



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO, DE CIÊNCIAS EXATAS E EDUCAÇÃO
DEPARTAMENTO DE ENG. DE CONTROLE, AUTOMAÇÃO E COMPUTAÇÃO
CURSO DE GRADUAÇÃO EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Robson Renan Steuck

**Desenvolvimento de software para controlador lógico programável e interface
homem-máquina de uma central geradora hidrelétrica**

Blumenau
2024

Robson Renan Steuck

**Desenvolvimento de software para controlador lógico programável e interface
homem-máquina de uma central geradora hidrelétrica**

Trabalho de Conclusão de Curso de Graduação em Engenharia de Controle e Automação do Centro Tecnológico, de Ciências Exatas e Educação da Universidade Federal de Santa Catarina como requisito para a obtenção do título de Engenheiro de Controle e Automação.

Orientador: Dr. Daniel Martins Lima

Blumenau

2024

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Steuck, Robson Renan

Desenvolvimento de software para controlador lógico programável e interface homem-máquina de uma central geradora hidrelétrica / Robson Renan Steuck ; orientador, Daniel Martins Lima, 2024.

55 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Campus Blumenau, Graduação em Engenharia de Controle e Automação, Blumenau, 2024.

Inclui referências.

1. Engenharia de Controle e Automação. 2. Automação. 3. Controlador lógico programável. 4. Interface homem-máquina. I. Lima, Daniel Martins. II. Universidade Federal de Santa Catarina. Graduação em Engenharia de Controle e Automação. III. Título.

Robson Renan Steuck

Desenvolvimento de software para controlador lógico programável e interface homem-máquina de uma central geradora hidrelétrica

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de “Engenheiro de Controle e Automação” e aprovado em sua forma final pelo Curso de Graduação em Engenharia de Controle e Automação.

Blumenau, 02 de fevereiro de 2024.

Banca Examinadora:

Prof. Dr. Daniel Martins Lima
Universidade Federal de Santa Catarina

Prof. Dr. Guilherme Brasil Pintarelli
Universidade Federal de Santa Catarina

Prof. Dr. Anelize Zomkowski Salvi
Universidade Federal de Santa Catarina

Este trabalho é dedicado a minha família, amigos,
professores e todos aqueles que participaram desta
caminhada.

AGRADECIMENTOS

Agradeço aos meus pais, Darlício Steuck e Lidiane de Souza, que sempre me incentivaram, deram os melhores exemplos e mostraram o caminho correto a ser seguido. Além de toda a minha família que sempre esteve presente em todos os momentos.

Agradeço também a todos os professores que participaram dessa jornada, em especial ao meu professor orientador, Daniel Martins Lima, que prestou apoio em todos os momentos que precisei. Assim como os amigos que fiz durante essa jornada e ficaram para a vida.

Não poderia deixar de agradecer também a WEG, pela confiança depositada ao autorizar o desenvolvimento deste trabalho, bem como os colegas de seção que sempre estiveram disponíveis para esclarecer quaisquer dúvidas.

"A verdade é aquilo que resiste ao teste da experiência"
(Albert Einstein)

RESUMO

Com o tamanho do potencial hídrico que o Brasil possui, é natural o surgimento de empreendimentos que aproveitam desse potencial para gerar energia. As usinas hidrelétricas são empreendimentos que por meio de equipamentos como turbina, gerador e construções como represas aproveitam da topografia local para gerar energia através do movimento da água. A facilitação na obtenção da documentação ambiental, quando comparada as demais hidrelétricas como as pequenas centrais hidrelétricas (PCHs) ou até mesmo as usinas hidrelétricas de energia (UHEs), aliada ao apelo ecológico fazem com que as centrais geradoras hidrelétricas (CGHs) se tornem uma oportunidade de negócio. Neste contexto, este trabalho foi executado dentro de um projeto de expansão de uma CGH feito pela WEG, e tem como objetivo apresentar a metodologia empregada para o correto desenvolvimento do software embarcado no controlador lógico programável (CLP), responsável por gerenciar todo o processo da usina, bem como o software inserido na interface homem-máquina (IHM), dispositivo pelo qual a operação da planta é realizada.

Palavras-chave: Automação; TPW04; Geração de Energia.

ABSTRACT

With the size of the hydroelectric potential that Brazil possesses, the emergence of ventures harnessing this potential to generate energy is natural. Hydroelectric power plants are projects that, through equipment such as turbines, generators, and structures like dams, leverage the local topography to generate energy through the movement of water. The ease in obtaining environmental documentation, when compared to other hydroelectric projects such as small hydroelectric power plants (SHPs) or even Hydroelectric Power Stations (HPSs), coupled with ecological appeal, makes small hydroelectric power plants (SHPs) a business opportunity. In this context, this work was carried out within an expansion project of an SHP undertaken by WEG, aiming to present the methodology employed for the correct development of the embedded software in the programmable logic controller (PLC), responsible for managing the entire plant process, as well as the software embedded in the human-machine interface (HMI), the device through which plant operation is conducted.

Keywords: Automation; TPW04; Power Generation.

LISTA DE FIGURAS

Figura 1 – Esquemático de uma Usina Hidrelétrica.	16
Figura 2 – Turbina Hidráulica Pelton.	18
Figura 3 – Turbina Hidráulica Francis.	19
Figura 4 – Turbina Hidráulica Kaplan.	19
Figura 5 – Exemplo de Hidrogerador CA.	20
Figura 6 – Conduto Forçado a Céu Aberto de uma CGH.	21
Figura 7 – Válvula Borboleta Utilizada em Usinas Hidrelétricas.	22
Figura 8 – Exemplo de Painel Elétrico.	25
Figura 9 – Controlador Lógico Programável Utilizado.	26
Figura 10 – Estrutura Utilizada pelo TPW04 para a Parametrização do Protocolo Modbus RTU.	28
Figura 11 – Exemplo de Parametrização da Rede Modbus RTU no TPW04.	29
Figura 12 – Exemplo de Utilização da Função MBUS no TPW04.	29
Figura 13 – Estrutura da Comunicação Estabelecida para a Função 03 do Protocolo Modbus.	30
Figura 14 – Deslocamento de Byte para Registrador Temporário.	31
Figura 15 – Alocação Final dos Dados Manipulados.	32
Figura 16 – Entrada Digital no Projeto Elétrico.	33
Figura 17 – Exemplo de Associação de DI com uma Memória Auxiliar por Meio da Instrução de SET.	34
Figura 18 – Exemplo de Código para Saída Digital do CLP.	34
Figura 19 – Exemplo de Código para o Desmembramento dos Comandos do Supervisor em Memórias Auxiliares.	36
Figura 20 – Exemplo Espelhamento dos Comandos em Memórias Auxiliares.	36
Figura 21 – Emprego de Válvula Borboleta e Válvula Bypass em uma CGH.	38
Figura 22 – Exemplo de Lógica para Abertura da Válvula Bypass.	38
Figura 23 – Exemplo de Lógica para que o Sincronismo Seja Habilitado.	39
Figura 24 – Exemplo de Organização de Símbolos em IHM.	41
Figura 25 – Símbolo de Bomba Utilizado em Projeto	42
Figura 26 – Estados de Um Símbolo Animado	43
Figura 27 – Configuração de Objeto para Abertura de uma Popup.	44
Figura 28 – Popup para Detalhamento de Bomba UHLM.	45
Figura 29 – Parametrização de Escravos no Ambiente de Desenvolvimento da IHM.	46
Figura 30 – Comunicação com Escravo Feita Diretamente por Endereçamento.	47
Figura 31 – Criação de Alarmes na IHM.	48
Figura 32 – Diagrama Unifilar UG3.	51
Figura 33 – Gerador UG3.	52

Figura 34 – Entradas e Saídas Analógicas UG3. 52

LISTA DE ABREVIATURAS E SIGLAS

AI	Analog Input
AIO	Analog Input/Output
AO	Analog Output
CGH	Central Geradora Hidrelétrica
CLP	Controlador Lógico Programável
DI	Digital Input
DIO	Digital Input/Output
IHM	Interface Homem Máquina
MBUS	Função Utilizada para a Comunicação via Modbus RTU por Meio de RS-485 no CLP TPW04
SCADA	Supervisory Control and Data Acquisition
setpoint	Valor de Referência
timeout	Tempo Máximo para a Confirmação de Envio/Recebimento de Dados
word	Registrador de 16 Bits

SUMÁRIO

1	INTRODUÇÃO	14
1.1	OBJETIVOS	14
1.1.1	Objetivo Geral	14
1.1.2	Objetivos Específicos	15
1.2	ESTRUTURA DO DOCUMENTO	15
2	REVISÃO BIBLIOGRÁFICA	16
2.1	USINA HIDRELÉTRICA	16
2.2	CENTRAL GERADORA HIDRELÉTRICA - CGH	17
2.3	TURBINA	17
2.4	GERADOR	19
2.5	CONDUTO FORÇADO DE ÁGUA	20
2.6	VÁLVULA BORBOLETA	21
2.7	SUBESTAÇÃO	22
2.8	CONTROLADOR LÓGICO PROGRAMÁVEL	23
3	DESENVOLVIMENTO	24
3.1	DIMENSIONAMENTO DO PROJETO	24
3.2	PROJETO ELÉTRICO	24
3.3	CONTROLADOR LÓGICO PROGRAMÁVEL	25
3.3.1	Hardware	25
3.3.2	Comunicação	27
3.3.3	Tratamento dos Dados Recebidos	30
3.3.4	Software	33
3.3.4.1	<i>Entradas e Saídas</i>	33
3.3.4.2	<i>Comandos Oriundos da IHM/Supervisório</i>	35
3.3.5	Descritivo Funcional	37
3.3.6	Documentação	40
3.4	INTERFACE HOMEM MÁQUINA	40
3.4.1	Hardware	40
3.4.2	Dimensionamento de Telas	40
3.4.3	Criação de Telas	41
3.4.4	Símbolos	41
3.4.5	Animação	42
3.4.6	Popups	43
3.4.7	Comunicação	45
3.4.7.1	<i>Parametrização dos Equipamentos</i>	46
3.4.7.2	<i>Endereçamento</i>	46
3.4.8	Alarmes	48

4	RESULTADOS	50
4.1	CLP	50
4.2	IHM	50
5	CONCLUSÃO	54
	REFERÊNCIAS	55

1 INTRODUÇÃO

A diferença de altitude entre dois pontos associado à uma massa de água, é capaz de fornecer um potencial energético para a geração de energia. Existem inúmeras maneiras de aproveitar a energia proveniente do movimento de massas. No Brasil, a mais difundida e conhecida pela população são as usinas hidrelétricas de acumulação, que são famosas pela grande área que ocupam para o armazenamento de água. No entanto, outra forma de gerar energia através desse mesmo princípio físico consiste em criar uma diferença de nível por meio de condutos, os quais podem se estender por mais de um quilômetro, desta forma, eliminando a necessidade de grandes reservatórios de água. Por esse e diversos outros motivos, as usinas hidrelétricas de fluxo contínuo estão se tornando cada vez mais relevantes no cenário energético nacional. Como prova disso estão as 704 centrais geradoras hidrelétricas (CGHs) que estão em pleno funcionamento no Brasil (ABRAPCH, 2024).

A WEG S.A., empresa com 62 anos de história, destaque mundial no segmento de motores elétricos, possui projetos no ramo de geração de energia, o que a fornece a base de conhecimentos necessária para o correto funcionamento de uma usina hidrelétrica. A empresa foi contratada para a realização de uma ampliação no sistema de geração do cliente, que já contava com duas turbinas em operação. O fornecimento previsto pela WEG foi o de painéis elétricos e outros equipamentos, tais como regulador de velocidade, regulador de tensão além do desenvolvimento do projeto elétrico e software para o CLP e IHM. Esses são alguns dos serviços que a WEG ficou encarregada de fornecer.

Para que um projeto que integra diversos equipamentos eletromecânicos funcione, existe a necessidade de um sistema de controle e monitoramento atuando constantemente. Alguns equipamentos são tipicamente empregados nessa tarefa, tais como CLP e IHM. Ambos os dispositivos necessitam do desenvolvimento de softwares específicos para o seu funcionamento. O autor deste trabalho de conclusão de curso atua na seção de Software Aplicativo para projetos de média tensão e sistemas, que se localiza na extensão do parque fabril II da WEG em Jaraguá do Sul/SC. E por isso, foi designado para o desenvolvimento do software que será implementado no CLP e IHM do projeto, softwares esses que compõem o escopo deste trabalho.

1.1 OBJETIVOS

Esta seção apresenta os objetivos geral e específicos que servem de base para o presente trabalho.

1.1.1 Objetivo Geral

Este trabalho tem como objetivo geral desenvolver a programação do CLP que controlará o sistema de geração do cliente, bem como o software implementado na IHM

que será utilizada para a operação da planta.

1.1.2 Objetivos Específicos

Ao levar em consideração o escopo proposto do projeto, os seguintes objetivos específicos são elencados:

- Analisar a elaboração do projeto elétrico que servirá de base para a programação do controlador lógico programável (CLP);
- Desenvolver a programação do controlador lógico programável, que controlará a operação da central geradora hidrelétrica (CGH), com base na documentação disponível;
- Elaborar o software que será implementado na interface homem-máquina.

1.2 ESTRUTURA DO DOCUMENTO

O presente trabalho foi dividido em 5 capítulos. Sendo que o atual tem o objetivo de contextualizar, dar uma breve introdução e definir os principais objetivos do projeto. No Capítulo 2 é realizada a revisão bibliográfica que trata de temas que servem de base para a compreensão do trabalho. O Capítulo 3 apresenta o desenvolvimento do projeto, necessário para atingir os objetivos propostos no Capítulo 1. No quarto capítulo são apresentados os resultados obtidos. Por fim, no Capítulo 5 são expressadas as considerações finais e conclusões do trabalho.

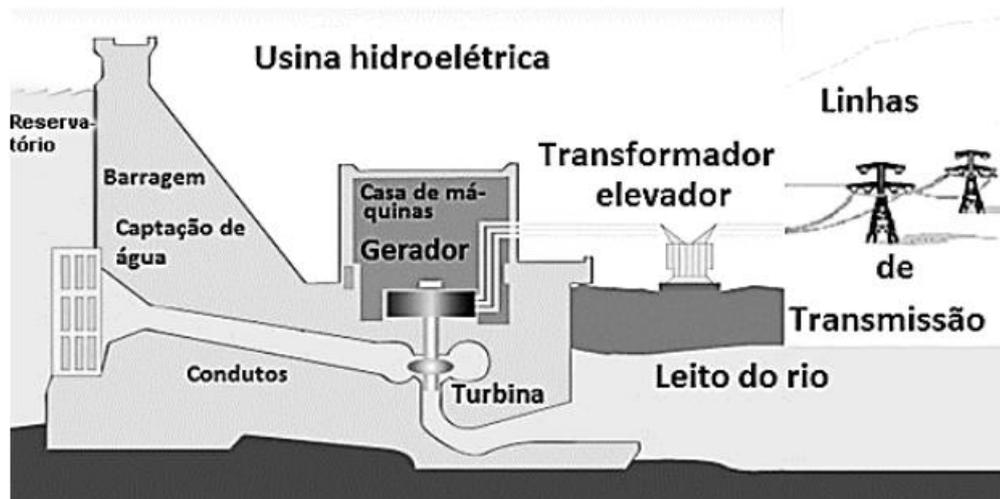
2 REVISÃO BIBLIOGRÁFICA

A seguir serão discutidos temas relacionados ao projeto sobre o qual o presente trabalho foi desenvolvido. Tendo como objetivo explicar o funcionamento dos principais componentes e procedimentos do projeto.

2.1 USINA HIDRELÉTRICA

As usinas hidrelétricas são construções que aproveitam as condições geológicas de determinados lugares, para que através da água em movimento, seja possível gerar energia. E o processo físico utilizado na geração de energia hidrelétrica é o de conservação de energia.

Figura 1 – Esquemático de uma Usina Hidrelétrica.



Fonte: Geração Hidroelétrica e Eólicoelétrica, 2024.

A partir da figura 1 existe uma diferença de altitude entre o reservatório e a turbina. O que dá origem à energia potencial gravitacional:

$$E = mgh \quad [J] \quad (1)$$

Em que m refere-se à massa, g à força gravitacional e h à altura. Note que quanto maior a massa e maior a altura, maior será a energia potencial gravitacional.

Outro ponto importante relacionado ao potencial de geração do local de instalação de uma usina, é a vazão que este é capaz de oferecer. Esta pode ser encontrada fazendo:

$$Q = \frac{V}{t} \quad \left[\frac{m^3}{s} \right] \quad (2)$$

Sendo que V refere-se ao volume de água e t ao tempo. Note que a equação de massa é dada por:

$$m = \rho V \quad [kg] \quad (3)$$

E associando a energia potencial gravitacional com o tempo, encontra-se a equação de potência:

$$\frac{E}{t} = \frac{mgh}{t} \quad [W] \quad (4)$$

Substituindo m por ρV encontra-se a equação:

$$\frac{E}{t} = \frac{\rho V gh}{t} \quad [W] \quad (5)$$

Observe que fica claro a associação da (5) com (3), da onde encontra-se a equação final (6):

$$P = Q\rho gh \quad [W] \quad (6)$$

Através da equação (6) fica explícito a relação entre a potência gerada pela usina com a diferença de altura entre a captação de água e a saída da turbina. Quanto maior a vazão e maior a energia potencial gravitacional gerada pela diferença de altura entre os dois pontos, maior será a capacidade de geração da planta (P.; MOURA; ROCHA, 2019).

2.2 CENTRAL GERADORA HIDRELÉTRICA - CGH

A CGH, consiste em uma usina hidrelétrica de pequeno porte, sendo consideradas as usinas com capacidade de geração de até 5 MW (ANEEL, 2024). Pelo seu tamanho reduzido, a infraestrutura necessária para a construção acaba sendo reduzida também, o que por sua vez abre uma ampla gama de possibilidades de locais para aplicação. As CGHs não necessitam de barragem de acumulação, a diferença de altitude entre a tomada da água e a turbina acaba sendo capaz de produzir energia cinética suficiente para vencer a inércia da turbina. Por isso o impacto ambiental gerado pela usina acaba sendo menor, uma vez que, por exemplo, o rio não precisa ser represado, bastando apenas o desvio de uma parte do seu fluxo. Um ponto muito importante e vantajoso na construção de uma CGH é o aspecto burocrático, uma vez que justamente pela sua infraestrutura e impacto ambiental reduzidos, não se faz necessário a concessão, permissão ou autorização junto a ANEEL (HIDROENERGIA, 2024) .

2.3 TURBINA

De acordo com (P.; MOURA; ROCHA, 2019) turbinas são equipamentos que podem gerar energia de uso mecânico e acionam certos tipos de máquinas. Trazendo os principais tipos de turbinas:

- Turbinas de ação são equipamentos que recebem energia na forma mecânica-hidráulica e a convertem em energia mecânica-motriz. A energia potencial é transformada em energia cinética antes de atingir às conchas do rotor da turbina. Atualmente, a turbina do tipo Pelton é a turbina de ação mais difundida. As turbinas Pelton possuem um ou mais injetores que jogam a água contra as pás e podem ser empregadas em alturas topográficas superiores a 1500m. A Figura 2 demonstra a sua estrutura.

Figura 2 – Turbina Hidráulica Pelton.



Fonte: WEG, 2024.

- Já as turbinas de reação, são equipamentos que convertem energia na forma mecânica-hidráulica para a forma mecânica-motriz. No caso das turbinas de reação, a energia na forma cinética é usada na turbina. Exemplos desse tipo de turbina são as turbinas do tipo Francis e Kaplan. As turbinas Francis são construídas para grandes aproveitamentos, tendo rendimentos superiores a 92% em grandes máquinas. Já as turbinas Kaplan são usadas para operar em desníveis de até 60m, sendo o rotor a diferença entre as turbinas Kaplan e Francis. As turbinas Kaplan possuem um servomotor para variar o ângulo de inclinação das pás. Na sequência, a Figura 3 e Figura 4 representam esses dois tipos de turbinas:

Figura 3 – Turbina Hidráulica Francis.



Fonte: WEG, 2024.

Figura 4 – Turbina Hidráulica Kaplan.



Fonte: WEG, 2024.

2.4 GERADOR

O gerador é amplamente utilizado na geração de energia elétrica. Esse dispositivo tem o seu princípio de funcionamento baseado no eletromagnetismo. Primeiramente se faz necessário entender os principais componentes que compõem um gerador de energia elétrica, que são eles: estator e rotor.

O estator é um componente fixo do gerador, no qual estão dispostas as bobinas pelas quais a energia elétrica será coletada. Já o rotor permanece em rotação durante o funcionamento do gerador. Utilizando o princípio da indução magnética descoberta por Michael Faraday, o qual diz que quando um fluxo magnético passando por uma bobina

de N espiras sofre uma variação, existe então a indução de uma força eletromotriz em cada espira, e por sua vez, a força eletromotriz total é o resultado da soma dessas forças eletromotrizes (FUNDAMENTOS DA FÍSICA - ELETROMAGNETISMO, 2016).

Figura 5 – Exemplo de Hidrogerador CA.

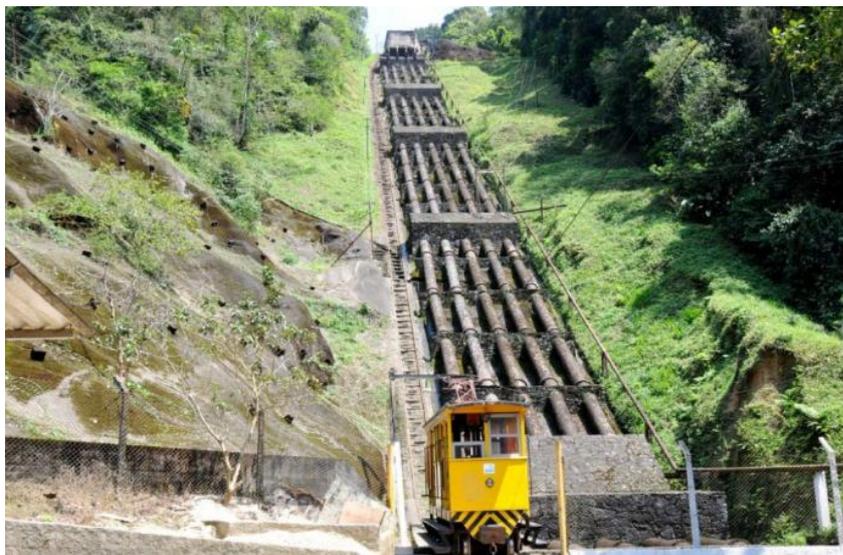


Fonte: WEG, 2024.

2.5 CONDUITO FORÇADO DE ÁGUA

Uma parte do fluxo da água de um rio é desviado do seu curso natural, através da tomada de água, ela entra então no conduto forçado de água, estes são condutos fechados para o caso de média e alta pressão de trabalho eles usualmente são fabricados em aço e suas partições são soldadas. Conforme percorre-se a diferença de nível entre a barragem a turbina, a pressão aumenta, de modo que a maior pressão de água é registrada na entrada da turbina (P.; MOURA; ROCHA, 2019). A figura 6 mostra este tipo de tubulação no conduto forçado a céu aberto de uma CGH.

Figura 6 – Conduto Forçado a Céu Aberto de uma CGH.



Fonte: CGH Libera Maria, 2024.

2.6 VÁLVULA BORBOLETA

Após percorrer todo o percurso do conduto forçado, temos energia cinética associada à massa de água. Por isso o seu contato direto com a turbina precisa ser evitado, uma vez que os danos ao equipamento seriam grandes. Esse é um dos motivos pelo qual utiliza-se a válvula borboleta, dispositivo acoplado próximo ao final do conduto forçado que bloqueia o fluxo de água. Pode-se elencar também como motivação o isolamento do fluxo em manutenção, além do sistema de fechamento por contra peso, fundamental em uma situação de emergência (HYDROWHEEL, 2024). A Figura 7 mostra um exemplo da válvula.

Figura 7 – Válvula Borboleta Utilizada em Usinas Hidrelétricas.



Fonte: HydroWheel, 2024.

Para que a água possa chegar até a turbina de maneira segura, a pressão dos dois lados da válvula borboleta precisa ser equalizada de maneira gradual. Essa equalização acontece por meio de uma válvula bypass, a qual interliga os dois lados da válvula borboleta e permite o fluxo gradual de água. No momento em que o processo é finalizado, a válvula principal inicia a sua abertura, até o momento em que temos 100% do fluxo de água passando por ela. Outra função muito importante deste equipamento é o bloqueio do fluxo de água no caso de falha no sistema. Pois em certos casos, o movimento da água precisa ser cessado imediatamente.

2.7 SUBESTAÇÃO

Os níveis de tensão provenientes do gerador de uma usina não se enquadram nos padrões de fornecimento da rede elétrica. Por isso, são necessários a utilização dos equipamentos que compõem uma subestação para, por exemplo, adequar o nível de tensão à um valor em que essa energia possa ser transmitida. A subestação desempenha também o papel de controle e proteção do circuito de geração, sendo capaz de isolar uma eventual falha. Na subestação são empregados diversos equipamentos eletroeletrônicos, como por exemplo o transformador (dispositivo responsável pelo aumento ou rebaixamento da tensão) e relé de proteção, que realiza o monitoramento de vários parâmetros da barra de geração e atua ao encontrar alguma desconformidade (P3 ENGENHARIA ELÉTRICA,

2018).

2.8 CONTROLADOR LÓGICO PROGRAMÁVEL

O controlador lógico programável CLP, é um equipamento que possui hardware e software compatível com aplicações industriais. Através do uso de linguagens de programação que podem ser implementadas em seu software, ele pode executar funções aritméticas, lógicas, de temporização, de contagem, entre outras. O seu hardware possui entradas para aquisição de dados e saídas para o acionamento de diversos equipamentos ou operações.

Desde o seu surgimento, o CLP revolucionou os processos industriais, uma vez que já não se fazia mais necessário a construção de painéis compostos por relés eletromagnéticos com um único propósito.

Até então, cada necessidade de mudança no processo significava a alteração física da estrutura dos painéis, o que além de aumentar os custos do processo, ainda levava tempo para ser realizado. Com a criação do CLP, tornou-se possível a reprogramação do processo, transformando as mudanças de hardware em mudanças de software (E-TEC BRASIL, 2016).

Segundo a norma (IEC_61131-3, 2003), o desenvolvimento do software do controlador pode ser desenvolvido por meio de quatro linguagens, sendo duas textuais e duas gráficas. A escolha da linguagem a ser utilizada pelo programador depende de alguns fatores, tais como:

- Formação do programador;
- Problema a resolver;
- Nível da descrição do problema;
- Estrutura do sistema de controle;
- Interface com outras pessoas/departamentos.

A linguagem de programação utilizada durante o desenvolvimento do presente trabalho será a de Diagrama Ladder LD, que é baseada na representação gráfica da lógicas de relés.

3 DESENVOLVIMENTO

Durante o processo de venda do projeto foi estabelecido que o software a ser implementado no CLP e na IHM seria desenvolvido pela WEG. A seção responsável pelo seu desenvolvimento foi a de software aplicativo, na qual o autor do presente trabalho atua, sendo responsável pelo desenvolvimento da lógica a ser empregada no software para ambos os equipamentos, CLP e IHM.

Nesse capítulo, as etapas de desenvolvimento do produto são apresentadas. Iniciando com a metodologia utilizada para a confecção do software do CLP, e em um segundo momento o software implementado na IHM.

3.1 DIMENSIONAMENTO DO PROJETO

A partir da coleta de informações sobre o local da usina. Toda a situação é analisada, por exemplo, topografia do local e quantidade de energia a ser gerada. Com esse processo concluído inicia-se o dimensionamento dos equipamentos necessários, turbina, gerador, válvulas e todos os demais equipamentos que vão desde a infraestrutura até o sistema de supervisão.

3.2 PROJETO ELÉTRICO

A base utilizada para o desenvolvimento de software é o projeto elétrico, elaborado de acordo com o levantamento das necessidades do cliente. De maneira geral tem-se como etapas do projeto elétrico:

- a) Levantamento de todos os equipamentos necessários para o projeto e barramentos de alimentação;
- b) Conexões entre entradas e saídas de equipamentos;
- c) Criação do layout dos equipamentos;

A Figura 8 ilustra a disposição dos equipamentos dentro do painel elétrico, assim como na parte inferior pode-se observar os borners citados anteriormente.

Figura 8 – Exemplo de Painel Elétrico.



Fonte: WEG, 2024.

3.3 CONTROLADOR LÓGICO PROGRAMÁVEL

Ao decorrer da seção serão expostos ao leitor os principais componentes do desenvolvimento do software projetado pelo autor. Que se inicia com a captação dos dados provenientes do projeto elétrico, passa pela parametrização da comunicação com os demais equipamentos além do desenvolvimento da lógica de operação da usina e chegando então na etapa de documentação.

3.3.1 Hardware

O CLP é composto por diversos componentes de hardware que juntamente ao aparato de software definem as suas principais características, tais como número de entradas/saídas, rede de comunicação, capacidade de processamento e desempenho de software.

Para o projeto estudado no presente trabalho, optou-se por utilizar o controlador lógico programável “TPW04” no modelo 360BR-D, fabricado pela empresa WEG. Para a sua escolha, além do custo benefício, levou-se em consideração o CLP que já estava em operação nas outras duas usinas do cliente, uma vez que isto resulta na redução do número de peças de reposição necessárias.

Por padrão são disponibilizadas três portas para comunicação, sendo que uma delas conta com uma porta Mini-USB que faz uso do protocolo USB para se conectar ao computador, e as outras duas se comunicam por meio do protocolo Modbus RTU, sendo que uma delas utiliza como meio físico RS-485 e a outra RS-232. Outras portas de comunicação podem ser utilizadas de acordo com o módulo de expansão empregado, sendo que existe a possibilidade de trabalhar com Modbus RTU utilizando as interfaces RS-485 ou RS-232, além de CANopen e TPC/IP por meio de Ethernet.

O modelo selecionado conta com trinta e seis entradas e vinte e quatro saídas digitais de fábrica. Durante o dimensionamento do projeto, se fez necessário a inclusão cinco módulos de entradas e dois de saídas digitais, além de dois módulos de entradas e uma de saída analógicas. Totalizando então cento e dezesseis entradas e oitenta saídas digitais. Já quanto às analógicas foram dezesseis entradas e duas saídas.

Os protocolos de comunicação utilizados foram Modbus TCP/IP por meio de ethernet e Modbus RTU utilizando como meio físico RS-485. A alimentação tanto do controlador quanto dos seus módulos de expansão são feitas todas com 24 Vcc. A Figura 9 mostra o CLP que foi utilizado no projeto.

Figura 9 – Controlador Lógico Programável Utilizado.



Fonte: Autor.

3.3.2 Comunicação

O protocolo de comunicação adotado no projeto para realizar a comunicação entre a maior parte dos equipamentos foi o Modbus TCP/IP por meio de ethernet. Por conta de sua facilidade de configuração, estabilidade da comunicação, além da possibilidade de grandes distâncias entre os equipamentos. Porém, alguns dispositivos utilizados no projeto ainda não disponibilizam estrutura para o uso desse protocolo, e por isso optou-se por construir uma rede de comunicação exclusiva para esses equipamentos, utilizando então o protocolo Modbus RTU por meio de RS-485.

Cada equipamento pode atuar como mestre ou escravo em determinadas situações. No caso da rede TCP/IP o CLP irá atuar somente como escravo, ou seja, respondendo as solicitações feitas pelos demais equipamentos. Porém no caso da rede Modbus RTU, o controlador será o mestre da rede.

Existem pontos que devem ser observados para o correto funcionamento da comunicação, dentre eles temos, por exemplo:

- A porta de comunicação que será utilizada no hardware deve ser apontada.
- Os parâmetros necessários para o protocolo devem ser configurados e estar idênticos em cada equipamento.
- Somente um aparelho deverá utilizar a rede por vez.
- As configurações em relação ao mapa Modbus de cada dispositivo deve ser respeitada.
- É necessário realizar o monitoramento da rede durante a comunicação, para evitar que um único aparelho ocupe a rede indefinidamente.

Por conta disto, utiliza-se uma rotina criada especialmente para cada projeto, em que todo o fluxo de informações é controlado. O primeiro passo deve ser o apontamento da porta utilizada para a comunicação via Modbus RTU.

Em seguida é feita a escolha dos parâmetros vinculados ao protocolo, como a velocidade da rede, quantidade de bits de paridade, bits de parada, etc. Por padrão, o TPW04 faz uso de um registrador de 16 bits específico em que o valor adicionado a ele representa todas as configurações pertinentes ao protocolo. A Figura 10 apresenta o funcionamento dessa estrutura de configuração, em que cada parâmetro compõem uma parte dos bits da word que será enviada ao registrador.

Figura 10 – Estrutura Utilizada pelo TPW04 para a Parametrização do Protocolo Modbus RTU.

Número do bit	Nome	Conteúdo	
		0 (Bit OFF)	1 (Bit ON)
B0	Comprimento do dado.	7 bits	8 bits
B1 B2	Paridade.	B2, B1:	(0,0):Nenhum (0,1):(IMPAR) (1,1):(PAR)
B3	Bit de parada.	1 bit	2 bits
B4 B5 B6 B7	Velocidade de comunicação (bps).	B7, B6, B5, B4:	(0, 1, 1,0):4800 (0, 1, 1,1):9600 (1, 0, 0,0):19200 (1, 0, 0,1):38400 (1, 0, 1,0):57600 (1, 0, 1,1):115200 (1, 1, 0,0):128000 (1, 1, 0,1):307200 (1, 1, 1,0):500000
B8 ⁽¹⁾	Caractere inicial.	Não	Sim, valor inicial: STX (02 H).
B9 ⁽¹⁾	Caractere final.	Não	Sim, valor inicial: ETX (03 H).
B10 a B15 ⁽²⁾	Indisponível.	–	–

Fonte: Manual de programação TPW04, 2015.

Então, por exemplo, no caso em que tivéssemos:

- Comprimento do dado = 8 bits
- Paridade = ímpar
- Bit de parada = 2 bits
- Velocidade de comunicação = 115200 bps
- Caractere inicial = não
- Caractere final = não

A palavra word, escrita do ponto de vista de bits, seria composta da seguinte maneira: 0000 0000 1011 1011.

Convertendo para decimal, temos: 187.

O registrador reservado para os parâmetros de comunicação Modbus RTU é o D6001. Portanto, para tal, basta mover a constante 187 para D6001. Como mostra a Figura 11, no momento em que o CLP vai para o modo *run*, que indica que o controlador está realizando a operação completa, a memória M8000 é acionada e os parâmetros da rede Modbus RTU é enviado ao registrador D6001.

Figura 11 – Exemplo de Parametrização da Rede Modbus RTU no TPW04.



Fonte: Autor, 2024.

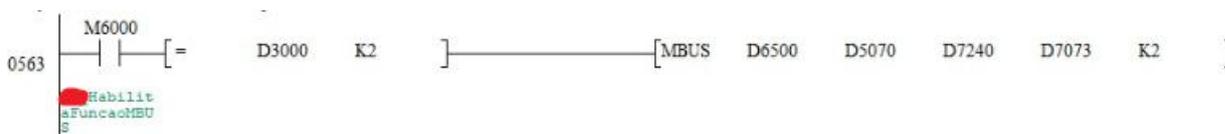
O próximo passo diz respeito às configurações relacionadas a cada escravo, sendo elas:

- Endereço do escravo
- Função Modbus
- Endereço do primeiro parâmetro na tabela de comunicação do escravo
- Quantidade de endereços consecutivos a serem lidos
- Endereço do primeiro registrador do CLP em que os dados recebidos começaram a ser escritos, ou no caso da função Modbus de escrita, o primeiro registrador a conter os dados a serem transmitidos.

Cada escravo deverá possuir suas próprias configurações. É possível utilizar a função MBUS para efetivar a comunicação. Porém é necessário realizar o gerenciamento da rede para o seu correto funcionamento. A lógica desenvolvida no software considera que pode haver comunicação entre apenas dois pontos por vez, pois se trata de comunicação serial. Ou seja, ele maneja todos os telegramas de maneira que somente um seja efetivado por vez. Outro fator muito importante diz respeito a falha de comunicação. Para isso, utiliza-se um *timeout*, que nada mais é do que um tempo limite no qual a função Modbus deve ser executada. Ou seja, por exemplo: no momento em que a função MBUS é acionada, um temporizador é inicializado, e enquanto a comunicação ocorre ele permanece ativo, se todo o processo é finalizado em um tempo menor do que o *timeout*, esse temporizador é resetado. Porém, caso contrário, uma ação é tomada: o telegrama atual é descartado e inicia-se uma nova chamada da função MBUS com os dados relativos ao próximo telegrama.

A Figura 12, demonstra um exemplo da utilização da função MBUS:

Figura 12 – Exemplo de Utilização da Função MBUS no TPW04.



Fonte: Autor, 2024.

No registrador 3000 deve constar o número da porta serial utilizada, caso o valor que consta em D3000 for igual a 2 (correspondente ao slot 2 das expansões de comunicação do hardware do CLP), então a função MBUS será liberada para ser executada.

Como primeiro parâmetro da função, o endereço do escravo e função Modbus devem constar em D6500. Em D5070 teremos o número de words que deverão ser lidas em sequência, já o valor disposto em D7240 deve corresponder ao primeiro endereço de registrador no qual os dados recebidos irão ser armazenados, enquanto em D7073 estará armazenado o número de words que deverão ser recebidas. Por fim, temos a constante $K2$, que indica à função MBUS o slot do CLP por onde a comunicação deverá ocorrer.

3.3.3 Tratamento dos Dados Recebidos

Finalizado o telegrama, a própria função MBUS aciona uma memória interna, M8323. O acionamento dessa memória representa para o software que o tratamento dos dados recebidos deve iniciar. Se faz necessária a manipulação das informações recebidas do escravo, uma vez que o protocolo Modbus RTU estabelece uma estrutura de comunicação a qual os equipamentos devem seguir para conseguirem se trocar informações de forma correta. O próprio protocolo trabalha com as informações divididas em bytes, porém os registradores do CLP trabalham com words, ou seja, 2 bytes. A Figura 13 demonstra a razão pela qual a manipulação desses dados é necessária.

Figura 13 – Estrutura da Comunicação Estabelecida para a Função 03 do Protocolo Modbus.

Request		Response	
Field Name	(Hex)	Field Name	(Hex)
Function	03	Function	03
Starting Address Hi	00	Byte Count	06
Starting Address Lo	6B	Register value Hi (108)	02
No. of Registers Hi	00	Register value Lo (108)	2B
No. of Registers Lo	03	Register value Hi (109)	00
		Register value Lo (109)	00
		Register value Hi (110)	00
		Register value Lo (110)	64

Fonte: Modbus Organization, 2012.

O primeiro byte do valor recebido do escravo que de fato será um dos dados requisitados, é o quarto byte do telegrama. Ou seja, o que seria a informação contida em um único registrador de 2 bytes no escravo, agora está dividida em duas words diferentes. E, por isso o tratamento dessas informações recebidas é feito, para que essas words sejam reordenados e as dados lidos pelo CLP possam ser interpretados de maneira correta.

Com esse intuito, o seguinte procedimento é feito no software do CLP:

1. Deslocar a parte baixa da word em que o primeiro byte de dados está alocado para um registrador temporário.

Para isso, o CLP utiliza a operação lógica WAND, que consiste na comparação *and* entre duas *words* assim como a Figura 14 ilustra.

Figura 14 – Deslocamento de Byte para Registrador Temporário.



Fonte: Autor, 2024.

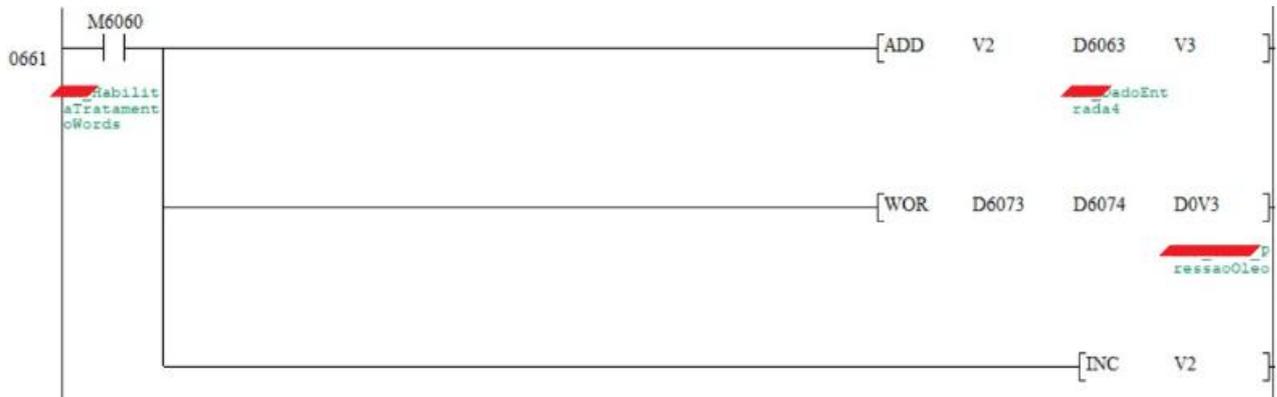
É realizada a operação *AND* entre o conteúdo do registrador D6251 e *HFF00*, em que *H* indica que os caracteres a seguir estão em hexadecimal, e *FF00*, que convertendo para binário: 1111111000000000. Ou seja, essa parte do código implementa o filtro da parte alta dos dados contidos em D6251. O resultado dessa operação será armazenado temporariamente também no byte mais significativo de D6073.

Da mesma maneira é realizado o filtro do byte menos significativo do registrador que contém a segunda parte da informação que no escravo compunha uma única *word*. Porém o seu conteúdo é enviado para ao byte menos significativo de um segundo registrador auxiliar, D6074. Note que para isso utiliza-se o código hexadecimal *FF*, que convertendo para binário: 0000000011111111.

Se faz necessário explicar que *V2* funciona como um offset, iniciando em 0. Implementado para automatizar o tratamento de maneira que com as mesmas linhas de código consiga-se percorrer todas as *words* que foram lidas do escravo.

2. Feito o deslocamento dos bytes de dados para os registradores auxiliares, será necessário apenas concatena-los para que tenhamos o resultado final da manipulação. A Figura 15 mostra como esse processo é feito.

Figura 15 – Alocação Final dos Dados Manipulados.



Fonte: Autor, 2024.

Um novo offset é adicionado $V3$, e ele funcionará como um ponteiro, indicando o endereço final do registrador em que os dados serão concatenados. Diferente de $V2$, ele iniciará com o valor do primeiro endereço final da sequência de dados. Para a concatenação, basta realizar a operação WOR , nos registradores auxiliares e enviar o resultado para o registrador no qual o dado ficará disponível para uso dentro da lógica do software. Por exemplo, caso os seguintes valores estivessem dispostos em $D6073$ e $D6074$:

$D6073 = 0000100000000000 = 2048$ (Dec.);

$D6074 = 0000000010101100 = 172$ (Dec.);

O resultado final da operação WOR seria:

$0010100010101100 = 2220$ (Dec.)

No final do ciclo do tratamento de dados atual, basta incrementar o valor de $V2$, e fazer com que a operação recomece a trabalhar nos registradores auxiliares (linha 0643), até que a quantidade de repetições seja igual ao número de dados (*words*) recebidos do escravo.

A manipulação de dados diz respeito a dados do tipo inteiro, de 16 bits. Porém o projeto exige que também seja trabalhado com dados do tipo ponto flutuante (*float*), para os quais se faz necessário o uso de 32 bits para o seu armazenamento. Nos escravos utilizados esses 32 bits são obtidos dividindo o valor a ser armazenado em duas partes de 16 bits, ou seja, em dois registradores em sequência. O que faz com que a leitura desses dados quando feita pelo CLP, necessite de um tratamento específico para tal. Assim como no caso de dados do tipo float, cada função Modbus implementada irá passar pelo processo de tratamento de dados que condiz com o seu contexto.

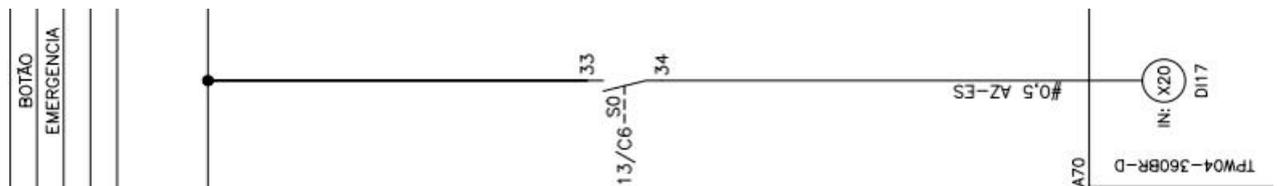
3.3.4 Software

3.3.4.1 Entradas e Saídas

A programação do CLP realizada por meio do software TPW-PCLINK se inicia com a associação dos terminais físicos do hardware com as suas memórias internas, que por sua vez serão utilizados durante a construção da lógica do software do controlador. Esse primeiro passo acontece da seguinte forma:

1. Localização do Sinal no Hardware do CLP: no projeto elétrico estão dispostos todos os detalhes necessários para que se identifique qual sinal está associado a cada conexão física dos equipamentos. Por exemplo, na Figura 16 há o contato do botão de emergência está associado a entrada digital 17 do CLP.

Figura 16 – Entrada Digital no Projeto Elétrico.



Fonte: Autor, 2024.

2. Associação da conexão física com uma memória interna do controlador: cada entrada e saída digital do CLP já possui uma memória previamente associada. Sendo que os contatos do tipo *X* correspondem ao status das entradas digitais (DIs) e *Y* as saídas digitais (DOs). Durante o desenvolvimento da lógica alguns dos status de entradas devem corresponder obrigatoriamente ao nível de acionamento atual da DI, porém existem casos em que se faz necessário que o pulso em uma das entradas faça com que um memória permaneça em nível lógico alto, até que por meio da lógica, seja reiniciada. E, por isso, utilizam-se as memórias auxiliares *M*, do próprio CLP para manter o sinal recebido do hardware. Essas memórias, como o próprio nome sugere, servem para auxiliar no processamento de dados no CLP, pois são internas ao software e não possuem nenhuma associação com DI/Os. Podem assumir somente o valor de ligado ou desligado, além de que no caso das memórias que vão de M500 a M1023 existe ainda um recurso que pode ser crucial no desenvolvimento da lógica, pois essas memórias são retentivas, ou seja, por mais que o CLP seja desenergizado, o seu status de acionamento é mantido.

Figura 17 – Exemplo de Associação de DI com uma Memória Auxiliar por Meio da Instrução de SET.



Fonte: Autor, 2024.

Assim como disposto na Figura 17 utiliza-se a função SET para manter o eventual sinal alto da entrada digital na memória auxiliar M0.

No caso das saídas digitais, também são utilizadas as memórias auxiliares, porém nesse caso temos a situação contrária, em que as memórias M acionam as saídas Y.

Figura 18 – Exemplo de Código para Saída Digital do CLP.



Fonte: Autor, 2024.

Os sinais de entradas/saídas analógicas podem ser feitas através de diferentes grandezas, porém no projeto em questão utilizou-se sinais de 4 a 20mA para tal. O CLP não consegue trabalhar com sinais contínuos internamente, e por isso utiliza um número finito aproximado para representar o sinal de 4 a 20mA que está chegando em sua entrada. Esse número finito é representado por 16 bits, o quais podem representar números de 0 a 32767. Esses 16 bits, também conhecidos por word são representados dentro do CLP por registradores e são simbolizados pela letra *D*. Por padrão, o TPW04 disponibiliza registradores fixos que irão receber os sinais e entradas e saídas analógicas no hardware, começando com o endereço D8436 e indo até D8495. O módulo de expansão utilizada para a leitura/escrita analógica trabalha com uma faixa fixa de valores, de posse dos mesmos e também do valor máximo assumido pelo sensor a ser lido, é possível construir uma escala, fazendo com que para encontrar o valor final baste multiplicar o valor lido no registrador correspondente a entrada analógica (AI) em questão, pela escala previamente calculada.

Quanto às saídas analógicas, existe um tratamento diferenciado, dependendo

do valor que o equipamento de destino espera ler será necessário enquadrar esse mesmo valor dentro da escala de 4 a 20mA. A parte final da saída analógica (AO), é simples, bastando enviar o valor final do sinal para um registrador previamente estabelecido pelo CLP para que esse valor seja associado a saída analógica em questão. No caso da primeira AO, o registrador de destino associado será D8381, já a segunda está associada a D8382 e assim sucessivamente, até o número máximo de saídas analógicas disponibilizadas pelo CLP que terá como seu registrador D8390.

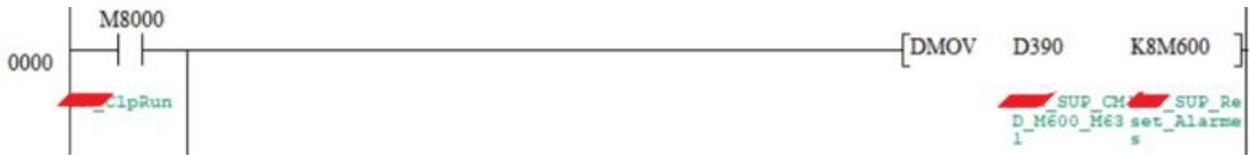
3.3.4.2 Comandos Oriundos da IHM/Supervisório

Existem algumas partes do processo que exigem a atuação do operador, são etapas que necessitam de uma confirmação, tais quais como a partida/parada da máquina e sincronização.

Esses sinais que tem como origem a ação do operador, são botões acionados via IHM/Supervisório, e são enviados ao CLP por meio da rede Modbus TCP/IP. Assim como os demais equipamentos, o CLP também possui uma tabela de endereços Modbus, através do qual é possível acessar os dados das memórias internas do equipamento. No caso do controlador, são acessíveis, por exemplo, as entradas/saídas digitais X/Y, as memórias auxiliares M e registradores D.

Existem algumas formas de enviar os comandos do mestre (IHM/SUP) para o escravo. Uma delas é utilizando a função Modbus 05, *Write Single Coil*, em que como o próprio nome sugere, escrevemos apenas um bit por vez. Outra forma pela qual podemos optar, seria por utilizar a função Modbus 16, que trabalha com a escrita de múltiplos registradores. A função 05 seria a forma mais direta, porém, em certos casos, ela pode gerar um congestionamento da rede, uma vez que a estrutura do telegrama Modbus para 1 bit ou 32 bits é a mesma, mudando apenas os parâmetros. Por esse motivo, em alguns casos, como o do supervisório, opta-se por agrupar os comandos em grupos de 32 bits, ou seja, 32 comandos diferentes, e enviá-los a dois registradores em sequência do CLP. Já no CLP, é realizada uma operação para desmembrar esses registradores e enviar cada bit para uma memória auxiliar separadamente. Para o caso em que os comandos são agrupados em 32 bits, a Figura 19 demonstra como esses dados são processados dentro da lógica do CLP.

Figura 19 – Exemplo de Código para o Desmembramento dos Comandos do Supervisório em Memórias Auxiliares.



Fonte: Autor, 2024.

Utiliza-se a função *DMOV* para enviar os dados que chegaram do supervisório em D390 e D391 para as memórias auxiliares. A função entende que são 32 bits que deverão ser transcritos por conta do uso do *K8*, que dentro do manual de programação do TPW04 é identificado como *elemento de bit*. Esse, quando seguido de endereços de memória auxiliares, indicam ao CLP que se tratam de grupos de bits, sendo que cada grupo contém 4 bits, ou seja, no caso de se utilizar *K8M600*, o *DMOV* entende que serão enviados 8 grupos de 4 bits (32 bits no total) de D390 e D391 para M600 em diante.

No caso da IHM, por conta do baixo fluxo de dados necessários na rede, optou-se por utilizar a função 05 e escrever os bits de comando diretamente no CLP. Novamente, o mesmo processo de transpor os sinais para outras memórias auxiliares, que foi realizado com as entradas/saídas digitais será implementado com os comandos oriundos da IHM ou Supervisório.

Figura 20 – Exemplo Espelhamento dos Comandos em Memórias Auxiliares.



Fonte: Autor, 2024.

A memória M42 armazena o estado de operação do processo, local ou remoto.

- Local: quando a operação é monitorada e controlada pela IHM que está junto da operação.
- Remoto: acionado quando a operação está sendo feita por meio de um SCADA (Supervisório), que pode estar rodando em um lugar diferente da planta.

3.3.5 Descritivo Funcional

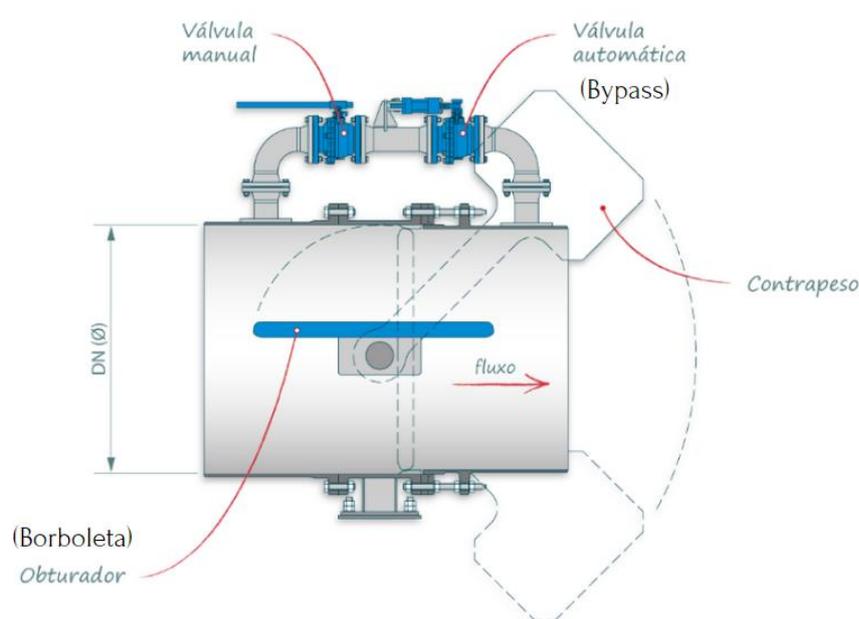
Todas as partes do projeto descritas até o momento fornecem a base para a implementação da lógica de funcionamento do processo. Para que o software possa ser desenvolvido de maneira correta, o desenvolvedor precisa ter conhecimento sobre o funcionamento do projeto. Quais são as etapas a ser seguidas para partir a máquina, as etapas para parar e os requisitos para que cada parte do processo possa acontecer. São todos detalhes que devem constar na documentação do projeto.

O descritivo funcional será o documento que irá fornecer todas essas informações ao desenvolvedor, desde o modo de operação de cada equipamento até o passo a passo que deve ser seguido para o correto funcionamento da usina.

Dentro do funcionamento da usina, podemos encontrar vários exemplos interessantes da importância de se seguir sistematicamente a lógica estabelecida em projeto, tais como:

- Equalização de pressão antes e depois da válvula borboleta: assim como visto na Seção 2.6, a válvula borboleta tem como função a interrupção do fluxo de água dos dutos para dentro da turbina. Isso pode ser necessário em algumas ocasiões, tais como na partida/parada da usina ou até mesmo em alguma situação de emergência. Quando falamos da partida da turbina, se faz necessário entender que a pressão com a qual a água chega no final dos dutos é bastante grande, e pelo fato da turbina estar inicialmente a vazio, não se pode deixar que essa pressão chegue a turbina abruptamente. Portanto, no início a válvula borboleta se encontra fechada, bloqueando o fluxo de água. O que faz com que a equalização da pressão seja feita gradualmente. A Figura 21 demonstra como os equipamentos necessários a equalização da pressão ficam dispostos na planta.

Figura 21 – Emprego de Válvula Borboleta e Válvula Bypass em uma CGH.



Fonte: Hydrowheel, 2024.

Uma vez que a água chega até a válvula borboleta (fechada) e todos os intertravamentos que dizem respeito à partida até esse ponto estão satisfeitos, a válvula bypass é aberta, fazendo com que haja um fluxo controlado de água em direção à turbina. A Figura 22 contém esse tipo de lógica.

Figura 22 – Exemplo de Lógica para Abertura da Válvula Bypass.



Fonte: Autor, 2024.

Gradualmente a diferença de pressão tenderá a zero. E nesse momento a válvula borboleta é aberta, fazendo com que o fluxo de água seja pleno para iniciar a geração de energia.

- Sincronização do gerador com a concessionária: Durante todo o processo de início da operação, até que o gerador esteja em carga nominal, a barra de geração permanece desconectada da concessionária. E no momento em que a planta atinge a sua faixa de operação, ela então deve ser conectada à rede. Porém, para que isso ocorra, é necessário realizar a sincronização entre barra

e concessionária. Alguns elementos como fase, frequência e tensão devem estar alinhados para que o disjuntor de geração possa ser armado.

Existem equipamentos específicos que realizam todo esse processo de sincronização da planta. Bastando enviar um sinal para que o dispositivo inicie o processo, e então receber o retorno de que o mesmo foi finalizado.

O comando de sincronização é enviado pelo CLP. E para que isso ocorra se faz necessário o conhecimento de todos os requisitos de operação para que o sinal ao sincronizador possa ser enviado.

Figura 23 – Exemplo de Lógica para que o Sincronismo Seja Habilitado.



Fonte: Autor, 2024.

A Figura 23 demonstra os intertravamentos específicos para que o comando de sincronismo seja enviado. Porém, esse é um dos últimos passos da partida, e para que cada passo seja efetivado, uma série de intertravamentos diferentes devem ser satisfeitos.

Os intertravamentos são situações em que um processo não pode acontecer sem que outro tenha acontecido previamente. E o conhecimento aprofundado do funcionamento do projeto se mostra crucial para o desenvolvimento da lógica implementada no CLP. Outras dessas etapas que pode-se citar são:

- Controle das bombas de lubrificação: esses equipamentos são cruciais para o correto funcionamento dos mancais.
- Controle de velocidade da turbina: realizado através do regulador de velocidade RHW100, de fabricação WEG. A análise das condições para a partida do regulador de velocidade é toda realizada dentro do software do CLP. Uma vez acionado, o RHW100 irá atuar na regulação das pás da turbina de modo a atingir o setpoint que lhe foi enviado.
- Acionamento do regulador de tensão: de forma similar ao regulador de velocidade, o regulador de tensão utilizado, ECW500, também de fabricação WEG, será acionado no momento em que a programação do CLP indicar que as condições para tal estão satisfeitas, tais como velocidade nominal da turbina dentro da faixa de operação. O ECW500 será o responsável por realizar o controle do gerador.

3.3.6 Documentação

Esse documento também chamado de *Tabela de comunicação* apresentará um descritivo completo de cada variável utilizada durante o desenvolvimento do programa do CLP. Nele está disposto o código da variável, um comentário explicando sua finalidade, tipo de dado, endereço de leitura correspondente a variável, etc.

Esse documento se mostra importante para o cliente, uma vez que pessoas irão ter contato com esse mesmo projeto, mas nem todas terão acesso à programação do CLP. Porém, em alguns casos, o técnico precisa saber, por exemplo, a origem de um determinado sinal, ou no caso de alguma outra empresa ser responsável pelo desenvolvimento do sistema SCADA, o técnico não terá acesso ao software do CLP, porém ele precisa saber em quais endereços irá encontrar as variáveis que ele necessita para o seu sistema.

3.4 INTERFACE HOMEM MÁQUINA

Durante a seção que segue serão apresentados as principais etapas e detalhes do desenvolvimento desta IHM que irá integrar o atual sistema do cliente que contava com outras duas unidades geradoras.

O software confeccionado pelo autor abrange tanto o processo de aquisição/envio de dados dos equipamentos que compõem o sistema, quando desenvolvimento das telas, criação de alarmes e também documentação.

3.4.1 Hardware

A escolha do equipamento que deverá ser utilizado no projeto depende de uma série de fatores, tais como: quantidade de informações necessárias em uma só tela (para a escolha das dimensões da IHM), capacidade de armazenamento de dados, software utilizado para o desenvolvimento, etc. Durante o dimensionamento dos equipamentos, o cliente, em conjunto aos técnicos da WEG, optou pela utilização da IHM WEG, de 10 polegadas modelo cMT2108X. A mesma conta com duas portas LAN, a qual será utilizada para trabalhar com Modbus TCP/IP, e também uma porta RS-485 para se comunicar por meio de Modbus RTU, além dessas, ainda dispõem outras porta de comunicação diferentes.

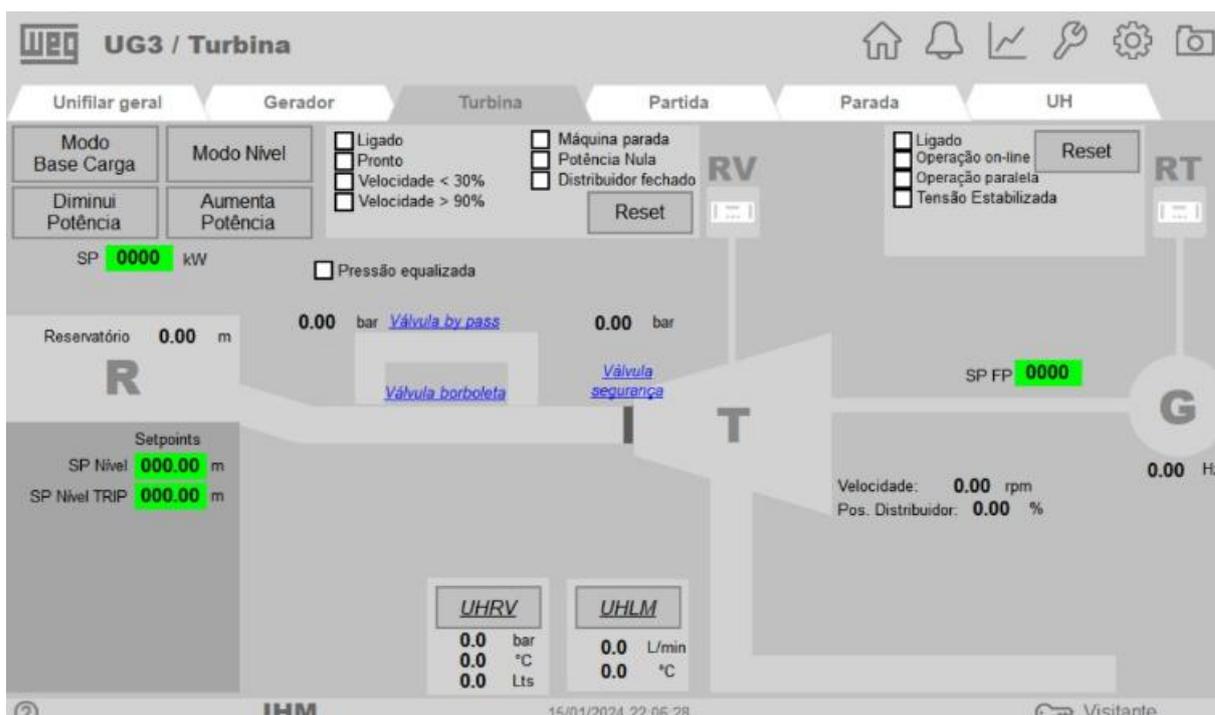
3.4.2 Dimensionamento de Telas

Algumas telas são dimensionadas por meio de fluxogramas, os quais servem de base para estabelecer a dinâmica que estará presente nas telas, já outras podem ser criadas com base nas dinâmicas conhecidas do sistema. Existem também as telas que por padrão devem constar na IHM, como por exemplo: alarmes, eventos e status de entradas/saídas do CLP.

3.4.3 Criação de Telas

Uma vez definidas as informações que necessitam estar em cada parte da IHM, é iniciada a sua criação. No princípio são criados os símbolos que em conjunto irão compor cada tela, uma vez que as conexões entre os equipamentos (elétrica e fluidos) são conhecidos, pode-se então conectá-los e encontrar a melhor posição para que fiquem dispostos.

Figura 24 – Exemplo de Organização de Símbolos em IHM.



Fonte: Autor, 2024.

A Figura 24 demonstra como isso é feito, note que algumas das informações que eram necessárias constar em tela foram botões, gerador, turbina, regulador de velocidade (RV) e regulador de tensão (RT). E como a dinâmica do processo já era de conhecimento do projetista, bastou conectá-los.

3.4.4 Símbolos

São figuras que podem, ou não, conter animação. Como exemplo de símbolos pode-se citar alguns bastante conhecidos, tais como as bombas, lâmpadas, atuadores, entre outros.

O primeiro passo a ser dado na criação de símbolos é a definição de padrão de cores. Desde 2015 a norma ISA-101 padroniza a forma como a IHM deve ser desenvolvida (ISA-101.1, 2015).

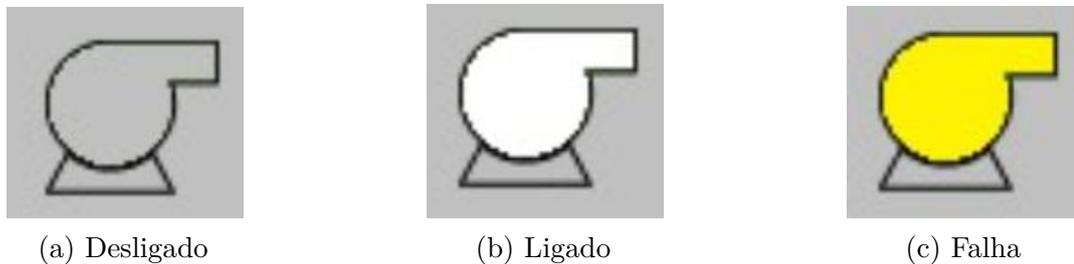
A WEG adotou um padrão de desenvolvimento de telas chamado de *High Performance*, o qual trabalha com a filosofia de minimização de detalhes em símbolos e utiliza

de um padrão de cores o mais próximo possível de cinza, dando foco aos detalhes que realmente são relevantes dentro da dinâmica do processo.

A Figura 24 é um exemplo do resultado que se obtém dessa filosofia de desenvolvimento. Note que os símbolos que se observam destacados em tela são os campos de setpoints (campos em verde) e os checkboxes (quadrados brancos) que de fato são as informações mais relevantes nesse contexto.

Os símbolos que contém animação para indicar diferentes estados do equipamento necessitam de utilizar cores diferentes, e mesmo essas cores são escolhidas pensando no desempenho do operador. Um bom exemplo seriam as bombas, nesse caso, existem basicamente três estados em que elas podem se encontrar:

Figura 25 – Símbolo de Bomba Utilizado em Projeto



Fonte: Autor, 2024.

Note que o único estado em que uma cor vibrante é utilizada será o de falha, indicando algo fora do padrão que está acontecendo no processo.

3.4.5 Animação

A animação dos símbolos pode ser realizada utilizando tags, ou então por meio do endereçamento direto da comunicação. Dependendo do valor atual da variável, uma determinada figura é exibida. Pode-se trabalhar com variáveis do tipo inteiro ou booleanas, de acordo com as necessidades do projeto. Inteiros são utilizados quando se faz necessário a associação de várias figuras à mesma tag, enquanto os booleanos são encontrados em situações de apenas dois estados possíveis, como ligado/desligado.

Pensando na situação do símbolo de bomba, por exemplo, uma tag do tipo booleana poderá ser utilizada para indicar o estado de ligado/desligado enquanto uma segunda tag, também booleana, indicará o estado de falha.

Figura 26 – Estados de Um Símbolo Animado



Fonte: Autor, 2024.

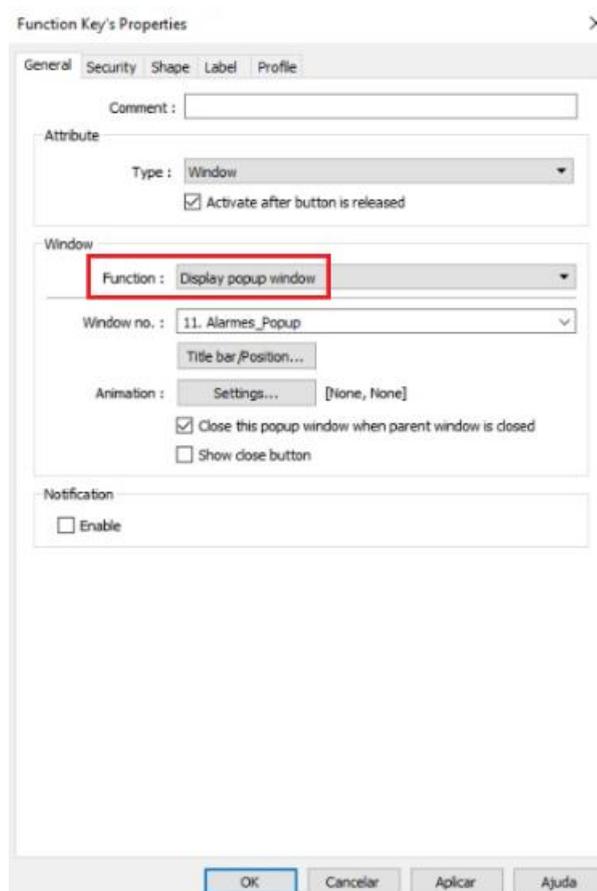
As figuras 26a e 26b demonstram como o status de acionamento é associado ao valor da tag no software de desenvolvimento da IHM.

3.4.6 Popups

Muito utilizadas por interfaces homem máquina e também em sistemas SCADA, as popups consistem em janelas flutuantes que são abertas na tela principal quando solicitadas. O seu conteúdo é composto por informações relevantes a alguma parte em específico da tela principal.

Dentro do ambiente de desenvolvimento da IHM, a criação de uma popup é feita da mesma forma que os demais displays. O que indicará ao software que a tela a ser aberta é na verdade uma popup, assim como a Figura 27 mostra, será a configuração feita pelo desenvolvedor indicando que após o toque do usuário em determinado objeto, um novo display deverá ser aberto, e na verdade o mesmo se trata de uma popup.

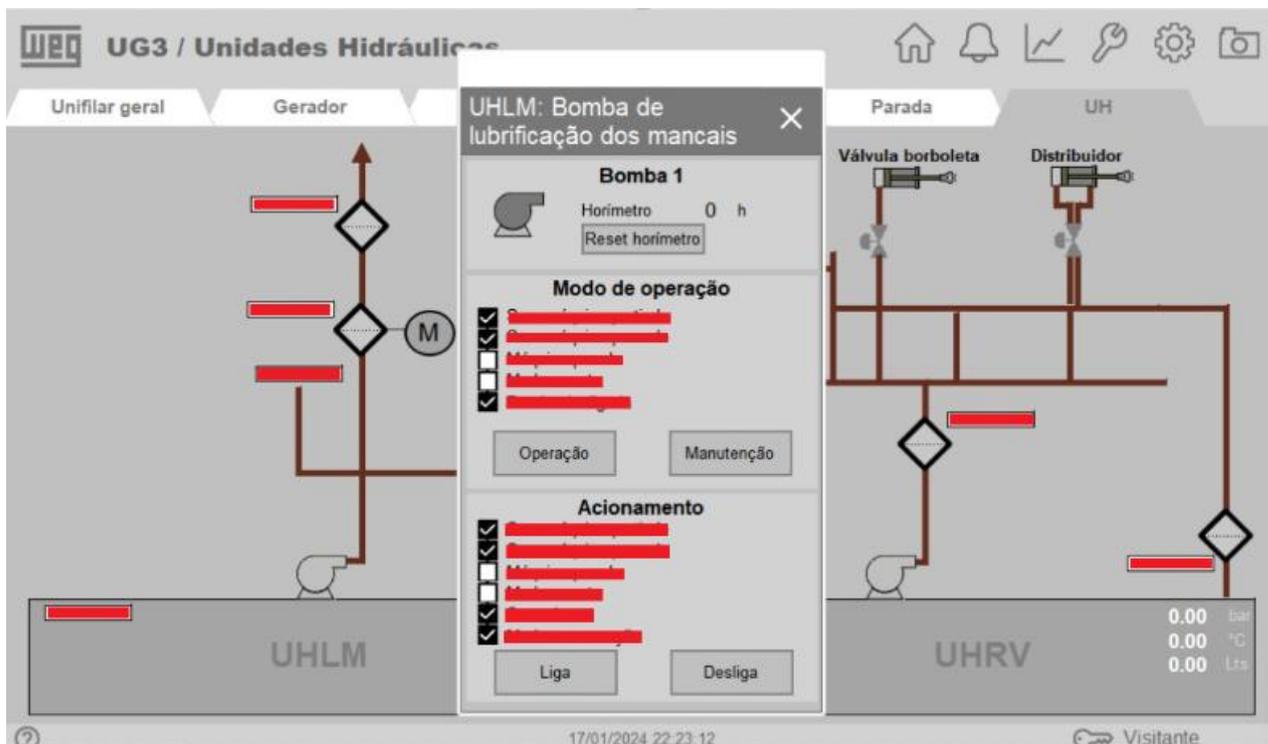
Figura 27 – Configuração de Objeto para Abertura de uma Popup.



Fonte: Autor, 2024.

Para a situação em que algum equipamento necessite de alguma outra forma de interação com o operador que vá além da visualização em forma de cores, pode-se fazer uso de popups. Um exemplo interessante a ser analisado é o caso de uma bomba.

Figura 28 – Popup para Detalhamento de Bomba UHLM.



Fonte: Autor, 2024.

Como observado na Figura 28, ao clicar sobre o símbolo de bomba na tela da UH (Unidade Hidráulica) a popup é aberta em meio ao display mostrando ao operador as informações pertinentes. Tais como condições para o seu acionamento, status atual, além de botões para acionamento ou seleção do modo de operação.

3.4.7 Comunicação

Existe a possibilidade de realizar o processamento de dados dentro da própria IHM, quando o seu hardware permite. Para a CGH em questão, a interface homem máquina irá, em sua maioria, apenas sintetizar as informações da planta e também fazer a interação entre CLP e operador.

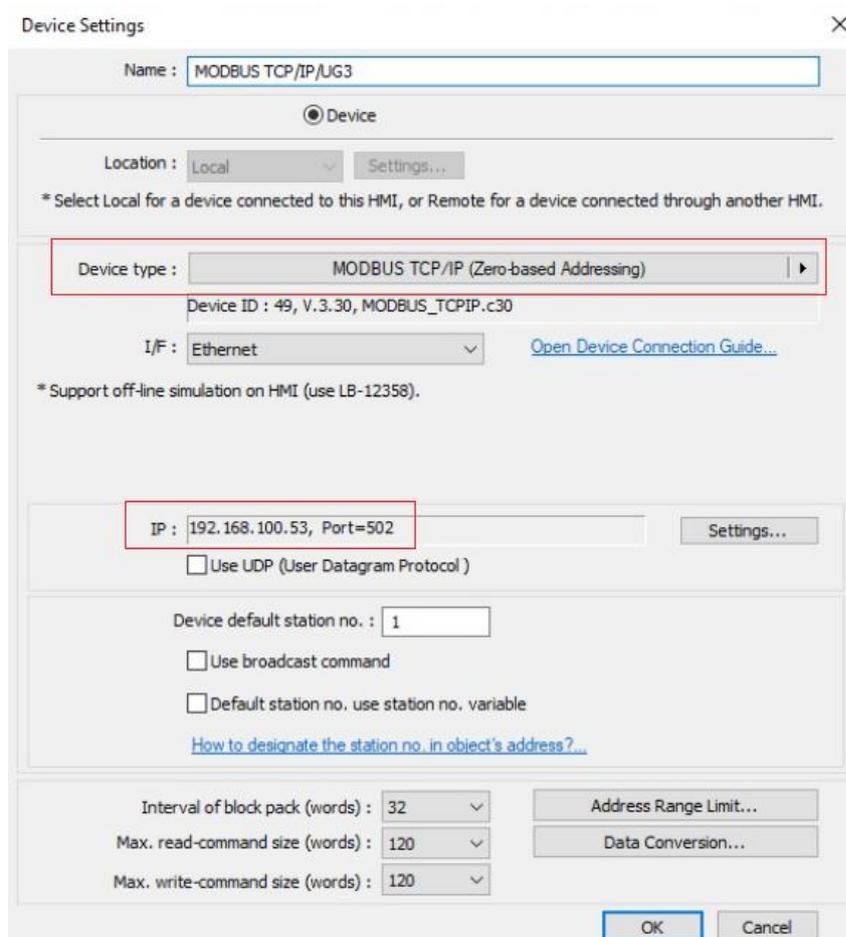
Portanto, as tags utilizadas no software precisam ter seus dados coletados/enviados aos demais equipamentos da planta, e nessa situação, se faz necessário o estabelecimento de uma rede de comunicação.

Assim como havia sido mencionado na Seção 3.3.2, a maior rede de comunicação do projeto é implementada com Modbus TCP/IP, abrangendo a maioria dos equipamentos. Nessa rede, a IHM irá atuar como mestre enquanto o CLP, relés de proteção e regulador de velocidade, irão ser apenas escravos.

3.4.7.1 Parametrização dos Equipamentos

Para que a rede funcione, é necessário a parametrização dos equipamentos que a compõem. No caso da IHM, todos os escravos com quem ela irá se comunicar devem ser inseridos em seu software, informando o seu respectivo endereço IP, porta de comunicação e demais parâmetros. A Figura 29 mostra como isto é realizado:

Figura 29 – Parametrização de Escravos no Ambiente de Desenvolvimento da IHM.



Fonte: Autor, 2024.

O primeiro passo é indicar ao software qual será o driver de comunicação que ele deverá utilizar, o que é feito por meio do *Device type*, em sequência os parâmetros do protocolo em questão são configurados, tais como endereço IP e o número da porta por onde a comunicação será feita.

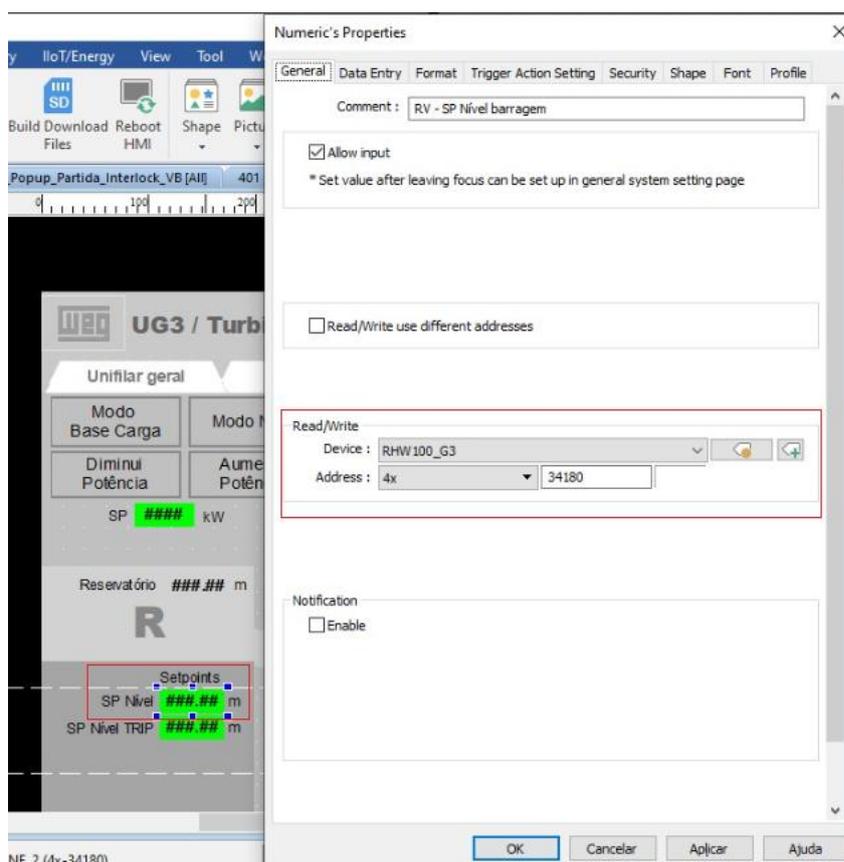
3.4.7.2 Endereçamento

Durante o desenvolvimento das telas, em inúmeras partes são necessárias o uso de informações provenientes de outros dispositivos, e uma vez que esses dispositivos já foram devidamente parametrizados como visto na Seção 3.4.7.1, basta então na hora de

configurar os campos de leitura/escrita em tela, informar ao software o endereço dos dados a serem lidos/enviados de outros equipamentos:

Para tal, se faz necessário apenas o conhecimento prévio da tabela de comunicação do equipamento de origem/destino. Uma vez que no desenvolvimento do software, o que estará atrelado ao campo de escrita do setpoint de nível da barragem, por exemplo, será diretamente o endereço no qual a IHM deve escrever no escravo esse dado.

Figura 30 – Comunicação com Escravo Feita Diretamente por Endereçamento.



Fonte: Autor, 2024.

Note que na Figura 30, a escrita do setpoint é atrelado diretamente ao endereço *34180* no regulador de velocidade RHW100.

Observe também que é passado como parâmetro ao software *4x*, que indica que se trata de um holding register e portanto permite a escrita no escravo.

No caso dos demais tipos de dados, como leitura/escrita de bits, leitura de words, etc, o software é desenvolvido da mesma maneira, mudando apenas o parâmetro que se refere ao formato do dado.

3.4.8 Alarmes

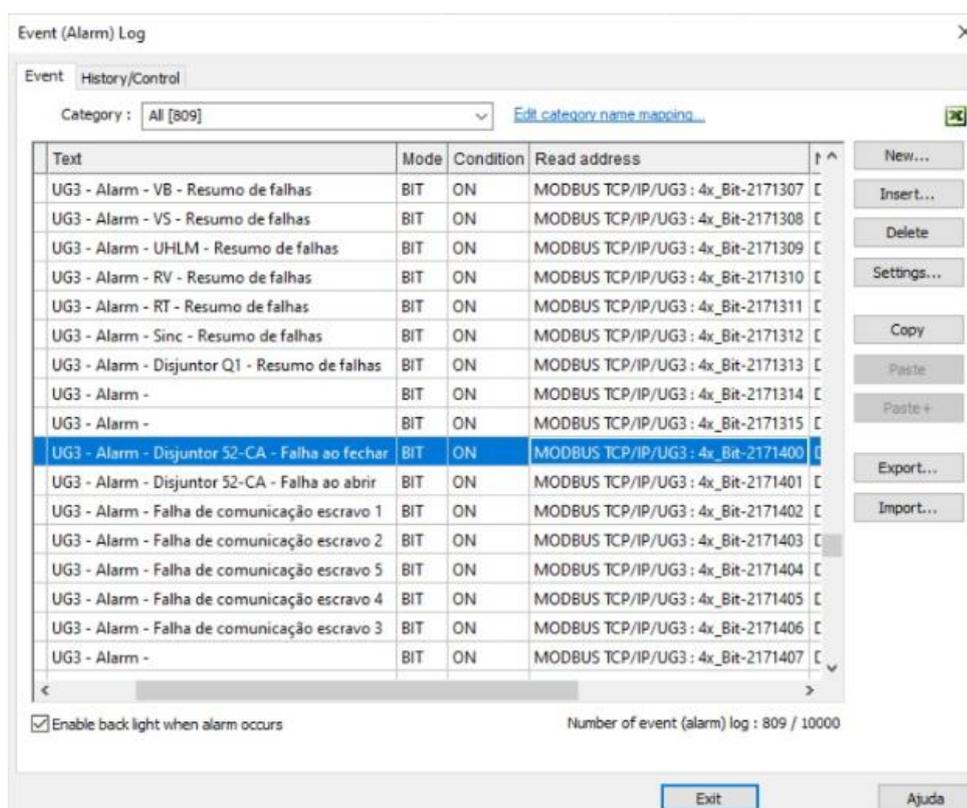
Os alarmes desempenham papel fundamental na operação de uma planta. Através deles o operador fica ciente dos problemas que estão ocorrendo em tempo real, além de que os textos associados aos alarmes fornecem uma boa noção da localização do problema, facilitando assim a tomada de decisão do operador.

Os equipamentos podem gerar alarmes que serão apresentados na IHM, porém todos são primeiramente enviados ao CLP para que após processados fiquem disponíveis para visualização na IHM.

Ainda no CLP, os alarmes gerados pela sua lógica são deslocados para algumas words. Já na IHM existe um espaço em seu ambiente de desenvolvimento dedicado a configuração dos alarmes. Nele são inseridas os endereços das words de alarmes oriundas do CLP e então bit a bit desta word é associado a algum texto, que ficará visível ao operador no momento em que aquele determinado bit da word estiver ativo.

A Figura 31 mostra como esses alarmes são criados.

Figura 31 – Criação de Alarmes na IHM.



Fonte: Autor, 2024.

Tendo como exemplo o alarme destacado na Figura 31, no lado esquerdo da figura, no campo *text*, está inserido o texto que o operador visualizará. Já ao lado direito da parte destacada, no campo *Read address*, tem-se os parâmetro abaixo.

- Modbus TCP/IP/UG3: nome dado ao equipamento no momento de inseri-lo na rede de comunicação da IHM.
- 4x_Bit: 4x indicará a software que se trata de um holding register e *Bit* sinaliza que está se trabalhando com um único bit de uma word.
- 2171400: 21714 é o endereço no qual está localizado a word de alarmes no escravo. Enquanto 00 se refere ao bit da word a ser considerado.

Note que o alarme abaixo do destacado termina em 01, indicando o próximo bit da word. Enquanto que o alarme acima do destacado tem como endereço 2171315, ou seja, último bit da word anterior.

4 RESULTADOS

Neste capítulo, será apresentado os resultados obtidos durante o desenvolvimento da programação do controlador lógico programável e da interface homem-máquina. Por procedimento interno da empresa, após a finalização do software do projeto, um outro projetista é encarregado de conferir tudo o que foi desenvolvido. Para verificar se o projeto está de acordo com as normas de confecção de software da WEG. E nesta verificação, foi constatado a correta confecção do projeto.

Na presente data de apresentação deste trabalho, todo o projeto desenvolvido se encontra na fase de comissionamento, e portanto ainda não há resultados definitivos que podem ser apresentados.

4.1 CLP

Iniciando com o CLP, durante os testes do painel fabricado pela WEG em que o TPW04 estava inserido, foi possível validar toda a comunicação entre o CLP e os demais equipamentos, tanto da rede Modbus RTU, em que ele atua como mestre, quanto na rede Modbus TCP/IP, em que o controlador é apenas um escravo. Além da comunicação, validou-se também os seus contatos de DIOS sendo que no caso das DIs, foram acionadas as suas entradas físicas e visualizados os seus contatos diretamente no software do controlador. Enquanto que para as saídas digitais, o processo contrário ocorreu, pois elas foram acionadas via software e então confirmadas fisicamente. Do mesmo modo, as AIOs foram testadas. O demais testes são realizados durante o comissionamento do projeto, pois é necessário que todos os equipamentos estejam devidamente instalados e operando para que a lógica toda possa ser validada.

4.2 IHM

A interface homem máquina foi testada através da simulação disponibilizada pelo ambiente de desenvolvimento da IHM e pôde-se testar todas as suas telas, da mesma forma que a Figura 24 foi gerada. A seguir serão demonstrados alguns resultados obtidos da simulação do software desenvolvido pelo autor.

Figura 32 – Diagrama Unifilar UG3.

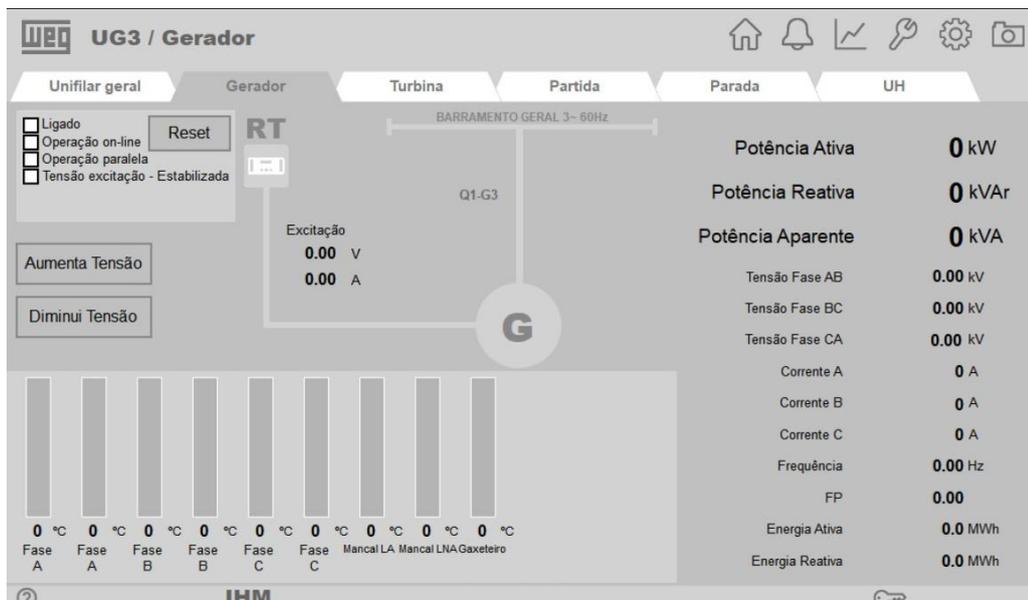


Fonte: Autor, 2024.

A Figura 32 apresenta o diagrama unifilar do terceiro gerador da usina e nela estão dispostos os relés de proteção, o multimetido MMW03, que ao clique do operador abrirá um *popup* que por sua vez mostrará valores referentes a diversas grandezas. Além de valores nominais do gerador e comandos de reinicialização destes dispositivos de proteção, a tela conta ainda com um disjuntor de proteção *Q1-G3*. Essa tela foi desenvolvida com base no diagrama unifilar presente no projeto elétrico.

Em sequência, temos a Figura 33, que apresenta todos os dados pertinentes ao gerador. Seu estado de operação, dados de grandezas instantâneas, além de comandos pertinentes ao operador. Os dados dispostos em tela tem como origem o relé de proteção que monitora todos esses valores e de acordo com a sua programação, pode atuar na proteção do gerador. Os comandos de aumenta/diminui tensão estão relacionados a tensão de excitação do gerador, que por sua vez tem influência direta na energia gerada. Esses comandos são enviados da IHM para o regulador de tensão, que por sua vez irá de fato realizar a operação através do seu software/hardware.

Figura 33 – Gerador UG3.



Fonte: Autor, 2024.

Figura 34 – Entradas e Saídas Analógicas UG3.



Fonte: Autor, 2024.

Uma última tela apresentada como resultado, a Figura 34, é um display importante no comissionamento e também no dia-a-dia de operação da planta. Durante o comissionamento do projeto, todos os componentes da planta deverão ser validados, e uma maneira de facilitar a atividade dos técnicos de campo é desenvolver uma tela em que

todos os dados referentes as leituras/escritas analógicas do CLP podem ser visualizadas concomitantemente.

5 CONCLUSÃO

Este trabalho, realizado em parceria com a WEG, teve como objetivo desenvolver a programação a ser implementada no CLP que controlará a operação de uma unidade geradora a ser instalada no cliente, além de demonstrar como essa programação tem origem em toda a documentação do projeto, como por exemplo o projeto elétrico. Alcançou-se o objetivo de desenvolver o software para a IHM com a inclusão de uma nova unidade geradora no cliente. A concepção deste trabalho só foi possível graças à autorização da empresa WEG e da colaboração de colegas da seção que prestaram todo o suporte necessário.

Ademais, desenvolvida a programação do CLP com base nisto. Desde a transferência dos sinais de entradas para memórias auxiliares, passando pela configuração da comunicação com os demais dispositivos, além da lógica de funcionamento da usina até a geração de alarmes.

Enquanto o software da IHM teve o início do seu desenvolvimento com a adição e parametrização dos equipamentos com os quais ela iria se comunicar, desde a seleção do protocolo de comunicação até a sua parametrização. Os símbolos expostos nas telas foram desenvolvidos com base na norma ISA-101 e tiveram o recebimento/envio de dados configurados individualmente. Os alarmes também foram configurados com base no endereçamento de dados realizado no CLP.

Os testes desenvolvidos tanto por meio de painel elétrico no caso do CLP, quanto por simulação no caso da IHM, demonstraram que o objetivo inicial de confecção de software para ambos os dispositivos foi validado. Os resultados definitivos do projeto serão obtidos no momento em que o comissionamento da unidade geradora for finalizado e sua entrada em operação, juntamente com todas as alterações de software que venham a ser necessárias.

REFERÊNCIAS

NÚMERO PCHs e CGHs em operação no Brasil? ABRAPCH. Disponível em: <https://abrapch.org.br/faq/numero-pchs-e-cghs-em-operacao-no-brasil/>. Acesso em: 28 jan. 2024.

DAVID HALLIDAY AND ROBERT RESNICK AND JEARL WALKER. **Fundamentos da Física: Eletromagnetismo**. Rio de Janeiro - RJ: LTC Editora, 2016. P. 812.

O que é uma CGH (Central Geradora Hidrelétrica)? HIDROENERGIA. Disponível em: <https://www.hidroenergia.com.br/blog/o-que-e-uma-cgh/>. Acesso em: 23 jan. 2024.

VÁLVULAS Borboleta. HydroWheel. Disponível em: <https://hydrowheel.com.br/produtos/valvulas/valvulas-borboleta/>. Acesso em: 29 jan. 2024.

ISA. **Introdução à Norma ISA-101: Interfaces Homem-Máquina**. [S.l.], 2015. Disponível em: <https://isasp.org.br/wp-content/uploads/2020/01/ISA-101-III-Simp%C3%B3sio-ISA-S%C3%A3o-Paulo-Sabesp-Nov2016.pdf>. Acesso em: 28 jan. 2024.

LEANDRO ROGGIA AND RODRIGO CARDOZO FUENTES. **Automação Industrial**. Santa Maria - RS: e-Tec Brasil, 2016. P. 102.

OUTORGAS. Ministério de Minas e Energia. Disponível em: <https://www.gov.br/aneel/pt-br/assuntos/geracao/outorgas>. Acesso em: 23 jan. 2024.

P., Moura Ailson; MOURA, Adriano Aron F.; ROCHA, Ednardo P. **Geração Hidroelétrica e Eólicoelétrica**. 1. ed. Ceará: Edições UFC, 2019. P. 291.

O que é e para que serve uma Subestação Elétrica? P3 Engenharia Elétrica. 2018.

Disponível em:

<https://p3engenharia.com.br/blog/o-que-e-e-para-que-serve-uma-subestacao-eletrica/>. Acesso em: 28 jan. 2024.

PLCOPEN. **IEC 61131-3: a norma para programação**. [S.l.], 2003. Disponível em: https://plcopen.org/sites/default/files/downloads/intro_iec_march04_portuguese.pdf. Acesso em: 5 fev. 2024.