Elcio Rezende Junior

Simulação Aplicada a uma Linha de Produção de Pães Automatizada

UBERLÂNDIA

2021

Elcio Rezende Junior

Simulação Aplicada a uma Linha de Produção de Pães Automatizada

Trabalho de Conclusão de Curso da Engenharia de Controle e Automação da Universidade Federal de Uberlândia - UFU - Câmpus Santa Mônica, como requisito para a obtenção do título de Graduação em Engenharia de Controle e Automação

Universidade Federal de Uberlândia – UFU

Faculdade de Engenharia Elétrica

Orientador: Prof. Dr. Renato Santos Carrijo

Uberlândia

2021

Rezende, Elcio

Simulação Aplicada a uma Linha de Produção de Pães Automatizada/ **Elcio Rezende Junior**. – **UBERLÂNDIA**, **2021**.

Orientador: Prof. Dr. Renato Santos Carrijo

Trabalho de Conclusão de Curso – Universidade Federal de Uberlândia – UFU Faculdade de Engenharia Elétrica . **2021**.
Inclui bibliografia.

1. Automação 2. CLP 3. SCADA 4.OPC I.Prof. Dr. Renato Santos Carrijo. II. Universidade Federal de Uberlândia. III. Faculdade de Engenharia Elétrica. IV. Engenharia de Controle e Automação.

Dedico este trabalho primeiramente a Deus por estar sempre presente em minha vida, me proporcionando força, saúde e fé para que eu possa continuar me esforçando todos os dias.

Aos meus familiares, pelo apoio incondicional e esforço para me ajudar em tudo que foi preciso para eu chegar até aqui.
Em especial ao meu avô paterno Aderbal (in memoriam), minhas avós Aparecida e Romilda, meu pai Elcio, minha mãe Celma e minha irmã Suelen, por sempre acreditarem muito em mim desde criança e me fazerem crer que com minha sabedoria e esforço eu poderia chegar em qualquer lugar que almejasse.

Agradecimentos

Aos meus pais Elcio Rezende e Celma Delfina da Silva, minha irmã Suelen Rezende, minhas avós e demais familiares por todo o apoio, confiança, e incentivo durante toda a graduação, por ajudarem e me darem forças em todos os momentos de dificuldades mas também por celebrarem todas minhas conquistas, sem vocês nada disso seria possível.

A minha namorada Aline Mendonça por sempre ter acreditado em mim, por toda ajuda durante o curso, isso foi essêncial, fazer esta graduação ao seu lado, literalmente junto com você, foi uma honra que nem todos podem ter.

Aos meus amigos, que foram muito mais que colegas de curso, por estarem sempre ao meu lado, por todas as várias horas de estudo seja durante o dia, noite, finais de semana ou feriados, junto a vocês o desafio se tornou mais simples.

A todos os professores e funcionários da Faculdade de Engenhearia Elétrica, principalmente aos que fazem parte do curso de Engenharia de Controle e Automação, por todo conhecimento e aprendizado de altíssima qualidade que proporcionaram. Em especial aos professores Josué Silva de Morais e Renato Santos Carrijo, ambos coordenadores do curso durante o momento de minha graduação, por sempre se dedicarem e proporcionarem o melhor aos seus alunos. Ao professor Renato Santos Carrijo ainda meus sinceros agradecimentos por ter sido meu orientador e ter desempenhado com muita dedicação esta função, e ter dado todo o suporte necessário durante o desenvolvimento do trabalho.

"Olho nenhum viu, ouvido
nenhum ouviu, mente nenhuma imaginou o que
Deus preparou para aqueles que o amam";

(1 Coríntios 2:9)

Resumo

A automação tem se tornado algo cada vez mais indispensável. A busca por processos mais rápidos, eficazes e de qualidade, de forma geral são o que impulsionam o crescimento da área. Utilizando programações em linguagens SFC e ST através do software CODESYS foi possível criar uma lógica capaz de efetuar todos os passos necessários que simulam uma linha de produção de pães automatizada desde o primeiro acionamento até a parte final de visualização do processo. Foi criado ainda um sistema de supervisão por meio do software Elipse SCADA que comunica via OPC com o CLP virtual presente no CODESYS o que possibilita testes mais eficazes e próximos da realidade. Dentro do cenário proposto obteve-se resultados satisfatórios das ferramentas utilizadas na simulação garantindo a proximidade e fidelidade do programa ao sistema proposto.

Abstract

Automation has become increasingly indispensable. The search for faster, more efficient and quality processes, in general, are what drive the area's growth. Using programming in SFC and ST languages through CODESYS software, it was possible to create a logic capable of performing all the necessary steps that simulate an automated bread production line, from the first activation to the final part of the process visualization. A supervision system was also created using Elipse SCADA software, which communicates via OPC with the virtual PLC present in CODESYS, which allows for more effective tests that are closer to reality. Within the proposed scenario, satisfactory results were obtained from the tools used in the simulation, ensuring the proximity and fidelity of the program to the proposed system.

Lista de ilustrações

Figura 1 - Pirâmide da Automação 18
Figura 2 - Exemplo de um CLP Moderno 19
Figura 3 - Especificação do Ciclo de Scan 21
Figura 4 - Exemplo de Representação SFC 22
Figura 5 - Exemplo de uma IHM 24
Figura 6 - Exemplo de uma sala de supervisão.....24
Figura 7 - Acesso a dados de processos com protocolo OPC.....26
Figura 8 - Compactação dos protocolos clássicos para o protocolo UA.....27
Figura 9 - Variáveis declaradas em SFC feita no CODESYS.....29
Figura 10 - Parte 1: Programação em SFC feita no CODESYS.....30
Figura 11 – Parte 2: Programação em ST feita no CODESYS 30
Figura 12 – Parte 3: Programação em ST feita no CODESYS 31
Figura 13 – Parte 4: Programação em ST feita no CODESYS 31
Figura 14 – Parte 5: Programação em ST feita no CODESYS 32
Figura 15- Declaração das variáveis em ST feita no CODESYS.....33
Figura 16 - Parte 1: Programação em ST feita no CODESYS.....34
Figura 17 - Parte 2: Programação em ST feita no CODESYS.....34
Figura 18 - Parte 3: Programação em ST feita no CODESYS.....35
Figura 19 - Parte 4: Programação em ST feita no CODESYS.....35
Figura 20 - Painel de controle feito no CODESYS 36
Figura 21 - Visualização do início do processo de simulação 36
Figura 22 - Visualização do fim do processo de simulação 36
Figura 23 - Comunicação do CLP com o computador local 37
Figura 24 - Tela de configuração do servidor OPC no Elipse SCADA 38
Figura 25 - Tela de configuração das tags selecionadas no CODESYS 39
Figura 26 - Tela de configuração das tags selecionadas no Elipse SCADA 40
Figura 27 - Tela inicial do sistema supervisório 41
Figura 28 - Menu de opções do sistema supervisório 41
Figura 29 - Tela de monitoramento do sistema supervisório 42
Figura 30 - Tela de alarmes do sistema supervisório 42
Figura 31 - Gráfico de tendencias do sistema supervisório 43
Figura 32 - Valores “setados” direto no CLP virtual do CODESYS 46

Figura 33 – Valores “setados” na tela de visualização do CODESYS 46
Figura 34 – Valores “setados” na tela de supervisão do Elipse SCADA 47
Figura 35 – Exemplificação do sistema de acesso 48
Figura 36 – Tela de alarmes do sistema supervisório 49
Figura 37 – Gráfico de tendencias do sistema supervisório 50

Lista de abreviaturas e siglas

A&E	*Alarms and Events*
CD	*Complex Data*
CLP	*Controlador Lógico Programável*
DA	*Data Access*
DX	*Data Exchange*
DDE	*Dynamic Data Exchange*
EPROM	*Erasable Programmable Read-Only Memory*
E2PROM	*Electrically-Erasable Programmable Read-Only Memory*
FBD	*Function Block Diagram*
GRAFCET	*Graphic de Commande Etape-Transition*
HDA	*Historical Data Access*
I/O	*Input/Output*
IEC	*International Electrotechnical Comittee*
IHM	*Inteface Homem Máquina*
IL	*Instruction List*
LD	*Ladder*
MFG	*Mark Flow Graph*
MTU	*Master Terminal Unit)*
OLE	*Object Linking and Embedding*
OPC	*Open Platform Communications (OLE For Process Control)*
PIB	*Produto Interno Bruto*
PLC	*Programmable Logic Controller*
RAM	*Random Access Memory*
RS-232	*Recommended Standard 232*
RS-485	*Recommended Standard 485*
RTU	Remote Terminal Unit
SCADA	*Supervisory Control and Data Aquisition*
SFC	*Sequential Function Chart*
ST	*Strutured Text*
UA	*Unified Architecture*
Xi	*Express Interface*
XML-DA	*eXtensible Markup Language Data Access*

Sumário

Agradecimentos 6

Resumo 8

Abstract 9

Lista de ilustrações 10

Lista de abreviaturas esiglas 12

1. Introdução 14

1.1. Justificativa 16

1.2. Objetivos 16

2. Referencial Teórico 17

2.1. Automação 17

2.2. Controlador Lógico Programável 19

2.3. Sistema de Supervisão e Aquisição de Dados 23

2.4. O Padrão OPC 26

3. Metodologia 28

3.1. Simulação do CLP 28

3.2. Comunicação com o sistema de supervisão 37

3.3. Configuração do Sistema Supervisório 39

4. Resultados e Discussões 45

5. Conclusão 52

Referências 54

1. Introdução

A automação se tornou essencial nos últimos anos, tanto para processos industriais quanto para aplicações menores. Segundo (ROSÁRIO, 2012) o sistema de automação se comporta como um operador humano que pensa e executa a melhor ação possível com base nas informações recebidas. Visando crescimento, rapidez e larga escala de produção, essa técnica tem sido cada vez mais bem vista nos mais diversos ambientes.

De acordo com (VIANA, 2019) a indústria de alimentos possui extrema importância social e econômica, uma vez que gera grande impacto tanto no PIB quanto na geração de empregos de um país. Desta forma se caracteriza como sendo uma das mais importantes no setor industrial, uma vez que é indispensável a produção do mesmo, e também vale ressaltar que essa importância é presente não somente em indústrias, a produção de alimentos é muito ativa também em pequenos empreendimentos.

(LAMB, 2015) Destaca vantagens de se usar a automação, e algumas delas são: operadores humanos com tarefas monótonas ou em ambientes perigosos podem ser substituidos, a produção é mais rápida e os custos menores, rigoroso padrão de produção, além de atuar como catalisador na economia. Visando essa utilidade e versatilidade, a grandeza de se pesquisar sobre automação em um lugar tão importante, obteve-se a proposta de um estudo sobre uma linha de produção de pães automatizada.

Para isso, será utilizado o software CODESYS, um simulador de CLP (Controlador Lógico Programável), que atráves da sua interface de programação representa fielmente as funcionalidades do mesmo.

De acordo com (RIBEIRO, 2001) o controlador lógico programável é um equipamento eletrônico, digital, que pode realizar funções como controlar um processo ou uma máquina, ser programado facilmente e também armazenar esse programa. De uma maneira geral, o CLP é um equipamento que se assemelha a um computador (hardware) onde é possível inserir um programa (software) para controlar e monitorar cargas (dispositivos de saídas) de acordo com parâmetros enviados ao CLP (dispositivos de entradas). O programa desenvolvido para um CLP é totalmente personalizável, composto por uma série de instruções ou funções específicas como lógica, sequenciamento, temporização, contagem e aritmética, o que o torna o CLP um equipamento muito dinâmico, que pode ser usado em qualquer processo automático de acionamento e ou monitoramento de máquinas e processos.

Neste caso a linguagem de programação utilizada foi o SFC (Sequential Function Chart) pois é feita em um formato gráfico e utilizada principalmente em processos sequenciais proporcionando mais margem para o manuseio e desenvolvimento do código.

(ROSÁRIO, 2012) Destaca em sua obra a importância de um sistem de supervisão, segundo ele esses sistemas garantem a monitoração e controle do processo de forma mais eficiente, disponibilizando em tempo real informações atualizadas sobre o que acontece na planta através de gráficos e relatorios fazendo com que operadores tomem a melhor decisão. Então para a parte de monitoramento e acompanhamento do processo utilizaremos tanto a parte de visualização presente no CODESYS quanto um Sistema de Supervisão e Aquisição de Dados, mais conhecido como SCADA (abreviação do termo em inglês Supervisory Control and Data Aquisition) que é responsável por toda parte de monitoramento das váriaveis de processo. Para isso será utilizado o software Elipse SCADA, que simula um sistema supervisório coletando e transmitindo em tempo real todas as váriaveis e disposistivos do sistema de controle.

Os dois softwares estarão conectados e exercendo a comunicação via OPC DA, que é um padrão de comunicação cliente-servidor responsável por levar os dados e informações do CLP exercido pelo CODESYS, para o sistema SCADA exercido pelo Elipse.

A combinação de todos os itens citados nos possibilita demonstrar o poder da automação através de uma simulação para uma possível implementação da linha de produção automatizada de pães. Uma solução, que busca a automatização do processo, tendo em vista menos dependência da mão de obra, a padronização e regularidade do produto, além da maior produção em um menor intervalo de tempo, tendo também mais segurança uma vez que é possível a total visualização e acompanhamento do processo através do sistema supervisório.

1.1. Justificativa

A automação de um sistema, seja em qualquer seguimento garante maior funcionalidade e abre caminho para que a aplicação se torne algo muito mais grandioso. O uso dessa prática se tornou indispensável, porém devido a barreiras como falta de conhecimento ou a complexidade de alguns sistemas ela nem sempre é utilizada.

Ciente disso a presente pesquisa visa mostrar por meio de um estudo de caso a implementação da automação em um ambiente muito importante, a indústria alimentícia, porém fazendo isso por meio de uma simulação, uma forma compacta e flexível, utilizando ferramentas didáticas, acessíveis e consolidadas. Monstrando como esse sitema funciona, podendo ser utilizado em grandes indústrias, em pequenos empreendimentos ou até mesmo como uma forma didática em estudos e pesquisas sobre automação industrial.

Foi pensando exatamente em demonstrar virtudes da automação, como flexibilização, segurança, confiabilidade, inovação, eficiência, enfim a simplificação do processo como um todo, que o trabalho foi desenvolvido de forma compreensível e didática e o tema escolhido foi sobre implementar a automação em uma linha de produção de pães.

1.2. Objetivos

O presente trabalho tem como objetivo geral desenvolver um simulador aplicado a uma linha de produção de pães, através de uma lógica capaz de efetuar todos os passos necessários. Solução que busca demonstrar o funcionamento da automação desde o primeiro acionamento até o fim do processo, destacando suas funcionalidades e virtudes dentro de uma linha de produção.

A pesquisa ainda tem como objetivos específicos:

- Exemplificar por meio de simulação como uma aplicação real funcionária na prática;
- correlacionar e destacar o uso de CLP e Sistemas Supervisórios, grandes pilares da automação industrial;
- demonstrar como é utilizada a comunicação OPC entre cliente e servidor em uma aplicação industrial.

2. Referencial Teórico

2.1. Automação

O termo automação encontra-se bastante em alta nos ultimos tempos, devido aos avanços tecnológicos esta prática se tornou essêncial, no entanto não é de hoje que existem diversos processos automatizados.

"A palavra automation foi inventada pelo marketing da indústria de equipamentos na década de 1960. O neologismo sem dúvida sonoro buscava enfatizar a participação do computador no controle automático industria. (MORAES e CASTRUCCI, 2007)".

A automação em uma definição resumida é qualquer processo onde há interação de um computador de forma há compactuar com o controle do mesmo. E tudo isso é feito com o intuito de ter a menor interferência humana possível, tanto pela segurança das pessoas quanto pelo poder de reação das máquinas que é muito maior contribuindo assim para maior eficiência.

> A automação envolve a implantação de sistemas interligados e assistidos por redes de comunicação,compreendendo sistemas supervisórios e interfaces homem máquina que possam auxiliar os operadores no exercício da supervisão e da análise dos problemas que porventura venham a ocorrer. (MORAES e CASTRUCCI, 2007).

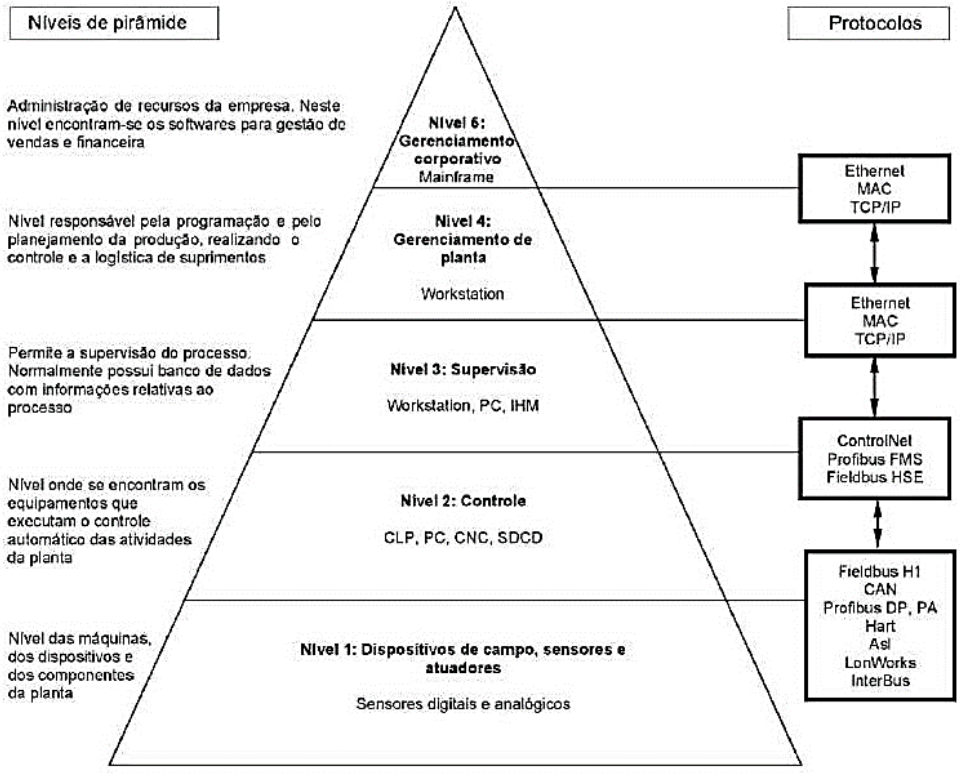
É importante também deixar claro que a automação veio para somar com a mão de obra humana. Apesar de as máquinas tomarem frente no trabalho duro e repetitivo são os humanos quem as coordenam e as operam.

> Muitas pessoas pensam e temem que a automação significa perda de empregos, quando pode ocorrer o contrário. De fato, falta de automação coloca muita gente para trabalhar. Porém, estas empresas não podem competir economicamente com outras por causa de sua baixa produtividade devida à falta de automação e por isso elas são forçadas a demitir gente ou mesmo encerrar suas atividades. Assim, automação pode significar ganho e estabilidade do emprego, por causa do aumento da produtividade, eficiência e economia. (RIBEIRO, 2001).

Vale ressaltar, de acordo com (DORF e BISHOP, 2001) “as máquinas são utilizadas para aumentar a produção per capita dos operários de uma fábrica com o objetivo de compensar o aumento dos salários e os custos inflacionários”.

Para compreender de uma forma resumida, a automação se divide em cinco camadas como mostra a figura 1.

Figura 1- Pirâmide da Automação.



Fonte**:** (MORAES e CASTRUCCI, 2007).

O Nível 1, compreende basicamente o chão de fábrica onde ficam os dispositivos de campo.

No Nível 2, está situado a parte de controle abrangendo os equipamentos que comandam as atividades da planta.

O Nível 3, se caracteriza pela supervisão do processo, alí está situado os sistemas SCADA.

No Nível 4, é onde há a programação e planejamento da produção, proporcionando o controle e logística dos suprimentos presentes.

O Nível 5, é responsável pela administração, pelo gerenciamento de todo o sistema, é o nível mais alto da pirâmide.

2.2. Controlador Lógico Programável

De acordo com (PRUDENTE, 2011) o CLP é definido como sendo um equipamento constituído de componentes eletrônicos que pode ou não ter uma memória programável e armazena dados e programas com a finalidade de ler e executar instruções descritas em forma de linguagem de programação, interagindo com um sistema que deve ser controlado por dispositivos de entrada ou saídas digitais ou analógicas. É importante dizer que ele foi projetado para operar em ambiente industrial suportando variações de temperatura, umidade, distúrbios elétricos dentre outras váriaveis existente neste ambiente.

Segundo (MORAES e CASTRUCCI, 2007) no fim da década de 1960 eram utilizados circuitos integrados, ou seja componentes eletrônicos para desenvolver computadores de pequeno porte que pudessem controlar processos. Na decada de 1970 esses controladores passam a ter microprocessadores e então foram chamados de CLP's, por fim na década de 1980 houve o aperfeiçoamento das comunicações e então começaram a serem utilizados em rede.

Com o avanço da automação se tornou indispensável a evolução dos equipamentos que a regem e o principal ou um dos pricipaís desses aparelhos é o CLP, amplamente utilizado nas indústrias devido a sua versatilidade e poder de processamento. Atualmente um controlador lógico programável é completamente robusto e desenvolvido, possui uma diversa variedade de processadores, boa capacidade de memória, vários meios de comunicação e bastante entradas e saídas para se conectar com os equipamentos de campo, tudo de acordo com a necessidade da aplicação.

Figura 2 - Exemplo de um CLP Moderno.

Fonte: (SIEMENS, 2020)

Segundo (PRUDENTE, 2011) o hardware de um CLP é constituído por três partes fundamentais; unidade central, unidade de entrada/saída I/O e unidade de programação. A unidade central é onde se organizam as funções de controle, a unidade de entradas e saídas é onde ocorre a ligação entre o dispositvo de controla (CLP) com os dispositivos de campo a ser controlados, a unidade de programação onde ocorre a interação do homem com a máquina que permite o armazenamento do programa escrito na memória do CLP.

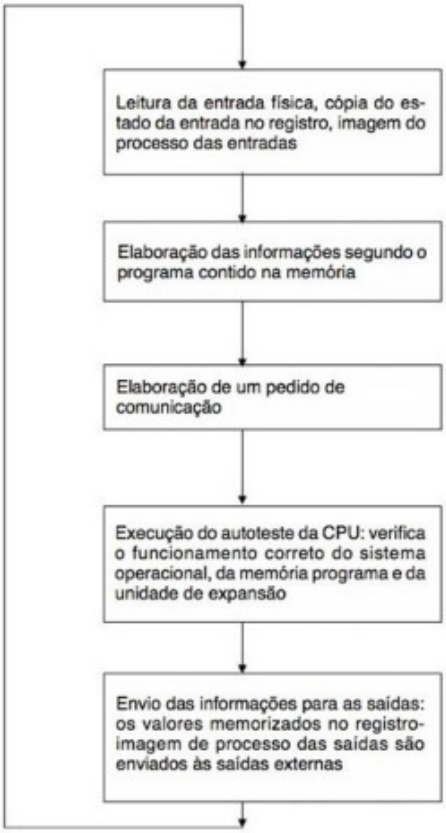
Explicando mais detalhadamente, (NATALE, 2018) descreve que a CPU de um CLP pode ser constituida tanto por um microprocessador quanto por um microcontrolador, uma memória RAM, local onde os programas são executados, e uma memória Flash EPROM ou E2PROM, responsável por armazenar uma cópia do programa que esta sendo executado. O sistema também é integrado por uma IHM, que é ligada a CPU e atua como um periférico que tem como função intermediar a comunicação do operador com o sistema.

É muito importante que se conheça a fundo as linguagens de programação presentes em um CLP pois é através da lógica criada através delas que tudo é possível funcionar, no entanto a forma com que o dispositvo lê e processa essas informações também deve ser entendida.

Seguindo o pensamento de (NATALE, 2018), como esse processamento é feito em tempo real quando o CLP recebe uma nova informação de entrada ela é comparada com informações presentes na memória, feito isso as decisões são tomadas e assim os comandos ou acionamentos são executados pelas saídas tudo acontece simultaneamente no desenrolar do processo.

O CLP basicamente lê as entradas e atualiza as saídas, porém ele trabalha de uma forma sequencial fazendo uma varredura. Essa leitura completa e em sequência é chamada de ciclo de scan, a figura 3 mostra de forma mais detalhada como esse ciclo acontece.

Figura 3- Especificação do Ciclo de Scan.



Fonte : (PRUDENTE, 2011)

De acordo com (MORAES e CASTRUCCI, 2007) existem muitas linguagens de programação que podem ser usada em um CLP. O International Electrotechnical Comittee – IEC é quem é responsável por padronizar essa linguagens, mais especificamente a norma IEC 61131-3 Programing Languages.

As cinco linguagens que são encontradas geralmente em programas de CLP e que são padronizadas pela norma são: Texto Estruturado (Strutured Text ou ST) e Lista de Instruções (Instruction List ou IL) sendo classificadas como linguagens textuais. Diagrama Ladder (LD) e Diagramas de Blocos Funcionais (Function Block Diagram ou FBD) classificadas como linguagens gráficas. Linguagem de Diagrama Sequencial (Sequential Flow Chart ou SFC) classificada como linguagem gráfica, porém permite a adição de textos.

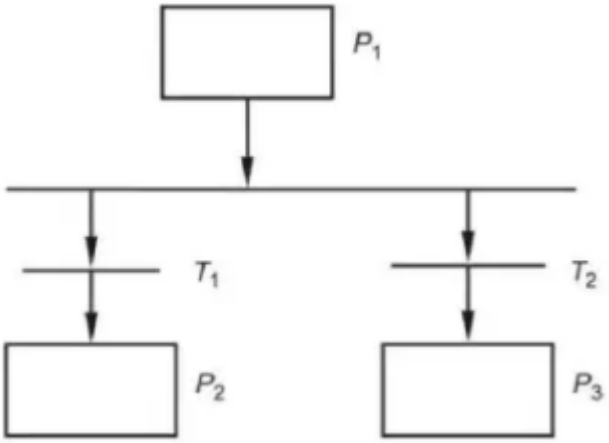
As linguagens utilizadas nesta pesquisa foram o ST para a construção gráfica da visualização no CODESYS e o SFC, que tem como característica criar passos seguidos por transições em forma gráfica, proporcionando praticidade e objetividade durante a construção do programa.

(DE MELLO, 2016) explica que o SFC é uma linguagem originada da Redes de Petri, como GRAFCET (Graphic de Commande Etape-Transition) da França e o MFG (Mark Flow Graph) do Japão. Cujo objetivo é descrever um processo de forma clara e explícita fazendo com que o programador expresse seu raciocínio lógico de forma prática e gráfica.

De acordo com (KATO e PORTO, 1999) a associação das Redes de Petri com a linguagem de programação CLP proporciona uma representação de fácil entendimento possibilitando a maior compreensão das pessoas envolvidas em um projeto.

Ainda segundo (KATO e PORTO, 1999) essa dinâmica de passo e transição presente nas Redes de Petri permite um manuseio melhor em relação as redes condição evento, permitindo ao projetista maior facilidade ao utilizar os recursos, como temporizadores, contadores e comparadores.

Figura 4- Exemplo de Representação SFC.



Fonte**:** (MORAES e CASTRUCCI, 2007).

Outra coisa que deve ser citada é a forma com que o CLP se comunica com o computador e os equipamentos de campo, isso é feito via rede porém existem diversos protocolos com tipologias reconhecidas para executar esse trabalho, alguns exemplos são Ethernet, Profibus, Modbus, DeviceNet entre outros. Vale ressaltar que a comunicação pode ser feita também via serial utilizando dispositivos de diferentes padrões como RS232 e RS485 por exemplo. Geralmente esses protocolos são padronizados, garantindo maior flexibilidade entre a comunicação de equipamentos de diferentes fabricantes.

2.3. Sistema de Supervisão e Aquisição de Dados

Como o próprio nome diz esse sistema é responsável pela supervisão e aquisição de dados do processo, o seu maior objetivo é mostrar com transparência os parâmetros que estão sendo medidos. É indispensável o uso desse recurso em uma automação pois com a análise que ele trás é possível melhorar a eficiência e até mesmo reduzir custos operacionais.

De acordo com (DA SILVA e SALVADOR, 2005) os sistemas supervisórios permitem que sejam feitas análises em tempo real de um processo produtivo ou uma instalação física. Essas análises são feitas através de coletas de informações feita por equipamentos de aquisição de dados e, em seguida, manipuladas, armazenadas e, apresentadas ao usuário.

Para (BOYER 2009) os sistemas supervisórios são de extrema importância nos ambientes industriais, pois além de prover todos os beneficíos de visualização e acompanhamento do processo, ele permite isso de forma remota ou seja, é possível ter a visão e controle do sistema mesmo a distâncias muito longas, de centenas ou milhares de quilômetros.

Ainda segundo (BOYER 2009) Os principais componentes dentro de um sistema SCADA são, o operador que está no centro de tudo e, faz a interface com o sistema ligado a algum dispositvos com entradas e saídas (I/O). Então, esse dispositivo se comunica com a unidade terminal mestre (MTU), por sua vez essa unidade mestre se comunica com as unidades terminais remotas (RTU) presentes no sistema, por fim essas unidades remotas se comunicam com sensores ou atuadores do processo.

Esse tipo de sistema geralmente está presente em um ambiente indústrial de duas formas; Por meio de uma IHM (Interfáce Homem-Máquina) normalmente presentes no chão de fábrica. Ou através dos computadores situados na sala de controle.

A IHM, foi projetada assim como o CLP para operar em ambientes agressivos suscetíveis a variações de temperatura, humidade, vibrações constantes, etc. Normalmente são instaladas na linha de produção proprocionando um contato direto com o operador.

Figura 5 - Exemplo de uma IHM.

Fonte: (SIEMENS, 2020)

Sistemas SCADA são também presentes em salas de controle, normalmente para supervisionar e controlar altas quantidades de variáveis presentes em processos diferentes por exemplo. Para isso o sistema de aquisição de dados tem como plataforma de operação um microcomputador, ao invés de paineis convencionáis a visualização é feita através de telas de monitoramento e algum controle ou acionamento necessário é feito através de mouse e teclado e não por botoeiras.

Figura 6 - Exemplo de uma sala de supervisão.

Fonte: (SIAUTEC, 2019)

O software trás todas as informações necessárias para controle e monitoramento de forma simples porém completa. O sitema é composto por acesso único, ou seja cada operador deve logar com seu usuário e senha, isso é importante pois fica registrado o nome das pessoas que tiveram acesso e modificaram algo, além de ser possível restringir as funções que cada usuário pode ter.

Faz parte também do funcionamento de um sistema supervisório dectectar e sinalizar alarmes e falhas de segurança. Geralmente dentro do programa existe uma tela própria para esta função que mostra de maneira clara alterações e falhas indesejadas que ocorreram no processo.

Junto ao monitoramento das variáveis do processo, ao planejamento de alarmes e falhas é comum conter em um sistema supervisório gráfico de tendências. Neste gráfico contém variáveis que são atualizadas simultâneamentes, proporcionando um acompanhamento em tempo real.

Mas para tudo isso ser possível é necessário existir uma comunicação entre o CLP e o Sistema SCADA. Segundo (MORAES e CASTRUCCI, 2007) essa comunicação é feita atráves de TAGs (Etiquetas) que o CLP envia junto aos sinais para o sistema de supervisão. Esses TAGs carregam informações como o endereço dentro do CLP e o tipo de TAG. São tipos de TAGs: Device, quando o dado se origina do CLP para o supervisório. DDE (Dynamic Data Exchange), quando os dados se originam de um outro computador da rede. Memory, quando os dados existem localmente no sistema de spervisório.

Ainda de acordo com (MORAES e CASTRUCCI, 2007) o que predomina em sistemas supervisórios é o padrão Windows da Microsoft, por ser mais convencional ajuda na interação com o operador, outra vantagem é a comunicação com outros produtos da rede, como o Excel ou o Olé.

Segundo (FONSECA 2002) hoje em dia um padrão muito utilizado para a comunicação entre os equipamentos do chão de fábrica e os sistemas de automação é o OPC (OLE for Process Control). O objetivo principal de um servidor OPC é estabelecer regras e padronizar os sistema para a comunicação entre dispositivos de campo como o CLP e sistemas de supervisão como o SCADA.

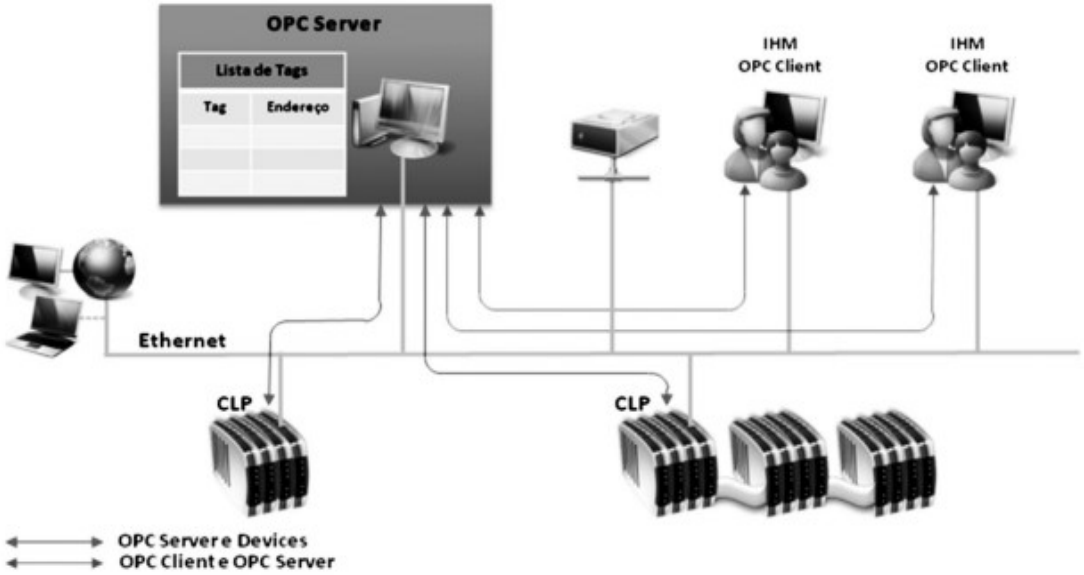
O servidor OPC possui um cliente e um servidor, há produtos no mercado que podem exercer ambos os papeis, porém normalmente o cliente são os sistemas de supervisão e o servidor o CLP. Um padrão bastante utilizado, inclusive nessa pesquisa é o OPC (DA) que permite leitura e escrita de dados em tempo real.

2.4. O Padrão OPC

(FRANCO, 2010) descreve o protocolo OPC como um padrão de comunicação de dados que concentra em um só lugar infromações providas de diversos pontos distintos . Uma grande vantagem de se usar o OPC se da por conta de ser um protocolo universal para envio de dados entre “Clientes” (aplicações) e “Servidores” (fonte de dados), fazendo com que o uso de drivers e conversores sejam minimamente utilizados.

O padrão OPC é amplamente utilizado em aplicações industriais justamente por possuir essa versatilidade de ser um protocolo universal que faz de forma eficaz e segura a comunicação entre os meios de fonte de dados e aplicações ou seja, a conexão entre o CLP e o sistema SCADA. A figura 7 exemplifica a conexão entre dispositivos via OPC.

Figura 7 - Acesso a dados de processos com protocolo OPC.



Fonte : (FRANCO, 2010)

Por ser tão amplamento utilizado, o padrão OPC possui algumas variações que abrangem as diversas necessidades dos clientes, é importante ressaltar também que o protrocolo sofreu algumas melhoras e atualizações ao longo dos anos, visando sempre sua otimização.

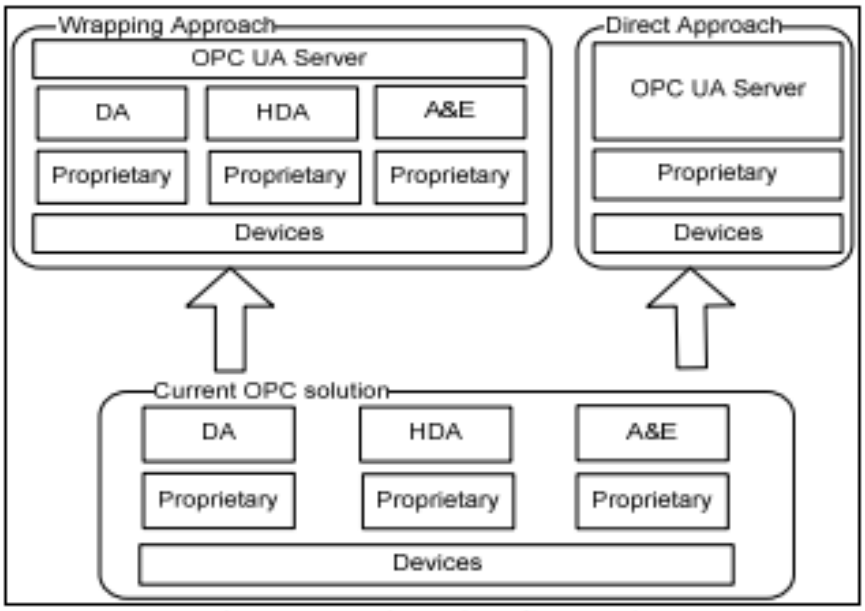
(GONZÁLEZ, 2019) cita que o chamado OPC clássico teria sido desenvolvido em 1996 através de uma grande operação da industria de automação com o intuito de fornecer um protocolo de comunicação para aplicativos de software e hardware de automação baseados em computador pessoal.

(GONZÁLEZ, 2019) explica ainda que atualmente o OPC possui dez especificações diferentes que são gerenciadas pela OPC Fundation, são elas: acesso de dados (DA), acesso a histórico de dados (HDA), alarmes e eventos (A&E), acesso a dados XML (XML-DA), troca de dados (DX), dados complexos (CD), segurança, lote, interface expressa (Xi) e arquiterua unificada (UA). O OPC clássico compreende as oitro primeiras especificações sendo o OPC DA o mais amplamente utilizado.

Apesar de ser presente em diversas aplicações o chamado OPC clássico deu espaço a uma nova tecnologia, com intuito de unir todas as especificações em uma só foi criado o protocolo OPC de arquitetura unificada.

(LEITNER, 2006) explica sobre os requisitos adicionais presentes no OPC UA, o conjunto de novos recursos conta com histórico de eventos, múltiplas hierarquias, métodos e programas. Outra grande conquista presente é um modelo de dados de nível superior diferentemente da arquitetura anterior, além de fornecer tipo de dados flutuantes e meta informações como unidade de engenharia.

Figura 8 – Compactação dos protocolos clássicos para o protocolo UA.



Fonte: (LEITNER, 2006)

3. Metodologia

Neste capítulo serão apresentados os métodos utilizados para desenvolver a pesquisa, os instrumentos necessários e a forma como foram explorados para obter os resultados.

O projeto teve início ao decorrer da graduação, mais especificamente durante o curso da disciplina de CLP. A partir deste momento começaram as pesquisas e análises sobre o tema em questão. A princípio o trabalho foi composto apenas pela parte de controle lógico utilizando as ferramentas de desenvolvimento e visualização do software CODESYS. Porém com o avanço das pesquisas obteve-se o intuito de fazer o aprimoramento deste projeto com a implementação de um sistema de supervisão SCADA por meio de uma comunicação OPC feita entre o software de controle e o de supervisionamento.

A seguir estão explicados todos os passos e direcionamentos utilizados para o desenvolvimento do trabalho.

3.1. Simulação do CLP

O primeiro passo para a realização do projeto é ter uma simulação fiel de um CLP fisíco, para isso foi escolhido o software CODESYS devido ao prévio conhecimento da ferramenta, por se tratar de um programa gratuito e, por atender muito bem todas as especificações necessárias. Muito popular e com um nome consolidado o software possui um poderoso simulador de CLP integrado, que segue a risca as funcionalidades de um controlador real. Ele também oferece variedade nas linguagens de programação, ambiente para simulação de telas e também a comunicação OPC com outros softwares de supervisão, atendendo totalmente as especificações do projeto.

Foi construída uma lógica que atendesse da melhor forma possível a ideia inicial. Por isso a linguagem escolhida foi o SFC, que como dito antes é caracterizado por criar passos seguidos de transições, exatamente o que foi proposto no projeto. As figuras a seguir ilustram as variáveis utilizadas e a lógica de programação, como forma também de demonstrar o ambiente de trabalho disponível no software.

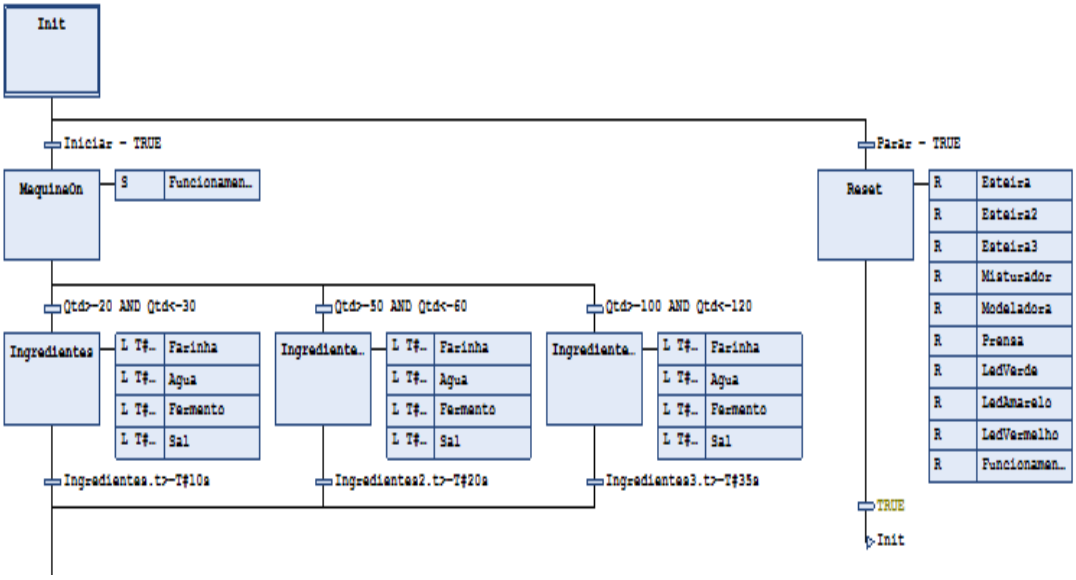
Figura 9 – Variáveis declaradas em SFC feita no CODESYS.

```
PROGRAM Programa_SFC
VAR
    TempEstufa: REAL;
    TempForno: REAL;
    Qtd: REAL;
    Descanso: TIME;
    Umid: REAL;
    Iniciar: BOOL;
    Parar: BOOL;
    Funcionamento: BOOL;

    Farinha: BOOL;
    Agua: BOOL;
    Fermento: BOOL;
    Sal: BOOL;

    Misturador: BOOL;
    Modeladora: BOOL;
    Prensa: BOOL;

    LedVerde: BOOL;
    LedAmarelo: BOOL;
    LedVermelho: BOOL;
    LedAzul: BOOL;

    Esteira: BOOL;
    Esteira2: BOOL;
    Esteira3: BOOL;
    Cilindro1: BOOL;
    Cilindro2: BOOL;
    Cilindro3: BOOL;
    Cilindro4: BOOL;
    Cilindro5: BOOL;

    PortaEstufa: BOOL;
    PortaForno: BOOL;
END_VAR
```

Fonte: *print screen* da tela de programação em SFC no CODESYS.

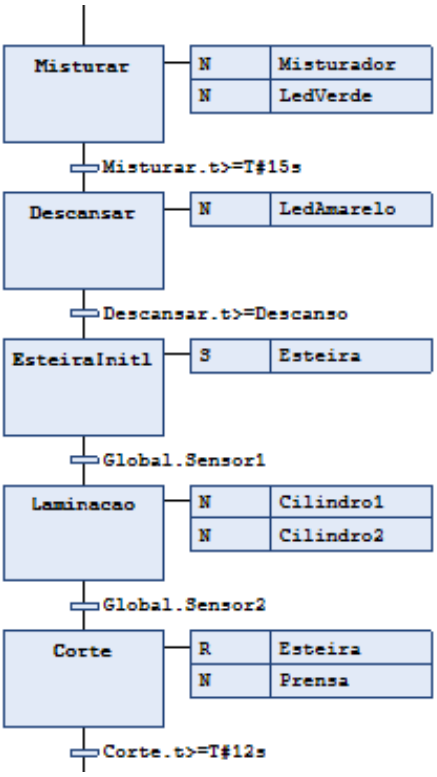
A “Figura 9” trás todas as variáveis declaradas no programa e seus respectivos tipos, foram utilizadas variáveis do tipo “REAL” para representar de fato números reais que serão setados nos equipamentos analógicos que mensuram quantidades e temperaturas, “TIME” para representar a unidade de tempo e, “BOOL” representando as variáveis digitais do tipo booleanas como por exemplo sensores, atuadores e LED’s.

Figura 10 – Programação em SFC feita no CODESYS.



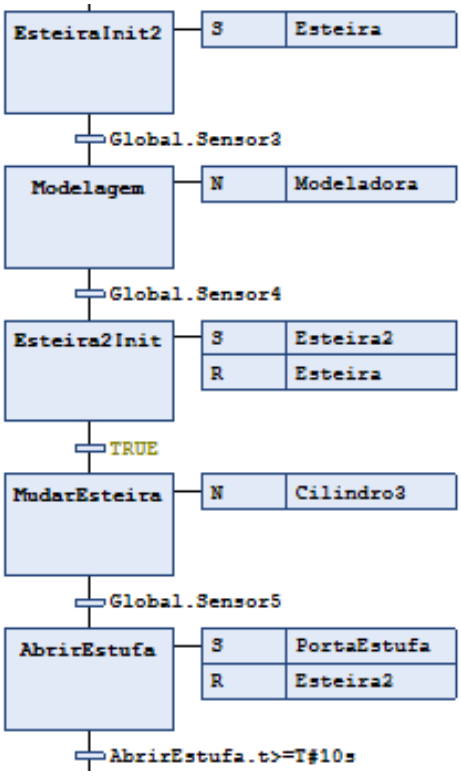
Fonte: *print screen* da tela de programação em SFC no CODESYS.

Figura 11 – Parte 2: Programação em SFC feita no CODESYS.



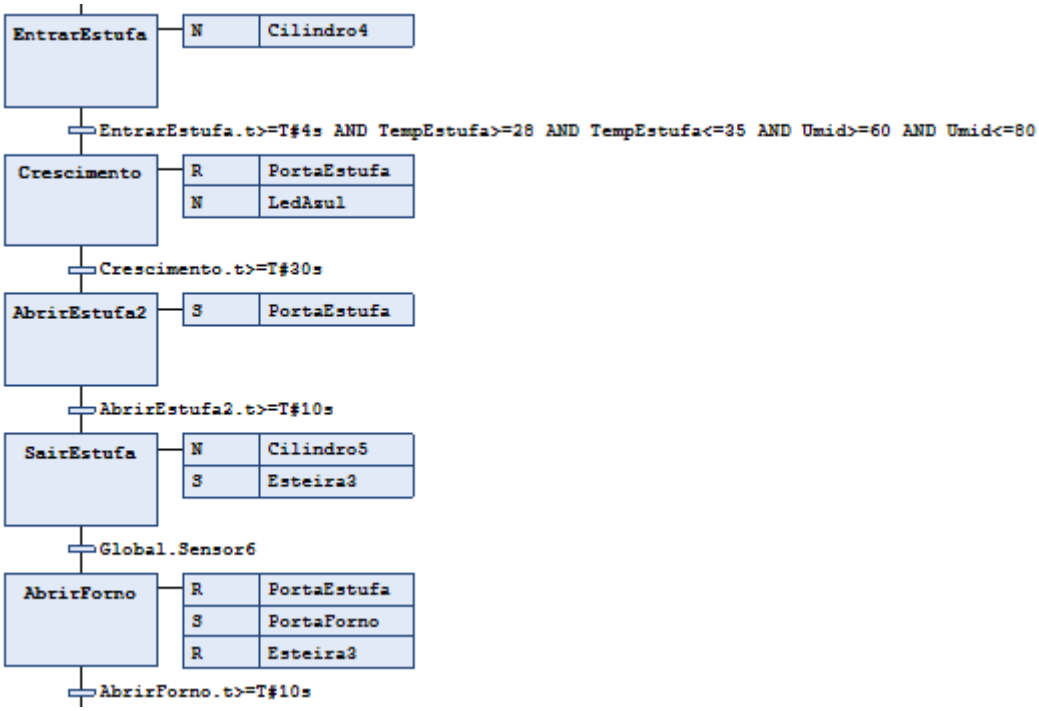
Fonte: *print screen* da tela de programação em SFC no CODESYS.

Figura 12 – Parte 3: Programação em SFC feita no CODESYS.



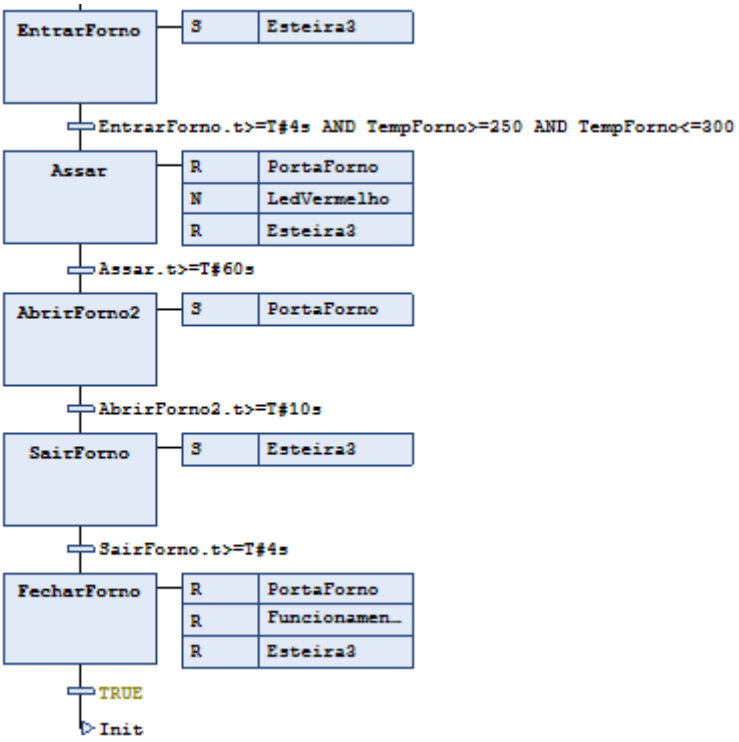
Fonte: *print screen* da tela de programação em SFC no CODESYS.

Figura 13 – Parte 4: Programação em SFC feita no CODESYS.



Fonte: *print screen* da tela de programação em SFC no CODESYS.

Figura 14 – Parte 5: Programação em SFC feita no CODESYS.



Fonte: *print screen* da tela de programação em SFC no CODESYS.

A ideia de passo e transição e a utilização desta linguagem veio pela forma em que se foi pensado o andamento do processo. A linha de produção será caracterizada, a princípio, por 6 etapas de processos para chegar ao resultado final do produto. Todas essas etapas serão automatizadas e seguirão a ordem definida, a transição de uma etapa para outra será feita através de uma esteira.

Essas 6 etapas descritas anteriormente são:

1. Homogeneização dos ingredientes no tanque de mistura.
2. Ativação de uma laminadora para a laminação da massa.
3. Ativação de uma prensa para o corte da massa.
4. Ativação de uma modeladora para dar molde a massa.
5. Ativação de uma estufa para o crescimento da massa.
6. Ativação de um forno para assar e finalizar o processo.

É necessário ressaltar que a simulação é ainda mais complexa e fiel a realidade, dentre esses passos são acionados sensores e esteiras a todo momento, e tudo isso é mostrado durante a simulação do programa.

Ainda utilizando os recursos e ferramentas do CODESYS, desta vez por meio da linguagem de programação de texto extruturado ST (Structured Text) foi criada uma interface que proporciona a visualização em tempo real dos acontencimentos de todas as etapas da simulação em forma gráfica e com imagens. As figuras abaixo mostram as variáveis utilizadas, a lógica de programação e, como é o ambiente de trabalho para a construção de uma visualização utilizando ST. Além disso estão dispostos também os resultados finais após a criação da tela.

Figura 15 – Declaração das variáveis em ST feita no CODESYS.

```
PROGRAM visual
VAR
    EsteiralX : DINT;
    EsteiralX2: DINT;
    PrensaY : REAL;
    aux : DINT;
    aux2: DINT;
    Cilindro3Y : DINT;
    ReceitaY: DINT;
    ReceitaX: DINT;
    aux3: DINT;
    aux4: DINT;
    aux5: DINT;
    Esteira3X: DINT;
    aux7: DINT;
    Esteira3X2: DINT;
END_VAR
```

Fonte: *print screen* da tela de programação em ST no CODESYS.

Para realizar a programação da parte de visualização em ST no CODESYS foi usada uma lógica de plano cartesiano X e Y para determinar a posição das imagens e também seu espaço de movimento.

Todas as variáveis exceto a “PrensaY” foram declaradas como “DINT” por necessitarem apenas de números inteiros para navegação no plano. A exceção foi justamente a prensa que necessitou de uma variável do tipo “REAL”, pois foram utilizados números reais com casas decimais para definir sua zona de movimentação no plano.

Figura 16 – Parte 1: Programação em ST feita no CODESYS.

```
1   IF Programa_SFC.Iniciar = 1 THEN
2       aux := 0;
3       aux2 := 0;
4       aux3 := 0;
5       aux4 := 0;
6       aux5 := 0;
7       aux7 := 0;
8   END_IF
9   IF Programa_SFC.Esteira = 1 THEN
10      EsteiralX := EsteiralX + 1;
11  END_IF
12
13  IF EsteiralX = 190 THEN
14      Global.Sensor1 := 1;
15  END_IF
16  IF EsteiralX = 420 THEN
17      Global.Sensor1 := 0;
18      Global.Sensor2 := 1;
19  END_IF
20  IF Programa_SFC.Prensa = 1 THEN
21      IF aux = 0 THEN
22          PrensaY := PrensaY + 0.5;
23          IF PrensaY = 150 THEN
24              aux := 1;
25          END_IF
26      END_IF
27      IF aux = 1 THEN
28          PrensaY := PrensaY - 0.5;
29          IF PrensaY = 0 THEN
30              aux := 2;
31          END_IF
```

Fonte: *print screen* da tela de programação em ST no CODESYS.

Figura 17 – Parte 2: Programação em ST feita no CODESYS.

```
32      END_IF
33  END_IF
34  IF EsteiralX = 620 THEN
35      Global.Sensor3 := 1;
36      Global.Sensor2 := 0;
37      aux5 := 1;
38  END_IF
39  IF Programa_SFC.Modeladora = 1 THEN
40      EsteiralX2 := EsteiralX2 + 1;
41  END_IF
42  IF EsteiralX = 1000 THEN
43      Global.Sensor4 := 1;
44      Global.Sensor3 := 0;
45  END_IF
```

Fonte: *print screen* da tela de programação em ST no CODESYS.

Figura 18 – Parte 3: Programação em ST feita no CODESYS.

```
46
47  IF Global.Sensor4 = 1 THEN
48      IF aux2 = 0 THEN
49      Cilindro3Y := Cilindro3Y + 1;
50      aux4 := 1;
51      IF Cilindro3y = 220 THEN
52          aux2 := 1;
53          END_IF
54      END_IF
55      IF aux2 = 1 THEN
56          Cilindro3Y := Cilindro3Y - 3;
57          IF Cilindro3Y = 0 THEN
58              aux2 := 2;
59          END_IF
60      END_IF
61  END_IF
62  IF Cilindro3Y >= 60 THEN
63      aux5 := 0;
64      IF aux3 = 0 THEN
65      ReceitaY := ReceitaY + 1;
66      END_IF
67      IF ReceitaY >= 160 THEN
68          aux3 := 1;
69      END_IF
70  END_IF
71  IF aux3 = 1 THEN
72          ReceitaX := ReceitaX - 1;
73  END_IF
```

Fonte: *print screen* da tela de programação em ST no CODESYS.

Figura 19 – Parte 4: Programação em ST feita no CODESYS.

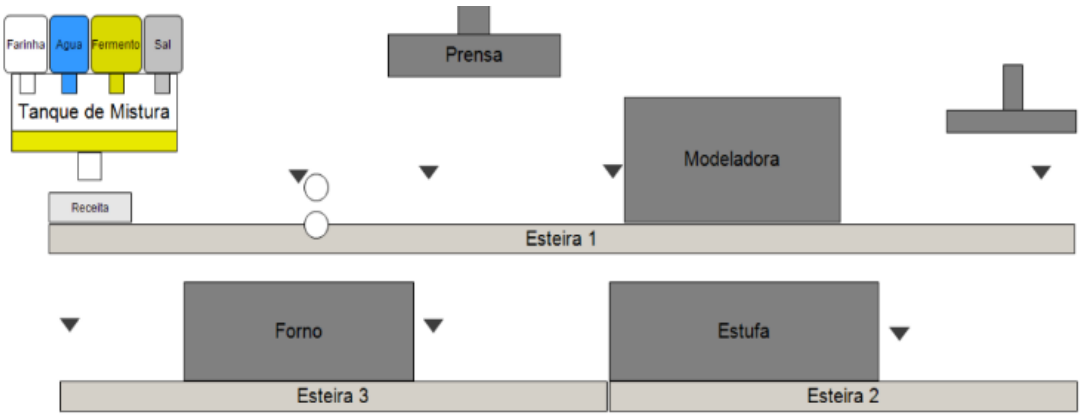
```
74  IF ReceitaX <= -78 THEN
75      aux3 := 2;
76      Global.Sensor5 := 1;
77      Global.Sensor4 := 0;
78  END_IF
79  IF Programa_SFC.Cilindro4 = 1 THEN
80      ReceitaX := ReceitaX - 1;
81  END_IF
82  IF Programa_SFC.Esteira3 = 1 THEN
83      Esteira3X := Esteira3X - 1;
84      IF Esteira3X = -200 THEN
85          Global.Sensor6 := 1;
86          Global.Sensor5 := 0;
87      END_IF
88  END_IF
89  IF Programa_SFC.LedVermelho = 1 THEN
90      aux7 := 1;
91  END_IF
92  IF Programa_SFC.Esteira3 = 1 AND aux7 = 1 THEN
93      Esteira3X2 := Esteira3X2 - 1;
94  END_IF
95  IF Esteira3X2 = -220 THEN
96      aux7 := 2;
97  END_IF
```

Fonte: *print screen* da tela de programação em ST no CODESYS.

Figura 20 – Painel de controle feito no CODESYS.

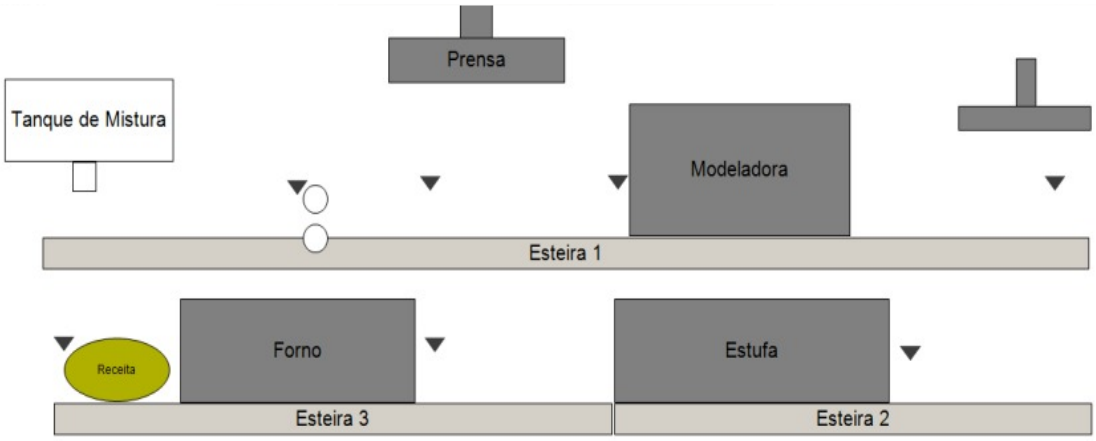
Fonte: *print screen* da tela de painel de controle do programa no CODESYS.

Figura 21 – Visualização do início do processo de simulação.



Fonte: *print screen* da tela de visualização do programa no CODESYS.

Figura 22 – Visualização do fim do processo de simulação.



Fonte: *print screen* da tela de visualização do programa no CODESYS.

3.2. Comunicação com o sistema de supervisão

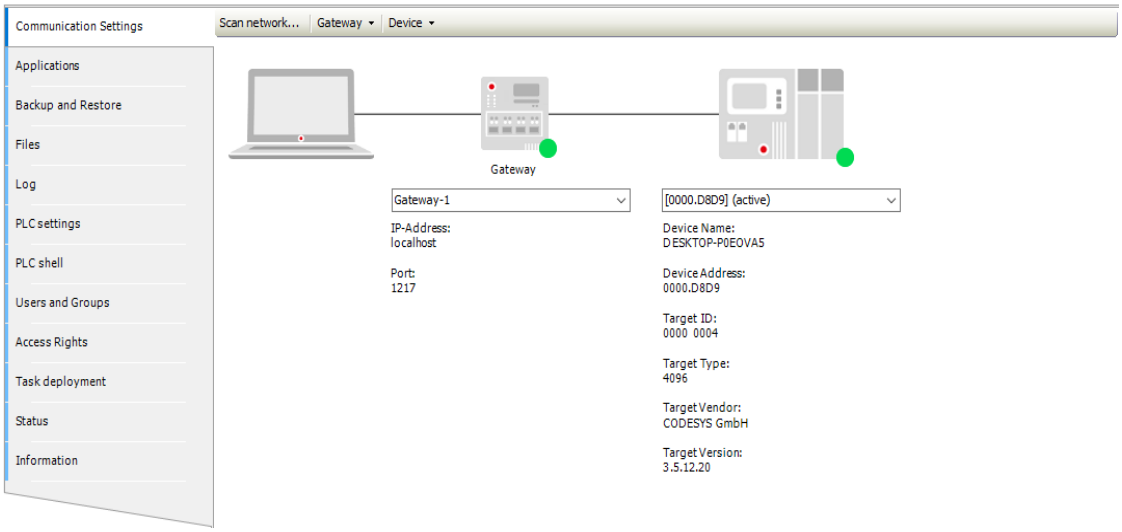
Feito todo o trabalho de arquitetura e funcionamento do código o segundo passo é conectar esse CLP com o sistema de supervisão. Como citado anteriormente nessa simulação a comunicação é feita via OPC DA, caracterizada por um padrão cliente-servidor que transmite dados entre eles em tempo real.

Neste segmento o software CODESYS, simulador do CLP atua como o cliente fornecendo ao servidor os dados que devem ser lidos e comunicados. Por sua vez o sistema de supervisão neste caso o Elipse SCADA recebe esses dados e os disponibilizam para leitura e criação do sistema supervisório. Porém antes de comunicarmos com o sistema de supervisão é necessário configurar a comunicação entre o computador local e o CLP virutal.

Sendo mais específico, esse CLP virtual existente no CODESYS que permite uma simulação fiel consiste em um soft-PLC denominado Codesys Control Win V3 que assim como o sistema OPC estão presentes no pacote de instalação. Esse soft-PLC é compatível com a norma IEC 61131-3 e por isso permite a programação nas linguagens presentes nesta norma.

A figura abaixo demonstra a tela de comunicação após a configuração entre o Codesys Control Win V3 e o computador local.

Figura 23 – Comunicação do CLP com o computador local.



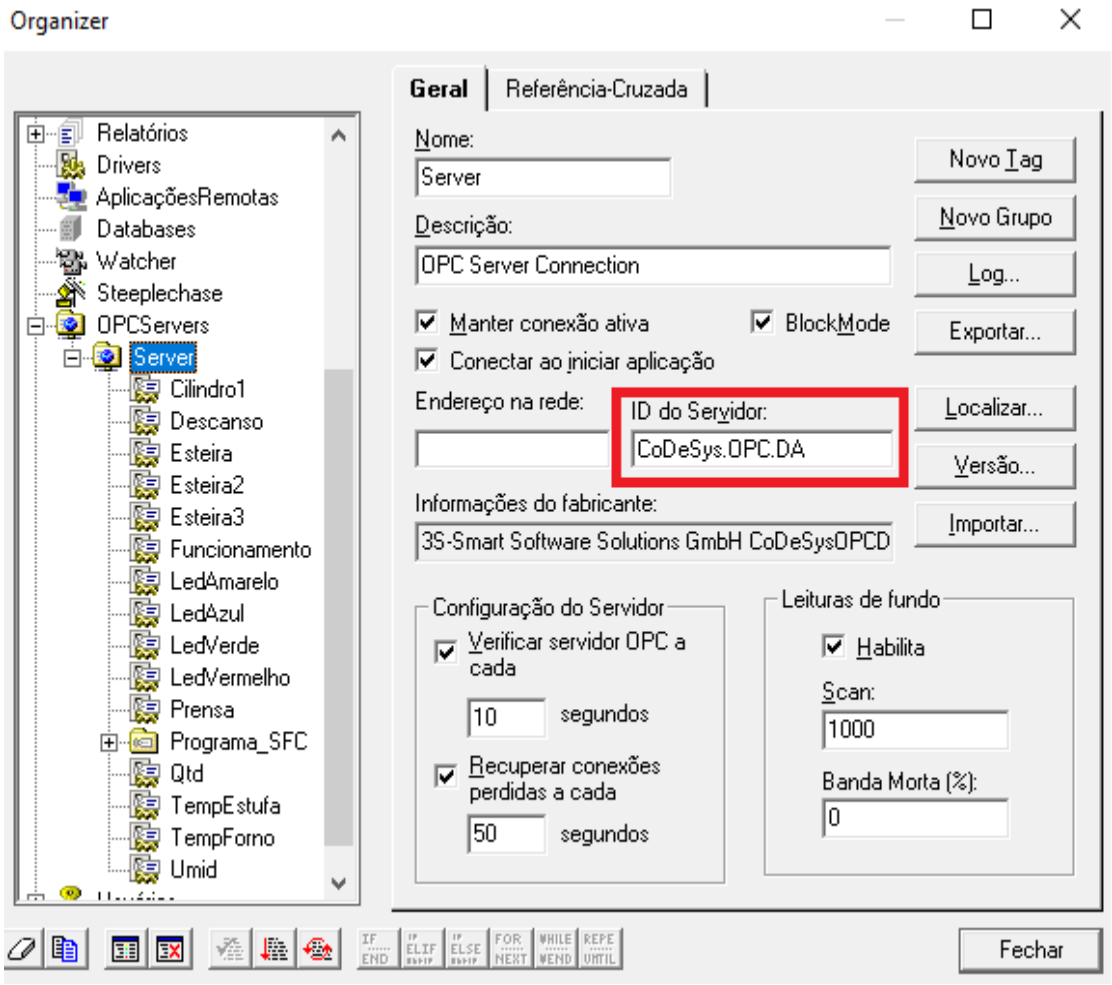
Fonte: *print screen* da tela de coomunicação do CLP com o computador local no CODESYS.

Feito essa comunicação entre os dispositivos, o CLP está agora conectado com a máquina e pronto para receber o download do código. O proximo passo é fazer a comunicação OPC com o sistema de supervisão. Parar isso é necessário dentro do Elipse SCADA configurar o servidor OPC para receber as informação do CODESYS.

Desta forma, com essa comunicação entre os dois softwares essenciais estabelecida resta fazer a configuração adequada para que as informações trocadas sejam apenas as necessárias.

A figura abaixo mostra a tela de configuração do Elipse SCADA após estabelecer a comunicação com o CODESYS.

Figura 24 – Tela de configuração do servidor OPC no Elipse SCADA.



Fonte: *print screen* da tela de configuração do servidor OPC no Elipse SCADA.

Seguidos todos os passos tem-se a comunicação local estabelecida com o simulador do CLP pronto para uso e também a interação entre o CODESYS e o sistema de supervisão via OPC.

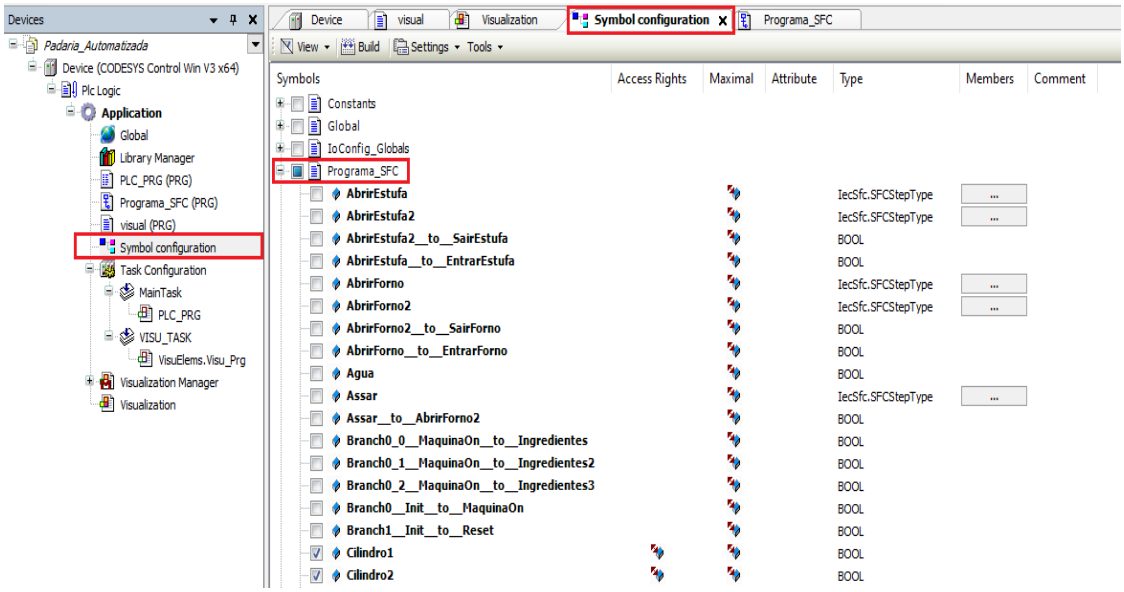
3.3. Configuração do Sistema Supervisório

O sistema supervisório proposto tem a finalidade de monitorar o funcionamento dos equipamentos e acompanhar o processo como um todo. O programa terá acesso as variáveis do processo, que são simuladas através do software CODESYS e serão transmitidas via OPC ao Elipse SCADA como dito anteriormente.

O sistema possui variáves analógicas e digitais que serão transmitidas em tempo real. A configuração para que isso seja possível começa no CODESYS pois essas variáves são originadas pelo programa. Tendo isso em mente podemos adotar o nome dessas variáves como "tags" que serão transmitidas de um software para o outro, basicamente então o primeiro processo é selecionar as tags que precisam e devem ser monitoradas.

A figura abaixo mostra detalhadamente onde extamente são selecionadas essas tags, na aba "Symbol Configuration" dentro do CODESYS.

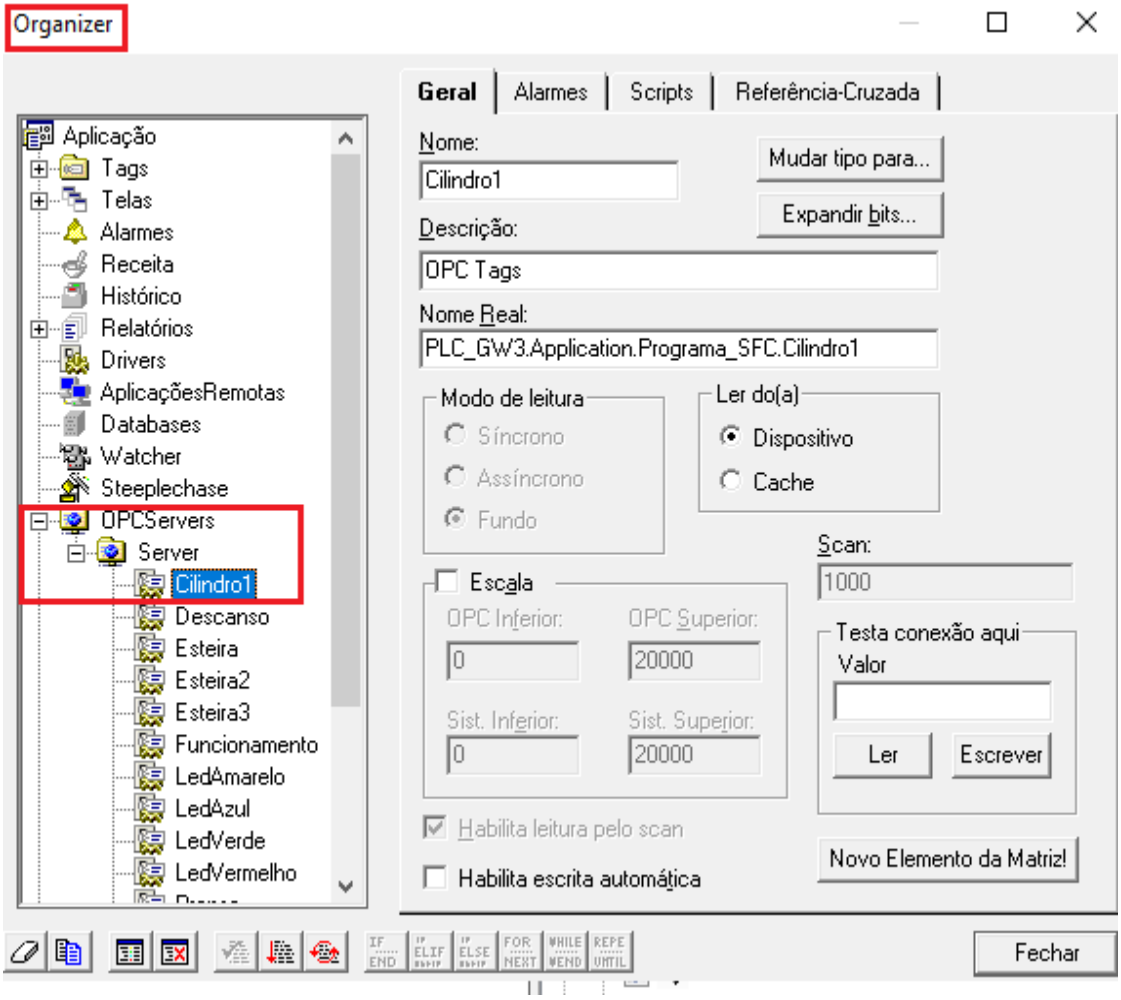
Figura 25 – Tela de configuração das tags selecionadas no CODESYS.



Fonte: *print screen* da tela de configuração das "tags" no CODESYS.

Então de acordo com o que foi explicado, após selecionar as tags necessárias o proximo passo é localiza-las e coleta-las no software de supervisão. Para isso é preciso ir dentro do Elipse SCADA na aba "Organizer" e depois em "OPC Servers" buscar as tags como demonstra a imagem abaixo.

Figura 26 – Tela de configuração das tags selecionadas no Elipse SCADA.



Fonte: *print screen* da tela de configuração das "tags" no Elipse SCADA.

Pode-se ver inclusive verificar que tanto na "Figura25" quanto na "Figura26" aparecem a tag "Cilindro1", mostrando a transmissão feita entre os dois softwares.

Após feita essa configuração de comunicação o próximo passo é construir de fato o sistema de supervisão, para isso foram feitas cinco telas de navegação, a primeira é de apresentação e login onde, se o operador insere os dados corretos ele é direcionado para a segunda tela, caso os dados forem incorretos ele continua na tela inicial. A segunda tela é um menu de opções que dá caminho a outras três telas: Monitoramento que é onde é feito toda a supervisão do processo, Alarmes e o Gráfico de tendências responsável por mostrar a variação da quantidade produzida conforme o tempo passa. Todas as telas possuem opções de retornar ao menu de opções ou de "Sair" e retornar a tela inicial.

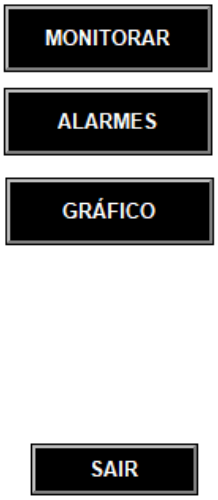
As cinco telas do sistema foram desenvolvidas para serem bastante simples e intuitivas, abaixo segue imagens representando cada uma delas:

Figura 27 – Tela inicial do sistema supervisório.

Fonte: elaborado pelo autor.

A primeira tela “Figura 27” consiste apenas em “Entrar” com usuário e senha, há também o botão “Sair” que fecha o programa. Como definições de segurança e acesso foi definido nas configurações um usuário “admin” padrão para entrar no sistema.

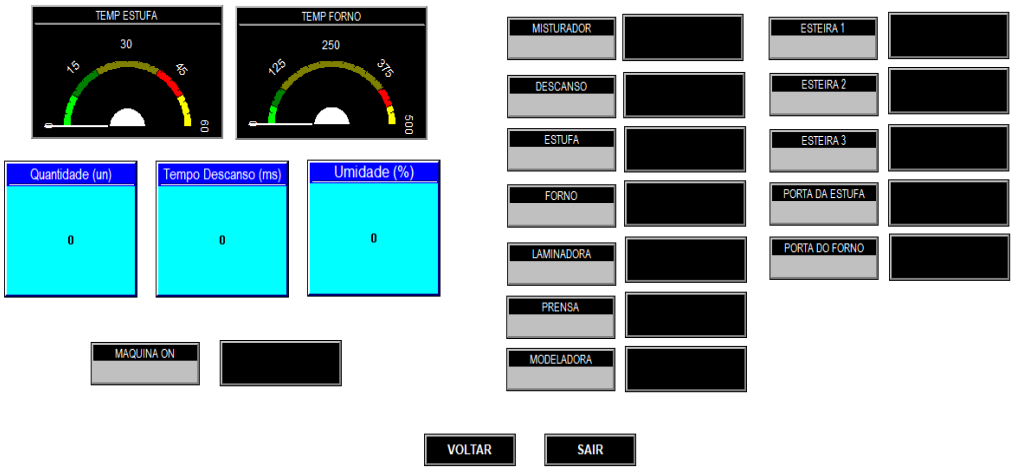
Figura 28 – Menu de opções do sistema supervisório.



Fonte: elaborado pelo autor.

A segunda tela “Figura 28” é um menu de opções onde o usuário pode escolher entre três telas de navegação (“Monitorar”, “Alarmes”, “Gráfico”), a primeira leva o usuário a central de monitoramento onde estão presentes todas as variáveis monitoráveis, a segunda mostram os alarmes detectados pelo sistema, e a tericera mostra o gráfico de tendências. É possível sair e voltar a tela inicial clicando no botão “Sair”.

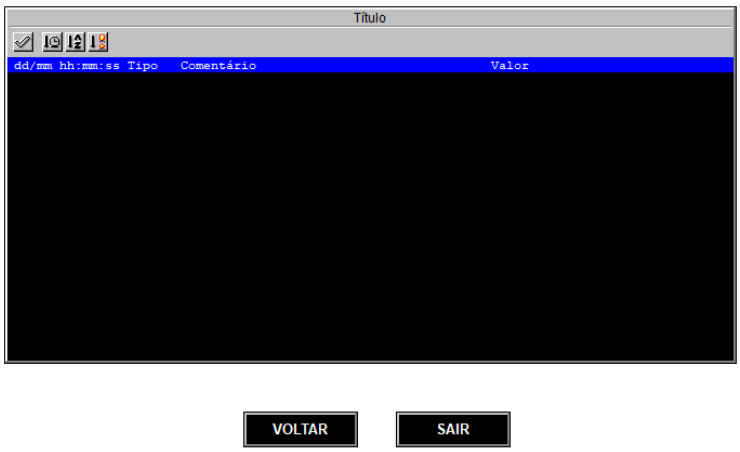
Figura 29 – Tela de monitoramento do sistema supervisório.



Fonte: elaborado pelo autor.

A terceira tela “Figura 29” é onde ocorre o monitoramento principal, nesta imagem estão todas as variáves necessárias para a supervisão do sistema, é possível verificar o funcionamento de todos os equipamentos da simulação (quando estão ativos é aceso uma luz verde ao lado do nome do mesmo) e também é possível visualizar os valores analógicos “setados” antes do início do funionamento. Existem ainda nessa tela o botão “Voltar” para retornar ao menu de opções e “Sair”, para retornar a tela inicial.

Figura 30 – Tela de alarmes do sistema supervisório.



Fonte: elaborado pelo autor.

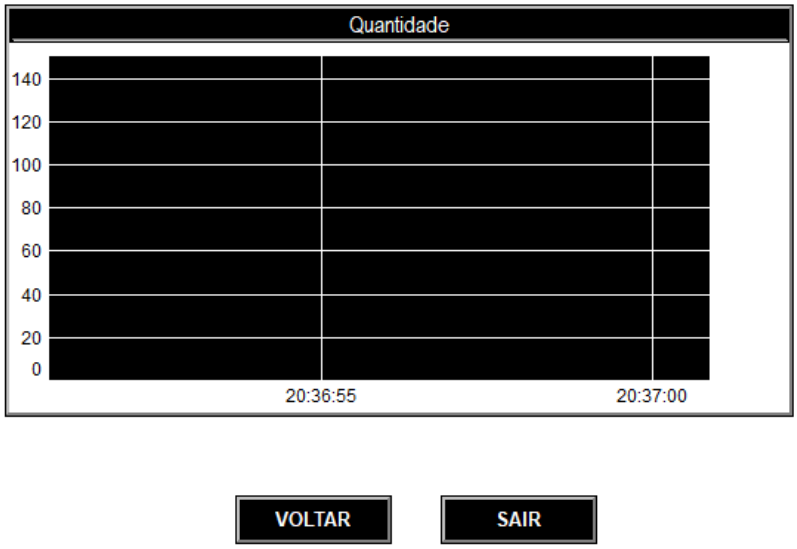
A quarta tela “Figura 30” é referente a visualização de alarmes. Os alarmes são importantes porque mantém os operadores informados de toda a situação da planta de forma simples e objetiva. Foram definidos quatro alarmes nessa simulação, todos eles possuem dois tipos, o “low” quando é uma situação em que os números estão abaixo do correto e o “high” que é exatamente a situação contrária, ou seja quando os números estão acima do correto.

Os quatro alarmes derivam de variáveis análogicas e são elas:

- **Quantidade** – Faixa de operação: 20 a 120 unidades. Alarme “low”: Abaixo de 20 unidades. Alarme “high”: Acima de 120 unidades.
- **Temperatura da Estufa** – Faixa de operação 28º a 35º. Alarme “low”: Abaixo de 28º. Alarme “high”: Acima de 35º.
- **Temperatura do Forno** – Faixa de operação 250º a 300º. Alarme “low”: Abaixo de 250º. Alarme “high”: Acima de 300º.
- **Umidade** – Faixa de operação 60% a 80%. Alarme “low”: Abaixo de 60%. Alarme “high”: Acima de 80%.

Nesta tela então, em uma representação gráfica todos os alarmes são mostrados de acordo com o que foi definido previamente. Como na tela anterior existem os botões de “Voltar” para retornar ao menu de opções e “Sair” para retornar ao início.

Figura 31 – Gráfico de tendencias do sistema supervisório.



Fonte: elaborado pelo autor.

A quinta e última tela “Figura 31” é responsável pelo gráfico de tendências, e assim como as telas anteriores possuem os botões de “Voltar” para retornar ao menu de opções e “Sair” para retornar a tela inicial do programa. O gráfico de tendências representa em tempo real a variação da variável de quantidade de pães produzida durante todo o período do dia e é importante para fazer por exemplo uma comparação que possa determinar os picos de maior funcionamento do sistema.

4. Resultados e Discussões

A fim de cumprir o objetivo proposto, o primeiro passo para a confecção do trabalho foi possuir um simulador de CLP que segue a risca as funcionalidades do mesmo. O Control Win V3, ferramenta presente no software CODESYS foi responsável por proporcionar este simulador fazendo o papel de "transformar" o computador local em um CLP industrial de alto desempenho, garantindo que a simulação tivesse credibilidade.

Para seguir com a simulação seria necessário estabelecer uma comunicação estável entre o CLP e o sistema SCADA. O resultado disso foi possível mais uma vez graças ao CODESYS, além de simular um CLP industrial no computador local proporciona uma interface OPC para a realização da troca de dados entre o cliente e o servidor.

O ultimo pilar essencial para a simulação de fato acontecer é a presença de um bom sistema de supervisão. O uso do software Elipse SCADA trouxe exatamente essa funcionalidade, nele foi possível receber as variáveis do sistema fornecidas pelo cliente e desta forma associa-las aos objetos gráficos corretos, possibilitando a criação uma tela de visualização organizada e eficiente.

Tendo a arquitetura do projeto definida, foi possível observar seu funcionamento da forma com que o sistema foi descrito. O CLP virtual fornecido pelo software CODESYS conseguiu carregar e compilar o código programado em sua interface, fornecer a comunicação OPC e, exportar de forma correta todas as variáveis ou "tags" para o sistema de supervisão. No Elipse SCADA, todas essas "tags" foram recebidas, foi criada a interface gráfica de visualização e desta forma a simulação tornou-se completa.

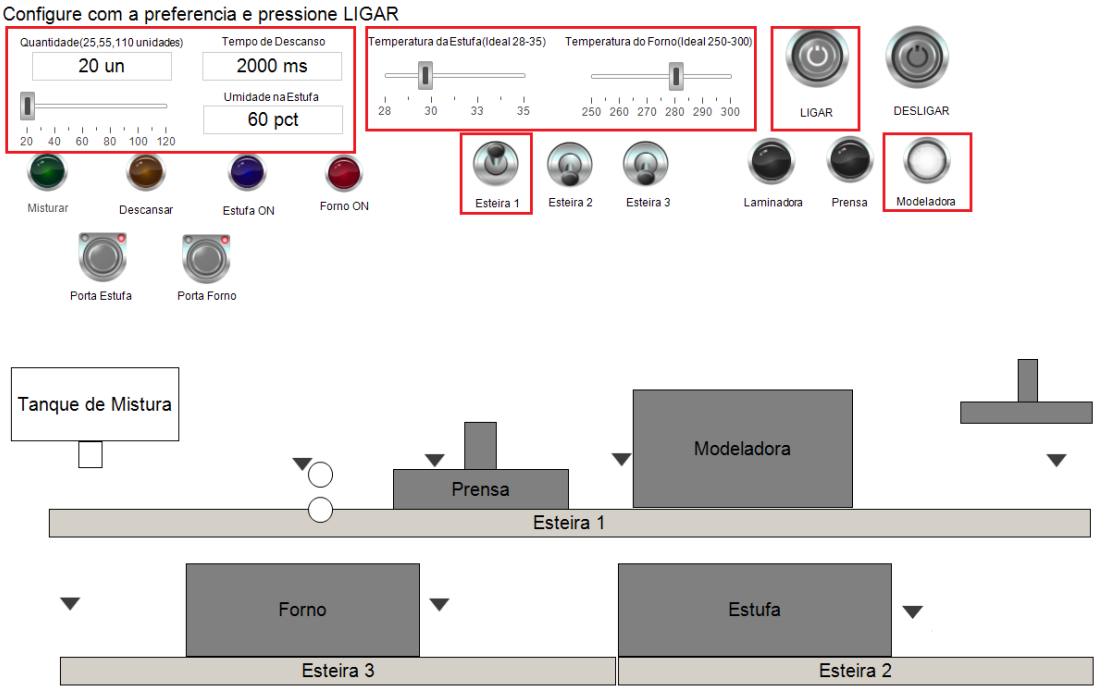
Com o intuito de demonstrar na prática os resultados do sistema passo a passo, foram anexadas algumas figuras. Primeiramente, nas Figuras 32, 33 e 34 é possível observar a sincronização entre o CLP virtual, a tela de visualização do CODESYS e a tela de supervisão do Elipse SCADA, pode-se notar que os valores setados assim como os sensores ativos são exatamente os mesmo nas diferentes plataformas.

Figura 32 – Valores “setados” direto no CLP virtual do CODESYS

Expression	Type	Value	Prepared value
TempEstufa	REAL	30	
TempForno	REAL	280	
Qtd	REAL	20	
Descanso	TIME	T#2s	
Umid	REAL	60	
Iniciar	BOOL	TRUE	
Parar	BOOL	FALSE	
Funcionamento	BOOL	TRUE	
Farinha	BOOL	FALSE	
Agua	BOOL	FALSE	
Fermento	BOOL	FALSE	
Sal	BOOL	FALSE	
Misturador	BOOL	FALSE	
Modeladora	BOOL	TRUE	
Prensa	BOOL	FALSE	
LedVerde	BOOL	FALSE	
LedAmarelo	BOOL	FALSE	
LedVermelho	BOOL	FALSE	
LedAzul	BOOL	FALSE	
Esteira	BOOL	TRUE	
Esteira2	BOOL	FALSE	

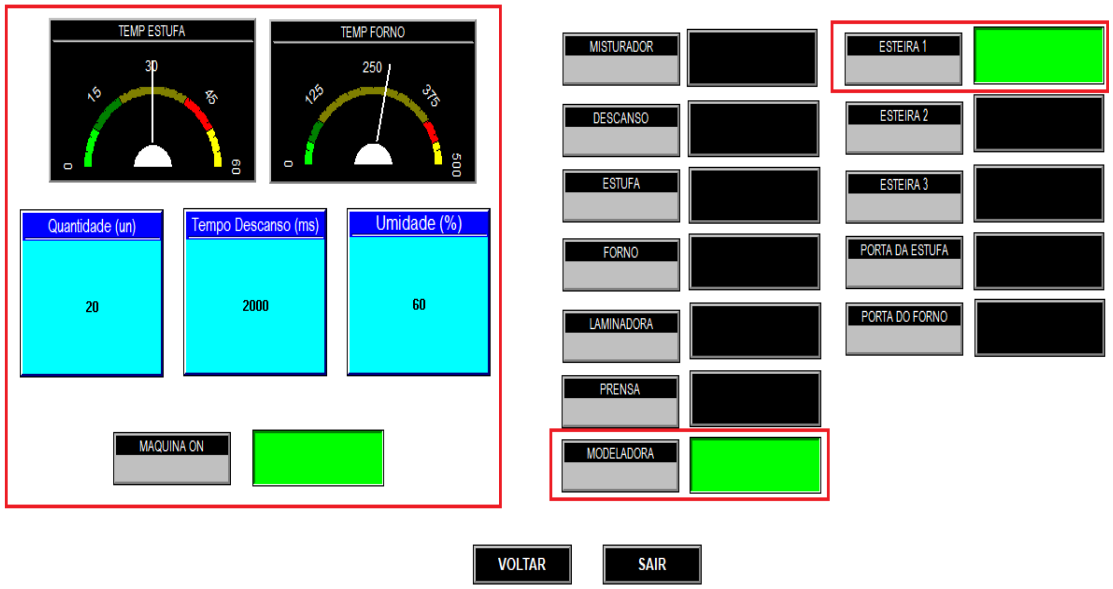
Fonte: *print screen* da tela de configuração de valores no CODESYS.

Figura 33 – Valores “setados” na tela de visualização do CODESYS



Fonte: *print screen* da tela de visualização elaborada pelo autor no CODESYS.

Figura 34 – Valores “setados” na tela de supervisão do Elipse SCADA



Fonte: elaborado pelo autor.

É importante ressaltar que os valores são realmente interativos, ou seja sincronizados em tempo real em qualquer uma dessas tela, isso mostra o dinamismo e qualidade da simulação.

Visto os resultados que garantem a comunicação entre o cliente e servidor pode-se agora explorar o funcionamento da tela de supervisão. Como dito no capítulo de metodologia o sistema supervisório é composto por cinco telas, a primeira delas é responsável pela segurança de acesso pois foi configurada para permitir a entrada apenas pessoas préviamente credenciadas. A segunta tela corresponde ao menu de opções que o usuario possui após acessar o sistema. A terceira tela é a de monitoramento e está exemplificada na “Figura 34”. A quarta tela funcionou como um sistema de alarmes de baixa “low” ou alta “high” para indicar possíveis problemas referentes as variáveis de quantidade, temperatura da estufa, temperatura do forno e umidade. A quinta e ultima tela atuou com o gráfico de tendencias que foi responsável por trazer uma variação da quantidade de pães que foram produzidos a cada operação do sistema ao longo do tempo.

A “Figura 35” demonstra o que acontece ao tentar inserir um usúario e senha que não condizem com o banco de acesso.

Figura 35 – Exemplificação do sistema de acesso

Fonte: elaborado pelo autor.

O resultado obtido acima é derivado de um teste feito ao colocar um usuário e senha não cadastrados, ao fazer isso o sistema retornou uma caixa de erro com a mensagem “Usuário ou senha incorretos”.

Mais uma vez foi possível ver a simulação sendo fiel e se comportando como um sistema real. A autenticação do sistema é de extrema importância para garantir a segurança do processo.

A “Figura 36” demonstra a detecção de alguns alarmes de “low” e “high” após os valores “setados” para a execução do sistema terem sido definido fora das faixas ideais para operação.

Figura 36 – Tela de alarmes do sistema supervisório

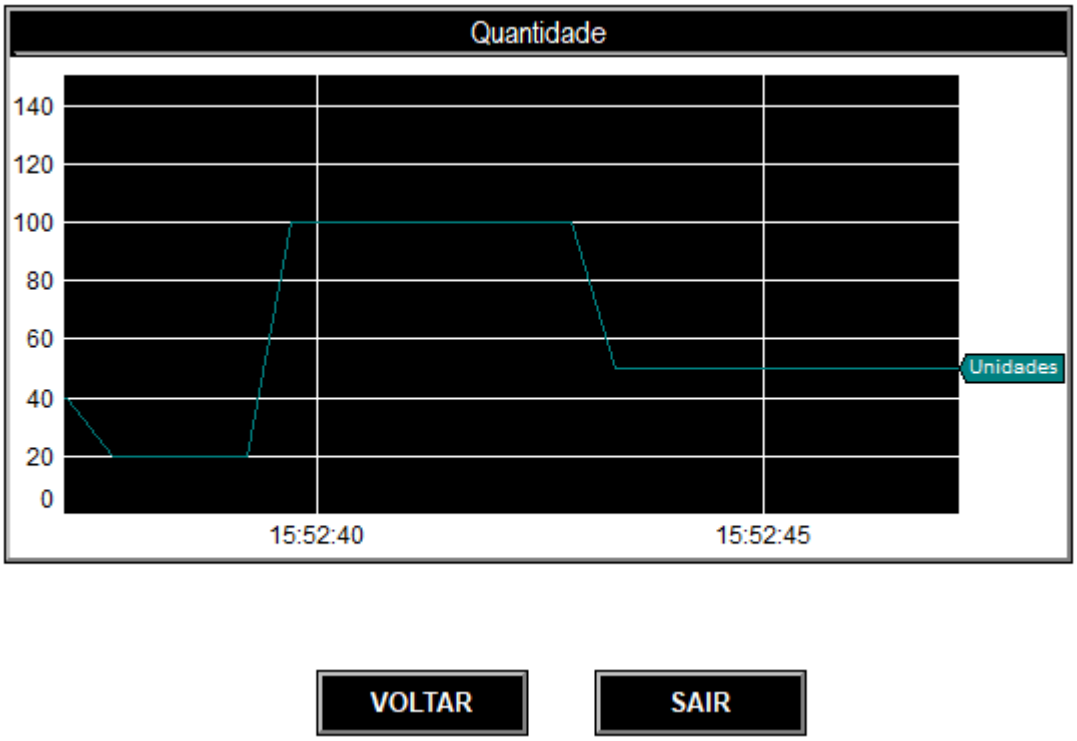
Fonte: elaborado pelo autor.

A simulação mais uma vez foi responsiva, ao “setar” valores fora da faixa pré programada no código do CODESYS o sistema supervisório por meio da comunicação OPC conseguiu identificar esses valores, com isso foi possível configurar os intervalos de alarmes de baixa “low” e alta “high”.

Segundo (MORAES e CASTRUCCI, 2007) os alarmes são importantes numa planta automatizada e tem como princípais características, notificar os operadores, alerta-los a tomar alguma providência, sinalizar falhas, fornecer o estado do funcionamento global do processo, etc.

A “Figura 37” então demonstra a ultima tela, responsável pela funcionalidade do gráfico de tendencias. Implementado para trazer respaldo sobre a quantidade de produção que esta sendo condicionada ao longo do tempo.

Figura 37 – Gráfico de tendencias do sistema supervisório



Fonte: elaborado pelo autor.

Para a realização e análise dos resultados obtidos pelo gráfico de tendencias foi feita uma variação da quantidade produzida em um pequeno periodo de tempo, e como pôde ser observado na imagem o gráfico foi bastante responsivo e exibiu em tempo real as variações de unidade.

Analisando o funcionamento do sistema como um todo, foi observado que em ambos os softwares, tanto no CODESYS responsável pelo CLP virtual , quanto no sistema SCADA do Elipse responsável pela tela de supervisão, obteve-se um bom resultado pois a comunicação entre eles aconteceu de forma correta e em tempo real como proposto anteriormente. Foi possível ver de forma gráfica todos os passos da linha de produção de pães acontecerem, desde o acionamento de um led até a indicação do funcionamento de uma esteira por exemplo.

O estudo feito sobre a linha de produção de pães automatizada trouxe um bom retorno e cumpriu seu papel, que era demonstrar a versátilidade e poder de otimização que a automação tem.

De acordo com (ROSÁRIO, 2012) a automação possui uma grande versátilidade para se adaptar a diferentes condições, fazendo com que as ações do sistema conduzam a resultados ótimos. Além disso, define a automação como sendo um sistema que tende a aumentar a eficiência dos processos.

Utilizando softwares conceituados e precisos a simulação mostrou-se funcional, dando uma prévia do que aconteceria em um ambiente real. Aliás, segundo (GARCIA, 2005) o conceito de simulação está aliado ao resultado temporal das variáves de interesse de um modelo quando as variáveis de entrada tem seus valores definidos, desta forma como "setamos" variáveis de entrada e obtemos uma saída a partir disso temos resultados embasados em uma simulação.

Por mais que esse modelo de projeto tenha sido construido para ser o mais fiel possível com a realidade uma simulação nunca chegará a ter 100% de similariedade com o mundo real, alguns fatores como por exemplo interferência externa, funcionamento dos equipamentos ou ambiente de operação podem interferir nos resultados finais, no entanto ao mesmo tempo tem-se uma certa garantia de lealdade pois apesar de a simulação não cobrir os possíveis problemas físicos sabe-se que o CLP foi projetado parar suprir essas adversidades. Segundo (ROSÁRIO, 2012) o CLP é amplamente utilizado em aplicações industriais pelo fato de ter uma grande capacidade de processamento e funcionamento em tempo real, alem de operar com várias entradas e saídas e poder operar em ambientes hostis pela sua capacidade de suportar grandes variações de temperaturas e até mesmo ruídos elétricos.

O sistema supervisório do projeto também atuou com êxito, demonstrando todas alterações das variáveis previamentes configuradas em suas telas, bem como os alarmes necessários. De acordo com (ROSÁRIO, 2012) o sistema SCADA é uma ferramenta muito utilizada na indústria e melhora muito a eficiência do processo de monitoração e controle. Isso passa uma segurança que o estudo foi bem feito, seguindo o princípio inicial que é chegar o mais próximo da realidade, demonstrando o poder da automação de um sistema.

5. Conclusão

Como foi falado desde o primeiro capítulo deste trabalho a automação se tornou essencial e indispensável devido aos tantos benefícios que ela traz, seja em qual for o ramo que aplicada. Então seguindo essa linha, a pesquisa demonstra a importância e funcionalidade dos sistemas que vieram para ajudar e otimizar os processos de uma aplicação.

Como previamente descrito no capítulo de objetivos essa pesquisa visava contextualizar e expressar de forma eficiente e didática o funcionamento de uma linha de produção industrial e todas as variáveis ao seu redor.

A simulação feita, desde o início da programação até o funcionamento gráfico das telas de visualização, do mais baixo nível até o mais alto nível da hierarquia dos termos técnicos, cumpriu os objetivos gerais e específicos propostos tendo um funcionamento conforme o descrito no projeto.

O objetivo geral que visava desenvolver um simulador aplicado a uma linha de produção de pães foi implementado e seguiu todos os passos propostos. A execução da lógica de programação se mostrou eficiente para a sequencia de funcionamento pré definida, as telas de visualização e supervisão funcionaram de forma responsiva relatando de forma gráfica todo o acontencimento do processo, seus alarmes e tendencias. O uso dos softwares CODESYS e Elipse SCADA para o desenvolvimento do projeto mostraram-se adequados, tendo em vista que a comunicação via OPC pré estabeleciada anteriormente funcionou, garantindo que a simulação tivesse em tempo real a troca de dados entre o cliente e o servidor, cumprindo assim também os objetivos específicos.

O estudo alinhou teoria e prática, citações e comprovações técnicas e, demonstrou como um sistema pode ser automatizado através de uma simulação. Esse tipo de conteúdo mostra-se muito importante para os dias de hoje uma vez que a área de pesquisa só cresce.

A automação ocupa um lugar de destaque atualmente, tanto pelas necessidades quanto pelos constantes avanços tecnológicos, e esse estudo vem para somar com tudo isso, pois além de possibilitar mais conteúdos para a área, abre espaço para que novas pesquisas sejam feitas.

Uma fonte de conhecimento, um estudo, um projeto sempre está aberto para melhoras até porque a tecnologia não para de evoluir. Além do mais em uma área tão ampla como a de automação melhoras acontecem com bastante frequencia.

Como sugestão para trabalhos futuros seria interessante modificar ou criar novas lógicas de programação partindo desta, dando margem para a linha de produção de pães automatizada ser ampliada também para outras areas da industria, visto que o desenvolvimento como um todo foi funcional com algumas alterações seria possível sim ampliar a area de atuação do projeto. Aliar o sistema de supervisão a um sistema gerenciador de banco de dados também é uma ótima sugestão para trabalhos futuros, tendo em vista que desta forma as informações coletadas poderiam ser armazenadas por uma grande quantidade de tempo dando a possíbilidade de fazer maiores análises através de gráficos e indicadores por exemplo.

Referências

BOYER, Stuart A. **SCADA: supervisory control and data acquisition**. International Society of Automation, 2009.

DA SILVA, Ana Paula Gonçalves; SALVADOR, Marcelo. **O que são sistemas supervisórios?**. São Paulo, 2005.

DE MELLO, André Torres Ferraz; BARBOSA, Marcelo Castro. **Transcritor de Redes de Petri para Linguagem de Programação de Máquina**. 2016.

DORF, R. C.; BISHOP, R. H. **Sistemas de Controle Moderno**; 8ª. Ed, LTC, Rio de Janeiro, 2001.

FRANCO, Ivan Carlos et al. **Utilização do protocolo de comunicação OLE for Process Control em processos industriais**. Exacta, v. 8, n. 3, p. 319-329, 2010.

FONSECA, Marcos. De. Oliveira. **Comunicação OPC - Uma abordagem prática**, 09 Outubro 2002. 1-12.

GARCIA, Claudio. **Modelagem e Simulação de Processos Industriais e de Sistemas Eletromecânicos Vol. 1**. [S.l.]: Edusp, 2005.

GONZÁLEZ, Isaías et al. **A literature survey on open platform communications (OPC) applied to advanced industrial environments**. Electronics, v. 8, n. 5, p. 510, 2019.

KATO, Edilson. Reis. Rodrigues.; PORTO, Arthur. José. Vieira. **Associação De Redes De Petri E Linguagens De Programação CLP Para O Projeto Do Controle De Sistemas De Manufatura Discreta**. 4º. SBAI - Simpósio Brasileiro de Automação Inteligente, São Paulo, 08-10 Setembro 1999.

LAMB, Frank. **Automação Industrial na Prática-Série Tekne**. [S.l.]: AMGH, 2015.

MORAES, C. C. D.; CASTRUCCI, P. D. L. **Engenharia de Automação Industrial 2º Edição**. São Paulo: LTC, 2007.

LEITNER, Stefan-Helmut; MAHNKE, Wolfgang. **OPC UA–service-oriented architecture for industrial applications**. ABB Corporate Research Center, v. 48, p. 61-66, 2006.

NATALE, Ferdinando. **Automação Industrial-Série Brasileira de Tecnologia**. Saraiva Educação SA, 2018.

PRUDENTE, Francesco. **Automação Industrial PLC:** Teoria e Aplicações 2º Edição. Milão: LTC, 2011.

RIBEIRO, Marco. Antonio. **Automação Industrial 4º Edição**. Salvador: Tek Treinamento & Consultoria Ltda, 2001.

ROSÁRIO, João Maurício. **Automação industrial**. Editora Baraúna, 2012.

SIEMENS. New Siemens. **Siemens**, 09 set. 2020. Disponivel em: <https://new.siemens.com/br/pt/produtos/automacao/sistemas-automacao/industrial/plc/s7-1200.html>.

VIANA, F. L. E. **Indústria de alimentos**. [S.l.]: Banco do Nordeste do Brasil, 2019.