

**UNIVERSIDADE DO VALE DO RIO DOS SINOS - UNISINOS  
UNIDADE ACADÊMICA DE GRADUAÇÃO  
CURSO DE ENGENHARIA ELÉTRICA**

**CÁSSIO PRUDÊNCIO CARDOSO**

**DESENVOLVIMENTO DO SISTEMA DE MONITORAMENTO DE  
ARREFECIMENTO DOS POSTOS DE CONTROLE LOCAIS A PARTIR DA  
ELABORAÇÃO DE SUPERVISÓRIO COMPLEMENTAR NA TRENSURB**

**São Leopoldo  
2021**

CÁSSIO PRUDÊNCIO CARDOSO

**DESENVOLVIMENTO DO SISTEMA DE MONITORAMENTO DE  
ARREFECIMENTO DOS POSTOS DE CONTROLE LOCAIS A PARTIR DA  
ELABORAÇÃO DE SUPERVISÓRIO COMPLEMENTAR NA TRENSURB**

Trabalho de Conclusão de Curso apresentado como requisito parcial para obtenção do título de Bacharel em Engenharia Elétrica - Ênfase em Controle e Automação, pelo Curso de Engenharia Elétrica da Universidade do Vale do Rio dos Sinos (UNISINOS).

Orientadora: Prof.<sup>a</sup> Dra. Ana Paula Mallmann

São Leopoldo

2021

## **AGRADECIMENTOS**

Agradeço primeiramente a Deus, por sempre me dar forças para enfrentar meus medos e buscar meus sonhos, além de me proporcionar uma família e amigos que foram essenciais nesta caminhada até aqui.

Aos meus pais, Derli e Kátia, que nunca mediram esforços físicos e financeiros para me proporcionar uma educação de qualidade e sempre foram minha inspiração de amor e companheirismo e ao meu irmão e colega de profissão Diego, por toda a parceria não apenas na vida, mas também na resolução de diversos problemas de engenharia.

A minha esposa e melhor amiga, Gabriela, por estar sempre comigo, me apoiar na busca dos meus sonhos, me motivar em todas as vezes que pensei em desistir e principalmente por quebrar o paradigma de uma vida cronometrada, me ensinando que é possível crescer juntos enquanto construímos nossa família.

Ao meu filho Victor, que nasceu no meio de uma pandemia e me ensinou o quão pouco precisamos para ser felizes.

A minha orientadora Ana, que é minha referência de profissional da educação e com excelência me guiou no desenvolvimento deste trabalho, colocando-se a disposição independente do dia e horário em que precisei.

A TRENURB por ceder espaço e confiar na proposta deste projeto e aos meus colegas de trabalho, com foco ao Miguel e Riboldi, por apoiarem esta solução e por toda assessoria no desenvolvimento e instalação do protótipo.

Por fim, mas não menos importante, a todos os colegas e professores que tanto me ensinaram ao longo destes 10 anos de curso, o meu muito obrigado!

“Se você pensa que pode ou se pensa que não pode, de qualquer forma você está certo”  
(Henry Ford).

## RESUMO

Para garantir a operação dos trens na via de forma segura, há um sistema de sinalizações e intertravamentos com diferentes níveis de segurança, onde caso um nível apresente defeito, o próximo deve garantir a integridade da operação. Foi verificado com a TREN SURB um histórico de falhas no segundo nível de segurança, causadas por sobretemperatura nos equipamentos eletrônicos, em decorrência de defeito no condicionador de ar responsável pela refrigeração do ambiente. O presente trabalho viabilizou o aumento da confiabilidade do sistema através da instalação de sensores para monitorar a temperatura ambiente e o funcionamento dos equipamentos responsáveis pela refrigeração, a coleta e armazenamento das leituras destes sensores no banco de dados timescaleDB e a visualização em tempo real destes dados através de um supervísório elaborado no aplicativo Grafana. Esta implantação aumentará a vida útil dos equipamentos e reduzirá o tempo de contato com a manutenção, quando for necessário intervir. Foi instalado um protótipo utilizando uma placa Arduino Mega 2560 e monitorado o funcionamento completo da proposta, onde os dados dos sensores foram devidamente armazenados e amostrados. Não houve perda de dados com os protocolos de comunicação ethernet e MQTT. O protótipo ocupou 13 % da memória do Arduino, tornando possível o controle dos condicionadores de ar de forma automática pela placa e manual pelo envio de comandos via MQTT. O comando dos condicionadores de ar instalados da fabricante Elgin ainda não possui compatibilidade com a biblioteca utilizada no Arduino, nos demais fabricantes o comando funcionou com normalidade.

**Palavras-chave:** sistema de monitoramento de temperatura; automação; supervísório; redes industriais; sensoriamento.

## LISTA DE FIGURAS

Figura 1 – Servidor Dell PowerEdge R210 II.....	18
Figura 2 – Fluxograma da metodologia.....	23
Figura 3 – Resumo de recursos da Arduino MEGA 2560 .....	24
Figura 4 – Sensor de umidade relativa e temperatura DHT22 .....	26
Figura 5 – Sensor de temperatura MF58 .....	26
Figura 6 – Circuito dos sensores de temperatura.....	28
Figura 7 – Leitura dos sensores de temperatura.....	28
Figura 8 – Sensor de corrente SCT-013-000 .....	29
Figura 9 – Circuito do sensor de corrente .....	30
Figura 10 – Leitura do sensor de corrente .....	31
Figura 11 - Arduino Ethernet Shield .....	32
Figura 12 – Circuito receptor e emissor infravermelho.....	36
Figura 13 – Aplicativos instalados no <i>docker</i> .....	38

## LISTA DE FOTOGRAFIAS

Fotografia 1 – Sistemas ATC e ATS do PCL do Pátio Trensurb .....	15
Fotografia 2 – Painel da remota de sinalizações.....	16
Fotografia 3 – Painel de operação local do PCL .....	17
Fotografia 4 – Termômetro tipo capela do PCL Pátio .....	18
Fotografia 5 – Sistema de refrigeração do PCL Pátio .....	20
Fotografia 6 - Leitura da corrente .....	31
Fotografia 7 – CSP Anchieta e PCL Pátio.....	39
Fotografia 8 – Instalação do novo condicionador de ar no PCL do Pátio.....	40
Fotografia 9 – Instalação dos sensores de temperatura e corrente .....	41
Fotografia 10 – Instalação do emissor infravermelho.....	42
Fotografia 11 – Conexão do cabo <i>ethernet</i> no <i>switch</i> de rede da TRENSURB .....	42
Fotografia 12 – Instalação do protótipo no PCL do Pátio .....	43
Fotografia 13 – Câmera térmica na saída do condicionador de ar.....	45

## LISTA DE GRÁFICOS

Gráfico 1 – Histórico de temperaturas na cidade de Porto Alegre em 2019.....	19
Gráfico 2 – Histórico de temperaturas na cidade de Porto Alegre em 2020.....	19
Gráfico 3 – Leitura dos sensores DHT22 e MF58 amostrados no Grafana.....	36
Gráfico 4 – Leitura gráfica dos sensores no Grafana .....	45

## LISTA DE QUADROS

Quadro 1 – Banco de dados supervisorio_PCL .....	34
Quadro 2 – Banco de dados cargas_PCL.....	35

## LISTA DE TABELAS

Tabela 1 – Falhas em aparelhos condicionadores de ar .....	21
Tabela 2 – Trabalhos correlatos .....	21
Tabela 3 – Especificações sensor DHT22 .....	25
Tabela 4 – Comparativo de sensores de temperatura .....	27
Tabela 5 – Comparativo de sensores de corrente.....	30

## LISTA DE SIGLAS

ASCII	<i>American Standard Code for Information Interchange</i>
ATC	<i>Automatic Train Control</i>
BTU	<i>British Thermal Unit</i>
CCO	Centro de Controle de Operações
CLP	Controladores Lógicos Programáveis
CSP	Cabine de Seccionamento e Paralelismo
CTC	Controle de Tráfego Centralizado
FURB	Universidade Regional de Blumenau
GESIS	Gerência de Sistemas
HTTP	<i>Hypertext Transfer Protocol</i>
IDE	<i>Integrated Development Environment</i>
IoT	<i>Internet of Things</i>
IP	<i>Internet Protocol</i>
MAC	<i>Media Access Control</i>
MQTT	<i>Message Queue Telemetry Transport</i>
NTC	<i>Negative Temperature Coefficient</i>
OTP	<i>One Time Programmable</i>
PCL	Posto de Controle Local
PWM	<i>Pulse Width Modulation</i>
SCT	<i>Split-core Current Transformer</i>
SCTE	Sistema de Controle e Tráfego e Energia
SEPRO	Setor de Projetos
SPI	<i>Serial Peripheral Interface</i>
SQL	<i>Structured Query Language</i>
TCP	<i>Transmission Control Protocol</i>
TMA	Temperatura Ambiente Máxima
TRENSURB	Empresa de Trens Urbanos de Porto Alegre
UART	<i>Universal Asynchronous Receiver/Transmitter</i>
UDP	<i>User Datagram Protocol</i>
MV	Máquina Virtual

## SUMÁRIO

<b>1 INTRODUÇÃO .....</b>	<b>12</b>
<b>2 FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>14</b>
<b>2.1 Conceitos básicos.....</b>	<b>14</b>
<b>2.2 Trabalhos correlatos .....</b>	<b>21</b>
<b>3 METODOLOGIA .....</b>	<b>23</b>
<b>3.1 Escolha do microcontrolador.....</b>	<b>24</b>
<b>3.2 Sensores de temperatura .....</b>	<b>25</b>
<b>3.3 Sensor de corrente.....</b>	<b>29</b>
<b>3.4 Protocolos de comunicação <i>ethernet</i>.....</b>	<b>32</b>
<b>3.5 Protocolo de comunicação MQTT .....</b>	<b>33</b>
<b>3.6 Banco de dados.....</b>	<b>33</b>
<b>3.7 Supervisório .....</b>	<b>35</b>
<b>3.8 Comando dos aparelhos condicionadores de ar .....</b>	<b>36</b>
<b>3.9 Configuração do servidor.....</b>	<b>37</b>
<b>3.10 Análise do histórico de falhas nos PCLs .....</b>	<b>38</b>
<b>3.11 Determinação de um PCL para instalação do protótipo .....</b>	<b>39</b>
<b>3.12 Instalação de um novo condicionador de ar.....</b>	<b>39</b>
<b>3.13 Instalação do protótipo.....</b>	<b>40</b>
<b>4 VALIDAÇÃO DO PROTÓTIPO E CONSIDERAÇÕES FINAIS .....</b>	<b>44</b>
<b>4.1 Validação do protótipo.....</b>	<b>44</b>
<b>4.2 Sugestões de trabalhos futuros.....</b>	<b>46</b>
<b>REFERÊNCIAS.....</b>	<b>47</b>

## 1 INTRODUÇÃO

Em abril de 1980, através do Decreto nº 84.640 (*apud* TRENSURB, [2020?]), foi criada a Empresa de Trens Urbanos de Porto Alegre S.A. (TRENSURB) para operar uma linha de trens urbanos na região metropolitana de Porto Alegre. Este projeto foi realizado para reduzir o fluxo de veículos na BR-116 e oferecer um transporte com baixo custo entre os municípios de Porto Alegre e Novo Hamburgo para um público estimado, na época, de 300 mil passageiros por dia (TRENSURB, [2020?]).

Para controlar a circulação de trens ao longo das vias metroviárias, o Sistema de Controle de Tráfego e Energia (SCTE) da Trensurb conta com o Controle de Tráfego Centralizado (CTC) que utiliza uma lógica de segurança baseada no princípio da falha segura. Esta permite a movimentação dos trens em segurança mesmo que ocorra uma falha em algum equipamento. Dentre as lógicas mencionadas, há as funções *Automatic Train Control* (ATC), do Inglês Controle Automático de Trens e *Automatic Train Stop* (ATS), do Inglês Parada Automática de Trens, que controlam a velocidade máxima que o trem pode estabelecer e impede a aproximação demasiada de um trem em relação a outro (TRENSURB, [2020?]).

O Posto de Controle Local (PCL) constitui-se no segundo nível de operação de intertravamento, sendo utilizado em uma possível situação de degradação do sistema CTC ou para implementar uma eventual estratégia de operação diferenciada por trechos. Existem 14 regiões de intertravamento vital, cada uma composta por um conjunto de equipamentos, interligados com as demais regiões. O deslocamento seguro de um trem ao longo de cada uma dessas regiões de intertravamento é garantido por relés vitais, máquinas de chaves e diversos equipamentos eletrônicos e eletromecânicos (TRENSURB, 2019).

Um PCL é composto por diversos equipamentos como transformadores, relés, retificadores, entre outros, que foram projetados para operar em regime contínuo e resfriados apenas por ventilação. Contudo, também há um painel de controle ligado a um servidor que precisa ser mantido sob refrigeração constante para o seu funcionamento ideal. Para refrigerar estes equipamentos atualmente utiliza-se apenas um aparelho condicionador de ar com funcionamento em tempo integral. No caso de falha deste condicionador, o servidor irá sobreaquecer e desligar, interrompendo a comunicação com o SCTE. Como esta é uma instalação não habitada, o Centro de

Controle de Operações (CCO) – subdivisão do SCTE – somente identifica a falha quando o servidor desliga, sendo necessário acionar a manutenção imediatamente.

Com foco no controle de temperatura e identificação de variações indesejadas, este trabalho tem o objetivo de garantir o funcionamento contínuo do servidor e prolongar a vida útil dos equipamentos, através de um método de monitoramento do sistema de arrefecimento do ambiente interno do PCL com redundância. Foi realizado o estudo do acréscimo de um aparelho condicionador de ar para dividir o regime de trabalho, operar em conjunto quando afetados por altas temperaturas externas e ainda garantir a refrigeração do servidor quando uma unidade falhar ou estiver em manutenção.

Também foi elaborado um sistema supervisorio via *software* Grafana para monitorar a temperatura interna do PCL e o funcionamento dos aparelhos condicionadores de ar. Isso permitirá que o CCO visualize o correto funcionamento dos equipamentos e acione a manutenção caso um destes apresente falha, assim podendo atuar de forma preventiva com a possibilidade de eliminar todas as falhas recorrentes por sobretemperatura.

Este trabalho está estruturado em quatro capítulos. O primeiro apresenta a introdução, o segundo apresenta a fundamentação teórica e o estado da arte. No terceiro capítulo é apresentado a metodologia utilizada no desenvolvimento em conjunto com a validação das escolhas tomadas, através da análise dos resultados obtidos. No quarto e último capítulo é apresentado as considerações finais e sugestões de trabalhos futuros.

## 2 FUNDAMENTAÇÃO TEÓRICA

Para o desenvolvimento de um sistema de arrefecimento do PCL é primordial conhecer a importância dos sistemas de segurança dele oriundos, o impacto e o histórico de falhas nestes sistemas, bem como a relevância de tornar o local assistido via supervisor complementar.

### 2.1 Conceitos básicos

O sistema de sinalização foi desenvolvido para garantir a operação dos trens na via com segurança, prevendo possíveis falhas humanas ou em equipamentos, dentre as funções básicas deste sistema estão:

- a) selecionar e autorizar as rotas necessárias para a operação dos trens;
- b) impedir a colisão de trens;
- c) informar ao piloto o limite de velocidade e impedir que esta seja ultrapassada;
- d) identificar a posição dos trens na via (TRENSURB, 2019).

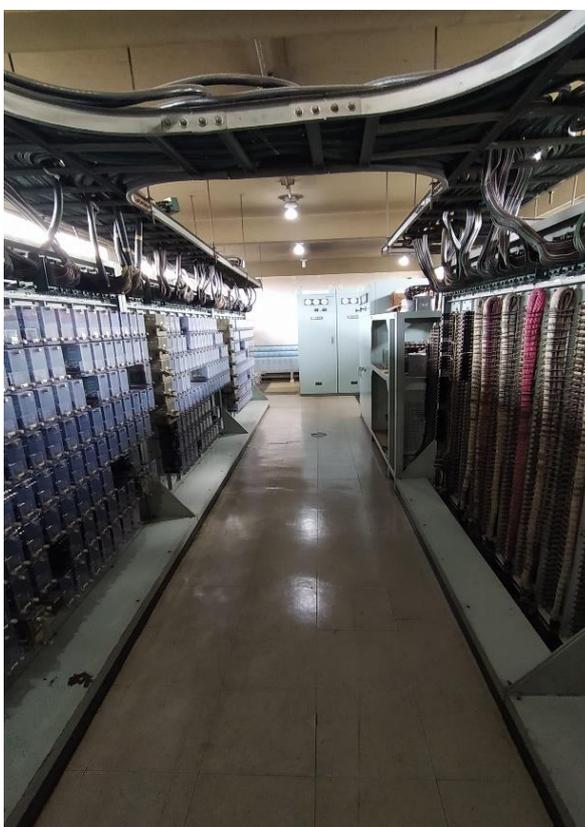
Devido ao número de trens circulando em uma via simultaneamente, é necessário manter um afastamento seguro entre estes. O sistema ATS permite que um trem trafegue na via com segurança utilizando um sinal de frequência de 60 Hz aplicado nos trilhos. Este sinal é recebido pela unidade em movimento caso o sentido de tráfego e o ponto de injeção do sinal sejam compatíveis. Na ausência desta condição ou do próprio sinal resultante de uma falha, atuará uma lógica que provocará uma frenagem de emergência que fará com que o trem pare (TRENSURB, 2019).

Superposta à frequência comercial de 60 Hz há sinais de áudio na faixa de 870 Hz a 930 Hz, gerados pelo conjunto ATC e têm como função permitir o controle de velocidade do trem em cada circuito de via. As velocidades possíveis no sistema são de 30, 50, 70 e 90 km/h, sendo que cada velocidade é configurada através da composição de duas frequências de áudio. Estas combinações são recebidas por antenas do trem e processadas pelos equipamentos de bordo. Na passagem de um trecho para outro onde o código de velocidade seja inferior, caso o operador não reduza a velocidade, o sistema de frenagem atuará de forma automática. Já quando

um trecho apresentar código de velocidade superior, a aceleração do trem será exclusiva por atuação do piloto (TRENSURB, 2019).

Os intertravamentos são formados por um conjunto de equipamentos eletrônicos e eletromecânicos localizados dentro do PCL, projetados para operar em regime contínuo sendo refrigerados apenas por ventilação. Na Fotografia 1 é possível observar alguns relés destes conjuntos; há 12 instalações neste formato, sendo elas os PCLs de Mercado, Rodoviária, Farrapos, Aeroporto, Pátio, CCO, Anchieta, Fátima, Mathias Velho, São Luís, Esteio e Sapucaia, localizadas ao longo da via em proximidade das estações de mesmo nome.

Fotografia 1 – Sistemas ATC e ATS do PCL do Pátio Trensurb



Fonte: Registrada pelo autor.

Interligados ao sistema ATC e ATS há um painel da remota de sinalizações, composto por 2 Controladores Lógicos Programáveis (CLP) PACSystems RX3i da EMERSON que operam com redundância recebendo e enviando comandos para os relés supracitados. Para operar os CLPs há um servidor Dell *PowerEdge* R210 II que permite realizar comandos de forma local no PCL e um sistema dedicado no CCO, isto é utilizado diariamente enquanto a rede interna da Trensurb estiver operacional e

de forma local no próprio PCL quando houver avaria nesta. A Fotografia 2 apresenta o interior do painel da remota de sinalizações.

Fotografia 2 – Painel da remota de sinalizações

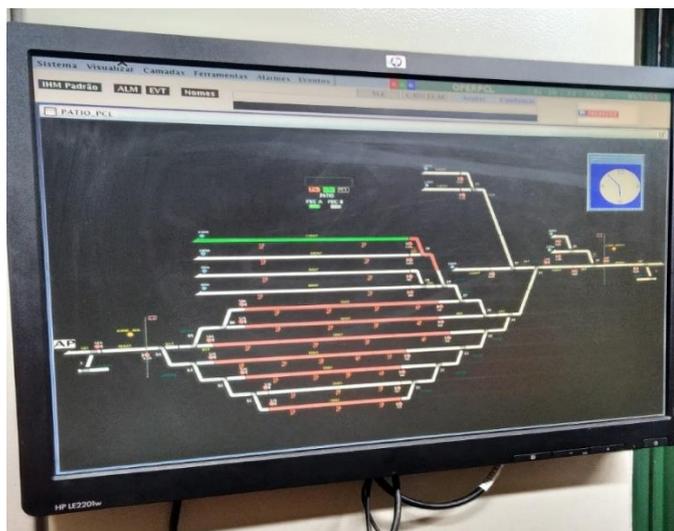


Fonte: Registrada pelo autor.

A principal finalidade do CTC é de monitorar e controlar as remotas de tráfego, o termo utilizado para esta atuação a distância é telemando, onde recebem indicações ou estados dos intertravamentos e enviam comandos para estes atuarem nos elementos de via. A interface no monitor permite a visualização dos elementos de campo que são controlados pelos intertravamentos, como circuitos de via, rotas estabelecidas, sinais e agulhas além de elementos lógicos que não possuem uma equivalência física ao longo da via, como bloqueios de sinal, situação do comando sobre o intertravamento, rotas automáticas, informações sobre falta de energia de tração e nomes das estações e trens (TRENSURB, 2013).

Para visualizar e operar o servidor em modo local, há um monitor na parte frontal do painel, com as mesmas informações supracitadas. Este será utilizado em caso de degradação do acesso remoto e pode ser observado na Fotografia 3.

Fotografia 3 – Painel de operação local do PCL



Fonte: Registrada pelo autor.

Este painel possui equipamentos eletrônicos sensíveis à operação em temperaturas elevadas, bem como a variação brusca da temperatura do ambiente, necessitando assim de refrigeração constante. A atual instalação conta com um aparelho condicionador de ar em operação ininterrupta, o que resulta na degradação deste. Devido ao local não ser habitado, uma falha no controle da temperatura desencadeia o desligamento do painel de controle, tornando inoperante a visualização e operação do trecho no CCO, através de acesso remoto, e no PCL, através de comando local. O comando local do PCL é utilizado em caso de extrema degradação do sistema de controle de tráfego da Trensurb, estando no segundo nível hierárquico e último onde há a possibilidade de comandar os equipamentos na via, enfatiza a criticidade de ser mantido em operação. Na falha do comando local, o sistema é fixado pelos relés eletromecânicos, agora sem visualização remota, e aumentando a responsabilidade dos pilotos na via.

Dentre os equipamentos instalados, o servidor Dell *PowerEdge* R210 II, demonstrado na Figura 1, é o mais frágil a altas temperaturas e por ser instalado em um rack fechado, pode ocorrer de a temperatura interna do rack ser superior à do ambiente da sala. Por isso, deve ser instalado em um local compatível com a Temperatura Ambiente Máxima (TMA) projetada pelo fabricante de 35 °C. Para a correta operação do servidor, a temperatura da sala deve estar entre 10 °C a 35 °C com variação máxima de 10 °C por hora e umidade relativa do ar entre 20 % e 80 % com variação máxima de 10 % por hora (DELL TECHNOLOGIES, 2020).

Figura 1 – Servidor Dell PowerEdge R210 II



Fonte: Site DELL TECHNOLOGIES (2020).

Para monitorar a temperatura interna do PCL, há instalado apenas um termômetro tipo capela, conforme Fotografia 4. Ele fornece três valores de temperaturas: atual, máxima e mínima. Com enchimento de mercúrio (Hg), possui um botão central com função de zerar o marcador de temperaturas máxima e mínima. A escala deste termômetro é de  $-40\text{ }^{\circ}\text{C}$  a  $50\text{ }^{\circ}\text{C}$  e precisão de  $\pm 1\text{ }^{\circ}\text{C}$ , conforme fabricante (INCOTERM, 2020). Nota-se que nesta foto do dia 23/10/2020 a temperatura máxima registrada foi de  $35\text{ }^{\circ}\text{C}$ , mesmo com o condicionador de ar em funcionamento. Contudo, não é possível saber a qual data este valor se refere.

Observa-se que é necessário aprimorar o atual sistema de medição de temperatura, bem como a implementação de um monitoramento e controle deste. Para isso, foram instalados sensores de temperatura em pontos estratégicos do ambiente e dentro do painel onde estão localizados os equipamentos mais sensíveis a alterações de temperatura. Estes sensores são monitorados em tempo real via implementação de supervisor e são utilizados para alterar os parâmetros dos condicionadores de ar que hoje estão configurados em uma temperatura fixa.

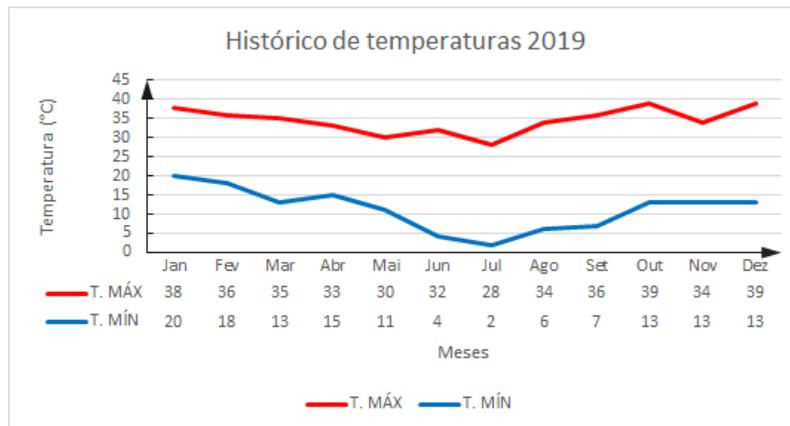
Fotografia 4 – Termômetro tipo capela do PCL Pátio



Fonte: Registrada pelo autor.

Para obter o histórico meteorológico na cidade de Porto Alegre nos anos de 2019 e 2020, foi consultado o site do Tempo. O período com maiores registros de temperatura é entre os meses de outubro a março do ano seguinte, com valores próximos a 40 °C. O Gráfico 1 apresenta os valores de temperatura obtidos em 2019.

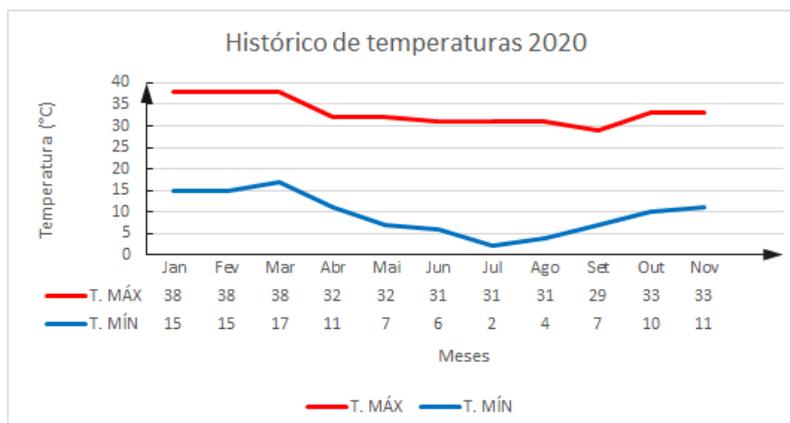
Gráfico 1 – Histórico de temperaturas na cidade de Porto Alegre em 2019



Fonte: Elaborado pelo autor, com base no site Tempo (HISTÓRICO..., 2019).

De mesma forma o Gráfico 2 apresenta os valores obtidos em 2020. O padrão apresentado é semelhante ao ocorrido em 2019. Percebe-se que a temperatura externa no período mencionado é superior à máxima indicada para operação do servidor do PCL, onde é essencial garantir o funcionamento adequado do aparelho condicionador de ar para resfriar o ambiente. Também há períodos em que a temperatura atinge valores próximos a 0 °C, e devido a não haver um controle do condicionador de ar, este permanecerá resfriando o ambiente ao invés de aquecê-lo acima de 10 °C, conforme orienta o fabricante.

Gráfico 2 – Histórico de temperaturas na cidade de Porto Alegre em 2020



Fonte: Elaborado pelo autor, com base no site Tempo (HISTÓRICO..., 2020).

Embora não haja registros da temperatura interna do PCL, a equipe de manutenção da Trensurb relatou que durante intervenções providas da falha do aparelho condicionador de ar no período de altas temperaturas, apresentado acima, depararam-se com o ambiente a aproximadamente 70 °C.

Um PCL, assim como um *datacenter*, é um local de missão crítica e a grande produção de calor sensível – calor responsável por alterar a temperatura do ar - necessita de um sistema inteligente de climatização com redundância, para garantir a refrigeração do ambiente na falha de um dos aparelhos condicionadores de ar (AGST, 2020).

Na Fotografia 5 é apresentado o antigo sistema de refrigeração instalado nos PCLs, onde havia um condicionador de ar tipo janela que foi substituído por um do tipo *split*, o aparelho antigo está desativado e deverá ser removido em breve.

Fotografia 5 – Sistema de refrigeração do PCL Pátio



Fonte: Registrada pelo autor.

Através do sistema ERP PLACOMP utilizado na TRENSURB, foi possível obter um relatório desde a implantação do programa em 2013, contendo os registros de defeitos em aparelhos condicionadores de ar nos PCLs. Na Tabela 1 pode-se observar que no ano de 2020 o número de defeitos aumentou significativamente,

sendo que este número pode ser ainda maior de acordo como a falha foi registrada no sistema.

Tabela 1 – Falhas em aparelhos condicionadores de ar

Quantitativo de Falhas	Ano										Total PCL
PCL	2013	2014	2015	2016	2017	2018	2019	2020			
AEROPORTO		2		1						3	
ANCHIETA						1				1	
ESTEIO		2								2	
FARRAPOS				1				1		2	
FÁTIMA		2								2	
FENAC					1		1	1		3	
MATHIASVELHO		1				1				2	
MERCADO								1		1	
NOVOHAMBURG		1	1					1		3	
SANTOAFONSO	1	4			2	2		1		10	
SÃO LEOPOLDO		4	4		1		2	9		20	
SÃO LUÍS		3				1				4	
UNISINOS	2	3			5	2	1			13	
PÁTIO								2		2	
<b>Total PCL</b>	<b>3</b>	<b>22</b>	<b>5</b>	<b>2</b>	<b>9</b>	<b>7</b>	<b>5</b>	<b>15</b>	<b>68</b>		

Fonte: Elaborado pelo autor.

No próximo subcapítulo é apresentado o estado da arte, iniciando a fundamentação teórica para o desenvolvimento de um sistema de arrefecimento dos PCLs monitorado por supervisor na Trensurb.

## 2.2 Trabalhos correlatos

Para o desenvolvimento deste trabalho, foram utilizados como base artigos e documentos correlatos conforme apresentado na Tabela 2.

Tabela 2 – Trabalhos correlatos

Título	Ano	Autor	Relação
CONTROLE DE RESFRIAMENTO DE SERVIDORES UTILIZANDO UM MICROCONTROLADOR	2013	TAVARES, Cristiano	Controle da refrigeração do servidor
Monitoramento de variáveis ambientais para um data center	2018	COSTA, Guilherme Rocha	Uso de um Arduino e do sensor DHT22 para medição da temperatura de data center
DC ALERTA: PROTÓTIPO DE UM SISTEMA DE MONITORAMENTO DA TEMPERATURA E UMIDADE PARA DATA CENTERS	2018	ANTÔNIO, Milson	Uso de um Arduino e do sensor DHT22 para medição da temperatura de data center com implantação de um servidor no <i>ThingSpeak</i>
Controle das variáveis ambientais de um data center utilizando softwares e ferramentas livres	2020	SOUZA, Felipe Vitorino ARAÚJO, Régia Talina Silva ALENCAR, José Jhonnathas Aires da Silva	Monitoramento de dados climáticos no aplicativo Zabbix e comando remoto pelo aplicativo Blynk

Fonte: Elaborado pelo autor.

No projeto de Tavares (2013) buscou-se controlar a velocidade das ventoinhas que fazem parte dos *data centers* através dos dados coletados por sensores de temperatura que são enviados a um servidor *web*. O foco deste trabalho está nos *data centers* verdes, que são aqueles com iniciativas para reduzir o consumo de energia e assim, minimizar o impacto ambiental.

Costa (2018) desenvolveu um projeto para medição da temperatura e umidade utilizando o sensor DHT22 e coletando os dados utilizando um Arduino ligado a um Esp8266 que se comunica com o roteador via rede sem fio. Por se tratar de um trabalho visando baixo custo, não foram desenvolvidos alarmes e controles remotos, apenas visualização da condição climática dos *data centers*.

Antônio (2018) deu continuidade a um trabalho de conclusão do curso de Ciência da Computação da Universidade Regional de Blumenau (FURB), aprimorando o protótipo utilizando *Internet of Things* (IoT), do inglês, Internet das Coisas para realizar o monitoramento ambiental de *data centers* utilizando um Arduino Mega 2560 como microcontrolador e sensores de temperatura DHT11, sugerindo que em trabalhos futuros este seja substituído pelo DHT22 devido a precisão do sensor. O monitoramento desta implementação foi feito através de um servidor no *ThingSpeak* que também pode ser acessado através de um aplicativo para a plataforma Android.

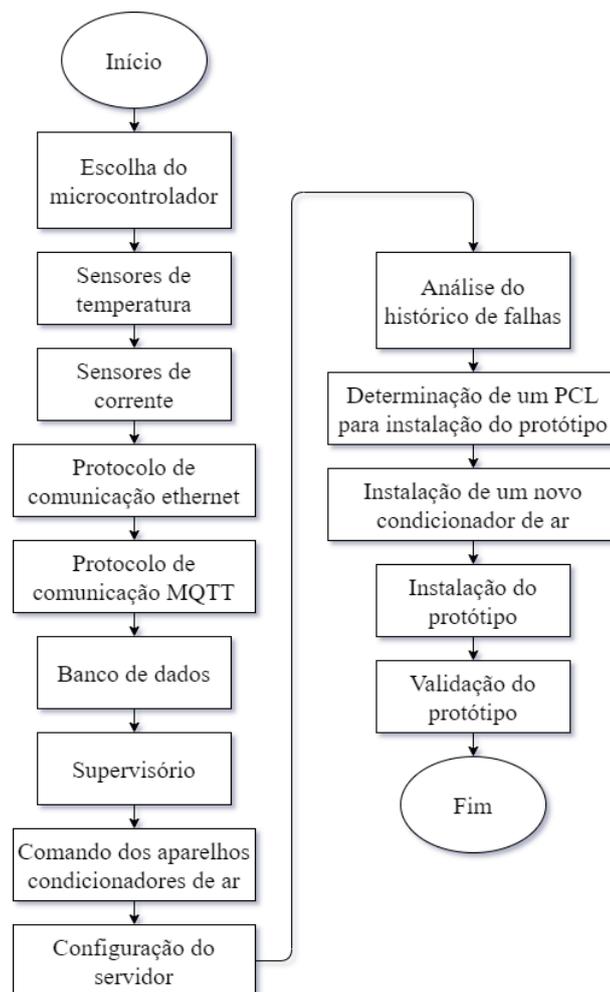
O trabalho do grupo formado por Souza *et al.* (2020) consiste no controle e monitoramento remoto da temperatura e umidade de um *data center* de uma instituição do Ceará, foi utilizado o *software* Zabbix para armazenamento e monitoramento dos dados coletados de sensores e o aplicativo Blynk para controlar o sistema por acesso remoto via smartphone. O projeto resultou em 18,75 % de economia energética e manteve o ambiente dentro dos valores adequados entre 24 °C a 27 °C.

Finalizada a fundamentação teórica, no próximo capítulo é apresentada a metodologia proposta para o desenvolvimento deste trabalho.

### 3 METODOLOGIA

A presente metodologia consiste na identificação de um PCL no qual seja possível instalar e coletar dados de sensores de temperatura estrategicamente posicionados para monitorar a temperatura ambiente e de operação do servidor. Para tanto, foi instalado um sensor para verificar o funcionamento do aparelho condicionador de ar e a possibilidade de instalar um aparelho reserva. Uma vez instalados os sensores, foi elaborado um sistema supervisorio em tempo real que permite monitorar a temperatura e operação dos equipamentos com integração a rede da TRENSURB, possibilitando o acesso de computadores interligados a esta. A Figura 2 apresenta o fluxograma da metodologia proposta.

Figura 2 – Fluxograma da metodologia



Fonte: Elaborado pelo autor.

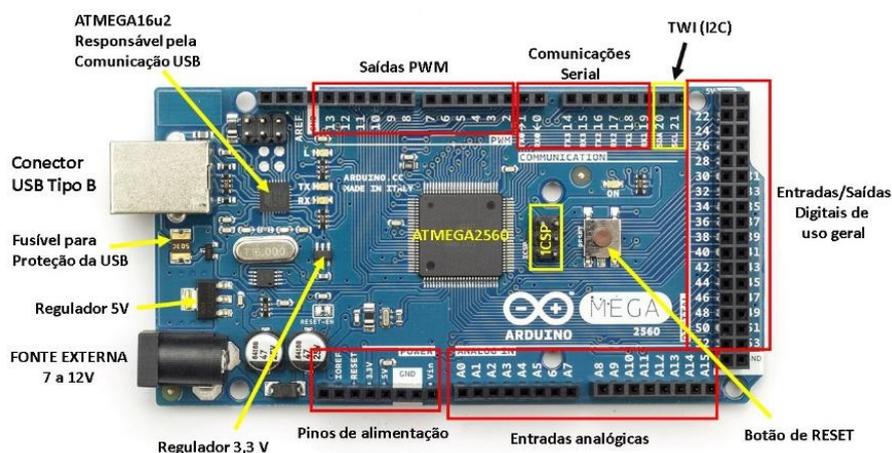
Para validação dos sensores foi utilizado um equipamento de medição auxiliar com precisão igual ou superior ao referido sensor, a fim de confirmar se a leitura está dentro dos limites construtivos registrados nos documentos dos fabricantes. Para os protocolos de comunicação, foi testado a comunicação entre o servidor e o cliente, com o envio e recebimento de informações. Todos os testes foram acompanhados por pelo menos 24 horas.

Com o supervisor em operação, foi avaliado a eficiência da execução destas melhorias implementadas e a necessidade de melhorias para que atenda a proposta de um correto arrefecimento dos PCLs.

### 3.1 Escolha do microcontrolador

A placa Arduino possui grande disponibilidade de bibliotecas compatíveis com sensores e módulos destes sensores otimizados para o uso nela, possui um próprio *Integrated Development Environment* (IDE), do inglês Ambiente de Desenvolvimento Integrado, de fácil uso utilizando linguagem de programação baseada em C e C++. Para o tratamento dos dados dos sensores e comunicação com a rede ethernet foi definido o uso de um Arduino Mega 2560 construído com o *Microcontroller Unit* (MCU), do inglês Unidade Microcontrolador, ATmega2560. O Arduino Mega possui maior número de entradas, saídas e principalmente maior memória *flash* em relação ao Arduino UNO, possibilitando não apenas a construção deste protótipo como também uma futura expansão do número de sensores utilizados. A Figura 3 apresenta a identificação dos principais componentes da placa.

Figura 3 – Resumo de recursos da Arduino MEGA 2560



Fonte: Site EMBARCADOS (2021).

O Arduino Mega 2560 possui 54 pinos de entradas e saídas digitais e destas 15 podem ser utilizadas como saídas *Pulse Width Modulation* (PWM), do inglês Modulação por Largura de Pulso, 16 entradas analógicas, 4 *Universal Asynchronous Receiver/Transmitter* (UART), do inglês Receptor/Transmissor Universal Assíncrono, e uma memória *flash* de 256 kB, onde 8 kB são utilizados pelo *bootloader* (ARDUINO, 2021). Como o Arduino é um *hardware* de código aberto, é possível construir uma placa própria com os arquivos disponibilizados no site do Arduino.

### 3.2 Sensores de temperatura

Para a medição da temperatura ambiente da instalação e do interior do painel da remota, foi selecionado o sensor DHT22 – também denominado como AM2302 – porque realiza medição de temperatura e umidade relativa do ar com sinal digital calibrado e garante excelente estabilidade de longo prazo. É construído com seus elementos de detecção conectados a um chip único de 8 b e com os coeficientes de calibração salvos em uma memória *One Time Programmable* (OTP), do inglês Programado Uma Vez. Esta é uma memória de somente leitura e é utilizada quando o sensor realiza uma medição. Com baixo consumo de energia (1 mA durante a medição e 40  $\mu$ A no modo repouso), transmissão por até 20 m de distância e taxa de atualização a cada 2 s, o sensor DHT22 mostra-se apropriado para execução deste projeto (*Osong Electronic Co.*, 2021). As especificações do sensor quanto a escala, precisão e resolução podem ser observadas na Tabela 3.

Tabela 3 – Especificações sensor DHT22

<b>DHT22</b>	<b>Escala</b>	<b>Precisão</b>	<b>Resolução</b>
<i>Umidade Relativa</i>	0 a 100 %	$\pm 2$ %	0,1 %
<i>Temperatura</i>	-40 a 80 °C	$\pm 0,5$ °C	0,1 °C

Fonte: Adaptado de *Osong Electronic Co.* (2021).

Quando o MCU solicita uma medição, o DHT22 muda do modo de baixo consumo de energia para o modo de operação e envia uma resposta de 40 b – sequenciada a partir do bit mais significativo – composta por cinco dados de 8 bits que correspondem respectivamente aos valores: inteiros da umidade relativa; decimais da umidade relativa; inteiros da temperatura; decimais da temperatura e por último como verificação da correta transmissão dos dados, uma soma dos 8 b inteiros da umidade

relativa e dos 8 b inteiros da temperatura (Osong Electronic Co, 2021). A Figura 4 apresenta o sensor DHT22, bem como os terminais deste.

Figura 4 – Sensor de umidade relativa e temperatura DHT22



Fonte: Adaptado de Osong Electronic Co. (2021).

O terminal 1 é referente a alimentação Vcc (+3,3 V a +6 V); o 2 é responsável pelo envio e recebimento dos dados; o 3 não é utilizado na construção do sensor e o 4 é o referencial GND. Conforme recomendação do fabricante, este deve ser instalado afastado das principais fontes de calor e sem exposição prolongada à luz, a fim de aumentar sua vida útil.

Já para a medição de temperatura da saída do condicionador de ar, foi escolhido o sensor de temperatura *Negative Temperature Coefficient* (NTC), do inglês Coeficiente Negativo de Temperatura, 10k MF58, que diferente do DHT22, este possui um revestimento produzido de aço inoxidável e o cabo revestido de material isolante, podendo ser instalado diretamente na fonte de calor e com características construtivas resistentes inclusive a ambientes úmidos e submersos na água. Com tensão de operação de 3,3 Vcc a 5 Vcc, o sensor NTC reduz sua resistência elétrica à medida que a temperatura aumenta. Apresenta precisão de  $\pm 1$  % e escala de  $-20$  °C a  $105$  °C. A Figura 5 apresenta o sensor MF 58, ressaltando que os terminais deste não possuem polaridade:

Figura 5 – Sensor de temperatura MF58



Fonte: (ELETROGATE, 2021).

A escolha destes sensores foi baseada na escala, precisão, tensão de alimentação e ambiente de instalação. Há uma vasta gama de sensores para esta aplicação disponíveis no mercado, dentre eles são apresentados na Tabela 4 os que melhor se destacam para esta aplicação.

Tabela 4 – Comparativo de sensores de temperatura

<i>Foto</i>					
<b>Sensor</b>	<b>DHT11</b>	<b>DHT22</b>	<b>MF58</b>	<b>BME280</b>	<b>BMP180</b>
<i>Leitura</i>	Temperatura Umidade	Temperatura Umidade	Temperatura	Temperatura Umidade Pressão	Temperatura Pressão
<i>Protocolo de comunicação</i>	Um fio	Um fio	Analógico	I2C SPI	I2C
<i>Tensão de alimentação</i>	3 a 5,5 Vcc	3 a 6 Vcc	3,3 a 5 Vcc	3,3 a 5 V	3,3 a 5 V
<i>Escala de temperatura</i>	0 a 50 °C	-40 a 80 °C	-20 a 105 °C	-40 a 85 °C	0 a 65 °C
<i>Precisão</i>	± 2°C	± 0,5 °C	± 1 °C	± 0,5 °C	± 0,5 °C

Fonte: Adaptado pelo autor (RANDOMNERDTUTORIALS, 2021).

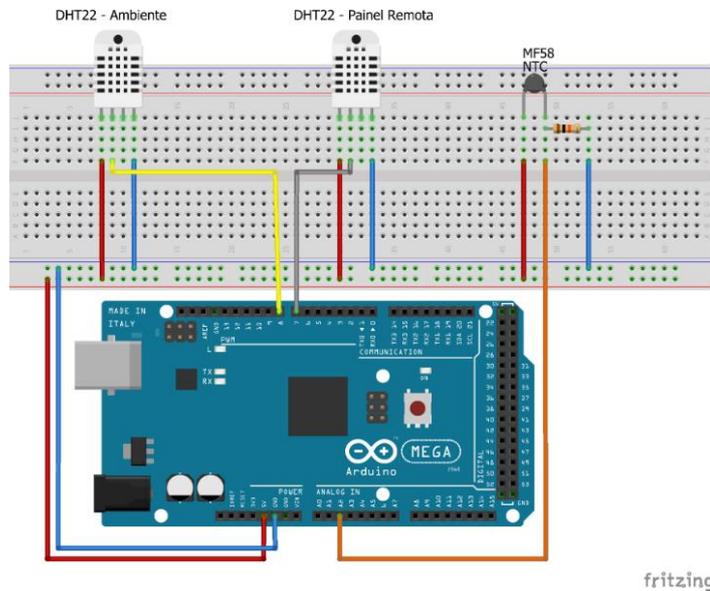
Como critério de eliminação na escolha dos devidos sensores, foi -considerado que:

- DHT11: realiza leitura de temperatura e umidade, mas com uma escala e precisão inferiores ao DHT22;
- BMP180: realiza leitura apenas de temperatura e pressão atmosférica;
- BME280: além de temperatura e umidade, também lê valores de pressão atmosférica, mas este dado não tem serventia neste trabalho e o sensor tem uma característica de aquecer em operação, causando um erro de leitura da temperatura de até  $\pm 2$  °C. É recomendado que no uso deste sensor, seja realizado calibração em laboratório e após isso corrigir a leitura via programação no microcontrolador (RANDOMNERDTUTORIALS, 2021).

Após a definição dos dois sensores de temperatura, foi realizado a montagem do circuito afim de validar esta escolha. Os dois sensores DHT22 e o sensor MF58 foram posicionados fisicamente um ao lado do outro e afastados de qualquer fonte de calor. Para realizar a leitura do MF58 é necessário inserir em série um resistor de

10 k $\Omega$  e utilizar uma entrada analógica do Arduino, já os sensores DHT22 utilizam diretamente uma porta digital. A Figura 6 representa a ligação destes sensores na placa Arduino.

Figura 6 – Circuito dos sensores de temperatura



Fonte: Elaborado pelo autor.

Foi utilizado a biblioteca dht.h para leitura dos sensores DHT22 e a biblioteca Thermistor.h para leitura do MF58. Para visualizar estes dados se fez uso do monitor serial disponível no IDE do Arduino. A Figura 7 apresenta os dados de temperatura e umidade coletados destes sensores:

Figura 7 – Leitura dos sensores de temperatura

```

COM4
02:48:26.900 -> Status          Umidade (%)          Temperatura (°C)
02:48:30.929 -> DHT22_1 Ok           63.5                 23.8
02:48:30.929 -> DHT22_2 Ok           66.0                 23.8
02:48:30.929 -> MF58                  23.8
02:48:35.952 -> DHT22_1 Ok           63.5                 23.8
02:48:35.952 -> DHT22_2 Ok           66.0                 23.8
02:48:35.952 -> MF58                  23.8
02:48:40.929 -> DHT22_1 Ok           63.6                 23.8
02:48:40.929 -> DHT22_2 Ok           66.1                 23.8
02:48:40.929 -> MF58                  23.8
02:48:45.939 -> DHT22_1 Ok           63.7                 23.8
02:48:45.939 -> DHT22_2 Ok           66.2                 23.8
02:48:45.939 -> MF58                  23.7
02:48:50.961 -> DHT22_1 Ok           63.6                 23.8
02:48:50.961 -> DHT22_2 Ok           66.2                 23.8
02:48:50.961 -> MF58                  23.8
02:48:55.969 -> DHT22_1 Ok           63.5                 23.8
02:48:55.969 -> DHT22_2 Ok           66.1                 23.8
02:48:55.969 -> MF58                  23.8
02:49:00.938 -> DHT22_1 Ok           63.5                 23.8
02:49:00.938 -> DHT22_2 Ok           66.0                 23.7
  
```

Fonte: Elaborado pelo autor.

Pode-se observar que a diferença na leitura da umidade entre os sensores DHT22 foi inferior a 4 % e entre a leitura da temperatura entre todos os três sensores foi quase exata em todas as medições, assim, validando a escolha para o desenvolvimento deste protótipo.

### 3.3 Sensor de corrente

Os sensores de corrente disponíveis no mercado são divididos em dois modelos: núcleo sólido e núcleo dividido. A vantagem de utilizar um sensor com núcleo dividido é a possibilidade da instalação sem a necessidade de abrir o circuito para realizar a leitura da corrente alternada. Visto isso, foi determinado o uso do sensor de corrente *Split-core Current Transformer (SCT) 013-000*, do inglês Transformador de Corrente de Núcleo dividido, que possui precisão de 1 % e é capaz de medir a corrente eficaz em escala de 0 a 100 A, gerando um sinal de corrente na saída de 0 a 50 mA, conforme Figura 8:

Figura 8 – Sensor de corrente SCT-013-000



Fonte: (VIDADESILICIO, 2021).

Há outros modelos deste mesmo sensor com capacidade de corrente inferior, mas as diferentes instalações podem possuir um condicionador de ar com potência diferente e pensando que o compressor quando em defeito pode ter a corrente eficaz superior a nominal. Optou-se para este protótipo uma escala que atenda todas as situações. Os sensores de núcleo sólido além da necessidade de abrir o circuito, também apresentaram a escala de corrente inferior ao escolhido. Alguns modelos destes sensores podem ser observados na Tabela 5.

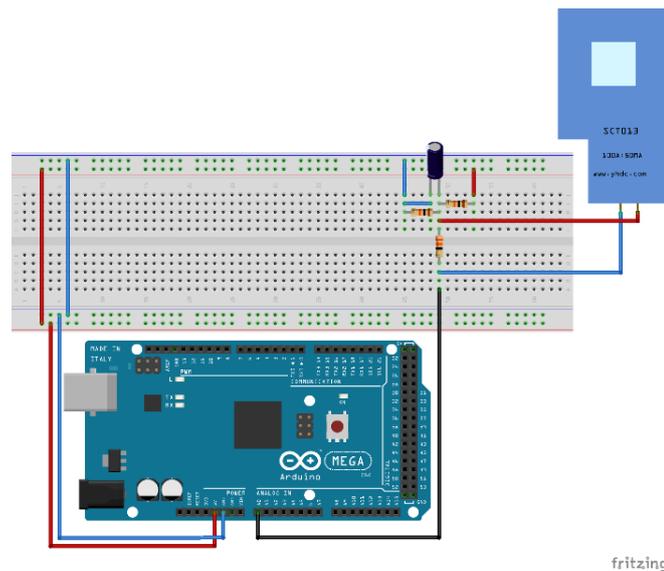
Tabela 5 – Comparativo de sensores de corrente

<i>Foto</i>		
<b>Sensor</b>	<b>ACS758LCB-050B</b>	<b>ACS712</b>
<b>Escala</b>	0 a 50 A	-20 a +20 A

Fonte: Adaptado pelo autor (BAUDA ELETRONICA, 2021).

Como o SCT013 gera um sinal de corrente na saída, para que este dado seja compatível com a leitura dos microcontroladores, é necessário converter esta corrente para um sinal de tensão. O circuito apresentado na Figura 9 demonstra a construção desta conversão e a ligação na placa Arduino.

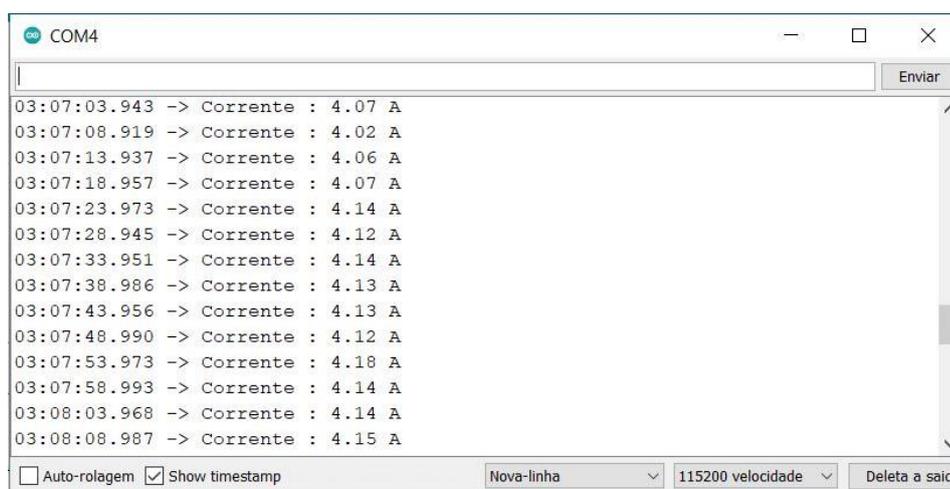
Figura 9 – Circuito do sensor de corrente



Fonte: Elaborado pelo autor.

Foi utilizado a bibliotecas EmonLib.h para leitura do sensor SCT013, e para visualizar estes dados também se fez uso do monitor serial disponível no IDE do Arduino. A Figura 10 apresenta os dados de corrente elétrica coletados deste sensor.

Figura 10 – Leitura do sensor de corrente



Fonte: Elaborado pelo autor.

De forma a verificar se a leitura está correta, foi utilizado um alicate amperímetro que utiliza do mesmo princípio que o sensor para a leitura de corrente. A Fotografia 6 apresenta a instalação do sensor SCT013 e a leitura com o amperímetro.

Fotografia 6 - Leitura da corrente



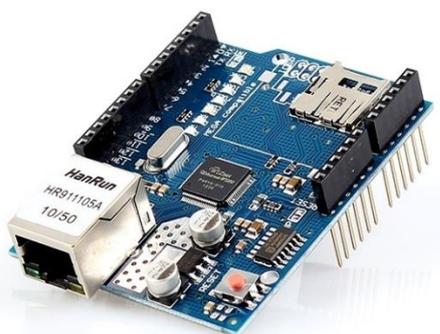
Fonte: Registrado pelo autor.

Para esta conferência foi medido a corrente de um compressor do condicionador de ar de 9000 *British Thermal Unit* (BTU), do inglês Unidade Térmica Britânica). A leitura foi realizada após o compressor estabilizar e manter o consumo demonstrado acima. Como a carga é dinâmica, a proximidade das leituras realizadas confirma a escolha do SCT013 para uso neste protótipo.

### 3.4 Protocolos de comunicação *ethernet*

Como os PCLs não possuem rede *wireless* – do inglês, sem fio – e a TRENSURB dispõe de rede cabeada disponível em quase todas as instalações, optou-se por utilizar o *Arduino Ethernet Shield* (do inglês, escudo Ethernet), demonstrado na Figura 11. Para o funcionamento deste módulo é necessário acoplá-lo ao Arduino e conectar um cabo de rede com terminal RJ45 a um ponto de rede existente.

Figura 11 - Arduino Ethernet Shield



Fonte: (FILIPEFLOP, 2021)

Este módulo faz conexão através da porta *Serial Peripheral Interface* (SPI), do inglês Interface Periférica Serial, com o Arduino e permite que o Arduino seja conectado à internet através do chip ethernet W5100, capaz de fornecer um *Internet Protocol* (IP), do inglês Protocolo de Internet, com conexão *Transmission Control Protocol* (TCP), do inglês Protocolo de Controle de Transmissão, e *User Datagram Protocol* (UDP), do inglês Protocolo de Datagramas de Usuário, com velocidade 10/100 Mb/s (FILIPEFLOP, 2021).

Para configurar a conexão com a rede, foi utilizado a biblioteca *Ethernet.h* onde é necessário atribuir um endereço *Media Access Control* (MAC), do inglês Controle de Acesso de Mídia, e um IP para o Arduino. A biblioteca sugere o uso do endereço MAC {0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED}, este foi utilizado para configuração do protótipo de um PCL, mas o *shield* de cada local deve ter um endereço MAC único. Quanto ao IP, foi verificado com o Setor de Projetos (SEPRO) a disponibilizado o endereço 172.16.201.138.

### 3.5 Protocolo de comunicação MQTT

O *Message Queue Telemetry Transport* (MQTT), do inglês Transporte e Telemetria de Fila de Mensagens, é um protocolo de mensagens assíncrono entre as partes, onde o emissor e o receptor da mensagem não estão diretamente conectados. É utilizado um modelo de publicação e assinatura, onde as mensagens são organizadas por tópicos e distribuídas por um *broker* (do inglês, intermediário) para os clientes, que são quaisquer dispositivos que possam interagir com o *broker*.

O MQTT tem vantagem em relação a outros protocolos, como o *Hypertext Transfer Protocol* (HTTP), do inglês Protocolo de Transferência de Hipertexto, que é síncrono – o cliente espera que o servidor responda – e não indicado para uma rede com grande número de dispositivos conectados pela dificuldade e custo para transmitir uma mensagem a todos eles. O HTTP é um protocolo com muitos cabeçalhos e regras, e é unidirecional, não permitindo que os clientes recebam comandos de forma passiva (IBM, 2021).

Foi utilizado como *broker* o módulo Python Mosquitto no servidor e a biblioteca MQTT.h no Arduino. Para padronizar os tópicos de assinatura foi criado uma sequência de: aplicação, local e sensor no serviço Python para coletar os dados dos sensores e armazenar em um banco de dados, como exemplo: “/supervisorio\_PCLs/PCL\_Patio/DTH22”; e uma sequência de: aplicação e local para enviar comandos ao Arduino de cada instalação; como exemplo: “/arduino/PCL\_Patio”.

### 3.6 Banco de dados

Para monitorar com segurança uma instalação com variáveis de temperatura e corrente de equipamentos em operação, além da coleta dos dados dos sensores, é necessário ter a informação da data e hora destas leituras. Isto permite armazenar um histórico dos eventos que pode ser utilizado para avaliar falhas e demais análises onde se deseja observar um determinado período dos dados coletados.

O TimescaleDB é um banco de dados de séries temporais de código aberto, desenvolvido com base no PostgreSQL, por onde é possível obter consultas de 10 a 100 vezes mais rápidas que ele e outros bancos de dados conhecidos, tais como

InfluxDB e MongoDB. O TimescaleDB foi otimizado para consultas usando séries de tempo e é capaz de gravar milhões de dados por segundo (TIMESCALE, 2021).

Foi utilizado o aplicativo Dbeaver para criar as tabelas e gerenciar os dados do banco de dados. No aplicativo é possível criar e editar as tabelas através de linguagem de programação *Structured Query Language* (SQL), do inglês Linguagem de Consulta Estruturada. Para organizar os dados coletados, foram criados dois bancos de dados, um para armazenar os dados de temperatura e umidade, nomeado de “supervisorio\_PCL” e outro para os dados de corrente elétrica, nomeado de “cargas\_PCL”.

O Arduino não tem uma biblioteca com conexão direta a esse banco de dados, contudo os dados são enviados através do protocolo MQTT, recebidos pelo serviço em Python e com o uso da biblioteca *psycopg2*, que é compatível com PostgreSQL. Assim foi possível gravar os dados dos sensores no banco de dados TimescaleDB. Para testar a conexão, foram posicionados sobre uma mesa os dois sensores DHT22 – nomeados de Painel Remota e Ambiente – e o sensor MF58 – nomeado de Condicionador de Ar – para medir a temperatura ambiente. O Quadro 1 apresenta os dados armazenados pelo banco “supervisorio\_PCL”, visualizados através do aplicativo Dbeaver.

Quadro 1 – Banco de dados supervisorio\_PCL

	time	abc_pcl_local	abc_pcl_sensor	abc_estado	123 temperatura	123 umidade
24517	2021-11-04 01:11:25.813 -0300	Patio	Painel Remota	Ok	25,3	79,5
24518	2021-11-04 01:11:25.960 -0300	Patio	Ambiente	Ok	25,3	81,1
24519	2021-11-04 01:11:26.005 -0300	Patio	Condicionado de Ar	[NULL]	25,27	[NULL]
24520	2021-11-04 01:11:30.814 -0300	Patio	Painel Remota	Ok	25,3	79,7
24521	2021-11-04 01:11:30.958 -0300	Patio	Ambiente	Ok	25,3	81,3
24522	2021-11-04 01:11:31.005 -0300	Patio	Condicionado de Ar	[NULL]	25,09	[NULL]
24523	2021-11-04 01:11:35.818 -0300	Patio	Painel Remota	Ok	25,3	80
24524	2021-11-04 01:11:35.953 -0300	Patio	Ambiente	Ok	25,3	81,4
24525	2021-11-04 01:11:36.000 -0300	Patio	Condicionado de Ar	[NULL]	25,18	[NULL]
24526	2021-11-04 01:11:40.825 -0300	Patio	Painel Remota	Ok	25,3	80,2
24527	2021-11-04 01:11:40.957 -0300	Patio	Ambiente	Ok	25,3	81,6
24528	2021-11-04 01:11:41.000 -0300	Patio	Condicionado de Ar	[NULL]	25,18	[NULL]
24529	2021-11-04 01:11:45.825 -0300	Patio	Painel Remota	Ok	25,3	80,2
24530	2021-11-04 01:11:45.960 -0300	Patio	Ambiente	Ok	25,3	81,7
24531	2021-11-04 01:11:46.016 -0300	Patio	Condicionado de Ar	[NULL]	25,18	[NULL]
24532	2021-11-04 01:11:50.828 -0300	Patio	Painel Remota	Ok	25,3	80,7
24533	2021-11-04 01:11:50.971 -0300	Patio	Ambiente	Ok	25,3	81,8
24534	2021-11-04 01:11:51.018 -0300	Patio	Condicionado de Ar	[NULL]	25,27	[NULL]

Fonte: Elaborado pelo autor.

Da mesma forma, o banco de dados cargas\_PCL é apresentado no Quadro 2.

Quadro 2 – Banco de dados cargas\_PCL

	time	pcL_local	pcL_sensor	corrente
2779	2021-11-09 17:23:56.288 -0300	Patio	Condicionador de Ar	4,84
2780	2021-11-09 17:23:51.284 -0300	Patio	Condicionador de Ar	4,88
2781	2021-11-09 17:23:46.280 -0300	Patio	Condicionador de Ar	4,83
2782	2021-11-09 17:23:41.277 -0300	Patio	Condicionador de Ar	4,9
2783	2021-11-09 17:23:36.273 -0300	Patio	Condicionador de Ar	4,87
2784	2021-11-09 17:23:31.271 -0300	Patio	Condicionador de Ar	4,86
2785	2021-11-09 17:23:26.266 -0300	Patio	Condicionador de Ar	4,85
2786	2021-11-09 17:23:21.262 -0300	Patio	Condicionador de Ar	4,8
2787	2021-11-09 17:23:16.261 -0300	Patio	Condicionador de Ar	4,84
2788	2021-11-09 17:23:11.257 -0300	Patio	Condicionador de Ar	4,84
2789	2021-11-09 17:23:06.264 -0300	Patio	Condicionador de Ar	4,8
2790	2021-11-09 17:23:01.251 -0300	Patio	Condicionador de Ar	4,83
2791	2021-11-09 17:22:56.247 -0300	Patio	Condicionador de Ar	4,83

Fonte: Elaborado pelo autor.

Neste teste de conexão foram enviadas as leituras dos sensores a cada 5 segundos e todas foram devidamente armazenados nos seus respectivos bancos de dados com a data e hora do registro, validando a escolha para este protótipo.

### 3.7 Supervisório

Foi escolhido o aplicativo Grafana para visualizar de forma gráfica as informações armazenadas nos bancos de dados. Com diversos modelos de gráficos otimizados para séries temporais, histogramas, listas de alertas, entre outros, é possível criar um único painel com os gráficos de diferentes fontes de dados e disponibilizar para toda empresa com atualizações em tempo real (GRAFANA, 2021).

Para validar o uso desta ferramenta e reforçar a escolha dos sensores de temperatura, foram posicionados os sensores DHT22 e o MF58 para medir a temperatura ambiente externa de uma residência. Nesta análise foram coletados os dados dos sensores a cada 5 s durante um período de 24 h. O Gráfico 3 apresenta o gráfico gerado destas amostras.

Gráfico 3 – Leitura dos sensores DHT22 e MF58 amostrados no Grafana



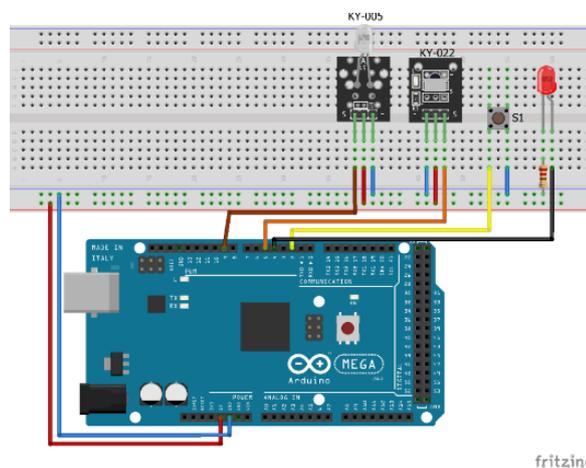
Fonte: Elaborado pelo autor.

Nesta análise não houve perda de dados recebidos dos sensores e é possível observar que os valores de temperatura medidos pelo sensor analógico – MF58 – são muito próximos em comparação ao sensor digital – DHT22 –, reforçando a escolha destes para instalação no PCL.

### 3.8 Comando dos aparelhos condicionadores de ar

Para controlar os aparelhos condicionadores de ar, foi utilizado um módulo emissor infravermelho para Arduino – KY-005 – em conjunto com um módulo receptor infravermelho – KY-022 – e a biblioteca IRremote.h. Ao pressionar o botão S1, configurado pela biblioteca PushButton.h, é ativado o modo gravação onde o receptor fica ativo e aceso o LED para indicar a espera de um sinal para gravação. A Figura 12 representa o circuito para receber e enviar comandos via infravermelho.

Figura 12 – Circuito receptor e emissor infravermelho



Fonte: Elaborado pelo autor.

Foi utilizado um controle remoto do condicionador de ar LG para validar este circuito. Após ativar o modo gravação e pressionar o botão de ligar do controle remoto, foi recebido um código *raw* do comando via serial. Um dado *raw* é um vetor de dados para armazenar sequências de bytes com comprimento fixo, em hexadecimal com referência à tabela *American Standard Code for Information Interchange* (ASCII), do inglês Código Padrão Americano para o Intercâmbio de Informação (UFRGSWIKI, 2021).

O código recebido para o comando de ligar foi armazenado em um vetor de nome *teclaA*, conforme a seguinte sequência:  $teclaA[ ] = \{4450,4250,600,1550,600,500,550,1550,600,1550,550,500,550,550,550,1550,550,500,600,500,550,1550,600,500,550,500,550,1600,550,1550,600,500,550,1550,550,550,550,1550,550,1600,600,1500,600,1550,550,500,600,1550,600,1550,550,1550,600,500,550,500,550,500,600,500,550,1550,550,500,600,500,550,1550,600,1550,600,1550,600,450,550,500,600,500,550,500,550,500,550,500,600,1550,550,1600,550,1550,550,1600,550,1550,600\}$ . Além do comando de ligar, também foram armazenados os comandos de desligar e modo turbo.

O programa no Arduino foi desenvolvido de forma a possibilitar o envio deste comando via protocolo MQTT. É permitido que a placa verifique as leituras dos sensores e envie comandos de forma automática para os condicionadores de ar e que estes comandos sejam enviados de um computador na rede de forma manual para controlar a operação dos equipamentos, ambas operações foram testadas e o condicionador de ar ligou com normalidade.

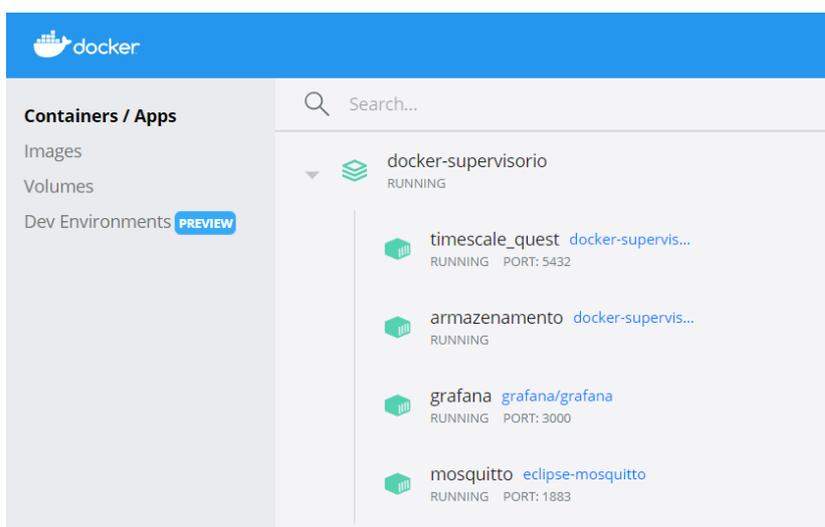
### 3.9 Configuração do servidor

Uma aplicação em *docker*, conhecida como container, tem vantagem em relação ao uso de uma Máquina Virtual (MV), pois a MV faz uso do *hardware* do computador e instala um sistema operacional em cada MV montada. Já na aplicação em container, é utilizada uma camada de aplicativo que utiliza o mesmo sistema operacional instalado, e isola os aplicativos enquanto está ativa (DOCKER, 2021).

Outra vantagem em relação a MV é que no *docker* é possível empacotar toda a aplicação, inclusive as dependências necessárias. Estas dependências podem ser a versão ou configuração de um aplicativo e instalação de *plugins* e *drivers*. Em caso de atualização da aplicação, a estrutura de camadas do *docker* viabiliza que apenas

a parte modificada seja transferida e ao utilizar *tags* nas imagens, caso uma atualização apresente algum erro, basta utilizar uma imagem com a tag de uma versão anterior para que a aplicação volte a funcionar (MundoDevOps, 2018). A Figura 13 apresenta a aplicação desenvolvida no *docker*.

Figura 13 – Aplicativos instalados no *docker*



Fonte: Elaborado pelo autor.

Neste container foram instalados o *broker* mosquitto, o banco de dados timescaleDB, o supervisorio Grafana e um serviço em Python que recebe os dados via MQTT e os envia para o banco de dados. Caso seja necessário substituir o computador onde o servidor está instalado, basta executar este arquivo *docker* que todas as configurações e aplicativos necessários para executar este projeto estarão prontas para uso.

### 3.10 Análise do histórico de falhas nos PCLs

Após analisar os dados da Tabela 1 é verificado que os PCLs com maior número de ocorrências causadas por falha no controle de temperatura são os de São Leopoldo, Unisinos e Santo Afonso. Contudo, estes estão localizados no interior da estação de mesmo nome, possuindo um sistema de refrigeração distribuído por dutos para toda a estação e diferente dos demais, que não são habitados, estes possuem funcionários trabalhando nas salas ao lado. Devido a esta proximidade, os funcionários percebem quando o alarme sonoro de avaria no PCL está ativo e acionam a manutenção. Considerando estas características, os PCLs deste formato não são o

foco deste trabalho, mas o supervisório aqui desenvolvido é compatível com todas as instalações.

O histórico de falhas entre as instalações das estações Mercado até Sapucaia apresenta uma média de aproximadamente duas falhas registradas por local. Isto se dá ao fato do sistema de refrigeração original – um condicionador de ar tipo janela – ter sido substituído por um condicionador de ar tipo *split*, reduzindo as ocorrências de falhas no sistema.

### 3.11 Determinação de um PCL para instalação do protótipo

Para instalação do protótipo deste trabalho, foi definido o PCL do Pátio Anchieta, devido à proximidade com a Cabine de Seccionamento e Paralelismo (CSP) Anchieta, onde há um *switch* para interligação do módulo *ethernet* do Arduino com a rede Trensurb. Na Fotografia 7 é possível visualizar estas duas instalações.

Fotografia 7 – CSP Anchieta e PCL Pátio



Fonte: Registrada e adaptada pelo autor.

Outro fator determinante para a escolha desta instalação é a proximidade com os setores administrativos, facilitando a logística para realizar a implantação e revisão do funcionamento do protótipo.

### 3.12 Instalação de um novo condicionador de ar

Ao longo do desenvolvimento deste projeto, foi apresentado ao SEPRO a importância da instalação de um condicionador de ar reserva nos PCLs. O SEPRO

acatou a proposta e tomou a responsabilidade de adquirir e instalar um novo aparelho ao menos no local do protótipo. A Fotografia 8 apresenta a instalação deste novo equipamento instalado no PCL do Pátio.

Fotografia 8 – Instalação do novo condicionador de ar no PCL do Pátio



Fonte: Registrada pelo autor.

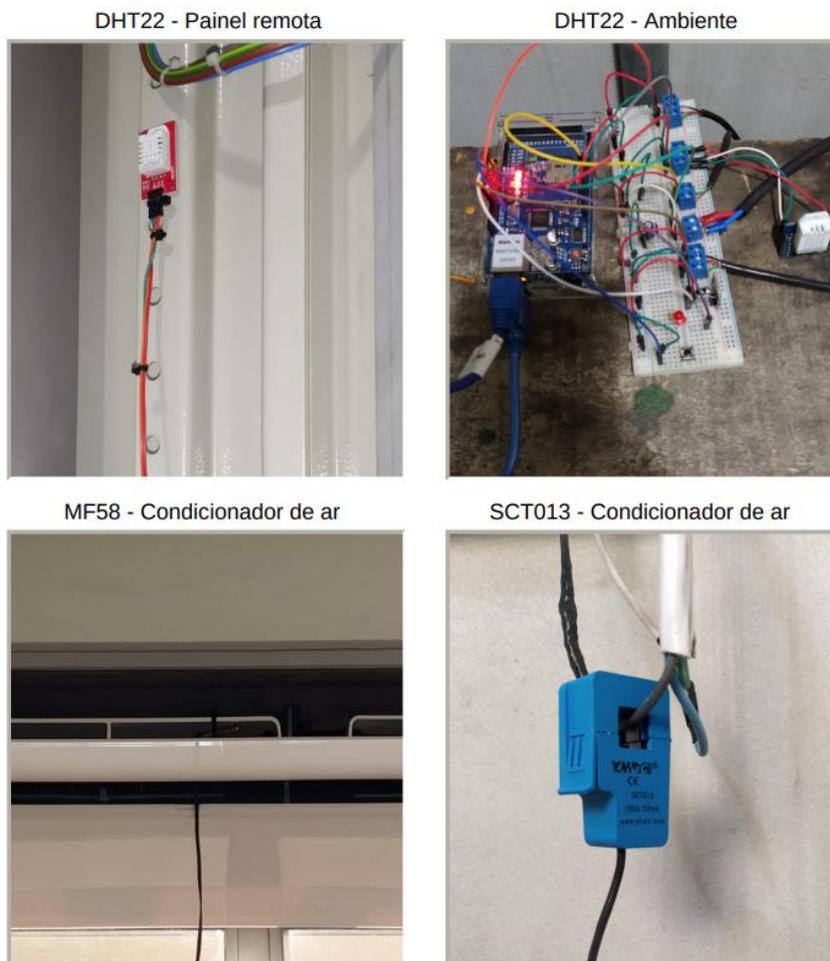
No mês de setembro do ano 2021, o SEPRO informou que havia finalizado a instalação não apenas no PCL do Pátio, mas também em todos os PCLs contemplados neste projeto, de Mercado até Sapucaia.

### 3.13 Instalação do protótipo

Foi iniciada a montagem do protótipo pela instalação dos sensores. Um sensor de temperatura DHT22 foi fixado no interior do painel da remota de sinalizações e o outro foi posicionado sobre uma bancada junto com a placa Arduino e uma *protoboard*, onde foram montados os circuitos auxiliares para o funcionamento dos sensores. O sensor de temperatura MF58 foi fixado na saída de ar do condicionador de ar e o

SCT013 fixado em uma das fases em que está ligado este condicionador. A Fotografia 9 apresenta a instalação destes sensores.

### Fotografia 9 – Instalação dos sensores de temperatura e corrente



Fonte: Registrada pelo autor.

O emissor infravermelho foi posicionado um pouco abaixo do condicionador de ar, a fim de evitar conflitos de distância e eliminar interferências de obstáculos que possam interferir no envio do sinal. Neste protótipo foi instalado apenas um emissor infravermelho, uma vez que os dois condicionadores de ar são de fabricantes diferentes e assim, possuem o sinal de comando diferente. Caso ambos os equipamentos sejam da mesma fabricante, é necessário aproximar ainda mais o sensor do seu devido condicionador de ar e instalar um bloqueio mecânico para evitar que o comando seja recebido pelos dois condicionadores. A Fotografia 10 apresenta a instalação do emissor infravermelho.

Fotografia 10 – Instalação do emissor infravermelho



Fonte: Registrada pelo autor.

Após a instalação dos sensores, foi conectado um cabo de rede do módulo *ethernet* do Arduino até o *switch* da rede TRENURB localizado na CSP Anchieta. Também foi instalado um novo *switch* no PCL, de caráter provisório, para testar a conexão do Arduino com o computador onde está instalado o servidor. Foi realizado um teste ponto a ponto entre os equipamentos conectados a rede, onde a conexão foi comprovada com totalidade. A Fotografia 11 mostra a conexão do cabo na CSP Anchieta.

Fotografia 11 – Conexão do cabo *ethernet* no *switch* de rede da TRENURB

Fonte: Registrada pelo autor.

Com os sensores e o servidor devidamente posicionados e com a conexão de rede testada, o protótipo ficou em operação para monitorar as leituras dos sensores e amostrar estes dados no aplicativo Grafana. A Fotografia 12 apresenta a instalação completa do protótipo no PCL do Pátio.

Fotografia 12 – Instalação do protótipo no PCL do Pátio



Fonte: Registrada pelo autor.

Após este teste, o computador com o servidor foi instalado em um ponto de rede no setor de manutenção da Gerência de Sistemas (GESIS), onde permaneceu recebendo os dados dos sensores durante 3 horas. Assim, comprovou o funcionamento do envio dos dados via protocolo MQTT. A conexão do servidor para o Arduino também foi testada através do envio de uma mensagem de teste para ele, a mensagem foi devidamente processada. Com isso, todos os dispositivos do protótipo confirmaram a comunicação de envio e recebimento de dados.

## 4 VALIDAÇÃO DO PROTÓTIPO E CONSIDERAÇÕES FINAIS

O objetivo do trabalho de tornar o PCL assistido de forma remota, com o controle das variáveis ambientais e de funcionamento dos equipamentos foi atingido de forma satisfatória. Para demonstrar a instalação e funcionamento do protótipo, é disponibilizado um vídeo através do *link*: [Apresentação do protótipo instalado](#).

Os detalhes dos testes e validação do protótipo estão detalhados no subcapítulo a seguir.

### 4.1 Validação do protótipo

O protótipo desenvolvido no Arduino Mega ocupou 13 % da memória total da placa. Foi realizado uma simulação utilizando o Arduino UNO e não foi possível gravar o código pois ultrapassou o limite de memória disponível. O módulo *ethernet* e dos sensores em conjunto com as bibliotecas disponíveis funcionaram com normalidade quando testados em bancada e após a instalação no PCL.

Foi observado *in loco* o funcionamento do protótipo durante 6 horas. Neste período o módulo do DHT22 alimentado por 5 Vcc realizou o envio de todos os dados sem apresentar erros. Já o outro módulo do DHT22 alimentado em 3,3 Vcc apresentou, sem uma taxa constante, alguns erros de *checksum*, que ocorrem quando há erro em uma leitura da temperatura ou umidade. Foram utilizados dois módulos de construção diferentes afim de verificar a compatibilidade dos modelos disponíveis no mercado.

Os sensores analógicos de temperatura MF58 e de corrente SCT013 não apresentaram erros durante a validação do protótipo. Para confirmar estes valores foi utilizado um alicate amperímetro e uma câmera térmica para medir a temperatura na saída do condicionador de ar. A Fotografia 13 apresenta esta leitura, e o valor foi confirmado nos registros do banco de dados.

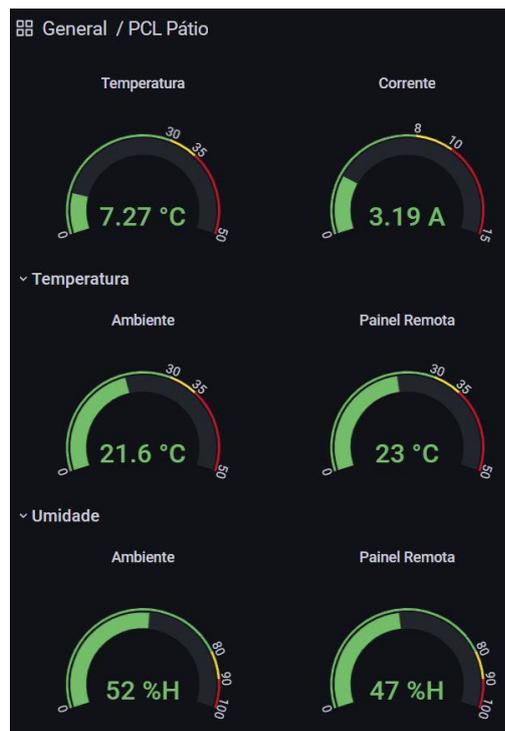
Fotografia 13 – Câmera térmica na saída do condicionador de ar



Fonte: Registrada pelo autor.

Os dados foram enviados pelos sensores para o Arduino, que através do protocolo MQTT, comunicou com o banco de dados onde todos os valores medidos e retornos da condição de operação dos sensores digitais foram devidamente registrados. O Grafana buscou corretamente os dados no timescaleDB e apresentou de forma gráfica o último valor lido de cada sensor, em tempo real, com atualização a cada 5 s, conforme apresentado no Gráfico 4.

Gráfico 4 – Leitura gráfica dos sensores no Grafana



Fonte: Elaborado pelo autor.

O último teste realizado foi o envio de um comando via infravermelho para o condicionador de ar. O comando foi enviado do computador conectado a rede através do método de publicação do MQTT. O Arduino recebeu e processou este sinal mas o condicionador de ar não o recebeu. Foi verificado na biblioteca IRremote.h que o fabricante do equipamento instalado – Elgin – ainda não foi otimizado para uso, pois o sinal *raw* é maior que o usual.

## 4.2 Sugestões de trabalhos futuros

Como sugestões de trabalhos destaca-se:

- a) analisar a viabilidade de substituição do computador onde está instalado o servidor por uma *Raspberry Pi*, ou semelhante que faça uso de Linux;
- b) verificar se há ganho de confiabilidade ao instalar sensores de tensão para monitorar se o condicionador de ar e o PCL estão com alimentação elétrica em normalidade;
- c) avaliar a utilização dos *plugins* de envio de alertas disponíveis no Grafana, bem como outros formatos de gráficos;
- d) analisar se a não compatibilidade de comunicação por infravermelho, pode ser substituída pelo uso de módulos relé.

## REFERÊNCIAS

ANTÔNIO, Milson. DC alerta: Protótipo de um sistema de monitoramento da temperatura e umidade para data centers. 2018. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) - Universidade Regional de Blumenau, Blumenau, 2018.

ARDUINO mega 2560 Rev3. *In*: Store. Disponível em: <https://store-usa.arduino.cc/products/arduino-mega-2560-rev3?selectedStore=us>. Acesso em: 06 out. 2021.

Baú da Eletrônica. Componentes eletrônicos. Disponível em: <https://www.baudaeletronica.com.br/>. Acesso em: 01 nov 2021.

COSTA, Guilherme Rocha da. Monitoramento de variáveis ambientais para um data center. 2018. Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Controle e Automação) - Universidade Estadual Paulista Júlio de Mesquita Filho, Sorocaba, 2018.

DELL TECHNOLOGIES. PowerEdge R210 II. Disponível em: <https://www.dell.com/support/home/pt-br/product-support/product/poweredge-r210-2/docs> Acesso em: 17 out. 2020.

DHT11 vs DHT22 vs LM35 vs DS18B20 vs BME280 vs BMP180. Disponível em: <https://randomnerdtutorials.com/dht11-vs-dht22-vs-lm35-vs-ds18b20-vs-bme280-vs-bmp180/>. Acesso em 01 nov 2021.

Docker. Use containers to Build, Share and Run your applications. Disponível em: <https://www.docker.com/resources/what-container>. Acesso em 01 out 2021.

ELETROGATE, Arduino, Robótica e Automação. Sensor de temperatura NTC 10k encapsulado MF58. Disponível em: [https://www.eletrogate.com/sensor-de-temperatura-ntc-10k-encapsulado-mf58?utm\\_source=Site&utm\\_medium=GoogleMerchant&utm\\_campaign=GoogleMerchant&gclid=CjwKCAjwz5iMBhAEEiwAMEAwGHE15wN3vE3gAleS9FYrw8TypDZNr9GJKG7oDg2qw3T\\_WmV6Zh3HoRoC](https://www.eletrogate.com/sensor-de-temperatura-ntc-10k-encapsulado-mf58?utm_source=Site&utm_medium=GoogleMerchant&utm_campaign=GoogleMerchant&gclid=CjwKCAjwz5iMBhAEEiwAMEAwGHE15wN3vE3gAleS9FYrw8TypDZNr9GJKG7oDg2qw3T_WmV6Zh3HoRoC). Acesso em: 06 nov 2021.

FILIFELOP. Ethernet Shield W5100 para Arduino. Disponível em: <https://www.filipeflop.com/produto/ethernet-shield-w5100-para-arduino/>. Acesso em: 10 nov 2021.

Grafana Labs. Dashboard anything. Observe everything. Disponível em <https://grafana.com/grafana/>. Acesso em 20 out 2021.

HISTÓRICO da previsão do tempo para Porto Alegre - RS. *In*: TEMPO. Porto Alegre, 2020. Disponível em: <https://www.tempo.com/porto-alegre-sactual.htm>. Acesso em: 21 nov. 2020.

IBM Brasil. Conhecendo o MQTT. Disponível em: <https://developer.ibm.com/br/articles/iot-mqtt-why-good-for-iot/>. Acesso em 10 nov 2021.

MundoDevOps. Por Que Utilizar Docker? Disponível em: <https://mundodevops.com/blog/por-que-utilizar-docker/>. Acesso em: 10 dez 2021.

SCT-013 – Sensor de Corrente Alternada com Arduino. Disponível em: <https://portal.vidadesilicio.com.br/sct-013-sensor-de-corrente-alternada/>. Acesso em: 10 nov 2021.

SENSOR de temperatura NTC 10k encapsulado MF58. In: Eletrogate. Disponível em: [https://www.eletrogate.com/sensor-de-temperatura-ntc-10k-encapsulado-mf58?utm\\_source=Site&utm\\_medium=GoogleMerchant&utm\\_campaign=GoogleMerchant&gclid=CjwKCAjwz5iMBhAEEiwAMEAwGHE15wN3vE3gAleS9FYrw8TypDZNr9GJKG7oDg2qw3T\\_WmV6Zh3HoRoCmfcQAvD\\_BwE](https://www.eletrogate.com/sensor-de-temperatura-ntc-10k-encapsulado-mf58?utm_source=Site&utm_medium=GoogleMerchant&utm_campaign=GoogleMerchant&gclid=CjwKCAjwz5iMBhAEEiwAMEAwGHE15wN3vE3gAleS9FYrw8TypDZNr9GJKG7oDg2qw3T_WmV6Zh3HoRoCmfcQAvD_BwE). Acesso em: 06 out. 2021.

SOUZA, ARAÚJO, ALENCAR. Controle das Variáveis ambientais de um data center utilizando softwares e ferramentas livres. 2019. Trabalho de Conclusão de Curso (Bacharelado em Tecnologia da Informação) - Instituto Federal de Educação, Ciência e Tecnologia do Ceará, Ceará, 2019.

TAVARES, Cristiano Araújo. Controle de resfriamento de servidores utilizando um microcontrolador. 2013. Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Computação) - Centro Universitário de Brasília, Brasília, 2013.

Termômetro máxima e mínima analógico enchimento ecológico. In: Incoterm. Disponível em: <https://www.incoterm.com.br/solucoes-em-medicao/termometro-de-maxima-e-minima-analogicos/produto/5201-02-0-00-termometro-maxima-e-minima-analogico-enchimento-ecologico>. Acesso em: 21 nov. 2020.

TimescaleDocs. TimescaleDB Overview. Disponível em: <https://docs.timescale.com/timescaledb/latest/overview/#accelerated-performance>. Acesso em 01 nov 2021.

TRENSURB. Apresentação Trensurb: Sistema de Sinalização de Segurança do Tráfego de Trens. Porto Alegre: setembro/2019.

TRENSURB. Descrição Funcional do Sistema SCTE. Porto Alegre, 2013.

TRENSURB. **História**. Porto Alegre: TRENSURB, [2020?]. Disponível em: [http://www.trensurb.gov.br/paginas/paginas\\_detalhe.php?codigo\\_sitemap=48](http://www.trensurb.gov.br/paginas/paginas_detalhe.php?codigo_sitemap=48). Acesso em: 04 out. 2020. (TRENSURB, [2020?]).

WkiR. Raw. Disponível em: <https://www.ufrgs.br/wiki-r/index.php?title=Raw>. Acesso em: 15 out 2021.