

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELÉTRICA
CURSO DE ENGENHARIA DE CONTROLE E AUTOMAÇÃO

GABRIEL PERES CABRINI

**ESTUDO E IMPLEMENTAÇÃO DE UM SISTEMA DIGITAL DE CONTROLE
DISTRIBUÍDO UTILIZANDO OS PROTOCOLOS MODBUS E PROFIBUS**

TRABALHO DE CONCLUSÃO DE CURSO

CORNÉLIO PROCÓPIO
2019

GABRIEL PERES CABRINI

**ESTUDO E IMPLEMENTAÇÃO DE UM SISTEMA DIGITAL DE CONTROLE
DISTRIBUÍDO UTILIZANDO PROTOCOLOS MODBUS E PROFIBUS**

Trabalho de Conclusão de Curso de graduação, apresentada ao Curso Superior de Engenharia de Controle e Automação da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para a obtenção do título de Bacharel.

Orientador: Prof. Dr. Wagner Endo

CORNÉLIO PROCÓPIO
2019



Universidade Tecnológica Federal do Paraná
Campus Cornélio Procópio
Departamento Acadêmico de Elétrica
Curso de Engenharia de Controle e Automação



FOLHA DE APROVAÇÃO

Gabriel Peres Cabrini

Estudo e implementação de um sistema digital de controle distribuído utilizando os protocolos modbus e profibus

Trabalho de conclusão de curso apresentado às 15:30hs do dia 25/06/2019 como requisito parcial para a obtenção do título de Engenheiro de Controle e Automação no programa de Graduação em Engenharia de Controle e Automação da Universidade Tecnológica Federal do Paraná. O candidato foi arguido pela Banca Avaliadora composta pelos professores abaixo assinados. Após deliberação, a Banca Avaliadora considerou o trabalho aprovado.

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

Prof(a). Dr(a). Emerson Ravazzi Pires da Silva - (Membro)

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

AGRADECIMENTOS

Primeiramente agradeço à Deus, por me conceder sabedoria e me guiar durante toda a minha caminhada terrestre.

À minha mãe Angélica F. I. Peres e meu padrasto José Gustavo Lazaretti, pelo constante incentivo e dedicação, sempre me ensinando os reais valores do estudo e da vida.

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

Aos Professores Dr. Cristiano Marcos Agulhari e Dr. Emerson Ravazzi Pires da Silva pelas imprescindíveis dicas de correção à proposta inicial deste trabalho.

Aos colegas do curso de graduação em Engenharia de Controle e Automação e todos aqueles que colaboraram com o desenvolvimento deste trabalho.

RESUMO

CABRINI, G. Peres. **Estudo e implementação de um sistema digital de controle distribuído utilizando protocolos modbus e profibus**. 60 f. Trabalho de Conclusão de Curso (Graduação) – Engenharia de Controle e Automação. Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2019.

Este trabalho apresenta um projeto e estudo de um sistema digital de controle distribuído, e sua implementação integrando um CLP definido como mestre em uma rede de comunicação com protocolo Profibus e uma bancada responsável por simular um sistema digital de controle distribuído. Essa integração entre ambos é realizada por um kit desenvolvido com unidades remotas, que utilizam um protocolo de comunicação Modbus. Este kit é responsável também pela interconexão de dois protocolos de comunicação, realizando a comunicação de ambos através de um *gateway*.

Palavras-chave: Redes industriais. Protocolos industriais. Sistema digital de controle distribuído. Modbus. Profibus. Unidade terminal remota.

ABSTRACT

CABRINI, G. Peres. **Study and implementation of a digital distributed control system using modbus and profibus protocols.** 60 f. Trabalho de Conclusão de Curso (Graduação) – Engenharia de Controle e Automação. Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2019.

This work presents a project and study of a digital distributed control system, and its implementation integrating a CLP defined as master in a communication network with Profibus protocol and a bench developed to simulate a distributed digital control system. This integration between the two is carried out by a kit developed with remote units, which use a Modbus communication protocol. This kit is also responsible for the interconnection of two communication protocols, performing the communication of both through a gateway.

Keywords: Industrial networks. Industrial protocols. Distributed digital control system. Modbus. Profibus. Remote terminal unit.

LISTA DE FIGURAS

Figura 01 - Cenário das redes industriais e seus protocolos.....	14
Figura 02 - Sistema SCADA realizando a aquisição de dados de CLPs.....	19
Figura 03 - DDC – Controle Digital Direto.....	20
Figura 04 - Visão geral de um SDCD.....	21
Figura 05 - Esquemático proposto para o trabalho.....	27
Figura 06 - Kit didático composto por unidades remotas.....	29
Figura 07 - Configuração do DigiGate Profibus.....	31
Figura 08 - Leitura dos quatro componentes e seus respectivos endereços.....	31
Figura 09 - Desenho da tela do supervisor implementado no ScadaBR.....	32
Figura 10 - Supervisor implementado no ScadaBR com suas representações gráficas.....	33
Figura 11 - Criação do data source no ScadaBR.....	33
Figura 12 - Criação dos data points no ScadaBR.....	34
Figura 13 - Simulação do CLP no software Elipse Modbus Simulator.....	35
Figura 14 - Simulação do CLP para motores, bombas e valores de set points.....	35
Figura 15 - Tela do supervisor ScadaBR após as alterações realizadas.....	36

LISTA DE SIGLAS

CIM	<i>Computer Integrated Manufacturing</i>
CLP	Controlador Lógico Programável
DDC	<i>Direct Digital Control</i>
HDLC	<i>High Level Data Link Control</i>
LAN	<i>Local Area Network</i>
RTU	<i>Remote Terminal Unit</i>
SCADA	<i>Supervisory Control and Data Acquisition</i>
SDCD	Sistema Digital de Controle Distribuído
UTR	Unidade Terminal Remota

SUMÁRIO

1 INTRODUÇÃO	10
1.1 OBJETIVOS	11
1.1.1 Objetivo geral.....	11
1.1.2 Objetivos específicos	12
2 FUNDAMENTAÇÃO TEÓRICA	13
2.1 REDES DE COMUNICAÇÃO INDUSTRIAL	13
2.2 PROTOCOLOS DE COMUNICAÇÃO INDUSTRIAL	14
2.2.1 MODBUS	15
2.2.2 PROFIBUS	16
2.3 SISTEMAS SUPERVISÓRIOS	17
2.3.1 SCADA (<i>Supervisory Control and Data Acquisition</i>)	18
2.4 SISTEMA DIGITAL DE CONTROLE DISTRIBUÍDO.....	19
3 ESTUDO DE UM SISTEMA INTEGRADO DE AUTOMAÇÃO INDUSTRIAL	23
3.1 UNIDADES DE ACESSO REMOTO	23
3.2 SDCD (Sistema Digital de Controle Distribuído)	24
3.3 INTEGRAÇÃO DE SISTEMAS DE AUTOMAÇÃO NA RELAÇÃO MESTRE- ESCRAVO	25
3.4 CIM (<i>Computer Integrated Manufacturing</i>).....	25
4 PROPOSTA DO TRABALHO.....	27
5 RESULTADOS E DISCUSSÕES.....	29
5.1 CONFIGURAÇÃO DAS UNIDADES REMOTAS	29
5.2 ELABORAÇÃO DA TELA DO SUPERVISÓRIO (SCADABR), IDENTIFICAÇÃO E CRIAÇÃO DOS DATAS POINTS	32
5.3 COMUNICAÇÃO ENTRE O SUPERVISÓRIO SCADABR E O PROTOCOLO DE COMUNICAÇÃO MODBUS	34
6 CONSIDERAÇÕES FINAIS	37

REFERÊNCIAS.....	38
APÊNDICE A – TUTORIAL DE INSTALAÇÃO DO SOFTWARE SCADABR	40
APÊNDICE B – TUTORIAL PARA CRIAÇÃO DO DATA SOURCE E DATA POINTS NO SCADABR.....	43
APÊNDICE C – TUTORIAL PARA CRIAÇÃO DE UMA REPRESENTAÇÃO GRÁFICA NO SCADABR.....	48
APÊNDICE D – INSTALAÇÃO DO SOFTWARE DIGICONFIG.....	53
APÊNDICE E – CONFIGURAÇÃO DAS UNIDADES REMOTAS DA MARCA NOVUS UTILIZANDO O SOFTWARE DIGICONFIG	55

1 INTRODUÇÃO

A automação envolve a implantação de sistemas conectados via redes de comunicação, englobando sistemas supervisórios e interfaces homem-máquina (IHM) que podem auxiliar os operadores na supervisão e na análise prévia dos problemas que podem vir a ocorrer. Quando se trata de sistemas que promovem informatização, os controladores lógicos programáveis (CLPs) são de extrema importância para que a automação industrial seja uma realidade onipresente (MORAES; CASTRUCCI, 2007).

Cada vez mais é exigido o projeto de redes industriais que se comuniquem de uma maneira eficaz e responsável com todo o processo na automação industrial. Com a evolução da eletrônica e da engenharia de software, surgiram os principais fatores para o desenvolvimento amplo dos sistemas de automação distribuídos em redes de campo industriais, ou então barramentos de campo (LUGLI; SANTOS, 2012). Componentes com alto desempenho, como microprocessadores, microcontroladores e sensores, bem como sistemas de supervisão cada vez mais eficazes contribuíram bastante para a integração da área computacional com os sistemas de automação industrial (LUGLI; SANTOS, 2010).

Com uma tendência global na utilização de sistemas baseados em dispositivos mais eficientes, torna-se necessária a comunicação entre estes dispositivos por meio de uma rede digital, onde tal comunicação torna possível a troca de recursos, como banco de dados e determinação de tarefas. Paralelamente ao crescimento do uso das LAN (*Local Area Network*) em ambientes de redes corporativas, foram desenvolvidas redes industriais, sendo estes barramentos de campo ou *Fieldbuses*. Pode-se dizer que o *Fieldbus* é a LAN para ambientes industriais (ALBUQUERQUE, 2009).

As redes industriais surgiram para dar maior flexibilidade ao processo de controle industrial, permitindo expansões futuras, e tornando-o mais acessível se comparado ao sistema centralizado, ou seja, com CLP. Estas possuem muitas vantagens sobre o sistema convencional, principalmente na questão do meio físico, o que resulta em um baixo custo de implantação e manutenção, pois exige menor quantidade de material, menor mão de obra, menor número de dispositivos, maior capacidade de I/Os, maior flexibilidade para expansões futuras e maior interconectividade (LUGLI; SANTOS, 2010).

Quando se fala em supervisão, é presente a ideia de controlar, inspecionar ou orientar um processo em um nível hierárquico superior. Em um sistema supervisório, é possível realizar operações como abrir ou fechar válvulas, acionar ou desligar bombas, ou seja, escrever na memória do CLP. Logo, as redes industriais permitem a troca de informações entre diferentes equipamentos computadorizados no ambiente industrial. Sua arquitetura faz com que as informações sejam transmitidas desde o chão de fábrica até o nível gerencial do processo automatizado (LUGLI; SANTOS, 2005).

Porém, para realização de tais integrações no processo industrial, existe uma preocupação na padronização dos protocolos de comunicação para os CLPs, a fim de proporcionar a comunicação entre equipamentos de diversos fabricantes, sendo aplicado aos sistemas supervisórios, controladores de processos e redes internas de comunicação (CARDOSO; ZANAROTTI, 2005).

Dessa forma, este trabalho propõe desenvolver uma integração entre um kit didático representado por um sistema de unidades remotas com uma bancada que simula um sistema digital de controle distribuído via inversores de frequência, e este processo será controlado via CLP, sendo supervisionado via interface gráfica através de um software supervisório ScadaBR. O desenvolvimento do kit é baseado em sistemas de supervisão e módulos de comunicação industrial Modbus e Profibus. É resultado da necessidade de criar uma aplicação que simule situações reais ao ambiente industrial, auxiliando os estudos em relação às redes industriais e sistemas supervisórios, assim como a integração e montagem.

1.1 OBJETIVOS

1.1.1 Objetivo geral

Este trabalho tem como objetivo estudar e implementar um sistema digital de controle distribuído utilizando os protocolos Modbus e Profibus, fazendo uso de um kit didático desenvolvido previamente a fim de comunicar uma bancada simulando um sistema digital de controle distribuído com um CLP, com apoio do software ScadaBR para supervisionar o sistema.

1.1.2 Objetivos específicos

Para atingir o objetivo geral, propõem-se os seguintes objetivos específicos:

- Estudar os equipamentos disponíveis e seus respectivos protocolos de comunicação;
- Realizar a manutenção das unidades remotas da marca Novus presentes no kit didático;
- Testar a integração entre as unidades remotas a fim de evitar possíveis impasses na implementação;
- Testar a comunicação entre os protocolos Modbus e Profibus pelo *gateway* da Novus;
- Configurar os endereços das unidades remotas previamente, realizar a leitura de cada unidade individualmente como também de todas as unidades interligadas;
- Realizar a comunicação do protocolo de comunicação com o *software* ScadaBR, garantindo a supervisão de dados.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão abordados os fundamentos teóricos nos quais são baseados a proposta deste trabalho, sendo estes uma revisão sobre redes de comunicação industrial, seus respectivos protocolos, sistemas supervisórios e SCADA.

2.1 REDES DE COMUNICAÇÃO INDUSTRIAL

Entre todas as tecnologias associadas ao controle industrial, as redes de comunicação são as que sofreram maiores evoluções na última década, seguindo a tendência global de evolução das comunicações que se tem sentido, praticamente em todos os ramos, desde as telecomunicações móveis, à Internet, às comunicações sem fio, etc (BORGES, 2007).

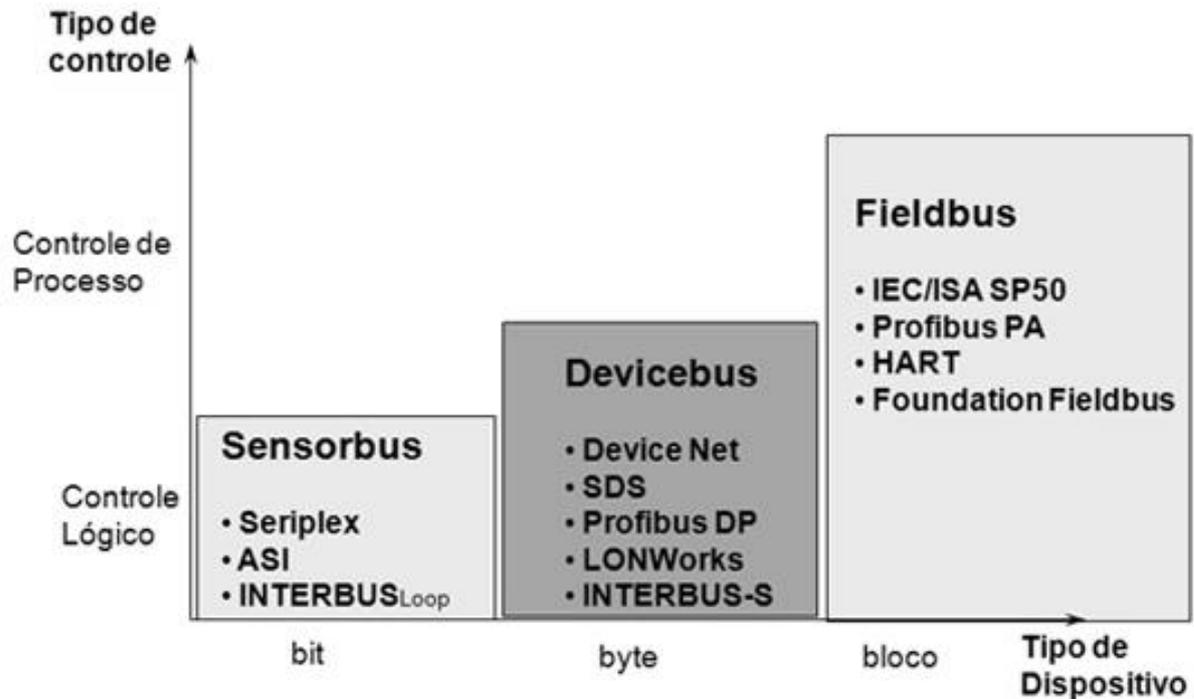
Com um expressivo destaque na automação da indústria, as redes digitais de instrumentação, ou barramentos de campo, tornam-se cada vez mais uma alternativa com grande aceitação. Depois da integração de microprocessadores a instrumentos de campo, surgiram os denominados instrumentos inteligentes aptos a realizarem a comunicação através de um barramento de campo, possibilitando que estes informem sua medição, a qualidade do sinal medido, além de outros possíveis diagnósticos (LIMA, 2004).

Com o advento dos barramentos de campo, a estratégia de controle centralizada pelo CLP pôde ser substituída por um controle descentralizado, favorecendo então que cada equipamento da rede pudesse assumir o papel de controlador. Sendo assim, vários controladores podem ser implementados na rede e até mesmo em instrumentos de campo, descentralizando a estrutura de controle (LIMA, 2004).

Tais barramentos de campo podem ser divididos em três tipos, sendo: rede de sensores ou *Sensorbus*, apropriadas para interligar sensores e atuadores discretos e que trabalham com *bits* de informação; rede de dispositivos ou *Devicebus*, apropriadas para integrar dispositivos como CLPs e outros dispositivos remotos de controle e aquisição de dados, operando com *bytes* de informação; e por fim rede de instrumentação ou *Fieldbus*, que realizam a integração entre instrumentos analógicos

de um ambiente industrial, como transmissores, válvulas de controle, etc., e trabalham com bloco de dados.

Figura 01 – Cenário das redes industriais e seus protocolos



Fonte: Adaptado de Lugli (2012, p. 27).

2.2 PROTOCOLOS DE COMUNICAÇÃO INDUSTRIAL

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

Os protocolos caracterizam os elementos de maior importância nas redes de automação industrial, tanto que normalmente estas passam a ser denominadas pelos respectivos protocolos utilizados, e seus protocolos definem o padrão operacional da rede de automação (MORAES; CASTRUCCI, 2007).

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

Para a aplicação proposta neste trabalho serão utilizados os protocolos MODBUS e PROFIBUS, que são protocolos empregados mundialmente no chão de fábrica.

2.2.1 MODBUS

O MODBUS é um protocolo para barramentos de campo criado pela MODICON, empresa fabricante de produtos para automação, visando o uso em seus próprios dispositivos. Porém, ao longo do tempo, este foi adotado por um grande número de fabricantes, autorizados pela empresa fabricante, passando então a ser um protocolo aberto (MORAES; CASTRUCCI, 2007).

Apesar de ser uma rede mais lenta do que outras redes já existentes, a rede MODBUS tem a vantagem de ser versátil e comunicar-se com inúmeros dispositivos dentre vários outros padrões de redes de comunicação utilizando apenas um par de cabo trançado (SOUZA, 2010).

Sua comunicação é baseada no modelo mestre-escravo, normalmente sendo CLPs os escravos e os sistemas supervisórios os mestres (LAGE, 2009), onde sua comunicação deve passar por um mestre, e os escravos não podem dialogar entre si. Neste modelo, o mestre pode solicitar resposta específica de um escravo e aguardar o retorno da informação, ou então pode enviar uma mensagem igual para todos os escravos. Este modelo de comunicação será explanado melhor na seção 3.3.

São divididos em três tipos, sendo MODBUS TCP/IP, MODBUS PLUS e MODBUS PADRÃO. O primeiro, é utilizado para comunicação entre sistemas de supervisão e CLPs, onde os dados são encapsulados no formato binário em quadros para serem transmitidos através de redes padrão Ethernet, com o mecanismo de controle e acesso CSMA-CD, cujas ações utilizam o modelo cliente-servidor. O segundo, é implementado para comunicar CLPs, chaves de partida eletrônica de motores, interfaces HM, etc., utilizando do RS-485, possuindo vários recursos adicionais de roteamento, diagnóstico, endereçamento e consistência, com o mecanismo de controle e acesso HDLC (*High Level Data Link Control*). O último, é utilizado para comunicar CLPs com dispositivos de entrada e saída de dados, IEDs como relés de proteção, atuadores de válvulas, transdutores de energia e etc.,

utilizando dos meios físicos RS-232 ou RS-485, conjuntamente com o modelo mestre-escravo. (MELO, 2005).

2.2.2 PROFIBUS

O PROFIBUS é um padrão aberto de rede de comunicação industrial, utilizado em um amplo leque de aplicações em automação da manufatura e de processos, possuindo total independência de fabricantes e possui sua padronização garantida pelas normas EN50170 e EN50250, permitindo assim que dispositivos de diferentes fabricantes possam comunicar-se entre si sem a necessidade de qualquer adaptação (LUGLI; SANTOS, 2010).

Possui um protocolo do tipo mestre-escravo, em que os controladores são descentralizados e estão conectados em rede no chão de fábrica. Os dispositivos intitulados como mestres são responsáveis por determinar a transmissão de dados. Já os dispositivos considerados escravos são dispositivos periféricos, incluindo dispositivos de entrada e saída, válvulas, sensores e transmissores de medidas, e não possuem direito de acesso direto ao barramento de campo, reconhecendo somente mensagens recebidas pelo mestre, ou retornar uma informação quando solicitados por estes (ALBUQUERQUE, 2009).

Este protocolo pode ser utilizado tanto em aplicações com transmissão de informações em alta velocidade como em tarefas complexas e extensas de comunicação, possuindo dois tipos de protocolos distintos, sendo PROFIBUS DP e PROFIBUS PA. Para tal aplicação, pode-se utilizar qualquer uma das três opções como meio de transmissão, sendo: RS485, Manchester ou fibra ótica (LUGLI; SANTOS, 2010). Os tipos do referido protocolo e suas características são apresentados a seguir.

Segundo Lugli e Santos (2010, p.68-69), o PROFIBUS DP, ou Periferia Descentralizada, “é o perfil mais frequentemente utilizado, otimizado para alta velocidade e conexão de baixo custo, projetado especialmente para a comunicação entre sistemas de controle de automação e seus respectivos I/Os”. Este pode ser empregado para substituir sistemas centralizados com CLPs em automação de manufatura, assim como para a transmissão de sinais de 4 a 20mA para automação de processos analógicos. Logo, esta é a versão otimizada para melhor desempenho, dedicado exclusivamente a aplicações com tempos críticos.

Segundo o mesmo autor citado acima, o PROFIBUS PA, ou Automação de Processo:

É o perfil menos frequentemente utilizado, desenvolvido para utilização apenas em sistemas de transmissão de sinais de 4 a 20mA ou HART para automação de processos analógicos. [...] sua grande vantagem em relação à versão DP é ter a possibilidade de configuração e parametrização integradas no instrumento de campo e a padrões de supervisão avançados do mercado, por exemplo FDT (*Field Decive Tool*) ou DTM (*Device Type Manager*). Ambas as tecnologias são conceitos de análise de falhas, diagnósticos e parâmetros avançados de uma rede de campo para instrumentos analógicos, como sensores de pressão ou temperatura. (2010, p.68-69).

2.3 SISTEMAS SUPERVISÓRIOS

Atividades de controle e supervisão de plantas industriais necessitam do uso de esquemas de automação sofisticados que devem garantir o acesso aos dados de produção, sendo estes através de grandes distâncias, pelos vários níveis da automação. Essas atividades podem ser realizadas pelo uso dos conhecidos sistemas supervisórios (BEJAN; IACOB; ANDREESCU, 2009).

Segundo Albuquerque (2009, p.213), “o termo controle supervisório denota o processo de monitorar à distância uma atividade, transmitindo diretrizes de operação aos controladores localizados à distância e recebendo de volta a indicação da realização das ações de controle”. Temos também, segundo Moraes e Castrucci (2007, p. 117) que “sistemas supervisórios são sistemas digitais de monitoração e operação da planta que gerenciam variáveis de processo, sendo estas atualizadas continuamente e podem ser guardadas em bancos de dados locais ou remotos para fins de registro histórico.

Este sistema é responsável pela comunicação do operador com todas as etapas do processo. Por meio deste o usuário pode acompanhar todo o processo industrial, podendo assim realizar eventuais intervenções. Através da utilização deste sistema, é possível gerenciar toda a operação das unidades de controle, de forma a obedecer intertravamentos hierárquicos de acordo com uma filosofia de controle

adequada, em função das vazões médias reais dos diversos segmentos do projeto e obedecendo a políticas de tomada de decisões (ALBUQUERQUE, 2009).

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

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

b) Flexibilidade: alterações no processo, correções ou implementações são facilmente realizáveis por meio de softwares, tanto que alguns permitem a alteração de telas sem interromper a operação normal do dispositivo que está sendo alterado, como se o sistema supervisório operasse nos dois modos simultaneamente.

c) Estrutura: toda planta de uma produção normalmente é dividida em áreas, logo é recomendável que a estrutura das telas do supervisório acompanhe essa divisão. Nas telas ocorre a exibição dos principais equipamentos e os instrumentos de medição e atuação do subsistema, permitindo sua visualização e controle. Visualizar nessa forma separada permite uma navegação direta e objetiva no processo, diminuindo o tempo de acesso às variáveis desejadas.

2.3.1 SCADA (*Supervisory Control and Data Acquisition*)

Supervisory Control and Data Acquisition, ou comumente chamado de SCADA, foi criado para supervisão e controle de quantidades elevadas de variáveis de entrada e saída digitais e analógicas distribuídas.

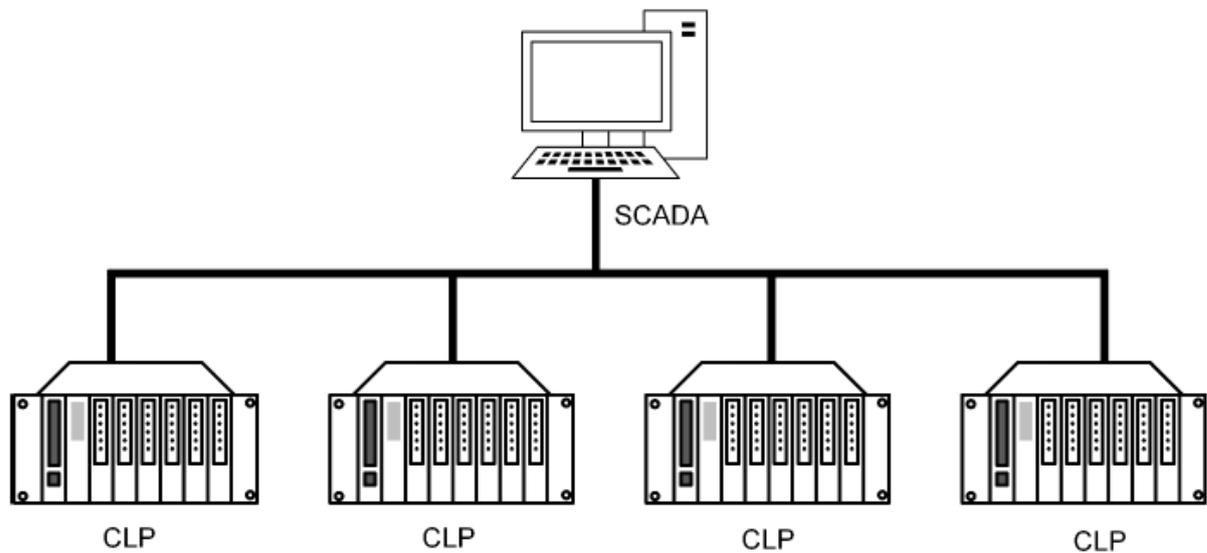
Estes sistemas de supervisão e controle são sistemas configuráveis, destinados à supervisão, ao controle e à aquisição de dados de plantas industriais, apresentando custo reduzido se comparado com os SDCCD (Sistemas Digitais de Controle Distribuído), sendo assim altamente populares nas indústrias (MORAES; CASTRUCCI, 2007)

Tal sistema possui funções de controle, monitoração e supervisão. A monitoração é a aquisição de dados para verificação das condições do sistema supervisionado, já a supervisão utiliza estes dados analisados para elaborar uma estratégia de operação. Sua arquitetura faz uso de processamento distribuído entre componentes de uma rede, não impedindo a existência de uma distância real entre esses componentes. Estão em constante coleta de dados, processando e

apresentando informações, possibilitando a entrada de comandos, parâmetros e *setpoints* do operador (ALBUQUERQUE, 2009).

Moraes e Castrucci (2007) afirmam que o dispositivo de controle que realiza comunicação com um sistema SCADA é denominado UTR (Unidade Terminal Remota), sendo geralmente um tipo de CLP. Na Figura 2 pode-se observar a maneira em que o sistema SCADA realiza a aquisição dos dados de 4 CLPs.

Figura 02 – Sistema SCADA realizando a aquisição de dados de CLPs

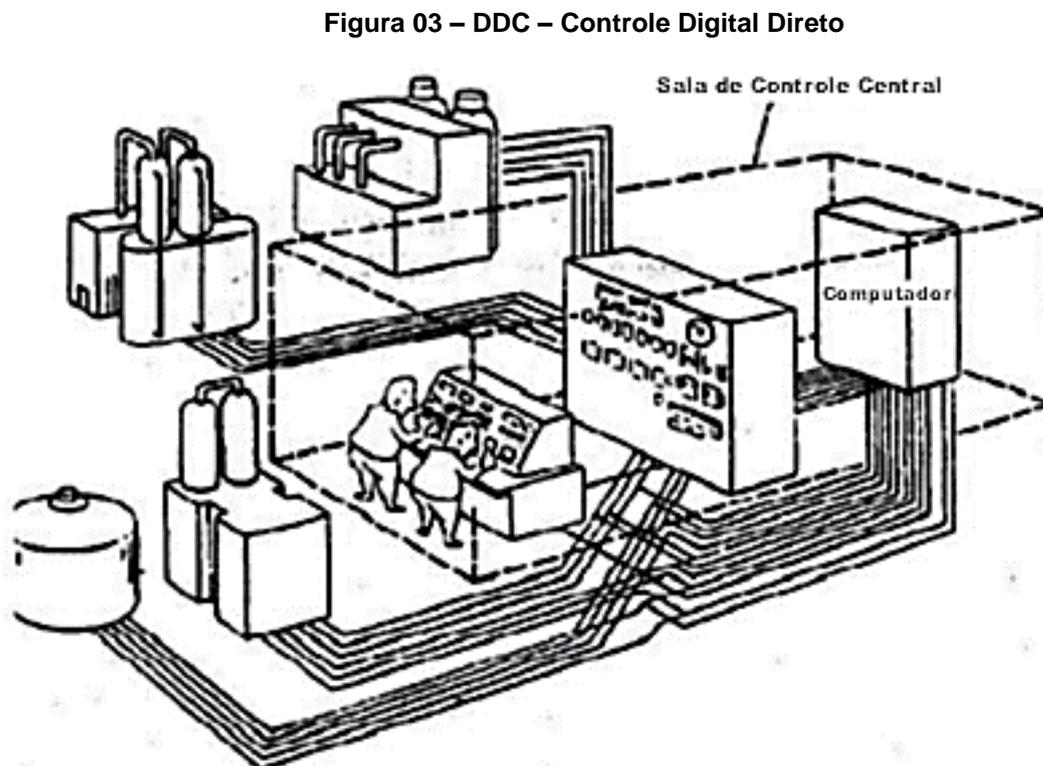


Fonte: Adaptado de Moras e Castrucci (2007, p. 121).

2.4 SISTEMA DIGITAL DE CONTROLE DISTRIBUÍDO

Até a década de 30, todos os instrumentos mecânicos eram conectados diretamente ao processo, sendo os operadores responsáveis pela monitoração de tais equipamentos. Na década seguinte, a instrumentação analógica pneumática foi um avanço devido a invenção do transmissor pneumático. Com tal foi possível retirar os controladores e registradores do ambiente de processo, dando origem ao conceito de sala de controle. Ali, os operadores podiam acompanhar o processo, registrando leituras, alterando *setpoints*, entre outros. Ao final da década de 50, com o advento dos transmissores e controladores eletrônicos analógicos, iniciou-se uma mudança do paradigma tecnológico, possibilitando a transmissão de sinais a distâncias consideráveis, permitindo o aumento na quantidade de informações obtidas do processo. Nos anos 60 iniciou-se o uso de computadores para o controle de

processos, dando origem ao conceito do *Direct Digital Control* (DDC) (Figura 3). Nesta época, o custo deste computador digital era muito elevado, fazendo com que seu uso fosse compartilhado, onde um único computador digital era usado para diferentes tarefas, operando centenas ou milhares de variáveis de processo. O uso do DDC auxiliou na eliminação dos indicadores, registradores e controladores de painel, tornando o controle de processo mais eficiente. Porém, esse avanço exigia profissionais especializados, com conhecimento em linguagem de programação, como também de cabeamento entre o campo e a sala de controle, pois cada equipamento do processo necessitava de um par de condutores (SILVA, 2004).



Fonte: RIBEIRO (2001, p. 166).

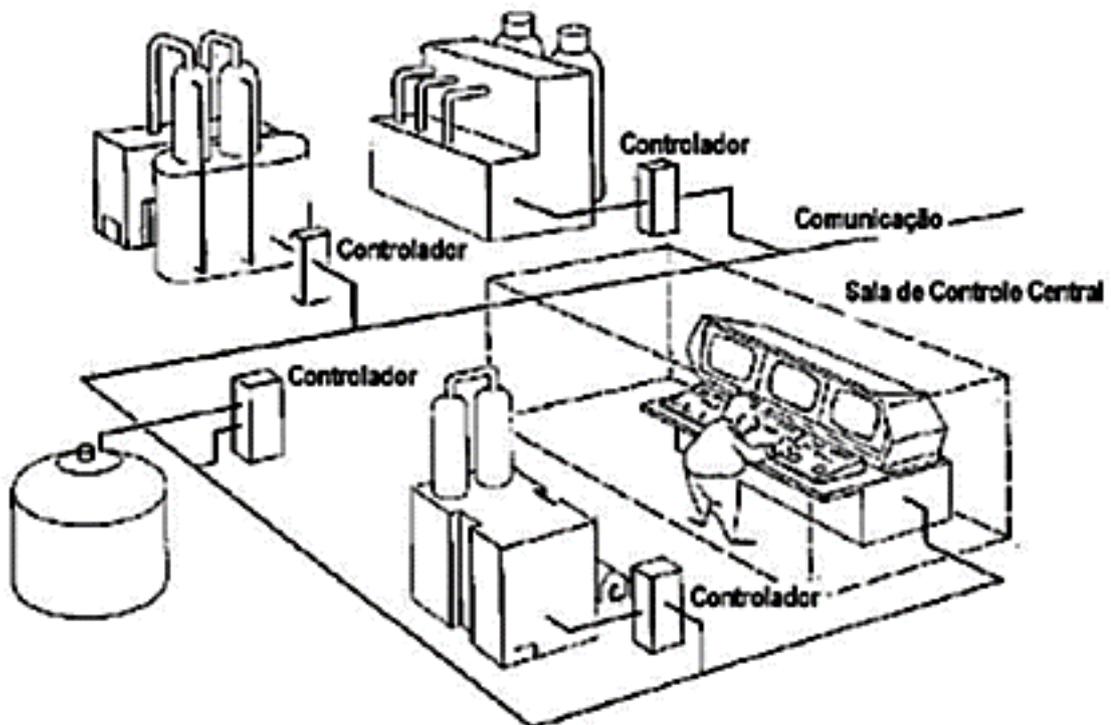
No início da década de 70, já eram presentes as redes de comunicação e surgiam os minicomputadores. Tais tecnologias foram essenciais para a origem de uma nova arquitetura de processos, denominada *Distributed Control System* (DCS), ou, em português, Sistema Digital de Controle Distribuído (SDCD). O processamento passou então a ser dividido entre computadores menores, ou os “controladores”. Estas estações de controle se comunicavam com uma sala de controle central, e caso houvesse alguma perda de comunicação, o processo não era interrompido.

O SDCD é composto por um pacote integrado de dispositivos que se auxiliam e se completam no cumprimento de suas principais funções, sendo estas funções o controle e a supervisão do processo industrial.

Para Finkel (2006), algumas das definições mais usuais de SDCD exibem que uma máquina é responsável pela ação de controle, enquanto outra é responsável pela IHM. O elemento de ligação entre os dois dispositivos microprocessados é a via de dados do sistema. Por essa definição, basta que os dois processadores sejam distintos para se ter um SDCD.

Este sistema é caracterizado como distribuído por ser dotado de redes redundantes, permitindo a descentralização do processamento de dados e decisões, através do uso de unidades remotas na planta. É através destas unidades remotas de processamento que os sinais dos equipamentos são processados, como condicionadores de sinais. Nestas arquiteturas, o controle é distribuído entre as estações remotas, sendo a estação central de controle um modo de facilitar a interface do sistema com o operador (PEREIRA, 2009).

Figura 04 – Visão geral de um SDCD



Fonte: RIBEIRO (2001, p. 171).

Tem-se então um sistema com arquitetura mista. De uma forma geral, as funções do SDCD podem ser estruturadas de maneira hierárquica, sendo definidos diversos níveis de atividades. Para Souza (1999), um SDCD pode ser dividido em subsistemas de acordo com suas características funcionais, como: subsistema de aquisição de dados, subsistema de monitoração e operação, subsistema de supervisão e otimização e subsistema de comunicação.

Já para Albuquerque (2009), os SDCD se caracterizam por equipamentos que fazem controle, monitoramento e supervisão de sistemas de automação industrial de grande porte, com capacidade de processamento elevada, alto número de entradas/saídas, IHM e um programa supervisor.

3 ESTUDO DE UM SISTEMA INTEGRADO DE AUTOMAÇÃO INDUSTRIAL

Neste capítulo será realizado o estudo de um sistema integrado de automação industrial, analisando sua estrutura, a fim de demonstrar como funciona este sistema integrado. Será feita uma revisão sobre condicionadores de sinais, sistemas digitais de controle distribuído e a relação mestre-escravo presente nos protocolos de comunicação e utilizado neste trabalho.

3.1 UNIDADES DE ACESSO REMOTO

Tendo em vista a necessidade de medições como temperatura, vibração, pressão, entre outras medidas realizadas por sensores, torna-se necessário o condicionamento do sinal para que o dispositivo responsável por tal aquisição seja capaz de efetuar a medição de forma eficaz e exata. Dentre estes condicionamentos, as principais tecnologias fornecem melhorias distintas no sinal, tanto no que diz respeito ao desempenho quanto à exatidão de tais sinais.

Logo, uma unidade de acesso remoto, ou então um condicionador de sinal, é um dispositivo que converte um tipo de sinal eletrônico em outro tipo de sinal, sendo esta a sua principal função, a fim de tornar mais fácil a leitura do sinal por instrumentação convencional. Existem vários tipos de condicionamento deste sinal, como amplificação, atenuação, isolação, filtragem, excitação, linearização, compensação de junção fria e configuração de ponte. Ambos métodos modificam o sinal inicial, a fim de facilitar a conversão deste sinal analógico para um sinal digital.

Dentre os variados tipos de unidades remotas, os fabricados pela Novus são amplamente utilizados e recomendados para este tipo de procedimento. Denominados como DigiRail, é uma linha de transmissores com interface de comunicação Modbus RTU, permitindo uma integração fácil entre sinais analógicos ou digitais aos sistemas de supervisão. Suas entradas são isoladas da alimentação e comunicação, a fim de evitar efeitos de interferência em sensores e equipamentos, eliminando a instabilidade nas medições.

São três tipos de unidades remotas, diferenciados pelos seus tipos de entrada ou saída, sendo eles:

a) Unidade remota com duas entradas analógicas universais, com sinais de entrada para termopares, Pt100, e outros sinais, como de 0 a 20 mV, 0 a 5 Vcc e 0 a 20 mA. Possui comunicação serial via RS485 a dois fios e protocolo Modbus RTU (DigiRail 2A);

b) Unidade remota com duas saídas a relé temporizados, com comunicação serial via RS485 a dois fios e protocolo Modbus RTU (DigiRail 2R);

c) Unidade remota com quatro entradas digitais contadoras, onde nível lógico 0 (baixo) corresponde ao intervalo de 0 a 1 Vcc e nível lógico 1 (alto) se refere ao intervalo de 4 a 35 Vcc. Possui também comunicação serial via RS485 a dois fios e protocolo Modbus RTU (DigiRail 4C).

3.2 SDCD (Sistema Digital de Controle Distribuído)

Um sistema digital de controle distribuído é caracterizado por uma integração de dispositivos, capazes de se auxiliarem diante do cumprimento de suas principais funções, como o controle e a supervisão do processo em nível industrial. Segundo algumas definições mais comuns, uma máquina é responsável pela ação de controle, enquanto outra máquina é responsável pela interface homem-máquina, ou supervisão pelo operador do processo. A ligação entre estes componentes é realizada via redes industriais e seus respectivos protocolos de comunicação, vistos previamente neste trabalho. Tal sistema é caracterizado como distribuído por ser dotado de redes industriais, o que favorece a descentralização do processamento de dados e a tomada de decisões necessárias no processo, através do uso das unidades remotas na planta. É por meio destas que o processamento dos sinais dos equipamentos é realizado, seguindo o conceito dos condicionadores de sinais.

Nestes SDCDs, o controle é distribuído entre as estações remotas do sistema, sendo a estação central de controle um modo de facilitar a interface entre o sistema e o operador. Como este sistema possui uma arquitetura considerada mista, pode-se estruturar seus níveis de atividades de uma forma hierárquica, considerando subsistemas do processo, como: aquisição de dados; monitoração e operação; supervisão e otimização; comunicação.

3.3 INTEGRAÇÃO DE SISTEMAS DE AUTOMAÇÃO NA RELAÇÃO MESTRE-ESCRAVO

Um sistema que utiliza uma comunicação com modelo mestre-escravo trabalha com requisição de dados e consequente resposta, visto que este modelo é baseado em estações ativas e passivas. O mestre, que são as estações ativas, é capaz de enviar mensagens independentemente de solicitações externas, requisitando algum dado ou apenas determinando alguma função para o referido escravo. Em barramentos extensos pode haver a presença de mais de um mestre, e para tal se utiliza um mecanismo de passagem de *token*, que funciona como um mecanismo de controle de acesso ao meio. As estações passivas, ou seja, os escravos, não possuem direito de acesso ao barramento, tendo como função apenas confirmar ao referido mestre que recebeu uma mensagem ou então responder a uma solicitação do mesmo.

Sendo assim, a principal função do mestre é controlar a rede de comunicação e concentrar os dados do sistema, operando como uma interface de operação remota e interface com sistemas de controle supervisórios. O escravo, por sua vez, possui a função de receber a informação do mestre e executá-la da melhor maneira possível, atuando em tarefas localizadas, podendo realizar processamento dos sinais, efetuar medidas e manipular eventos de forma pré-determinada (definidas pelo mestre). Existem também os escravos que não possuem processamento local, sendo estes considerados RTU (*Remote Terminal Unit*), ou então Unidades Terminais Remotas.

3.4 CIM (*Computer Integrated Manufacturing*)

Sistemas que se caracterizam pelo gerenciamento de processos de forma integrada, em modo geral, são designados pelo nome de Manufatura Integrada por Computador, ou então CIM. Tal sistema é responsável pela integração do projeto, da produção, distribuição e funções dentro de um sistema coerente suportado por uma rede de sistemas computacionais, formado basicamente por computadores, bancos de dados e CLPs.

No nível mais baixo desta estrutura, há pequenas quantidades de dados e muitos nós, exigindo protocolos com baixas taxas de transmissão. Nos níveis mais altos, os nós são menores, porém a quantidade de dados aumenta, exigindo redes cada vez mais velozes.

Na automação, o CIM vem como uma síntese da ultramecanização, racionalização (melhor integração a fim de obter menos desperdício, menos esforço, menos custo, etc.), processamento contínuo e controle automático. Tais sistemas CIM possibilitaram a interligação dos níveis de gerenciamento, controle e supervisão dos sistemas de automação de forma hierárquica, com o uso de algoritmos complexos, controle distribuído e centralização de decisões de grande escala, possibilitando o gerenciamento do processo tanto técnico quanto administrativo. Atualmente, a base de um CIM é formada por SDCCD, que representa praticamente todos os níveis de controle e execução do processo.

É relevante ressaltar também a importância dos softwares de aquisição de dados, supervisão e controle, ou SCADA, e das redes de comunicação com seus protocolos industriais (*fieldbus*), pois são essenciais para o funcionamento do CIM e do SDCCD.

4 PROPOSTA DO TRABALHO

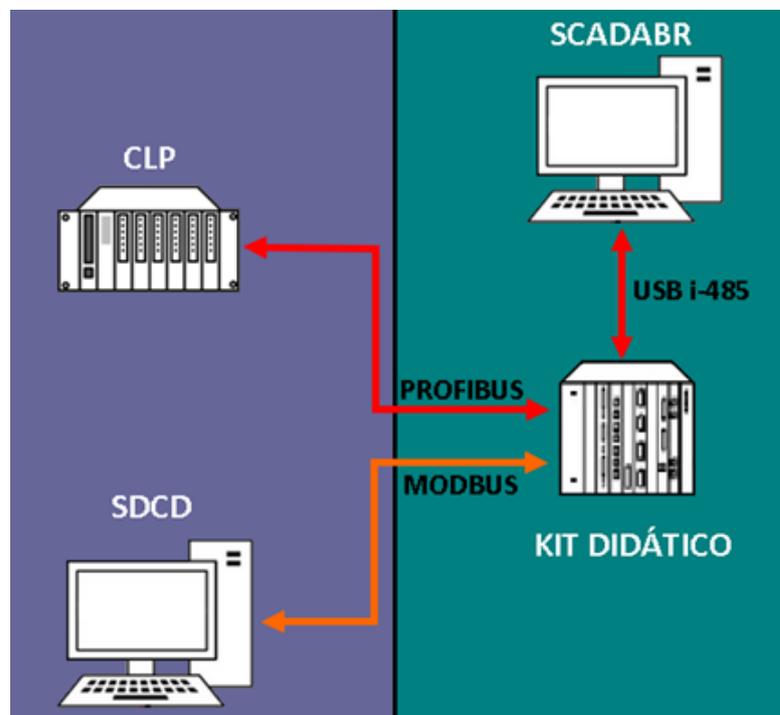
A proposta deste trabalho consiste no estudo de um sistema digital de controle distribuído, a fim de compreender seu funcionamento na automação industrial, como também implementá-lo em um processo real.

Inicialmente é realizada a manutenção das unidades remotas presentes no kit didático, e então testada toda sua integração, conferindo o pleno funcionamento de ambos os módulos, e configurando via *software* os respectivos endereços para cada unidade remota.

Essa implementação é proposta entre um kit didático composto por unidades remotas da Novus com o protocolo de comunicação Modbus utilizando do software ScadaBR para supervisão e aquisição dos dados. Este kit didático foi desenvolvido baseado em sistemas de supervisão e módulos de comunicação industrial Modbus e Profibus (RONQUI, 2011).

Na Figura 05 pode-se observar o esquemático proposto para o trabalho, exemplificando as conexões presentes em tal.

Figura 05 – Esquemático proposto para o trabalho



Fonte: Autoria própria.

Esta integração é controlada por um CLP, configurado como mestre no protocolo Profibus, e o kit de unidades remotas como Profibus escravo, este composto por um *gateway*. A função deste equipamento é a interconexão entre uma rede Profibus DP e uma rede Modbus RTU. Assim, de acordo com sua pré-configuração, este componente lê os dados dos demais aparelhos (escravos) da rede Modbus e repassa estes valores lidos ao mestre Profibus, e da mesma maneira, ele escreve nas saídas dos escravos Modbus conforme a solicitação do mestre da rede Profibus, facilitando e proporcionando assim um total controle pela rede Profibus sobre os aparelhos da rede Modbus.

Os conteúdos abordados nesta proposta do trabalho estão relacionados com as disciplinas de redes industriais, instrumentação industrial e controle do curso de engenharia de controle e automação.

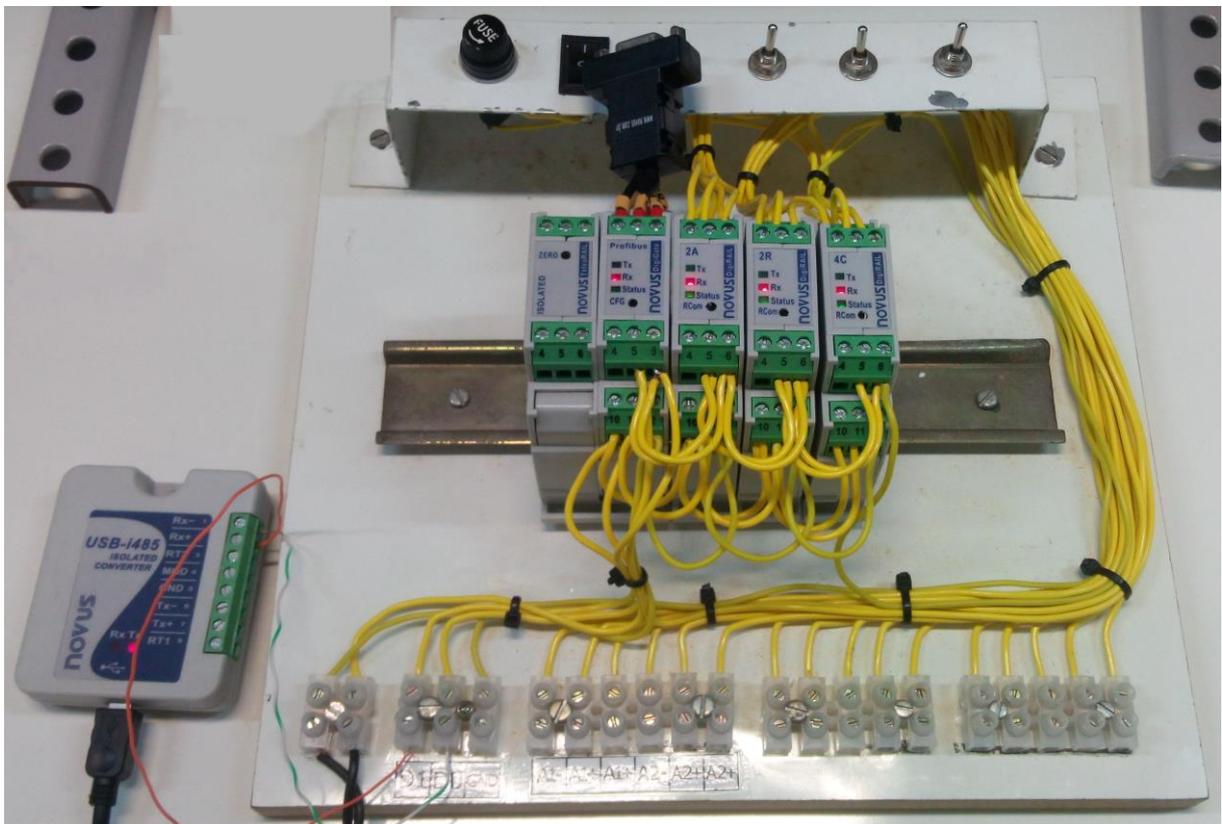
5 RESULTADOS E DISCUSSÕES

Neste capítulo serão apresentados os resultados obtidos com a configuração e a simulação das comunicações a fim de comprovar o êxito da proposta apresentada e agregar conhecimento a trabalhos futuros que podem surgir. Tais resultados são referentes à configuração das unidades remotas (da marca Novus), à comunicação do kit didático com o computador, à elaboração da tela do supervisor e à comunicação do protocolo Modbus.

5.1 CONFIGURAÇÃO DAS UNIDADES REMOTAS

Inicialmente, na Figura 06, pode-se observar o Kit composto de suas unidades remotas, explanadas na seção 3.1. Estas unidades remotas são da marca Novus.

Figura 06 – Kit didático composto por unidades remotas



Fonte: Autoria própria.

Para realizar a comunicação das unidades remotas, é necessário de um Conversor Digital USB-i485. Este conversor é responsável por realizar a interface entre um computador e barramentos de comunicação industrial RS485 ou RS422. Ao ligá-lo à porta USB de um computador, ele é detectado e automaticamente instalado como uma porta COM nativa, compatível com qualquer aplicativo de comunicação serial.

Portanto, para a configuração de cada unidade remota, isola-se sua comunicação, conectando seus terminais D1 e D0 aos terminais Rx+ e Rx-, respectivamente, do Conversor Digital USB-i485.

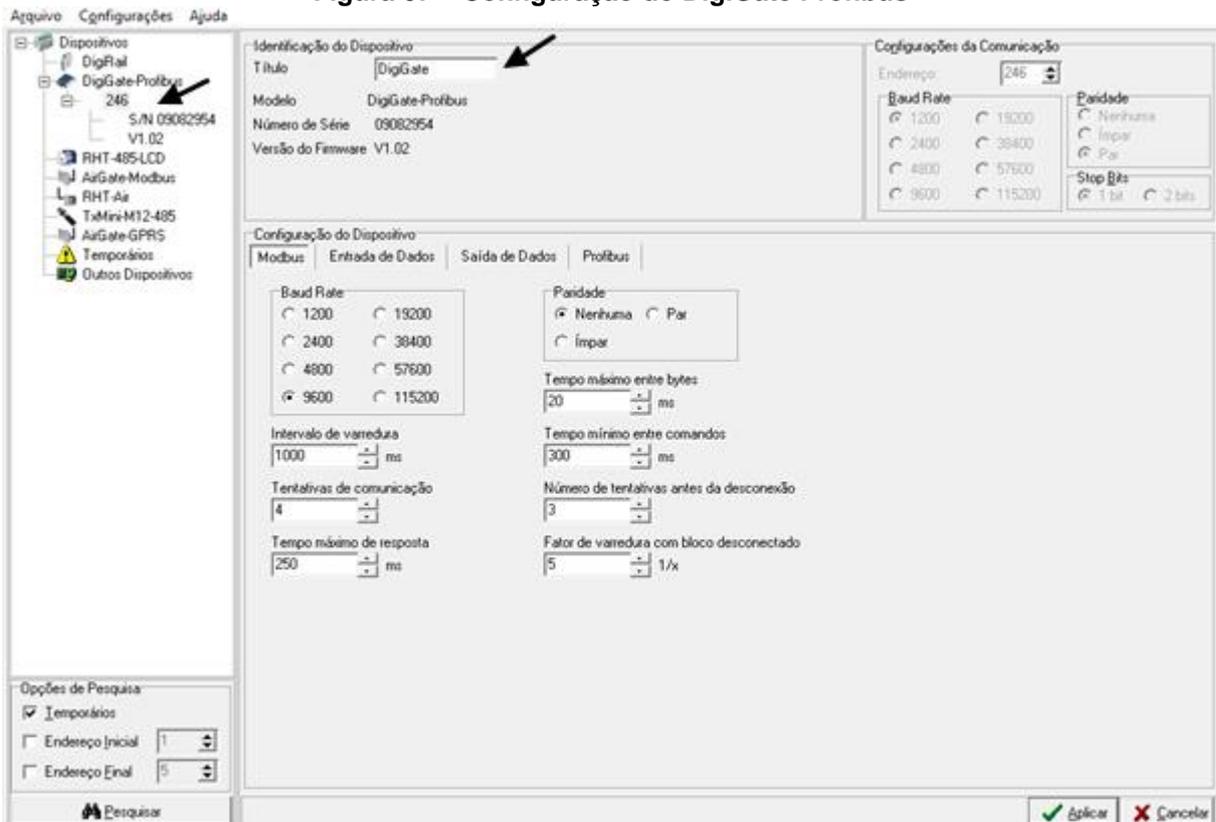
Este procedimento foi realizado para as três unidades remotas presentes no kit (DigiRail 2A, DigiRail 2R e DigiRail 4C), como também para o *gateway* (DigiGate Profibus).

Nesta configuração inicial, pôde-se determinar o endereço de cada unidade, para então ser possível fazer a leitura de todas as unidades remotas simultaneamente. O procedimento para estas configurações está descrito no Apêndice E deste trabalho. Na Figura 08, observa-se a leitura dos quatro componentes, com seus respectivos endereços já configurados para posterior utilização.

Para a configuração do DigiGate Profibus (Figura 07), o *gateway* responsável pela comunicação entre os dois protocolos de comunicação, é necessário alterar seu modo de funcionamento para um “modo Configuração”. Este procedimento é necessário para o aparelho deixar de se comportar como mestre na rede Modbus, e sim como escravo nesta rede, aceitando assim comandos do software de configuração, sendo este último o mestre.

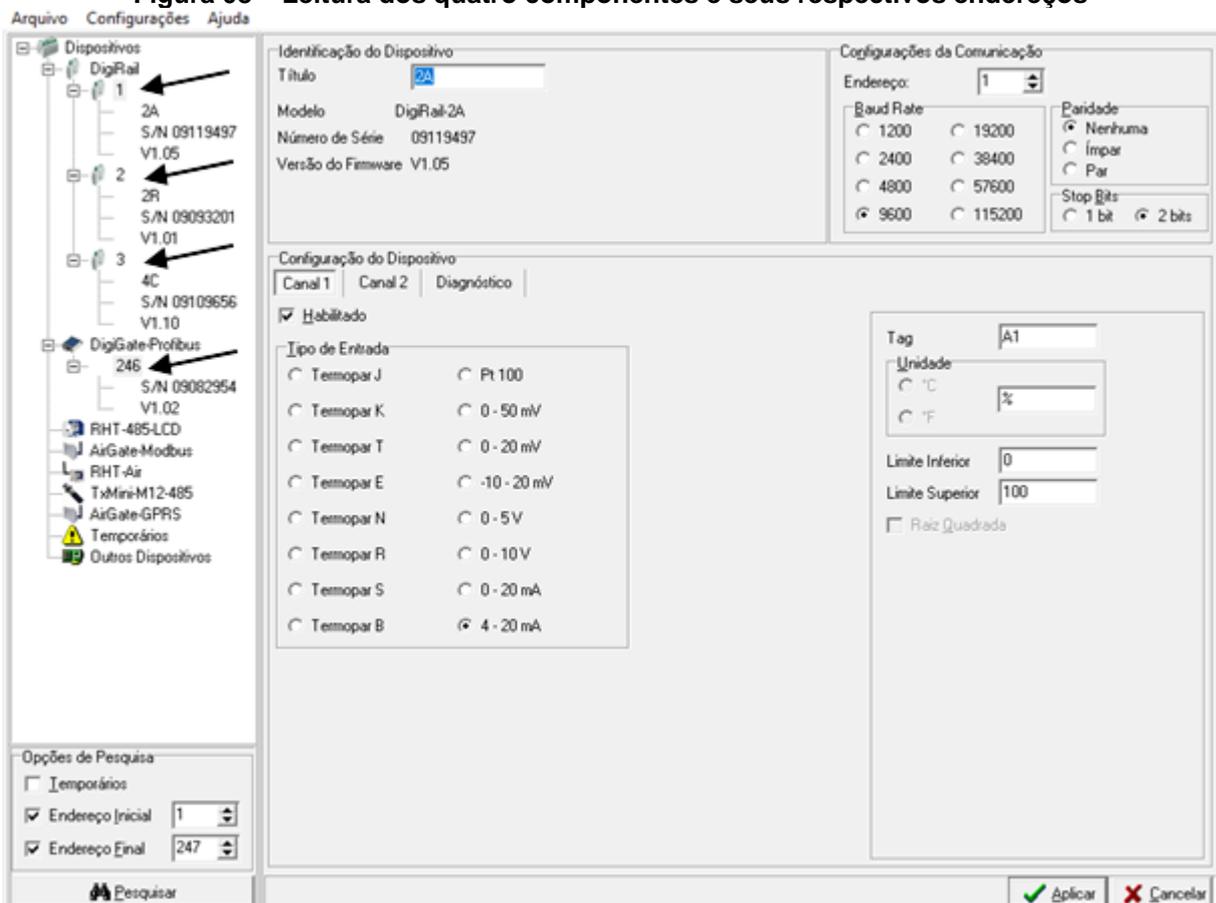
Para acessar este “modo Configuração”, deve-se pressionar a tecla CFG presente na unidade remota, observando que o led Status passa a piscar lentamente. Assim, foi determinado o título para a unidade remota, visto que seu endereço é fixo (246), sendo possível alterar seu *baud rate* e outras configurações relacionadas a varredura, tempo de resposta, etc. Na janela “Profibus”, é possível alterar o endereço Profibus, quando o diagnóstico revela o número de identificação, bloco de leitura de dados, bloco de escrita de dados e o *baud rate*.

Figura 07 – Configuração do DigiGate Profibus



Fonte: Autoria própria.

Figura 08 – Leitura dos quatro componentes e seus respectivos endereços



Fonte: Autoria própria.

5.2 ELABORAÇÃO DA TELA DO SUPERVISÓRIO (SCADABR), IDENTIFICAÇÃO E CRIAÇÃO DOS DATAS POINTS

Com o objetivo de demonstrar a comunicação no âmbito industrial, simulando um processo real com suas variáveis, foi elaborado uma tela para o supervisor ScadaBR. O desenho desta tela foi realizado utilizando a ferramenta Paint 3D. A tela representa um processo de mistura de corantes à um líquido qualquer.



Na Figura 09, o processo de mistura é operado pelas ações das bombas dos tanques e dos motores alimentadores e misturadores, além do alimentador geral, de modo a se ter o controle de acionamento e a velocidade de rotação de cada atuador.

Na Figura 10 foram evidenciados os botões de ON/OFF e a indicação de velocidade dos respectivos motores e bombas. Foram então definidos o *data source* utilizado para a comunicação entre o supervisor e o protocolo de comunicação Modbus, e os *data points* para a realização da comunicação com o equipamento (Figuras 11 e 12). Vale ressaltar que são sete motores e três bombas, totalizando 10 bobinas, e também 10 registradores.

Figura 10 – Supervisório implementado no ScadaBR com suas representações gráficas

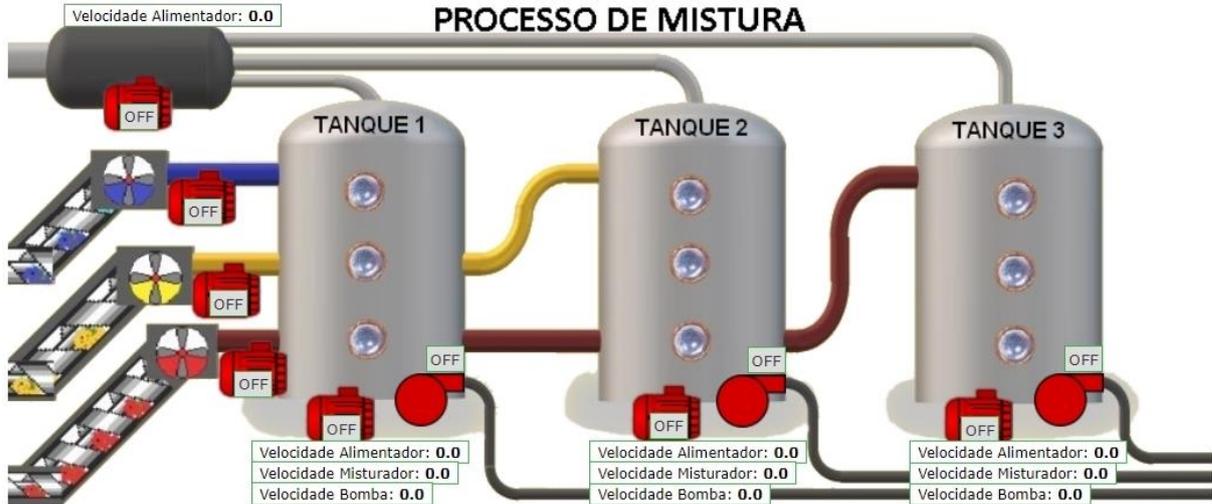


Figura 11 – Criação do *data source* no ScadaBR

Propriedades do modbus IP

Nome: Processo Mistura

Export ID (XID): DS_322790

Período de atualização: 1 segundo(s)

Quantificação:

Timeout (ms): 500

Tentativas: 2

Apenas quantidades contínuas:

Criar pontos de monitor de escravo:

Máxima contagem de leitura de bits: 2000

Máxima contagem de leitura de registradores: 125

Máxima contagem de escrita de registradores: 120

Tipo de transporte: TCP

Host: 192.168.100.91

Porta: 502

Encapsulado:

Níveis de alarme de eventos

Exceção de data source: Urgente

Exceção de leitura de data point: Urgente

Exceção de escrita em data point: Urgente

Fonte: Autoria própria.

Figura 12 – Criação dos *data points* no ScadaBR

Data points						
Nome	Tipo de dado	Status	Escravo	Faixa	Offset (baseado em 0)	
Alimentador Geral	Binário		1	Status do coil	9	
Alimentador1	Binário		1	Status do coil	0	
Alimentador2	Binário		1	Status do coil	1	
Alimentador3	Binário		1	Status do coil	2	
Bomba1	Binário		1	Status do coil	3	
Bomba2	Binário		1	Status do coil	4	
Bomba3	Binário		1	Status do coil	5	
Misturador1	Binário		1	Status do coil	6	
Misturador2	Binário		1	Status do coil	7	
Misturador3	Binário		1	Status do coil	8	
Vel_Alimentador1	Numérico		1	Registrador holding	0	
Vel_Alimentador2	Numérico		1	Registrador holding	1	
Vel_Alimentador3	Numérico		1	Registrador holding	2	
Vel_AlimentadorGeral	Numérico		1	Registrador holding	9	
Vel_Bomba1	Numérico		1	Registrador holding	3	
Vel_Bomba2	Numérico		1	Registrador holding	4	
Vel_Bomba3	Numérico		1	Registrador holding	5	
Vel_Misturador1	Numérico		1	Registrador holding	6	
Vel_Misturador2	Numérico		1	Registrador holding	7	
Vel_Misturador3	Numérico		1	Registrador holding	8	

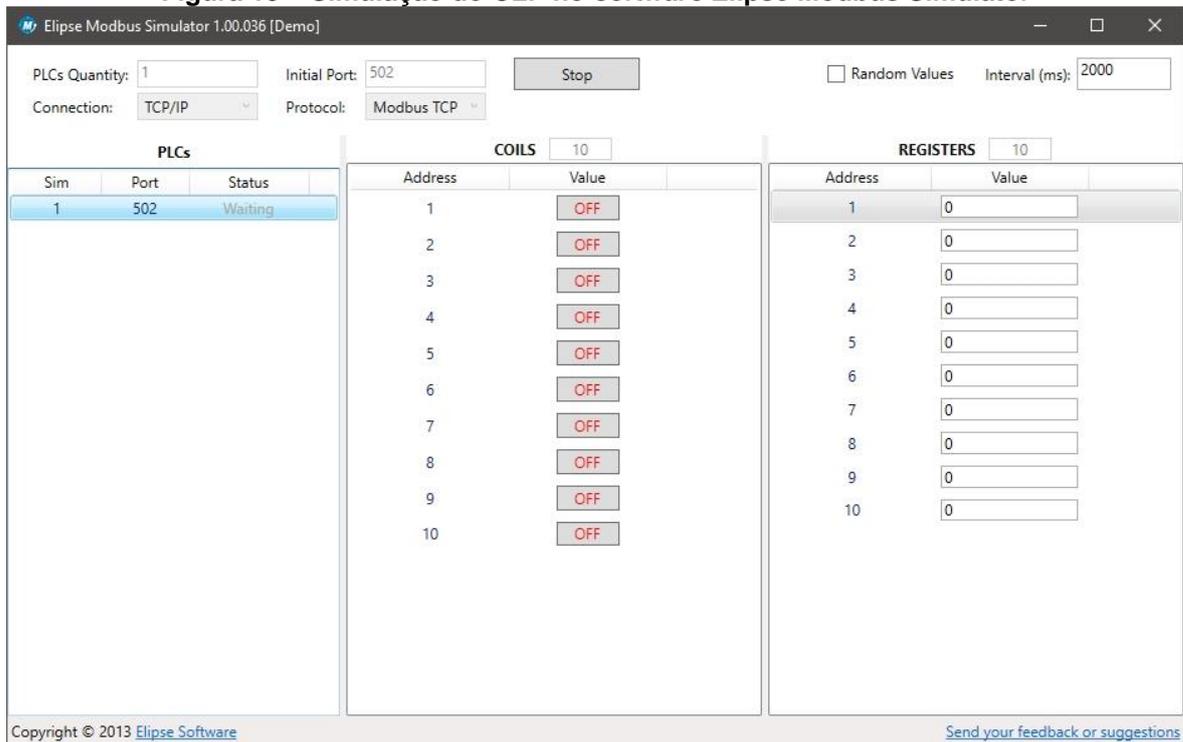
Fonte: Autoria própria.

5.3 COMUNICAÇÃO ENTRE O SUPERVISÓRIO SCADABR E O PROTOCOLO DE COMUNICAÇÃO MODBUS

Com o objetivo de simular o envio de dados do chão de fábrica para o sistema supervisório ScadaBR, foi utilizado um *software* que opera simulando um CLP. Este é o Eclipse Modbus Simulator, que para esta operação utiliza o protocolo de comunicação industrial Modbus (Figura 13).

No simulador foi definido a utilização de um CLP, com a função de acionamento para os seus atuadores (motores e bombas) presentes no processo, como também a definição de *set points* para as velocidades de rotação de cada um destes atuadores. Este CLP é composto por dez bobinas, responsáveis pelas ações dos motores de alimentação, de mistura, e para as bombas dos tanques, e dez registradores, responsáveis por definir a velocidade de rotação destes atuadores.

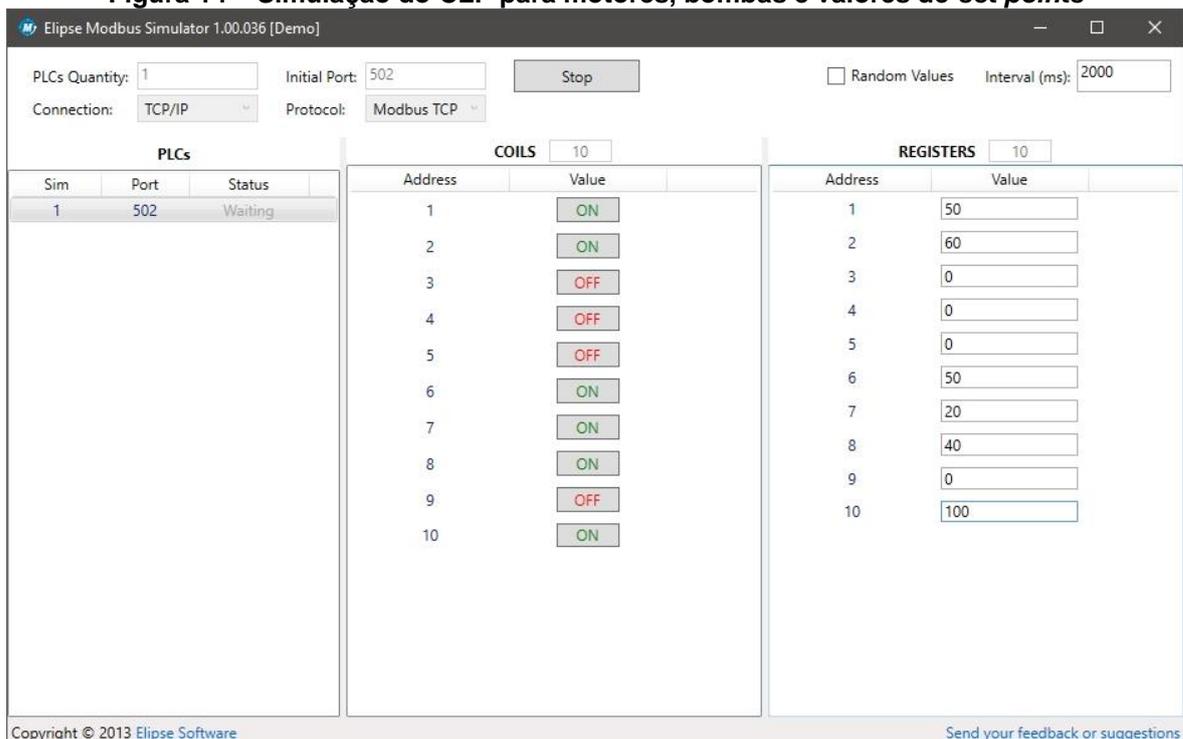
Figura 13 – Simulação do CLP no software Elipse Modbus Simulator



Fonte: Autoria própria.

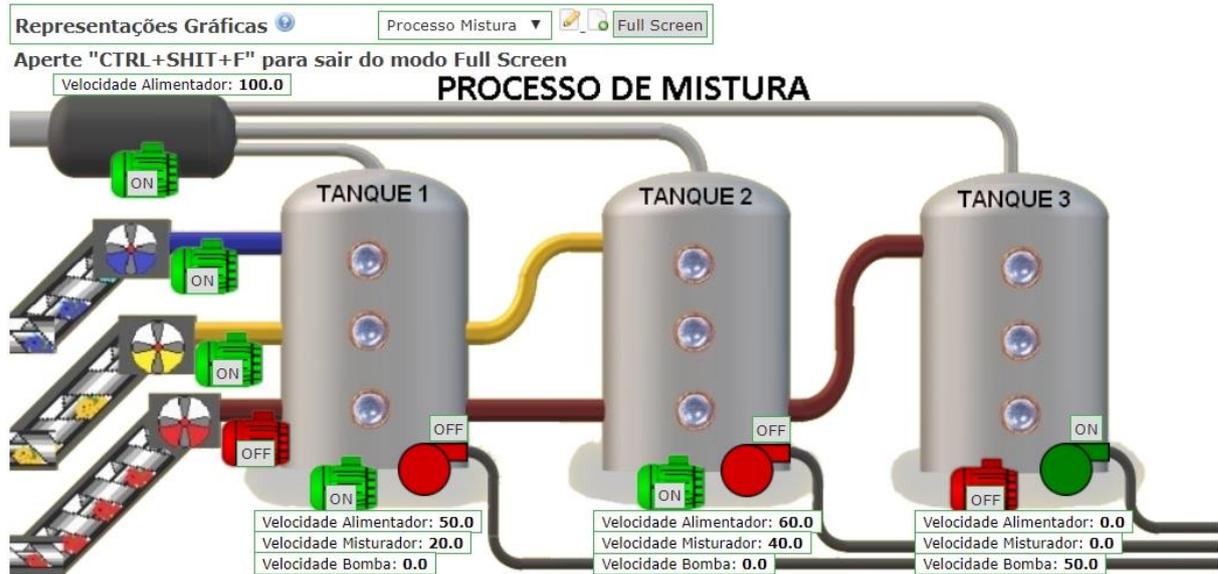
Podemos visualizar na Figura 14 que cada comando realizado no simulador é refletido na tela do sistema supervisor da Figura 15, representando então a comunicação entre o protocolo de comunicação industrial Modbus e o supervisor ScadaBR.

Figura 14 – Simulação do CLP para motores, bombas e valores de *set points*



Fonte: Autoria própria.

Figura 15 – Tela do supervisor ScadaBR após as alterações realizadas



Fonte: Autoria própria.

Observando os resultados obtidos com a simulação e a comunicação, podemos afirmar que as aplicações para sistemas de supervisão no *software* ScadaBR são amplas, pois opera com diferentes tipos de comunicação e protocolos de comunicação e tem o diferencial de ser um servidor web, com seu acesso a partir de um navegador de internet padrão.

6 CONSIDERAÇÕES FINAIS

Neste trabalho foi apresentado o estudo de um sistema digital de controle distribuído utilizando dos protocolos de comunicação Modbus e Profibus. Foi fundamentado teoricamente toda a base deste trabalho, como seus protocolos de comunicação, os sistemas supervisórios, e também foi realizado o estudo de um sistema integrado de automação industrial, no qual faz parte os condicionadores de sinais, ou unidades remotas, os sistemas digitais de controle distribuído, e a relação mestre-escravo presente atualmente nos protocolos de comunicação industriais.

O foco principal deste trabalho foi garantir a comunicação do supervisório ScadaBR com o protocolo de comunicação Modbus, simulando a comunicação com um CLP. Esta comunicação se mostrou eficaz e confiável, através dos testes realizados, além de ser totalmente editável com suas representações gráficas, tornando o sistema supervisório mais didático e objetivo, auxiliando na supervisão dos dados de equipamentos do chão de fábrica. Foi realizada também a manutenção e configuração das unidades remotas da Novus e a comprovação de seu funcionamento e integração.

Diante dos resultados obtidos, tanto com a configuração das unidades remotas da Novus, quanto com a simulação da comunicação industrial com o protocolo de comunicação Modbus, comunicando-o com o sistema supervisório ScadaBR, permite-se concluir que o objetivo geral foi alcançado.

Este trabalho é uma contribuição excelente aos alunos de Engenharia de Controle e Automação, pois permite o acesso ao conhecimento sobre os protocolos de comunicação industrial, que são amplamente implementados nas indústrias, ou seja, aproxima os futuros profissionais às necessidades da indústria, além de trazer novos conceitos sobre os sistemas supervisórios, como o ScadaBR, que é um *software* livre, gratuito e de código-fonte aberto.

Futuramente, sugere-se a integração deste trabalho com um CLP, realizando a supervisão e aquisição dos dados por um acesso remoto, já que o supervisório ScadaBR permite esta ação, operando num navegador de internet comum.

REFERÊNCIAS

ALBUQUERQUE, Pedro U. B. de. **Redes industriais: aplicações em sistemas digitais de controle distribuído protocolos industriais, aplicações SCADA**. 2. ed. São Paulo: Ensino Profissional, 2009.

BORGES, Fátima. **Redes de comunicação industrial**. Documento técnico n. 2. 2007. Disponível em: <http://www.schneiderelectric.pt/documents/product-services/training/doctecnico_redes.pdf> Acesso em 13 maio 2017.

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

CARDOSO, M. C. F.; ZANAROTTI, G. S. CLP – **Evolução e tendências. Mecatrônica Atual – Automação Industrial de Processos e Manufatura**. Ano 4, n. 25, p. 17-21, 2005.

FINKEL, Vitor Schmidt. **Instrumentação Industrial**. Rio de Janeiro: Editora Interciência, 2006.

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

LIMA, Fábio Soares de. **Estratégia de Escalonamento de Controladores PID Baseado em Regras Fuzzy para Redes Industriais Foundation Fieldbus Usando Blocos Padrões**. 2004. 68p. Dissertação de Mestrado. Universidade Federal do Rio Grande do Norte – UFRN – Natal – RN, 2004.

LUGLI, Alexandre Baratella; SANTOS, Max Mauro Dias. **Redes industriais para automação industrial: AS-I, PROFIBUS e PROFINET**. 1. ed. São Paulo, SP: Érica, 2010.

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

LUGLI, Alexandre B.; SANTOS, Max Mauro Dias. **Redes industriais para automação industrial: AS-I, PROFIBUS e PROFINET**. 2. ed. São Paulo, SP: Érica, 2012.

MELO, Walmy. **Visão geral dos Protocolos Modbus**. 2005. Disponível em: <<http://docslide.com.br/documents/visao-geral-dos-protocolos-modbus.html>> Acesso em 28 abr. 2017.

MORAES, Cícero Couto de; CASTRUCCI, Plínio de Lauro. **Engenharia de Automação Industrial**. 2. ed. Rio de Janeiro: LTC, 2007.

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

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

RONQUI, Larissa Agnes Pereira dos Santos. **Kit para sistemas supervisórios e rede industrial de comunicação baseado no protocolo modbus**. 2011. 69 f. Trabalho de Conclusão de Curso (Tecnólogo) – Curso de Tecnologia em Automação Industrial, Universidade Tecnológica Federal do Paraná, Cornélio Procópio-PR, 2011.

RIBEIRO, Marco Antonio. **Automação Industrial**. 4. ed, Salvador-BA, 2001.

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

SOUZA, Eduardo Luís da Silva. **Redes de comunicação industrial**. 2010. 49 f. Trabalho de Conclusão de Curso (Graduação) – Curso de Engenharia Elétrica, Universidade São Francisco, Campinas-SP, 2010.

SOUZA, F. M. **Automação Básica e Circuitos de Intertravamento e Alarmes**. Programa de Certificação do Pessoal de Manutenção – COM. Trabalho realizado em parceria SENAI/CST (Companhia Siderúrgica de Tubarão), Vitória, ES. p. 93-105, 1999.

APÊNDICE A – TUTORIAL DE INSTALAÇÃO DO SOFTWARE SCADABR

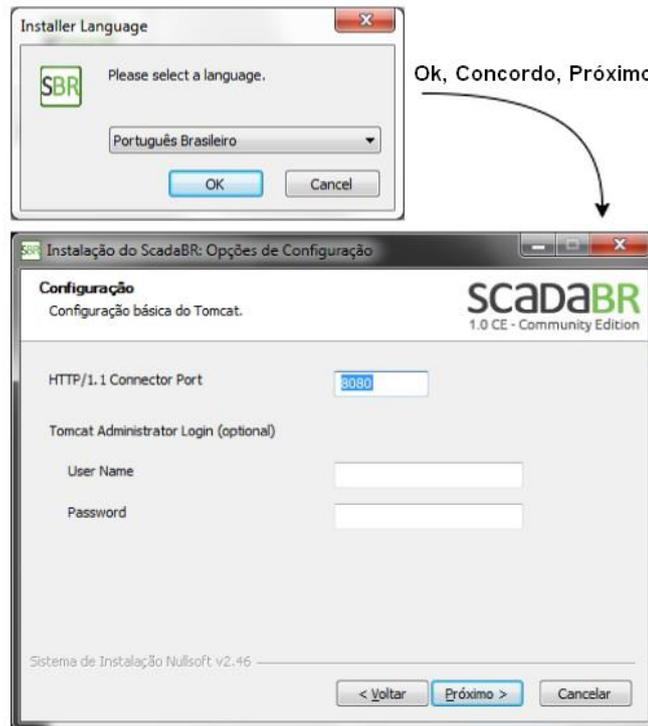
Neste apêndice será apresentado um tutorial de instalação do *software* ScadaBR, utilizado neste trabalho. Para sua instalação ser bem-sucedida, é necessário verificar as seguintes condições:

- Java 6.0: para que o software funcione corretamente é necessário instalar o Java 6.0. Outras versões podem não ser compatíveis. Caso possua alguma outra versão instalada, será necessário removê-la.
- Firewall: desative o Firewall do Windows.
- Porta 8080: abra o seu navegador padrão e digite <http://localhost:8080>. Caso não carregue a página significa que a porta está livre para instalação. Caso contrário, utilize outra porta (8081, 8082, etc).
- Remova arquivos existentes: caso já tenha tentado instalar o software anteriormente, será necessário remover todas as pastas criadas.

Passo 01 – realizar o download gratuito do *software* no site www.scadabr.com.br.

Passo 02 – ao abrir o instalador, basta clicar em OK e avançar as etapas. Este procedimento está demonstrado na Figura 16.

Figura 16 – Etapas para instalação do software ScadaBR



Fonte: Autoria própria.

Passo 03 – na configuração da máquina virtual Java, deve-se especificar o caminho da pasta JRE onde está instalado o Java 6.0. Obs.: o instalador normalmente realiza este procedimento automaticamente, porém pode ser necessário indicar este caminho manualmente.

Figura 17 – Configuração da máquina virtual Java



Fonte: Autoria própria.

Passo 04 – na tela do banco de dados é recomendada a utilização do Derby, visto que este já vem instalado, enquanto o MySQL necessita de instalação e configuração manual.

Figura 18 – Configuração do banco de dados

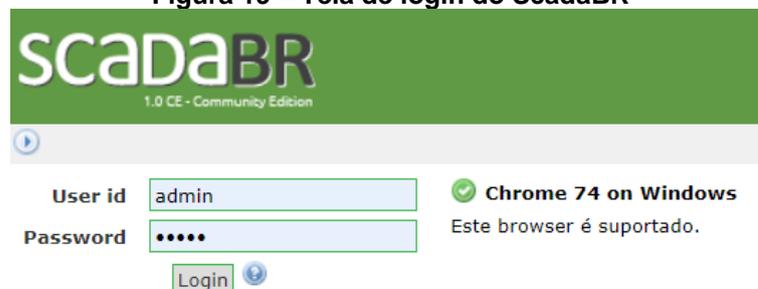


Fonte: Autoria própria.

Passo 05 – pronto, caso tenha seguido cada passo deste pequeno tutorial, o *software* ScadaBR já deve estar funcionando perfeitamente.

Passo 06 – para testá-lo, inicie o Tomcat como administrador do Windows, e em seguida abra o *browser* do navegador. Digite o endereço <http://localhost:8080/scadabr> e efetue o login no sistema preenchendo os campos usuário e senha com “admin”.

Figura 19 – Tela de login do ScadaBR



Fonte: Autoria própria.

APÊNDICE B – TUTORIAL PARA CRIAÇÃO DO *DATA SOURCE* E *DATA POINTS* NO SCADABR

Neste apêndice será apresentado um tutorial para a criação do *data source* e dos *data points* no supervisor ScadaBR. Estas são as “fontes” de informações, ou seja, é a configuração dos equipamentos que serão supervisionados.

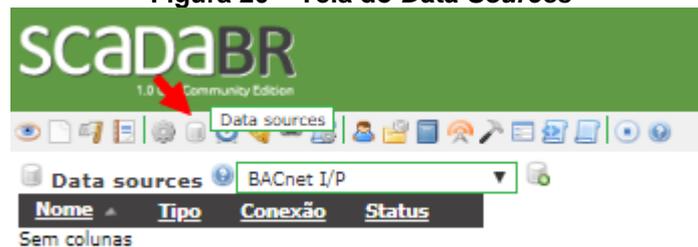
Um *data source* pode ser um barramento de comunicações usando um protocolo do tipo ASCII, Modbus, SQL, OPC, etc. Cada *data source* pode ter múltiplos *data points*, ou “tags”, que são os pontos de medição e controle.

Passo 01 – iniciar o ScadaBR no endereço <http://localhost:8080/scadabr>.

Passo 02 – efetuar o login inserindo usuário e senha “admin”.

Passo 03 – entrar na janela do *Data Sources*.

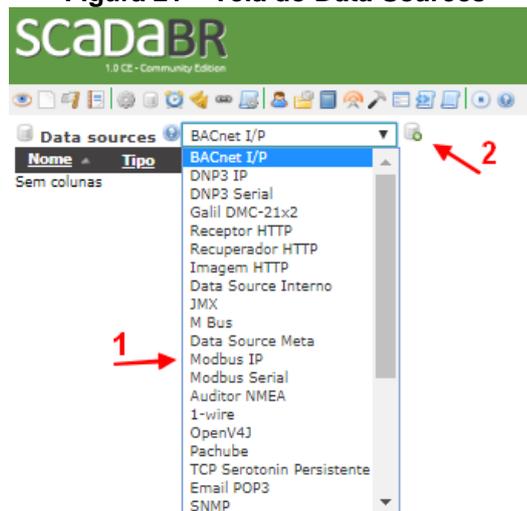
Figura 20 – Tela do *Data Sources*



Fonte: Autoria própria.

Passo 04 – selecionar o protocolo desejado e então clicar no botão “Adicionar”. Para dar sequência ao tutorial, foi escolhido o protocolo “Modbus IP”.

Figura 21 – Tela do *Data Sources*



Fonte: Autoria própria.

Passo 05 – nesta nova janela que se abrirá, faça as alterações indicadas na imagem a seguir, como “Nome”, “Período de atualização”, “Host” e “Porta”. Vale

ressaltar que em “Host”, deve ser inserido o IP do equipamento que esteja se comunicando com o protocolo Modbus. Neste caso, o IP utilizado é o do próprio computador, pois é onde o *software* de simulação do Modbus está em operação.

Figura 22 – Tela de criação do Data Source

The screenshot shows the 'Propriedades do modbus IP' configuration window in SCADA BR. The fields are as follows:

- Nome: [Empty]
- Export ID (XID): DS_877947
- Período de atualização: 5 minuto(s)
- Quantificação: [Checked]
- Timeout (ms): 500
- Tentativas: 2
- Apenas quantidades contínuas: [Unchecked]
- Criar pontos de monitor de escravo: [Checked]
- Máxima contagem de leitura de bits: 2000
- Máxima contagem de leitura de registradores: 125
- Máxima contagem de escrita de registradores: 120
- Tipo de transporte: TCP
- Host: [Empty]
- Porta: 502
- Encapsulado: [Unchecked]
- Níveis de alarme de eventos:
 - Exceção de data source: Urgente
 - Exceção de leitura de data point: Urgente
 - Exceção de escrita em data point: Urgente

Fonte: Autoria própria.

Figura 23 – Tela do simulador do CLP

The screenshot shows the Elipse Modbus Simulator 1.00.036 [Demo] interface. The configuration options are:

- PLCs Quantity: 1
- Initial Port: 502
- Connection: TCP/IP
- Protocol: Modbus TCP
- Random Values: [Unchecked]
- Interval (ms): 2000

The main display area is divided into three sections:

- PLCs:** A table with columns Sim, Port, and Status. It shows one PLC with Sim 1, Port 502, and Status Waiting.
- COILS:** A table with columns Address and Value. It shows three coils with addresses 1, 2, and 3, all with values OFF.
- REGISTERS:** A table with columns Address and Value. It shows three registers with addresses 1, 2, and 3, all with values 0.

Copyright © 2013 Elipse Software. Send your feedback or suggestions.

Fonte: Autoria própria.

Passo 06 – após feitas as alterações, clique em “Salvar”.

Figura 24 – Tela de configuração do *Data Source*

SCADA BR
1.0 CE - Community Edition

Propriedades do modbus IP

Nome: Tutorial

Export ID (XID): DS_877947

Período de atualização: 1 segundo(s)

Quantificação:

Timeout (ms): 500

Tentativas: 2

Apenas quantidades contínuas:

Criar pontos de monitor de escravo:

Máxima contagem de leitura de bits: 2000

Máxima contagem de leitura de registradores: 125

Máxima contagem de escrita de registradores: 120

Tipo de transporte: TCP

Host: 192.168.100.91

Porta: 502

Encapsulado:

Níveis de alarme de eventos

Exceção de data source: Urgente

Exceção de leitura de data point: Urgente

Exceção de escrita em data point: Urgente

Fonte: Autoria própria.

Passo 07 – ao salvar um novo *data source*, abaixo aparecerá os respectivos *data points* que serão criados.

No simulador, estamos utilizando 3 bobinas e 3 registradores, e portanto, devemos criar 6 *data points*.

Passo 08 – clique para criar um novo *data point*

Figura 25 – Tela de criação do *data point*

Data points

Nome	Tipo de dado	Status	Escravo	Faixa	Offset (baseado em 0)

Detalhes do data point

Nome:

Export ID (XID): DP_802830

Id do escravo: 1

Faixa do registro: Status do coil

Tipo de dados modbus: Binário

Offset (baseado em 0): 0

Bit: 0

Número de registradores: 0

Codificação de caracteres: ASCII

Configurável:

Multiplicador: 1

Aditivo: 0

Fonte: Autoria própria.

Nesta nova janela que aparecerá, vamos entender alguns pontos:

- Nome: nome que será dado à atual “tag” (*data point*).

- Id do escravo: se refere ao equipamento conectado pelo protocolo. Neste caso, temos apenas 1 CLP, então a ID será 1.
- Tipo de dados modbus: binário, inteiro, float, etc. Neste caso, utilizaremos “binário”, pois está associado à bobina, que assume valor “0 e 1” (OFF/ON ou false/true).
- Offset (baseado em 0): se refere à qual bobina (coil) será o *data point*. Neste caso, temos três bobinas, ou seja, o valor deste campo pode ser “0, 1 ou 2”.

Feito as alterações para as três bobinas, temos:

Figura 26 – Tela dos *data points* criados

Nome	Tipo de dado	Status	Escravo	Faixa	Offset (baseado em 0)
Bobina1	Binário		1	Status do coil 0	
Bobina2	Binário		1	Status do coil 1	
Bobina3	Binário		1	Status do coil 2	

Fonte: Autoria própria.

Passo 09 – criar os *data points* para os registradores

Figura 27 – Tela de criação dos *data points*

Detalhes do data point	
Nome	Registrador1
Export ID (XID)	DP_641321
Id do escravo	1
Faixa do registro	Registrador holding
Tipo de dados modbus	Inteiro de 2 bytes sem sinal
Offset (baseado em 0)	0
Bit	0
Número de registradores	0
Codificação de caracteres	ASCII
Configurável	<input checked="" type="checkbox"/>
Multiplicador	1
Aditivo	0

Fonte: Autoria própria.

Para os registradores, vale ressaltar alguns pontos:

- Id do escravo: igual para as bobinas, o valor continua 1 pois temos apenas um CLP sendo simulado.
- Tipo de dados modbus: agora, o registrador passa a obter valores inteiros, e não binários como anteriormente.
- Offset (baseado em 0): temos três registradores, ou seja, este valor será “0, 1 e 2”.

Feito as alterações para os três registradores, temos:

Figura 28 – Tela dos *data points* criados

Data points					
Nome	Tipo de dado	Status	Escravo	Faixa	Offset (baseado em 0)
Bobina1	Binário		1	Status do coil	0
Bobina2	Binário		1	Status do coil	1
Bobina3	Binário		1	Status do coil	2
Registrador1	Numérico		1	Registrador holding	0
Registrador2	Numérico		1	Registrador holding	1
Registrador3	Numérico		1	Registrador holding	2

Fonte: Autoria própria.

Passo 10 – habilitar todos os *data points*

Figura 29 – Tela dos *data points* habilitados

Data points					
Nome	Tipo de dado	Status	Escravo	Faixa	Offset (baseado em 0)
Bobina1	Binário		1	Status do coil	0
Bobina2	Binário		1	Status do coil	1
Bobina3	Binário		1	Status do coil	2
Registrador1	Numérico		1	Registrador holding	0
Registrador2	Numérico		1	Registrador holding	1
Registrador3	Numérico		1	Registrador holding	2

Fonte: Autoria própria.

Pronto, agora que já temos o *data source* e os *data points* criados e configurados, podemos criar uma representação gráfica para estes (Apêndice C).

APÊNDICE C – TUTORIAL PARA CRIAÇÃO DE UMA REPRESENTAÇÃO GRÁFICA NO SCADABR

Neste apêndice será apresentado um tutorial para criar uma representação gráfica no supervisor ScadaBR. Esta representação gráfica é onde você consegue criar seus próprios “painéis de controle”, podendo incluir monitoramento em “tempo-real” dos *data points*, gráficos, botões, etc.

Como visto anteriormente, temos um CLP simulado com três bobinas e três registradores, e já criamos seus respectivos *data points*. Vamos então supor que as três bobinas são responsáveis pelo acionamento de três motores, e que os três registradores são responsáveis por controlar a velocidade em porcentagem destes motores (0% até 100%).

Passo 01 - elaborar um fundo para sua representação gráfica. Neste caso, não utilizaremos um fundo, pois não é esta a intenção.

Passo 02 – clique em “Representação gráfica” e então em “Nova Representação”.

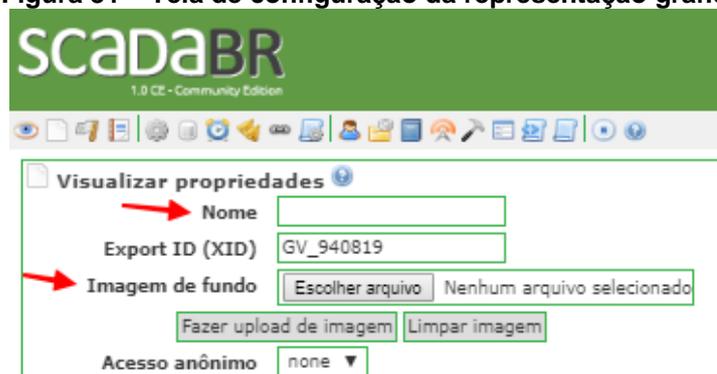
Figura 30 – Tela da Representação Gráfica



Fonte: Autoria própria.

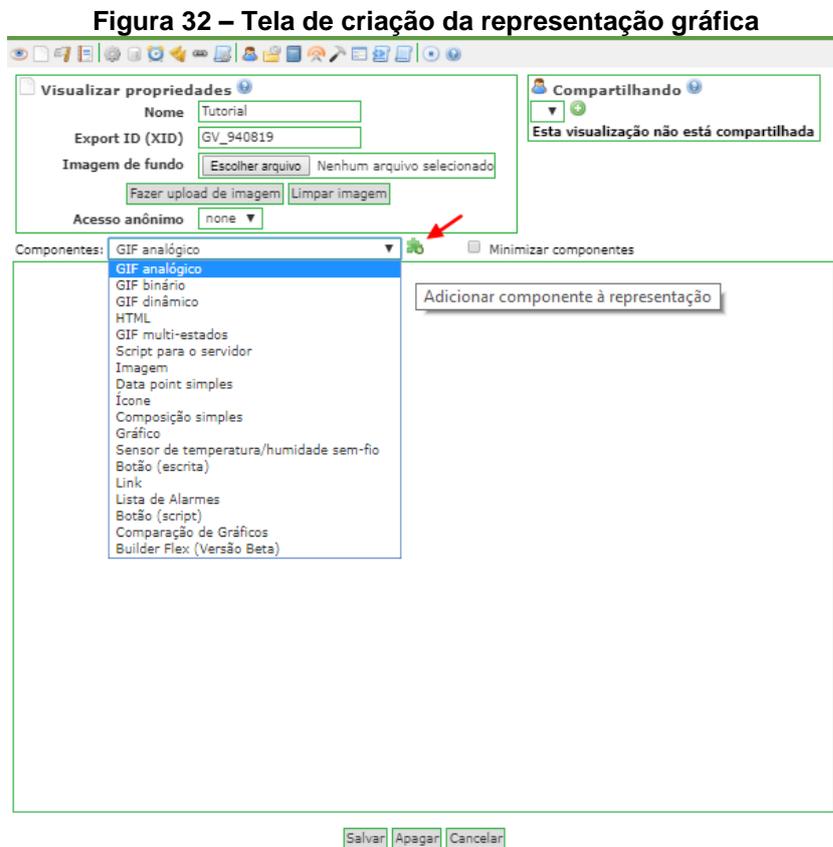
Passo 03 – dê um nome para esta tela, e faça o *upload* da imagem de fundo desejada.

Figura 31 – Tela de configuração da representação gráfica



Fonte: Autoria própria.

Passo 04 – selecione o componente que deseja adicionar e clique em “Adicionar componente à representação”.



Fonte: Autoria própria.

Neste caso, selecionaremos “GIF binário” para representar os três motores desejados.

Figura 33 – Detalhes de um GIF binário no ScadaBR



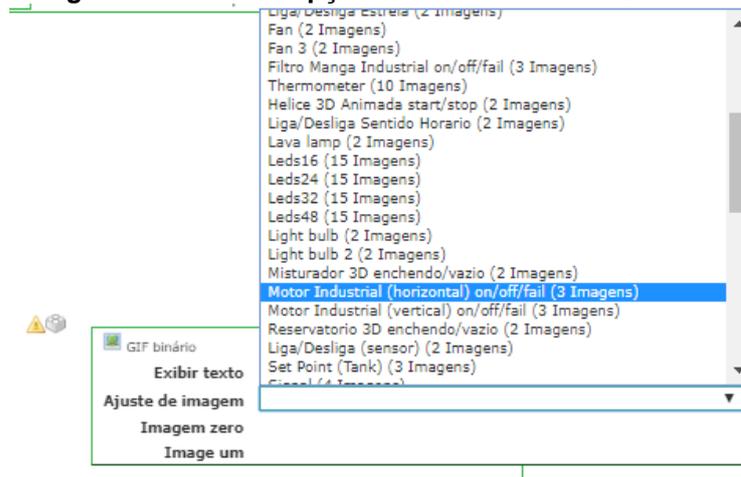
Fonte: Autoria própria.

Ao clicar em “Adicionar componente à representação”, nos deparamos com esta imagem.

- A opção 1 é responsável por “Editar configurações de componente de data point”.
- A opção 2 é responsável por “Editar processador gráfico”.
- A opção 3 é responsável por “Apagar componente de data point”.

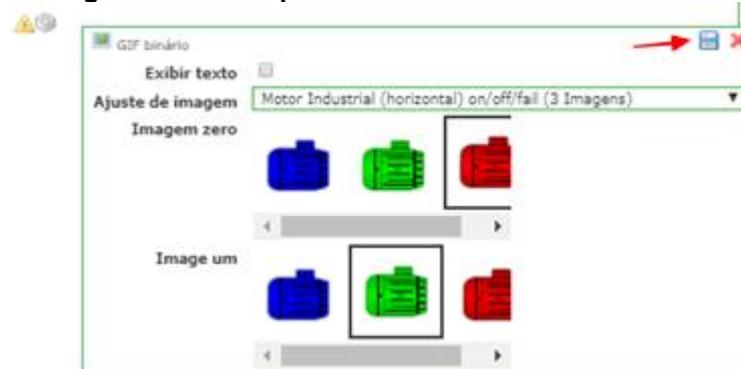
Passo 05 – escolher a representação gráfica que se assemelhe à um motor através da opção 2 acima. Neste caso, o supervisor ScadaBR possui diversas opções para escolha, auxiliando o usuário. Selecionamos então o “Motor Industrial”, e escolhemos qual imagem representará o motor “zero” (OFF) e o motor “um” (ON). Feito isto, clicamos em “Salvar”.

Figura 34 – Tela de opções de GIF binário no ScadaBR



Fonte: Autoria própria.

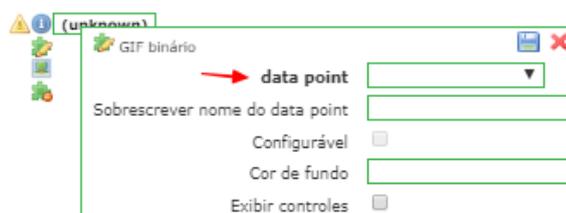
Figura 35 – Exemplo de um GIF binário no ScadaBR



Fonte: Autoria própria.

Passo 06 – identificar a qual *data point* se refere a representação gráfica criada. Para isto, clicamos na opção 1, abrindo uma nova janela.

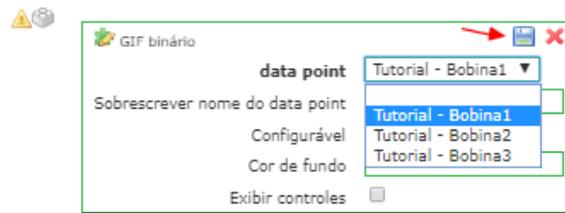
Figura 36 – Tela de configuração do GIF binário



Fonte: Autoria própria.

Nesta janela, clicamos em “data point” e selecionamos o *data point* referente à bobina um, e então clicamos em “Salvar”.

Figura 37 – Tela de configuração do GIF binário



Fonte: Autoria própria.

Portanto, a representação gráfica criada assumirá a imagem de um motor vermelho (desligado).

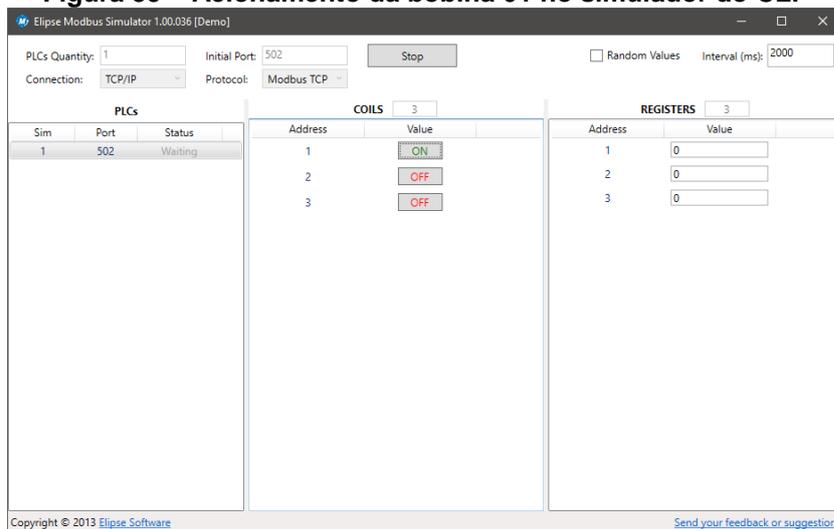
Figura 38 – Representação gráfica de um motor desligado



Fonte: Autoria própria.

Ao acionarmos esta bobina no simulador do CLP, ou seja, alterando seu valor para “ON” (true), a representação gráfica sofre uma alteração, apresentando a imagem de um motor verde (ligado).

Figura 39 – Acionamento da bobina 01 no simulador do CLP



Fonte: Autoria própria.

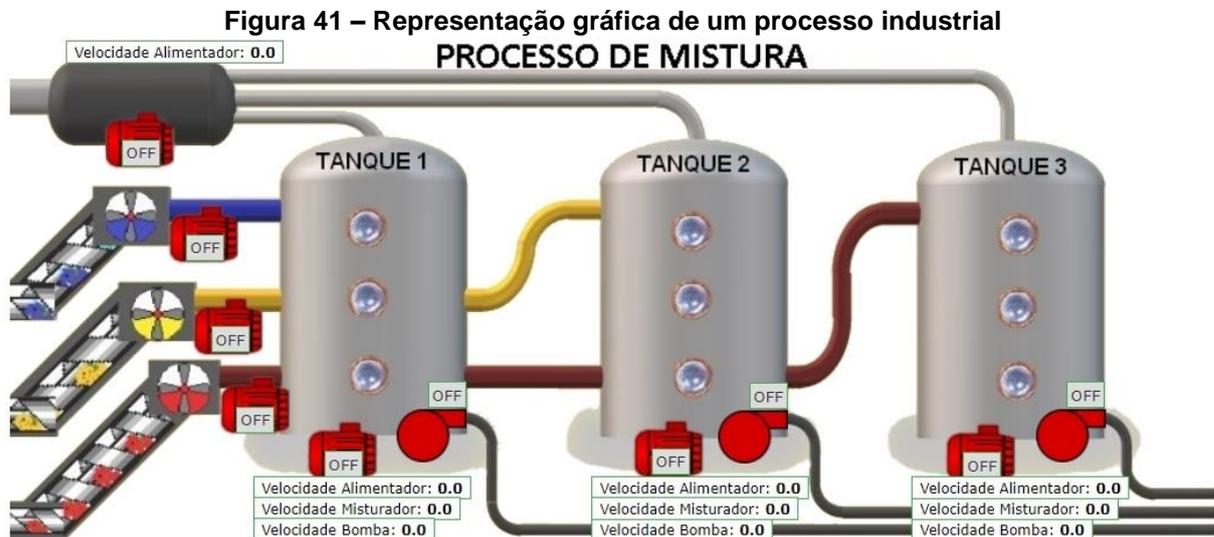
Figura 40 – Representação gráfica de um motor ligado



Fonte: Autoria própria.

Passo 07 – ao finalizar a criação da sua representação gráfica, o último procedimento a ser realizado é salvá-la, clicando no botão “Salvar” ao final da página.

Para ilustrar este tutorial, temos na Figura X a representação gráfica do sistema supervisor utilizado neste trabalho.



Fonte: Autoria própria.

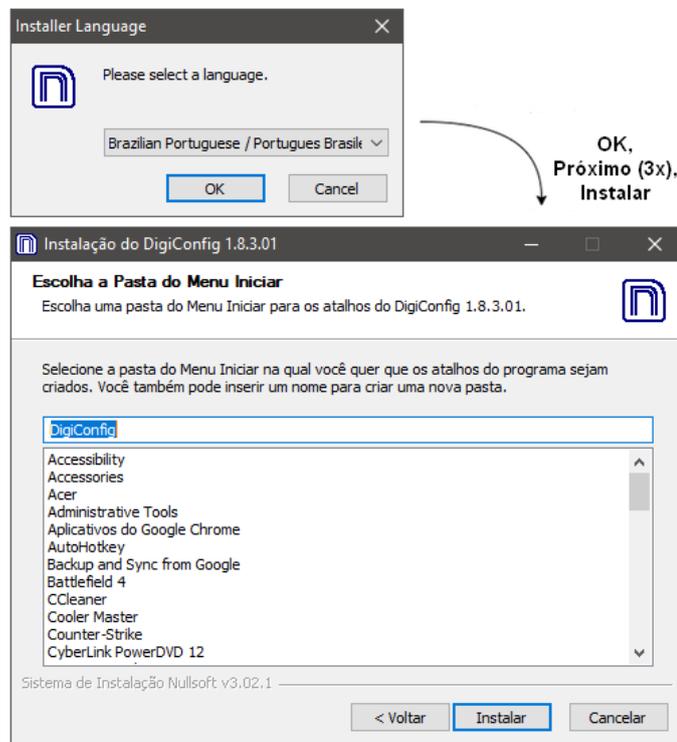
APÊNDICE D – INSTALAÇÃO DO SOFTWARE DIGICONFIG

Inicialmente, para a configuração das unidades remotas da marca Novus, há a necessidade da instalação de um *software*, o DigiConfig. O download deste software pode ser feito através do site da Novus, através do link: <http://www.novus.com.br/arquivos/636170>.

Passo 01 - para realizar sua instalação, basta abrir o arquivo executável disponibilizado pelo *download* no link acima.

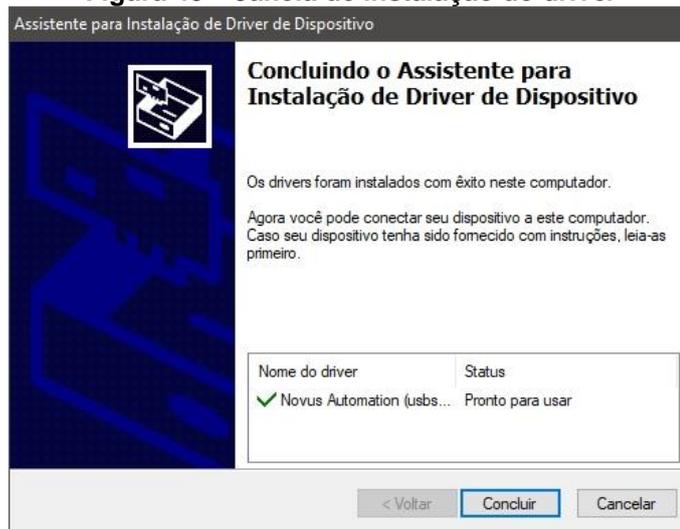
Passo 02 – na janela que se abrirá, basta clicar em “OK, próximo, próximo, próximo e instalar”.

Figura 42 – Etapas para a instalação do software DigiConfig



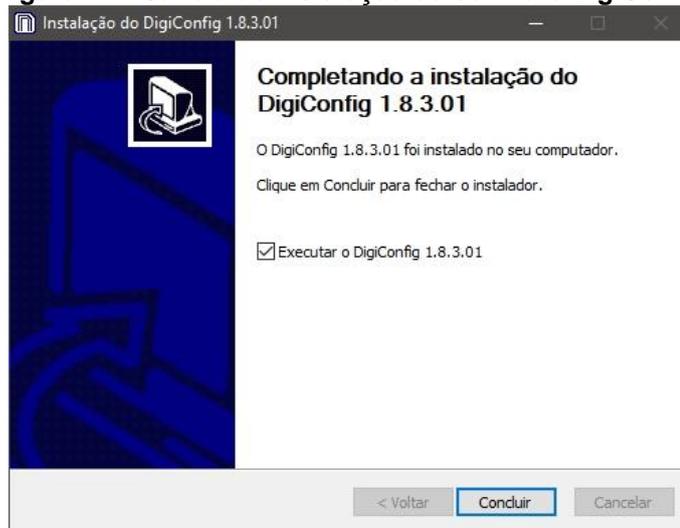
Fonte: Autoria própria.

Passo 03 – após clicar em “Instalar”, durante a instalação, aparecerá uma janela para a instalação do *driver* da Novus. Basta clicar em “Avançar” e “Concluir” para realizar a instalação do *driver* exigido.

Figura 43 – Janela de instalação do *driver*

Fonte: Autoria própria.

Pronto, o *software* DigiConfig está instalado e pronto para ser executado.

Figura 44 – Janela de instalação do *software* DigiConfig

Fonte: Autoria própria.

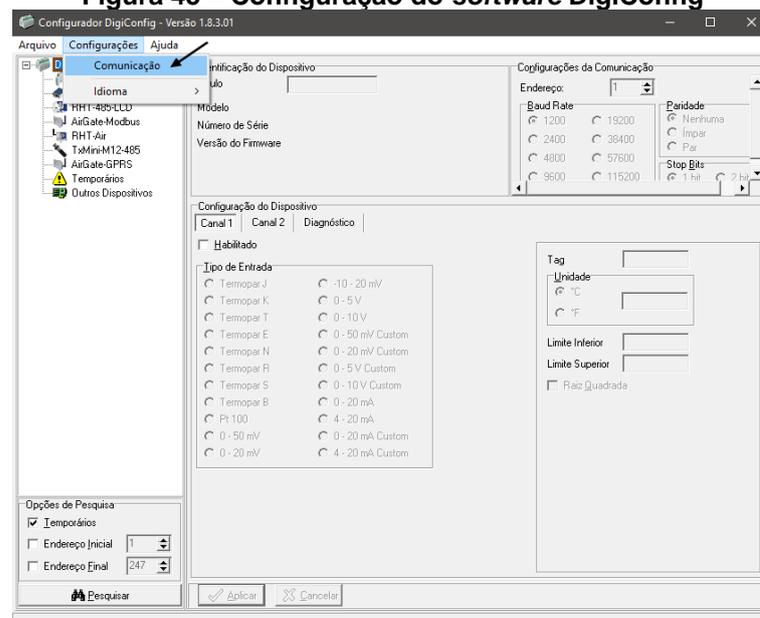
APÊNDICE E – CONFIGURAÇÃO DAS UNIDADES REMOTAS DA MARCA NOVUS UTILIZANDO O SOFTWARE DIGICONFIG

Com o *software* para configuração devidamente instalado, podemos então elaborar um tutorial para a configuração das unidades remotas presentes no kit didático utilizado neste trabalho, sendo estas da marca Novus.

Passo 01 - inicialmente, para a configuração de cada unidade remota, deve-se isolar a sua comunicação, conectando seus terminais D1 e D0 aos terminais Rx+ e Rx-, respectivamente, do Conversor Digital USB-i485, onde este é responsável por fazer a interface entre o computador e barramentos de comunicação industrial.

Passo 02 – abrir o *software* DigiConfig, clicar em “Configurações” e então em “Comunicação”.

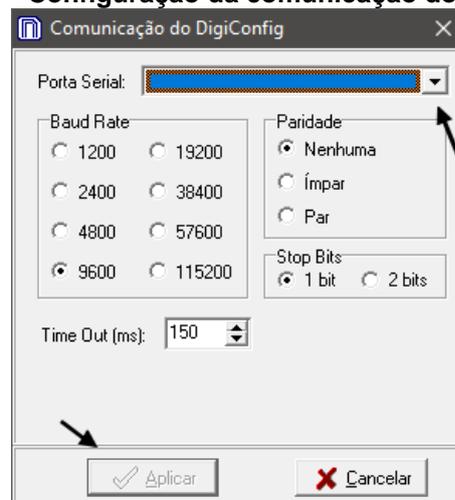
Figura 45 – Configuração do software DigiConfig



Fonte: Autoria própria.

Passo 03 – selecionar a Porta Serial que estará disponível ao conectar o Conversor Digital à uma porta USB do computador, e então clicar em “Aplicar”.

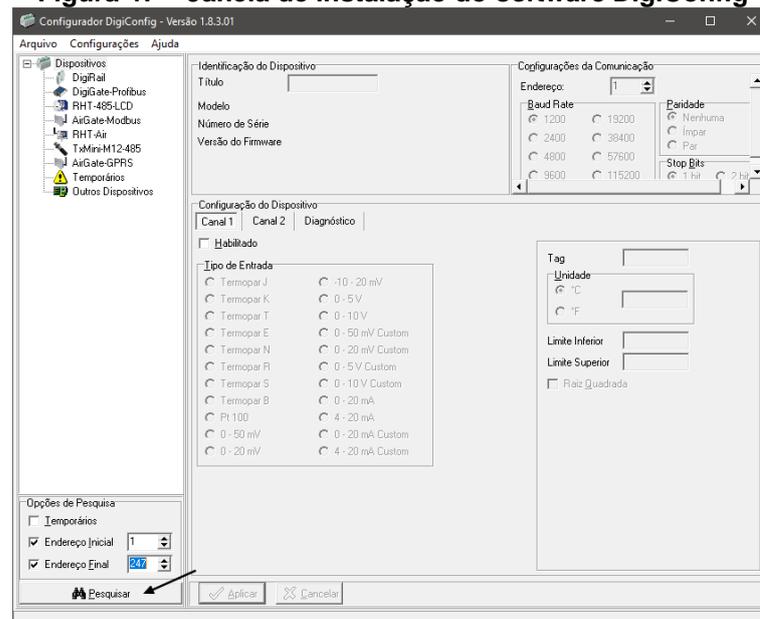
Figura 46 – Configuração da comunicação do DigiConfig



Fonte: Autoria própria.

Passo 04 – selecionar as opções “Endereço Inicial” e “Endereço Final” e clicar em “Pesquisar”.

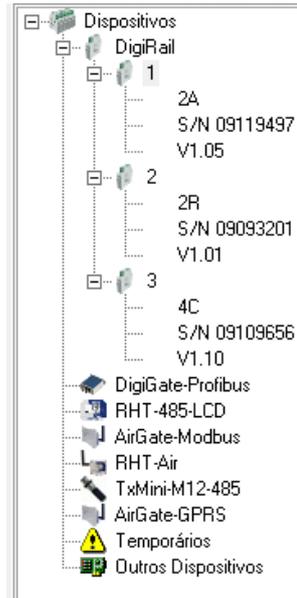
Figura 47 – Janela de instalação do software DigiConfig



Fonte: Autoria própria.

Passo 05 – após a pesquisa dos dispositivos terminar, inicia-se então a configuração de cada um deles, que no caso são: DigiRail 2A; DigiRail 2C; DigiRail 4R; DigiGate Profibus.

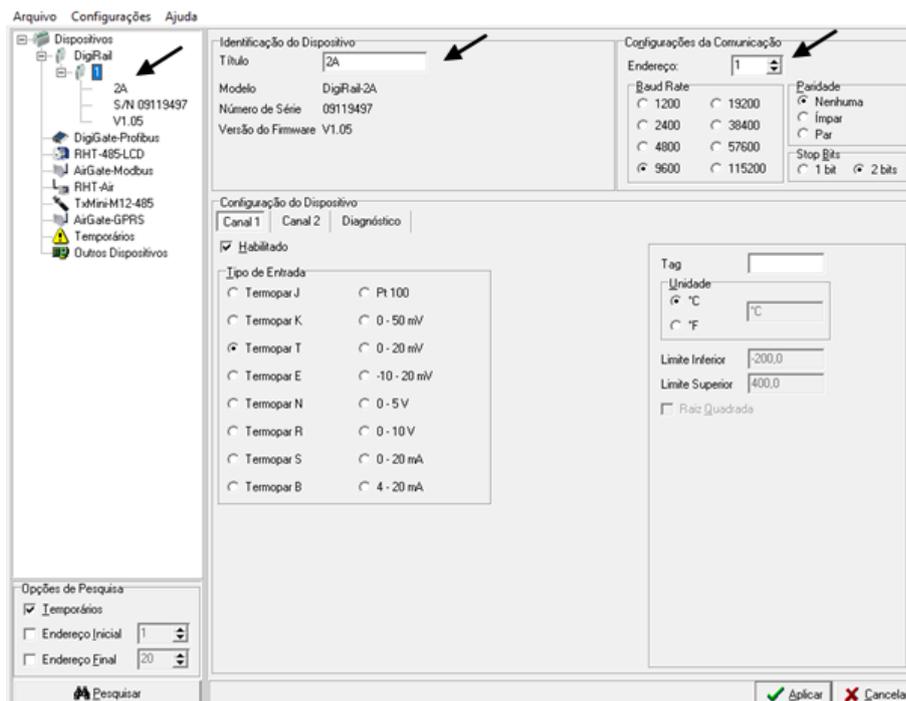
Figura 48 – Dispositivos encontrados no software DigiConfig



Fonte: Autoria própria.

Na configuração do DigiRail 2A, foi definido seu endereço e seu título, indicado pelas setas. Nesta configuração, é possível também habilitar ou desabilitar os canais 1 e 2, definir o seu tipo de entrada dentre os disponíveis, como também determinar uma tag de identificação, a fim de facilitar a identificação do dispositivo e de suas saídas.

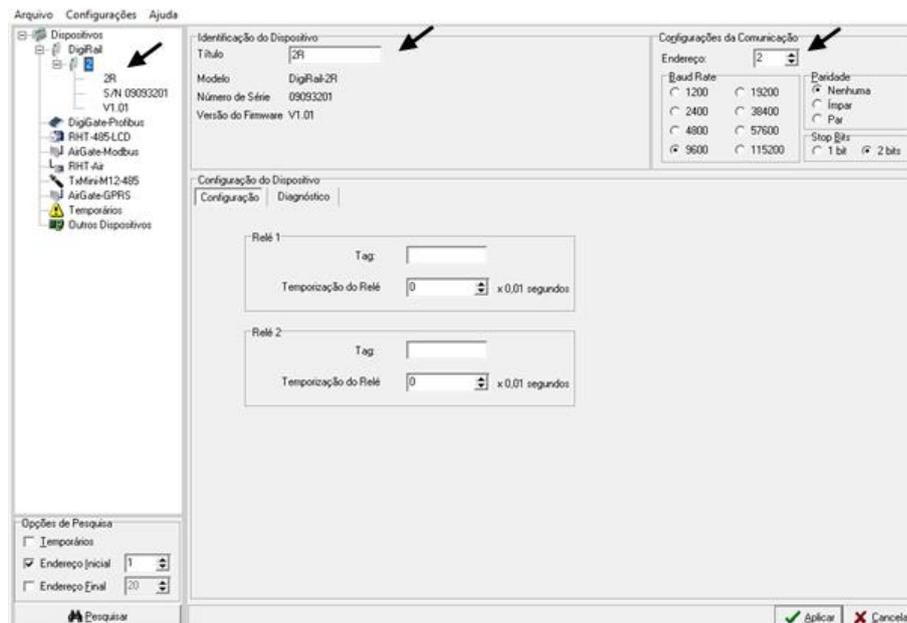
Figura 49 – Configuração do DigiRail 2A



Fonte: Autoria própria.

Assim, em sequência, foi configurado o DigiRail 2R, definindo seu endereço e seu título, indicado pelas setas. Na janela de configuração, também é possível configurar os relés 1 e 2, determinar um nome para a tag do referido relé, como também uma temporização para o relé desarmar se desejado.

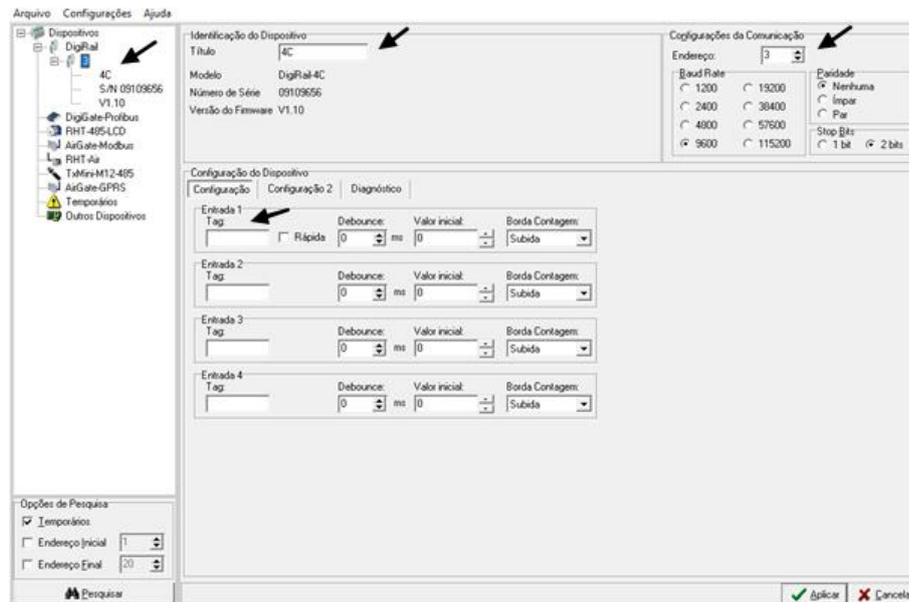
Figura 50 – Configuração do DigiRail 2R



Fonte: Autoria própria.

Na configuração do DigiRail 4C, foi definido também seu endereço e seu título, indicado pelas setas – vale ressaltar que os endereços configurados precisam ser diferentes um do outro, a fim de facilitar a leitura posterior. Na janela de configuração, é possível determinar uma tag para cada uma das 4 entradas digitais, assim como um *debounce*, valor inicial e escolher se a borda de contagem será de subida ou descida.

Figura 51 – Configuração do DigiRail 4C

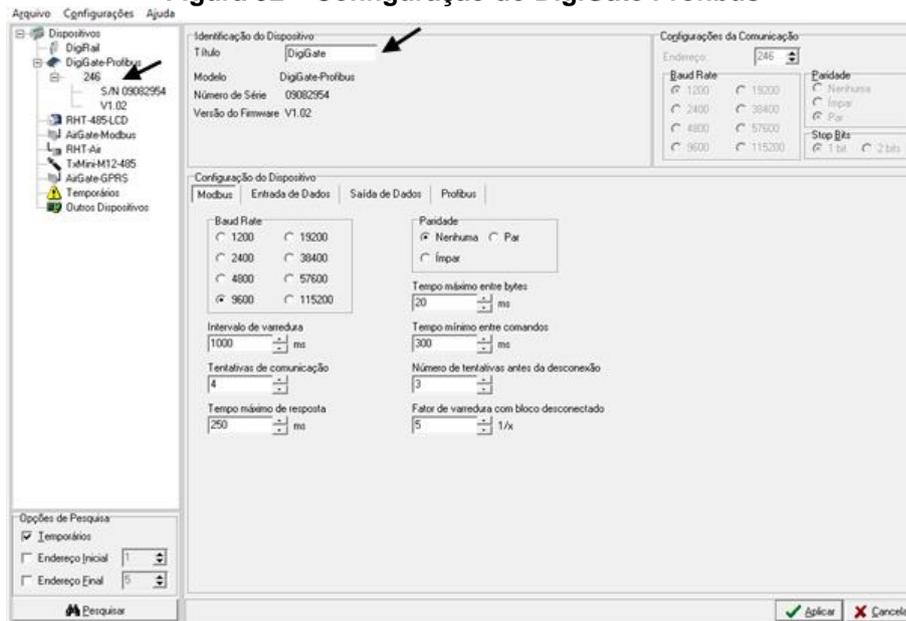


Fonte: Autoria própria.

Por fim, na configuração do DigiGate Profibus, é necessário alterar o seu modo para um “modo Configuração”. Neste modo, o aparelho deixa de se comportar como mestre na rede Modbus e passa a se comportar como escravo, aceitando assim comandos do software de configuração, sendo este último o mestre. Para acessar este modo, deve-se pressionar a tecla CFG presente na unidade remota, observando que o led Status passa a piscar lentamente.

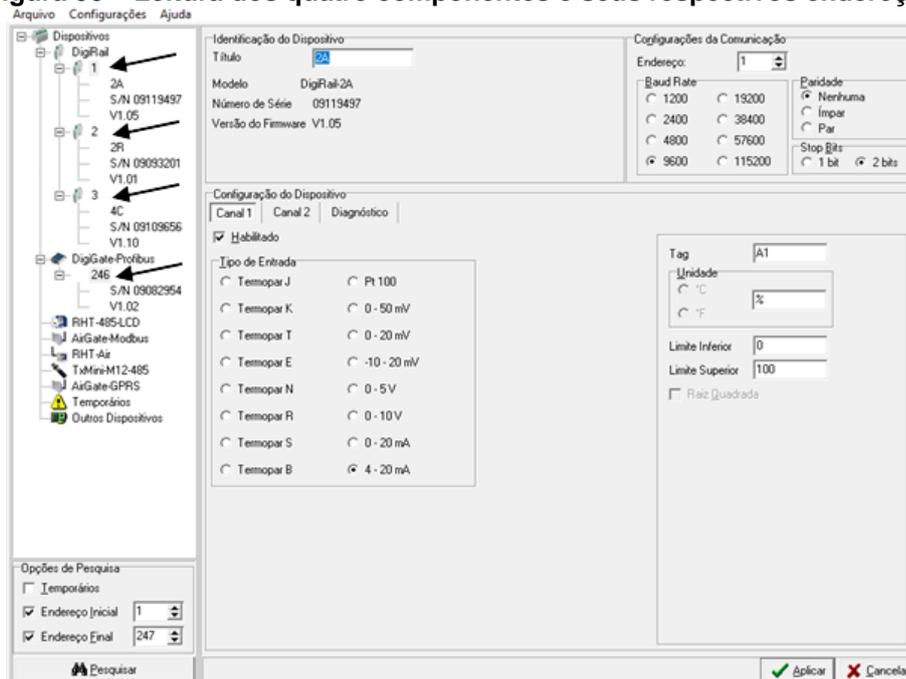
Assim, foi determinado o título para a unidade remota, visto que seu endereço é fixo (246), sendo possível alterar seu *baud rate*, outras configurações relacionadas a varredura, tempo de resposta, etc. Na janela “Profibus”, é possível alterar o endereço Profibus, quando o diagnóstico revela o número de identificação, bloco de leitura de dados, bloco de escrita de dados e o *baud rate*.

Figura 52 – Configuração do DigiGate Profibus



Fonte: Autoria própria.

Figura 53 – Leitura dos quatro componentes e seus respectivos endereços



Fonte: Autoria própria.