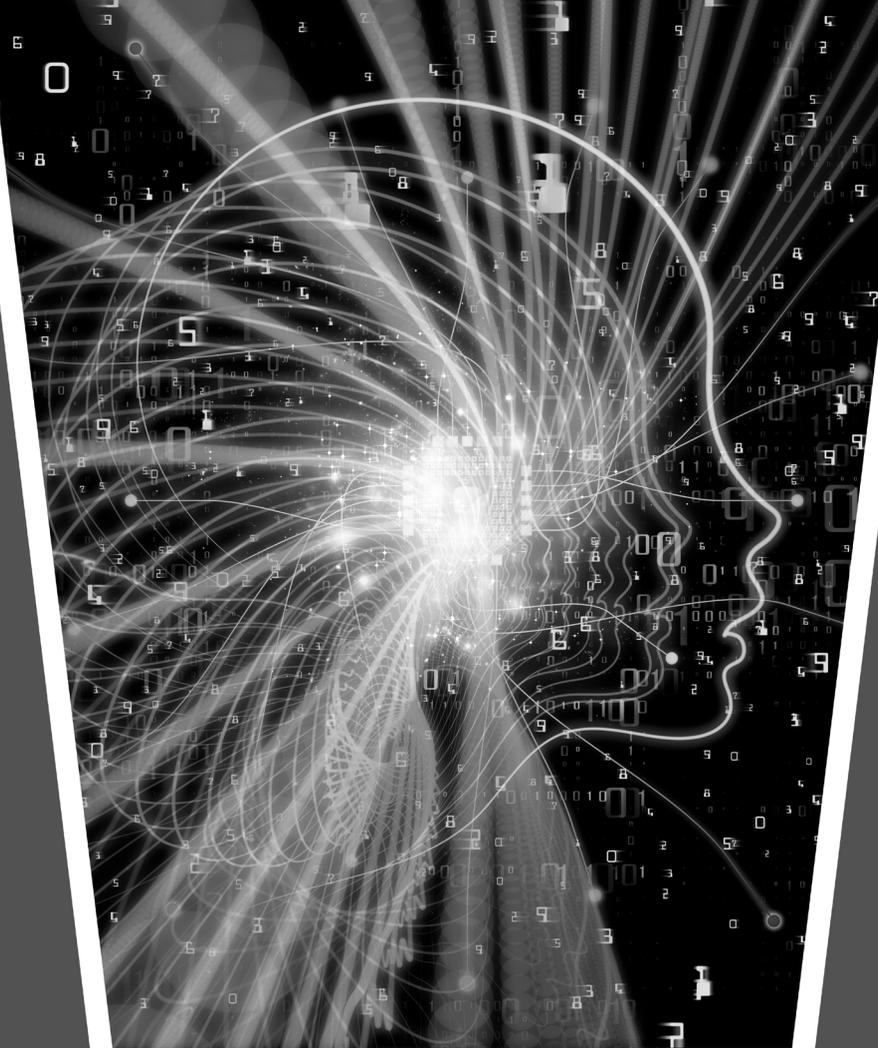


Engenharia Moderna: Soluções para Problemas da Sociedade e da Indústria 2

Filipe Alves Coelho
Monica Tais Siqueira D'Amelio Felipe
Vicente Idalberto Becerra Sablón
(Organizadores)

**Atena**
Editora
Ano 2021



Engenharia Moderna: Soluções para Problemas da Sociedade e da Indústria 2

Filipe Alves Coelho
Monica Tais Siqueira D'Amelio Felipe
Vicente Idalberto Becerra Sablón
(Organizadores)

Atena
Editora

Ano 2021

Editora Chefe

Profª Drª Antonella Carvalho de Oliveira

Assistentes Editoriais

Natalia Oliveira

Bruno Oliveira

Flávia Roberta Barão

Bibliotecária

Janaina Ramos

Projeto Gráfico e Diagramação

Natália Sandrini de Azevedo

Camila Alves de Cremo

Luiza Alves Batista

Maria Alice Pinheiro

Imagens da Capa

Shutterstock

Edição de Arte

Luiza Alves Batista

Revisão

Os Autores

2021 by Atena Editora

Copyright © Atena Editora

Copyright do Texto © 2021 Os autores

Copyright da Edição © 2021 Atena Editora

Direitos para esta edição cedidos à Atena Editora pelos autores.



Todo o conteúdo deste livro está licenciado sob uma Licença de Atribuição *Creative Commons*. Atribuição-Não-Comercial-NãoDerivativos 4.0 Internacional (CC BY-NC-ND 4.0).

O conteúdo dos artigos e seus dados em sua forma, correção e confiabilidade são de responsabilidade exclusiva dos autores, inclusive não representam necessariamente a posição oficial da Atena Editora. Permitido o *download* da obra e o compartilhamento desde que sejam atribuídos créditos aos autores, mas sem a possibilidade de alterá-la de nenhuma forma ou utilizá-la para fins comerciais.

Todos os manuscritos foram previamente submetidos à avaliação cega pelos pares, membros do Conselho Editorial desta Editora, tendo sido aprovados para a publicação com base em critérios de neutralidade e imparcialidade acadêmica.

A Atena Editora é comprometida em garantir a integridade editorial em todas as etapas do processo de publicação, evitando plágio, dados ou resultados fraudulentos e impedindo que interesses financeiros comprometam os padrões éticos da publicação. Situações suspeitas de má conduta científica serão investigadas sob o mais alto padrão de rigor acadêmico e ético.

Conselho Editorial

Ciências Humanas e Sociais Aplicadas

Prof. Dr. Alexandre Jose Schumacher – Instituto Federal de Educação, Ciência e Tecnologia do Paraná

Prof. Dr. Américo Junior Nunes da Silva – Universidade do Estado da Bahia

Prof. Dr. Antonio Carlos Frasson – Universidade Tecnológica Federal do Paraná

Prof. Dr. Antonio Gasparetto Júnior – Instituto Federal do Sudeste de Minas Gerais
Prof. Dr. Antonio Isidro-Filho – Universidade de Brasília
Prof. Dr. Carlos Antonio de Souza Moraes – Universidade Federal Fluminense
Prof. Dr. Crisóstomo Lima do Nascimento – Universidade Federal Fluminense
Prof^ª Dr^ª Cristina Gaio – Universidade de Lisboa
Prof. Dr. Daniel Richard Sant’Ana – Universidade de Brasília
Prof. Dr. Deyvison de Lima Oliveira – Universidade Federal de Rondônia
Prof^ª Dr^ª Dilma Antunes Silva – Universidade Federal de São Paulo
Prof. Dr. Edvaldo Antunes de Farias – Universidade Estácio de Sá
Prof. Dr. Elson Ferreira Costa – Universidade do Estado do Pará
Prof. Dr. Eloi Martins Senhora – Universidade Federal de Roraima
Prof. Dr. Gustavo Henrique Cepolini Ferreira – Universidade Estadual de Montes Claros
Prof^ª Dr^ª Ivone Goulart Lopes – Instituto Internazionele delle Figlie de Maria Ausiliatrice
Prof. Dr. Jadson Correia de Oliveira – Universidade Católica do Salvador
Prof. Dr. Julio Candido de Meirelles Junior – Universidade Federal Fluminense
Prof^ª Dr^ª Lina Maria Gonçalves – Universidade Federal do Tocantins
Prof. Dr. Luis Ricardo Fernandes da Costa – Universidade Estadual de Montes Claros
Prof^ª Dr^ª Natiéli Piovesan – Instituto Federal do Rio Grande do Norte
Prof. Dr. Marcelo Pereira da Silva – Pontifícia Universidade Católica de Campinas
Prof^ª Dr^ª Maria Luzia da Silva Santana – Universidade Federal de Mato Grosso do Sul
Prof^ª Dr^ª Paola Andressa Scortegagna – Universidade Estadual de Ponta Grossa
Prof^ª Dr^ª Rita de Cássia da Silva Oliveira – Universidade Estadual de Ponta Grossa
Prof. Dr. Rui Maia Diamantino – Universidade Salvador
Prof. Dr. Urandi João Rodrigues Junior – Universidade Federal do Oeste do Pará
Prof^ª Dr^ª Vanessa Bordin Viera – Universidade Federal de Campina Grande
Prof. Dr. William Cleber Domingues Silva – Universidade Federal Rural do Rio de Janeiro
Prof. Dr. Willian Douglas Guilherme – Universidade Federal do Tocantins

Ciências Agrárias e Multidisciplinar

Prof. Dr. Alexandre Igor Azevedo Pereira – Instituto Federal Goiano
Prof^ª Dr^ª Carla Cristina Bauermann Brasil – Universidade Federal de Santa Maria
Prof. Dr. Antonio Pasqualetto – Pontifícia Universidade Católica de Goiás
Prof. Dr. Cleberton Correia Santos – Universidade Federal da Grande Dourados
Prof^ª Dr^ª Daiane Garabeli Trojan – Universidade Norte do Paraná
Prof^ª Dr^ª Diocléa Almeida Seabra Silva – Universidade Federal Rural da Amazônia
Prof. Dr. Écio Souza Diniz – Universidade Federal de Viçosa
Prof. Dr. Fábio Steiner – Universidade Estadual de Mato Grosso do Sul
Prof. Dr. Fágner Cavalcante Patrocínio dos Santos – Universidade Federal do Ceará
Prof^ª Dr^ª Girlene Santos de Souza – Universidade Federal do Recôncavo da Bahia
Prof. Dr. Jael Soares Batista – Universidade Federal Rural do Semi-Árido
Prof. Dr. Júlio César Ribeiro – Universidade Federal Rural do Rio de Janeiro
Prof^ª Dr^ª Lina Raquel Santos Araújo – Universidade Estadual do Ceará
Prof. Dr. Pedro Manuel Villa – Universidade Federal de Viçosa
Prof^ª Dr^ª Raissa Rachel Salustriano da Silva Matos – Universidade Federal do Maranhão
Prof. Dr. Ronilson Freitas de Souza – Universidade do Estado do Pará
Prof^ª Dr^ª Talita de Santos Matos – Universidade Federal Rural do Rio de Janeiro
Prof. Dr. Tiago da Silva Teófilo – Universidade Federal Rural do Semi-Árido

Prof. Dr. Valdemar Antonio Paffaro Junior – Universidade Federal de Alfenas

Ciências Biológicas e da Saúde

Prof. Dr. André Ribeiro da Silva – Universidade de Brasília

Prof^ª Dr^ª Anelise Levay Murari – Universidade Federal de Pelotas

Prof. Dr. Benedito Rodrigues da Silva Neto – Universidade Federal de Goiás

Prof^ª Dr^ª Débora Luana Ribeiro Pessoa – Universidade Federal do Maranhão

Prof. Dr. Douglas Siqueira de Almeida Chaves – Universidade Federal Rural do Rio de Janeiro

Prof. Dr. Edson da Silva – Universidade Federal dos Vales do Jequitinhonha e Mucuri

Prof^ª Dr^ª Elizabeth Cordeiro Fernandes – Faculdade Integrada Medicina

Prof^ª Dr^ª Eleuza Rodrigues Machado – Faculdade Anhanguera de Brasília

Prof^ª Dr^ª Elane Schwinden Prudêncio – Universidade Federal de Santa Catarina

Prof^ª Dr^ª Eysler Gonçalves Maia Brasil – Universidade da Integração Internacional da Lusofonia Afro-Brasileira

Prof. Dr. Ferlando Lima Santos – Universidade Federal do Recôncavo da Bahia

Prof. Dr. Fernando Mendes – Instituto Politécnico de Coimbra – Escola Superior de Saúde de Coimbra

Prof^ª Dr^ª Gabriela Vieira do Amaral – Universidade de Vassouras

Prof. Dr. Gianfábio Pimentel Franco – Universidade Federal de Santa Maria

Prof. Dr. Helio Franklin Rodrigues de Almeida – Universidade Federal de Rondônia

Prof^ª Dr^ª Iara Lúcia Tescarollo – Universidade São Francisco

Prof. Dr. Igor Luiz Vieira de Lima Santos – Universidade Federal de Campina Grande

Prof. Dr. Jefferson Thiago Souza – Universidade Estadual do Ceará

Prof. Dr. Jesus Rodrigues Lemos – Universidade Federal do Piauí

Prof. Dr. Jônatas de França Barros – Universidade Federal do Rio Grande do Norte

Prof. Dr. José Max Barbosa de Oliveira Junior – Universidade Federal do Oeste do Pará

Prof. Dr. Luís Paulo Souza e Souza – Universidade Federal do Amazonas

Prof^ª Dr^ª Magnólia de Araújo Campos – Universidade Federal de Campina Grande

Prof. Dr. Marcus Fernando da Silva Praxedes – Universidade Federal do Recôncavo da Bahia

Prof^ª Dr^ª Maria Tatiane Gonçalves Sá – Universidade do Estado do Pará

Prof^ª Dr^ª Mylena Andréa Oliveira Torres – Universidade Ceuma

Prof^ª Dr^ª Natiéli Piovesan – Instituto Federaci do Rio Grande do Norte

Prof. Dr. Paulo Inada – Universidade Estadual de Maringá

Prof. Dr. Rafael Henrique Silva – Hospital Universitário da Universidade Federal da Grande Dourados

Prof^ª Dr^ª Regiane Luz Carvalho – Centro Universitário das Faculdades Associadas de Ensino

Prof^ª Dr^ª Renata Mendes de Freitas – Universidade Federal de Juiz de Fora

Prof^ª Dr^ª Vanessa Lima Gonçalves – Universidade Estadual de Ponta Grossa

Prof^ª Dr^ª Vanessa Bordin Viera – Universidade Federal de Campina Grande

Ciências Exatas e da Terra e Engenharias

Prof. Dr. Adélio Alcino Sampaio Castro Machado – Universidade do Porto

Prof. Dr. Carlos Eduardo Sanches de Andrade – Universidade Federal de Goiás

Prof^ª Dr^ª Carmen Lúcia Voigt – Universidade Norte do Paraná

Prof. Dr. Cleiseano Emanuel da Silva Paniagua – Instituto Federal de Educação, Ciência e Tecnologia de Goiás

Prof. Dr. Douglas Gonçalves da Silva – Universidade Estadual do Sudoeste da Bahia

Prof. Dr. Eloi Rufato Junior – Universidade Tecnológica Federal do Paraná
Prof^ª Dr^ª Érica de Melo Azevedo – Instituto Federal do Rio de Janeiro
Prof. Dr. Fabrício Menezes Ramos – Instituto Federal do Pará
Prof^ª Dra. Jéssica Verger Nardeli – Universidade Estadual Paulista Júlio de Mesquita Filho
Prof. Dr. Juliano Carlo Rufino de Freitas – Universidade Federal de Campina Grande
Prof^ª Dr^ª Luciana do Nascimento Mendes – Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte
Prof. Dr. Marcelo Marques – Universidade Estadual de Maringá
Prof. Dr. Marco Aurélio Kistemann Junior – Universidade Federal de Juiz de Fora
Prof^ª Dr^ª Neiva Maria de Almeida – Universidade Federal da Paraíba
Prof^ª Dr^ª Natiéli Piovesan – Instituto Federal do Rio Grande do Norte
Prof^ª Dr^ª Priscila Tessmer Scaglioni – Universidade Federal de Pelotas
Prof. Dr. Takeshy Tachizawa – Faculdade de Campo Limpo Paulista

Linguística, Letras e Artes

Prof^ª Dr^ª Adriana Demite Stephani – Universidade Federal do Tocantins
Prof^ª Dr^ª Angeli Rose do Nascimento – Universidade Federal do Estado do Rio de Janeiro
Prof^ª Dr^ª Carolina Fernandes da Silva Mandaji – Universidade Tecnológica Federal do Paraná
Prof^ª Dr^ª Denise Rocha – Universidade Federal do Ceará
Prof. Dr. Fabiano Tadeu Grazioli – Universidade Regional Integrada do Alto Uruguai e das Missões
Prof. Dr. Gilmei Fleck – Universidade Estadual do Oeste do Paraná
Prof^ª Dr^ª Keyla Christina Almeida Portela – Instituto Federal de Educação, Ciência e Tecnologia do Paraná
Prof^ª Dr^ª Miranilde Oliveira Neves – Instituto de Educação, Ciência e Tecnologia do Pará
Prof^ª Dr^ª Sandra Regina Gardacho Pietrobon – Universidade Estadual do Centro-Oeste
Prof^ª Dr^ª Sheila Marta Carregosa Rocha – Universidade do Estado da Bahia

Conselho Técnico Científico

Prof. Me. Abrãao Carvalho Nogueira – Universidade Federal do Espírito Santo
Prof. Me. Adalberto Zorzo – Centro Estadual de Educação Tecnológica Paula Souza
Prof. Dr. Adaylson Wagner Sousa de Vasconcelos – Ordem dos Advogados do Brasil/Seccional Paraíba
Prof. Dr. Adilson Tadeu Basquerote Silva – Universidade para o Desenvolvimento do Alto Vale do Itajaí
Prof. Dr. Alex Luis dos Santos – Universidade Federal de Minas Gerais
Prof. Me. Alexandro Teixeira Ribeiro – Centro Universitário Internacional
Prof^ª Ma. Aline Ferreira Antunes – Universidade Federal de Goiás
Prof. Me. André Flávio Gonçalves Silva – Universidade Federal do Maranhão
Prof^ª Ma. Andréa Cristina Marques de Araújo – Universidade Fernando Pessoa
Prof^ª Dr^ª Andreza Lopes – Instituto de Pesquisa e Desenvolvimento Acadêmico
Prof^ª Dr^ª Andrezza Miguel da Silva – Faculdade da Amazônia
Prof^ª Ma. Anelisa Mota Gregoleti – Universidade Estadual de Maringá
Prof^ª Ma. Anne Karynne da Silva Barbosa – Universidade Federal do Maranhão
Prof. Dr. Antonio Hot Pereira de Faria – Polícia Militar de Minas Gerais
Prof. Me. Armando Dias Duarte – Universidade Federal de Pernambuco
Prof^ª Ma. Bianca Camargo Martins – UniCesumar

Profª Ma. Carolina Shimomura Nanya – Universidade Federal de São Carlos
Prof. Me. Carlos Antônio dos Santos – Universidade Federal Rural do Rio de Janeiro
Prof. Me. Christopher Smith Bignardi Neves – Universidade Federal do Paraná
Prof. Ma. Cláudia de Araújo Marques – Faculdade de Música do Espírito Santo
Profª Drª Cláudia Taís Siqueira Cagliari – Centro Universitário Dinâmica das Cataratas
Prof. Me. Clécio Danilo Dias da Silva – Universidade Federal do Rio Grande do Norte
Prof. Me. Daniel da Silva Miranda – Universidade Federal do Pará
Profª Ma. Daniela da Silva Rodrigues – Universidade de Brasília
Profª Ma. Daniela Remião de Macedo – Universidade de Lisboa
Profª Ma. Dayane de Melo Barros – Universidade Federal de Pernambuco
Prof. Me. Douglas Santos Mezacas – Universidade Estadual de Goiás
Prof. Me. Edevaldo de Castro Monteiro – Embrapa Agrobiologia
Prof. Me. Eduardo Gomes de Oliveira – Faculdades Unificadas Doctum de Cataguases
Prof. Me. Eduardo Henrique Ferreira – Faculdade Pitágoras de Londrina
Prof. Dr. Edwaldo Costa – Marinha do Brasil
Prof. Me. Eliel Constantino da Silva – Universidade Estadual Paulista Júlio de Mesquita
Prof. Me. Ernane Rosa Martins – Instituto Federal de Educação, Ciência e Tecnologia de Goiás
Prof. Me. Euvaldo de Sousa Costa Junior – Prefeitura Municipal de São João do Piauí
Prof. Dr. Everaldo dos Santos Mendes – Instituto Edith Theresa Hedwing Stein
Prof. Me. Ezequiel Martins Ferreira – Universidade Federal de Goiás
Profª Ma. Fabiana Coelho Couto Rocha Corrêa – Centro Universitário Estácio Juiz de Fora
Prof. Me. Fabiano Eloy Atilio Batista – Universidade Federal de Viçosa
Prof. Me. Felipe da Costa Negrão – Universidade Federal do Amazonas
Prof. Me. Francisco Odécio Sales – Instituto Federal do Ceará
Profª Drª Germana Ponce de Leon Ramírez – Centro Universitário Adventista de São Paulo
Prof. Me. Gevair Campos – Instituto Mineiro de Agropecuária
Prof. Me. Givanildo de Oliveira Santos – Secretaria da Educação de Goiás
Prof. Dr. Guilherme Renato Gomes – Universidade Norte do Paraná
Prof. Me. Gustavo Krahl – Universidade do Oeste de Santa Catarina
Prof. Me. Helton Rangel Coutinho Junior – Tribunal de Justiça do Estado do Rio de Janeiro
Profª Ma. Isabelle Cerqueira Sousa – Universidade de Fortaleza
Profª Ma. Jaqueline Oliveira Rezende – Universidade Federal de Uberlândia
Prof. Me. Javier Antonio Albornoz – University of Miami and Miami Dade College
Prof. Me. Jhonatan da Silva Lima – Universidade Federal do Pará
Prof. Dr. José Carlos da Silva Mendes – Instituto de Psicologia Cognitiva, Desenvolvimento Humano e Social
Prof. Me. Jose Elyton Batista dos Santos – Universidade Federal de Sergipe
Prof. Me. José Luiz Leonardo de Araujo Pimenta – Instituto Nacional de Investigación Agropecuaria Uruguay
Prof. Me. José Messias Ribeiro Júnior – Instituto Federal de Educação Tecnológica de Pernambuco
Profª Drª Juliana Santana de Curcio – Universidade Federal de Goiás
Profª Ma. Juliana Thaisa Rodrigues Pacheco – Universidade Estadual de Ponta Grossa
Profª Drª Kamilly Souza do Vale – Núcleo de Pesquisas Fenomenológicas/UFPA
Prof. Dr. Kárpio Márcio de Siqueira – Universidade do Estado da Bahia
Profª Drª Karina de Araújo Dias – Prefeitura Municipal de Florianópolis
Prof. Dr. Lázaro Castro Silva Nascimento – Laboratório de Fenomenologia & Subjetividade/UFPR

Prof. Me. Leonardo Tullio – Universidade Estadual de Ponta Grossa
Prof^ª Ma. Lillian Coelho de Freitas – Instituto Federal do Pará
Prof^ª Ma. Liliansi Aparecida Sereno Fontes de Medeiros – Consórcio CEDERJ
Prof^ª Dr^ª Lívia do Carmo Silva – Universidade Federal de Goiás
Prof. Dr. Lucio Marques Vieira Souza – Secretaria de Estado da Educação, do Esporte e da Cultura de Sergipe
Prof. Dr. Luan Vinicius Bernardelli – Universidade Estadual do Paraná
Prof^ª Ma. Luana Ferreira dos Santos – Universidade Estadual de Santa Cruz
Prof^ª Ma. Luana Vieira Toledo – Universidade Federal de Viçosa
Prof. Me. Luis Henrique Almeida Castro – Universidade Federal da Grande Dourados
Prof^ª Ma. Luma Sarai de Oliveira – Universidade Estadual de Campinas
Prof. Dr. Michel da Costa – Universidade Metropolitana de Santos
Prof. Me. Marcelo da Fonseca Ferreira da Silva – Governo do Estado do Espírito Santo
Prof. Dr. Marcelo Máximo Purificação – Fundação Integrada Municipal de Ensino Superior
Prof. Me. Marcos Aurelio Alves e Silva – Instituto Federal de Educação, Ciência e Tecnologia de São Paulo
Prof^ª Ma. Maria Elanny Damasceno Silva – Universidade Federal do Ceará
Prof^ª Ma. Marileila Marques Toledo – Universidade Federal dos Vales do Jequitinhonha e Mucuri
Prof. Me. Pedro Panhoca da Silva – Universidade Presbiteriana Mackenzie
Prof^ª Dr^ª Poliana Arruda Fajardo – Universidade Federal de São Carlos
Prof. Me. Ricardo Sérgio da Silva – Universidade Federal de Pernambuco
Prof. Me. Renato Faria da Gama – Instituto Gama – Medicina Personalizada e Integrativa
Prof^ª Ma. Renata Luciane Polsaque Young Blood – UniSecal
Prof. Me. Robson Lucas Soares da Silva – Universidade Federal da Paraíba
Prof. Me. Sebastião André Barbosa Junior – Universidade Federal Rural de Pernambuco
Prof^ª Ma. Silene Ribeiro Miranda Barbosa – Consultoria Brasileira de Ensino, Pesquisa e Extensão
Prof^ª Ma. Solange Aparecida de Souza Monteiro – Instituto Federal de São Paulo
Prof^ª Ma. Taiane Aparecida Ribeiro Nepomoceno – Universidade Estadual do Oeste do Paraná
Prof. Me. Tallys Newton Fernandes de Matos – Faculdade Regional Jaguaribana
Prof^ª Ma. Thatianny Jasmine Castro Martins de Carvalho – Universidade Federal do Piauí
Prof. Me. Tiago Silvio Dedoné – Colégio ECEL Positivo
Prof. Dr. Welleson Feitosa Gazel – Universidade Paulista

Editora Chefe: Profª Drª Antonella Carvalho de Oliveira
Bibliotecária: Janaina Ramos
Diagramação: Camila Alves de Cremo
Correção: Giovanna Sandrini de Azevedo
Edição de Arte: Luiza Alves Batista
Revisão: Os Autores
Organizadores: Filipe Alves Coelho
 Monica Tais Siqueira D'amelio Felipe
 Vicente Idalberto Becerra Sablón

Dados Internacionais de Catalogação na Publicação (CIP)

E57 Engenharia moderna: soluções para problemas da sociedade e da indústria 2 / Organizadores Filipe Alves Coelho, Monica Tais Siqueira D'amelio Felipe, Vicente Idalberto Becerra Sablón. – Ponta Grossa - PR: Atena, 2021.

Formato: PDF
 Requisitos de sistema: Adobe Acrobat Reader
 Modo de acesso: World Wide Web
 Inclui bibliografia
 ISBN 978-65-5706-999-8
 DOI 10.22533/at.ed.998211304

1. Engenharia. I. I. Coelho, Filipe Alves (Organizador). II. Felipe, Monica Tais Siqueira D'amelio (Organizadora). III. Sablón, Vicente Idalberto Becerra (Organizador). IV. Título.
 CDD 620

Elaborado por Bibliotecária Janaina Ramos – CRB-8/9166

Atena Editora
 Ponta Grossa – Paraná – Brasil
 Telefone: +55 (42) 3323-5493
www.atenaeditora.com.br
contato@atenaeditora.com.br

DECLARAÇÃO DOS AUTORES

Os autores desta obra: 1. Atestam não possuir qualquer interesse comercial que constitua um conflito de interesses em relação ao artigo científico publicado; 2. Declaram que participaram ativamente da construção dos respectivos manuscritos, preferencialmente na: a) Concepção do estudo, e/ou aquisição de dados, e/ou análise e interpretação de dados; b) Elaboração do artigo ou revisão com vistas a tornar o material intelectualmente relevante; c) Aprovação final do manuscrito para submissão.; 3. Certificam que os artigos científicos publicados estão completamente isentos de dados e/ou resultados fraudulentos; 4. Confirmam a citação e a referência correta de todos os dados e de interpretações de dados de outras pesquisas; 5. Reconhecem terem informado todas as fontes de financiamento recebidas para a consecução da pesquisa.

APRESENTAÇÃO

A ciência tenta obter conhecimento sobre a estrutura fundamental do mundo utilizando observações sistemáticas e experimentais. A engenharia explora o campo do desconhecido procurando sistematicamente por novas soluções para problemas práticos. O GPS, a Internet, antibióticos, dentre outros, surgiram em meio às dificuldades das guerras. O Brasil, apesar de não estar envolvido em nenhuma, vive outras batalhas diárias.

No primeiro volume deste livro trouxemos um pouco da produção científica de um grupo de pesquisadores da região de Campinas e neste novo volume, não diferente, apresentamos mais engenharia e ciência aos serviços da sociedade e da indústria. Entretanto, desta vez a produção ocorreu durante um dos eventos de mudança mais rápida observada na sociedade recente: a quarentena imposta pela pandemia de COVID-19.

O ano de 2020 será lembrado por todos como o ano mais atípico das nossas vidas. O distanciamento social afastou pesquisadores do contato diário com colegas e de seus materiais de trabalho. Pesquisar de casa parecia impossível. Vimos ao longo de 2020 que nossos alunos conseguiam fazer pesquisa nas empresas que trabalhavam. Que, com os devidos cuidados, poderíamos usar os laboratórios. Que a internet aproximou os distantes grupos de pesquisa. Que ciência se faz com pessoas dedicadas e apaixonadas pelo trabalho.

Pesquisamos. E este livro é a amálgama do árduo trabalho de produzir ciência e tecnologia em 2020. É a flor do mandacaru: aos olhos de quem vê, surgiu no ambiente aparentemente improvável e inóspito. O ano que passou fortaleceu nosso grupo de pesquisa e parcerias foram criadas e/ou fortalecidas. Reforçamos, porém, que este livro está mais para um *tweet* diante do livro que foi 2020. Um ano longo, com muito aprendizado, muitas quebras de paradigmas e que de certa maneira, parece ainda insistir em estar entre nós. Este livro foi um recorte das nossas vidas acadêmicas, uma lembrança que será registrada nos anais da academia, mas com significado muito particular para cada um dos autores que aqui depositaram as lembranças do que melhor fizeram neste período.

O ano que se adentra rapidamente traz a esperança de renovação, de mudanças não mais tão bruscas e de um ano que se inicia em regime laminar. E nesta correnteza que é a vida, celebramos neste volume trabalhos que envolvem inteligência artificial aplicada (inclusive para a COVID-19), aplicação ou desenvolvimento de materiais, melhorias de processos industriais e da gestão de linhas de produção, geração de energia, dentre outros temas.

Finalmente, agradecemos a Editora Atena por abraçar esta iniciativa, abrindo as portas para a divulgação do conhecimento para a comunidade científica e a sociedade.

Filipe Alves Coelho

Monica Tais Siqueira D'Amelio

Vicente Idalberto Becerra Sablón

SUMÁRIO

CAPÍTULO 1..... 1

THE INFLUENCE OF MEDICAL IMAGE ANALYSIS FOR COVID-19 AS A TECHNOLOGICAL MECHANISM TO SUPPORT THE GLOBAL PANDEMIC

Ana Carolina Borges Monteiro
Reinaldo Padilha França
Rangel Arthur
Giulliano Paes Carnielli
Vicente Idalberto Becerra Sablón
Yuzo Iano

DOI 10.22533/at.ed.9982113041

CAPÍTULO 2..... 11

THE IMPACT OF COMPUTATIONAL INTELLIGENCE FOR COVID-19 AS A TECHNOLOGICAL RESOURCE TO SUPPORT THE GLOBAL PANDEMIC

Reinaldo Padilha França
Ana Carolina Borges Monteiro
Rangel Arthur
Andrea Coimbra Segatti
Vicente Idalberto Becerra Sablón
Yuzo Iano

DOI 10.22533/at.ed.9982113042

CAPÍTULO 3..... 21

***MACHINE LEARNING* PARA DELINEAMENTO EXPERIMENTAL EM ESTUDOS DA DOR - *IOT*, REDE NEURAL, *K-MEANS* E ÁRVORE DE DECISÃO**

Fábio Andrijauskas
Glaucilene Ferreira Catroli
Eduardo Keizo Horibe Junior
Matheus Gaboardi Tralli
Rafael Soares Torres
João Marcos Santos

DOI 10.22533/at.ed.9982113043

CAPÍTULO 4..... 33

RASTREX – SISTEMA DE RASTREAMENTO VEICULAR

Sergio Henrique Matukava
Vinicius Stanisoski Perassolli
Vicente Idalberto Becerra Sablón
Annete Silva Faesarella

DOI 10.22533/at.ed.9982113044

CAPÍTULO 5	47
AMBIENTE DE APRENDIZADO PARA ESTUDO DE MÁQUINAS VIRTUAIS EM SISTEMA EMBARCADO	
Renan Romão Oliveira Regimar Francisco dos Santos Glaucilene Ferreira Catroli Fábio Andrijauskas	
DOI 10.22533/at.ed.9982113045	
CAPÍTULO 6	58
GERADOR DE ENERGIA PIEZOELÉTRICO: AQUISIÇÃO, MONITORAMENTO E CONDICIONAMENTO DO SINAL GERADO	
Darilson Francisco das Dores Antunes Vicente Idalberto Becerra Sablón	
DOI 10.22533/at.ed.9982113046	
CAPÍTULO 7	70
SUORTE PARA MÓDULO FOTOVOLTAICO COM INCLINAÇÃO VARIÁVEL	
Felipe de Marco Costa Rafael Aparecido Bragante Annete Silva Faesarella Filipe Alves Coelho	
DOI 10.22533/at.ed.9982113047	
CAPÍTULO 8	83
VIABILIZAÇÃO DO USO DE MANUFATURA ADITIVA NOS PROCESSOS DE AGITAÇÃO E MISTURA	
Tadeu Henrique Aparecido da Silva Mateus Bueno Veris Monica Tais Siqueira D'Amelio	
DOI 10.22533/at.ed.9982113048	
CAPÍTULO 9	95
MODELAGEM E SIMULAÇÃO DO PROCESSO DE FERMENTAÇÃO CONTÍNUA EM MICRO BIORREATOR	
João Paulo Fioritti Godoy Guilherme Brandão Silva Filipe Alves Coelho	
DOI 10.22533/at.ed.9982113049	
CAPÍTULO 10	107
CELULOSE NANOFIBRILADA: ESTUDO DA OBTENÇÃO E APLICAÇÃO NA INDÚSTRIA PAPELEIRA	
Marcela Renata Zenni	

Caroline Pereira dos Santos
Roberta Martins da Costa Bianchi

DOI 10.22533/at.ed.99821130410

CAPÍTULO 11..... 120

DESENVOLVIMENTO DE BIOPOLÍMERO A PARTIR DO AMIDO DE CHUCHU E AVALIAÇÃO DA INCORPORAÇÃO DO RESÍDUO DE CAFÉ E ÓLEO DE BURITI

Fernanda Andrade Tigre da Costa
Jairo Paschoal Júnior
Rosana Zanetti Baú

DOI 10.22533/at.ed.99821130411

CAPÍTULO 12..... 135

ROLHA DE RESÍDUO: A INOVAÇÃO A PARTIR DO DESCARTE DE *PALLETS*

Laura Bisetto Zanella
Liliani Alves da Silva
Tainah Cristina Cunha Muner
Monica Tais Siqueira D'Amelio

DOI 10.22533/at.ed.99821130412

CAPÍTULO 13..... 148

PRODUÇÃO DE COSMECÊUTICOS COM ÓLEO DE CAFÉ PARA PREVENÇÃO DO FOTOENVELHECIMENTO

Vanessa Cristina de Barros Mariano
Natália Cristina de Brito Lopes
Iara Lúcia Tescarollo

DOI 10.22533/at.ed.99821130413

CAPÍTULO 14..... 161

SMLP - SISTEMA DE MONITORAMENTO DE LINHA DE PRODUÇÃO

Igor Vieira Lima
Kaique Franco Jarussi
Annete Silva Faesarella
Vicente Idalberto Becerra Sablón

DOI 10.22533/at.ed.99821130414

CAPÍTULO 15..... 174

SISTEMA DE MICRODRENAGEM

Beatriz de Souza Elias
Luiz Henrique Mascaro de Mendonça
Cristina das Graças Fassina
Renata Lima Moretto

DOI 10.22533/at.ed.99821130415

CAPÍTULO 16	187
CASCA DE BANANA COMO BIOADSORVEDOR DE PIGMENTOS DE MEIO AQUOSO	
Gláucia Rodrigues	
Brenda Gabriela	
Monica Tais Siqueira D'Amelio Felipe	
DOI 10.22533/at.ed.99821130416	
CAPÍTULO 17	199
MINIMIZAÇÃO DE SOBRECARGA ESTRUTURAL NA BLINDAGEM DA RADIOATIVIDADE	
André Augusto Gutierrez Fernandes Beati	
Heitor Berger Campos	
Angela Aparecida Brandão	
Natália Ribeiro da Silva	
DOI 10.22533/at.ed.99821130417	
SOBRE OS ORGANIZADORES	220
ÍNDICE REMISSIVO	221

AMBIENTE DE APRENDIZADO PARA ESTUDO DE MAQUINAS VIRTUAIS EM SISTEMA EMBARCADO

Data de aceite: 16/03/2021

Data de submissão: 15/01/2021

Renan Romão Oliveira

Universidade São Francisco
Engenharia de Computação
Itatiba – SP

<http://lattes.cnpq.br/4582228530919310>

Regimar Francisco dos Santos

Universidade São Francisco
Engenharia de Computação
Itatiba - SP

<http://lattes.cnpq.br/7602724854566864>

Glaucilene Ferreira Catroli

UNICAMP – Instituto de Biologia
Campinas - SP

<http://lattes.cnpq.br/4914553972592247>

Fábio Andrijauskas

Universidade São Francisco
Engenharia de Computação
Itatiba - SP

<http://lattes.cnpq.br/7771878233635494>

RESUMO: Computadores são máquinas utilizadas para a execução das mais variadas atividades, porém essas máquinas têm um funcionamento relativamente complexo, o que acaba por distanciar indivíduos que não são da área de computação e obriga a aquisição de conhecimento específico sobre o funcionamento e programação dessas máquinas. Porém, tais máquinas são divididas em diferentes níveis de abstração em sua construção, e têm sua

complexidade mitigada quando analisadas em níveis mais baixos, o que pode ser um ótimo artifício didático, principalmente se considerasse a ausência de ferramentas consagradas no mercado atual. O presente trabalho visa o desenvolvimento de uma máquina virtual, com linguagem de máquina própria, didática e que possibilite o aprendizado direcionado ao desenvolvimento de *software* e arquitetura de computadores por leigos ou iniciantes da área, partindo de um ambiente de baixo nível e, conseqüentemente, de menor complexidade. Para tal, será desenvolvido um interpretador que terá como plataforma de execução um sistema embarcado de baixo custo, mais especificamente um microcontrolador ESP32. A aplicação desenvolvida será de código aberto, permitindo que terceiros desenvolvam extensões e melhorias, mantendo o ambiente atualizado.

PALAVRAS-CHAVE: Desenvolvimento de software, arquitetura de computadores, compilador, máquina virtual, sistema embarcado.

LEARNING ENVIRONMENT FOR STUDYING VIRTUAL MACHINES IN THE EMBEDDED SYSTEM

ABSTRACT: Computers are machines used to execute a great variety of tasks but these machines have a relatively complex operation, which ends up distancing individuals who are not from the computing area and forces the acquisition of specific knowledge about the functioning and programming of these machines. However, these machines are divided into different levels and have their complexity mitigated when analyzed at lower levels, this can be a great didactic artifice, in addition to the fact that there are no established tools in the market. The present work aims the development of a virtual machine

with its machine language, didactic and provide the learning of software development and computer architecture for beginners, starting from a low level and, consequently, less complex environment. An interpreter will be developed with this purpose, it will have a low-cost embedded system as its execution platform, an ESP32 microcontroller. The application developed will also be open-source, allowing others to develop extensions and improvements, keeping the environment updated.

KEYWORDS: Software development, computer architecture compiler, virtual machine, embedded system.

1 | INTRODUÇÃO

Computadores modernos são máquinas muito complexas por conta da grande quantidade de componentes internos e das tecnologias avançadas envolvidas em seu funcionamento. Ao longo do tempo, cientistas da computação desenvolveram teorias e modelos para descrever o funcionamento de tais máquinas, permitindo sua utilização ou mesmo a programação, por usuários que não possuem conhecimento específico na área. De forma resumida, Tanenbaum *et al.* (2013) descrevem os computadores como máquinas que podem resolver problemas para as pessoas executando instruções que lhes são dadas. Tal afirmação deixa claro como a codificação de programas e sua execução são processos essenciais dentro do campo da computação, porém o funcionamento de certa forma mistificado destas máquinas, e a forma como executam instruções de código, faz com que leigos ou mesmo profissionais de campos relacionados à tecnologia, não demonstrem interesse pelo desenvolvimento de *software*. Cria-se assim, o estigma de uma área complexa, uma vez que existem poucas ferramentas capazes de elevar o aprendizado de programação do nível mais básico para o intermediário. Existem muitas aplicações voltadas para programadores iniciantes com código em bloco, jogos com mecânicas baseadas em pseudocódigo ou ferramentas para usuários mais avançados, como linguagens de programação de sintaxe simplificada, no entanto, tais opções não possibilitam que um usuário básico/intermediário produza uma aplicação funcional, algo crucial para deixá-lo engajado.

Com isso, apresenta-se uma máquina virtual, com linguagem de máquina própria visando, de maneira simples e didática, o desenvolvimento e execução de aplicações que conseguem descrever o processo de computador. Tal produção foi constituída de um interpretador que será uma máquina teórica e servirá de ambiente de execução para seu próprio código de máquina. Essa máquina será executada em um ESP32, um sistema embarcado de baixo custo, com baixo consumo de energia e um número considerável de interfaces de entrada e saída, como terminais digitais e analógicos, compatibilidade nativa com protocolos de comunicação variados e até mesmo interfaces de comunicação sem fio. Tais características permitem ao usuário o desenvolvimento de aplicativos que realizam interações físicas, a compreensão do funcionamento de um computador digital a partir dos níveis mais básicos e a evolução gradativa do seu conhecimento na área.

2 | AMBIENTE DE APRENDIZADO PARA ESTUDO DE MÁQUINAS VIRTUAIS EM SISTEMA EMBARCADO

Como especificado na teoria, o sistema desenvolvido tem como *hardware* de execução um sistema embarcado, optou-se por utilizar um microcontrolador da série ESP32, produzido pela empresa chinesa *Espressif Systems* e voltado especificamente para aplicações de IoT (*Internet of Things*). Os microcontroladores são equipamentos programáveis de pequeno porte e de baixo custo, desenvolvidos para controlar ações e eventos. São compostos, basicamente, por uma unidade processadora, memórias, entradas e saídas, controle temporal e conversores analógicos e digitais. Esse tipo de dispositivo, segundo Silva (2012), surgiu em meados da década de 1970, criado por uma equipe da *Texas Instruments*, derivado dos então microprocessadores criados para realizar cálculos e, posteriormente, tomada de decisões. No desenvolvimento dos microcontroladores, foram incorporados memórias e outros componentes através de circuitos integrados, possibilitando uma evolução que resultou nos equipamentos que conhecemos atualmente.

A característica integradora dos microcontroladores, ou seja, ligações com meios externos através de entradas e saídas, os torna muito práticos para a execução de funções complexas, principalmente na automação ou automatização. Sua conectividade também é algo fundamental para explicar a grande difusão e aplicações desses aparatos, que têm capacidade de atuação na internet e tornam-se a melhor opção para o desenvolvimento de um sistema de controle e monitoramento integrado com outros dispositivos. Com todos os recursos, um microcontrolador pode atuar como *host* para uma página *web* entre outras aplicações (Santos e Junior, 2019).

O ESP32 é amplamente utilizado para desenvolvimento de sistemas embarcados móveis, por possuir uma arquitetura *RISC* (*acrônimo de Reduced Instruction Set Computer*) de 32 *bits*, microcontrolador *dual-core* com *clock* máximo de 240 MHz com baixo consumo e uma excelente performance. Possui grande capacidade de memória ROM de 448 KB usada para *boot* e funções principais mais memória SRAM de 520 KB usadas para dados e instruções de programas. O ESP32 possui muitos recursos o que o torna interessante para IoT, tais como: 34 GPIO, 3 SPI, 2 I2S, 18 canais ADC, 3 UART, 10 pinos de leitura capacitiva e PWM.

A linguagem de programação padrão utilizada para desenvolvimento do ESP32 com o ESP-IDF, e neste trabalho, é a linguagem C, com sintaxe e funcionalidades familiares para a maioria dos programadores e eficiência de execução por ser compilada e gerar código de máquina otimizado. Apesar de ser considerada uma linguagem de programação de alto nível, a linguagem C permite a realização de operações de baixo nível, como manuseio de ponteiros e alocação dinâmica de memória que, apesar de serem funcionalidades relativamente avançadas da linguagem, estão presentes em sua biblioteca padrão e são essenciais para o desenvolvimento de uma máquina virtual.

Uma vez que com a linguagem C é possível criar estruturas de dados customizadas (*structs*) para o armazenamento dos dados necessários na execução dos programas de maneira geral. A linguagem C é rápida e sem gerar muitos metadados, dados utilizados para manuseamento do programa, otimizando a utilização da memória que é um recurso um pouco escasso em um sistema embarcado, além de permitir futuras otimizações de

desempenho e alterações de design mais facilmente. Em contrapartida essa abordagem faz com que seja necessária a verificação dos recursos utilizados e disponíveis, para não causar erros como um estouro da pilha de execução.

3 | APLICAÇÃO E UTILIZAÇÃO

A arquitetura desta máquina está exemplificada no diagrama da Figura 1, na qual é possível observar os principais componentes que fazem parte desta: os registradores, utilizados para armazenamento de valores temporários durante a execução das instruções, contando inclusive com registradores especiais como o Acumulador (ACC), utilizado para operações aritméticas; o Contador de Programa (PC) que aponta para a próxima instrução a ser realizada e o Registrador de Interrupção dos Dispositivos (DIR), utilizado na operação dos dispositivos conectados à máquina virtual. Conectada aos registradores pelo barramento de dados está a memória endereçável de 6 *Kilobytes*, na qual ficam armazenados tanto os códigos do programa em execução, quanto os dados trabalhados por este programa, além de contar com um espaço reservado, de 256 *bytes*, para a utilização por dispositivos conectados. Mais detalhes sobre os componentes da arquitetura estão nos parágrafos a seguir.

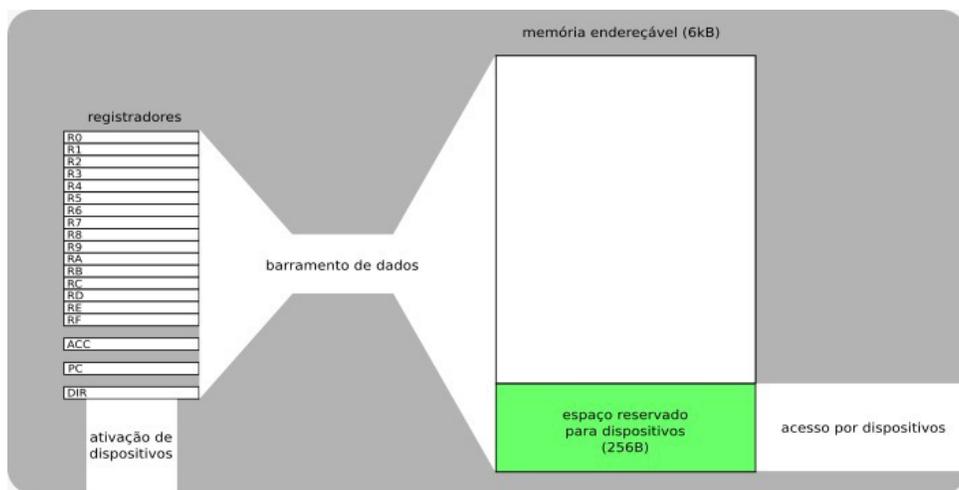


Figura 1 - Diagrama da arquitetura da máquina virtual desenvolvida.

Fonte: Figura de autoria própria.

A arquitetura da máquina virtual desenvolvida é baseada em **registradores** e, assim como àquela de um processador físico, tem dezenove registradores de dezesseis *bits* cada um, sendo estes:

- Dezesseis registradores de propósito geral (R0 a RF, indexados por um algarismo hexadecimal), chamado de *GRs* ou *General Registers*, utilizados livremente pelo programador durante o desenvolvimento das aplicações;

- Um acumulador (ACC), utilizado pela máquina para realização de operações aritméticas. Este também pode ser referenciado diretamente pelo usuário em instruções que utilizam *GRs*;
- Um contador de programa, chamado de *PC* ou *Program Counter*, utilizado pela máquina virtual para saber qual a posição de memória da próxima instrução que será executada. Esse registro é incrementado automaticamente a cada instrução e modificado em instruções de pulo de execução do programa;
- Um registrador de interrupções dos dispositivos, chamado de *DIR* ou *Device Interrupts Register*, utilizado para ativar as interrupções de dispositivos externos, para que estes tomem controle da aplicação. O funcionamento dos dispositivos é explicado na seção dispositivos externos.

A máquina virtual também possui seis *kilobytes*, ou 65536 *bytes*, de memória endereçável, ou seja, todos os endereços possíveis do contador de programa de 16 *bits* podem ser utilizados, porém, os 256 primeiros *bytes* da memória (endereços 0x0000 a 0x0099 em hexadecimal) são utilizados para acesso por dispositivos, assim, o valor do contador de programa no início da execução do código é 256 (0x0100 em hexadecimal). A aplicação desenvolvida também foi baseada na arquitetura de Von Neumann, portanto, as instruções do código de máquina e os dados utilizados no programa ficam armazenados na mesma memória.

A aplicação desenvolvida também implementa o conceito de **dispositivos externos virtuais**, programas que podem utilizar parte da memória da máquina para realizar operações que envolvam a interação da máquina com o meio externo, ou operações de entrada e saída (*I/O* ou *Input/Output*). Conforme comentado anteriormente, a área reservada para uso dos dispositivos externos consiste dos *bytes* nos endereços 0 a 255, na qual cada dispositivo reserva certa quantidade de *bytes*. Os dispositivos são conectados à máquina antes da execução do código de fato e funcionam basicamente tomando controle da execução após uma instrução, por meio de uma interrupção e utilizando a memória reservada a eles. As interrupções de dispositivos são chamadas pelo próprio programa, ou seja, são interrupções internas, colocando um *bit* relativo à interrupção como alto (*high* ou *1*) no registrador de interrupções dos dispositivos (o *DIR*). Portanto, como no caso da memória, a quantidade de *bits* reservados por dispositivo varia e sempre se inicia do *bit* menos significativo do registrador, na ordem em que os dispositivos são conectados à máquina.

Apenas dispositivos de saída foram utilizados em testes na máquina virtual, pois, uma vez que todas as interrupções são internas, dispositivos de entrada não são tão efetivos, já que as chamadas seriam síncronas e partiriam do próprio programa executado. É importante considerar que a implementação de sub-rotinas e interrupções externas está entre as possíveis melhorias futuras para a arquitetura da máquina virtual.

O dispositivo mais utilizado e deixado nativamente (dispositivo com suporte direto no sistema) na máquina foi um mini *display oled*, que pode ser observado na Figura 2. Neste, é possível escrever linhas de texto de forma semelhante a um terminal e foi escolhido por ser uma forma muito prática de obter saída dos programas durante a realização de testes.

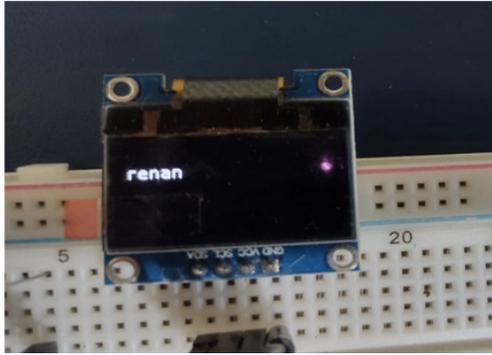


Figura 2 - *Display OLED* utilizado como dispositivo de saída da máquina virtual, em funcionamento.

Fonte: Figura de autoria própria.

O **código de máquina** da arquitetura desenvolvida consiste em 37 instruções, cada qual com um código de operação de um *byte* e parâmetros que variam de tamanho dependendo da instrução. Essa variabilidade vai de instruções sem parâmetros, que têm apenas um *byte* de tamanho, até instruções com um valor literal, de dezesseis *bits*, e um endereço de memória, também de 16 *bits*, totalizando 5 *bytes* de tamanho total da instrução. O tempo de execução da máquina é capaz de lidar com as instruções de diferentes tamanhos de maneira automática. As instruções implementadas foram categorizadas em 5 grupos diferentes, dependendo de suas funcionalidades, são eles:

- **Instruções de movimentação:** têm a função de mover valores entre diferentes localidades, como registradores e posições de memória;
- **Instruções de pulo:** têm como função realizar pulos na execução do programa, ou seja, alterar o valor do contador de programa para outra posição da memória dependendo de certas condições;
- **Instruções de adição:** realizam somas de valores no valor de registradores;
- **Instruções de subtração:** realizam subtrações de valores no valor de registradores;
- **Instruções de incremento/decremento:** incrementa ou decrementa o valor de registradores;
- **Instruções de mudança de *bits*:** realizam operações binárias de mudança de *bits* no valor de registradores;
- **Instruções de operações lógicas:** realizam operações de lógica binária (*AND*, *OR*, *XOR* e *NOT*) no valor de registradores.

Por ter uma visão didática, implementada de maneira que preza pela simplicidade,

o ciclo de instruções da máquina ou ciclo de busca-execução, é realizado por uma única função, que chama uma macro da linguagem. Essa macro, que pode ser observada na Figura 3, primeiramente decodifica a instrução comparando os *bytes* mais e menos significativos de seu *opcode*. As instruções possuem valores preestabelecidos para definir a categoria da instrução e, logo depois, a instrução em particular dentro da sua categoria, realizando este processo em uma estrutura de seleção na forma de árvore. Após a decodificação da instrução, a operação é executada instantaneamente com os operandos necessários para ela, já que, uma vez que os tipos de operandos são definidos por instrução, nenhuma verificação necessita ser realizada. Ainda é possível ver no próprio código, qual instrução de máquina é executada por cada linha da macro e quais tipos de operandos utilizam por meio dos comentários que estão na frente de cada uma das linhas. Aqui, as instruções estão representadas na forma de mnemônicos semelhantes aos de uma linguagem *assembly*, assim esse trecho crítico do código já conta com uma forma de documentação básica, porém eficiente, junto do próprio código.

```

#define DECODE_INSTRUCTIONS(a,b) \
a(0x0, \
  b(0x0, WORD; BYTE; Ry = x;); /* MOV lit, reg */ \
  b(0x1, BYTE; BYTE; Ry = Rx;); /* MOV reg, reg */ \
  b(0x2, BYTE; WORD; write_word(y, Rx);); /* MOV reg, addr */ \
  b(0x3, WORD; BYTE; Ry = read_word(x);); /* MOV addr, reg */ \
  b(0x4, WORD; WORD; write_word(y, x);); /* MOV lit, addr */ \
  b(0x5, BYTE; BYTE; Ry = read_word(Rx);); /* MOV addr_ptr(reg), reg */ \
  b(0x6, BYTE; BYTE; write_word(Ry, Rx);); /* MOV reg, addr_ptr(reg) */ \
) \
a(0x1, \
  b(0x0, WORD; WORD; if (x != _ACC)_PC = y;); /* JNE lit, addr */ \
  b(0x1, BYTE; WORD; if (Rx != _ACC)_PC = y;); /* JNE reg, addr */ \
  b(0x2, WORD; WORD; if (Rx == _ACC)_PC = y;); /* JEQ lit, addr */ \
  b(0x3, BYTE; WORD; if (Rx == _ACC)_PC = y;); /* JEQ reg, addr */ \
  b(0x4, WORD; WORD; if (x < _ACC)_PC = y;); /* JLT lit, addr */ \
  b(0x5, BYTE; WORD; if (Rx < _ACC)_PC = y;); /* JLT reg, addr */ \
  b(0x6, WORD; WORD; if (x > _ACC)_PC = y;); /* JGT lit, addr */ \
  b(0x7, BYTE; WORD; if (Rx > _ACC)_PC = y;); /* JGT reg, addr */ \
  b(0x8, WORD; WORD; if (x <= _ACC)_PC = y;); /* JLE lit, addr */ \
  b(0x9, BYTE; WORD; if (Rx <= _ACC)_PC = y;); /* JLE reg, addr */ \
  b(0xa, WORD; WORD; if (x >= _ACC)_PC = y;); /* JGE lit, addr */ \
  b(0xb, BYTE; WORD; if (Rx >= _ACC)_PC = y;); /* JGE reg, addr */ \
  b(0xc, WORD; _PC = x;); /* UCJ addr */ \
) \
a(0x2, \
  b(0x0, BYTE; BYTE; _ACC = Rx + Ry;); /* ADD reg, reg */ \
  b(0x1, WORD; BYTE; _ACC = x + Ry;); /* ADD lit, reg */ \
) \
a(0x3, \
  b(0x0, BYTE; BYTE; _ACC = Rx - Ry;); /* SUB reg, reg */ \
  b(0x1, WORD; BYTE; _ACC = x - Ry;); /* SUB lit, reg */ \
) \
a(0x4, \
  b(0x0, BYTE; Rx++;); /* INC reg */ \
  b(0x1, BYTE; Rx--;); /* DEC reg */ \
) \
a(0x5, \
  b(0x0, BYTE; WORD; Rx = Rx << y;); /* LSF reg, lit */ \
  b(0x1, BYTE; BYTE; Rx = Rx << y;); /* LSF reg, reg */ \
  b(0x2, BYTE; WORD; Rx = Rx >> y;); /* RSF reg, lit */ \
  b(0x3, BYTE; BYTE; Rx = Rx >> y;); /* RSF reg, reg */ \
) \
a(0x6, \
  b(0x0, BYTE; WORD; _ACC = Rx & Ry;); /* AND reg, lit */ \
  b(0x1, BYTE; BYTE; _ACC = Rx & y;); /* AND reg, reg */ \
  b(0x2, BYTE; WORD; _ACC = Rx | Ry;); /* OR reg, lit */ \
  b(0x3, BYTE; BYTE; _ACC = Rx | y;); /* OR reg, reg */ \
  b(0x4, BYTE; WORD; _ACC = Rx ^ Ry;); /* XOR reg, lit */ \
  b(0x5, BYTE; BYTE; _ACC = Rx ^ y;); /* AND reg, reg */ \
  b(0x6, BYTE; _ACC = -Rx;); /* NOT reg */ \
)

```

Figura 3 - Macro que decodifica e executa todas as instruções da arquitetura da máquina virtual.

Fonte: Figura de autoria própria.

Após a execução de cada instrução, uma função é chamada para verificar se alguma interrupção de dispositivo foi chamada e, em caso positivo, a máquina verifica quais dispositivos foram acionados realizando operações de máscara de *bits* no registrador de interrupção de dispositivos (o *DIR*). Após, passa à execução do programa para eles realizarem as operações necessárias com seus respectivos trechos de memória mapeados.

A metaprogramação utilizando macros mostrou-se facilmente extensível, uma vez que a maioria das instruções foram adicionadas e muitas alterações na execução de instruções foram realizadas após o ciclo de execução da máquina já estar completamente implementado. Isso prova que essa é uma maneira completamente eficaz de se implementar o funcionamento de um processador virtual, já que este realiza uma vasta gama de tarefas simples que muitas vezes precisam ser modificadas ou incrementadas.

Foi implementado um **sistema de arquivos** com persistência na aplicação da máquina virtual, permitindo que o código a ser executado possa ser carregado de um computador via cabo e que tal código permaneça gravado no dispositivo mesmo após ser desligado e religado. O sistema de arquivo implementado foi o *SPIFFS (SPI Flash File System)*, que é de código aberto e foi desenvolvido especificamente para funcionar na memória *flash* de dispositivos embarcados e tem suporte nativo pela plataforma do ESP32. Com tal funcionalidade implementada, tudo que o precisa ser feito para inserir código na máquina virtual via cabo USB é exemplificado na figura 4, que mostra a utilização da ferramenta gratuita *mkspiffs*. Tal ferramenta compacta a pasta *fs_folder*, com apenas um arquivo e que, obrigatoriamente, deve ser chamado *program* e é o código do programa em formato binário para uma imagem de partição SPIFFS, que aqui se chama *spiffs.bin* e tem um tamanho de 960 *Kilobytes*, ou 0xF0000 *bytes* em hexadecimal. A imagem deve ter este tamanho devido à tabela de partição do sistema de arquivos. Para a gravação, é utilizada a ferramenta de linha de comando do ESP-IDF *esptool.py*. A imagem gerada na memória *flash* da placa, no *offset* 0x110000 que é um valor que foi definido na tabela de partições. Apesar dos comandos no exemplo terem sido executados em um sistema operacional Linux, todas as ferramentas utilizadas estão disponíveis para a maioria dos sistemas operacionais de computadores *desktop* populares, como o *Microsoft Windows* e o *macOS* da *Apple*.

Até o momento esta é a única forma de enviar código para a máquina virtual, porém, outras maneiras podem ser desenvolvidas futuramente, uma vez que o sistema de arquivos pode ser acessado por código, inclusive para a escrita, e levando em conta as diversas opções de conectividade oferecidas pelo *hardware* da ESP32, como *Wi-Fi* e *Bluetooth*

Era parte do planejado para o projeto da plataforma de desenvolvimento embarcado a criação de uma **interface para programação visual**, utilizando de uma linguagem chamada de *assembly visual*. Tal interface combinaria a facilidade das linguagens de programação em blocos com a robustez e simplicidade do *assembly*, também chamado de linguagem de montagem. Apesar do compilador, ou *assembler*, da linguagem de máquina de nossa máquina virtual não ter sido desenvolvido ainda, a linguagem em si, com todos os seus mnemônicos e funcionalidades foi definida, de forma que é possível traduzir um programa para ela e, mesmo que isso não seja funcional, esta programação se mostra de mais fácil compreensão. Com a sintaxe desenvolvida, todas as 37 instruções foram simplificadas em 19 mnemônicos, uma vez que as instruções variam de acordo com o tipo

de parâmetro utilizado.

Além dos mnemônicos, outra funcionalidade muito útil do *assembly* são os *labels*, nome dado a certo endereço de memória. Aqui, os *labels* podem ser usados tanto para nomear trechos do código e realizar pulos condicionais para a localidade destes trechos, quanto para inserir valores constantes diretamente na memória, como se fossem variáveis em uma linguagem de programação de alto nível. Assim, se torna desnecessário manter o endereço físico de tais localidades da memória, como acontece quando se programa diretamente em código de máquina. A seguir, na Figura 4, apresenta-se um exemplo de código em *assembly visual*, mostrando ambos os usos dos *labels*:

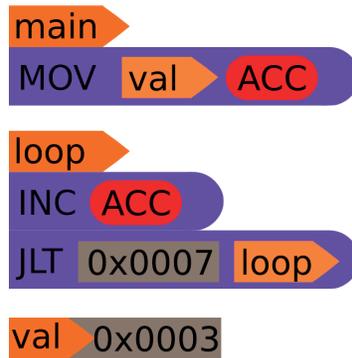


Figura 4 - Trecho de código em *assembly visual*, mostrando utilização dos *labels*.

Fonte: Figura de autoria própria.

Na figura acima, observa-se a *label main*, que sempre aponta o início da execução do programa. A primeira instrução movimenta o valor apontado pela *label val* para o registrador acumulador (o *ACC*), seguida pela *label loop*, na qual é executada uma instrução de incremento no acumulador. Depois disso, está inserida uma instrução de pulo condicional que retoma a execução do programa para a localização apontada pela *label loop*, caso o valor do acumulador seja menor do que 7, ou seja, o trecho executa uma iteração, contando do valor 3 até 7, algo similar a um *loop* (ou laço) em uma linguagem de programação de alto nível.

Uma observação importante é a de como os formatos e cores têm significados definidos: a seta laranja sempre representa uma *label*, a elipse vermelha, um registrador, e o retângulo cinza, um valor literal. Esse padrão foi escolhido apenas para prova de conceito de como o artifício do *assembly visual* tornaria a programação mais simples se comparada àquela que utiliza uma linguagem *assembly* tradicional, em texto.

A seguir, é apresentado um exemplo de programa para a máquina virtual, com uma funcionalidade bastante básica: mostrar uma mensagem de “olá mundo” no *display* utilizado como dispositivo de saída. Para isso, foram utilizadas 6 das 37 instruções disponíveis na arquitetura, as quais são mostradas e explicadas no Quadro 1.

Opcode	Mnemônico	Operandos		Descrição
		1	2	
0x00	MOV	literal (16 bits)	registrador (8 bits)	Move o valor literal para o registrador especificado
0x05	MOV	registrador (8 bits)	registrador (8 bits)	Move o valor do endereço de memória apontado pelo registrador 1 para o registrador 2
0x06	MOV	registrador (8 bits)	registrador (8 bits)	Move o valor do registrador 1 para o endereço de memória apontado pelo registrador 2
0x10	JNE	literal (16 bits)	endereço (16 bits)	Pula a execução do programa para o endereço de memória, caso o valor do acumulador seja diferente do literal
0x40	INC	registrador (8 bits)		Incrementa o valor no registrador
0xFF	HLT			Termina a execução do programa

Quadro 1 - Instruções utilizadas no programa de exemplo.

Fonte: Quadro de autoria própria.

O código do programa de exemplo, da mesma forma em que está distribuído na memória da máquina virtual em formato hexadecimal, é mostrado no Quadro 2. Lembrando que as instruções de um programa começam no endereço 256 (ou 0x0100 em hexadecimal) pois a memória antes desta posição é reservada para os dispositivos conectados à máquina e, sendo assim, ficam vazios no início da execução de um programa.

	0x00	0x01	0x02	0x03	0x04	0x05	0x06	0x07	0x08	0x09	0x0A	0x0B	0x0C	0x0D	0x0E	0x0F
0x0100	00	01	1F	00	00	00	00	01	05	00	02	06	02	01	40	00
0x0110	40	01	05	00	10	10	00	00	01	08	00	00	01	11	FF	48
0x0120	65	6C	6C	6F	20	77	6F	72	6C	64	00	00	00	00	00	00

Legenda	
	Opcodes
	Parâmetros
	Dados
	Não utilizado

Quadro 2 - Código em hexadecimal do programa de exemplo.

Fonte: Quadro de autoria própria.

Uma outra forma de escrever um programa é utilizando-se da linguagem *assembly* (Figura 5). O *assembly visual* para a máquina virtual não foi desenvolvido, porém seus mnemônicos e funcionalidades foram projetados. A Figura 5, mostra como ficaria o programa de teste referido nesta seção se desenvolvido na linguagem *assembly visual* que foi projetada para a máquina virtual.

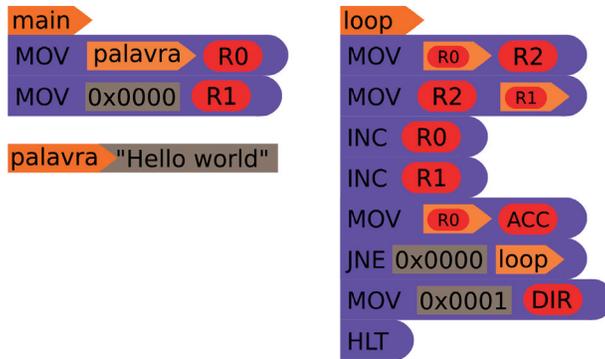


Figura 5 - Código em *assembly visual* do programa de exemplo.

Fonte: Figura de autoria própria.

Com este trabalho, foi comprovado ser possível o desenvolvimento de implementações virtuais de computadores a partir do zero, tendo por base conteúdos relativos a arquiteturas físicas que antes tinham sido abordados apenas de forma teórica pelos desenvolvedores dessa aplicação. No desenvolvimento foram utilizados conceitos de programação intermediários, como os ponteiros de memória e macros da linguagem C, assim, foi criado outro nível de abstração capaz de executar aplicações completamente funcionais e desenvolvidas de forma significativamente mais simples do que aquelas para o nível inferior, propicia para o estudo de sistemas embarcados. Além de tudo isso, o projeto é de código aberto (<https://github.com/renanOliveira/ppvm>), e foi desenvolvido de maneira completamente extensível, tanto nas instruções quanto nos dispositivos virtuais, permitindo que toda a comunidade de desenvolvedores utilize-o e realize alterações e melhorias como bem entender.

REFERÊNCIAS

SANTOS, Jean Willian; JUNIOR, Renato Capelin de Lara. **Sistema de automatização residencial de baixo custo controlado pelo microcontrolador ESP32 e monitorado via smartphone**. 2019. 46 f. Trabalho de conclusão de curso (Curso superior de Tecnologia em Automação Industrial) Universidade Federal do Paraná, Paraná, 2019.

SILVA, Lucas M. **Uma máquina virtual para uso didático**. 2012. 236 f. Monografia (Graduação em Sistemas de Informação) - Universidade Federal de Santa Catarina, Florianópolis, 2012.

TANENBAUM, A. S.; AUSTIN, T. **Organização estruturada de computadores**. 6ª edição. São Paulo: Pearson Education do Brasil, 2013.

ÍNDICE REMISSIVO

A

Análises 21, 22, 24, 26, 30, 122, 125, 126, 127, 129, 130, 131, 132, 133, 161, 191, 204, 207, 208, 217

Antioxidante 122, 156, 157

Aplicação 22, 28, 30, 47, 48, 50, 51, 54, 57, 59, 60, 62, 64, 81, 83, 86, 107, 109, 110, 114, 115, 119, 131, 133, 134, 135, 142, 148, 149, 156, 170, 171, 172, 187, 188, 190, 211, 217

Aquisição 31, 33, 47, 58, 59, 60, 61, 64

B

Bioplástico 122

C

Casca de banana 187, 188, 189, 191, 192, 193, 194, 197, 198

Celulose 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 121, 122, 155

Ciclo de vida 136, 146

Computador 48, 54

Corantes 187, 189, 193, 195, 196, 197, 198

Cosméticos 83, 148, 149, 151, 152, 158, 159, 187, 188

D

Dados 21, 22, 24, 25, 26, 27, 28, 29, 30, 31, 33, 35, 36, 38, 41, 42, 43, 44, 45, 49, 50, 51, 59, 60, 61, 62, 64, 68, 69, 71, 73, 74, 75, 76, 77, 95, 98, 99, 100, 101, 103, 104, 105, 122, 131, 145, 146, 151, 163, 167, 168, 170, 174, 181, 183, 190, 191, 193, 194, 214, 218

defletores 85

Desenvolvimento 21, 23, 24, 30, 31, 33, 37, 38, 39, 42, 47, 48, 49, 50, 54, 57, 58, 60, 64, 76, 81, 83, 95, 98, 108, 119, 120, 122, 132, 145, 148, 149, 150, 151, 152, 155, 157, 158, 159, 160, 162, 174, 175, 185, 188, 199, 200, 203, 206, 207, 218, 220

Dimensionamento 80, 81, 177, 178

E

Eficiência 21, 49, 59, 70, 71, 72, 74, 75, 76, 78, 80, 81, 97, 100, 109, 114, 161, 162, 164, 173, 188, 213, 219, 220

Efluentes industriais 187, 198

Energia 48, 58, 59, 60, 63, 64, 68, 70, 71, 72, 73, 74, 75, 76, 78, 80, 81, 82, 84, 95, 121, 135, 136, 137, 177, 189, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 217, 219

Energia Solar 70, 71, 72, 73, 82

G

Géis 151, 155, 157

GPS 4, 33, 34, 36, 37, 38, 44, 46

I

Impelidores 83, 84, 85, 86, 87, 88, 89, 90, 93, 94

Indústria 4.0 30, 162, 163, 165, 173

Informação 23, 26, 27, 36, 37, 57, 68, 162, 169, 181, 218

Inteligência artificial 220

IoT 21, 22, 30, 38, 49, 68, 162, 163

I-Pai Wu 177

K

K-means 28, 29

L

Linha de produção 161, 162, 164, 165, 166, 167, 170, 171

M

Microcontrolador 30, 31, 37, 38, 39, 40, 47, 49, 57, 168

Microdrenagem 7, 174, 175, 177, 179, 184, 185

Modelagem 34, 59, 68, 82, 95, 98, 100, 105, 220

Modelo matemático 95, 98, 101, 105

Monitoramento 19, 33, 34, 49, 58, 60, 64, 161, 162, 163, 167, 169, 170, 171, 173, 175

N

Nanotecnologia 108

O

Óleo de café 148, 151, 154, 155, 157, 160

P

Papel 107, 108, 109, 110, 114, 115, 116, 117, 118, 119, 151, 189, 202

Piezoelétrico 58, 59, 60, 63, 64, 68

Programação 38, 40, 41, 47, 48, 49, 54, 55, 57, 100, 101, 173

R

Rastreamento 33, 34, 39, 45, 83, 88

Rastreamento de partículas 83

Reator 95, 96, 97, 98, 99, 100, 101, 103, 104, 105, 204

Rede neural 21, 24, 25

Rendimento 82, 97, 98, 99, 100, 120, 121, 123, 126, 131, 164, 192, 197

Rolhas de pallets 139

S

Saúde 203, 208, 217, 219

Simulação 34, 39, 64, 67, 75, 76, 77, 95, 100, 104, 105, 145, 171, 220

Solubilidade 120, 123, 126, 131, 132, 210

T

Testes comportamentais 21, 24

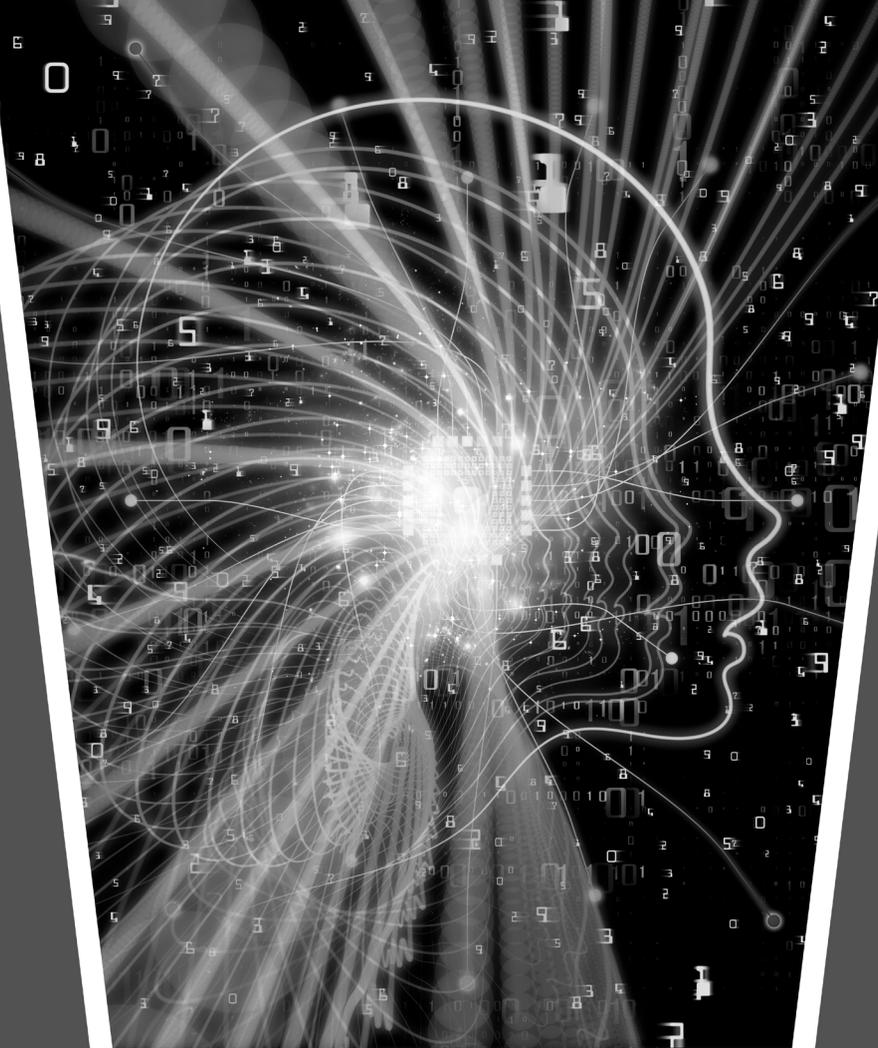
Transformação digital 163

V

Veículos 33, 34, 64

Virtual 12, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57

Vórtices 84, 85, 91



Engenharia Moderna: Soluções para Problemas da Sociedade e da Indústria 2



www.atenaeditora.com.br



contato@atenaeditora.com.br



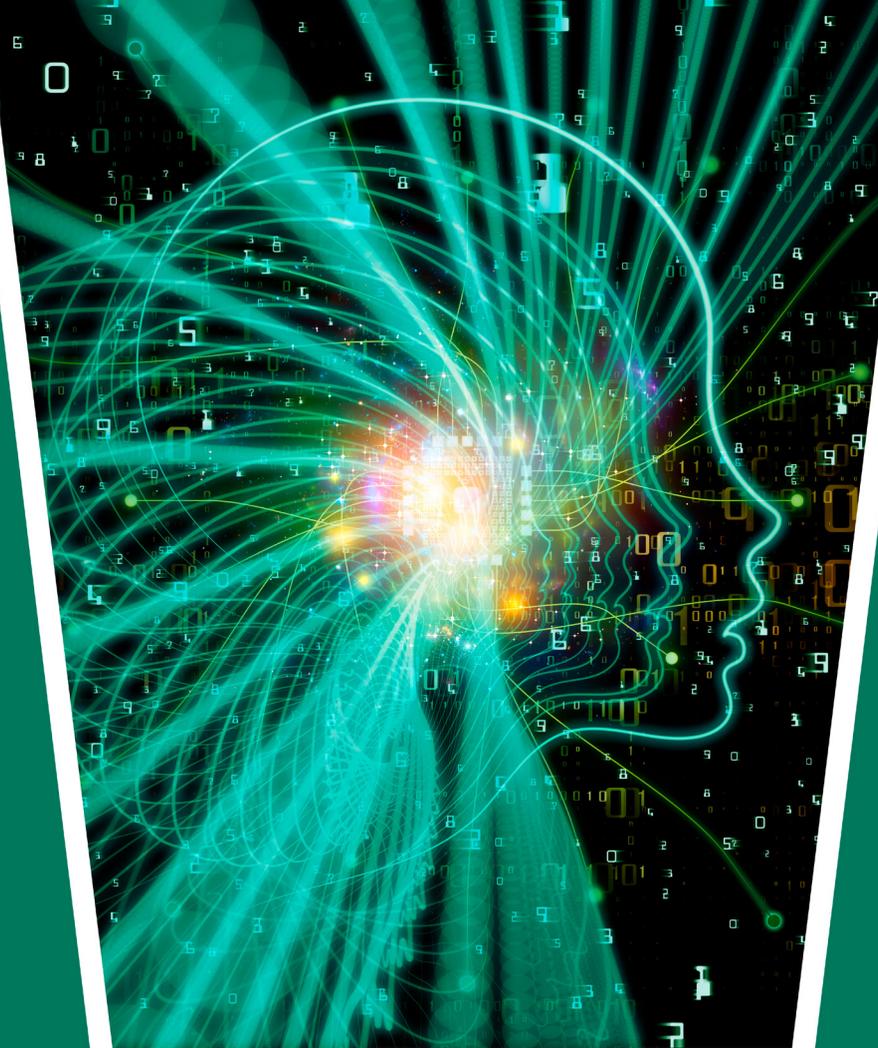
[@atenaeditora](https://www.instagram.com/atenaeditora)



www.facebook.com/atenaeditora.com.br

Atena
Editora

Ano 2021



Engenharia Moderna: Soluções para Problemas da Sociedade e da Indústria 2

-  www.atenaeditora.com.br
-  contato@atenaeditora.com.br
-  [@atenaeditora](https://www.instagram.com/atenaeditora)
-  www.facebook.com/atenaeditora.com.br


Editora
Ano 2021