



UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE TECNOLOGIA E GEOCIÊNCIAS
DEPARTAMENTO DE ENGENHARIA ELÉTRICA
CURSO DE GRADUAÇÃO EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

LUCAS VINÍCIUS DOS SANTOS FALCÃO

**PROJETO DE AUTOMAÇÃO DE UM MÓDULO DE ULTRAFILTRAÇÃO DE ÁGUA
UTILIZANDO CLP'S E IHM DA ALTUS**

Recife
2024

LUCAS VINÍCIUS DOS SANTOS FALCÃO

**PROJETO DE AUTOMAÇÃO DE UM MÓDULO DE ULTRAFILTRAÇÃO DE ÁGUA
UTILIZANDO CLP'S E IHM DA ALTUS**

Trabalho de Conclusão de Curso apresentado ao departamento de Engenharia Elétrica da Universidade Federal de Pernambuco, como requisito parcial para obtenção do grau de Engenheiro de Controle e Automação.

Orientador: Prof. Dr. Herbert Albérico de Sá Leitão

Recife
2024

Ficha de identificação da obra elaborada pelo autor,
através do programa de geração automática do SIB/UFPE

Falcão, Lucas Vinícius dos Santos.

Projeto de automação de um módulo de ultrafiltração de água utilizando CLP's
e IHM da Altus / Lucas Vinícius dos Santos Falcão. - Recife, 2024.

62 p. : il., tab.

Orientador(a): Herbert Albérico de Sá Leitão

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de
Pernambuco, Centro de Tecnologia e Geociências, Engenharia de Controle e
Automação - Bacharelado, 2024.

Inclui referências.

1. Ultrafiltração. 2. Membranas. 3. CLP. 4. IHM. 5. Instrumentação Industrial.
I. Leitão, Herbert Albérico de Sá. (Orientação). II. Título.

620 CDD (22.ed.)

LUCAS VINÍCIUS DOS SANTOS FALCÃO

**PROJETO DE AUTOMAÇÃO DE UM MÓDULO DE ULTRAFILTRAÇÃO DE ÁGUA
UTILIZANDO CLP'S E IHM DA ALTUS**

Trabalho de Conclusão de Curso apresentado ao departamento de Engenharia Elétrica da Universidade Federal de Pernambuco, como requisito parcial para obtenção do grau de Engenheiro de Controle e Automação.

Aprovado em: 20/03/2024.

BANCA EXAMINADORA

Prof. Dr. Herbert Albérico de Sá Leitão (Orientador)
Universidade Federal de Pernambuco

Prof. Dr. Geraldo Leite Maia (Examinador Interno)
Universidade Federal de Pernambuco

M.Sc Eduardo Augusto Oliveira Barbosa (Examinador Interno)
Universidade Federal de Pernambuco

AGRADECIMENTOS

Primeiramente, gostaria de agradecer a Deus por ter me dado essa grande oportunidade em minha vida. Sem dúvidas alguma, nada seria possível sem Ele.

Agradecer aos meus pais, que me ajudaram e me incentivaram para ser quem eu sou hoje e por toda a educação que me foi passada. Só vocês conseguem entender o quanto essa jornada foi difícil.

Agradecer aos meus primos pela indicação à H2O Engenharia Química e Ambiental para a participação nesse projeto de ultrafiltração, assim como por terem trabalhado e visto o desenvolvimento do mesmo junto comigo.

Agradecer as amigadas que foram criadas e que me ajudaram significativamente para a chegada até aqui. Em especial, aos meus grandes amigos que dividiram diversas disciplinas comigo: Levi Lima, Vinícius Campelo, Natacha Correia e Felipe José. Com toda certeza, vocês deixaram todo esse processo mais fácil.

Por fim, gostaria de agradecer aos professores do Departamento de Engenharia Elétrica, que puderam contribuir, direta ou indiretamente, com o conhecimento que hoje tenho adquirido. Em especial, Herbert, por ter me dado a oportunidade de tê-lo como orientador do Trabalho de Conclusão de Curso, me dando toda a contribuição possível.

RESUMO

Qualidade e conservação dos recursos hídricos vêm sendo atividades com grande foco nas grandes indústrias nos dias atuais. Isso porque, além de deixá-las conforme visa as leis ambientais, ajudam as mesmas a atingirem uma ótima eficiência operacional. Para isso, surge como solução a implementação de membranas responsáveis por fazer a ultrafiltração da água, processo que é capaz de tratar a água mesmo que ela esteja em condições insalubres. Tal processo serve tanto para a produção de água potável quanto para pré-tratamento de determinados procedimentos, como é o caso da osmose reversa e da dessalinização. Este trabalho apresenta um projeto aplicado em uma indústria que consistiu na utilização de um módulo com 6 membranas de ultrafiltração, no qual há dois Controladores Lógicos Programáveis (CLP's) e uma Interface Homem Máquina (IHM) da Altus para realizar o controle e o monitoramento do processo, respectivamente. Além disso, fizeram parte deste projeto de automação uma ampla instrumentação industrial, composta por atuadores pneumáticos, válvulas solenoides, hidrômetros de pulso, inversor de frequência, bombas dosadoras, entre outros instrumentos, que serão tratados e detalhados neste trabalho.

Palavras-chave: Ultrafiltração; Membranas; CLP; IHM; Instrumentação industrial.

ABSTRACT

Quality and conservation of water resources have been one of the main focuses of large industries nowadays. That's because, in addition to ensuring compliance with environmental laws, they help them to achieve optimal operational efficiency. For this, the implementation of membranes responsible for ultrafiltration of water appears as a solution, a process which is capable of treating water even if it is in unhealthy conditions. This process serves both or the production of drinking water and for pre-treatment of specific procedures, such as reverse osmosis and desalination. The project carried out consists of using a module containing 6 ultrafiltration membranes, which is composed of two Programmable Logic Controllers (PLC's) and a Human Machine Interface (HMI) from Altus that performs the control and monitoring of the process. Other than that, a broad industrial instrumentation is implicit in this automation, such as: pneumatic actuators, solenoid valves, pulse hydrometers, frequency inverter, metering pumps, among others instruments which will be discussed and detailed in this work.

Keywords: Ultrafiltration; Membranes; PLC; HMI; Industrial instrumentation.

LISTA DE ILUSTRAÇÕES

Figura 1: Tela inicial após criação de projeto no MasterTool IEC XE.....	17
Figura 2: Zoom da tela inicial após criação de projeto no MasterTool IEC XE	17
Figura 3: Tela inicial após criação de projeto no <i>FvDesigner</i>	18
Figura 4: Membrana de ultrafiltração de água.....	20
Figura 5: Topologia mestre-escravo.....	24
Figura 6: <i>Broadcast</i> versus <i>multicast</i>	26
Figura 7: Bits dominantes e recessivos no CAN	27
Figura 8: Instruções básicas em <i>ladder</i>	28
Figura 9: Exemplos de blocos de função em <i>ladder</i>	28
Figura 10: Contato selo em <i>ladder</i>	29
Figura 11: Exemplo de um processo industrial.....	31
Figura 12: Máquina de estados	31
Figura 13: Passos do desenvolvimento do projeto	33
Figura 14: Fração da máquina de estados do modo <i>Air Scour</i>	35
Figura 15: Estado inicial do modo ' <i>Air Scour</i> '	35
Figura 16: Acionamento da válvula de drenagem através da máquina de estados...	36
Figura 17: Sub-rotinas criadas no <i>software</i> do CLP	36
Figura 18: Comentários de variáveis	37
Figura 19: Comentário de <i>rungs</i>	38
Figura 20: Leitura da entrada analógica do medidor de temperatura	39
Figura 21: Lógica em Ladder utilizada para o hidrômetro de pulso	41
Figura 22: Configuração do servidor Modbus.....	42
Figura 23: Janela para adicionar o protocolo de comunicação no CLP.....	42
Figura 24: Endereçamento de dados através do Modbus TCP	43
Figura 25: Configuração da IHM para realizar comunicação.....	44
Figura 26: Endereçamento de uma variável na IHM	45
Figura 27: Implementação de dispositivos na rede CAN.....	46
Figura 28: Mapeamento de entradas e saídas do CLP XP315	47
Figura 29: Fluxograma de navegação das telas da IHM	47
Figura 30: Tela de Login - IHM.....	50
Figura 31: Tela de Fluxograma - IHM.....	50

Figura 32: Tela de Operação - IHM.....	51
Figura 33: Tela de Instrumentação - IHM.....	51
Figura 34: Tela de Registradores - IHM	52
Figura 35: Tela de Alarmes - IHM	52
Figura 36: Tela de Manutenção - IHM.....	53
Figura 37: Tela do <i>Web Server</i>	54
Figura 38: Registro de funcionamento da tela de operação	55
Figura 39: Registro de funcionamento da tela de fluxograma	56
Figura 40: Registro de funcionamento da tela de alarmes	57
Figura 41: Registro de funcionamento do <i>Web Server</i>	58
Figura 42: Registro da membrana de ultrafiltração	58
Figura 43: Registro dos tanques de químicos	59

LISTA DE TABELAS

Tabela 1 – Tipos de dados Modbus	24
--	----

LISTA DE ABREVIATURAS E SIGLAS

CAN	<i>Controller Area Network</i>
CLP	Controlador Lógico Programável
FSM	<i>Finite State Machine</i>
IHM	Interface Homem Máquina
I/O	<i>Input/Output</i>
ORP	<i>Oxydation Reduction Potencial</i>
OSI	<i>Open Systems Interconnection</i>
PDU	<i>Protocol Data Unit</i>
PH	Potencial Hidrogeniônico
RTD	<i>Resistance Temperature Detector</i>
RTU	<i>Remote Terminal Unit</i>
TCP	<i>Transmission Control Protocol</i>

SUMÁRIO

1	INTRODUÇÃO	14
1.1	OBJETIVOS	15
1.1.1	Geral.....	15
1.1.2	Específicos	16
1.2	<i>SOFTWARES</i>	16
1.3	INSTRUMENTOS E EQUIPAMENTOS UTILIZADOS.....	18
1.4	ORGANIZAÇÃO DO TRABALHO.....	22
2	FUNDAMENTAÇÃO TEÓRICA	23
2.1	PROTOCOLO MODBUS	23
2.2	PROTOCOLO CAN	25
2.3	LINGUAGEM <i>LADDER</i>	27
2.3.1	Máquina de Estados Finitos.....	30
3	DESENVOLVIMENTO DO TRABALHO.....	33
3.1	ELABORAÇÃO DA MÁQUINA DE ESTADOS FINITOS	34
3.2	BOAS PRÁTICAS DE PROGRAMAÇÃO ADEQUADAS	36
3.3	PARAMETRIZAÇÃO DOS MEDIDORES	38
3.4	HIDRÔMETROS DE PULSO.....	40
3.5	UTILIZAÇÃO DO PROTOCOLO MODBUS TCP.....	41
3.6	UTILIZAÇÃO DO PROTOCOLO CANOPEN.....	45
3.7	CRIAÇÃO DAS TELAS DA IHM	47
3.8	WEB SERVER.....	53
3.9	RESULTADOS	55
4	CONCLUSÕES E PROPOSTAS DE CONTINUIDADE	60
	REFERÊNCIAS.....	61

1 INTRODUÇÃO

A água é um recurso essencial para o desenvolvimento da humanidade. Além de ser o meio básico para a sobrevivência, ela está presente em diversas atividades do dia a dia. A preocupação que se deve tomar para garantir esse recurso com qualidade e quantidade para as gerações futuras é gigantesca. No entanto, resultados mostram o contrário: a qualidade e a quantidade da água atualmente estão cada vez menores, em ambos os aspectos. Sendo assim, cabe a população a tomar medidas eficazes para a preservação desse bem.

De acordo com o relatório da ONU (BISCHOFF, 2023), o mundo pode entrar, no futuro, em uma crise global de escassez de água, sendo a América do Sul o continente que há uma grande probabilidade em termos de agravação dessa problemática. O aumento do consumo de água populacional e da poluição são uma das causas dessa preocupação.

Em relação ao Brasil, constata-se que o país possui um histórico com dados negativos em relação à distribuição de água potável. No ano de 2022, registrou-se uma perda de aproximadamente 40,1% de toda água produzida (HOJE, 2022). Isso significa que, a cada 100 litros de água, quase 40 litros são perdidos. Além disso, as mudanças climáticas, decorrentes do aquecimento global, têm afetado a capacidade dos reservatórios de água doce do país (MODELLI, 2023), o que pode levar a adoção de medidas drásticas, caso não sejam tomadas providências no presente.

Diante disso, é necessário que haja medidas a fim de minimizar as chances de escassez de água no futuro. Diminuir a poluição, principalmente em rios e mares, seria algo que de imediato traria bons resultados. Ou, em outros casos, realizar o tratamento da água para que a mesma volte a se tornar potável ou com qualidade semelhante.

Estando ciente disso, o presente trabalho tem como objetivo incentivar as empresas e fábricas a realizarem o tratamento de água através do módulo de ultrafiltração. A ultrafiltração consiste na separação das substâncias sólidas presentes na água através de membranas semipermeáveis. Esse processo consegue deixar a água em uma ótima qualidade, independentemente de sua apresentação em forma

bruta, deixando-a dentro das normas regulamentadas pelo Ministério da Saúde (FUSATI, 2020).

Através da ultrafiltração, é possível tratar até mesmo a água do mar, tornando-a adequada para diversos fins. Esse processo não apenas serve como tratamento primário, mas também pode ser utilizado como pré-tratamento para osmose reversa (FERRARO, 2008) ou nanofiltração (AGBOOLA, MAREE e MBAYA, 2014). É altamente recomendado que as empresas invistam nessa tecnologia, considerando seu alto consumo de água e a necessidade de contar com um recurso de qualidade para garantir eficiência nos processos.

Neste trabalho, é apresentado um projeto de implantação de um módulo de ultrafiltração HYDRAcap MAX 60 (Capillary Ultrafiltration Module HYDRAcap® MAX 60 Performance), composto por 6 membranas semipermeáveis e que opera em cinco diferentes modos de operação. Será vista a necessidade da automação para um projeto desse nível, uma vez que deixa o mesmo com capacidade de funcionamento automático, possuindo um monitoramento eficaz e detalhado, além de um controle mais preciso.

Vale salientar que este projeto foi desenvolvido em nome da H2O Engenharia Química e Ambiental e implantado em uma unidade da Unilever, empresa reconhecida por sua atuação na área de produtos de limpeza, higiene pessoal e alimentação. Os detalhes das instrumentações e equipamentos utilizados serão apresentados na Seção 1.3, assim como, posteriormente, todo o seu processo de desenvolvimento até a chegada do resultado final. O objetivo principal é a realização do tratamento de água para que ela possa ser aplicada dentro dos processos da empresa.

1.1 Objetivos

1.1.1 Geral

Desenvolver a programação nos CLP's (Controladores Lógicos Programáveis) e na IHM (Interface Homem Máquina) da Altus para controlar e monitorar o módulo de ultrafiltração de água.

1.1.2 Específicos

Segue abaixo os objetivos específicos deste trabalho:

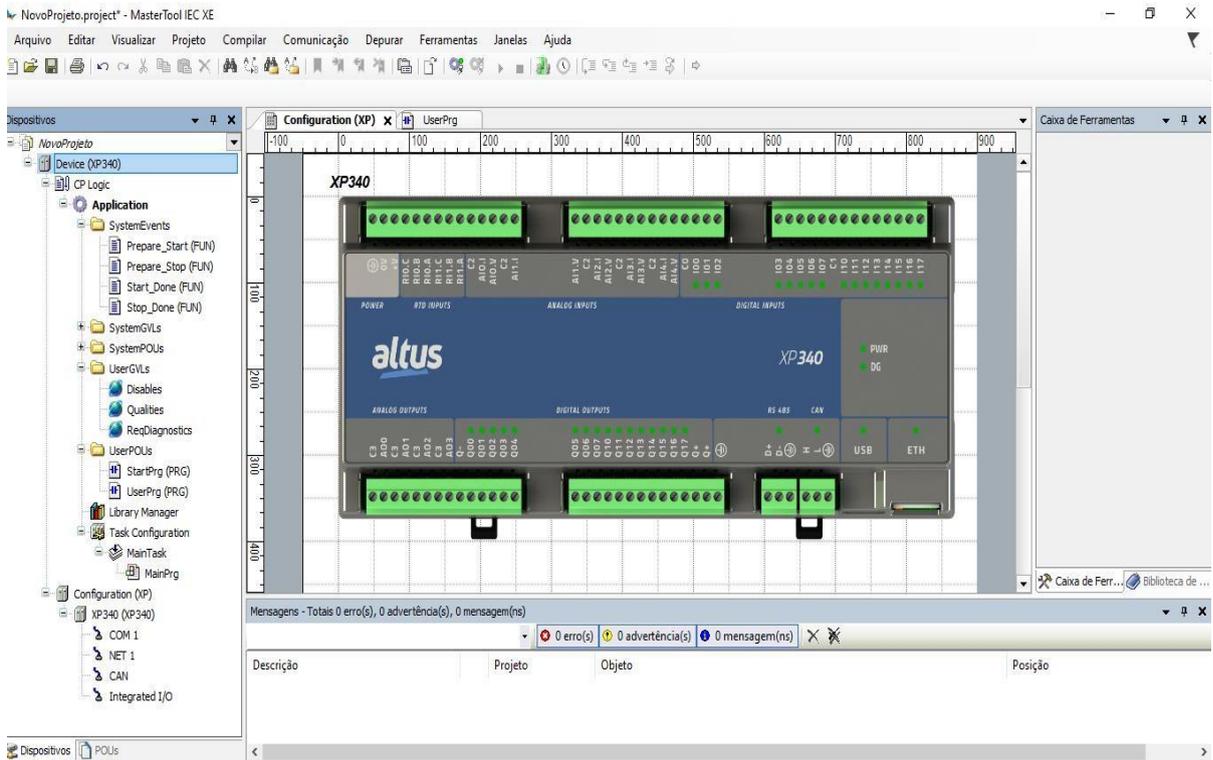
- Realizar a programação, em linguagem *Ladder*, do CLP mestre que controla o módulo de ultrafiltração de água;
- Realizar a programação na IHM para monitorar e também controlar a planta;
- Criar um *web server* através do *software* de programação da Altus, a fim de exibir os dados em tempo real do módulo em dispositivos em rede;
- Estabelecer a comunicação correspondente dos equipamentos e instrumentos com o CLP mestre;
- Deixar o módulo de ultrafiltração pronto para utilização, deixando-o dentro da segurança necessária de operação.

1.2 Softwares

A seguir são descritos os *softwares* utilizados para o desenvolvimento do projeto:

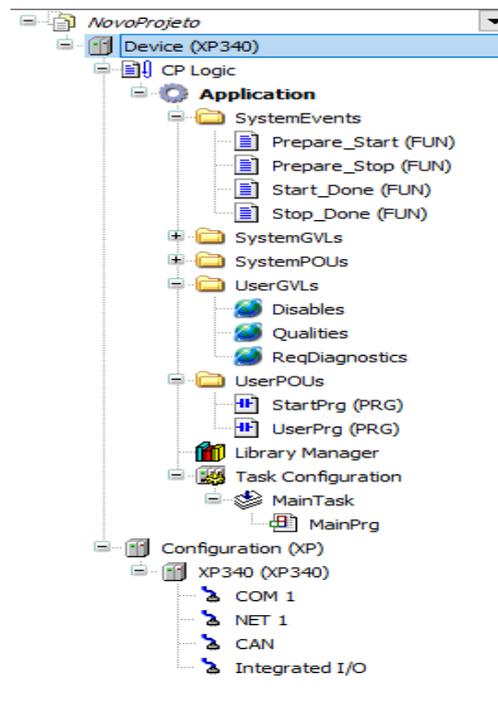
- MasterTool IEC XE, versão 3.34 (MasterTool IEC XE MT8500), para a programação do CLP XP340 da Altus. Este software é bastante didático, o que torna fácil a implementação da lógica de programação desejada. A Figura 1 mostra a tela inicial após a criação de um projeto nesse *software*, enquanto a Figura 2 mostra um *zoom* aplicado na parte lateral esquerda dessa mesma tela. Através da Figura 2 é possível realçar alguns pontos importantes, como, por exemplo: na pasta *SystemEvents* é possível realizar a adição de novas sub-rotinas de programação, de forma a deixar o programa mais estruturado; a pasta *UserGLV*, em “*Disables*”, é possível introduzir *tags* globais; na pasta *UserPOUs*, em “*UserPrg*”, tem-se o programa principal em linguagem *Ladder* do projeto. Demais informações são encontradas na aba de ajuda do próprio MasterTool.

Figura 1: Tela inicial após criação de projeto no MasterTool IEC XE



Fonte: De autoria própria

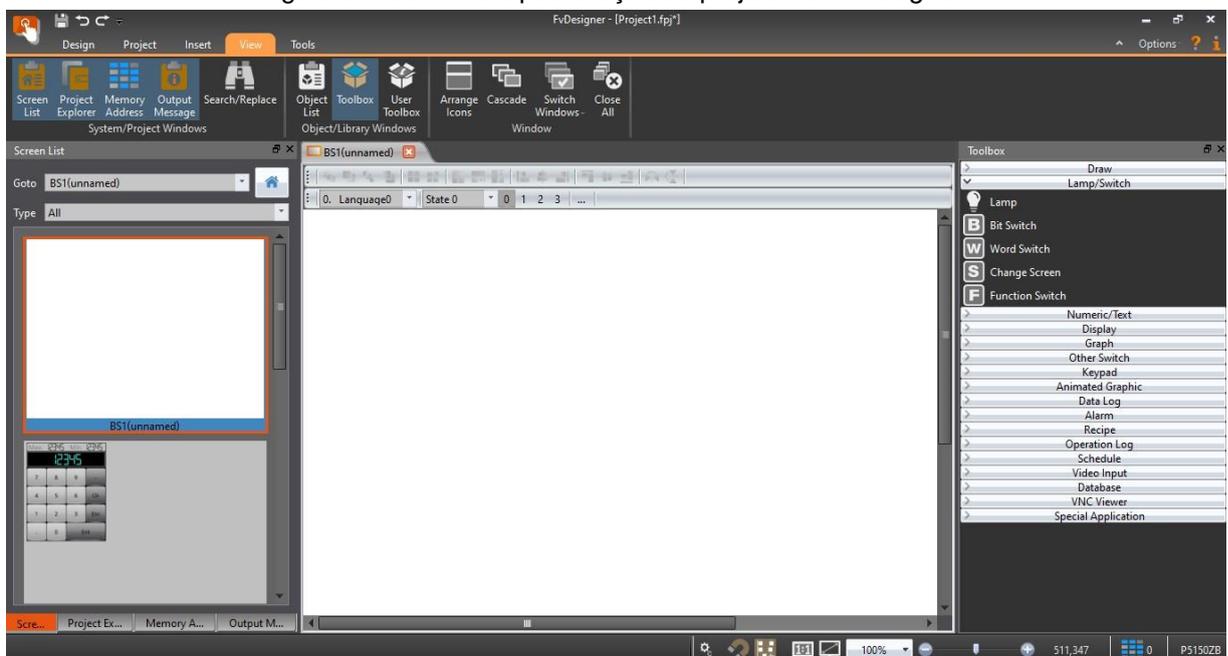
Figura 2: Zoom da tela inicial após criação de projeto no MasterTool IEC XE



Fonte: De autoria própria

- FvDesigner, versão 1.6 (FvDesigner Manual de Utilização, 2021), para a criação das telas da IHM da Altus. Da mesma forma que o MasterTool IEC XE, o *software* é fácil de se trabalhar e possui diversas ferramentas gráficas que podem facilitar no monitoramento de diversos processos. A tela inicial, após a criação de um projeto novo, pode ser vista na Figura 3. Percebe-se no menu lateral direito a variedade de objetos e funções que é possível implementar durante a programação. Também, conta com recursos como *scripts*, *data logs*, receitas, entre outros, que ajudam no desenvolvimento de projetos mais avançados.

Figura 3: Tela inicial após criação de projeto no *FvDesigner*



Fonte: De autoria própria

1.3 Instrumentos e Equipamentos Utilizados

Segue abaixo os instrumentos industriais utilizados no projeto:

- CLP XP340 (Características Técnicas XP3xx), atuando como equipamento mestre de todo o processo. Nele está localizada toda a lógica de programação, incluindo a parametrização da comunicação entre os demais instrumentos da planta. Esse modelo conta com 16 entradas e 16 saídas

digitais, 5 entradas analógicas, 2 entradas analógicas RTD (*Resistance Temperature Detector*), 4 saídas analógicas e com capacidade de desenvolvimento de um *web server* no próprio *software*;

- CLP XP315 (Características Técnicas XP3xx), servindo como uma expansão de I/O's (*Inputs/Outputs*). Dispositivo escravo do XP340, tendo a comunicação com o mesmo através do protocolo *CAN-open* – protocolo baseado na Rede CAN (*Controller Area Network*), porém que atua acima da camada física e da camada de enlace do Modelo OSI (*Open Systems Interconnection*). Instrumento que possui 16 entradas e 16 saídas digitais, 5 entradas analógicas e 2 entradas analógicas RTD;
- IHM P2101NA de 10,1 polegadas (Terminais de Operação Série P2), que realiza uma comunicação com o CLP mestre por meio do protocolo Modbus TCP (*Transmission Control Protocol*). Informações detalhadas sobre esse protocolo podem ser encontradas na Seção 2.6. No contexto do projeto, o *display* é utilizado para monitorar, coletar informações importantes e até mesmo operar a planta de forma manual;
- Inversor de frequência Optidrive ODE 3 (Invertek Optidrive E3 IP20-IP66 Manual User Guider), cuja finalidade é a realização do controle da bomba de alimentação do módulo de ultrafiltração. Também é um dos dispositivos escravos do XP340 e tem uma comunicação com o mesmo através do protocolo *CAN-open*;
- Dois hidrômetros de pulsos, sendo um localizado na entrada do módulo e um outro na saída. Ambos enviam um pulso ao CLP, através da entrada digital correspondente, quando se passam 100 litros de água;
- Dez válvulas solenoides para o controle das etapas dos processos. Oito delas estão equipadas com atuadores pneumáticos e monitoramento de posição, enquanto duas são elétricas e não possuem monitoramento;
- Quatro medidores de pressão. Dois localizados na entrada do módulo, antes e após um filtro de disco, e os outros dois nas saídas do módulo. O *range* de operação vai de 0 a 100 bar, embora a pressão do módulo seja muito baixa, chegando, aproximadamente, em 0,7 bar em condições

favoráveis de operação. Neste caso, foram implementados filtros a fim de minimizar as variações das medições;

- Medidor de temperatura, localizado na entrada do módulo, após um filtro de disco. Seu principal objetivo é garantir que a água não entre em temperaturas elevadas, para evitar danos nas membranas. O instrumento possui uma faixa de medição de 0 a 100° C;
- Medidor de Potencial de Oxidação-Redução (ORP), localizado na saída da água filtrada do módulo. Sua função é detectar a capacidade de oxidação ou redução da água, com o objetivo de avaliar sua qualidade. A faixa de medição do equipamento varia de -1000 a 1000 mV.

Na Figura 4, é possível visualizar a cápsula HYDRAcap MAX 60. Nela, são apresentadas algumas informações importantes, como: entrada de água da membrana, representada em amarelo por *'feed'*; drenagem da água da membrana, representada em vermelho por *'drain'*; entrada de ar, representada em cinza por *'air'*; saída da água pós filtração, representada em azul por *'filtrate'* e; saída para purga, representada em vermelho por *'concentrate'*.

Figura 4: Membrana de ultrafiltração de água



Fonte: (HYDRAcap MAX Technical Manual, 2019)

Já em relação aos modos de operação nos quais ela pode funcionar, podem ser vistos a seguir, de acordo com o manual (HYDRAcap MAX Technical Manual, 2019):

- *Filtration*: neste modo de operação, a água bruta entra pelas membranas sob baixa pressão e é transformada em água filtrada, podendo ser direcionada para algum reservatório. Esse modo ainda pode ser subdividido em dois grupos: *Direct Filtration* e *Crossflow Filtration*. A diferença reside no fato de que o *Crossflow Filtration* é necessário apenas quando a turbidez da água está elevada, o que demanda a abertura periódica de uma válvula de purga para garantir a eficiência do processo;
- *Air Scour*: consiste em uma limpeza rápida utilizando ar de baixa pressão de forma a retirar as partículas sólidas que ficam nas superfícies das membranas. Ao encerrar o processo do modo *Air Scour*, é dado retorno ao modo *Filtration*;
- *Maintenance Clean*: consiste em uma limpeza química para restaurar o módulo. Este modo de operação pode ser subdividido nos modos de operação “MC1”, “MC2”, “MC3” e “MC1+2”. No modo “MC1”, realiza-se uma limpeza com cloro. No modo “MC2”, realiza-se uma limpeza com soda cáustica. No modo “MC3”, realiza-se uma limpeza com ácidos. Por fim, no modo “MC1+2”, realiza-se a junção dos processos dos modos MC1 e MC2. Ao encerrar a limpeza pelo modo *Maintenance Clean*, é dado retorno ao modo *Filtration*;
- *Recovery Clean*: consiste na realização de uma limpeza química mais intensa, restaurando completamente as membranas do módulo. Ocorre de forma menos frequente que os demais modos de operação, geralmente sendo uma vez a cada 30 a 60 dias. Da mesma forma que os

modos anteriores, após o término do processo, o módulo é retornado ao modo *Filtration*;

- *Membrane Integrity Testing*: este modo de operação realiza um procedimento de retenção de pressão de ar para verificar falha nas fibras. Esse modo não foi implementado na lógica de programação do projeto, uma vez que foi decidido que seria realizado de forma manual.

1.4 Organização do Trabalho

Este trabalho foi dividido em quatro capítulos de forma a organizar e estruturar as informações decorrentes do projeto de ultrafiltração.

No Capítulo 2, é apresentada a fundamentação teórica, explicando detalhadamente alguns conceitos importantes relacionados ao projeto desenvolvido, como: o que é o Modelo OSI e sua divisão em camadas, exemplos de protocolos de comunicações e outros tópicos de suma relevância para o desenvolvimento do projeto.

No Capítulo 3, é mostrado o desenvolvimento do trabalho, onde é possível verificar, passo a passo, os procedimentos aplicados para chegar aos resultados esperados.

Já no Capítulo 4, são apresentadas a conclusão do projeto e algumas propostas de continuidade.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, encontra-se a explicação básica para o entendimento do desenvolvimento do projeto, abrangendo conhecimentos a cerca de alguns protocolos de comunicação utilizados, como é o caso do protocolo Modbus e do protocolo CAN (*Controller Area Network*), assim como também das funcionalidades principais da linguagem *ladder*.

2.1 Protocolo Modbus

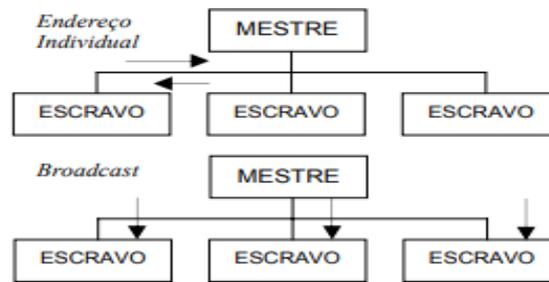
O protocolo Modbus encontra-se na camada de Aplicação do Modelo OSI e é responsável pelo estabelecimento da comunicação entre dispositivos. É amplamente utilizado nos dias atuais devido à sua versatilidade, permitindo a troca fácil de dados entre equipamentos de fabricantes distintos.

Existem várias variações do protocolo Modbus. A primeira a surgir foi o protocolo Modbus RTU (*Remote Terminal Unit*), desenvolvido no final da década de 70 pela Modicon (Protocolo Modbus (RTU / TCP & ASCII), 2022). Posteriormente, surgiram outras variações, como o protocolo Modbus ASCII e o protocolo Modbus TCP, sendo este último utilizado no projeto atual. A principal diferença entre essas variações reside na infraestrutura de rede e no meio físico. Por exemplo, o Modbus RTU utiliza comunicação serial, enquanto o Modbus TCP faz uso de redes Ethernet baseadas em TCP/IP.

A comunicação é feita baseada na topologia mestre-escravo, onde o dispositivo mestre atua iniciando a comunicação e realizando a solicitação das informações desejadas para os escravos e, os escravos, enviam as informações que lhe foram solicitadas.

Na Figura 5 a seguir, é possível visualizar a demonstração da topologia mestre-escravo. Como é possível perceber, as mensagens das trocas de dados podem ser endereçadas de forma individual ou, para o caso de uma arquitetura de rede TCP/IP, através do *broadcast*, onde o dispositivo mestre consegue acessar todos os dispositivos escravos da rede de uma só vez.

Figura 5: Topologia mestre-escravo



Fonte: (NASCIMENTO e LUCENA, 2003)

Basicamente, a troca de dados é feita através do PDU (*Protocol Data Unit*), que tem a função de especificar o tipo de dado que vai ser lido e se vai servir apenas de leitura, escrita ou ambos. Abaixo, na Tabela 1, segue o nome das funções, o tamanho do dado a ser feito a troca de informação e o seu nível de acesso:

Tabela 1 – Tipos de dados Modbus

Função	Tamanho	Acesso
<i>Coil Status (0X)</i>	1 bit	Leitura e escrita
<i>Input Status (1X)</i>	1 bit	Somente leitura
<i>Input Registers (3X)</i>	16 bits	Somente leitura
<i>Holding Registers (4X)</i>	16 bits	Leitura e escrita

Fonte: Retirado de (Modbus TCP, Manual do Usuário, 2013)

É fundamental prestar atenção nas duas formas de nomenclaturas das funções (a segunda está indicada entre parênteses na Tabela 1). Isso se deve ao fato de que, dependendo do *software* utilizado, pode-se optar por uma das duas abordagens. Para ilustrar essa diferença, no projeto, o MasterTool IEC XE chama uma função através do seu nome completo, enquanto o FvDesigner utiliza a sua forma abreviada (0X, 1X, 3X ou 4X) como exemplificado.

Com o conhecimento dessas informações, o programador pode manipular os dados entre os dispositivos da maneira que melhor se adequa às necessidades do sistema. Por exemplo, ao acionar um contato aberto da lógica de um CLP por meio de uma IHM, estando utilizando o protocolo Modbus, é possível endereçá-lo como

uma *coil*. Da mesma forma, para efetuar a leitura de uma entrada analógica e visualizá-la na IHM, basta endereçar esse dado como um *input register*.

Seguindo essa abordagem, é possível mapear todos os endereços necessários para a troca de dados. É crucial enfatizar que cada dado deve possuir um endereço único para evitar conflitos na transferência de informações. Dessa forma, embora a aplicação do protocolo seja vantajosa, é importante ter em mente o tempo que é necessário para realizar o endereçamento e a classificação de todos os dados que são necessários, principalmente em um projeto de grande escala.

2.2 Protocolo CAN

Inicialmente, o protocolo CAN foi criado com uma finalidade específica: atender às demandas da indústria automobilística. A empresa alemã BOSCH desenvolveu essa tecnologia para a implementação de um meio de comunicação entre os dispositivos eletrônicos de caminhões e ônibus, por volta da década de 80 (CAN Bus: Conceituação). Com o passar dos anos, esse protocolo evoluiu e deu origem a diversas variações, de maneira semelhante ao que ocorreu com o protocolo Modbus. Como exemplo, tem-se o DeviceNet, desenvolvido pela *Allen Bradley*, e o protocolo CANopen – utilizado no projeto atual para a comunicação com o inversor Optidrive ODE 3 e o CLP XP315.

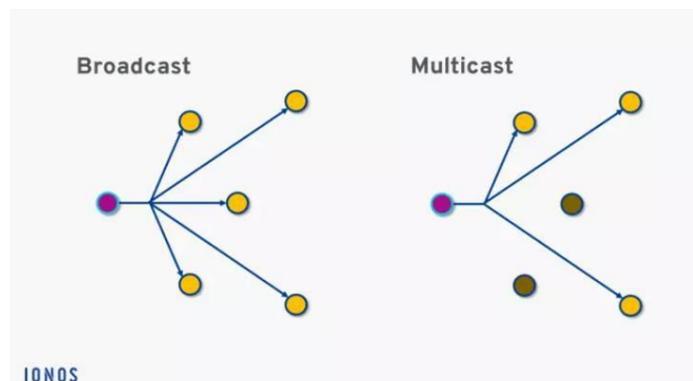
O protocolo CAN opera nas camadas Física e de Enlace de Dados do Modelo OSI e é um protocolo de comunicação serial. Ele é capaz de trocar informações em uma topologia mestre-escravo, como discutido anteriormente na seção anterior sobre o protocolo Modbus, e também suporta a topologia multi-mestre. Na topologia multi-mestre, qualquer controlador pode assumir o papel de mestre em momentos específicos e transmitir mensagens na rede de forma independente.

É importante notar que as mensagens enviadas neste protocolo de comunicação utilizam a técnica de comunicação *multicast*, permitindo que um único dispositivo envie a mesma mensagem para vários receptores na rede. Enquanto o *broadcast* envia mensagens para todos os dispositivos da rede simultaneamente, o *multicast* oferece a flexibilidade de selecionar receptores específicos de interesse. Para uma

compreensão mais clara sobre a diferença entre o *broadcast* e o *multicast*, basta visualizar a Figura 6.

No CAN, os dados são transmitidos através de bits representando níveis lógicos altos (1) e baixos (0). Os primeiros bits do pacote de dados estão associados à prioridade da mensagem a ser transmitida. Essa abordagem permite identificar qual mensagem possui prioridade superior, garantindo que ela seja transmitida antes das demais. Esse processo contribui para estabelecer uma comunicação segura e eficiente da rede CAN.

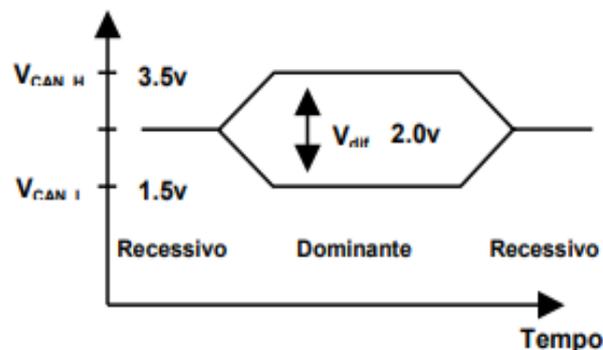
Figura 6: *Broadcast versus multicast*



Fonte: (IONOS, 2023)

Nas redes CAN, os barramentos geralmente são conectados em rede através de dois fios. Os sinais transmitidos são diferenciados como *CAN High* e *CAN Low*, e os bits são interpretados como dominantes ou recessivos nesses dois sinais. Essa diferenciação é essencial para evitar colisões de dados, especialmente em topologias multi-mestre, onde vários controladores podem tentar escrever dados simultaneamente. A Figura 7 ilustra um exemplo de bits dominantes e recessivos para uma melhor compreensão. É importante notar que a tensão para os bits dominantes e recessivos são fornecidas com magnitudes diferentes.

Figura 7: Bits dominantes e recessivos no CAN



Fonte: (GUIMARÃES e SARAIVA, 2002)

Em comparação com o protocolo CAN, o CANopen atua na camada de Aplicação do Modelo OSI, o que o coloca em uma camada mais elevada. Inicialmente, o CANopen teve como foco principal sua aplicação em máquinas orientadas ao movimento (CANopen - A Rede Embarcada Padronizada). Este protocolo oferece a capacidade de conectar diversos dispositivos em uma rede, mantendo um alto nível de padronização, tornando-o particularmente mais relevante em ambientes industriais.

Um aspecto crítico para estabelecer a comunicação entre os dispositivos, o que se aplica a todos os protocolos baseados no CAN, é a configuração precisa dos endereços dos nós dos dispositivos. Qualquer configuração incorreta pode resultar em problemas de funcionamento na rede.

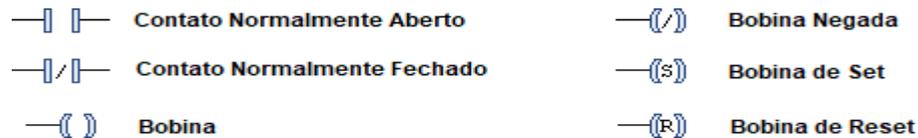
2.3 Linguagem *Ladder*

A linguagem *ladder* é uma das linguagens mais populares para programação de Controladores Lógicos Programáveis (CLP's). Ela foi desenvolvida tomando como referência comandos elétricos e, através dela, é possível visualizar o comportamento da lógica de controle por trás dos sistemas eletrônicos e elétricos.

Na Figura 8 é possível ver os comandos mais básicos na linha de programação que estão presentes em quase todo o projeto. São eles: contato normalmente aberto, contato normalmente fechado, bobina, bobina negada, bobina de set e bobina de

reset. O contato normalmente aberto conduz quando está em nível lógico alto, enquanto o contato normalmente fechado possui a lógica inversa. O mesmo se aplica às bobinas, a única diferença se encontra nas bobinas de set e reset. A bobina de set deixa mantido em nível lógico alto a saída, enquanto a bobina de reset a desliga.

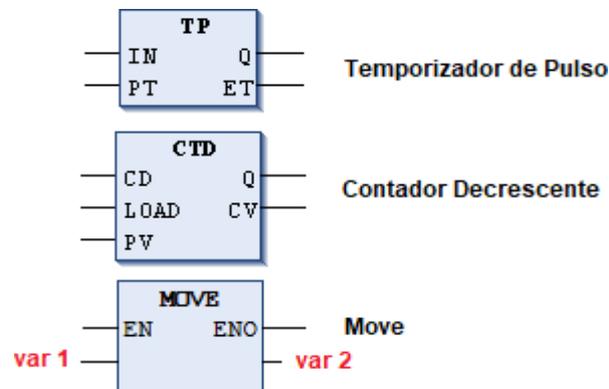
Figura 8: Instruções básicas em *ladder*



Fonte: De autoria própria

Também é possível encontrar diversos blocos de função dentro do programa. Cada função contém uma funcionalidade específica. Alguns exemplos podem ser vistos na Figura 9, onde são apresentados blocos que agem como um temporizador de pulso, um contador decrescente e o bloco *move*. O temporizador de pulso é um temporizador que, independentemente do tempo que ele é ativado, mantém a saída em nível lógico alto pelo tempo preestabelecido pelo usuário. O contador decrescente serve para realizar a contagem de forma regressiva e, o bloco *move* para mandar o valor de uma variável para uma outra.

Figura 9: Exemplos de blocos de função em *ladder*



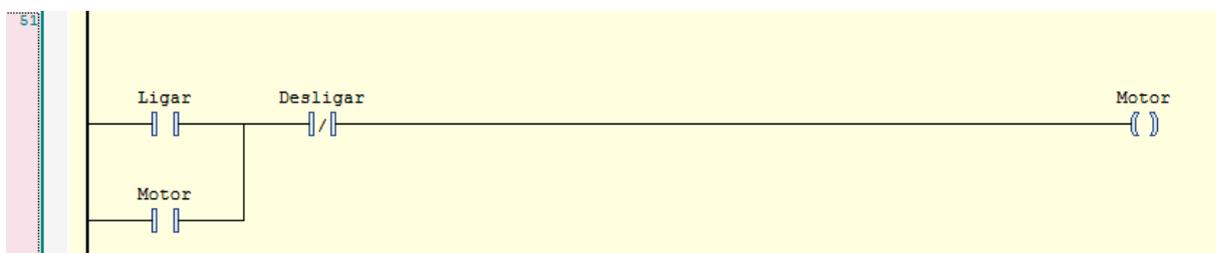
Fonte: De autoria própria

São só algumas das mais variadas instruções de lógica que podem ser aplicadas ao código. Blocos com funcionalidades mais complexas também existem e são aplicados dependendo da necessidade.

Dentro do projeto, a lógica vai sendo dividida em '*rungs*', que representam as linhas na programação. Cada linha que o programador adiciona pode conter uma lógica distinta. É importante ressaltar que a leitura e execução pelo CLP ocorrem de forma sequencial, da esquerda para a direita e de cima para baixo.

Um exemplo comumente aplicado na programação em *ladder* é o chamado 'contato de selo', como demonstrado na Figura 10. Essa lógica envolve a ativação de um motor após um pulso gerado por uma botoeira, contato normalmente aberto representado pela *tag* 'Ligar'. Uma vez acionado, o contato de selo mantém o motor em funcionamento, mesmo quando o botão momentâneo de partida retorne ao nível lógico zero. O motor, representado pela bobina com a *tag* 'Motor', permanece ligado até que o botão de desligar, contato normalmente fechado, representado pela *tag* 'Desligar', seja acionado.

Figura 10: Contato selo em *ladder*



Fonte: De autoria própria

Esse foi só um dos inúmeros exemplos que podem ser feitos através da linguagem *ladder*. O projeto atual consiste em uma junção de diversas linhas de comandos com os mais variados tipos de blocos de função para a obtenção do resultado esperado.

Dependendo do projeto, a complexidade pode acabar se tornando algo bem relevante. Para isso, é feito alguns métodos de forma a simplificar a linha de raciocínio. Um desses, que foi introduzido no projeto atual de ultrafiltração, foi a máquina de estados. Pode ser visto detalhadamente no próximo tópico a seguir.

2.3.1 Máquina de Estados Finitos

A máquina de estados finitos (FSM – *Finite State Machine*), mais conhecida apenas como máquina de estados, é uma técnica de programação que consiste em um modelo matemático que é capaz de representar processos cíclicos, podendo ser aplicada em processos de automação industrial (SENA, 2013). A aplicação é feita de forma bem prática, o que permite a qualquer pessoa com o conhecimento no assunto realizar algum projeto complexo.

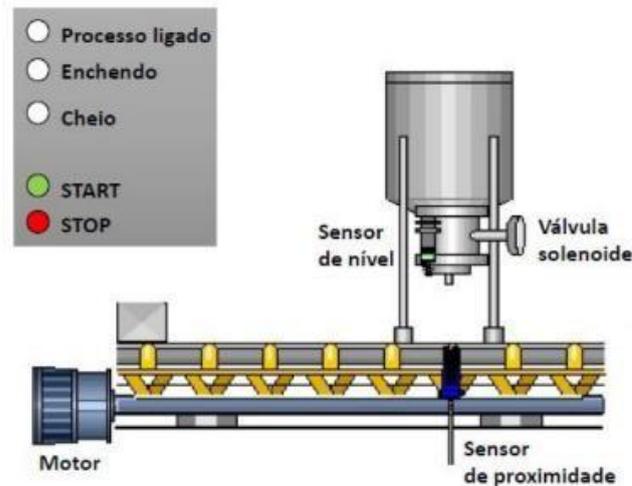
Basicamente, um estado representa uma condição atual do processo e, em *ladder*, está associado a uma memória. Dessa forma, cada condição diferente em que o processo se encontra, um estado específico será criado. A mudança de uma condição para outra, neste caso de um estado para outro, ocorre quando satisfaz a condição do que é chamado de transição. A transição é imposta pelo programador e vai dizer qual comportamento que o processo deve cumprir para que migre de um estado a outro. Geralmente as transições estão relacionadas às entradas digitais de um CLP, como a atuação de um sensor, por exemplo. Como cada estado é único, em uma FSM, é preciso que seja declarado quais saídas estarão atuadas e quais não vão estar atuadas, para que a lógica funcione adequadamente.

Para a implementação dessa técnica, é recomendado que a pessoa responsável pela programação faça um levantamento das entradas e das saídas de todo o projeto. Em seguida, que seja realizada a montagem da máquina de estados, juntamente com o mapeamento das transições. Por fim, basta especificar quais ações devem ser tomadas em cada estado.

Um exemplo pode ser visto na Figura 11. Ela mostra um processo de automação que carrega automaticamente caixas que passam pela esteira.

Segue a descrição do processo requerido: o motor deve ligar, atuando a esteira, de forma a conduzi-la até o sensor de proximidade. Chegando nesta posição, a esteira deve ser desligada para dar início ao processo de enchimento da mesma. A válvula solenoide é então aberta para dar início ao carregamento e é fechada até o sensor de nível atuar, sinalizando que o carregamento está completo. Ao chegar nessa condição, a esteira é novamente ligada e então a máquina, cíclica, aguarda uma nova caixa para repetir a operação.

Figura 11: Exemplo de um processo industrial

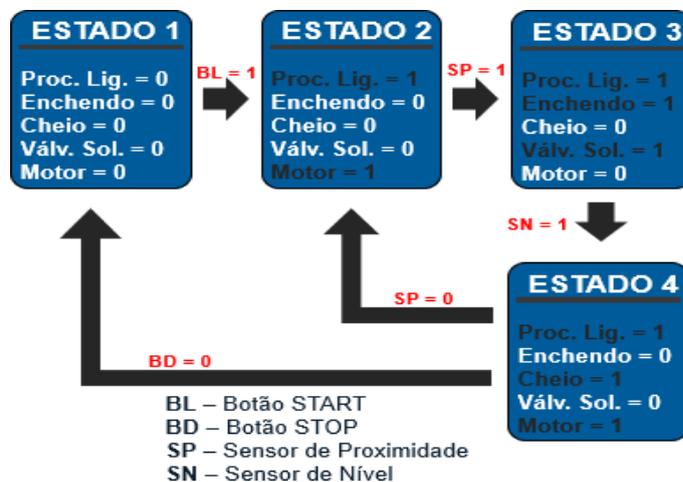


Fonte: (NETO, 2021)

Algumas outras condições são: iniciar o processo quando o botão 'start' for pressionado, além de acender o sinaleiro 'processo ligado'; interromper o processo quando o botão 'stop' for pressionado, além de desligar o sinaleiro 'processo ligado'; ligar o sinaleiro 'enchendo' apenas quando estiver na etapa de enchimento da caixa e; ligar o sinaleiro 'cheio' quando o sensor de nível alto estiver atuado.

A máquina de estados correspondente a esse processo pode ser vista a seguir na Figura 12:

Figura 12: Máquina de estados



Fonte: De autoria própria, inspirado em (NETO, 2021)

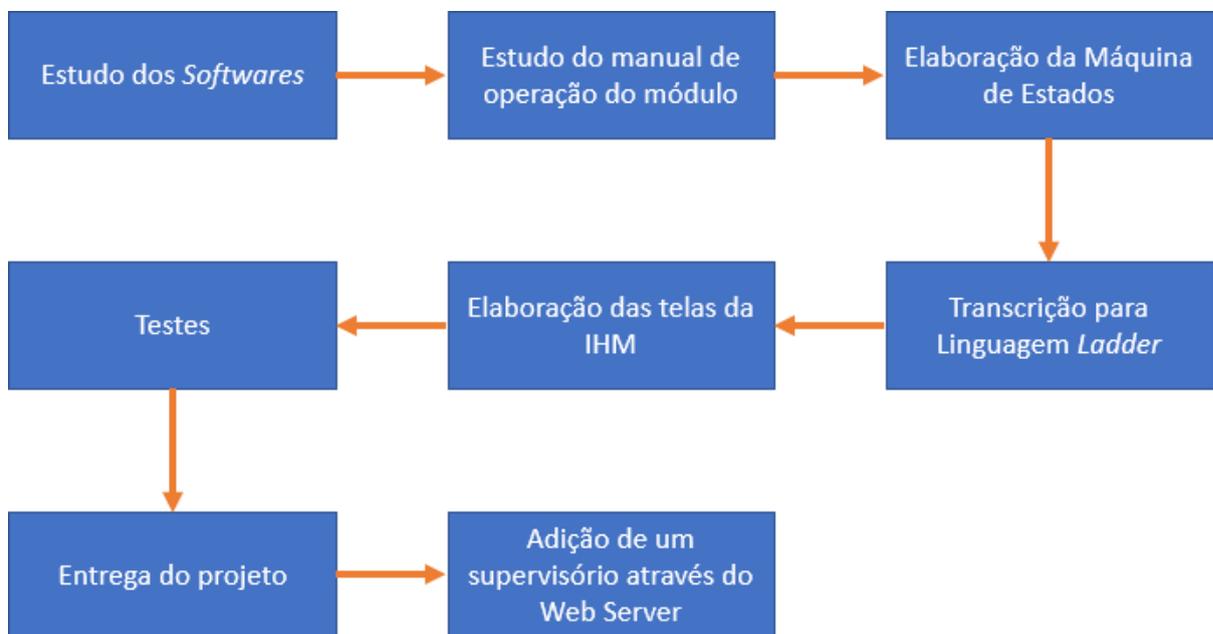
Como é possível perceber, o estado 1 corresponde à máquina desligada. O estado 2 quando a máquina está com a esteira ligada, movimentando a caixa. O estado 3 quando a caixa já se encontra na posição de enchimento e, o estado 4 quando a caixa enche por completo e está na iminência de sair da posição devido a esteira ligada.

Para a transição desse diagrama de estados para *ladder*, basta realizar lógicas simples para atuação de memórias, entradas e saídas correspondentes a cada estado e suas respectivas ações.

3 DESENVOLVIMENTO DO TRABALHO

Este capítulo será dividido em seções, abordando detalhadamente o desenvolvimento de alguns temas específicos, de forma que fique claro como se chegou no resultado esperado. Inicialmente, é fundamental compreender a sequência cronológica que se foi adotada, o que pode ser visto, de forma geral, analisando a Figura 13 a seguir:

Figura 13: Passos do desenvolvimento do projeto



Fonte: De autoria própria

Como é possível observar, toda a preparação do projeto foi feita através de estudos realizados a respeito dos *softwares* utilizados e, não menos importante, do manual de operação do módulo de ultrafiltração. As demais etapas, como mencionado, são aprofundadas nos subcapítulos a seguir.

3.1 Elaboração da Máquina de Estados Finitos

A máquina de estados finitos (FSM) foi a técnica de programação utilizada no desenvolvimento do projeto, desempenhando um papel fundamental na obtenção dos resultados desejados. Esta abordagem foi uma das primeiras etapas do trabalho, sendo elaborada antes mesmo de iniciar a programação em linguagem *ladder*. A razão para essa abordagem inicial é a importância de estabelecer uma estrutura clara no código, evitando confusões à medida que o projeto progride.

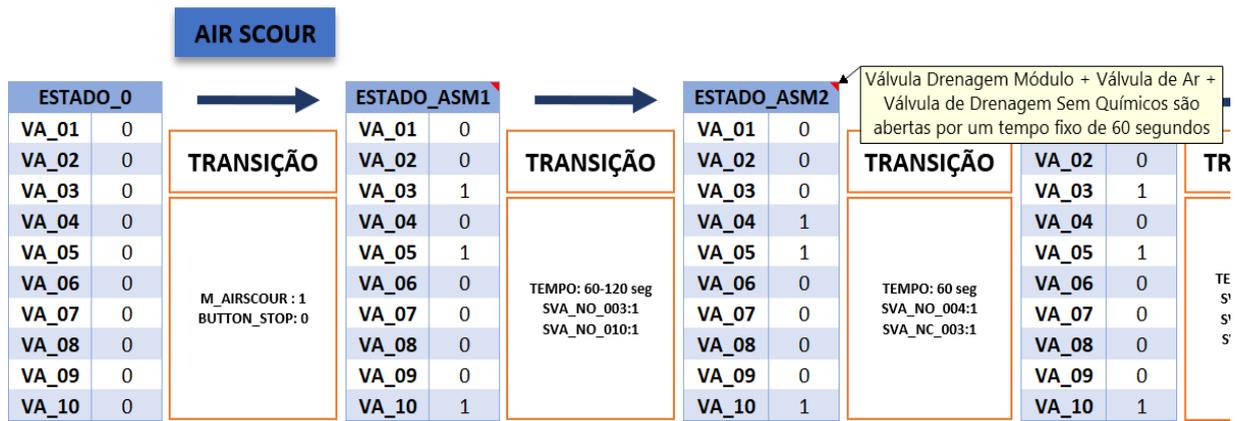
Apesar de existirem diversas saídas digitais que impactam na operação do módulo de ultrafiltração, a máquina de estados que foi criada visava apenas a lógica de abertura e fechamento das dez válvulas presentes na planta. Ela foi criada com base no manual de operação do módulo (HYDRAcap MAX Technical Manual, 2019) e, desenvolvida, visualmente, no *Microsoft Excel*.

Vale ressaltar que para cada modo de operação, foi desenvolvida uma máquina de estados diferente. Sendo assim, como há quatro modos de operação que foram trabalhados no projeto de automação, elaborou-se quatro máquinas de estados distintas. Por exemplo, no modo de operação '*Air Scour*', a máquina de estados passa por até seis estados diferentes antes de retornar ao modo de filtração. Uma parte dessa elaboração pode ser vista na Figura 14.

Como pode ser observado, a representação gráfica está devidamente detalhada, com comentários explicativos sobre o que ocorre em cada etapa do processo e quais válvulas estarão abertas ou fechadas. Além disso, ela apresenta com clareza os critérios de transição de uma etapa para a próxima.

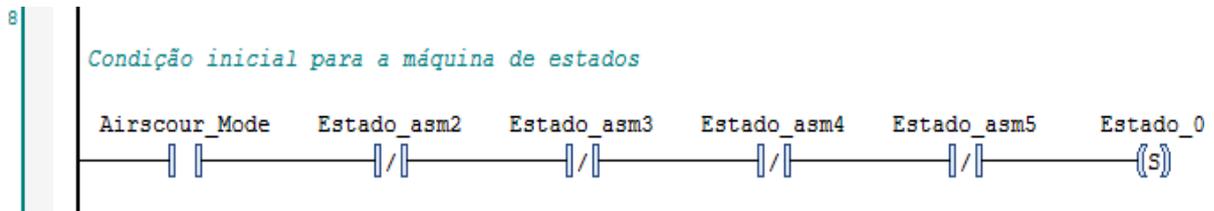
Para a implementação em linguagem *ladder*, basta realizar alguns processos relativamente simples. O 'Estado_0', visível na imagem, representa o módulo de ultrafiltração desligado. Sendo assim, para implementar no CLP essa lógica, basta colocar as memórias dos demais estados com contato fechado ligados em série e, a memória do estado inicial como uma bobina de *set*. Pode ser observada a implementação na Figura 15.

Figura 14: Fração da máquina de estados do modo *Air Scour*



Fonte: De autoria própria

Figura 15: Estado inicial do modo '*Air Scour*'

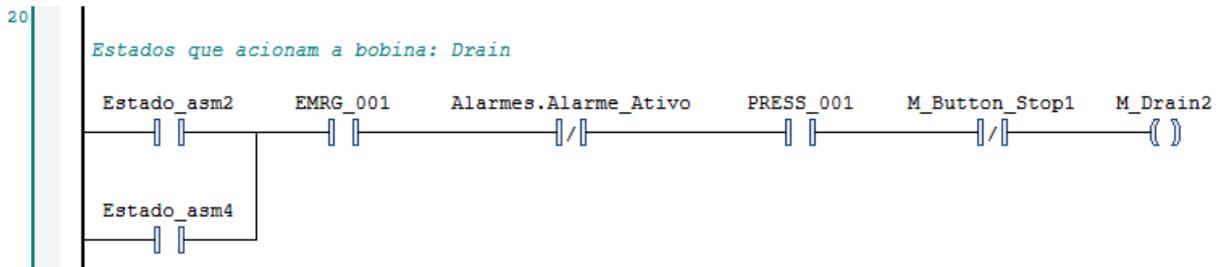


Fonte: De autoria própria

Para mudar de um estado para outro, basta colocar as condições de transição em uma nova *rung*. E, quando satisfeita a condição, acionar uma bobina para desligar o estado atual e uma outra bobina para ligar o estado seguinte.

Em relação a ativação das saídas, basta colocar em paralelo quais estados que as ativarão. É possível visualizar o exemplo para o acionamento da válvula de drenagem do projeto durante o modo de operação '*Air Scour*' na Figura 16. Vale notar os demais contatos em série antes da chegada da bobina. Eles estão relacionados aos alarmes criados para impedir que ela seja acionada em alguma ocasião crítica estabelecida.

Figura 16: Acionamento da válvula de drenagem através da máquina de estados



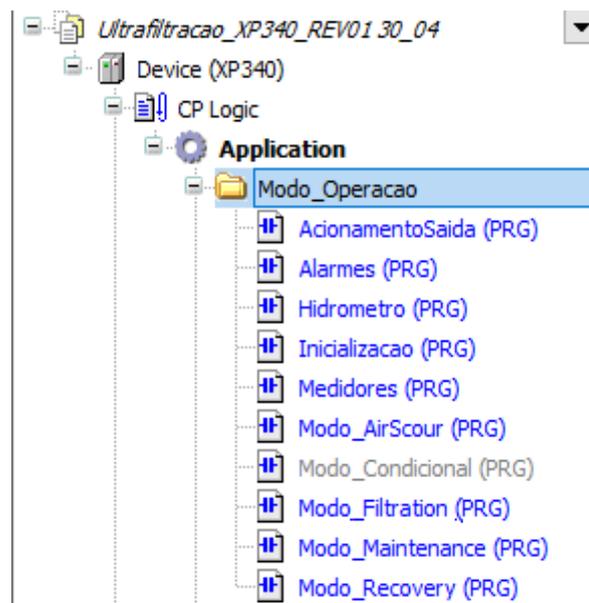
Fonte: De autoria própria

3.2 Boas Práticas de Programação Aderidas

Visando facilitar alguma possível manutenção futura, o projeto foi elaborado seguindo algumas boas práticas de programação. Isso ajuda a tornar o processo mais rápido e prático, além de informar com clareza os passos que foram realizados em cada parte do código.

Primeiramente, foi realizada uma divisão de sub-rotinas dos códigos elaborados em linguagem *ladder*, onde cada sub-rotina apresenta uma característica específica do projeto. É possível visualizar essa divisão na Figura 17 a seguir:

Figura 17: Sub-rotinas criadas no software do CLP



Fonte: De autoria própria

Ao analisar a Figura 17, nota-se que há uma sub-rotina para cada um dos modos de operação do módulo de ultrafiltração de água, além de haverem outras com funcionalidades mais específicas, como os hidrômetros e os alarmes. No caso da necessidade de modificação em alguma parte do projeto, a organização proporcionada oferece grande facilidade. É fundamental lembrar que, para as sub-rotinas atuarem adequadamente dentro do MasterTool, é preciso chamá-las dentro do programa principal.

Outra boa prática adotada foi a inclusão de comentários para as variáveis criadas. Em uma área que há muitos instrumentos semelhantes, é comum encontrar nomenclaturas parecidas dentro do projeto. Dessa forma, inserir um comentário ajuda a identificar qual a relação da variável com a parte do código que é preciso trabalhar. Um trecho contendo algumas delas pode ser visto na Figura 18:

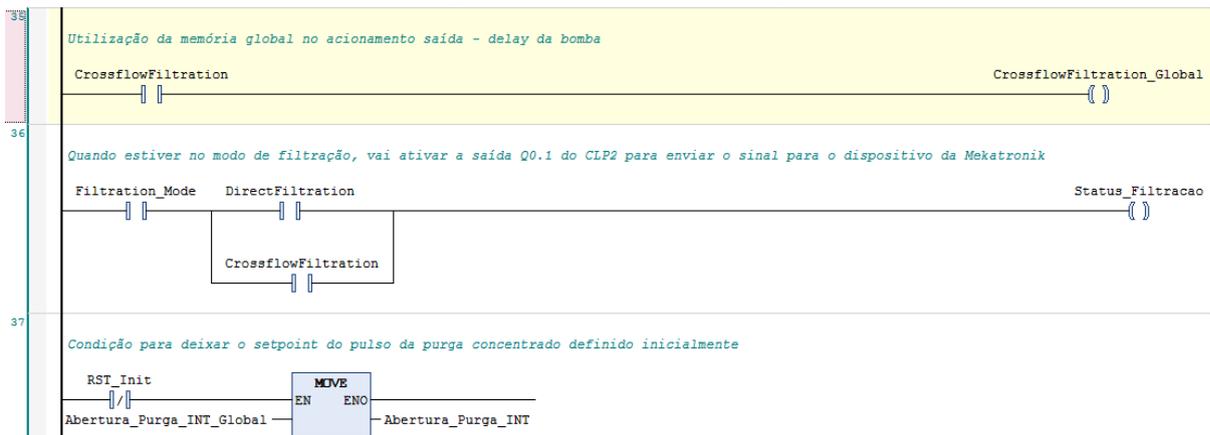
Figura 18: Comentários de variáveis

	Escopo	Nome	Endereço	Tipo de dados	Inicialização	Comentário
10	VAR_GLOBAL	LLS_004		REFERENCE TO BOOL	DigitalInputs.I02	I02_NIVEL_BAIXO_SEGURANCA_BAL
11	VAR_GLOBAL	LHS_002		REFERENCE TO BOOL	DigitalInputs.I03	I03_NIVEL_ALTO_LIG/DESL
12	VAR_GLOBAL	LLS_008		REFERENCE TO BOOL	DigitalInputs.I04	I04_NIVEL_BAIXO_TRC001_PROTECAO_BRC
13	VAR_GLOBAL	LLS_005		REFERENCE TO BOOL	DigitalInputs.I05	I05_NIVEL_BAIXO_TQ006_PROTECAO_BDAC_001
14	VAR_GLOBAL	LLS_006		REFERENCE TO BOOL	DigitalInputs.I06	I06_NIVEL_BAIXO_TQ005_PROTECAO_BDNAOH_001
15	VAR_GLOBAL	LLS_007		REFERENCE TO BOOL	DigitalInputs.I07	I07_NIVEL_BAIXO_TQ004_PROTECAO_BDNAOCL_002_BDNAOCL_003
16	VAR_GLOBAL	SVA_NO_001		REFERENCE TO BOOL	DigitalInputs.I10	I10_STATUS_VALV_FEED_OPEN
17	VAR_GLOBAL	SVA_NO_002		REFERENCE TO BOOL	DigitalInputs.I11	I11_STATUS_VALV_FILTRATE_OPEN
18	VAR_GLOBAL	SVA_NO_003		REFERENCE TO BOOL	DigitalInputs.I12	I12_STATUS_VALV_CONCENTRATE_OPEN
19	VAR_GLOBAL	SVA_NO_004		REFERENCE TO BOOL	DigitalInputs.I13	I13_STATUS_VALV_DRAIN_OPEN
20	VAR_GLOBAL	SVA_NO_006		REFERENCE TO BOOL	DigitalInputs.I14	I14_STATUS_VALV_DRAINFILTRATE_OPEN
21	VAR_GLOBAL	SVA_NO_007		REFERENCE TO BOOL	DigitalInputs.I15	I15_STATUS_VALV_RC_OPEN
22	VAR_GLOBAL	SVA_NO_009		REFERENCE TO BOOL	DigitalInputs.I16	I16_STATUS_VALV_DRAINCHIMICAL_OPEN
23	VAR_GLOBAL	SVA_NO_010		REFERENCE TO BOOL	DigitalInputs.I17	I17_STATUS_VALV_DRAIN_NOCHIMICAL_OPEN
24	VAR_GLOBAL	BRC_001		REFERENCE TO BOOL	DigitalOutputs.Q00	Q00_BOMBA_BRC
25	VAR_GLOBAL	BDNAC_001		REFERENCE TO BOOL	DigitalOutputs.Q01	Q01_BOMBA_NAC

Fonte: De autoria própria

Além de deixar comentadas as variáveis do projeto, também foram incluídos comentários em quase todas as *rungs*. Essa prática é de alta relevância, especialmente em projetos extensos, uma vez que reduz o tempo do programador para o entendimento da lógica em algum momento futuro. A Figura 19 mostra alguns comentários criados para as *rungs*. Com esse exemplo, já é notável que, através da *rung* 36, o CLP XP315 tem uma de suas saídas digitais acionadas para enviar um sinal a um equipamento de outra marca.

Figura 19: Comentário de rungs



Fonte: De autoria própria

Como o cliente na maioria das vezes pede mudanças no projeto, deixar a estrutura do mesmo organizada e comentada ajuda a ganhar tempo na hora das modificações.

3.3 Parametrização dos Medidores

Como mencionado anteriormente, os instrumentos de medições utilizados no projeto de ultrafiltração incluem um medidor de ORP, medidor de temperatura e quatro medidores de pressão. Todos eles sendo representados por sinais de 4 a 20 mA. Isso significa que, no limite inferior de sua faixa de medição, esses medidores fornecem um sinal de 4 mA ao CLP, assim como fornecem um sinal de 20 mA para a medida máxima.

Como o range dos medidores de pressão é relativamente alto em comparação à pressão de trabalho do módulo, foi necessário estabilizar o sinal medido por meio de um filtro digital de 10 segundos. Apesar de não ser o adequado, pois deixa o tempo de resposta lento para um controle crítico do funcionamento do módulo, o filtro digital estabilizou o sinal de pressão medido, ainda deixando o projeto funcionando dentro do esperado. Sendo assim, quando ocorria alta pressão no módulo, a detecção conseguia ser eficiente, no entanto com a resposta um pouco demorada, mas não o suficiente para causar danos ao maquinário. Todos os medidores de pressão estão

associados às entradas analógicas do CLP XP315, que está no projeto atuando como uma expansão de I/O's.

O transmissor e indicador de temperatura foi selecionado com um range adequado para o projeto, de 0 até 100°C. Para este caso, foi implementado um filtro digital de 1 segundo apenas para deixar o sinal um pouco mais robusto. A marcação de temperatura vista pelo CLP, após a devida conversão em *ladder*, estava de acordo com a medição mostrada no equipamento.

Já o medidor de ORP foi o único instrumento de medição que estava se apresentando um pouco inconsistente. Segundo o manual, o seu range de operação vai de -1000 a 1000 mV. No entanto, após as conversões em *ladder*, a medição indicada no instrumento apresentava um valor diferente do CLP para certas faixas de medição. Para esse caso em particular, como a faixa de operação necessária não era tão longa (aproximadamente de 200 a 800 mV), foi então calibrado para o CLP retornar o valor mais próximo possível da indicação dentro dessa faixa.

É possível visualizar na imagem abaixo o bloco 'LIN_TRAFO' que faz automaticamente a conversão da corrente 4 a 20 mA para a unidade específica de medida. Neste caso, para a medição da temperatura em graus Celsius.

Figura 20: Leitura da entrada analógica do medidor de temperatura



Fonte: De autoria própria

Na segunda linha vista no código da Figura 20, é possível visualizar o bloco 'MOVE', que está servindo para transpor o valor da medição da temperatura em uma outra variável, deixando mais fácil manipulá-la posteriormente dentro do projeto. Essa mesma *tag* está sendo redirecionada para a IHM.

3.4 Hidrômetros de Pulso

O hidrômetro é um instrumento de medição que consegue quantificar o consumo de água que passa por ele. Quando se refere a um 'hidrômetro de pulso', significa que ele emite um pulso para uma entrada digital de um CLP (se esse for o dispositivo de controle) quando um determinado volume de água é consumido. Em situações em que a única necessidade é contabilizar o volume total de água consumido e registrar os dados, é possível implementar um simples contador para executar essa tarefa.

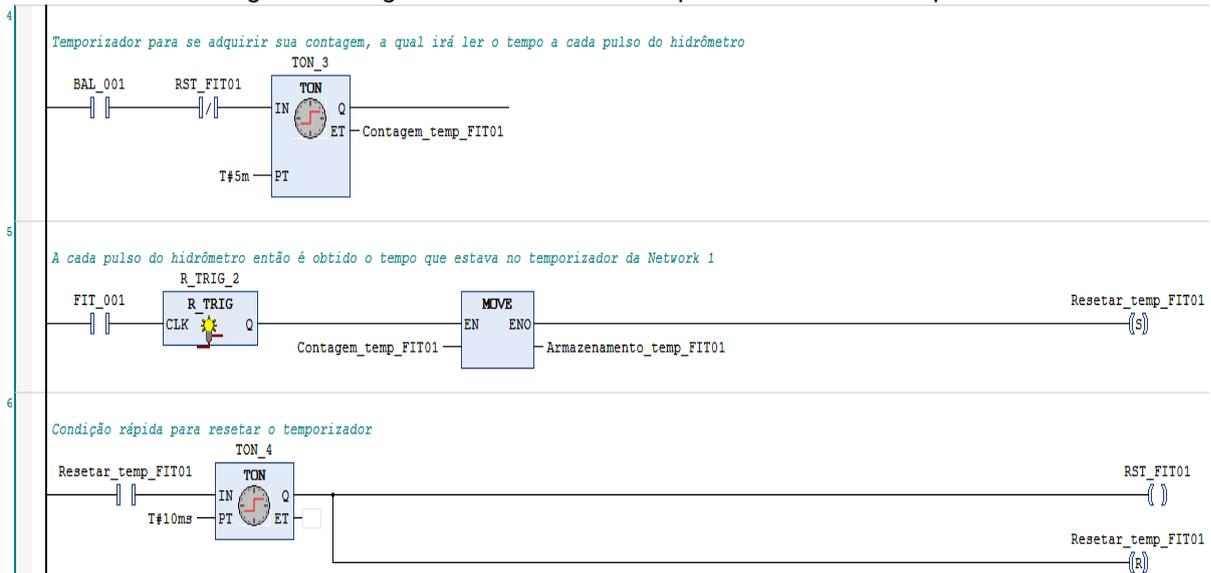
No contexto do projeto atual, a necessidade não se limita apenas a contabilizar a vazão total. Também, foi fundamental realizar o controle automatizado do inversor para ajustar a frequência da bomba de alimentação com base na leitura do hidrômetro. Nesse cenário, o operador seleciona um valor de referência para a vazão, e o sistema, utilizando os dados do hidrômetro, ajusta automaticamente a velocidade da bomba.

O ponto crucial para chegar no resultado esperado é a obtenção do tempo em que o hidrômetro está dando cada pulso, para obter, dessa forma, a vazão por unidade de tempo. É possível visualizar na Figura 21 a lógica implementada para obter o tempo que o hidrômetro leva para dar cada pulso.

Conforme observado na Figura 21, sempre que a bomba de alimentação (BAL_001) é ativada, um temporizador inicia a contagem do tempo. Quando o hidrômetro emite um pulso, indicando que leu 100 litros (no cenário atual), o valor do tempo decorrido no temporizador é imediatamente transferido para uma variável de armazenamento. Além disso, a contagem do temporizador é reiniciada, permitindo a captura de um novo registro. Esse processo permite determinar o tempo de duração entre os ciclos de pulsos do hidrômetro.

Com a descoberta desse tempo e tendo noção do valor do volume de água que é registrado, é possível realizar o controle automatizado da bomba de alimentação através do inversor de frequência.

Figura 21: Lógica em Ladder utilizada para o hidrômetro de pulso



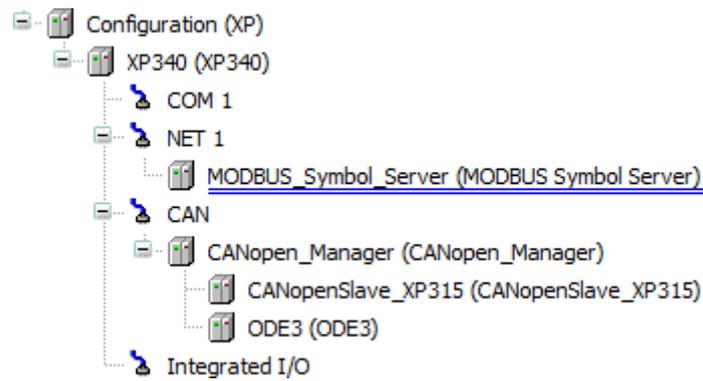
Fonte: De autoria própria

3.5 Utilização do Protocolo Modbus TCP

Sabendo como funciona o protocolo Modbus TCP, visto na Seção 2.1, fica simples manipular os dados para que sejam enviados para outro dispositivo. Neste caso, o protocolo Modbus TCP foi utilizado para realizar a comunicação entre a IHM e o CLP mestre da arquitetura de rede.

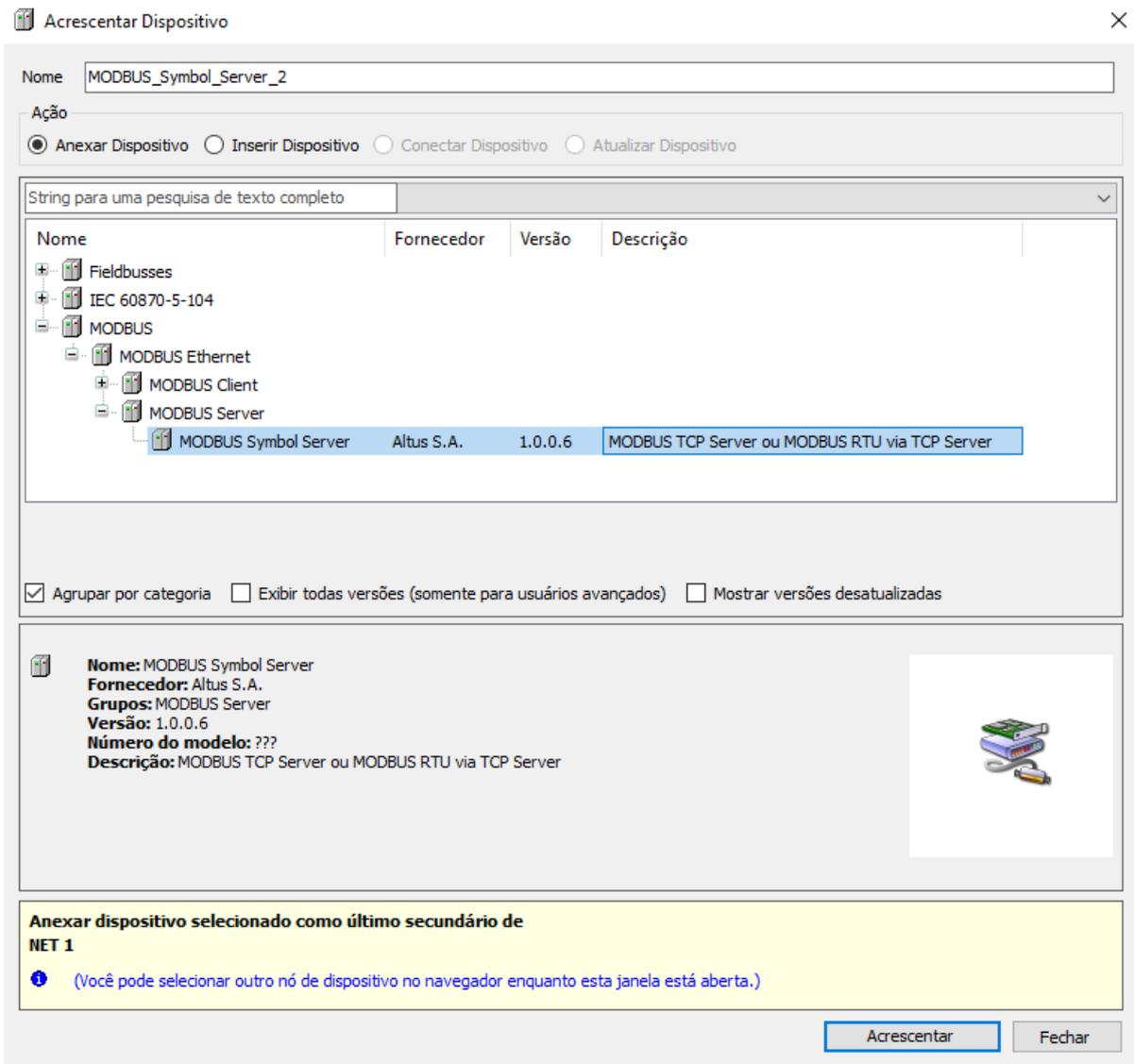
Inicialmente, antes de parametrizar os endereços desejados, é necessário adicionar um servidor Modbus à interface de programação do MasterTool IEC XE. É possível visualizar o servidor já inserido no *software* na Figura 22. Para alcançar esta etapa, basta clicar com o botão direito em 'NET 1' e selecionar 'Adicionar dispositivo'. Ao adicionar o dispositivo, basta especificar o protocolo a ser usado, que neste caso é o servidor Modbus. Pode ser visto a janela que é aberta para adicionar um dispositivo na Figura 23.

Figura 22: Configuração do servidor Modbus



Fonte: De autoria própria

Figura 23: Janela para adicionar o protocolo de comunicação no CLP



Fonte: De autoria própria

Após ser acrescentado o servidor Modbus, faz-se necessário conectar fisicamente o CLP à IHM, além de realizar as devidas configurações no *software* FvDesigner, que será comentado em breve. Dando sequência, basta endereçar os dados conforme a Tabela 1 (Seção 2.1). É possível visualizar na Figura 24 alguns dados endereçados para a troca de informação com a IHM. Também, é possível perceber que o modo de conexão do protocolo Modbus é o modo TCP.

Figura 24: Endereçamento de dados através do Modbus TCP

Configurações						
Porta TCP	502	Filtros...	Avançado...			
Modo de Conexão	TCP					
Mapeamentos						
Variável de Valor	Tipo do Dado	Endereço Inicial do Dado	Endereço Inicial Absoluto do Dado	Tamanho do Dado	Faixa de Dados	
AcionamentoSaida.M_Manutencao_VA_006	Coil	36	36	1	36..36	
AcionamentoSaida.M_Manutencao_VA_007	Coil	37	37	1	37..37	
AcionamentoSaida.M_Manutencao_VA_008	Coil	38	38	1	38..38	
AcionamentoSaida.M_Manutencao_VA_009	Coil	39	39	1	39..39	
AcionamentoSaida.M_Manutencao_BRC_001	Coil	40	40	1	40..40	
AcionamentoSaida.M_Manutencao_BDNAC_001	Coil	41	41	1	41..41	
AcionamentoSaida.M_Manutencao_BDNACH_001	Coil	42	42	1	42..42	
AcionamentoSaida.M_Manutencao_BDNAOCL_002	Coil	43	43	1	43..43	
AcionamentoSaida.M_Manutencao_BDNAOCL_003	Coil	44	44	1	44..44	
Button_Stop	Coil	45	45	1	45..45	
Medidores.Temp_Entrada	Holding R...	46	400046	2	46..47	
Medidores.Turb_Entrada	Holding R...	48	400048	2	48..49	
Medidores.Turb_Saida	Holding R...	50	400050	2	50..51	
Medidores.PH_Saida	Holding R...	52	400052	2	52..53	
Medidores.ERP_Saida	Holding R...	54	400054	2	54..55	
Medidores.Med_PT_001	Holding R...	56	400056	2	56..57	
Medidores.Med_PT_002	Holding R...	58	400058	2	58..59	

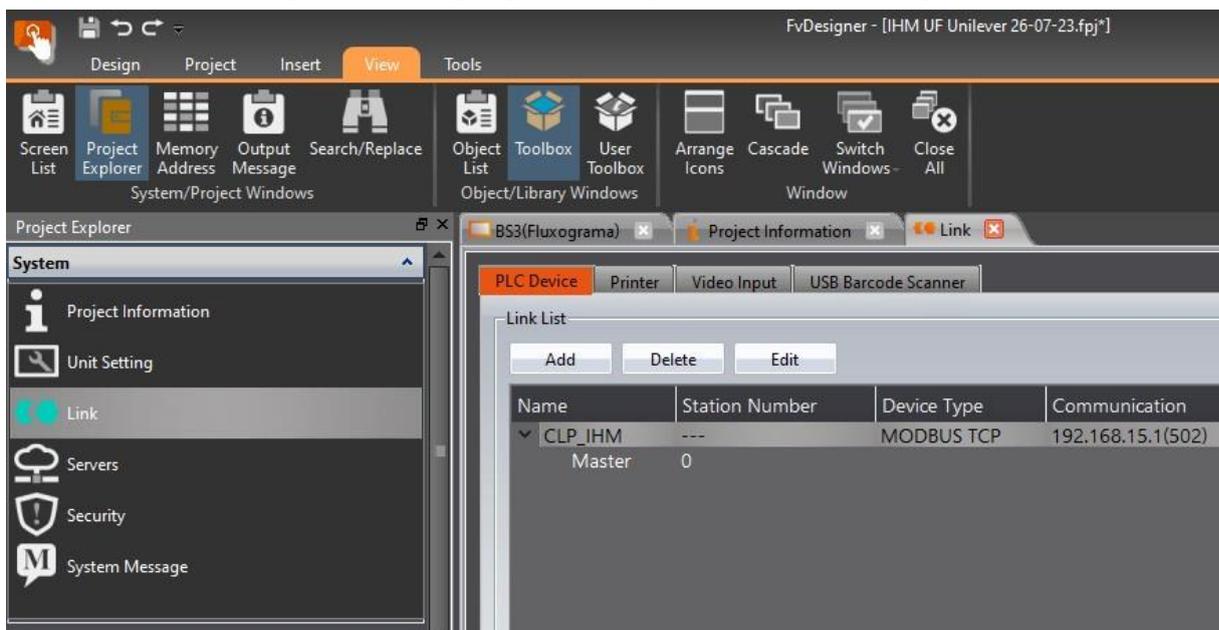
Fonte: De autoria própria

Como é perceptível na Figura 24, as variáveis *booleanas* estão sendo endereçadas como uma '*Coil*' e as variáveis do tipo Real, do tipo Inteiro e similares, estão sendo endereçadas como uma '*Holding Register*'. Dessa forma, todos os dados podem ser visualizados e escritos através da interface homem-máquina, sem que haja problema algum.

Um ponto importante a ser observado é o número do endereço atribuído a cada variável. Nenhum endereço deve sobrescrever outro, tornando cada um deles único. Os dados do tipo *holding register* possuem um endereço absoluto inicial a partir do 400001, enquanto os dados *booleanos* correspondem ao 1. Portanto, é possível selecionar o mesmo endereço para um dado do tipo *coil* e um do tipo *holding register*, uma vez que seus endereços absolutos são diferentes. No entanto, no projeto, optou-se por seguir a numeração padrão do endereçamento, independentemente do tipo de dado.

Já em relação a configuração da IHM, para a utilização do protocolo Modbus, basta ir no menu 'Project Explorer' e selecionar a opção 'Link', conforme a Figura 25.

Figura 25: Configuração da IHM para realizar comunicação

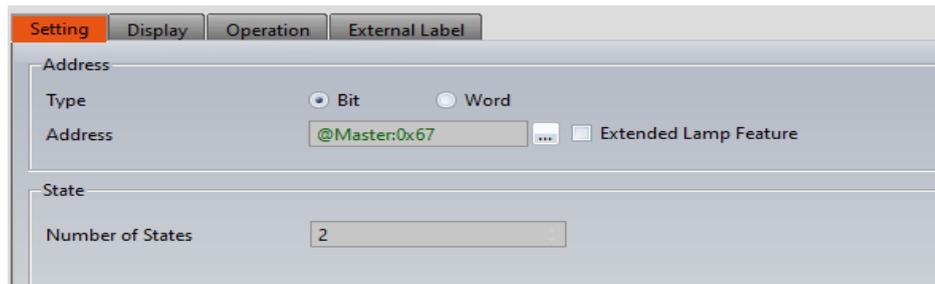


Fonte: De autoria própria

Em seguida, será aberta uma janela, também visível na Figura 25, a opção de adicionar o dispositivo para o estabelecimento da comunicação. Basta adicionar o CLP com seu respectivo endereço de IP e selecionar o protocolo Modbus TCP. Feito isso, ambos os dispositivos conseguem trocar dados entre si. Tudo isso aplicado dentro do *software* FvDesigner.

Para que a IHM receba os dados de uma variável em uma tela específica, basta fornecer ao objeto criado o endereço correspondente estabelecido no CLP. O endereço do objeto é visível quando acessado a sua aba de propriedades. Um exemplo pode ser visto na Figura 26.

Figura 26: Endereçamento de uma variável na IHM



Fonte: De autoria própria

Como mencionado na Fundamentação Teórica, o MasterTool IEC XE representa a função através do seu nome, como visto na Figura 27, enquanto o FvDesigner através da abreviação, como na imagem acima. No exemplo da Figura 29, percebe-se que está utilizando a função 'Coil' que está no endereço 67 (0X67).

Sabendo realizar essas parametrizações, é possível então transcrever qualquer variável do CLP para que ela possa ser lida ou escrita pela IHM. O projeto conta com um total aproximado de 205 tags adicionadas na troca de informação através do protocolo Modbus TCP.

3.6 Utilização do Protocolo CANopen

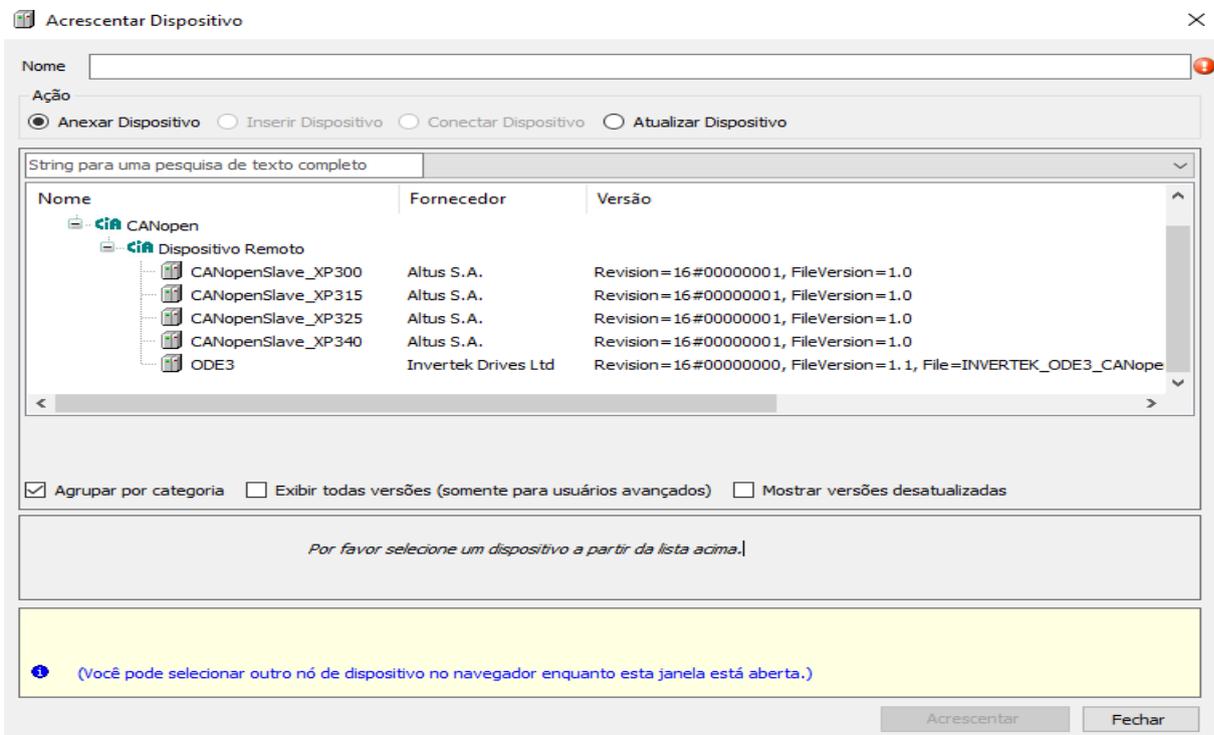
O protocolo CANopen foi utilizado no projeto para estabelecer a comunicação entre o inversor Optidrive ODE 3 e o CLP XP315 com o CLP XP340, que atua como mestre na topologia mestre-escravo. A implementação desses dispositivos no MasterTool IEC XE é relativamente simples. Basta clicar com o botão direito em 'CAN', localizado no menu lateral esquerdo, conforme mostrado na Figura 22, e adicionar o 'CANopen_Manager'. Em seguida, repetir o processo sobre o novo objeto adicionado

e adicionar o equipamento que irá trocar informações com o CLP, como ilustrado na Figura 27.

É importante ressaltar que o inversor Optidrive ODE 3 inicialmente não será exibido entre as opções de dispositivos. Para incluí-lo, é necessário baixar o arquivo no site da Altus e implementá-lo no *software* através da guia 'Ferramentas' no menu superior, acessando o repositório de dispositivos.

Com os equipamentos já adicionados ao projeto, é possível realizar algumas configurações básicas e importantes, como, por exemplo, ajustar o endereçamento do nó do dispositivo. Além disso, é apresentado o mapeamento das entradas e saídas de cada dispositivo, permitindo a utilização das variáveis em outras partes do projeto. Na Figura 28, é possível visualizar a localização de algumas entradas digitais do CLP XP315. Em relação ao inversor, algumas variáveis, como referência de velocidade, corrente atual, temperatura, status de falha, entre outras, são mostradas para serem utilizadas conforme desejado dentro do projeto.

Figura 27: Implementação de dispositivos na rede CAN



Fonte: De autoria própria

Figura 28: Mapeamento de entradas e saídas do CLP XP315

Variável	Mapeamento	Canal	Endereço	Tipo	Unidade	Descrição
SVA_NC_001		Bit0	%IX0.0	BOOL		I2_00_STATUS_VALV_FEED_CLOSED
SVA_NC_002		Bit1	%IX0.1	BOOL		I2_01_STATUS_VALV_FILTRATE_CLOSED
SVA_NC_003		Bit2	%IX0.2	BOOL		I2_02_STATUS_VALV_CONCENTRATE_CLOSED
SVA_NC_004		Bit3	%IX0.3	BOOL		I2_03_STATUS_VALV_DRAIN_CLOSED
SVA_NC_006		Bit4	%IX0.4	BOOL		I2_04_STATUS_VALV_DRAINFILTRATE_CLOSED
SVA_NC_007		Bit5	%IX0.5	BOOL		I2_05_STATUS_VALV_RC_CLOSED
SVA_NC_009		Bit6	%IX0.6	BOOL		I2_06_STATUS_VALV_DRAINCHIMICAL_CLOSED
SVA_NC_010		Bit7	%IX0.7	BOOL		I2_07_STATUS_VALV_DRAIN_NOCHIMICAL_CLOSED
PT_001		Digital_Inputs_2	%IB1	USINT		
		Analog_Inputs_1	%IW2	INT		

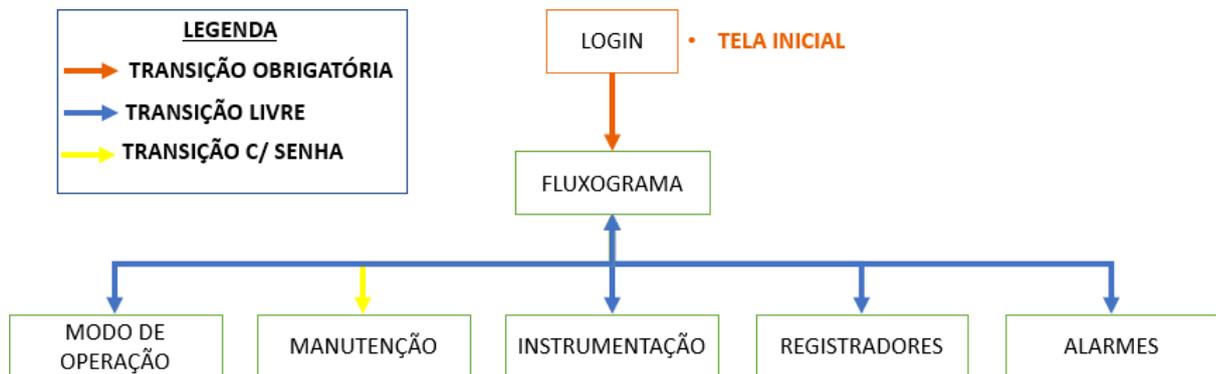
Fonte: De autoria própria

3.7 Criação das Telas da IHM

A IHM concentra-se no maior poder operacional do módulo de ultrafiltração de água deste projeto. Através dela, o operador consegue programar o modo de operação da máquina, ajustar *set points* de vazão, entre outras funcionalidades. Ela foi elaborada de forma a passar todas as informações necessárias do processo de maneira simples e objetiva, permitindo que o próprio operador a manuseie conforme desejado.

Inicialmente, é interessante analisar a Figura 29 a seguir, que mostra o fluxograma de navegação das telas criadas, para facilitar o entendimento posteriormente.

Figura 29: Fluxograma de navegação das telas da IHM



OBS.: TODOS OS BLOCOS VERDES CONSEGUEM SE DIRECIONAR À TELA INICIAL DE FORMA DIRETA!

Fonte: De autoria própria

Como é perceptível, o projeto conta com 7 telas distintas, sendo a tela de *login* a tela de inicialização da IHM (Figura 30). Ela exige que o usuário entre em sua conta para visualizar as demais telas, sendo direcionado, inicialmente, à tela de fluxograma do processo, conforme Figura 31. A partir da tela de fluxograma, o usuário pode retornar à tela de *login* ou navegar entre outras, como a tela de seleção do modo de operação do módulo (Figura 32), tela de visualização dos medidores (Figura 33), tela de visualização dos registradores (Figura 34), tela de visualização dos alarmes (Figura 35) e a tela de manutenção (Figura 36). Esta última sendo acessível apenas para usuários que efetuarem o *login* com uma conta de nível superior.

Há dois níveis de acesso neste projeto, sendo o nível básico direcionado aos operadores e o nível mais elevado destinado aos mantenedores e/ou supervisor(es). A diferença, como mencionado, é que o nível mais alto possibilita o acesso à tela de manutenção, permitindo realizar ajustes de parâmetros mais técnicos. Isso inclui o ajuste de *set points* de alguns pontos de pressão do módulo, ligar ou desligar válvulas e bombas a qualquer instante, resetar a contagem dos ciclos de filtração, reiniciar a IHM, alterar a senha dos usuários e acessar uma janela de configurações internas da própria interface homem-máquina.

A tela do fluxograma foi elaborada seguindo as instalações reais, com as localizações correspondentes dos equipamentos. Foram incluídas animações para tornar o processo mais intuitivo, como a mudança na cor do tubo quando está passando algum líquido por ele. Além disso, há animações que indicam em verde quando o equipamento está desligado e, em vermelho, quando ele está sendo acionado. Com exceção dos tanques, que possuem uma animação que permite tornar o objeto invisível, possibilitando que o operador perceba quando estão cheios ou necessitando de líquidos. Também, é possível visualizar os valores dos instrumentos de medição por meio de *numerics displays* localizados próximo à sua representação no fluxograma. Para concluir, o usuário tem a opção de navegar entre as telas ao tocar no botão 'Menu', que o leva a um *Pop Up*. Além disso, pode escolher entre operar a máquina no modo manual ou automático, bem como ligá-la ou desligá-la.

A tela da Figura 32 serve para o operador selecionar qual modo de operação a máquina irá adotar. Através dos ciclos, localizados na parte direita da tela, a própria máquina consegue percorrer todos os modos de operação automaticamente, sem necessidade de intervenção humana. Um ciclo corresponde à uma etapa de filtração

seguida de um *Air Scour*. Na parte inferior da tela, é possível visualizar alguns temporizadores, com contagem crescente para o modo *Filtration* e decrescente para os demais modos de operação. Embora visualmente o *display* pareça desorganizado, essa aparência decorre do fato de o projeto do FvDesigner não estar em simulação. Isso resulta na exibição de todos os objetos criados na tela de uma só vez. O mesmo aplica-se às demais figuras.

Os medidores são exibidos na tela de instrumentação do projeto. São mostrados todos em uma única tela para tornar mais prático a visualização dos mesmos. Como se pode perceber, ainda há alguns medidores que não foram mencionados até o momento, como a medição do potencial hidrogeniônico (PH) e da turbidez, tanto de entrada quanto de saída do módulo. Isso devido ao fato de ser uma ideia de instalação futura, tornando necessário apenas configurar no CLP. Também, é possível visualizar um botão com o nome 'Vazão' ainda nessa tela, onde direciona o usuário, ao ser pressionado, a um *Pop Up* para a visualização do acumulado de água e da vazão atual de cada um dos hidrômetros.

A tela dos registradores foi elaborada com a finalidade de registrar a data e a hora em que o módulo entra em algum dos modos de limpeza, proporcionando um controle mais eficiente do processo.

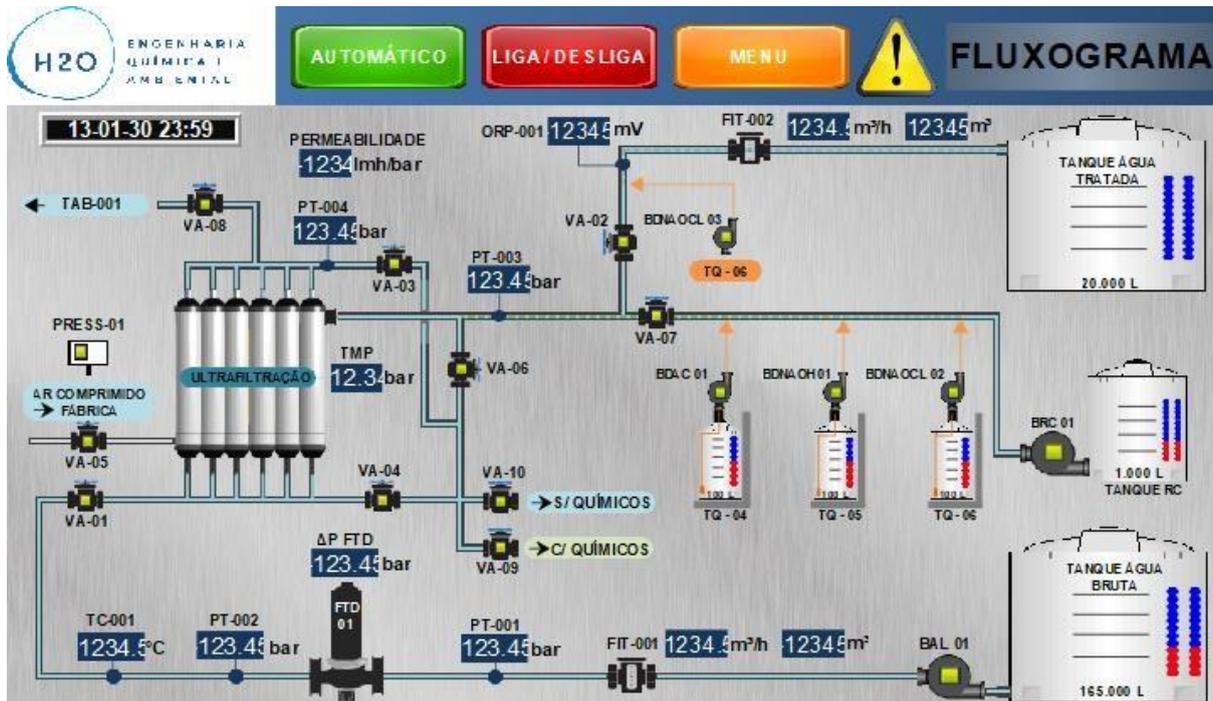
Por fim, há a tela de alarmes, que exibe o histórico completo dos alarmes ocorridos, destacando aqueles que estão ativos com prioridades no topo, caso estejam em falha no momento. Nesta tela, é possível realizar diversas ações, como limpar o histórico de alarmes, filtrá-los para uma rápida localização, executar *reset* para retornar a máquina ao modo de operação e silenciar o alarme sonoro por 60 segundos.

Figura 30: Tela de Login - IHM



Fonte: De autoria própria

Figura 31: Tela de Fluxograma - IHM



Fonte: De autoria própria

Figura 32: Tela de Operação - IHM



Fonte: De autoria própria

Figura 33: Tela de Instrumentação - IHM



Fonte: De autoria própria

Figura 34: Tela de Registradores - IHM

REGISTRO DE CICLOS

Data	Hora	MC1	MC2	MC3	MC1+2	RC
01/04/2010	13:20:30	9999	9999	9999	9999	9999
01/04/2010	13:20:30	9999	9999	9999	9999	9999
01/04/2010	13:20:30	9999	9999	9999	9999	9999

Fonte: De autoria própria

Figura 35: Tela de Alarmes - IHM

ALARMES

Data	Acionado	Mensagem
dd/M...	hh:mm:ss	...

LIMPAR
FILTRAR
RESET
MUTE

-01-30 23:

Fonte: De autoria própria

Figura 36: Tela de Manutenção - IHM



Fonte: De autoria própria

3.8 Web Server

Com o projeto já em funcionamento, deu-se início à elaboração do *Web Server*. Ele tem como finalidade exibir as principais informações do projeto para os dispositivos conectados à rede em que o CLP está inserido, de forma que facilite a monitorização dos processos sem a necessidade de estar fisicamente próximo ao equipamento. Esse sistema de análise remota traz praticidade ao cliente, principalmente à equipe da gestão, possibilitando tomadas de decisões mais eficientes.

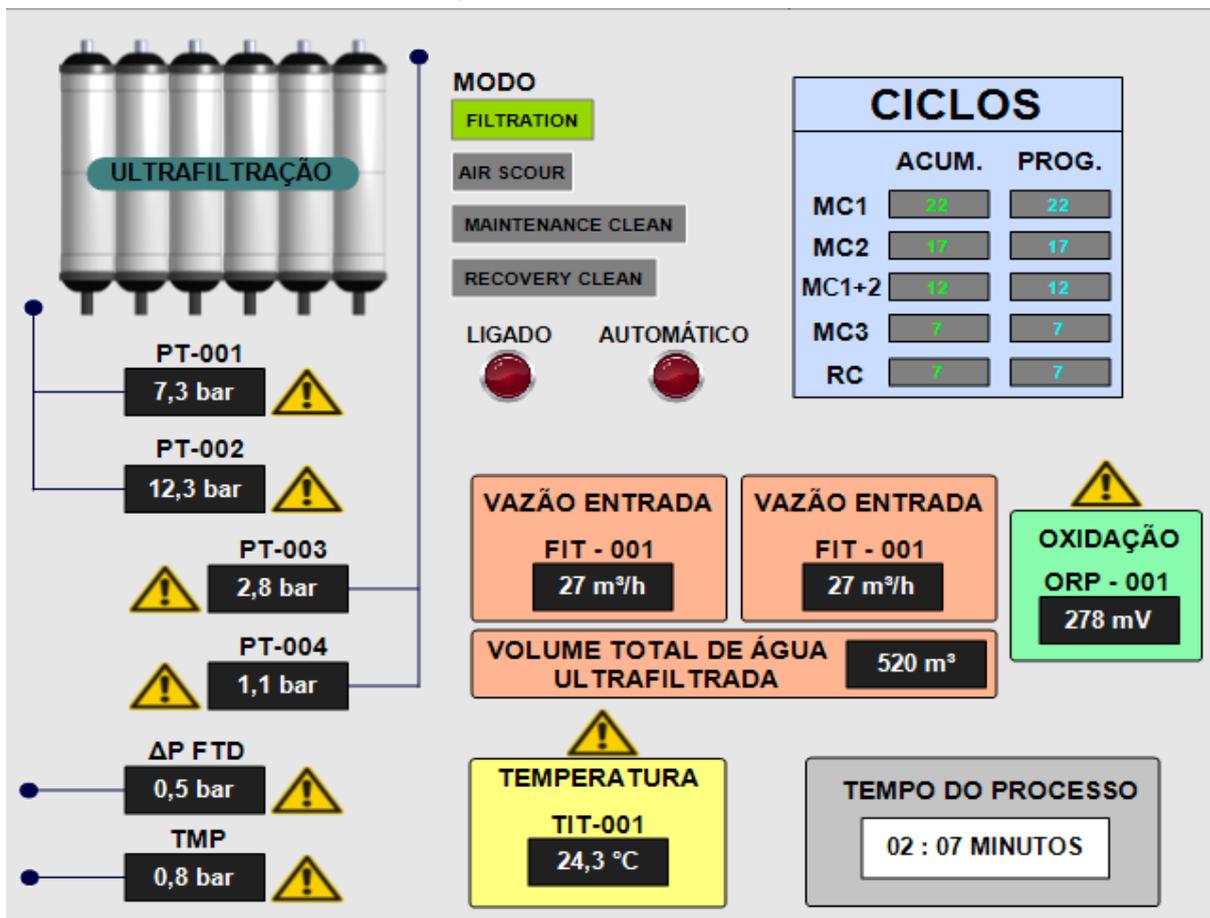
Como mencionado, o CLP XP340 da Altus já vem com a funcionalidade de criar um *Web Server* através do MasterTool IEC XE de maneira prática, semelhante a programação de uma IHM. Para adicioná-lo dentro do programa, basta clicar com o botão direito em '*Application*', que está no menu lateral esquerdo conforme Figura 2, e adicionar um novo objeto de Gerenciador de Visualização (*Visualization Manager*), para que seja feito as configurações básicas do *Web Server*, e adicionar um novo objeto para a criação das telas (*Visualization*). Ao inserir o objeto de visualização e acessá-lo, abre-se a tela para realizar o designer desejado.

Seguindo esses passos, foi então configurada e elaborada a única tela que há no servidor. A pedido do cliente, o *Web Server* ficou destinado apenas para ler os dados, sendo impedido, dessa maneira, de ser implementado alguma *tag* com a funcionalidade de escrita. É possível visualizar na Figura 37 a tela elaborada para o monitoramento do tratamento de água através dessa ferramenta de programação.

Os dados são meramente ilustrativos uma vez que foi obtido diretamente do MasterTool, sem estar com o CLP *online*. No entanto, é possível perceber que através dessa tela é possível obter todas as principais informações do processo.

Para acessá-la através de algum equipamento com acesso à rede, é necessário abrir algum navegador de *internet* e inserir, na barra de pesquisa, o IP do CLP mestre, a porta do *Web Server* (neste caso, 8080), o nome da aplicação inserida dentro do *software* e a extensão 'htm'. Um exemplo pode ser visto a seguir: 192.168.1.10:8080/WebServer.htm.

Figura 37: Tela do *Web Server*



Fonte: De autoria própria

3.9 Resultados

Antes de entregar o projeto, foram realizados diversos tipos de testes para garantir que não haveria falhas no código desenvolvido. Procedimentos de geração de falhas foram executados para verificar se a máquina seguiria os padrões desejados. Além disso, foi realizado um teste em todos os modos de operação do módulo. A tela de IHM foi rigorosamente testada para assegurar que nenhum dado importante estivesse faltando e, principalmente, para garantir que não impactassem no funcionamento do processo.

Alguns registros do processo foram feitos. Como pode ser visto na Figura 38, a IHM na tela de operação.

Figura 38: Registro de funcionamento da tela de operação



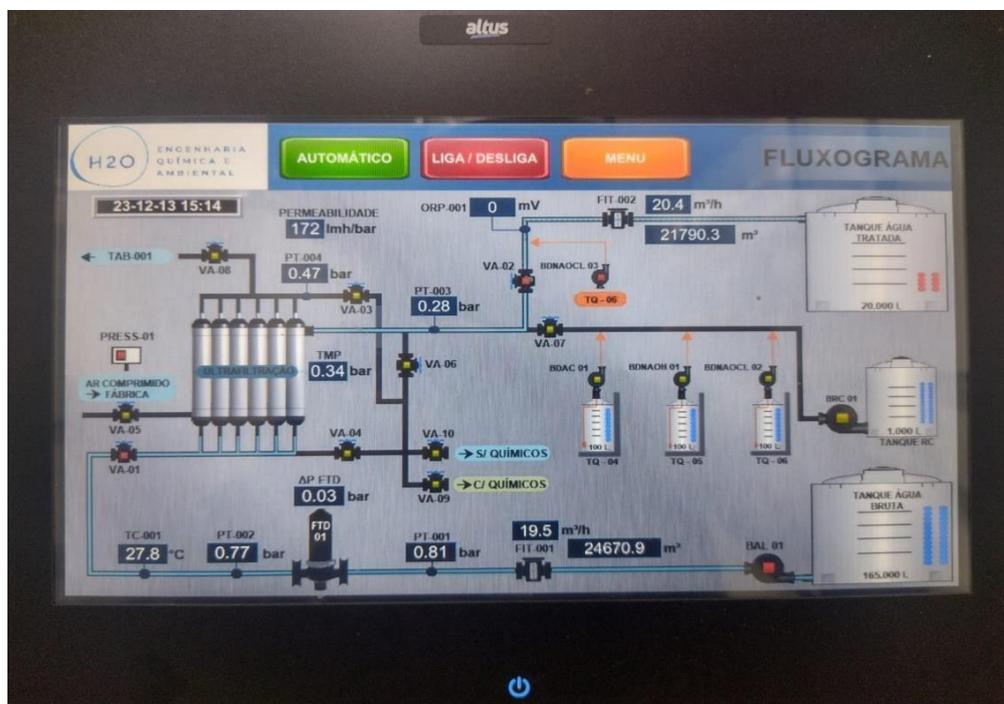
Fonte: De autoria própria

Como é possível perceber, o módulo de ultrafiltração está operando no modo *Air Scour*, faltando um pouco mais de três minutos para retornar ao modo *Filtration*. Além

disso, é possível verificar alguns ciclos já concluídos e os programados, conforme mostrado na lateral direita da imagem.

Outro registro pode ser observado na Figura 39, onde a IHM exibe o fluxograma. É possível observar o volume acumulado de água que passou pelos hidrômetros, bem como sua vazão correspondente. Também, é possível visualizar dados sobre a pressão do módulo e a circulação de água através da animação das tubulações, juntamente com outras informações presentes na tela.

Figura 39: Registro de funcionamento da tela de fluxograma



Fonte: De autoria própria

Na Figura 40, é possível ver o registro da IHM quando a mesma se encontra na tela de alarmes. Ela exibe alguns alarmes que foram ativados, como o nível baixo dos tanques de químicos. Essa tela é bastante importante, uma vez que armazena as falhas ocorridas durante o processo de ultrafiltração ou limpeza do módulo.

Figura 40: Registro de funcionamento da tela de alarmes

Data	Acionado	Mensagem
12/12/23	16:53:17	BAIXA CONCENTRAÇÃO DE CLORO
12/12/23	16:55:48	NÍVEL BAIXO - TQ ÁCIDO CÍTRICO
12/12/23	16:55:01	NÍVEL BAIXO - TQ RC
12/12/23	16:54:32	NÍVEL BAIXO - TQ ÁCIDO CÍTRICO
12/12/23	16:54:10	NÍVEL BAIXO - TQ HIDRÓXIDO DE SÓDIO
12/12/23	16:54:03	NÍVEL BAIXO - TQ HIDRÓXIDO DE SÓDIO
12/12/23	16:53:48	NÍVEL BAIXO - TQ HIDRÓXIDO DE SÓDIO
12/12/23	16:52:19	NÍVEL BAIXO - TQ HIDRÓXIDO DE SÓDIO
12/12/23	16:52:36	NÍVEL BAIXO - TQ HIPOCLORITO DE SÓDIO
12/12/23	16:52:19	NÍVEL BAIXO - TQ HIPOCLORITO DE SÓDIO
12/12/23	15:16:19	NÍVEL BAIXO - TQ ÁCIDO CÍTRICO
12/12/23	15:14:24	NÍVEL BAIXO - TQ HIDRÓXIDO DE SÓDIO
12/12/23	15:14:37	NÍVEL BAIXO - TQ ÁCIDO CÍTRICO
12/12/23	14:21:47	NÍVEL BAIXO - TQ HIDRÓXIDO DE SÓDIO
12/12/23	14:19:46	NÍVEL BAIXO - TQ ÁCIDO CÍTRICO

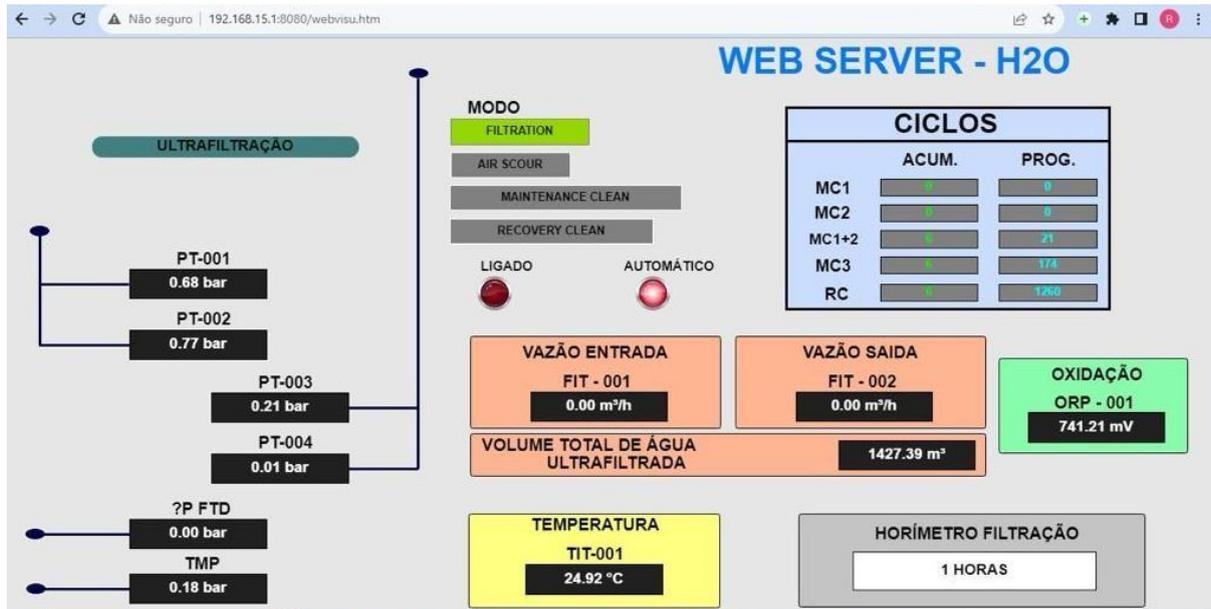
Fonte: De autoria própria

Em relação ao *Web-Server*, ele pode ser visualizado na Figura 41, sendo acessado por um computador conectado à mesma rede dos equipamentos de controle. Por meio desse acesso, é possível ver as informações mais relevantes do módulo de ultrafiltração, bem como alguns alarmes específicos. Durante este teste, a imagem do módulo não foi carregada, como mostrado na Figura 37. No entanto, nos demais dispositivos que acessavam o servidor, permaneceu com a tela de acordo com o projetado.

Em relação aos equipamentos físicos, os quais todos se encontram dentro de um container, é possível visualizar alguns deles na Figura 42 e na Figura 43, registrados durante o processo de montagem. Uma mostra o módulo com as seis cápsulas *Hydracap*, enquanto a outra apresenta os três tanques de químicos juntamente com suas respectivas bombas.

Ao final dos testes, ainda dentro do ambiente da H2O Engenharia Química e Ambiental, o projeto foi levado para a Unilever, onde permaneceu em testes por um bom tempo. Durante esse período, a água passou por diversas análises para comprovar a eficácia do tratamento.

Figura 41: Registro de funcionamento do Web Server



Fonte: De autoria própria

Figura 42: Registro da membrana de ultrafiltração



Fonte: De autoria própria

Figura 43: Registro dos tanques de químicos



Fonte: De autoria própria

4 CONCLUSÕES E PROPOSTAS DE CONTINUIDADE

O projeto realizado demandou muitos dias de trabalho, contudo, foi entregue dentro do prazo estabelecido pelo cliente. A experiência adquirida foi de grande importância para a formação profissional, dando a oportunidade não apenas de aplicar o conhecimento teórico na prática, mas também de escolher a estratégia de programação desejada. Elaborar um projeto do zero e acompanhar seu desenvolvimento é uma experiência bastante significativa.

Após a entrega, foi necessário aguardar um certo tempo com todo o módulo em teste para confirmar a eficácia da água originada da ultrafiltração. Com essa confirmação, a água tratada foi liberada para utilização nos processos industriais da Unilever – destinatária do projeto. Através desse módulo de ultrafiltração, a empresa conseguiu maximizar a utilização de seus recursos hídricos, reconhecendo a importância de mantê-los em boa quantidade e qualidade para sustentar as possíveis expansões.

Tanto a H2O Engenharia Química e Ambiental quanto a Unilever expressaram sua aprovação ao projeto. Este êxito serve como um estímulo para que outras empresas considerem a possibilidade de investimento em sistemas avançados de tratamento de água.

Como sugestão para continuidade do projeto, é possível destacar a expansão do *Web Server* para torna-lo acessível de qualquer lugar, não se limitando apenas à rede da fábrica. Assim também, como implementar a tecnologia do *Node-Red* para enviar mensagens aos celulares dos mantenedores e supervisores quando a máquina apresentar falhas, visando reduzir o tempo de manutenção.

REFERÊNCIAS

- AGBOOLA, Oluranti; MAREE, Jannie; MBAYA, Richard. Characterization and Performance of Nanofiltration Membranes. **CORE**, 2014. Disponível em: <<https://core.ac.uk/download/pdf/83592845.pdf>>.
- BISCHOFF, Wesley. G1. **Relatório da ONU Aponta Crise Global por Escassez de Água**, 2023. Disponível em: <<https://g1.globo.com/meio-ambiente/noticia/2023/03/21/relatorio-da-onu-aponta-risco-de-crise-global-por-escassez-de-agua.ghtml>>.
- CAN Bus: Conceituação. **alexag**. Disponível em: <[https://www.can-cia.org/canopen/](http://www.alexag.com.br/CAN_Bus_Parte_2.html#:~:text=O%20CAN%20%C3%A9%20um%20protocolo,de%20tempo%20conhecidos%20e%20regulares).>>.</p>
<p>CANOPEN - A Rede Embarcada Padronizada. can-cia. Disponível em: <.
- CAPILLARY Ultrafiltration Module HYDRAcap® MAX 60 Performance. **membranes**. Disponível em: <<https://membranes.com/wp-content/uploads/Documents/Element-Specification-Sheets/UF-MF/HYDRAcap-MAX/HYDRAcap-MAX-60.pdf>>. Acesso em: 29 Setembro 2023.
- Características Técnicas XP3xx. [S.l.], p. 20.
- FERRARO, Ronaldo J. S. lyceumonline. **Universidade São Francisco**, 2008. Disponível em: <<https://lyceumonline.usf.edu.br/salavirtual/documentos/1524.pdf>>.
- FUSATI. O Que é Ultrafiltração e Seus Benefícios. **FUSATI**, 2020. Disponível em: <[HOJE, Jornal. G1. **Brasil Desperdiça 40,1% de Toda Água Potável Produzida**, 2022. Disponível em: <<https://g1.globo.com/jornal-hoje/noticia/2022/06/01/brasil-desperdica-401percent-de-toda-agua-potavel-produzida-diz-pesquisa.ghtml>>.

HYDRAcap MAX Technical Manual. \[S.l.\], p. 49. 2019.

INVERTEK Optidrive E3 IP20-IP66 Manual User Guider. Disponível em: <<https://inverterdrive.com/file/Invertek-Optidrive-E3-Manual>>. Acesso em: 28 Setembro 2023.

IONOS. O que é multicast?, 2023. Disponível em: <<https://www.ionos.com/digitalguide/server/know-how/multicast/>>.

MasterTool IEC XE MT8500. \[S.l.\], p. 8.

MODBUS TCP, Manual do Usuário. **servicedrive**, 2013. Disponível em: <<https://www.servicedrive.com.br/wp-content/uploads/Manual-do-Usu%C3%A1rio-Modbus-TCP-PLC300-Servicedrive-19-3012-6360.pdf>>.

MODELLI, Laís. UOL. **Perda de Água é Alerta Para as Mudanças Climáticas**, p. 1-86, Dec. 2023. Disponível em: <<https://noticias.uol.com.br/ultimas-noticias/deutschewelle/2023/03/22/perda-de-agua-e-alerta-para-as-mudancas-climaticas.htm>>.](https://www.fusati.com.br/o-que-e-ultrafiltracao-e-seus-beneficios/#:~:text=Ultrafiltra%C3%A7%C3%A3o%20%C3%A9%20uma%20variedade%20de,s%C3%B3lidos%20s%C3%A3o%20capturados%20e%20descartados.>>.</p>
<p>FvDesigner Manual de Utilização. [S.l.], p. 497. 2021.</p>
<p>GUIMARÃES, Alexandre D. A.; SARAIVA, Antônio M. O Protocolo CAN: Entendendo e Implementando uma Rede de Comunicação Serial de Dados Baseada no Barramento)

NASCIMENTO, João M. A. D.; LUCENA, Pedro B. D. Protocolo Modbus, 2003. Disponível em: <https://www.dca.ufrn.br/~affonso/FTP/DCA447/trabalho3/trabalho3_13.pdf>.

NETO, Rafael C. **Máquina de Estados e Implementação de Algoritmos de Controle a partir do Grafcet**. [S.l.]. 2021.

PROTOCOLO Modbus (RTU / TCP & ASCII), 2022. Disponível em: <<https://www.envisia.com.br/2022/08/02/protocolo-modbus-rtu-tcp-ascii/>>.

SENA, Hanilton. Técnicas de Programação - Máquinas de Estados e Ladder, 2013. Disponível em: <<https://pt.slideshare.net/hjssena/tecnicas-de-programao-maquinas-de-estados-e-ladder>>.

Terminais de Operação Série P2. [S.l.], p. 20.