INSTITUTO FEDERAL DO ESPÍRITO SANTO

ENGENHARIA DE CONTROLE E AUTOMAÇÃO

**LUCAS COELHO GARBOSSA**

**GUILHERME OLIVEIRA**

**IMPLEMENTAÇÃO EM HARDWARE DE INSTRUÇÃO DE PROCESSADOR ARM MONOCÍCLO**

SERRA

2025

LUCAS COELHO GARBOSSA

GUILHERME OLIVEIRA

**IMPLEMENTAÇÃO EM HARDWARE DE INSTRUÇÃO DE PROCESSADOR ARM MONOCÍCLO**

Projeto de desenvolvimento apresentado ao professor Rafael Emerick Z. De Oliveira como requisito para aprovação na disciplina de Arquitetura de Computadores.

SERRA 2025

**SUMÁRIO**

**1 INTRODUÇÃO..........................................................................................................7**

**2 FUNDAMENTAÇÃO TEÓRICA.................................................................................7**

2.1.................................................................................................8

2.2 ...................................................................................8

**3 H.................................................................................................................9**

**4 R........................................................9**

**5 MATERIAIS E MÉTODOS.......................................................................................12**

**6 REFERÊNCIAS.......................................................................................................15**

**1 Introdução**

A Arquitetura de um hardware é definida pelo conjunto de instruções ou comandos que uma máquina deve saber obedecer e o conjunto de regras de manipulação dos operandos que esses comandos sabem utilizar e operar. Ou seja, ela é uma perspectiva do ponto de vista do programador do computador.

Para executar tais tarefas, é preciso um processador, que pode ser classificado como RISC (que trabalha com um conjunto de instruções reduzidas, sendo menor e mais rápido, ou seja, são projetados para eficiência e desempenho, além de ter um hardware mais simples e um software mais complexo) ou CISC (que tem instruções e hardware mais complexos, o que permite um software mais simples e menor).

Vistos esses aspectos e, devido às suas características, podemos classificar os processadores ARM Monocíclicos como RISC.

O presente relatório da disciplina de Arquitetura de Computadores, do Curso de Engenharia de Controle e Automação do Instituto Federal do Espírito Santo, tem como objetivo descrever a implementação em hardware das instruções MOV, CMP, TST e EOR através de ferramentas que permitam simulação e desenvolvimento de novas instruções em código HDL após entendimento do funcionamento da hierarquia e módulos internos de um processador ARM.

**2 Desenvolvimento**

**2.1 Testando o processador ARM Monocíclico**

Como o processador ARM monocíclico é classificado como RISC, já sabemos que ele contará com um software mais complexo quando pensarmos em desenvolver funções que seu hardware não pode executar por si só, já que é desenvolvido para ser mais simples, menor e mais eficiente.

Portanto, é imprescindível entender afundo o funcionamento de suas partes internas, incluindo sua hierarquia, como o Testbench, e seus módulos mais internos, presentes dentro do DUT (Device Under Test). O Testbench pode ser entendido como nosso ambiente de simulação, sendo capaz de testar e verificar o comportamento do DUT, para dessa forma, sabermos o que esperar da resposta e prever o que deve acontecer em cada ciclo executado no programa. Assim, poderemos garantir que tudo está funcionando conforme esperado. O DUT inclui todos os módulos internos do processador, como a ALU (unidade lógica aritmética), os registradores, flags, etc, que trabalham juntos para executar corretamente as instruções do processador.

**2.1.1 Esquemático ARM monocíclico**

Para entender todo esse funcionamento, foi desenvolvido um esquemático que representa todas as partes e módulos internos de um processador ARM Monocíclico. Ele se mostrou muito eficaz em facilitar o entendimento do funcionamento do processador, além de auxiliar na visualização dos pontos que deveriam ser alterados para a implementação das novas funções. Ele pode ser visto na Figura 1, Figura 2 e Figura 3 que aparecem logo abaixo. Veja como fica didático e compreensível o funcionamento de todos os componentes do processador, desde a unidade de controle até os registradores. É possível visualizar como todos os dados são tratados em cada módulo, sendo possível perceber o que é entrada, saída, etc em cada um deles. Por exemplo, na Figura 1, no módulo “arm”, não temos dificuldade em perceber que as entradas são “Instr” e “ReadData”, ambas de 32 bits, e as saídas são “MemWrite”, de 1 bit, e “WriteData”, “ALUResult” e “PC” de 32 bits. Assim podemos ver como cada um dos blocos tem funções definidas e como eles devem se conectar aos outros.

Diagrama, Esquemático

Descrição gerada automaticamente

Figura 1 - Esquemático processador ARM (parte 1)

Diagrama

Descrição gerada automaticamente

Figura 2 - Esquemático processador ARM (parte 2)

Diagrama, Esquemático

Descrição gerada automaticamente

Figura 3 - Esquemático processador ARM (parte 3)

**2.1.2 Simulação e preenchimento da Tabela 1**

Antes de tudo, foi necessário entender o significado de cada coluna da Tabela 1:

* **SrcA**: Esta coluna representa a fonte A, que é um dos operandos utilizados pela ALU (Unidade Lógica Aritmética) para realizar operações aritméticas e lógicas.
* **SrcB**: Esta coluna representa a fonte B, que é o segundo operando utilizado pela ALU para realizar operações aritméticas e lógicas.
* **ALUResult**: Esta coluna mostra o resultado da operação realizada pela ALU com os operandos SrcA e SrcB.
* **MemWrite**: Esta coluna indica se uma operação de escrita de memória está sendo realizada. Se for 1, a memória está sendo escrita; se for 0, não está sendo escrita.
* **Branch**: Esta coluna indica se uma instrução de desvio condicional está sendo executada. Se for 1, o desvio será realizado; se for 0, não será realizado.
* **ReadData**: Esta coluna mostra os dados que foram lidos da memória durante uma operação de leitura.
* **WriteData**: Esta coluna representa os dados que estão sendo escritos na memória durante uma operação de escrita.

Com os caminhos traçados no esquemático, uma visão mais macro de todo o processo e o entendimento do significado dos parâmetros de entrada e saída o preenchimento da tabela se tornou uma tarefa mais simples e dinâmica.

Então, foi feito o preenchimento da Tabela 1 de acordo com nossas previsões e de acordo com o significado de cada coluna.

Gráfico

Descrição gerada automaticamente com confiança média

Tabela 1 - Previsões de resultados da execução do código memfile.s

Preenchida a tabela, percebemos que a instrução final “STR”, escreverá o **valor 7** (em decimal) na **posição de memória 100** (em decimal).

Seguimos para a simulação no Modelsim (Figura 4). Para incluir os valores desejados, o esquemático das Figuras 1,2 e 3 ajudou bastante, pois com ele foi possível navegar dentro dos módulos do processador ARM no Modelsim com facilidade, visto que é possível identificar a localização de cada variável seguindo os blocos lógicos do desenho.

Assim, com todos os sinais adicionados na janela de ondas, foi possível fazer toda a conferência dos valores da Tabela 1 com a simulação, rolando a barra em amarelo, clock após clock, conferindo valor a valor e numerando cada ciclo na tabela para um clock diferente. Observe como o primeiro ciclo da tabela (1) tem os valores em concordância com a simulação:

* Scra = 8;
* Scrb = 8;
* AluResult = 0, e assim por diante.

Interface gráfica do usuário

Descrição gerada automaticamente

Figura 4 - Simulação no Modelsim

Agora, podemos confrontar nossa previsão em tabela com a simulação do processador. Através da Figura 5, visualizamos a última instrução (“str r2, [r0, #0x64]”, que significa: escreva o valor de r2 (7) na posição de memória r0+100 (100)), que nos mostra que a ALUResult indica o valor 100, WriteData indica 7 e MemWrite indica 1 (habilitando a função de escrita na memória). Tudo isso resulta no valor 7 sendo gravado na posição de memória 100, confirmando perfeitamente nossas previsões.

Tela de computador com jogo

Descrição gerada automaticamente

Figura 5 - Última instrução da simulação que confirma as previsões feitas em tabela

**2.2 Modificando o processador ARM Monocíclico**

Após total compreensão do processo, principalmente em nível de microarquitetura, foi possível iniciar o processo de desenvolvimento das novas instruções a serem processadas, dentre elas: MOV, CMP, TST e EOR.

Para isso, precisamos saber também a ordem de interpretação e o que cada bit representa em uma instrução. É possível ter essa informação através da Figura 6.

Diagrama

Descrição gerada automaticamente

Figura 6 - Codificação de instruções de processamento de dados de um processador ARM monocíclico

Além disso, consultaremos também a Tabela 2, para seguir o padrão de comandos da arquitetura arm.

Tabela

Descrição gerada automaticamente

Tabela 2 - Instruções de processamento de dados

**2.2.1 Função MOV**

A função MOV em assembly é usada para mover. Em um processador ARM, a instrução MOV copia o valor de um registrador ou um valor imediato (a depender de como a instrução será passada) para outro registrador. Por exemplo, MOV R1, R2 copia o valor do registrador R2 para o registrador R1.

Para implementar a função MOV em HDL (Hardware Description Language) no hardware de um processador ARM monocíclico, é necessário modificar a unidade de controle para reconhecer a instrução MOV e direcionar os sinais de controle adequados para os registradores. Basicamente, a unidade de controle deve gerar os sinais que permitem a leitura do valor do registrador de origem (ou imediato) e a escrita desse valor no registrador de destino.

Podemos dividir em três etapas:

1. **Decodificação:** a unidade responsável pelo controle deve ser capaz de identificar quando uma função MOV deverá ser executada. Para isso, ele deve saber identificá-la através dos bits 20:27, que carregam os dados do Opcode. Nos preocuparemos mais especificamente com a secção funct da Figura 6, onde nos atentaremos para o I (registrador (0) ou imediato (1)) e o cmd (operação a ser executada, nesse caso, o MOV - 1101 de acordo com a Tabela 2 - );
2. **Geração de sinais de controle:** com a unidade de controle sendo capaz de identificar a função MOV, devemos alterá-la para que consiga também gerar sinais de controle, ou seja, o “controller”, através dos seus submódulos “decoder” e “condlogic”, deverá gerar sinais de saída corretos (RegSrc, RegWrite, ImmSrc, etc – ver esquemático das Figuras 1, 2 e 3 - ) para o “datapath”;
3. **Execução da Instrução:** no “datapath”, onde os sinais de controle são recebidos, os diversos submódulos, que compõem multiplexadores, registradores, etc (como “regfile”, “alu”, “mux2”, “adder”, etc) são direcionados para que as operações aritméticas e movimentação de dados sejam executadas corretamente.

Com as etapas bem definidas, podemos iniciar o desenvolvimento das novas funções. Para que o “controller” seja capaz de identificar a função MOV, criamos o “MovFlag” (1 bit). Com nosso esquemático bem estabelecido, ficou mais simples expandir esse bit para os locais que precisam identificá-lo para o correto funcionamento.

Como já dito e podendo ser verificado no esquemático, ela é passada como parâmetro inicialmente no bloco arm, sendo sinalizada em seguida para o controller, para o datapath, e vai sendo expandida de fora para dentro em todos os outros locais nos quais é necessária. Além disso, foi criada uma variável interna para “espelhar” o MovFlag, a “FlagM”, para servir como barramento interno e ser passada como parâmetro para o decoder e condlogic. Vale ressaltar que é importante seguir uma ordem de variáveis para passar o MovFlag, respeitando a regra de big ou little endian a qual o processador respeita.

Além disso, foi necessário criar um novo mux2, o “movmux” para passar os parâmetros da instrução MOV.

**2.1.1) Especificações**

* Placa transmissora de sinal
* Tensão de entrada: 3,3 ~ 5,5 V;
* Tensão de saída: 0 ~ 2,3V;
* Corrente de trabalho: 3 ~ 6mA;
* Faixa de Medição TDS: 0 ~ 1000ppm;
* Precisão de medição TDS: ± 10% FS (25 ℃);
* Tamanho do módulo: 42 \* 32mm;
* Interface do módulo: PH2.0-3P;
* Interface do eletrodo: XH2.54-2P.
* Sonda TDS
  + - Número de agulha: 2;
    - Comprimento total: 83 cm;
    - Interface de conexão: XH2.54-2P;
    - Cor: Preto;
    - Outros: Sonda à prova d’água.

**2.1.2) Visão Geral**

Abaixo pode ser verificadas as conexões do TDS através das numerações da Figura 1, as quais estão descritas na Tabela 1:

|  |
| --- |
|  |
| Figura 1 – TDS   |  | | --- | |  | | Tabela 1 – Descrição das conexões do TDS | |

**2.1.3) Diagrama de Ligação**

Na figura 2 pode ser observada como serão feitas as ligações entre os conectores do TDS e o Arduíno UNO.

|  |
| --- |
|  |
| Figura 2 – Diagrama de ligação Arduíno-TDS |

**3 Desenvolvimento**

**4 Redação da revisão bibliográfica**

Utilizamos a Scielo Brasil, que é uma biblioteca virtual de revistas científicas para buscar informações. Foram observados dez artigos sobre geração e impactos de hidrelétricas, sendo selecionados os cinco que estavam mais relacionados com o tema. Os critérios para seleção foram escorados nos objetivos específicos do trabalho e nas justificavas ambientais para evitar a implementação de barragens, tomando como assuntos chave a capacidade e controle de um gerador, Assim, os artigos escolhidos destacam assuntos sobre a qualidade da água em superfícies em reservatórios de hidrelétricas, a história e memória de locais que foram alagados para produção de energia, potência e força eletromotriz de um gerador, técnicas de controle em um gerador e identificação do elemento fundamental de tensão da rede. A partir disso, foram coletadas informações a fim de entender melhor como se dá o funcionamento de geradores e como a tensão e frequência variam de acordo com sua rotação. Informações as quais são necessárias para prover o controle adequado da pressão e vazão de água na tubulação para suprir a demanda elétrica que estiver ligada na carga. Além disso, foi verificada a importância de gerar energia sem grandes danos ao meio ambiente através de uma alternativa viável, após análise de informações que mostram os impactos que uma hidrelétrica pode causar. Então, mostraremos como podemos nos beneficiar da geração de energia através da água de uma forma diferente, mas que pode trazer grandes benefícios sem a degradação ambiental que uma barragem gera.

Sabemos que a construção de uma usina hidrelétrica traz diversos benefícios devido ao fato de ser considerada uma forma de geração energética limpa, principalmente no Brasil, país com grande abundância de água e rios. Porém, como observado na pesquisa de Costa (2022, p.894), existem outros fatores que muitas vezes não são analisados:

[...] a sua construção está associada a diversos impactos socioambientais, como transformação na estrutura da comunidade aquática (**LI et al., 2019**), alterações no regime fluvial e na qualidade da água (**XIN et al., 2015**), perda de patrimônio genético (fauna e flora) (**DODDS; WHILES, 2020**) e mudanças na hidrodinâmica do transporte de sedimentos (**LI *et al*., 2019**; **MEBARKIA; BOUFEKANE, 2020**). (COSTA, 2022, p.894)

Grandes problemas podem surgir no futuro se uma barragem não for construída com os devidos cuidados, e mesmos assim, sempre deixará grandes impactos socioambientais que serão irreversíveis.

De acordo com André a frequência mecânica e a frequência elétrica variam de acordo com a rotação do gerador, ao passo em que os ímãs de mesma polaridade passam pelas bobinas dos enrolamentos do gerador:

[...] a frequência do rotor, denominada de frequência mecânica (Fm), é igual à frequência com que os ímãs (com a mesma polaridade) passam pelas bobinas, e a mesma da tensão elétrica gerada, denominada de frequência elétrica (Fel). A distinção entre essas frequências ocorre quando, por exemplo, se dobra a quantidade de ímãs no rotor. Para esse caso, o valor da Fel é duplicada em relação a Fm. (ALVES, 2021, p.3)

Ele afirma também que as duas se relacionam proporcionalmente e de acordo com o número de pólos do gerador, onde a frequência mecânica multiplicada pelo número de pólos é igual a duas vezes a frequência elétrica.

Essa informação se torna muito importante quando se visa controlar a frequência de saída de acordo com a rotação do eixo do gerador. Verificamos que para isso precisaremos de uma rotação de 3600 RPM para atender a rede que utilizamos que é de 60Hz.

Segundo Alves (2021, p.8) “[...] a força eletromotriz cresce linearmente com a velocidade de rotação do gerador” o que nos mostra que é possível controlar uma carga ligada ao gerador controlando a velocidade de rotação do mesmo, ou seja, é possível moldá-lo de acordo com a demanda de energia elétrica que pretendemos suprir.

E por consequência de experimentos Alves (2021) nos mostra que a tensão eficaz gerada (Vrms) depende diretamente da resistência da carga (R), da resistência interna do gerador e da força eletromotriz eficaz.

Para controlar a tensão através da velocidade do gerador são necessárias técnicas de controle robustas e seguras, onde podemos utilizar controladores contínuos e com parâmetros fixos projetados com as técnicas que já conhecemos. Como diz Moutinho (2012):

[...] No caso dos Reguladores de Velocidade (RVs) e Reguladores Automáticos de Tensão (RATs), normalmente são utilizados controladores contínuos com parâmetros fixos projetados com técnicas lineares convencionais. Nesse tipo de projeto são utilizados modelos matemáticos lineares obtidos por meio de técnicas de linearização aplicadas a modelos não-lineares representativos do SEP (Larsen and Swann., 1981). (MOUTINHO, 2012, p.571)

Ele afirma também, que esse tipo de controle pode ser feito de forma mais avançada:

[...] O controle de um SEP de forma estável e confiável em várias condições operacionais também pode ser realizado por meio da utilização de estruturas de controle projetadas com base em técnicas avançadas de controle digital. (MOUTINHO, 2012, p.571)

Assim, podemos perceber que a capacidade de geração elétrica de um gerador pode ser controlada através de sua rotação, onde a frequência e a tensão serão fixadas por sistemas de controle que envolvem técnicas conhecidas, sendo elas avançadas ou mais simples, a fim de manter uma carga conhecida ou desconhecida alimentada energeticamente de acordo com sua demanda.

Mas, para fazer esse controle, precisamos de um ponto de referência a fim de mapear os parâmetros do controlador. Para isso, precisamos detectar preferencialmente de forma simultânea a frequência, o ângulo de fase e a amplitude da tensão da rede, como cita:

[...] A detecção da frequência, do ângulo de fase e da amplitude do componente fundamental da tensão da rede é de grande interesse para sistemas de controle, aplicações eletrônicas e sistemas elétricos de potência de uma forma geral. (MOUTINHO, 2012, p.573)

Entretanto, poucos algoritmos conseguem detectar esses três parâmetros em simultaneidade hoje em dia. As dificuldades estão principalmente na rapidez e precisão dos algoritmos em regime permanente. Pesquisas recentes têm conseguido contornar esses problemas através da utilização da Transforma Discreta de Fourier. No estudo de Moutinho (2012) ele consegue usá-la na identificação dos três parâmetros e usá-la para conexão de um gerador síncrono na rede elétrica.

Marcelo mostra em suas análises todos os métodos necessários para se obter a onda e frequência fundamental. É possível verificar também como poderá ser feita a correção da defasagem, sintetização de uma onda unitária e como detectar a amplitude da rede. Por final, é comprovada a eficácia do experimento e do controlador através de simulações.

Concluímos que seria possível instalar um gerador a uma tubulação com água, fazendo seu controle de geração de energia elétrica através da variação de vazão e pressão de água à montante do mesmo. Sua conexão à rede elétrica poderá ser feita com um controlador que observe a frequência, ângulo de fase e amplitude da tensão na carga demandada e altera o fluxo de água no conjunto turbogerador a fim de controlar sua rotação em função do consumo de energia elétrica.

**5 Materiais e métodos**

O estudo foi feito com cunho quantitativo e qualitativo, tendo como base fundamentações teóricas presentes em revistas científicas consultadas online, comparando os dados teóricos de geradores com os impactos sociais e ambientais, a fim de mostrar uma alternativa eficaz e prática para geração de energia elétrica.

Para atingir os objetivos propostos, foi feito um estudo em um rio existente na cidade de Santa Leopoldina, localizada no estado do Espírito Santo, Brasil, em maio de 2023. O local se caracteriza por ser uma região serrana, com declives acentuados e que se mostraram totalmente adequados para implementação do trabalho proposto.

Para isso seguiu-se o cronograma que se inicia no dia 05 de Junho de 2023 da seguinte forma: o primeiro dia foi reservado para instalação da tubulação no rio, o segundo dia feita a base de concreto, terceiro dia instalados a turbina e gerador na base, quarto dia, instalação da fiação do gerador até o inversor e no último dia (09 de junho de 2023) foram feitas todas as medições de geração elétrica.

A primeira análise a ser feita foi a angulação do rio, que se mostrou com grande potencial de geração de energia, tendo ângulos em torno de 20 a 30 graus, o que pôde ser feito de forma visual, sem necessidade de equipamentos sofisticados. Feito isso se iniciou o processo de montagem, onde foram utilizados 6 varões de 75 milímetros de diâmetro e 6 metros de comprimento, da marca Tigre. O ponto de partida foi escolhido mais acima do rio, onde havia empoçamento natural de água. A partir daí, fixou-se o tubo com a boca submersa com filtro, para evitar passagem de folhas de árvores. Foram encaixados todos os varões seguindo pela lateral do rio, a fim de evitar que os mesmos fossem carregados pela água em épocas chuvosas. Entre os tubos foi utilizada uma válvula esfera, também da marca tigre, para controle da passagem de fluxo de água.

Concluído esse processo, iniciou-se o primeiro teste, que consiste na abertura da válvula. Feito isso, percebeu-se grande pressão e fluxo de água, confirmando a eficácia do sistema. Foi verificada também ausência de vazamentos na tubulação, a fim de garantir potência máxima e total aproveitamento de energia potencial da água para a geração. Vale ressaltar, que o desvio de água foi mínimo comparado com o fluxo total do rio, sem contar que ela passa pela tubulação e retorna para o mesmo, o que torna o processo totalmente renovável e sem impactos ambientais.

Na próxima etapa, na parte de baixo do rio, após fechamento da válvula, foram instalados a turbina e o gerador de 1000 Watts, da marca Pelton e Weg respectivamente, em cima de uma base de concreto para suporte do sistema gerador, na qual foram feitas furações com uma furadeira do tipo martelete, onde foram chumbados. Após isso, o eixo de rotação de turbina foi conectado ao do gerador.

O gerador foi conectado a cabos elétricos de 2,5 milímetros e levado até um inversor da marca Weg, o qual poderá futuramente ser conectado em sistema on-Grid (injetado diretamente na rede elétrica da concessionária) ou off-Grid (conectado em baterias). Com o sistema todo instalado, foi feita abertura da válvula esfera, e iniciou-se o processo de geração de energia elétrica.

Enfim, foram feitas análises de tensão e corrente utilizando multímetro e amperímetro. Os resultados foram muito satisfatórios, visto que foi observada uma tensão de 220 Volts constante na saída do gerador. Juntamente com a corrente analisada, verificamos que o gerador estava produzindo cerca de 0,8KWh de energia. Isso resulta em aproximadamente 19KWh por dia e 576KWh por mês. Isso seria o suficiente para abastecer praticamente três casas com consumo moderado de energia elétrica.

Foi feito um balanço aproximado do investimento, que ficou em torno de 4 a 5 mil reais, o que é considerado um capital muito baixo se comparado com as fontes de energia limpa que temos disponíveis hoje no mercado (como a solar e a eólica). A geração ainda superou as expectativas e se mostrou muito mais eficiente, prática e de fácil instalação que essas outras fontes.

O projeto se mostrou coerente com a pesquisa, onde conseguimos manter a rotação da turbina constante, resultando em uma frequência contínua em 60Hz na saída do gerador, conforme disse André (2021). Com o inversor conseguimos manter o controle de tensão em 220 Volts conforme estipulado através do estudo da pesquisa de Moutinho (2012).

Foi possível concluir que a pesquisa confirmou a hipótese de que é viável produzir energia através da água canalizada para uma turbina sem impactos ambientais e de forma barata, rápida e prática. Através do processo, surpreendentemente notou-se que a geração tem potencial muito maior do que o esperado, sendo suficiente para abastecer não só uma casa, mas até três, o que torna o projeto ainda mais barato do que o previsto.

O sistema não foi conectado diretamente a uma casa ou direto na rede elétrica, ficando em aberto e como sugestão a implementação de novos estudos que executem tal ação a fim de comprovar na prática a utilização do projeto em sistema on-Grid ou off-Grid.

**6 Referências**

ALVES, A. L. et al. **Potência e força eletromotriz em um gerador didático de corrente alternada.** **Revista Brasileira de Ensino de Física**, v. 44, 2022. Disponível em : <https://doi.org/10.1590/1806-9126-RBEF-2021-0379>. Acesso em: 01/06/2023.

CARLOS et al. **História e memória da pesquisa e acervo arqueológico da Usina Hidrelétrica de Balbina.** v. 17, n. 2, 1 jan. 2022. Disponível em: <https://doi.org/10.1590/2178-2547-BGOELDI-2021-0043>. Acesso em: 28/05/2023.

MOUTINHO, M. N. et al. **Técnica de controle preditivo baseado em modelo aplicada ao controle de tensão de um gerador síncrono - resultados experimentais.** **Sba: Controle & Automação Sociedade Brasileira de Automatica**, v. 23, n. 5, p. 570–582, out. 2012. Disponível em: :<https://doi.org/10.1590/S0103-17592012000500005>. Acesso em: 10/06/2023.

PÁDUA, M. B. et al. **Metodologia para identificação do componente fundamental da tensão da rede baseada no algoritmo recursivo da TDF**. v. 18, n. 3, p. 381–396, 1 set. 2007. Disponível em: <https://doi.org/10.1590/S0103-17592007000300010>. Acesso em: 01/06/2023.

PRISCILA DA COSTA et al. **Dinâmica espaço temporal da qualidade das águas superficiais de dois grandes reservatórios de usinas hidrelétricas brasileiras.** v. 27, n. 5, p. 893–907, 1 out. 2022. Disponível em: <https://doi.org/10.1590/S1413-415220210233>. Acesso em: 26/05/2023.