



## Notas Técnicas

CBPF-NT-003/25

agosto 2025

Sistema supervisorio de medidas em estrutura de banco de dados  
aplicados a equipamentos geofísicos

Ítalo Maurício, Sergio Fontes, Geraldo Cernicchiaro

## Sistema supervisorio de medidas em estrutura de banco de dados aplicados a equipamentos geofísicos

*Supervisory System for Measurement in Database Structure Applied to Geophysical Equipment*

Ítalo Maurício<sup>1,2</sup>, Sergio Fontes<sup>2</sup>, Geraldo Cernicchiaro<sup>1</sup>

<sup>1</sup> Centro Brasileiro de Pesquisas Físicas - CBPF/MCTI Rua Dr. Xavier Sigaud, 150 – Rio de Janeiro – RJ – Brasil.

<sup>2</sup> Observatório Nacional - ON/MCTI Rua General José Cristino, 77 - Rio de Janeiro – RJ – Brasil

Submetido em 22/04/2024 Aceito: 25/06/2025

**Resumo** - O sistema supervisorio apresentado neste trabalho, aplicado ao monitoramento e aquisição de dados geofísicos, desenvolvido em Python e baseado em microprocessadores, possibilita o monitoramento e reconfiguração dos sistemas de medidas, armazenamento de dados em nuvem e transmissão de dados em tempo real para multiusuários, simultaneamente. A estrutura de dados chave/valor da arquitetura do módulo de comunicação é implementada no banco de dados NoSQL, Redis, adequado para monitoramento de sistemas em tempo real. Os sistemas de medidas geofísicas estabelecem conectividade com o microprocessador de comunicação serial e o servidor hospedeiro do banco de dados. Os dados adquiridos pelos equipamentos geofísicos são enviados ao microprocessador, armazenados em um diretório local e em seguida alocados em chaves. Desta maneira, os multiusuários com acesso a estrutura de dados podem enviar comando de medidas aos equipamentos geofísicos, consultar seus respectivos status, além de terem acesso aos dados adquiridos pelos sistemas de medidas instalados em campo. O protótipo foi testado com êxito em dois sistemas de medidas geofísicas utilizados em projetos de pesquisa do Observatório Nacional.

**Palavras-chaves:** Sistema supervisorio, sistemas de medidas, banco de dados, Redis, microprocessador, equipamentos geofísicos.

**Abstract** - The supervisory system presented in this work, applied to the monitoring and acquisition of geophysical data, developed in Python and based on microprocessors, enables the monitoring and reconfiguration of measurement systems, data storage in the cloud, and real-time data transmission to multiple users simultaneously. The key/value data structure of the communication module architecture is implemented in the NoSQL database, Redis, suitable for real-time system monitoring. Geophysical measurement systems establish connectivity with the serial communication microprocessor and the hosting server of the database. The data acquired by the geophysical equipment is sent to the microprocessor, stored in a local directory, and then allocated to keys. In this way, multiple users with access to the data structure can send measurement commands to the geophysical equipment, check their respective statuses, and access the data acquired by the measurement systems installed in the field. The prototype was successfully tested on two geophysical measurement systems used in research projects at the National Observatory.

**Keywords:** Supervisory system, measurement systems, database, Redis, microprocessor, geophysical equipment.

## 1. Introdução

O monitoramento em tempo real dos equipamentos geofísicos é fundamental para garantir qualidade e confiabilidade das medidas adquiridas e reduzir o custo das atividades de campo, evitar o deslocamento de técnicos até a estação, muitas vezes situadas em lugares de difícil e custoso acesso. Além do mais, monitorar os equipamentos possibilita corrigir falhas dos sistemas com maior precisão e rapidez. Os equipamentos geofísicos empregados nos métodos magnetotelúricos e eletrorresistividade são utilizados nas aquisições de dados geofísicos para estimar a condutividade elétrica da subsuperfície. Essas medidas são utilizadas para diagnosticar riscos de deslizamento, rompimento de barragens, dinâmicas de aquíferos, aterros sanitários e mapear estruturas condutivas, alcançando profundidades desde alguns poucos metros a algumas centenas de quilômetros.

Algumas destas estruturas podem ser indicadores de depósitos minerais ou reservatórios de hidrocarbonetos. Normalmente, estas medidas são realizadas por operadores situados em campo, interagindo diretamente com os equipamentos. Em ambientes que ofereçam risco à vida dos operadores, a coleta diária das medidas realizadas por esses equipamentos poderá ser inviabilizada em função das dificuldades encontradas. Além disso, o sistema de monitoramento remoto possibilita a supervisão dos sistemas de medida 24h por dia por diferentes operadores ou algoritmos automatizados de identificação de falhas, ampliando a funcionalidade e a capacidade operacional dos equipamentos geofísicos existentes no mercado.

Os projetos de pesquisa que utilizam estações geofísicas de eletrorresistividade e magnetotelúricas, instaladas em atividades de campo, têm motivado a supervisão e o monitoramento destas instalações. O monitoramento em tempo real dos equipamentos geofísicos, realizado em uma sala de controle, são importantes para garantir o controle da qualidade e a continuidade dos dados, o processamento dos dados em tempo real, assegurando assim agilidade no diagnóstico das estruturas geológicas monitoradas.

Os sistemas de medidas geofísicas normalmente são configurados para realizar medidas com apenas um arranjo de aquisição por vez. Utilizando o sistema de monitoramento remoto, é possível programar os equipamentos para diferentes configurações sem a

necessidade da intervenção presencial do operador. Diante dessa possibilidade, é possível reduzir consideravelmente os custos das campanhas de aquisição e otimizar os recursos financeiros dos projetos de pesquisa, direcionando-os para outras demandas.

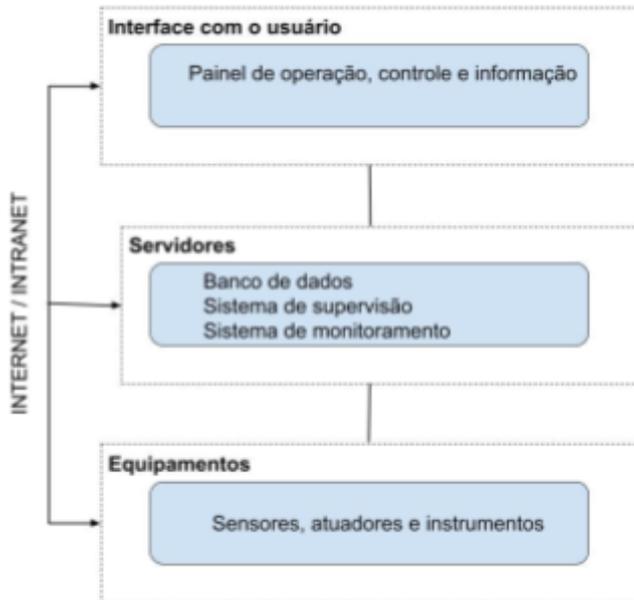
A transmissão dos dados é baseada em uma arquitetura de banco de dados NoSQL, viabilizando a aplicação do módulo de comunicação em sistemas de medidas geofísicos instalados em estações de aquisição de dados em tempo contínuo.

Um dos objetivos do módulo de comunicação é ampliar a capacidade operacional dos sistemas de medidas, visto que a maioria destes equipamentos não tem um sistema de transmissão e monitoramento de dados disponibilizado pelo fabricante. A capacidade de coletar os dados, reconfigurar os equipamentos e monitorar a dinâmica das estações remotamente, oferecida pelo módulo, acrescenta robustez aos sistemas geofísicos, além de aumentar as chances de obter medidas com boa qualidade.

O protótipo utilizado nas campanhas de aquisição de dados magnetotelúricos do projeto de pesquisa PETRONAS MULTIFÍSICA do Observatório Nacional garantiu a supervisão da estação em tempo real e acesso às medidas remotamente. Tais contribuições proporcionaram avaliação contínua das medidas, acesso diários aos dados e seu respectivo processamento, algo que até então não tinha sido realizado pelos pesquisadores envolvidos no projeto. Do ponto de vista técnico-científico, houve um avanço significativo neste novo modelo de aquisição e monitoramento dos sistemas de medidas.

## 2. Sistema Supervisório - SCADA

Sistemas de Controle e Aquisição de Dados Supervisionados, conhecidos como SCADA (do inglês *Supervisory Control and Data Acquisition*), referem-se a sistemas que empregam tanto software quanto hardware para monitorar e supervisionar variáveis, dispositivos, subprocessos e processos industriais e não industriais [18][26]. Esses sistemas também gerenciam equipamentos conectados por meio de servidores de comunicação dedicados[1][2][15] Na Figura 1, apresentamos um fluxograma simplificado que ilustra a estrutura e o funcionamento do modelo SCADA.



**Figura 1 - Arquitetura simplificada em três camadas do sistema SCADA**

Por meio do SCADA, é possível realizar medições dos instrumentos instalados em campo, criar gráficos de tendência, manter registros históricos das variáveis do processo, plotar séries temporais e realizar análise espectral dos dados adquiridos [11]. A capacidade de analisar remotamente os dados em tempo real aprimora o diagnóstico da dinâmica dos processos e o controle dos sistemas de medição monitorados. Essas análises proporcionam uma resposta ágil para a tomada de decisões assertivas, permitindo correções ou a manutenção dos processos de forma eficiente e estável.[1][8]

Os sistemas SCADA requerem um meio físico para realizar a aquisição de dados dos sistemas de medidas in situ e a transmissão das informações ao controlador. Geralmente, esse meio físico adere a protocolos de comunicação como RS-232, USB ou Ethernet. Inerentemente, os sistemas SCADA oferecem robustez, praticidade e modularidade, sendo compatíveis com diferentes sistemas operacionais e hardwares, e mantendo-se constantemente atualizados. Além disso, é crucial manter as informações e dados manipulados em um nível elevado de segurança para garantir a privacidade e veracidade das informações. Os SCADA normalmente são desenvolvidos por empresas comerciais com base em códigos e arquiteturas fechadas. No entanto, neste projeto, o SCADA apresenta uma arquitetura, baseada em

código aberto, que pode ser facilmente replicada, mantendo os níveis necessários de segurança para preservar as informações [31]. A autenticação de login e senha do usuário no banco de dados se mostrou suficiente para garantir a segurança do sistema e evitar violações na estrutura de dados do Redis, um banco de dados NoSQL de código aberto, detalhado na seção 3.

Os recursos empregados na implementação do SCADA deste projeto são exclusivamente constituídos por softwares e hardwares de código aberto. Destaca-se o computador de placa única Raspberry, utilizado para estabelecer a comunicação entre os sistemas de aquisição, como equipamentos geofísicos, e os softwares, servidores e interfaces desenvolvidos na linguagem Python 3[12][13] [20][28].

### 3. Banco de Dados

Um banco de dados consiste em um conjunto de informações organizadas e estruturadas de acordo com padrões definidos pelo desenvolvedor. Para utilizar o banco de dados de maneira eficiente e segura, os usuários precisam familiarizar-se com os padrões e características da sua estrutura, permitindo consultas, atualizações e armazenamento seguro de novas informações. Na era digital, o termo 'banco de dados' está intrinsecamente relacionado às informações armazenadas eletronicamente em servidores dedicados ou *storages*. [23]

Existem diferentes tipos de bancos de dados, sendo os dois principais:

1. Bancos de dados relacionais (RDBMS - *Relational Database Management System*): Esses bancos de dados organizam as informações em tabelas, onde os dados são estruturados em linhas e colunas. O SQL (*Structured Query Language*) é uma linguagem comumente usada para consultar e manipular bancos de dados relacionais. [23]

2. Bancos de dados não relacionais (NoSQL ou Not Only SQL): Esse tipo de banco de dados abrange várias tecnologias que não seguem o modelo relacional. Eles são mais flexíveis e adequados para casos de uso específicos, como armazenamento e processamento de grandes volumes de dados não estruturados.

Os bancos de dados desempenham um papel crucial em aplicativos e sistemas modernos, abrangendo desde websites e aplicativos móveis até sistemas empresariais complexos, assegurando eficiência, segurança e escalabilidade na manipulação de informações. No âmbito deste projeto, o banco de dados é essencial para garantir o funcionamento do sistema supervisorio. Todas as configurações, comandos de medição e armazenamento dos dados coletados são realizados por meio das estruturas de dados definidas no banco de dados Redis.[32]

### 3.1 Redis

O Redis, que significa *Remote Dictionary Server*, é um banco de dados NoSQL de código aberto, baseado em chave-valor. Notável por proporcionar tempos de resposta inferiores a um milissegundo, o Redis é ideal para aplicações em tempo real, sistemas de mensagens e transmissão de mídia.[24] [25][32]

Algumas características são listadas abaixo:

- Conjuntos ordenados: conjuntos ordenados por um valor
- Hashes: uma estrutura de dados para armazenar uma lista de campos e valores
- Bitmaps: um tipo de dados que oferece operações de nível de bits
- HyperLogLogs: uma estrutura de dados probabilística para estimar os itens únicos em um conjunto de dados
- Transmissões: uma fila de mensagens de estrutura de dados de log
- Dados geoespaciais: um mapa de registros com base em longitude/latitude, “proximidades”
- JSON - um objeto aninhado e semiestruturado de valores nomeados que suportam números, strings, booleanos, matrizes e outros objetos

O Redis tem uma grande variedade de estruturas de dados para atender às necessidades de suas aplicações. Os tipos de dados do Redis incluem:

- Strings - dados em texto ou binários com tamanho de até 512 MB

- Listas - uma coleção de strings na ordem em que foram adicionadas
- Conjuntos: uma coleção não ordenada de strings com a capacidade de fazer intersecção, união e diferenciação de outros tipos de conjunto

Todos os dados do Redis são armazenados na memória principal do servidor, possibilitando o acesso a dados com baixa latência e alta taxa de transferência. Ao contrário dos bancos de dados tradicionais, os *datastores* na memória eliminam a necessidade de acessos ao disco, reduzindo a latência do mecanismo para microssegundos. Além disso, para garantir persistência, o Redis oferece suporte a backups em momentos anteriores, copiando o conjunto de dados do Redis para o disco [21].

O Redis é compatível com diversas linguagens de programação, como Java, Python, PHP, C, C++, C#, JavaScript, Node.js, Ruby, R, Go, entre outras. Uma das principais vantagens do Redis é sua simplicidade e facilidade para consultar e armazenar dados.

**Consulta** - Acessa o valor armazenado na chave através do comando GET

```
redis.GET ('CHAVE')
```

**Armazenar** - Armazena um valor a chave através do comando SET

```
redis.SET ('CHAVE', 'VALOR')
```

O acesso e armazenamento de dados podem ser realizados em qualquer momento do processo, bastando estar conectado ao servidor onde o Redis está instalado.[6]

## 4. Métodos Geofísicos

Existem diversas técnicas geofísicas destinadas a estimar os contrastes das estruturas geológicas na subsuperfície. Neste estudo, empregamos dois métodos nos quais a propriedade física de interesse para delimitar essas estruturas é a condutividade elétrica. O método MT é amplamente utilizado pelo grupo de pesquisadores do Observatório Nacional desde a década de 70 e está presente em diversos projetos da instituição.

Por sua vez, o método de eletrorresistividade está sendo avaliado e testado para o monitoramento de barragens, um tema de grande sensibilidade e relevância para o cenário brasileiro.

#### 4.1 Eletrorresistividade

O método da eletrorresistividade fundamenta-se na análise do potencial elétrico, tanto dos campos elétricos naturais presentes na crosta terrestre quanto dos campos induzidos por fontes antrópicas. Através das medições do potencial elétrico em dois pontos da superfície, é possível determinar a condutividade dos corpos condutores no subsolo e estimar a distribuição das estruturas geológicas existentes.[4][27]

No método geofísico de eletrorresistividade existem algumas técnicas de aquisição de dados. Podemos separar as técnicas existentes em dois grandes grupos: Sondagem Elétrica Vertical - SEV e Caminhamentos Elétricos. Existe uma grande gama de arranjos e configurações possíveis dos eletrodos. Os levantamentos utilizando as SEV's são executados quando se deseja obter informações da variação vertical da resistividade da subsuperfície. Por sua vez, os caminhamentos são aplicados quando há interesse em obter informações da variação lateral da resistividade, mantendo-se uma profundidade teoricamente constante. O arranjo para realizar medidas de resistividade consiste basicamente em um sistema de quatro eletrodos, dois deles (**A** e **B**) são usados para injetar uma corrente elétrica contínua conhecida (**I**) e dois eletrodos **M** e **N** são usados para medir a diferença de potencial ( $\Delta V$ ) do espaçamento entre eles. A partir da relação entre corrente e tensão, estima-se a resistividade aparente ( $\rho_a$ ) da subsuperfície, Eq. 1. Vale ressaltar que o espaçamento entre os eletrodos é diretamente proporcional à profundidade de investigação. Algumas características do solo, como a saturação, podem influenciar nos valores da medida e na determinação da profundidade de investigação. [4][9]

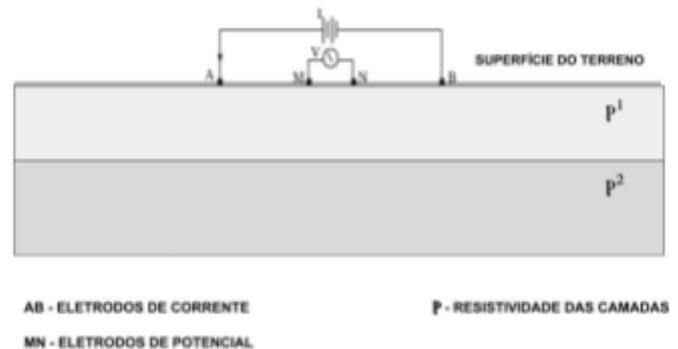
$$\rho_a = k \cdot \Delta V / I \quad (1)$$

- $\rho_a$  - Resistividade aparente
- $k$  - Fator geométrico
- $\Delta V$  - Diferença de potencial
- $I$  - Corrente injetada no solo

Os arranjos de aquisição podem ser classificados em quatro grupos:

- Schlumberger
- Wenner
- Polo-Dipolo
- Dipolo-Dipolo

Os arranjos no método de eletrorresistividade são caracterizados e diferenciados pelo espaçamento entre os eletrodos de corrente e tensão. A Figura 2 sintetiza o princípio fundamental do método de eletrorresistividade.[4]



**Figura 2 - Arranjo geral do sistema de aquisição de dados de eletrorresistividade**

#### 4.2 Magnetotelúrico

O método geofísico magnetotelúrico (MT), amplamente empregado em prospecções minerais e de hidrocarbonetos, fundamenta-se no registro das variações dos campos magnéticos e elétricos naturais da Terra para estimar a impedância, conforme a Equação 2, da subsuperfície terrestre. Esse método pressupõe que o campo eletromagnético da Terra é influenciado pelas características elétricas da subsuperfície [16]. Em essência, o MT determina a condutividade elétrica das formações rochosas a partir da razão entre os campos elétricos ( $E$ ) e magnético ( $B$ ) [3][5][7].

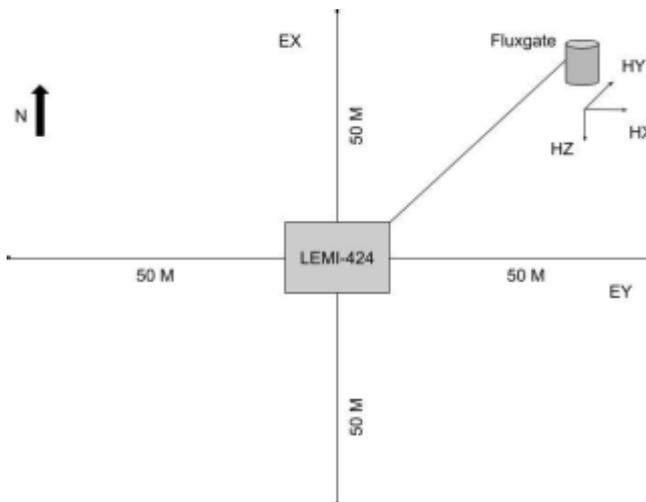
$$Z = E * (i, j) / B * (i, j) \quad (2)$$

As equações fundamentais do eletromagnetismo que descrevem a propagação da onda eletromagnética no meio são a base do método MT. Dentre as equações podemos destacar duas. A equação da continuidade do campo magnético (B), equação 3 e a equação de Maxwell para o campo elétrico (E), equação 4.

$$\nabla \times B = \mu_0 \sigma E + \mu_0 \epsilon_0 dE/dt \quad (3)$$

$$\nabla \times E = -dB/dt \quad (4)$$

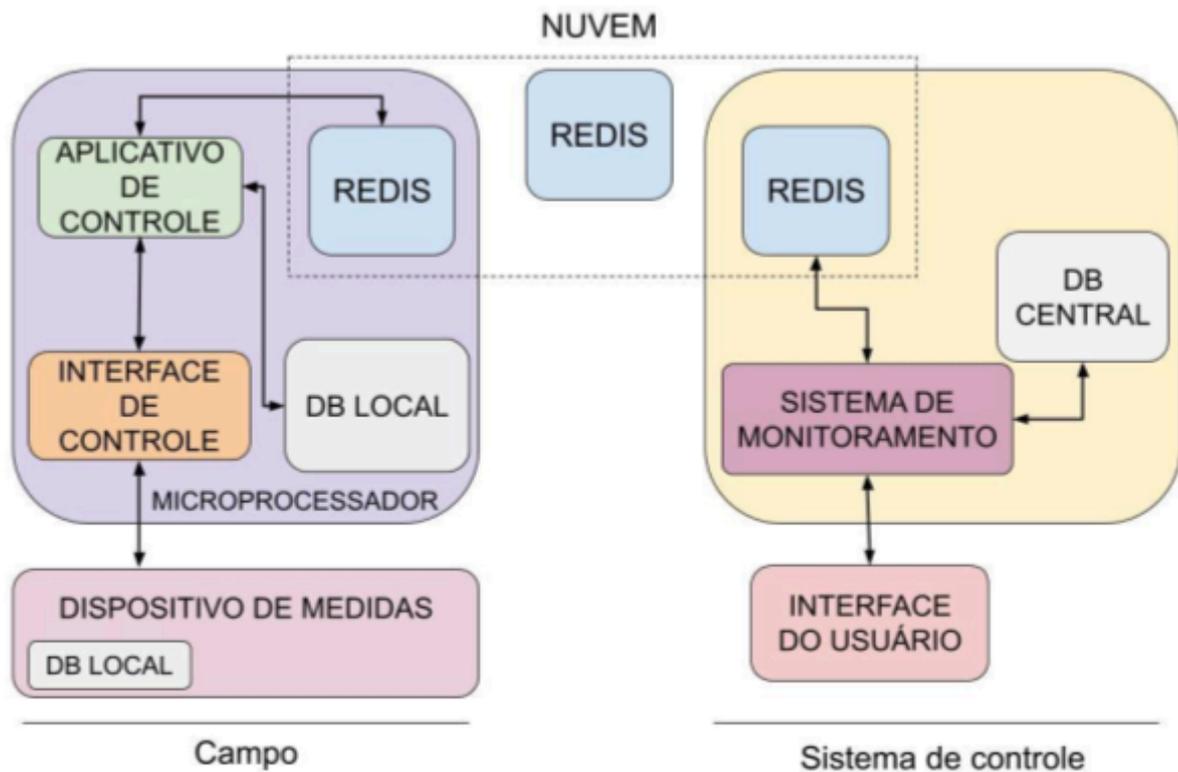
O arranjo das instalações dos sistemas de medidas magnetotélúricos obedecem a perpendicularidade entre os campos elétricos e magnéticos, de modo que as componentes registradas são ortogonais entre si. A seguir, na figura 3 é apresentado um modelo de arranjo bastante utilizado nas aquisições de MT [22].



**Figura 3 - Arranjo utilizado na aquisição de dados MT**

## 5. Visão Geral da Arquitetura

O módulo de comunicação foi concebido mediante a integração de hardwares e softwares de código aberto, inicialmente desenvolvidos para aplicações diversas. A união dessas ferramentas confere ao projeto uma abordagem multidisciplinar. A linguagem de programação Python 3, em conjunto com algumas bibliotecas escritas para o Redis e o mini-computador Raspberry, são as principais ferramentas empregadas na implementação deste módulo de comunicação. A arquitetura do sistema supervisorio, conforme ilustrado na Figura 4, compreende a comunicação entre os sistemas de medidas, o microprocessador e o banco de dados Redis. A comunicação remota entre a sala de controle e o equipamento, localizado fora do campo de visão do usuário, é realizada por meio do banco de dados instalado em um servidor dedicado [10]. No contexto deste projeto, o servidor dedicado está instalado em uma máquina virtual, disponibilizada gratuitamente pelo site <https://www.redis.com> [31][32].



**Figura 4 - Diagrama de blocos do sistema supervisório**

A arquitetura do sistema supervisório proposta neste trabalho viabiliza a interação remota com os sistemas de medidas por meio de interfaces gráficas e aplicativos de controle. As interfaces de controle podem ser desenvolvidas em Python, conforme as especificações dos sistemas de medidas, ou utilizar o software fornecido pelo fabricante. Em ambos os casos, a interface de controle envia comandos aos equipamentos pela porta serial e recebe os valores das medidas realizadas. Os comandos enviados podem incluir operações de calibração, aquisição, reconfiguração ou reset. Os valores das medidas são então transmitidos ao aplicativo de controle e armazenados na memória local do microprocessador, registrados em um arquivo de texto, e alocados nas chaves do Redis.

O aplicativo de controle é encarregado de gerenciar o fluxo de comandos e armazenar os valores das medidas nas chaves do banco de dados Redis. O acesso às chaves e valores do banco de dados ocorre em tempo real. A estrutura do aplicativo de controle, desenvolvida em Python, estabelece uma conexão com o Redis ao ser inicializada [19][25].

O banco de dados Redis permite que todos os comandos e armazenamentos de medidas sejam efetuados em nuvem. Um servidor, seja uma máquina dedicada ou uma máquina virtual, armazena a estrutura de dados chave/valor do banco de dados. Após a instalação, a conexão com o Redis é apresentada no front-end do usuário após exibirem os valores das chaves estabelecida por meio do endereço IP, porta de acesso e senha. Isso significa que o acesso às informações pode ser realizado de qualquer máquina ou aplicativo que tenha a biblioteca do Redis instalado e as respectivas informações de acesso e segurança.

Os usuários remotos terão à disposição interfaces gráficas (GUIs) desenvolvidas em PySimpleGUI. Por meio dessas interfaces, é possível consultar a disponibilidade dos equipamentos e enviar solicitações de medidas.

Através das GUIs mencionadas nos parágrafos anteriores, os usuários inserem requisições de consulta e comandos de aquisição. No processo de solicitação de medidas, o *front-end* do módulo de comunicação associa a opção selecionada às chaves do Redis. Para cada valor armazenado nas respectivas chaves, há uma rotina em Python que executa o tipo de medida selecionada. O microprocessador, conectado aos sistemas de medidas, consulta as chaves do Redis e envia o comando de configuração à interface de controle. O equipamento é devidamente configurado e em seguida, inicia o processo de aquisição. Após a aquisição, os dados são armazenados na estrutura chave/valor dedicada a receber o valor das medidas. Na solicitação de consulta, status e resultados, o front-end verifica os valores das chaves dedicadas a armazenar essas informações e os exibe ao requisitante. O conteúdo dessas chaves é apresentado no front-end do usuário após exibirmos valores das chaves do Redis.

O sistema de monitoramento tem a capacidade de interagir com o equipamento por meio dos comandos enviados ao banco de dados ou diretamente pela conexão com o microprocessador Raspberry utilizando o VNC Viewer [30].

No entanto, por questões de segurança do sistema supervisorio, apenas usuários root têm acesso ao microprocessador através do VNC Viewer. Esse tipo de acesso, possibilita o controle direto do mini-computador e a transferência de arquivos tanto para a máquina remota, na sala de controle, quanto para o microprocessador instalado em campo, por meio do SSH - *Secure Socket Shell* nativo desta aplicação. Estes arquivos podem incluir arquivos de configuração, rotinas em Python, drivers de comunicação e arquivos de medidas salvos na memória do microprocessador.

Existem diversos microprocessadores disponíveis no mercado, e entre eles, a Raspberry Pi 3 foi a escolhida para a implementação do projeto. A familiaridade com o dispositivo, instalação do sistema operacional, linguagem python nativa e facilidade conexão com os periféricos, foram fatores cruciais para a seleção. Além disso, as especificações técnicas do hardware foram consideradas no processo de escolha do microprocessador. A capacidade de utilizar interfaces LAN, WLAN, portas USB e saída HDMI contribuem positivamente para a implementação da arquitetura sugerida neste projeto.[13][29]

O sistema operacional embarcado no microprocessador é o Raspberry PI OS, um sistema baseado no Linux Debian, estável e amplamente utilizado em projetos de prototipagem. [17]

Conforme indicado no diagrama da Figura 4, há redundância no armazenamento das medidas. Os dados são registrados em quatro instâncias: na memória local do equipamento, no *storage* do microprocessador, no servidor do Sistema de Monitoramento e no *storage* da máquina do usuário. Caso seja necessário liberar espaço no banco de dados, os valores armazenados nas chaves podem ser excluídos. Isso significa que os dados armazenados no servidor do Sistema de Monitoramento podem ser temporários, e o usuário root tem autonomia para definir por quanto tempo os valores das medidas deverão ser mantidos.

## 5.1 Processos de configuração do sistema

Nesta seção, serão abordadas algumas etapas de configuração e inicialização do sistema. É crucial ressaltar que os fluxogramas simplificados ilustram as etapas realizadas para configurar o banco de dados, instalado tanto no servidor quanto no microprocessador, dedicado a estabelecer a conexão com os sistemas de medidas instalados em campo.

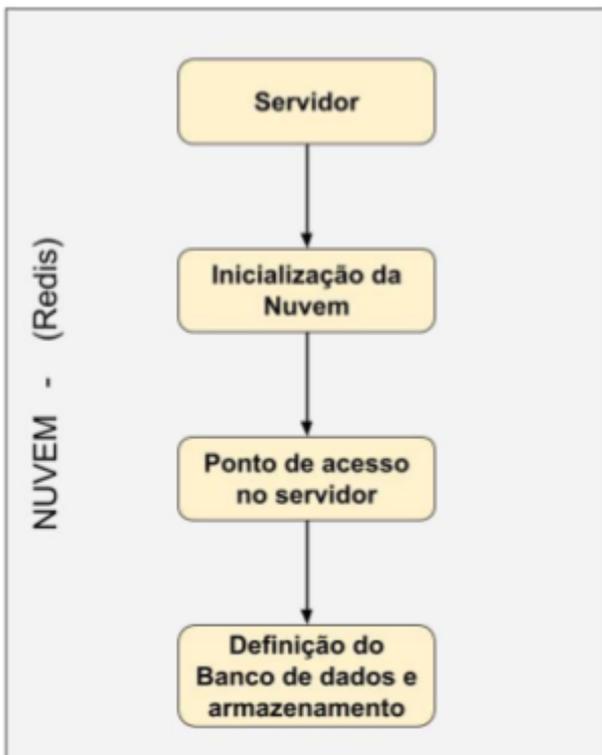
### 5.1.1 Configuração do servidor em nuvem

O fluxograma apresentado na Figura 5 ilustra as etapas necessárias para configurar o banco de dados. Inicialmente, é essencial definir o servidor e suas especificações. O banco de dados Redis pode ser instalado localmente em um servidor dedicado ou em um servidor virtual na nuvem. Por uma questão de simplicidade e didática, neste trabalho, o Redis foi hospedado em um servidor virtual fornecido pela RedisLab [6][31]. Os valores armazenados nas chaves do Redis podem ser acessados por qualquer máquina que tenha as informações de login e senha do banco de dados em nuvem.

Considerando o uso do RedisLab, a inicialização da nuvem e suas respectivas configurações, incluindo o ponto de acesso e a capacidade de armazenamento limitada a 30 MB, são determinadas pelos mantenedores do site. O ponto de acesso do Redis é definido pelos seguintes parâmetros:

- ``host_redis`` (string): Semelhante ao endereço URL.
- ``port_redis`` (inteiro): Porta de acesso.
- ``password_redis`` (string): Senha definida pelo usuário."

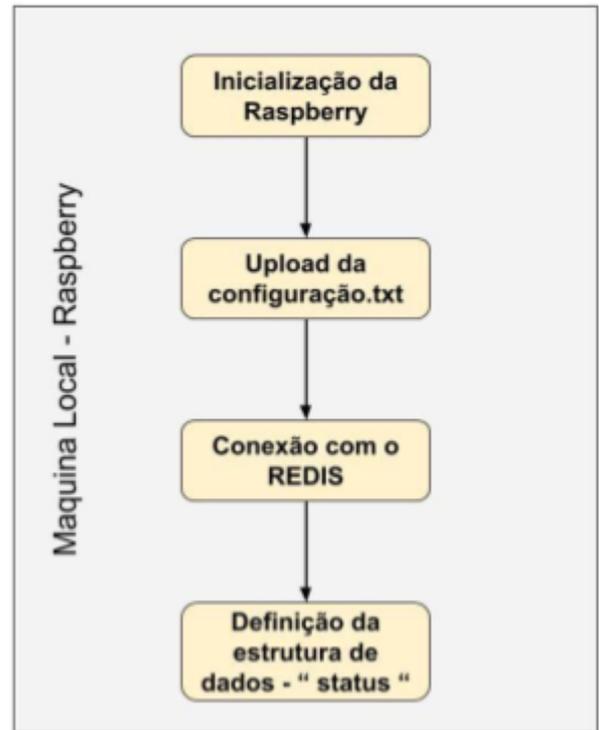
Após a configuração do servidor, conforme indicado no fluxograma, torna-se possível utilizar as definições do banco de dados nos códigos em Python. Na próxima seção, será exemplificado como estabelecer conexão com o Redis e ter acesso às respectivas estruturas de dados.



**Figura 5- Fluxograma de inicialização do banco de dados**

### 5.1.2 Inicialização da máquina local

A máquina local, Raspberry, é encarregada de estabelecer a conexão entre os sistemas de medidas e o banco de dados. Após a conexão com o Redis, torna-se possível identificar os valores associados às chaves da estrutura de dados e enviar comandos ao equipamento. O acesso às medidas pode ser realizado diretamente através da leitura da porta serial ou dos arquivos de texto armazenados na memória local do microprocessador. Para tanto, é necessário seguir as etapas representadas pelo fluxograma apresentado na Figura 6.



**Figura 6 - Fluxograma de inicialização do sistema**

A inicialização da Raspberry ocorre quando o sistema de alimentação de 5V/2A é devidamente acionado. Nesta etapa, considera-se que o sistema operacional Raspbian já está instalado e configurado no cartão SD.

Após a inicialização, uma rotina em Python é executada automaticamente. Nessa rotina, ocorre a configuração da porta serial à qual o sistema de medida está conectado, bem como do ponto de acesso do Redis. Os parâmetros de configuração são definidos em um arquivo chamado `file_config.txt`, presente no mesmo diretório da sub-rotina escrita em Python e da interface de controle. Uma vez estabelecida a conexão com o Redis, automaticamente é criada uma estrutura de dados chave/valor pré-definida no código Python. Essa estrutura define as chaves necessárias para realizar a configuração, medida e status dos sistemas de medidas. A Tabela 1 apresenta a estrutura de dados simplificada do sistema supervisorio.

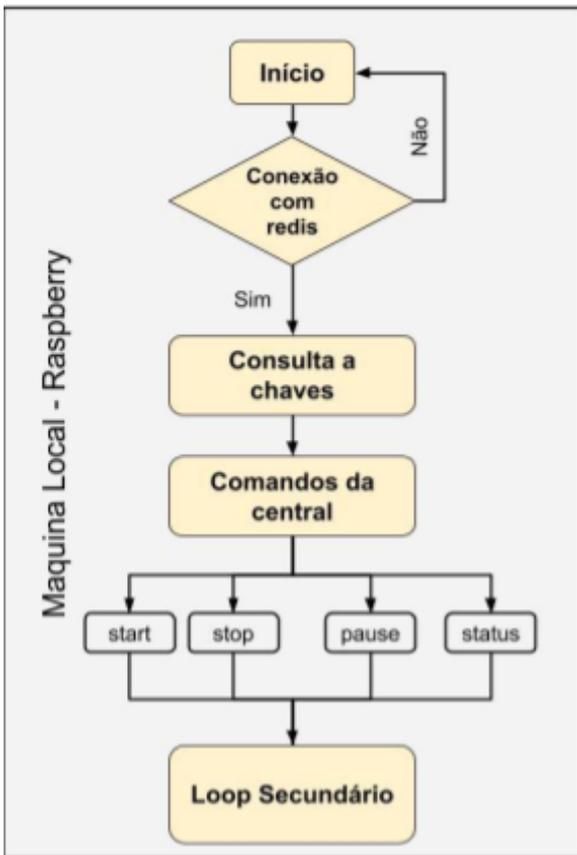
STATUS	inteiro	Condição do equipamento
MEDIDA	float	Armazena as medidas
COMANDO	string	Medida a ser realizada
EQUIPAMENTO	string	Modelo do Equipamento
ESTAÇÃO	string	Local de instalação

**Tabela 1 - Estrutura de dados simplificada do sistema de monitoramento**

**5.1.3 Loop principal - Autenticação e Inicialização do sistema de aquisição**

Após a inicialização e configuração da máquina local, o programa principal é executado para estabelecer a comunicação entre os dispositivos de medida e o banco de dados Redis. O fluxograma apresentado na Figura 7 sintetiza o processo de conexão entre o banco de dados e a máquina local. Antes de iniciar o processo de medidas, é necessário validar as requisições enviadas pelos usuários por meio do banco de dados.

A estrutura de dados definida pelas chaves do Redis é verificada antes de iniciar o processo de medida. A verificação das chaves indica a possibilidade de executar o tipo de medida desejada, a disponibilidade do equipamento e seu respectivo status. Após a verificação, os comandos básicos: play, pause, stop e status são enviados para o segundo estágio do fluxo de processamento - configuração do sistema de medida.

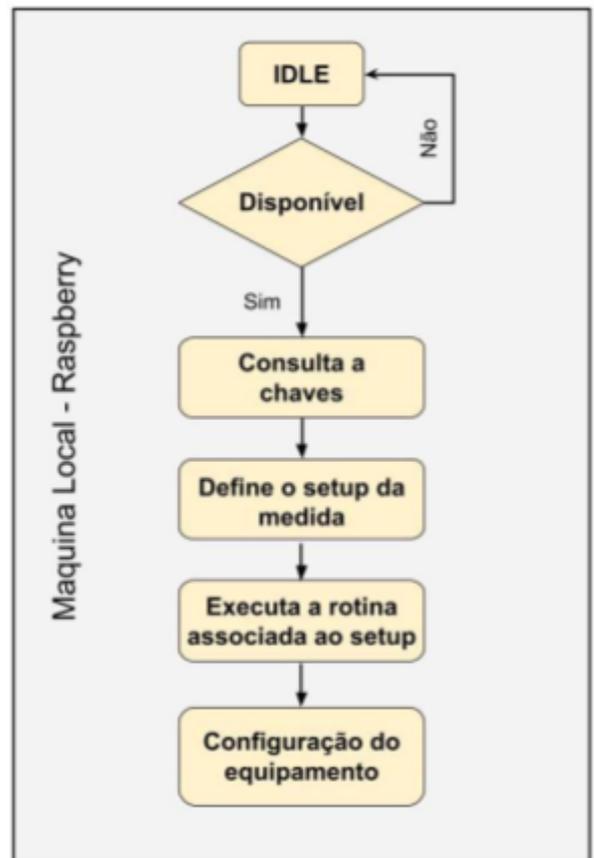


**Figura 7: Fluxograma de Autenticação e Inicialização do sistema de aquisição**

**5.1.4 - Loop secundário - Configuração do sistema de medida**

Após a validação das requisições, ou seja, a confirmação da disponibilidade do equipamento, é necessário configurar o sistema de medida para o arranjo desejado. A validação da requisição é realizada pela sala de controle, considerando os requisitos de disponibilidade dos equipamentos e o tipo de medida desejada. Após a validação, a estrutura de dados é verificada, e os valores armazenados nessa estrutura são associados a rotinas de configuração do equipamento geofísico.

Após a configuração, o sistema estará habilitado e apto a executar medidas. O fluxograma apresentado na Figura 8 resume o processo de configuração do equipamento. Os comandos utilizados nas sub-rotinas de configuração são comandos ASCII enviados pela interface de controle por meio da porta serial.



**Figura 8 : Loop de configuração do sistema de medida**

### 5.1.5 Loop de Medidas

O terceiro bloco do sistema de aquisição consiste em executar a rotina do tipo de medida selecionada pelo usuário e armazenar os respectivos valores no Redis e na memória local do microprocessador, conforme representado no fluxograma da Figura 9. Para cada tipo de medida, há uma sub-rotina específica, cujos comandos em ASCII são enviados pelo microprocessador ao equipamento via porta serial.

Após executar as medidas, o equipamento envia os valores através da porta serial ao microprocessador. A sub-rotina recebe os valores, armazena-os no banco de dados local em um arquivo de texto e, em seguida, após verificar a conexão com o Redis, salva as medidas nas chaves da estrutura de dados definida previamente. Caso a conexão com o Redis não seja estabelecida, após cinco tentativas, um flag indicando que não foi possível salvar a medida é adicionado a um arquivo de texto. Após restabelecer a conexão, uma rotina é executada para realizar o upload dos dados salvos no arquivo de texto nas chaves do Redis. Ao fim do processo as chaves são atualizadas para a condição inicial e um novo processo de aquisição poderá ser submetido.

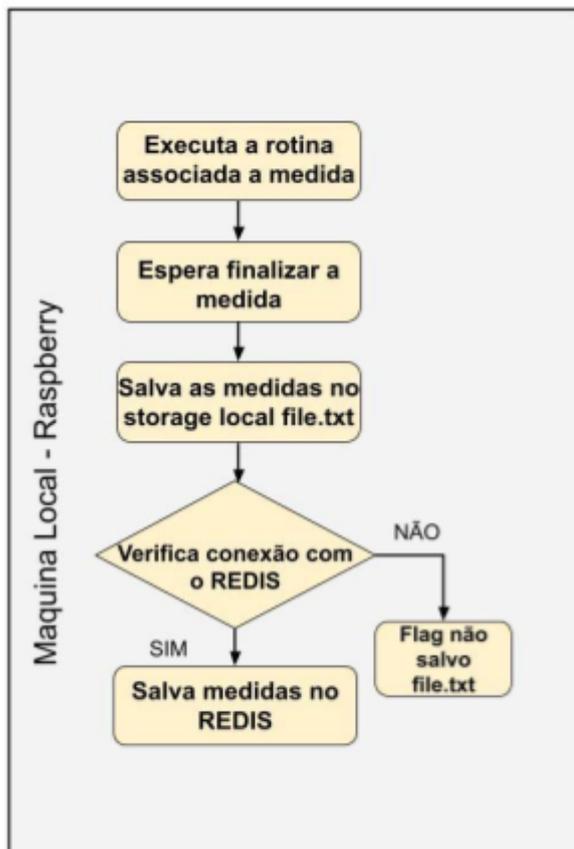


Figura 9: Loop de medidas

## 6. Sistemas de medidas

### 6.1 Medidas de resistividade - SuperStingR8

Trata-se de um resistivímetro multicanal amplamente utilizado em pesquisas geofísicas, reconhecido como um dos equipamentos mais precisos e menos suscetíveis a ruídos disponíveis no mercado. Sua capacidade de registrar dados em 8 canais simultâneos proporciona uma redução significativa no tempo de aquisição.[27]

Esse equipamento é utilizado para estimar a condutividade da subsuperfície. Suas aplicações abrangem diversas áreas, incluindo detecção de cavidades, exploração de água subterrânea e identificação de áreas contaminadas, sendo bastante comum em estudos geofísicos. Interface de controle

Para este sistema de medidas, não existem interfaces desenvolvidas pelo fabricante compatíveis com a Raspberry Pi, sendo assim, a interface de controle foi implementada em Python. Alguns comandos em ASCII são enviados por meio da conexão com a porta serial. As rotinas de configuração e aquisição de medidas foram escritas e são executadas automaticamente no loop de medidas do programa.

Neste caso, para determinar quais são os comandos ASCII associados às funções do equipamento, foi necessário monitorar o tráfego da porta serial quando o software do fabricante, instalado no sistema operacional Windows 7, foi utilizado para configurar o SuperSting. Utilizamos o software Serial Monitor para identificar os comandos necessários para estabelecer a conexão e a comunicação bilateral com o equipamento, além de identificar os valores em ASCII associados aos comandos de configuração do sistema de medida.

Após esta etapa um código em python foi escrito para estabelecer comunicação com o equipamento e controlá-lo através dos comandos enviados pelo teclado do computador. Que posteriormente foram enviados automaticamente pelas rotinas escritas em Python.

## 6.2 Medidas Magnetotelúricas - LEMI 424

O LEMI-424 é um sistema de aquisição de dados eletromagnéticos projetado para o método magnetotelúrico. O sistema utiliza cinco canais independentes, sendo três magnéticos e dois elétricos, para registrar as componentes do campo eletromagnético terrestre. O sistema inclui um sensor magnético do tipo fluxgate, destinado a medir a variação do campo magnético em nT nas componentes Norte-Sul (Bx), Leste-Oeste (By) e Vertical (Bz). Além disso, possui dois pares de eletrodos não polarizáveis responsáveis pelo registro da variação do campo elétrico em mV/m, nas direções Norte-Sul e Leste-Oeste [3][7].

O LEMI-424 faz parte de uma categoria de equipamentos magnetotelúricos classificados como “longo período”, cuja baixa taxa de amostragem é na ordem de 1 Hz, o que implica em um sinal de baixíssima frequência. Essa resolução permite estimar a condutividade das estruturas geológicas presentes em grandes profundidades, desde alguns quilômetros até cerca de 400 Km. Normalmente o tempo de aquisição necessário para determinar a condutividade dessas estruturas pode variar de 12 a 45 dias contínuos.

Devido ao longo período de aquisição requerido por esse sistema e à alta sensibilidade nas condições de operação, o monitoramento remoto dos dados adquiridos é crucial. Por exemplo, caso o sensor magnético do tipo fluxgate altere sua posição vertical e/ou horizontal, ou se os eletrodos forem desconectados durante o processo de aquisição, as medidas serão invalidadas, exigindo uma reinstalação dos elementos sensíveis do sistema de medidas.

- **Interface de controle**

Para este modelo de equipamento, o fabricante disponibilizou uma interface gráfica compatível com o hardware Raspberry Pi e o sistema operacional Raspbian. Essa interface possibilita a configuração e a realização das medidas diretamente no equipamento. A fim de executar esses comandos de maneira autônoma, foi desenvolvida uma rotina em Python capaz de inicializar o programa executável fornecido pelo fabricante e acionar o comando para iniciar a aquisição de dados.

Após a inicialização da interface, o equipamento inicia o processo de medidas contínuas. Essas medidas são armazenadas no armazenamento local do equipamento, no diretório da máquina local, e em seguida são transferidas para o banco de dados REDIS, conforme detalhado na seção anterior.

## 7. Resultados

Os resultados apresentados nesta seção destacam a eficácia do modelo de arquitetura do sistema supervisorio proposto no módulo de comunicação detalhado neste artigo. Esses resultados são divididos em duas categorias principais: os dados coletados pelos sistemas de medição, apresentados graficamente, e as conexões automáticas estabelecidas entre os componentes de hardware e software, notavelmente o banco de dados em nuvem Redis.

### 7.1 - Conectividade

O servidor hospedeiro do banco de dados Redis foi criado por meio de uma conta gratuita fornecida no site [www.redislab.com](http://www.redislab.com) [31]. A máquina virtual do RedisLab oferece um espaço de 30MB para contas gratuitas, o suficiente para testar o modelo da arquitetura proposta neste trabalho. Para criar uma conta gratuita, é necessário fornecer um e-mail válido e definir uma senha. Após o registro, é possível criar um *endpoint* do banco de dados. As seguintes informações são então configuradas para estabelecer a conexão com o Redis: endereço IP do servidor, porta de acesso e a senha definida pelo usuário. Esses parâmetros são essenciais para garantir uma conexão adequada ao banco de dados Redis hospedado no RedisLab.

```
redis = redis.Redis (host_redis = IP, port_redis =
porta de acesso, password_redis = senha)
```

#### # Bloco de comunicação com o Redis

```
host_redis="redis-15032.c8.us-east-1-2.ec2.cloud.
re dislabs.com" # STRING
```

```
port_redis = 15032 # INTEIRO password_redis =
"senha" #STRING
```

Após a conexão bem-sucedida, os usuários têm a capacidade de criar e acessar estruturas de dados específicas de seu interesse usando o modelo chave/valor no banco de dados Redis hospedado no RedisLab. No sistema supervisorio proposto, as rotinas escritas em Python automatizam a comunicação com o Redis sempre que são inicializadas. Isso implica que, ao iniciar o sistema supervisorio, as rotinas Python automaticamente estabelecem uma conexão com o banco de dados Redis, permitindo acesso contínuo e dinâmico às informações armazenadas. Essa abordagem automatizada simplifica e agiliza o processo de comunicação e manipulação de dados no contexto do sistema supervisorio.

Nesta fase do projeto, a implementação do sistema ocorreu através da conexão do microprocessador a um modem 3G. Após estabelecer a conectividade com a Internet, o sistema supervisorio obteve acesso às chaves do REDIS para armazenar as medições e verificar os comandos enviados pelos usuários. Esse modelo de conectividade foi aplicado aos sistemas de medição mencionados neste texto, tanto nas aplicações de campo quanto nos testes realizados em laboratório.

Os sistemas de medição permaneceram operacionais por mais de 24 horas ininterruptas. O sistema de medição magnetotelúrico gerou aproximadamente 13 MB de dados adquiridos por dia. De maneira semelhante, o resistivímetro utilizado nos testes em laboratório gerou um pouco mais de 1 MB por medida. Durante os testes, observou-se que a estabilidade do módulo supervisorio está intimamente relacionada à conectividade com o servidor do banco de dados.

A necessidade de verificar as chaves, tanto para armazenar as medições quanto para verificar os comandos, torna-se inviável em caso de falha na conectividade do modem com a internet. Para contornar essa situação, será necessário implementar, em etapas subsequentes, a conectividade com o servidor por meio de outro protocolo de comunicação e, de preferência, armazenar o servidor do Redis em uma máquina dedicada. Neste projeto, optou-se por uma solução simples: armazenar as medições em um arquivo no armazenamento local e, posteriormente, ao restabelecer a conectividade, realizar o upload para o banco de dados, conforme descrito anteriormente.

## 7.2 - Medidas realizadas pelos sistemas Geofísicos

As medidas obtidas evidenciam quantitativamente o funcionamento da arquitetura de comunicação proposta. Os dados adquiridos pelo equipamento geofísico SuperSting R8 foram registrados em tempo real no banco de dados REDIS e no diretório da Raspberry, conforme a rotina programada. Para este sistema de medida, a interface de controle foi desenvolvida em Python, e a comunicação foi estabelecida diretamente com o equipamento por meio de comandos ASCII enviados para a porta serial. Para este modelo de aquisição, as rotinas em python envia comandos ASCII pré-definidos na rotina de acordo com o tipo de arranjo solicitado. Neste experimento foi utilizado o arranjo de teste de canais, para verificar o funcionamento do modelo da arquitetura e avaliar a repetibilidade do sistema de medidas.

Na Figura 10, é possível observar as cinco primeiras medidas, assim como a última, realizadas em intervalos de 10 minutos. Na Figura 11, são apresentadas as medidas efetuadas no intervalo de 1 hora. É possível observar que a dinâmica do sistema de medidas permaneceu estável, fornecendo os valores de medidas esperados. Os valores de resistência encontram-se na faixa recomendada pela fabricante, situando-se entre 490 e 515 mΩ. Esses dados referem-se às medidas realizadas na 'Test Box' para a verificação dos canais dos eletrodos. Essas medidas prévias são sumamente importantes para avaliar a confiabilidade do equipamento antes das medidas de campo serem realizadas.



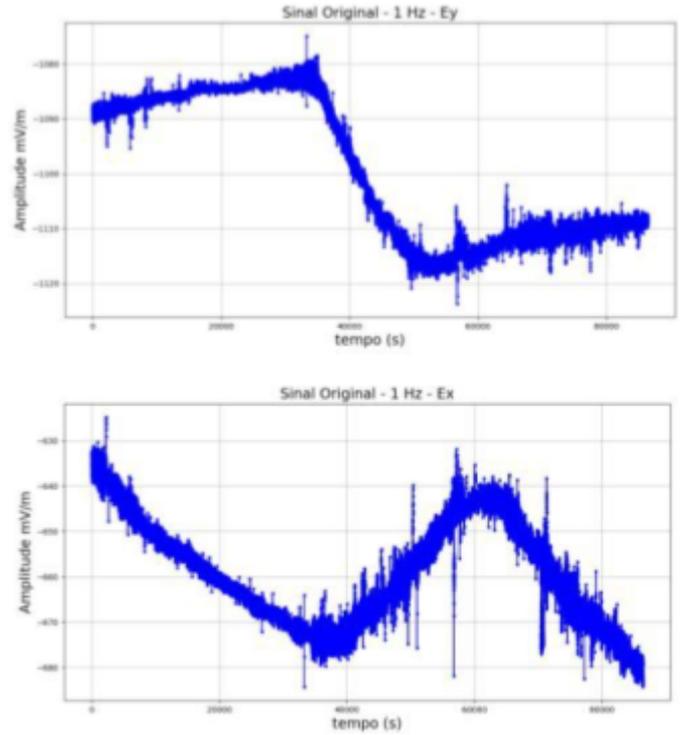
Figura 10 - Medidas realizadas em intervalos de 10 min



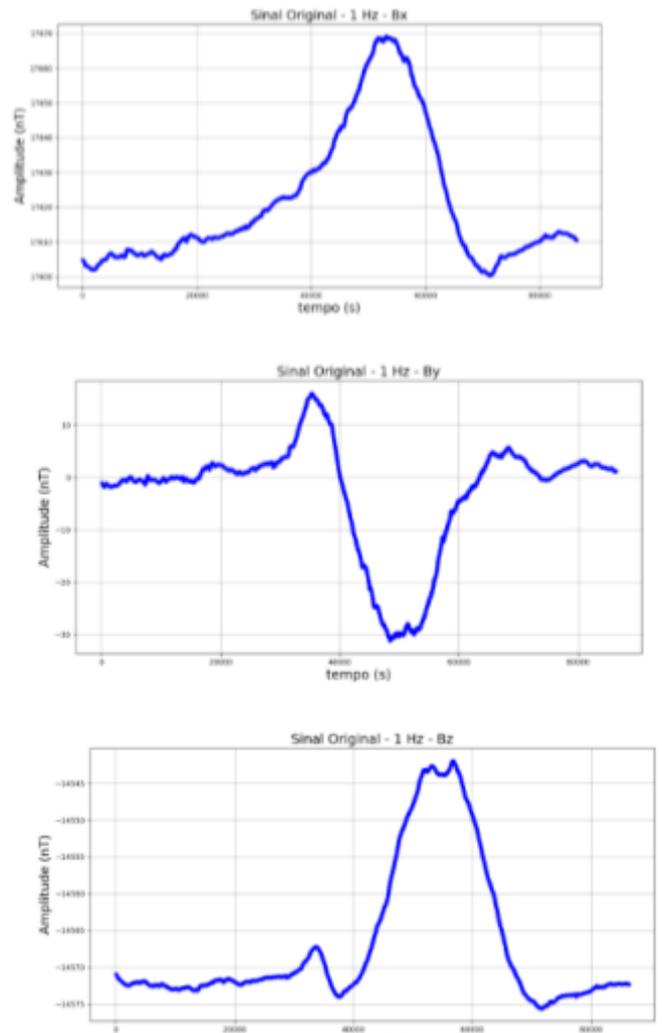
**Figura 11 - Medidas realizadas em intervalos de 1h**

As medidas realizadas com o LEMI-424 e armazenadas no banco de dados podem ser visualizadas na Figura 12. Os valores dos 5 canais registrados são exatamente idênticos aos valores armazenados na memória do equipamento e no armazenamento local da Raspberry. Essas medidas foram realizadas na cidade de Ibertioga-MG durante uma campanha de campo promovida pelo Observatório Nacional, em maio de 2023, como parte do projeto INVESTIGAÇÕES MULTIFÍSICA, financiado pela PETRONAS.

As medidas representam o registro do campo eletromagnético adquirido com a taxa de amostragem de 1Hz. As componentes do campo elétrico são registradas nas direções norte/sul ( $E_x$ ) e leste/oeste ( $E_y$ ), Figura 12a. Da mesma forma, os campos magnéticos são registrados nas direções norte ( $B_x$ ), leste ( $B_y$ ) e vertical ( $B_z$ ), Figura 12b. Esses dados representam as informações fundamentais do campo eletromagnético que estão sendo monitoradas e registradas para análise no contexto do sistema supervisorio [14].



**Figura 12a - Medidas dos campos elétricos  $E_x$  e  $E_y$  realizadas pelo LEMI-424 em 17 de Maio de 2023 na cidade de Ibertioga-MG.**



**Figura 12b - Medidas dos campos elétricos magnéticos  $B_x$ ,  $B_y$  e  $B_z$  realizadas pelo LEMI-424 em 17 de Maio de 2023 na cidade de Ibertioga - MG.**

## 8. Conclusão

O sistema supervisorio desenvolvido neste trabalho, aplicado ao monitoramento e aquisiçao de dados geofisicos, representa uma inovacao significativa na gestao e operacao de equipamentos utilizados em pesquisas de magnetoteluricos e eletroresistividade. Desenvolvido em Python e baseado em microprocessadores, o sistema permite não apenas o monitoramento contínuo dos sistemas de medidas, mas também a reconfiguracao remota, armazenamento eficiente em nuvem e transmissao em tempo real para múltiplos usuarios simultaneamente.

A arquitetura do módulo de comunicacao, com sua estrutura de dados chave/valor implementada no banco de dados NoSQL Redis, destaca-se como uma solucao eficaz para o monitoramento em tempo contínuo. O Redis, adequado para aplicacoes de tempo real, oferece conectividade eficaz entre os equipamentos geofisicos, microprocessador e servidor hospedeiro do banco de dados.

A capacidade de estabelecer conectividade com os equipamentos geofisicos através da porta serial, armazenar dados localmente e, em seguida, aloca-los em chaves no Redis, cria um ambiente dinâmico onde múltiplos usuarios podem interagir, enviando comandos, consultando status e acessando dados adquiridos pelos sistemas de medidas em campo.

O prototipo foi submetido a testes bem-sucedidos realizados em laboratorio e em duas etapas do projeto de pesquisa conduzido pelo Observatório Nacional. Esses testes ocorreram entre maio e julho de 2023 em Ibertioga-MG e entre setembro e novembro do mesmo ano em Lagoa Dourada-MG, como parte do projeto INVESTIGAÇÕES MULTIFÍSICAS, financiado pela PETRONAS. Durante essas fases, o sistema demonstrou eficacia na supervisao remota das estações geofisicas magnetoteluricas de longo período.

Esta conquista não só contribui para a reducao de custos operacionais, otimizando os recursos financeiros em campanhas de aquisiçao, mas também apresenta uma abordagem inovadora para aprimorar a avaliacao da qualidade e continuidade dos dados adquiridos. No âmbito técnico-científico, o sistema oferece uma gestao mais eficiente dos equipamentos, marcando um passo significativo para o futuro da aquisiçao de dados geofisicos.

## Referências

- [1] A. Fernández-Pacheco, S. Martin, and M. Castro, "Implementation of an Arduino remote laboratory with raspberry Pi," in Proc. IEEE Global Eng. Educ. Conf. (EDUCON), Apr. 2019, pp. 1415–1418.
- [2] Boyer, S. A. (2016). Supervisory Control and Data Acquisition (SCADA) Systems (4th ed.). International Society of Automation.
- [3] Cagniard, L. (1953). "Basic theory of the magnetotelluric method." *Geophysics*, 8, 605-635.
- [4] Chambers, J.E., Meldrum, P.I., Gunn, D.A., Wilkinson, P.B., Kuras, O. (2006). Electrical resistivity tomography and its application to geotechnical investigations. Geological Society, London, Engineering Geology Special Publications.
- [5] Ferguson, I.J., Loudon, K.E. (1973). A review of the principles and applications of the magnetotelluric method. *Geophysics*.
- [6] J. Carlson, *Redis in Action*. Greenwich, CT, USA: Manning Publications, 2013.
- [7] Jones, A. G., & Ferguson, I. J. (2018). *The Magnetotelluric Method: Theory and Practice*. Cambridge University Press. ISBN-10: 052181927X, ISBN-13: 978-1107008014.
- [8] J. Sarik and I. Kymissis, "Lab kits using the Arduino prototyping platform," in Proc. IEEE Frontiers Educ. Conf. (FIE), Oct. 2010, pp. T3C-1–T3C-5.
- [9] Kearey, P., Brooks, M., & Hill, I. (2013). *An Introduction to Geophysical Exploration* (3rd ed.). Wiley-Blackwell. ISBN-13: 978-0632049295.
- [10] Kurose, J. F., & Ross, K. W. (2021). *Redes de Computadores e a Internet: Uma Abordagem Top-Down* (8th ed.). Bookman. ISBN-10: 8582605587.
- [11] Lathi, B. P. (2006). *Sinais e Sistemas Lineares* (2nd ed.). Bookman.
- [12] Lutz, M. (2013). *Learning Python: Powerful Object-Oriented Programming*. O'Reilly Media.
- [13] M. Maksimović, V. Vujović, N. Davidović, V. Milošević, and B. Perišić, "Raspberry Pi as Internet of Things hardware: Performances and constraints," in Proc. 1st Int. Conf. Elect., Electron. Comput. Eng., 2014, vol. 3, no. 8, pp. 1–6.
- [14] McKinney, W. (2018). *Python Para Análise de Dados: Tratamento de Dados com Pandas, NumPy e IPython*. Novatec Editora.
- [15] Ogata, K. (2010). *Engenharia de Controle Moderno* (4th ed.). Pearson. ISBN-10: 8587918230.
- [16] Okwueze, E.C., Udensi, E.E., Eluozo, S.N. (2015). A review of magnetotelluric method and its applications to petroleum exploration in Nigeria. *Journal of Petroleum & Gas Exploration Research*.
- [17] P. Jamieson and J. Herdtner, "More missing the Boat—Arduino, Raspberry Pi, and small prototyping boards and engineering education needs them," in Proc. IEEE Frontiers Educ. Conf. (FIE), Oct. 2015, pp. 1–6.

- [18] Oppenheim, A. V., & Willsky, A. S. (2015). *Sinais e Sistemas* (2nd ed.). Pearson.
- [19] P. Orduña, L. Rodriguez-Gil, I. Angulo, U. Hernandez, A. Villar, and J. Garcia-Zubia, "WebLabLib: New approach for creating remote laboratories," in *Cyber-Physical Systems and Digital Twins*, M. E. Auer and B. K. Ram, Eds. Cham, Switzerland: Springer, 2020, pp. 477–488.
- [20] Richardson, S. (2017). *Programação do Raspberry Pi com Python: Aprenda a Programar no Pequeno Computador Mais Popular do Mundo*. Novatec Editora.
- [21] S. Chen, X. Tang, H. Wang, H. Zhao, and M. Guo, "Towards scalable and reliable in-memory storage system: A case study with Redis," in *Proc. IEEE Trustcom/BigDataSE/ISPA*, Aug. 2016, pp. 1660–1667.
- [22] Simpson, F., & Bahr, K. (2005). *Practical Magnetotellurics*. Cambridge University Press. ISBN-10: 1108462553, ISBN-13: 978-1108462556.
- [23] Silberschatz, A., Korth, H. F., & Sudarshan, S. (2011). *Fundamentos de Sistemas de Bancos de Dados* (6th ed.). LTC. ISBN-13: 9788521617475.
- [24] S. Werner, A. Lauber, J. Becker, and E. Sax, "Cloud-based remote virtual prototyping platform for embedded control applications: Cloud-based infrastructure for large-scale embedded hardware-related programming laboratories," in *Proc. 13th Int. Conf. Remote Eng. Virtual Instrum. (REV)*, Feb. 2016, pp. 168–175.
- [25] Villar-Martínez, A., et al. (2019). "Improving the Scalability and Replicability of Embedded Systems Remote Laboratories." *IEEE ACCESS*, 7, 164164-164185. ISSN: 2169-3536.
- [26] Wonham, W. M., & Cai, K. (2019). *Supervisory Control of Discrete-Event Systems*. Springer Cham.
- [27] Advanced Geosciences, Inc. (2005). *The SuperSting with Swift automatic resistivity and IP system Instruction Manual*. Austin, Texas.
- [28] Python.org. (Acesso em 29 de janeiro de 2024). Recuperado de <https://www.python.org/>.
- [29] Raspberry Pi Documentation. (Acesso em 29 de setembro de 2023). Recuperado de <https://www.raspberrypi.com/documentation>.
- [30] RealVNC. (Acesso em 04 de fevereiro de 2024). Recuperado de <https://www.realvnc.com/en/>.
- [31] Redis Labs. (Acesso em 29 de janeiro de 2024). Recuperado de <https://redis.com/>.
- [32] Redis Explained. (Acesso em 21 de setembro de 2022). Recuperado de <https://architecturenotes.co/redis/>.

Notas Técnicas é uma publicação de trabalhos técnicos relevantes, das diferentes áreas da física e afins, e áreas interdisciplinares tais como: Química, Computação, Matemática Aplicada, Biblioteconomia, Eletrônica e Mecânica entre outras.

Cópias desta publicação podem ser obtidas diretamente na página web <http://revistas.cbpf.br/index.php/nt> ou por correspondência ao:

Centro Brasileiro de Pesquisas Físicas  
Área de Publicações  
Rua Dr. Xavier Sigaud, 150 – 4<sup>o</sup> andar  
22290-180 – Rio de Janeiro, RJ  
Brasil  
E-mail: [alinecd@cbpf.br](mailto:alinecd@cbpf.br)/[valeria@cbpf.br](mailto:valeria@cbpf.br)  
<http://portal.cbpf.br/publicacoes-do-cbpf>

Notas Técnicas is a publication of relevant technical papers, from different areas of physics and related fields, and interdisciplinary areas such as Chemistry, Computer Science, Applied Mathematics, Library Science, Electronics and Mechanical Engineering among others.

Copies of these reports can be downloaded directly from the website <http://notastecnicas.cbpf.br> or requested by regular mail to:

Centro Brasileiro de Pesquisas Físicas  
Área de Publicações  
Rua Dr. Xavier Sigaud, 150 – 4<sup>o</sup> andar  
22290-180 – Rio de Janeiro, RJ  
Brazil  
E-mail: [alinecd@cbpf.br](mailto:alinecd@cbpf.br)/[valeria@cbpf.br](mailto:valeria@cbpf.br)  
<http://portal.cbpf.br/publicacoes-do-cbpf>