

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CAMPUS CORNÉLIO PROCÓPIO
CURSO DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA

RENAN MAINARDES

**INTEGRAÇÃO DE INVERSORES DE FREQUÊNCIA NA
ARQUITETURA DE UM SISTEMA DIGITAL DE CONTROLE
DISTRIBUÍDO**

TRABALHO DE CONCLUSÃO DE CURSO

CORNÉLIO PROCÓPIO
2016

RENAN MAINARDES

**INTEGRAÇÃO DE INVERSORES DE FREQUÊNCIA NA
ARQUITETURA DE UM SISTEMA DIGITAL DE CONTROLE
DISTRIBUÍDO**

Trabalho de Conclusão de Curso de graduação, apresentado como um dos pré-requisitos ao curso de Engenharia Elétrica da Universidade Tecnológica Federal do Paraná para obtenção do título de Engenheiro Eletricista.

Orientador: Prof. Dr. Wagner Endo

CORNÉLIO PROCÓPIO
2016



Universidade Tecnológica Federal do Paraná
Campus Cornélio Procópio
Departamento de Engenharia Elétrica
Curso de Engenharia Elétrica



FOLHA DE APROVAÇÃO

Renan Mainardes

Integração de inversores de frequência na arquitetura de um sistema digital de controle distribuído

Trabalho de conclusão de curso apresentado às 16:00hs do dia 10/06/2016 como requisito parcial para a obtenção do título de Engenheiro Eletricista no programa de Graduação em Engenharia Elétrica da Universidade Tecnológica Federal do Paraná. O candidato foi arguido pela Banca Avaliadora composta pelos professores abaixo assinados. Após deliberação, a Banca Avaliadora considerou o trabalho aprovado.

Prof(a). Dr(a). Wagner Endo - Presidente (Orientador)

Prof(a). Dr(a). Cristiano Marcos Agulhari - (Membro)

Prof(a). Dr(a). Rodrigo Rodrigues Sumar - (Membro)

AGRADECIMENTOS

Aos meus pais Regina e Francisco, pelo constante incentivo e extrema dedicação, sempre me ensinando o valor do estudo e me guiando pelo melhor caminho possível.

Ao Prof. Dr. Wagner Endo pela prontidão e conselhos valiosos durante a orientação deste trabalho.

Aos Professores Dr. Marcos Banheti Rabello Vallim e Dr. Rodrigo Rodrigues Sumar pelas imprescindíveis dicas de correção à proposta inicial deste trabalho.

Aos alunos de iniciação científica Rodrigo Pita Rolle e Lucas Belório de Castro pelo apoio e prontidão.

A minha namorada Ana Paula Gonçalves pelo apoio, incentivo e companheirismo.

Aos colegas do curso de graduação em Engenharia Elétrica e todos aqueles que direta ou indiretamente colaboraram com o desenvolvimento deste trabalho.

RESUMO

MAINARDES, Renan. INTEGRAÇÃO DE INVERSORES DE FREQUÊNCIA NA ARQUITETURA DE UM SISTEMA DIGITAL DE CONTROLE DISTRIBUÍDO. 76 f. Trabalho de Conclusão de Curso - Curso de Graduação em Engenharia Elétrica, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2016.

Este trabalho propõe a integração de inversores de frequência na arquitetura de um sistema digital de controle distribuído. Para se atingir tal objetivo é construída uma bancada com intuito de representar a arquitetura por meio de um sistema supervisão, inversores de frequência e dispositivos de campo. Toda aquisição de dados feita pelo sistema supervisão utiliza o protocolo *Modbus* RTU. Após o detalhamento do projeto da bancada, realiza-se três testes para validar a integração proposta. Esses testes visam demonstrar o funcionamento da comunicação serial, do regulador PID e da função *soft*PLC do inversor de frequência. Posteriormente é realizada uma discussão sobre possíveis aplicações industriais e didáticas da arquitetura montada, bem como sugestões para trabalhos futuros.

Palavras-chave: Bancada didática; Modbus; OPC; Redes industriais; Sistema supervisão.

ABSTRACT

MAINARDES, Renan. INTEGRATION OF FREQUENCY INVERTERS IN THE ARCHITECTURE OF A DISTRIBUTED CONTROL SYSTEM. 76 f. Trabalho de Conclusão de Curso - Curso de Graduação em Engenharia Elétrica, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2016.

This work proposes the integration of frequency inverters in the architecture of a distributed control system. In order to achieve this objective a stand is built with the purpose of representing the architecture by using a supervisory system, frequency inverters and field devices. All the data acquisition by the supervisory system are made using Modbus RTU protocol. After the didactic stand project detailing, three tests are made in order to validate the proposed integration. These tests demonstrates the operation of the serial communication, the PID regulator and the softPLC function of the frequency inverter. After that, a discussion about possible industrial and didactic applications is made, as well as future works suggestions.

Keywords: Didactic stand; Modbus; OPC; Industrial Networks; Supervisory System.

LISTA DE QUADROS

Quadro 1 - Benefícios do uso do CLP em sistemas de controle.	20
Quadro 2 - Lista de materiais e equipamentos utilizados na bancada.	37
Quadro 3 - Equipamentos utilizados na bancada para o primeiro teste.	41
Quadro 4 - Classificação da rede utilizada no primeiro teste.	42
Quadro 5 - Leitura das variáveis em cada inversor pelo software SuperDrive G2 do fabricante WEG.	43
Quadro 6 - Aplicações do inversor de frequência com a função softPLC.	50

LISTA DE FIGURAS

Figura 1 - Arquitetura da automação industrial.....	13
Figura 2 - Sensores e atuadores trabalham em conjunto para controlar um processo industrial por meio da lógica de controle executada pelo CLP.	19
Figura 3 – <i>Hardware</i> de um CLP.....	22
Figura 4 - Ciclo de varredura de um CLP.....	23
Figura 5 - Classificação das linguagens de programação da norma IEC 61131-3....	24
Figura 6 - Sistema supervisório SCADA representado por um servidor comunicando-se com quatro CLP's.	27
Figura 7 - Sistema supervisório SCADA interligado representado por um servidor transmitindo os dados dos quatro CLP's para dois ou mais clientes.....	28
Figura 8 - Controle centralizado.	29
Figura 9 - Barramento de campo.....	30
Figura 10 - Barramento de campo distribuído.	30
Figura 11 - Sistema digital de controle distribuído.....	31
Figura 12 - Arquitetura proposta neste trabalho. Nesta arquitetura, têm-se unidades de monitoração, controle e dispositivos.	34
Figura 13 - Desenho dimensional da bancada que irá simular a arquitetura proposta.	35
Figura 14 - Diagrama de ligações da bancada.....	36
Figura 15 - Módulo de comunicação serial, indispensável para a integração proposta.	36
Figura 16 - Bancada que irá simular a arquitetura proposta construída em estrutura de metalon.....	37
Figura 17 - Interface gráfica do sistema supervisório.....	39
Figura 18 - Arquitetura proposta no primeiro teste.....	40
Figura 19 - Montagem da arquitetura do primeiro teste.	40
Figura 20 - Ligação do transdutor de pressão ao inversor de frequência com 2 fios.	41
Figura 21 - Leitura das variáveis em cada inversor pelo sistema supervisório via rede Modbus RTU.....	43
Figura 22 - Leitura das variáveis em cada inversor pela aplicação cliente via OPC.	44

Figura 23 - Arquitetura proposta no segundo teste.	45
Figura 24 - Montagem da arquitetura do segundo teste.....	46
Figura 25 - Ligação do transdutor de pressão ao inversor de frequência para o segundo teste.....	46
Figura 26 - Gráfico de resposta da velocidade do motor em função da variável de processo com setpoint de 50%.	47
Figura 27 - Gráfico de resposta da velocidade do motor em função da variável de processo com setpoint de 25%.	48
Figura 28 - Linguagem ladder desenvolvida para o teste da função “softPLC”.	49

LISTA DE SIGLAS

CLP	Controlador Lógico Programável
CPU	<i>Central Processing Unit</i>
DC	<i>Direct Current</i>
DDC	<i>Direct Digital Control</i>
DP	<i>Decentralized Periphery</i>
FBD	<i>Function Block Diagram</i>
FMS	<i>Field Message Specification</i>
IEC	<i>International Electrotechnical Commission</i>
IHM	Interface Homem-Máquina
IL	<i>Instruction List</i>
LD	<i>Ladder Diagram</i>
OLE	<i>Object Linking and Embedding</i>
OPC	<i>OLE – Object Linking and Embedding – Process Control</i>
PA	<i>Process Automation</i>
PID	Proporcional Integral Derivativo
RAM	<i>Random Access Memory</i>
RS232	<i>Recommended Standard 232</i>
RS485	<i>Recommended Standard 485</i>
RTU	<i>Remote Terminal Unit</i>
SCADA	<i>Supervisory Control and Data Acquisition</i>
SDCD	Sistema Digital de Controle Distribuído
SFC	<i>Sequential Function Chart</i>
ST	<i>Structured Text</i>

SUMÁRIO

1 INTRODUÇÃO	12
1.1 OBJETIVOS	14
1.1.1 Objetivo geral	14
1.1.2 Objetivos específicos.....	14
2 FUNDAMENTAÇÃO TEÓRICA	16
2.1 INVERSORES DE FREQUÊNCIA	16
2.1.1 Parâmetros do inversor	16
2.2 SENSOREAMENTO	17
2.3 CONTROLADORES LÓGICOS PROGRAMÁVEIS	17
2.3.1 Utilização em sistemas de controle	18
2.3.2 <i>Hardware</i> e princípio de funcionamento	21
2.3.3 Linguagens de programação.....	23
2.4 SISTEMAS SUPERVISÓRIOS.....	25
2.4.1 Características de sistemas supervisórios	26
2.4.2 SCADA	26
2.5 REDES DE COMUNICAÇÃO INDUSTRIAL.....	28
2.5.1 Sistemas centralizados e distribuídos	29
2.5.2 Parâmetros de uma rede de comunicação industrial	31
2.6 PROTOCOLOS DE COMUNICAÇÃO INDUSTRIAL.....	32
2.6.1 MODBUS.....	33
2.6.2 OPC.....	33
3 PROJETO E IMPLEMENTAÇÃO DA ARQUITETURA DO SDCD	34
3.1 DEFINIÇÃO DA ARQUITETURA	34
3.2 PROJETO E IMPLEMENTAÇÃO	35
4 RESULTADOS E ENSAIOS DE IMPLEMENTAÇÃO DO SDCD	38
4.1 TESTE DE COMUNICAÇÃO SERIAL	39
4.2 TESTE DO REGULADOR PID.....	44
4.3 TESTE DE APLICAÇÃO DO CLP INTEGRADO.....	48
5 CONSIDERAÇÕES FINAIS	51
REFERÊNCIAS	52
ANEXO A - PARAMETRIZAÇÃO DOS INVERSORES DE FREQUÊNCIA	54

ANEXO B - CONFIGURAÇÃO DO SISTEMA SUPERVISÓRIO (SERVIDOR).....	60
ANEXO C - CONFIGURAÇÃO DO SISTEMA SUPERVISÓRIO (CLIENTE)	70

1 INTRODUÇÃO

Atualmente muitos sistemas e técnicas tem sido desenvolvidas para o controle de operação, supervisão e gerenciamento de processos industriais. É neste contexto que fica evidenciada a importância da automação industrial (LOPEZ, 2000).

Entende-se por automação industrial qualquer sistema, apoiado em computadores, que substitua o trabalho humano em favor da segurança das pessoas, da qualidade dos produtos, da rapidez da produção e da redução de custos podendo, desta maneira, aperfeiçoar os complexos objetivos das indústrias e dos serviços. A automação industrial envolve a implementação de sistemas interligados baseados em redes de comunicação, compreendendo também sistemas supervisórios que facilitam a análise de problemas que porventura venham a ocorrer (MORAES e CASTRUCCI, 2007).

No entanto, em meados de 1990, o cenário da automação industrial era diferente. Os sistemas convencionais para a conexão de sensores e atuadores ao controlador apresentavam um custo elevado, inflexibilidade para expansões e maior demanda de tempo para as fases de projeto, instalação e manutenção. A solução encontrada para estes problemas foi a integração dos dispositivos em rede serial através dos denominados sistemas de rede industrial (LOPEZ, 2000).

A preocupação com a integração evoluiu tanto que associações técnicas normativas foram criadas para oferecer soluções padronizadas para a comunicação e troca de dados entre equipamentos de diferentes fabricantes. Esta é a razão pela qual pode-se encontrar, atualmente, dezenas de padrões e protocolos de comunicação abertos para realizar a integração e controle de dispositivos distribuídos por toda a arquitetura da automação industrial (Figura 1) (CAPELLI, 2006).

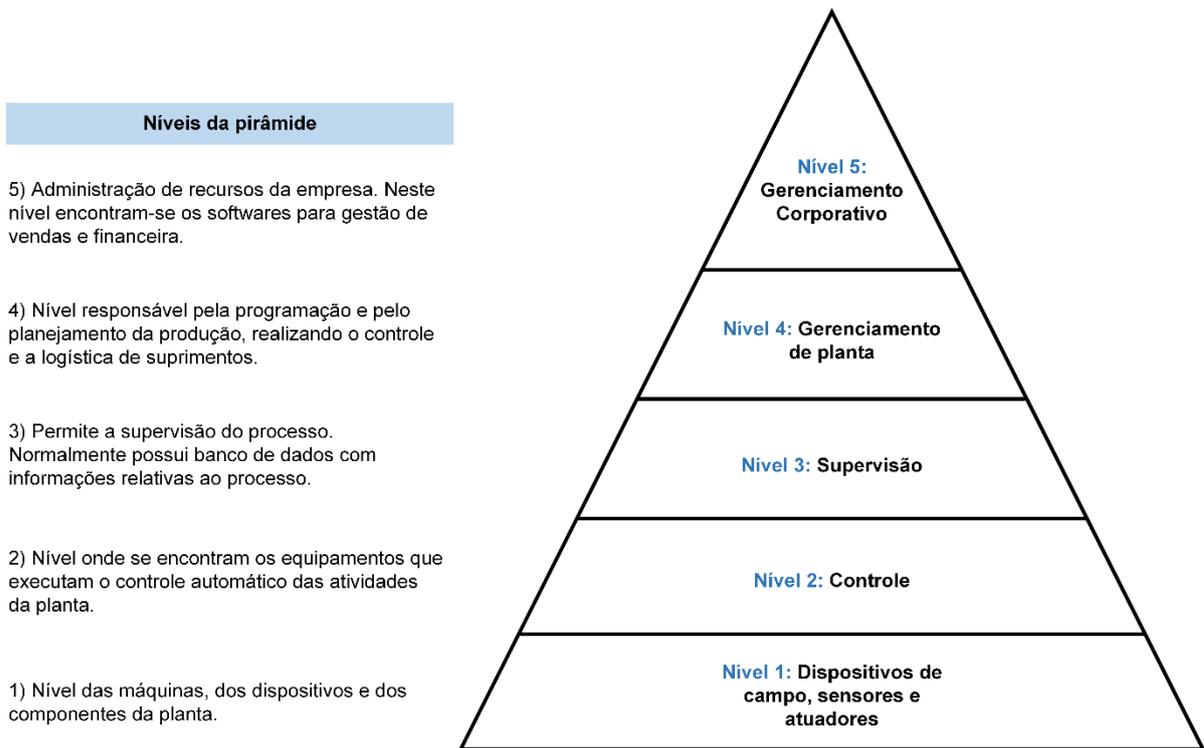


Figura 1 - Arquitetura da automação industrial.
Fonte: Adaptado de Moraes e Castrucci (2007)

Neste contexto, um exemplo de elemento final de controle (base da pirâmide) são os motores de indução trifásicos que acionam bombas, atuadores, dentre outros elementos do chão de fábrica. Este tipo de motor é amplamente utilizado em aplicações industriais e as razões de sua popularidade incluem simplicidade, confiabilidade e baixo custo (EL-HAWARY, 2002).

O acionamento e controle destes motores pode ser feito por meio de inversores de frequência, também utilizados para coletar dados de sensores e para comunicação com sistemas supervisórios.

O problema, no entanto, consiste em se encontrar meios de interligar física e logicamente esses equipamentos em uma arquitetura robusta e flexível mesmo com a variedade de tecnologias e fabricantes existentes.

Visando resolver tal problema este trabalho propõe a integração de inversores de frequência na arquitetura de um sistema digital de controle distribuído, ilustrando a conexão entre os três primeiros níveis da arquitetura da automação industrial.

O uso e integração de inversores de frequência em uma arquitetura se justifica devido as vantagens que esses equipamentos possuem em relação a outros tipos de acionamento de motores. Capelli (2006) descreve essas vantagens como:

a) Otimização do processo: O inversor contribui para a redução de perdas e consumo de material na produção.

b) Suavização de operação das máquinas: O número de partidas e paradas bruscas é sensivelmente reduzido. Através de rampas de aceleração e frenagem o impacto direto sobre os componentes mecânicos é evitado ao máximo, o que aumenta a vida útil do equipamento.

c) Menor manutenção: Toda a tecnologia em corrente alternada dispensa manutenção preventiva. Como não há comutação entre escovas e coletor, típica dos motores de corrente contínua, a vida útil do sistema é muito maior.

d) Economia de energia elétrica: O inversor economiza energia elétrica. Dois clássicos exemplos são bombas e ventiladores, em que o consumo é reduzido ao cubo. Um motor desse tipo, quando ligado a um inversor a meia velocidade, gasta apenas 12,5% do que consumiria se estivesse ligado à rede elétrica diretamente.

1.1 OBJETIVOS

1.1.1 Objetivo geral

Este trabalho tem como objetivo integrar inversores de frequência na arquitetura de um sistema digital de controle distribuído.

1.1.2 Objetivos específicos

Para se atingir o objetivo geral, são propostos os seguintes objetivos específicos:

- Estudar os equipamentos e protocolos de comunicação;

- Construir uma bancada para representar a arquitetura de um sistema digital de controle distribuído;
- Realizar três testes para validar a integração proposta.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão abordados os fundamentos teóricos que serão utilizados na proposta apresentada neste trabalho, como uma breve revisão sobre inversores de frequência, sensoreamento, controladores lógicos programáveis, sistemas supervisórios, redes e protocolos de comunicação industrial.

2.1 INVERSORES DE FREQUÊNCIA

São chamados de inversores de frequência os dispositivos eletrônicos capazes de controlar a variação de velocidade de motores de indução. Através deste dispositivo, é possível controlar e monitorar parâmetros do motor como tensão, corrente, torque, tempo de aceleração e desaceleração (MASCHERONI; LICHTBLAU; GERARDI, 2009)

O inversor de frequência é um dos principais dispositivos da automação industrial e seu uso tem contribuído consideravelmente para a otimização de plantas fabris (CAPELLI, 2006).

2.1.1 Parâmetros do inversor

Ferreira (2012) explica que para um inversor funcionar de forma adequada é necessário informá-lo em que condições de trabalho ele irá operar e isso é feito através de sua parametrização. Segundo o autor, os parâmetros de um inversor podem ser classificados da seguinte maneira:

a) Parâmetros de leitura: são variáveis que podem ser visualizadas no *display*, onde são exibidos os valores programados nos parâmetros de regulação, configuração e do motor;

b) Parâmetros de regulação: consistem em valores ajustáveis a serem utilizados pelas funções do inversor e como exemplo pode-se citar os tempos de aceleração e desaceleração;

c) Parâmetros de configuração: definem as características do inversor, bem como as funções que serão executadas. Como exemplo pode-se citar o tipo de controle que será realizado (escalar ou vetorial).

d) Parâmetros do motor: referem-se as características nominais do motor que será acionado pelo inversor como, por exemplo, tensão, corrente, velocidade, frequência e potência nominais.

2.2 SENSOREAMENTO

Sensores são dispositivos que transformam variáveis físicas, como posição, velocidade, temperatura, pressão, entre outros, em variáveis convenientes. Esses dispositivos constituem a base da automação, seja ela industrial, predial ou comercial (CAPELLI, 2009).

Para a automação, o principal objetivo é comandar eventos baseados na leitura desses sensores. Existem aqueles em que a amplitude do sinal elétrico de saída reproduz a amplitude do sinal de entrada, são os sensores de medição ou transdutores, fundamentais no campo do controle dinâmico dos processos (MORAES; CASTRUCCI, 2007).

2.3 CONTROLADORES LÓGICOS PROGRAMÁVEIS

Um controlador lógico programável (CLP) pode ser definido como um dispositivo capaz de armazenar instruções para implementação de funções de controle (sequências lógicas, temporização e contagem, por exemplo), além de realizar operações lógicas, aritméticas, manipulação de dados e comunicação em rede (GEORGINI, 2000).

Os CLPs foram inicialmente desenvolvidos para atender as exigências da General Motors em 1968, substituindo os circuitos de relé utilizados na época. Os maiores requisitos eram que esses equipamentos fossem facilmente programados e reprogramados, de fácil manutenção, menores e mais baratos que os circuitos de relés que iriam substituir e capazes de se comunicar com sistemas de coleta de dados (GALLOWAY; HANCKE, 2013).

Prudente (2007) salienta que “naqueles anos a indústria automobilística suspendia a atividade produtiva toda vez que mudava a produção de um modelo de montagem do mesmo tipo de automóvel”. O autor explica que desta maneira, mudar o modelo de peça a ser produzida significava mudar todos os quadros elétricos, gerando elevados custos para a empresa.

É nesse contexto que a introdução dos CLPs na indústria se tornou tão importante, pois seu uso resultou não somente na redução de custos, mas na redução do tempo de parada de máquinas e no aumento da competitividade. Com o passar do tempo, o CLP transformou-se em um equipamento mais simples de usar, mais confiável e eficiente, tornando-se essencial em vários processos produtivos (CAPELLI, 2006).

2.3.1 Utilização em sistemas de controle

Franchi e Camargo (2013) explicam que, apesar de tamanhos e complexidades diferentes, os sistemas de controle (Figura 2) podem ser divididos em três partes com funções bem definidas:

a) Sensores/transdutores: transdutor é um dispositivo que converte uma condição física do elemento sensor em um sinal elétrico para ser utilizado pelo CLP através da conexão às suas entradas.

b) Atuadores: sua função é converter o sinal elétrico proveniente do CLP em uma condição física, normalmente ligando ou desligando algum dispositivo.

c) Controladores: de acordo com os estados das suas entradas, o controlador utiliza um programa de controle para calcular os estados das suas saídas.

Características do sistema com CLP	Benefícios
Uso de componentes de estado sólido	- Alta confiabilidade
Memória programável	- Simplifica mudanças - Flexibiliza o controle
Pequeno tamanho	- Necessita de um espaço mínimo para instalação
Microprocessador	- Capacidade de comunicação - Alto nível de performance - Alta qualidade dos produtos - Possibilidade de trabalhar com várias funções simultaneamente
Contadores/temporizadores via software	- Facilidade para alterar presets - Elimina hardware
Controle de relés via software	- Reduz custo em hardware/cabeamento - Redução de espaço
Arquitetura modular	- Flexibilidade para instalação - Redução de custos de hardware - Expansibilidade
Variiedade de interfaces de I/O	- Controle de uma grande variedade de I/O - Elimina um controle dedicado
Estações remotas de I/O	- Elimina cabeamentos longos
Indicadores de diagnóstico	- Reduz tempo de manutenção - Sinaliza a operação correta/incorrecta do sistema de controle
Interfaces modulares de I/O	- Facilita a manutenção - Facilita o cabeamento
Variáveis de sistema alocadas na memória de dados	- Facilita gerenciamento/manutenção - Podem ser colocadas na forma de um relatório de saída

Quadro 1 - Benefícios do uso do CLP em sistemas de controle.
Fonte: Adaptado de Franchi e Camargo (2013)

2.3.2 Hardware e princípio de funcionamento

Os CLPs são projetados e construídos para operarem em ambientes severos devendo, portanto, resistir a altas temperaturas, ruídos elétricos, poluição atmosférica e ambientes úmidos (FRANCHI; CAMARGO, 2013).

De acordo com Prudente (2007), o *hardware* (equipamento físico) de um CLP (Figura 3) é constituído basicamente de três unidades fundamentais:

a) Unidade central: é a unidade do CLP que organiza todas as funções de controle, sendo subdividida em CPU (*Central Processing Unit*), memórias e fonte de alimentação.

a.1) CPU: lê os sinais de entrada, carregando as informações provenientes de um botão, interruptor, sensor e/ou transdutor. Caso exista uma variação de sinal, a CPU reage elaborando novos dados segundo a lógica interna do programa e gerando os sinais na amplitude certa de saída. Este sinal de saída comanda um dispositivo atuador, permitindo o acionamento de diversos equipamentos.

a.2) Memórias: como citado anteriormente a CPU de um CLP precisa ler e escrever informações sendo que os dispositivos que permitem tais operações são as memórias, que as executam por meio de circuitos integrados. As principais características dos tipos de memória são a possibilidade de ler e escrever, velocidade de escrita, modalidade de cancelar, comportamento em caso de falta de alimentação elétrica e quantidade de informação memorizável (capacidade). Quando se desenvolve uma aplicação industrial com um CLP, a memória necessária depende dos números das entradas, saídas e da complexidade do diagrama de blocos de funcionamento do sistema controlado.

a.3) Fonte de alimentação: a fonte de alimentação é utilizada para converter uma tensão alternada da rede (geralmente 220 V) em uma tensão contínua (geralmente 12/24 V) para o funcionamento da CPU e dos dispositivos de entrada (sensor, transdutor, botão, etc.). A fonte de alimentação do CLP pode ser integrada ou ser externa ao CLP.

b) Unidade de entrada e saída: esta unidade é composta de dispositivos apropriados de modo a permitir o diálogo do CLP com os sinais de entrada e saídas provenientes do campo. A unidade de entrada opera sob o sinal de entrada proveniente dos sensores da instalação para torna-la compatível com a CPU do

CLP. Já a unidade de saída, trata os sinais emitidos pela CPU para enviá-los aos dispositivos que devem ser acionados, chamados atuadores (motores elétricos, lâmpadas, válvulas pneumáticas, entre outros).

c) Unidade de programação: É a interface entre homem/máquina. É, basicamente, o dispositivo que permite escrever o programa na memória do CLP.

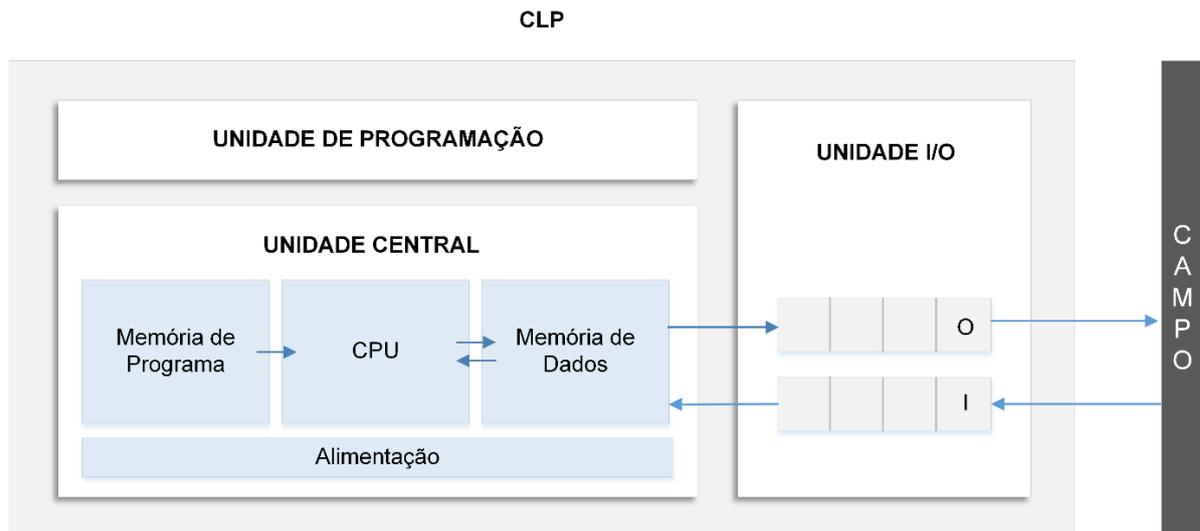


Figura 3 – Hardware de um CLP.
Fonte: Adaptado de Prudente (2007)

Em relação ao seu funcionamento, um CLP realiza suas operações em ordem sequencial, à qual se dá o nome de ciclo de *scan* ou varredura (FRANCHI; CAMARGO, 2013). Esse ciclo é dividido em 3 etapas, como mostrado na Figura 4.

Na primeira etapa a CPU executa a leitura dos status (condições, estados) dos dispositivos de entrada por meio dos circuitos de entrada e saída. Em seguida, na segunda etapa, esses status são armazenados na memória RAM (*Random Access Memory*) para serem processados pelo programa de aplicação (desenvolvido pelo usuário). Após a execução do programa de aplicação, o processador atualiza os status dos dispositivos de saída (terceira etapa), realizando a lógica de controle (GEORGINI, 2000).

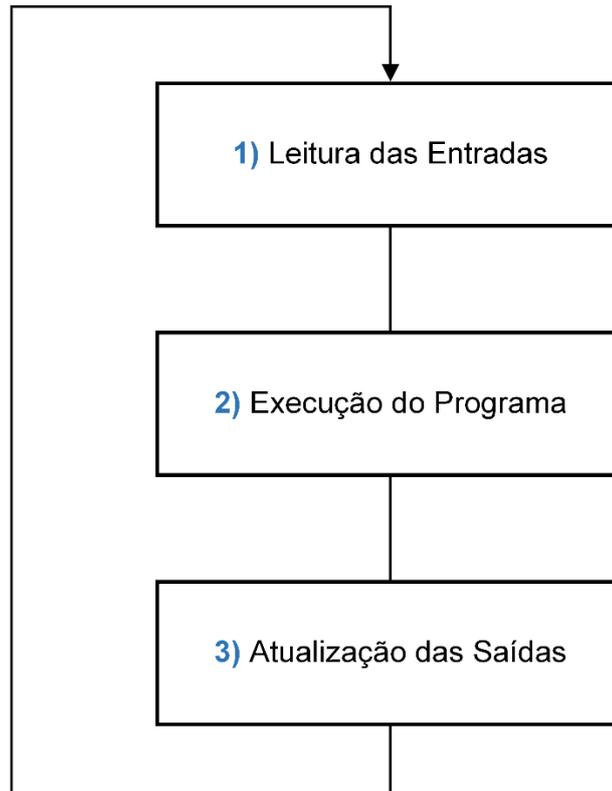


Figura 4 - Ciclo de varredura de um CLP.
Fonte: Franchi e Camargo (2013)

2.3.3 Linguagens de programação

Franchi e Camargo (2013) salientam que para se compreender os tipos de linguagens de programação utilizadas em um CLP, é necessário primeiro esclarecer alguns termos utilizados como:

a) Linguagem de programação: é um conjunto padronizado de instruções que um sistema computacional é capaz de reconhecer.

b) Programar: é o ato de fornecer uma série de instruções a um sistema com capacidade computacional, de maneira que este seja capaz de comportar-se deterministicamente, executando de forma automática as decisões de controle em função do estado atual, das entradas e das saídas do sistema num dado instante.

c) Programador: é a pessoa responsável por prever as situações possíveis do sistema, planejar uma estratégia de controle e codificar as instruções em uma linguagem de programação padronizada para posteriormente serem passadas ao sistema computacional.

Em relação ao seu histórico, até o início de 1990 não existia uma técnica padrão de programação de CLPs, sendo que Base, Forthram, C e várias outras linguagens eram utilizadas para esta finalidade. O fato dos fabricantes utilizarem linguagens diferentes acarretava em desperdício de tempo, alto custo em treinamento e falta de integração entre sistemas (CAPELLI, 2009).

Com base nesse problema, a comunidade industrial internacional reconheceu que era necessário estabelecer um padrão aberto para os CLPs, visando a uniformização de procedimentos dos diversos fabricantes. Para isso, foi criado um grupo de trabalho na IEC (*International Electrotechnical Commission*) para estabelecer normas sobre o desenvolvimento de CLPs, incluindo projeto de hardware, instalação, testes documentação, programação e comunicação (FRANCHI; CAMARGO, 2013)

A norma de referência para a programação dos CLPs é a IEC 61131-3, que resume todas as modalidades de programação do CLP. Essas modalidades de programação podem ser classificadas em duas grandes categorias (PRUDENTE, 2007), como mostra a Figura 5.

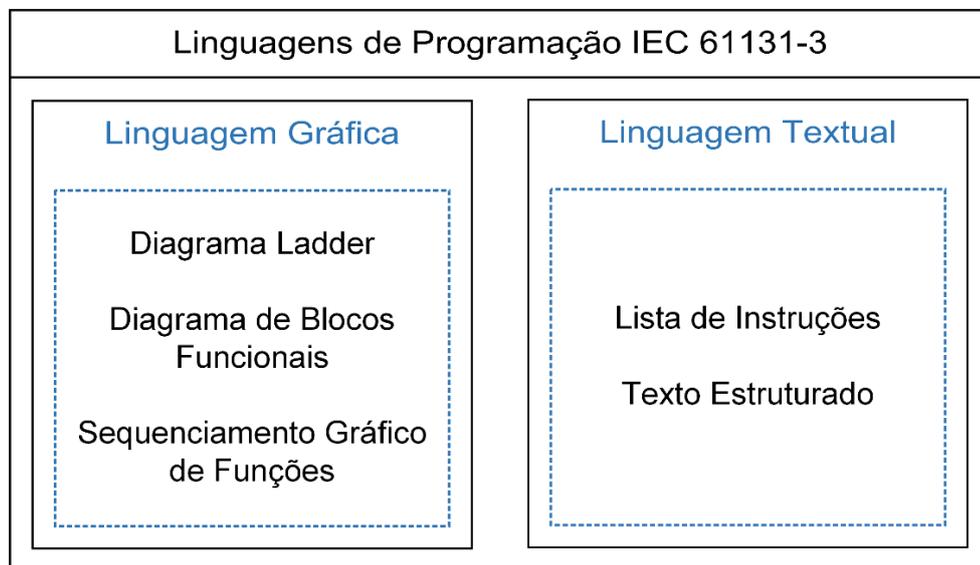


Figura 5 - Classificação das linguagens de programação da norma IEC 61131-3.
 Fonte: Adaptado de Prudente (2007)

Prudente (2007) distingue as linguagens de programação basicamente da seguinte maneira:

a) Diagrama Ladder – LD (*Ladder Diagram*): esta é a linguagem de programação mais utilizada, sendo semelhante a um esquema elétrico funcional.

b) Diagrama de Blocos Funcionais – FBD (*Function Block Diagram*): é uma linguagem que consiste em portas lógicas, permitindo que seja desenhado um esquema clássico de eletrônica digital, sendo altamente utilizado em sistemas de controle de grande instalação de processos.

c) Sequenciamento Gráfico de Funções – SFC (*Sequential Function Chart*): é uma linguagem desenvolvida na França com o nome de linguagem *Grafcet*. Representa o funcionamento por passos de um processo automático, similar a um diagrama de blocos.

d) Lista de instruções – IL (*Instruction List*): esse tipo de linguagem utiliza diretamente as instruções do microcomputador. Tal sistema é pouco intuitivo e pouco prático, não possui nenhuma representação gráfica e requer muito tempo do programador para a pesquisa de falhas no programa.

e) Texto estruturado – ST (*Structured Text*): essa linguagem de programação é de alto nível como as linguagens Pascal, Basic e C++ sendo, às vezes, indispensável em determinada aplicação ou em redes de comunicação.

2.4 SISTEMAS SUPERVISÓRIOS

Tarefas de controle e supervisão de plantas industriais requerem o uso de esquemas de automação sofisticados que devem garantir acesso aos dados de produção através de grandes distâncias, pelos vários níveis da automação. Essas tarefas podem ser realizadas pelo uso dos chamados sistemas supervisórios (BEJAN; IACOB; ANDREESCU, 2009).

Pode-se definir sistemas supervisórios como sistemas digitais de monitoração e operação da planta que gerenciam variáveis de processo, onde são atualizadas continuamente e podem ser guardadas em bancos de dados locais ou remotos para fins de registro histórico (MORAES; CASTRUCCI, 2007).

O propósito de um sistema supervisório é prover ao usuário a capacidade de exercer controle sobre um determinado componente e confirmar se sua performance está de acordo com as diretrizes tomadas (ACKERMAN; BLOCK, 1992).

2.4.1 Características de sistemas supervisórios

Segundo Moraes e Castrucci (2007), os sistemas supervisórios possuem as seguintes características:

a) Facilidade de interpretação: a representação da planta por áreas e equipamentos de processo facilita a sua rápida interpretação, especialmente quando a tela é animada com cores e animações.

b) Flexibilidade: sistemas supervisórios permitem alterações no processo facilmente realizáveis por meio de softwares, sendo que alguns permitem a alteração de telas sem interromper a operação normal do dispositivo em que a alteração está sendo feita.

c) Estrutura: toda planta de produção é naturalmente dividida em áreas e é recomendável que a estrutura das telas do sistema supervisório acompanhe essa divisão natural. Nas telas são exibidos os principais equipamentos e os instrumentos de medição e atuação da planta, permitindo a sua visualização e controle. A visualização nessa forma estruturada permite uma navegação objetiva pelo processo, diminuindo o tempo de acesso do operador às variáveis supervisionadas.

2.4.2 SCADA

Os sistemas de supervisão e controle comumente chamados de sistemas SCADA (*Supervisory Control and Data Acquisition*) são sistemas configuráveis, destinados à supervisão, ao controle e à aquisição de dados de plantas industriais. Basicamente, um SCADA é um software implementado em um nível acima do nível de controle na hierarquia de uma rede industrial (GALLOWAY; HANCKE, 2013).

Um sistema SCADA engloba a coleta de informações e a transmissão das mesmas à uma unidade central para que então possam ser analisadas por um determinado número de operadores em um *display*. Após isso, as ações de controle necessário são direcionadas novamente para o processo (BEJAN; IACOB; ANDREESCU, 2009).

Moraes e Castrucci (2007) explicam que o dispositivo de controle que se comunica com um sistema SCADA é denominado RTU (*Remote Terminal Unit*) e geralmente é um tipo de CLP. Um exemplo pode ser visto na Figura 6, onde um sistema SCADA faz a aquisição de dados de quatro CLPs.

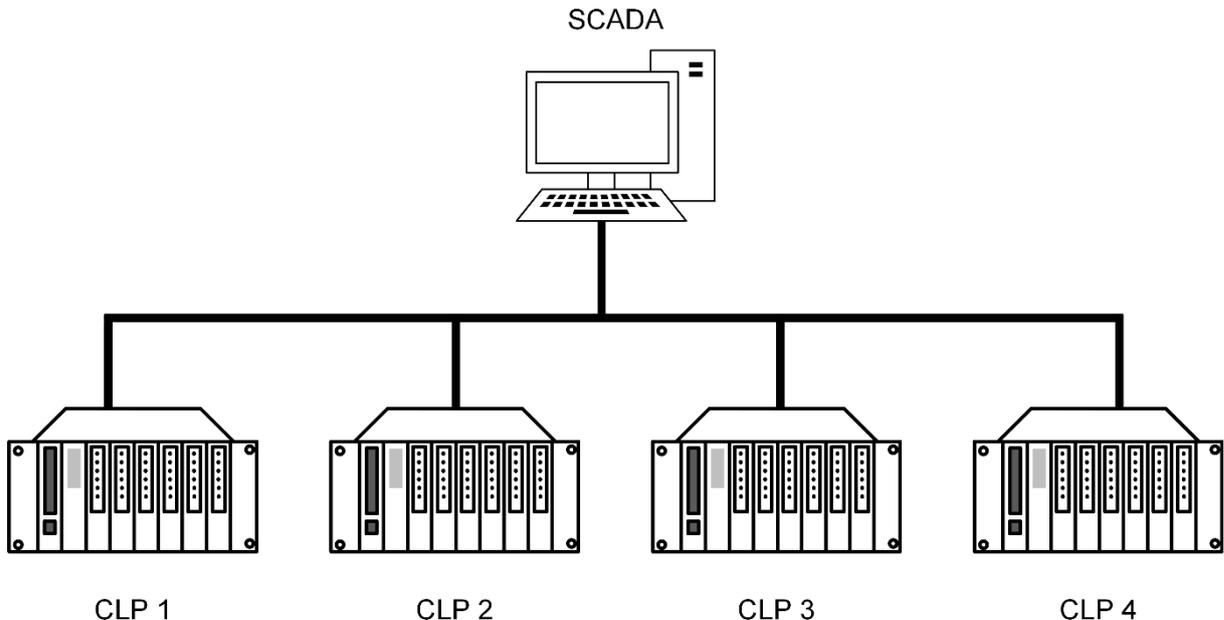


Figura 6 - Sistema supervisor SCADA representado por um servidor comunicando-se com quatro CLP's.

Fonte: Adaptado de Moraes e Castrucci (2007)

Para grandes aplicações, onde vários computadores operam a planta, existe a necessidade de se estabelecer um critério de como será feita a aquisição de dados dos CLPs, de estações remotas e demais equipamentos inteligentes do chão-de-fábrica. Uma maneira de resolver o problema mencionado é através de um sistema SCADA que possa acessar os dados dos CLPs e disponibilizá-los para outros sistemas através de uma rede entre computadores, totalmente independente da rede de CLPs. Essa transferência de dados entre computadores é realizada empregando-se uma filosofia cliente/servidor. A máquina que acessa os dados do CLP passa a ser o servidor de dados, enquanto as demais funcionam como clientes (Figura 7) (MORAES; CASTRUCI, 2007).

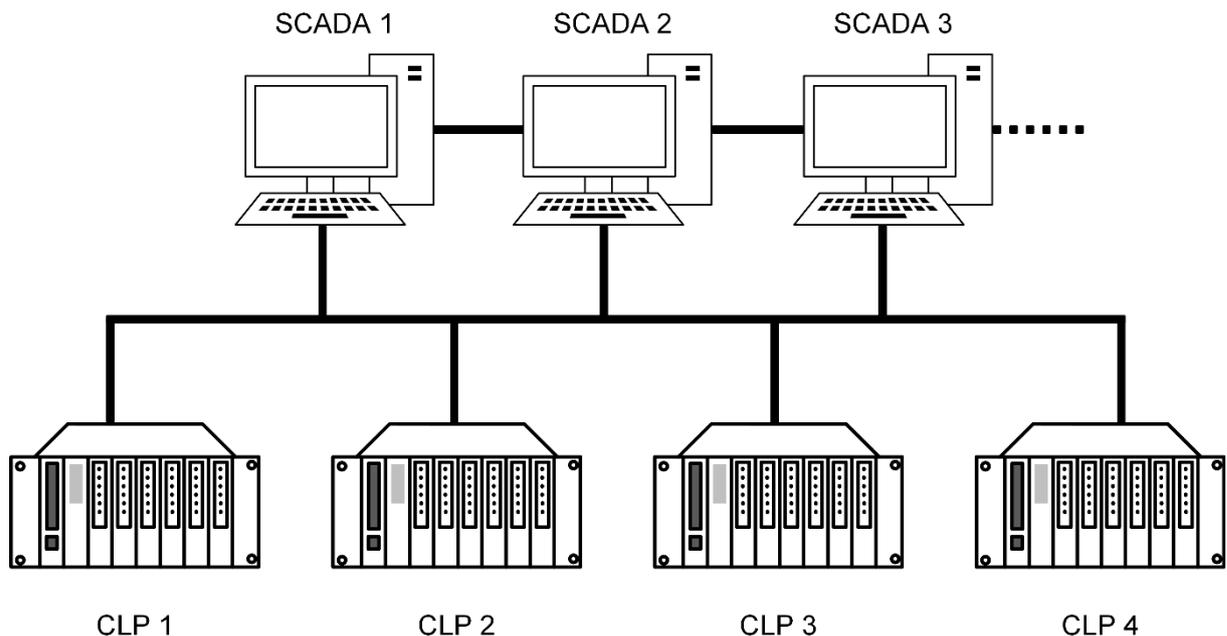


Figura 7 - Sistema supervisório SCADA interligado representado por um servidor transmitindo os dados dos quatro CLP's para dois ou mais clientes.
 Fonte: Adaptado de Moraes e Castrucci (2007)

2.5 REDES DE COMUNICAÇÃO INDUSTRIAL

Quando vários elementos como computadores, CLPs, sensores e atuadores trabalham interligados, trocando informações de forma precisa, com o objetivo de supervisionar e controlar um determinado processo, pode-se afirmar que formam uma rede de comunicação industrial (SILVA, 2011).

A utilização de redes permite a comunicação rápida e confiável entre equipamentos, fatores atualmente indispensáveis no conceito de produtividade industrial (NOGUEIRA, 2009).

Esta seção apresenta alguns conceitos sobre redes de comunicação industrial, como a evolução dos sistemas centralizados para distribuídos e os parâmetros necessários a uma rede para seu correto funcionamento.

2.5.1 Sistemas centralizados e distribuídos

Nos anos 60, inicia-se o uso de computadores para o controle de processos industriais, onde surge o conceito de *Direct Digital Control* (DDC). Neste sistema, um único computador digital (unidade de controle e monitoração) era usado para diferentes tarefas envolvidas com a operação das centenas ou milhares variáveis do processo (Figura 8). O DDC trouxe a vantagem de se eliminarem os indicadores, registradores e controladores de painel, tornando o processo mais eficiente. Entretanto, o uso de computadores exigia profissionais mais qualificados, com conhecimentos de linguagens de programação complexas, além do inconveniente do excesso de cabeamento entre o campo e a sala de controle, fator que causava altos custos de instalação e manutenção. (SILVA, 2004).

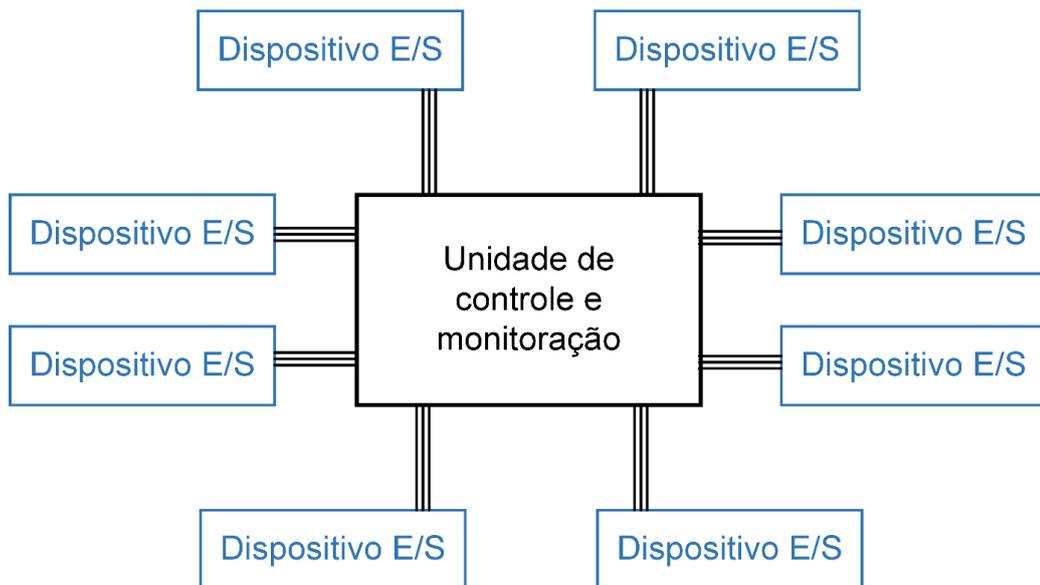


Figura 8 - Controle centralizado.
Fonte: Adaptado de Moraes e Castrucci (2007)

De acordo com Moraes e Castrucci (2007) para superar essas dificuldades, sistemas de automação de barramento de campo foram desenvolvidos (Figura 9). Nesses sistemas a estação de controle comunica-se com os dispositivos de entrada e saída através de um barramento. Suas características são:

- Controle centralizado;
- Transmissão digital de dados em uma topologia de barramento;

- Padrões RS232 ou RS485 para transmissão.



Figura 9 - Barramento de campo.
 Fonte: Adaptado de Moraes e Castrucci (2007)

Os autores explicam que, posteriormente, o avanço na tecnologia e demanda do mercado levaram ao desenvolvimento de sistemas de controle conhecidos como de barramento de campo distribuído (Figura 10). As características deste sistema são:

- Inteligência distribuída, utilizando microcontroladores ao longo do barramento;
- Redução de cabeamento e custos de instalação;
- Unidades de conexão (*gateways, bridges e repeaters*).

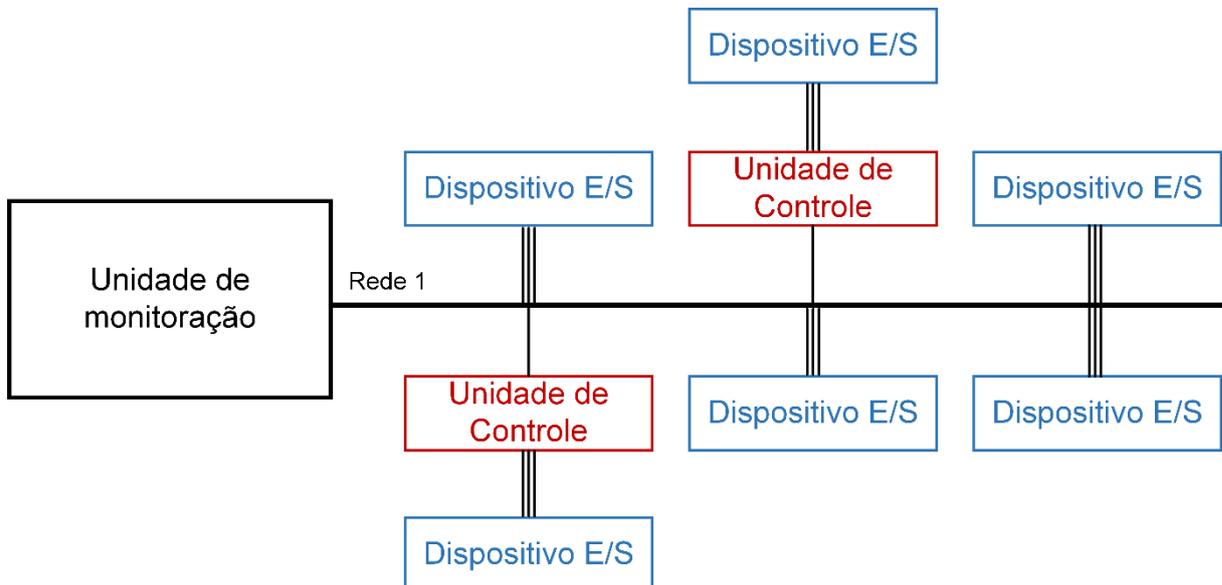


Figura 10 - Barramento de campo distribuído.
 Fonte: Adaptado de Moraes e Castrucci (2007)

Atualmente, implementando melhorias em relação a esses sistemas, foram desenvolvidos os chamados Sistemas Digitais de Controle Distribuído (SDCD) (Figura 11), que se caracterizam por:

- Meios variados de comunicação;
- Implementação mais completa para sistemas abertos, ou seja, não-proprietários;
- Flexibilidade completa para topologias de rede;
- Softwares e ferramentas de desenvolvimento mais simples para utilização do operador (MORAES; CASTRUCCI, 2007).

Em um SDCCD, o controle não é concentrado em um dispositivo central, mas distribuído em estações remotas. A estação central não é um elemento essencial a continuidade da operação, mas um dispositivo com intuito de facilitar e oferecer maiores recursos para a interface com o operador (PEREIRA, 2009).

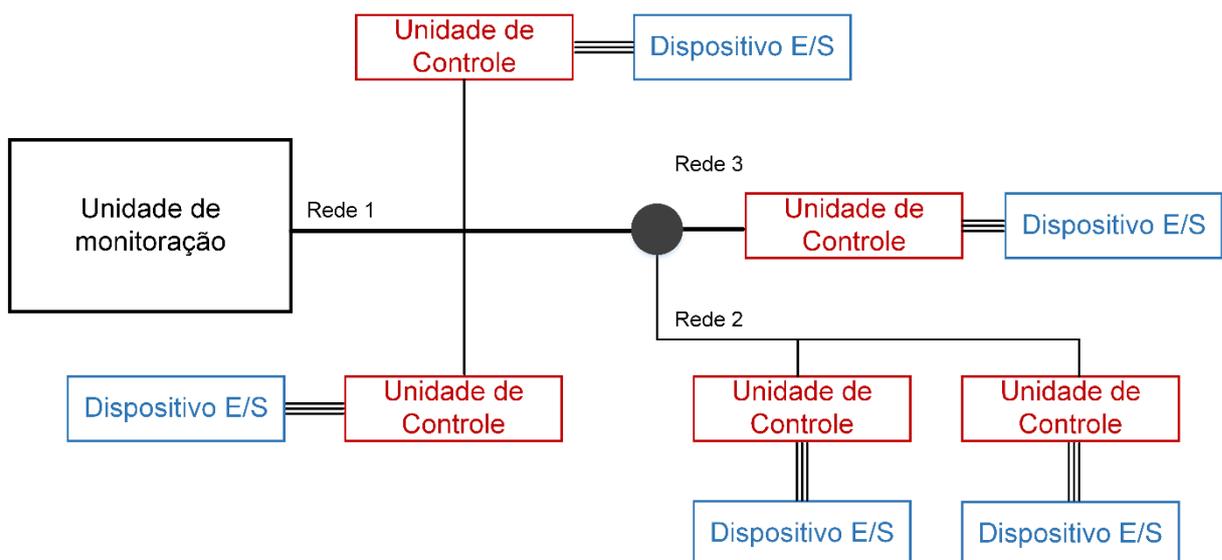


Figura 11 - Sistema digital de controle distribuído.
 Fonte: Adaptado de Moraes e Castrucci (2007)

2.5.2 Parâmetros de uma rede de comunicação industrial

Existem várias possibilidades de configuração para uma rede industrial, o que as torna flexíveis e aptas a atenderem as mais diversas necessidades. As redes podem ser caracterizadas por parâmetros distintos como taxa de transmissão,

topologia física da rede, meio físico de transmissão, tecnologia de comunicação e algoritmo de acesso ao barramento. (LAGE, 2009).

Moraes e Castrucci (2007) sintetizam esses parâmetros basicamente da seguinte maneira:

a) Taxa de transmissão: é a quantidade média de dados a serem transmitidos na rede por um período de tempo, medido em kilobits por segundo (kbps).

b) Topologia física da rede: representa a disposição construtiva na qual os dispositivos estão conectados na rede, em outras palavras, refere-se à aparência física da rede.

c) Meio físico de transmissão: representa todo o cabeamento utilizado para a interconexão dos dispositivos.

d) Tecnologia de comunicação: é a maneira como a comunicação entre os pontos (nós) da rede é gerenciada.

e) Algoritmo de acesso ao barramento: refere-se ao modo como os nós da rede acessam ou disponibilizam informações no barramento.

2.6 PROTOCOLOS DE COMUNICAÇÃO INDUSTRIAL

As redes industriais são constituídas por protocolos de comunicação com a finalidade de transportar sinais que trafegam sob um barramento de comunicação comum, para a interligação de tarefas, com o objetivo de controlar processos industriais (LUGLI; SANTOS, 2005).

Por definição um protocolo é um conjunto de regras sobre a comunicação entre os elementos de uma rede industrial e sua escolha depende da aplicação. Existem diversos protocolos utilizados: MODBUS, ETHERNET, PROFIBUS, entre outros (NOGUEIRA, 2009).

Para a aplicação proposta neste trabalho serão utilizados os protocolos de comunicação industrial MODBUS e OPC.

2.6.1 MODBUS

É um dos mais antigos protocolos usados em redes de controladores lógicos programáveis para aquisição de sinais de instrumentos e comando de atuadores usando uma porta serial. Pelo fato de ser um protocolo aberto, é usado em milhares de equipamentos e é uma das soluções mais economicamente viáveis a serem utilizadas em automação industrial (NOGUEIRA, 2009).

Trata-se de um protocolo de comunicação do tipo mestre-escravo, onde os escravos são usualmente controladores lógico programáveis e os mestres são sistemas supervisórios (LAGE, 2009).

2.6.2 OPC

Os servidores OPC (OLE – Object Linking and Embedding – Process Control) são softwares executados em plataforma Microsoft que fornecem uma interface para aplicações em dispositivos contendo padrão de comunicação proprietário. A vantagem do OPC é a utilização da arquitetura cliente/servidor. O cliente, que pode ser uma interface homem-máquina na sala de controle ou uma aplicação de PC, somente precisa reconhecer uma interface para obter dados de qualquer dispositivo (LOPEZ, 2000).

3 PROJETO E IMPLEMENTAÇÃO DA ARQUITETURA DO SDCD

3.1 DEFINIÇÃO DA ARQUITETURA

Devido à sua flexibilidade, robustez e ampla utilização na indústria, a integração proposta neste trabalho será implementada na arquitetura de um sistema digital de controle distribuído (Figura 12).

Nesta arquitetura, como explicado no capítulo 2.4.1 deste trabalho, o controle não é concentrado em um dispositivo central, mas distribuído em estações remotas denominadas unidades de controle, aqui representadas pelos inversores de frequência. Ainda nesta comparação, têm-se o conjunto de servidor e clientes como unidade de monitoração. O servidor desta unidade recebe e envia dados aos inversores por meio do protocolo de comunicação *Modbus RTU*. Os clientes, por sua vez, se comunicam com o servidor por meio do protocolo *OPC*.

Em resumo, todas as variáveis de processo obtidas pelos dispositivos de campo são coletadas e controladas pelos inversores, que enviam essas informações diretamente para um sistema supervisório.

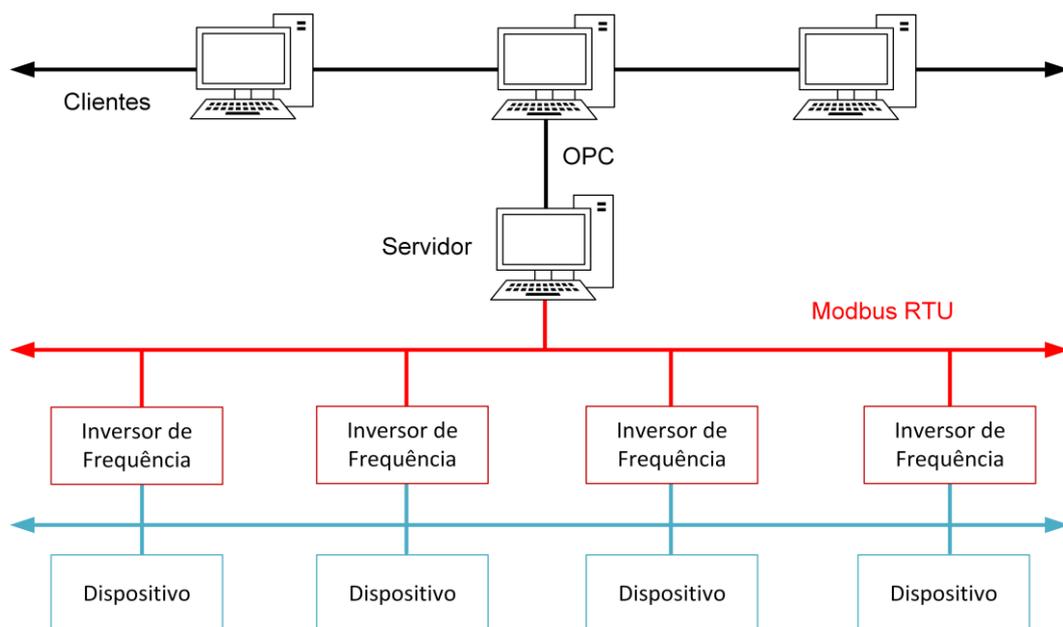


Figura 12 - Arquitetura proposta neste trabalho. Nesta arquitetura, têm-se unidades de monitoração, controle e dispositivos.

Fonte: Autoria própria

3.2 PROJETO E IMPLEMENTAÇÃO

Uma vez definida a arquitetura do sistema, uma bancada é construída para representá-la. Além de servir para validar a integração proposta, a bancada poderá ser utilizada por futuros alunos da instituição no aprendizado de redes industriais e outros assuntos correlacionados.

A seguir serão apresentadas informações sobre o projeto da bancada. O desenho dimensional da estrutura (vista frontal e lateral) é exibido na Figura 13. Neste trabalho optou-se por utilizar o inversor de frequência CFW-11 do fabricante WEG, devido a sua disponibilidade e funções especiais.

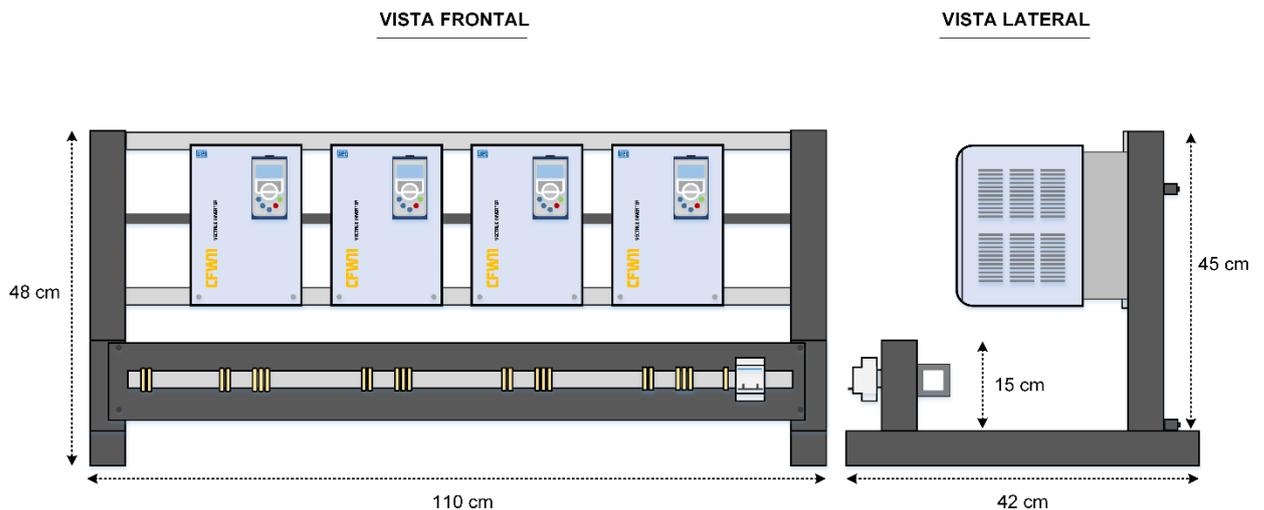


Figura 13 - Desenho dimensional da bancada que irá simular a arquitetura proposta.
Fonte: Autoria própria

O diagrama de ligações da bancada encontra-se na Figura 14. Nele, é possível observar o circuito de alimentação e de comunicação conectados paralelamente em cada inversor, bem como o circuito de aterramento.

É importante ressaltar que toda comunicação serial só pode ser realizada por meio de um acessório do inversor denominado “Módulo de comunicação serial RS485-01” (Figura 15). O CFW-11 possui vários módulos além deste usado na comunicação serial, como o módulo de expansão de entradas e saídas digitais usado para a função “softPLC”. Esta função, que agrega funcionalidades de um CLP ao inversor, será discutida posteriormente neste trabalho.

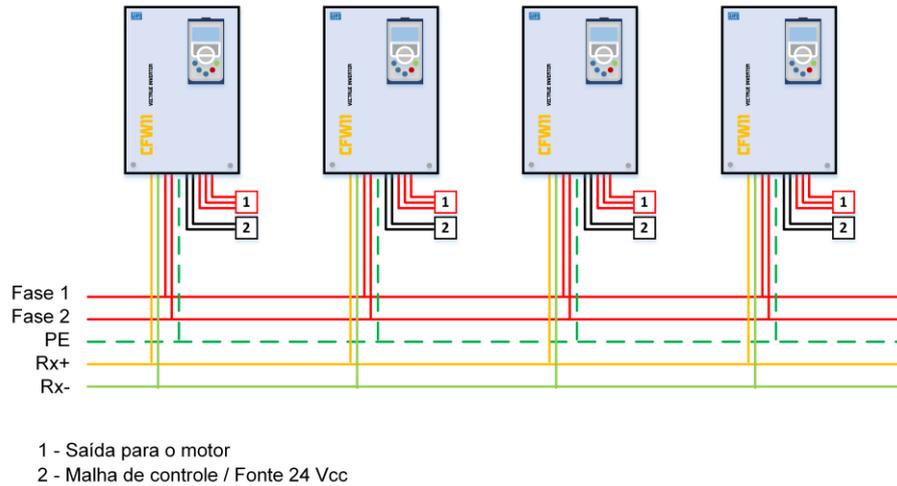


Figura 14 - Diagrama de ligações da bancada.
Fonte: Autoria própria



Figura 15 - Módulo de comunicação serial, indispensável para a integração proposta.
Fonte: Adaptado de WEG S.A. (2010)

A lista de materiais e equipamentos utilizados na bancada é exibida no Quadro 2. Neste quadro têm-se o tipo do material ou equipamento, fabricante, modelo e quantidade utilizada no projeto.

MATERIAL / EQUIPAMENTO	FABRICANTE	MODELO	QUANTIDADE
Trilho de metal	Aedem	Alumínio	3
Canaleta	Hellermann Tyton	Semi-aberta	1
Borne	Strahl	SAK	23
Disjuntor	JNG	DZ47-63	1
Inversor de frequência	WEG	CFW-11	4
Conversor digital	Novus	USB-i485	1
Cabo flexível amarelo	SIL	0,75 mm	6 m
Cabo flexível verde	SIL	0,75 mm	6 m
Cabo flexível preto	SIL	0,75 mm	4 m
Cabo flexível vermelho	SIL	1,5 mm	10 m
Cabo flexível verde	SIL	1,5 mm	4 m

Quadro 2 - Lista de materiais e equipamentos utilizados na bancada.

Fonte: Autoria própria

Após a definição do projeto, a bancada é construída em uma estrutura de metalon (Figura 16), com barras de dimensões 30x40 mm. O tamanho e modo como as barras de metalon foram fixadas garante a mobilidade da bancada, caso precise ser desmontada para realocação.



Figura 16 - Bancada que irá simular a arquitetura proposta construída em estrutura de metalon.

Fonte: Autoria própria

4 RESULTADOS E ENSAIOS DE IMPLEMENTAÇÃO DO SDCD

Para validar a integração proposta são realizados três testes. O objetivo dos testes é demonstrar o funcionamento da arquitetura montada e sua flexibilidade, tornando-a apta para ser adaptada para um processo industrial real.

O primeiro teste consiste em demonstrar a confiabilidade da rede ao comparar os dados recebidos via software do fabricante dos inversores com aqueles obtidos pelo sistema supervisório criado.

Já o segundo teste consiste em obter gráficos de resposta da variação de velocidade de um motor em função da variável de processo medida, com intuito de validar o funcionamento e comunicação do regulador PID do inversor.

O terceiro e último teste é feito com o principal diferencial do inversor utilizado, a função “*softPLC*” que agrega funcionalidades de um CLP ao inversor.

A escolha dos testes se justifica devido ao papel da automação no controle de processos industriais, onde é preciso monitorar e corrigir variáveis em tempo real para se obter um controle preciso das mesmas.

A interface gráfica do sistema supervisório, responsável por integrar os dispositivos, é construída no software Elipse SCADA (Figura 17). O procedimento para se configurar um sistema supervisório é exibido em detalhes no Anexo B deste trabalho.

Nesse sistema, *tags* de comunicação são usadas para receber e escrever dados nos inversores. Essas *tags* devem indicar o endereço do inversor, tipo de operação e número do registrador a ser acessado.

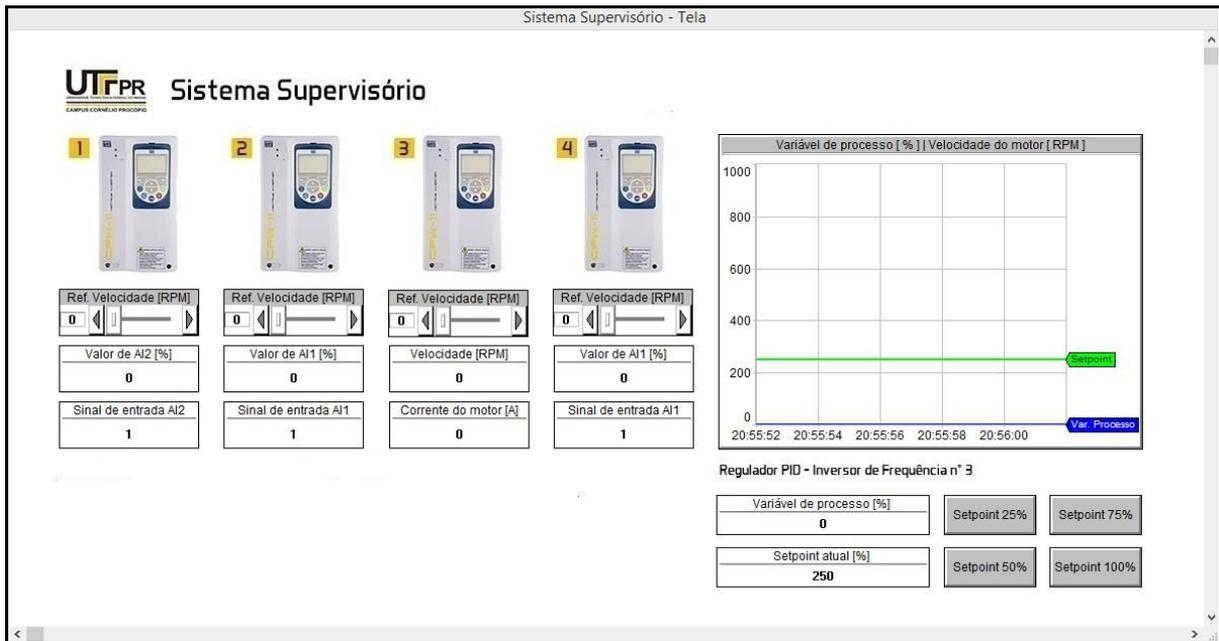


Figura 17 - Interface gráfica do sistema supervisório.
Fonte: Autoria própria.

4.1 TESTE DE COMUNICAÇÃO SERIAL

O objetivo deste teste, como exposto anteriormente, é demonstrar a confiabilidade da rede montada. Para sua execução, a arquitetura da Figura 18 é montada. Nela têm-se um dispositivo conectado em cada inversor de frequência e estes, por sua vez, conectados a uma unidade de monitoração (servidor e cliente).

Toda a comunicação entre os inversores de frequência e o servidor são feitas usando o protocolo de comunicação *Modbus* RTU. A escolha desse protocolo é baseada nas necessidades da aplicação, sendo que o mesmo é suficiente para realizar todos os testes propostos.

Na Figura 19 é possível ver a arquitetura montada para o primeiro teste com os equipamentos devidamente numerados. A descrição de cada equipamento é exibida no Quadro 3.

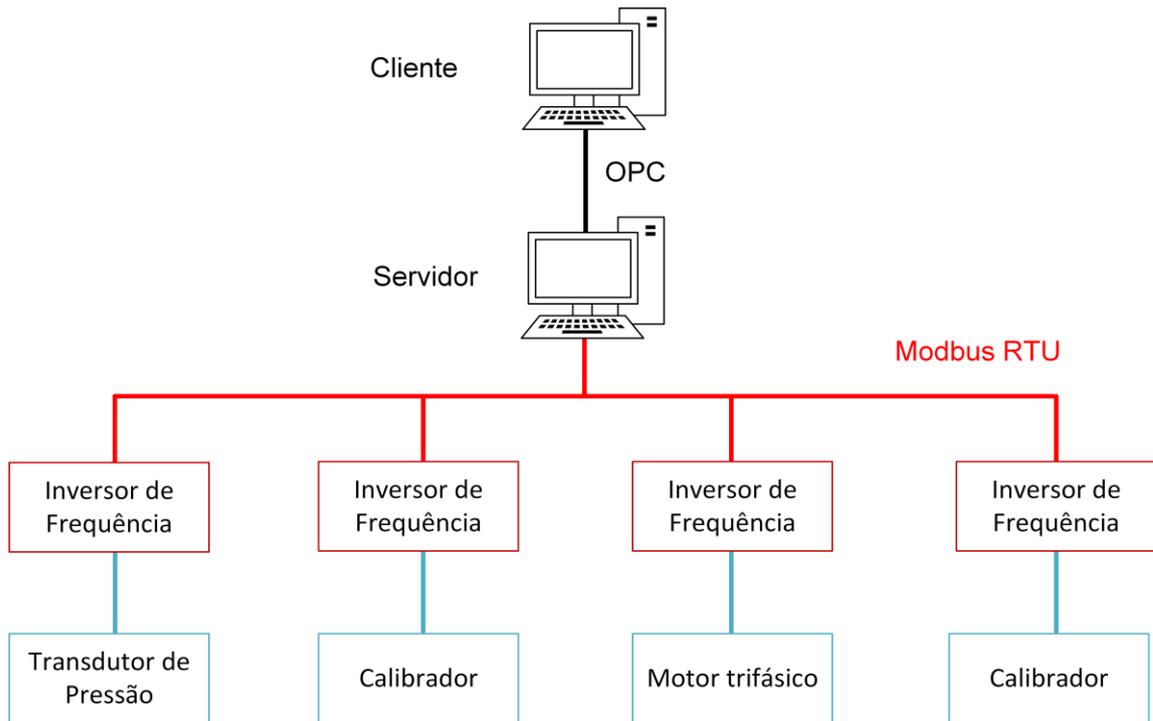


Figura 18 - Arquitetura proposta no primeiro teste.
Fonte: Autoria própria.

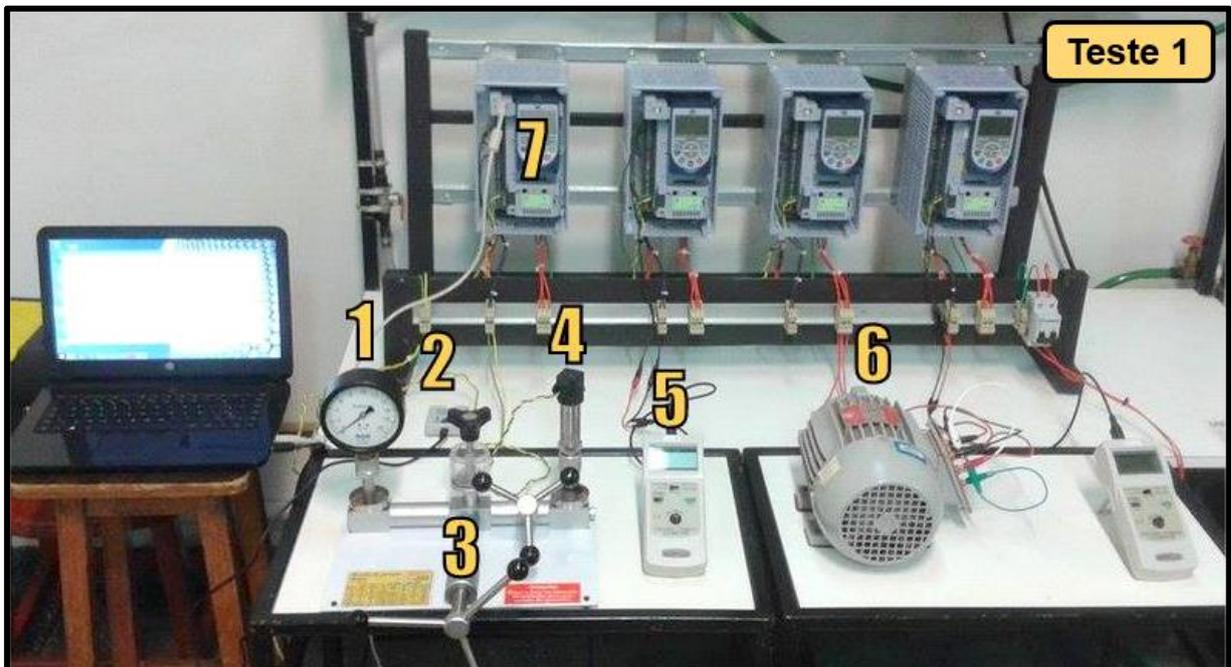


Figura 19 - Montagem da arquitetura do primeiro teste.
Fonte: Autoria própria

N°	DESCRIÇÃO	FABRICANTE	MODELO	QUANTIDADE
1	Manômetro	Multimedição	-	1
2	Conversor USB-i485	Novus	USB-i485	1
3	Bomba de aferição	Zurich	800-D	1
4	Transdutor de pressão	Zurich	PSI. 420	1
5	Calibrador	Instrutherm	CC-124	2
6	Motor de indução trifásico	WEG	80 393	1
7	Módulo de comunicação	WEG	RS485-01	4

Quadro 3 - Equipamentos utilizados na bancada para o primeiro teste.
Fonte: Autoria própria

A ligação do transdutor de pressão ao inversor foi feita na configuração a 2 fios, onde o sinal do transdutor é compartilhado com a alimentação 24 Vcc. A Figura 20 apresenta esta ligação. Os calibradores por sua vez foram ligados na entrada analógica AI1. Após a ligação dos equipamentos, os inversores são parametrizados para realizar a comunicação serial.

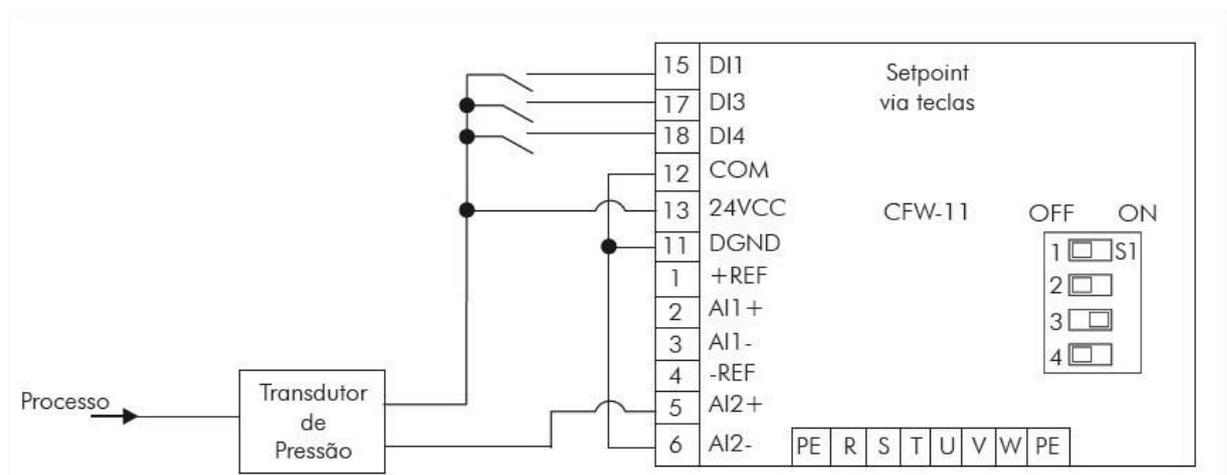


Figura 20 - Ligação do transdutor de pressão ao inversor de frequência com 2 fios.
Fonte: WEG S.A. (2012)

A rede montada para o primeiro teste é classificada (Quadro 4) de acordo com os parâmetros descritos no capítulo 2.4.2 deste trabalho. Essa classificação fornece uma representação técnica da rede em sua essência.

PARÂMETRO	DESCRIÇÃO
Taxa de transmissão	9600 kbps
Topologia física da rede	Barramento
Meio físico de transmissão	Par trançado (cabos flexíveis de 0,75 mm)
Tecnologia de comunicação	<i>Unicast</i>
Algoritmo de acesso ao barramento	Padrão <i>Modbus</i> RTU

Quadro 4 - Classificação da rede utilizada no primeiro teste.

Fonte: A autoria própria

Como descrito no quadro anterior, para se comunicar com os inversores o sistema supervisor, por meio do protocolo *Modbus* RTU, envia mensagens do tipo *unicast*. Esse tipo de mensagem consiste em enviar uma requisição para um escravo definido e este retornar uma resposta para o mestre. Também é possível enviar mensagens do tipo *broadcast* onde o mestre envia uma requisição para todos os escravos e nenhuma resposta é enviada para o mestre. Neste teste, para validar a comunicação, utiliza-se mensagens do tipo *unicast*.

O inversor utilizado possui um software próprio para parametrização, chamado *SuperDrive* G2. Por meio desse software, é possível comunicar-se diretamente com os inversores via porta USB. O objetivo é comparar os dados exibidos neste software com os valores obtidos via comunicação em rede pelo sistema supervisor criado. O procedimento para configuração do software *SuperDrive* G2 é descrito em detalhes no Anexo A deste trabalho.

No Quadro 5 é possível observar os dados lidos pelo software *SuperDrive* G2 em cada equipamento conectado ao inversor. Neste quadro, a coluna “Leitura” representa em porcentagem do fundo de escala (4 a 20 mA) o valor medido nos calibradores e no transdutor de pressão. Os valores dos sinais de entrada de AI1 e AI2 podem assumir valores de 0 a 3. Estes valores foram escolhidos aleatoriamente apenas para fins de validação da comunicação.

Em seguida, os mesmos dados são lidos pelo sistema supervisor via rede *Modbus* RTU (Figura 21). É importante esclarecer que os valores transmitidos são sempre números inteiros, independente de possuírem representação com casa decimal. Desta forma, o valor “9,5” seria transmitido como sendo “95” via serial.

SUPERDRIVE G2			
EQUIPAMENTO	VALOR	PARÂMETRO	LEITURA
Transdutor de pressão	10 kgf/cm ²	P0019 - Valor de AI2	49,11 [%]
		P0238 - Sinal de Entrada AI2	1
Calibrador	12 mA	P0018 - Valor de AI1	52,61 [%]
		P0233 - Sinal de Entrada AI1	1
Motor de indução trifásico	250 RPM	P0002 - Velocidade do Motor	284 [RPM]
		P0003 - Corrente do Motor	5,8 [A]
Calibrador	16 mA	P0018 - Valor de AI1	76,77 [%]
		P0233 - Sinal de Entrada AI1	1

Quadro 5 - Leitura das variáveis em cada inversor pelo software SuperDrive G2 do fabricante WEG.

Fonte: Autoria própria

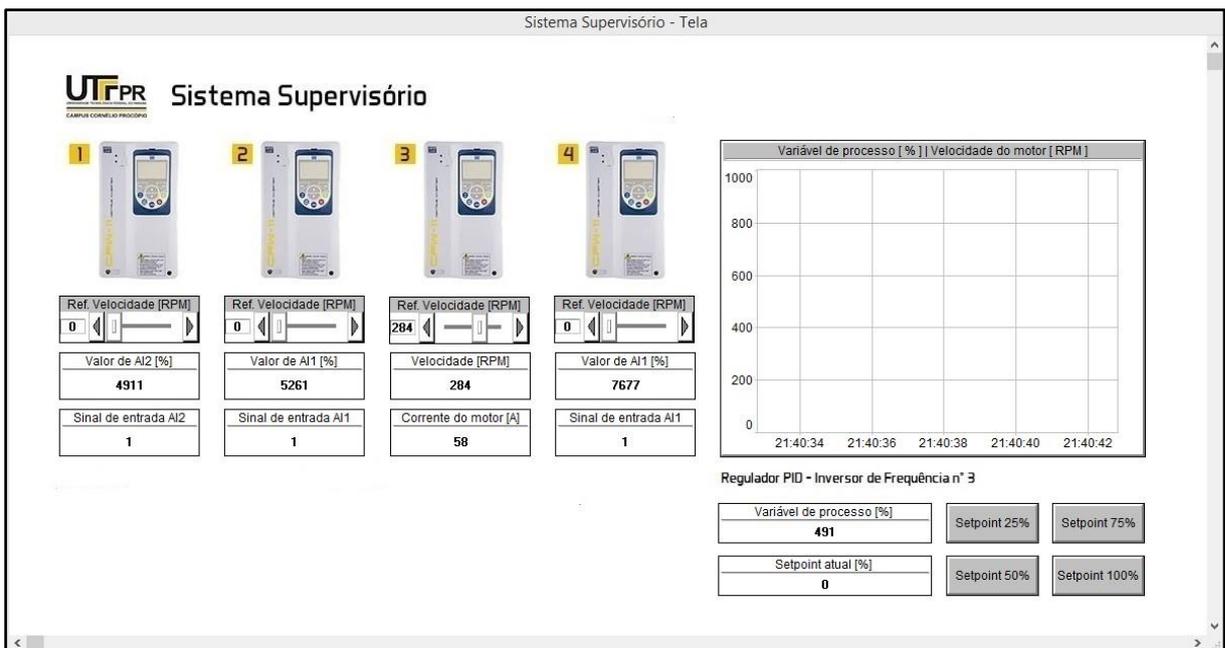
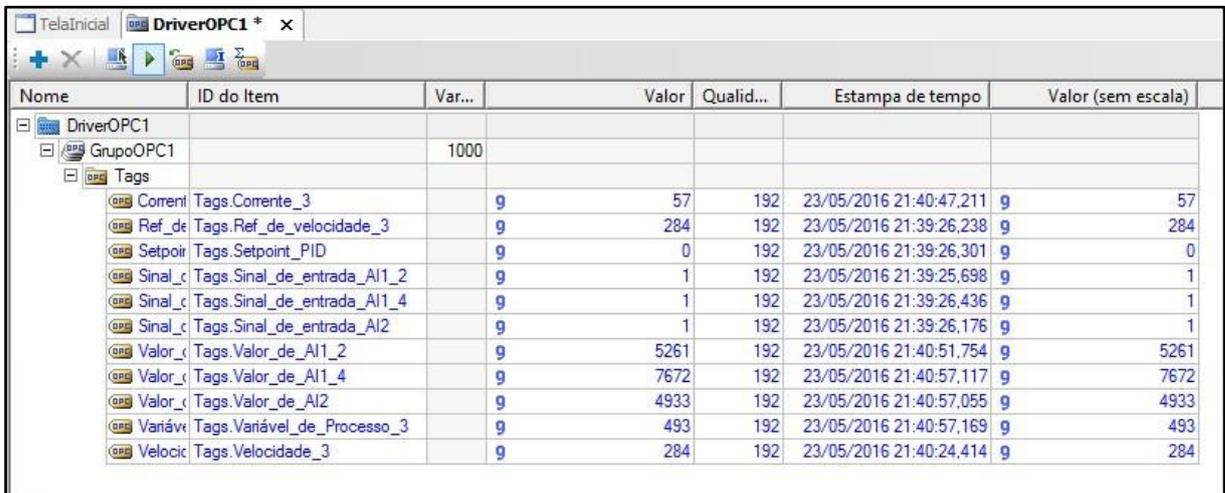


Figura 21 - Leitura das variáveis em cada inversor pelo sistema supervisório via rede Modbus RTU.

Fonte: Autoria própria

Na Figura 22 é possível visualizar também as variáveis recebidas por meio do protocolo OPC pela aplicação cliente criada no software Eclipse E3. Para efeito de teste, utilizou-se o servidor e cliente no mesmo computador em rede local. O

procedimento para a criação da aplicação cliente é exibida no Anexo C deste trabalho.



Nome	ID do Item	Var...	Valor	Qualid...	Estampa de tempo	Valor (sem escala)
DriverOPC1						
GrupoOPC1						
Tags						
Corrent	Tags.Corrente_3	g	57	192	23/05/2016 21:40:47,211	g 57
Ref_de	Tags.Ref_de_velocidade_3	g	284	192	23/05/2016 21:39:26,238	g 284
Setpoint	Tags.Setpoint_PID	g	0	192	23/05/2016 21:39:26,301	g 0
Sinal_c	Tags.Sinal_de_entrada_AI1_2	g	1	192	23/05/2016 21:39:25,698	g 1
Sinal_c	Tags.Sinal_de_entrada_AI1_4	g	1	192	23/05/2016 21:39:26,436	g 1
Sinal_c	Tags.Sinal_de_entrada_AI2	g	1	192	23/05/2016 21:39:26,176	g 1
Valor_c	Tags.Valor_de_AI1_2	g	5261	192	23/05/2016 21:40:51,754	g 5261
Valor_c	Tags.Valor_de_AI1_4	g	7672	192	23/05/2016 21:40:57,117	g 7672
Valor_c	Tags.Valor_de_AI2	g	4933	192	23/05/2016 21:40:57,055	g 4933
Variável	Tags.Variável_de_Processo_3	g	493	192	23/05/2016 21:40:57,169	g 493
Velocid	Tags.Velocidade_3	g	284	192	23/05/2016 21:40:24,414	g 284

Figura 22 - Leitura das variáveis em cada inversor pela aplicação cliente via OPC.
Fonte: Autoria própria

Ao comparar os valores obtidos via comunicação em rede *Modbus* RTU pelo sistema supervisório (servidor e cliente) com os valores reais obtidos pelo software *SuperDrive G2*, é possível concluir que a comunicação é executada com sucesso, tanto para leitura quanto para escrita de variáveis. Possíveis discrepâncias entre os valores deve-se a constante atualização dos mesmos, pois no momento em que cada tela é capturada esses valores podem sofrer pequenas variações.

4.2 TESTE DO REGULADOR PID

Este teste visa demonstrar o funcionamento do regulador PID do inversor e sua comunicação com o sistema supervisório. A importância deste teste reside no fato do regulador PID ser essencial no controle de variáveis de processo.

A arquitetura usada e sua montagem podem ser vistas nas Figuras 23 e 24, respectivamente. São usados um inversor de frequência, um motor trifásico, um manômetro e um transdutor de pressão acoplado a uma bomba de aferição.

Neste teste, têm-se o transdutor de pressão enviando sinais de 4 a 20 mA equivalentes a pressão medida na escala de 0 a 20 kgf/cm² para o inversor. A

pressão é gerada por meio da bomba de aferição (representada pelo bloco “processo”) ao girar-se a válvula em sua extremidade. O inversor de frequência irá aumentar ou diminuir a velocidade do motor na tentativa de manter a variável de processo (pressão) no *setpoint* escolhido.

Como o intuito do teste é apenas demonstrar o funcionamento do regulador PID do inversor, a referência de pressão é gerada manualmente, não existindo conexão física entre o motor e a bomba de aferição.

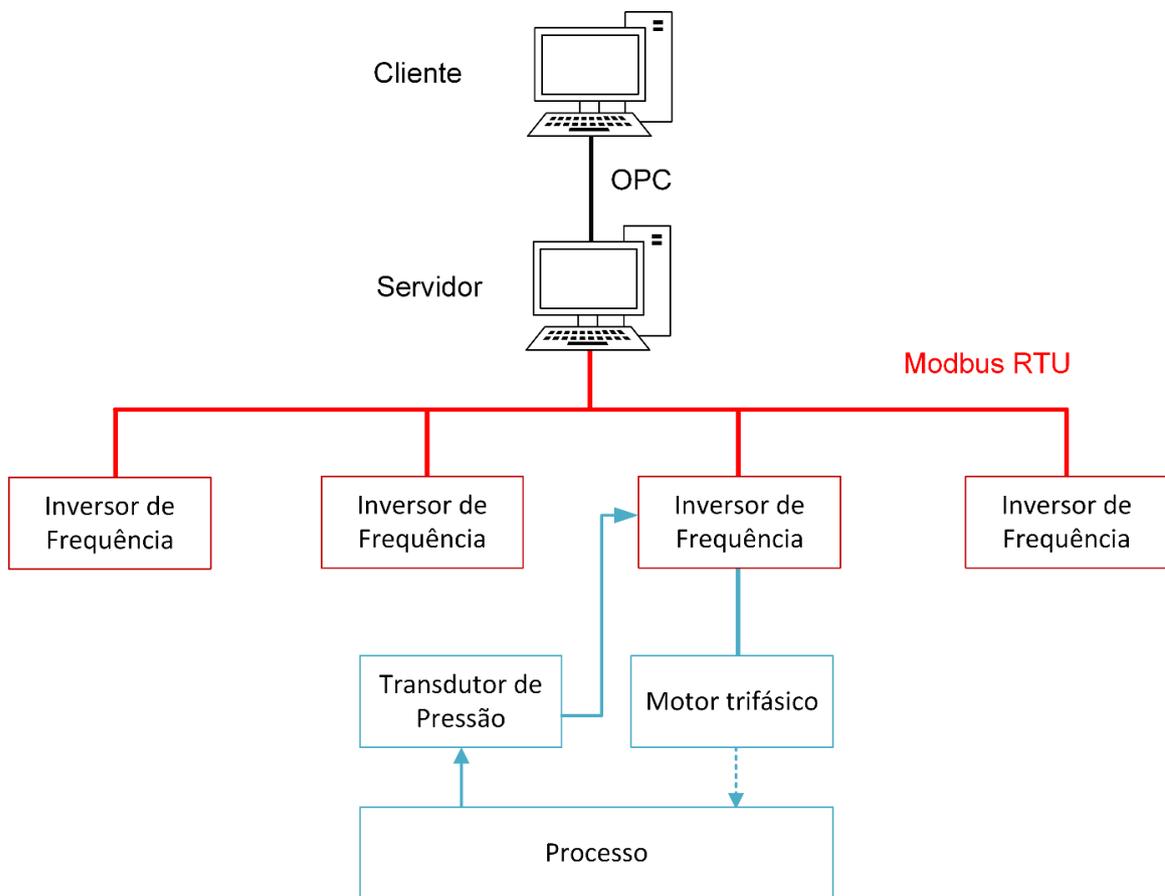


Figura 23 - Arquitetura proposta no segundo teste.
 Fonte: Autoria própria



Figura 24 - Montagem da arquitetura do segundo teste
Fonte: Autoria própria

As ligações elétricas do motor e transdutor de pressão ao inversor de frequência são exibidas na Figura 25. Nessa figura é importante destacar que além das ligações elétricas é preciso configurar a chave 3 corretamente na posição “ON” e as demais na posição “OFF” para receber um sinal de corrente elétrica na entrada analógica AI2. Posteriormente, é preciso configurar os parâmetros do inversor para acessar a função especial do PID.

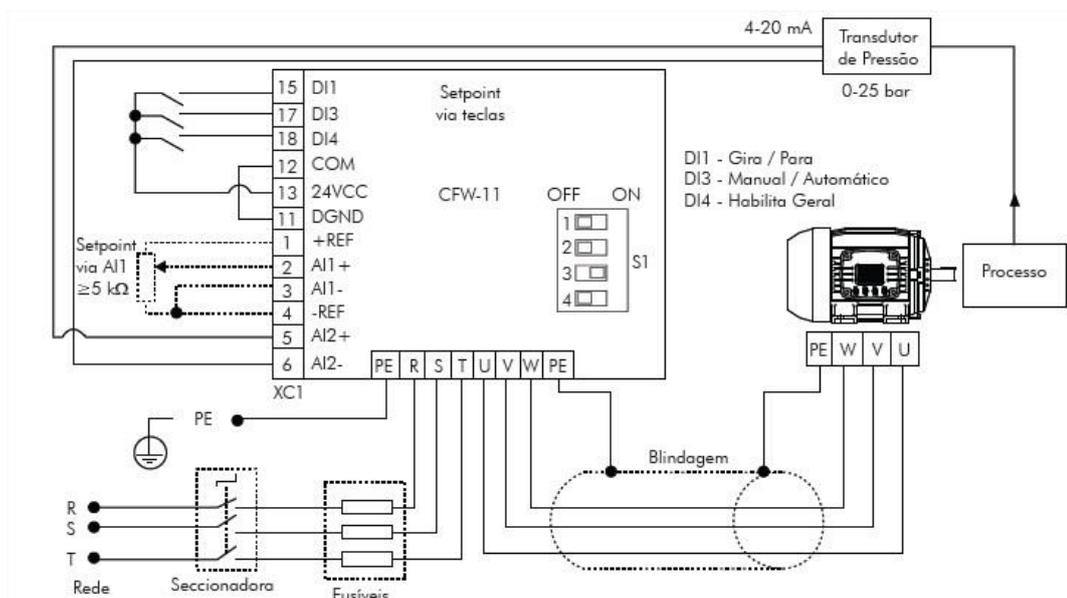


Figura 25 - Ligação do transdutor de pressão ao inversor de frequência para o segundo teste.
Fonte: WEG S.A. (2012)

Em relação ao sistema supervisório criado, têm-se um gráfico associando a velocidade atual do motor, o *setpoint* escolhido e o status da variável de processo medida. Nos gráficos gerados (Figuras 26 e 27) é possível observar a resposta da velocidade do motor perante a alteração da variável de processo com 2 *setpoints* diferentes (50% e 25%). Como o PID foi configurado com ação reversa uma diminuição da variável de processo acarreta num aumento da velocidade do motor e vice-versa, na tentativa de manter essa variável o mais próximo possível do *setpoint*. O tempo de resposta do inversor pode ser melhorado alterando-se parâmetros do regulador PID.

Esse teste também demonstra a utilidade de se representar e registrar variáveis graficamente. Utilizando-se esse artifício a análise de falhas é facilitada uma vez que é possível ver exatamente o momento da ocorrência de um determinado problema e sua relação com outras variáveis.

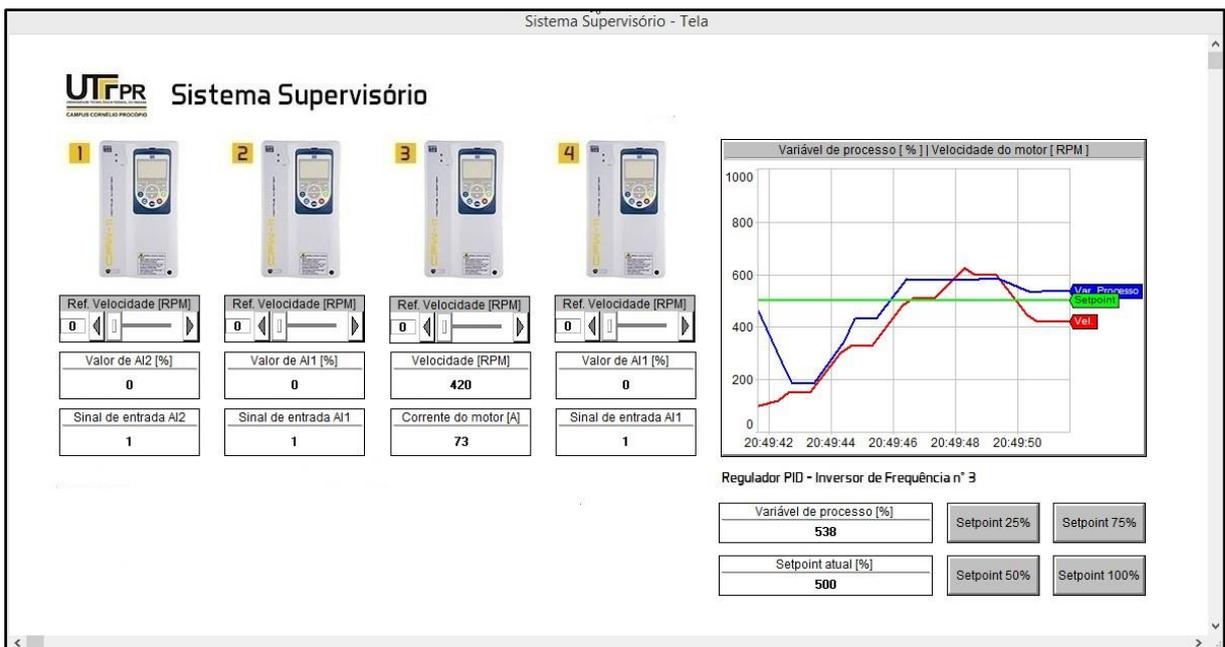
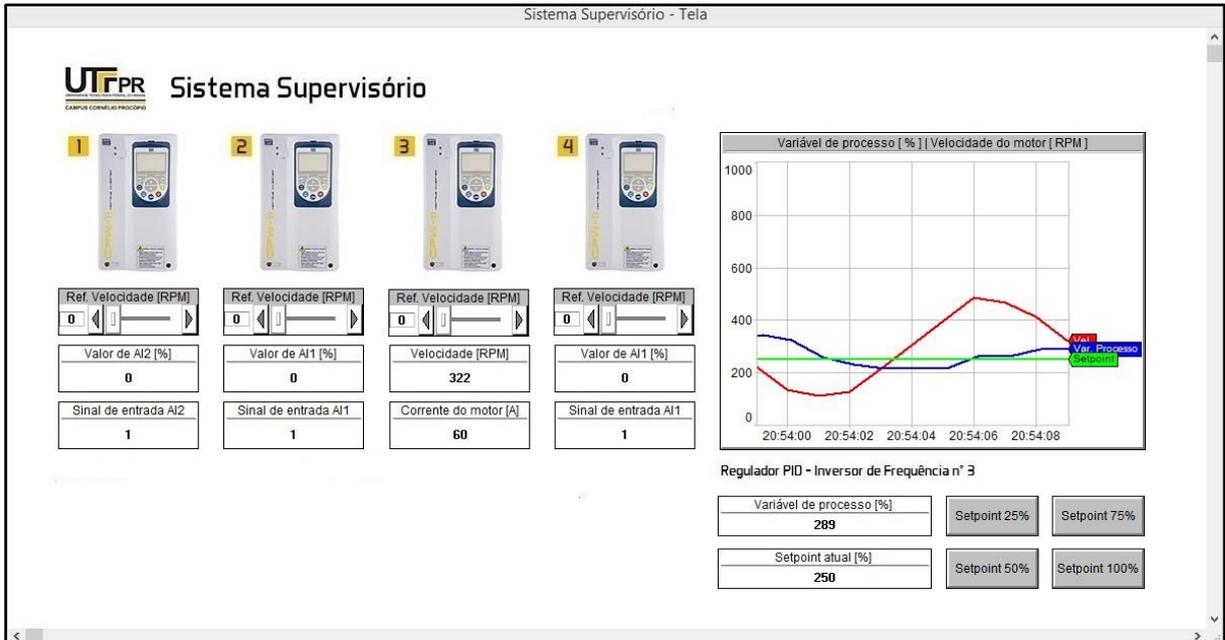


Figura 26 - Gráfico de resposta da velocidade do motor em função da variável de processo com setpoint de 50%.
Fonte: Autoria própria.



**Figura 27 - Gráfico de resposta da velocidade do motor em função da variável de processo com setpoint de 25%.
Fonte: Autoria própria.**

4.3 TESTE DE APLICAÇÃO DO CLP INTEGRADO

Este teste visa demonstrar um dos maiores diferenciais do inversor de frequência utilizado, a função “softPLC” que permite que o inversor assuma funções de um CLP. Essa função possui as seguintes características:

- Software de programação WLP (*Weg Ladder Programmer*) gratuito;
- Linguagem de programação ladder;
- Blocos de CLP, matemáticos e controle
- Acesso a todos os parâmetros do inversor;
- Monitoração remota.

Com o software WLP, é possível editar lógicas em linguagem ladder e fazer o *download* dos programas diretamente nos inversores.

Na configuração de fábrica, o inversor possui 6 entradas e 3 saídas digitais, porém é possível expandir esse valor por meio de módulos PLC. Uma aplicação teste é criada (Figura 28) utilizando-se 4 entradas digitais. O objetivo da aplicação é habilitar o funcionamento do inversor e configurar sua referência de velocidade pelo bloco “*Set-Speed*”.

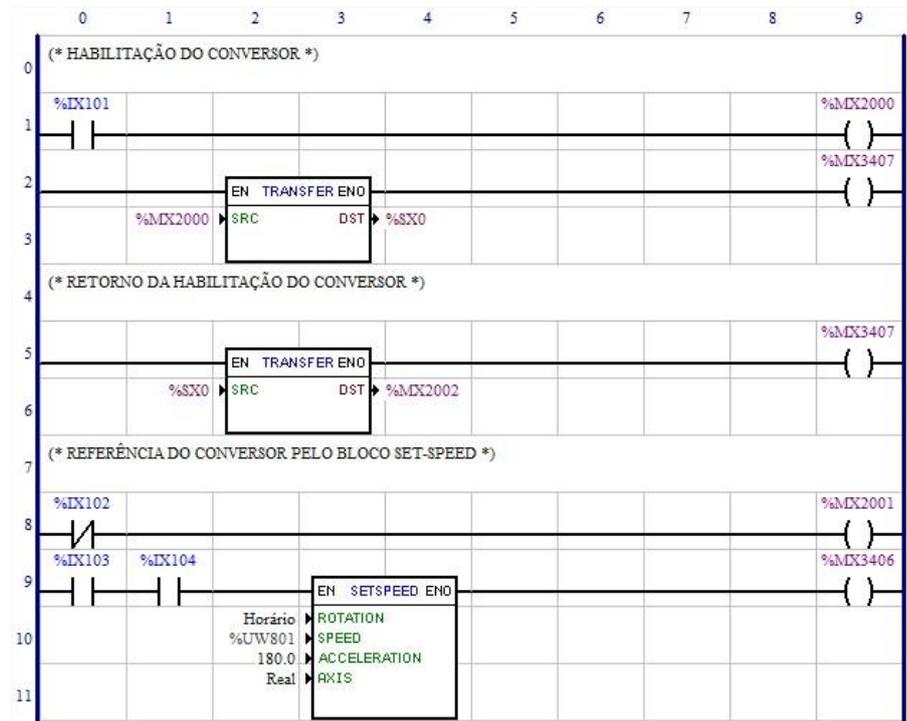


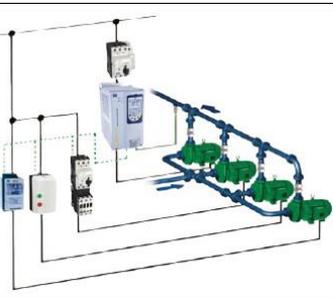
Figura 28 - Linguagem ladder desenvolvida para o teste da função “softPLC”.
Fonte: Autoria própria

Ao receber um sinal de nível alto nas entradas digitais %IX103 e %IX104, o bloco “Set-speed” carrega a referência de velocidade pré-programada pelo usuário após o inversor ser habilitado pela entrada %IX101.

Como explicado nesse trabalho, o CLP atualiza suas saídas baseado nas suas entradas por meio de uma lógica de comando, que pode ser escrita em diferentes linguagens de programação. No CFW-11 as saídas podem representar portas digitais e analógicas ou parâmetros do inversor.

A presença de um controlador lógico programável integrado ao inversor representa redução de custos, uma vez que pode-se usar seu controlador integrado em rede ao invés de um controlador externo.

A aplicação simples realizada nesse teste ilustra a flexibilidade e robustez do inversor CFW-11 pois combinando a lógica criada com um sistema supervisório é possível monitorar as entradas e saídas digitais dos inversores de frequência, facilitando a análise de eventuais falhas no controlador. Para ilustrar sua flexibilidade, exemplos de aplicação da função “softPLC” explorada neste teste e seus benefícios podem ser vistos no Quadro 6.

APLICAÇÃO	DESCRIÇÃO	EXEMPLO
Bombas e ventiladores	Redução do consumo de energia elétrica, controle preciso de variáveis de um processo (pressão, vazão, temperatura) com regulador PID, sinalização de alarmes de segurança e manutenção.	
Compressores	Redução do consumo de energia elétrica, melhora o rendimento e controle de demanda, diminui a corrente de partida, evita desgastes mecânicos.	
Controle multibombas	Permite manter a pressão de linha de uma tubulação constante, independente das flutuações de demanda de vazão.	
Elevação de cargas	Modo vetorial garante precisão de parada e controle de velocidade mesmo em baixas rotações.	

Quadro 6 - Aplicações do inversor de frequência com a função softPLC.
Fonte: Adaptado de WEG S.A. (2013)

5 CONSIDERAÇÕES FINAIS

Neste trabalho foi apresentada a integração de inversores de frequência na arquitetura de um sistema digital de controle distribuído. Para que tal objetivo fosse atingido, construiu-se uma bancada para representar a arquitetura em questão.

Após a construção da bancada e parametrização dos inversores foi possível realizar três testes para validar a integração proposta.

Utilizando o protocolo de comunicação *Modbus* RTU o sistema supervisório criado no software Elipse SCADA foi capaz de coletar dados da unidade de controle e dos dispositivos de entrada de forma confiável e eficaz. As funcionalidades do regulador PID e do CLP integrado do inversor também foram exploradas com sucesso.

Com o exposto pode-se concluir que o objetivo da integração proposta foi atingido uma vez que os elementos da arquitetura foram interligados física e logicamente apesar de possuírem tecnologias distintas. Tal fato torna a arquitetura, com eventuais adaptações necessárias, apta para atuar em qualquer aplicação industrial real.

Pelo fato de arquiteturas como a utilizada neste trabalho envolverem diferentes áreas da Engenharia Elétrica, a bancada construída poderá auxiliar alunos no aprendizado de redes industriais, automação e controle discreto, instrumentação industrial e acionamento de motores, aproximando os futuros profissionais das necessidades da indústria.

Para trabalhos futuros, sugere-se a implementação de um laboratório remoto envolvendo a bancada construída, de modo que os alunos possam acessar e fazer experimentos pela internet em períodos de tempo determinados.

Essa implementação poderia aumentar o contato dos alunos com equipamentos utilizados na indústria, melhorar a qualidade do aprendizado e reduzir custos de instalação de laboratórios nas universidades.

REFERÊNCIAS

ACKERMAN, W. J.; Block, W. R. (1992). Understanding supervisory systems. Computer Applications in Power, **IEEE**. New Jersey, v. 5, n. 4, p.37-40, oct. 1992.

BEJAN, Cristina A.; IACOB, Mihai; ANDREESCU, Gheorghe-Daniel. SCADA automation system laboratory, elements and applications. In: INTERNACIONAL SYMPOSIUM, 7., 2009, Subotica, Serbia. **Conference...**Subotica, Serbia: IEEE, 2009. P. 181-186.

CAPELLI, Alexandre. **Automação Industrial: Controle do Movimento e Processos Contínuos**. São Paulo: Érica, 2006.

EL-HAWARY, Mohamed. **Induction motors and their control**. New Jersey: Wiley-IEEE Press, 2002.

FERREIRA JÚNIOR, Milton Gontijo. **Controle de um inversor de frequência via CLP**. 2012. 39 f. Termo de Conclusão de Curso (Graduação) - Curso de Engenharia, Controle e Automação, Universidade Federal de Ouro Preto, Ouro-preto, 2012.

FRANCHI, Claiton Moro; CAMARGO, Valter Luís Arlindo de. **Controladores Lógicos Programáveis: Sistemas Discretos**. 2. ed. São Paulo: Érica, 2013.

GALLOWAY, Brendan; HANCKE, Gerhard P. Introduction to industrial control networks. **Communications Surveys & Tutorials, IEEE**. New Jersey, v. 15, n.2, p. 860-880, second quarter 2013.

GEORGINI, Marcelo. **Automação Aplicada: Descrição e Implementação de Sistemas Seqüenciais com PLCs**. São Paulo: Érica, 2000.

LAGE, Felipe Stiergert. **Comunicação Industrial**. 2009. 115 f. Trabalho de Conclusão de Curso (Graduação) - Curso de Engenharia, Controle e Automação, Universidade Federal de Ouro Preto, Ouro-preto, 2009.

LOPEZ, Ricardo Aldabó. **Sistemas de Redes para Controle e Automação: Rede Industrial, Tecnologias de Controle, Meios de Transmissão, Modelo OSI Rede Fieldbus Industriais, Sistemas Residenciais e rede Ethernet**. Rio de Janeiro: Book Express, 2000.

LUGLI, Alexandre B.; SANTOS, Max M. D. **Redes industriais: evolução, motivação e funcionamento**. Intech América do Sul. 2005.

MASCHERONI, José M.; LICHTBLAU, M.; GERARDI, D. **Guia de aplicação: inversores de frequência**. 2009.

MORAES, Cícero Couto de; CASTRUCCI, Plínio de Lauro. **Engenharia de Automação Industrial**. Rio de Janeiro: LTC, 2007.

NOGUEIRA, Thiago A. **Redes de comunicação para sistemas de automação industrial**. 2009. 80 f. Trabalho de Conclusão de Curso (Graduação) - Curso de Engenharia, Controle e Automação, Universidade Federal de Ouro Preto, Ouro-preto, 2009.

PEREIRA, Warley H. **Sistemas digitais de controle distribuído**. 2009. 44 f. Trabalho de Conclusão de Curso (Graduação) - Curso de Engenharia, Controle e Automação, Universidade Federal de Ouro Preto, Ouro-preto-MG, 2009.

PRUDENTE, Francesco. **Automação Industrial: PLC: Teoria e Aplicações: Curso Básico**. Rio de Janeiro: LTC, 2007.

SILVA, Gustavo Vitorino Monteiro da. **As redes de campo em instrumentação e controle industrial**. 2004. 128 f. Tese (Doutorado) - Curso de Engenharia Eletrotécnica e de Computadores, Escola Superior de Tecnologia de Setúbal, Setúbal, 2004.

SILVA, Lyvia Regina Biagi. **Estudo aplicado de topologias de controle de processos utilizando uma planta didática industrial**. 2011. 115 f. Trabalho de Conclusão de Curso (Graduação) - Curso de Engenharia Elétrica, Universidade Tecnológica Federal do Paraná, Cornélio Procópio-PR, 2011.

WEG S.A. **CFW-11: Inversor de Frequência**. Jaraguá do Sul - SC. 20 p. 2013.

WEG S.A. **Manual da Comunicação Serial RS232/RS485**. Blumenau - SC. 34 p. 2010.

WEG S.A. **Manual de Programação: Inversor de Frequência CFW-11**. Blumenau - SC. 294 p. 2012.

ANEXO A - PARAMETRIZAÇÃO DOS INVERSORES DE FREQUÊNCIA

Para utilizar o inversor de frequência, é preciso fornecer parâmetros ao mesmo para que funcione de maneira adequada. Esta parametrização pode ser feita de duas maneiras, utilizando-se a IHM do dispositivo ou por meio do software SuperDrive G2 conectando-se um cabo USB ao inversor.

1) Utilizando a IHM

A IHM do inversor CFW-11 possui botões de navegação para se inserir ou alterar os parâmetros. A figura abaixo mostra a função de cada botão.

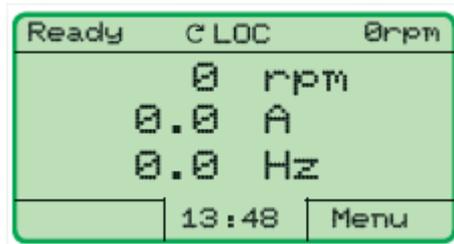


1.1) Inserindo as configurações do motor e da aplicação

De acordo com recomendações do fabricante, para aplicações simples, como é o caso da utilização deste trabalho, é possível parametrizar os inversores por meio da função “Start-up orientado”. Esta função consiste em informar ao inversor características básicas sobre o motor e a aplicação a qual está destinado.

Para acessar o startup orientado basta seguir os seguintes passos:

Passo 1 - No modo de “Monitoração”, pressione “Menu” utilizando a “soft key” direita para acessar o “Menu”.

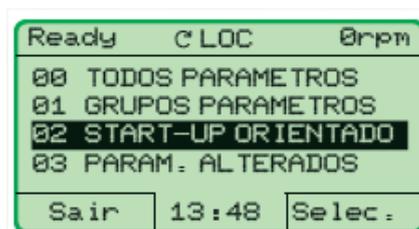


Passo 2 - Acesse a opção “00 Todos os parâmetros” e em seguida selecione o parâmetro P0000, alterando seu valor para 5. Selecione a opção “salvar”.



Passo 3 - No modo de “Monitoração”, pressione “Menu” utilizando a “soft key” direita.

Passo 4 - Com as teclas de navegação, vá até a opção 02 de parâmetros, denominado “Start-up orientado” e entre neste grupo pressionando o botão “Selec.”.



Passo 5 - Altere o valor do parâmetro P0317 (Start-up orientado) para “Sim” com as teclas de navegação e selecione “Salvar”.



Passo 6 - Neste momento é iniciada a rotina do “Start-up orientado”. Com as teclas de navegação, altere os valores dos parâmetros de acordo com seu motor e aplicação.

1.2) Inserindo as configurações de comunicação serial

Para configurar corretamente a comunicação serial do inversor, é preciso acessar o grupo de parâmetros destinado a tal função por meio dos seguintes passos:

Passo 1 - No modo de “Monitoração”, pressione “Menu” utilizando a “soft key” direita.

Passo 2 - Com as teclas de navegação, acesse a opção “01 Grupos Parâmetros”.

Passo 3 - Uma vez nos grupos de parâmetros, escolha o grupo “49 Comunicação”.

Passo 4 - No grupo “49 Comunicação”, defina os parâmetros da rede que deseja alterar, um exemplo é o parâmetro P0312 que permite escolher o protocolo de comunicação serial.

2) Utilizando o software SuperDrive G2

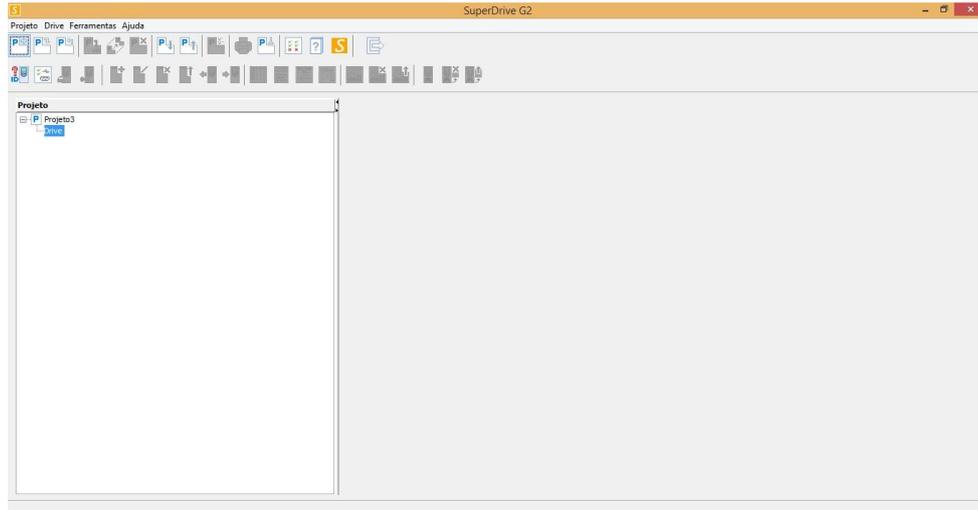
Devido a grande quantidade de parâmetros a serem configurados, uma maneira mais rápida de realizar a parametrização é por meio do software SuperDrive G2, disponível no site do fabricante do inversor.

Para utilizá-lo, basta seguir os seguintes passos:

Passo 1 - Baixe o software SuperDrive G2 no site do fabricante WEG.

Passo 2 - Extraia o arquivo em uma pasta de sua escolha e execute o instalador.

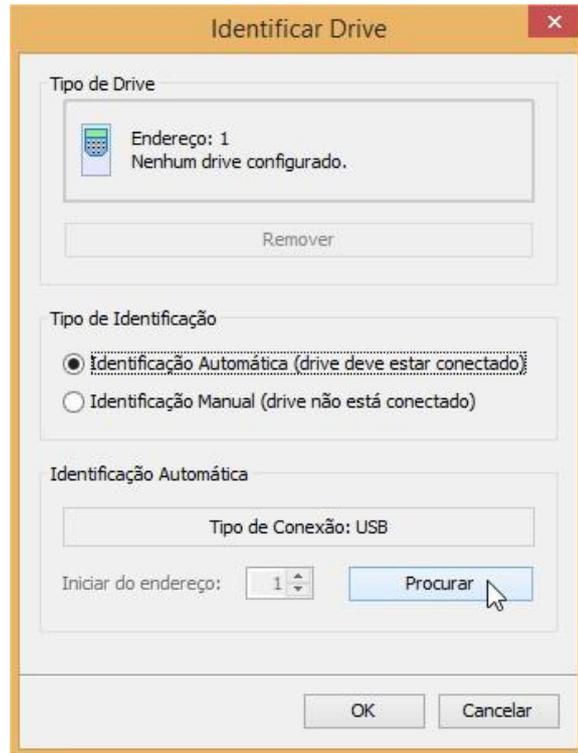
Passo 3 - Após instalado, abra o programa.



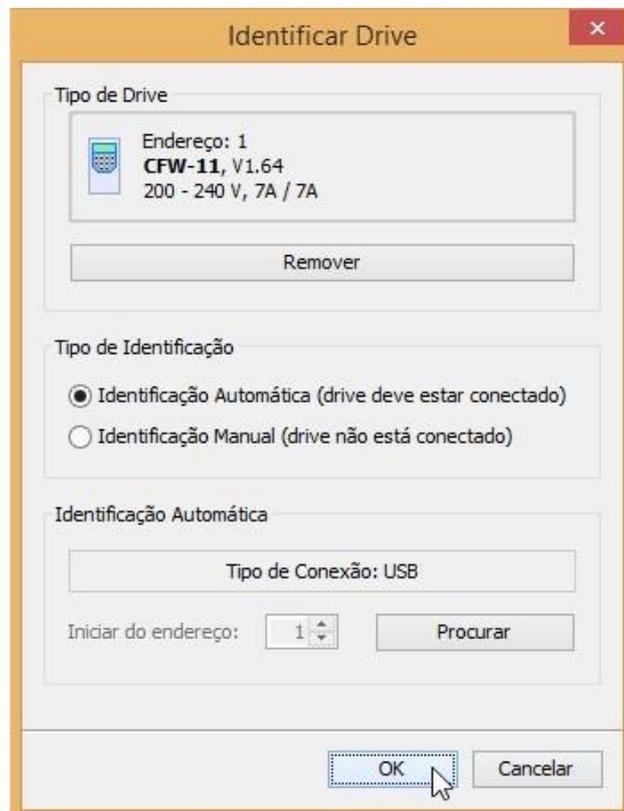
Passo 4 - Conecte o inversor ao computador por meio de um cabo USB e clique em “Identificar Drive” no canto superior esquerdo.



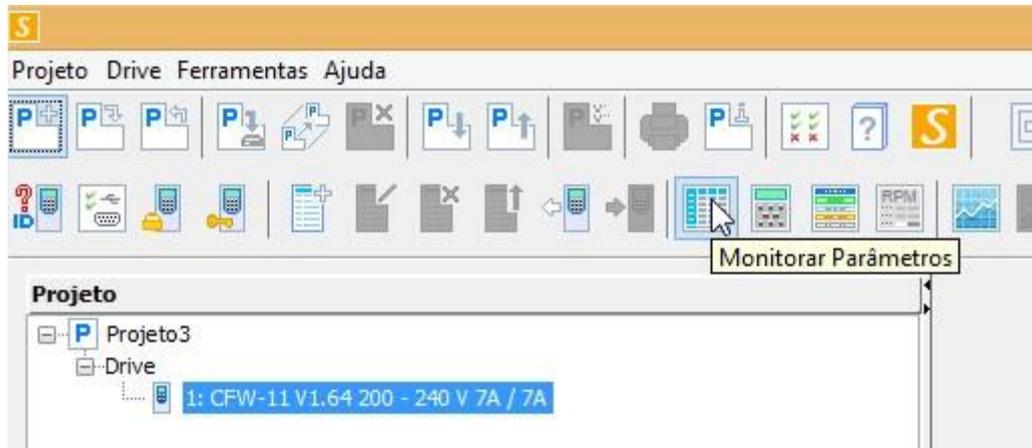
Passo 5 - Na janela que se abre, clique para marcar a caixa “Identificação Automática (drive deve estar conectado)” e em seguida clique em “Procurar”.



Passo 6 - Após a identificação do drive, suas informações devem ser exibidas no campo “Tipo de Drive”. Clique em “OK”.



Passo 7 - Depois de da identificação do drive, clique no botão “Monitorar Parâmetros” no menu superior.



Passo 8 - Finalmente, na janela que se abre é possível monitorar e alterar os parâmetros desejados clicando em cada célula da coluna “Ajuste do Usuário”. É importante lembrar que somente podem ser alterados os parâmetros que não foram classificados como “Somente leitura”.

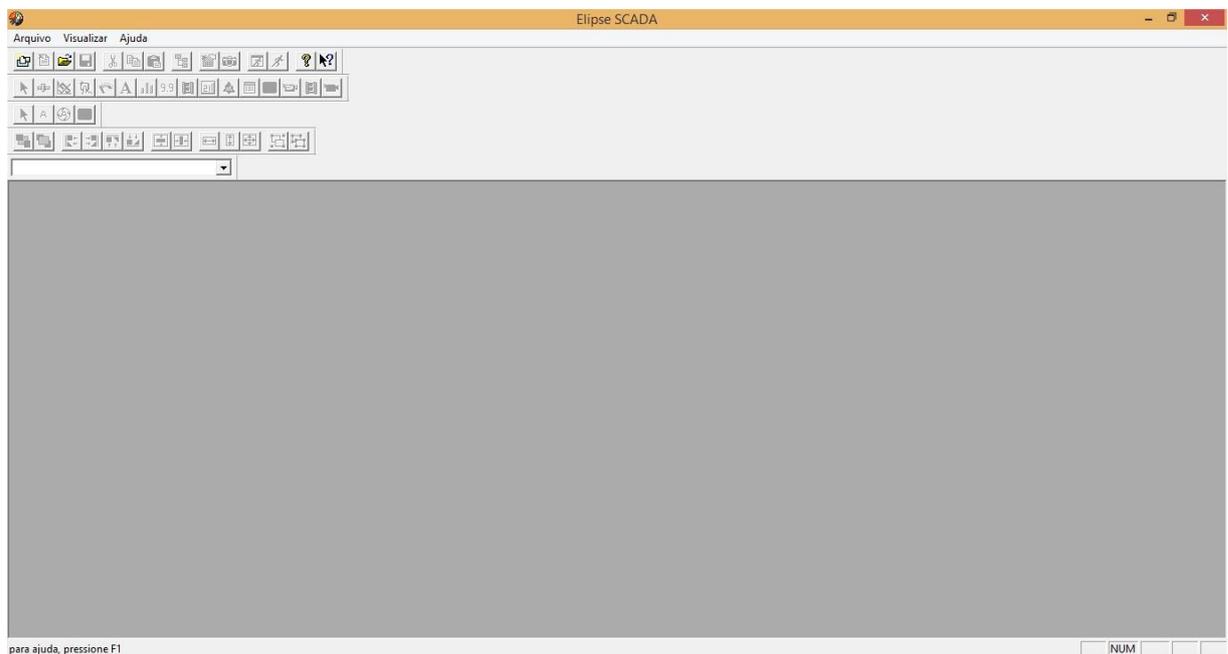
Número	Função	Mínimo	Máximo	Ajuste de Fábrica	Ajuste do Usuário	Unidade	Comparação
0	Acesso aos Parâ...	0	9999	0	5		Alterado
1	Referência Veloci...	0	18000	0	0	rpm	
2	Velocidade do Motor	0	18000	0	0	rpm	
3	Corrente do Motor	0	4500	0	0	A	
4	Tensão Barram.C...	0	2000	0	322	V	
5	Frequência do Mo...	0	300	0	0	Hz	
6	Estado do Inversor	0	7	0: Ready (Pronto)	0: Ready (Pronto)		
7	Tensão de Saída	0	2000	0	0	V	
9	Torque no Motor	-1000	1000	0	0	%	
10	Potência de Saída	0	6553,5	0	0	kW	
12	Estado DI8 a DI1	00000000b	11111111b	00000000b	00000100b		
13	Estado DO5 a DO1	00000000b	00011111b	00000000b	00000001b		
14	Valor de AO1	0	100	0	0	%	
15	Valor de AO2	0	100	0	0	%	
16	Valor de AO3	-100	100	0	0	%	
17	Valor de AO4	-100	100	0	0	%	
18	Valor de AI1	-100	100	0	0	%	
19	Valor de AI2	-100	100	0	0	%	
20	Valor de AI3	-100	100	0	0	%	

35 ms Online

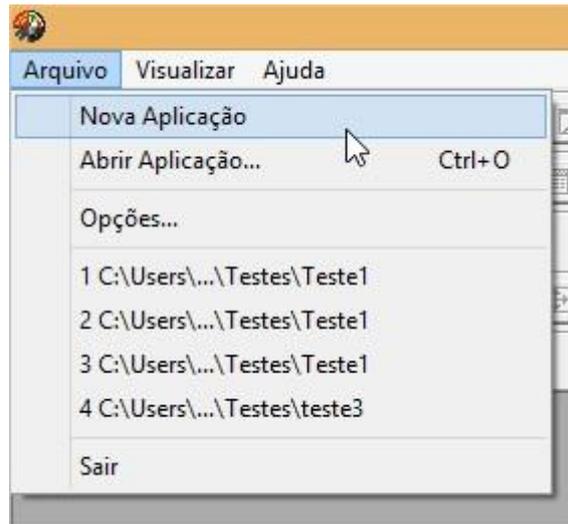
ANEXO B - CONFIGURAÇÃO DO SISTEMA SUPERVISÓRIO (SERVIDOR)

Este anexo tem como objetivo auxiliar o usuário na configuração do driver de comunicação *Modbus* RTU e tags no sistema supervisório criado no software Elipse SCADA atuando como servidor.

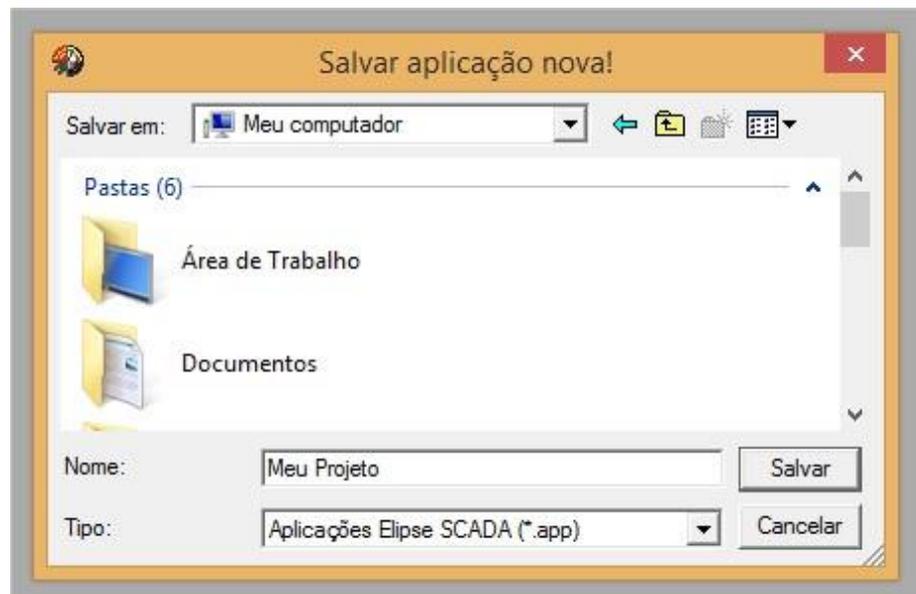
Passo 1 - Abra o software Elipse SCADA.



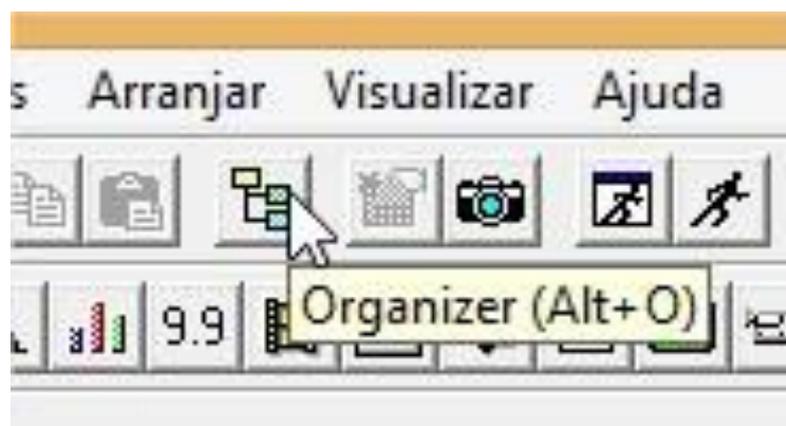
Passo 2 - Clique em “Arquivo” e em “Nova Aplicação”.



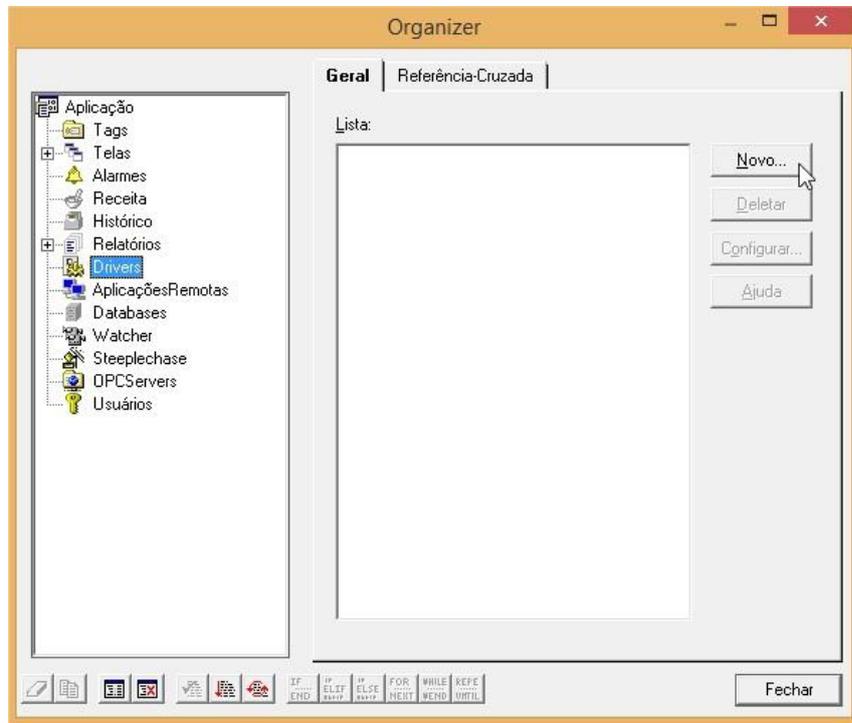
Passo 3 - Escolha um nome e um diretório para salvar sua aplicação.



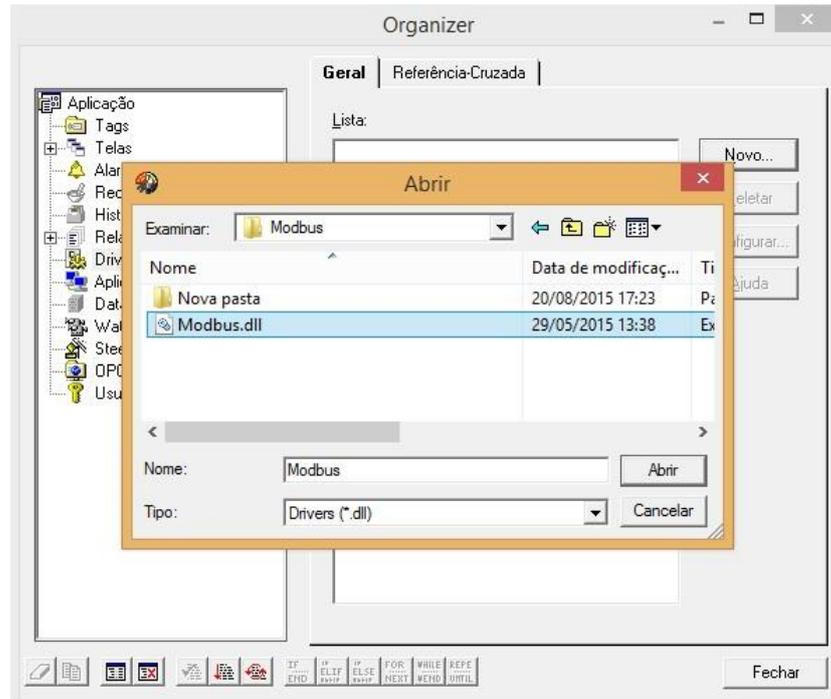
Passo 4 - Após criado o projeto, clique no botão “Organizer” no menu superior.



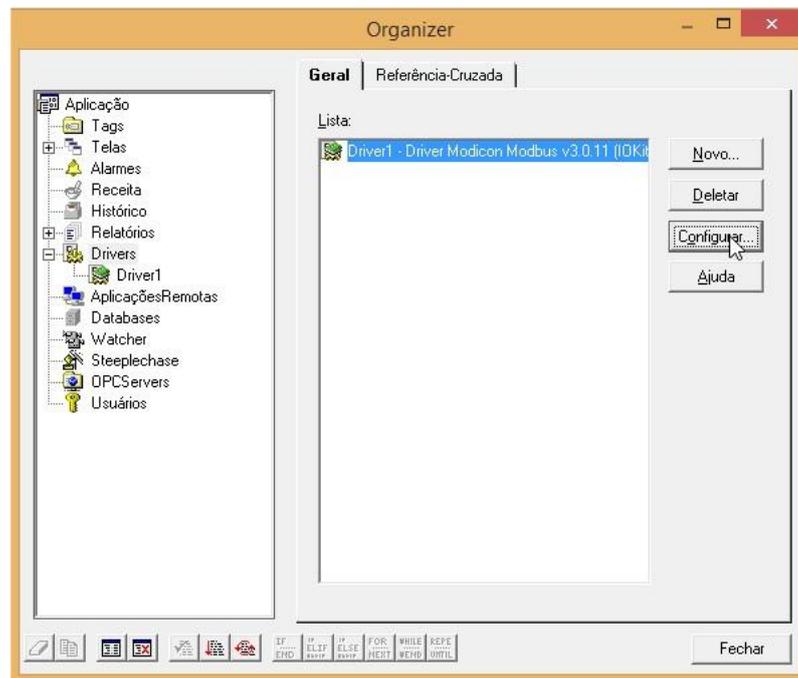
Passo 5 - Na janela “Organizer”, clique em “Drivers” no menu na lateral esquerda. Em seguida, clique no botão “Novo...”.



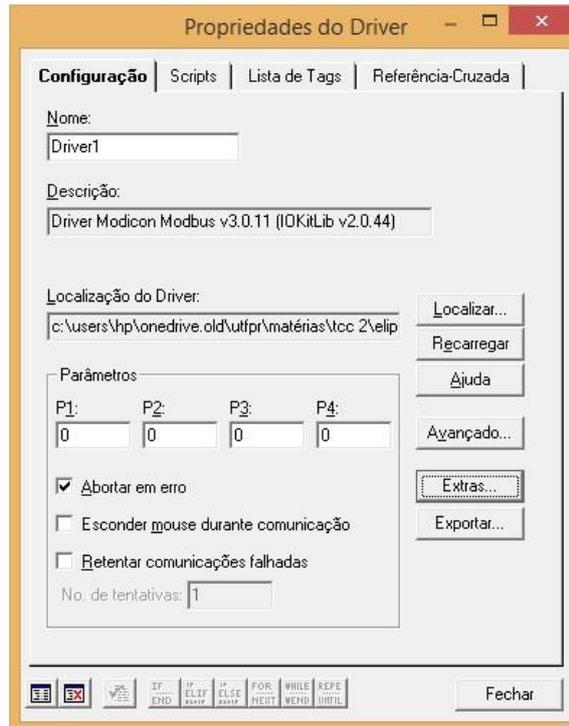
Passo 6 - Escolha o diretório e em seguida clique no driver que deseja utilizar (arquivo com formato .dll). Neste trabalho foi utilizado o driver Modbus disponível no site do desenvolvedor do software Elipse SCADA.



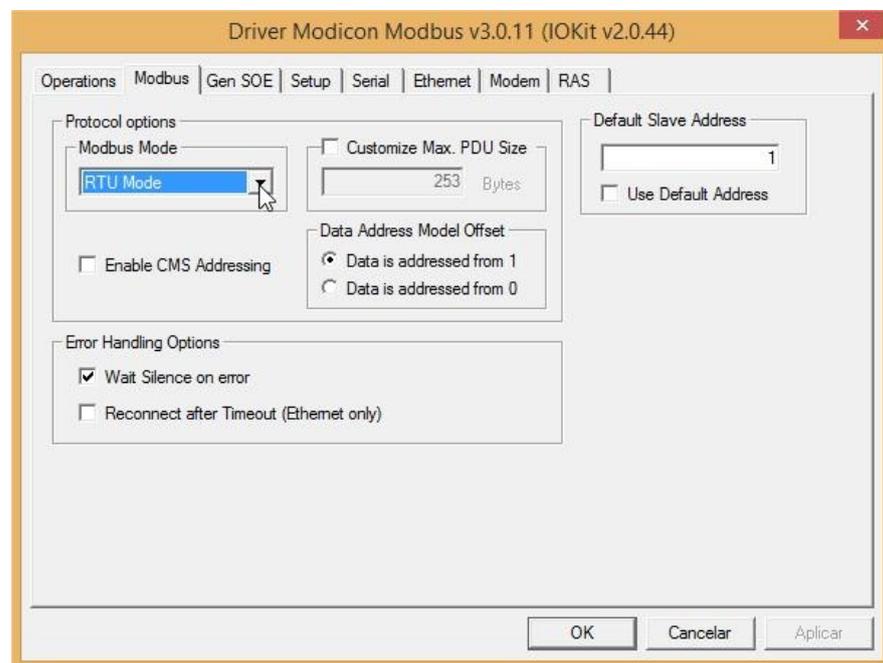
Passo 7 - Após escolhido o driver, clique no botão “Configurar...”.



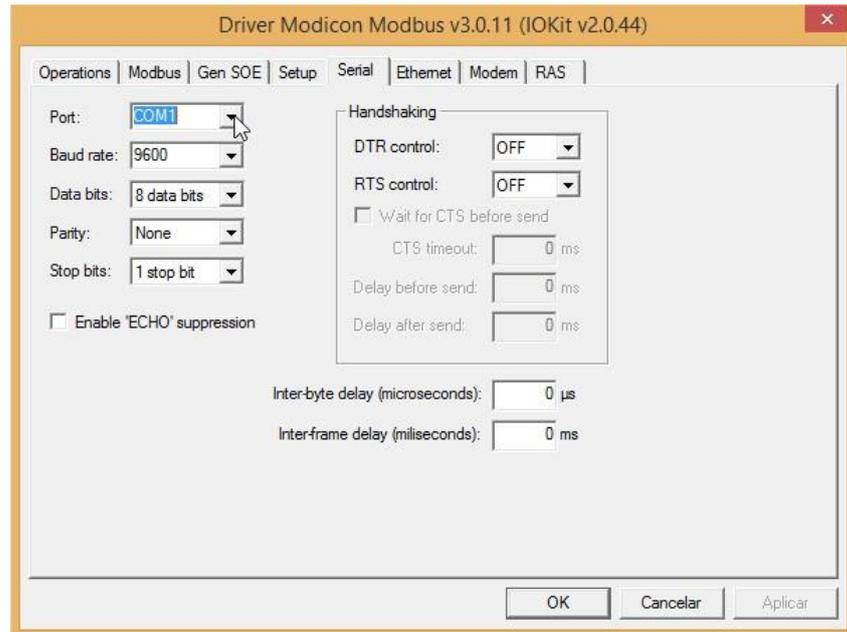
Passo 8 - Na janela “Propriedades do Driver”, clique no botão “Extras...”.



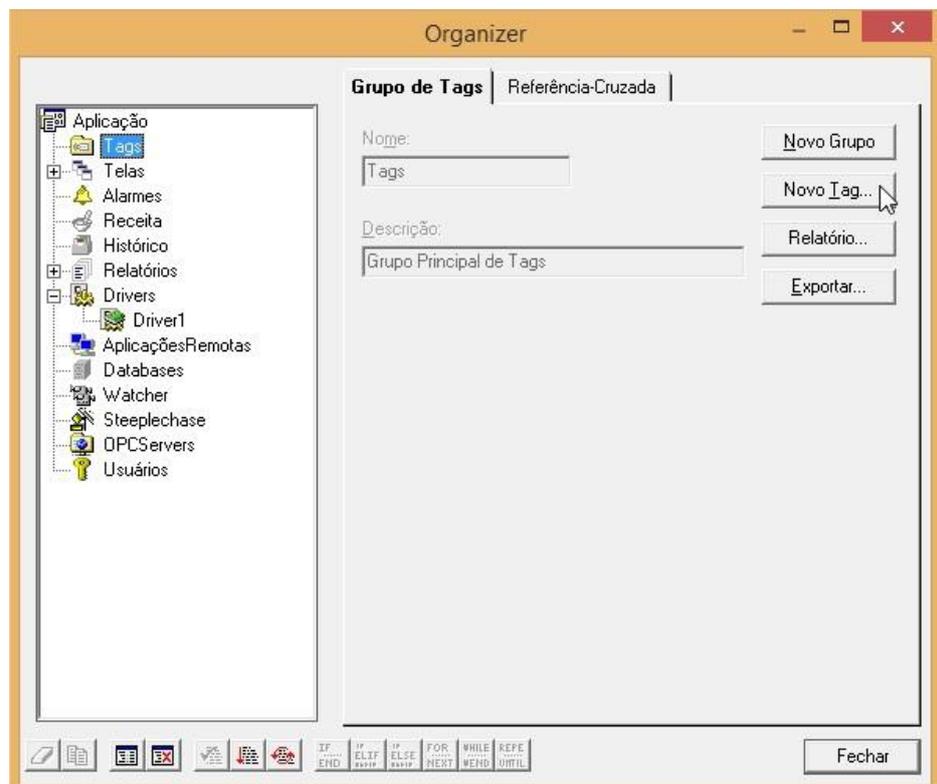
Passo 9 - Na janela de configuração do driver, clique na aba “Modbus” e se certifique que o modo do protocolo esteja correto na seção “Modbus Mode”.



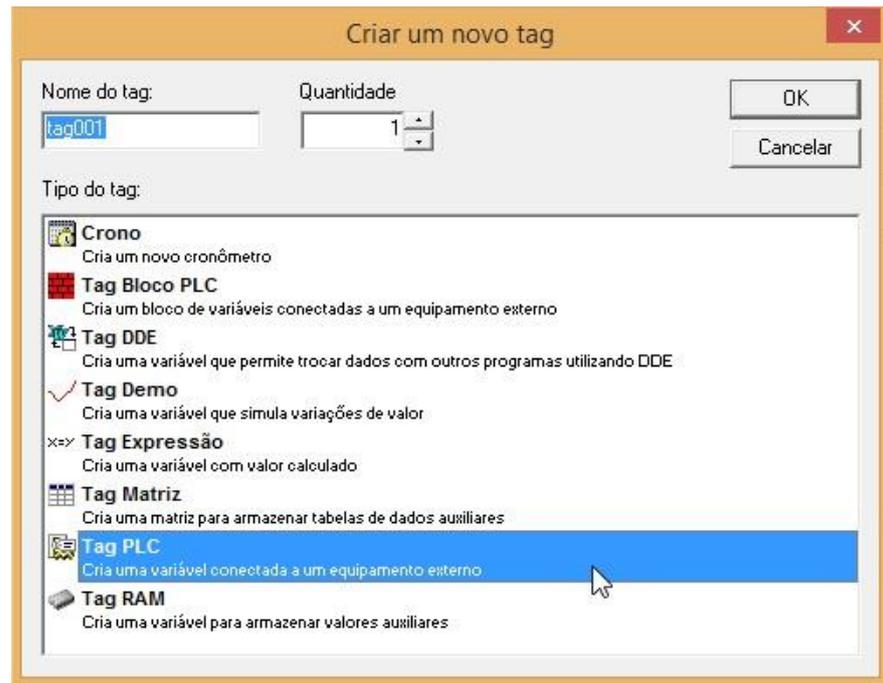
Passo 10 - Na aba “Serial”, verifique se os itens Port, Baud rate, Data bits, Parity e Stop bits estão de acordo com o que foi ajustado no inversor no grupo de parâmetros “Comunicação serial”.



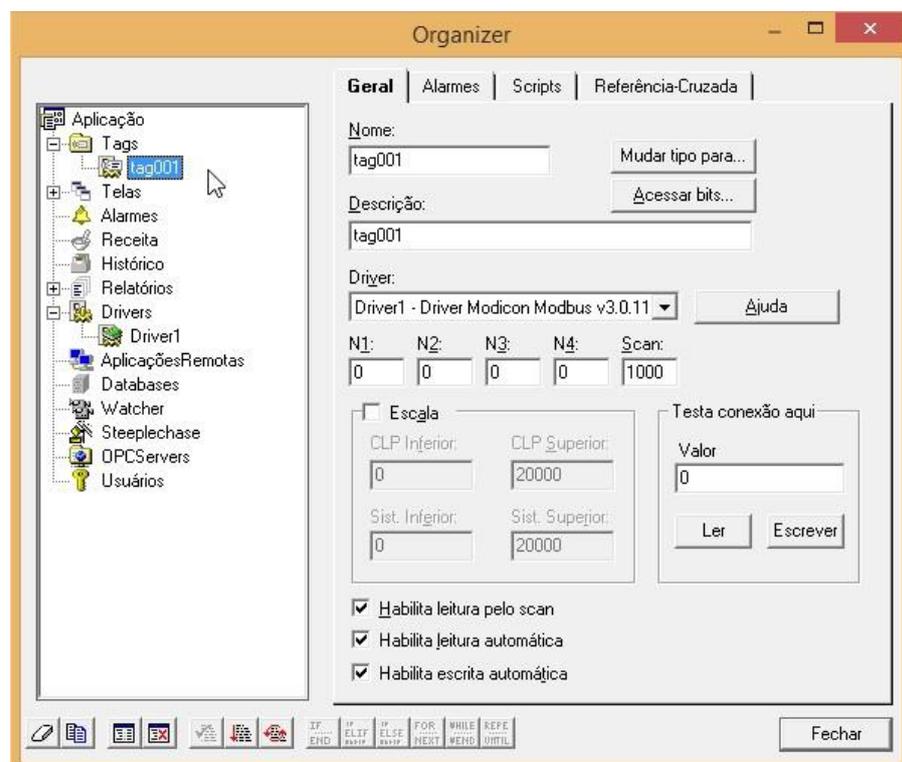
Passo 11 - De volta a janela principal do “Organizer”, clique na opção “Tags” no menu esquerdo e em seguida clique em “Novo Tag...”.



Passo 12 - Escolha a opção “Tag PLC” e clique em OK.

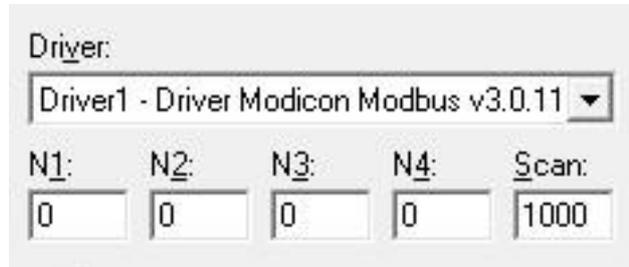


Passo 13 - Em seguida, clique na tag criada (tag001) para poder configurar seus parâmetros.



Passo 14 - Na tela de configuração da tag, é possível alterar os parâmetros N1 a N4. Estes parâmetros representam respectivamente o endereço do

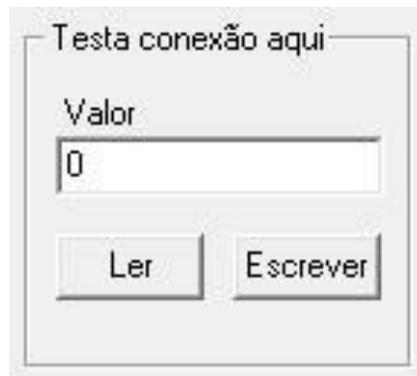
equipamento escravo, código da operação (leitura ou escrita), parâmetro adicional (geralmente não usado) e endereço do registrador que se deseja acessar.



Driver:
Driver1 - Driver Modicon Modbus v3.0.11 ▼

N1:	N2:	N3:	N4:	Scan:
0	0	0	0	1000

Passo 15 - Após inserir os parâmetros N1 a N4, teste a conexão no campo “Testa conexão aqui”.



Testa conexão aqui

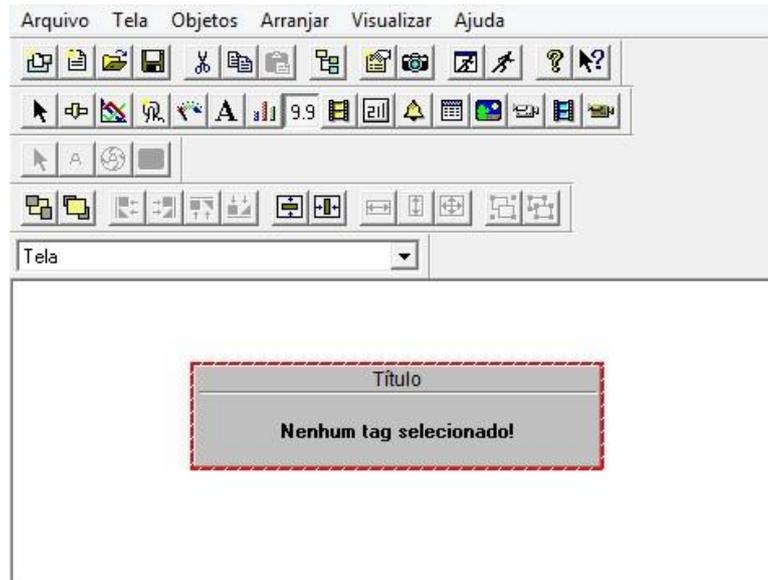
Valor
0

Ler Escrever

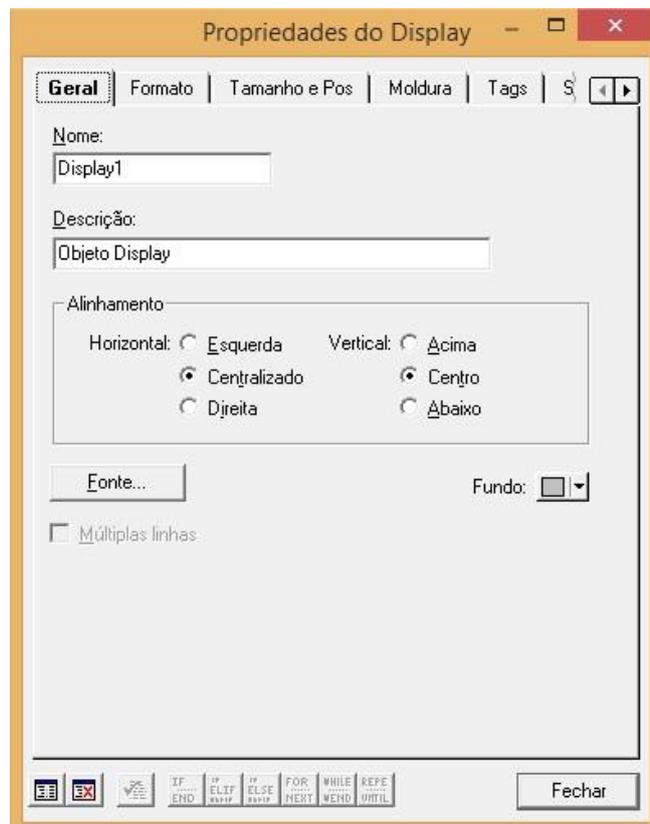
Passo 16 - Existem várias maneiras de exibir as informações de uma tag e neste exemplo iremos utilizar um display. Clique no botão “Display” no menu superior.



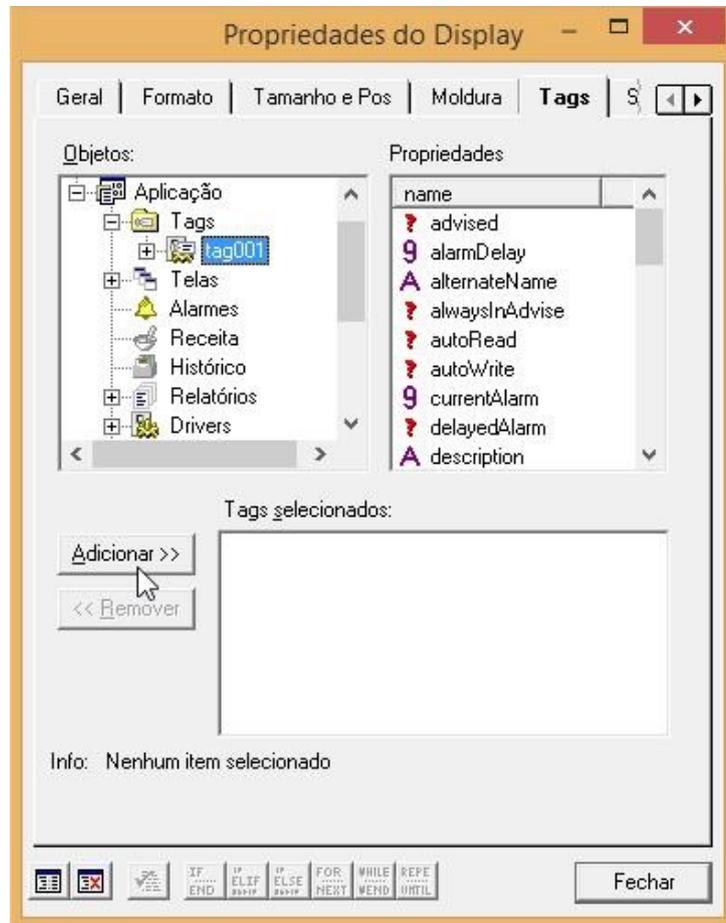
Passo 17 - Insira o display em um espaço vazio na tela de edição.



Passo 18 - Dê um duplo clique no display criado para abrir a janela “Propriedades do Display”. Nela é possível alterar o nome do botão, formato, tamanho, moldura entre outras opções.



Passo 19 - Clique na aba “Tags” para associar a tag criada em passos anteriores ao display recentemente criado. Para isso, localize a tag no campo “Objetos” e clique no botão “Adicionar” na parte inferior.



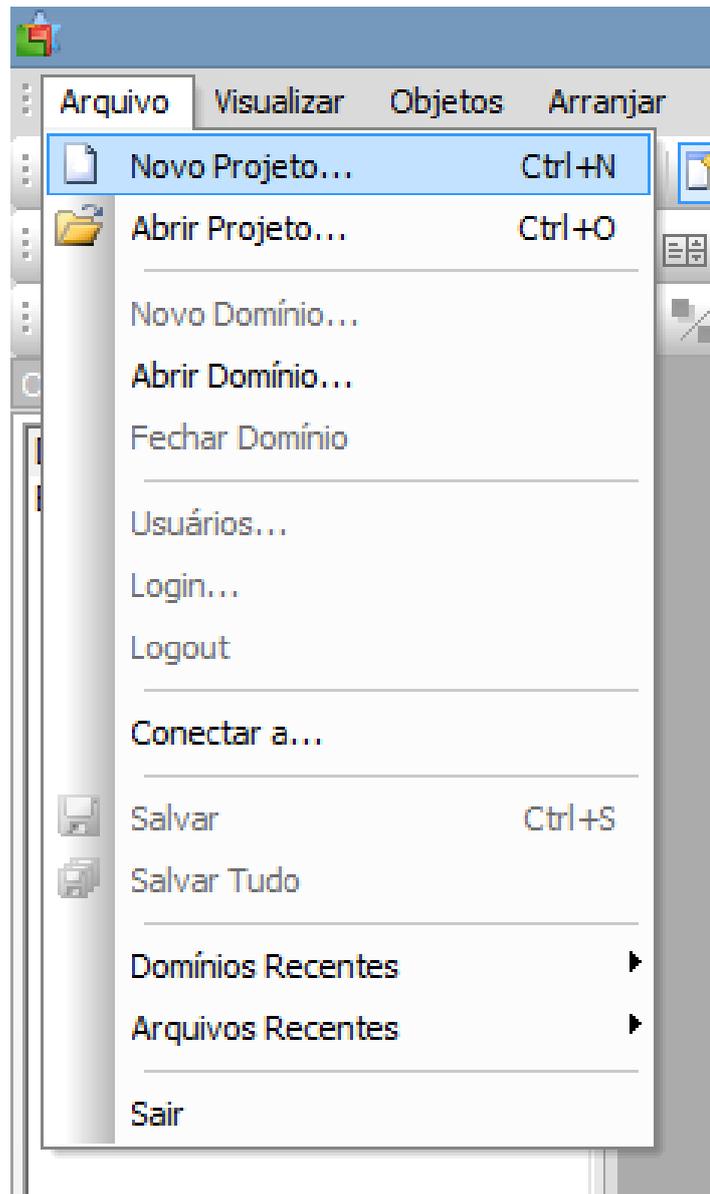
Passo 20 - Após a configuração do driver, criação e associação da tag, é possível executar a aplicação para ver o sistema supervisório criado funcionar. Para isso clique no botão “Monitara a Tela” no menu superior ou aperte o atalho “F8”.



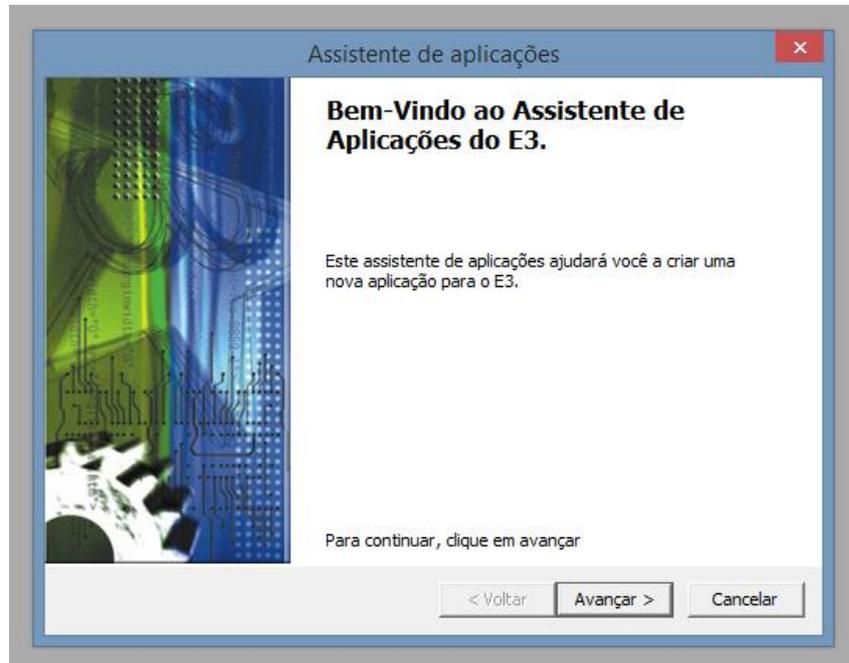
ANEXO C - CONFIGURAÇÃO DO SISTEMA SUPERVISÓRIO (CLIENTE)

Este anexo tem como objetivo auxiliar o usuário na configuração do driver de comunicação OPC e tags no sistema supervisório criado no software Elipse E3 atuando como cliente.

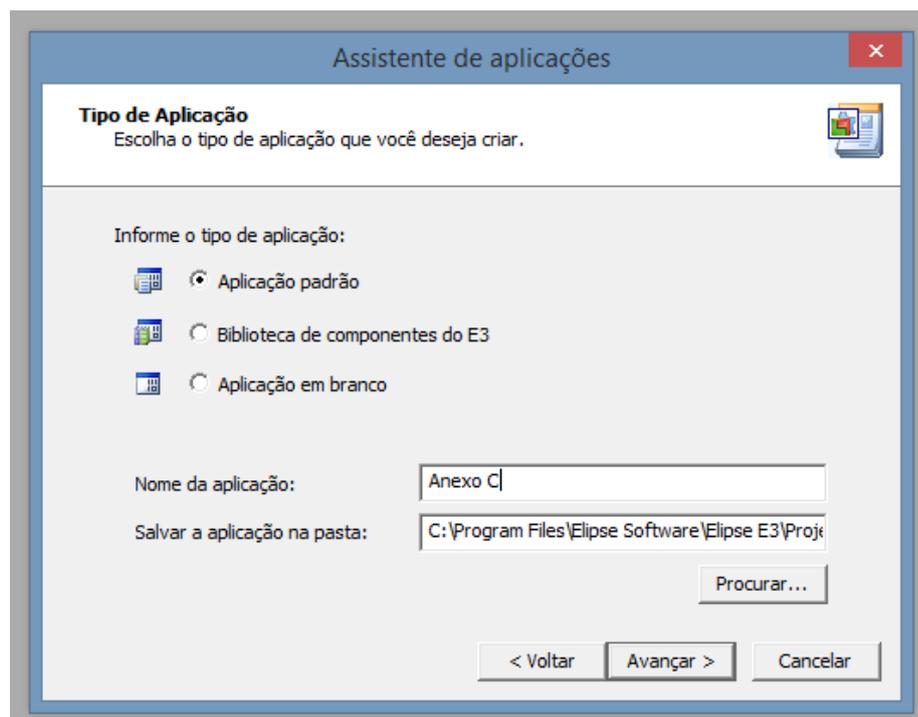
Passo 1 - Abra o software Elipse E3 e clique em “Novo Projeto” na aba “Arquivo”.



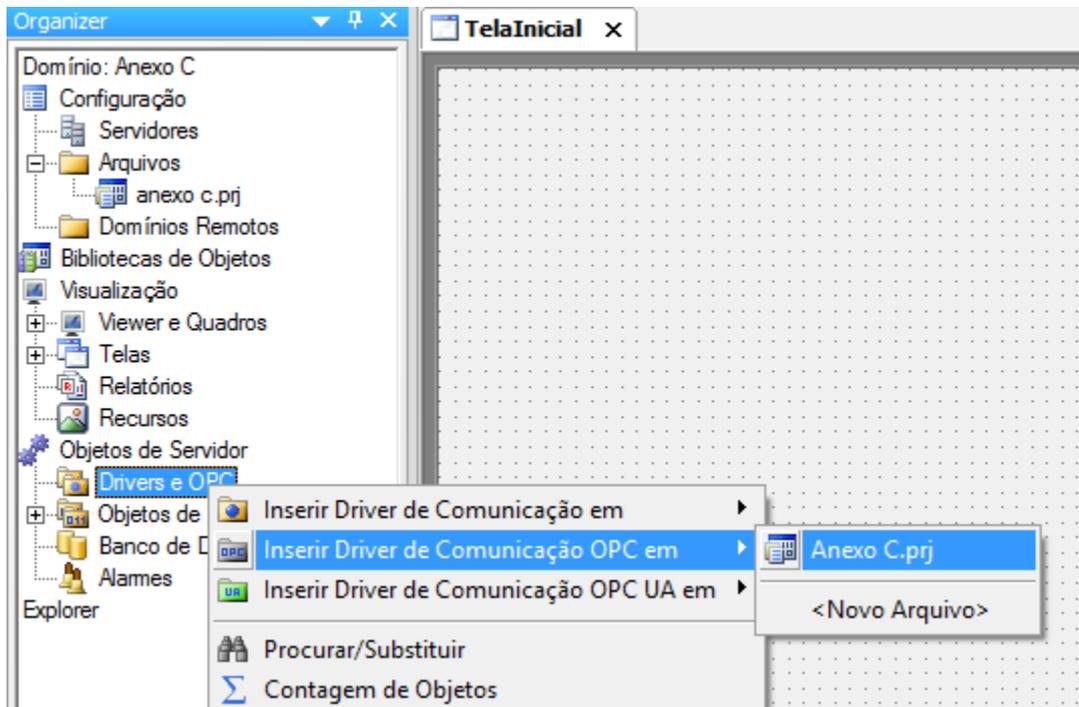
Passo 2 - Uma janela se abrirá, clique em avançar.



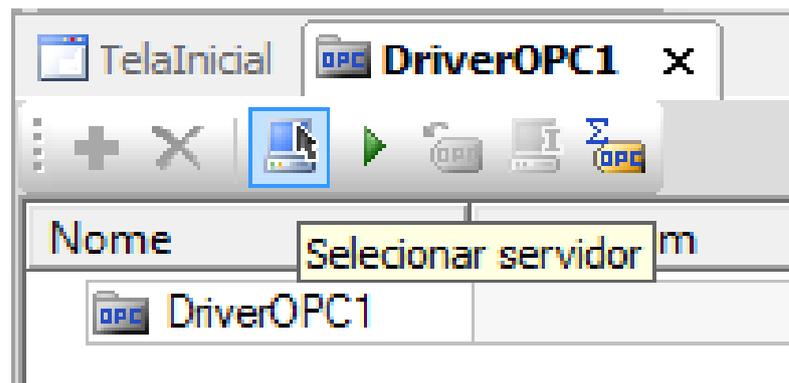
Passo 3 - Coloque o nome do seu projeto no campo “Nome da aplicação” e em seguinte clique em avançar até o final da configuração.



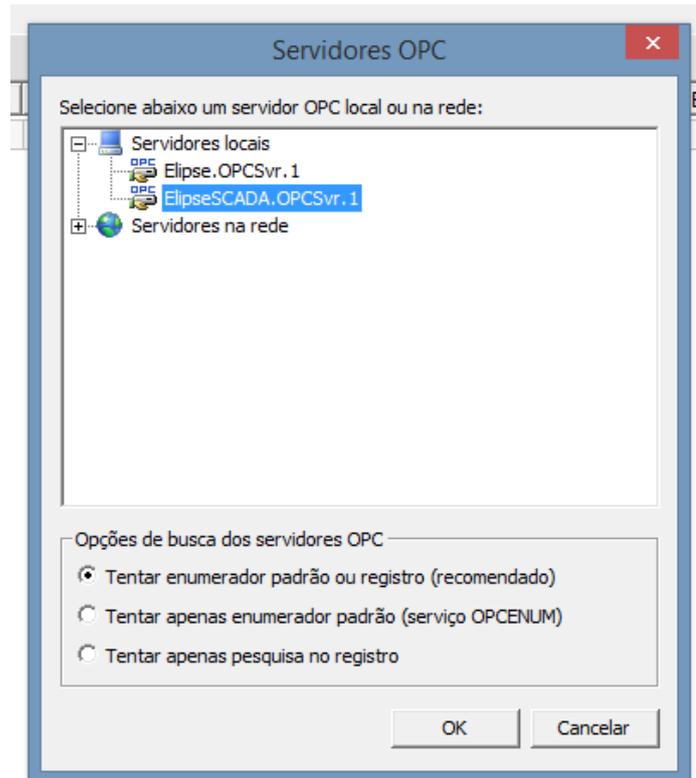
Passo 4 - Clique com o botão direito em “Drivers e OPC” na janela “organizer”, em seguida clique em “Inserir Driver de Comunicação OPC em” e depois clique no nome escolhido para seu projeto.



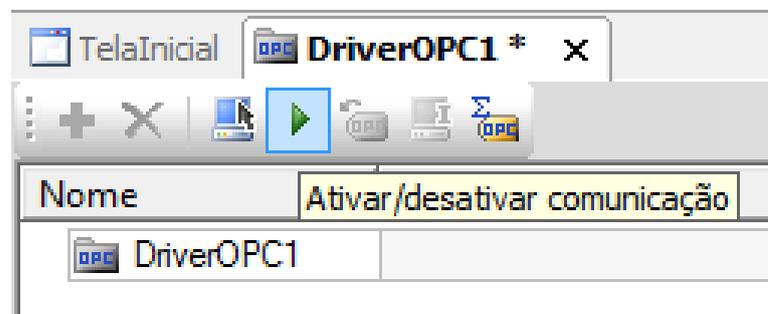
Passo 5 - Clique no botão “Selecionar servidor”.



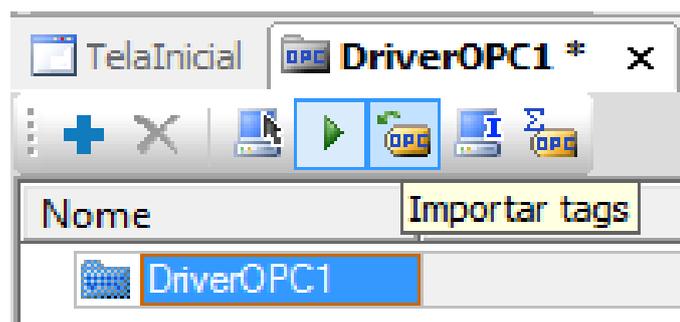
Passo 6 - Em seguida, na janela que se abre, clique no servidor OPC desejado, neste exemplo usou-se um servidor local.



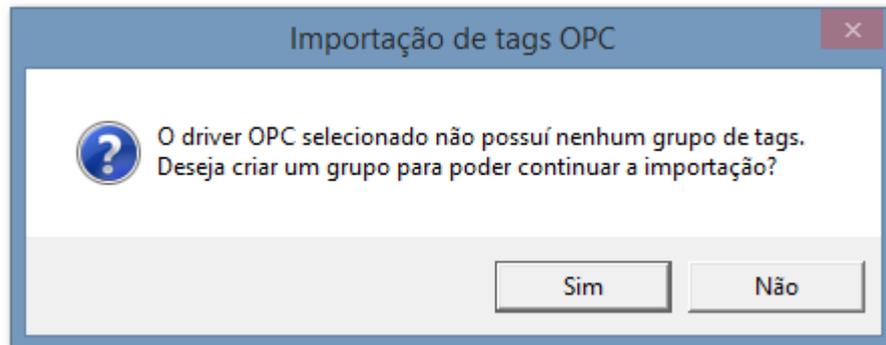
Passo 7 - Clique no botão “Ativar/desativar comunicação”.



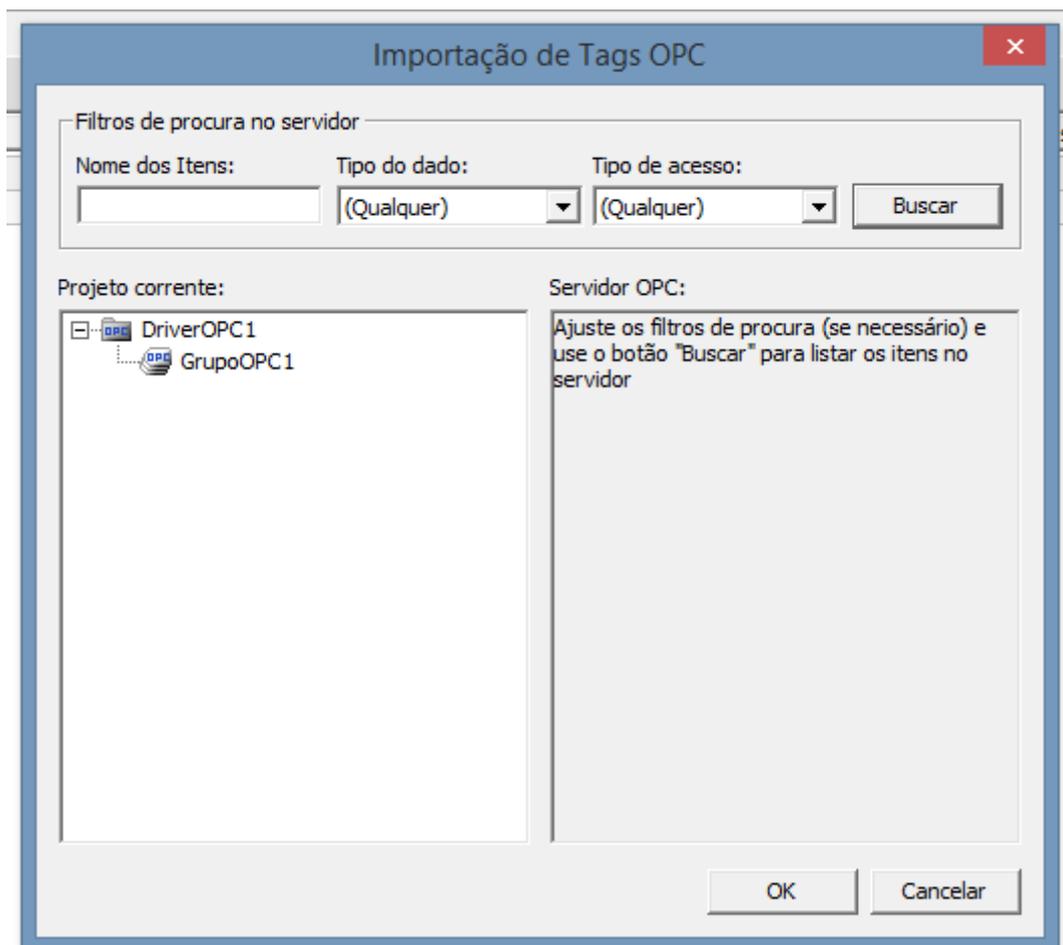
Passo 8 - Clique no botão “Importar tags”.



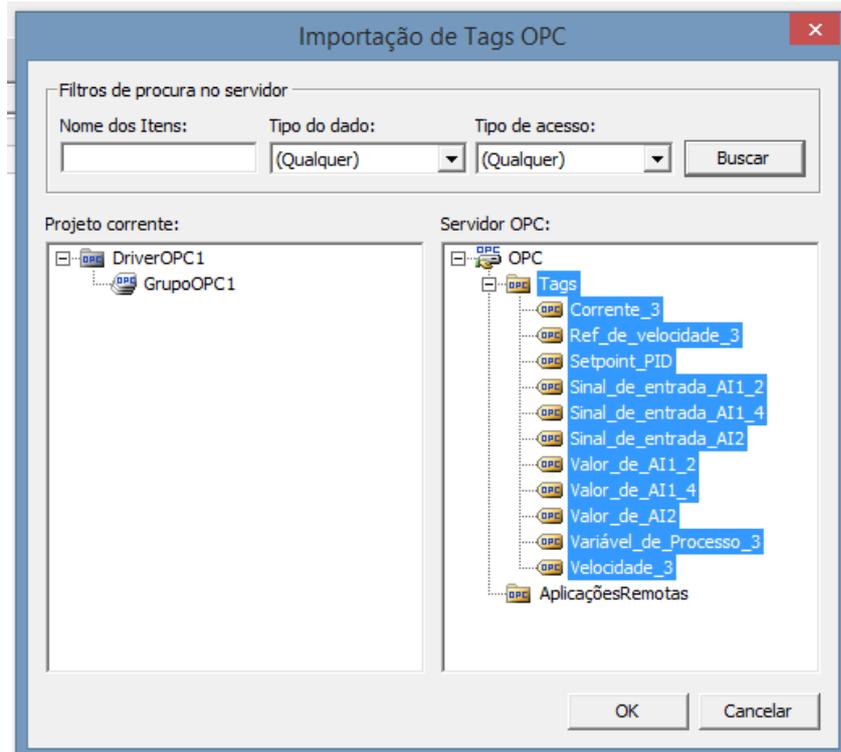
Passo 9 - Uma janela se abrirá perguntando se o usuário deseja criar um grupo de tags. Clique em no botão “Sim”.



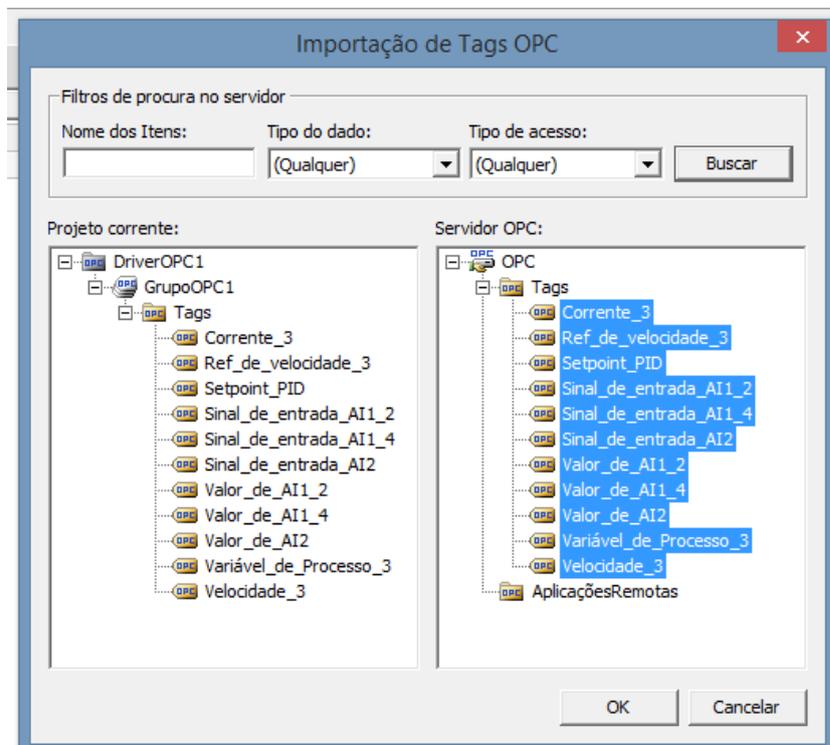
Passo 10 - Na janela que se abre, clique no botão “Buscar”.



Passo 11 - Em seguida, clique na pasta “Tags” para selecionar todas as tags inseridas nela.



Passo 12 - Arraste a pasta tags para dentro da pasta “Grupo OPC1” para copiar as tags.



Passo 13 - Agora é possível monitorar as tags recebidas do servidor. Para salvar o progresso, clique em “Salvar” na aba “Arquivo”.

Nome	ID do Item	Var...	Valor	Qualid...	Estampa de tempo	Valor (sem escala)
DriverOPC1						
GrupoOPC1		1000				
Tags						
Coment	Tags.Comente_3		Null	0	26/05/2016 16:08:18,853	Null
Ref_de	Tags.Ref_de_velocidade_3		Null	0	26/05/2016 16:08:18,853	Null
Setpoir	Tags.Setpoint_PID		Null	0	26/05/2016 16:08:18,853	Null
Sinal_c	Tags.Sinal_de_entrada_AI1_2		Null	0	26/05/2016 16:08:18,853	Null
Sinal_c	Tags.Sinal_de_entrada_AI1_4		Null	0	26/05/2016 16:08:18,853	Null
Sinal_c	Tags.Sinal_de_entrada_AI2		Null	0	26/05/2016 16:08:18,853	Null
Valor_c	Tags.Valor_de_AI1_2		Null	0	26/05/2016 16:08:18,853	Null
Valor_c	Tags.Valor_de_AI1_4		Null	0	26/05/2016 16:08:18,853	Null
Valor_c	Tags.Valor_de_AI2		Null	0	26/05/2016 16:08:18,853	Null
Variável	Tags.Variável_de_Processo_3		Null	0	26/05/2016 16:08:18,853	Null
Velocic	Tags.Velocidade_3		Null	0	26/05/2016 16:08:18,853	Null