



## Monitoramento em Tempo Real do Peso das Balanças em Usinas de Cana-de-Açúcar

Silvano Antonio da Silva, Leandro Carlos Fernandes

<sup>1</sup> Faculdade de Computação e Informática  
Universidade Presbiteriana Mackenzie (UPM) – São Paulo, SP – Brazil

10408081@mackenzista.com.br

**Abstract.** *The use of Internet of Things (IoT) technology in industrial processes has become increasingly essential for improving operational efficiency. This paper presents a real-time monitoring system for the sugarcane industry, where weight sensors are integrated into the scales used to measure sugarcane loads. The data is collected and transmitted in real-time to a central system, generating reports that allow administrators to make informed decisions about production. The implementation of this IoT system aims to reduce human error, optimize processes, and align with the Sustainable Development Goal 9 (Industry, Innovation, and Infrastructure). This study also discusses the challenges and benefits of adopting IoT in the sugarcane sector, offering a new perspective on how technology can streamline production in agricultural industries.*

**Resumo.** *O uso da tecnologia de Internet das Coisas (IoT) em processos industriais tornou-se cada vez mais essencial para melhorar a eficiência operacional. Este artigo apresenta um sistema de monitoramento em tempo real para o setor de cana-de-açúcar, onde sensores de peso são integrados às balanças utilizadas para medir as cargas de cana. Os dados são coletados e transmitidos em tempo real para um sistema central, gerando relatórios que permitem aos administradores tomar decisões informadas sobre a produção. A implementação desse sistema IoT visa reduzir erros humanos, otimizar processos e alinhar-se ao Objetivo de Desenvolvimento Sustentável 9 (Indústria, Inovação e Infraestrutura). O estudo também discute os desafios e benefícios da adoção de IoT no setor sucroenergético, oferecendo uma nova perspectiva de como a tecnologia pode agilizar a produção em indústrias agrícolas.*

### 1. Introdução

Nos últimos anos, a indústria sucroalcooleira tem enfrentado o desafio de otimizar seus processos produtivos para atender à crescente demanda por eficiência e sustentabilidade. O uso de tecnologias como a Internet das Coisas (IoT) surge como uma solução inovadora para o setor, permitindo o monitoramento em tempo real de diversos aspectos da produção. Em especial, o monitoramento de balanças em usinas de cana-de-açúcar é crucial para garantir a precisão na pesagem das cargas e, consequentemente, melhorar o controle de estoque e a tomada de decisões gerenciais. A aplicação de sensores IoT em

balanças industriais possibilita a coleta e transmissão de dados em tempo real, permitindo a geração de relatórios automáticos e otimizando o processo produtivo.

A relevância deste projeto está em sua capacidade de reduzir erros manuais, aumentar a precisão e automatizar etapas críticas da produção, como a pesagem de cana-de-açúcar. Além disso, o monitoramento em tempo real oferece aos administradores maior controle sobre a operação, possibilitando uma reação rápida a possíveis inconsistências e falhas no processo. Este estudo busca propor uma solução eficiente para o setor, alinhada aos Objetivos de Desenvolvimento Sustentável (ODS) da ONU, em particular o ODS 9, que trata da promoção de uma infraestrutura industrial resiliente, inovação e modernização. A adoção de IoT no monitoramento de balanças contribui diretamente para a inovação no setor e a melhoria contínua dos processos industriais, promovendo sustentabilidade e competitividade no mercado global.

### **1.1. Motivação do Projeto**

A motivação para o desenvolvimento deste projeto decorre da necessidade de modernizar processos industriais e reduzir os desperdícios causados por erros humanos e falhas nos sistemas tradicionais de pesagem. O setor sucroenergético desempenha um papel fundamental na economia agrícola, e a precisão no controle de produção é essencial para aumentar a competitividade no mercado global. A introdução de sensores IoT para monitoramento das balanças de pesagem visa criar uma solução robusta, eficiente e em sintonia com as exigências da indústria moderna.

### **1.2. Alinhamento com ODS 9**

O projeto também se alinha com o Objetivo de Desenvolvimento Sustentável 9 da ONU, que preconiza a necessidade de construir infraestruturas resilientes, promover a inovação tecnológica e modernizar indústrias. Ao propor o uso de tecnologias IoT no setor de cana-de-açúcar, este trabalho busca modernizar um processo crítico da cadeia produtiva, trazendo benefícios como maior eficiência, redução de custos e otimização de recursos.

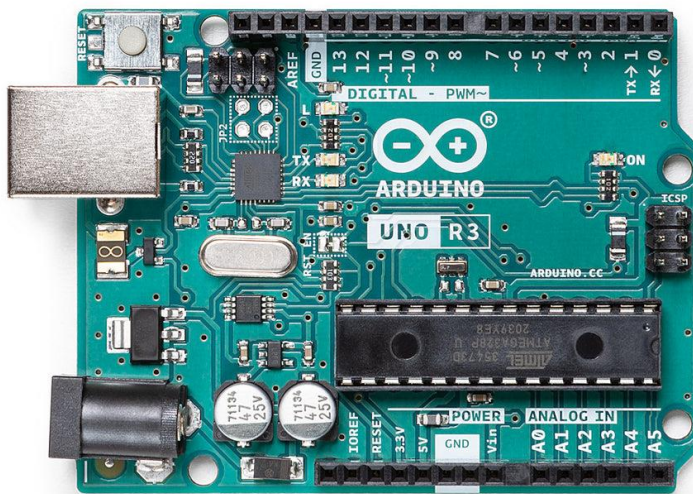
## 2. Materiais e métodos

### 2.1. Arduino UNO

- **Descrição:** O Arduino UNO é a placa principal do projeto, responsável por processar os dados vindos dos sensores e enviar as informações para o sistema via MQTT.

- **Função no projeto:** Ele será o controlador principal, recebendo os sinais do sensor de peso e se comunicando com o sistema.

- Imagem: <https://store.arduino.cc/arduino-uno-rev3>



### 2.2. Sensor de Carga (Load Cell)

- **Descrição:** O sensor de carga (geralmente um **HX711** ou outro sensor de célula de carga) é usado para medir o peso colocado na balança.

- **Função no projeto:** Converte a força aplicada em um sinal elétrico proporcional, permitindo que o Arduino interprete o peso.

- Imagem: [https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711\\_english.pdf](https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf)

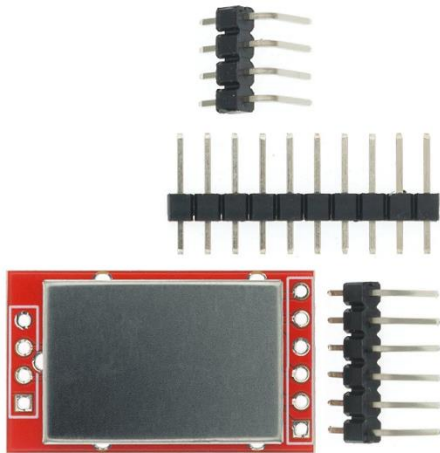


### 2.3. Amplificador HX711

- **Descrição:** Um amplificador de célula de carga que é necessário para aumentar o sinal gerado pelo sensor de carga para que o Arduino possa processá-lo.

- **Função no projeto:** Amplifica os sinais do sensor de carga e os envia para o Arduino para interpretação.

- **Imagem:** [https://www.sparkfun.com/datasheets/Sensors/ForceFlex/hx711\\_english.pdf](https://www.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf)

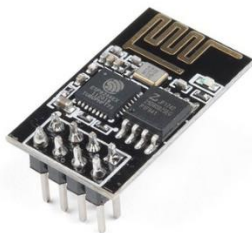


### 2.4. Módulo Wi-Fi ESP8266

- **Descrição:** Módulo de comunicação Wi-Fi que conecta o Arduino à Internet, permitindo o envio de dados via protocolo MQTT.

- **Função no projeto:** Responsável por enviar os dados de pesagem para o servidor através do protocolo MQTT.

- **Imagem:** [https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf)



### 2.5. MQTT (Message Queuing Telemetry Transport)

- **Descrição:** Protocolo de comunicação leve, adequado para dispositivos de IoT, que envia dados de forma eficiente para o sistema via Internet.

- **Função no projeto:** Transmissão dos dados de pesagem para o sistema em tempo real, utilizando tópicos para comunicação via servidor MQTT.

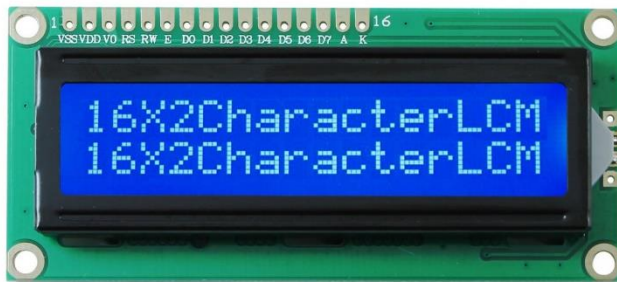
- **Imagem:** <https://www.ibm.com/docs/en/ibm-mq/9.1?topic=mqtt-sample-scenario>

## 2.6. Atuador: Display LCD (16x2)

- **Descrição:** Um display LCD pode ser usado como atuador, mostrando os resultados da pesagem em tempo real na balança.

- **Função no projeto:** Exibir os dados de pesagem para o usuário diretamente no dispositivo.

- **Imagem:** <https://www.adafruit.com/datasheets/TC1602A-01T.pdf>



## 2.7. Fonte de Alimentação

- **Descrição:** Fonte de alimentação de 9V ou 12V para alimentar o Arduino e os componentes eletrônicos.

- **Função no projeto:** Fornecer energia estável para os componentes.

- **Imagem:** <https://www.adafruit.com/datasheets/9V.pdf>



## 2.8. Componentes adicionais

- **Fios de Conexão (Jumpers):** Para realizar as conexões elétricas entre os componentes.
- **Protoboard:** Para montagem temporária dos circuitos.
- **Resistores e capacitores:** Para ajustes nos circuitos de acordo com a necessidade de precisão.

## Ferramentas e Métodos

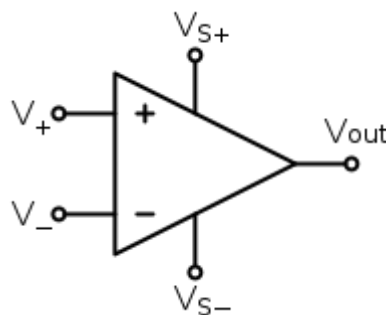
- **IDE Arduino:** Será utilizada para programar o Arduino, com o código responsável por interpretar os dados do sensor de carga e se comunicar via MQTT.
- **Plataforma MQTT:** Broker MQTT para gerenciar a troca de mensagens entre o Arduino e o sistema.
- **Software de Monitoramento:** Ferramenta para receber e processar os dados das pesagens, gerando relatórios.

## 2.9 Funcionamento:

### Amplificador Operacional

O módulo HX711 utilizado neste tutorial é um conversor analógico-digital e amplificador operacional de 24 bits. Ele é usado para amplificar o sinal da célula de carga de 50 kg, além de realizar a conversão do sinal analógico para um sinal digital. Isso é necessário, pois o sinal de saída da célula está na escala de milivolts (mV). Portanto para que seja um sinal que possa ser lido pelo microcontrolador, ele precisa ser amplificado para o nível adequado da placa utilizada.

Um amplificador operacional geralmente possui uma entrada variável e uma entrada constante, e então ele retorna como saída a diferença das tensões, de acordo com o ganho do amplificador. Essa matemática pode ser expressa através de  $V_{out} = A * [ (V+) - (V-) ]$ , onde o ganho "A" costuma ser fornecido como um valor fixo pelo fabricante para o desenvolvimento do circuito.



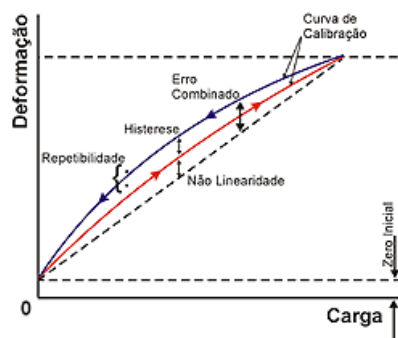
### Amplificador Operacional

Fonte: [Wikipedia](#)

Os pinos "Vs+" e "Vs-" são responsáveis pela alimentação positiva e negativa, respectivamente, do amplificador.

### Funcionamento da Célula de Carga

Células de carga possuem um funcionamento parecido, em relação à variação de resistência, com potenciômetros, pois elas possuem extensômetros que se deformam de acordo com a carga aplicada sobre sua área de contato. Essa deformação faz com que sua resistência aumente. Entretanto existem alguns fatores que podem influenciar na variação ôhmica do extensômetro, como pode ser visto abaixo.



Curva da Deformação pela Carga

Fonte: Portal Célula de Carga

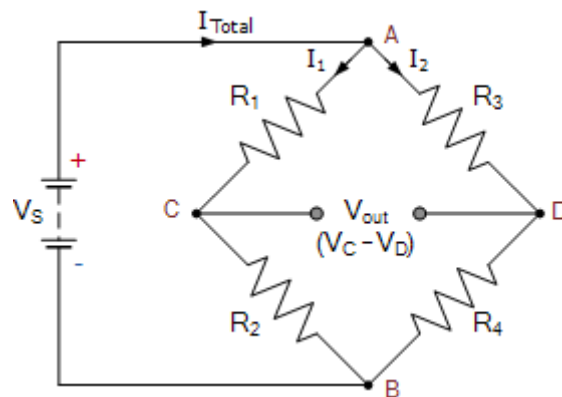
Em um mundo ideal, o gráfico da diferença de potencial em relação à carga aplicada em uma célula de carga seria uma reta. A **linearidade** é um erro representado pela diferença entre a curva real de resposta da célula quando aplicado um peso sobre sua área de contato, quando comparado com a curva ideal de expectativa teórica. A **histerese** também é um fenômeno comum quando testamos relações teóricas na prática. Como é possível ver na imagem acima, a curva desta relação acaba se deslocando para cima ou para baixo durante a aplicação ou a remoção de peso na célula. Ou seja, ao aplicar um peso de um quilo à célula sem carga, por exemplo, teríamos como resposta um valor diferente do que se removermos um quilo da mesma célula já carregada com dois quilos. A **repetibilidade** é a capacidade da célula de carga em retornar o mesmo valor quando aplicado o exato mesmo peso sobre o sensor, portanto ela deve ser levada em conta na hora de escolha do sensor adequado para realizar uma certa aplicação. A **fluência** é a perda de característica de transdutor (componente que transforma uma grandeza física, neste caso o peso em um sinal elétrico) do material, causando uma variação inesperada no sinal elétrico de saída. O **erro combinado** da curva acima é um erro calculado a partir de dois ou mais erros citados anteriormente. Todos esses fatores de distúrbio do componente devem ser levados em conta durante a calibração.

### Ponte de Wheatstone

A **ponte de Wheatstone** é, se não o único, o método mais utilizado para realizar medições sobre a variação da célula de carga. Com ela é possível



identificar valores de resistência ôhmica desconhecida. Ela é utilizada para determinar resistências muito pequenas, podendo ser utilizada para medir uma escala de mili-ohms. O circuito da ponte nada mais é do que duas disposições série-paralelo de dois resistores, preferencialmente de mesmo valor. Com o valor igualado em todos os resistores da ponte, a diferença de potencial no centro da ponte quando medido entre as disposições de resistores deve ser igual a zero. A imagem abaixo mostra um exemplo do circuito da ponte de Wheatstone.



### Ponte de Wheatstone

Fonte: [\*Electronics Tutorials\*](#)

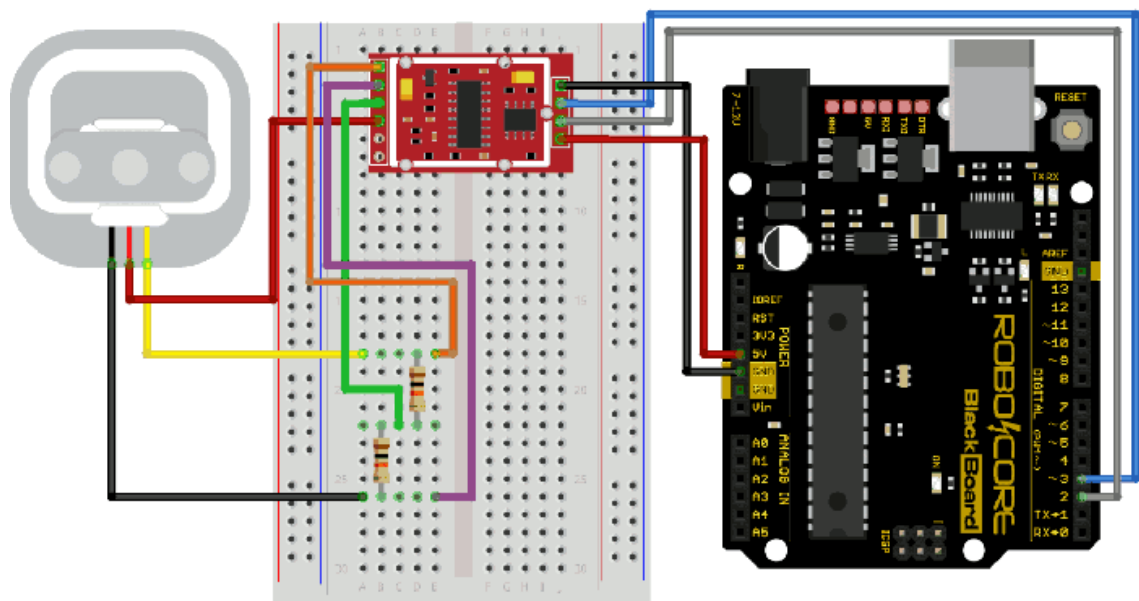
Para a ponte de Wheatstone, a fórmula matemática para calcular as resistências necessárias para o valor  $V_{out}$  ser igual a zero é  $V_{out} = V_C - V_D = [R_2 / (R_1 + R_2)] * V_{in} - [R_4 / (R_3 + R_4)] * V_{in}$ . Nessa condição, pode-se usar a relação entre os resistores  $R_1 / R_2 = R_3 / R_4$ . Assim é possível calibrar manualmente a tensão de saída da ponte, já que ao variar uma das resistências, o valor será alterado. Para realizar uma calibragem de hardware através das resistências, é necessário substituir um dos resistores de cada lado da ponte, por exemplo  $R_1$  e  $R_3$ , então alimentar a ponte com 5 V e GND nos pontos "A" e "B", respectivamente, como pode ser visto na imagem acima. Então, com um multímetro, variar os potenciômetros para que o valor lido entre os pontos "C" e "D" seja igual a zero.

### Calibração Via Software

#### Circuito

O circuito a seguir utiliza o módulo HX711 para realizar a comparação entre as tensões de entrada do circuito (excitações), com os pinos "E+" e "E-", e a tensão de entrada do amplificador operacional, com os pinos "A-" e "A+". Observe a montagem na imagem abaixo.





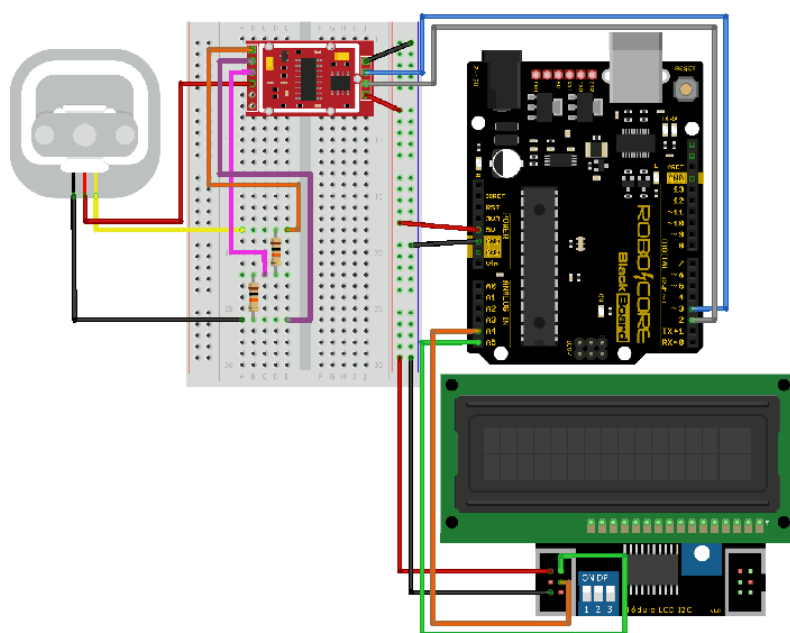
fritzing

### Circuito Calibração via Software

A calibração será feita por um fator que será calculado com o valor retornado pelo módulo HX711, para converter o valor lido para uma unidade de peso (neste caso, utilizaremos a medida em quilogramas).

### Circuito

Aproveitando o circuito utilizado para a calibração via software, iremos acrescentar um display LCD, com um módulo I2C para facilitar a prototipagem e diminuir o número de portas utilizadas, para mostrar o peso que está sendo aplicado na área de contato da célula de carga. O circuito completo pode ser visto na imagem abaixo.



fritzing

## Circuito Balança Digital

Como é possível observar na imagem acima, mantivemos o circuito utilizado anteriormente para realizar a calibração via software, porém incluímos o display LCD com módulo I2C para simular uma balança digital. Neste exemplo será possível apresentar pelo display o peso medido pela célula em quilogramas.

### Modulo de Comunicação:

O modulo de comunicação escolhido foi o troca do Arduino R3 pelo R4 Wifi no qual conecta o Arduino à Internet, permitindo o envio de dados via protocolo MQTT. Segue abaixo trecho do código que faz a comunicação com o servidor MQTT para envio dos dados:

```
#include <HX711.h>          // Biblioteca para a célula de carga
#include <WiFi.h>            // Biblioteca para conexão WiFi
#include <PubSubClient.h>    // Biblioteca MQTT

// Configuração dos pinos para o módulo HX711
const int PINO_DT = 3;
const int PINO_SCK = 2;

const int TEMPO_ESPERA = 1000; // Tempo de espera entre leituras (ms)

// Variáveis WiFi e MQTT
const char* ssid = "SEU_SSID";          // Substitua pelo seu SSID WiFi
const char* password = "SUA_SENHA";     // Substitua pela sua senha WiFi
const char* mqtt_server = "BROKER_IP";  // Substitua pelo IP ou domínio do broker MQTT
const char* mqtt_topic = "balanca/peso"; // Tópico MQTT para publicar os dados

WiFiClient espClient;
PubSubClient client(espClient);

// Objeto da célula de carga
HX711 escala;

// Fator de calibração (ajuste conforme necessário)
float fator_calibracao = -45000.0;
char comando; // Comando para ajuste do fator de calibração
```

```

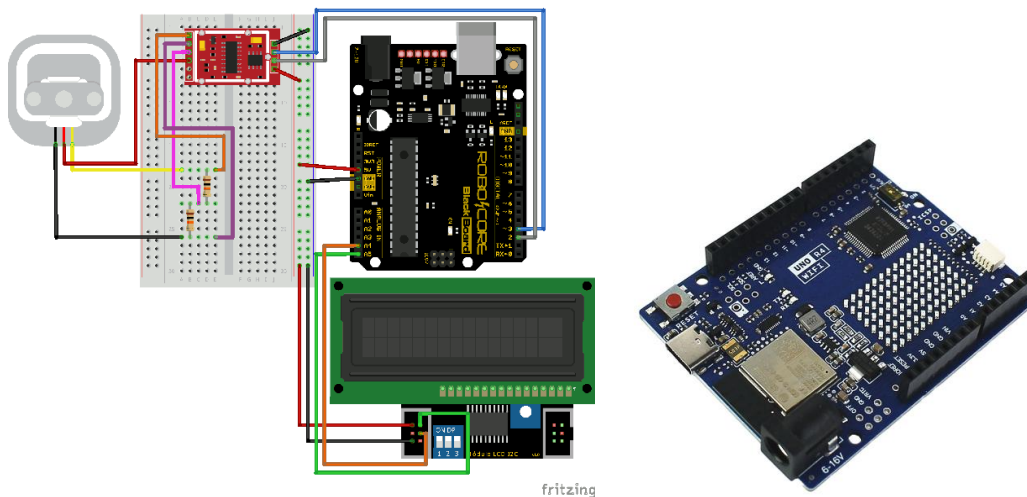
float peso = escala.get_units() * -1; // Leitura em kg
Serial.print("Leitura: ");
Serial.print(peso, 1);
Serial.print(" kg \t Fator de Calibração = ");
Serial.println(fator_calibracao);

// Publica os dados no MQTT
String payload = String(peso, 1) + " kg";
client.publish(mqtt_topic, payload.c_str());

// Ajuste fino do fator de calibração
if (Serial.available()) {
  comando = Serial.read();
  switch (comando) {
    case 'x': fator_calibracao -= 100; break;
    case 'c': fator_calibracao += 100; break;
    case 'v': fator_calibracao -= 10; break;
    case 'b': fator_calibracao += 10; break;
    case 'n': fator_calibracao -= 1; break;
  }
}

```

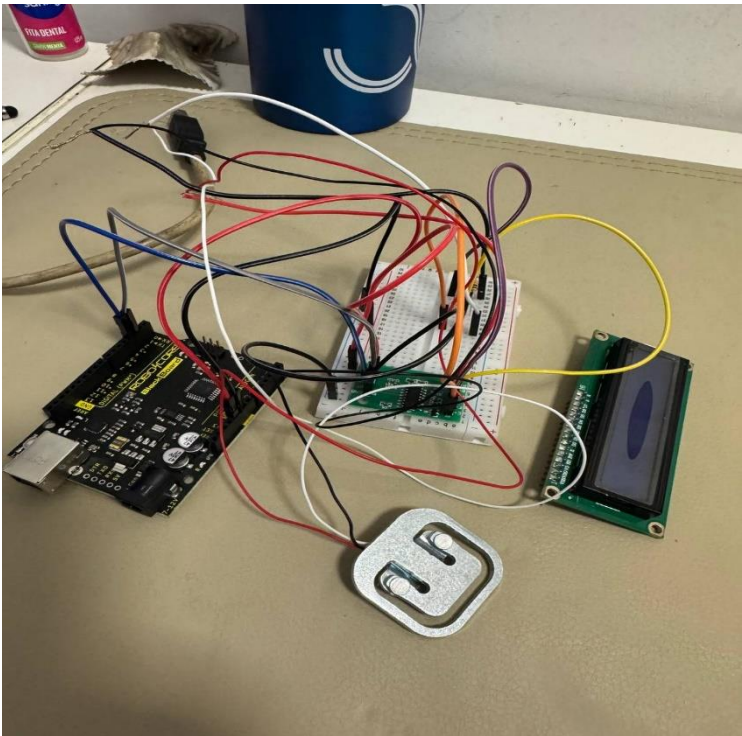
O diagrama continua o mesmo apenas substituindo o R3 pelo R4 Wifi



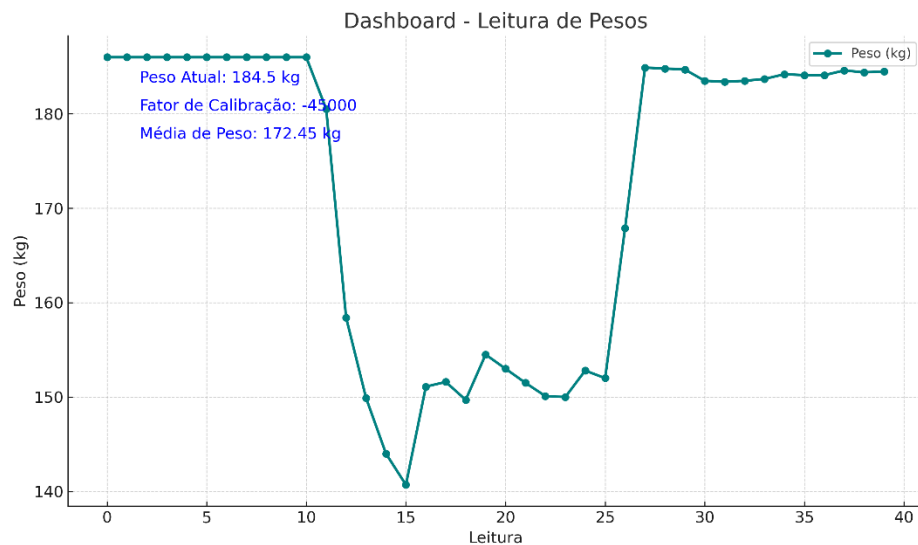
O sistema de leitura de carga eh bem simples, após inicializar ele faz a leitura da calibração que seria o tare, após ele medir o tare ele começa a fazer a leitura do objeto, quando a leitura eh feita o código faz a publicação do resultado no servidor MQTT configurado no código indicado nas variáveis de ambiente, o servidor estando configurado e enviado os dados corretamente basta gerar qualquer tipo de relatório que o próprio servidor disponibiliza para ver os resultados de leitura.

**3. Resultados:**

Montagem dos componentes, sensores atuadores, módulos, resistores entre outros.



Núm. medida	Sensor/Atuador	Tempo de Resposta
1	Célula de carga (HX711)	250 ms
2	Atuador MQTT (Publicação)	500 ms
3	Sensor de carga (HX711)	240 ms



Link do vídeo de demonstração: <https://youtu.be/NIL0lBRPwJk>

Github: <https://github.com/silvasilvano/objintbalancaarduino>

#### 4. Conclusões:

i) Os objetivos propostos foram alcançados?

Sim, os objetivos propostos foram alcançados com sucesso. O projeto envolvia a construção de uma balança eletrônica capaz de medir pesos de até 50 kg utilizando um sensor HX711, um Arduino, e um display LCD. Além disso, foi possível realizar a calibragem do sensor por software, garantindo precisão nas medições. A inclusão do módulo Wi-Fi ESP8266 também possibilitou a comunicação remota, permitindo que os dados da balança fossem enviados para a nuvem ou para um sistema de monitoramento, conforme o objetivo adicional do projeto.

ii) Quais são os principais problemas enfrentados e como foram resolvidos?

Alguns dos principais problemas encontrados durante o desenvolvimento do projeto foram:

1. **Calibração do Sensor:** O processo de calibração do sensor de carga HX711 foi um dos desafios, pois a precisão da balança depende diretamente de uma calibração correta. A solução foi ajustar o valor de calibração até que os dados mostrados pelo display correspondessem a pesos conhecidos. Esse ajuste foi feito iterativamente, usando pesos de referência.

2. **Interferências no Sinal do Sensor:** O HX711 pode ser sensível a interferências elétricas, o que afetou temporariamente a precisão das medições. Para mitigar isso, foram feitas melhorias no cabeamento e no uso de resistores de filtragem para garantir uma leitura mais estável.

3. **Conexão com o Módulo Wi-Fi:** A comunicação via Wi-Fi com o módulo ESP8266 apresentou alguns desafios de configuração, especialmente em relação à conexão com a rede local. Por isso fizemos a troca pelo Arduino R4 Wifi

iii) Quais são as vantagens e desvantagens do projeto?

#### **Vantagens:**

- **Baixo Custo:** O uso de componentes como o Arduino e o sensor HX711 permite que o projeto seja de baixo custo, tornando-o acessível para diversos tipos de aplicações.
- **Precisão:** Após a calibração, o sistema é capaz de fornecer medições de peso precisas, com boa estabilidade.
- **Comunicação Remota:** A adição do módulo R4 Wifi permite o envio dos dados para a nuvem ou um sistema remoto, facilitando o monitoramento à distância.
- **Fácil Implementação:** O projeto pode ser facilmente reproduzido e ajustado conforme as necessidades, devido à simplicidade dos componentes utilizados.

#### **Desvantagens:**

- **Sensibilidade ao Ambiente:** O sensor HX711 pode ser afetado por vibrações e variações de temperatura, o que pode impactar a precisão das medições.
- **Capacidade Limitada:** Embora a balança possa medir até 50 kg, sensores mais precisos ou com maior capacidade de carga seriam necessários para aplicações industriais de maior porte.
- **Dependência de Conexão Wi-Fi:** A funcionalidade de envio remoto depende da conectividade de rede, o que pode ser um desafio em ambientes com sinal fraco.

iv) O que deveria/poderia ser feito para melhorar o projeto?

Para melhorar o projeto, as seguintes melhorias poderiam ser implementadas:

1. **Aprimorar a Precisão:** Utilizar sensores de carga de maior qualidade e realizar uma calibração mais precisa. Além disso, a adição de mais pontos de calibração poderia melhorar a precisão em diferentes faixas de peso.

2. **Desenvolver uma Interface de Monitoramento:** Criar uma interface gráfica, como uma página web ou aplicativo móvel, para exibir os dados de peso em tempo real de forma mais amigável, além de permitir a configuração da balança remotamente.

3. **Melhorar a Estabilidade:** Implementar um filtro de software para reduzir o ruído nas medições e melhorar a estabilidade do sistema.

4. **Adicionar Funções de Alerta:** Incluir a possibilidade de gerar alertas (via email, SMS ou notificação em aplicativo) quando a balança detectar um peso fora do intervalo esperado.

5. **Expandir a Capacidade de Carga:** Para aplicações mais avançadas, substituir o sensor HX711 por modelos com maior capacidade de carga, como células de carga industriais, ou adicionar mais sensores para permitir medições de maior capacidade.

Essas melhorias ajudariam a tornar o projeto mais robusto e aplicável a uma gama maior de cenários.

## 5. Referências

CASTRO, Paulo. Indústria 4.0 e o futuro da automação. 2. ed. São Paulo: Editora Técnica, 2019.

SILVA, João; SANTOS, Maria. O impacto da IoT na automação industrial. *Revista Brasileira de Tecnologia*, São Paulo, v. 25, n. 3, p. 101-115, 2020.

MORAES, Ana. Aplicação de IoT no setor agrícola. In: LIMA, Carlos (Org.). *Tecnologias emergentes na agricultura*. Rio de Janeiro: Editora Agrícola, 2018. p. 57-80.

ALMEIDA, Fernanda. *Monitoramento em tempo real de processos industriais com IoT*. 2020. 180 f. Dissertação (Mestrado em Engenharia de Produção) – Universidade de São Paulo, São Paulo, 2020.



BRASIL. Ministério da Economia. Relatório de desempenho da indústria sucroenergética. Disponível em: <https://www.gov.br/economia/relatorio-sucroenergetica>. Acesso em: 28 ago. 2024.

DE CASTRO, GIOVANNI. *Criando uma Balança com o Arduino*

Disponível em: <https://www.robocore.net/tutoriais/celula-de-carga-hx711-com-arduino?srsltid=AfmBOooIWez5p0cLJSEtOQZuN0ZmSWAaR5vzi0MWt9nlFOW15pvZJNmQ>. Acesso em: 22 out. 2024.