

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE CAMPINAS
CENTRO DE CIÊNCIAS EXATAS, AMBIENTAIS E DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM GESTÃO DE REDES DE
TELECOMUNICAÇÕES

ISAIAS MENDES DE OLIVEIRA

DESENVOLVIMENTO DE PLATAFORMA EXPERIMENTAL DE ELETRICIDADE BÁSICA
COM ACESSO E CONTROLE REMOTOS: UMA ABORDAGEM IMPLEMENTADA
DIRETAMENTE EM MICROCONTROLADOR

CAMPINAS

2022

ISAIAS MENDES DE OLIVEIRA

DESENVOLVIMENTO DE PLATAFORMA EXPERIMENTAL DE ELETRICIDADE BÁSICA
COM ACESSO E CONTROLE REMOTOS: UMA ABORDAGEM IMPLEMENTADA
DIRETAMENTE EM MICROCONTROLADOR

Dissertação apresentada ao Programa de Pós-Graduação *Stricto Sensu* em Gestão de Redes de Telecomunicações do Centro de Ciências Exatas, Ambientais e de Tecnologias, da Pontifícia Universidade Católica de Campinas, como exigência para obtenção do título de Mestre em Gestão de Redes de Telecomunicações.

Orientadora: Prof.^a Dra. Cecília de Freitas
Morais

CAMPINAS

2022

Ficha catalográfica elaborada por Adriane Elane Borges de Carvalho CRB 8/9313
Sistema de Bibliotecas e Informação - SBI - PUC-Campinas

006.22
O48d

Oliveira, Isaias Mendes de

Desenvolvimento de plataforma experimental de eletricidade básica com acesso e controle remotos: uma abordagem implementada diretamente em microcontrolador / Isaias Mendes de Oliveira. - Campinas: PUC-Campinas, 2022.

93 f.: il.

Orientador: Cecília de Freitas Morais.

Dissertação (Mestrado em Gestão de Redes de Telecomunicações) - Programa de Pós-Graduação em Gestão de Redes de Telecomunicações, Centro de Ciências Exatas, Ambientais e de Tecnologia, Pontifícia Universidade Católica de Campinas, Campinas, 2022.

Inclui bibliografia.

1. Internet das coisas. 2. Controle remoto. 3. Tecnologia de ponta e educação. I. Morais, Cecília de Freitas. II. Pontifícia Universidade Católica de Campinas. Centro de Ciências Exatas, Ambientais e de Tecnologia. Programa de Pós-Graduação em Gestão de Redes de Telecomunicações. III. Título.

CDD - 22. ed. 006.22

ISAIAS MENDES DE OLIVEIRA

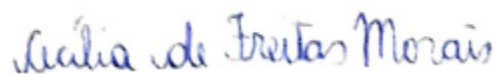
DESENVOLVIMENTO DE PLATAFORMA EXPERIMENTAL DE ELETRICIDADE BÁSICA
COM ACESSO E CONTROLE REMOTOS: UMA ABORDAGEM IMPLEMENTADA
DIRETAMENTE EM MICROCONTROLADOR

Dissertação apresentada como exigência para a obtenção do título de Mestre em Gestão de Redes de Telecomunicações ao Programa de Pós-Graduação *Stricto Sensu* em Gestão de Redes de Telecomunicações do Centro de Ciências Exatas, Ambientais e de Tecnologias.

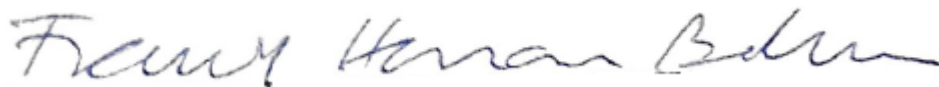
Área de Concentração: Gestão de Redes e Serviços

Orientadora: Prof.^a Dra. Cecília de Freitas Morais

Dissertação defendida e aprovada em 12 de dezembro de 2022 pela Comissão Examinadora constituída dos seguintes professores:



Prof.^a Dra. Cecília de Freitas Morais
Orientadora da Dissertação e Presidente da Comissão Examinadora
Pontifícia Universidade Católica de Campinas



Prof. Dr. Frank Herman Behrens
Pontifícia Universidade Católica de Campinas



Prof. Dr. Pedro Luís Dias Peres
Universidade Estadual de Campinas - UNICAMP

Dedico a minha família pelo apoio e companheirismo.

Dedico ao IFSP Câmpus Hortolândia pela oportunidade de me dedicar integralmente a essa pesquisa.

AGRADECIMENTOS

Prof.^a Dra. Cecília de Freitas Morais

Orientadora e incentivadora dos meus trabalhos de pós-graduação na Faculdade de Engenharia Elétrica da Pontifícia Universidade Católica de Campinas, pelo apoio, atenção e amizade.

A minha esposa Joseni Santos, meus filhos Caio, Elisa e Benício

Por estarem ao meu lado em todos os momentos e pelo incentivo para que esse sonho se tornasse realidade.

RESUMO

A Internet das Coisas (IoT) é considerada a tecnologia do século XXI. Na área de ensino, dentro do contexto da Educação 4.0, as tecnologias relacionadas à IoT merecem destaque por apresentarem um grande potencial inovador ao serem aplicadas na melhoria do processo de ensino-aprendizagem, especialmente contribuindo com o desenvolvimento de laboratórios remotos de baixo custo. A popularização dos computadores e microcontroladores de placa única, bem com uma ampla disponibilidade de sensores, fornecem infinitas possibilidades para a construção desses laboratórios. Com a pandemia da Covid-19, esse avanço tecnológico foi acelerado, pois várias instituições de ensino tiveram suas atividades presenciais suspensas, recomendando a educação à distância como medida de contenção de disseminação do vírus e alternativa para a continuidade de suas atividades. No ensino à distância existe uma dificuldade de formar nos alunos as habilidades que só são adquiridas com a experimentação prática em laboratórios. Apesar de haver vários estudos demonstrando que, em termos de engajamento e ganhos de aprendizagem para os alunos, os laboratórios remotos são equivalentes aos laboratórios tradicionais, seu uso no Brasil ainda é muito limitado. Nesse contexto, este trabalho apresenta o desenvolvimento de uma plataforma experimental para o ensino de conceitos de eletricidade básica para estudantes do ensino médio e/ou técnico, com acesso e controle remotos, que, diferentemente da maioria das soluções encontradas na literatura, é implementada diretamente em um microcontrolador de placa única. Para isso, é proposta uma nova arquitetura simplificada em comparação a outros laboratórios remotos. Finalmente, para tornar mais intuitiva a interação do usuário do laboratório remoto, reduzindo a necessidade de supervisão e suporte adicional, também é desenvolvida uma interface gráfica na forma de um gêmeo digital do experimento prático.

Palavras-chave: Laboratório remoto. Educação 4.0. Gêmeo digital.
Microcontrolador de placa única. Eletricidade básica. Internet das Coisas.

ABSTRACT

The Internet of Things (IoT) is considered the technology of the 21st century. In the learning scenario, within the context of Education 4.0, IoT-related technologies deserve to be highlighted because they have great innovative potential when applied to improve the teaching-learning process, especially contributing to the development of low-cost remote laboratories. The popularization of single-board computers and microcontrollers, as well as the wide availability of sensors, provide endless possibilities for building these laboratories. With the Covid-19 pandemic, this technological advance was accelerated, as several educational institutions had their face-to-face activities suspended, recommending distance education as a measure to contain the spread of the virus and an alternative for the continuity of their activities. Despite several studies demonstrating that, in terms of engagement and learning gain, the remote laboratories are equivalent to the traditional ones, their employment in Brazil is still very limited. In distance learning, there is a difficulty in training students in skills that are only acquired with practical experimentation in laboratories. In this context, this work presents the development of an experimental platform for teaching basic electricity concepts for high school and or technical students, with remote access and control, which, unlike most solutions found in the literature, is implemented directly in a single-board microcontroller. For this, it is proposed the use of an innovative and simpler architecture compared to other remote laboratories. Finally, to make the remote laboratory user interaction more intuitive, reducing the need for supervision and additional support, a graphical interface is also developed in the form of a digital twin of the hands-on experiment.

Keywords: Remote laboratory. Education 4.0. Digital twin. Single-board microcontroller. Basic electricity. Internet of Things.

LISTA DE FIGURAS

Figura 1 – Evolução da Internet.....	18
Figura 2 – Da Educação 1.0 à Educação 4.0.....	27
Figura 3 – Quatro elementos centrais da Educação 4.0.	28
Figura 4 – Arquitetura Cliente – Servidor com <i>software</i> de controle – Experimento.	30
Figura 5 – Arquitetura Cliente – Servidor com <i>software</i> de controle – placa DAQ – Experimento.	31
Figura 6 – Arquitetura Cliente – Servidor <i>proxy</i> – Nós do laboratório – Experimentos.	31
Figura 7 – Arquitetura Cliente – Servidor com SCADA – PLC – Experimento.....	32
Figura 8 – Arquitetura Cliente – SBC – SBM ou FPGA – Experimento.	33
Figura 9 – Classificação das arquiteturas de placa única e suas principais plataformas.	34
Figura 10 – Arquitetura do Painel Elétrico CC e Painel Elétrico CA desenvolvidos no projeto RELLE da UFSC.....	37
Figura 11 – Diagrama de bancada de emulação mostrando componentes e conexões.	38
Figura 12 – Diagrama de blocos da bancada incluindo os diferentes tipos de cabos usados em sua construção.	39
Figura 13 – <i>Layout</i> de <i>hardware</i> para laboratório remoto IoT composto de braço robótico, bomba CC, sensor de fluxo de água e câmeras.....	40
Figura 14 – Arquitetura do laboratório remoto.	40
Figura 15 – Projeto de laboratório remoto para ensino de levitação magnética.	41
Figura 16 – Arquitetura do laboratório remoto.	42
Figura 17 - Arquitetura do sistema laboratorial.....	43
Figura 18 – Metodologia do trabalho.....	44
Figura 19 – Arquitetura simplificada proposta neste trabalho.	55
Figura 20 – Possibilidades de adaptações e expansão da arquitetura proposta.	57
Figura 21 – Organização da plataforma proposta em módulos.	58
Figura 22 – Esquema elétrico do módulo circuito.....	61
Figura 23 – Partes principais de um gêmeo digital.	65
Figura 24 – Imagem do gêmeo digital desenvolvido para o projeto.	66
Figura 25 – Código RGB adotado para ilustrar com diferentes tons de amarelo no gêmeo digital a intensidade de luminosidade nas lâmpadas do circuito real.	67
Figura 26 – Imagem salva de um circuito montado.....	68
Figura 27 – Tela responsiva do laboratório remoto.....	70

Figura 28 – Diferentes montagens da parte externa do experimento.....	74
Figura 29 – Resultados do teste de desempenho do servidor <i>Web</i>	76
Figura 30 – Pontuação obtida pela página <i>Web</i> no Google Lighthouse.....	79
Figura 31 – Combinação 1	91
Figura 32 – Combinação 2	91
Figura 33 – Combinação 3	91
Figura 34 – Combinação 4	91
Figura 35 – Combinação 5	91
Figura 36 – Combinação 6	91
Figura 37 – Combinação 7	91
Figura 38 – Combinação 8	91
Figura 39 – Combinação 9	91
Figura 40 – Combinação 10	91
Figura 41 – Combinação 11	91
Figura 42 – Combinação 12	91
Figura 43 – Combinação 13	92
Figura 44 – Combinação 14	92
Figura 45 – Combinação 15	92
Figura 46 – Combinação 16	92
Figura 47 – Combinação 17	92
Figura 48 – Combinação 18	92
Figura 49 – Combinação 19	92
Figura 50 – Combinação 20	92
Figura 51 – Combinação 21	92
Figura 52 – Combinação 22	92
Figura 53 – Combinação 23	92
Figura 54 – Combinação 24	92
Figura 55 – Combinação 25	93
Figura 56 – Combinação 26	93
Figura 57 – Combinação 27	93

LISTA DE TABELAS

Tabela 1 – Bibliotecas utilizadas nos SBMs.	53
Tabela 2 – Custo unitário por item e total geral do projeto em reais (R\$) e dólares (U\$).	69
Tabela 3 – Estados das chaves nas 17 configurações distintas de circuitos elétricos.	72
Tabela 4 – Comparativo da medição de tensão nos pontos de interesse (M1 a M4).	73

SUMÁRIO

1	Introdução	17
1.1	Motivação	20
1.2	Justificativa	21
1.3	Objetivo	24
1.3.1	Objetivos específicos	24
1.4	Organização do texto	24
2	Fundamentação teórica	26
2.1	Educação 4.0	26
2.2	Laboratórios didáticos	28
2.2.1	Laboratórios remotos	29
2.3	Arquiteturas de placa única	33
3	Trabalhos relacionados.....	35
3.1	Laboratórios remotos em funcionamento.....	35
3.2	Propostas de laboratórios remotos.....	39
3.3	Análise dos trabalhos relacionados.....	43
4	Metodologia.....	44
4.1	Materiais e métodos	44
4.1.1	Microcontroladores de placa única.....	44
4.1.2	Protocolo I2C.....	47
4.1.3	Medidor de corrente e tensão.....	49
4.1.4	Display OLED.....	50
4.1.5	Módulo de relés.....	50
4.1.6	Linguagem de marcação	51
4.1.7	Linguagem de script	51
4.1.8	Folha de estilos	52
4.1.9	API da Web.....	52
4.1.10	IDE Arduino	52
4.1.11	Linguagem de programação C++.....	53
4.1.12	Bibliotecas do ESP32 e ESP32-CAM	53
5	Plataforma proposta	55
5.1	Considerações sobre a arquitetura proposta.....	56
5.2	Organização da plataforma	57
5.2.1	Módulo câmera	58

5.2.2	Módulo fonte de alimentação	59
5.2.3	Módulo controlador	59
5.2.4	Módulo circuito.....	60
5.3	Funcionamento da plataforma	64
5.4	Gêmeo digital.....	64
5.5	Custos.....	68
6	Resultados	70
6.1	Aspectos construtivos e vista externa do experimento	74
6.2	Teste de desempenho do servidor <i>Web</i>	74
6.3	Teste de desempenho da página <i>Web</i>	77
7	Conclusão	80
8	Trabalhos futuros	82
	Referências bibliográficas	83
	APÊNDICE A – Proposta de Roteiro de Aprendizagem.....	89
	APÊNDICE B – Combinações para formação de um circuito.....	91

LISTA DE ABREVIações E SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
API	<i>Application Programming Interface</i>
ARPANET	<i>Advanced Research Projects Agency Network</i>
CA	Corrente Alternada
CC	Corrente Contínua
CMOS	<i>Complementary Metal-Oxide-Semiconductor</i>
CPU	<i>Central Process Unit</i>
CSS	<i>Cascading Style Sheets</i>
DAQ	<i>Data Acquisition</i>
DIY	<i>Do it Yourself</i>
FPGA	<i>Field Programmable Gate Array</i>
GPIO	<i>General Purpose Input/Output</i>
HTML	<i>Hypertext Markup Language</i>
HTTP	<i>Hypertext Transfer Protocol</i>
I2C	<i>Inter Integrated Circuit</i>
I2S	<i>Inter-IC Sound</i>
IDE	<i>Integrated Development Environment</i>
IEC	<i>International Electrotechnical Commission</i>
IoT	<i>Internet of Things</i>
LabVIEW	<i>Laboratory Virtual Instrument Engineering Workbench</i>
LE	<i>Low Energy</i>
LED	<i>Light-Emitting Diode</i>
LPWAN	<i>Lower Power Wide Area Networks</i>
MATLAB	<i>MATrix LABoratory</i>
MOOC	<i>Massive Open Online Course</i>
OLED	<i>Organic Light-Emitting Diode</i>
PC	<i>Personal Computer</i>
PLC	<i>Programmable Logic Controller</i>
PWA	<i>Progressive Web App</i>
PWM	<i>Pulse Width Modulation</i>
RELLE	<i>Remote Labs Learning Environment</i>
RFID	<i>Radio Frequency Identification</i>
RGB	<i>Red Green Blue</i>

SBA	<i>Single Board Architectures</i>
SBC	<i>Single Board Computer</i>
SBM	<i>Single Board Microcontroller</i>
SCADA	<i>Supervisory Control and Data Acquisition</i>
SCCB	<i>Serial Camera Control Bus</i>
SCL	<i>Serial Clock</i>
SD	<i>Secure Digital</i>
SDA	<i>Serial Data</i>
SEO	<i>Search Engine Optimization</i>
SPI	<i>Serial Peripheral Interface</i>
SPIFFS	<i>SPI Flash File System</i>
SPST	<i>Single Pole Single Throw</i>
SVG	<i>Scalable Vector Graphics</i>
TCP/IP	<i>Transmission Control Protocol/Internet Protocol</i>
TIC	Tecnologias da Informação e Comunicação
UART	<i>Universal Asynchronous Receiver-Transmitter</i>
UFSC	Universidade Federal de Santa Catarina
UI	<i>User Interface</i>
UNICAMP	Universidade Estadual de Campinas
URL	<i>Uniform Resource Locator</i>
USB	<i>Universal Serial Bus</i>
UXGA	<i>Ultra-Extended Graphics Array</i>

1 INTRODUÇÃO

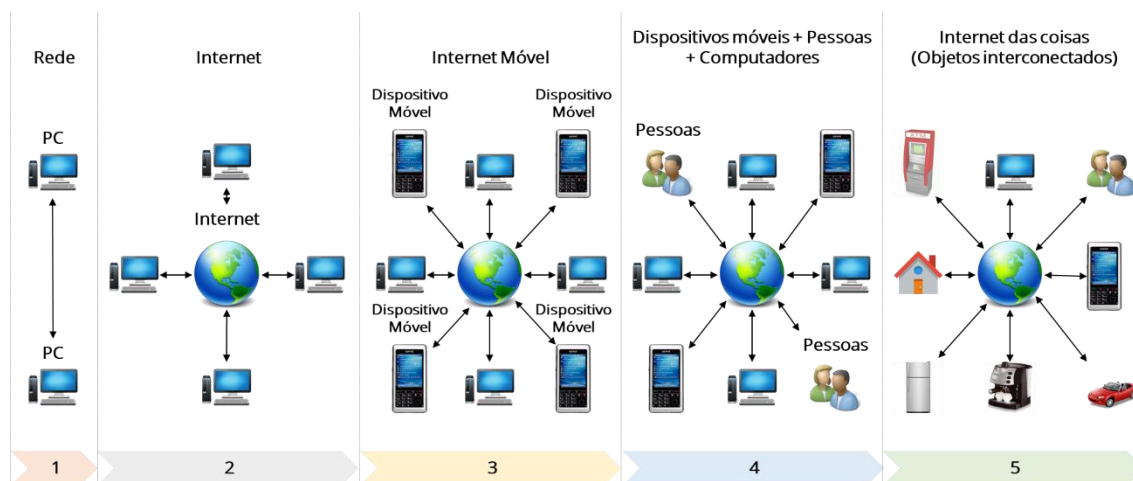
O envio e o recebimento de dados via rede de computadores vêm evoluindo temporalmente. A história da comunicação entre computadores se inicia no final da década de 1960, mais especificamente, em 1969. Nesse ano acontece o envio da primeira mensagem entre dois computadores. Esse fato histórico só foi possível graças a ARPANET (Rede da Agência para Projetos de Pesquisa Avançada, do inglês *Advanced Research Projects Agency Network*), considerada a primeira rede de computadores (PERERA, 2017). A ARPANET interligava diversos departamentos de pesquisas em todo o Estados Unidos visando possibilitar a transmissão de dados sigilosos e militares.

Em 1973 foi registrada a patente da primeira etiqueta de RFID (Identificação por Radiofrequência, do inglês *Radio Frequency Identification*), tecnologia de identificação automática através de sinais de rádio, a qual atualmente é usada nas redes de dispositivos IoT (Internet das Coisas, do inglês *Internet of Things*). Depois do surgimento da RFID, os próximos marcos históricos de comunicação entre computadores ou dispositivos eletrônicos incluem o surgimento da Internet, das redes móveis e do conceito de IoT (PERERA, 2017).

A evolução da Internet, ilustrada pela **Figura 1**, acontece em cinco fases. Inicia-se com a conexão de dois computadores e depois avança para a criação da rede mundial de computadores (*World Wide Web*), quando um grande número de computadores está conectado. A Internet Móvel surgiu quando os dispositivos móveis, como os celulares, adquiriram a capacidade de se conectarem à Internet. A quarta fase é caracterizada pela conexão entre pessoas, quando a *Web* passa a ser uma via de mão dupla em que os usuários podem abastecer as plataformas com informações permitindo uma produção de conteúdo mais fluída (redes sociais, *blogs*, *wikis*, etc). Finalmente, na quinta fase, surgiu a Internet das Coisas, quando objetos do cotidiano são envelopados com uma camada digital e adicionados à Internet, ou seja, são capazes de interagir com pessoas e outros

objetos permitindo cruzamento de dados e tomadas de decisão mais precisas (PERERA; LIU; JAYAWARDENA; CHEN, 2015).

Figura 1 – Evolução da Internet.



Fonte: Adaptado de (PERERA; LIU; JAYAWARDENA; CHEN, 2015)

O termo IoT é relativamente novo, foi inventado em 1999 por Kevin Ashton, na época diretor-executivo da Auto-ID Center, para promover a tecnologia RFID, mas a ideia de ter vários dispositivos conectados em redes remonta aos anos 1970 e naquela época era chamada “computação difusa”.

A IoT não chamou a atenção de grandes empresas e de pesquisadores até 2011 e só atingiu os grandes mercados no início de 2014 (LUETH, 2014). Desde então, a IoT vem revolucionando a área de tecnologia da informação, sendo considerada a tecnologia do século XXI, com previsão de 75 bilhões de dispositivos conectados até 2025 (STATISTA RESEARCH DEPARTMENT, 2016).

Apesar de existirem diferentes visões e perspectivas, em linhas gerais, pode-se definir IoT como um conceito que engloba uma grande variedade de coisas: objetos, sensores e dispositivos conectados por sistemas de comunicações e suportados por uma infraestrutura de informação para fornecer serviços com valor agregado (PERERA, 2017).

Dentre as principais aplicações da IoT, pode-se citar:

- Automação residencial;
- Dispositivos vestíveis;

- Automação industrial;
- Monitoramento da saúde;
- Monitoramento de atividades esportivas;
- Laboratórios e experimentos de baixo custo, etc.

No contexto do último tópico de aplicação de IoT listado acima, observou-se um crescimento recente das investigações e pesquisas em decorrência da pandemia da Covid-19, a qual fez com que várias instituições de ensino recomendassem a educação à distância como medida de contenção de disseminação do vírus e como alternativa para a continuidade das aulas e atividades de ensino (VARGAS; CUERO; TORRES, 2020).

Em 2020, durante a pandemia da Covid-19, 29 países, dos 33, da América Latina e Caribe continuaram com o processo educacional através de diferentes modalidades como virtual, à distância e mista (CEPAL-UNESCO, 2020).

Um dos principais problemas do ensino à distância é a dificuldade de formar nos alunos as habilidades que, em geral, só são adquiridas com o uso da prática em laboratórios. Paralelamente, nos cursos das áreas de ciências, engenharias e tecnologias, as atividades práticas em laboratórios são essenciais no processo de ensino-aprendizagem. Entretanto, o acesso presencial aos laboratórios está limitado ao período em que a instituição de ensino está aberta, em funcionamento. Uma alternativa viável que ajudaria a superar esse obstáculo, seria a utilização de laboratórios remotos (POLIAKOV; RIDA, 2020).

Segundo (VILELA; GERMANO; MONTEIRO; CARVALHO, 2019), um laboratório remoto

permite a realização de experimentos reais, a partir de qualquer lugar, em qualquer horário, através de uma interface de controle remoto, possibilitando a um usuário com acesso à Internet configurar um equipamento experimental e iniciar uma experiência, que poderá ser observada em tempo real na tela de

um computador que recebe as imagens, captadas por câmeras locais.

Ao disponibilizar um laboratório remoto, uma instituição de ensino pode otimizar a utilização desse recurso educacional visto que, diferentemente de uma estrutura física acessada presencialmente, um laboratório remoto está disponível 24 horas e em qualquer dia da semana, atendendo um número maior de alunos. Para os alunos, um dos benefícios é a possibilidade de acessar um laboratório e realizar seus experimentos em horários alternativos e em qualquer lugar, bastando para isso, ter acesso a um computador ou celular conectado à Internet.

1.1 MOTIVAÇÃO

A principal motivação deste trabalho é promover o acesso e uso de laboratórios remotos permitindo que:

- Alunos que não possuem um laboratório em sua instituição tenham acesso a plataformas para realização de atividades práticas;
- Instituições com limitação de recursos nas quais um grande número de alunos precisa compartilhar um espaço físico restrito possam aumentar seu alcance;
- O acesso daqueles alunos que pretendem refazer os experimentos em horários alternativos seja facilitado;
- Os conceitos aprendidos em aulas teóricas sejam fixados e implementados na prática de maneira autônoma pelos alunos;
- Experimentos básicos sejam realizados sem o requerimento de grandes recursos computacionais.

Adicionalmente, outra motivação foi a necessidade observada durante a pandemia da Covid-19, de desenvolvimento de plataformas que facilitem o acesso dos alunos à realização de atividades experimentais, as quais complementam os conhecimentos teóricos adquiridos, em especial na área de ciências exatas, quando a entrada no espaço físico das instituições é restrita.

1.2 JUSTIFICATIVA

No Brasil, o uso de laboratórios remotos para fins educacionais ainda é bem limitado. Em uma revisão da literatura conduzida em (DA SILVA BERALDO; DE OLIVEIRA; STRINGHINI, 2021), abrangendo os anos de 2004 a 2020, foram encontrados somente 13 artigos descrevendo o uso e implantação de laboratórios remotos no Brasil.

Um estudo comparativo entre um laboratório tradicional e um laboratório remoto realizado em (VILELA; GERMANO; MONTEIRO; CARVALHO, 2019) concluiu que a atividade experimental realizada remotamente oferece oportunidades de aprendizado aos alunos similares às aquelas atividades experimentais realizadas presencialmente.

Segundo (SILVA, Rafael de Amorim; NOVA; VASCONCELOS; CALADO; BRANCO; BRAGA, 2017), há diversas tecnologias que podem ser aplicadas na educação para melhorar o processo de ensino-aprendizagem, sendo que as tecnologias relacionadas à IoT são as que têm os maiores potenciais para essa tarefa.

Segundo (MIRANDA; LOPEZ; NAVARRO; BUSTAMANTE; MOLINA; MOLINA, 2019), atualmente a educação vive a era da chamada Educação 4.0, caracterizada por avanços didáticos e tecnológicos, especialmente das TICs (Tecnologias da Informação e Comunicação), no processo de ensino e aprendizagem e, nesse contexto, as TICs são alimentadas por tecnologias presentes na IoT e

os avanços nos sistemas de comunicação têm um papel essencial nas práticas educacionais atuais, pois são responsáveis por fornecer plataformas de ensino à distância através de diferentes técnicas de aprendizagem, considerando o uso de recursos *online*, cursos *online* abertos e massivos (do inglês *Massive Open Online Course* – MOOC), laboratórios remotos, redes e bibliotecas digitais, entre outros.

Nesse contexto, com a popularização da IoT e surgimento da Educação 4.0, o desenvolvimento de laboratórios remotos tornou-se mais simples e mais barato, o que contribuiu para que mais alunos tenham acesso a essas tecnologias educacionais, aumentando as possibilidades de desenvolverem as competências essenciais necessárias à sua formação.

Na última década, placas de desenvolvimento, computadores de placa única (do inglês *Single Board Computers* – SBCs) e microcontroladores de placa única (do inglês *Single Board Microcontrollers* – SBMs) integrados com sensores e tecnologias de comunicação tornaram-se uma solução muito popular e interessante para o desenvolvimento e implementação de soluções, entre elas os laboratórios remotos, que ajudariam no desenvolvimento de competências essenciais por meio da aprendizagem colaborativa e mais rica (DRACHSLER; SCHNEIDER, 2018; RIQUELME; MUNOZ; MAC LEAN; VILLARROEL; BARCELOS; DE ALBUQUERQUE, 2019). Tais recursos são cada vez mais utilizados em robótica, domótica, teste e medição, projetos DIY (Faça Você Mesmo, do inglês *Do It Yourself*), dispositivos IoT como suporte no ensino de ciência, tecnologia, engenharia e matemática (ÁLVAREZ; MOZO; DURÁN, 2021).

No que se refere aos equipamentos físicos, programas, tipos e quantidades de itens utilizados, diferentes abordagens podem ser usadas para o desenvolvimento de laboratórios remotos. Em (KALÚZ; GARCÍA-ZUBÍA; FIKAR; ČIRKA, 2015), por exemplo, são descritas 5 arquiteturas frequentemente usadas para o desenvolvimento e implementação de laboratórios remotos. Dentre as arquiteturas descritas, a **Cliente → SBC → SBM ou FPGA (Matriz de Portas Programáveis, do inglês *Field Programmable Gate Array*) → Experimento**, merece destaque por ser usualmente empregada em laboratórios remotos de baixo custo, como os apresentados em (MELLOS; SILVA; LIMA; SIMÃO, 2016), em (SARAVANA MOHAN; KARTHIKEYAN; RAM KUMAR; RUPESH, 2019), em (FERNÁNDEZ-PACHECO; MARTIN; CASTRO, 2019), em (CHAVES; ASSUMPÇÃO; FERREIRA; CARDIERI;

BRANQUINHO; FRUETT, 2021) e (CHAVES, 2022), em (GORBOUNOV; CHEN, 2021), em (AZAD, 2021) e em (NENOV; EVSTATIEV; KADIROVA, 2022). Nesta arquitetura, obrigatoriamente é utilizado um SBC (e.g. Raspberry Pi), mais dispendioso que um SBM (e.g. Arduino), como dispositivo responsável pelo processamento, supervisão de dados e pela geração do vídeo em tempo real a partir de uma câmera. Além disso, também é empregada uma FPGA ou um SBM como unidade de controle.

Nesse sentido, essa dissertação de mestrado apresenta o desenvolvimento de uma plataforma experimental com acesso e controle remotos via dispositivos conectados à Internet, a qual, diferentemente da maioria das soluções encontradas na literatura, é implementada diretamente em um microcontrolador de placa única ou SBM. Essa plataforma consiste em um laboratório que faz uso de recursos e tecnologias relacionadas à IoT, visando disponibilizar uma ferramenta de suporte para o ensino de conceitos básicos de eletricidade.

Para implementar o laboratório remoto, propõe-se uma nova arquitetura, distinta das listadas em (KALÚZ; GARCÍA-ZUBÍA; FIKAR; ČIRKA, 2015), a qual consiste na simplificação do modelo **Cliente → SBC → SBM ou FPGA → Experimento**.

Além da arquitetura simplificada, outra contribuição deste trabalho é o uso do conceito “gêmeo digital”, disponibilizando uma representação digital interativa do experimento físico. Nesse ponto, é importante ressaltar que um gêmeo digital consiste em uma representação virtual de um produto ou processo físico, usado para entender e prever as características de desempenho da contraparte física (SIEMENS, [s.d.]).

Tendo em vista a evolução das tecnologias de IoT e a expansão do conceito de Educação 4.0, várias plataformas experimentais com acesso remoto similares à proposta nesta dissertação vêm sendo desenvolvidas na última década. No entanto, a realização deste trabalho justifica-se por suas características inovadoras em termos de: arquitetura simplificada, emprego de dispositivos de baixo custo, e

reduzida necessidade de suporte local ou remoto para realização das atividades experimentais.

1.3 OBJETIVO

O objetivo deste trabalho é desenvolver uma plataforma experimental de suporte para ensino de conceitos de eletricidade básica para estudantes do ensino médio e/ou técnico, com acesso e controle remotos via dispositivos conectados à Internet e implementação direta em um microcontrolador de placa única ou SBM.

1.3.1 *Objetivos específicos*

- Aplicar uma arquitetura simplificada no desenvolvimento do laboratório remoto;
- Utilizar apenas SBMs no processamento, supervisão de dados e controle;
- Aplicar o conceito de gêmeo digital;
- Utilizar sensores, dispositivos eletrônicos e microcontroladores de baixo custo para aumentar a acessibilidade de diferentes instituições que pretendam replicar o projeto.

1.4 ORGANIZAÇÃO DO TEXTO

O texto desta dissertação está organizado conforme descrito a seguir. No Capítulo 1 foram expostas as principais justificativas para realização deste trabalho e foi explicado o cenário que motivou a realização desta pesquisa, além de serem listados os objetivos que devem ser alcançados.

No Capítulo 2 é apresentada a **Fundamentação teórica** descrevendo conceitos relevantes com os quais o leitor deve se familiarizar para proceder com a leitura, como: Educação 4.0, laboratórios didáticos, laboratórios remotos e arquiteturas de placa única, os quais são a base dessa pesquisa.

No Capítulo 3 são apresentados os **Trabalhos relacionados** com essa dissertação. A partir de uma abrangente revisão bibliográfica, foram selecionados

alguns laboratórios remotos em funcionamento ou desenvolvimento (propostas de laboratórios remotos) que inspiraram essa pesquisa, além de realizar-se uma análise das arquiteturas empregadas nos trabalhos citados.

No Capítulo 4 é descrita a **Metodologia** empregada neste trabalho, bem como são listados os materiais e métodos necessários para se replicar a plataforma experimental proposta, em termos de dispositivos de *hardware* e soluções de *software*.

No Capítulo 5 é introduzida a **Plataforma proposta** descrevendo-a por completo.

No Capítulo 6 são apresentados os **Resultados** obtidos, incluindo testes de desempenho.

No Capítulo 7 é apresentada uma **Conclusão** com base nos resultados obtidos e comparações com o que está disponível na literatura.

No Capítulo 8 apresenta-se uma lista de sugestões de **Trabalhos Futuros** para dar continuidade a essa pesquisa.

Finalmente, são apresentados dois apêndices, o primeiro com uma proposta de roteiro de aprendizagem a ser adaptado pelo docente que fará uso da plataforma, e o segundo com uma lista de figuras mostrando as diferentes montagens em série/paralelo que podem ser obtidas com o mesmo circuito experimental proposto.

2 FUNDAMENTAÇÃO TEÓRICA

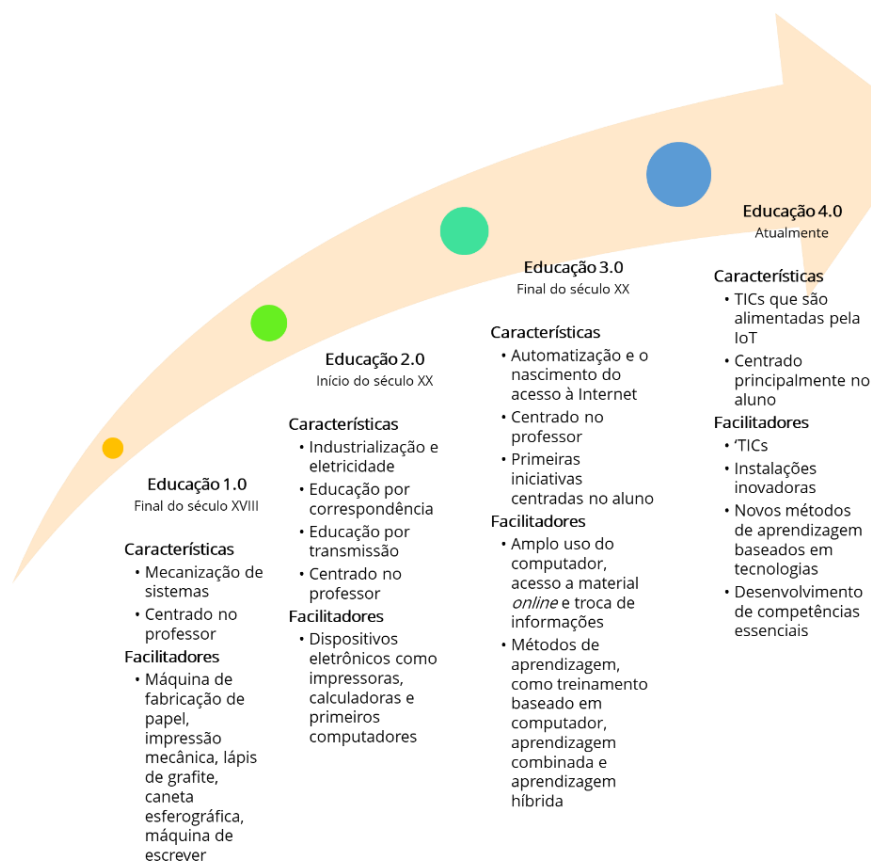
Neste capítulo, são apresentadas as definições que fundamentam a proposta do trabalho. Inicialmente, é abordado o conceito de Educação 4.0 e como se relaciona com esta pesquisa. Em seguida, são listadas as classificações de laboratórios didáticos, bem como são discutidas as principais arquiteturas usadas nos diferentes tipos de laboratórios. Finalmente, é apresentada a classificação das arquiteturas de placa única e suas principais plataformas, com um enfoque especial na subdivisão entre computadores de placa única ou SBCs, microcontroladores de placa única ou SBMs e matrizes de portas programáveis ou FPGAs.

2.1 EDUCAÇÃO 4.0

A educação sempre evolui temporalmente de acordo com o surgimento de novas tecnologias, avanços no processo de ensino-aprendizagem e alterações nas dinâmicas sociais regionais e locais. Atualmente a educação está vivenciando um período chamado Educação 4.0, no qual são empregados métodos de ensino e aprendizagem, instalações inovadoras alinhadas com tecnologias emergentes visando preparar os alunos para enfrentar problemas globais, regionais ou locais, atuais e futuros (MIRANDA; LOPEZ; NAVARRO; BUSTAMANTE; MOLINA; MOLINA, 2019).

Observa-se que existe uma relação entre a evolução da educação com as revoluções industriais, pois são nessas revoluções que se dão os grandes avanços tecnológicos, os quais naturalmente são facilitadores do progresso da educação (MIRANDA; LOPEZ; NAVARRO; BUSTAMANTE; MOLINA; MOLINA, 2019). A **Figura 2** ilustra a evolução da educação desde o final do século XVIII (período conhecido como Educação 1.0) até os dias atuais (Educação 4.0), destacando as características de cada período e listando os principais facilitadores.

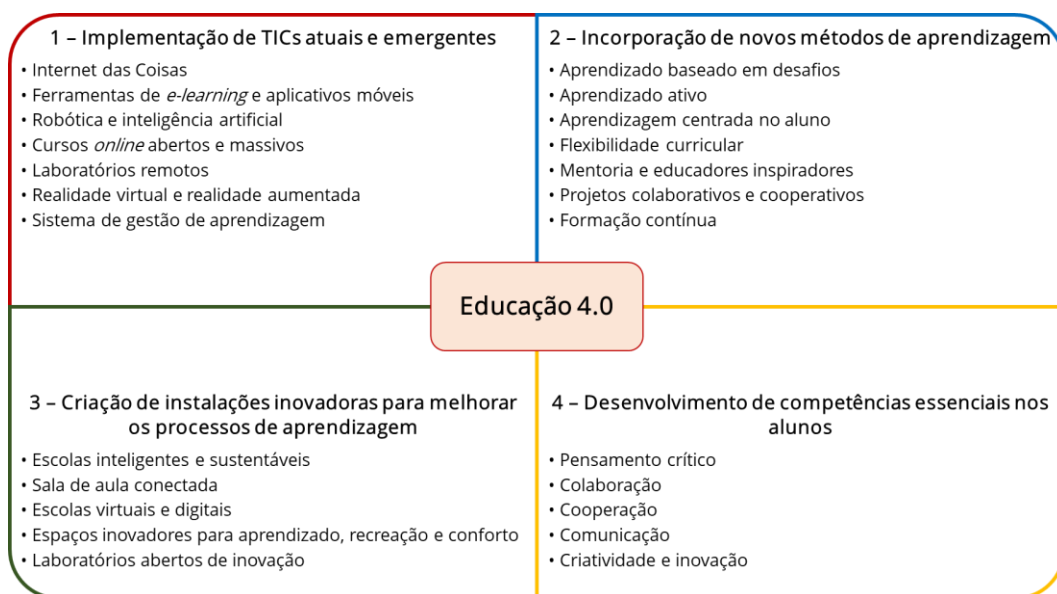
Figura 2 – Da Educação 1.0 à Educação 4.0.



Fonte: Adaptado de (MIRANDA; LOPEZ; NAVARRO; BUSTAMANTE; MOLINA; MOLINA, 2019)

Segundo (MIRANDA; LOPEZ; NAVARRO; BUSTAMANTE; MOLINA; MOLINA, 2019), a era da Educação 4.0 é promovida por quatro elementos: (1) implementação de TICs atuais e emergentes, (2) incorporação de novos métodos de aprendizagem, (3) criação de instalações inovadoras para melhorar os processos de aprendizagem e (4) desenvolvimento de competências essenciais nos alunos, conforme ilustrado pela **Figura 3**.

Figura 3 – Quatro elementos centrais da Educação 4.0.



Fonte: Adaptado de (MIRANDA; LOPEZ; NAVARRO; BUSTAMANTE; MOLINA; MOLINA, 2019)

Este trabalho se relaciona com o conceito de Educação 4.0 ao utilizar tecnologias da IoT, ao contribuir para a criação de espaços inovadores para aprendizado, promovendo o aprendizado ativo de alunos e, principalmente, ao propor a implementação e uso de um laboratório remoto.

2.2 LABORATÓRIOS DIDÁTICOS

A versão *online* do dicionário (MICHAELIS, 2022) define um laboratório didático como um local ou sala especial de trabalho, experimentação e investigações científicas, equipada com aparelhagem específica para pesquisa e experimentos. Os laboratórios didáticos podem ser classificados de várias maneiras. Por exemplo, (LIMA; NETO; MARTINS, 2005) usam dois critérios de classificações de laboratórios: **(1)** de acordo com a forma de acesso – presencial ou remoto; **(2)** de acordo com a estrutura física – real ou virtual. Por outro lado, (HERADIO; DE LA TORRE; DORMIDO, 2016) combinam esses dois critérios e classificam os laboratórios como:

- 1. Experimento real de acesso local** – São os tradicionais laboratórios práticos, onde o aluno fica em frente a um computador conectado ao experimento real.

2. **Experimento simulado de acesso local** – Todo o ambiente é desenvolvido via *software* e a interface de experimentação funciona sobre um recurso simulado, virtual e fisicamente inexistente instalado em um computador.
3. **Experimento real de acesso remoto** – O experimento real é acessado através da Internet. O usuário opera e controla remotamente um experimento real através de uma interface de experimentação. Essa abordagem é chamada de **laboratório remoto**.
4. **Experimento simulado de acesso remoto** – Esta forma de experimentação é semelhante à anterior, mas substituindo o sistema físico por um modelo matemático. O aluno opera com a interface de experimentação em um sistema virtual acessado pela Internet. Isso é o que normalmente é conhecido como **laboratório virtual**.

Seguindo as classificações apresentadas na literatura, considera-se que o laboratório desenvolvido neste trabalho consiste em um experimento real de acesso remoto, por esse motivo, uma análise mais aprofundada das diferentes arquiteturas de laboratórios remotos será apresentada a seguir.

2.2.1 Laboratórios remotos

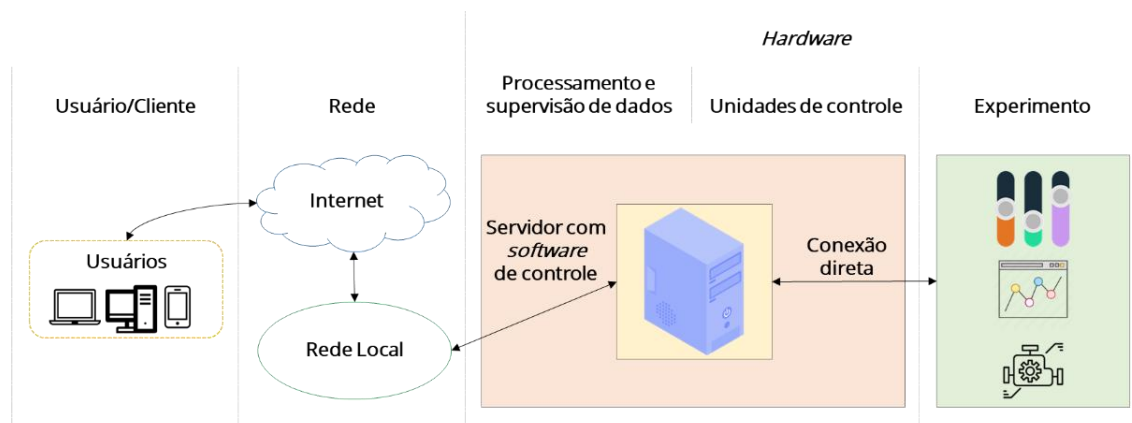
Existem vários laboratórios remotos em funcionamento no Brasil e no exterior. Conforme mencionado anteriormente, (KALÚZ; GARCÍA-ZUBÍA; FIKAR; ČIRKA, 2015) lista 5 arquiteturas diferentes encontradas na literatura para o desenvolvimento de laboratórios remotos, as quais variam em relação às estruturas de *hardware* e *software* utilizadas. A seguir há uma breve descrição de como cada uma dessas arquiteturas funcionam.

2.2.1.1 Arquitetura Cliente – Servidor com *software* de controle – Experimento

Nessa arquitetura, há uma conexão direta, via porta serial ou USB (Barramento Serial Universal, do inglês *Universal Serial Bus*), entre um PC

(Computador Pessoal, do inglês *Personal Computer*) que tem a função de servidor, como ilustrado pela **Figura 4**.

Figura 4 – Arquitetura Cliente – Servidor com *software* de controle – Experimento.



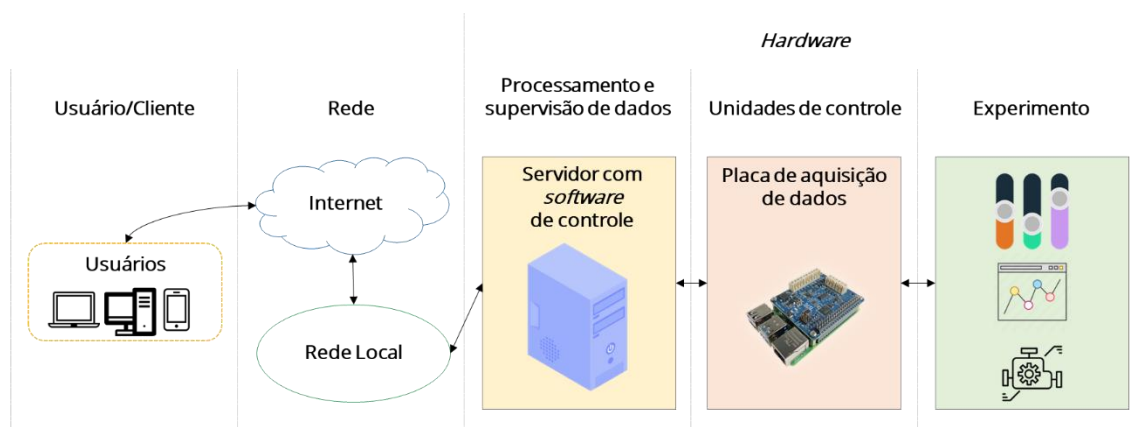
Fonte: Adaptado de (KALÚZ; GARCÍA-ZUBÍA; FIKAR; ČIRKA, 2015)

O PC possui um *software* próprio para comunicação com o cliente e também para o controle do experimento. Geralmente, o servidor apresenta soluções de *software* como, por exemplo, MATLAB (do inglês *MATrix LABoratory*) e LabVIEW (do inglês *Laboratory Virtual Instrument Engineering Workbench*). A principal vantagem deste tipo de arquitetura está na sua baixa dificuldade de implementação. Em contrapartida, a utilização de *softwares* comerciais como MATLAB e LabVIEW aumenta o custo final do laboratório remoto (KALÚZ; GARCÍA-ZUBÍA; FIKAR; ČIRKA, 2015).

2.2.1.2 Arquitetura Cliente – Servidor com *software* de controle – Placa DAQ – Experimento

Essa arquitetura é muito similar à arquitetura mencionada anteriormente. A diferença é que entre o servidor e o experimento há uma placa DAQ (Aquisição de Dados, do inglês *Data Acquisition*), como ilustrado pela **Figura 5**.

Figura 5 – Arquitetura Cliente – Servidor com *software* de controle – placa DAQ – Experimento.



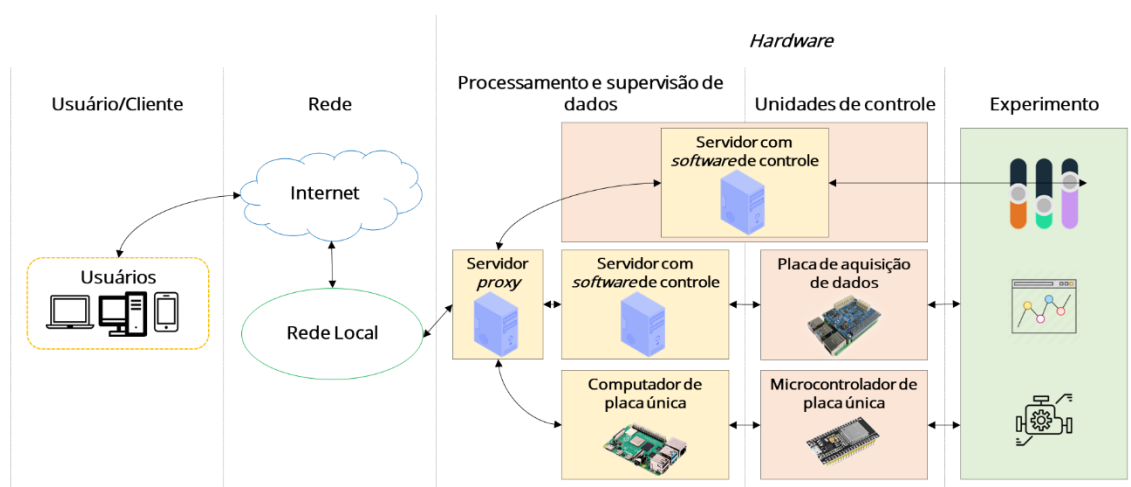
Fonte: Adaptado de (KALÚZ; GARCÍA-ZUBÍA; FIKAR; ČIRKA, 2015)

A placa DAQ geralmente é usada quando não é possível interconectar diretamente o servidor com o experimento. Esse tipo de configuração de *hardware* geralmente reduz os requisitos do *software* de processamento de sinal incluído no servidor, mas também aumenta o custo dos elementos físicos que constituem o experimento (KALÚZ; GARCÍA-ZUBÍA; FIKAR; ČIRKA, 2015).

2.2.1.3 Arquitetura Cliente – Servidor *proxy* – Nós dos laboratórios – Experimentos

Essa arquitetura, também conhecida como arquitetura ramificada, fornece uma capacidade maior de possíveis conexões quando comparada a outras, como ilustrado pela Figura 6.

Figura 6 – Arquitetura Cliente – Servidor *proxy* – Nós do laboratório – Experimentos.



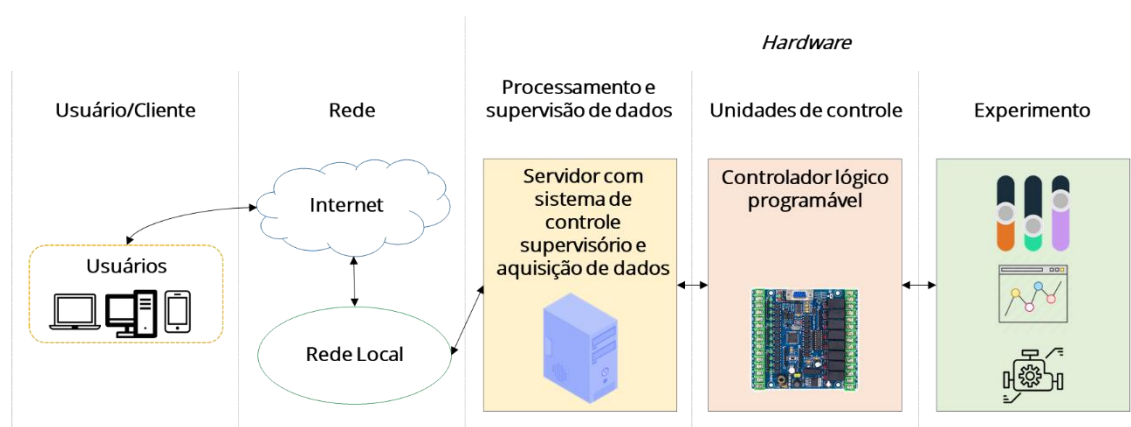
Fonte: Adaptado de (KALÚZ; GARCÍA-ZUBÍA; FIKAR; ČIRKA, 2015)

Um servidor *proxy* é usado para gerenciar as conexões de diferentes clientes aos diferentes nós do laboratório. Cada nó geralmente consiste em um servidor local equipado com *software* e *hardware* apropriados para aquisição de dados e um experimento. A principal vantagem dessa arquitetura é que o gerenciamento do acesso dos usuários via servidor *proxy* permite que os nós dos laboratórios possam estar localizados, até mesmo, em diferentes países (KALÚZ; GARCÍA-ZUBÍA; FIKAR; ČIRKA, 2015).

2.2.1.4 Arquitetura Cliente – Servidor com SCADA – PLC – Experimento

Nessa arquitetura, há uma combinação do sistema SCADA (Sistema de Controle Supervisório e Aquisição de Dados, do inglês *Supervisory Control and Data Acquisition*) e um PLC (Controlador Lógico Programável, do inglês *Programmable Logic Controller*), como ilustrado pela **Figura 7**.

Figura 7 – Arquitetura Cliente – Servidor com SCADA – PLC – Experimento.



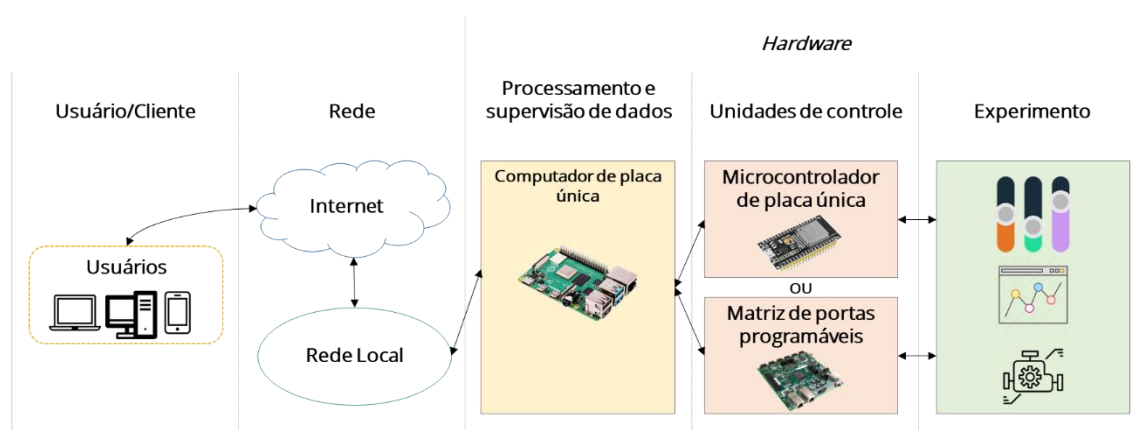
Fonte: Adaptado de (KALÚZ; GARCÍA-ZUBÍA; FIKAR; ČIRKA, 2015)

Essa arquitetura se caracteriza pelo seu alto custo decorrente do uso de *softwares* e *hardwares* comerciais, entretanto há redução significativa do esforço e tempo necessário para sua implementação (KALÚZ; GARCÍA-ZUBÍA; FIKAR; ČIRKA, 2015).

2.2.1.5 Arquitetura Cliente – SBC – SBM ou FGPA – Experimento

Esta arquitetura se difere das demais pelo uso de componentes baratos, sendo conhecida como uma arquitetura de baixo custo. É baseada em SBC, SBM ou FPGA, como ilustrado pela **Figura 8**.

Figura 8 – Arquitetura Cliente – SBC – SBM ou FPGA – Experimento.



Fonte: Adaptado de (KALÚZ; GARCÍA-ZUBÍA; FIKAR; ČIRKA, 2015)

Apesar do custo dos componentes físicos empregados nessa arquitetura ser incomparavelmente inferior ao dos *hardwares* industriais, há um esforço maior na implementação dos *softwares*, pois devem ser desenvolvidos para cada solução específica (KALÚZ; GARCÍA-ZUBÍA; FIKAR; ČIRKA, 2015). Por outro lado, em se tratando de laboratórios remotos, esse esforço poderia ser aproveitado didaticamente como parte integrante do aprendizado dos alunos no desenvolvimento de *softwares* e *firmwares*.

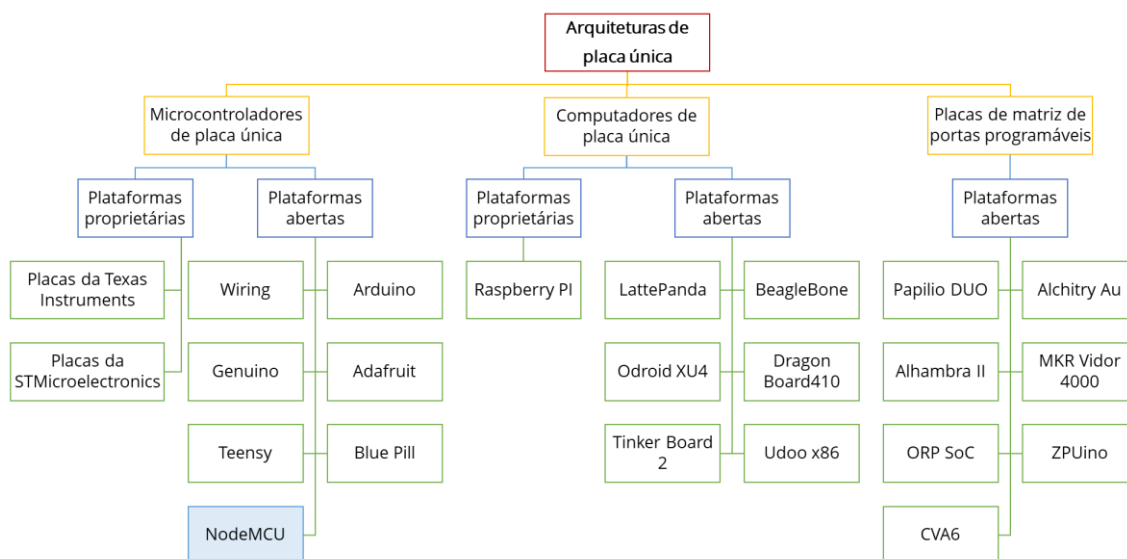
2.3 ARQUITETURAS DE PLACA ÚNICA

Segundo (ÁLVAREZ; MOZO; DURÁN, 2021), as SBAs (Arquiteturas de Placa Única, do inglês *Single Board Architectures*) se desenvolvem progressivamente de forma compacta, com baixo custo, com alta, baixa e média capacidade de processamento, sendo muito difundidas para o uso de sistemas de aquisição de dados em inúmeras aplicações.

As SBAs são semelhantes a um computador em termos dos elementos básicos que as compõem: memória, portas de entrada/saída e processador. A grande diferença entre um computador comum e uma SBA é que em uma SBA esses componentes básicos estão incluídos em um único *chip* monolítico (ÁLVAREZ; MOZO; DURÁN, 2021).

Pode-se classificar as SBAs em três formas: SBMs, SBCs e FPGAs, como ilustrado pela **Figura 9**.

Figura 9 – Classificação das arquiteturas de placa única e suas principais plataformas.



Fonte: Adaptado de (ÁLVAREZ; MOZO; DURÁN, 2021)

Embora originalmente não listada por (ÁLVAREZ; MOZO; DURÁN, 2021), a plataforma NodeMCU¹ (incluída em azul na **Figura 9**), projetada em 2014 pela empresa Espressif Systems, também possui SBMs amplamente utilizados em projetos IoT. É uma plataforma aberta e seus SBMs, além de serem de baixo custo, são capazes de se conectar em redes Wi-Fi, possuem servidores *Web* embutidos, suportam diversos protocolos de rede, incluindo a maioria dos protocolos usados para projetar sistemas IoT (DUTTA; KHURANA; DEVDUTT, 2021).

¹ Disponível em https://www.nodemcu.com/index_en.html

3 TRABALHOS RELACIONADOS

Neste capítulo são apresentados alguns trabalhos relacionados ao desenvolvimento e uso de laboratórios remotos de baixo custo. Os trabalhos foram organizados em duas seções: **laboratórios remotos em funcionamento** e **propostas de laboratórios remotos**. No final é realizada uma classificação dos trabalhos citados relacionando-os com a proposta desta dissertação.

3.1 LABORATÓRIOS REMOTOS EM FUNCIONAMENTO

Nessa seção é apresentada uma visão geral de três laboratórios remotos: o Painel Elétrico CC e o Painel Elétrico CA, disponibilizados pelo RELLE (do inglês *Remote Labs Learning Environment*) da Universidade Federal de Santa Catarina (UFSC) e uma bancada de emulação de redes de sensores sem fio desenvolvida por Leonardo Delforno no Programa de Pós Graduação (PPG) em Gestão de Redes de Telecomunicações da PUC-Campinas (DELFORNO, 2020) e reformulado para acesso remoto via plataforma NEWBen (*Network Environment Workbench*) desenvolvida no PPG em Engenharia Elétrica da Universidade Estadual de Campinas (UNICAMP) (CHAVES; ASSUMPÇÃO; FERREIRA; CARDIERI; BRANQUINHO; FRUETT, 2021).

O RELLE é composto de vários experimentos nas áreas de física (incluindo o estudo dos meios de propagação de calor por convecção e irradiação, segunda lei de Newton usando plano inclinado, conversão de energia elétrica pelo efeito fotovoltaico, óptica, condução de calor em barras metálicas e pêndulos), biologia (incluindo propostas ainda em teste do estudo de tecidos via microscópio e a relação entre ambiente e biodiversidade pela observação da água), química e robótica (incluindo várias plataformas para programação visual, controle de sensores e atuadores via Arduino). Contudo, dentre os diversos experimentos disponibilizados na página do projeto (<http://relle.ufsc.br/>), dois deles merecem destaque por terem inspirado o desenvolvimento do laboratório remoto proposto nessa dissertação: Painel Elétrico CC e Painel Elétrico CA.

O primeiro laboratório abordado é o **Painel Elétrico CC**, que possibilita o estudo das associações em série, paralela e mista em redes de corrente contínua (CC). Segundo o seu Manual Técnico, disponibilizado em (MELLOS; SILVA; LIMA; SIMÃO, 2016), o experimento Painel Elétrico CC tem como objetivo:

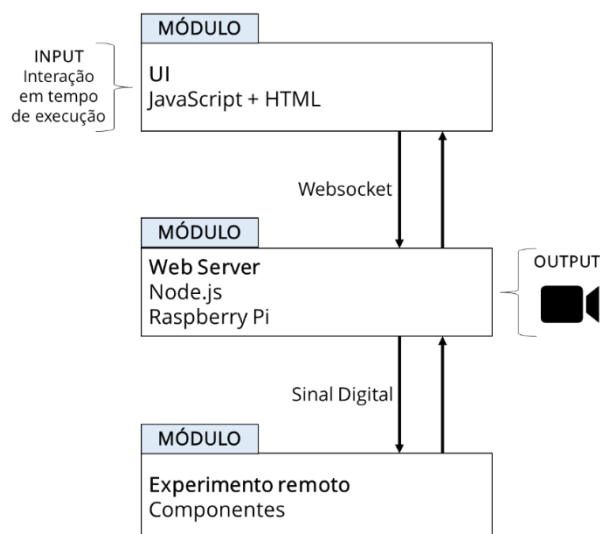
Auxiliar os estudantes do Ensino Médio e do Ensino Superior a efetuar práticas relacionadas à associação de resistores (série, paralela e mista), bem como, a análise de circuitos eletrônicos em corrente contínua usando as leis básicas de análise de circuitos: Lei de Ohm, Lei de Kirchhoff das Tensões ou Malhas e a Lei de Kirchhoff das Correntes ou Nós, a partir do uso de um enfoque prático da aplicação destas leis.

O segundo laboratório é o **Painel Elétrico CA**, que possibilita o estudo das associações em série, paralela e mista em redes de corrente alternada (CA). Segundo o seu Manual Técnico, disponibilizado em (SILVA, Juarez Bento da; LIMA; SIMÃO; CARLOS; PEREIRA, 2016), o experimento Painel Elétrico CA tem como objetivo:

Prover uma melhor visualização para estudos de associação em série, paralela e mista. Proporcionando a identificação de fenômenos físicos presentes no funcionamento de uma lâmpada incandescente.

A **Figura 10** ilustra a arquitetura dos laboratórios Painel Elétrico CC e Painel Elétrico CA desenvolvidos no projeto RELLE da UFSC.

Figura 10 – Arquitetura do Painel Elétrico CC e Painel Elétrico CA desenvolvidos no projeto RELLE da UFSC.



Fonte: (DA SILVA, Juarez Bento; DE LIMA; SIMÃO; CARLOS; PEREIRA, 2016; MELLOS; SILVA; LIMA; SIMÃO, 2016)

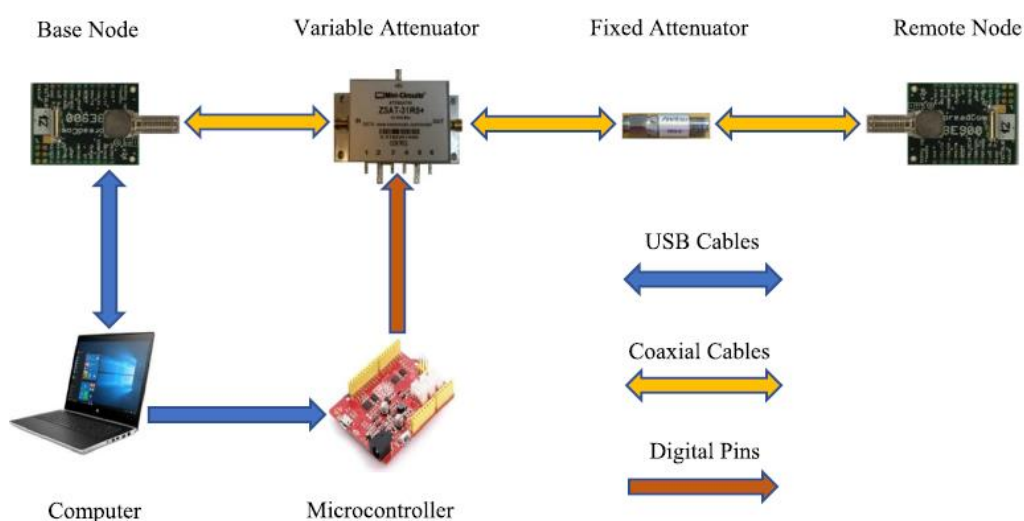
A arquitetura desse laboratório é organizada em 3 módulos: UI (Interface do Usuário, do inglês *User Interface*), *Web Server* (servidor *Web*) e Experimento remoto. Os módulos UI e *Web Server* são armazenados em um SBC Raspberry Pi. A interface do usuário é composta de uma página *Web*, arquivos de folha de estilos e *scripts* em JavaScript. Para acessar o experimento o usuário precisa de um navegador *Web* que irá carregar os arquivos necessários para disponibilizar o experimento para uso. A partir desse momento uma conexão via *WebSocket* é criada entre o navegador do usuário e o módulo *Web Server*. O módulo *Web Server* então é responsável por enviar os sinais digitais para a placa de controle do experimento.

O terceiro laboratório foi nomeado **NEWBen** e é uma continuação do trabalho proposto por (DELFORNO, 2020). Consiste em um ambiente de emulação de rede remoto, de baixo custo e de código aberto, capaz de reproduzir o comportamento de enlaces não guiados e de baixa potência sob diferentes configurações sendo aplicado no ensino de conceitos relacionados a comunicação sem fio investigando os aspectos fundamentais das comunicações sem fio utilizadas no desenvolvimento de aplicações IoT, como características espaciais e

temporais, distribuições e probabilidades (CHAVES; ASSUMPÇÃO; FERREIRA; CARDIERI; BRANQUINHO; FRUETT, 2021) (CHAVES, 2022).

A proposta inicial desse laboratório era composta por um PC conectado, via cabos USB, a um nó base e um microcontrolador. Um atenuador variável digital e atenuadores fixos são colocados entre os nós base e remotos, conectados via cabos coaxiais, eliminando as antenas, conforme ilustrado pela **Figura 11**.

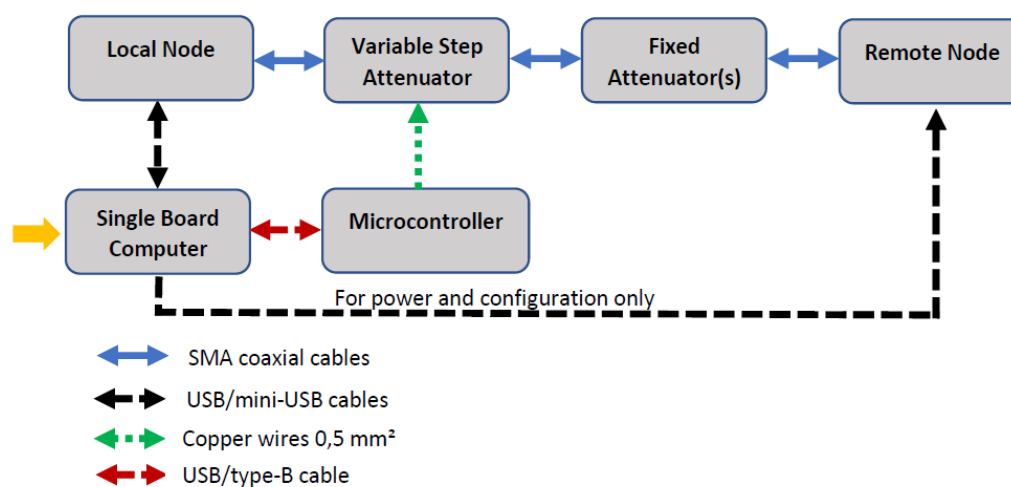
Figura 11 – Diagrama de bancada de emulação mostrando componentes e conexões.



Fonte: (CHAVES; ASSUMPÇÃO; FERREIRA; CARDIERI; BRANQUINHO; FRUETT, 2021)

Esse laboratório utilizado em um curso de pós-graduação e uma pesquisa no final de curso com os alunos indicou que o ambiente contribuiu para a compreensão dos princípios que regem a conectividade *over-the-air*, sugerindo que este ambiente remoto é uma ferramenta útil para investigações acadêmicas sobre as particularidades dos canais sem fio de baixa potência (CHAVES; ASSUMPÇÃO; FERREIRA; CARDIERI; BRANQUINHO; FRUETT, 2021). Na versão atual dessa plataforma, o PC foi substituído por um SBC Raspberry Pi 2, e a arquitetura representada na **Figura 11** foi alterada para uma nova estrutura ilustrada pela **Figura 12**.

Figura 12 – Diagrama de blocos da bancada incluindo os diferentes tipos de cabos usados em sua construção.



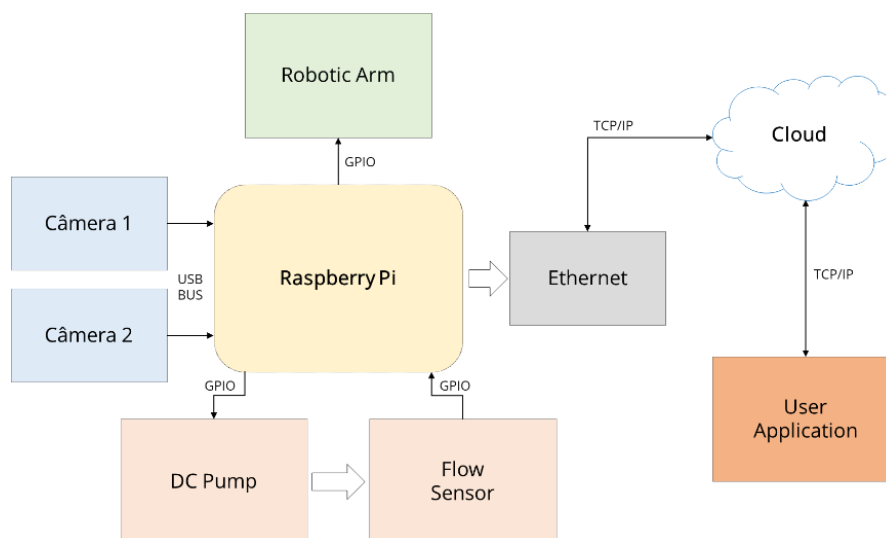
Fonte: (CHAVES, 2022)

3.2 PROPOSTAS DE LABORATÓRIOS REMOTOS

Em (SARAVANA MOHAN; KARTHIKEYAN; RAM KUMAR; RUPESH, 2019) é proposto um laboratório de acesso remoto em tempo real baseado na *Web* usando IoT. Esse laboratório remoto usa instrumentos reais como braço robótico, bomba CC, sensor de fluxo de água e câmeras. Segundo os autores, a arquitetura proposta (ver **Figura 13**):

- Ajuda a melhorar a segurança para o usuário trabalhar em ambientes perigosos ou no manuseio de substâncias químicas perigosas;
- Cria novas oportunidades de ensino a distância entre alunos e pesquisadores;
- Oferece uma solução econômica para multiusuários acessarem instrumentos reais via Internet.

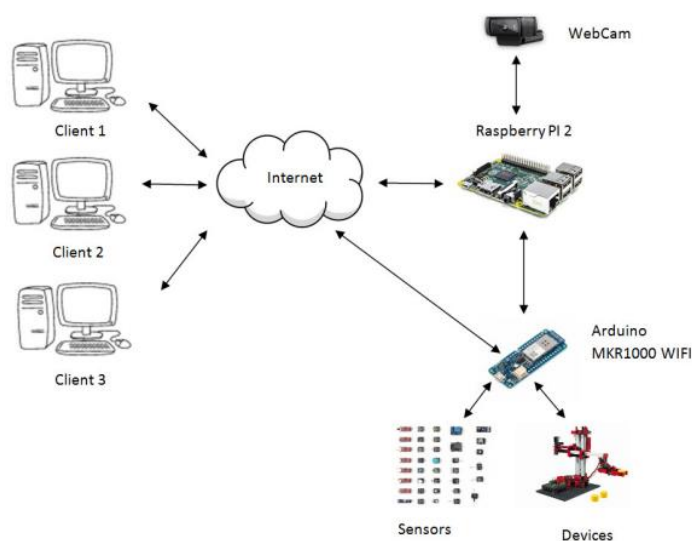
Figura 13 – *Layout* de *hardware* para laboratório remoto IoT composto de braço robótico, bomba CC, sensor de fluxo de água e câmeras.



Fonte: (SARAVANA MOHAN; KARTHIKEYAN; RAM KUMAR; RUPESH, 2019)

Em (FERNÁNDEZ-PACHECO; MARTIN; CASTRO, 2019) é proposto um laboratório remoto para dar suporte a ambientes de experimentação de aprendizado de IoT *online*. Em um SBM Arduino são conectados dispositivos como: cubo de LED (Diodo Emissor de Luz, do inglês *Light-Emitting Diode*), guindaste mecânico, sensor, etc. Um SBC Raspberry Pi é responsável pela geração do *streaming* de vídeo e pela hospedagem da página *Web* que controla o experimento. A **Figura 14** ilustra a arquitetura desse laboratório.

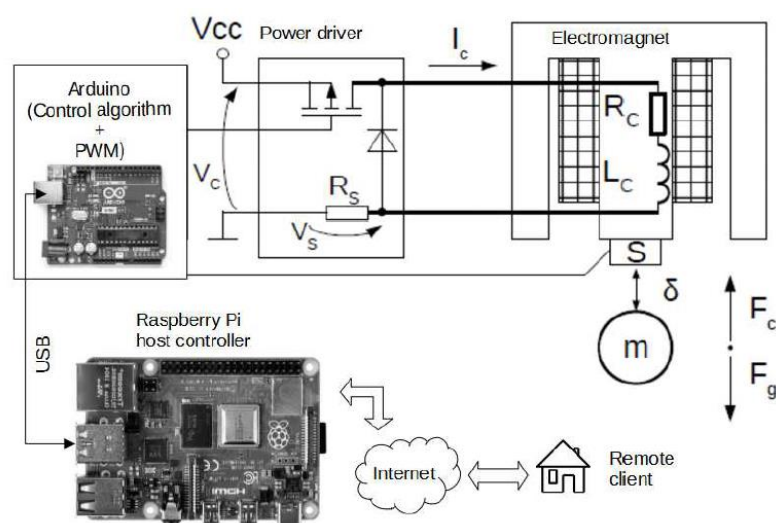
Figura 14 – Arquitetura do laboratório remoto.



Fonte: (FERNÁNDEZ-PACHECO; MARTIN; CASTRO, 2019)

Em (GORBOUNOV; CHEN, 2021) é apresentado um conceito de laboratório remoto de baixo custo para experimentos ciberfísicos, que permite o ensino de conceitos relacionados a levitação magnética, conforme ilustrado pela **Figura 15**. O componente principal é um controlador de *host*, nesse caso a escolha foi por um SBC Raspberry Pi, que executa um sistema operacional baseado em Linux, hospeda um servidor *Web* e que suporta comunicação via GPIO (Portas Programáveis de Entrada e Saída de Dados, do inglês *General Purpose Input/Output*). Os parâmetros do experimento físico são controlados em baixo nível por um dispositivo escravo, que pode ser outro SBC, um SBM ou um dispositivo lógico programável. A resposta do dispositivo físico pode ser observada tanto no terminal do usuário, quanto visualmente por meio de uma câmera (GORBOUNOV; CHEN, 2021).

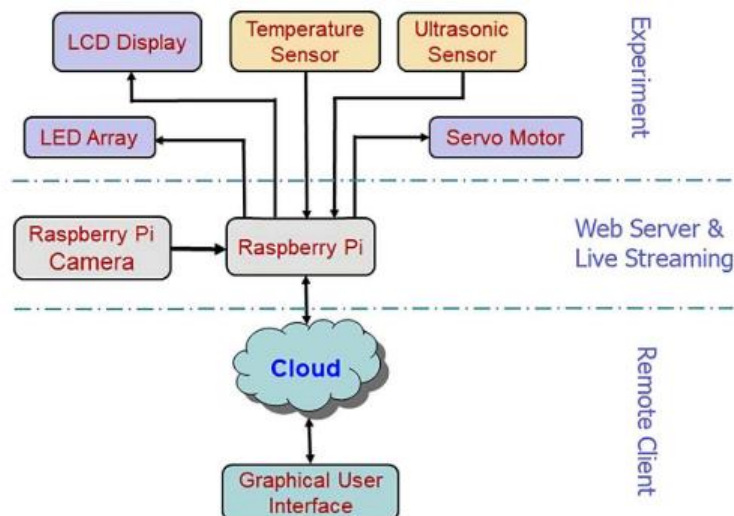
Figura 15 – Projeto de laboratório remoto para ensino de levitação magnética.



Fonte: (GORBOUNOV; CHEN, 2021)

Em (AZAD, 2021) é proposto um laboratório para o ensino de IoT. O objetivo do laboratório é fornecer aos alunos uma plataforma para aprender a desenvolver um sistema embarcado. Os estudantes podem manipular os códigos disponíveis e observar o comportamento de diversos dispositivos como, por exemplo, cubo de LED, servo motor, sensor ultrassônico e sensor de temperatura. A arquitetura do laboratório é ilustrada pela **Figura 16**.

Figura 16 – Arquitetura do laboratório remoto.

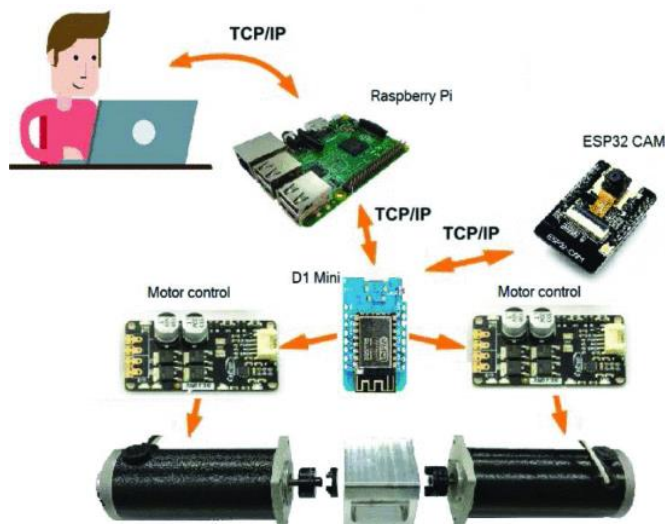


Fonte: (AZAD, 2021)

A arquitetura da **Figura 16** é dividida em três camadas: **Cliente remoto**, **Servidor Web** e **Vídeo em tempo real** e **Experimento**. O servidor *Web* está hospedado em um SBC Raspberry Pi. Uma câmera é conectada ao SBC para a geração do vídeo em tempo real. O SBC então faz o controle dos diversos experimentos.

Em (NENOV; EVSTATIEV; KADIROVA, 2022) é desenvolvido um modelo de laboratório para medição das características de carga de um motor elétrico, prevendo o estudo de um motor DC com excitação paralela e série. O modelo do laboratório, ilustrado na **Figura 17**, inclui o motor elétrico testado, um gerador CC através do qual o motor é excitado e um reostato para carregar o gerador. Um controlador DC com modulação PWM (Modulação por Largura de Pulso, do inglês *Pulse Width Modulation*) e dispositivos para medição das respectivas grandezas são utilizados para regular as correntes de excitação e a velocidade é medida por meio de um tacômetro DC.

Figura 17 - Arquitetura do sistema laboratorial.



Fonte: (NENOV; EVSTATIEV; KADIROVA, 2022)

3.3 ANÁLISE DOS TRABALHOS RELACIONADOS

Como mencionado anteriormente, a grande maioria dos laboratórios remotos de baixo custo é baseada na estrutura **Cliente – SBC – SBM ou FPGA – Experimento**, o que é recorrente nos trabalhos apresentados neste capítulo. Note que praticamente todas as arquiteturas apresentadas da **Figura 10** até a **Figura 17** empregam o SBC Raspberry Pi ou um PC, encarecendo o custo dos projetos quando comparados à proposta desenvolvida nessa dissertação, a qual será melhor detalhada nos capítulos seguintes.

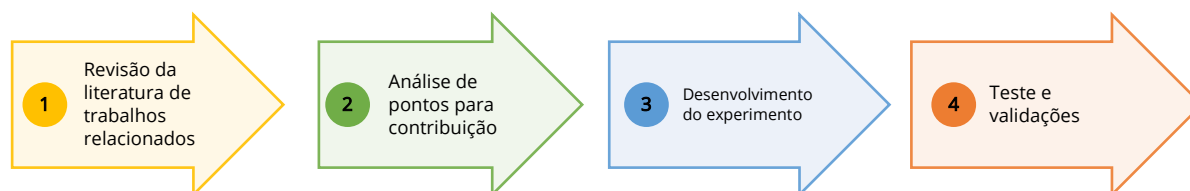
Com base na ampla revisão bibliográfica realizada e no estado da arte dos laboratórios remotos de baixo custo, considera-se que eliminar esse componente intermediário de controle do experimento (SBC), garantindo um desempenho similar da plataforma, ao mesmo tempo em que se reduz os gastos com *hardware* e simplifica-se a arquitetura, constitui uma das contribuições que justificam a realização deste trabalho.

4 METODOLOGIA

A pesquisa realizada nesse trabalho, sob o ponto de vista da sua natureza, é uma pesquisa aplicada. Segundo (PROVDANOV; FREITAS, 2013), esse tipo de pesquisa objetiva gerar conhecimentos para aplicação prática dirigidos à solução de problemas específicos.

Em relação ao objetivo geral e aos objetivos específicos, este trabalho pode ser classificado como uma pesquisa exploratória. Segundo (PROVDANOV; FREITAS, 2013), uma pesquisa exploratória tem como finalidade proporcionar mais informações sobre o assunto que se investiga, orientando a fixação dos objetivos e a formulação das hipóteses ou descobrindo um novo tipo de enfoque para o assunto de interesse, que é o caso dessa pesquisa. Em termos metodológicos, essa pesquisa segue os quatro passos descritos na **Figura 18**.

Figura 18 – Metodologia do trabalho.



Fonte: Autoria Própria

4.1 MATERIAIS E MÉTODOS

Nesta seção é apresentada uma descrição dos dispositivos de *hardware* e dos componentes de *software* necessários para o desenvolvimento da plataforma proposta.

4.1.1 *Microcontroladores de placa única*

Neste trabalho são utilizados dois SBMs: um NodeMCU ESP32 (para processamento, supervisão de dados e controle) e um NodeMCU ESP32-CAM (para

transmissão de vídeo em tempo real do experimento), os quais serão detalhados a seguir.

4.1.1.1 NodeMCU ESP32- WROOM-32

O NodeMCU ESP32 é equipado com o microcontrolador ESP32-WROOM-32, fabricado pela empresa Espressif Systems. Esse microcontrolador é um módulo genérico com Wi-Fi, Bluetooth e Bluetooth LE (Baixo Consumo, do inglês *Low Energy*) que visa uma ampla variedade de aplicações, desde redes de sensores de baixa potência até as tarefas mais exigentes, como codificação de voz, *streaming* de música e decodificação de MP3 (SYSTEMS ESPRESSIF, 2022).

O microcontrolador é equipado com dois núcleos de CPU (Unidade Central de Processamento, do inglês *Central Process Unit*) que podem ser controlados individualmente com uma frequência de *clock* ajustável de 80 MHz a 240 MHz. Possui também um coprocessador de baixo consumo que pode ser usado no lugar da CPU para economizar energia ao executar tarefas que não exigem muito poder de computação, como monitoramento de periféricos. Por ser um microcontrolador com dois núcleos de CPU, ele é mais rápido do que outros SBMs disponíveis no mercado, inclusive do que o Arduino (FOLTÝNEK; BABIUCH; ŠURÁNEK, 2019).

Dentre os atributos do microcontrolador escolhido que merecem destaque, pode-se citar seu amplo conjunto de periféricos, desde sensores de toque capacitivos, sensores Hall, interface de cartão de memória SD (do inglês *Secure Digital*), Ethernet, SPI (Interface Periférica Serial, do inglês *Serial Peripheral Interface*) de alta velocidade, UART (Receptor-Transmissor Assíncrono Universal, do inglês *Universal Asynchronous Receiver-Transmitter*), I2S (do inglês *Inter-IC Sound*) e I2C (do inglês *Inter Integrated Circuit*) (SYSTEMS ESPRESSIF, 2022).

A escolha pelo uso desse microcontrolador se deu pela versatilidade de suas características descritas previamente, mas também pela disponibilidade de 38 pinos (quantidade superior a de outros microcontroladores equiparáveis,

possibilitando a conexão com um maior número de dispositivos), além de possuir os seguintes itens: um servidor *Web* (necessário para disponibilizar a página *Web* para controle do experimento); um sistema de arquivos SPIFFS (do inglês *SPI Flash File System*), necessário para armazenar a página *Web*, *script* e folha de estilos; conexão nativa para uma rede Wi-Fi (facilitando a configuração de acesso à rede local) e apresentar um baixo custo de aquisição.

Outro ponto que pode ser destacado em relação a esse microcontrolador é que ele pode ser programado utilizando a IDE (Ambiente de Desenvolvimento Integrado, do inglês *Integrated Development Environment*) do Arduino, que é uma das plataformas *open-source* (código aberto) de SBM mais utilizadas e conhecidas pelos desenvolvedores de projetos, contribuindo com o uso de recursos, que inicialmente foram desenvolvidos para Arduino mas que também são válidos para outras placas de SBM como a própria ESP32.

4.1.1.2 NodeMCU ESP32-CAM

O NodeMCU ESP32-CAM é baseado no NodeMCU ESP32 e por isso herda diversas configurações e recursos como, por exemplo, Wi-Fi, *Bluetooth*, interface de cartão de memória SD, processador *dual core*, etc. O NodeMCU ESP32-CAM é um SBM específico para a geração de vídeo, trabalhando sempre em conjunto com uma câmera.

Nessa pesquisa é utilizado um NodeMCU ESP32-CAM com uma câmera OV2640. Esta câmera opera com um sensor CMOS (Semicondutor de Óxido Metálico Complementar, do inglês *Complementary Metal-Oxide-Semiconductor*) o qual suporta imagens com a qualidade UXGA (Matriz de Gráficos Ultra Estendidos, do inglês *Ultra-Extended Graphics Array*) e resolução máxima de 1600 x 1200, equivalente a 2 Megapixels, com potência de 140mW e baixo consumo de energia e corrente. O controle é feito via interface SCCB (Barramento Serial de Controle de Câmera, do inglês *Serial Camera Control Bus*), semelhante à I2C, ou seja, requer apenas duas linhas de comunicação: uma para os dados e outra para o *clock*. É

válido destacar que câmeras que usam sensores CMOS são de baixo custo, eficientes em termos de energia e têm componentes menores; portanto são mais apropriadas para uso em dispositivos IoT (ONER, 2021).

O ESP32-CAM não possui conexão USB para a comunicação com a IDE de desenvolvimento sendo, portanto, necessário utilizar um adaptador USB para essa comunicação. Nessa pesquisa foi empregado o adaptador ESP32-CAM-MB, responsável simultaneamente pela comunicação com a IDE e alimentação do SBM.

4.1.2 Protocolo I2C

O protocolo I2C descreve o funcionamento de um barramento serial de baixa velocidade para comunicação eficiente entre dispositivos. Foi desenvolvido em 1982 pela Philips Semiconductors (atualmente NXP Semiconductors). O barramento I2C é composto por duas linhas, uma bidirecional para transferência de dados (nomeada SDA, do inglês *Serial Data*) e outra para sinal de sincronização entre os dispositivos (nomeada SCL, do inglês *Serial Clock*). É um protocolo muito empregado em dispositivos IoT, pois grande parte dos SBMs possui suporte para sua utilização (CHU, 2018).

O barramento I2C suporta diversos dispositivos conectados que podem transmitir e receber dados entre si. Cada dispositivo recebe um endereço de identificação para permitir o acesso via barramento serial. Durante a operação, um dos dispositivos conectados ao barramento funciona como mestre e os demais como escravos. O mestre gera um sinal de *clock*, para sincronização, na linha SCL e coordena a transferência de dados pela linha SDA. Um escravo lê o barramento e responde quando seu endereço de identificação é acessado. Finalmente, o mestre e o escravo designado trocam dados através da linha SDA (CHU, 2018).

Neste trabalho, o protocolo de comunicação serial I2C é usado para controlar os medidores de tensão/corrente e os *displays*.

4.1.2.1 Multiplexador I2C

Na eletrônica um multiplexador é um dispositivo com um circuito que seleciona um ou mais sinais de entrada e os encaminha para uma saída específica, sendo frequentemente utilizado para ampliar o número de terminais associados a uma única porta, ou seja, permitir que mais dispositivos sejam controlados por um mesmo sistema.

Neste trabalho, conforme será discutido a seguir, os sinais elétricos do circuito a ser controlado via experimento remoto são monitorados por quatro medidores de corrente/tensão adequadamente posicionados e os valores medidos são respectivamente exibidos em quatro *displays*. Tanto os *displays* quando os medidores de corrente/tensão podem se comunicar com o SBM via protocolo I2C, ou seja, são oito dispositivos compartilhando o mesmo barramento I2C, constituído por uma linha de dados SDA e outra de *clock* SCL.

Para conectar esses oito dispositivos nessas duas linhas há duas opções: (1) garantir que cada um desses oito dispositivos tenha um endereço específico que deve ser indicado para o microcontrolador durante o processo de envio e recebimento de dados; ou (2) usar um multiplexador com uma porta (canal) específico para cada dispositivo que deve ser indicado para o microcontrolador durante o processo de envio e recebimento de dados.

Para usar a primeira opção seria necessário alterar fisicamente (realizar uma conexão com solda) o endereço de cada dispositivo, pois é comum virem de fábrica com um mesmo endereço padrão. Essa alternativa, além de mais trabalhosa pode causar danos físicos ao dispositivo, dificultando sua troca ou substituição.

Por outro lado, a segunda opção, que foi a escolhida neste trabalho, requer apenas o uso de um multiplexador para conectar cada dispositivo em um canal e indicar essa associação canal/dispositivo ao microcontrolador. Nessa opção não é necessária a alteração de endereços dos dispositivos, visto que eles são acessados

via um canal do multiplexador. Diferentemente da alternativa anterior, não há grandes riscos de danificar o componente, o que facilita sua manutenção ou substituição.

Portanto, para facilitar o endereçamento e o acesso aos componentes foi utilizado um multiplexador I2C modelo TCA9548A de oito canais (de baixo custo e grande disponibilidade no mercado). Esse multiplexador possui oito chaves de conversão bidirecionais que são controladas através do barramento I2C, de forma que qualquer canal individual (ou combinação de canais) pode ser selecionado (TEXAS INSTRUMENTS, 2019).

4.1.3 Medidor de corrente e tensão

Como esta pesquisa propõe o desenvolvimento de um laboratório para o ensino de conceitos relacionados à eletricidade básica, é obrigatório o uso de algum dispositivo para medir corrente e tensão dentro de um determinado circuito. No mercado existem vários dispositivos que medem essas grandezas. Neste trabalho são utilizados quatro medidores de corrente/tensão, modelo INA219, em pontos específicos do circuito elétrico.

O INA219 é um derivador (*shunt*) de corrente e monitor de potência com interface compatível com I2C. Para ler a tensão, ele monitora a queda de tensão no *shunt* e a tensão de alimentação do barramento, com tempos de conversão e filtragem programáveis. Um valor de calibração programável, combinado com um multiplicador interno, permite leituras diretas de corrente em amperes (TEXAS INSTRUMENTS, 2015).

A escolha desse medidor se deu pelos seguintes motivos: baixo custo, capacidade de calibração, uso do barramento I2C, opera com tensões de 0 a 26V (a tensão máxima no circuito é de 12V), mede corrente nos dois sentidos e tem erro máximo de 0,5% nas medições.

4.1.4 *Display OLED*²

Para exibir as leituras de tensão e corrente na bancada experimental são utilizados quatro *displays* com telas OLED (Diodo Orgânico Emissor de Luz, do inglês *Organic Light-Emitting Diode*). Foi escolhido o modelo SSD1306 com 128 linhas e 64 colunas, o qual permite comunicação com o barramento I2C, apresenta baixo consumo de energia, alta taxa de atualização e baixo custo (SOLOMON SYSTECH, 2008).

A tecnologia OLED é caracterizada pelo menor consumo de energia e qualidade superior quando comparada a outras tecnologias de exibição gráfica. Esse tipo de tela também consegue exibir a cor preta mais profundamente e as demais cores de forma mais realista. Esse é um atributo importante porque a tela será filmada por uma câmera e a qualidade da imagem exibida no *display* OLED pode influenciar na percepção de cores de quem assiste o *streaming* de vídeo.

4.1.5 *Módulo de relés*

Um relé é um comutador mecânico operado eletricamente (eletromecânico) que funciona como um interruptor em um circuito. Ao receber um sinal de controle, o relé permite ou não a passagem de corrente em um determinado circuito de carga (BRAGA, 2012). Nesta pesquisa é utilizado um módulo com oito relés. Esse módulo pode operar com tensões de até 30V em CC e 250V em CA, com níveis de corrente de até 10A. Seguindo sua descrição, a função desses relés no circuito elétrico a ser controlado no experimento é funcionar como chaves para permitir a montagem de diferentes associações de lâmpadas em série ou paralelo, permitindo a análise de diferentes configurações de circuitos.

² O diodo orgânico emissor de luz (sigla OLED, em inglês: *Organic Light-Emitting Diode*) é um LED em que a camada de emissão eletroluminescente é um filme orgânico que emite luz em resposta a uma corrente elétrica.

4.1.6 Linguagem de marcação

A linguagem de marcação HTML (Linguagem de Marcação de Hipertexto, do inglês *Hypertext Markup Language*) descreve a estrutura de páginas *Web*. É a principal tecnologia para a construção de páginas *Web*, tornando-se um padrão para desenvolvimento de *sítes*. A linguagem permite a inserção de diversos tipos de informações como, por exemplo, tabelas, vídeos, parágrafos, títulos, etc, em uma página *Web* (TABARÉS, 2021). Neste trabalho foi utilizado HTML, na versão 5, para o desenvolvimento da página *Web* que permite controlar o experimento.

4.1.7 Linguagem de script

O JavaScript é uma linguagem de *script* muito utilizada no desenvolvimento de páginas *Web*. A ampla maioria dos sites usa JavaScript e todos os navegadores modernos incluem interpretadores JavaScript, tornando-a uma das linguagens de programação mais utilizadas. JavaScript faz parte da tríade de tecnologias que são base da *Web* atual: HTML, para especificar a estrutura de páginas *Web*; CSS (Folhas de Estilo em Cascata, do inglês *Cascading Style Sheets*), para especificar a apresentação dessas páginas; e JavaScript, para especificar o comportamento delas (FLANAGAN, 2012).

Neste trabalho são escritos *scripts* em JavaScript com o intuito de: gerenciar as conexões *WebSockets*, alterar as características gráficas do gêmeo digital (alterar as cores dos ramos energizados associados aos circuitos equivalentes, alterar a cor que representa a luminosidade das lâmpadas, cores e estados das chaves, etc), receber os valores dos medidores de corrente e tensão e inseri-los na página *Web*, gerenciar o acesso dos usuários (permitindo um único usuário por vez) e controlar o tempo de uso do experimento.

Essa linguagem de programação não foi projetada para rodar em servidores (como linguagens *server-side*), mas no navegador do usuário (linguagem *client-side*), ou seja, seu uso não compromete o desempenho do servidor.

4.1.8 Folha de estilos

O CSS é uma linguagem projetada para descrever a aparência de documentos escritos em uma linguagem de marcação, como HTML. Com CSS o projetista pode controlar a cor do texto, o estilo das fontes, o espaçamento entre os parágrafos, como as colunas são dimensionadas e dispostas, quais imagens ou cores de fundo são usadas e uma variedade de outros efeitos visuais (POUNCEY; YORK, 2011). Nesta pesquisa a versão 3 do CSS é utilizada para estilizar a página *Web* do experimento.

4.1.9 API da Web

API (Interface de Programação de Aplicação, do inglês *Application Programming Interface*) é um conjunto de regras e padrões usados para facilitar a troca de dados e informações entre sistemas diferentes. Neste trabalho, utiliza-se uma API para programar o protocolo *WebSocket*.

O *WebSocket* é um protocolo de comunicação, da pilha de protocolos TCP/IP (protocolo de controle de transmissão/protocolo da internet, do inglês *Transmission Control Protocol/Internet Protocol*), projetado para navegadores e servidores da *Web*, mas, diferentemente do HTTP (protocolo de transferência de hipertexto, do inglês *Hypertext Transfer Protocol*), protocolo padrão para o acesso de páginas na *Web*, o *WebSocket* fornece comunicação *full-duplex* em uma única conexão TCP (XIAO, 2018). O *WebSocket* possibilita mais interação entre um navegador e um servidor *Web*, permite a transferência de dados em tempo real e fluxos de mensagens e é mais eficiente, em relação a outros métodos de transmissão, na transmissão de dados em tempo real (MITROVIC; DORDEVIC; VELJKOVIC; DANKOVIC, 2021). Nesta pesquisa o *WebSocket* é utilizado para troca de dados em tempo real entre a página *Web* e o SBM ESP32.

4.1.10 IDE Arduino

O IDE (Ambiente de Desenvolvimento Integrado, do inglês *Integrated Development Environment*) do Arduino foi utilizado para escrever, compilar e

enviar os códigos-fonte para os SBMs ESP32 e ESP32-CAM. O IDE Arduino foi inicialmente desenvolvido para trabalhar com SBMs do tipo Arduino, mas atualmente também suporta outros tipos de SBMs, como é o caso dos SBMs da família ESP.

É um *software* que está em constante atualização, possuindo diversas versões (nessa pesquisa foi empregada a versão 1.8), de fácil instalação, configuração e uso. Sua documentação está disponível em <https://www.arduino.cc/reference/pt/>.

4.1.11 Linguagem de programação C++

Para programar os microcontroladores ESP32 e ESP32-CAM foi utilizada a linguagem de programação C++ padrão da IDE Arduino. Essa linguagem foi originalmente desenvolvida por Bjarne Stroustrup em 1983 no Bell Labs e recebeu seu nome por originalmente significar C com classes, ou seja, apesar de baseada na linguagem C, novas características foram adicionadas como funções virtuais, sobrecarga de operadores, herança múltipla, gabaritos e tratamento de exceções.

4.1.12 Bibliotecas do ESP32 e ESP32-CAM

Uma das vantagens do uso da IDE Arduino é a quantidade de bibliotecas disponíveis e o vasto suporte *online*. Neste trabalho foram utilizadas diversas bibliotecas, todas gratuitas, na programação dos SBMs ESP32 e ESP32-CAM. Suas versões e os seus repositórios estão elencados na **Tabela 1**.

Tabela 1 – Bibliotecas utilizadas nos SBMs.

Biblioteca	Versão	SBM	Repositório
Adafruit_GFX	1.10.12	ESP32	https://github.com/adafruit/Adafruit-GFX-Library
Adafruit_INA219	1.1.1	ESP32	https://github.com/adafruit/Adafruit_INA219
Adafruit_SSD1306	2.5.1	ESP32	https://github.com/adafruit/Adafruit_SSD1306
CameraPins	1.0	ESP32-CAM	https://github.com/espressif/arduino-esp32/tree/master/libraries/ESP32/examples/Camera
EspCamera	2.0.0	ESP32-CAM	https://github.com/espressif/esp32-camera
FS	1.0	ESP32	https://github.com/espressif/arduino-esp32/tree/master/libraries/FS
SPIFFS	1.0	ESP32	https://github.com/espressif/arduino-esp32/tree/master/libraries/SPIFFS

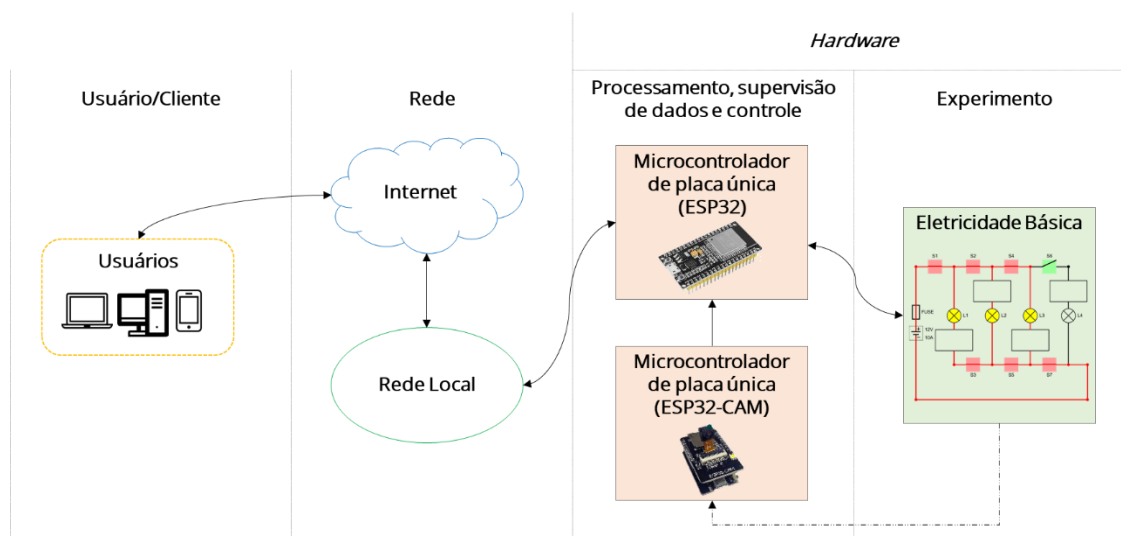
WebServer	1.0	ESP32	https://github.com/espressif/arduino-esp32/tree/master/libraries/WebServer
WebSocketsServer	2.3.6	ESP32	https://github.com/Links2004/arduinoWebSockets
WiFi	1.0	ESP32 e ESP32CAM	https://github.com/espressif/arduino-esp32/tree/master/libraries/WiFi
WiFiClient	1.0	ESP32 e ESP32CAM	https://github.com/espressif/arduino-esp32/tree/master/libraries/WiFi
Wire	1.0.1	ESP32	https://github.com/espressif/arduino-esp32/tree/master/libraries/Wire

Fonte: Autoria Própria

5 PLATAFORMA PROPOSTA

Para implementar o laboratório remoto propõe-se uma nova arquitetura, distinta das listadas em (KALÚZ; GARCÍA-ZUBÍA; FIKAR; ČIRKA, 2015), a qual consiste na simplificação do modelo **Cliente** → **SBC** → **SBM ou FPGA** → **Experimento** para o esquema **Cliente** → **SBMs** → **Experimento**, ilustrado pela **Figura 19**.

Figura 19 – Arquitetura simplificada proposta neste trabalho.



Fonte: Autoria Própria

Na arquitetura proposta, um SBM é responsável pelo processamento e supervisão de dados acumulando também a função de unidade de controle, enquanto outro SBM, hierarquicamente inferior, é responsável pela geração do vídeo em tempo real. Com isso, elimina-se a necessidade de um SBC, reduzindo os custos da plataforma e tornando-a mais acessível para instituições que pretendam replicar o projeto.

Essa arquitetura simplificada é possível por dois principais motivos:

1. O SBM utilizado para o processamento, supervisão de dados e unidade de controle é um ESP32. Esse microcontrolador possui nativamente acesso à rede Wi-Fi, sistema de arquivos e um servidor *Web*.

2. O SBM utilizado para a geração do vídeo em tempo real é um ESP32-CAM que também possui nativamente acesso a rede Wi-Fi e gera a transmissão (*streaming*) de vídeo diretamente no microcontrolador.

5.1 CONSIDERAÇÕES SOBRE A ARQUITETURA PROPOSTA

A arquitetura proposta pode ser expandida e adaptada para diferentes usos e situações, conforme destacado em amarelo na **Figura 20**³. A seguir são listadas algumas observações sobre a arquitetura:

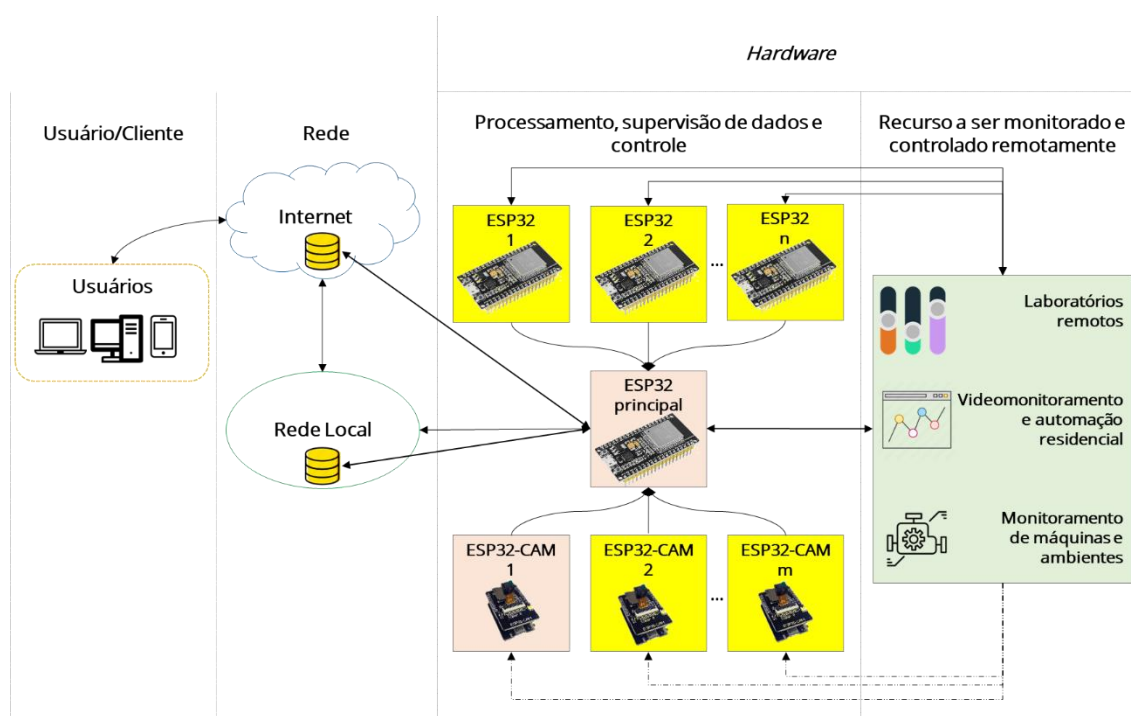
- É possível aproveitá-la para uso em laboratórios com outros tipos de experimentos, que não a análise de um circuito elétrico;
- Permite a operação simultânea de diversas câmeras (pois cada ESP32-CAM pode trabalhar de maneira autônoma) o que facilitaria no seu emprego em monitoramento e controle centralizado de ambientes ou equipamentos com unidades esparsas;
- Apesar da aparente limitação de terminais disponíveis em um único ESP32, possibilita trabalhar com mais de um SBM, visto que esse dispositivo pode ser configurado como um *Access Point* ou como uma *Station* em uma rede Wi-Fi. Nesse sentido é válido destacar que SBMs da família ESP podem operar com o protocolo de comunicação ESP-NOW, o qual é bidirecional e possibilita a troca de dados entre vários ESPs de forma rápida e segura;
- Foi desenvolvida com o intuito de ser aplicada em outras áreas como como no contexto de automação industrial na Indústria 4.0 para, por exemplo, realizar o monitoramento remoto do estado de máquinas,

³ Observe que a diferença da **Figura 20** para a arquitetura originalmente proposta (ver **Figura 19**), constituída de um único SBM para controle e outro para *streaming*, é que nessa adaptação admite-se o uso de n ESP32s e m ESP32-CAMs conectados a um ESP32 central configurado como *Access Point*.

permitindo não somente a visibilidade em tempo real, por meio de interfaces gráficas intuitivas chamadas de gêmeos digitais, como também possibilitando intervenção imediata, o que contribui para que um único técnico ou engenheiro de manutenção tenha acesso a diferentes plantas industriais, reduzindo os custos operacionais e aumentando a velocidade de detecção e correção de falhas e melhorando a eficiência das operações a longo prazo;

- Permite configurar os SBMs para acessar banco de dados armazenados em nuvem ou localmente, abrindo a possibilidade de armazenamento de dados para análise futura.

Figura 20 – Possibilidades de adaptações e expansão da arquitetura proposta.

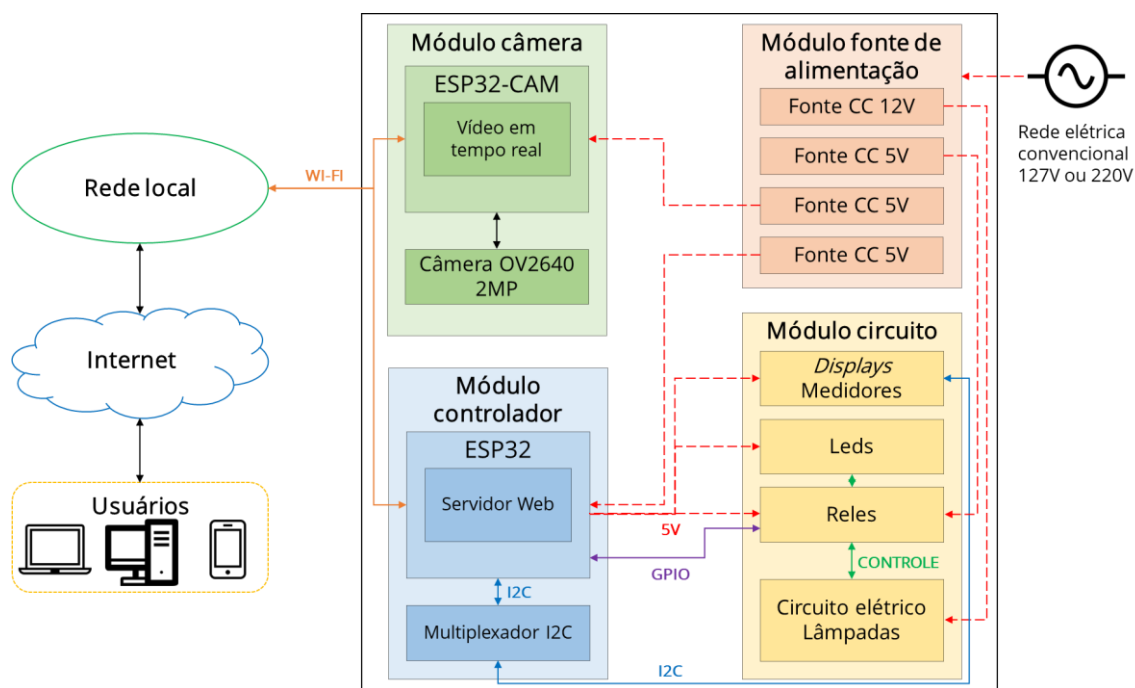


Fonte: Autoria Própria

5.2 ORGANIZAÇÃO DA PLATAFORMA

A plataforma experimental está disposta em 4 módulos: **módulo câmera**, **módulo fonte de alimentação**, **módulo controlador** e **módulo circuito**. A Figura 21 ilustra a organização e a integração de cada um desses módulos, os quais serão melhor detalhados a seguir.

Figura 21 – Organização da plataforma proposta em módulos.



Fonte: Autoria Própria

5.2.1 Módulo câmera

O módulo câmera é responsável pela geração do *streaming* de vídeo. Ele é composto por:

- **Um SBM modelo ESP32-CAM** – Esse SBM se conecta à rede local através de uma conexão Wi-Fi e é alimentado por uma das fontes de 5V disponíveis no módulo fonte de alimentação. Assim que é ligado, gera automaticamente um *streaming* de vídeo, cujas configurações de resolução e qualidade de imagem estão armazenadas em sua memória *flash*, portanto não precisa ser reconfigurado toda vez que é desligado. O *streaming* de vídeo é acessado por meio de uma URL (Localizador Uniforme de Recursos, do inglês *Uniform Resource Locator*) que é inserida na página *Web* do experimento. Suporta diversos modelos de câmeras.
- **Uma câmera de 2 Megapixels modelo OV2640** – Essa câmera é conectada diretamente ao ESP32-CAM. Apesar de permitir gravação de imagens na proporção 4:3 com resoluções de 160x120 até

1600x1200, neste projeto está configurada para operar com resolução máxima.

5.2.2 Módulo fonte de alimentação

O módulo fonte de alimentação é responsável pelo suprimento de energia elétrica do módulo controlador, do módulo câmera e do módulo circuito. Ele é composto por:

- **Uma fonte de 12V e 10A** – Responsável pela alimentação das lâmpadas do circuito elétrico controlado durante o experimento remoto.
- **Três fontes de 5V e 2A** – A primeira é responsável pela alimentação das bobinas dos relés que operam como chaves eletrônicas no circuito elétrico experimental, permitindo ou impedindo a passagem de corrente elétrica e ajudando a montar diferentes configurações de malhas a serem analisadas pelo usuário. A segunda alimenta o SBM ESP32-CAM e a terceira o ESP32.

Apesar de não ser necessário, optou-se por trabalhar com três fontes de 5V para isolar a alimentação das bobinas dos relés da alimentação dos SBMs, evitando que alguma queda de tensão indevida, decorrente do acionamento dos relés, ocasione o travamento dos dispositivos responsáveis pelo processamento do experimento e também para que o ESP32-CAM possuísse uma fonte exclusiva, aumentando seu desempenho e facilitando sua manutenção.

5.2.3 Módulo controlador

O módulo controlador é responsável por controlar e gerenciar todo o experimento, sendo composto por:

- **Um SBM modelo ESP32 com 38 pinos** – Esse SBM é responsável por realizar a leitura ou escrita nos seguintes dispositivos usados no experimento: medidores de corrente e tensão, *displays* para visualização dos valores de corrente e tensão medidos em diferentes

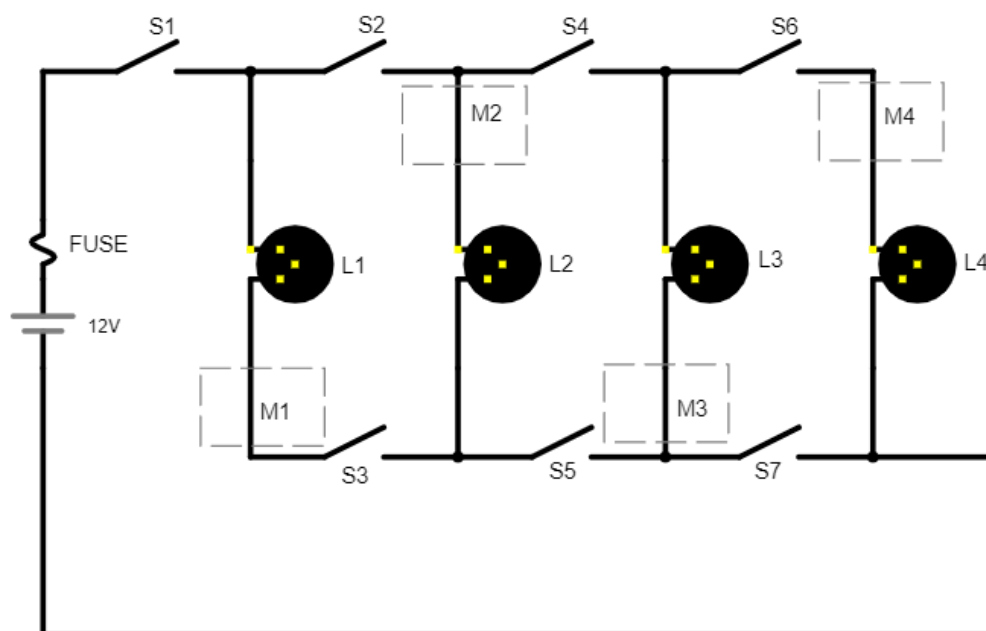
nós do circuito elétrico e o módulo dos relés que atuam como chaves no circuito elétrico. Controla, via GPIO, cada canal do módulo de relés e os LEDs indicadores do estado de cada uma das chaves. Se conecta à rede local através de uma conexão Wi-Fi, tem uma memória *flash* de 4MB que é dividida, por padrão, da seguinte forma: 1,3MB para o *firmware* (conjunto de instruções operacionais embutidas pelo fabricante no *hardware* do microcontrolador), 1,2MB para armazenar o código fonte (conjunto de instruções escritas de forma organizada por um programador para executar funções específicas) e 1,5MB para o sistema de arquivo SPIFFS. Possui um servidor *Web* capaz de executar páginas *Webs*, folhas de estilos e *scripts*. Se comunica com o multiplexador (descrito a seguir) através do protocolo I2C.

- **Um multiplexador I2C modelo TCA9548A de 8 canais** – Recebe os dados dos medidores de corrente e tensão posicionados em nós específicos do circuito elétrico e os encaminha para o ESP32 via protocolo de comunicação serial I2C. Recebe do ESP32 os dados dos medidores, após tratamento, e os envia para exibição nos *displays*, também via I2C.

5.2.4 Módulo circuito

O módulo circuito é o objeto de estudo do usuário do laboratório remoto. Ele foi concebido com o intuito de prover diferentes montagens em série e paralelo de quatro lâmpadas pelo acionamento de sete chaves posicionadas convenientemente, como ilustrado na **Figura 22**.

Figura 22 – Esquema elétrico do módulo circuito.



LEGENDA

L1, L2, L3 e L4 = lâmpadas de 12V e 5W

M1, M2, M3 e M4 = medidores de corrente (A) no ramo e tensão (V) em relação ao terra

C1, C2, C3, C4, C5, C6 e C7 = chaves (relês) de acionamento para a construção do circuito desejado

Fonte: Autoria Própria

Para proporcionar um aprendizado adequado de medição de corrente e tensão, baseado nas Leis de Kirchhoff das Tensões (LKT) e das Correntes (LKC), que segundo (CHARLES K. ALEXANDER; SADIKU, 2013) são importantes ferramentas para análise de circuitos elétricos, são acrescentados quatro medidores de corrente e tensão colocados em posições estratégicas do circuito.

Nesse ponto, é válido destacar que embora os laboratórios remotos disponibilizados pelo RELLE da UFSC, a saber: **Painel Elétrico CC** (MELLOS; DA SILVA; DE LIMA; SIMÃO, 2016) e **Painel Elétrico CA** (DA SILVA, Juarez Bento; DE LIMA; SIMÃO; CARLOS; PEREIRA, 2016) sejam a referência para o desenvolvimento do experimento controlado pela plataforma proposta neste trabalho, há algumas diferenças as quais são listadas a seguir:

- A plataforma proposta também controla um circuito CC assim como o **Painel Elétrico CC** do RELLE, no entanto, esse último usa resistores,

enquanto o experimento proposto utiliza lâmpadas automotivas visando um aprendizado mais lúdico dos conceitos ensinados;

- O **Painel Elétrico CA** do RELLE também faz o uso de lâmpadas, mas são lâmpadas incandescentes que atualmente são difíceis de encontrar, a alimentação do **Painel Elétrico CA** é em corrente alternada (diferente da plataforma proposta que opera em corrente contínua) e, finalmente, a plataforma proposta disponibiliza as medições de correntes nos ramos e tensões nos nós, facilitando a compreensão das LKT e LKC pelos alunos, enquanto o **Painel Elétrico CA** do RELLE não apresenta essa funcionalidade;
- A plataforma proposta possibilita a criação de uma variedade maior de arranjos elétricos, ou seja, um número maior de combinações de conexões em série/paralelo e circuitos para estudo.

Roteiros de aprendizagem, como o apresentado no **Apêndice A**, especialmente desenvolvidos para estudantes do ensino médio e/ou técnico, disponibilizados na página de entrada do experimento, podem orientar tutores e alunos em como usar esse circuito.

O estudo desse circuito pode ser aplicado com suporte complementar para o ensino das Leis de Kirchhoff em situações como: prática complementar de atividades feitas em sala de aula, atividades extraclasse, atividades de treinamentos, etc.

Os componentes do módulo circuito são descritos a seguir:

- **Quatro *Displays* OLED 128x64** – Recebem os dados do multiplexador TCA9548A, via I2C, e os exibem. Cada *display* está associado a um único medidor de corrente e tensão, auxiliando o usuário na leitura dessas variáveis.
- **Quatro medidores de corrente e tensão INA219** – Realizam a medição de corrente nos ramos do circuito elétrico em que foram posicionados

e calculam a diferença de potencial do nó em relação ao terra. Envia os dados coletados, via I2C, para o multiplexador TCA9548A, e por sua vez ao ESP32 controlador.

- **Dois módulos relé de 8 canais 5V e 10A** – Um módulo é responsável pelo chaveamento dos canais (correspondente a cada uma das chaves do circuito) sendo controlado via GPIO pelo ESP32. Esse módulo tem as bobinas alimentadas por uma das fontes de 5V descritas no módulo fonte de alimentação. O outro módulo é responsável pelo acionamento dos LEDs, sendo também controlado via GPIO pelo ESP32, mas tendo as bobinas alimentadas diretamente pelo ESP32.
- **Quatro lâmpadas automotivas bipolares de 12V e 5W** – Indicam visualmente a intensidade e o caminho percorrido pela corrente elétrica para as diferentes malhas de circuito montadas de acordo com o fechamento ou abertura das chaves presentes no gêmeo digital (que por sua vez controlam as chaves eletrônicas - relés). São ligadas (em série ou paralelo) à fonte de alimentação de 12V.
- **Sete módulos de LEDs RGB (Vermelho Verde Azul, do inglês Red Green Blue)** – Funcionam como uma indicação visual de abertura ou fechamento das chaves do circuito elétrico. Assim, quando uma chave está acionada, o LED vermelho correspondente é ligado indicando perigo (circuito energizado). Por outro lado, quando a chave é desligada o LED verde correspondente é ligado indicando que a passagem de corrente por aquele ramo foi interrompida. Cada módulo está associado a uma única chave, sendo controlado via GPIO pelo ESP32. Os LEDs azuis não são utilizados.

5.3 FUNCIONAMENTO DA PLATAFORMA

Para acessar remotamente o experimento, a URL do ESP32 deve ser digitada em um navegador *Web*. O servidor *Web* presente no ESP32 exibirá a página de entrada do experimento com tutoriais, roteiros de aprendizagem e outros materiais de apoio.

A princípio, não há necessidade de *login*/senha para ingressar na página de entrada, a qual permite o acesso simultâneo de vários usuários. Em seguida, basta clicar no botão ACESSAR para obter o controle do experimento de forma exclusiva durante um tempo pré-definido, impedindo a admissão de outros usuários nesse mesmo intervalo de tempo.

Os requisitos de segurança de acesso a plataforma podem ser implementados levando em consideração a política de segurança da instituição de ensino que disponibilizará o experimento remoto. Com isso cada instituição tem autonomia para implementar os requisitos de segurança que julgar necessário.

5.4 GÊMEO DIGITAL

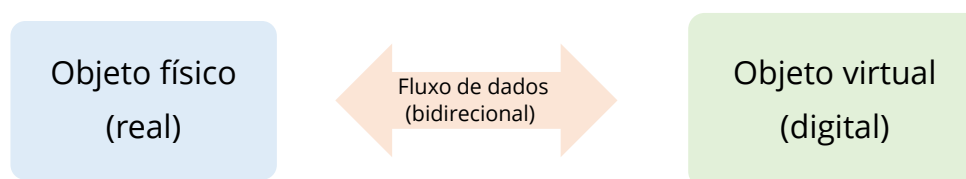
O termo gêmeo digital refere-se ao desenvolvimento de modelos virtuais de processos, produtos ou serviços que permitam a simulação do comportamento do sistema, compreensão e previsão das características de desempenho da contraparte física que ele representa (WUTTKE; HENKE; HUTSCHENREUTER, 2020). Os autores de (SULEMA; DYCHKA; SULEMA, 2019) consideram que o conceito de gêmeo digital pode ser usado na formação futura de engenheiros.

Segundo (KRITZINGER; KARNER; TRAAR; HENJES; SIHN, 2018) existem várias definições sobre o conceito de gêmeo digital, mas em linhas gerais, há um entendimento de que os gêmeos digitais são contrapartes digitais de objetos físicos com os dados fluindo de forma bidirecional do objeto físico para o objeto digital e vice-versa. Esses autores também propõem uma classificação dos gêmeos digitais em três subcategorias, de acordo com seu nível de integração de dados:

- **Modelo digital** – é uma representação digital de um objeto físico existente ou planejado que não usa nenhuma forma de troca automatizada de dados entre o objeto físico e o objeto digital. Há o uso de dados reais, mas a troca de dados é feita de forma manual e, portanto, uma mudança no estado do objeto físico não tem efeito direto no objeto digital e vice-versa.;
- **Sombra digital** – é uma representação digital conectada ao objeto real, mas o fluxo de dados é somente em um único sentido, do objeto físico para o objeto virtual;
- **Gêmeo digital** – é uma representação digital de um objeto físico onde o fluxo de dados ocorre em ambos os sentidos, do objeto físico para o virtual e do virtual para o físico. Uma mudança no estado do objeto físico leva diretamente a uma mudança no estado do objeto digital e vice-versa.

Com base nessas definições elencadas anteriormente, um gêmeo digital é composto de três partes principais: objeto físico (real), objeto virtual (digital) e fluxo de dados bidirecional, conforme ilustrado na **Figura 23**.

Figura 23 – Partes principais de um gêmeo digital.



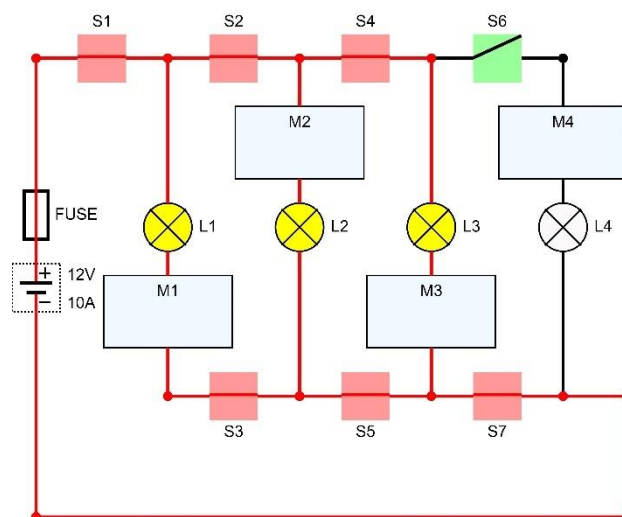
Fonte: Autoria Própria

Seguindo essa ideia, nesta seção discute-se como foram empregados componentes virtuais no laboratório remoto proposto a fim de interagir com a bancada eletrônica real do experimento.

O gêmeo digital foi inteiramente desenvolvido em HTML utilizando recursos de SVG (do inglês *Scalable Vector Graphics*). Uma grande vantagem de usar SVG em HTML é que as imagens podem ser animadas ou modificadas via CSS e/ou

JavaScript. Por ser um gráfico vetorial, as imagens criadas via SVG podem ter seu tamanho alterado, por exemplo, aplicando-se um *zoom* na página, e mesmo assim elas não perdem sua qualidade. Outra vantagem é que, como são criadas via HTML, elas exigem menos requisições para o servidor *Web*. A **Figura 24** ilustra o gêmeo digital desenvolvido neste projeto.

Figura 24 – Imagem do gêmeo digital desenvolvido para o projeto.



Fonte: Autoria Própria

O gêmeo digital do laboratório remoto proposto tem uma imagem com as seguintes dimensões: 389 *pixels* de altura e 330 *pixels* de largura. Cada item ou componente (chave, linhas, lâmpadas, pontos, etc.) são identificados com um código identificador específico. Esse código permite que os componentes possam ser manipulados individualmente via CSS e/ou JavaScript.

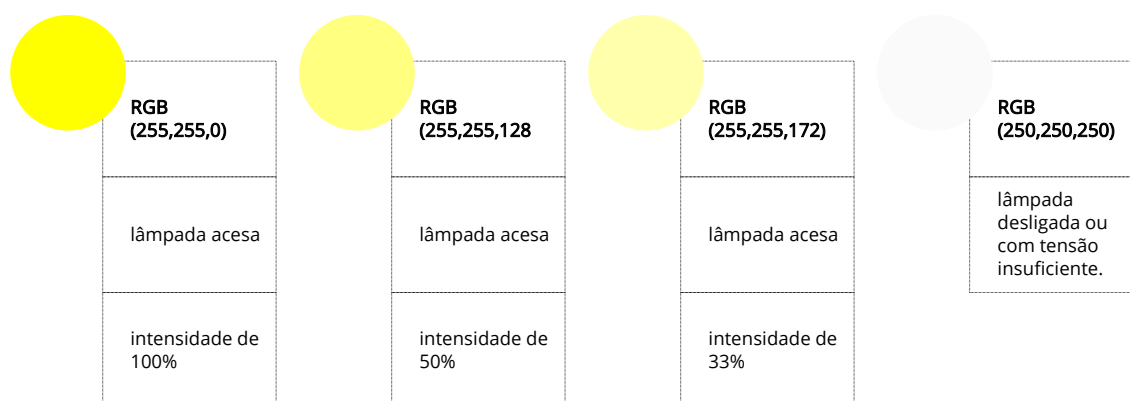
O ícone usado para representar as chaves eletrônicas respeita o símbolo padrão empregado para a representação de chaves em esquemas de circuitos elétricos apresentando a cor vermelha para indicar que estão ligadas e a cor verde quando estão desligadas, seguindo a norma IEC⁴ 60073:2002 (INTERNATIONAL ELECTROTECHNICAL COMMISSION, 2002). As chaves são botões clicáveis,

⁴ Comissão Eletrotécnica Internacional (do inglês *International Electrotechnical Commission*).

permitindo o controle do circuito via plataforma virtual de maneira remota por dispositivos com acesso à Internet. Seus ícones são animados para indicar o estado da chave: ligada ou desligada, conforme ilustrado na **Figura 24** (observe que a diferença de cor e formato entre chave S6, aberta, e as demais, fechadas).

A passagem de corrente nos ramos do circuito é indicada pela coloração vermelha seguindo a norma ABNT⁵ NBR⁶ 5410:2004 (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, 2004). As lâmpadas podem representar quatro intensidades diferentes de luminosidade (de acordo com a tensão aplicada em seus terminais), de forma que cada intensidade está associada a uma tonalidade de amarelo descrita por códigos RGB, conforme ilustrado pela **Figura 25**.

Figura 25 – Código RGB adotado para ilustrar com diferentes tons de amarelo no gêmeo digital a intensidade de luminosidade nas lâmpadas do circuito real.



Fonte: Autoria Própria

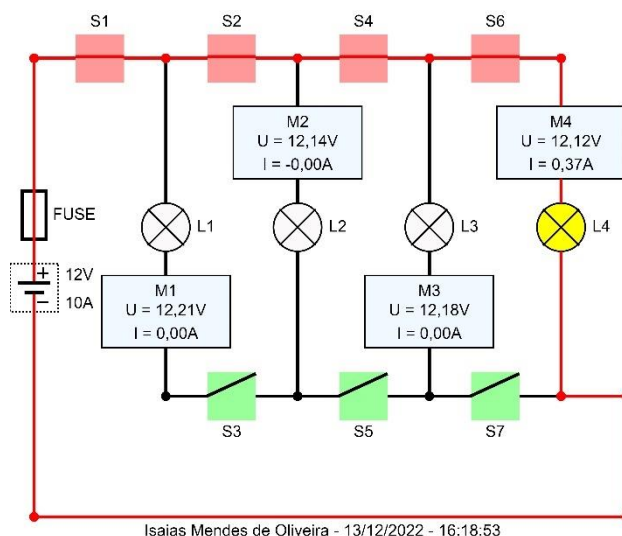
A imagem do gêmeo digital pode ser salva como um recurso didático para montagem de relatórios e posterior análise. Tal imagem exhibe o estado das chaves e lâmpadas no momento, além de indicar os valores medidos de corrente e tensão de acordo com a arranjo elétrico configurado no circuito. Na parte inferior da

⁵ Associação Brasileira de Normas Técnicas

⁶ Norma Brasileira

página onde encontra-se o gêmeo digital, o usuário pode clicar no botão **Salvar Circuito** e, em seguida, é aberta uma janela para que o usuário possa inserir uma identificação (por exemplo, seu nome completo). Ao clicar no botão **OK** é salva uma imagem com o estado do circuito naquele momento e com o nome inserido pelo usuário, a data e a hora da geração da imagem, conforme a exibido na **Figura 26**.

Figura 26 – Imagem salva de um circuito montado.



Fonte: Autoria Própria

É importante salientar que o gêmeo digital desenvolvido neste trabalho é de implementação mais simples pois é composto basicamente de formas geométricas comuns: quadrados, linhas, pontos e retângulos. Por isso foi possível desenvolver a imagem vetorial inteiramente em SVG, manipulando-a com CSS e JavaScript. Para gêmeos digitais que requerem mais sofisticação e animações mais elaboradas estão disponíveis ferramentas gratuitas que poderão ser utilizadas para automatizar esse processo de conversão da imagem desejada para um gráfico vetorial escalonável (SVG).

5.5 CUSTOS

Para finalizar, também foi avaliado o custo de investimento necessário para reproduzir o experimento remoto, uma vez que um dos principais requisitos para desenvolvimento da plataforma era manter um baixo custo a fim de aumentar o alcance do projeto, o qual poderia ser replicado por outras instituições. Conforme

descrito na **Tabela 2** foram investidos R\$ 725,80 (ou U\$ 161,29, considerando a cotação do dólar a R\$ 4,50 na época de compra dos componentes).

Tabela 2 – Custo unitário por item e total geral do projeto em reais (R\$) e dólares (U\$).

Item	Qtde	Valor unit. (U\$)	Valor unit. (R\$)	Valor total (U\$)	Valor total (R\$)
ESP32	1	14,07	63,32	14,07	63,32
Módulo Reles (8 canais)	2	9,56	43,00	19,12	86,00
Módulo LED RGB	7	1,09	4,89	7,61	34,23
Kit com 4 Lamp/soquete	1	6,64	29,90	6,64	29,90
Display	4	5,89	26,49	23,55	105,96
Medidores	4	6,54	29,45	26,18	117,80
Fonte 12V	1	10,00	45,00	10,00	45,00
Fonte 5V	3	5,11	23,00	15,33	69,00
ESP32-CAM + câmera	1	15,55	69,99	15,55	69,99
Protoboard	4	3,98	17,90	15,91	71,60
Multiplexador I2C	1	7,33	33,00	7,33	33,00
Total geral				161,29	725,80

Fonte: Aatoria Própria

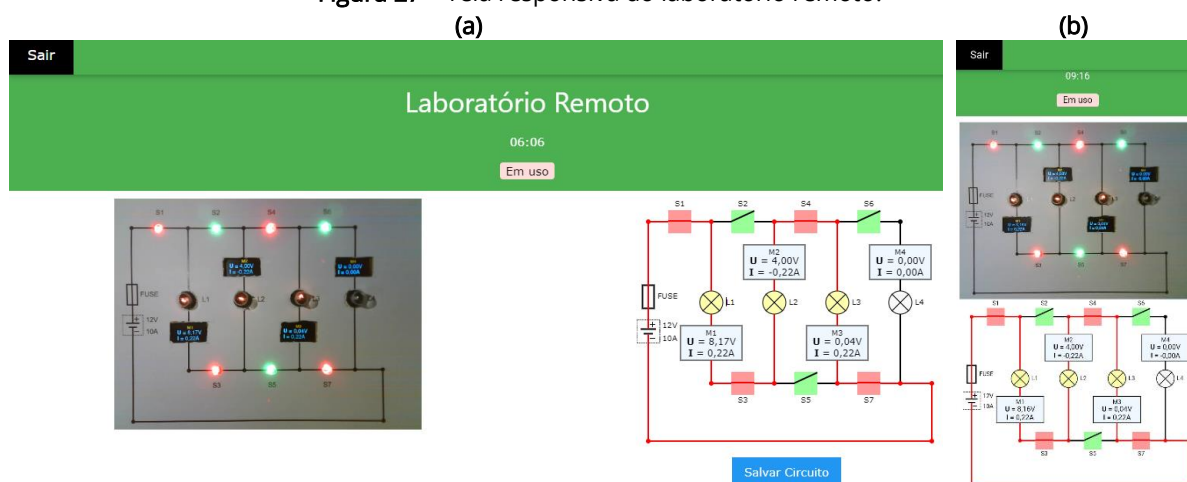
Esse investimento é relativamente baixo se comparado com o custo que o mesmo projeto teria se também fosse utilizado um SBC. Usualmente, o SBC que é recorrentemente utilizado em laboratórios remotos é o Raspberry Pi, que custa em média R\$ 1.250,00. Nesse caso, ainda seria necessário substituir a ESP32-CAM por uma *Webcam* USB com custo aproximado de R\$ 120,00, ou seja, o investimento total ficaria em torno de R\$ 1.948,58 (ou U\$ 433,02), o que permitiria a montagem de quase três plataformas idênticas à proposta.

6 RESULTADOS

A arquitetura simplificada proposta, mostrada na **Figura 19**, além de poder ser utilizada em outros contextos, mostrou-se adequada para o desenvolvimento de laboratórios remotos reduzindo os custos necessários para a implementação dos mesmos.

O protótipo desenvolvido mostrou-se funcional. Nos testes realizados os atrasos (*delays*) entre o momento que uma chave é acionada via gêmeo digital até a execução da ação pelo ESP32 não comprometem a experiência do usuário e o controle do experimento. A página *Web* para visualização do experimento apresentou uma velocidade de carregamento adequada, com boa responsividade mesmo em telas pequenas (comuns em dispositivos móveis). A **Figura 27 (a)** exibe a visualização do experimento em uma tela grande (associada a um navegador *Web* de um computador pessoal) e a **Figura 27 (b)** em uma tela pequena (*smartphone*).

Figura 27 – Tela responsiva do laboratório remoto.



Fonte: Autoria Própria

O recurso de salvamento da imagem indicando o estado atual do circuito elétrico e medições de corrente/tensão também funcionou conforme o previsto.

O gêmeo digital se comportou adequadamente, indicando corretamente os ramos de circuito alimentados por corrente, a intensidade das lâmpadas e o estado de cada chave.

O controle de usuários também funcionou da maneira esperada, impedindo a ocorrência de múltiplos acessos simultâneos ao experimento. O *streaming* de vídeo, mesmo na resolução mais alta, mostrou-se eficiente, exibindo o comportamento das lâmpadas com atualização em tempo quase real.

Em relação ao circuito elétrico, as 7 chaves permitem 128 combinações possíveis. Das 128 combinações, apenas 27 formam um circuito com passagem de corrente, ou seja, com pelo menos uma lâmpada acesa.

Desses 27 circuitos, pode ser observada a equivalência de algumas combinações, sendo que apenas 17 configurações distintas de circuitos podem ser alcançadas, as quais foram testadas e os valores de tensão e corrente obtidos pelos medidores instalados na bancada e enviados ao *display* e ao gêmeo digital pelo ESP32 foram comparados com um valor teórico calculado. A **Tabela 3** exibe quais chaves devem ser abertas ou fechadas para criar essas 17 configurações distintas de circuitos elétricos. No **Apêndice B** são exibidas as imagens do gêmeo digital com as 27 combinações onde há formação de um circuito com passagem de corrente e pelo menos uma lâmpada é acesa.

Tabela 3 – Estados das chaves nas 17 configurações distintas de circuitos elétricos.

Combinação	Referência no Apêndice B	Chaves							Circuito
		C1	C2	C3	C4	C5	C6	C7	
1	Figura 31	1	1	1	1	1	1	1	1
2	Figura 32	1	1	1	1	1	0	1	2
3	Figura 33	1	1	1	0	1	0	1	3
4	Figura 34	1	1	1	0	1	1	1	3
5	Figura 35	1	0	1	0	1	0	1	4
6	Figura 36	1	0	1	1	1	0	1	4
7	Figura 37	1	0	1	0	1	1	1	4
8	Figura 38	1	0	1	1	1	1	1	4
9	Figura 39	1	0	1	1	0	1	1	5
10	Figura 40	1	0	1	1	1	1	0	6
11	Figura 41	1	0	1	1	0	0	1	7
12	Figura 42	1	0	1	1	0	1	0	8
13	Figura 43	1	0	1	0	1	1	0	9
14	Figura 44	1	1	0	1	1	1	1	10
15	Figura 45	1	1	0	1	1	0	1	11
16	Figura 46	1	1	0	0	1	0	1	12
17	Figura 47	1	1	0	0	1	1	1	12
18	Figura 48	1	1	0	0	1	1	0	13
19	Figura 49	1	1	0	1	0	1	1	14
20	Figura 50	1	1	1	1	0	1	1	14
21	Figura 51	1	1	0	1	0	0	1	15
22	Figura 52	1	1	1	1	0	0	1	15
23	Figura 53	1	1	1	0	1	1	0	16
24	Figura 54	1	1	0	1	0	1	0	17
25	Figura 55	1	1	1	1	0	1	0	17
26	Figura 56	1	1	0	1	1	1	0	17
27	Figura 57	1	1	1	1	1	1	0	17

Fonte: Autoria Própria

Para obter o valor teórico calculado para a tensão e corrente nos quatros pontos de medição e permitir a validação do circuito, foi usado o simulador *online* Falstad, disponível em <http://www.falstad.com/circuit/circuitjs.html>, com as seguintes configurações: lâmpadas com potência nominal de 5W e tensão nominal de 12V, fonte CC de 12,20V (tensão real da fonte) e chaves SPST (Um Polo/Acionamento Único, do inglês *Single Pole Single Throw*). As medições de tensão nos pontos de interesse M1, M2, M3 e M4 (ver imagem e legenda da **Figura 22**) em cada circuito equivalente realizadas no protótipo são aproximadamente as mesmas obtidas pela simulação com diferenças de no máximo 2,9%, conforme indicado na **Tabela 4**.

Tabela 4 – Comparativo da medição de tensão nos pontos de interesse (M1 a M4).

Combinação	Valor teórico calculado				Protótipo				Diferença			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
1	0,00	12,09	0,00	12,09	0,14	11,95	0,12	11,92	0,1%	-1,2%	0,1%	-1,4%
2	0,00	12,12	0,00	0,00	0,14	12,00	0,12	0,02	0,1%	-1,0%	0,1%	0,0%
3	0,00	12,15	0,00	0,00	0,12	12,04	0,02	0,01	0,1%	-0,9%	0,0%	0,0%
4	0,00	12,15	0,00	0,00	0,11	12,03	0,04	0,00	0,1%	-1,0%	0,0%	0,0%
5	0,00	0,00	0,00	0,00	0,07	0,00	0,01	0,00	0,1%	0,0%	0,0%	0,0%
6	0,00	0,00	0,00	0,00	0,07	0,00	0,01	0,00	0,1%	0,0%	0,0%	0,0%
7	0,00	0,00	0,00	0,00	0,07	0,00	0,01	0,00	0,1%	0,0%	0,0%	0,0%
8	0,00	0,00	0,00	0,00	0,07	0,00	0,01	0,00	0,1%	0,0%	0,0%	0,0%
9	6,41	0,00	0,00	0,00	6,58	0,84	0,01	0,79	2,7%	0,8%	0,0%	0,8%
10	6,41	5,78	6,40	5,78	6,53	5,65	6,51	5,61	1,9%	-2,2%	1,7%	-2,9%
11	8,12	4,06	0,00	0,00	8,12	4,00	0,02	0,00	0,0%	-1,5%	0,0%	0,0%
12	8,12	4,06	4,06	4,06	8,18	4,05	4,02	3,98	0,7%	-0,2%	-1,0%	-2,0%
13	8,12	8,12	8,12	4,06	8,11	8,05	8,06	4,04	-0,1%	-0,9%	-0,7%	-0,5%
14	12,12	12,12	0,00	12,12	12,16	11,97	0,08	11,94	0,3%	-1,2%	0,1%	-1,5%
15	12,15	12,15	0,00	0,00	12,18	12,01	0,80	0,00	0,2%	-1,2%	0,8%	0,0%
16	12,17	12,17	0,00	0,00	12,20	12,06	0,01	0,00	0,2%	-0,9%	0,0%	0,0%
17	12,17	12,17	0,00	0,00	12,20	12,06	0,01	0,00	0,2%	-0,9%	0,0%	0,0%
18	12,18	12,18	8,12	4,06	12,20	12,10	8,00	3,99	0,2%	-0,7%	-1,5%	-1,7%
19	12,15	12,15	0,00	12,15	12,17	12,07	0,06	11,99	0,2%	-0,7%	0,1%	-1,3%
20	12,15	12,15	0,00	12,15	12,17	12,07	0,06	11,99	0,2%	-0,7%	0,1%	-1,3%
21	12,17	12,17	0,00	0,00	12,20	12,12	0,06	0,00	0,2%	-0,4%	0,1%	0,0%
22	12,17	12,17	0,00	0,00	12,20	12,12	0,06	0,00	0,2%	-0,4%	0,1%	0,0%
23	11,56	12,18	11,56	5,79	11,38	12,12	11,34	5,65	-1,6%	-0,5%	-1,9%	-2,4%
24	12,17	12,17	12,17	12,17	12,20	12,12	12,14	12,06	0,2%	-0,4%	-0,2%	-0,9%
25	12,17	12,17	12,17	12,17	12,20	12,12	12,14	12,06	0,2%	-0,4%	-0,2%	-0,9%
26	12,17	12,17	12,17	12,17	12,20	12,12	12,14	12,06	0,2%	-0,4%	-0,2%	-0,9%
27	12,17	12,17	12,17	12,17	12,20	12,12	12,14	12,06	0,2%	-0,4%	-0,2%	-0,9%

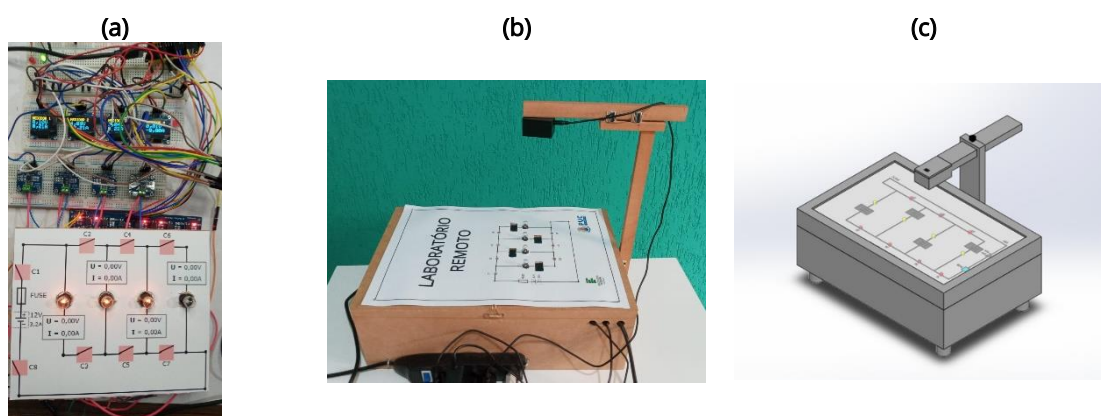
Fonte: Autoria Própria

Observa-se que a variação entre os valores teóricos calculados e os medidos pelo protótipo é bem pequena e não compromete o entendimento do funcionamento do circuito. Essa variação pode ser decorrente de vários motivos como, por exemplo, comprimento e secção transversal dos fios, variação da resistência das lâmpadas pela temperatura, etc. Uma vez que essa análise foge do escopo deste trabalho, uma investigação mais aprofundada dessas diferenças percentuais de medição e suas causas pode ser proposta como objeto de estudo dos alunos que farão uso do experimento.

6.1 ASPECTOS CONSTRUTIVOS E VISTA EXTERNA DO EXPERIMENTO

Posto que a plataforma desenvolvida se trata de um protótipo, ou seja, um produto projetado para testes, é possível que antes de sua aplicação final em uma instituição de ensino ainda ocorram algumas adequações. Por esse motivo, o experimento foi montado sobre *protoboards*, conforme indicado na **Figura 28 (a)**. A montagem eletrônica foi depositada dentro de uma caixa de madeira na qual as *protoboards* foram fixadas (para transporte e proteção), conforme indicado na **Figura 28 (b)**. Posteriormente, é recomendado que, para uso em ensino, sejam desenvolvidas placas de circuito impresso, de forma que a versão final tenha uma aparência similar àquela ilustrada pela **Figura 28 (c)**.

Figura 28 – Diferentes montagens da parte externa do experimento.



Fonte: Autoria Própria

6.2 TESTE DE DESEMPENHO DO SERVIDOR *WEB*

Diferentes parâmetros podem ser usados para medir o desempenho de um servidor *Web*. Um parâmetro comumente adotado em testes de desempenho de servidores *Web* divulgados na literatura é o número de solicitações por segundo que o servidor pode responder sob diferentes condições de carga (LIND, 2014).

Dois outros parâmetros de desempenho usualmente avaliados são o tempo de resposta, medido em milissegundos, e a taxa de transferência, medida em *bytes* por segundo. Em laboratórios remotos um tempo de resposta baixo é extremamente importante para não comprometer a experiência do usuário.

A taxa de transferência também é um parâmetro importante, mas é afetada por diferentes fatores que não podem ser controlados pelo programador da plataforma como, por exemplo, tipo de conexão do usuário, distância entre o dispositivo do usuário e o servidor *Web*, entre outros. Adicionalmente, a confiabilidade do teste realizado está associada à ferramenta de teste escolhida, à definição dos parâmetros de configuração e à elaboração de um cenário que simule de maneira precisa a ação dos usuários, para que eventualmente seja possível comparar o desempenho da plataforma com os resultados obtidos por outros servidores similares.

Na literatura existem diversos resultados e análises de testes de servidores como em (LIND, 2014), (NGUYEN, 2017), (JADER; ZEEBAREE; ZEBARI, 2019) e (SURADKAR; LOMTE, 2020) mas nenhum desses trabalhos está focado na análise de servidores *Web* embarcados em SBMs. Além disso, ainda não foi desenvolvida uma metodologia padrão para a realização desses testes. Cabe destacar que os testes realizados por (LIND, 2014) foram em servidores *Web* embarcados em SBCs e não em SBMs.

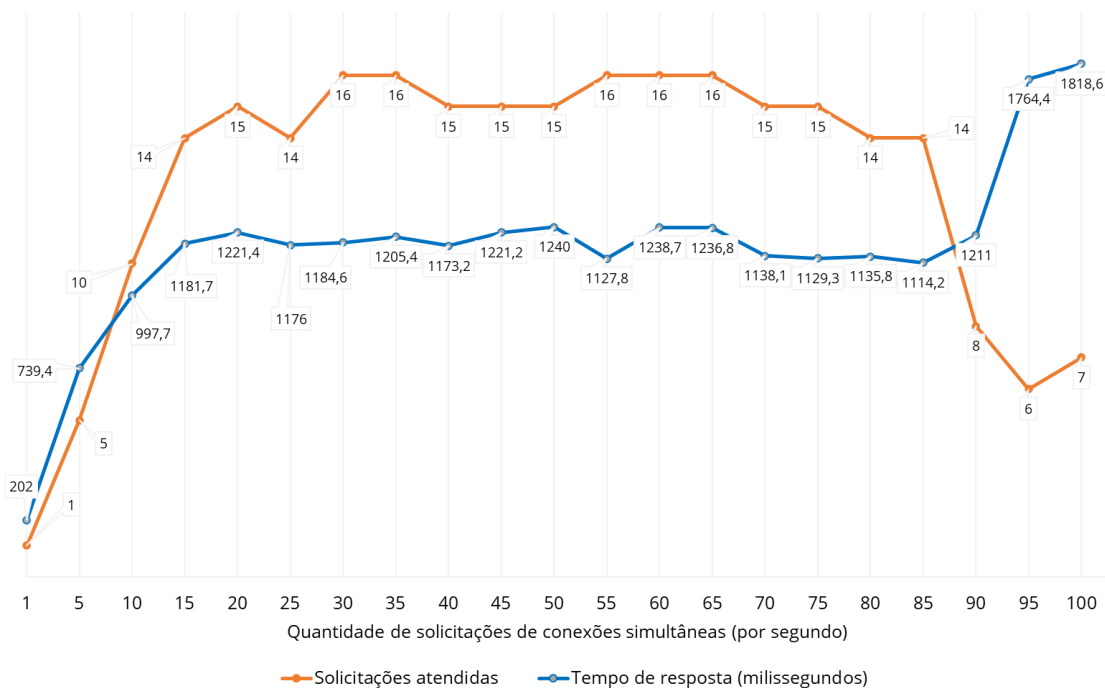
Levando em consideração o cenário descrito anteriormente, é importante destacar que há uma carência de estudos voltados à análise do desempenho de servidores *Web* embarcados em SBMs e que isso abre portas para investigações futuras.

Neste trabalho é realizado, a título de ilustração, um teste simples para medir minimamente o desempenho do servidor *Web* embarcado no ESP32. Para isso foi utilizada a ferramenta gratuita HTTPERF⁷. O teste foi realizado em uma rede local utilizando uma conexão Wi-Fi. Foram enviadas várias solicitações de conexões

⁷ A ferramenta **httperf** (<https://github.com/httpperf/httpperf/blob/master/README.md>) foi desenvolvida com o propósito de fornecer um recurso flexível para gerar várias cargas de trabalho HTTP e medir o desempenho do servidor.

para o servidor *Web*, iniciando com 1 até 100 solicitações simultâneas. A **Figura 29** ilustra graficamente o resultado do teste.

Figura 29 – Resultados do teste de desempenho do servidor *Web*.



Fonte: Autoria Própria

Segundo (NIELSEN, 2010), para que o usuário tenha uma boa experiência em aplicações *Web* é necessário que o tempo de resposta esteja, pelo menos, na casa dos 1000 milissegundos. Nos testes realizados é possível notar que o servidor *Web* satisfaz esse requisito com até 10 solicitações de conexões de forma simultânea. Adicionalmente, o servidor continua atendendo satisfatoriamente até 15 conexões simultâneas com 1240 milissegundos de atraso máximo na resposta, muito próximo do que é considerado adequado para que a experiência do usuário não seja prejudicada. É possível observar que, conforme o número de solicitações de conexões é incrementado, há uma saturação (por volta de 15 solicitações) do servidor *Web*, enquanto que para 90 solicitações ocorre um aumento abrupto do tempo de resposta e uma diminuição brusca da quantidade de solicitações atendidas. Para aplicações onde não ocorre a conexão simultânea de dezenas de usuários, o servidor *Web* mostrou-se adequado e satisfatório.

Importante salientar que o teste foi realizado com o ESP32 respondendo, simultaneamente, às requisições: do servidor *Web*, dos sensores, dos envios e dos recebimentos de mensagens via *Websockets* da página *Web*. Em outras situações, onde o ESP32 é menos exigido, o teste pode ser ainda mais satisfatório.

6.3 TESTE DE DESEMPENHO DA PÁGINA *WEB*

Ao disponibilizar uma aplicação *Web* para uso é importante se preocupar em como essa aplicação é carregada e exibida em diferentes tipos de dispositivos e taxas de transferência de redes. Por isso é essencial que os arquivos que compõem essa aplicação tenham tamanhos reduzidos facilitando o seu carregamento. Também é importante se preocupar com a acessibilidade e a aplicação de boas práticas de programação e desenvolvimento para que a aplicação tenha um bom desempenho e proporcione uma experiência adequada para o usuário.

Os testes de desempenho de páginas *Web* podem ser realizados por diversas ferramentas entre as quais a Google Lighthouse tem ganhado cada vez mais atenção nos últimos anos. Google Lighthouse é uma ferramenta de código aberto que fornece métricas de desempenho, acessibilidade, otimização de mecanismos de busca e aplicativos *Web* progressivos e, se necessário, indica como alterar essas métricas visando um melhor desempenho da página *Web* (CHROME DEVELOPERS, [s.d.]).

A ferramenta Google Lighthouse disponibiliza cinco categorias de métricas de desempenho que pontuam de 0 a 100, classificando de 0 a 49 como ruim, de 50 a 89 como razoável e de 90 a 100 como ótimo. As cinco categorias são:

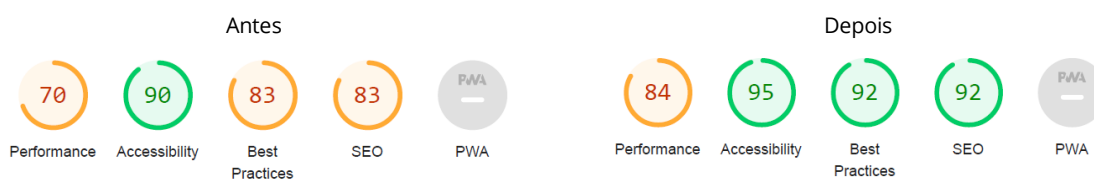
- Desempenho (*Performance*) – garantem que sua página seja otimizada para que os usuários possam ver e interagir com o conteúdo das páginas *Web* adequadamente. Mede a rapidez com que sua página carrega ou com que os principais elementos ficam disponíveis para usuários em dispositivos móveis;

- Acessibilidade⁸ (*Accessibility*) – destacam oportunidades para melhorar a acessibilidade da aplicação *Web*. Essa métrica recebe como pontuação uma média ponderada de cerca de 35 características de acessibilidade;
- Práticas recomendadas (*Best Practices*) – destacam oportunidades para melhorar a integridade geral do código da aplicação *Web*;
- Otimização para mecanismos de busca (do inglês *Search Engine Optimization* - SEO) – garantem que as páginas *Web* sejam otimizadas para melhorar a classificação de resultados em mecanismos de busca e pesquisas;
- Aplicativos *Web* Progressivos (do inglês *Progressive Web App* - PWA) – caso a aplicação seja um PWA, validam os aspectos desse tipo de desenvolvimento. Com base na lista de verificação básica PWA, calcula o nível de integridade do seu site.

Neste trabalho foi utilizada a ferramenta Google Lighthouse para melhorar a qualidade da página *Web* (que controla o experimento) e conseqüentemente obter um melhor carregamento da mesma. Foram realizados vários ajustes para que todas as categorias de métricas, com exceção da PWA, uma vez que a página não usa essa metodologia no seu desenvolvimento, ficassem com uma pontuação ótima (entre 90 e 100), conforme pode ser visto no comparativo de antes e depois dos ajustes, ilustrado pela **Figura 30**.

⁸ Acessibilidade da aplicação *Web* refere-se ao projeto de conteúdo de forma que usuários com uma ampla gama de deficiências (incluindo visuais, auditivas, físicas, de fala, cognitivas, de linguagem, de aprendizagem e neurológicas) possam acessá-la. Dependendo da finalidade da página *Web*, a acessibilidade se sobrepõe a outras categorias, como *Web design* responsivo, usabilidade e otimização de mecanismos de pesquisa, visto que *sites* acessíveis podem ter maior alcance de audiência, melhores resultados de busca e demonstrar responsabilidade social aos usuários.

Figura 30 – Pontuação obtida pela página *Web* no Google Lighthouse.



Fonte: Autoria Própria

Dentre os ajustes que foram realizados pode-se citar: minificação⁹ (do inglês *Minification*) dos arquivos HTML, CSS e JavaScript, remoção de códigos CSS e JavaScript não utilizados, desenvolvimento de página *Web* responsiva, otimização das imagens, compactação de arquivos, etc.

Na categoria Performance não foi possível atingir uma pontuação entre 90 e 100 pontos pois era necessário alterar a versão do protocolo HTTP do servidor *Web* do ESP32. Por padrão, o ESP32 trabalha com a versão 1.1 e a sugestão do Google Lighthouse é usar a versão 2.0. Apesar de ser uma alteração possível, isso implicaria que toda a programação deste projeto deveria ser refeita, o que é inviável. Por esse motivo, sugere-se que autores de trabalhos futuros considerem empregar a versão 2.0 desde o início do projeto.

⁹ Minificação é o processo de remover todos os caracteres desnecessários do código-fonte de linguagens de programação interpretadas ou linguagens de marcação sem alterar sua funcionalidade.

7 CONCLUSÃO

O desenvolvimento da plataforma experimental com controle e acesso remotos foi finalizada com sucesso. A funcionalidade do experimento foi testada e considerada adequada: as medições do circuito elétrico foram validadas, mesmo com a latência, presente em qualquer aplicação que dependa de uma rede de computadores para funcionar, entre o momento que uma chave é acionada via gêmeo digital até a execução da ação pelo ESP32 não há comprometimento da experiência do aluno e a qualidade visual para usuário foi considerada apropriada independentemente do tamanho da tela do dispositivo de acesso à página *Web* que controla o experimento.

Conclui-se que o objetivo principal do trabalho foi atingido: desenvolver uma plataforma de laboratório remoto para ensino de elétrica básica, apresentando como contribuição a proposta de uma arquitetura alternativa mais simplificada e barata, com recursos virtuais intuitivos que permitem uma boa interatividade e melhor compreensão da contraparte física do experimento.

Os objetivos específicos também foram alcançados: foi proposta, desenvolvida e aplicada uma arquitetura simplificada no desenvolvimento de um laboratório remoto; foram usados somente SBMs no processamento, supervisão de dados e controle do experimento, eliminando a necessidade do uso de um SBC ou PC para esta função; criou-se um gêmeo digital utilizando linguagens de programação *Web* amplamente aplicadas no desenvolvimento de aplicações *Web*; os sensores, microcontroladores e demais componentes eletrônicos empregados na parte de *hardware* da plataforma experimental são acessíveis em termos de custo.

Além de alcançar os objetivos inicialmente propostos, após o desenvolvimento do trabalho foram observadas outras contribuições relevantes desta pesquisa, as quais são listadas a seguir:

- A arquitetura simplificada proposta é escalável, podendo ser adaptada, modificada e aplicada em outras áreas de forma que o experimento objeto de estudo pode ser alterado e a arquitetura aproveitada;
- A modularidade da plataforma desenvolvida facilita manutenção e a reprodução do laboratório remoto por outras instituições;
- Os resultados desta dissertação podem ser usados como um tutorial para replicar a arquitetura proposta em disciplinas de eletrônica embarcada e IoT, permitindo que os alunos usem os erros e acertos deste trabalho para pular etapas e possam implementar outros tipos de montagens que possam ser acessadas, monitoradas e controladas remotamente, permitindo multiplicar a quantidade de experimentos com acesso remoto disponíveis em uma mesma instituição.

Trabalho publicado

Os resultados parciais desta dissertação foram publicados em:

- Isaias M. Oliveira, Cecília F. Morais. Plataforma experimental de eletricidade básica com acesso e controle remotos implementada diretamente em microcontrolador. In Versão Preliminar dos Anais do XXIV Congresso Brasileiro de Automática (CBA2022), pp 1–8, Fortaleza, CE, Outubro 2022. Disponível em https://www.sba.org.br/cba2022/wp-content/uploads/artigos_cba2022/paper_1494.pdf

Repositório da plataforma

Os arquivos e documentos relacionados ao desenvolvimento da plataforma estarão disponíveis em:

- <https://github.com/isaiasmendesoliveira/Plataforma-experimental-de-eletricidade-basica>

8 TRABALHOS FUTUROS

Nesta sessão são elencadas algumas possibilidades de trabalhos futuros que podem melhorar o uso da arquitetura e plataforma apresentadas neste trabalho:

1. Como existem várias bibliotecas disponíveis para criar o servidor *Web* no ESP32 seria importante realizar novos testes de desempenho utilizando outras bibliotecas para verificar se ocorrerá alteração no desempenho;
2. Criar mecanismos para gerenciar uma fila de usuários.
3. Aplicar o roteiro de aprendizagem com vistas a obter um *feedback* dos alunos e professores;
4. Criar os arquivos para impressão da placa de circuito;
5. Criar os arquivos para a impressão 3D da parte externa do experimento;
6. Buscar parecerias com outros autores que desenvolvem laboratórios remotos para testar a arquitetura proposta em outros tipos de experimentos;
7. Aplicar a arquitetura proposta, bem como suas variações, em outras áreas como, por exemplo, no monitoramento na indústria 4.0.

REFERÊNCIAS BIBLIOGRÁFICAS

ÁLVAREZ, José Luis; MOZO, Juan Daniel; DURÁN, Eladio. Analysis of single board architectures integrating sensors technologies†. *Sensors*, v. 21, n. 18, p. 6303, 21 set. 2021. Disponível em: <<https://www.mdpi.com/1424-8220/21/18/6303>>.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 5410: Instalações elétricas de baixa tensão**. Rio de Janeiro, RJ: [s.n.], 2004.

AZAD, Abul Kashem Mohammed. Design and Development of Remote Laboratories with Internet of Things Setting. *Advances in Internet of Things*, v. 11, n. 03, p. 95–112, 2021.

BRAGA, Newton C. **Relés - Conceitos e Aplicações**. 1ª ed. São Paulo: Instituto Newton C. Braga, 2012.

CEPAL-UNESCO. La educación en tiempos de la pandemia de COVID-19. **Informe COVID-19**, p. 21, ago. 2020. Disponível em: <https://repositorio.cepal.org/bitstream/handle/11362/45904/1/S2000510_es.pdf>.

CHARLES K. ALEXANDER; SADIKU, Matthew N. O. **Fundamentos de Circuitos Elétricos**. 5. ed. Porto Alegre: [s.n.], 2013.

CHAVES, Pedro Rinaldo. **A contribution to the teaching of wireless connectivity for the internet of things: a network environment for training and experimentation**. 2022. 216 f. Universidade Estadual de Campinas, 2022. Disponível em: <<http://repositorio.unicamp.br/Busca/Download?codigoArquivo=552010>>.

CHAVES, Pedro Rinaldo; ASSUMPÇÃO, Raphael M.; FERREIRA, Luiz C.; CARDIERI, Paulo; BRANQUINHO, Omar C.; FRUETT, Fabiano. A remote emulation environment for the teaching of low-power wireless communications. *Computer Applications in Engineering Education*, v. 29, n. 6, p. 1453–1464, 2021.

CHROME DEVELOPERS. **Lighthouse**. Disponível em: <<https://developer.chrome.com/docs/lighthouse/overview/>>. Acesso em: 1 nov. 2022.

CHU, Pong P. **FPGA Prototyping by SystemVerilog Examples**. [S.l.]: Wiley, 2018.

DA SILVA, Juarez Bento; DE LIMA, João Paulo Cardoso; SIMÃO, José Pedro Schardosim; CARLOS, Lucas Mellos; PEREIRA, Josiel. **Manual Técnico - Painel Elétrico CA**. . Araranguá: Laboratório de Experimentação Remota. Disponível em: <<http://relle.ufsc.br/docs/57ff85eaefc2f.pdf>>. , 2016

DELFORNO, LEONARDO. **BANCADA DE EMULAÇÃO DE CANAL DE RÁDIO PARA**

RSSF EM INTERNET DAS COISAS. 2020. PONTIFÍCIA UNIVERSIDADE CATÓLICA DE CAMPINAS, 2020. Disponível em: <<https://repositorio.sis.puc-campinas.edu.br/handle/123456789/15067>>.

DRACHSLER, Hendrik; SCHNEIDER, Jan. JCAL Special Issue on Multimodal Learning Analytics. **Journal of Computer Assisted Learning**, v. 34, n. 4, p. 335–337, ago. 2018. Disponível em: <<https://onlinelibrary.wiley.com/doi/10.1111/jcal.12291>>.

DUTTA, Dr. Umesh; KHURANA, Nilansh; DEVDUTT. **The Internet of Thing Using NodeMCU: A Pratical Approach to Master IoT**. [S.l.]: Blue Rose Publishers, 2021.

FERNÁNDEZ-PACHECO, Atilano; MARTIN, Sergio; CASTRO, Manuel. Implementation of an arduino remote laboratory with raspberry pi. **IEEE Global Engineering Education Conference, EDUCON**, v. April-2019, p. 1415–1418, 2019.

FLANAGAN, David. **JavaScript: O Guia Definitivo**. 6. ed. [S.l.]: Bookman, 2012.

FOLTÝNEK, Petr; BABIUCH, Marek; ŠURÁNEK, Pavel. Measurement and data processing from Internet of Things modules by dual-core application using ESP32 board. **Measurement and Control (United Kingdom)**, v. 52, n. 7–8, p. 970–984, 2019.

GORBOUNOV, Yassen; CHEN, Hao. Low-Cost Remote Laboratory for Cyber-Physical Experiments. p. 149–152, 2021.

HERADIO, Ruben; DE LA TORRE, Luis; DORMIDO, Sebastian. Virtual and remote labs in control education: A survey. **Annual Reviews in Control**, v. 42, p. 1–10, 2016. Disponível em: <<http://dx.doi.org/10.1016/j.arcontrol.2016.08.001>>.

INTERNATIONAL ELECTROTECHNICAL COMMISSION. **IEC 60073: Basic and safety principles for man-machine interface, marking and identification - Coding principles for indicators and actuators**. [S.l: s.n.], 2002.

JADER, Omid H.; ZEEBAREE, Subhi R.M.; ZEBARI, Rizgar R. A state of art survey for web server performance measurement and load balancing mechanisms. **International Journal of Scientific and Technology Research**, v. 8, n. 12, p. 535–543, 2019.

KALÚZ, Martin; GARCÍA-ZUBÍA, Javier; FIKAR, Miroslav; ČIRKA, Ľuboš. A Flexible and Configurable Architecture for Automatic Control Remote Laboratories. **IEEE Transactions on Learning Technologies**, v. 8, n. 3, p. 299–310, 2015.

KRITZINGER, Werner; KARNER, Matthias; TRAAR, Georg; HENJES, Jan; SIHN, Wilfried. Digital Twin in manufacturing: A categorical literature review and classification. **IFAC-PapersOnLine**, v. 51, n. 11, p. 1016–1022, 2018. Disponível em:

<<https://linkinghub.elsevier.com/retrieve/pii/S2405896318316021>>.

LIMA, Joselice Ferreira; NETO, João da Rocha Medrado; MARTINS, Carlos A. P. S. Laboratório virtual: apresentação, conceituação, análise e uma proposta de definição. 2005. Disponível em: <<http://www.abenge.org.br/cobenge/legado/arquivos/14/artigos/MG-15-51506386687-1119055068933.pdf>>.

LIND, Daniel. **Performance evaluation of HTTP web servers in embedded systems**. 2014. KTH Industrial Engineering and Management, 2014.

LUETH, Knud Lass. **Why the Internet of Things is called Internet of Things: Definition, history, disambiguation**. Disponível em: <<https://iot-analytics.com/internet-of-things-definition/>>. Acesso em: 2 dez. 2021.

MELLOS, Lucas Carlos; SILVA, Juarez Bento da; LIMA, João Paulo Cardoso de; SIMÃO, José Pedro Schardosim. **Manual Técnico - Painel Elétrico CC**. . Araranguá: Laboratório de Experimentação Remota. Disponível em: <<http://relle.ufsc.br/docs/57ff98e8277a1.pdf>>. , 2016

MELLOS, Lucas Carlos; DA SILVA, Juarez Bento; DE LIMA, João Paulo Cardoso; SIMÃO, José Pedro Schardosim. **Manual Técnico - Painel Elétrico CC**. . Araranguá: Laboratório de Experimentação Remota. Disponível em: <<http://relle.ufsc.br/docs/57ff98e8277a1.pdf>>. , 2016

MICHAELIS. **Michaelis - Dicionário Brasileiro da Língua Portuguesa**. Disponível em: <<https://michaelis.uol.com.br/busca?r=0&f=0&t=0&palavra=laboratório>>. Acesso em: 18 maio 2022.

MIRANDA, Jhonattan; LOPEZ, Claudia S.; NAVARRO, Sergio; BUSTAMANTE, Martin R.; MOLINA, Jose Martin; MOLINA, Arturo. Open Innovation Laboratories as Enabling Resources to Reach the Vision of Education 4.0. **Proceedings - 2019 IEEE International Conference on Engineering, Technology and Innovation, ICE/ITMC 2019**, 2019.

MITROVIC, Nikola; DORDEVIC, Milan; VELJKOVIC, Sandra; DANKOVIC, Danijel. Implementation of WebSockets in ESP32 based IoT Systems. **2021 15th International Conference on Advanced Technologies, Systems and Services in Telecommunications, TELSIKS 2021 - Proceedings**, p. 261–264, 2021.

NENOV, T R; EVSTATIEV, B I; KADIROVA, S Y. Design and Implementation of a Remote Lab for Investigation of DC Motors. 30 jun. 2022, [S.l.]: IEEE, 30 jun. 2022. p. 1–5. Disponível em: <<https://ieeexplore.ieee.org/document/9831406/>>.

NGUYEN, Van Nam. Comparative Performance Evaluation of Web Servers. **VNU**

Journal of Science: Comp. Science & Com. Eng, v. 31, n. 3, p. 28–34, 2017.

NIELSEN, Jakob. **Website Response Times**. Disponível em: <<https://www.nngroup.com/articles/website-response-times/>>. Acesso em: 13 dez. 2022.

ONER, Vedat Ozan. **Developing IoT Projects with ESP32: Automate Your Home Or Business with Inexpensive Wi-Fi Devices**. [S.l.]: Packt Publishing, 2021.

PERERA, Charith; LIU, Chi Harold; JAYAWARDENA, Srimal; CHEN, Min. A Survey on Internet of Things from Industrial Market Perspective. **IEEE Access**, v. 2, p. 1660–1679, 2015.

PERERA, Charith. **Sensing as a Service for Internet of Things: A Roadmap**. [S.l.: s.n.], 2017. Disponível em: <<https://hal.archives-ouvertes.fr/hal-01443043>>.

POLIAKOV, Mykhailo; RIDA, Ibrahim. Remote laboratories for engineering education: Status and prospects. **2020 Advances in Science and Engineering Technology International Conferences, ASET 2020**, 2020.

POUNCEY, Ian; YORK, Richard. **Beginning CSS: Cascading Style Sheets for Web Design**. [S.l.]: Wiley, 2011.

PROVDANOV, Cleber Cristiano; FREITAS, Ernani Cesar de. **Metodologia do trabalho científico: métodos e técnicas da pesquisa e do trabalho acadêmico**. 2. ed. Novo Hamburgo: [s.n.], 2013.

RIQUELME, Fabian; MUNOZ, Roberto; MAC LEAN, Roberto; VILLARROEL, Rodolfo; BARCELOS, Thiago S.; DE ALBUQUERQUE, Victor Hugo C. Using multimodal learning analytics to study collaboration on discussion groups: A social network approach. **Universal Access in the Information Society**, v. 18, n. 3, p. 633–643, 22 ago. 2019. Disponível em: <<http://link.springer.com/10.1007/s10209-019-00683-w>>.

SARAVANA MOHAN, M; KARTHIKEYAN, P; RAM KUMAR, D; RUPESH, M. Remote laboratory. **International Journal of Engineering and Advanced Technology**, v. 8, n. 6 Special issue, p. 554–559, 2019. Disponível em: <<https://www.ijeat.org/wp-content/uploads/papers/v8i6S/F11120886S19.pdf>>.

SIEMENS. **Digital Twin**. Disponível em: <<https://www.plm.automation.siemens.com/global/en/our-story/glossary/digital-twin/24465>>. Acesso em: 10 maio 2022.

SILVA, Juarez Bento da; LIMA, João Paulo Cardoso de; SIMÃO, José Pedro Scharodosim; CARLOS, Lucas Mellos; PEREIRA, Josiel. **Manual Técnico - Painel**

Elétrico CA. . Araranguá: Laboratório de Experimentação Remota. Disponível em: <<http://relle.ufsc.br/docs/57ff85eaefc2f.pdf>>. , 2016

SILVA, Rafael de Amorim; NOVA, João Vila; VASCONCELOS, Rubem; CALADO, Ivo; BRANCO, Kalinka Castelo; BRAGA, Rosana. Aplicando Internet das Coisas na Educação: Tecnologia, Cenários e Projeções. 27 out. 2017, [S.l: s.n.], 27 out. 2017. p. 1256. Disponível em: <<http://www.br-ie.org/pub/index.php/wcbie/article/view/7514>>.

SOLOMON SYSTECH. SSD1306 Datasheet. n. April, p. 65, 2008. Disponível em: <<https://www.crystalfontz.com/controllers/SolomonSystech/SSD1306/339/>>.

STATISTA RESEARCH DEPARTMENT. **Internet of Things - number of connected devices worldwide 2015-2025.** . [S.l: s.n.], 2016. Disponível em: <<https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/>>.

SULEMA, Yevgeniya; DYCHKA, Ivan; SULEMA, Olga. Multimodal Data Representation Models for Virtual, Remote, and Mixed Laboratories Development. 2019, Cham: Springer International Publishing, 2019. p. 559–569.

SURADKAR, Nitin; LOMTE, Santosh. VMware ESXi: Virtual Web Server performance evaluation with weighttp Benchmark. **Proceedings of IEEE International Conference on Advent Trends in Multidisciplinary Research and Innovation, ICATMRI 2020**, p. 4–7, 2020.

SYSTEMS ESPRESSIF. ESP32WROOM32 Datasheet. p. 27, 2022. Disponível em: <https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32_datasheet_en.pdf>.

TABARÉS, Raúl. HTML5 and the evolution of HTML; tracing the origins of digital platforms. **Technology in Society**, v. 65, n. February, 2021.

TEXAS INSTRUMENTS. Bidirectional Current/Power Monitor INA219 Datasheet. n. December, p. 39, 2015. Disponível em: <<http://www.ti.com/lit/ds/symlink/ina219.pdf>>.

_____. Multiplexer TCA9548A Datasheet. n. November, p. 39, 2019. Disponível em: <<https://www.ti.com/lit/ds/symlink/tca9548a.pdf>>.

VARGAS, Javier; CUERO, Jairo; TORRES, Camilo. Laboratorios Remotos e IOT una oportunidad para la formación en ciencias e ingeniería en tiempos del COVID-19: Caso de Estudio en Ingeniería de Control. **Espacios**, v. 41, n. 42, p. 188–198, 2020.

VILELA, Douglas Carlos; GERMANO, José Silvério Edmundo; MONTEIRO, Marco

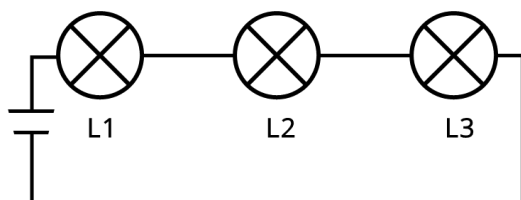
Aurélio Alvarenga; CARVALHO, Samuel José de. Estudo comparativo de um experimento de eletrodinâmica: Laboratório Tradicional x Laboratório Remoto. **Revista Brasileira de Ensino de Física**, v. 41, n. 4, 2019. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1806-11172019000400505&tIng=pt>.

WUTTKE, Heinz-Dietrich; HENKE, Karsten; HUTSCHENREUTER, René. Digital Twins in Remote Labs. 2020, Cham: Springer International Publishing, 2020. p. 289.

XIAO, Perry. **Designing Embedded Systems and the Internet of Things (IoT) with the ARM Mbed**. [S.l.]: Wiley, 2018.

APÊNDICE A – PROPOSTA DE ROTEIRO DE APRENDIZAGEM

- 1) Inserir um texto de apresentação e orientações iniciais
- 2) Inserir um texto de apresentação dos objetivos
 - a) Objetivo 1 – classificação de tipos de circuitos
 - b) Objetivo 2 – 1ª Lei de Kirchhoff
 - c) Objetivo 3 – 2ª Lei de Kirchhoff
- 3) Inserir um texto explicando sobre os tipos de circuitos (série, paralelo e misto)
- 4) Inserir um texto explicando sobre a 1ª Lei de Kirchhoff (Lei das Correntes ou Leis dos Nós)
- 5) Inserir um texto explicando sobre a 2ª Lei de Kirchhoff (Lei das Tensões ou Lei das Malhas)
- 6) Reproduza o circuito abaixo no laboratório remoto e em seguida responda as questões:



- a) Quais chaves foram abertas e quais foram fechadas para a construir o circuito? Preencha a tabela abaixo:

Chave	Condição
1	
2	
3	
4	
5	
6	
7	

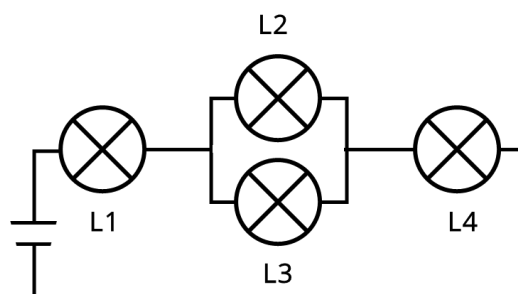
- b) Preencha a tabela abaixo com os valores de tensão e corrente em cada um dos quatros medidores:

Medidor	Corrente	Tensão
1		
2		
3		
4		

- c) Por que o valor da corrente em alguns ramos é negativo, enquanto em outros é positivo? Justifique.
- d) Sabendo que o valor de tensão apresentado no medidor é entre o nó e o terra, qual é a tensão aplicada nas lâmpadas?

Lâmpada	Tensão
1	
2	
3	
4	

- e) Valide o circuito com base na 1ª Lei de Kirchhoff
 - f) Valide o circuito com base na 2ª Lei de Kirchhoff
 - g) Classifique o circuito em série, paralelo ou misto, justificando sua resposta.
- 7) Reproduza o circuito abaixo no laboratório remoto e em seguida responda as questões:



- a) Quais chaves foram abertas e quais foram fechadas para a construir o circuito? Preencha a tabela abaixo:

Chave	Condição
1	
2	
3	
4	
5	
6	
7	

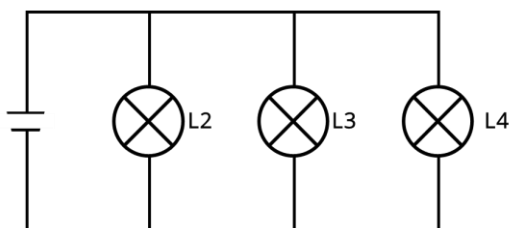
- b) Preencha a tabela abaixo com os valores de tensão e corrente em cada um dos quatros medidores:

Medidor	Corrente	Tensão
1		
2		
3		
4		

- c) Sabendo que o valor de tensão apresentado no medidor é entre o nó e o terra, qual é a tensão aplicada nas lâmpadas?

Lâmpada	Tensão
1	
2	
3	
4	

- d) Valide o circuito com base na 1ª Lei de Kirchhoff
- e) Valide o circuito com base na 2ª Lei de Kirchhoff
- f) Classifique o circuito em série, paralelo ou misto, justificando sua resposta.
- 8) Reproduza o circuito abaixo no laboratório remoto e em seguida responda as questões:



- a) Quais chaves foram abertas e quais foram fechadas para a construir o circuito? Preencha a tabela abaixo:

Chave	Condição
1	
2	
3	
4	
5	
6	
7	

- b) Preencha a tabela abaixo com os valores de tensão e corrente em cada um dos quatros medidores:

Medidor	Corrente	Tensão
1		
2		
3		
4		

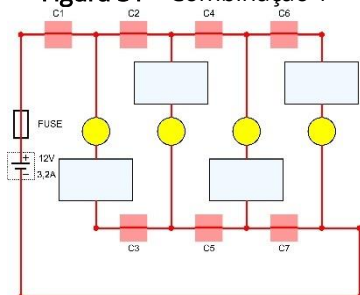
- c) Sabendo que o valor de tensão apresentado no medidor é entre o nó e o terra, qual é a tensão aplicada nas lâmpadas?

Lâmpada	Tensão
1	
2	
3	
4	

- d) Valide o circuito com base na 1ª Lei de Kirchhoff
- e) Valide o circuito com base na 2ª Lei de Kirchhoff
- f) Classifique o circuito em série, paralelo ou misto, justificando sua resposta.

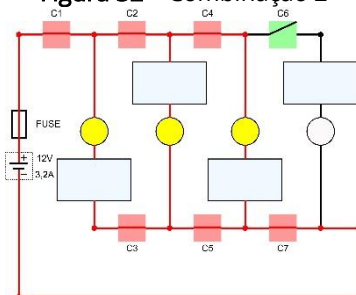
APÊNDICE B – COMBINAÇÕES PARA FORMAÇÃO DE UM CIRCUITO

Figura 31 – Combinação 1



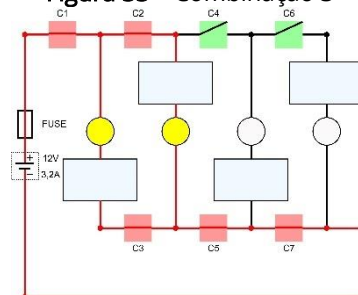
Fonte: Autoria Própria

Figura 32 – Combinação 2



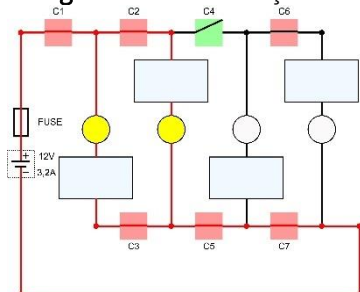
Fonte: Autoria Própria

Figura 33 – Combinação 3



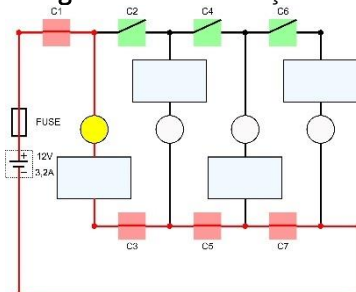
Fonte: Autoria Própria

Figura 34 – Combinação 4



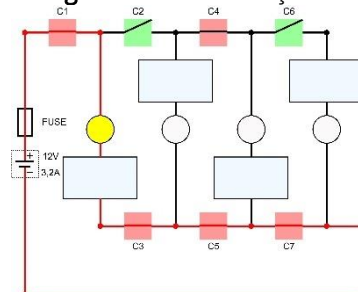
Fonte: Autoria Própria

Figura 35 – Combinação 5



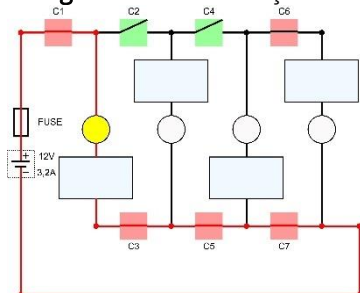
Fonte: Autoria Própria

Figura 36 – Combinação 6



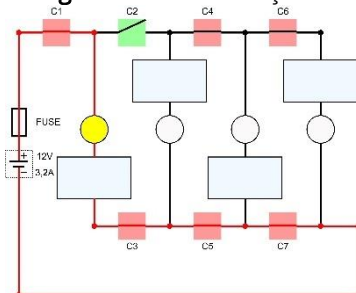
Fonte: Autoria Própria

Figura 37 – Combinação 7



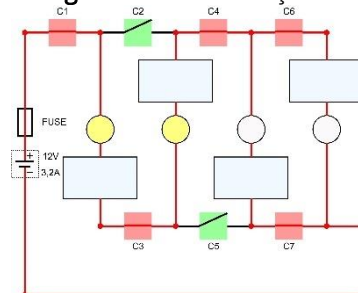
Fonte: Autoria Própria

Figura 38 – Combinação 8



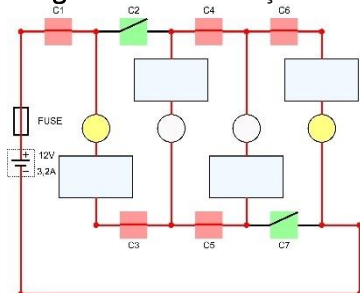
Fonte: Autoria Própria

Figura 39 – Combinação 9



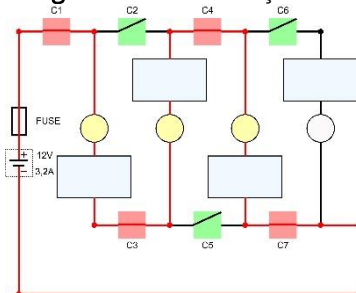
Fonte: Autoria Própria

Figura 40 – Combinação 10



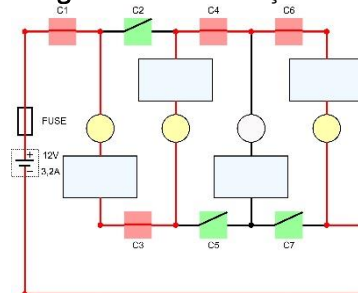
Fonte: Autoria Própria

Figura 41 – Combinação 11

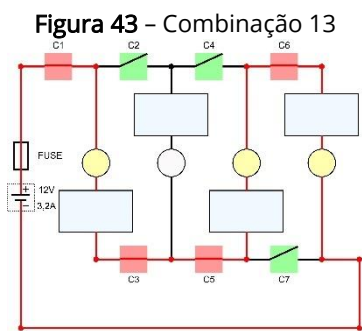


Fonte: Autoria Própria

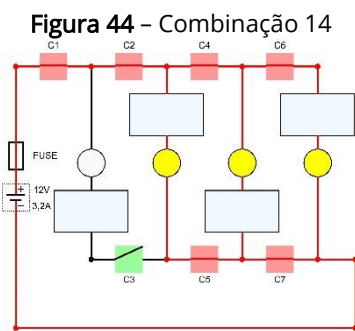
Figura 42 – Combinação 12



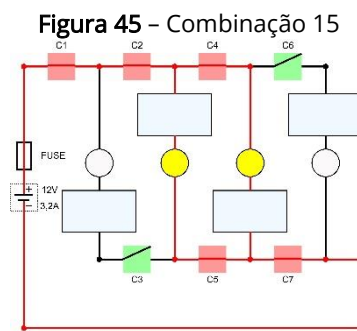
Fonte: Autoria Própria



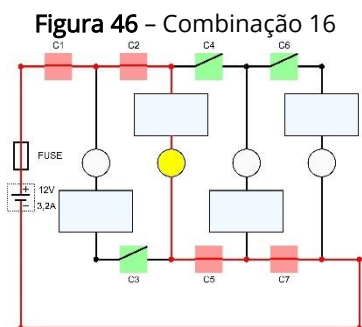
Fonte: Autoria Própria



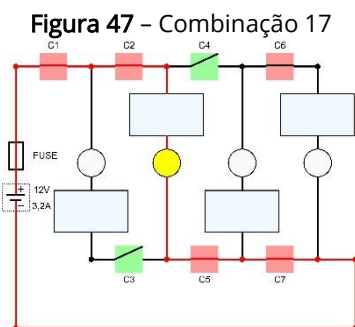
Fonte: Autoria Própria



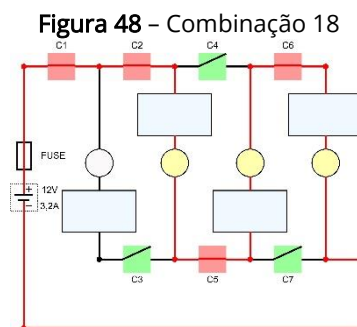
Fonte: Autoria Própria



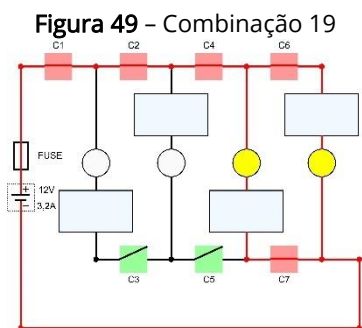
Fonte: Autoria Própria



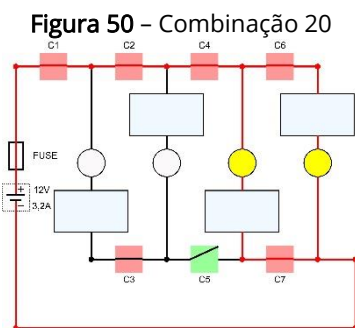
Fonte: Autoria Própria



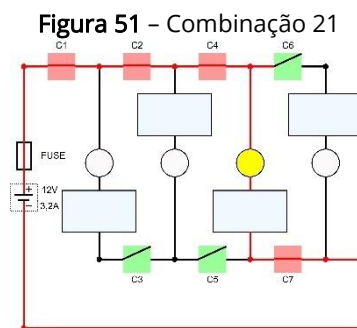
Fonte: Autoria Própria



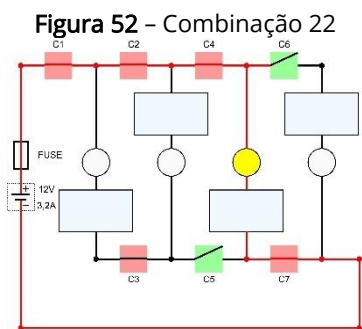
Fonte: Autoria Própria



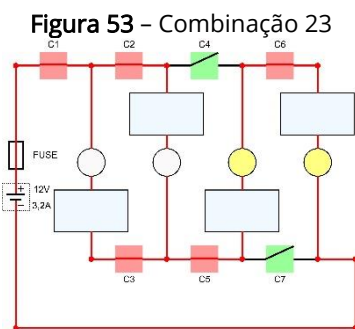
Fonte: Autoria Própria



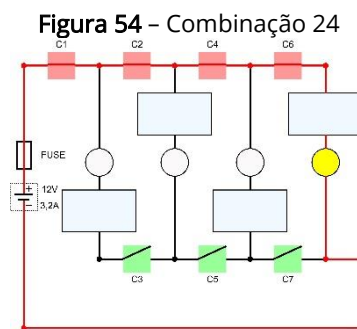
Fonte: Autoria Própria



Fonte: Autoria Própria

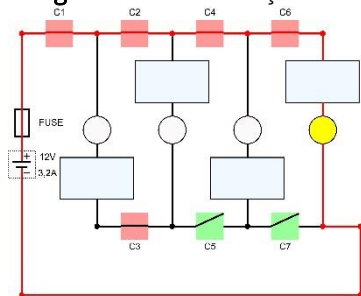


Fonte: Autoria Própria



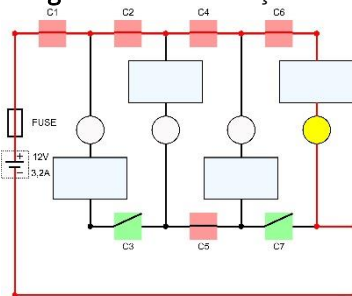
Fonte: Autoria Própria

Figura 55 – Combinação 25



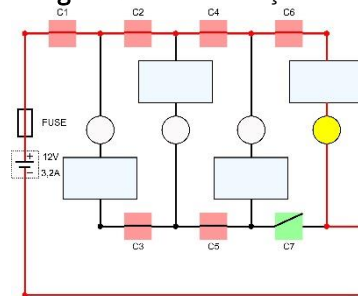
Fonte: Autoria Própria

Figura 56 – Combinação 26



Fonte: Autoria Própria

Figura 57 – Combinação 27



Fonte: Autoria Própria