

**FUNDAÇÃO OSWALDO ARANHA
CENTRO UNIVERSITÁRIO DE VOLTA REDONDA
CURSO DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA
TRABALHO DE CONCLUSÃO DE CURSO**

**FILIFE FERNANDES ALVES
LAÍS CAMPOS LANDIM**

**IMPLEMENTAÇÃO DE UM SISTEMA SUPERVISÓRIO NO
LABORATÓRIO DE AUTOMAÇÃO INDUSTRIAL**

**VOLTA REDONDA
2020**

**FUNDAÇÃO OSWALDO ARANHA
CENTRO UNIVERSITÁRIO DE VOLTA REDONDA
CURSO DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA
TRABALHO DE CONCLUSÃO DE CURSO**

**IMPLEMENTAÇÃO DE UM SISTEMA SUPERVISÓRIO NO
LABORATÓRIO DE AUTOMAÇÃO INDUSTRIAL**

Monografia apresentada ao Curso de Engenharia Elétrica do UniFOA como requisito à obtenção do título de Engenheiro Eletricista.

Alunos:

Filipe Fernandes Alves

Laís Campos Landim

Orientador:

Prof. M. Sc. Edson de Paula Carvalho

Coorientador:

Prof. Esp. Aloano Regio de Almeida Pereira

**VOLTA REDONDA
2020**

RESUMO

Este trabalho tem por objetivo explicar e realizar todo o processo de implantação de um sistema supervisório nos computadores do laboratório de automação industrial do Centro Universitário de Volta Redonda – UniFOA. Depois de apresentado todo o embasamento teórico, a implementação teórica e prática é descrita, através do uso do supervisório (Elipse SCADA), para que seja possível a comunicação deste software com os CLPs existentes no interior do laboratório, a fim de que o programa possa ser utilizado em aulas práticas, através de exemplos e criações de telas para supervisionar inúmeros processos programados nos controladores lógicos programáveis. Esta experiência na área de automação industrial serve para enriquecer o conhecimento dos alunos dos cursos que utilizam o laboratório e a quantidade de materiais disponíveis, até mesmo para os próprios professores e funcionários. Os softwares supervisórios são muito utilizados na maioria dos ambientes industriais atualmente, o que traz também experiências únicas para os alunos.

Palavras-chave: Supervisório, Elipse SCADA, CLP, Automação.

ABSTRACT

This work aims to explain and carry out the entire process of implementing a supervisory system in the computers of the industrial automation laboratory of Centro Universitário of Volta Redonda - UniFOA. After presenting all the theoretical basis, the theoretical and practical implementation is described, through the use of supervisory (Eclipse SCADA), so that it is possible to communicate this software with the existing PLCs inside the laboratory, so that the program can be used in practical classes, through examples and screen creations to supervise numerous processes programmed in programmable logic controllers. This experience in the area of industrial automation serves to enrich the knowledge of students of the courses that use the laboratory and the amount of materials available, even for the teachers and employees themselves. Supervisory software is widely used in most industrial environments today, which also brings unique experiences for students.

Keywords: Supervisory, Eclipse SCADA, CLP, Automation.

LISTA DE FIGURAS

Figura 1 – Esquema geral do CLP	13
Figura 2 – Ciclo de Varredura do CLP.....	16
Figura 3 – CLP de estrutura compacta.....	18
Figura 4 – CLP de estrutura modular.	19
Figura 5 – Portas lógicas, expressões lógicas e linguagem LADDER.....	20
Figura 6 – Sensores fim de curso.....	23
Figura 7 – Sensor indutivo.....	24
Figura 8 – Sensor capacitivo.	24
Figura 9 – Sensor fotoelétrico.....	25
Figura 10 – Sensor magnético.....	26
Figura 11 – Sensor magnético.....	26
Figura 12 – Sensor RFID.....	27
Figura 13 – Sensor de corrente contínua.	28
Figura 14 – Sensor de tensão AC.	29
Figura 15 – Sensor de tensão AC.	30
Figura 16 – Atuadores elétricos.....	30
Figura 17 – Atuadores pneumáticos.....	31
Figura 18 – Contatores.....	32
Figura 19 – Interface Homem-Maquina	34
Figura 20 – Sistema de controle com CLP	35
Figura 21 – Supervisório Laquis	36
Figura 22 – Elipse da Elipse Software.....	37
Figura 23 – Supervisório iFix.....	39
Figura 24 – Supervisório Indusoft.....	40
Figura 25 – Supervisório SCADA BR	41
Figura 26 – Exemplo de Tela de Supervisório.....	45
Figura 27 – Tela que aparece ao abrir o arquivo Elipse SCADA.....	47
Figura 28 – Tela inicial do Elipse SCADA.	47
Figura 29 – Tela da função organizer no Elipse SCADA.....	48

Figura 30 – Tela para escolha do driver de comunicação no Elipse SCADA.	49
Figura 31 – Tela que aparece o driver no Elipse SCADA.	49
Figura 32 – Tela para configuração de driver no Elipse SCADA.	50
Figura 33 – Aba Modbus de Comunicação no Elipse SCADA.	51
Figura 34 – Aba setup de comunicação no Elipse SCADA.	53
Figura 35 – Aba serial de comunicação no Elipse SCADA.	53
Figura 36 – Exemplo de tabela de pontos necessários para um projeto no Elipse SCADA.	54
Figura 37 – Tela principal (organizer) no Elipse SCADA.	55
Figura 38 – Tela para criar novo tag no Elipse SCADA.	55
Figura 39 – Tela para configurar uma tag criada no Elipse SCADA.	57
Figura 40 – Tela exemplo com uma lista de tags criadas no Elipse SCADA.	58
Figura 41 – Tela para teste conforme configuração de tags no Elipse SCADA.	58
Figura 42 – Tela de controle de nível de tanque criada no supervisório Elipse SCADA.	59
Figura 43 – Lista de tags utilizadas na tela.	60
Figura 44 – Tag LIGA.	60
Figura 45 – Tag DESLIGA.	61
Figura 46 – Tag NIVEL ALTO.	61
Figura 47 – Tag NIVEL BAIXO.	62
Figura 48 – Programação básica de um controle de nível de tanque criada no CLP TPW03.	62
Figura 49 – Fluxograma do Controle de Nível de Tanque.	63
Figura 50 – Tela do controle de nível de tanque acusando nível alto.	64

LISTA DE QUADROS

Quadro 1 – Critério de escolha dos sensores de proximidade.	23
Quadro 2 – Descrições dos valores de leitura no Elipse SCADA.	51
Quadro 3 – Descrições dos valores de leitura no Elipse SCADA.	52
Quadro 4 – Dados referentes ao CLP e ao Elipse SCADA.	56

LISTA DE ABREVIATURA, SIGLAS E SÍMBOLOS

AC	<i>Alternative Current</i> (Corrente Alternada)
CAN	Campus Área Network (Rede de Área do Campus)
CLP	Controlador Lógico Programável
CPU	<i>Central Processor Unit</i> (Unidade Central de Processamento)
EPI	Equipamento de Proteção Individual
HART	<i>Highway Addressable Remote Transduce</i> (Via de Dados Endereçável por Transdutor Remoto)
HMI	<i>Human Machine Interface</i>
I/O	<i>Input/Output</i> (Entrada/Saída)
IHM	Interface Homem Máquina
LAN	Local Área Network (Rede de Área Local)
LED	<i>Light Emitting Diode</i> (Diodo Emissor de Luz)
MAC	<i>Media Access Control</i> (Controle de Acesso de Mídia)
NA	Normalmente Aberto
NF	Normalmente Fechado
NR12	Norma Regulamentadora 12
OPC	<i>Open Platform Communications</i> (Plataforma Aberta de Comunicações)
OSI	<i>Open System Interconnection</i> (Sistema Aberto de Interconexão)
PID	Proporcional integral derivativo
PLC	<i>Programmable Logic Controller</i>
PV	Variável Primária
QV	Variável Quartenária
RFID	<i>Radio Frequency Identification</i> (Identificação por Radiofrequência)
SCADA	<i>Supervisory Control and Data Acquisition</i> (Sistemas de Supervisão e Aquisição de Dados)
SV	Variável Secundária
TCP/IP	<i>Transmission Control Protocol</i> (Protocolo de Controle de Transmissão) / Internet Protocol (Protocolo de Internet)
TV	Variável Terciária

SUMÁRIO

1 INTRODUÇÃO	10
1.1 Objetivos.....	11
1.1.1 Objetivo Geral.....	11
1.1.2 Objetivos Específicos.....	11
1.2 Metodologia.....	11
2 REVISÃO BIBLIOGRÁFICA	12
2.1 Controlador Lógico Programável – CLP	12
2.2 Componentes do CLP	13
2.3 Componentes de Hardware.....	13
2.3.1 Módulos de Entrada	13
2.3.2 Módulos de Saída	14
2.4 CPU.....	15
2.5 Classificação dos CLPs quanto a tamanhos e modelos.....	17
2.6 Linguagem de programação dos CLPs	19
2.7 Redes.....	20
2.8 Sensores	21
2.8.1 Sensores discretos	22
2.8.1.1 Sensores fim de curso.....	22
2.8.2 Sensor de proximidade.....	23
2.8.2.1 Sensores indutivos	23
2.8.2.2 Sensores capacitivos.....	24
2.8.2.3 Sensor fotoelétrico.....	24
2.8.2.4 Sensor magnético	25
2.8.2.5 Sensor ultrassônico.....	26
2.8.2.6 Sensor RFID.....	27

2.8.2.7 Sensor de medição de corrente	27
2.8.2.8 Sensor de medição de tensão.....	28
2.9 Atuadores	29
2.9.1 Atuadores hidráulicos	29
2.9.2 Atuadores elétricos.....	30
2.9.3 Atuadores pneumáticos.....	31
2.10 Contatores.....	31
2.11 Supervisório	32
2.11.1 Monitoramento por wireless	33
2.11.2 Monitoramento por IHM.....	33
2.11.3 Sistema de controle com CLP	35
2.11.3.1 Elipse da Elipse Software	36
2.11.3.2 Laquis.....	37
2.11.3.3 iFIX da General Electric.....	38
2.11.3.4 InduSoft Web Studio.....	39
2.11.3.5 SCADA BR	40
3 DESENVOLVIMENTO APLICADO AO LABORATÓRIO.....	42
3.1 Lista de Materiais	42
3.2 Normas aplicadas a um laboratório (educacional)	43
3.2.1 Normas aplicadas a laboratórios	43
3.2.2 Adequação dos laboratórios.....	44
3.3 Implementação	45
3.3.1 Carregar o driver de comunicação	48
3.3.2 Configurar driver de comunicação ModBus.....	50
3.3.3 Tipo de comunicação	52
3.3.4 Configuração da porta de comunicação	53
3.3.5 Criação de TAGS	54
3.4 Aplicação para o laboratório.....	59
4 CONCLUSÃO.....	65
REFERÊNCIAS.....	66
ANEXO A: PROTOCOLO MODICON MODBUS PARA ELIPSE SCADA	69

1 INTRODUÇÃO

Segundo Brusamarello (2007) no século atual, a automação industrial é de extrema importância para o desenvolvimento de qualquer indústria de qualquer ramo. Partindo do ponto de vista de que uma indústria não pode parar, trabalhar com máquinas é essencial. Para que isso seja possível, é fundamental que tais ferramentas sejam, além de operadas, automatizadas para que as empresas atinjam o fim comum de produzir o seu produto.

A automação industrial teve berço na década de 1950, época que começaram a surgir mecanismos para movimentar automaticamente os materiais. Após a 1ª Revolução Industrial, ocorrida no século XVIII, na Inglaterra, passaram a ser utilizados na produção máquinas a vapor e teares. Depois de um tempo, no século XIX, veio a 2ª Revolução, na qual, pela primeira vez, ocorreu o emprego da eletricidade no processo produtivo. Já no século XX, na 3ª Revolução, a grande demanda dos processos produtivos implicou no avanço desenfreado das tecnologias e dos computadores, proporcionando avanços significativos para a indústria. Hoje em dia estamos passando pelo que podemos chamar de 4ª Revolução Industrial, ou Indústria 4.0, que é um conceito muito difundido no que diz respeito às trocas de dados que ocorrem entre os mecanismos de produção dentro das indústrias aliadas à globalização e modernização dos equipamentos. Ex.: Internet das Coisas e Computação em Nuvem.

A necessidade da existência de máquinas automatizadas hoje é muito grande, pois não apenas a produção está aumentando, mas também devido a viabilidade econômica de cada empresa no que se diz respeito a auto sustentabilidade com seus próprios produtos, diante dos custos mais baixos de componentes de automação das máquinas, da qualidade e da estabilidade de novos produtos, que substituem ocupações de risco dos trabalhadores.

1.1 Objetivos

1.1.1 Objetivo Geral

O objetivo geral deste trabalho é enriquecer o aprendizado e ampliar o conhecimento dos alunos que frequentam o laboratório de Automação Industrial do Centro Universitário de Volta Redonda – UniFOA em suas aulas neste local, por meio da inserção de um novo recurso técnico e muito utilizado atualmente, que é o sistema supervisório.

1.1.2 Objetivos Específicos

Os objetivos específicos deste trabalho variam desde a aquisição e a instalação do software supervisório dentro do laboratório até a experimentação deste por meio da criação de telas. São eles:

- Instalação do software supervisório nos computadores do laboratório;
- Execução bem sucedida da configuração de comunicação do supervisório com os CLPs do laboratório;
- Estudo teórico e prático de CLPs, supervisórios, programações e telas;
- Adequação do ambiente do laboratório ao supervisório e vice versa;
- Possíveis testes futuros para o supervisório em processos existentes no laboratório.

1.2 Metodologia

A metodologia adotada para a construção de toda a parte de fundamentação teórica foi a prática de pesquisas acadêmicas em diversas referências. O segmento prático desta obra foi embasado em conceitos da área de Automação, além da experimentação do uso do CLP TPW03 na produção de uma programação e do software supervisório Elipse na criação de uma simulação de uma tela exemplo. A montagem da tela foi feita depois de muitas pesquisas, possibilitando aos autores uma grande oportunidade de explorar este software até aprender a utilizá-lo.

2 REVISÃO BIBLIOGRÁFICA

Para que ocorra efetivamente a implementação do sistema supervisorio no laboratório, é necessário ter conhecimento de todos os pré-requisitos e equipamentos necessários. Desta maneira, neste capítulo, serão explicados todos os equipamentos utilizados para a realização do procedimento prático do trabalho. São eles: CLP, tipos de Rede, Softwares Supervisorios e sensores.

2.1 Controlador Lógico Programável – CLP

Segundo Antonelli (1998), foi desenvolvido pela General Motors na década de 60, inicialmente conhecido como CP (Controlador Programável). Depois, tornou-se CLP (Controlador Lógico Programável), ou, do inglês, PLC (*Programmable Logic Controller*). No início, ele era utilizado apenas para intertravamentos e, somente na década de 70, foram acrescentadas a esse programa funções lógicas, temporizações, contagens e manipulações de dados. Vale ressaltar também que no fim da década de 70 os CLP'S começaram a realizar funções muito mais avançadas como controle PID (Proporcional – Integral – Derivativo).

Defina-se então o CLP, ou Controlador Lógico Programável, como a inteligência no processo de automatização, é um computador que controla funções específicas através de um programa criado por um ser humano. O CLP, diferente do computador convencional que utilizamos no dia a dia, é programado para executar processos com mais precisão, rapidez, efetividade e eficiência, devido a alta competitividade do mercado em um tempo em que a tecnologia reina.

Assim, com a utilização do Controlador Lógico Programável, basta um computador e uma pessoa para alterar o programa de forma simples, o que tornou extremamente possível simplificar a alteração dos processos, reduzindo tempo e mão-de-obra e obtendo lucros significativos.

Dentre as vantagens na utilização do CLP, temos:

- Opção mais viável na redução de custos;
- Permitem a realização de operações mais complexas;
- São equipamentos de fácil configuração e programação;

- Possuem fácil manutenção e resistência;
- Podem ser controlados a distância;
- Grande versatilidade.

2.2 Componentes do CLP

Segundo Franchi (2011), como todo equipamento, o CLP também deve funcionar de alguma maneira. Para que o funcionamento seja possível, serão apresentadas as principais partes do CLP, no que diz respeito tanto a parte física (hardware) quanto a parte programável (software).

São elas: os módulos de entrada, a unidade central de processamento e os módulos de saída, como ilustrado na figura 1.



Figura 1– Esquema geral do CLP

Fonte: <http://saladaautomacao.com.br/como-funciona-o-clp/> Acesso em 15 fev.2020

2.3 Componentes de Hardware

Os componentes da parte física do CLP são: módulos de entrada e módulos de saída. (FRANCHI, 2011).

2.3.1 Módulos de Entrada

Os cartões de entrada são aqueles que fornecem as entradas em corrente contínua ou alternada, as quais são responsáveis pelo recolhimento das informações. Recebem os comandos de um sistema e encaminham para a CPU, onde é feito o processamento de tudo que está sendo enviado.

Como exemplos de sinais de entrada, pode-se destacar: botoeiras, contatos de relés, sensores (de presença de peças, de temperatura, de vazão, de velocidade etc.), encoders, pressostatos, chaves de fim de curso e todos os tipos de dispositivos usados para monitorar o processo e fornecer um retorno de informação ao CLP.

As entradas do CLP podem ser internas e externas. Nas entradas externas, o sinal é proveniente de um sensor, enquanto nas internas ele é originado por alguma outra entrada dentro do próprio equipamento. Além disso, existem também as entradas digitais e as entradas analógicas.

As entradas analógicas são aquelas que recebem sinais analógicos, que variam sua intensidade de acordo com o tempo. Esse tipo de sinal é utilizado para representar grandezas físicas que variam com o tempo (vazão, temperatura, nível, deslocamento). Tem-se como exemplo um sinal enviado por um gerador para controlar a rotação de um motor. A tensão aumenta continuamente à medida que aumenta a rotação do motor. Este sinal trabalha com um alcance de valores em corrente ou tensão que variam sua intensidade de acordo com a variação da grandeza apresentada. Geralmente, esse sinal trabalha com valores de corrente de 4 a 20mA e tensão de 0 a 10V.

As entradas digitais são aquelas que recebem sinais digitais – também conhecidos como sinais lógicos ou discretos – que só possuem dois valores que são denominados de nível alto, representado pelo algarismo 1 (ligado/fechado: com sinal elétrico no CLP), e nível baixo, representado pelo algarismo 0 (desligado/aberto: sem sinal elétrico no CLP). Como exemplo deste sinal, existem os botões em um interruptor. Quando o botão está atuado, a entrada digital do CLP reconhece o sinal elétrico como 1 (ligado → enviando sinal elétrico). Quando não está atuado, a entrada digital do CLP reconhece o sinal elétrico como 0 (desligado → não enviando sinal elétrico) (FRANCHI, 2011).

2.3.2 Módulos de Saída

Os cartões de saída são pontos de conexão onde ligamos os dispositivos de campo e pelos quais o CLP pode alimentar uma carga. Este acionamento é realizado através do envio de sinal elétrico do equipamento para os dispositivos de saída propriamente ditos.

Como exemplos de sinais de saída, pode-se destacar: contatores que partem motores, sinalizadores, válvulas solenoides, inversores de frequência, colunas luminosas etc.

As saídas, como as entradas, também podem ser externas e internas. As saídas externas são aquelas responsáveis por mandarem sinais a componentes externos ao CLP, como é o caso de comandar um motor, por exemplo. Já as saídas internas podem ser as bobinas de um temporizador, isto é, componentes internos do controlador.

Tais quais as entradas, as saídas também podem ser analógicas ou digitais.

As saídas analógicas são as interfaces através das quais o CLP pode variar a tensão ou a corrente sobre uma carga, continuamente no tempo. É o caso de um motor cuja velocidade oscila através do uso de um inversor de frequência. Cada variação na intensidade do sinal analógico é uma variação na velocidade do motor.

Como as entradas desse tipo, as saídas digitais são aquelas que recebem um sinal digital, ou seja, um sinal cujos valores só podem ser de nível alto (1) ou baixo (0). Alguns exemplos de saídas digitais são lâmpadas, qualquer dispositivo que exija ser alimentado com tensão nominal ou desligado, válvulas solenoides, entre outros. Como sinais digitais de saída, existem três categorias: saída a transistor (sinal de saída 24Vcc, com capacidade máxima de corrente 0,5A), saída a triac/tiristor (sinal de saída 110Vac, com capacidade máxima de corrente entre 0,5 e 1A e a saída a relé, que trabalha com qualquer valor de tensão numa corrente máxima de 5A – isso acontece pois a saída aciona um relé interno que disponibiliza um contato para o usuário (FRANCHI, 2011).

2.4 CPU

De acordo com Antonelli (1998), no interior do CLP existe a chamada CPU (*Central Processor Unit.*) em português, Unidade Central de Processamento. A CPU nada mais é que o “cérebro” do CLP, é para onde são jogadas as informações provindas dos sinais de entrada e de onde são transferidas essas informações atualizadas para os sinais de saída. A CPU permite que o CLP leia diferentes dados e realize inúmeras operações para que o programador atinja seu objetivo final.

É formada por um circuito eletrônico composto de microprocessadores e memórias programáveis pelo usuário. Esta programação é baseada na lógica de comandos elétricos, feita através de um microcomputador.

Para que ocorra a inicialização do CLP, acontece um processo denominado ciclo de varredura (ou scan). Dentro desse ciclo, primeiro acontece a varredura das entradas: a CPU lê todas as entradas e guarda essas informações como Memória Imagem de Entrada. Logo em seguida, é feita a varredura do programa, na qual as informações da Memória Imagem de Entrada são interpretadas de acordo com a programação realizada anteriormente pelo usuário, e, segundo a lógica dessa programação, modifica os estados de saída e armazena essas informações na Memória Imagem de Saída. Por último, na varredura das saídas, as saídas são atualizadas conforme foi registrado na Memória Imagem de Saída. Este ciclo é repetitivo e seu tempo depende do tamanho da programação, além do número de entradas e de saídas. A figura 2 representa o esquema de funcionamento do CLP.



Figura 2 – Ciclo de Varredura do CLP

Fonte: <http://saladaautomacao.com.br/como-funciona-o-clp/>. Acesso em 15 fev. 2020

O tempo máximo permitido para a execução do Scan é chamado de *Watchdog Timer* (temporizador “cão de guarda”). Se esse tempo for excedido ocorre

um erro fatal, a CPU força o modo de programação e as saídas são desligadas. Caso contrário, o valor do scan time (varredura) é armazenado numa variável apropriada e é reiniciado junto com o *Watchdog*.

Todos os erros que possam vir a ocorrer são sinalizados por flags (bits internos da CPU), ou por LEDs externos. Algumas CPUs também dispõem de uma variável destinada ao armazenamento do código de erro ocorrido durante a execução do último scan (ANTONELLI ,1998).

2.5 Classificação dos CLPs quanto a tamanhos e modelos

Os CLPs são produzidos por diversas empresas, como por exemplo a WEG, Telemecanique, Siemens, Rockwell e etc. Existem algumas maneiras de caracterizar o dimensionamento físico de um CLP: a quantidade e características de I/O (entrada/saída), a capacidade de memória da CPU, a alimentação e funções específicas e as condições ambientais.

Um CLP de estrutura compacta tem por característica ser pequeno e alimentado pela rede elétrica (110/220 Vac), através da figura 3 pode-se observar um modelo com essas descrições.

Os componentes de um CLP compacto são:

- CPU;
- Entradas;
- Saídas;
- Fonte interna de 24 Vcc.



Figura 3 – CLP de estrutura compacta

Fonte: <https://www.dixautomacao.com.br/produtos/rele-programavel/rele-programavel-compacto>. Acesso em 15 fev. 2020

Já um CLP de estrutura modular representado pela figura 4, tem por características, ser fabricados em tamanhos maiores e possuir maior quantidade de componentes, sendo eles:

- CPU: a memória e o tempo de processamento variam de fabricante ara fabricante;
- Módulos de entradas e saídas: a tensão de operação, a capacidade de corrente e a quantidade de pontos variam. Existem módulos que realizam funções especiais (pesagem, redes de comunicação, entre outras);
- Módulo de alimentação: alimenta os módulos. Varia na capacidade de corrente;
- Rack: estrutura para alojar os módulos (cartões). Varia no tamanho.



Figura 4 – CLP de estrutura modular.

Fonte: <https://www.altus.com.br/busca?palavrachave=CLP+de+estrutura+modular>. Acesso 27 fev. 2020.

2.6 Linguagem de programação dos CLPs

Segundo Moraes e Castrucci (2007), existem alguns tipos de linguagens com as quais se pode utilizar e programar um CLP, mas a principal e mais usada é a linguagem LADDER. Esse “idioma” se tornou um padrão dentre todas as formas de programação, no que diz respeito a um CLP, devido a sua simplicidade e a sua similaridade com a linguagem de relés utilizada no desenvolvimento de circuitos elétricos convencionais.

A Eletrônica Digital, tem como base a Álgebra de Boole para manejo dos bits de programação (exemplos: A e B). Dentro dessa teoria, existem várias portas lógicas que efetuam a operação entre os bits. Na figura 5 é possível observar a correlação entre as portas lógicas NOT, AND e OR com as suas respectivas representações na linguagem LADDER.

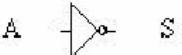
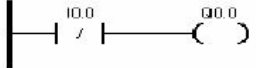
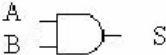
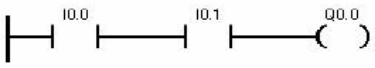

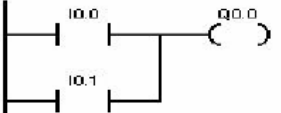
Portas Lógicas	Símbolo	Expressão	Ladder
NOT		$S = \bar{A}$	
AND		$S = A \cdot B$	
OR		$S = A + B$	

Figura 5 – Portas lógicas, expressões lógicas e linguagem LADDER.

Fonte: <https://ensinandoeletrica.blogspot.com/2016/02/clp-controladores-logicos-programaveis.html>. Acesso 27 fev. 2020

IO.0 e IO.1 são bits de entrada, enquanto Q0.0 é um bit de saída. Mais adiante, será especificado o CLP utilizado para o experimento prático deste trabalho.

2.7 Redes

Carvalho (2003) diz que toda rede de comunicação, de computadores, de dados e outras, se trata de um conjunto de dois ou mais computadores que se relacionam, estando interligados por meio de um sistema de comunicação digital. As máquinas falam a mesma linguagem e, assim, conseguem conversar uma com a outra.

Uma rede de comunicação tem como principal atributo o direcionamento da distribuição de dados dentro de procedimentos industriais, gerenciando inúmeros processos de automação de forma rápida e eficiente na troca de informações entre os componentes, tais como: sensores, computadores, etc. Todo o processo é realizado de forma programada, levando em conta o ambiente que está sendo utilizado.

- **Rede Modbus:** Segundo Tanenbaum (1994), o protocolo modbus é um conjunto de regras e convenções para conversação. Essas regras definem a comunicação entre dois equipamentos, sejam eles computadores ou máquinas. Nos protocolos são definidas as sintaxes de como os equipamentos irão ordenar os dados de forma que fiquem entendidos por ambos os lados que fazem parte da comunicação. Este protocolo define uma estrutura de mensagem que os controladores reconhecerão e usarão, independentemente do tipo de rede acima deles. O protocolo Modbus descreve também o processo que um controlador usa para pedir acesso a outros

dispositivos, como responderá a pedidos desses outros dispositivos, como serão descobertos erros da comunicação e como serão informados à sua origem. Em outras palavras, o protocolo fornece um formato comum para o plano e conteúdo de campos de mensagem. Sendo o Modbus uma linguagem informática independente do material, esta permite o diálogo entre equipamentos de natureza e construtores diferentes. Também é importante ressaltar que existem várias implementações do protocolo Modbus. Como sua especificação é bastante ampla, as vezes não é necessário implementar todas as suas especificações para se ter uma rede industrial padrão Modbus em funcionamento, ou seja, tem-se como padrão de mercado o formato simples de mensagens que o Modbus utiliza. Outra definição para o Modbus a nível de controladores é que ele provê o padrão interno que os controladores usam para analisar gramaticalmente as mensagens. É o Modbus que possibilita a um controlador que o mesmo reconheça uma mensagem dirigida a ele, determine o tipo de ação a ser efetuada e extraia os dados contidos na mensagem. Se uma resposta é requerida, o controlador construirá a mensagem de resposta e enviará a mesma usando o protocolo Modbus.

2.8 Sensores

Segundo Cassiolato e Alves (2008), O primeiro sensor foi desenvolvido ainda nos 50 como um meio para facilitar o processo industrial. Visto como a peça fundamental da automação e definido como a chave principal que converte um fenômeno físico em um sinal, o sensor é o sentido em um sistema de controle, sem sensor não há dados, popularmente definiria o sensor como a alma de um processo de controle.

A principal função de um sensor é interagir com o ambiente, possibilitando a medição de alguma alteração sofrida, sendo assim responsáveis por qualquer movimentação fabril como contagem de material, nível de fluidos e verificação do material dentro do recipiente.

Existem também sensores responsáveis pela segurança dos profissionais que operam máquinas que obedecem a norma regulamentadora (NR-12). Quando um funcionário tenta infringir qualquer norma de segurança que possa causar algum dano físico ao funcionário, a máquina desliga.

No ambiente industrial é mais comum encontrar várias aplicações como: sensores de pressão, sensores de temperatura, sensores de nível, sensores de vazão além dos mais comuns como indutivo, capacitivo, fotoelétrico, magnético e ultrassônico.

Segundo Bryan M. Bueno (Engenheiro) o funcionamento dos sensores é baseado na alteração do ambiente, como: aproximação, calor, luz e etc. por isso são infinitas as soluções possíveis que podemos adquirir com sensores, importante é saber escolher o sensor correto para cada atividade e precisão necessária.

Pode-se classificá-los como sensores discretos e sensores de medição, definindo-os como:

2.8.1 Sensores discretos

Segundo Cassiolato e Alves (2008), sinais elétricos com saídas binárias (0 - 1 / ON - OFF) são utilizados para detecção de eventos, por exemplo, chegada de um objeto a uma posição, um nível de um fluido a um valor etc. Dentro dessa classificação, duas principais classes são: os sensores discretos e os de proximidades.

2.8.1.1 Sensores fim de curso

A figura 6 mostra sensores fim de curso. Nestes sensores, é necessária uma força entre o sensor e o objeto para que o mesmo seja detectado. Estes dispositivos possuem o corpo robusto para suportar forças mecânicas devido ao contato com o objeto.

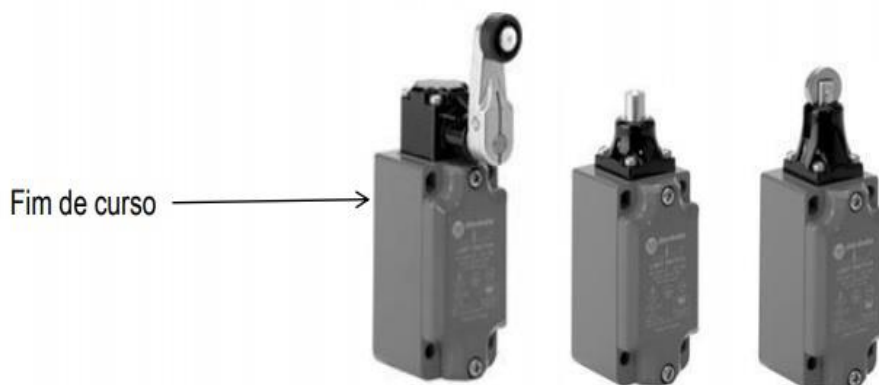


Figura 6 – Sensores fim de curso.

Fonte: FENG. <https://www.politecnica.pucrs.br>. Acesso 02 abr. 2020.

Sua configuração é baseada em:

- Chave de contato elétrico normalmente aberto ou normalmente fechado;
- Após o acionamento, o contato pode ser permanente ou momentâneo;
- Dois ou quatro pares de contatos elétricos;
- Lentidão na abertura e fechamento dos contatos;
- Atuação por pressão.

2.8.2 Sensor de proximidade

Segundo Washington (2012), nestes sensores, a detecção do objeto é feita através da aproximação do mesmo. O quadro 1, descreve o critério para escolha do melhor sensor de proximidade de acordo com a necessidade.

Quadro 1 – Critério de escolha dos sensores de proximidade.

Tecnologia	Indutiva	Capacitiva	Magnética	Ultrassônica	Fotoelétrica
Alcance	<80 mm	<60 mm	<100	<15 m	<200 m
Preço	R\$ 182,00 - 1212,00	R\$ 600,00 – 1212,00	R\$ 121,00 – 727,00	R\$ 1212,00 – 6060,00	R\$ 363,00 – 1818,00
Materiais detectados	Metais	Todos os materiais	Objetos magnetizados (imantados)	Todos os materiais independentemente da forma em que se apresentem (sólidos, líquidos, etc.)	Objetos com poder refletor

Fonte: <https://www.politecnica.pucrs.br>. Acesso 02 abr. 2020.

2.8.2.1 Sensores indutivos

Segundo Washington (2012) os sensores indutivos, representados pela figura 7, detectam a alteração no campo eletromagnético, utilizados para objetos metálicos. Este sensor de proximidade é o mais utilizado devido ao seu preço de mercado. O mesmo possui um raio de alcance relativamente curto de até 80 mm, tendo como características física uma alta resistência e uma forma construtiva robusta.



Figura 7 – Sensor indutivo.

Fonte: FENG. <https://www.politecnica.pucrs.br>. Acesso 02 abr. 2020.

2.8.2.2 Sensores capacitivos

Os sensores capacitivos, representados pela figura 8, têm o princípio de funcionamento bem análogo ao indutivo, esse sensor detecta através da alteração do campo eletrostático, utilizado para objetos não metais, esse sensor tem um raio de alcance curto apenas 60 mm, valor de mercado relativamente maior que o indutivo, possui uma velocidade elevada.



Figura 8 – Sensor capacitivo.

Fonte: FENG. <https://www.citisystems.com.br/sensor-capacitivo/> . Acesso 02 abr. 2020.

2.8.2.3 Sensor fotoelétrico

É conhecido por dispositivo óptico, aquele que detecta variação no espectro infravermelho recebido, ilustrado pela figura 9. Este dispositivo pode ser utilizado em três classificações: através de barreiras, retro refletido ou Reflex e difuso. Apesar de

ser um sensor extremamente sensível a impurezas, sua utilização possui diversos benefícios, como: distância considerada de detecção, cerca de 200 metros e detectam qualquer tipo de objeto, até objetos transparentes (WASHINGTON, 2012).



Figura 9 – Sensor fotoelétrico.

Fonte: FENG. <https://www.politecnica.pucrs.br>. Acesso 02 abr. 2020.

2.8.2.4 Sensor magnético

O sensor de efeito hall, representado pela figura 10, detecta a alteração no campo magnético e tem seu princípio de funcionamento muito análogo ao indutivo, porém esse sensor possui uma lâmina com ímã que se magnetiza e desmagnetiza rapidamente pelo objeto, portanto o objeto detectado precisa possuir um ímã ou ser magnetizado. Apesar do seu tamanho, é um sensor que possui um enorme alcance, e uma das suas principais características é o baixo custo, além de ser insensível a vibrações e a impurezas (WASHINGTON, 2012).



Figura 10 – Sensor magnético.

Fonte: FENG.

https://www.politecnica.pucrs.br/professores/tergolina//Automacao_e_Control/APRESENTACAO_-_Aula_03_Sensores_Industriais.pdf. Acesso 02 abr. 2020.

2.8.2.5 Sensor ultrassônico

O funcionamento do sensor ultrassônico é baseado na emissão e recepção de ondas acústicas de alta frequência. Com a reflexão da onda, é possível detectar a presença e medir a distância do objeto, podendo ser utilizado no modo barreira ou difuso. Este sensor, representado pela figura 11, destina-se a aplicações bastante específicas, como detecção a grandes distâncias em ambientes agressivos (WASHINGTON, 2012).



Figura 11 – Sensor magnético.

Fonte: <https://www.politecnica.pucrs.br>. Acesso 02 abr. 2020.

2.8.2.6 Sensor RFID

De acordo com Washington (2012) o sensor RFID, representado pela figura 12, é o sensor de identificação por radiofrequência, onde possui algumas partes, sendo elas: antena, interface, transceivers (que contém o circuito integrado e a informação transmitida) e tag. A antena transmite a informação, emitindo o sinal do circuito integrado para transmitir suas informações para o leitor, que por sua vez converte as ondas de rádio do RFID para informações digitais. Agora, depois de convertidas, elas poderão ser lidas e compreendidas por um computador para então ter seus dados analisados.



Figura 12 – Sensor RFID.

Fonte: https://www.politecnica.pucrs.br/professores/tergolina//Automacao_e_Control/APRESENTACAO_-_Aula_03_Sensores_Industriais.pdf. Acesso 02 abr. 2020.

2.8.2.7 Sensor de medição de corrente

Segundo Vishay Dale (2016) sensores de corrente contínua (Figura 24) servem para verificar a intensidade da corrente elétrica. Este equipamento segue o princípio da Lei de Ohm, gerando uma tensão elétrica proporcional à corrente que passa através dele. Através da figura 13 pode-se observar um exemplo deste sensor.

São aplicados em situações nas quais a exatidão e a taxa de repetição em várias situações são muito importantes, exceto quando é necessário a utilização de valores muito elevados de correntes. Alguns exemplos dessas situações são: sensores de força, fontes de alimentação comutadas, projetos que envolvem o uso de motores de corrente contínua, medidores de consumos residenciais e todos os processos nos quais a corrente deve ser lida.



Figura 13 – Sensor de corrente contínua.

Fonte: <https://www.filipeflop.com/blog/medidor-de-corrente-sct013-com-arduino/>. Acesso 02 abr. 2020.

2.8.2.8 Sensor de medição de tensão

Os sensores de tensão AC (corrente contínua), detectam e indicam tanto a passagem quanto a presença de tensão AC em um circuito. Através da figura 14, pode-se observar um exemplo deste sensor.



Figura 14 – Sensor de tensão AC.

Fonte: <https://www.eletronicacastro.com.br/categoria-nao-definida/14050-p8-modulo-sensor-de-tensao-ac-0000000140508.html>. Acesso 02 abr. 2020.

2.9 Atuadores

Segundo Cyrino (2017), são equipamentos que realizam a conversão de energia elétrica, hidráulica ou pneumática em energia mecânica. A energia mecânica gerada pelos atuadores é enviada aos vários tipos de processos para que haja movimento.

Os atuadores são fundamentais no funcionamento de máquinas e equipamentos e existem diversas maneiras de converter seus movimentos em energia mecânica e, por consequência, realizar um determinado trabalho. Os atuadores são classificados em hidráulicos, elétricos ou pneumáticos, e também existem os contadores.

2.9.1 Atuadores hidráulicos

Os atuadores hidráulicos são acionados por fluidos em movimento, como o óleo, que geralmente é utilizado por ser facilmente pressurizado para proporcionar tal movimento. Um exemplo físico desse atuador é representado pela figura 15.

Os sistemas hidráulicos são ligados a válvulas direcionais (pré-atuadores), que definem a direção do deslocamento do fluido nos atuadores, seguindo sinais emitidos por algum controle. Requerem um controle contínuo e preciso de

posicionamento e velocidade. A relação entre a potência mecânica do atuador e seu peso é elevada, assim é possível obter unidades compactas com altas potências (CYRINO, 2017).



Figura 15 – Sensor de tensão AC.

Fonte: <https://www.manutencaoemfoco.com.br/atadores-de-maquinas-e-equipamentos/>. Acesso 03 abr. 2020.

2.9.2 Atuadores elétricos

Os atuadores elétricos produzem movimentos majoritariamente rotativos e atendem a comandos de controle de forma manual ou programada, no próprio local ou de maneira remota.

Esses atuadores são responsáveis diretamente pelo controle do movimento de rotação e deslocamentos. motores de passo, motores elétricos de corrente contínua, motores de corrente alternada, ou etc. são exemplos de atuadores elétricos, representados pela figura 16. (CYRINO, 2017).



Figura 16 – Atuadores elétricos.

Fonte: <https://www.manutencaoemfoco.com.br/atuadores-de-maquinas-e-equipamentos/>. Acesso 03 abr. 2020

2.9.3 Atuadores pneumáticos

Os atuadores pneumáticos, representados pela figura 17, usam como fluido o ar comprimido e tem um princípio de funcionamento muito semelhante aos atuadores hidráulicos, mas, em contrapartida, tem uma capacidade inferior ao conter a pressão do fluido.

Também são conectados a válvulas direcionais (pré-atuadores) que definem a direção do deslocamento do fluido nos atuadores, seguindo sinais emitidos por algum controle.

O ar comprimido utilizado é produzido por um compressor para fornecer movimento ao atuador pneumático através das válvulas direcionais. (CYRINO, 2017).



Figura 17 – Atuadores pneumáticos.

Fonte: <https://www.manutencaoemfoco.com.br/atuadores-de-maquinas-e-equipamentos/>. Acesso 03 abr. 2020.

2.10 Contatores

Segundo Pirelli (2012), define-se por contator um dispositivo eletromecânico que realiza o acionamento de motores trifásicos, resistências industriais e quaisquer outras cargas que exijam correntes elétricas muito maiores e que apresentem potências maiores. Pode-se observar um exemplo de contator através da figura 18.

Dentro destes dispositivos, estão presentes uma bobina, um núcleo e um conjunto de contatos de força e comando. O eletromagnetismo é responsável pelo

acionamento desses contatos, que ligam ou desligam os equipamentos a eles conectados.

Em sistemas trifásicos, nos quais os equipamentos requerem três linhas de fase para que funcionem, podemos encontrar com mais frequência a atuação de contadores de potência, que operam de modo muito semelhante a um interruptor, o que garante maior vida útil às máquinas acionadas por eles, além de apresentarem uma maior segurança e facilidade de manuseio.



Figura 18 – Contadores.

Fonte: <https://www.dceeletrica.com.br/contadores>. Acesso 03 abr. 2020.

Os contadores podem ter três tipos de contatos:

- Contatos normalmente abertos (NA): Ficam abertos e, quando acionados, se fecham, permitindo a passagem de corrente elétrica.
- Contatos normalmente fechados (NF): Ficam fechados e, quando acionados, se abrem, interrompendo a passagem de corrente elétrica.
- Contatos comutadores: Possuem ambas as funções acima no mesmo contato (uma parte NA e outra NF). São utilizados para comutar entre diferentes partes de um circuito.

2.11 Supervisório

Segundo Andrade (2018), em meados das décadas de 70 e 80, com o avanço tecnológico da eletrônica, surgiram os primeiros microprocessadores e os computadores tornaram-se peça fundamental no meio industrial. Surgiram então os primeiros supervisórios. Quando visitamos indústrias, vemos ao lado de máquinas robustas um visor no qual estão sendo demonstrados gráficos e tabelas referentes ao serviço realizado. Basicamente, um sistema supervisório é definido por capturar e armazenar em um banco de dados informações sobre o processo de produção, que são captadas por sensores que capturam dados específicos (conhecidos com variáveis do processo).

A principal utilidade do Supervisório é facilitar a interação do usuário com o chão de fábrica (máquina, equipamento, linha de produção, processo de fabricação, etc). A facilidade fica clara não só ao usuário que está realizando a interação, mas também ao desenvolvedor da aplicação. Esse tipo de sistema busca otimizar a administração dos dados industriais, facilitando a interação entre seu usuário e todo o chão de fábrica. Ele engloba diferentes instâncias, podendo ir desde a interface até o desenvolvimento de aplicativos específicos.

O supervisório utiliza alguns recursos tais como tags, telas sinóticas, alarmes e eventos, históricos e relatórios.

Existem diversas formas de se montar um supervisório para a coleta dos dados, sendo elas, monitoramento por wireless, monitoramento por IHM e sistema de controle com CLP.

2.11.1 Monitoramento por wireless

Segundo Martins (2012) nesse tipo de montagem, toda a comunicação entre os instrumentos é feita utilizando o protocolo wireless HART, e seu gerenciamento é feito através do gateway wireless HART. O gateway tem como foco principal gerenciar a rede, coletar informações e fazer a integração os dados no sistema de supervisão.

2.11.2 Monitoramento por IHM

Segundo Martins (2012), a IHM faz a interface entre os operadores com o processo industrial, representado pela figura 19. Nesse tipo de monitoramento IHM

(Interface Homem Máquina), a supervisão é realizada no campo e traz diversas funções dentro dela, que podem suprir várias aplicações.

Seu funcionamento é definido da seguinte forma: inicialmente, é necessário realizar a leitura do gateway wireless HART. Nesse ponto, o gateway está fornecendo dado utilizando Modbus TCP/IP, com mapa configurado.

A IHM necessita de um driver de comunicação para ler este padrão de comunicação, e assim integrar os dados na interface gráfica.

Além disso, a IHM irá armazenar as informações no seu banco de dados. Após isso, é possível montar uma interface gráfica conectando com os valores enviados pelos instrumentos de medição Wireless HART no campo.

Nesta aplicação, não temos comando de escrita. Significa que não acionaremos nada utilizando o sistema supervisorio, apenas serão lidas e armazenadas as informações do campo.



Figura 19 – Interface Homem-Maquina

Fonte: <https://aucavili.com.br/blog/?p=214>. Acesso 05 abr. 2020.

A IHM pode ser utilizada de três formas diferentes em uma aplicação industrial:

- Substituição de painel por botões: Você pode substituir o painel por botões, assim a interpretação dos dados será muito mais simples e você ganha espaço no painel.

- Monitoramento dos dados de processo: Nessa aplicação, a IHM será um supervisor para criação de gráficos do processo, além de alarmes e acionamentos.
- Visualização: Essa aplicação é basicamente os dois modos citados acima, funcionando juntamente com um sistema supervisor SCADA.

2.11.3 Sistema de controle com CLP

Segundo Cassiolato (2011), o sistema de controle apresentado na figura 20, possui variáveis para leitura e acionamento. Todos os equipamentos realizam a comunicação HART com cartão HART do CLP. Neste contexto, desejamos um sistema supervisor tipo SCADA (*Supervisory Control and Data Acquisition*) para supervisão dessa malha de controle.

Como os instrumentos estão se comunicando por HART, isto quer dizer que o que está sendo lido são as 4 variáveis de processos: PV (Variável Primária), SV (Variável Secundária), TV (Variável Terciária), QV (Variável Quartenária).

A comunicação entre o CLP e o sistema supervisor SCADA será feita utilizando a comunicação OPC que é baseada no protocolo ethernet. Os sistemas SCADA habitualmente podem ter inúmeros drivers de comunicação para integração com diferentes fabricantes.



Figura 20 – Sistema de controle com CLP

Fonte: <https://www.solucoesindustriais.com.br>. Acesso 03 abr. 2020.

O SCADA é composto por um servidor, onde serão salvas informações do sistema nos bancos de dados que podem ser em redundância. Existe também o

estúdio para desenvolvimento das telas e a interface de visualização para realização da operação, conforme identificado na figura 21.

Atualmente, no mercado, temos diversos sistemas supervisórios utilizados, como:

- Elipse SCADA da Elipse Software;
- Laquis;
- FactoryTalk View SE da Rockwell Automation;
- iFIX da General Electric;
- InduSoft Web Studio da InduSoft;
- ProcessView da SMAR;
- SCADA BR (open source) da MCA Sistemas;
- SIMATIC Wincc da Siemens;
- VijeoCitect da Schneider Electric;
- WondewareinTouch da Invensys.



Figura 21 – Supervisório Laquis

Fonte: SCADA Laquis. Disponível em: <https://www.lcds.com.br/laquis.asp>. Acesso em: 28 fev.2020.

2.11.3.1 Elipse da Elipse Software

Segundo Elipse (2020), é uma plataforma HMI/SCADA utilizada em aplicações avançadas e distribuídas, muita utilizada em sistemas críticos e centro de controle. O Elipse é considerado líder de mercado, é uma ferramenta SCADA para

monitoramento e controle de processos, oferece escalabilidade e constante evolução para inúmeras aplicações, desde o simples até o mais complexo processo de interface HMI em tempo real. A figura 22 demonstra um exemplo desta plataforma.

Foi desenvolvido para atender os mais exigentes requisitos de conectividade, flexibilidade e confiança.

Podemos citar vários benefícios em utilizar o Elipse como: Conecta com a maioria dos PLCs do mercado, resposta rápida, investimento duradouro entre outros.

As principais características são: Multiusuários e multiprojetos, editor de tela completo, fácil gerenciamento da aplicação entre outras.

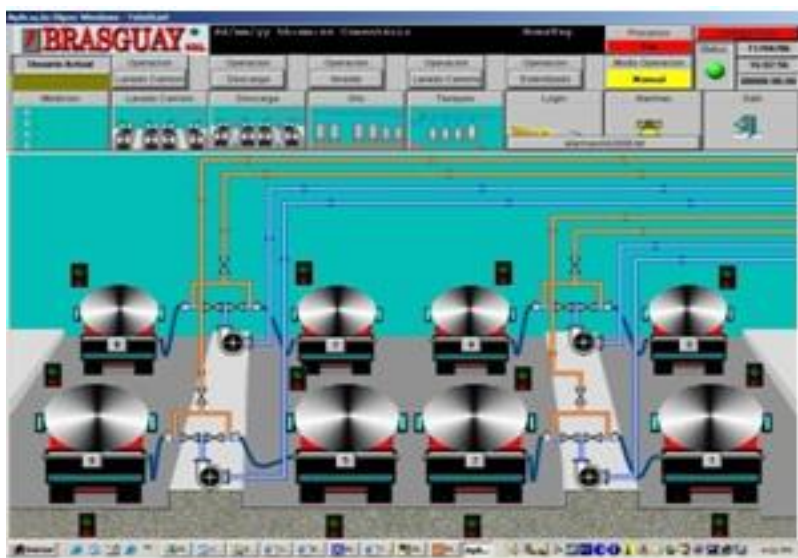


Figura 22 – Elipse da Elipse Software

Fonte: www.elipse.com.br/produto/elipse-e3/. Acesso em: 29 abr. 2020.

2.11.3.2 Laquis

O Laquis é um módulo responsável pela comunicação com um PLC, onde é possível definir as variáveis que serão utilizadas nos processos que serão controlados. O protocolo de comunicação entre o Laquis e o PLC varia de acordo com cada equipamento, um desses protocolos é o ModBus que pode ser por via serial ou Ethernet.

Os dados podem ser armazenados em um banco de dados formando um histórico, a partir dos dados arquivados podemos gerar relatórios que podem ser formatados de acordo com cada necessidade, desde simples listagens das tags até

cálculos sobre valores montados de acordo com a norma, podemos também customizar os relatórios com informações e cálculos, um tipo de informação pode ser monitorada, calculada ou armazenada no sistema SCADA é o alarme, o alarme é uma tag que varia de acordo com a condição disparada.

As telas do sistema Laquis podem ser replicadas na rede, podem ser monitoradas por diversos usuários através da web, outros computadores podem acessar este servidor através da rede. Em aplicações mais avançadas ou específicas são possíveis programar ações sobre dados através de scripts.

Indicar tags na planilha (entradas e saídas) é uma das aplicações básicas do Laquis, cada tag ou variável podem conter quantas propriedades forem necessárias. Nas aplicações mais avançadas o Laquis proporciona a possibilidade de programação através de linguagem visual estruturada para supervisão e automação industrial.

O Laquis permite que driver seja desenvolvido dentro do próprio sistema, que possui uma linguagem totalmente voltada para esse tipo de operação.

2.11.3.3 iFIX da General Electric

Segundo a iFIX (2020), um sistema muito consagrado (Figura 34) para supervisão e controle de qualquer processo industrial. Muito flexível e aplicável a projetos de grande e pequeno porte. Oferece robustez e confiabilidade com uma interface totalmente completa que permite a operação e monitoração industrial. A figura 23 apresenta um exemplo deste sistema.

Com recursos amigáveis e intuitivos permite a construção de telas que representam processos, essas telas permitem visualizar melhor dados e comandos. O sistema de controle apresenta inúmeras ferramentas para análise básica com gráfico e lista de alarme, ao qual facilita a interpretação de dados.

Permite o armazenamento de dados em um pequeno histórico suficiente para atender as necessidades.



Figura 23 – Supervisório iFix

Fonte: <https://www.aquarius.com.br/produto/ge-ifix/>. Acesso em: 29 abr. 2020.

2.11.3.4 InduSoft Web Studio

Segundo a InduSoft (2020), uma poderosa ferramenta voltada para a automação industrial, que possibilita o desenvolvimento HMI, SCADA. Esse sistema tem grandes vantagens, como:

- Economia de tempo: Através da intranet é possível acessar toda operação até de um telefone celular.
- Linguagem Fácil: Utiliza linguagem Unicode.
- Flexibilidade: Utiliza mais de 250 drivers disponíveis.
- Está no mercado desde 1997, ano que sua venda foi altíssima e até hoje é muito utilizado, sendo vendido mundialmente.

A figura 24 apresenta um exemplo de supervisório deste sistema.

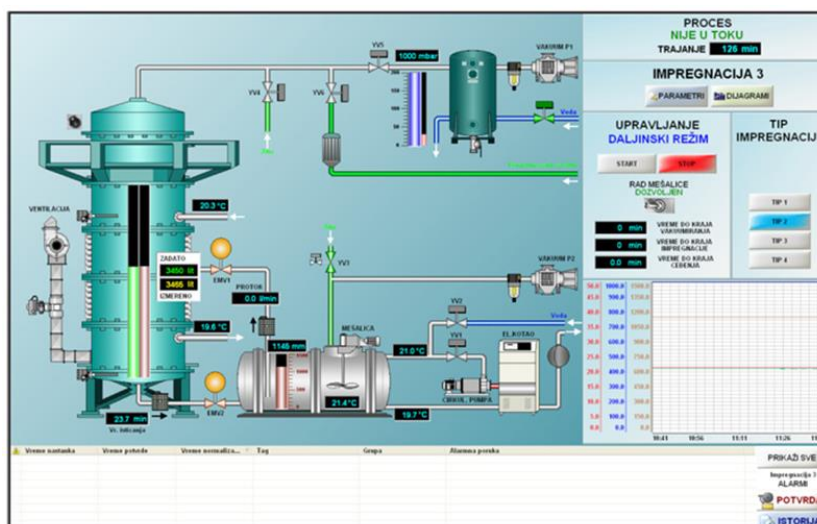


Figura 24 – Supervisório Indusoft

Fonte: <https://www.indusoft.com/br/Recursos/Conhecendo-o-InduSoft-Web-Studio>. Acesso em: 29 abr. 2020.

2.11.3.5 SCADA BR

Segundo Cassiolato (2011), o SCADA BR é um software voltado para automação totalmente gratuito e de código fonte aberto, este software serve para automatizar processos industriais como medição e automação, através dele pode-se acessar e controlar diversos dispositivos. Também é possível salvar os dados em uma base de dados. A figura 25 exemplifica um modelo de tela do SCADA BR.

SCADA BR deve ser instalado em um computador, que será conhecido como o "servidor da aplicação" ou "Servidor SCADA BR". Tanto as telas "HMI" quanto as configurações do SCADA BR serão acessadas através de um navegador de Internet.

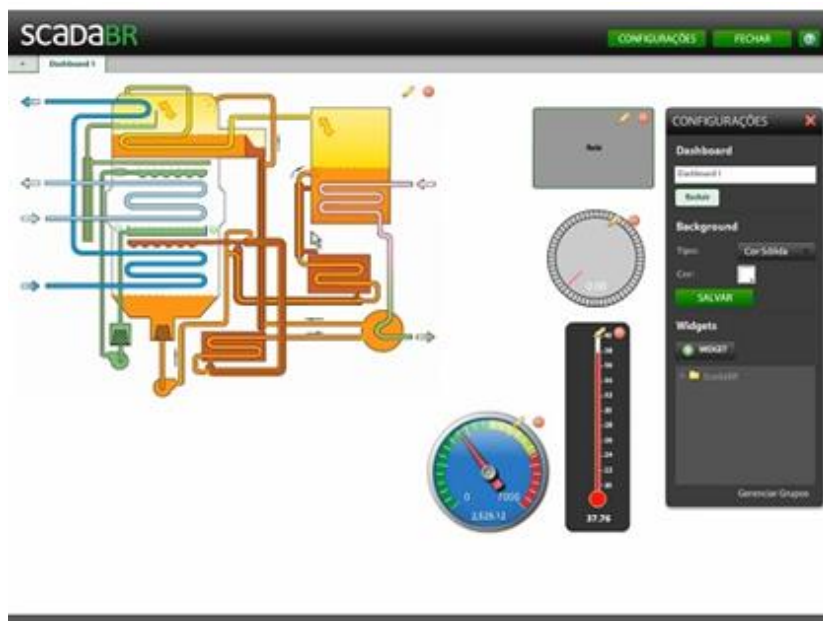


Figura 25 – Supervisório SCADA BR

Fonte: https://sites.google.com/a/certi.org.br/certi_scadabr/home/minicursos/iniciando-scadabr.
Acesso em: 29 abr. 2020

3 DESENVOLVIMENTO APLICADO AO LABORATÓRIO

O objetivo principal deste trabalho é realizar a instalação e aplicação prática de um sistema supervisório no laboratório de automação do Centro Universitário de Volta Redonda – Fundação Oswaldo Aranha (UniFOA), para que seja possível um maior e melhor aproveitamento do aprendizado, principalmente no que se diz respeito a programações nos CLPs em linguagem LADDER. O CLP utilizado no laboratório em todos os computadores é o TPW3-PCLINK, do fabricante WEG.

Após algumas pesquisas, foi decidido que o melhor Sistema Supervisório para ser utilizado no laboratório seria o Eclipse SCADA, devido a sua fácil implementação e baixo custo, podendo ser rodado na versão demo com até 20 tags. Para que a implementação seja possível, é necessário instalar o supervisório em um computador para que o sistema efetue a supervisão dos experimentos, assim é possível controlar a programação feita no CLP.

O meio utilizado para a comunicação física do supervisório com o CLP será através do cabo de programação ou porta de comunicação RS-485, uma vez que os CLPs utilizados no laboratório já possuem essa porta de comunicação.

Em seguida, será apresentada uma lista com os materiais utilizados no processo de implementação.

3.1 Lista de Materiais

- Computadores com desktop Windows 7 ou superior presentes no laboratório;
- Controlador Lógico Programável modular TPW3-PCLINK;
- Software Supervisório eclipse SCADA;
- Cabo RS-485 para comunicação;
- Cabo de comunicação RS232;
- Conversor RS232 para RS485.

3.2 Normas aplicadas a um laboratório (educacional)

A seguir serão apresentadas algumas normas educacionais gerais (podem ser aplicadas em vários campus universitários e escolas) para uso e segurança de um laboratório de Automação Industrial. É de suma importância que ocorra o cumprimento total de cada uma das regras listadas abaixo, para que as atividades dentro do laboratório sejam lícitas e que não apresentem nenhum risco à saúde física e/ou psicológica dos alunos, professores e funcionários.

3.2.1 Normas aplicadas a laboratórios

Segundo UFRS (2017), as Normas Regulamentadoras (NR) “tratam-se do conjunto de requisitos e procedimentos relativos à segurança e medicina do trabalho, de observância obrigatória às empresas privadas, públicas e órgãos do governo que possuam empregados regidos pela Consolidação das Leis do Trabalho (CLT).” Estas normas também se aplicam em ambientes laboratoriais universitários e em escolas. As normas aplicáveis a ambientes e equipamentos de laboratório são:

- **NR-06:** Equipamentos de Proteção Individual (EPI). EPI é todo dispositivo de uso individual destinado a proteger a saúde e a integridade. A empresa é obrigada a fornecer gratuitamente EPI adequado ao risco e em perfeito estado. Também é responsabilidade da empresa garantir que os EPIs estejam sendo utilizados.
- **NR-10:** Segurança em Instalações e Serviços em Eletricidade. Fixa as condições mínimas para garantir a segurança de quem trabalha em instalações elétricas e a segurança de usuários e terceiros. Trata-se de normas para instalações seguras e da proteção de partes energizadas. É aplicável não só a equipamento, mas também ao espaço físico e instalações elétricas.
- **NR-12:** Segurança no Trabalho em Máquinas e Equipamentos. Define referências técnicas, princípios fundamentais e medidas de proteção para garantir a saúde e a integridade física. Trata de acesso a partes girantes, quentes ou energizadas. Especifica tensão dos painéis de comando operado e proteção das partes móveis. O

não cumprimento pode acarretar em aplicação das penalidades previstas na legislação pertinente.

- NR-17: Ergonomia. Visa a estabelecer parâmetros que permitam a adaptação das condições de trabalho às características psicofisiológicas dos trabalhadores, de modo a proporcionar um máximo de conforto, segurança e desempenho eficiente. Trata de levantamento, transporte e descarga individual de materiais, mobiliário dos postos de trabalho, equipamentos dos postos de trabalho, condições ambientais de trabalho e organização do trabalho.
- ABNT NBR 5410: Instalações elétricas de baixa tensão. Estabelece as condições para instalações elétricas de baixa tensão, a fim de garantir a segurança de pessoas e animais, o funcionamento adequado da instalação e a conservação dos bens. Aplica-se aos circuitos elétricos, a toda fiação e as linhas elétricas fixas de sinal alimentados sob tensão nominal igual ou inferior a 1000V em corrente alternada, com frequências inferiores a 400 Hz, ou a 1500V em corrente contínua.
- ABNT NBR 13967: Estações de Trabalho. Especifica as características físicas e dimensionais e classifica os sistemas de estação de trabalho. Define parâmetros geométricos, limites de capacidade e proteção. O atendimento a essa norma deve ser comprovado pela apresentação de um laudo.

3.2.2 Adequação dos laboratórios

O laboratório é um espaço educativo de suma importância na formação profissional: neles encontramos máquinas, equipamentos, bancadas energizadas etc. É um ambiente bem fiel ao ambiente industrial, o que já traz uma vivência desta área para os alunos. Entretanto, possuem riscos, principalmente devido à falta de experiência do aluno.

Dentro do ambiente industrial, existem normas e procedimentos de segurança que devem ser seguidas rigorosamente, e não é diferente nos laboratórios. São elas:

- Os laboratórios devem ser adequados conforme as prerrogativas da NR-10;

- Máquinas e equipamentos devem ser adequados conforme a NR-12, o que para os equipamentos mais antigos pode implicar na inclusão de dispositivos de segurança e proteção;
- A verificação da conformidade de ambientes e equipamentos deve ser feita por um profissional de segurança do trabalho devidamente credenciado.

3.3 Implementação

Para início da implementação do sistema de controle e supervisão, será necessário a instalação de um software supervisorio (SCADA).

Um sistema supervisorio se trata de um software capaz de supervisionar, controlar um processo já programado por um CLP. Ao abrirmos o programa, teremos a opção de construir uma tela para controle de determinado processo. Através figura 26, pode-se verificar um exemplo de uma tela para controlar os níveis de fluido dentro de três tanques.

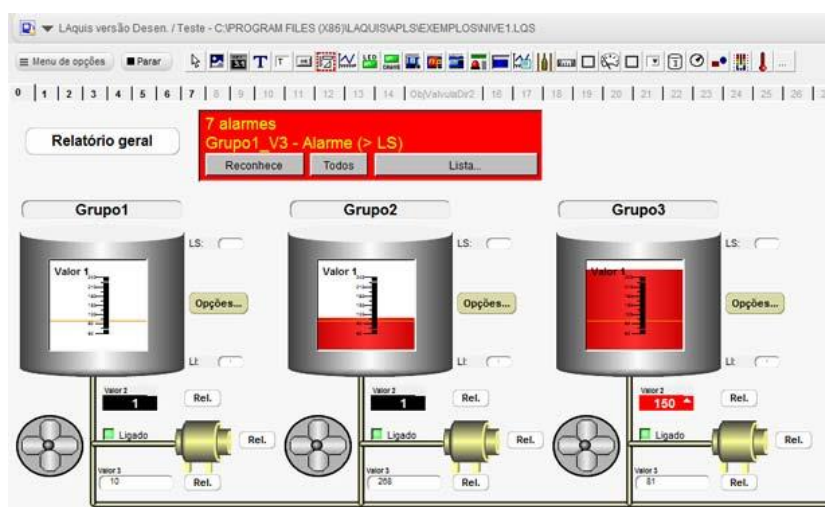


Figura 26 – Exemplo de Tela de Supervisorio.

Fonte: <https://www.saladaeletrica.com.br/supervisorio-laquis/>. Acesso em: 31 mai. 2020.

No ambiente do laboratório, o supervisorio estará em comunicação direta com o CLP, uma vez que o funcionamento da tela depende da programação deste software. Na tela, são atribuídas tags. As tags são palavras que servem justamente como uma etiqueta e ajudam na hora de organizar informações, agrupando aquelas que receberam a mesma marcação, facilitando encontrar outras relacionadas. No software, não é diferente. Cada tag vem com uma “regra” para que cada elemento

da tela cumpra. Podemos usar como exemplo: o botão para ligar possui a tag “TAG1LIGA”, o botão para desligar possui a tag “TAG2DESLIGA”, e assim por diante. Note que qualquer outro elemento da tela com a tag responsável por ligar também terá essa função, por isso cada componente de uma tela deve ter sua devida função, para que seja possível o controle dos processos.

Para instalar o sistema supervisório no laboratório de Automação Industrial, é necessário que o software seja baixado e instalado nas máquinas que serão utilizadas.

Para que seja possível realizar a comunicação do sistema supervisório com os CLPs, é preciso instalar os drivers de comunicação, que são baixados juntamente com o sistema supervisório. O protocolo de comunicação escolhido foi o modbus, devido a sua facilidade de implementação.

Modbus serial é um protocolo de comunicação de dados para comandar atuadores, e pode usar os cabos RS-232 ou RS-485 e Ethernet para que a comunicação seja possível. O modbus utiliza o método de comunicação mestre escravo, que define como será a troca de dados entre o CLP e Computador.

Neste método, o mestre envia uma requisição de informação, enquanto o escravo produz a informação de resposta e reenvia para o mestre. Porém, o mestre também pode enviar informações para revisar os dados mandados pelo escravo, diretamente. A estrutura da mensagem deste protocolo exige que cada controlador tenha seu endereço para ser conectado à rede, o que facilita a comunicação de qual escravo está se comunicando com qual mestre.

A interface gráfica do supervisório é gerada pela IHM contida no mesmo, baseada na programação do CLP como plano de fundo. Os elementos gráficos estão associados a endereços das portas do CLP, e assim é possível dizer se recebem sinais analógicos ou digitais, o que possibilita a criação de tags, dando informações em tempo real sobre o processo. Teoricamente, os sinais, no caso, são digitais, assim os endereços seguem o mesmo processo de criação.

A seguir, será mostrado todo o caminho, desde a instalação do software até a sua comunicação com o CLP.

Após ser baixado o software SCADA do site da Elipse (www.elipse.com.br), será necessário executar o arquivo de instalação. Após a instalação, abrir o arquivo. Porém, quando o arquivo abrir irá aparecer a tela representada pela figura 27.

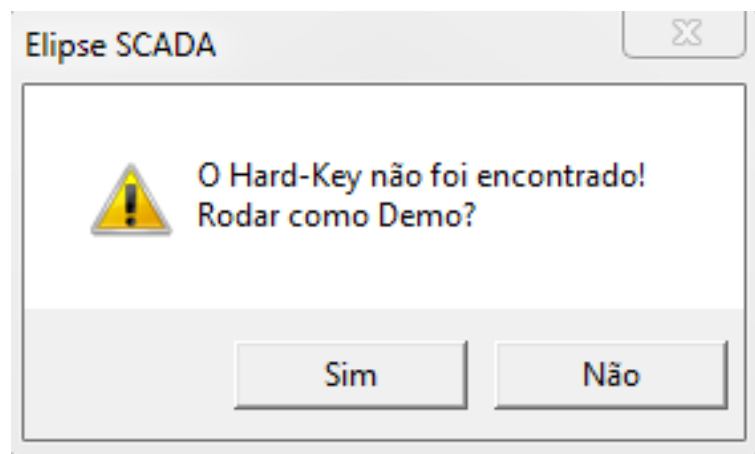


Figura 27 – Tela que aparece ao abrir o arquivo Elipse SCADA.
Fonte: os autores, 2020.

Como todo software supervisor, faz-se necessário ter uma licença para rodar em sua versão completa. Com o que será demonstrado e aplicado no laboratório, não será necessário rodar o programa em sua versão completa. A liberação do programa em sua versão completa será através de um *hard key* ou um *soft key*, no caso do Elipse SCADA. Como iremos rodar na versão demo, o aplicativo irá reconhecer até 20 tags somente, o que não será nenhum ponto agravante para o laboratório, uma vez que os experimentos não irão ultrapassar esse limite. A figura 28 representa a tela inicial do aplicativo.

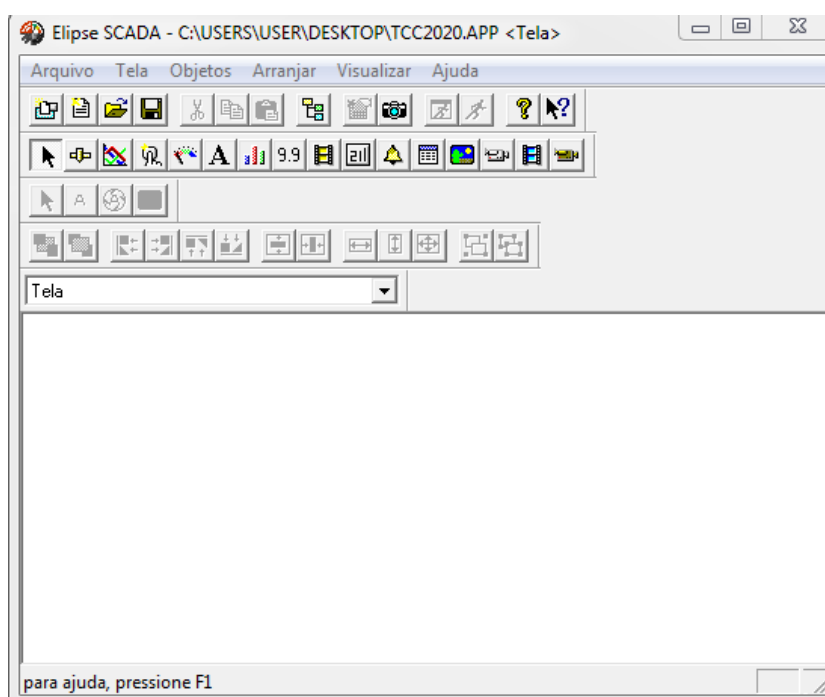


Figura 28 – Tela inicial do Elipse SCADA.
Fonte: os autores, 2020.

Através da tela organizer demonstrada pela figura 29, é possível efetuar a principais configurações para comunicação entre o CLP e o PC que estará rodando o supervisório. Para criar essas configurações, será necessário executar os seguintes passos:

1. Carregar o driver de comunicação;
2. Configurar o driver de comunicação Modbus;
3. Tipo de comunicação;
4. Configuração da porta de comunicação;
5. Criação das tags.

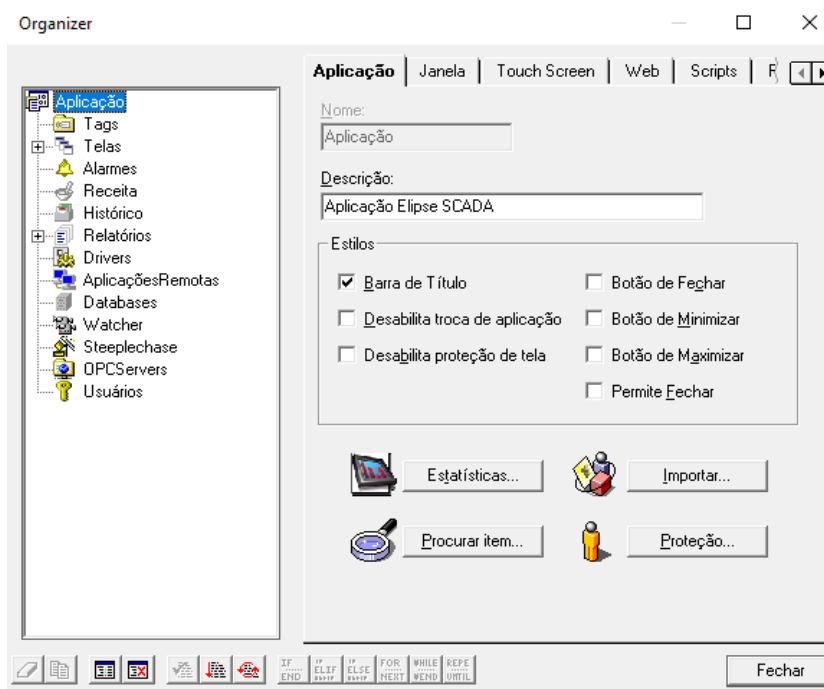


Figura 29 – Tela da função organizer no Elipse SCADA
Fonte: os autores, 2020.

3.3.1 Carregar o driver de comunicação

Como foi determinado que a comunicação seria a Modbus RTU devido às suas facilidades de configuração, será necessário a escolha do driver de comunicação, conforme figura 30.

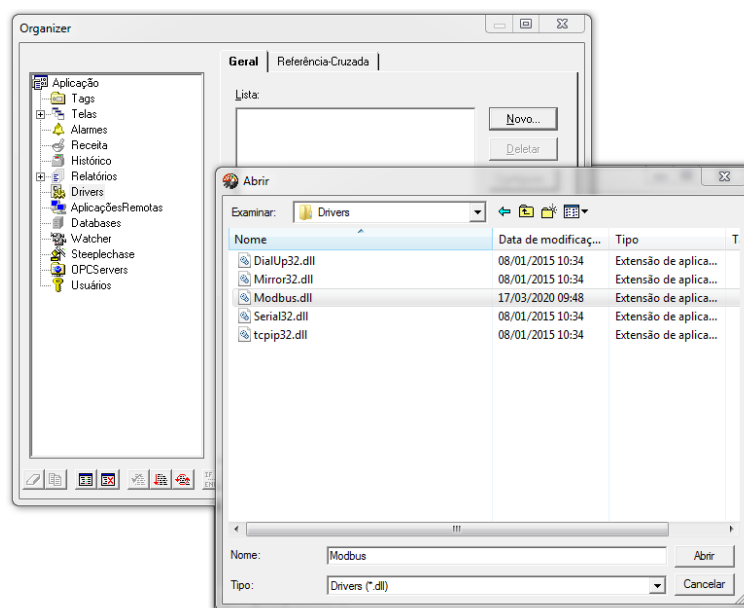


Figura 30 – Tela para escolha do driver de comunicação no Elipse SCADA.
Fonte: os autores, 2020.

Nessa tela, deverá ser escolhido um novo driver, e esse driver deverá estar instalado e disponível na unidade C do PC. Então, conforme o protocolo de comunicação escolhido foi o modbus, o arquivo de comunicação deverá ser o “Modbus.dll”. Escolhido o arquivo, o driver aparecerá conforme a tela apresentada pela figura 31.

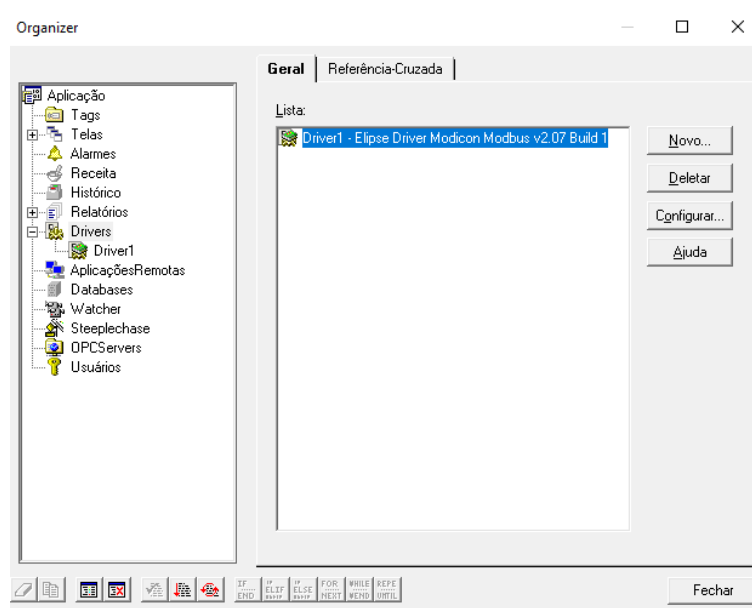


Figura 31 – Tela que aparece o driver no Elipse SCADA.
Fonte: os autores, 2020.

Esse driver deverá ser baixado no site do Elipse SCADA, sendo necessário escolher qual o fabricante do CLP utilizado. Esse driver poderá mudar de versão.

3.3.2 Configurar driver de comunicação ModBus

Essa etapa é de suma importância, pois será nessa configuração que serão configuradas as funções de leitura e escrita entre o CLP e o sistema supervisor. Essa configuração só poderá ser feita após a seleção do driver modbus, e se dará através da tela mostrada na figura acima. Selecionar o campo configurar, mostrado acima, irá aparecer a tela indicada na figura 32.

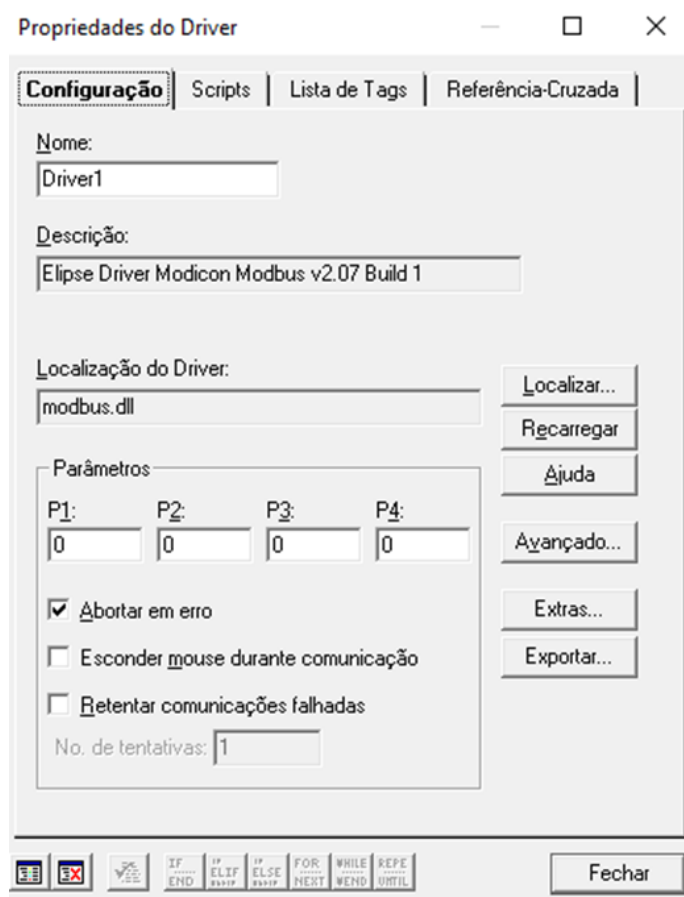


Figura 32 – Tela para configuração de driver no Elipse SCADA.

Fonte: os autores, 2020.

Essa configuração é importante, pois será ela que dará o início da criação do que será lido e escrito no CLP. Para se iniciar a configuração, deve-se estar na tela indicada na figura acima e será necessário selecionar a opção extras. Ao selecionar o campo extras, aparecerá a tela apresentada pela figura 33.

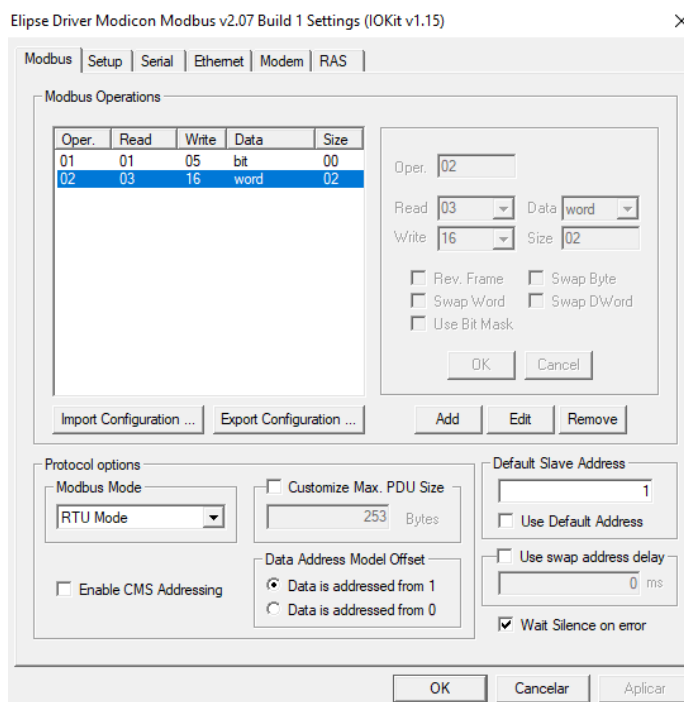


Figura 33 – Aba Modbus de Comunicação no Elipse SCADA.
Fonte: os autores, 2020.

O preenchimento dos campos será em função do que se espera que o supervisor execute. Essa configuração será através da função que o supervisor irá executar sobre o CLP, e essa escolha será através da função executada. Essa função está descrita no manual do driver de comunicação (anexo 1) e será através do tipo de dado e variáveis que se deseja ler ou escrever. Segue um exemplo demonstrado nos quadros 2 e 3.

Quadro 2 – Descrições dos valores de leitura no Elipse SCADA.

LEITURA	
Valor	Descrição
1	Leitura de Bit (Read Coil Status - 0x)
2	Leitura de Bit (Read Input Status - 1x)
3	Leitura de Words (Read Holding Registers - 4x)
4	Leitura de Words (Read Input Registers - 3x)
7	Leitura de Status (Read Exception Status)
20	Leitura de Memória Estendida (Read General Reference - 6x)

Fonte: os autores, 2020.

Quadro 3 – Descrições dos valores de leitura no Elipse SCADA.

ESCRITA	
Valor	Descrição
5	Escrita de Bit (Force Single Coil - 0x)
6	Escrita de Word Simples (Preset Single Register - 4x)
15	Escrita de Bits (Force MultipleCoils - 0x)
16	Escrita de Words (PresetMultipleRegister - 4x)
21	Escrita de Memória Estendida (Write General Reference - 6x)

Fonte: os autores, 2020.

Note que foram escolhidas as funções 1 e 2, que, para as aplicações no laboratório serão as mais utilizadas, devido aos experimentos serem informações binárias e analógicas.

A função 1 será utilizada para se ler uma informação digital, ou seja, bit a bit, sendo que se poderá ler uma informação de entrada com nível lógico 1 ou 0 e escrever uma informação digital 1 ou 0.

A função 2 será utilizada para se ler uma informação analógica, ou seja, um valor que estará alterando no tempo, como, por exemplo, poderá ser usado para valores atuais de temporizadores e contadores.

Outro ponto que se deve levar em consideração é o tipo de protocolo utilizado e qual o endereço do escravo. Como a comunicação será entre 2 pontos CLP e PC (onde estará rodando o supervisório) deverá se utilizar o protocolo Modbus RTU, pois esse tipo de protocolo só roda em função de 2 pontos somente. O endereço utilizado no processo é definido como 1, pois sendo que será considerado o PC como mestre e o CLP como escravo (slave), deverá ser informado que o CLP será o escravo e, com isso, terá o endereço 1. Desta forma, haverá uma troca de informação entre CLP e PC.

3.3.3 Tipo de comunicação

O próximo passo será referente a escolha do tipo de transmissão a ser feita. Isso será feito na próxima aba setup. Como será utilizada a comunicação RS232 ou RS485, será uma comunicação serial, conforme a figura 34.

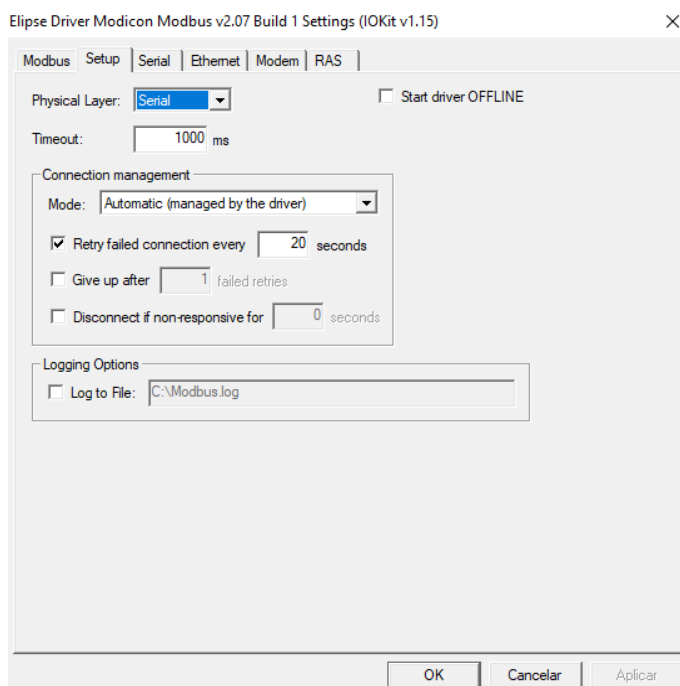


Figura 34 – Aba setup de comunicação no Elipse SCADA.
Fonte: os autores, 2020.

3.3.4 Configuração da porta de comunicação

Nessa etapa, será determinada a configuração da comunicação da porta serial. Como, por exemplo, qual a porta a ser utilizada, velocidade, tamanho da palavra, paridade e stop bits. Observe na figura 35, as seguintes configurações.

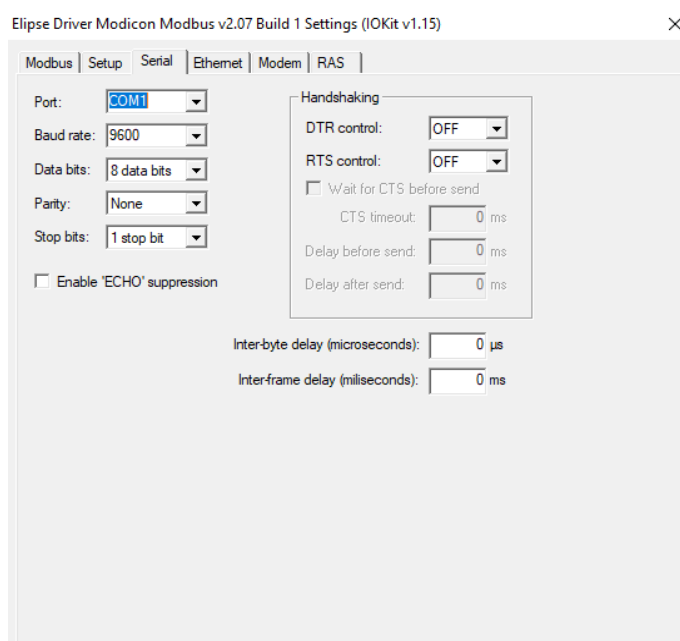


Figura 35 – Aba serial de comunicação no Elipse SCADA.
Fonte: os autores, 2020.

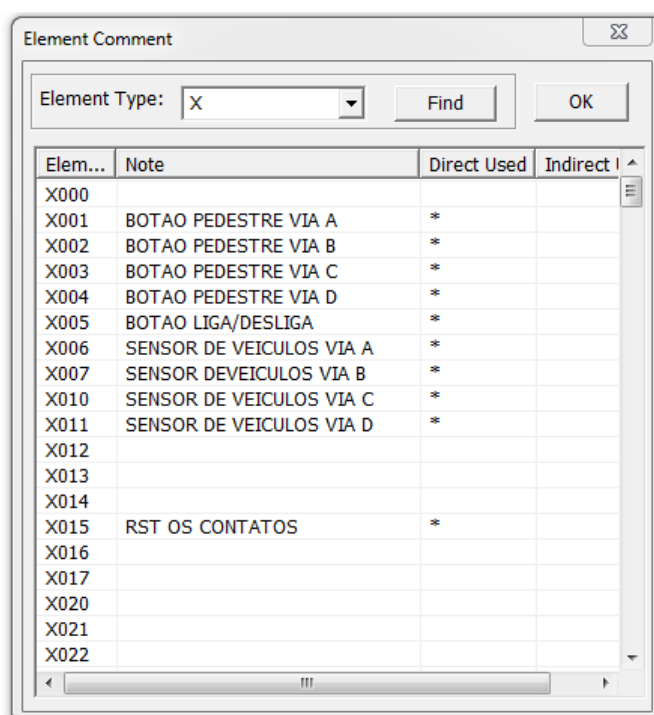
Essa configuração deverá estar de acordo com a configuração do CLP, ou seja, deverá estar igual para que a comunicação aconteça. Após essa configuração, retornar à tela inicial do *organizer*.

3.3.5 Criação de TAGS

Essa etapa é para configurar o local e o tipo de dados que o supervisor irá buscar no CLP e receber do CLP. Esses dados são chamados de tags.

Para todos os parâmetros ou pontos que se deseja verificar no supervisor, é necessário que se crie uma tag. É através dessa tela que serão animados os experimentos.

Para se criar uma tag, primeiramente é preciso ter uma tabela com todos os pontos necessários para os projetos, conforme exemplificado na figura 36. Essa tabela poderá ser facilmente extraída do CLP em questão.



Elem...	Note	Direct Used	Indirect I
X000			
X001	BOTAO PEDESTRE VIA A	*	
X002	BOTAO PEDESTRE VIA B	*	
X003	BOTAO PEDESTRE VIA C	*	
X004	BOTAO PEDESTRE VIA D	*	
X005	BOTAO LIGA/DESLIGA	*	
X006	SENSOR DE VEICULOS VIA A	*	
X007	SENSOR DEVEICULOS VIA B	*	
X010	SENSOR DE VEICULOS VIA C	*	
X011	SENSOR DE VEICULOS VIA D	*	
X012			
X013			
X014			
X015	RST OS CONTATOS	*	
X016			
X017			
X020			
X021			
X022			

Figura 36 – Exemplo de tabela de pontos necessários para um projeto no Elipse SCADA.
Fonte: os autores, 2020.

Após, deverá retornar para a tela principal (*organizer*), como mostra a figura 37.

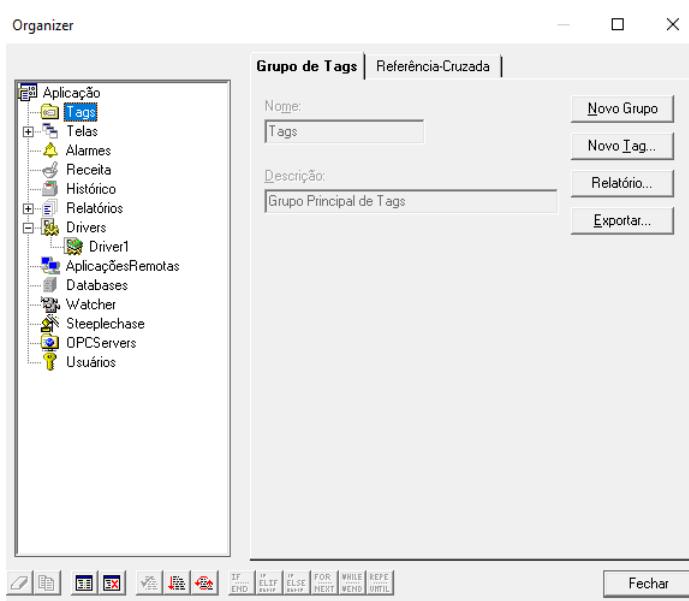


Figura 37 – Tela principal (organizer) no Elipse SCADA.
Fonte: os autores, 2020.

Nessa tela, deverá selecionar a função tag e o botão novo tag, com isso irá abrir uma nova tela para que se inicie a configuração.

Como os dados a serem supervisionados serão de um CLP, as tags deverão ser tags de CLP, pois esses dados serão lidos e escritos no CLP, conforme tela representada pela figura 38.

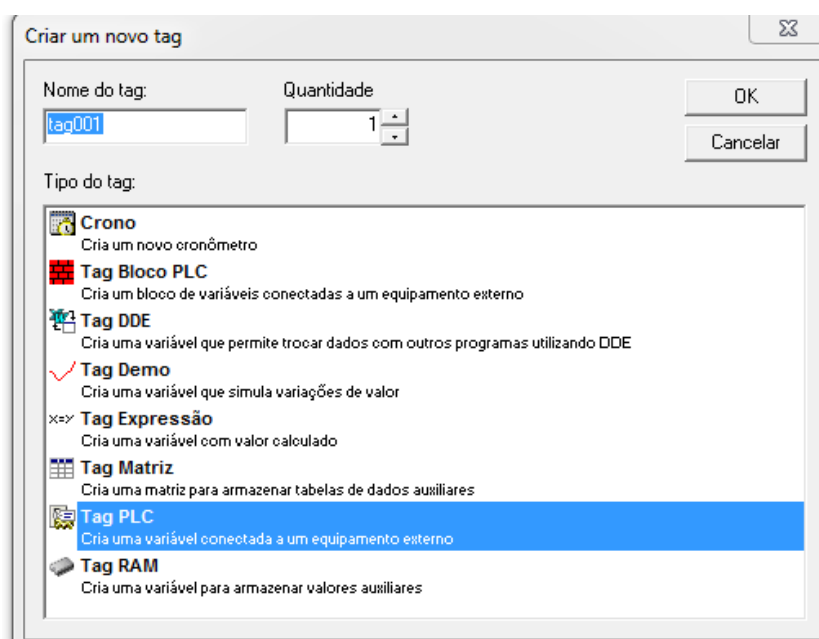


Figura 38 – Tela para criar novo tag no Elipse SCADA.
Fonte: os autores, 2020.

Uma vez selecionado o tipo de tag, se dará início a configuração de cada tag, conforme a tabela criada de tag.

O primeiro passo será dar um nome para a tag, como por exemplo “tag001” ou o nome que irá se utilizar no projeto, como por exemplo “início”.

Para o próximo passo, será necessário configurar os dados referentes aos campos N1, N2, N3 e N4, que são os dados de onde serão lidos e escritos entre CLP e supervisório.

Para isso, é necessário ter os dados referentes ao CLP e ao Elipse SCADA, conforme quadro 4.

**Quadro 4 – Dados referentes ao CLP e ao Elipse SCADA.
Mapeamento de Memória - Bits**

Endereços Modbus		Descrição dos Bits	Nº de Pontos	Comentário
Decimal	Hexadecimal			
0 - 255	0000H - 00FFH	X0 - X377	256 Pontos	Endereços disponíveis para as funções 01H, 05H e 0FH
256 - 511	0100H - 01FFH	Y0 - Y377	256 Pontos	
512 - 2047	0200H - 07FFH	M0 - M1535	1536 Pontos	
2048 - 3071	0800H - 0BFFH	S0 - S1023	1024 Pontos	
3072 - 3583	0C00H - 0DFFH	T0 - T511	512 Pontos	
3584 - 3839	0E00H - 0EFFH	C0 - C255	256 Pontos	
3840 - 4095	0F00H - 0FFFH	M8000 - M8255	256 Pontos	
4096 - 4351	1000H - 10FFH	M8256 - M8511	256 Pontos	
4352 - 7423	1100H - 1CFFH	S1024 - S4095	3072 Pontos	
7424 - 13567	1D00H - 34FFH	M1536 - M7679	6144 Pontos	
13568 - 16383	3500H - 3FFFH	Reservado	Reservado	

Fonte: Catálogo Modbus. 2020.

O quadro 4 se refere ao endereço de dados do CLP que deverá ser configurado no supervisório para cada tag. Por exemplo, se quiser ler a entrada X0, deverá ser preenchido no campo N4 como 1. Isso fará com que o supervisório entenda que será a primeira entrada apresentada pelo CLP. Caso queira efetuar a leitura do contador C0, deverá entrar com o valor 3585 e, com isso, o supervisório conseguirá ler o contador C0.

O próximo passo é descrever a função dos campos que deverão ser configurados, conforme representado pela figura 39. Estes campos são: N1, N2, N3 e N4.

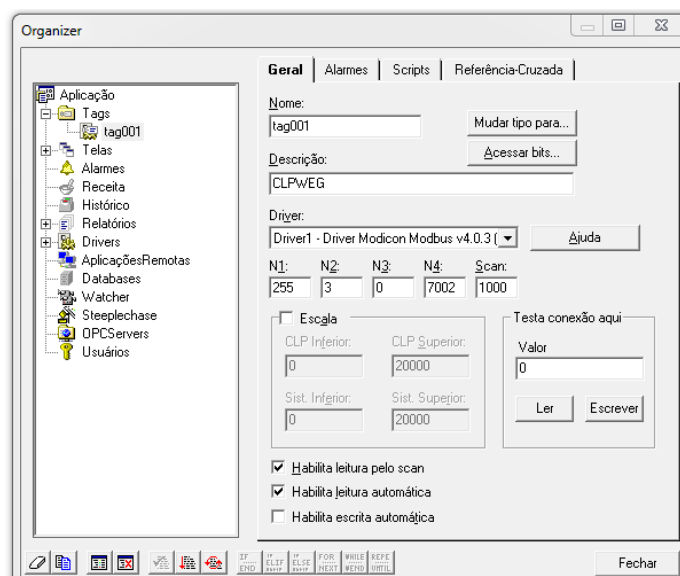


Figura 39 – Tela para configurar uma tag criada no Eclipse SCADA.

Fonte: os autores, 2020.

- **N1:** Endereço do equipamento escravo na rede. Este endereço é usado em redes seriais e pode variar de 1 a 247. Pode-se ainda configurar este parâmetro com o valor 0 (zero). Com isto, este tag trabalha em modo Broadcast, enviando a mensagem para todos os equipamentos escravos (CLP) que estiverem na rede.
- **N2:** Código da operação. Faz referência a uma das operações citadas na página anterior, na configuração do driver. É em geral modificado em função do tamanho do dado a ser lido.
- **N3:** Parâmetro Adicional. Em geral não é usado e pode ser deixado em 0 (zero).
- **N4:** Endereço do registrador, variável ou bit do dispositivo que se deseja ler. Funciona como offset, pois o valor retornado será o valor do Endereço Base + N4 (Offset).

É importante salientar que cada tag deverá ter sua configuração desses campos. A figura 40 demonstra uma tela com vários tags criados.

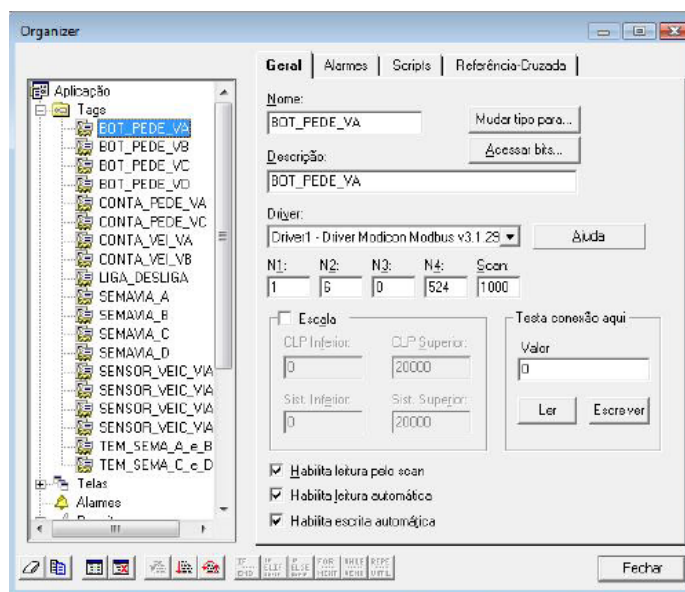


Figura 40 – Tela exemplo com uma lista de tags criadas no Elipse SCADA.
Fonte: os autores, 2020.

É possível efetuar teste conforme a configuração da tag, demonstrado na figura 41.

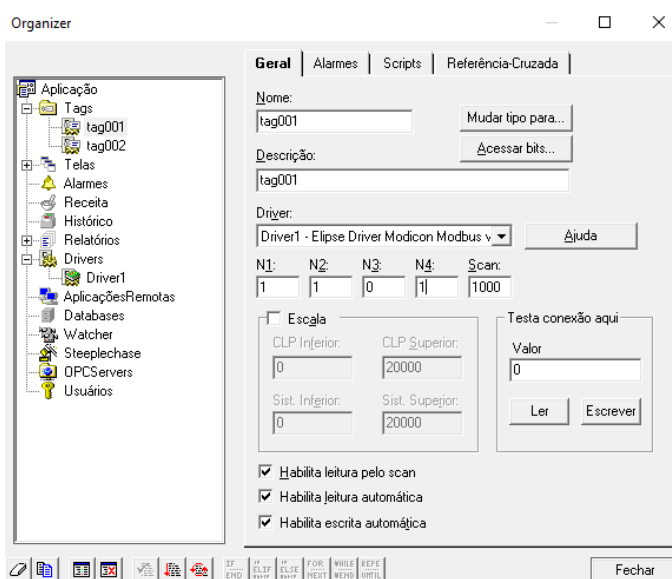


Figura 41 – Tela para teste conforme configuração de tags no Elipse SCADA.
Fonte: os autores, 2020.

Nessa tela, pode-se notar que foi configurado no campo N1 = 1. Esse valor corresponde ao endereço do equipamento escravo, ou seja, no nosso caso será o CLP. No campo N2 = 1 foi configurado conforme a função que a tag irá ler, configurado no drive. No campo N3 = 0, por padrão, nesse caso não irá ser utilizado. Por último, o N4 = 1 será a leitura da entrada X0 do CLP. Para se fazer o teste, é

preciso ir no campo e testar a conexão, e colocar um valor, no caso binário, e mandar ler ou escrever. Se na entrada X0 estiver igual a 1, ao se acionar ler no campo, o valor que irá aparecer será 1, ou se quiser escrever é só clicar nesse campo que irá escrever, na lógica do CLP, o valor inserido.

3.4 Aplicação para o laboratório

Após ter sido realizada uma comunicação bem sucedida entre o CLP e o software supervisor Elipse SCADA, é possível utilizar os recursos do programa para efetuar a criação de telas para controle de processos devidamente programados em LADDER. Estas telas são muito empregadas em salas de controle de indústrias, onde ocorre todo o comando de máquinas de produção.

No ambiente laboratorial educativo de uma universidade, cabe ao estudante estabelecer uma relação amigável com o software supervisor, a fim de aprender a manusear o programa da melhor maneira possível. Neste trabalho, foi feita uma tela-exemplo de um controle do nível de um tanque de água, como pode ser visto na figura 42.

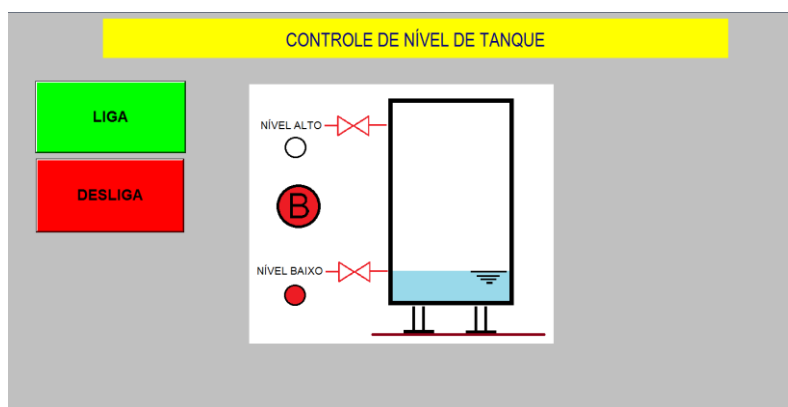


Figura 42 – Tela de controle de nível de tanque criada no supervisor Elipse SCADA.

Fonte: os autores, 2020.

As tags utilizadas na programação estão apresentadas pelas telas ilustradas pelas figuras 43, 44, 45, 46 e 47.

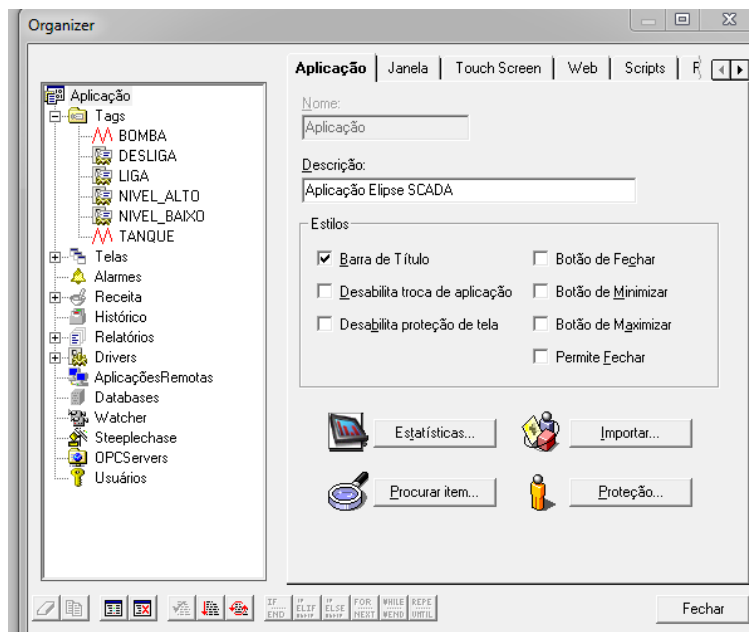


Figura 43 – Lista de tags utilizadas na tela.

Fonte: os autores, 2020.

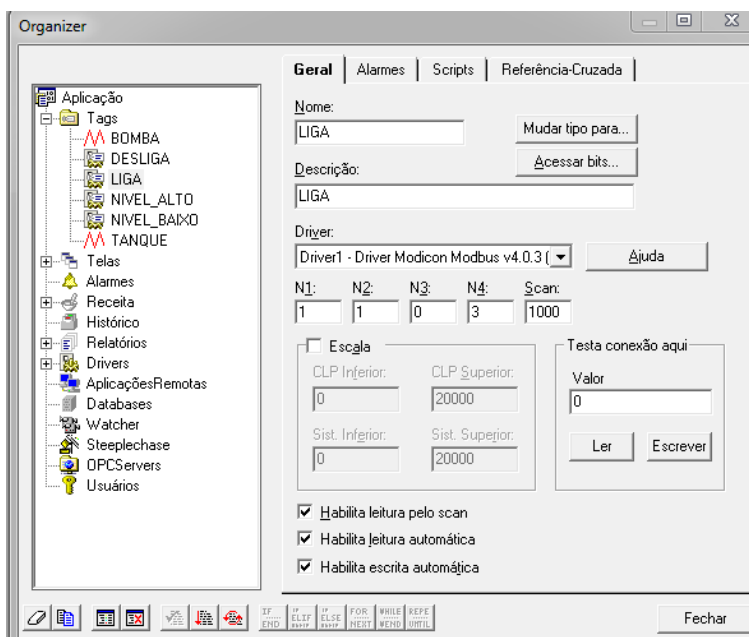


Figura 44 – Tag LIGA.

Fonte: os autores, 2020.

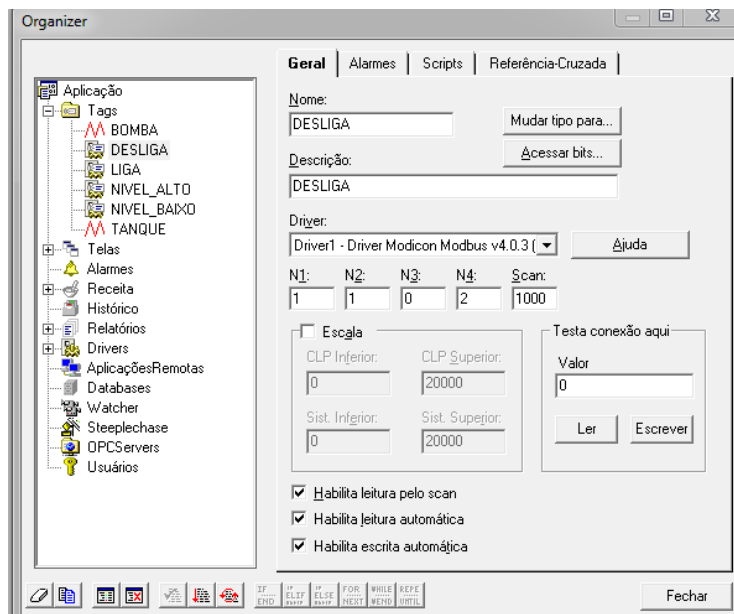


Figura 45 – Tag DESLIGA.

Fonte: os autores, 2020.

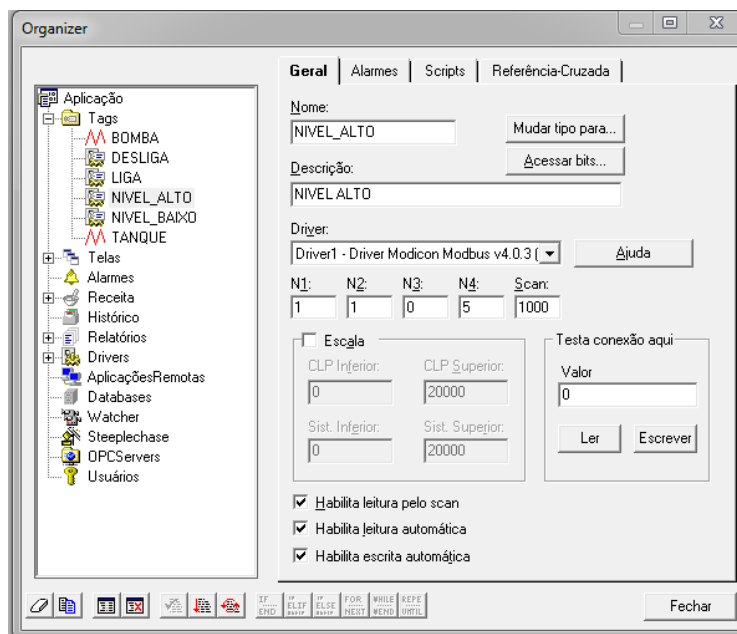


Figura 46 – Tag NIVEL ALTO.

Fonte: os autores, 2020.

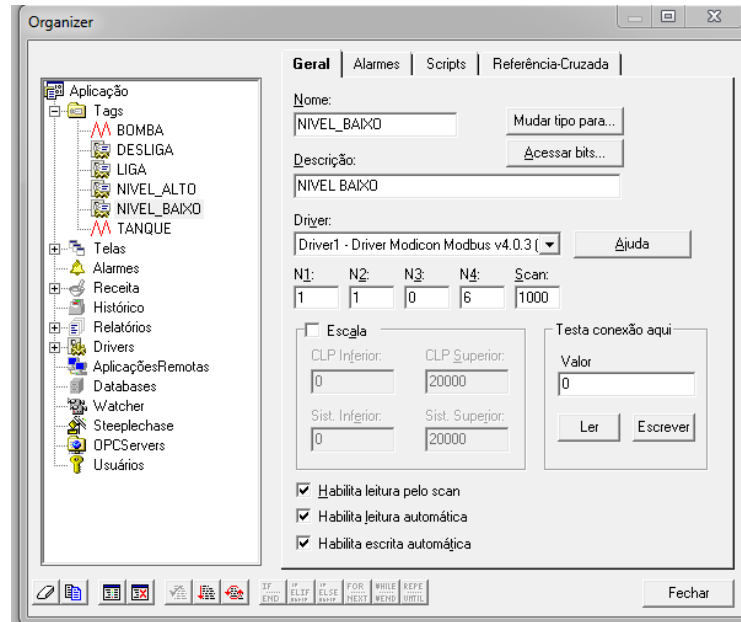


Figura 47 – Tag NIVEL BAIXO.
Fonte: os autores, 2020.

A programação demonstrada pela figura 48 foi construída através do CLP TPW03, utilizando os princípios das TAGs indicadas na figura 47.

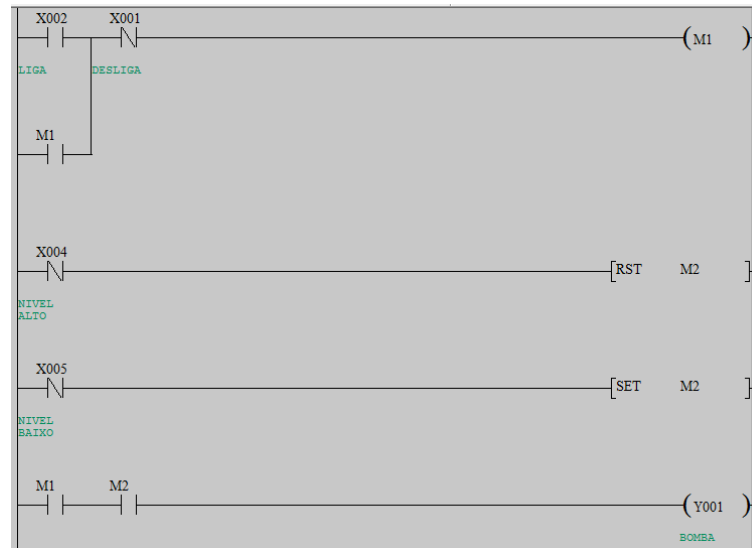


Figura 48 – Programação básica de um controle de nível de tanque criada no CLP TPW03.
Fonte: os autores, 2020.

A programação funciona da seguinte maneira: inicialmente, vamos adotar que o tanque esteja vazio. Ao se executar a programação, temos o contato (bit) X002 normalmente aberto. Ao se pressionar o botão liga, essa informação muda o estado desse bit, passando para 1 e tornando-se verdadeiro. Com isso, M1 – o bit de memória – irá a 1, dando condição para que o processo entre em automático, isso

pode ser visto na quarta linha da programação LADDER. Quando o nível do tanque atingir o sensor de nível alto, o sinal indicativo de nível alto irá se acender na cor verde, conforme tag mostrado na figura 46 (N4 = “5”) , indicando que a bomba deverá ser desligada e, assim, segundo a programação, X004 (NÍVEL ALTO) resetará uma memória M2 que passará para 0 (não acionada). Depois disso, o tanque começará a se esvaziar até que atinja o sensor de nível baixo (x005). Isso fará com que o sinal chegue até o CLP, o contato (bit) X005 torne-se verdadeiro e o sistema efetue o start da bomba, setando através da função SET M2. Esse sinal nível baixo será representado no supervisório conforme figura 47 e, com isso, irá alterar a sua cor para vermelho, indicando tanque vazio. Somente após subir o nível, sua cor irá sair do vermelho. A configuração desse sinal será conforme mostrado na tag ilustrada na figura 47 (N4 = “6”). Como o sistema estará em automático, o processo irá se repetir. O esquema segue ilustrado pelo fluxograma conforme figura 49.

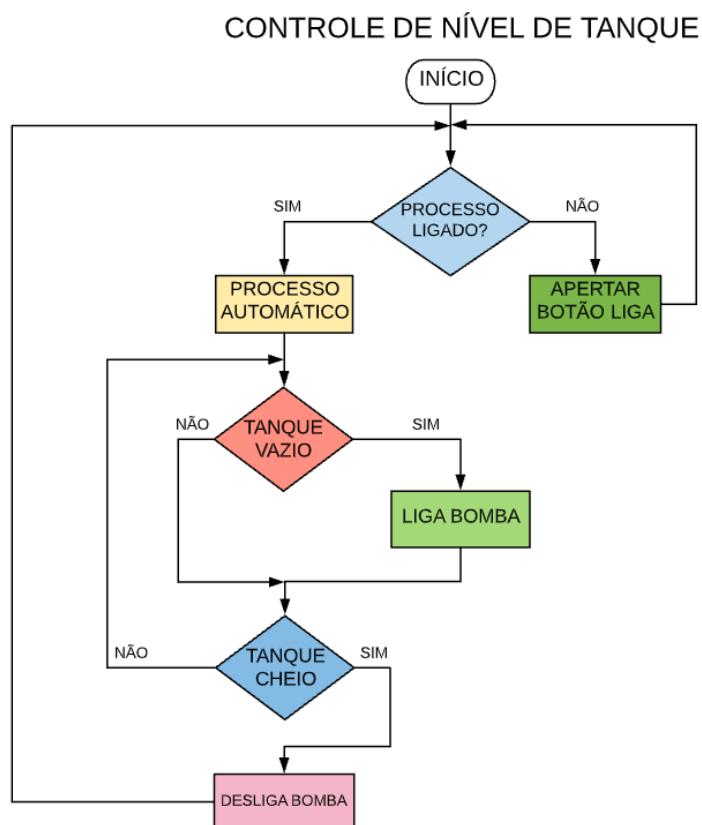


Figura 49 – Fluxograma do Controle de Nível de Tanque.
Fonte: os autores, 2020.

Na tela, foi desenhado o botão LIGA, que conforme descrito acima, liga o sistema. O tag para esse botão ficará conforme a tabela tag apresentado pela figura 44 (N4 = “3”), que, ao apertar, faz a bomba ligar e o tanque encher. A animação do tanque começará a ser preenchida com água até chegar no nível alto, como indicado na figura 45. O botão DESLIGA (configurado na figura 45, de tag N4 = “2”) será utilizado quando se deseja efetuar uma parada no sistema e com isso o processo estará fora de automático. Esta situação hipotética mostra que é possível controlar inúmeros e diferentes processos remotamente com este recurso que é o supervisor. A figura 50 representa a situação de tanque cheio, ou seja, com nível alto.

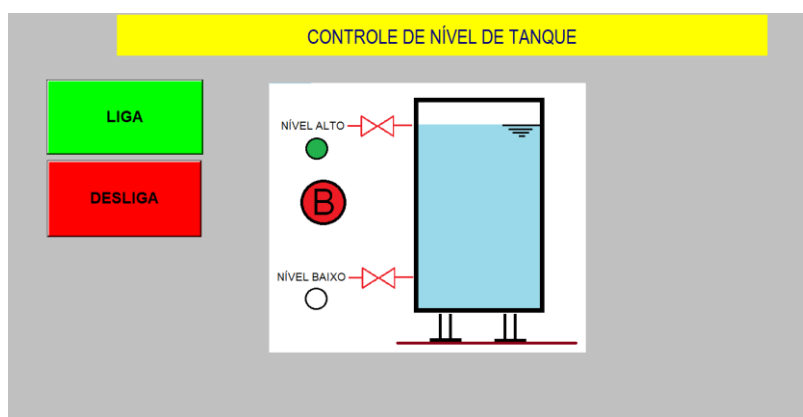


Figura 50 – Tela do controle de nível de tanque acusando nível alto.
Fonte: os autores, 2020.

4 CONCLUSÃO

As pesquisas realizadas e a parte remota desenvolvida foram de grande aproveitamento, a partir das quais podemos analisar o CLP como a inteligência no processo de automatização. São produzidos por diversas empresas, porém, optamos em utilizar o da WEG modelo TPW3-PCLINK, que possui um grande número de funções e uma capacidade de processamento altíssima, o que o habilita para desenvolver uma infinidade de tarefas. O sistema supervisório escolhido foi o Eclipse SCADA, por proporcionar desde o simples até o mais complexo processo de interface HMI em tempo real e também por ser atualmente o líder no mercado industrial. Com o objetivo de construir uma tela para controle de determinado processo fundamental na indústria, iniciamos com nossos encontros semanais, o que gerou primeiro desafio, ser acompanhado remotamente e explorar algo novo. Conforme o andamento do projeto, foram identificados novos desafios e dúvidas, como o meio utilizado para a comunicação física do supervisório com o CLP, logo definida através de cabo de programação ou porta de comunicação RS-485. Seguimos o projeto e passamos para a programação do software, cada parte gerando seu determinado desafio e somando com muito conhecimento e experiência.

No decorrer deste trabalho, pudemos examinar como são importantes os equipamentos de Automação para apresentar um controle de acesso mais seguro para os ambientes industriais e para comprovar que as ferramentas utilizadas no ramo da Automação Industrial são bastante completas a ponto de realizar funções que muitas vezes não envolvem controle de máquinas e processos necessariamente.

Ainda há muito a se explorar nessa área e a universidade tem muito a oferecer e grande capacidade em desenvolvimento de projetos de controle em nosso laboratório.

REFERÊNCIAS

TECHSOUP. **O que são Redes? Noções Básicas.** 2019. Disponível em: <https://www.techsoupbrasil.org.br/node/1916>. Acesso em: 05 fev. 2020.

OFICINA DA NET. **Topologia de Redes.** 2013. Disponível em: https://www.oficinadanet.com.br/artigo/2254/topologia_de_redes_vantagens_e_desvantagens. Acesso em: 07 fev.2020.

FARIAS, G.; MEDEIROS, E.; **Introdução à Computação.** 1ª Ed. Disponível em: <http://producao.virtual.ufpb.br/books/camyle/introducao-a-computacao-livro/livro/livro.chunked/ch07s03.html>. Acesso em: 07 fev. 2020.

CALDEIRA Pires, Armando. **Sensores e sua utilização: História e Definições.** 2017. 18f. Dissertação de Mestrado: Faculdade de Tecnologia: Departamento de Engenharia Mecânica. Brasília, 2015.

BRUSAMARELLO, V.J; BALBINOT, A; **Instrumentação e Fundamentos de Medida** (Volumes 1 e 2). São Paulo: Livros Técnicos e Cient. Editora, 2007.

PUC-RS **Sensores Industriais.** 2015. 53f.. Porto Alegre, 2015. Disponível em: <https://www.politecnica.pucrs.br/professores/tergolina/Automacao_e_Control/APR_ESENTACAO_-_Aula_03_Sensores_Industriais.pdf>. Acesso em: 16 fev. 2020.

DIRECTINDUSTRY **Que sensor de proximidade escolher?** Disponível em: <<http://guide.directindustry.com/pt/que-sensor-de-proximidade-escolher/>>. Acesso em: 16 fev. 2020.

DRIEMEIER, Larissa; ALVES, Marcílio; MOURA, Rafael T.; **Métodos Experimentais em Sistemas Mecânicos.** Aula Introdutória. 122f. GMSIE – Grupo de Mecânica dos Sólidos e Impacto em Estruturas. São Paulo. Disponível em: https://edisciplinas.usp.br/pluginfile.php/4119238/mod_resource/content/1/01%20Introdu%C3%A7%C3%A3o%20a%20sistemas%20de%20medi%C3%A7%C3%A3o.pdf. > Acesso em: 20 fev. 2020.

Eicos Sensores. **Sensor de Nível: o que é?** Disponível em: <http://eicos.blog.br/sensor-de-nivel-o-que-e/>. Acesso em: 25 fev. 2020.

AUGUSTO, Heber. **Sistema Supervisório: Entenda o que é.** 2016. Disponível em: <<https://www.hitecnologia.com.br/blog/sistema-supervisorio-entenda-o-que-e/>>. Acesso em: 01 mar. 2020.

MOTT, Anderson. **O que são Sistemas Supervisórios?** 2019. Disponível em: <<https://www.automacaoindustrial.info/o-que-sao-sistemas-supervisorios/>> Acesso em: 03 mar. 2020.

ANDRADE, Fabrício. **Tudo sobre sistema supervisório!** 2018. Disponível em: <https://automacaoecartoons.com/2018/01/15/sistema-supervisorio-scada-ihm/>> Acesso em: 03 mar. 2020.

ENGPROCESS. **Sistemas supervisórios: o que são?** 2017. Disponível em: <<https://engprocess.com.br/sistemas-supervisorios/>> Acesso em: 03 mar. 2020

EMBARCADOS. **Sensores de Corrente VishayDale.** 2016. Disponível em: <<https://www.embarcados.com.br/sensores-de-corrente-vishay-dale/>> Acesso em: 04 mar. 2020.

SCADA LAquis. **Supervisório LAquis.** <Disponível em: <https://www.lcds.com.br/laquis.asp>> Acesso em: 21 fev. 2020.

USINAINFO Eletrônica & Robótica. **Sensor de Tensão AC 127/220V – P8.** Disponível em: <<https://www.usinainfo.com.br/sensor-de-tensao-arduino/sensor-de-tensao-ac-127220v-p8-3728.html>> Acesso em: 20 abr. 2020.

CYRINO, Luis. Manutenção em Foco. **Atuadores de Máquinas e Equipamentos.** Disponível em: <<https://www.manutencaoemfoco.com.br/atuadores-de-maquinas-e-equipamentos/>> Acesso em: 29 abr. 2020.

INSTITUTO FEDERAL DO RIO GRANDE DO SUL (IFRS). **Regulamento de uso dos laboratórios.** Ibirubá: 2017.

Centro Universitário UNIVATES. **Regulamento dos laboratórios de uso específico.** Lajeado: 2016.

EXSTO TECNOLOGIA. **Normas de Segurança Aplicáveis a Laboratórios de Ensino.** Disponível em: <<https://exsto.com.br/download/ebook-normas-seguranca-laboratorios-ensino.pdf>> Acesso em: 29 abr. 2020.

CERTI – **SCADABR.** Disponível em: <https://sites.google.com/a/certi.org.br/certi_scadabr/home/minicursos/iniciando-scadabr>. Acesso em: 29 abr. 2020.

INDUSOFTWEBSTUDIO. **Conhecendo o InduSoftWebStudio.** Disponível em: <https://www.indusoft.com.br/Recursos/Conhecendo-o-InduSoft-Web-Studio>. Acesso em: 29 abr. 2020.

AQUARIUS. **ProficyiFix.** Disponível em: <https://www.aquarius.com.br/produto/ge-ifix/>. Acesso em: 29 abr. 2020.

ELIPSE SOFTWARE. **Elipse E3.** Disponível em: <http://www.elipse.com.br/produto/elipse-e3/>. Acesso em: 29 abr. 2020.

BLUELUX. **Contatores: o que são e para que servem?** Disponível em: <https://www.bluelux.com.br/contatores-o-que-sao-e-para-que-servem>. Acesso em: 01 jun. 2020.

KRAKECHE, I.; ZWIRTES, M; FISTAROL, A; TIBOLLA, V. **Sensores de Temperatura,** Disponível em: <<http://hermes.ucs.br/ccet/demc/vjbrusam/inst/temp2.pdf>>, Acesso em: 27 abr. 2020

ANTONELLI, Pedro Luis. **Introdução aos controladores lógicos programáveis (CLPs)**. Disponível em: < www.ejm.com.br/download/Introducao%20CLP.pdf>. Acesso em: 17 mar. 2020.

FRANCHI, Claiton Moro; CAMARGO, Valter Luís Arlindo de. **Controladores lógicos programáveis - sistemas discretos**. São Paulo: Érica, 2008. Acesso em: 17 mar. 2020.

CASSILO, Danielle. **Automação e controle**. Universidade Federal Rural Do SemiÁrido (UFERSA). Disponível em: < <http://www2.ufersa.edu.br/portal/view/uploads/setores/166/arquivos/Automacao>>. Acesso em: 10 jan. 2020.

MORAES; CASTRUCCI. **Engenharia de Automação Industrial**. [S.l.]: Editora LTC, 2007. Acesso em: 10 jan. 2020

CASSIOLATO, C.; ALVES, E. **Medição de vazão**. Revista Controle & Instrumentação, 2008. Acesso em: 27 jan. 2020

CARVALHO, P. C. de. **Arquiteturas de sistemas de automação industrial utilizando CLPS** - parte 2. Mecatrônica Atual, 2003. Acesso em: 29 de janeiro de 2020.

MARTINS, M. G. **Princípios de Automação Industrial**. [S.l.]: Universidade Federal de Santa Maria, 2012. Acesso em: 07 mar. 2020.

CASSIOLATO, C. **Sistemas de supervisão e aquisição de dados**. Eletrônica Industrial, 2011. Acesso em: 09 mar. 2020.

MURARO, R. M. **A automação e o futuro do homem**. [S.l.]: Editora Vozes, 1969. Acesso em: 09 mar. 2020.

WASHINGTON, B. **Fenômenos de Transporte para Engenharia**. [S.l.]: Editora LTC, 2012. Acesso em: 10 mar. 2020.

Guia PIRELLI. **Contatores elétricos**. 2012. Acesso em 11 mar. 2020.

ANENBAUM, Andrew S.. **Redes de computadores**. 2. Ed. Rio de Janeiro: Campus, 1994. Acesso em: 12 mar. 2020.

SCHNEIDER ELETRONICS. **Modbus Protocol 2000**. Disponível em: <<http://www.modicon.com/techpubs/toc7.html>>. Acesso em: 12 mar. 2020.

ANEXO A: PROTOCOLO MODICON MODBUS PARA ELIPSE SCADA

*elipse*software

MODICON MODBUS PARA ELIPSE SCADA

*elipse*software

Nome do arquivo:	MODBUS32.DLL
Fabricante:	Schneider Electric
Protocolo:	Modicon Modbus ASC/RTU/TCP
Última versão:	22/12/2004

Introdução

Esse driver implementa o protocolo Modbus Master/Slave, permitindo uma aplicação Elipse atuar como Mestre em uma rede Modbus, comunicando-se com qualquer equipamento escravo que implemente os protocolos Modbus modos ASCII ou RTU, ou Modbus TCP.

O protocolo MODBUS é um protocolo baseado em mensagens, posicionado no nível 7 do modelo OSI, que possibilita comunicação cliente/servidor entre equipamentos conectados a diferentes tipos de redes. É um protocolo baseado em mensagens de comando e resposta, oferecendo serviços com funções definidas por um código de 8 bits.

Existem três categorias de códigos de funções:

- **Códigos de Funções Públicas:** Funções bem definidas pelo protocolo, com garantia de unicidade, validadas pela comunidade Modbus.org e publicamente documentadas em MB IETF RFC.
- **Códigos de Funções Definidas pelo Usuário:** Funções definidas pelo usuário não são padronizadas e não precisam de aprovação pela Modbus.org, não tendo portando qualquer garantia de unicidade, podendo ser livremente implementadas.
- **Códigos de Funções reservadas:** Códigos atualmente usados por alguns fabricantes em produtos antigos, não estando disponíveis para uso público.

Esse driver implementa todas as funções públicas com exceção das funções 22, 23 e 43, bem como algumas funções específicas de fabricantes.

O protocolo Modbus foi desenvolvido inicialmente pela Modicon (atual Schneider Electric), em 1979, sendo hoje um padrão aberto, implementado por centenas de fabricantes em milhares de equipamentos.

Maiores informações referentes ao protocolo Modbus podem ser obtidas no site www.modbus.org.

Parâmetros de Configuração

Parâmetros [P] de configuração do driver

- P1** Porta de comunicação (COM1 = 1, COM2 = 2, COMn = n)
- P2** Paridade (veja Tabela 1) + Baudrate (veja Tabela 2)
- P3** Databits (veja Tabela 3) + Stopbits (veja Tabela 4)
Ex: 9600bps e Paridade Par = 60 + 2, logo P2 = 62
- P4** Timeout em milisegundos
Ex: 8 Databits e 2 Stopbits = 8 + 20, logo P3 = 28

elipsesoftware

Tabela 1 (Paridade)

Valor	Paridade
0	Sem Paridade
1	Paridade Ímpar
2	Paridade Par

Tabela 2 (Baudrate)

Valor	Baudrate (bps)
0	110
10	300
20	600
30	1200
40	2400
50	4800
60	9600
70	19200
80	28800
90	38400
100	56000
110	57600
120	115200

Tabela 3 (Databits)

Valor	Databits
7	7
8	8

Tabela 4 (Stopbits)

Valor	Stopbits
10	1
20	2

NOTA: Para Network: Os parâmetros P1, P2 e P3 serão ignorados.

Parâmetros [N] de endereçamento de tags PLC

- N1 Endereço da CPU.
- N2 Número do "Func", criado em "Extras" para leitura e escrita.
- N3 NA
- N4 Endereço da variável.

NOTA: Se for utilizada a função Modbus 20 ou 21, N3 será utilizado para informar o número do arquivo.

Se N1 = 9999 será processado o "Código de Exceção", enviado pelo Modbus Slave.

Modo Leitura: Obtém o último Código de Exceção do Modbus Slave, registrado pelo driver.

Modo Escrita: Altera/Zera o último Código de Exceção, registrado pelo driver.

*elipse*software

Se **N1 = 9998** será processado o controle da função "Halt" (se habilitado em Extras) sobre o driver.

Modo Leitura: Obtém o estado atual do driver.

Possíveis valores de Leitura:

0 = driver em estado Normal.

1 = driver em estado Halt.

Modo Escrita: Modifica o estado atual do driver.

Possíveis valores de Escrita:

0 = força para o estado Normal. (caso o driver esteja em estado Halt)

1 = força para o estado Halt. (caso o driver esteja em estado Normal)

Para Network

N1 Endereço da CPU

N2 Número do "Func" criado em "Extras" para leitura e escrita

N3 Número do "Func" criado em "Extras" para endereçamento do IP

N4 Endereço da variável

NOTA: Se for utilizada a função Modbus **20** ou **21**, **N3** será utilizado para informar o número do arquivo e a parte alta de **N2** indicará o número do "Func" criado em "Extras" para endereçamento do IP e a parte baixa de **N2** indicará o número do "Func" criado em "Extras" para leitura e escrita.

Para Modem

N1 -1 (menos um)

N2 Comando do modem (veja Tabela 5)

N3 Se discar (**N2=1**), número da portadora.

N4 Não utilizado, deixar em zero.

Tabela 5 (comandos do modem)

N2	Função ou Valor	R/W?
N2 = 0	Número do telefone	R/W (String)
N2 = 1	Disca	W (Command Write)
N2 = 2	Status do modem	R (String)
N2 = 3	Carrier	R (Bool)
N2 = 4	Escape	W (Command Write)
N2 = 5	Baud Rate	R (Double)

Parâmetros [B] de endereçamento de tags bloco

B1 Endereço da CPU

B2 Número do "Func" criado em "Extras" para leitura e escrita

B3 NA

B4 Endereço da variável

NOTA: Se for utilizada a função Modbus **20** ou **21**, **B3** será utilizado para informar o número do arquivo.

elipsesoftware

Para Network

- B1** Endereço da CPU
- B2** Número do "Func." criado em "Extras" para leitura e escrita
- B3** Número do "Func." criado em "Extras" para endereçamento do IP
- B4** Endereço da variável.

NOTA: Se for utilizada a função Modbus 20 ou 21, B3 será utilizado para informar o número do arquivo e a parte alta de B2 indicará o número do "Func" criado em "Extras" para endereçamento do IP e a parte baixa de B2 indicará o número do "Func" criado em "Extras" para leitura e escrita.

Observações para Comunicação Network:

- A porta padrão para comunicação com MODBUS/TCP é 502.
- Certifique-se que a opção "Use Network" esteja marcada.

Parâmetros Extras do Driver

Através desta Janela, é possível configurar parâmetros adicionais do driver:

Dialog Log

Pode ser especificada a criação de um arquivo de Log, incluindo o método de criação do arquivo: Create New (Novo) ou Append Data (insere os novos dados no final do arquivo) e o caminho do mesmo.

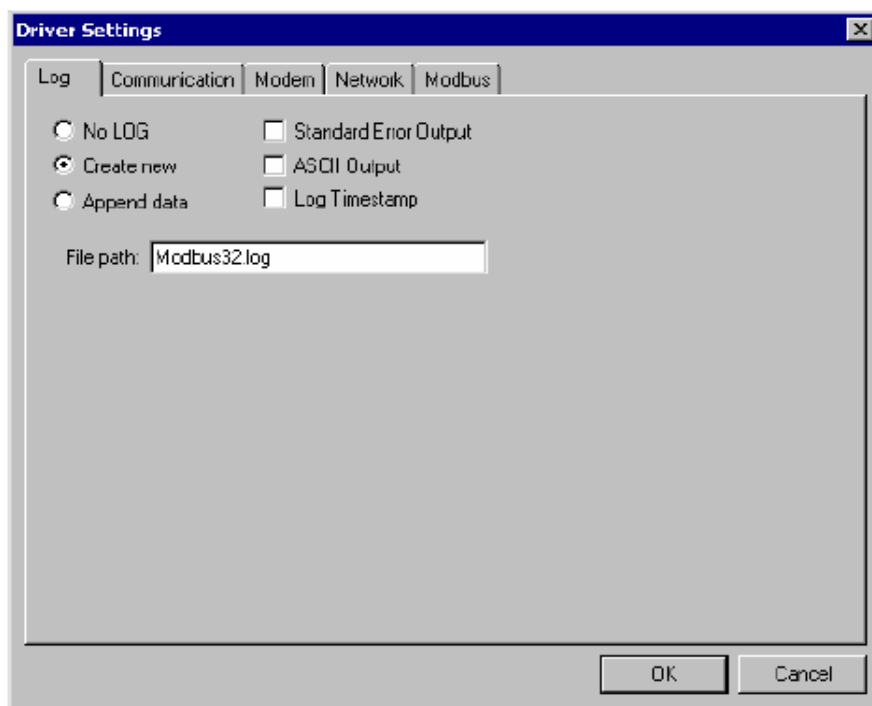


Figura 1: Configurações do Driver

elipsesoftware

Dialog Communication

Podem ser especificados o controle de fluxo e delays de comunicação.

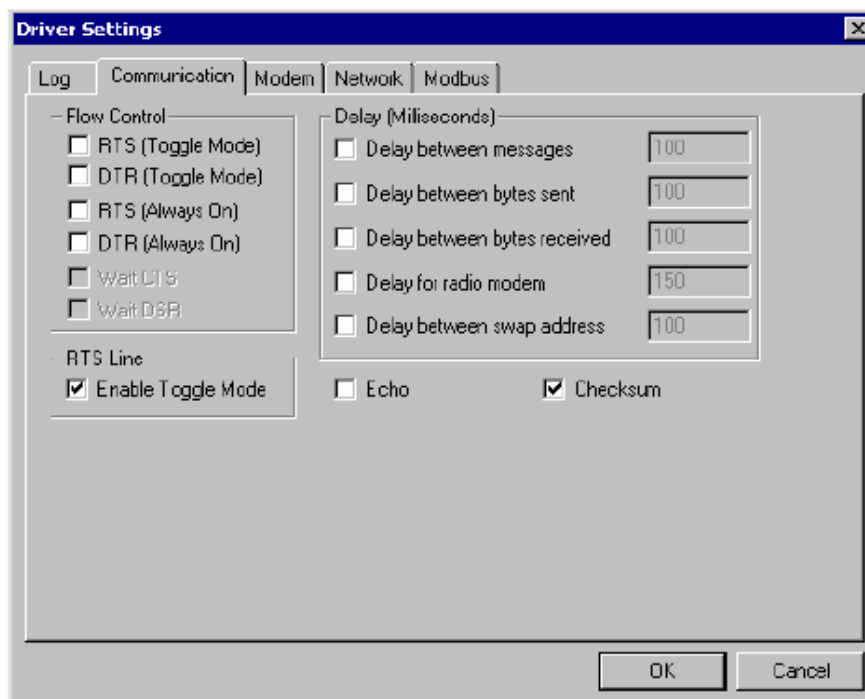


Figura 2:: Configurações de Comunicação do Driver

As opções disponíveis são as seguintes:

- **RTS (Toggle Mode):** Indica que o driver vai controlar, via software, o controle de fluxo do pino RTS da porta serial. O RTS será ligado antes da transmissão e desligado após a mesma, a fim de esperar a mensagem de resposta.
- **DTR (Toggle Mode):** Indica que o driver vai controlar, via software, o controle de fluxo do pino DTR da porta serial. O DTR será ligado antes da transmissão e desligado após a mesma, a fim de esperar a mensagem de resposta.
- **RTS (Always On):** Indica que o driver vai deixar o pino RTS da porta serial sempre ligado, a partir do momento que iniciar a comunicação. Isto deve ser feito quando o modem a ser utilizado é de uma linha privada full-duplex em alguns casos, ou a depender da situação.
- **DTR (Always On):** Indica que o driver vai deixar o pino DTR da porta serial sempre ligado, a partir do momento que iniciar a comunicação. Isto deve ser feito quando o modem a ser utilizado é de uma linha privada full-duplex em alguns casos, ou a depender da situação.
- **Wait CTS:** Indica que o driver deve esperar a confirmação do sinal CTS, para começar a transmitir. O sinal CTS é uma resposta do modem quando se seta o pino RTS.
- **Wait DSR:** Indica que o driver deve esperar a confirmação do sinal DSR, para começar a transmitir. O sinal DSR é uma resposta do modem quando se seta o pino DTR.
- **RTS Line: Enable Toggle Mode:** Este modo é utilizado quando o modo de chaveamento via software não opera conforme esperado, especialmente em sistemas operacionais como o

elipsesoftware

Windows NT e 2000, onde o acesso ao hardware não é direto, causando delays no chaveamento. Nesse caso, deixa-se ao encargo do sistema operacional o chaveamento do RTS.

- **Echo:** Indica se o driver deve, a cada caractere enviado, ler o mesmo caractere da porta. Isso somente é utilizado quando o modem possui eco, o que implicaria em caracteres a mais do que o esperado na porta serial.
- **Checksum:** Especifica se o driver fará ou não a verificação de checksum.
- **Delays:** Podem ser especificados delays entre cada uma das mensagens, entre cada byte enviado ou o timeout entre cada byte recebido, em cada uma das propriedades respectivas. Além disso, ainda pode ser feito Delay para radio modem ou forçar um Delay para cada troca de endereço dos equipamentos em rede, se necessário.

Dialog Modem

Pode ser especificado o modem e numero de discagem. A opção Accept Incoming Calls permite que o driver receba chamadas do equipamento. A conexão pode ser monitorada através do tag Carrier (N1=1, N2=3), e o número de telefone pode ser fixo, conforme propriedade Dial Number, ou ainda modificado através do tag Numero do Telefone (N1=-1, N2=0).

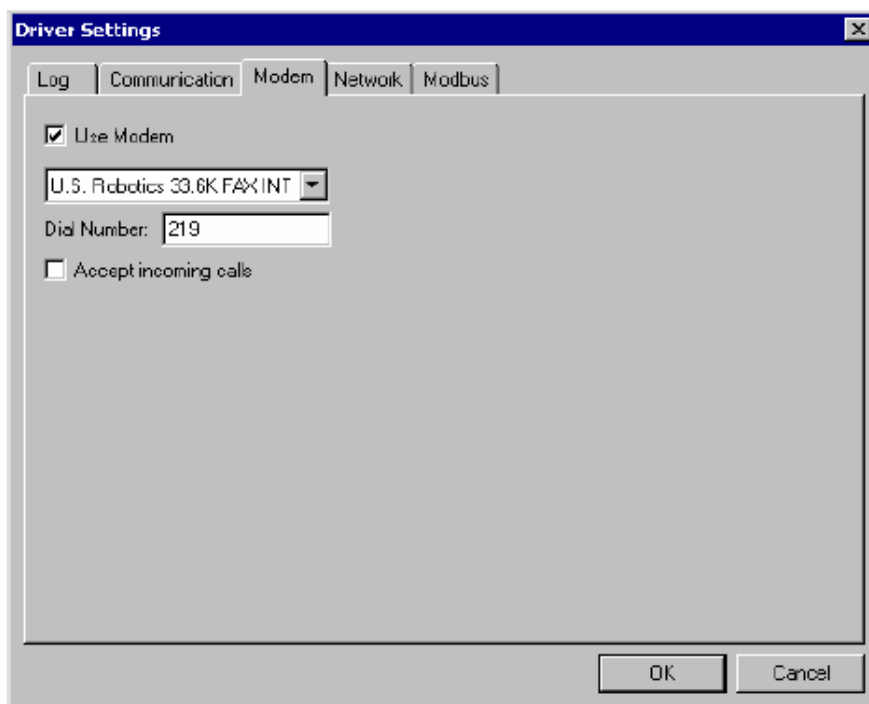


Figura 3: Configurações de Modem do Driver

elipsesoftware

Dialog Network

Na comunicação, via Network, devem ser especificados os endereços IP e Portas (sockets) dos dispositivos a serem conectados. Os endereços seqüenciais da lista devem ser referenciados no N3/B3 de cada tag.

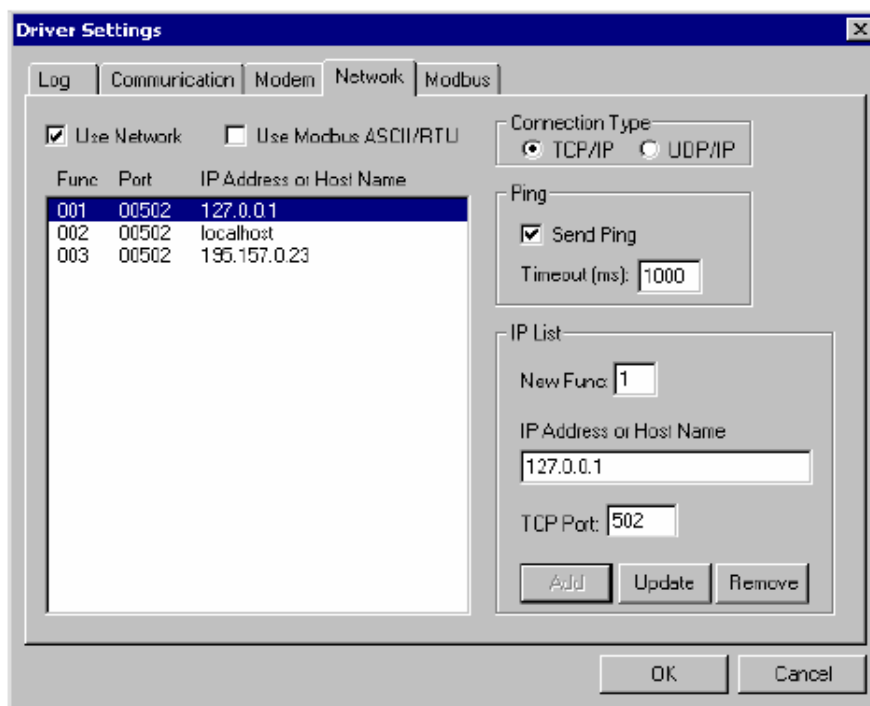


Figura 4: Configurações de Network do Driver

As opções disponíveis são as seguintes:

- **Use Network:** Habilita comunicação em rede.
- **Use Modbus ASCII/RTU:** Habilita o uso do protocolo Modbus ASCII/RTU via comunicação em rede. Se essa opção não estiver marcada, será utilizado o protocolo padrão Modbus/TCP.
- **New Func:** Número da "Func" que será utilizada nos parâmetros N2/B2.
- **Connection Type:** Define o tipo de conexão a ser utilizado, TCP/IP ou UDP/IP.
- **Send Ping:** Habilita um teste de conexão sobre o endereço de IP.
- **Timeout (ms):** Tempo de espera, em milisegundos, para a resposta do Ping.
- **IP Address or Host Name:** Endereço de IP do equipamento escravo.
- **TCP Port:** Define o número da porta a ser utilizado pelo socket. No protocolo Modbus/TCP, a porta padrão é 502.

elipsesoftware

Os botões Add/Update/Remove definem a lista de IP e Portas que serão utilizadas no sistema de rede.

- **Botão Add:** Adiciona um novo item a lista.
- **Botão Update:** Atualiza um item selecionado na lista.
- **Botão Remove:** Remove um item selecionado na lista.

Dialog Modbus

Aqui são especificadas as funções Modbus que serão utilizadas nos parâmetros N2/B2 dos tags. A cada código de função deve ser especificada uma função Modbus para Leitura e outra para Escrita, além do tipo de dado que será manipulado, e o tamanho do campo.

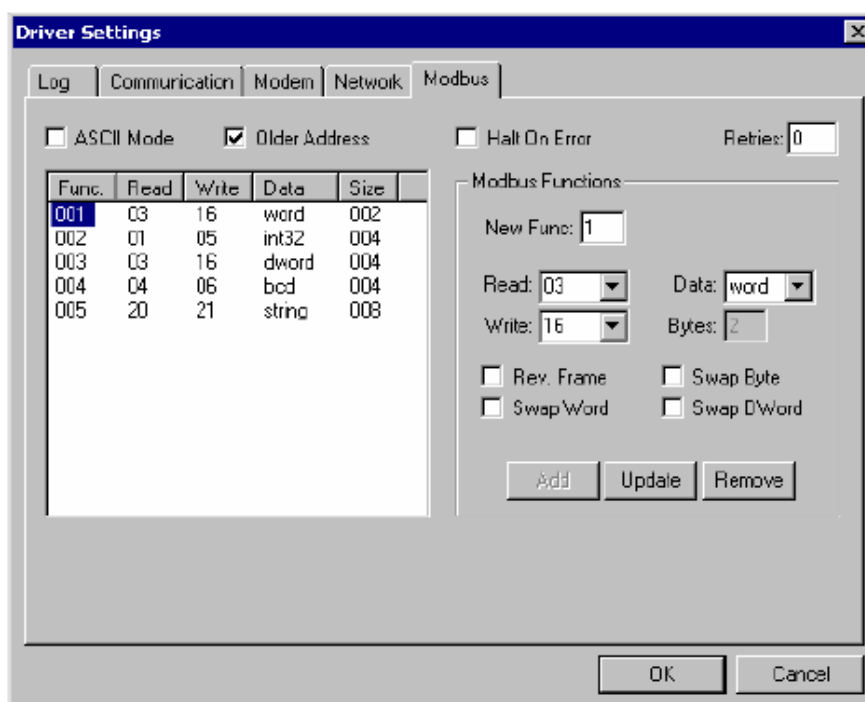


Figura 5: Configurações de ModBus do Driver

As opções disponíveis são as seguintes:

- **New Func:** Número da "Func" que será utilizada nos parâmetros N2/B2.
- **Retries:** Número de tentativas internas do driver, caso não consiga obter uma resposta válida do equipamento escravo.
- **Halt On Error:** Habilita/Desabilita a entrada do driver em modo "halt" após falharem todas as tentativas internas do driver (definido em "Retries"). O modo "Halt" é um recurso utilizado para forçar uma pausa na comunicação e liberar imediatamente o erro de leitura/escrita, sem precisar aguardar pelo estouro do tempo de "time-out".

Segue abaixo, as funções Modbus suportadas pelo driver:

Leitura

Valor	Descrição
01	Leitura de Bit (Read Coil Status - 0x)
02	Leitura de Bit (Read Input Status - 1x)
03	Leitura de Words (Read Holding Registers - 4x)
04	Leitura de Words (Read Input Registers - 3x)
07	Leitura de Status (Read Exception Status)
20	Leitura da Memória Extendida (Read General Reference - 6x)

Funções Especiais

Valor	Descrição
65 03	Leitura da Memória de Massa (somente para o equipamento ABB MGE 144)

Escrita

Valor	Descrição
05	Escrita de Bit (Force Single Coil - 0x)
06	Escrita de Word Simples (Preset Single Register - 4x)
15	Escrita de Bits (Force Multiple Coils - 0x)
16	Escrita de Words (Preset Multiple Registers - 4x)
21	Escrita na Memória Extendida (Write General Reference - 6x)

Funções Especiais

Valor	Descrição
65 01	Reset medidor de energia (somente para o equipamento ABB MGE 144)
65 02	Reset memória de máximo e mínimo (somente para o equipamento ABB MGE 144)

Tipos de Dados

Tipo	Descrição
Char	Palavra de 8 bits, caractere.
Byte	Palavra de 8 bits sem sinal.
Int8	Palavra de 8 bits com sinal.
Int16	Palavra de 16 bits com sinal.
Int32	Palavra de 32 bits com sinal.
Word	Palavra de 16 bits sem sinal.
Dword	Palavra de 32 bits sem sinal.
Float	Ponto Flutuante de 32 bits (IEEE 754) (4 bytes na ordem: EXP F2 F1 F0).
Double	Real de 64 bits.
String	Palavra de N chars (texto).
BCD	Palavra de 8 bits com código BCD.

- **Size:** Deve ser informado o tamanho em bytes do dado, a depender do tipo de dado selecionado. Exemplo: Dword deve ser de tamanho 4, assim como o float. Já um tipo string vai depender do tamanho do string esperado.
- **Rev Frame:** Indica que o sentido dos bytes no frame está invertido.
- **Swap Word:** Indica que o driver deverá inverter a ordem dos bytes dois a dois (em words) para obter o valor.
- **Swap Byte:** Indica que o driver deverá inverter a ordem dos bytes um a um para obter o valor.

elipsesoftware

- **Swap DWord:** Indica que o driver deverá inverter a ordem dos bytes quatro a quatro (em dwords) para obter o valor.
- **ASC Mode:** Habilita o uso do protocolo Modbus ASCII. Se essa opção não estiver marcada, será utilizado o protocolo padrão Modbus RTU.
- **Older Address:** Habilita informar, de modo direto, o valor da posição de memória (automaticamente o driver fará a subtração por 1 do valor informado, antes de enviar o frame de comunicação).

Os botões Add/Update/Remove definem a lista de funções Modbus que serão utilizadas no sistema.

- **Botão Add:** Adiciona um novo item a lista.
- **Botão Update:** Atualiza um item selecionado na lista.
- **Botão Remove:** Remove um item selecionado na lista.

Histórico das revisões do driver

Versão	Data	Autor	Comentários
v1.01	22.12.2004	C. Mello	- Adicionado suporte ao endereço escravo ZERO, para enviar comandos em modo broadcast. - Correção do problema de leitura/escrita nas funções Modbus que usam formato de dado do tipo "string". - Adicionada opção do registro em log do valor de "Timestamp", dos processos de leitura de dados. - Adicionado contador de transações para o protocolo Modbus TCP. - Suporte para nomes de IP no formato "host name". - Adicionado o parâmetro "Halt On Error", na página "Modbus", do dialog de Extras. - Substituição da opção "Log Date & Time" por "Log Timestamp", na página de "Log", do dialog de Extras. (a opção "Log Date & Time" será utilizada por padrão pelo driver) - Ajustes gerais na página "Network", do dialog de Extras. - Ajustes gerais na página "Modbus", do dialog de Extras. - Revisão geral de todo o código fonte.
		F. Englert	- Adicionada opção de função "Halt", para tratamento de erros da comunicação. (Case 4429)
		A. Quites	- Adicionado o parâmetro de tentativas internas do driver, na página "Modbus", em "Extras". (Case 3365)
v1.00			- Todas as publicações anteriores ao controle de revisões.