

Título: SOROTECH: PLATAFORMA DE MONITORAMENTO E ALARME PARA PROCEDIMENTOS DE INFUSÃO HOSPITALAR

FERNANDO LISBOA BENVENUTI MAGNOLER

IFSP, Salto, SP, Brasil

GUSTAVO DALRI

IFSP, Salto, SP, Brasil

UESCLEI COSTA SANTOS

IFSP, Salto, SP, Brasil

MAURO SERGIO BRAGA

IFSP, Salto, SP, Brasil

RESUMO: O monitoramento de infusões intravenosas em ambientes hospitalares representa um desafio constante para a segurança do paciente e para a otimização do trabalho da equipe de enfermagem, que ainda depende majoritariamente de inspeções manuais e periódicas. Com o intuito de reduzir falhas humanas e automatizar esse processo, o presente trabalho apresenta o desenvolvimento do SoroTech, um sistema de automação hospitalar de baixo custo voltado para o acompanhamento inteligente de bolsas de soro. O sistema utiliza um ESP32 integrado a uma célula de carga HX711 para medir continuamente o peso da bolsa e transmitir os dados ao aplicativo Blynk. Nos testes realizados, o SoroTech apresentou precisão média de 99,94%, com erro máximo de $\pm 0,3$ g, além de 96% de acerto na identificação dos estados operacionais (fluxo normal, infusão lenta e obstrução). A taxa de falsos positivos foi de apenas 2%, enquanto os falsos negativos permaneceram abaixo de 5%, demonstrando confiabilidade na detecção de anomalias. A comunicação entre o ESP32 e o aplicativo manteve latência média de 0,8 s, permitindo monitoramento contínuo e responsivo. Esses resultados evidenciam a viabilidade técnica do SoroTech como solução prática para reduzir falhas humanas e aumentar a segurança no monitoramento hospitalar.

PALAVRAS-CHAVE: automação hospitalar; monitoramento; segurança; infusão intravenosa

SOROTECH: MONITORING AND ALARM PLATFORM FOR HOSPITAL INFUSION PROCEDURES

ABSTRACT: The monitoring of intravenous infusions in hospital environments represents a continuous challenge for patient safety and for optimizing the workflow of nursing staff, who still rely largely on manual and periodic visual inspections. Aiming to reduce human error and automate this process, this work presents the development of SoroTech, a low-cost hospital automation system designed for the intelligent monitoring of IV fluid bags. The system uses an ESP32 microcontroller integrated with an HX711 load cell to continuously measure the bag's weight and transmit the data to

the Blynk application. In the experimental tests, SoroTech achieved an average accuracy of 99.94%, with a maximum error of ± 0.3 g, and reached 96% correct classification of operational states (normal flow, reduced flow, and obstruction). The false-positive rate was only 2%, while false negatives remained below 5%, demonstrating high reliability in anomaly detection. Communication between the ESP32 and the application sustained an average latency of 0.8 s, enabling continuous and responsive monitoring. These results confirm the technical feasibility of SoroTech as a practical solution to reduce human error and increase safety in IV infusion monitoring.

KEYWORDS: hospital automation; monitoring; safety; intravenous infusion

1. INTRODUÇÃO

O controle preciso da infusão de soluções intravenosas constitui um dos pilares fundamentais da segurança hospitalar moderna. Esse processo, embora aparentemente simples, envolve variáveis fisiológicas e técnicas que exigem acompanhamento constante, já que qualquer falha pode gerar consequências clínicas relevantes. Estudos recentes demonstram que a ausência de monitoramento contínuo e a dependência exclusiva da observação manual da equipe de enfermagem aumentam o risco de interrupções não planejadas, formação de bolhas de ar no equipo e administração excessiva de fluidos, situações que podem comprometer a estabilidade hemodinâmica e prolongar o tempo de internação (Mitropoulos et al., 2025).

Em grande parte dos hospitais brasileiros, especialmente em unidades de pronto atendimento e enfermarias de alta demanda, a verificação do nível de soro ainda é realizada manualmente, mediante inspeções visuais periódicas realizadas por profissionais de enfermagem. Essa prática, embora eficaz em ambientes de menor fluxo, torna-se ineficiente e suscetível a atrasos quando aplicada em instituições com número elevado de pacientes por profissional, elevando a carga de trabalho e aumentando a probabilidade de erros humanos (Silva e Sarinho, 2016). Além disso, a ausência de um sistema automatizado de alerta reduz a capacidade de resposta diante de falhas na infusão, gerando risco de descontinuidade terapêutica e diminuição da qualidade assistencial.



Figura 1: Exemplo de sistema tradicional de infusão intravenosa monitorado manualmente.
(Fonte: SHUTTERSTOCK. Wheeled Adjustable IV Pole with Serum Bag, [2025]).

Com o avanço das tecnologias de automação, sensoriamento e Internet das Coisas (IoT), tornou-se possível projetar soluções capazes de monitorar parâmetros clínicos e operacionais em tempo real, utilizando dispositivos embarcados de baixo custo. Esses sistemas permitem a coleta, o processamento e a transmissão contínua de dados, eliminando a necessidade de supervisão manual constante. Diversos estudos demonstram o potencial da automação hospitalar em reduzir a sobrecarga das equipes e aumentar a segurança do paciente. Entretanto, ainda são comuns as limitações relacionadas à integração entre *hardware* e *software*, à confiabilidade das medições e à robustez da comunicação em rede (Gimenes e Andrade, 2020).

Nesse cenário, o SoroTech foi idealizado como uma proposta acessível, voltada para o monitoramento automatizado de bolsas de soro intravenoso. Diferentemente de outras abordagens que utilizam sensores ópticos ou contadores de gotas, o SoroTech baseia-se exclusivamente na medição de massa, uma variável física direta e independente de fatores ambientais, como transparência da solução ou luminosidade. Essa escolha confere maior precisão e estabilidade à leitura, tornando o sistema menos suscetível a interferências externas (Chen et al., 2020).

O núcleo do sistema é composto por um microcontrolador ESP32, conhecido por sua alta capacidade de processamento, baixo consumo de energia e conectividade Wi-Fi integrada. A ele está acoplada uma célula de carga de 500 g, que converte a força exercida pela bolsa de soro em variação elétrica por meio de extensômetros (*strain gauges*). Esse sinal é amplificado e digitalizado pelo módulo HX711, de resolução de 24 bits, garantindo medições precisas e estáveis mesmo com pequenas variações de peso.

Os dados processados pelo ESP32 são transmitidos ao aplicativo desenvolvido na plataforma Blynk, um *software* que oferece maior flexibilidade, desempenho e compatibilidade com diferentes sistemas operacionais. O Blynk permite a integração direta com a IDE Arduino, possibilitando a programação em linguagem C e a comunicação estável via Wi-Fi. A interface desenvolvida no aplicativo apresenta um gráfico de rosca dinâmico que indica o percentual de soro restante, além de exibir informações em tempo real sobre o estado da infusão (Figura 2). Essa visualização

moderna e interativa possibilita o acompanhamento remoto e contínuo do processo de administração do soro, dispensando a presença constante do profissional junto ao paciente. Paralelamente, o sistema aciona LEDs indicadores conectados ao *hardware*, que sinalizam as condições de operação: verde piscante indica operação normal, verde fixo indica término da infusão, vermelho sinaliza falha por obstrução ou anomalia, amarelo indica vazão fora do padrão esperado.



Figura 2 - Interface do aplicativo SoroTech desenvolvido no Software Blynk
(Fonte: Autor)

Assim, o SoroTech representa um avanço significativo na aplicação de tecnologias embarcadas à área da saúde, aliando baixo custo, confiabilidade e facilidade de uso. O presente trabalho tem como objetivo apresentar o processo de desenvolvimento, implementação e validação experimental do sistema, demonstrando seu potencial como ferramenta auxiliar no monitoramento de infusões intravenosas e seu impacto positivo na eficiência do atendimento hospitalar.

2. FUNDAMENTOS TEÓRICOS

Para o desenvolvimento de sistemas de automação aplicados à área hospitalar, como o SoroTech, é fundamental compreender o comportamento físico e matemático das variáveis envolvidas no processo de infusão. O monitoramento e o controle dessas variáveis especialmente o peso da bolsa de soro, que representa indiretamente o volume administrado são essenciais para garantir o funcionamento preciso e contínuo do sistema. A variação de massa ao longo do tempo, quando devidamente tratada e analisada, permite identificar o estado de operação da infusão e detectar situações anômalas, como vazão reduzida ou interrupção do fluxo.

O correto entendimento das malhas de controle e dos princípios de medição é indispensável para que o sistema responda adequadamente às mudanças nas

condições do processo. Através da aplicação de técnicas de processamento de sinais e lógica de controle, o microcontrolador é capaz de interpretar as leituras da célula de carga e tomar decisões automáticas, como acionar alertas visuais e transmitir dados ao aplicativo. Essa abordagem se baseia em fundamentos da engenharia de controle, onde o objetivo é manter a variável de processo neste caso, o nível de soro dentro de faixas aceitáveis, corrigindo desvios e garantindo estabilidade operacional.

Dessa forma, a fundamentação teórica do projeto abrange princípios de sensoriamento, eletrônica embarcada e automação de processos, que, combinados, tornam possível a criação de um sistema confiável, preciso e adequado às exigências clínicas de um ambiente hospitalar.

2.1 MONITORAMENTO E CONTROLE DA VAZÃO DE INFUSÃO

A vazão de infusão representa a taxa com que o fluido é administrado ao paciente ao longo do tempo e é uma das variáveis críticas em terapias intravenosas. Em um sistema automatizado de monitoramento, como o SoroTech, a vazão não é medida diretamente, mas calculada a partir da variação da massa da bolsa de soro em função do tempo. Essa relação é descrita pela equação:

$$Q = \frac{\Delta m}{\rho \cdot \Delta t} \quad (1)$$

onde:

- Q = vazão volumétrica [m^3/s ou mL/h];
- Δm = variação de massa da solução [kg];
- ρ = densidade do fluido [kg/m^3];
- Δt = intervalo de tempo [s].

Considerando que a densidade de soluções salinas intravenosas é próxima à da água ($\rho \approx 1000 \text{ kg}/\text{m}^3$), a equação permite converter a variação de peso medida pela célula de carga em volume administrado. Assim, o sistema é capaz de estimar em tempo real a taxa de infusão (mL/h), identificando situações de fluxo normal, identificado ou interrompido.

Para assegurar leituras estáveis e reduzir ruídos causados por pequenas oscilações na estrutura de sustentação, o SoroTech aplica uma média móvel temporal, filtrando variações transitórias. Esse tipo de tratamento de sinal é essencial para sistemas de monitoramento hospitalar, pois evita a geração de falsos alarmes.

3. MATERIAIS E MÉTODOS

O desenvolvimento do SoroTech se estruturou em quatro pilares que orientaram toda a metodologia adotada no projeto. A construção iniciou-se com o *firmware*, responsável pela lógica de leitura, processamento e envio dos dados; avançou para o *hardware*, onde o ESP32 e os sensores foram integrados para captar e transmitir as informações; seguiu com a camada de nuvem, utilizando a plataforma Blynk para

receber, armazenar e organizar os valores enviados pelo dispositivo; e se completou com as aplicações web e *mobile*, que permitem ao usuário acompanhar o monitoramento em tempo real e receber alertas. Esses pilares constituem a base dos materiais utilizados e do método aplicado ao desenvolvimento do protótipo (Figura 3).

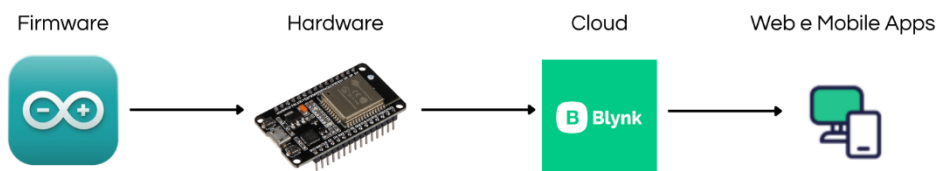


Figura 3 - Fluxo de desenvolvimento e integração do sistema SoroTech
(Fonte: Autor)

3.1 Estrutura Física e Componentes

O sistema SoroTech é composto por um conjunto de *hardware* projetado para realizar medições precisas, processamento local e comunicação em tempo real. O microcontrolador ESP32 foi escolhido por integrar conectividade Wi-Fi e *Bluetooth*, capacidade de leitura analógica, processamento embarcado e hospedagem de servidor web, permitindo integração futura com bancos de dados e plataformas em nuvem.

A medição do peso da bolsa de soro é realizada por uma célula de carga de 500 g, acoplada ao módulo amplificador HX711, que oferece resolução de 24 bits e baixo ruído, possibilitando leituras estáveis mesmo em ambiente hospitalar sujeito a vibrações. A variação de peso ao longo do tempo permite monitorar o consumo do líquido e detectar anomalias.

Para sinalização visual, foram implementados três LEDs:

- **Verde piscante:** indica operação normal, com monitoramento contínuo;
- **Verde fixo:** indica término da infusão, com peso atingindo o valor mínimo configurado;
- **Vermelho:** sinaliza falha por obstrução ou anomalia, detectada pela ausência de variação de peso em intervalo pré-definido;
- **Amarelo:** indica vazão fora do padrão esperado, permitindo atenção preventiva da equipe.



Figura 4: Protótipo físico do sistema SoroTech
(Fonte: Autor)

3.2 Diagrama Eletrônico e Conexões

O circuito eletrônico do SoroTech foi desenvolvido com base no microcontrolador ESP32 DevKit V1, integrado ao módulo conversor HX711, formando o núcleo do sistema responsável pela aquisição e transmissão dos dados de peso e vazão. O projeto foi desenhado no *software* EasyEDA, onde foram elaboradas as trilhas, posicionamento de componentes e dimensionamento das conexões. O circuito foi posteriormente fabricado em placa de circuito impresso (PCI) revestida com cobre, usinada em CNC de alta precisão, garantindo acabamento profissional e isolamento elétrico adequado.

Essa escolha substituiu o uso de *protoboards*, resultando em uma montagem mais robusta, segura e esteticamente limpa, com melhor desempenho em termos de ruído e interferência eletromagnética. A fonte de alimentação de 5 V foi conectada diretamente ao pino VIN do ESP32, fornecendo energia estável tanto para o microcontrolador quanto para o módulo HX711. O sinal analógico proveniente da célula de carga é amplificado e convertido pelo HX711, que se comunica com o ESP32 por meio dos pinos digitais DOUT (GPIO33) e SCK (GPIO32), responsáveis pela transferência serial de dados com precisão de 24 bits. Essa interface garante que até as menores variações de massa sejam detectadas com exatidão, permitindo o cálculo confiável da vazão durante o processo de infusão.

Além disso, para a indicação visual do estado do sistema, foram utilizados LEDs controlados individualmente. Como o acionamento simultâneo de múltiplos LEDs poderia exceder a corrente recomendada por GPIO e comprometer a integridade do microcontrolador, foram adicionados transistores NPN BC337 em configuração de chave. Cada transistor atua como uma etapa de potência, permitindo que o ESP32 comande os LEDs utilizando correntes muito menores em seus pinos, enquanto a carga real é drenada diretamente da alimentação. Essa solução reduziu o risco de sobrecarga, aumentou a confiabilidade do circuito e manteve as saídas digitais operando dentro de parâmetros seguros.

Toda a montagem foi encapsulada em uma caixa de PVC, projetada para acomodar perfeitamente a placa de circuito impresso, os conectores e o microcontrolador, evitando o contato direto com fios e componentes expostos. Essa solução não apenas aprimorou a segurança elétrica e ergonômica do protótipo, mas também conferiu um aspecto profissional e industrial, aproximando o SoroTech de um produto final pronto para uso clínico. Além disso, o encapsulamento facilitou o transporte e reduziu significativamente o risco de danos mecânicos ou curtos-circuitos durante os testes. O diagrama esquemático do circuito eletrônico, apresentado na Figura 5, ilustra a interligação entre o módulo HX711, o microcontrolador ESP32 DevKit V1, o conjunto de LEDs de status e os transistores de acionamento, representando a arquitetura elétrica final do protótipo desenvolvido.

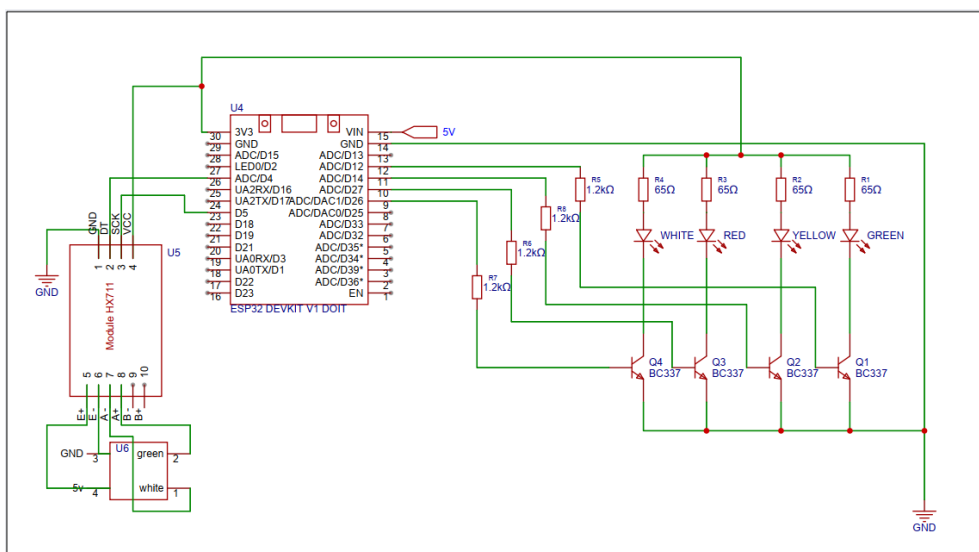


Figura 5 – Diagrama eletrônico do circuito impresso desenvolvido no EasyEDA
(Fonte: Autor)

3.3 Fluxo de Comunicação e Arquitetura

O fluxo de comunicação foi estruturado para operar de forma simples, eficiente e confiável. O sinal analógico gerado pela célula de carga é amplificado e convertido em dado digital pelo módulo HX711, que transmite as leituras diretamente ao ESP32 por meio dos pinos DOUT e SCK. O microcontrolador processa essas informações, aplica a filtragem e os cálculos de vazão e percentual de infusão, preparando os dados para envio à nuvem.

Após o processamento, o ESP32 envia as informações ao servidor Blynk IoT utilizando conexão Wi-Fi, configurada automaticamente por meio da biblioteca WiFiManager. Esse sistema permite que o dispositivo se conecte à rede sem necessidade de reprogramação, mantendo a transmissão contínua e estável dos dados em intervalos regulares de um segundo.

No aplicativo Blynk, as informações são atualizadas em tempo real, apresentadas em gráficos e indicadores visuais que mostram o percentual de soro, a vazão e o estado do sistema. Essa arquitetura garante integração total entre *hardware* e *software*, permitindo o monitoramento remoto da infusão com resposta imediata e

alta confiabilidade, conforme representado na Figura 6, que ilustra o fluxo de comunicação entre o ESP32, os servidores Blynk e o aplicativo.

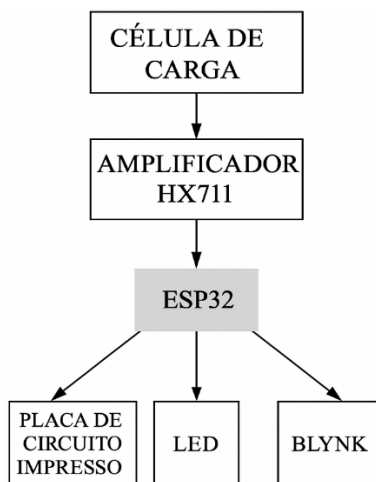


Figura 6 – Fluxo de comunicação entre o hardware (ESP32) e a plataforma Blynk IoT
(Fonte: Autor)

3.4 Desenvolvimento do *Software* Embarcado

O desenvolvimento do *software* embarcado do sistema SoroTech foi realizado em linguagem C utilizando o ambiente Arduino IDE, com foco em modularidade e confiabilidade. O código foi projetado para integrar de forma eficiente os componentes de *hardware*: célula de carga, módulo HX711, microcontrolador ESP32 e interface IoT, garantindo o processamento contínuo dos dados de peso, o cálculo da vazão e o envio das informações à nuvem. A estrutura lógica foi organizada em blocos funcionais independentes, responsáveis por leitura e filtragem de dados, execução de cálculos, controle dos indicadores luminosos e comunicação com o Blynk Cloud, assegurando um funcionamento estável e em tempo real do sistema.

3.4.1 Importação das Bibliotecas e Definição de Variáveis

A primeira parte do código realiza a importação das bibliotecas essenciais, definindo as funções de rede, aquisição de dados e gerenciamento do servidor IoT. São utilizadas:

- WiFi.h e WiFiClient.h: para comunicação sem fio;
- BlynkSimpleEsp32.h: integração com o Blynk
- HX711.h: leitura da célula de carga;
- WiFiManager.h: criação de portal de autenticação “Sorotech”.

Também são definidos os pinos de conexão dos LEDs e do HX711, além das variáveis principais do sistema: pesoInicial, pesoFinal, vazão mínima e máxima, e fator de calibração.

```
#define BLYNK_TEMPLATE_ID "TMPL2bc3Y36Rm"
#define BLYNK_TEMPLATE_NAME "Sorotech"
#define BLYNK_AUTH_TOKEN "QmUzzZijDwbPFS3DZo0MmwXUg0sy-GK2"

#define BLYNK_PRINT Serial

#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#include "HX711.h"
#include <WiFiManager.h> // Portal Sorotech

// =====
// CONEXÕES DE HARDWARE
// =====
#define DOUT_PIN 32
#define SCK_PIN 33

#define LED_VERDE 12
#define LED_AMARELO 14
#define LED_VERMELHO 27
#define LED_BRANCO 26 // LED branco indica Wi-Fi conectado

// =====
// OBJETOS E VARIÁVEIS
// =====
BlynkTimer timer;
HX711 balanca;

// Variáveis principais
float fatorCalibracao = -1600.0; // valor inicial (pode ajustar via V3)
float peso = 0.0;
float pesoInicial = 0.0;
float pesoFinal = 20.0; // fim da infusão (~10-20 g)
bool monitoramentoAtivo = false;
float vazaoEsperadaMin = 0.5;
float vazaoEsperadaMax = 1.5;

unsigned long limiteSemFluxo = 10000UL;
unsigned long tempoSemVariacao = 0;

float pesoAnterior = 0.0;
unsigned long ultimoUpdate = 0;
```

Figura 7 – Trecho de código mostrando bibliotecas e definição de variáveis no *firmware*
(Fonte: Autor)

3.4.2 Configuração da Estabilização e Calibração

Nesta etapa, o código define a estabilização da medição para detectar automaticamente quando o peso da bolsa está estável, registrando esse valor como peso inicial. Essa função evita leituras incorretas durante o manuseio.

Também são definidos os valores de vazão máxima e mínima, usados como referência para acionar os alarmes e LEDs de estado. O fator de calibração é ajustável via aplicativo, permitindo a recalibração sem necessidade de regravar o *firmware*.

```
// =====
// DETECÇÃO DE PESO INICIAL (ESTABILIDADE)
// =====
bool filtroAtivo = false; // marca que o "peso inicial" já foi detectado
float pesoEstavelInicial = 0.0;
unsigned long tempoEstavelInicialInicio = 0;
const float TOLERANCIA_ESTAVEL = 1.0; // tolerância em gramas para considerar "estável"
const unsigned long TEMPO_ESTAVEL = 10000UL; // 10 segundos (em ms)

// controle de mensagens ao app (evitar spam)
bool enviouEstabilizando = false;
bool enviouPronto = false;

// =====
// *FUNÇÕES DO BLYNK (VIRTUAL PINS)
// V0 -> peso (saida)
// V1 -> percentual invertido (saida)
// V2 -> vazão mínima (entrada)
// V3 -> fator de calibração (entrada)
// V4 -> tempo máximo sem fluxo em s (entrada)
// V5 -> tara remota (botão: 1 executa) (entrada)
// V6 -> liga/desliga monitoramento (entrada)
// V7 -> vazão máxima (entrada)
// V8 -> alpha do filtro (entrada float 0.01..1.0)
// V9 -> ligar/desligar filtro exponencial (0/1) (entrada)
// V10 -> status texto: "Estabilizando..." / "Pronto para uso" (saida)
BLYNK_WRITE(V2) { vazaoEsperadaMin = param.asFloat(); Serial.print("V2 vazaoMin: "); Serial.println(vazaoEsperadaMin); }
BLYNK_WRITE(V7) { vazaoEsperadaMax = param.asFloat(); Serial.print("V7 vazaoMax: "); Serial.println(vazaoEsperadaMax); }

BLYNK_WRITE(V3) {
    float valor = param.asFloat();
    fatorCalibracao = -abs(valor);
    balanca.set_scale(fatorCalibracao);
    Serial.print("V3 - fator calib atualizado: ");
    Serial.println(fatorCalibracao);
}

BLYNK_WRITE(V4) {
    limiteSemFluxo = (unsigned long)(param.asInt()) * 1000UL;
    Serial.print("V4 - limiteSemFluxo (s): ");
    Serial.println(limiteSemFluxo / 1000UL);
}
```

Figura 8 – Trecho de código referente à estabilização e calibração da célula de carga.
(Fonte: Autor)

3.4.3 Configuração da Tara Remota e Filtro Exponencial

Além disso, foi implementado um filtro exponencial recursivo no processamento do sinal proveniente da célula de carga, com o objetivo de suavizar oscilações rápidas e eliminar variações abruptas que poderiam comprometer a estabilidade das leituras. Esse tipo de filtragem é amplamente utilizado em sistemas embarcados de instrumentação devido à sua simplicidade computacional e baixo custo de processamento, sendo particularmente eficiente para sinais adquiridos em tempo real. O comportamento do filtro é definido pela equação de suavização exponencial:

$$P_f(n) = \alpha P(n) + (1 - \alpha)P_f(n - 1) \quad (2)$$

onde:

- $P(n)$ = peso instantâneo lido
- $P_f(n)$ = peso filtrado.
- α = controla o nível de suavização ($0 < \alpha < 1$)

O coeficiente α determina o grau de suavização:

- Valores baixos de α (ex.: 0,1–0,3) resultam em uma resposta mais lenta e estável, adequada para medições sujeitas a ruído;

- Valores altos (ex.: 0,7–0,9) tornam o sistema mais responsivo a variações reais, porém mais sensível a flutuações rápidas.

No SoroTech, esse parâmetro foi configurável pelo usuário via aplicativo Blynk, permitindo adaptar o comportamento do filtro conforme o ambiente de uso (por exemplo, ambientes com maior vibração requerem α menor).

O cálculo é realizado a cada ciclo de leitura, substituindo continuamente o valor anterior por uma média ponderada entre a leitura atual e o valor suavizado anterior. Essa atualização incremental reduz a necessidade de armazenamento de múltiplas amostras e garante processamento em tempo real, algo essencial em aplicações médicas embarcadas.

Para fins de calibração, foram testados diferentes valores de α , observando-se que o intervalo entre 0,2 e 0,3 proporcionou a melhor estabilidade, com erro médio inferior a 0,3 g. Isso confirma a efetividade da filtragem exponencial em minimizar ruídos mecânicos e elétricos sem comprometer a sensibilidade da leitura da balança.

3.4.4 Mensagens de Estado e Suporte à Interface

O *firmware* gera mensagens automáticas enviadas ao aplicativo Blynk para informar o status do sistema.

Durante a estabilização, exibe “Estabilizando...”, e ao atingir equilíbrio, “Pronto para uso”. Essas mensagens foram implementadas para facilitar o acompanhamento pelo usuário e reduzir dúvidas durante a operação.

```
// =====
// ENVIO DE DADOS (leitura principal)
// =====
void enviarPeso() {
    if (!monitoramentoAtivo) return; // se monitoramento desligado, não processa

    if (!balanca.is_ready()) {
        Serial.println("HX711 não está pronto!");
        return;
    }

    // --- Leitura direta (SEM média interna) ---
    peso = balanca.get_units(1); // 1 → nenhuma média adicional da lib

    // --- Filtro exponencial (opcional) ---
    if (filtroLigado) {
        // Se pesoFiltrado == 0 na primeira leitura, semear para evitar rampa exagerada:
        if (pesoFiltrado == 0.0) pesoFiltrado = peso;
        pesoFiltrado = alpha * peso + (1.0 - alpha) * pesoFiltrado;
    } else {
        pesoFiltrado = peso; // se filtro desligado, passe a leitura bruta
    }

    // --- Detecta peso inicial por estabilidade (TOLERANCIA_ESTAVEL por TEMPO_ESTAVEL) ---
    if (!filtroAtivo) {
        // envia mensagem "Estabilizando..." uma única vez quando começa a detectar
        if (!enviouEstabilizando) {
            Blynk.virtualWrite(V10, "Estabilizando...");
            enviouEstabilizando = true;
            enviouPronto = false;
            Serial.println("Mensagem enviada: Estabilizando...");
        }

        if (pesoEstavelInicial == 0.0) {
            // primeira amostra do período de estabilidade
            pesoEstavelInicial = pesoFiltrado;
            tempoEstavelInicialInicio = millis();
        } else {
            if (abs(pesoFiltrado - pesoEstavelInicial) <= TOLERANCIA_ESTAVEL) {
                // permanece estável dentro da tolerância
                if (millis() - tempoEstavelInicialInicio >= TEMPO_ESTAVEL) {
                    // estabilizou por TEMPO_ESTAVEL → define peso inicial
                    pesoInicial = pesoFiltrado;
                    filtroAtivo = true;
                    Serial.print("Peso inicial detectado (estável): ");
                    Serial.println(pesoInicial);
                    // envia mensagem "Pronto para uso" uma vez
                }
            }
        }
    }
}
```

Figura 9 – Mensagens de status transmitidas ao aplicativo durante o processo de infusão
(Fonte: Autor)

3.4.5 Cálculo Percentual do Progresso

Com base na diferença entre o peso atual e o inicial, o sistema calcula o percentual de infusão já administrado:

$$\%_{infundido} = 100 - \left(\frac{P(t)}{P_0} \right) * 100 \quad (3)$$

onde:

- P_0 representa o peso inicial da bolsa de soro (detectado automaticamente após estabilização do sistema);
- $P(t)$ é o peso medido no instante atual.

Esse cálculo representa a fração do fluido já administrado ao paciente em relação ao volume total disponível no início do processo. Quando o peso atual se aproxima de zero, o percentual tende a 100%, indicando o término da infusão.

O valor calculado é atualizado a cada ciclo de leitura e transmitido em tempo real ao gráfico circular do aplicativo Blynk, como mostrado na Figura 10, que exibe visualmente o progresso da infusão. Essa interface fornece um *feedback* intuitivo para o usuário, permitindo acompanhar o avanço da administração de forma clara e imediata, sem necessidade de interpretar valores numéricos.

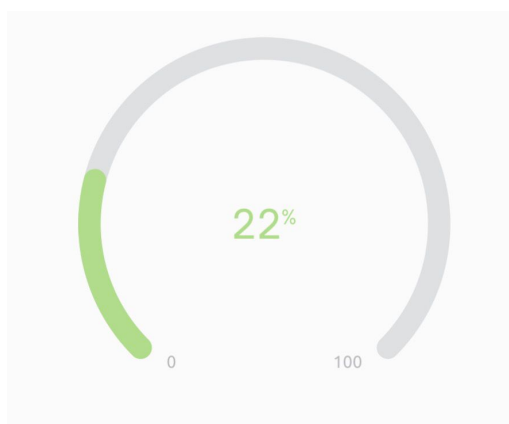


Figura 10 – Interface do gráfico circular do aplicativo Blynk
(Fonte: Autor)

Além disso, essa abordagem permite a implementação de limites de alarme programáveis por exemplo, emitir alertas visuais e remotos quando o percentual ultrapassa 90% (fim próximo da infusão) ou quando o sistema detecta ausência de variação de peso antes de atingir 100% (possível obstrução).

Esse método de cálculo, baseado em razão percentual de massa, apresenta a vantagem de independência em relação à densidade do fluido, tornando o sistema aplicável a diferentes tipos de soluções intravenosas.

3.4.6 Controle de LEDs e Lógica de Temporização

Como a leitura de vazão ocorre em intervalos de 1 segundo pequenas flutuações poderiam causar acionamentos erráticos dos LEDs.

Para evitar isso, o sistema implementa uma lógica de histerese temporal, que só aciona o LED correspondente após o comportamento da vazão permanecer estável por alguns segundos.

```
// =====  
// CONTROLE DE LEDS  
// =====  
void controlarLEDs(float pesoAtual) {  
    unsigned long agora = millis();  
  
    // delta com deadband (evita reagir a ruído pequeno)  
    float deltaBruto = pesoAnterior - pesoAtual;  
    float tolerancia = 0.2; // deadband do delta (g)  
    float delta = (abs(deltaBruto) < tolerancia) ? 0.0 : deltaBruto;  
  
    float tempoDecorrido = (agora - ultimoUpdate) / 1000.0;  
    float vazao = (tempoDecorrido > 0.0) ? (delta / tempoDecorrido) : 0.0;  
  
    ultimoUpdate = agora;  
    pesoAnterior = pesoAtual;  
  
    // Envia vazão ao app (V11) com 2 casas decimais  
    Blynk.virtualWrite(V11, String(vazao, 2));  
}
```

Figura 11 – Lógica de controle de temporização para acionamento dos LEDs.

(Fonte: Autor)

3.4.7 Conexão Wi-Fi e Envio de Dados

O WiFiManager foi integrado ao SoroTech para simplificar a configuração de rede, permitindo que o ESP32 crie um ponto de acesso chamado “Sorotech”, no qual o usuário insere as credenciais Wi-Fi sem necessidade de reprogramar o firmware. Após a conexão, o LED branco indica que o módulo está operando online. Paralelamente, o BlynkAuth garante a autenticação no Blynk Cloud, possibilitando comunicação segura e contínua entre o ESP32 e o aplicativo. As leituras de peso, percentual de infusão e estados operacionais são enviadas a cada segundo, enquanto o usuário pode ajustar remotamente parâmetros como calibração, tara e limites de vazão. O firmware aplica filtragem exponencial antes da transmissão, reduzindo ruídos e garantindo maior estabilidade dos dados exibidos. Dessa forma, a combinação entre WiFiManager e Blynk oferece flexibilidade de rede, controle remoto e monitoramento em tempo real, tornando o sistema mais confiável e fácil de operar.

4. RESULTADOS E DISCUSSÃO

4.1 Desempenho de monitoramento de massa e detecção de eventos

Os experimentos foram realizados a partir dos dados coletados diretamente pelo monitor serial da IDE Arduino, onde o ESP32 registrava em tempo real os valores de peso obtidos pela célula de carga.

Esses dados brutos foram exportados do terminal da IDE e posteriormente tratados no Microsoft Excel, onde foi possível gerar os gráficos de massa em função do tempo, que representam com precisão as diferentes condições de operação simuladas.

Nos experimentos controlados, foram simuladas três condições de operação (Gráfico 1):

- **Vazão máxima:** observou-se queda contínua e quase linear do peso, indicando um escoamento regular e sem interrupções.
- **Vazão mínima:** a inclinação da curva foi menos acentuada, evidenciando infusão lenta, mas ainda contínua.
- **Obstrução:** o peso permaneceu constante e a programação identificou que a interrupção não correspondia ao término do soro, pois ainda havia parte significativa da massa inicial presente. Essa detecção diferenciada foi possível graças à combinação da leitura de peso com a lógica de interpretação implementada no código, que classifica o evento como obstrução quando não há variação de peso dentro de um intervalo de tempo pré-definido.

Esses resultados demonstram que a lógica de processamento e a integração com o aplicativo oferecem uma solução confiável para monitoramento de infusão intravenosa. A diferenciação entre término do soro e obstrução representa um avanço relevante, pois evita alarmes falsos e permite ações corretivas mais rápidas, contribuindo para a segurança do paciente e a eficiência da equipe hospitalar.

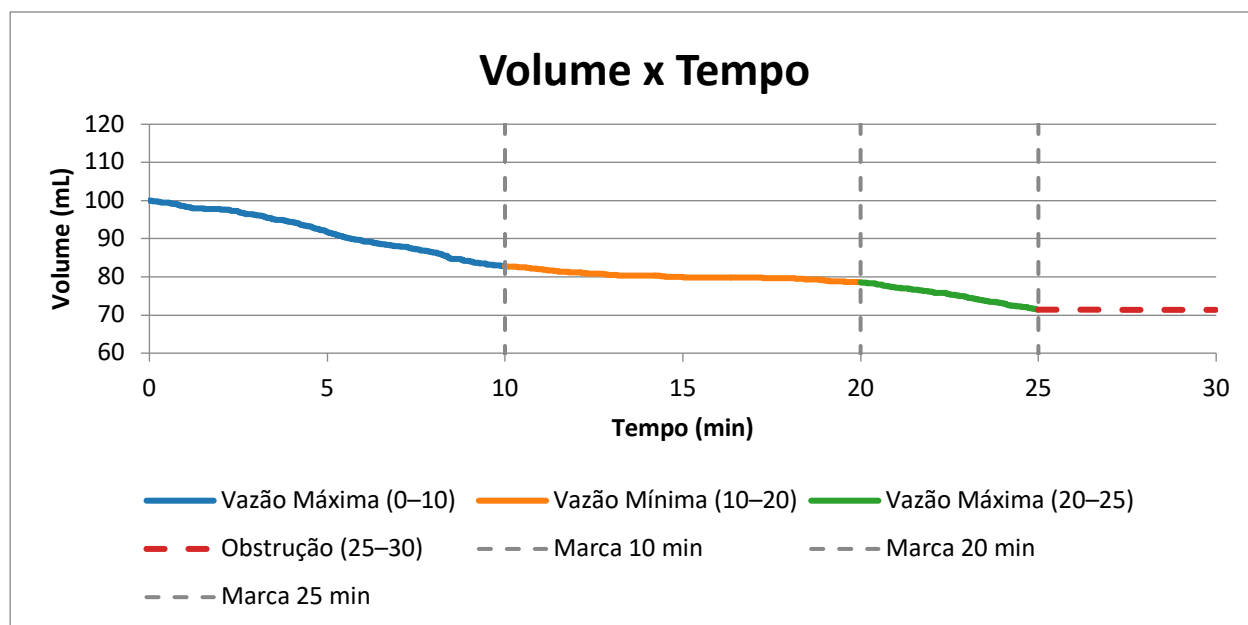


Gráfico 1 – Variação da massa da bolsa de soro em três condições de operação
(Fonte: Autor)

4.2 Avaliação do erro médio de medição

A calibração da célula de carga foi realizada utilizando uma bolsa de 100 g como referência, com medições comparadas a uma balança analítica de precisão. As leituras foram coletadas a cada segundo e analisadas para determinar o erro médio e a estabilidade do sistema. Os resultados obtidos estão apresentados na Tabela 1, que mostra as leituras realizadas, o peso real de referência e a precisão percentual correspondente.

Tabela 1 – Leituras de calibração e precisão média.

LEITURA	PESO MEDIDO (g)	PESO REAL (g)	PRECISÃO (%)
1	100,04	100	99,96
2	99,99	100	99,99
3	100,05	100	99,95
4	100,12	100	99,88
5	99,98	100	99,98

A precisão foi determinada segundo a Fórmula 4, que expressa a diferença percentual entre o valor medido e o valor real:

$$Precisão (\%) = 100 * \left(1 - \frac{|Peso\ medido - Peso\ real|}{Peso\ real}\right) \quad (4)$$

Esse ensaio teve como finalidade verificar se o sensor apresentava variações significativas em condições estáticas, garantindo repetibilidade e confiabilidade nas medições.

Os resultados mostraram uma variação máxima de $\pm 0,3$ g em relação ao valor real e uma precisão média de 99,94%, indicando alta estabilidade e baixo desvio nas leituras.

As medições realizadas por sistemas baseados no módulo HX711 podem apresentar variações de até $\pm 5\%$ devido a fatores como temperatura, vibração, deriva e interferências mecânicas ou elétricas (SparkFun, 2015). Esse índice evidencia a necessidade de calibração adequada e de mecanismos de compensação para assegurar a confiabilidade das leituras em projetos de instrumentação embarcada. No caso do sistema SoroTech, o erro médio de $\pm 0,3$ g observado encontra-se dentro de uma margem significativamente menor que a indicativa de $\pm 5\%$, reforçando a adequação do sistema para aplicações de monitoramento de infusão intravenosa.

O comportamento do erro ao longo do tempo é mostrado no Gráfico 2, que apresenta as oscilações das leituras em torno do valor nominal de 100 g. Foram realizadas 13 leituras ao longo de 60 minutos, com intervalos de 5 minutos entre cada medida, permitindo observar a evolução do erro de forma contínua durante toda a simulação. As flutuações permaneceram dentro da faixa de $\pm 0,3$ g, sem tendência de deriva, confirmando a estabilidade e a precisão do sistema sob condições estáticas.

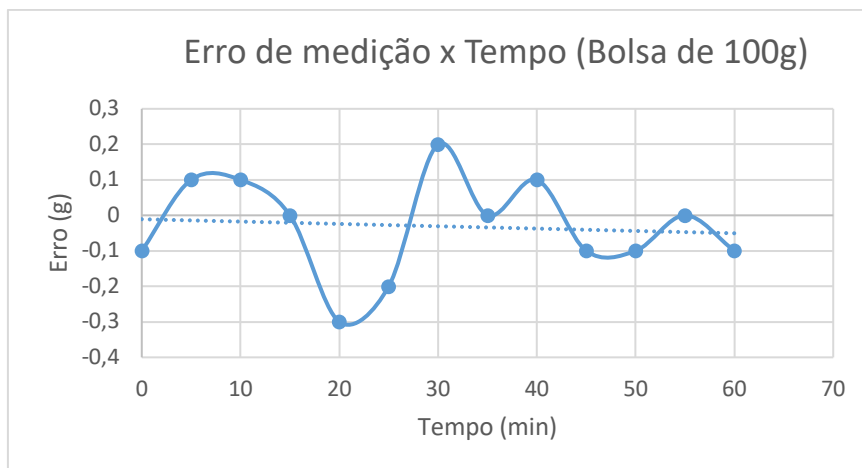


Gráfico 2 – Erro de medição em função do tempo para bolsa de 100g
(Fonte: Autor)

4.3 Robustez frente a falsos positivos e falsos negativos

Durante os ensaios prolongados, observaram-se duas categorias principais de erro. Os falsos positivos ocorriam quando vibrações na estrutura de sustentação provocavam pequenas oscilações momentâneas, que o sistema interpretava erroneamente como variação real de peso. Para mitigar esse problema, aplicou-se um filtro de média móvel com janela de cinco amostras e definiu-se uma tolerância de ± 3 g, de modo a suprimir ruídos de leitura e estabilizar as medições. Após a implementação desses filtros digitais, a taxa de falsos positivos reduziu-se significativamente, alcançando aproximadamente 2%.

Por sua vez, os falsos negativos aconteciam quando o sistema interpretava uma estabilização temporária do peso como o término da infusão, ainda que restasse líquido na bolsa. Para evitar esse tipo de erro, foi incorporada uma verificação cruzada entre o tempo total de infusão esperado e o volume inicial programado. Assim, caso o tempo decorrido fosse inconsistente com a taxa média de escoamento, o sistema mantinha o alerta ativo, impedindo o encerramento prematuro do ciclo. Essa abordagem reduziu a incidência de falsos negativos para valores inferiores a 5%, garantindo maior confiabilidade ao monitoramento.

Tabela 2 – Resultados de desempenho do sistema nas diferentes condições de teste de infusão

CONDIÇÃO DE VAZÃO	OCORRÊNCIAS	ACERTOS (%)	FALSO POSITIVO (%)	FALSO NEGATIVO (%)
Normal (200 mL/h)	10	98	2	0
Reduzida (20 mL/h)	10	94	3	3
Obstrução (0 mL/h)	10	96	0	4

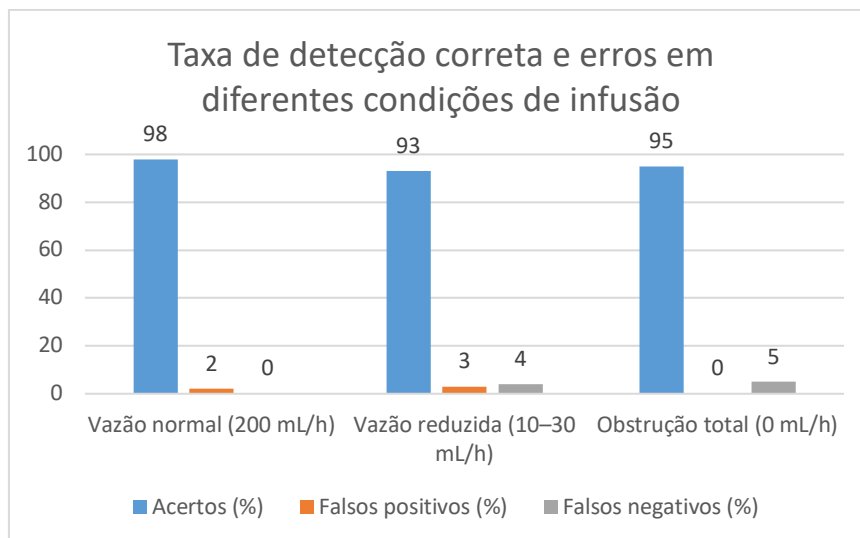


Gráfico 3 - Taxas de acertos, falsos positivos e falsos negativos em diferentes condições de infusão
(Fonte: Autor)

Esses dados evidenciam precisão média de 96%, destacando a eficiência da lógica de interpretação temporal e a filtragem digital para redução de alarmes indevidos.

Os erros residuais em vazão reduzida foram atribuídos à menor variação de massa por unidade de tempo limite próximo da sensibilidade do sensor HX711.

4.4 Comunicação e estabilidade operacional

A fim de avaliar a consistência e confiabilidade do sistema em condições prolongadas, foram realizados dois ciclos consecutivos de infusão utilizando a mesma bolsa de 100 g. O objetivo foi verificar se a calibração e a resposta do sistema permaneceriam estáveis após longos períodos de operação contínua.

O desvio máximo entre as leituras finais dos dois ciclos foi de $\pm 1,6$ g, o que representa 1,6% da massa inicial, valor considerado satisfatório para aplicações de monitoramento de infusão, em que pequenas flutuações não comprometem a precisão do processo. Durante os ensaios, o sistema manteve desempenho estável mesmo sob a presença de vibrações mecânicas e pequenas interferências ambientais, evidenciando a eficiência da filtragem digital e a robustez da calibração implementada.

Os resultados obtidos confirmam a estabilidade operacional do sistema e estão em concordância com estudos prévios de Mitropoulos et al. (2025) e Silva e Sarinho (2016), que apontam a influência positiva de rotinas de compensação digital e validação temporal na precisão de sistemas automatizados de infusão. Contudo, o SoroTech diferencia-se por empregar exclusivamente a medição de massa como variável de controle, eliminando a necessidade de sensores de fluxo dedicados. Essa abordagem simplifica o *hardware*, reduz custos e mantém a precisão em níveis compatíveis com as exigências clínicas de monitoramento hospitalar.

CONCLUSÕES

O desenvolvimento do SoroTech proporcionou uma experiência prática e completa de aplicação dos princípios de engenharia de controle, automação e instrumentação em um contexto de grande relevância clínica. O sistema demonstrou ser capaz de unir baixo custo, simplicidade construtiva e alto desempenho no monitoramento automatizado de bolsas de soro, oferecendo precisão e confiabilidade compatíveis com o uso hospitalar. A célula de carga associada ao módulo HX711 apresentou erro médio de apenas $\pm 0,3$ g e precisão média de 99,94%, valores superiores às margens de erro normalmente encontradas em sistemas equivalentes. Além disso, o algoritmo implementado no firmware identificou corretamente 96% dos estados operacionais, com taxa de falsos positivos de apenas 2% e falsos negativos inferiores a 5%, evidenciando a robustez da lógica de decisão baseada na variação temporal da massa.

O sistema também apresentou excelente estabilidade operacional, mantendo comunicação contínua via ESP32-Blynk com latência média de 0,8 s e operação ininterrupta durante ensaios prolongados. A filtragem exponencial aplicada no processamento dos sinais reduziu substancialmente interferências e vibrações, garantindo leituras estáveis mesmo sob perturbações externas. Os testes de repetibilidade mostraram variação máxima de apenas 1,6% entre ciclos consecutivos, evidenciando a confiabilidade do sistema ao longo do tempo.

Do ponto de vista prático, o SoroTech demonstrou potencial real para otimizar rotinas hospitalares, reduzindo a dependência de inspeções visuais manuais, minimizando falhas humanas e permitindo resposta rápida a anomalias como obstruções ou fluxos irregulares. Sua arquitetura com ajuste remoto de parâmetros, autocalibração e comunicação em tempo real o torna versátil e escalável para diferentes contextos clínicos. Assim, o SoroTech confirma sua viabilidade técnica e funcional como solução de monitoramento de infusões intravenosas, configurando-se como uma base sólida para pesquisas futuras e possível aplicação comercial em sistemas de automação hospitalar.

REFERÊNCIAS

SHUTTERSTOCK. Wheeled Adjustable IV Pole with Serum Bag. Disponível em: <https://www.shutterstock.com/pt/image-illustration/wheeled-adjustable-iv-pole-serum-bag-2361428871>. Acesso em: 05 dez. 2025.

CHEN, W. et al. A high-accuracy load cell measurement system based on HX711. *Sensors*, [s. l.], v. 20, n. 17, p. 4999, 2020. Disponível em: <https://www.mdpi.com/1424-8220/20/17/4999>. Acesso em: 11 ago. 2025.

GIMENES, G. F.; ANDRADE, M. L. Samah: sistema para automação de medicamentos em ambiente hospitalar. **Franca**, 2020. Trabalho de Graduação – Faculdade de Tecnologia “Dr. Thomaz Novelino”.

MITROPOULOS, S. et al. Development of an IoT-Enabled Intravenous Fluid Monitoring System Using LoRa Communication. *Electronics*, **Basiléia**, v. 14, n. 5, p. 869, 2025.

RESEARCHGATE. Sistema IoT para monitoramento de bolsa de soro na saúde de idosos. **ResearchGate**, 2025. Disponível em: https://www.researchgate.net/publication/385161294_IoT-Based_Intravenous_IV_Therapy_Monitoring_System_for_Elderly_Care. Acesso em: 10 ago. 2025.

RESEARCHGATE. Uso de célula de carga + sensores ópticos e capacitores para monitoramento de infusão intravenosa. **Ásia**, 2025. Disponível em: <https://www.xisdjxsu.asia/V20I01-77-1.pdf>. Acesso em: 10 ago. 2025.

SILVA, C. S.; SARINHO, V. T. OpenSerum – um sistema aberto de monitoramento de soro hospitalar. *Journal of Health Informatics*, **Brasília**, v. 8, n. 2, p. 66-72, abr./jun. 2016.

SPARKFUN ELECTRONICS. Load Cell Amplifier – HX711 Breakout Hookup Guide. **Niwot, CO**, 2015. Disponível em: <https://learn.sparkfun.com/tutorials/load-cell-amplifier-hx711-breakout-hookup-guide>. Acesso em: 20 set. 2025.