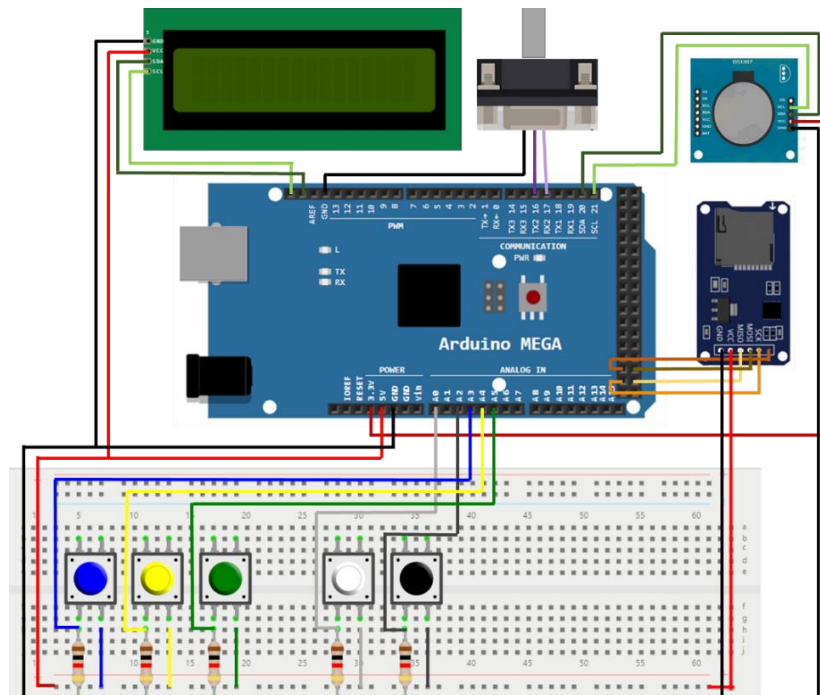
Figura 35 – Representação de conexão dos botões



Fonte: Autor (2024).

Por fim, juntando os módulos a configuração dos botões, o protótipo final apresenta a forma mostrada na Figuras 36.

Figura 36 – Representação de todas as conexões do projeto.



Fonte: Autor (2024).

O código⁹ é dividido em quatro partes: Definições, Funções, Setup e Loop.

Nas Definições foram adicionados as variáveis globais e definições de nomenclatura de pinos.

Na Funções tem-se algumas funções criadas para o funcionamento do projeto, a função de checagem de funcionamento do RTC, de checagem de funcionamento do módulo de cartão SD, de atualização de data e hora no display, de atualização de pesagem no display e pôr fim a função de escrita dos dados no arquivo txt.

Na parte do Setup temos a inicialização dos módulos RTC, cartão SD, Display, a inicialização e configuração dos botões e da serial.

E por fim temos a parte do Loop no qual tem-se o código para a atualização da data e hora no display a cada minuto, o código de acionamento dos botões e suas respectivas funções após seu acionamento.

3.5 CAIXA DE ALOCAÇÃO DO SISTEMA

Nesta seção, são discutidas as etapas de desenvolvimento do compartimento que protege todo o circuito eletrônico de sujeira, vibrações e umidade, além de oferecer uma aparência agradável ao projeto. Foi levado em consideração fatores como o tamanho e a forma do circuito, a facilidade de montagem e desmontagem, e a acessibilidade a componentes e conexões. É abordado o processo de design e fabricação do case, incluindo a modelagem 3D, e a montagem final do produto.

3.5.1 Modelagem

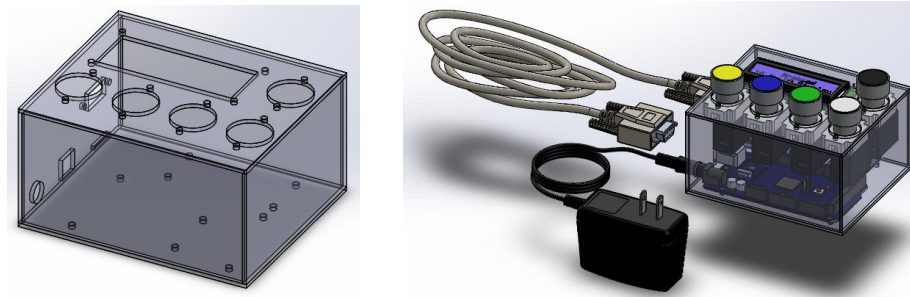
Para o desenvolvimento do case, foi necessário utilizar ferramentas virtuais que permitissem o desenvolvimento do projeto com as dimensões adequadas. Para isso foi utilizado o Solidworks¹⁰ para cumprir com esse propósito.

O design do compartimento foi projetado de forma a encaixar os componentes eletrônicos, a modelagem final pode ser verificada através das Figura 37 e suas fichas técnicas estão em na sessão de Apêndice A.

⁹ Disponível em: https://github.com/Rebeca-Couto04/DESENVOLVIMENTO-DE-MONITORAMENTO-DE-PESAGEM-COM-A-UTILIZA-O-DE-UM-SISTEMA-EMBARCADO/blob/main/Projeto_vf.ino

¹⁰ Disponível em: <https://www.solidworks.com/pt-br>

Figura 37 – Caixa vazia a esquerda e com os componentes a direita.



Fonte: Autor (2024).

3.5.2 Fabricação

Para a fabricação deste protótipo, foi usufruído da disponibilidade de um centro de usinagem no local de realização do projeto e a caixa, Figura 38, foi confeccionada de acordo com as medidas preestabelecida com o uso do poliacetal como material de confecção devido sua alta resistência para um ambiente industrial e de possíveis choque mecânicos.

Porém também foi estudado o método de fabricação via impressão 3D, por ser uma forma rápida, barata e fácil de prototipação de produtos e de rápida confecção para caixas futuras.

Figura 38 – Modelo físico do sistema completo.



Fonte: Autor (2024).

3.6 DASHBOARD

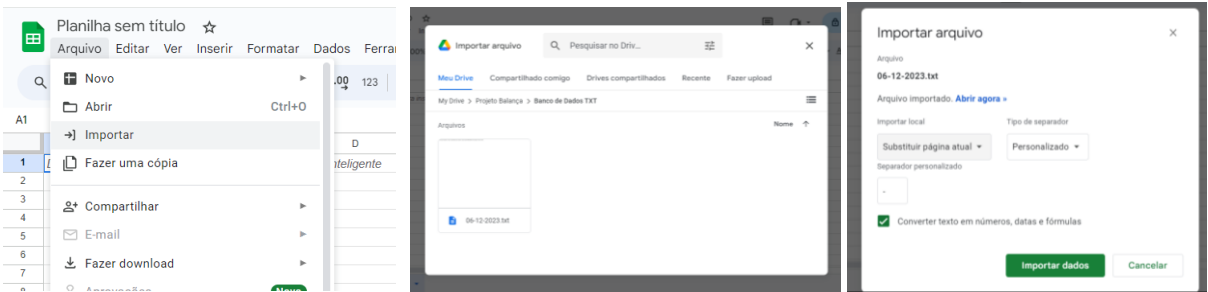
Esta subseção aborda o desenvolvimento da aplicação encarregada de exibir os dados extraídos pelo módulo do cartão SD. Nesse contexto, são detalhadas as configurações do banco de dados na nuvem, os parâmetros estabelecidos e o design atribuído a gestão visual do projeto.

3.6.1 Banco de Dados

Através da transferência dos arquivos em formato TXT do cartão de memória

integrado ao sistema desenvolvido, é necessário a conversão desses arquivos em documentos compatíveis com o Google Planilhas. Esse procedimento pode ser executado por meio da importação dos dados do arquivo TXT para a planilha Google, conforme ilustrado na Figura 39.

Figura 39 – Passos para conversão de TXT para Google Planilhas.



Fonte: Autor (2024).

Em seguida, é necessário fazer a junção de todos os arquivos criados no cartão de memória em apenas um banco de dados, para isso foi criado uma nova planilha e desenvolvido um script essa finalidade de junção dos arquivos. Na Figura 40 tem-se uma exemplificação dessa junção realizada.

Figura 40 – Planilha com junção dos dados.

1	MATERIAL	DATA	HORÁRIO	PESAGEM					
14	B	04/12/2023	10:00	29.3					
15	B	04/12/2023	13:00	28.5					
16	B	04/12/2023	16:00	29					
17	B	04/12/2023	19:00	30.4					
18	B	04/12/2023	22:00	29.6					
19	C	04/12/2023	03:00	28.6					
20	C	04/12/2023	06:00	28.5					
21	C	04/12/2023	09:00	29.4					
22	C	04/12/2023	12:00	30.5					
23	C	04/12/2023	15:00	28.2					
24	C	04/12/2023	18:00	29.9					
25	C	04/12/2023	21:00	29.5					
26	A	06/12/2023	13:00	29.4					
27	B	07/12/2023	15:00	28.1					
28	C	08/12/2023	16:00	28					
29	C	02/12/2023	00:00	29.5					
30	B	02/12/2023	01:00	28.2					
31	A	02/12/2023	02:00	29					
32	C	02/12/2023	03:00	30.4					
33	B	02/12/2023	04:00	30.8					
34	A	02/12/2023	05:00	29.4					

Fonte: Autor (2024).

3.6.2 Parâmetros

Para o desenvolvimento da gestão visual é necessários parâmetros para fins de comparação e análise, algum desses parâmetros já são recebidos diretamente dos arquivos txt, provenientes do código do protótipo, sendo esses a data, horário, matéria prima e peso, como pode ser visto na Figura 40.

Contudo, alguns parâmetros importantes para as análises como o turno de operação e se a pesagem está dentro da condição padrão aceitável para o processo

não são diretamente obtidos no arquivo txt e foram feitas formulas dentro da plataforma google para obtenção desses parâmetros. Com isso, a configuração final do banco de dados com todos os parâmetros estabelecidos é a mostrada na Figura 41. Contendo os seguintes parâmetros: turno, dia da semana, dia, mês, ano e condição (fora por falta, fora por excesso ou ok).

Figura 41 – Planilha com dados e parâmetros completo.

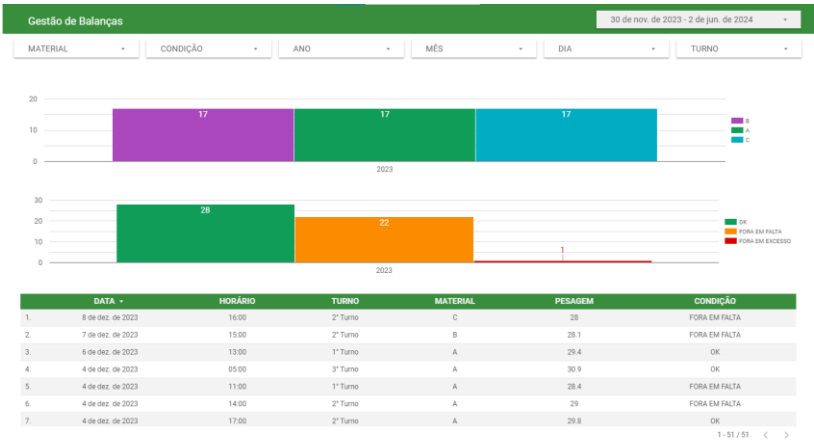
	A	B	C	D	E	F	G	H	I	J	K
1	MATERIAL	DATA	HORÁRIO	PESAGEM	TURNO	DIA DA SEMANA	DIA	MÊS	ANO	CONDIÇÃO	
14	B	04/12/2023	10.00	29.3	1° Turno	segunda-feira	04	DEZ	2023	OK	
15	B	04/12/2023	13.00	28.5	1° Turno	segunda-feira	04	DEZ	2023	FORA EM FALTA	
16	B	04/12/2023	16.00	29	2° Turno	segunda-feira	04	DEZ	2023	FORA EM FALTA	
17	B	04/12/2023	19.00	30.4	2° Turno	segunda-feira	04	DEZ	2023	OK	
18	B	04/12/2023	22.00	29.6	3° Turno	segunda-feira	04	DEZ	2023	OK	
19	C	04/12/2023	03.00	28.6	3° Turno	segunda-feira	04	DEZ	2023	FORA EM FALTA	
20	C	04/12/2023	06.00	28.5	1° Turno	segunda-feira	04	DEZ	2023	FORA EM FALTA	
21	C	04/12/2023	09.00	29.4	1° Turno	segunda-feira	04	DEZ	2023	OK	
22	C	04/12/2023	12.00	30.5	1° Turno	segunda-feira	04	DEZ	2023	OK	
23	C	04/12/2023	15.00	28.2	2° Turno	segunda-feira	04	DEZ	2023	FORA EM FALTA	
24	C	04/12/2023	18.00	29.9	2° Turno	segunda-feira	04	DEZ	2023	OK	
25	C	04/12/2023	21.00	29.5	2° Turno	segunda-feira	04	DEZ	2023	OK	
26	A	06/12/2023	13.00	29.4	1° Turno	quarta-feira	06	DEZ	2023	OK	
27	B	07/12/2023	15.00	28.1	2° Turno	quinta-feira	07	DEZ	2023	FORA EM FALTA	
28	C	08/12/2023	16.00	28	2° Turno	sexta-feira	08	DEZ	2023	FORA EM FALTA	
29	C	02/12/2023	00.00	29.5	3° Turno	sábado	02	DEZ	2023	OK	
30	B	02/12/2023	01.00	28.2	3° Turno	sábado	02	DEZ	2023	FORA EM FALTA	
31	A	02/12/2023	02.00	29	3° Turno	sábado	02	DEZ	2023	FORA EM FALTA	
32	C	02/12/2023	03.00	30.4	3° Turno	sábado	02	DEZ	2023	OK	
33	B	02/12/2023	04.00	30.8	3° Turno	sábado	02	DEZ	2023	OK	
34	A	02/12/2023	05.00	29.4	3° Turno	sábado	02	DEZ	2023	OK	

Fonte: Autor (2024).

3.6.3 Disposição e Visualização dos Dados

Com o banco de dados consolidado e os parâmetros devidamente definidos, torna-se viável realizar o *design* da gestão visual. Para essa finalidade, utilizou-se a plataforma Looker Studio¹¹, resultando na configuração do dashboard conforme ilustrado na Figura 42.

Figura 42 – Modelo do Dashboard.

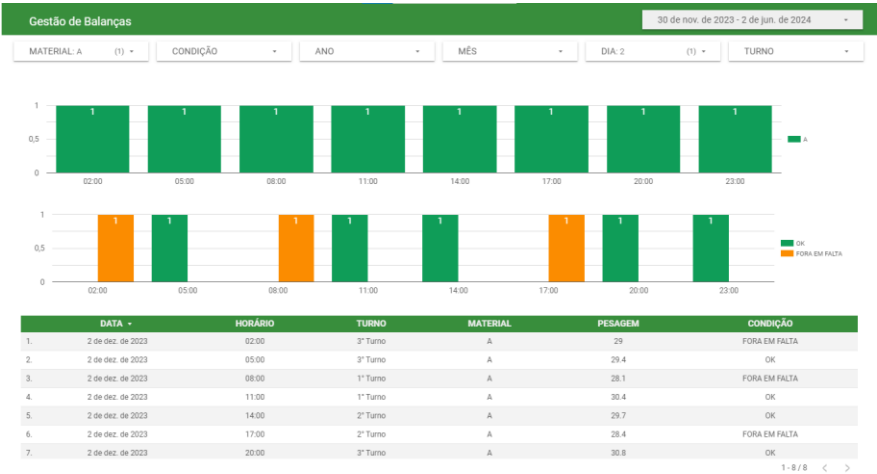


Fonte: Autor (2024).

¹¹ Disponível em: <https://lookerstudio.google.com/u/0/navigation/reporting>

A plataforma utilizada possibilita a interação com os dados apresentados por meio de filtros e cliques na tela, facilitando uma análise mais aprofundada dos parâmetros em questão. Na Figura 43, exemplifica-se essa capacidade, analisando as pesagens do material A na data 02/12/2023 durante todo os horários do dia que houve pesagem.

Figura 43 – Exemplificação de aprofundamento dos dados.



Fonte: Autor (2024).

4 RESULTADOS E DISCUSSÕES

Esta seção é reservada para a discussão dos testes em bancada para validação do protótipo do sistema desenvolvido para validação das funcionalidades estabelecidas nos objetivos específicos.

4.1 VALIDAÇÃO DAS COMUNICAÇÕES DO SISTEMA

O sistema a aquisição correta dos dados é fundamental para as análises de especificações dos parâmetros e rastreo das cargas pesadas, com isso se faz necessário à validação desses dados, sendo eles a pesagem, material e data/horário das coletas.

Para validação das pesagens foram usados pesos padrões de 50g, 100g, 500g e de 10kg, Figura 44, afim de mostrar a precisão do sistema e sua confiabilidade.

Figura 44 – Pesos padrões.



Fonte: Autor (2024).

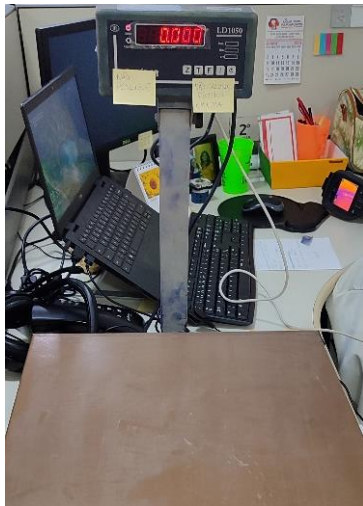
A partir disso foram feitas pesagens com cada um dos pesos padrões em cada um dos materiais para garantir que o material que foi escolhido na pesagem está de acordo com o selecionado. Na Tabela 8 tem a listagem de todas as pesagens realizadas na sequência de realização delas e na Figura 45 tem-se os registros dos testes feitos.

Tabela 8 – Detalhamento das pesagens realizadas.

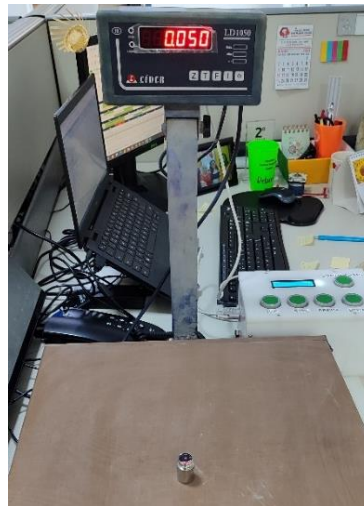
Peso/Material	1502	DYNE	RESINA
0,000	Pesagem nº13	Pesagem nº14	Pesagem nº15
0,050	Pesagem nº4	Pesagem nº5	Pesagem nº6
0,100	Pesagem nº7	Pesagem nº8	Pesagem nº9
0,500	Pesagem nº10	Pesagem nº11	Pesagem nº12
10,022	Pesagem nº3	Pesagem nº2	Pesagem nº1

Fonte: Autor (2024).

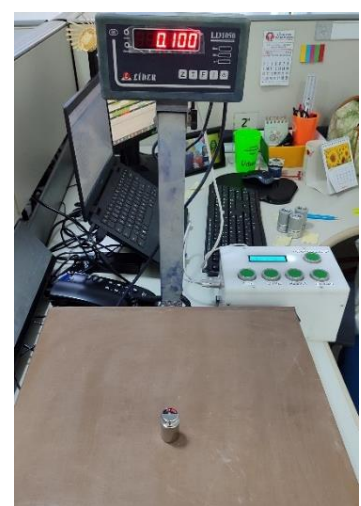
Figura 45 – Pesagem realizadas com cada peso padrão.



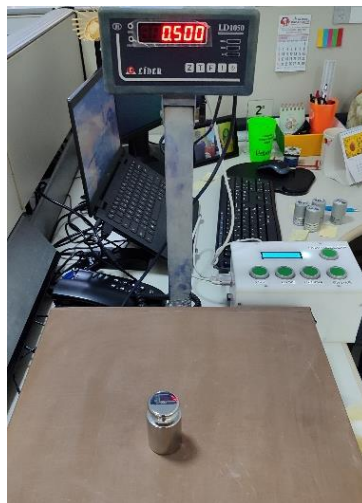
(a) Balança sem carga



(b) Pesagem com 50g



(c) Pesagem com 100g



(d) Pesagem com 500g



(e) Pesagem com 10kg

Fonte: Autor (2024).

Como pode ser visto nas Figuras 46, os testes tiveram sucesso com o material, o valor do peso definidos para testagens compatíveis com os dados visto no display do sistema.

Figura 46 – Dados mostrado no display para cada peso padrão.



(a) Display com balança sem carga



(b) Display com pesagem com 50g



(c) Display com pesagem com 100g



(d) Display com pesagem com 500g

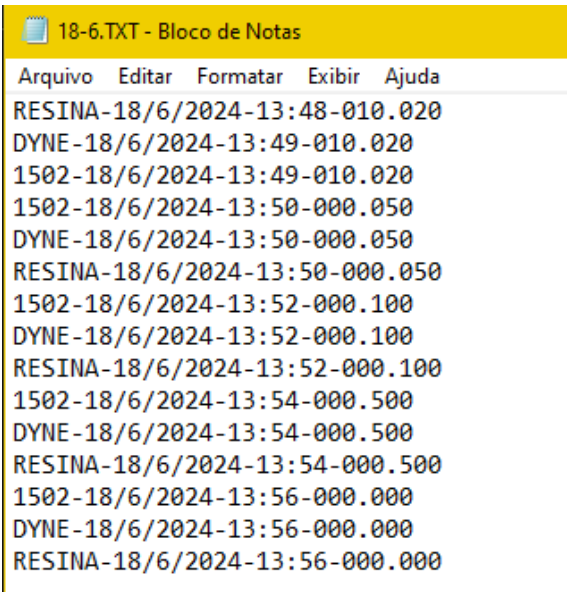


(e) Display com pesagem com 10kg

Fonte: Autor (2024).

Porém além das informações contidas no display é necessário à validação dos dados registrados arquivo txt do cartão de memória, na Figura 47 segue registro do arquivo gerado por essas pesagens, na qual também foi obtido um resultado satisfatório.

Figura 47 – Arquivo gerado pelas pesagens com o peso padrão.



Fonte: Autor (2024).

Além dos dados de pesagem e material, a data e hora são parâmetros fundamentais para rastreamento da carga, para isso foram feitas medições hora a

hora entre 8h as 17h durante dois dias, 20/06/2024 e dia 21/06/2024. Nessa coleta foram registradas as pesagens com a matéria prima intitulada de 1502 e pesagens diversas, Figura 48 com registro de uma das medições realizadas.

Figura 48 – Um dos registros de pesagem hora a hora.



(a) Pesagem 9h30min dia 20/06/2024 (b) Pesagem 8h35min dia 21/06/2024

Fonte: Autor (2024).

Analogamente as analise feitas para pesagem e material para os dados de data e hora obteve resultados satisfatórios em relação as informações mostradas no display e registradas no arquivo txt, Figura 49.

Figura 49 – Arquivo gerado pelas pesagens hora a hora durante dois dias

20-6.TXT - Bloco de Notas					
Arquivo	Editar	Formatar	Exibir	Ajuda	
1502-20/6/2024-8:39-000.000					
1502-20/6/2024-8:41-000.106					
1502-20/6/2024-8:59-000.120					
1502-20/6/2024-9:10-000.122					
1502-20/6/2024-9:30-000.110					
1502-20/6/2024-9:52-000.112					
1502-20/6/2024-10:31-000.108					
1502-20/6/2024-11:58-000.126					
1502-20/6/2024-12:53-000.124					
1502-20/6/2024-12:59-000.126					
1502-20/6/2024-13:0-000.000					
1502-20/6/2024-13:32-000.002					
1502-20/6/2024-14:21-000.000					
1502-20/6/2024-14:25-000.000					
1502-20/6/2024-14:34-000.106					
1502-20/6/2024-14:45-000.110					
1502-20/6/2024-14:53-000.112					
1502-20/6/2024-15:51-000.106					
1502-20/6/2024-15:51-000.000					
1502-20/6/2024-15:59-000.106					
1502-20/6/2024-16:15-000.000					
1502-20/6/2024-16:46-000.106					
1502-20/6/2024-17:1-000.106					
1502-20/6/2024-17:2-000.000					

21-6.TXT - Bloco de Notas					
Arquivo	Editar	Formatar	Exibir	Ajuda	
1502-21/6/2024-8:0-000.000					
1502-21/6/2024-8:5-000.000					
1502-21/6/2024-8:41-000.074					
1502-21/6/2024-9:19-000.056					
1502-21/6/2024-9:36-000.000					
1502-21/6/2024-9:49-000.182					
1502-21/6/2024-10:29-000.000					
1502-21/6/2024-11:26-000.034					
1502-21/6/2024-11:26-000.034					
1502-21/6/2024-11:49-000.034					
1502-21/6/2024-11:55-000.034					
1502-21/6/2024-12:38-000.000					
1502-21/6/2024-13:16-000.034					
1502-21/6/2024-14:11-000.036					
DYNE-21/6/2024-15:47-000.202					
DYNE-21/6/2024-16:54-000.198					
DYNE-21/6/2024-16:57-000.010					

(a) Pesagens do dia 20/06/2024 (b) Pesagens do dia 21/06/2024

Fonte: Autor (2024).

4.2 SIMULAÇÃO DO AMBIENTE PRODUTIVO

Um dos objetivos do projeto é permitir sua utilização no ambiente produtivo

industrial com isso é necessário realizar teste para simular o ambiente em que o protótipo será inserido, com isso algumas premissas devem ser consideradas como funcionamento por 24h do sistema ligado durante vários dias.

Dito isso para essa simulação foi realizada a premissa do sistema ficar ligado por no mínimo 48h e não apresentar temperatura elevada, que causaria riscos ao sistema e até possível queima dos componentes.

Para esse teste foi utilizada uma câmera termográfica, Flir E8¹², com os parâmetros de emissividade de 0.95, Temperatura refletida 20°C e distância de 1 metro, como pode ser visto na Figura 50.

Figura 50 – Parâmetros definidos na câmera termográficas.



Fonte: Autor (2024).

O teste foi iniciado no dia 20/06/2024 às 8h39min, foi enviado uma medição de pesagem para esse registro de informação, além disso foi registrado a temperatura base inicial do sistema com a câmera termográfica, Figura 51, sendo ela de 39.0°C.

Figura 51 – Registro da temperatura inicial do sistema.

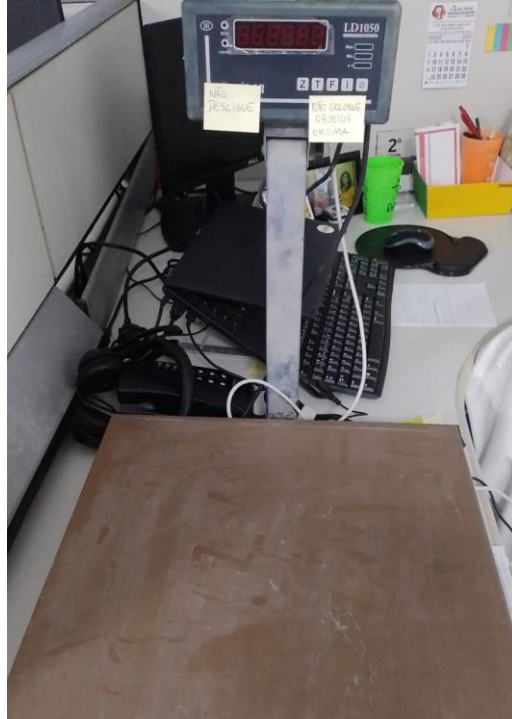


Fonte: Autor (2024).

¹² Disponível em: <https://www.flir.com.br/products/e8-pro/?vertical=condition%20monitoring&segment=solutions>

Contudo durante a noite desse mesmo dia ocorreu uma queda de energia no local onde estava acontecendo o teste e com isso o sistema foi desligado, Figura 52.

Figura 52 – Registro de balança sem energia.



Fonte: Autor (2024).

Apesar da interrupção da testagem esse acontecimento foi válido para mostrar que após uma falta de energia o sistema volta as suas funcionalidades normalmente, sendo apenas necessário sua religação para operar, na Figura 53 é mostrado a religação do sistema e registro de data e horário normalizados.

Figura 53 – Registro do religamento do sistema



Fonte: Autor (2024).

Com a energia reestabelecida, foi reiniciado o teste no dia 21/06/2024 as 8h e finalizado dia 25/06/2024 as 8h58min, seguindo a mesma premissa de registro descrita anteriormente, sendo assim o sistema ficou ligado durante 96h. Nas Figura 54 é mostrado as temperaturas resultantes do início, 35.9°C, e a temperatura final,

35.9°C, do teste e na Figura 55 o registro de envio durante esses dias.

Figura 54 – Temperatura registradas no início e no final do teste.



(a) Temperatura inicial (21/06/2024) (b) Temperatura final (25/06/2024)

Fonte: Autor (2024).

Figura 55 – Arquivo gerado pelas pesagens hora a hora durante dois dias

21-6.TXT - Bloco de Notas	25-6.TXT - Bloco de Notas
Arquivo Editar Formatar Exibir Ajuda	Arquivo Editar Formatar Exibir Ajuda
1502-21/6/2024-8:0-000.000	1502-25/6/2024-7:57-000.000
1502-21/6/2024-8:5-000.000	1502-25/6/2024-8:4-000.000
1502-21/6/2024-8:41-000.074	1502-25/6/2024-8:14-000.106
1502-21/6/2024-9:19-000.056	1502-25/6/2024-8:58-000.008
1502-21/6/2024-9:36-000.000	1502-25/6/2024-13:12-000.010
1502-21/6/2024-9:49-000.182	
1502-21/6/2024-10:29-000.000	
1502-21/6/2024-11:26-000.034	
1502-21/6/2024-11:26-000.034	
1502-21/6/2024-11:49-000.034	
1502-21/6/2024-11:55-000.034	
1502-21/6/2024-12:38-000.000	
1502-21/6/2024-13:16-000.034	
1502-21/6/2024-14:11-000.036	
DYNE-21/6/2024-15:47-000.202	
DYNE-21/6/2024-16:54-000.198	
DYNE-21/6/2024-16:57-000.010	

(a) Pesagens do dia 21/06/2024 (b) Pesagens do dia 25/06/2024

Fonte: Autor (2024).

Logo é notável que o sistema ficou dentro das especificações, entre -40°C a 85°C como citada na sessão 3.3.1 (Arduino Mega 2560) deste trabalho, e não apresentou nenhuma anomalia ou temperaturas elevadas críticas durante esse tempo de testagem.

5 CONCLUSÕES

Com as informações geradas ao longo deste projeto, é possível entender o potencial que o sistema embarcado, em específico a estrutura do arduino, estão tomando na tecnologia atualmente. Com recursos hardware e software facilmente acessíveis, foi possível criar um sistema que viabiliza a aquisição de dados e acompanhados a cada pesagem para tomadas de decisão mais assertivas.

O desenvolvimento deste projeto iniciou com a problemática de como se poderia realizar a aquisição de dados de um ambiente de pesagem para análise de desperdício. Logo foram definidas as funcionalidades que o sistema iria assumir de forma a garantir praticidade de operação e eficiência na rastreabilidade e aquisição das informações e com custo reduzido. Com isso, foi possível estabelecer a proposta de desenvolvimento do projeto.

Com o funcionamento definido, foram selecionados os componentes eletrônicos que iriam compor o sistema, analisando o custo benefício desses itens. Nesta etapa, foram selecionados componentes responsáveis pela pesagem, comunicação de dados, central de processamento e registro de dados.

Esses itens foram, em seguida, testados individualmente, unidos em um único sistema e conectado ao instrumento de medição, foi realizado o código, através da IDE do arduino, para o sistema realizar as funções definidas. Com o código e o sistema eletrônico montado, foi possível continuar o trabalho ao desenvolver a interface que exibe as informações coletadas pelo sistema embarcado.

Com a lógica e a parte eletrônica pronta, a etapa de desenvolvimento foi finalizada com a modelagem da caixa em software CAD. Essa caixa pode ser enviada para ser fabricado dentro de um centro de usinagem ou em uma máquina de impressão 3D.

Por fim, foi realizado a testagem de validação de informação, utilizando pesos padrões, e também foi realizado a simulação de um ambiente produtivo com a análise termográfica do sistema, garantido assim a confiabilidade do sistema e atingimento dos objetivos desse trabalho.

5.1 TRABALHOS FUTUROS

Como proposta para futuro desenvolvimento do projeto, tem-se:

- Realização de testes no meio produtivo para mais validações dos módulos;
- Adição de um módulo de identificação automática da matéria prima;
- Adição de impressora com etiqueta para liberação do peso apenas pelo especificado para a próxima etapa do processo produtivo;

REFERÊNCIAS

AFONSO, Júlio Carlos; SILVA, Raquel Medeiros da. A evolução da balança analítica. Química Nova, v. 27, 2004.

ALEXANDRE BALBINOT; VALNER JOÃO BRUSAMARELLO. Instrumentação e Fundamentos de Medidas - Vol. 2. [s.l: s.n.].

ALVES, J. Protocolos da comunicação Serial. Disponível em: <https://curtocircuito.com.br/blog/Categoria%20Arduino/protocolo-de-comunicacao-serial#caracteristicas>. Acesso em: 18/11/2023.

ALVES, Reinaldo Galo Febrônio; MANFIO, Edio Roberto; DE GENOVA BUGATTI, Ildeberto. DOSAGEM PROGRAMADA DE MATÉRIA-PRIMA PARA OS DIVERSOS PROCESSOS DA INDÚSTRIA. Revista Eletrônica e-Fatec, v. 6, n. 1, p. 8-8, 2016.

ARDUINO. Arduino. 2020. <https://www.arduino.cc/> . Acesso em: 27 de jan. 2024.

ATMEL. Atmel ATmega640/V-1280/V-1281/V-2560/V-2561/V. Atmel Corporation, 2014. 435p. Disponível em: http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-2549-8-bit-AVR-Microcontroller-ATmega640-1280-1281-2560-2561_datasheet.pdf. Acesso em: 27 de jan. 2024.

B180 BALANÇA - Descrição Técnica. [s.l: s.n.]. Disponível em: <https://www.liderbalancas.com.br/wp-content/uploads/produtos/files/1651278494.pdf>. Acesso em: 30 jun. 2024.

B180-LD1050. [s.l: s.n.]. Disponível em: <https://www.liderbalancas.com.br/wp-content/uploads/produtos/manual/balanca-b180-ld1050.pdf>. Acesso em: 8 jul. 2024.

BOLTON, W. Instrumentação & Controle. São Paulo: Hemus, 2002.

CARDOSO, Marcelo O. UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO DEPARTAMENTO ACADÊMICO DE ELETRÔNICA CURSO DE ESPECIALIZAÇÃO EM AUTOMAÇÃO INDUSTRIAL INDÚSTRIA 4.0: a quarta revolução industrial, 2016. Disponível em: https://repositorio.utfpr.edu.br/jspui/bitstream/1/17086/1/CT_CEAUT_2015_08.pdf

DE ALMEIDA, Rodrigo Maximiano Antunes; DE MORAES, Carlos Henrique Valério; SERAPHIM, Thatyana de Faria Piola. Programação de sistemas embarcados: Desenvolvendo software para microcontroladores em linguagem C. Elsevier Brasil, 2017.

DAWOUD, Dawoud Shenouda; DAWOUD, Peter. Serial communication protocols and standards. CRC Press, 2022.

DS1307 datasheet. Disponível em: <https://semiconductors.es/DS1307.html> . Acesso em: 30 jun. 2024.

EXXACTA. Comunicação serial o que é, para que serve e quais as vantagens? Disponível em: <https://www.dakol.com.br/comunicacao-serial/>. Acessado em 02/11/2023.

FERNANDES, Jefferson Doolan. Sistema embarcado para monitoramento de variáveis físicas do rotor de máquinas elétricas. 2017. 103f. Tese (Doutorado em Ciência e Engenharia de Petróleo) - Centro de Ciências Exatas e da Terra, Universidade Federal do Rio Grande do Norte, Natal, 2017.

FIALHO, Instrumentação Industrial: Conceitos, Aplicações e Análises. 7. ed. São Paulo: Érica, 2012.

GREFIM, Daniel; SILVA, Renan Tavares da; CRUZ, Willian Carlos Teixeira da. Máquina de medir por coordenadas utilizando microcontrolador e sistema supervisorio. 2018. 100 f. Trabalho de Conclusão de Curso (Superior de Tecnologia em Mecatrônica Industrial) - Universidade Tecnológica Federal do Paraná, Curitiba, 2018.

HD44780 Datasheet Disponível em: <https://blogmasterwalkershop.com.br/arquivos/datasheet/Datasheet%20HD44780.pdf> . Acesso em: 21 jul. 2024.

HEATH, S. Embedded System Design, 2. ed. Oxford: Newnes, 2003.

MARCELINO, Mateus et al. Aplicação da ponte de wheatstone com extensômetros elétricos: instrumentação de um link para medições das forças de tração e compressão. Revista Científica Semana Acadêmica. Fortaleza, ano MMXVIII, Nº. 000128, 07/08/2018. 87 Disponível em: <https://semanaacademica.com.br/>

artigo/aplicacao-da-ponte-de-wheatstonecom-extensometros-eletricos-instrumentacao-de-um-link-para . Acesso em: 11 jan. 2024.

MARTINS, Gleison Hidalgo Hidalgo et al. Projeto de redução do desperdício de matéria-prima: estudo de caso na indústria de embalagens de papel no Brasil. *Gestão e Projetos: GeP*, v. 4, n. 3, p. 141-167, 2013.

MELLIS, D. A. Arduino - Introduction. Arduino, 2009. Disponível em: <http://www.arduino.cc/en/Guide/Introduction> . Acesso em: 16 jan. 2024.

MIYADAIRA, Alberto Noboru. Microcontroladores PIC18: Aprenda e programe em linguagem C. Saraiva Educação SA, 2009.

MICROSD Datasheet (PDF) - PJS008-1000 - Yamaichi Electronics Co., Ltd. Disponível em: https://www.alldatasheet.com/category/index.jsp?sSearchword=Microsd&gad_source=1&gclid=EAlaIqObChMI4f_e_KWEhwMVR2BIAB0efAmtEAAAYASAAEgKnqvD_BwE . Acesso em: 30 jun. 2024.

NACHAZEL, Ted. What is a Strain Gauge and How Does it Work? Michigan Scientific Corporation, 2020. Disponível em: <https://www.michsci.com/what-is-a-strain-gauge/> . Acesso em: 11 jan. 2024.

NASCIMENTO, F. P. D.; SOUSA, F. L. L. Metodologia Da Pesquisa Cientifica: TEORIA E PRATICA. THESAURUS, 2016.

OHNO, Taiichi. O sistema Toyota de produção além da produção. Bookman, 1997.

OLIVEIRA, Euler. Como usar com Arduino - Módulo Real Time Clock RTC DS1307 - BLOG MASTERWALKER SHOP, 2019. Disponível em: <https://blogmasterwalkershop.com.br/arduino/como-usar-com-arduino-modulo-real-time-clock-rtc-ds1307>. Acessado em 12/11/2023.

RHEINBOLDT, Heinrich. Historia da balanca: a vida de J. J. Berzelius. São Paulo: Nova Stella, 1988.

RODRIGUES, Evando; PEDÓ, Ricardo; TEDESCO, Leonel Pablo. Sistemas Embarcados e sua Aplicação na Indústria. Santa Cruz do Sul: Universidade de Santa Cruz do Sul, 2013.

RTCLIB - Arduino Reference. Disponível em: <https://www.arduino.cc/reference/en/libraries/rtclib/> . Acessado em 12/11/2023

SD - Arduino Reference. Disponível em: <https://www.arduino.cc/reference/en/libraries/sd/> . Acessado em 11/11/2023.

SOUZA, F. Arduino MEGA 2560. Disponível em: <https://embarcados.com.br/arduino-mega-2560/> . Acessado em 11/11/2023.

LD1050 TERMINAL - Descrição Técnica. [s.l: s.n.]. Disponível em: <https://www.liderbalancas.com.br/wp-content/uploads/produtos/files/1450359761.pdf> . Acesso em: 30 jun. 2024.

LIQUIDCRYSTAL - Arduino Reference. Disponível em: https://www.arduino.cc/reference/en/libraries/liquidcrystal/?_gl=1 . Acessado em 12/11/2023.

APÊNDICE A – FICHAS TÉCNICAS DA CAIXA

Figura 56 – Ficha Técnica da caixa

