

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

DANIEL DE MORAES
MAURÍCIO TOMIM SARAIVA DA SILVA
RODRIGO DE ANDRADE ARRUDA

**DESENVOLVIMENTO DE UM SISTEMA DE COMUNICAÇÃO EM
REDE PROFIBUS ENTRE CLP's SIEMENS S7 300 E
IMPLEMENTAÇÃO DE SISTEMA SUPERVISÓRIO**

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA

2017

DANIEL DE MORAES
MAURÍCIO TOMIM SARAIVA DA SILVA
RODRIGO DE ANDRADE ARRUDA

**DESENVOLVIMENTO DE UM SISTEMA DE COMUNICAÇÃO EM REDE
PROFIBUS ENTRE CLP's SIEMENS S7 300 E IMPLEMENTAÇÃO DE UM
SISTEMA SUPERVISÓRIO**

Trabalho de Conclusão de Curso de Graduação, apresentado à disciplina de TCC 2, do Curso de Engenharia de Controle e Automação e Engenharia Elétrica do Departamento Acadêmico de Eletrotécnica - DAELT - da Universidade Tecnológica Federal do Paraná - UTFPR, como requisito parcial para obtenção do título de Engenheiro de Controle e Automação e Engenheiro Eletricista.

Orientador: Prof. Me. Daniel Balieiro Silva

CURITIBA
2017

DANIEL DE MORAES
MAURÍCIO TOMIM SARAIVA DA SILVA
RODRIGO DE ANDRADE ARRUDA

DESENVOLVIMENTO DE UM SISTEMA DE COMUNICAÇÃO EM REDE
PROFIBUS ENTRE CLP's SIEMENS S7 300 E IMPLEMENTAÇÃO DE SISTEMA
SUPERVISÓRIO

Este Trabalho de Conclusão de Curso de Graduação foi julgado e aprovado como requisito parcial para a obtenção do Título de Engenheiro de Controle e Automação, do curso de Engenharia de Controle e Automação do Departamento Acadêmico de Eletrotécnica (DAELT) da Universidade Tecnológica Federal do Paraná (UTFPR).

Curitiba, ____ de _____ de _____.

Prof. Paulo Sérgio Walenia, Esp.
Coordenador de Curso de
Engenharia de Controle e Automação

Profa. Annemarien Gehrke Castagna, Mestre
Responsável pelos Trabalhos de Conclusão de
Curso de Engenharia de Controle e Automação do
DAELT

ORIENTAÇÃO

BANCA EXAMINADORA

Daniel Balieiro Silva, Mestre
Universidade Tecnológica Federal do Paraná
Orientador

Daniel Balieiro Silva, Mestre
Universidade Tecnológica Federal do Paraná

Jorge Assade Leludak, Doutor
Universidade Tecnológica Federal do Paraná

José da Silva Maia, Mestre
Universidade Tecnológica Federal do Paraná

"Descobri que, quanto mais eu trabalho, mais sorte eu pareço ter."

Thomas Jefferson

RESUMO

ARRUDA, Rodrigo De Andrade. DA SILVA, Maurício Tomim Saraiva. MORAES, Daniel de. DESENVOLVIMENTO DE UM SISTEMA DE COMUNICAÇÃO EM REDE PROFIBUS ENTRE CLP's SIEMENS S7 300 E IMPLEMENTAÇÃO DE SISTEMA SUPERVISÓRIO. 2017. 109 f. Trabalho de Conclusão de Curso (Engenharia de Controle e Automação e Engenharia Elétrica) – Departamento Acadêmico de Eletrotécnica, Universidade Tecnológica Federal do Paraná. Curitiba, 2017.

Esta pesquisa apresenta uma abordagem teórica e prática referente à comunicação entre CLP's (Siemens), com foco em Rede Profibus. Apresenta detalhes de configuração em um passo-a-passo no Apêndice A e um vídeo disponibilizado no *Youtube* ("Tutorial - Comunicação entre dois CLP's Siemens S7-300 através de rede Profibus") que auxiliará futuros acadêmicos interessados no assunto.

O trabalho é complementado por um exemplo prático de aplicação de rede Profibus (semáforos controlados por CLP's), o leitor terá estas habilidades necessárias.

Será acrescentado detalhes de configuração referentes à IHM Eaton, no Apêndice B, utilizada neste trabalho. Tais detalhes auxiliam na comunicação quando se trata de várias estações interligadas em um único barramento. Estes são detalhes adicionais ao utilizado na referência INOUE E PALAR.

Além de toda a configuração entre CLP's, será feito um detalhamento, no Apêndice C, do software de supervisão WinCC que ajudará o estudante.

Traz como resultado do estudo, um aprendizado básico de Redes Industriais que poderá ser utilizado em indústrias de pequeno, médio e grande porte.

Palavras-chave: Redes Industriais. Rede Profibus. Comunicação entre CLP's. Semáforos controlados por CLP's. CLP Siemens.

ABSTRACT

ARRUDA, Rodrigo De Andrade. DA SILVA, Maurício Tomim Saraiva. MORAES, Daniel de. DEVELOPMENT OF A PROFIBUS NETWORK COMMUNICATION SYSTEM BETWEEN SIEMENS S7 300 PLC's AND SUPERVISORY SYSTEM IMPLEMENTATION. 2017. 109 p. Trabalho de Conclusão de Curso (Control and Automation Engineering and Eletric Engineering– Eletrotechnology Academic Department, Federal Technology University - Paraná. Curitiba, 2017.

This research presents a theoretical and practical approach to communication between PLCs (Siemens), focusing on Profibus Network. It presents configuration details in a step-by-step in Appendix A and a video available on Youtube ("Tutorial - Communication between two Siemens S7-300 CLPs through Profibus network") that will help future academics interested in the subject.

The work is complemented by a practical example of applying Profibus network (semaphores controlled by PLCs), the reader will have these required skills.

Configuration details for the Eaton HMI will be added in Appendix B, used in this work. Such details aid in communication when it comes to multiple stations interconnected on a single bus. These are additional details to that used in the INOUE AND PALAR reference. In addition to the entire configuration between PLCs, a detail will be given in Appendix C of the WinCC supervisory software that will help the student. It brings as a result of the study, a basic learning of Industrial Networks that can be used in small, medium and large industries.

Palavras-chave: Industrial Networks. Profibus Network. Communication between PLC's. Semaphores controlled by PLC's. Siemens PLC's.

LISTA DE FIGURAS

Figura 1 - Conexões entre dispositivos	15
Figura 2 - Distribuição de Módulos e Componentes CLP S7-300.....	22
Figura 3 - CPL S7 300	23
Figura 4 - Fonte de Alimentação Siemens para CLP S7-300.....	24
Figura 5 - CPU 313C	25
Figura 6 - Módulo de E/S Digitais, mod. 6ES7323- 1BH01-0AA0.....	26
Figura 7 - Módulo de Entradas Analógicas, modelo 6ES7331-7HF01-0AB0 .	27
Figura 8 - Diagrama de blocos de um PLC	28
Figura 9 - Ciclo de processamento (scan).....	29
Figura 10 - CLP Siemens S7-300, CPU 313C.....	31
Figura 11 - Redes de Comunicação Industrial	32
Figura 12 - Pirâmide de Automação (reeditada).....	35
Figura 13 - Exemplo de Topologia em Barramento.....	37
Figura 14 - Áreas de Aplicação do PROFIBUS	39
Figura 15 - Níveis de Sinais em Barramento RS-485.....	42
Figura 16 - Barramento RS-485 (reeditada).....	43
Figura 17 - Passagem de Token entre Mestres	45
Figura 18 - Protocolo Único de Acesso à Rede.....	46
Figura 19 - Conexão em Topologia Barramento	49
Figura 20 - Pinagem dos Conectores	51
Figura 21 - Vista de um Conector de Rede Profibus	53
Figura 22 - Sistema de supervisão e controle	56
Figura 23 - Elipse SCADA	58
Figura 24 - Elipse E3	59
Figura 25 - SIMATIC Logon Service no sistema WinCC.....	62
Figura 26 - Exemplo de IHM utilizando o software WinCC.....	64
Figura 27 - Blocos utilizados para o projeto de comunicação.	68
Figura 28 - Cruzamento desenvolvido no software WinCC.....	69
Figura 29 - Botão para solicitação de pedestre desenvolvido no software Galileo.....	70

Figura 30 - Projeto de comunicação de CLP's.	70
Figura 31 - Tela Inicial com a estações mestre e escravo.	77
Figura 32 – Abrir projeto estação mestre.	78
Figura 33 – Abrir configurações de hardware estação mestre.	78
Figura 34 – Tela de configuração de hardware estação mestre.	79
Figura 35 – Abrir tela de configuração de rede DP da estação mestre.	79
Figura 36 – Tela Propriedades – DP – (R0/S2.1) estação mestre.	80
Figura 37 – Abrir tela Propriedades – DP – (R0/S2.1) estação mestre.	80
Figura 38 – Criar nova rede de comunicação.	81
Figura 39 – Configuração da rede PROFIBUS.	82
Figura 40 – Tela Propriedades – DP – (R0/S2.1).	82
Figura 41 – Confirmar configurações da rede PROFIBUS.	83
Figura 42 – Salvar configurações da rede PROFIBUS.	83
Figura 43 – Abrir projeto referente a estação escravo.	84
Figura 44 – Configurar o hardware referente a estação escravo.	84
Figura 45 - Abrir tela de configuração de rede DP da estação escravo.	85
Figura 46 - Abrir tela Propriedades – DP – (R0/S2.1) estação escravo.	85
Figura 47 - Tela Propriedades – DP – (R0/S2.1) estação escravo.	86
Figura 48 – Configurar a estação como escravo.	87
Figura 49 – Salvar e fechar as configurações da estação escravo.	87
Figura 50 – Abrir projeto da estação mestre.	88
Figura 51 – Abrir configuração de hardware da estação mestre.	88
Figura 52 – Abrir o catálogo.	89
Figura 53 – Expandir a lista PROFIBUS DP do catálogo.	89
Figura 54 – Expandir a lista de configuração de estações.	90
Figura 55 – Inserir a CPU 31x na rede PROFIBUS.	91
Figura 56 – Acoplar a estação escravo a estação mestre.	91
Figura 57 – Confirmar o acoplamento da estação escravo.	92
Figura 58 – Salvar e fechar o projeto da estação mestre.	92
Figura 59 – Abrir projeto da estação escravo.	93
Figura 60 – Abrir configuração de hardware da estação escravo.	93
Figura 61 – Abrir configuração DP da estação escravo.	94
Figura 62 – Abrir nova configuração DP na estação escravo.	94
Figura 63 – Tela Propriedades – DP – (R0/S2.1) estação escravo.	95

Figura 64 – Frame de comunicação mestre para o escravo.	96
Figura 65 - Frame de comunicação mestre para o escravo.	97
Figura 66 – Confirmando as configurações do frame de comunicação.	97
Figura 67 – Criar novo frame de comunicação.....	98
Figura 68 – Frame de comunicação da estação escravo / mestre.....	98
Figura 69 - Frame de comunicação da estação escravo / mestre.....	99
Figura 70 – Confirmar configurações do frame de comunicação.	99
Figura 71 – Verificar segundo frame criado e confirmar configurações.	100
Figura 72 – Salvar e fechar o projeto da estação escravo.	100
Figura 73 – Download do projeto da estação escravo.	101
Figura 74 – Download do projeto da estação mestre.	101
Figura 75 – Tela de configuração dos frames de comunicação.	102
Figura 76 - Programação Software SIMATIC Manager.....	103
Figura 77 - Programação Software SIMATIC Manager.....	104
Figura 78 - Tela inicial de um projeto no WinCC.	107
Figura 79 - Aba Connections.	108
Figura 80 - Parâmetros de conexão.	108
Figura 81 - Drivers de comunicação com o WinCC Flexible 2008.	109

LISTA DE QUADROS

Quadro 1 - Áreas de Aplicação do PROFIBUS	41
Quadro 2 - Características do Cabo AL-2303	50
Quadro 3 - Comprimento máximo para segmento e expansão em redes PROFIBUS utilizando RS-485	50
Quadro 4 - Requisitos de Hardware	61
Quadro 5 – Relação entre bits das estações mestre e escravo	103
Quadro 6 - Parâmetros de Configuração das IHM's Mestre e Escravo (exemplo Semáforo)	105
Quadro 7 - Número de Estações (exemplo Semáforo)	106

LISTA DE SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
CLP	Controlador Lógico Programável
CLPs	Controladores Lógicos Programáveis
CNC	Comando Numérico Computadorizado
CP	Módulo de Comunicação
DAELT	Departamento Acadêmico de Eletrotécnica
DIN	Deutsches Institut für Normung (Instituto Alemão de Normalização)
DN	Departamento Nacional
DP	Decentralized Peripherals (Periféricos Descentralizados)
DPM1	Mestre DP Classe 1
EPROM	Erasable programmable read-only memory (Memória Somente Leitura Programável e Apagável)
E/S	Entradas e saídas
FET	Field Effect Transistor (Transistor de efeito de campo)
FM	Módulo de Função
Grafcet	Graphe Fonctionnel de Commande (Controle Gráfico Funcional)
IEC	International Electrotechnical Commission (Comissão de Eletrotécnica Internacional)
IHM	Interface Humano Máquina
MAC	Controle de Acesso ao Meio
MODICON	Modular digital controller (Controlador digital modular)
MPI	Multi-point Interface
PC	Personal Computer (Computador pessoal)
PID	Proporcional Integral Derivativo
PROFIBUS	Process Field Bus
PS	Fonte de Alimentação
RAM	Random Access Memory (Memória de acesso aleatório)
S7	Step 7

SCADA	Supervisory Control and Data Acquisition (Controle de Supervisão e Aquisição de Dados)
SDCD	Sistema Digital de Controle Distribuído
SFC	Sequence functional chart (Gráfico funcional da seqüência)
SM	Módulo de Sinal
TRIAC	Triode for Alternating Current (Triode para corrente alternada)
UCP	Unidade Central de Processamento
UTFPR	Universidade Tecnológica Federal do Paraná

SUMÁRIO

1	INTRODUÇÃO	14
1.1	TEMA	14
1.2	PROBLEMAS E PREMISSAS	16
1.3	OBJETIVOS	16
1.3.1	Objetivo Geral	16
1.3.2	Objetivos Específicos	16
1.4	JUSTIFICATIVA	17
1.5	PROCEDIMENTOS METODOLÓGICOS	18
1.6	RECURSOS	18
2	CONTROLADORES LÓGICOS PROGRAMÁVEIS	19
2.1	HISTÓRIA	20
2.2	PRINCIPAIS COMPONENTES DE UM CLP	22
2.2.1	Fonte de alimentação	23
2.2.2	Memória EPROM	24
2.2.3	Memória do usuário	24
2.2.4	Unidade Central de Processamento – UCP	25
2.2.5	Memória de Dados	25
2.2.6	Memória Imagem das E/S	25
2.2.7	Módulos de Saídas	26
2.2.8	Módulo de Entradas	27
2.2.9	Terminal de Programação	27
2.2.10	Terminais remotos de E/S	28
2.3	PRINCÍPIO DE FUNCIONAMENTO	28
2.4	LINGUAGENS DE PROGRAMAÇÃO	29

2.5	FABRICANTES	30
3	REDES INDUSTRIAIS	31
3.1	AUTOMAÇÃO	34
3.2	TIPOS DE COMUNICAÇÃO	35
3.2.1	Comunicação Paralela	36
3.2.2	Comunicação Série	36
3.1.2.1.	Comunicação Serial Síncrona.....	36
3.1.2.2.	Comunicação Serial Assíncrona.....	36
3.3	TOPOLOGIA DE REDE	37
3.4	TECNOLOGIA PROFIBUS	38
3.5	A FAMÍLIA PROFIBUS	39
3.6	ARQUITETURA	40
3.7	1ª CAMADA - TRANSMISSÃO TIA/EIA-485 (RS-485).....	41
3.7.1	Interface Diferencial Balanceada	42
3.7.2	Operação Multiponto.....	43
3.7.3	Vantagens.....	43
3.7.4	Meio Físico.....	44
3.8	2ª CAMADA – ACESSO À REDE	44
3.9	PROFIBUS DP.....	46
3.9.1	Procedimentos do Sistema	47
3.9.2	Transmissão Cíclica de Dados	47
3.10	PROFIBUS FMS	48
3.11	PROFIBUS DP E PROFIBUS FMS	48
3.11.1	Instalação.....	48
3.11.2	Dispositivos	48
3.11.3	Cabeamento.....	49

3.11.4 Conector.....	51
4 SISTEMAS SUPERVISÓRIOS	53
4.1 SCADA – DEFINIÇÃO	54
4.2 ARQUITETURA SCADA.....	55
4.3 INTERFACE HUMANO-MÁQUINA.....	56
4.4 SISTEMAS SUPERVISÓRIOS EXISTENTES.....	57
4.4.1 Elipse Scada	57
4.4.2 Elipse E3.....	58
4.4.3 Genesis32.....	60
4.4.4 Labview	60
4.4.5 SIMATIC WinCC	60
4.4.5.1 Gerenciamento Integrado ao Usuário	62
4.4.5.2 Interface de Usuário.....	62
4.4.5.3 Funções de Zoom	63
4.4.5.4 Sistema de Arquivamento	63
5 SISTEMA INTEGRADO PROPOSTO.....	64
5.1 PROJETO DE COMUNICAÇÃO ENTRE SEMÁFOROS.....	65
5.1.1 Caracterização dos estágios.....	66
5.1.2 O Projeto	67
6 CONCLUSÃO	71
REFERÊNCIAS	73
APÊNDICE A	77
APÊNDICE B	105
APÊNDICE C.....	107

1 INTRODUÇÃO

Segundo Capelli (2007, p. 15), a automação na indústria teve início no século XVIII na Inglaterra com a revolução industrial, onde o foco visava maiores lucros, menores custos e maior produtividade, tudo isso impulsionado pelo crescimento populacional que aumentou a demanda de produtos e mercadorias. Os dispositivos precisavam ser desenvolvidos para cada tarefa e devido à natureza mecânica, tinham uma pequena vida útil.

Na década de 1920, os dispositivos mecânicos foram substituídos por relés, que viabilizou o desenvolvimento de funções de controle mais complexas e sofisticadas. Eles se mostraram uma alternativa de custo viável, especialmente para a automação de pequenas máquinas com um número limitado de transdutores e atuadores (FRANCHI E CAMARGO, 2009, p. 21).

Para Silveira e Santos (1998, p. 17), até o início da década de 60, a utilização de relés eletromecânicos era a única opção possível. Com o advento dos dispositivos microprocessados, vieram os controladores lógicos programáveis (CLP), em que a forma básica de programação era oriunda da lógica de programação dos diagramas elétricos a relés.

Segundo Moraes e Castrucci (2001, p. 29), foi na década de 1980 que houve aperfeiçoamento das funções de comunicação dos CLP's, sendo então utilizados em rede.

1.1 TEMA

O trabalho estará restrito à comunicação entre os CLP's e o software de supervisão em razão da falta de dispositivos para comunicação em rede. A plataforma de integração será composta de *hardware* e *software* de um único fabricante, neste caso, Siemens.

1.1.1. Delimitação do Tema

Este trabalho buscará apresentar algumas técnicas da engenharia de automação industrial e aplicá-las para o desenvolvimento de uma lógica de programação e configuração de hardware e software que seja capaz de comunicar dois CLP's (marca Siemens, modelo S7-313) através de uma rede Profibus, bem como interligar junto à IHM (marca EATON) onde os comandos poderão ser simulados.

Serão feitas conexões entre dois CPL's, utilizando como interface uma IHM e um PC para cada CLP. A Figura 1 ilustra as conexões a serem realizadas.

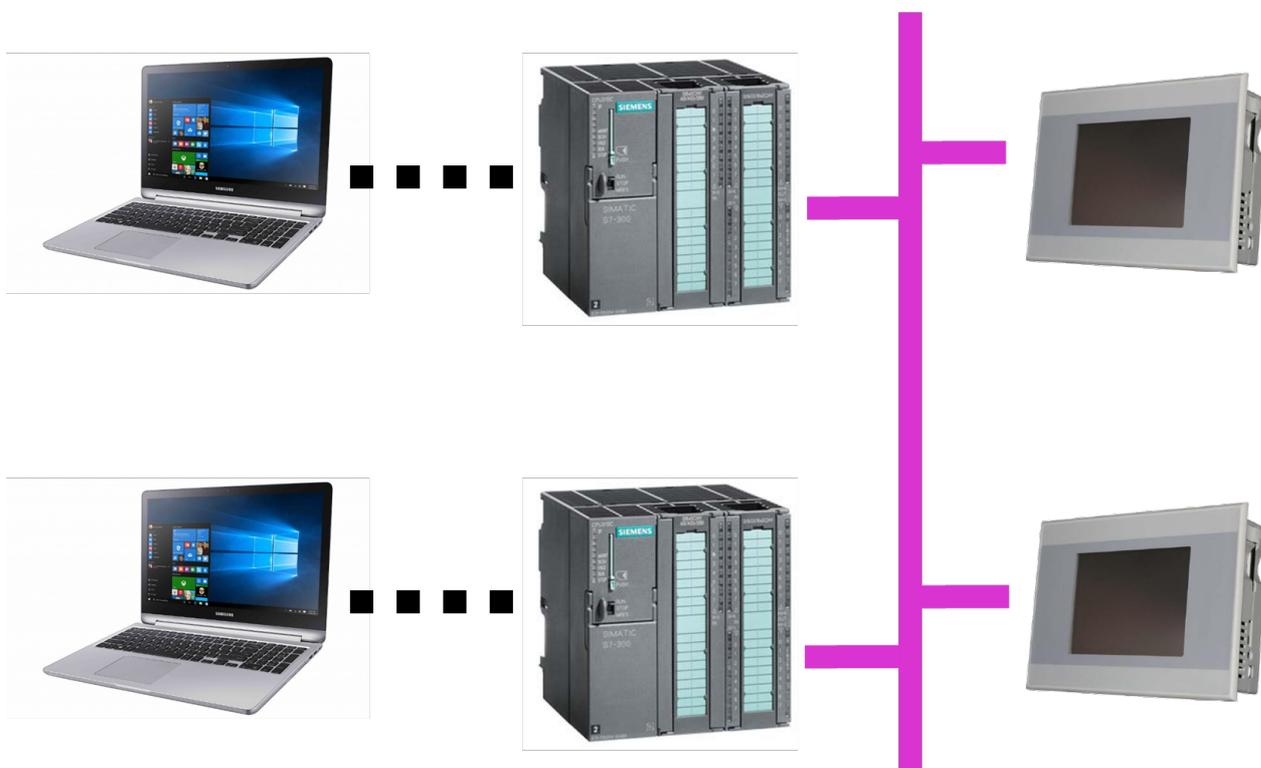


Figura 1 - Conexões entre dispositivos
Fonte: Autoria própria

Verifica-se que, na Figura 1, existem conexões representadas por linhas contínuas e pontilhadas. As linhas contínuas ilustram a conexão via rede PROFIBUS e as linhas pontilhadas conexão MPI. Tais conexões serão descritas posteriormente.

1.2 PROBLEMAS E PREMISSAS

As indústrias automatizadas, em sua maioria, se aperfeiçoam rapidamente e dão origem a novos aprendizados que muitas vezes os cursos de aprimoramento profissional e universidades nacionais não conseguem acompanhar esse desenvolvimento.

Logo: Quais processos e métodos da engenharia de automação as indústrias utilizam?

Este trabalho tem como premissa comunicar dois controladores lógicos programáveis muito utilizados em indústrias automatizadas e que ainda não está inserido no meio acadêmico da Universidade Tecnológica Federal do Paraná.

1.3 OBJETIVOS

1.3.1 Objetivo Geral

Elaborar material didático para a integração de um sistema de automação SCADA composto de CPL's e IHM's a ser utilizado nas disciplinas dos cursos de Automação Industrial do DAELT.

1.3.2 Objetivos Específicos

- ✓ Estudar a literatura referente à família de CLPs Siemens S7-300, rede Profibus e supervisório WinCC.
- ✓ Levantar os principais problemas de comunicação entre os CLP's.
- ✓ Estudar as principais técnicas de Engenharia de Automação Industrial existentes.

- ✓ Integrar um sistema de automação entre CLP Siemens S7-313 e software de supervisão WinCC.
- ✓ Realizar a ligação em rede Profibus entre dois CLPs Siemens S7-313.
- ✓ Realizar a conexão entre os CLPs Siemens S7-313 e IHM.
- ✓ Descrever as vantagens e desvantagens do sistema desenvolvido com os demais sistemas utilizados no mercado.
- ✓ Analisar os resultados obtidos.
- ✓ Descrever um passo-a-passo para futuros acadêmicos em pdf e entregue ao professor orientador deste trabalho para disponibilização a alunos da UTFPR.
- ✓ Produzir um vídeo do passo-a-passo de configuração de comunicação entre dois CLP's Siemens S7-313 e disponibilizá-los aos futuros acadêmicos.

1.4 JUSTIFICATIVA

O presente trabalho surge devido à constante atualização tecnológica na área de automação e comunicação, o que acaba dificultando o acompanhamento por parte das instituições de ensino. É possível referenciar, também, a necessidade das indústrias em se obter cada vez mais a gestão de equipamentos, bem como a garantia de que eles trabalhem de maneira sincronizada e interligada refletindo em maior produtividade, custo-benefício e segurança as pessoas e processos.

Como uma alternativa aos alunos da universidade, para aprofundar o conhecimento em comunicação de CLP's em rede de maneira simplificada, autodidata e prática uma vez que da conclusão do trabalho o mesmo ficará disponível para consulta e testes no laboratório da Universidade Tecnológica Federal do Paraná, Campus Curitiba, Unidade Sede.

1.5 PROCEDIMENTOS METODOLÓGICOS

O trabalho será desenvolvido primeiramente pelo levantamento e estudo das técnicas de implementação de uma Rede Profibus e o funcionamento do Step7, buscando assim todo o conhecimento necessário para realizar as análises e avaliações dos resultados a serem obtidos.

Após o levantamento e o estudo do referencial teórico, será realizada a primeira análise de programação através do simulador do Step7, em que será desenvolvida uma lógica de programação para simular a comunicação de dois CLPs.

Tendo a primeira avaliação finalizada, será iniciado a parte prática que consiste em fazer a configuração do hardware nos dois CLPs, e após, iniciar testes de configuração e implementar a lógica de programação já desenvolvida.

Será também desenvolvido um sistema supervisório através do WinCC para que se mostre claramente que o objetivo principal foi alcançado. A aplicação do sistema supervisório a ser desenvolvido será definida posteriormente aos estudos teóricos para analisar a melhor forma de demonstrar a comunicação entre dois CLP's.

1.6 RECURSOS

Os recursos de hardware utilizados serão os que estão disponíveis na UTFPR:

- ✓ 2 CPL's: Marca: Siemens, modelo: S7 S313;
- ✓ 2 IHM's: Marca: EATON, modelo: XV-102-B2-35TQR-10;
- ✓ 2 PC's (computadores pessoais);
- ✓ 3 cabos de Rede PROFIBUS;
- ✓ 2 cabos MPI;

Os softwares utilizados estão dispostos a seguir com seus respectivos hardwares:

- ✓ Step7 Siemens (PC's);
- ✓ Software de Supervisão WinCC (PC's);
- ✓ Galileo (IHM's);

2 CONTROLADORES LÓGICOS PROGRAMÁVEIS

Com a Automação tomando seu espaço na indústria, surgiu também a necessidade de se criar um dispositivo que substituísse os painéis de controle a relés e que fosse prático realizar uma mudança de lógica de produção sem perder tempo alterando contadores. Desta maneira, atendendo a esta necessidade da indústria, surgiu o CLP.

Controlador Lógico Programável ou Controlador Programável, conhecido também por suas siglas CLP ou CP e pela sigla de expressão inglesa PLC (Programmable Logic Controller), é um computador especializado, baseado num microprocessador que desempenha funções de controle - controle de diversos tipos e níveis de complexidade. Geralmente as famílias de Controladores Lógicos Programáveis são definidas pela capacidade de processamento de um determinado número de pontos de Entradas e/ou Saídas (E/S). Controlador Lógico Programável é um equipamento eletrônico digital com hardware e software compatíveis com aplicações industriais. Segundo a NEMA (National Electrical Manufacturers Association), é um aparelho eletrônico digital que utiliza uma memória programável para armazenar internamente instruções e para implementar funções específicas, tais como lógica, seqüenciamento, temporização, contagem e aritmética, controlando, por meio de módulos de entradas e saídas, vários tipos de máquinas ou processos. (SILVA MELLO, p. 2)

Segundo Moraes e Castrucci (2001, p. 23), as principais características dos controladores lógicos programáveis são:

- Linguagens de programação de alto nível, caracterizando um sistema bastante amigável ao operador. Após o programa estar concluído, ele pode ser transferido para outros CLPs, garantindo confiabilidade na sua utilização;
- Simplificação nos quadros e painéis elétricos: Toda a fiação do comando fica resumida a um conjunto de entradas e saídas. Assim, qualquer alteração necessária torna-se mais rápida e barata.

- Confiabilidade operacional: Já que as alterações podem ser realizadas através do *software*, necessitando de pouca ou nenhuma alteração da fiação elétrica, a possibilidade de haver erros é minimizada, garantindo sucesso no desenvolvimento ou melhoria a ser implementada.
- Funções avançadas: Os CLPs podem realizar uma grande variedade de tarefas de controle através de funções matemáticas, controle da qualidade e informações para relatórios;
- Comunicação de rede: Através de interfaces de operação, controladores e operadores em rede permitem coleta de dados e um enorme intercâmbio de troca de dados em relação aos níveis da Pirâmide de Automação.

2.1 HISTÓRIA

O controlador lógico programável é definido pela IEC (International Electrotechnical Commission) como:

Sistema eletrônico operando digitalmente, projetado para o uso em um ambiente industrial, que usa uma memória programável para a armazenagem interna de instruções orientadas para o usuário para implementar funções específicas, tais como lógica, sequencial, temporização, contagem e aritmética, para controlar, através de entradas e saídas digitais ou analógicas, vários tipos de máquinas ou processos. O controlador programável e seus periféricos associados são projetados para serem facilmente integráveis em um sistema de controle industrial e facilmente usados em todas as suas funções previstas. (FRANCHI E CAMARGO, 2009, p.23).

Stéfano (2007, p.4) define o CLP através do conceito apresentado pela ABNT, que se refere a um “equipamento eletrônico digital com hardware e software compatíveis com aplicações industriais”.

Os sistemas de controle surgiram durante o período da Revolução Industrial, que teve início no século XVIII, na Inglaterra, com a mecanização dos sistemas, tal revolução foi marcada principalmente pela transformação nos processos de produção artesanal para a produção por máquinas (CAPELLI, 2007).

Desencadeada pela burguesia que buscava maiores lucros, menores custos e produção acelerada na fabricação de mercadorias, outro fator foi o crescimento populacional, que trouxe maior diligência de produtos (CAPELLI, 2007).

A Revolução Industrial foi responsável por causar mudanças econômicas, sociais e tecnológicas, no mundo mediante o desenvolvimento tecnológico (CAPELLI, 2007).

Durante o período da Revolução industrial, as linhas de montagem da época contavam com engenhosos dispositivos mecânicos, os quais automatizavam algumas tarefas críticas e repetitivas, dispositivos estes substituídos em 1920, por painéis de relés eletromecânicos, permitindo o desenvolvimento de funções mais complexas e sofisticadas (FRANCHI E CAMARGO, 2009).

Com a inserção dos painéis, a dificuldade passou a ser as atualizações e modificações, economicamente não atrativas devido à grande complexidade para alterar a lógica, que implicava na modificação física de fios, cabos e componentes, acarretando na aquisição de novos painéis (FRANCHI E CAMARGO, 2009).

O primeiro CLP foi desenvolvido no final de 1960, pela companhia americana chamada Bedford Associated que ofereceu a empresa General Motors, o dispositivo de computação chamado de MODICON 084 (Modular digital controller), referia-se a um dispositivo que poderia realizar várias operações distintas e facilmente programável (CAPELLI, 2007).

Stéfano (2007) destaca que os CLPs foram primeiramente utilizados em indústrias automobilísticas, após isso, passaram a ser utilizados nos outros segmentos industriais. Assim como a sua aceitação, a procura por mais funções, maior capacidade de memória e de pontos de entrada e saída, aumentou.

Segundo Moraes e Castrucci (2001, p.29) na década de 1970 passou-se a incorporar na fabricação dos CLPs microprocessadores, o que aumentou a sua aplicação, e em 1980 passou a ser utilizado em redes de comunicação de dados, sendo suas principais características, as listadas abaixo:

- Com uma linguagem de programação de alto nível.
- Sistema de fácil utilização pelo operador.
- Possibilita a intercambialidade do programa para outros CLPs.
- Simplifica o projeto e instalação dos sistemas elétricos.

- Diminui o volume de fiação.

Estas características tornaram as futuras alterações e ampliações dos sistemas de produção mais ágeis e com menor custo, contribuindo para o aumento da confiabilidade do CLP na indústria. (MORAES E CASTRUCCI, 2001, p.30).

2.2 PRINCIPAIS COMPONENTES DE UM CLP

Na Figura 2, temos o esquemático da distribuição dos módulos e componentes de um CLP S7-300.

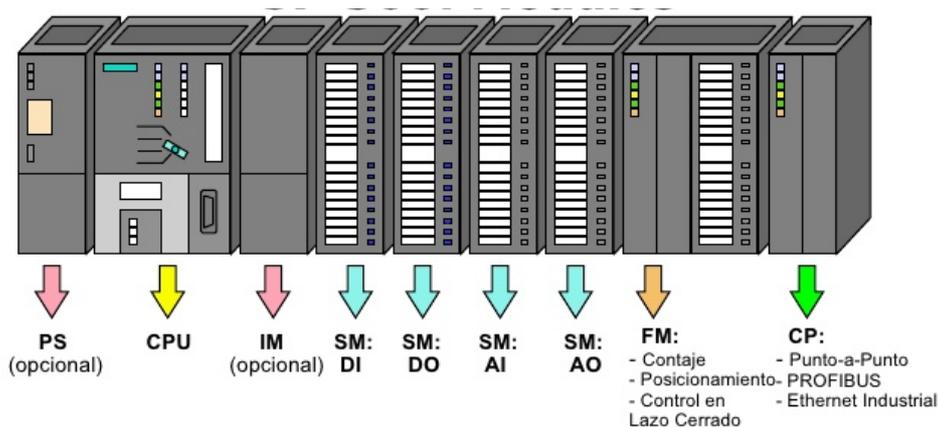


Figura 2 - Distribuição de Módulos e Componentes CLP S7-300
Fonte: INACAP

O CPL utilizado neste trabalho é ilustrado pela Figura 3.



Figura 3 - CPL S7 300
Fonte: SIEMENS (1)

2.2.1 Fonte de alimentação

As fontes de alimentação, podem ser de dois tipos **source** quando é interna ao controlador ou **sink** quando é externa. É responsável pela conversão da corrente alternada para contínua a qual o controlador é alimentado. (MORAES E CASTRUCCI, 2001, p.31).

Como segurança existe uma bateria que em caso de falta de energia da rede evita a perda do programa do usuário. (MORAES E CASTRUCCI, 2001, p.31).

A Figura 4 é uma fonte de alimentação Siemens, modelo 6ES7307-1BA01-0AA0, para CLP S7-300.



Figura 4 - Fonte de Alimentação Siemens para CLP S7-300
Fonte: Siemens

2.2.2 Memória EPROM

A memória EPROM contém o programa elaborado pelo fabricante do CLP, que realiza as funções básicas como *start-up*, armazenamento e gerenciamento da sequência de operações, esta memória não é acessível pelos usuários. (MORAES E CASTRUCCI, 2001, p.32).

2.2.3 Memória do usuário

A memória do usuário possui dois estados: RUN, quando está em operação; e PROG, quando está parado aguardando o carregamento do programa no CLP. Responsável pelo armazenamento do programa elaborado pelo usuário que terá seus dados internos e imagens de E/S processados pela UCP, depois de processado irá recebê-los para novamente armazená-los. (MORAES E CASTRUCCI, 2001, p.32).

2.2.4 Unidade Central de Processamento – UCP

A UCP é responsável pela execução do programa do usuário, atualização das memórias de dados, imagem e das entradas e saídas. Para Silveira e Santos (1998, p.84) a UCP é o “cérebro” que controla todas as ações do CLP sendo constituída pelo processador, memórias e barramento de interligação (MORAES E CASTRUCCI, 2001, p.31).

A Figura 5 é uma CPU 313C:



Figura 5 - CPU 313C
Fonte: Siemens

2.2.5 Memória de Dados

É uma tabela de valores manipuláveis onde estão as informações referentes ao processamento do programa do usuário. (MORAES E CASTRUCCI, 2001, p.32).

2.2.6 Memória Imagem das E/S

Reproduz o estado dos periféricos de entrada e saída que podem ser digitais com níveis 0 ou 1, ou analógicos com variações de tensão e corrente. (MORAES E CASTRUCCI, 2001, p.33).

2.2.7 Módulos de Saídas

Stéfano (2007) define um ponto de saída como sendo o sinal produzido pelo controlador programável para acionar dispositivos ou componentes do sistema de controle (atuadores). Exemplo: lâmpadas, solenoides, motores e etc.

Moraes e Castrucci (2001, p.34) descrevem que basicamente são três tipos de saídas:

- A relé com capacidade até 5,0A apresenta uma ótima robustez e imune aos possíveis transientes de rede, contudo possui uma baixa vida útil em relação aos demais, em torno de 150.000 e 300.000 acionamentos.
- Saída a Triac é utilizado quando a fonte é de corrente alternada, tendo uma vida útil de aproximadamente 1×10^6 acionamentos com uma capacidade de 1,0A.
- Saída a Transistor podendo ser do tipo NPN ou do tipo FET, geralmente é o mais utilizado, sendo recomendado quando é utilizado fontes de corrente contínua e há a necessidade de acionamentos em grande frequência de operação, com capacidade até 1,0A e vida útil de 1×10^6 acionamentos.

Um módulo de entradas e saídas digitais está representado na Figura 6.



Figura 6 - Módulo de E/S Digitais, mod. 6ES7323- 1BH01-0AA0
Fonte: Siemens

2.2.8 Módulo de Entradas

Stéfano (2007) destaca que um ponto de entrada é considerado o sinal recebido pelo controlador programável, através de um dispositivo ou componente externo (sensores). Exemplo: micro chaves, botões, termopares, relés, etc.

O sensor é um componente que transforma uma condição física do elemento em um sinal elétrico para ser utilizado pelo CLP, através da conexão de entrada. Um exemplo típico consiste em ao pressionar um botão de pressão momentânea, um sinal elétrico é enviado ao CLP, indicando sua condição atual que pode ser pressionado ou liberado. (FRANCHI E CAMARGO, 2009).

Os módulos de entrada quando recebem o sinal, através de opto isoladores fazem circular uma corrente no circuito de baixa potência do CLP, possuem uma capacidade de 100mA e vida útil de 10×10^6 acionamentos. (MORAES E CASTRUCCI, 2001, p.35).

Um módulo de entradas analógicas está representado na Figura 7.



Figura 7 - Módulo de Entradas Analógicas, modelo 6ES7331-7HF01-0AB0
Fonte: Siemens

2.2.9 Terminal de Programação

É responsável pela comunicação entre o usuário e o CLP, pode ser um computador ou um dispositivo portátil que permite o autodiagnóstico, alterações

online, programação de instruções, monitoramento, gravar ou apagar a memória. (MORAES E CASTRUCCI, 2001, p.37).

2.2.10 Terminais remotos de E/S

Também chamados de unidades remotas, fazem comunicação com CLP, em situações em que é inviável a conexão de todas as entradas e saídas devido a quantidade e distância da instalação, sendo esse o conceito de rede de remotas. (MORAES E CASTRUCCI, 2001, p.38).

2.3 PRINCÍPIO DE FUNCIONAMENTO

Consiste na execução por parte da UCP do programa do fabricante, este realiza ciclicamente a leitura das entradas, realiza a execução do programa do usuário e atualiza os valores das saídas. (SILVEIRA E SANTOS, 1998, p.83).

A Figura 8 representa um diagrama de blocos de um CLP.

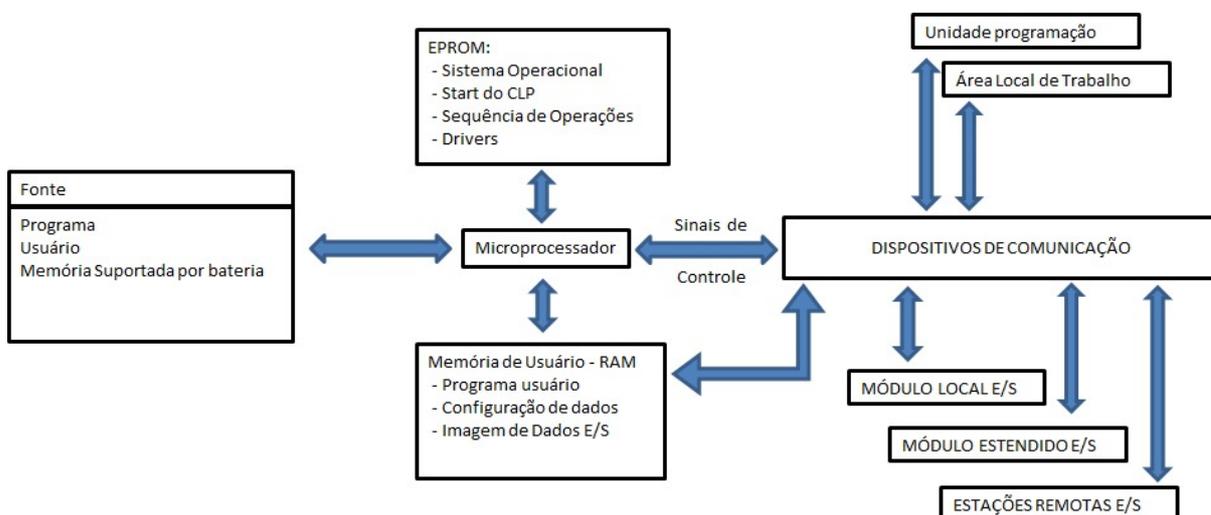


Figura 8 - Diagrama de blocos de um PLC
Fonte: MORAES E CASTRUCCI

O tempo para execução das tarefas é em média de milissegundos até microssegundos, sendo, conhecido como ciclo de varredura ou *scanning*, depende principalmente da velocidade e características do processador utilizado, do tamanho

do programa do usuário e da quantidade e tipo de pontos de E/S. (SILVEIRA E SANTOS, 1998, p.83).

A varredura é processada em ciclo fechado, onde o controlador lê a porta de entrada, grava as informações na imagem de entrada, processa as informações e por fim copia a imagem de saída na porta de saída, sendo que ao inicializar o CLP a imagem das variáveis de entrada é zerada, conforme o diagrama da Figura 9. (MORAES E CASTRUCCI, 2001, p.37).

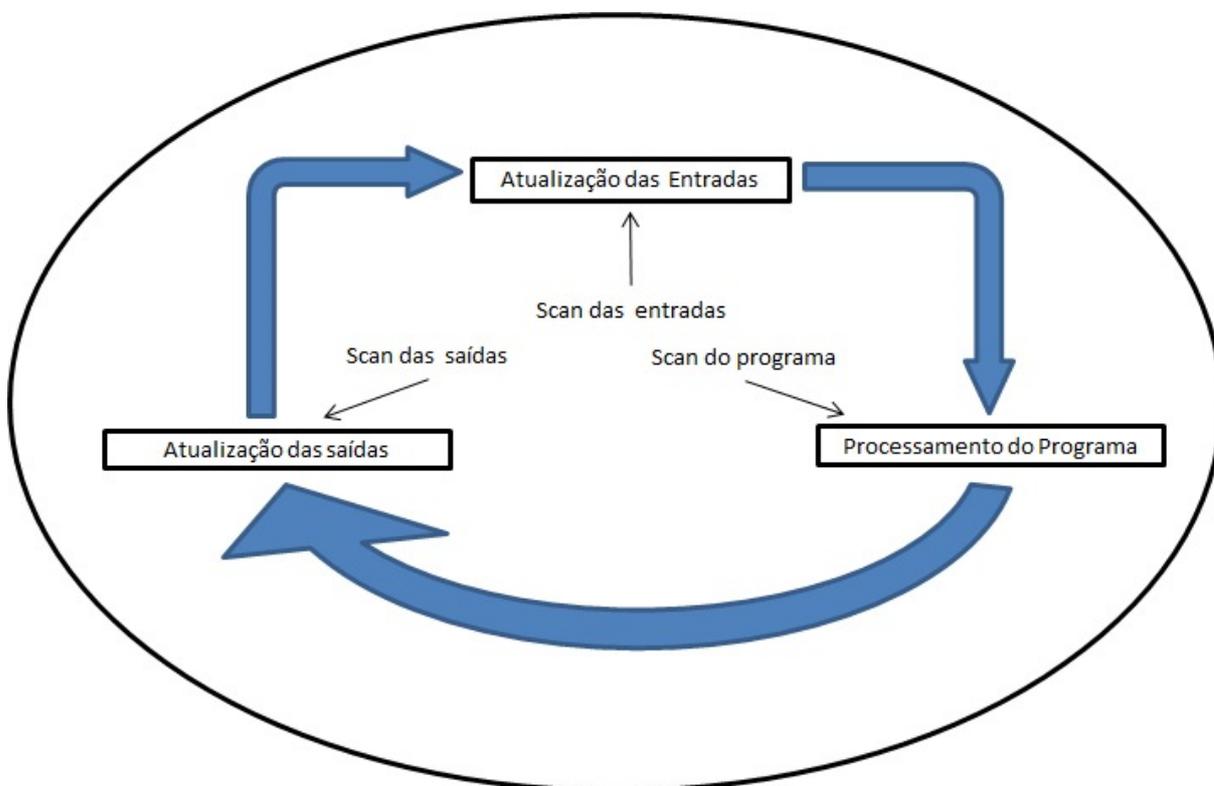


Figura 9 - Ciclo de processamento (scan)
Fonte: MORAES e CASTRUCCI

2.4 LINGUAGENS DE PROGRAMAÇÃO

A norma IEC 1331-3 Programming Languages é a principal norma no que se refere a padronização das linguagens de programação existentes. Para o desenvolvimento de um programa é necessário seguir alguns passos básicos, tais

como, escrever as instruções, editar o programa, verificar e corrigir erros de sintaxe e testá-lo no controlador. (MORAES E CASTRUCCI, 2001, p.43).

A demanda pelos CLPs surgiu da necessidade de otimizar os painéis de controles a relés e foi neste contexto que a linguagem de programação de diagrama de contatos “Ladder” surgiu, com o intuito de facilitar a programação pelos técnicos e engenheiros já familiarizados com esta lógica (SILVEIRA E SANTOS, 1998, p.92).

Posteriormente devido as diversas atualizações e melhorias as quais os CLPs passaram, com seus novos processadores, utilizando novos algoritmos, surgiu a necessidade da implementação de novas linguagens de programação para atendimento as particularidades de funções as quais o diagrama de contatos não abrangia, predominantemente as linguagens de programação são: diagramas de contatos, blocos funcionais, mnemônicos booleanos e parâmetros idiomáticos. (SILVEIRA E SANTOS, 1998, p.92 e 93).

2.5 FABRICANTES

Para especificação de um CLP é necessário levar em consideração a compatibilidade entre a instalação elétrica, equipamentos eletromecânicos, os sinais aceitáveis e os pontos de E/S, a existência de chaves de proteção, tipo e forma de endereçamento, entre outros. (MORAES E CASTRUCCI, 2001, p.39).

Apresenta-se, a seguir, o modelo de CLP S7-300 utilizado neste trabalho e suas principais características.

Fabricado pela Siemens, é uma família de CLPs modulares de pequeno porte, possui UCPs de diferentes níveis de performance, tempo de varredura de 0,06 ms/kbits dependendo do modelo, expansível até 32 módulos, possui interface de comunicação MPI, PROFIBUS e Industrial Ethernet. (SIEMENS (1), 2016, pag. 1-37).

A arquitetura modular do S7-300 traz economia de espaço, flexibilidade de configuração e rápida expansão, instalado diretamente sobre trilho DIN padrão. (SIEMENS (2), 2016).

A Figura 10 é um exemplo de CLP Siemens S7-300, CPU 313C.



Figura 10 - CLP Siemens S7-300, CPU 313C
Fonte: Siemens

Esse modelo de CPL possibilita a configuração para aplicações como por exemplo a indústria automobilística, construção de máquinas especializadas e em série, processamento de plásticos, indústria de embalagens, indústria alimentícia e de cigarros, saneamento, automação predial entre outros, suas principais características são: (SIEMENS (1), 2016, p. 1-37).

- Módulo de E/S (SM) – Digitais ou analógicas.
- Módulo de Comunicação (CP) – Profibus DP/FMS, Ethernet, AS-Interface, Serial Ponto-a-ponto ou Modbus.
- Módulo de Função (FM) – Contadores rápidos, saídas de pulso rápido, controle de posição, controle de motor de passo, controle em malha fechada (PID).
- Possibilita a combinação de até 32 módulos.

3 REDES INDUSTRIAIS

Neste capítulo, aborda-se uma fundamentação teórica referentes aos conceitos básicos de Redes Industriais, os quais serão utilizados posteriormente, onde também alguns detalhes adicionais serão comentados.

Com a crescente necessidade se de conectar computadores e CLP, surgiram as Redes Industriais. Essa interligação em rede permitiu o compartilhamento de

recursos e bases de dados, que passaram a ser único, proporcionando mais segurança aos usuários da informação.

As Redes Industriais, para Regazzi, (2005, p. III.1 – 5) “representam um importante meio de integração. Pode ser reconhecida por seus atributos físicos bem como pelas suas estruturas de dados especiais definidos pelos protocolos.”

Uma rede industrial é um sistema de comunicação bi-direcional, em tempo real, que permite a troca de informação digital entre dispositivos como: sensores, atuadores e controladores.

As redes industriais são, em geral, física e eletronicamente mais robustas que as redes de computadores convencionais. O cabeamento também tende a ser mais imune a ruídos. Além disso, os sistemas de conexão usados nas redes industriais são mais resistentes a desconexões acidentais e a condições ambientais. (REGAZZI, 2005, p. III.1 – 5)

Na Figura 11, pode-se se verificar os diferentes tipos de redes com base no tipo de instrumento e no tipo de controle.

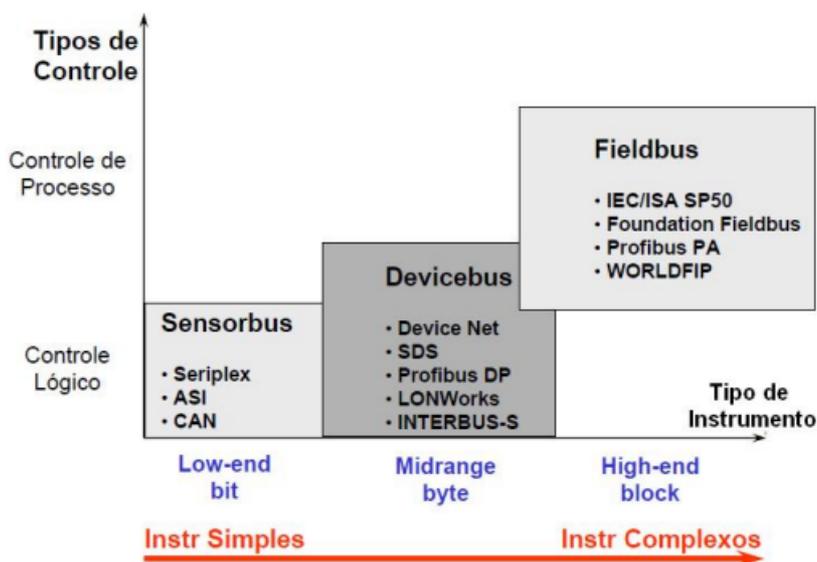


Figura 11 - Redes de Comunicação Industrial
Fonte: GUEDES, 2005

As apostilas CLP e Redes Industriais do SENAI-DN apresentam os três grupos de redes da Figura 2 com as seguintes características:

- **SensorBus:** Efetua a comunicação entre os níveis 1 e 2 da Pirâmide de Automação (sensores e atuadores com dispositivos de controle). É uma rede de baixo custo, que envia mensagens curtas (bits), possuem uma configuração fixa,

não são intrinsecamente seguras, possuem par trançado como meio físico de transmissão e permite-se uma distância máxima de 500 metros.

- **DeviceBus:** Efetua a comunicação entre os níveis 2 e 3 (dispositivos de controle e sistemas de supervisão). É uma rede que envia *bytes* ou *words* e possui uma frequência de comunicação de dezenas de milissegundos.

- **FieldBus:** Trabalha com dispositivos inteligentes realizando a comunicação entre os níveis 3 e 4 (sistemas de produção e gerenciamento de produção). Esta rede envia e recebe mensagens de dados de *words* ou blocos e possui uma frequência de comunicação de centenas de milissegundos.

Além destes três grupos, existe também o Data Bus. Este é responsável pela comunicação dos dois níveis superiores da pirâmide. Realiza uma transferência maciça de dados entre equipamentos utilizando mensagens de blocos ou arquivos. Sua frequência de comunicação é de segundos ou minutos e pode ser feita para longas distâncias (LAN / WAN / Internet).

Ainda segundo Regazzi (2005, p. III.1 – 5), as principais redes industriais do mercado são:

- AS-I: “é classificada como sensorbus e oferece muitos benefícios das redes industriais com menores custos e simplicidade de instalação. É baseada em tecnologia digital e otimizada para dispositivos binários”.

- DeviceNet: “é classificada como sensorbus. É baseada em tecnologia digital e cada nó da rede pode ser controlador, sensor ou atuador. Por ser uma rede produtor/ suporta múltiplos tipos de comunicação e priorização de mensagens.”

- Profibus: “é classificada como fieldbus. Consiste de um sistema multimestre com operação mútua de sistemas de automação, engenharia e visualização. Os mestres, também denominados de dispositivos ativos, definem o tráfego da rede.”

- Interbus: “é classificada como sensorbus. Foi desenvolvida para ser uma rede industrial rápida. A topologia da rede é em anel, o que permite excelente característica de temporização cíclica.”

- FieldBus Foundation: “é classificado como uma rede fieldbus. A rede Fieldbus Foundation é totalmente digital, serial e permite a comunicação bidirecional entre sensores, atuadores e controladores.” Complementa ainda: “é uma rede local (LAN) para automação e instrumentação de controle de processos, com capacidade de distribuir o controle no campo.”

- Modbus: “é um protocolo na camada de aplicação do modelo OSI que disponibiliza um serviço de comunicação cliente/servidor entre dois dispositivos conectados a uma rede.” Além disso, Regazzi afirma que “é um

protocolo do tipo consulta/resposta que fornece serviços específicos através de códigos de função. ”

- LonWorks: “pode ser classificada como devicebus. Foi desenvolvida pela Echelon e sua principal aplicação é em automação predial. Existem diversas aplicações usando LonWorks em sistemas de ventilação, aquecimento e ar-condicionado pelo mundo, inclusive no Brasil, permite um número elevado de nós (pontos). ”

Estes diferentes tipos de rede vêm para atender as específicas necessidades da indústria no que diz respeito à comunicação entre os elementos.

3.1 AUTOMAÇÃO

A automação é a implantação de sistemas interligados por redes de comunicação e compreende sistemas supervisórios, IHM (Interface Homem-Máquina), com o objetivo de supervisionar e analisar os eventuais problemas no processo, sendo o CLP o equipamento fundamental (MORAES E CASTRUCCI, 2001, p. 16).

Os controles de processos industriais e a automação da manufatura é sem dúvida uma das aplicações de maior impacto. É também onde se alcançou maior sucesso comercial do CLP e pode ser empregado em qualquer tipo de sistema que se deseja tornar automático - inteligente (NATALE, 2000, p. 12).

Para Silveira e Santos (1998, p. 23), automação industrial é oferecer e gerenciar soluções, pois ela sai do nível de chão de fábrica para voltar seu foco para o gerenciamento da informação.

Como a automação atua em muitas áreas abrangendo inúmeras funcionalidades, criou-se a Pirâmide de Automação. É uma maneira de representar a hierarquia dos níveis de automação. Segundo Eduardo Bento Pereira, professor da Universidade Federal de São João Del-Rei, esta hierarquia é dividida em cinco níveis:

Nível 1 – Dispositivos de Campo, Sensores e Atuadores: Sensores (que podem ser digitais ou analógicos) e atuadores (por exemplo: válvulas, cilindros, inversores de frequência ou servomotores);

Nível 2 – Controle: tais como CLP, PC, CNC ou SDCD;

Nível 3 – Supervisão (exemplo: Workstation, PC ou IHM);

Nível 4 – Gerenciamento de Planta (exemplo: Workstation);

Nível 5 – Gerenciamento Corporativo (ex.: Mainframe).

A Pirâmide de Automação está representada na Figura 12 hierarquizando estes 5 níveis.

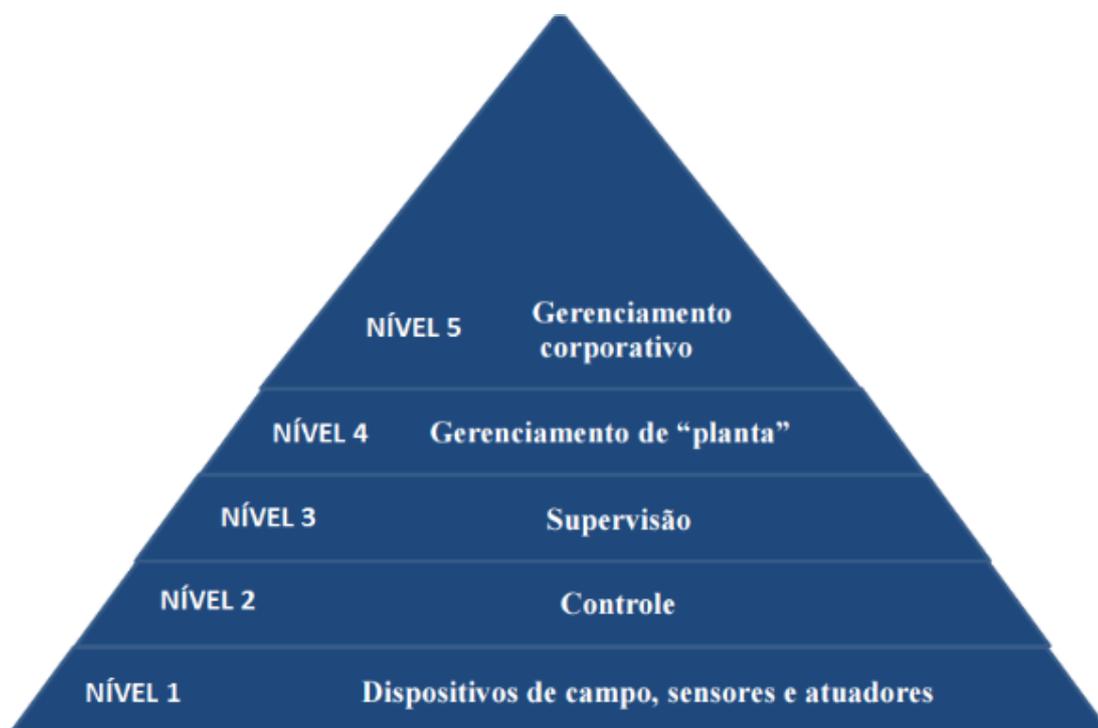


Figura 12 - Pirâmide de Automação (reeditada)
Fonte: BENTO, 2013

3.2 TIPOS DE COMUNICAÇÃO

A seguir são descritos dois tipos de comunicação: paralela e série; bem como suas vantagens.

3.2.1 Comunicação Paralela

Este tipo de transmissão de dados é mais custoso e complexo que a Comunicação Série pois, necessita de mais de um canal de comunicação. Além disso, possui uma baixa imunidade a ruídos e é utilizada em curtas distâncias.

A vantagem da Comunicação Paralela é que ela possui uma velocidade de transmissão de dados maior que a série, justamente por possuir mais de um canal de comunicação. (BETINI, 2014)

3.2.2 Comunicação Série

Já este tipo de transmissão atende a necessidade de comunicação com equipamentos a uma maior distância. Entretanto, possui apenas um canal de transmissão, desta maneira, a comunicação é mais imune a ruídos e tem menor custo de instalação. (BETINI, 2014)

3.1.2.1. Comunicação Serial Síncrona

Neste tipo de comunicação serial, um sinal de *clock* é associado com o dado. (BETINI, 2014)

3.1.2.2. Comunicação Serial Assíncrona

Já na transmissão assíncrona, não existe sincronismo entre transmissor e receptor. A re-sincronização é feita caractere por caractere. (BETINI, 2014)

3.3 TOPOLOGIA DE REDE

Os dispositivos de uma rede podem ser arranjados de diversas maneiras dependendo do protocolo de comunicação adotado. Estrela, anel, barramento e árvores são exemplos de topologias de rede.

Neste trabalho, será adotada a topologia em barramento, atendendo ao padrão RS-485 (detalhado no ítem 3.7), conforme especificado.

Na Figura 13, estão representadas as Estações de Trabalho (ET) enviando e recebendo dados a partir de suas Interfaces de Acesso ao Meio (IAM), que estão conectadas ao barramento.

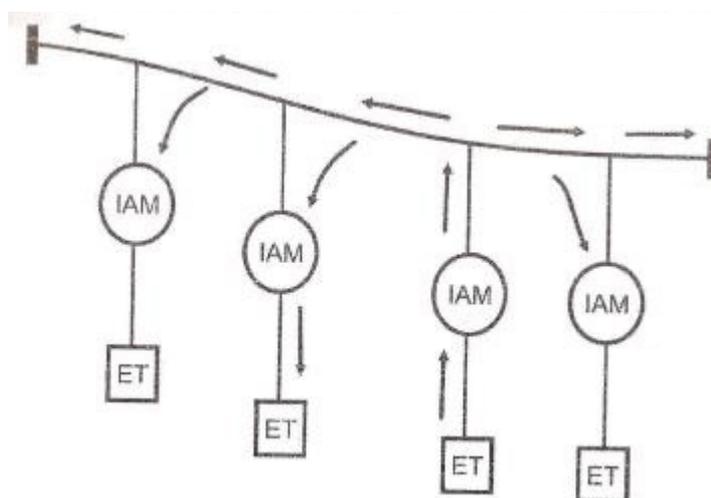


Figura 13 - Exemplo de Topologia em Barramento
Fonte: BETINI

A vantagem desta topologia é uma fácil e simples instalação necessitando de menos cabos. Com isso, a compreensão e entendimento da rede também se torna fácil (BETINI, p. 43). Além disso, os dados podem ser transmitidos em ambos sentidos.

A desvantagem é que, com o uso intenso da rede, ela fica mais lenta. Assim a de Barramento também apresenta dificuldades em se isolar um possível problema (BETINI, p. 43).

3.4 TECNOLOGIA PROFIBUS

Neste item, detalha-se a tecnologia de redes industriais Profibus aderida neste trabalho.

As redes de campo estão sendo utilizadas cada vez mais como o sistema de comunicação entre sistemas de automação e dispositivos de campo. A experiência tem mostrado que o uso da tecnologia de rede de campo pode economizar cerca de 40% dos custos de instalação, configuração, e manutenção da fiação em relação à tecnologia convencional. Nas redes de campo, apenas um par de fios é necessário para transmitir as informações que podem ser dados de entrada ou saída, parâmetros, diagnósticos, programas ou alimentação para os dispositivos de campo.

As redes de campo vêm sendo utilizadas há algum tempo, porém, as primeiras eram específicas e incompatíveis, com elevados custos de configuração ou interfaceamento entre equipamentos diferentes. As novas redes surgidas, oferecem padrões abertos, dispensando interfaces complicadas. Os sistemas abertos deixam o usuário final livre para escolher, entre uma variada gama de produtos, a solução melhor e mais econômica para sua aplicação.

PROFIBUS é a rede de campo líder na Europa, desfrutando de grande aceitação no resto do mundo. Suas áreas de aplicação incluem Manufatura, Controle de Processo e Automação Predial.

PROFIBUS é uma rede de campo aberta, padronizada na Europa, mas de uso internacional, definida na Norma EN 50170. A padronização garante proteção aos investimentos dos fornecedores e dos usuários, além de viabilizar a independência de fornecedor.

Os mais importantes fabricantes mundiais de tecnologia de automação oferecem interfaces PROFIBUS, para seus dispositivos. (ALTUS, 2013)

A tecnologia PROFIBUS foi adotada neste trabalho de conclusão de curso haja vista suas inúmeras aplicações na indústria. Além disso, a Universidade Tecnológica Federal do Paraná conta com novos CLP's que atendem a esta tecnologia.

3.5 A FAMÍLIA PROFIBUS

Ela é dividida basicamente em três categorias: PROFIBUS DP, PROFIBUS PA e PROFIBUS FMS. Cada categoria atende um tipo de comunicação atendendo diferentes níveis da Pirâmide de Automação.

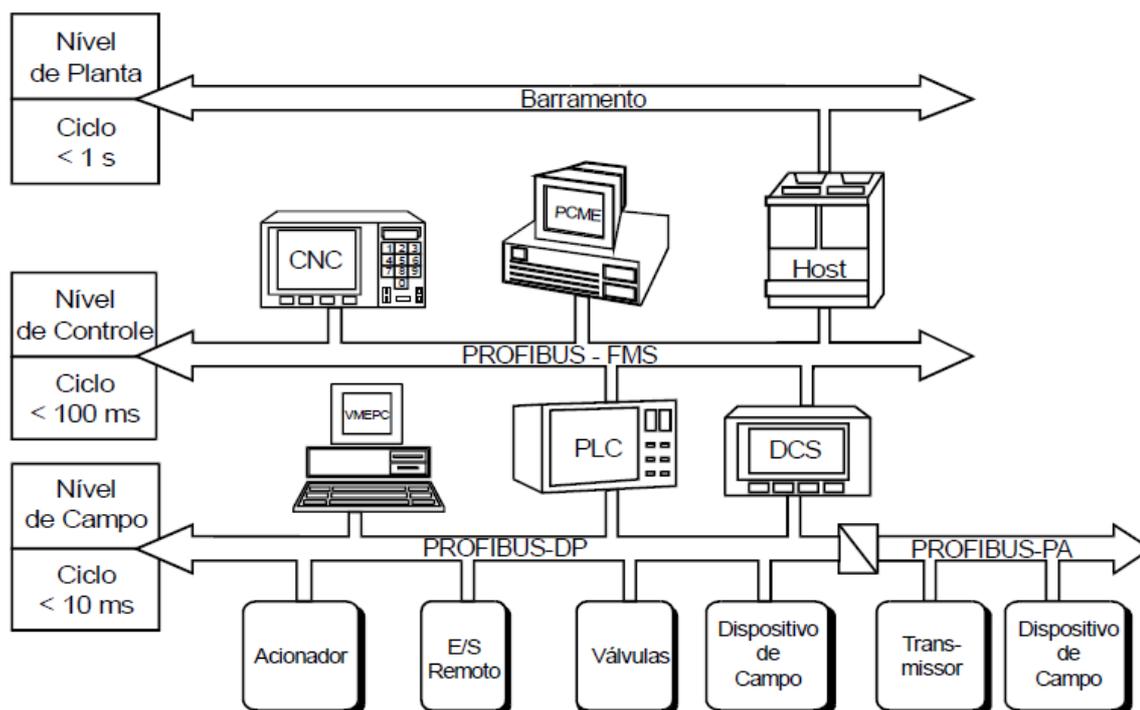


Figura 14 - Áreas de Aplicação do PROFIBUS
Fonte ALTUS, 2013.

Na Figura 14, o padrão PROFIBUS é representado atuando em diferentes níveis de uma hierarquia de uma indústria.

O PROFIBUS é um padrão de rede de campo independente de fornecedor e aberto, aplicável a uma grande gama de aplicações, incluindo Processo, Manufatura e Automação Predial. A independência de fornecedor e a compatibilidade são garantidas pela Norma PROFIBUS EN50170. Com o PROFIBUS, dispositivos de diferentes fabricantes podem comunicar-se entre si, sem modificações nas interfaces. (ALTUS, 2013)

O PROFIBUS DP é otimizado para conexão rápida e barata. Esta versão de PROFIBUS é destinada especialmente para comunicação entre

sistemas de controle de automação e E/S distribuídos. PROFIBUS-DP pode ser usado para substituir transmissão de sinais como 24 Vdc ou 4 a 20 mA. (ALTUS, 2013)

PROFIBUS-PA foi projetado especialmente para aplicação em processos contínuos. Possibilita a conexão de sensores e atuadores em barramento único comum, em áreas intrinsecamente seguras. (ALTUS, 2013)

PROFIBUS-PA possibilita a comunicação de dados e alimentação no mesmo barramento, usando tecnologia a dois fios, de acordo com o padrão internacional IEC 1158-2. (ALTUS, 2013)

PROFIBUS-FMS é uma solução de comunicação genérica. Os serviços FMS abrangem uma larga faixa de aplicações e proporcionam grande flexibilidade. PROFIBUS-FMS geralmente é utilizado a nível de controle. (ALTUS, 2013)

Como a proposta deste trabalho acadêmico refere-se à comunicação entre dois CLP's mestre/escravo (sistemas de controle) via Rede PROFIBUS, a categoria utilizada nele será PROFIBUS DP.

3.6 ARQUITETURA

O protocolo PROFIBUS segue padrões internacionais e, por isso, segue os critérios do modelo OSI. Ele divide uma rede em 7 camadas: Física (Camada 1), Ligação de Dados ou Enlace de Dados, Rede, Transporte, Sessão, Apresentação e Aplicação (Camada 7).

Desta maneira, cada camada efetua tarefas precisamente definidas. Dentre as 7:

- Camada Física: define as características da transmissão física.
- Camada de Transporte: define o protocolo de acesso à rede.
- Camada Aplicação: define as funções de aplicação.

Estas três camadas enunciadas acima são utilizadas no protocolo PROFIBUS-DP, enquanto as demais não são definidas no mesmo.

3.7 1ª CAMADA - TRANSMISSÃO TIA/EIA-485 (RS-485)

O RS-485 é um padrão que especifica apenas as características elétricas dos geradores (drivers) e receptores (receivers) que poderão compor uma rede multiponto. Não define um protocolo de comunicação ou os tipos de conectores (especificação de pinagem) que poderão ser utilizados. Também não especifica outras características, como a qualidade do sinal e a temporização. A definição destes itens fica a cargo do projetista ou de outro padrão que utilize o RS-485 como referência (ELECTRONIC..., 1983, p. 1), como é o caso do Profibus, que é um padrão para redes de controle industriais, e que utiliza as especificações RS-485 como padrão de camada física. (AZEVEDO, p. 23)

Este protocolo é comumente utilizado na indústria onde há sistemas automatizados ligados em rede e, por isso, será objeto de estudo teórico e prático neste trabalho.

O Quadro 1 resume as principais características deste tipo de transmissão.

Característica	Implementação
Topologia da rede	Barramento linear, terminação ativa em ambas as extremidades, extensões de conexão são permitidas em velocidades abaixo de 1,5 Mbit/s
Meio de transmissão	Cabo blindado com par trançado (AL-2303)
Número de estações	32 estações em cada segmento sem repetidores, e 127 estações, com repetidores.
Conectores	Conector 9 pinos tipo D subminiatura (AL-2601e AL-2602)

Quadro 1 - Áreas de Aplicação do PROFIBUS
Fonte ALTUS, 2013.

3.7.1 Interface Diferencial Balanceada

Em um sistema balanceado, a tensão é produzida pelo gerador em um par de linhas onde só pode trafegar um sinal por vez. A diferença entre as tensões nas linhas determinará se o gerador está transmitindo um sinal lógico “1” ou “0”. (AZEVEDO, p. 24)



Figura 15 - Níveis de Sinais em Barramento RS-485
Fonte AZEVEDO, p.24

O gerador (representado na Figura 15 pelo triângulo com a letra D) produz uma tensão diferencial com amplitude entre $\pm 1,5\text{ V}$ e $\pm 6,0\text{ V}$ através de seus terminais (TEXAS..., 2008a, p. 2). Um sinal lógico “1” na entrada do gerador fará com que a linha A se torne positiva em relação à linha B, em sua saída. Para um sinal lógico “0” na entrada do gerador, ocorre exatamente o contrário: a linha B é que se torna mais positiva em relação à linha A. (AZEVEDO, p. 24)

O receptor (também representado na Figura 21 pelo triângulo com a letra R) detecta o sinal transmitido pelo par de fios. Se a diferença de potencial entre as linhas A e B for superior a 200 mV , o receptor reconhecerá um nível lógico “1”. Caso contrário, para uma diferença inferior a -200 mV , será reconhecido um nível “0”. A característica balanceada do sinal, aliada ao acoplamento através de cabos de par trançado, que são indicados para o padrão RS-485, confere à comunicação uma boa rejeição de ruídos. Isso ocorre porque, um sinal de ruído aplicado aos cabos por uma fonte externa, incidirá igualmente nas duas linhas, sendo cancelado pelas entradas diferenciais dos receptores (TEXAS..., 2008a, p. 2).

3.7.2 Operação Multiponto

Um barramento RS-485 é normalmente composto de geradores, receptores, no mínimo um par de linhas de transmissão e resistores de terminação.

Na Figura 16 está representado um típico barramento RS-485 com um mestre (Master) e três escravos (Slave).

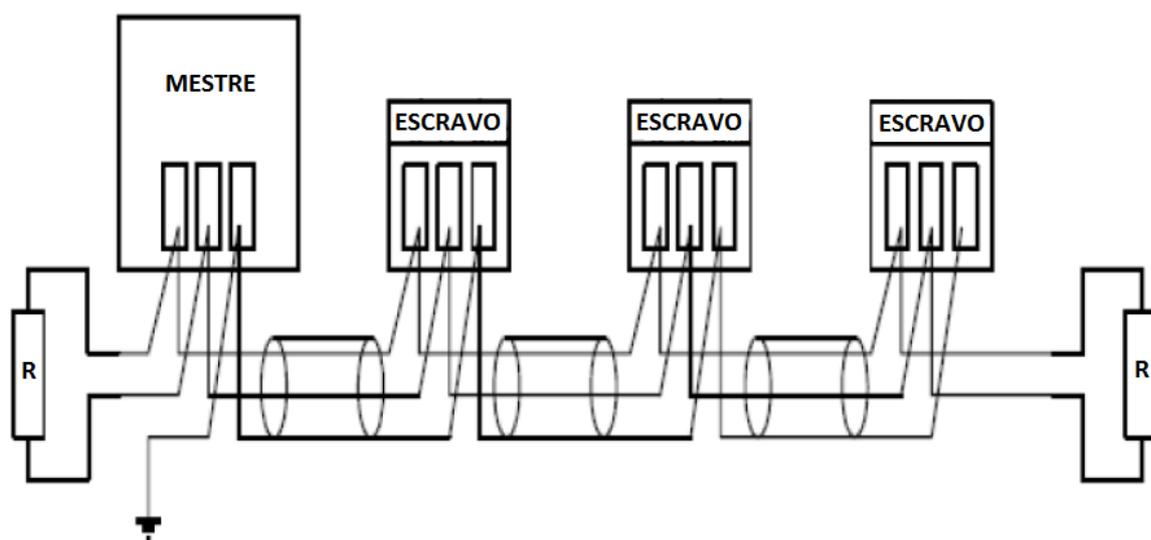


Figura 16 - Barramento RS-485 (reeditada)
Fonte AZEVEDO, p.24

A norma RS-485 especifica que um número máximo de 32 unidades de carga pode participar do compartilhamento.

3.7.3 Vantagens

Este protocolo de comunicação permite a interligação de vários equipamentos que podem se comunicar entre si através do mesmo cabo.

3.7.4 Meio Físico

Há apenas um par de fios (par trançado) para transmissão e recepção que deve ser compartilhado.

3.8 2ª CAMADA – ACESSO À REDE

Ela garante a segurança dos dados, a administração dos protocolos de transmissão e envio dos dados em forma de “telegramas”. (ALTUS, 2013)

Toda a família PROFIBUS (DP, PA e FMS) utiliza o mesmo padrão de protocolo de acesso à rede (protocolo único).

No PROFIBUS, o nível 2 (2ª camada) é chamado “Fieldbus Data Link” (FDL). O Controle de Acesso ao Meio (MAC) especifica o procedimento quando uma estação está autorizada a transmitir dados. O MAC deve assegurar de que apenas uma estação tem o direito de transmitir dados a cada instante. O protocolo PROFIBUS foi projetado para atender a duas principais exigências no controle de acesso ao meio:

- Durante a comunicação entre sistemas de automação (mestres), deve ser assegurado que cada uma dessas estações tenha suficiente tempo para executar suas tarefas de comunicação dentro de um intervalo de tempo precisamente definido.

- A transmissão de dados cíclica, em tempo real tem que ser implementada mais rápida e simples o quanto possível, para a comunicação entre um controlador programável e seus dispositivos de E/S (escravos). (ALTUS, 2013)

Este protocolo de acesso à rede adota o procedimento de passagem de “token” (“passe”) entre os dispositivos mestres (representada na Figura 15). Já os dispositivos escravos utilizam o procedimento mestre-escravo para se comunicarem com os seus respectivos mestres.

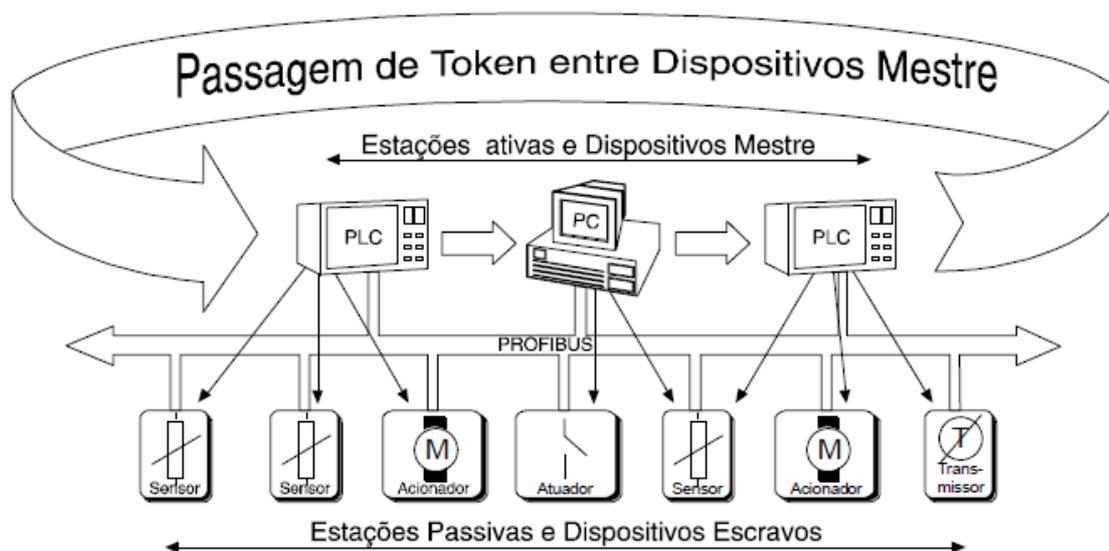


Figura 17 - Passagem de Token entre Mestres
 Fonte ALTUS, 2013, p.10

Na figura 17, o sistema possui três dispositivos mestres e sete dispositivos escravos (cada um associado ao seu mestre).

O procedimento de passagem do “token” garante que o direito de acesso à rede é designado a cada mestre dentro de um espaço de tempo definido. (ALTUS, 2013, p. 10)

Este espaço de tempo pré-definido determina o período de tempo que o mestre tem para conversar com outros mestres e seus escravos.

A mensagem de “token”, (...), deve circular em torno do anel, uma vez para cada mestre, dentro de um tempo de rotação máximo definido. (ALTUS, 2013, p. 10)

A sequência que é passado o “token” é definida a partir do número de IP de cada mestre em ordem crescente. Sendo assim, cada mestre tem seu momento certo de estar ativo durante o ciclo.

Na fase de inicialização do sistema, a tarefa do Controle de Acesso ao Meio (MAC) das estações ativas deve procurar os endereços presentes e estabelecer o anel de “token”. Na fase operacional, as estações ativas defeituosas ou desligadas, devem ser removidas do anel e novas estações ativas devem ser adicionadas. (ALTUS, 2013, p. 10)

O tempo de conservação do “token” em um mestre depende do tempo de rotação configurado. Também a detecção de defeitos do meio de

transmissão ou no receptor, assim como a detecção de erros no endereçamento das estações (por ex: endereços múltiplos), ou na passagem do “token” (tokens múltiplos ou perda dele) são características a cargo do MAC PROFIBUS. (ALTUS, 2013, p. 11)

O nível 2 do PROFIBUS opera no modo sem conexão. Além da lógica de transmissão “peer-to-peer”, proporciona comunicações “multi-peer” (Broadcast ou Multicast). A Comunicação “broadcast” significa que uma estação ativa envia mensagens sem resposta para todas as outras estações (mestres e escravos). (ALTUS, 2013, p. 11)

Na comunicação “Multicast”, uma estação ativa envia mensagens sem resposta para um grupo pré-determinado de estações (mestres e escravos). (ALTUS, 2013, p. 11)

3.9 PROFIBUS DP

Dentro da Família PROFIBUS, será dado um enfoque sobre o PROFIBUS DP que possibilita a existência de sistemas com um único mestre ou vários mestres.

Divide-se os dispositivos em três categorias: Mestre DP Classe 1 (CLP's), Mestre DP Classe 2 (Sistema Supervisórios) e Escravo DP (acionadores, IHM's, válvulas, entre outros).

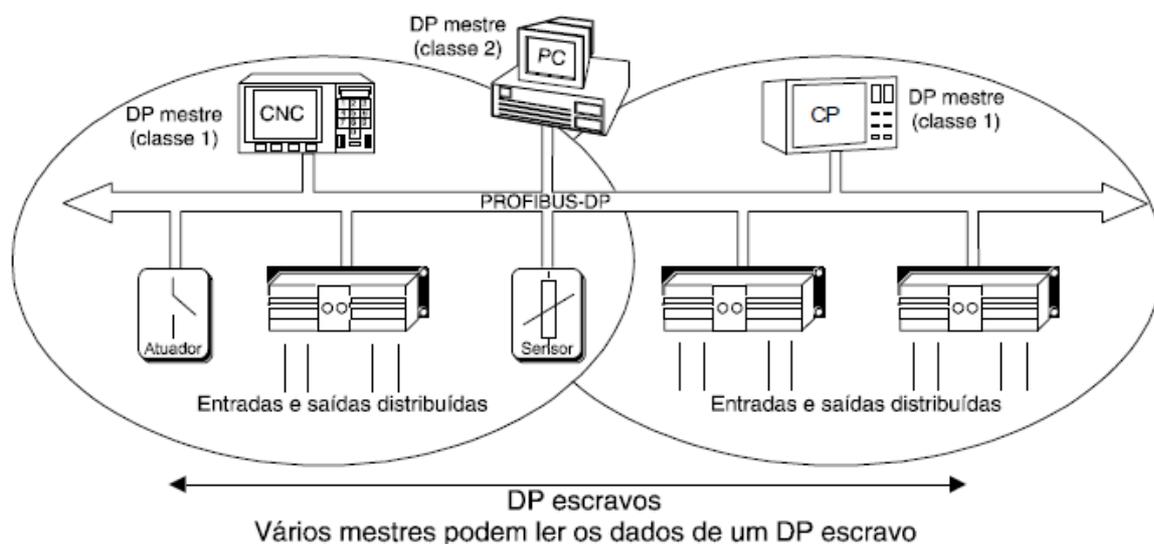


Figura 18 - Protocolo Único de Acesso à Rede
Fonte ALTUS, 2013, p.10

Nas configurações de múltiplos mestres, vários deles estão conectados no barramento. Estes mestres são subsistemas independentes, cada um contém um mestre DPM1 com seus escravos DP, ou uma configuração de dispositivos de diagnóstico (ver Figura 18). As imagens das entradas e saídas dos escravos DP podem ser lidas por todos os mestres.

Entretanto, somente um mestre DP (o DPM1 associado durante a configuração) pode escrever nas saídas. Os Sistemas de múltiplos mestres produzem um tempo de ciclo de barramento maior que os sistemas de mestre único. (ALTUS, 2013, p. 13)

3.9.1 Procedimentos do Sistema

O comportamento do Sistema é composto por três estados básicos:

- Parado (ou “stop”): não há transmissão de dados;
- “*Clear*”: o DPM1 efetua a leitura de entrada dos escravos;
- Operação: o DPM1 transfere os dados (escrita).

3.9.2 Transmissão Cíclica de Dados

A transmissão de dados entre o DPM1 e seus escravos DP é executada automaticamente pelo DPM1, em uma ordem definida. Ao configurar o sistema de rede o usuário especifica as associações dos escravos DP aos respectivos DPM1 e quais escravos DP virão a ser incluídos ou excluídos da transmissão cíclica. (ALTUS, 2013)

Existem três fases que compõe esta transmissão entre DPM1 e escravos DP: parametrização, configuração e transferência de dados.

3.10 PROFIBUS FMS

Esta categoria possibilita a comunicação entre controladores programáveis. Ela possui serviços de acessos a variáveis, transmissão de programas e controle de sua execução e transmissão de eventos.

3.11 PROFIBUS DP E PROFIBUS FMS

3.11.1 Instalação

Abaixo estarão descritos os detalhes de instalação dos dispositivos bem como seus periféricos.

3.11.2 Dispositivos

Na figura 25, representa-se uma rede PROFIBUS composta de:

- Dispositivos com Interface PROFIBUS (1);
- Cabo Especial PROFIBUS (2);
- Conector com Terminação AL-2602 (3);
- Conector AL-2601 (4);
- Repetidor (5).

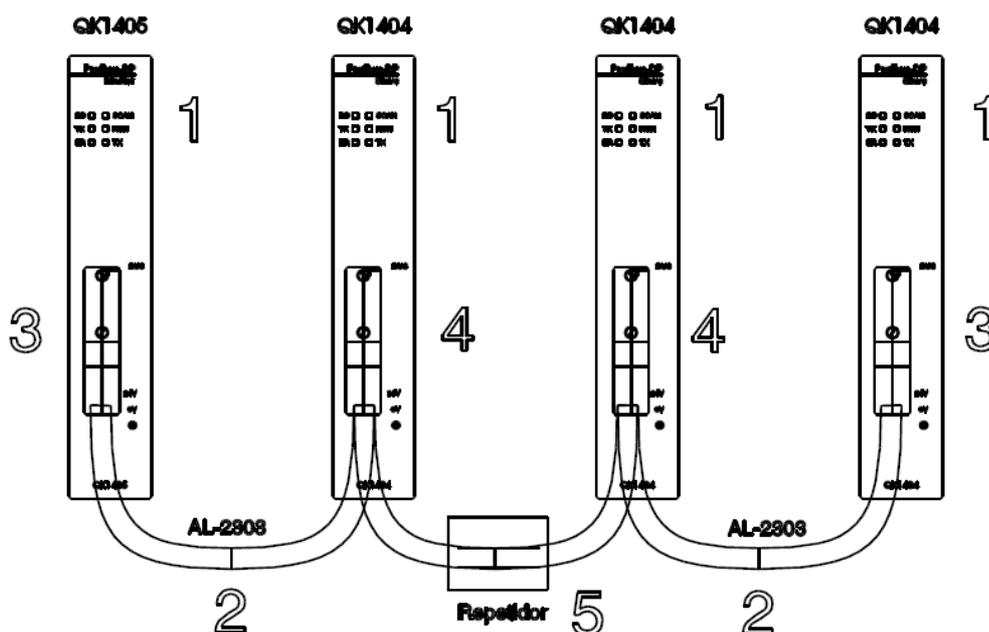


Figura 19 - Conexão em Topologia Barramento
 Fonte ALTUS, 2013, p.10

Todos os dispositivos PROFIBUS devem ser conectados em uma estrutura de barramento (Figura 19). Até 32 estações (mestres ou escravos) podem ser conectadas em um segmento. O barramento é terminado em cada extremidade do segmento por uma terminação. O cabo é separado em segmentos, cada um sendo parafusado ao conector, formando uma cadeia. Esta forma de instalação permite a troca de qualquer módulo durante a operação da rede (troca quente). Para conectar mais de 32 estações, deve ser utilizado um Repetidor. (ALTUS, 2013, p.35)

Segundo a norma CENELEC EN 50170, um máximo de 4 Repetidores são permitidos entre duas estações quaisquer. Entretanto, dependendo do fabricante e das características técnicas do Repetidor, a utilização de uma quantidade maior de Repetidores é possível. Não se recomenda a utilização de um número maior de Repetidores devido aos atrasos que são embutidos na rede e o comprometimento do Slot Time, que consiste no tempo máximo que o Mestre irá esperar uma resposta do Escravo (SILVA).

3.11.3 Cabeamento

A Norma EN50170 define as características do cabeamento utilizado no barramento da Rede PROFIBUS citadas no Quadro 2.

Características do Cabo Tipo A	Especificação
Impedância	135 a 165 Ohms
Capacitância	< 30 Pf/m
Resistividade	110 Ohms/km
Bitola do fio	0.64mm
Área do condutor	> 0.34mm ²

Quadro 2 - Características do Cabo AL-2303
Fonte ALTUS, 2013, p.36

Seu comprimento é limitado de acordo com a velocidade de transmissão adotada (especificações na Quadro 3).

Velocidade de Transmissão [kbit/s]	Tamanho Máximo de um Segmento [m]	Expansão Máxima da Rede (9 Repetidores) [m]
9,6	1200	10000
19,2	1200	10000
93,75	1200	10000
187,5	1000	10000
500	400	4000
1500	200	2000
3000	100	1000
6000	100	1000
12000	100	1000

Quadro 3 - Comprimento máximo para segmento e expansão em redes PROFIBUS utilizando RS-485
Fonte: SILVA

Em altas velocidades de transmissão, o sinal é atenuado mais rapidamente do que em velocidades mais baixas implicando, portanto, em um comprimento máximo menor para os segmentos de rede com velocidades maiores. (SILVA, Lopes)

3.11.4 Conector

Algumas determinações são consideradas a seguir:

- Deve ser utilizado um conector de 9 pinos tipo D para conexões do PROFIBUS.
- Ele conecta cada dispositivo da rede permitindo sua “troca à quente”.
- O cabo chega e sai deste conector sem interromper a rede.
- Um outro tipo de conector que contenha terminação deve ser utilizado nos extremos da rede.

Um exemplo de pinagem de conectores entre duas estações e uma terminação de barramento são apresentados na Figura 20.

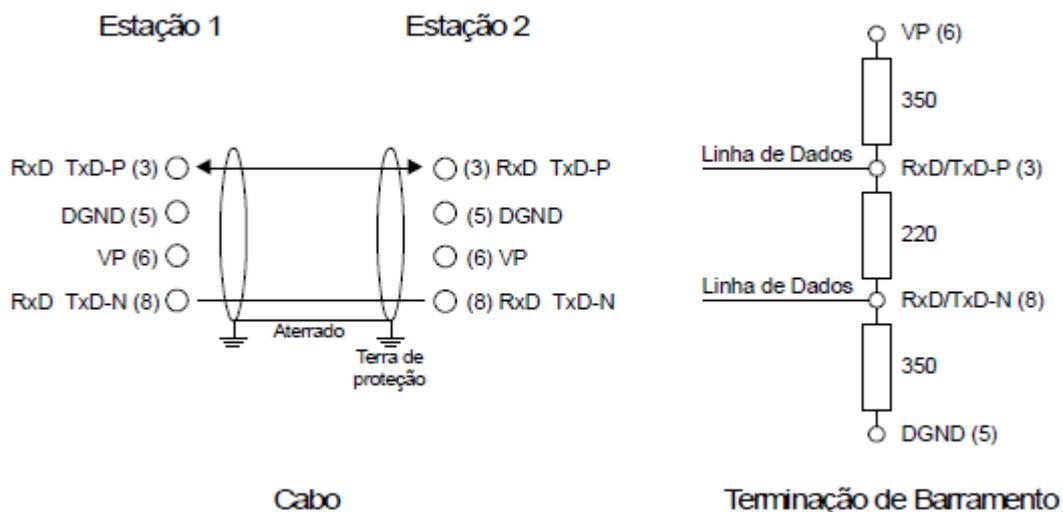


Figura 20 - Pinagem dos Conectores
Fonte ALTUS, 2013, p.36

Cada segmento de uma rede PROFIBUS DP deve possuir uma terminação em cada extremidade. Esta terminação, definida pela Norma EN50170, pode ficar, por conveniência, no conector do cabo, facilitando a instalação e retirada de módulos. As terminações da rede PROFIBUS são *ativas* (recebem alimentação do conector dos dispositivos) (ALTUS, 2013, p.37).

Estas terminações possuem uma resistência no qual evita possíveis reflexões do sinal que podem causar distorção e perda da informação transmitida.

Os resistores de terminação em uma rede RS-485 buscam casar a impedância característica do cabo, que é dependente do seu diâmetro, distância entre os condutores, tipo de isolamento, etc. Estes resistores ajudam absorver a energia do sinal e deste modo impedem a reflexão do mesmo no barramento. Eles possuem o valor de 220 Ohms.

Existem também os resistores de polarização que são utilizados para manter as linhas de sinal RxD/TxD-POS e RxD/TxD-NEG no estado inativo, quando nenhum transmissor está ativo na linha. No PROFIBUS, um dos resistores de polarização é conectado entre a linha de sinal RxD/TxD-POS e a tensão +5Vcc (VP), enquanto que o outro resistor é conectado entre a linha de sinal RxD/TxD-NEG e o terra (DGND). Manter as linhas de sinal no estado inativo impede a recepção de falsos Start Bits pelos receptores, evitando deste modo, a recepção de falsas mensagens. Estes resistores possuem o valor de 390 Ohms (SILVA, Lopes).

Nota-se que na Figura 21 estão representados um resistor de terminação (central no valor de 200 Ohms) e dois de polarização (nas extremidades no valor de 350 Ohms). Estes resistores de polarização variam em torno de 350 a 390 Ohms.

Alguns fabricantes possuem conectores com e sem terminação com formatos idênticos, mudando apenas a cor da carcaça plástica. Pode-se notar os detalhes de um conector na Figura 21.

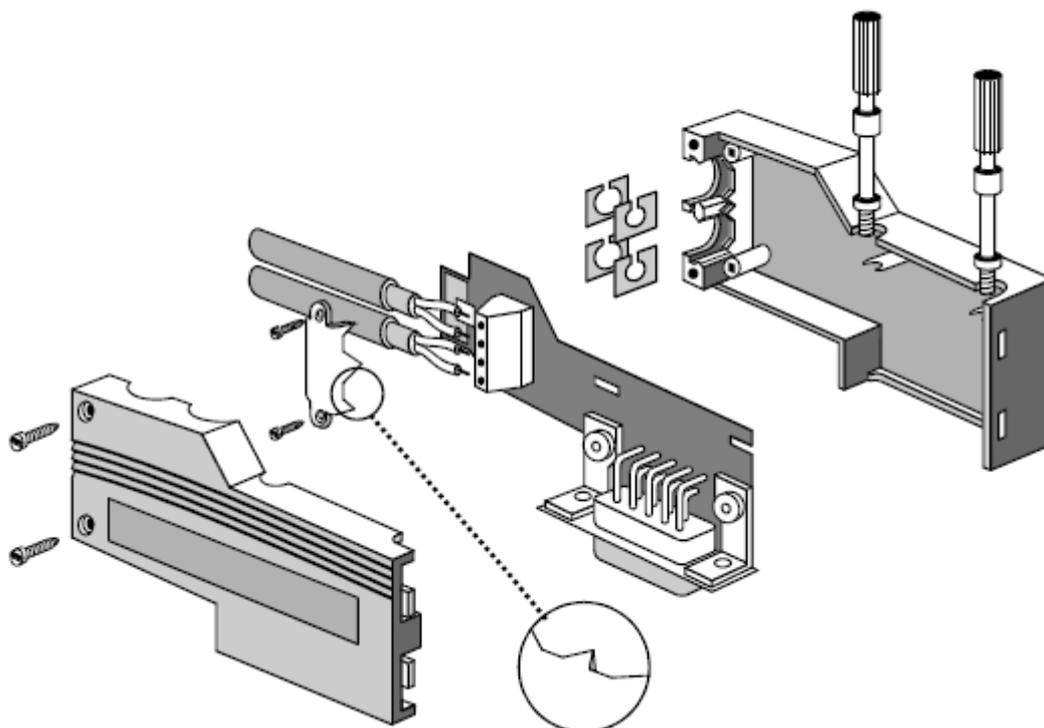


Figura 21 - Vista de um Conector de Rede Profibus
Fonte ALTUS, 2013, p.37

4 SISTEMAS SUPERVISÓRIOS

Nos primeiros dois terços do século XX, segundo Boyle (2009, p.19), a engenharia tinha uma difícil missão para o desenvolvimento de aviões, foguetes, previsão de tempo e outros parâmetros geofísicos, pois a coleta de dados para acompanhamento e monitoramento era difícil ou impossível de ser recolhida.

Um claro exemplo eram os primeiros dias de uma nova aeronave que havia espaço para o piloto, mas pouco espaço para o projeto e engenheiros de teste acompanhar o veículo e monitorar as centenas de sensores instalados para avaliar tensões e deformações na estrutura e no motor (Boyle, 2009, p.19).

Sentiu-se a necessidade de algo que proporcionasse uma atualização de informação em tempo real sobre o estado atual do processo remoto a ser monitorado, assim como a capacidade para manipular o processo remotamente.

No final da década de 60, segundo Shaw (2006, p.3), o circuito integrado foi inventado, tornando-se possível construir complexos e sofisticados dispositivos e

computadores *mainframes*, bem como sistemas capazes de coletar informações em tempo real, sem uso de muito *hardware*, que hoje são denominados de sistemas supervisórios ou SCADA (Supervisory Control and Data Acquisition).

4.1 SCADA – DEFINIÇÃO

Na fabricação moderna e processos industriais, indústrias de mineração, automotiva, petróleo e gás, é muitas vezes necessário conectar equipamentos e sistemas separados por grandes distâncias. Isto pode variar desde alguns metros até milhares de quilômetros.

A telemetria (uso de equipamentos especiais para fazer medições de temperatura, velocidade, pressão, umidade etc., cujos resultados são transmitidos a uma estação distante, onde são armazenados e analisados (MICHAELIS, 2016)), é utilizada para enviar comandos, programas e recebe informações a partir desses locais remotos.

SCADA refere-se à combinação de telemetria e aquisição de dados. Abrange a coleta das informações, transfere-o de volta para o site central, realiza qualquer análise e controle necessário e, em seguida, mostra a informação em telas de operação ou displays. As ações de controle necessárias são então transmitidas de volta para o processo (BAILEY & WRIGHT, 2003, p.13).

Já Thomas & McDonald (2015, p.4), cita que os sistemas SCADA são definidos como um conjunto de equipamentos que irá fornecer um operador em um local remoto com informações suficientes para determinar o estado do equipamento particular ou um processo e causar ações a tomar lugar sobre esse equipamento ou processo sem estar fisicamente presente.

Boyle (2009, p.2), os sistemas de Supervisão e Aquisição de Dados (SCADA) permitem que sejam monitoradas e rastreadas informações de um processo produtivo ou instalação física. Tais informações são coletadas através de equipamentos de aquisição de dados e, em seguida, manipulados, analisados, armazenados e, posteriormente, apresentados ao usuário.

Os sistemas SCADA identificam os *tags*, que são todas as variáveis numéricas ou alfanuméricas envolvidas na aplicação, podendo executar funções computacionais (operações matemáticas, lógicas, com vetores ou strings, etc) ou representar pontos de entrada/saída de dados do processo que está sendo controlado e, com base nos valores dos tags, os dados coletados são apresentados ao usuário (SILVA & SALVADOR, 2011).

4.2 ARQUITETURA SCADA

Krutz (2006, p.7) exemplifica que o sistema SCADA possui cinco elementos em sua arquitetura:

- **Operador:** Operador Humano que monitora o sistema SCADA e executa funções de controle de supervisão para as operações da planta remotos.
- **Interface homem-máquina (IHM):** apresenta dados para o operador e prevê entradas de controle numa variedade de formatos, incluindo gráficos, esquemas, janelas, menus suspensos, telas sensíveis ao toque, etc.
- **Unidade Terminal Master (MTU):** Equivalente a uma unidade mestre em uma arquitetura mestre/escravo. A MTU apresenta dados para o operador através da IHM, reúne dados a partir do local distante, e transmite sinais de controle para o local remoto.
- **Meios de comunicação:** método de comunicação entre a MTU e controladores remotos. A comunicação pode ser através da Internet, redes com ou sem fios, ou a rede telefônica pública comutada.
- **Unidade Terminal Remota (RTU):** Funciona como um escravo na arquitetura mestre/escravo. Envia sinais de controle para o dispositivo controlado, adquire dados a partir desses dispositivos, e transmite os dados para o MTU. Um RTU pode ser um CLP (Krutz, 2006, p.7).

Silva & Salvador (2011) exemplificam que um sistema de supervisão pode ser dividido em componentes físicos e lógicos:

Componentes físicos: sensores e atuadores, rede de comunicação e estações remotas (aquisição/controle) e de monitoração central - sistema computacional SCADA. (SILVA E SALVADOR, 2011). Ilustra-se os componentes na Figura 22.

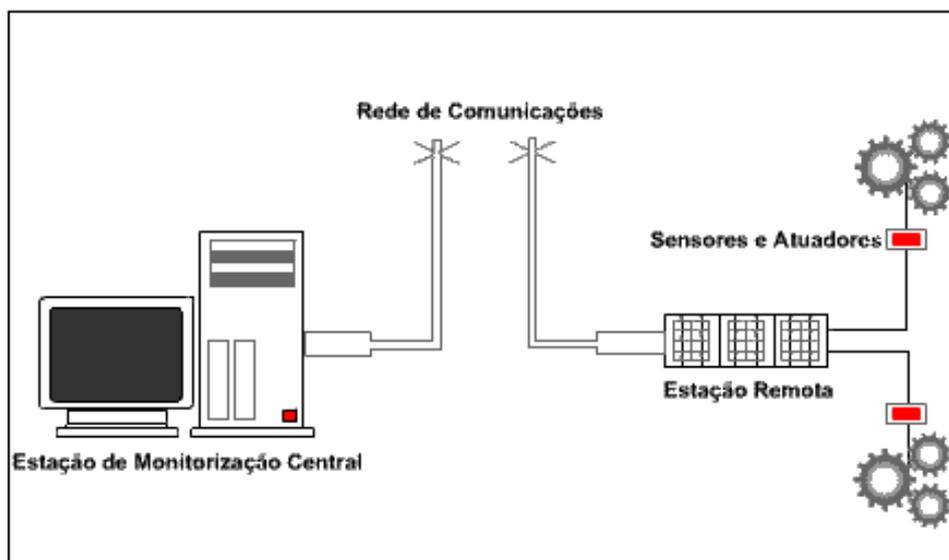


Figura 22 - Sistema de supervisão e controle
 Fonte: Silva e Salvador, 2011

Componentes lógicos: Núcleo de processamento, comunicação com PLCs/RTUs, gerenciamento de alarmes, históricos e banco de dados, lógicas de programação interna (scripts) ou controle, Interface gráfica, relatórios, comunicação com outras estações SCADA e comunicação com sistemas externos/corporativos. Silva & Salvador (2011)

4.3 INTERFACE HUMANO-MÁQUINA

A Interface Humano-Máquina (IHM) surgiu da necessidade de controlar sistemas automatizados complexos de forma amigável, eficiente e ergonômica, equipamento em que o objetivo é permitir a supervisão e comando de plantas automatizadas.

Segundo Moraes e Castrucci (2001, p. 119), há várias aplicações e utilizações para uma IHM:

- Visualização de alarmes gerados por alguma condição anormal do sistema.
- Visualização de dados dos motores e/ou equipamentos de uma linha de produção.
- Visualização de dados de processo da máquina.

- Alteração de parâmetros do processo.
- Operação em modo manual de componentes da máquina.
- Alteração de configurações de equipamentos.

4.4 SISTEMAS SUPERVISÓRIOS EXISTENTES

Sabendo que há um vasto número de sistemas supervisórios existentes no mercado, serão abordados alguns exemplos de *softwares* de supervisão de processos industriais e suas principais características.

4.4.1 Elipse Scada

Produzido pela Elipse Software, a qual possui sua sede em Porto Alegre, é totalmente configurável pelo usuário, permite a monitoração de variáveis em tempo real, através de gráficos e objetos que estão relacionados com as variáveis físicas de campo. Também é possível fazer acionamentos e enviar ou receber informações para equipamentos de aquisição de dados (ELIPSE SOFTWARE, 2015).

O Elipse SCADA está disponível em quatro pacotes diferentes para atender as demandas de personalização dos clientes:

- **Elipse View:** Indicado para aplicações simples, como por exemplo, uma interface com o operador para monitoração e acionamentos;
- **Elipse MMI (Man Machine Interface):** software de supervisão completo. Possui banco de dados proprietário, relatórios formatados, históricos, receitas, alarmes e Controle Estatístico de Processos, facilmente implementáveis;
- **Elipse Pro:** É a mais avançada ferramenta do Elipse SCADA. Permite trocar dados em tempo real com outras estações, transferir ou atualizar bancos de dados, realizar comandos e programar setpoints através de rede local ou linha discada.

- **Elipse SCADA CE:** Este pacote permite executar aplicações Elipse SCADA em dispositivos baseados no sistema operacional Windows CE, como IHMs, dispositivos sem disco em geral e outros dispositivos móveis.

Um exemplo de tela deste sistema de monitoramento pode ser conferido na Figura 23.

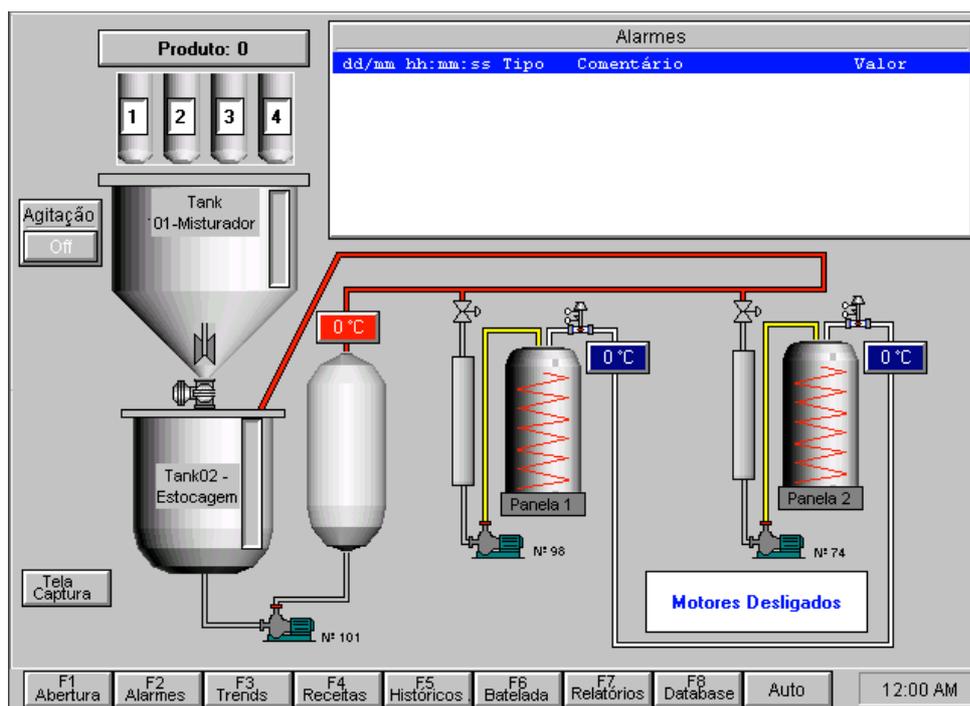


Figura 23 - Elipse SCADA
Fonte: Elipse Software

4.4.2 Elipse E3

Segundo Elipse Software (2016), o Elipse E3 é um sistema de supervisão e controle de processos desenvolvido para atender os atuais requisitos de conectividade, flexibilidade e confiabilidade, sendo ideal para uso em sistemas críticos. Com uma arquitetura de operação em rede que compõe um verdadeiro sistema multicamadas, o software oferece uma plataforma de rápido desenvolvimento de aplicações, alta capacidade de comunicação e garantia de expansão, preservando os investimentos. A solução permite a comunicação com

inúmeros protocolos e equipamentos, podendo acomodar tanto os sistemas locais quanto os geograficamente distribuídos.

Assim como o Elipse SCADA, o Elipse E3 também possui licenças específicas para cada usuário:

- **E3 IHM:** É uma interface humano-máquina para operação local com capacidade de comunicação, mas sem registro de dados em disco. Apresenta versões de até 1.000 tags, com limite de até 4 drivers de comunicação e sem conexão com Bancos de Dados.
- **E3 Lite:** É um pacote com menor número de pontos para pequenas aplicações com registro histórico, de alarmes e relatórios. Possui todos os principais recursos do Elipse E3, porém com menor número de *tags* (20, 50, 150, 250, 500 e 750) e não pode ter *viewers* adicionais.
- **E3 Gateway:** É indicado para aplicações de tradução de protocolos, com versões de 500 a 20 mil tags.

Um exemplo de tela deste sistema de monitoramento pode ser conferido na Figura 24.

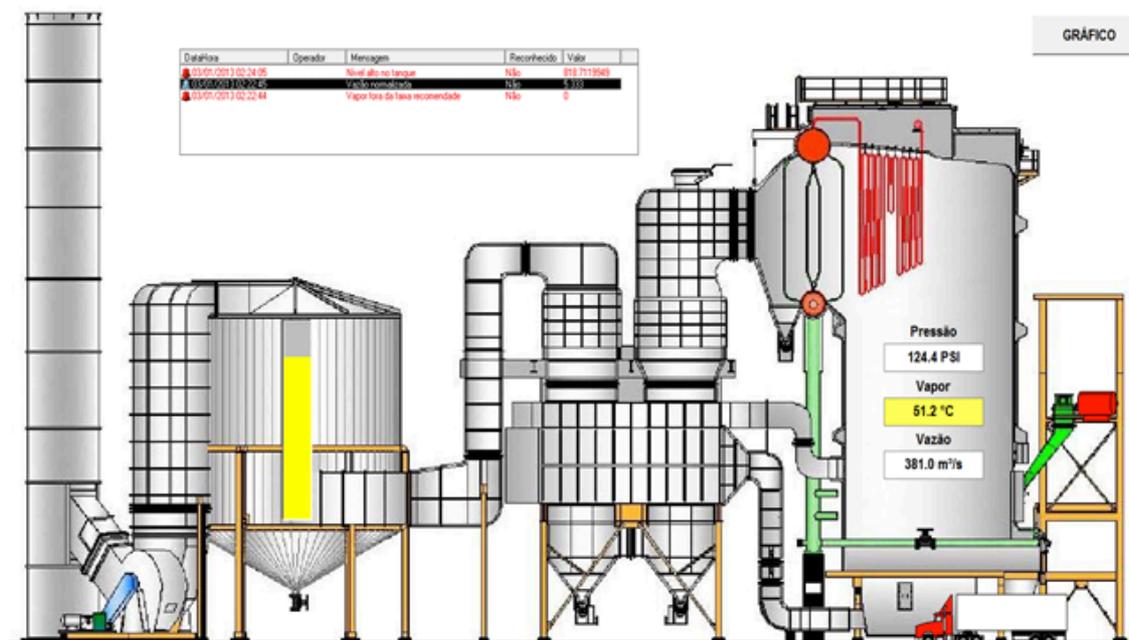


Figura 24 - Elipse E3
Fonte: Elipse E3 Tutorial, 2016

4.4.3 Genesis32

Sistema SCADA criado pela Inconics que oferece ao seu cliente uma interface que incorpora tecnologias como OPC, Visual Basic e ActiveX. Inclui também inúmeras ferramentas para coleta de dados em toda a empresa, registro, gráficos, relatórios e sistemas de análise (ICONICS, 2016).

4.4.4 Labview

Criado pela National Instruments, tendo sua primeira versão para Macintosh em 1986 e conseqüentemente para Windows, Linux e Solaris, o Labview é um ambiente de desenvolvimento integrado, criado especificamente para engenheiros e cientistas.

O LabVIEW é baseado na linguagem de programação gráfica G, que utiliza um modelo de fluxo de dados em vez de uma seqüência de linhas de texto. Dessa forma, pode ser escrito códigos funcionais usando um layout mais visual e intuitivo.

Através dele, é possível criar interfaces para instrumentação virtual, como osciloscópios e geradores de função. Possui um grande número de cartões para aquisição de dados em campo e facilidade para construir aplicativos que podem ser executados em rede ou pela internet (NATIONAL INSTRUMENTS, 2016).

4.4.5 SIMATIC WinCC

SIMATIC WinCC é um Sistema de Controle e Aquisição de dados (SCADA) e, ao mesmo tempo, um sistema de interface homem-máquina produzido pela Siemens.

Segundo o site oficial da Siemens (2016), as áreas de aplicação do SIMATIC WinCC são as seguintes:

- **Visualização de processo para todas as indústrias e tecnologias:** Foi projetado de modo a não ser específico para indústria e tecnologia, ele é modular e pode ser estendido de forma flexível. O WinCC pode ser usado tanto em aplicações para um único usuário como em soluções complexas para múltiplos usuários ou até sistemas distribuídos incluindo vários servidores e clientes – também baseados na Internet – para visualização da planta. WinCC se estabeleceu como um padrão da indústria no campo de visualização de processo, não importando se é pensado como um Sistema SCADA autônomo (Supervisory Control and Data Acquisition) ou

componente HMI de sistemas de controle como SIMATIC PCS 7 ou Spectrum PowerCC para gerenciamento de energia. Um grande número de Opcionais e Add-ons para WinCC permitem criar soluções específicas para cada tipo de indústria, ex: na indústria farmacêutica ou aplicações de tratamento de água.

- **Visualização de processo para implantação global:** A interface de configuração do WinCC foi projetada desde o princípio para implantação global: você pode alternar entre alemão, francês, inglês, espanhol e Italiano apertando apenas um botão. Se você usar a versão ASIA do WinCC você pode alternar entre os quatro idiomas asiáticos: Chinês, Taiwanês, Coreano, Japonês e Inglês. Os textos do aplicativo Runtime podem ser editados e exibidos no idioma que você quiser. Isso quer dizer que você pode usar a mesma solução de visualização em vários mercados alvo.
- **Visualização de processo para qualquer companhia:** Todos os canais de comunicação essenciais para conectar aos controladores SIMATIC S5/ S7/ 505, como para AllenBradley Ethernet IP, Modbus TCP/IP e canais de outros fabricantes como PROFIBUS/PROFINET e OPC já estão incluídos na licença WinCC. Devido ao fato de cada fabricante de controlador também oferecer respectivos servidores OPC, praticamente não existem limites na capacidade de comunicação do WinCC.

SIMATIC WinCC inclui um arquivamento poderoso e escalonável de dados baseado em Microsoft SQL Server, que pode ser utilizado como um hub central de informações para efetiva integração com a área de negócios e de TI. Isso também permite conexão com sistemas MES (Manufacturing Execution Systems) e sistemas ERP (Enterprise Resource Planning) (SIEMENS, 2016).

Para a instalação do software, o hardware deve obedecer aos requisitos conforme o Quadro 4.

Categoria	mínimo XP	mínimo Vista	recomen- dado XP	recomen- dado Vista
Processador	Pentium 3 1GHz	Pentium 42,5GHz	Pentium 4 3GHz	Pentium 4 3,4GHz
RAM (memória principal)	1Gb	2Gb	>1Gb	2Gb
Placa de vídeo	16Mb	128Mb	32Mb	>128Mb
HD	20Gb	20Gb	80Gb	80Gb
Espaço livre em disco para instalação	1,5Gb	>1,5Gb	>10Gb	>10Gb
Espaço livre em disco na partição do sistema	100Mb	100Mb	100Mb	100Mb
Drive DVD/R	sim	sim	sim	sim

Quadro 4 - Requisitos de Hardware
Fonte: Siemens

A seguir será apresentado os principais diferenciais do WinCC com relação aos demais sistemas supervisórios.

4.4.5.1 Gerenciamento Integrado ao Usuário

O WinCC possui o “*SIMATIC Logon*” (figura 31). Com ele, o usuário pode designar e controlar direitos de acessos a usuários para configuração e runtime, ou seja, pode-se designar aos usuários (máximo de 128) direitos apropriados de acesso às funções do software (Figura 25).

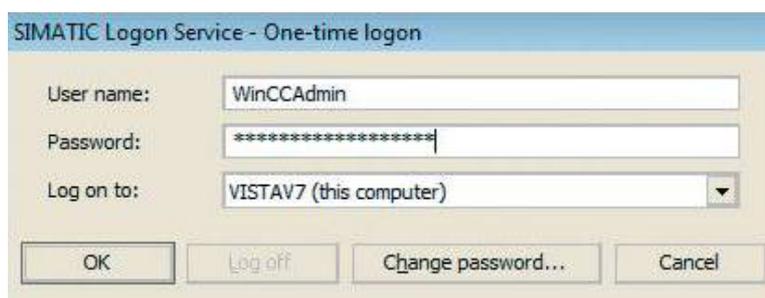


Figura 25 - SIMATIC Logon Service no sistema WinCC.
Fonte: Siemens,2016

4.4.5.2 Interface de Usuário

Usando os padrões WinCC, é possível criar interfaces de usuário com configuração personalizada para qualquer aplicação – para um controle seguro do processo e otimização de toda a sua produção. (SIEMENS, 2016)

Segue a seguir as principais ferramentas:

- Abrangente grupo de objetos padrão configuráveis;
- Botões, Caixas de Marcação e Seleção e Deslizadores;
- Objetos gráficos (Formatos Vector e Bitmap);
- Objetos de tubo (qualquer caminho pode ser mostrado);
- Janelas de imagem e aplicação;
- Objetos OLE, Controles ActiveX (ex: alarme, tendência, controles de tabela);
- Controle de mídia (GIFs animados, AVIs, suporte de .NET e XAML);
- Campos de entrada e saída, listas de texto;
- Barras 2D e 3D, exibições de grupo e status;
- Objetos centrais cambiáveis (*faceplates*);

- Troca de idioma em tempo de execução.

4.4.5.3 Funções de Zoom

O WinCC possui funções inteligentes de zoom, onde é possível representar tamanhos de tela de até 10.000 x 10.000 pixels através do scroll do mouse. (SIEMENS, 2016)

4.4.5.4 Sistema de Arquivamento

SIMATIC WinCC não apenas coleta mensagens de processo e eventos locais, mas também armazena em arquivos circulares e disponibiliza essas informações conforme desejado, de forma filtrada ou organizada. Valores históricos ou progressões de valores são armazenados em arquivos de processo. Além de valores de processo, o WinCC também arquiva mensagens e dados da aplicação.

Foram vários os motivos que levaram o WinCC ser o escolhido para utilizar nesse projeto, entre eles:

- Por ser um software com boa documentação e de fácil acesso a manuais e tutoriais através da internet;
- Pela experiência prévia de um dos componentes da equipe;
- Pouco contato com o WinCC durante a graduação, podendo ser uma importante ferramenta para futuros acadêmicos de engenharia elétrica e engenharia de controle e automação;
- Porque o *software* apresenta todas as funcionalidades necessárias para a construção do sistema supervisorio, sendo também mais simples a comunicação com o Step 7;
- O WinCC é utilizado na maioria das empresas multinacionais como um padrão, bem como empresas de Curitiba e região metropolitana tal como Renault, Volkswagen e Volvo do Brasil.

Na Figura 26, pode-se observar um exemplo prático do Simatic WinCC.

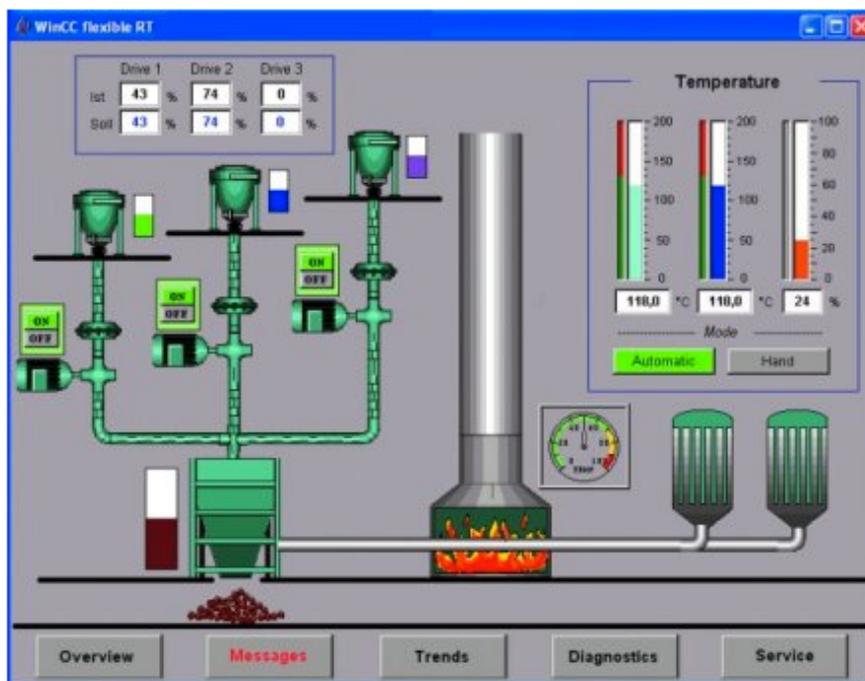


Figura 26 - Exemplo de IHM utilizando o software WinCC.
Fonte: Elec-Intro, 2017.

5 SISTEMA INTEGRADO PROPOSTO

O Sistema Integrado de Controle de Processos proposto é composto por dois notebooks com o software WinCC, os quais trabalham como supervisão do processo do semáforo, dois CLPs Siemens S7-300 e duas IHMs Eaton, que estão inclusas nos kits de CLP do laboratório utilizado na UTFPR.

Cada CLP possui comunicação via PROFIBUS com uma única IHM, nesta IHM consta o botão que representa o botão de pedestre do nosso semáforo, e cada IHM representa um cruzamento de nossa avenida, cujo projeto será apresentado no próximo tópico.

Os CLP's estão interligados pela rede PROFIBUS, e trocam informações como por exemplo sincronismo do semáforo, programação horária e o teste de comunicação.

Cada CLP está interligado por rede MPI a um notebook, o qual terá a função de supervisão do processo através de programa elaborado no software WinCC.

5.1 PROJETO DE COMUNICAÇÃO ENTRE SEMÁFOROS

Para uma demonstração de comunicação em rede Profibus entre os CLP's, foi proposto um projeto que demonstra claramente uma das infinitas possibilidades para utilizar-se uma comunicação: dois cruzamentos que se comunicam entre si em uma estrutura mestre-escravo.

Foi utilizado como exemplo os cruzamentos das ruas Westphalen com a Avenida Sete de Setembro e ruas Westphalen com a Avenida Silva Jardim, ruas localizadas próximas ao campus sede da UTFPR em Curitiba. No entanto, cada semáforo possui 13 lâmpadas, ou seja, cada semáforo possui 6 lâmpadas verdes, 6 lâmpadas vermelhas e uma lâmpada amarela. Deste modo, o controle de um cruzamento é feito através de um CLP, e o controle do outro cruzamento por um segundo CLP.

Foi realizado uma coordenação dos tempos de abertura dos semáforos, supondo uma determinada velocidade da via, ou seja, um tempo hipotético para a abertura do segundo semáforo, em função da abertura do primeiro cruzamento.

Também foi adicionado um sistema de pedestre para auxílio à passagem dos mesmos. Os botões de chamada que possibilita a passagem dos pedestres são simulados nas IHMs Eaton 102, sendo um para cada IHM. Utilizou-se a referência de INOUE E PALAR para realizar a configuração das IHM's. Os detalhes adicionais necessários para configuração delas com o várias estações de trabalho num mesmo barramento estão comentados no Apêndice B.

No software WinCC Flexible, pode ser visualizado os semáforos dos dois cruzamentos, cada cruzamento visualizado em um computador diferente.

Foi previsto um sistema de colocação dos semáforos em alerta, ou seja, as lâmpadas amarelas piscarão a partir de um determinado horário, que pode ser visualizado em tempo real o horário de ativação do sistema, na tela do WinCC

Flexible. Este horário pode ser alterado pelo operador, adicionando o tempo inicial e final.

Deste modo, é possível ter uma integração total dos dois sistemas didáticos com IHM's, comunicação dos CLP's em rede e supervisão do processo.

5.1.1 Caracterização dos estágios

- **Estágio 1 (semáforo 1 em verde estacionário e dinâmico e semáforo 2 em vermelho estacionário):** O semáforo permanecerá em condição estacionária em seu estado inicial, lâmpadas verdes acesas, por 15s. Após esse intervalo de tempo, faz-se o deslocamento de bit até que todas as lâmpadas verdes estejam apagadas. Durante todo este período, as lâmpadas vermelhas do semáforo 2 ficam acesas em estado estacionário.
- **Estágio 2 (semáforo 1 em amarelo e semáforo 2 em vermelho dinâmico):** Neste estágio, teremos o semáforo 1 em amarelo. Neste mesmo instante, tem início o deslocamento de bit do semáforo 2 (lâmpadas vermelhas). Quando ocorrem 4 deslocamentos de bits das lâmpadas vermelhas do semáforo 2, o semáforo 1 ficará em vermelho. Ou seja, antes que o semáforo 2 ative as lâmpadas verdes, o semáforo q já deverá estar em vermelho.
- **Estágio 3 (semáforo 1 em vermelho estacionário e semáforo 2 em verde estacionário e dinâmico):** O semáforo 2 permanecerá em condição estacionária em seu estado inicial, lâmpadas verdes acesas por 15s. Após esse intervalo de tempo faz-se o deslocamento de bit até que todas as lâmpadas verdes estejam apagadas. Durante todo esse período, as lâmpadas vermelhas do semáforo 1 ficam acesas em estado estacionário.
- **Estágio 4 (semáforo 2 em amarelo e semáforo 1 em vermelho dinâmico):** Neste estágio, teremos o semáforo 2 em amarelo. Neste mesmo instante, tem início o deslocamento de bit do semáforo 1 (lâmpadas vermelhas). Quando ocorrem 4 deslocamentos de bits das lâmpadas vermelhas do semáforo 1, o

semáforo 2 ficará em vermelho. Ou seja, antes, que o semáforo 1 ative as lâmpadas verdes, o semáforo 2 já deveria estar em vermelho.

Após o desenvolvimento dos quatro estágios, um novo ciclo terá início.

- **Rotina do pedestre:** A rotina do pedestre será solicitada através do comando de um botão das IHM's. Esta rotina será ativada somente após o término dos estágios 2 e 4. Quando ativada a rotina do pedestre, as lâmpadas vermelhas dos dois semáforos serão acesas e a lâmpada verde do pedestre ativada. Este estado permanecerá por 15s. Após a etapa de verde do pedestre, a lâmpada vermelha do pedestre deverá piscar por 4s.

Ao término desta rotina, o sistema voltará ao seu ciclo natural. Ou seja, caso estivéssemos no primeiro ciclo, (estágios 1 e 2), deverá ser ativo o segundo ciclo (estágios 3 e 4). De outro modo, se estivéssemos no ciclo 2 (estágios 3 e 4) quando a rotina do pedestre foi solicitada, deveremos ativar o primeiro ciclo.

5.1.2 O Projeto

Para a lógica descrita no tópico anterior, foi utilizado um bloco FB em linguagem graph7 (FB1 – Graph Principal). No Graph7, tanto para o mestre, como para o escravo, foram utilizadas 31 etapas. A figura 27 apresenta os blocos utilizados no clp mestre:

Object name	Symbolic name	Created in language	Size in th
System data	---	---	
OB1	Geral	LAD	
OB10	TOD_INT0	LAD	
FB1	Graph Principal	GRAPH	
FC1	Comunicação	LAD	
FC2	Temporizadores Semáforos	LAD	
FC3	Programação Horária	LAD	
FC72	G7_STD_3	STL	
DB1		DB	
DB2	Botão IHM Pedestre	DB	
DB4	Dados_Prog_Horária	DB	
Teste	Teste		
SFC1	READ_CLK	STL	
SFC17	ALARM_SQ	STL	
SFC18	ALARM_S	STL	
SFC52	WR_USMSG	STL	
SFC64	TIME_TCK	STL	

Figura 27 - Blocos utilizados para o projeto de comunicação.
Fonte: Autoria própria, 2017.

No bloco FC2 (Temporizadores Semáforos) foram inseridos todos os temporizadores e as lógicas de auxílio ao graph7. Já o FC1 (Comunicação), o grupo optou por inserir todas as comunicações entre o CLP mestre e o CLP escravo, pois facilita na visualização de entradas e saídas de comunicação.

No bloco FC3 (Programação Horária), está inserido a lógica para que os semáforos pudessem entrar na etapa de “amarelo piscando”, a partir do momento em que o usuário define no sistema supervisor WinCC.

Nos blocos DB2 e DB4 estão inseridos alguns dados necessários para o funcionamento do botão das IHM's e da programação horária. No programa também há uma VAT para fazer testes quando necessário. O bloco OB1 (Geral) contém todas os blocos FB's e FC's citados anteriormente. Caso algum bloco não esteja inserido em OB1, o respectivo bloco não funcionará. Demais blocos foram gerados automaticamente na criação de um bloco em linguagem Graph7.

Para comprovar efetivamente a comunicação entre os dois CLP's, foi utilizado um byte para envio de sinais do mestre para o escravo e outro byte para envio de sinais do escravo para o mestre. Esses bytes foram utilizados para envio das seguintes informações:

- Bits para coordenar os tempos de abertura dos semáforos, sendo que o cruzamento mestre está sempre adiantado alguns segundos (determinado pelo usuário) com relação ao escravo;

- Bits para envio de informação do mestre para o escravo, para o caso de a programação horária estar ativa;
- Bits para envio de informação do mestre para o escravo assim que o mestre receber informação de solicitação do pedestre para o cruzamento escravo.

A figura a 28 apresenta um dos cruzamentos do projeto, que foi desenvolvido no WinCC:

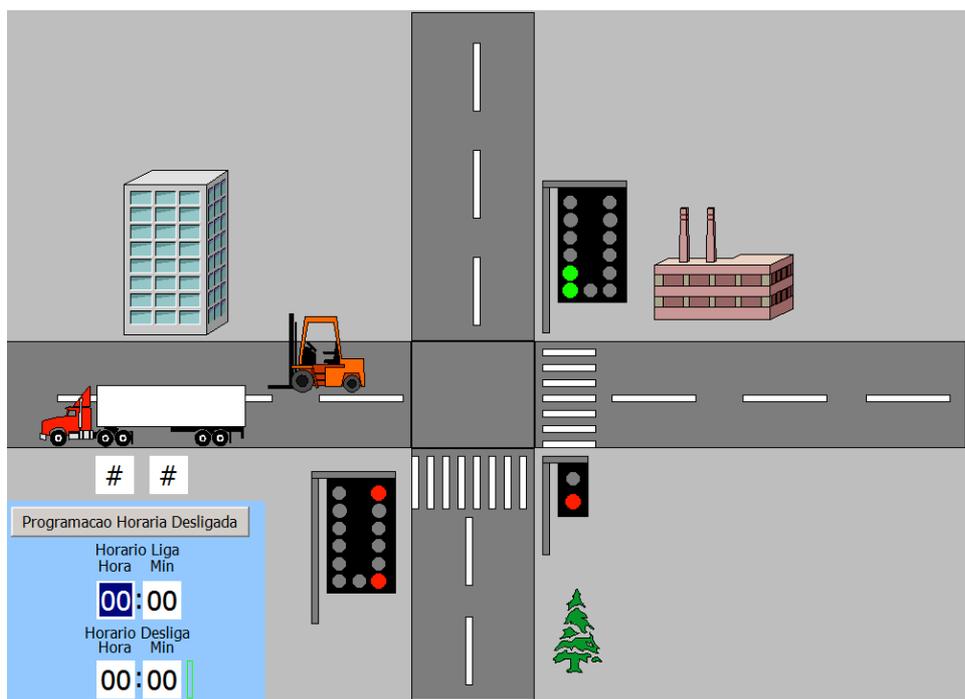


Figura 28 - Cruzamento desenvolvido no software WinCC.
Fonte: Autoria própria, 2017.

A figura 29 apresenta uma das IHM's utilizadas como botoeiras para solicitação de passagem de pedestres. A tela da IHM foi desenvolvida no software Galileo e foi utilizado uma IHM da Eaton, que está inserida nos kits didáticos disponibilizados no laboratório de CLP.



Figura 29 - Botão para solicitação de pedestre desenvolvido no software Galileo.
Fonte: Autoria própria, 2017.

Na figura 30 apresenta-se a parte prática deste projeto, montada como foi proposto anteriormente. À esquerda, inserido no kit didático é apresentado o CLP mestre, juntamente com a IHM de solicitação do pedestre e o cruzamento mestre. À direita, inserido no kit didático é apresentado o CLP escravo, juntamente com a IHM de solicitação do pedestre e o cruzamento escravo.

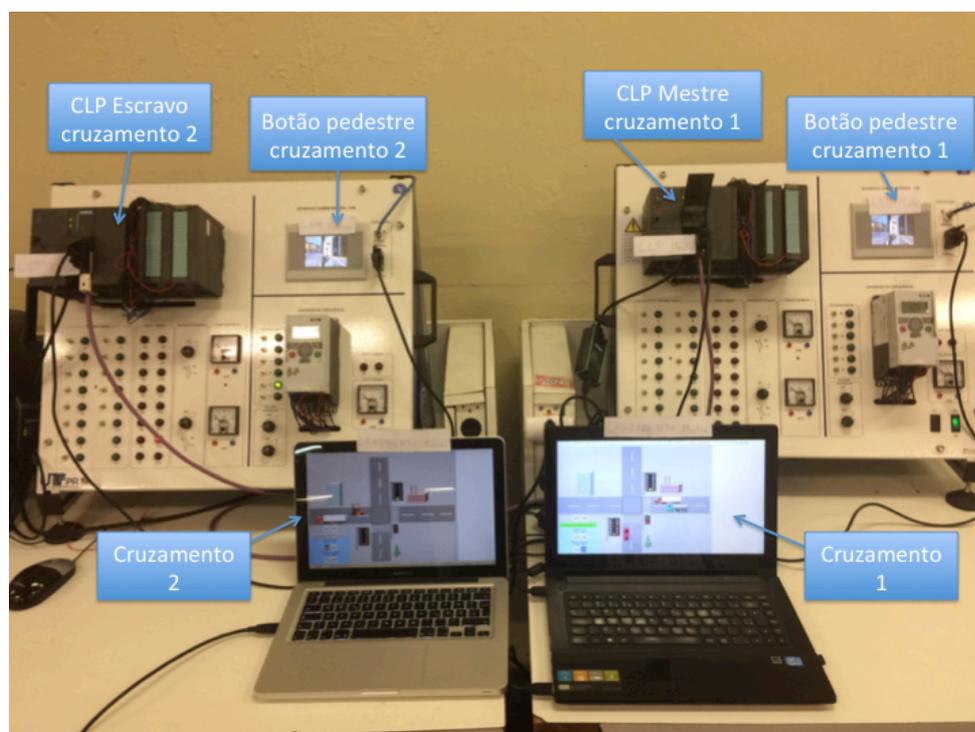


Figura 30 - Projeto de comunicação de CLP's.
Fonte: Autoria própria, 2017.

6 CONCLUSÃO

O desenvolvimento do presente estudo possibilitou uma análise de como configurar a comunicação entre dois CLP's SIEMENS S7 313. Conforme planejado, foi criado um passo-a-passo em pdf no intuito de orientar alunos de graduação de engenharia do DAELT da UTFPR.

A primeira e única dificuldade encontrada entre a comunicação entre dois CPL's foi referente à conexão física entre ambos. O cabo utilizado não estava com os terminais conectados da forma correta exigida pelo padrão RS485. Logo, após algumas tentativas, foi levantada a possibilidade de abrir os conectores e verificar os terminais. Corrigido isto, a comunicação foi efetuada perfeitamente.

Além disso, também foi elaborado um exemplo de aplicação em dois cruzamentos de ruas com o objetivo de ilustrar a sincronização entre semáforos. Neste modelo, a rede era composta por duas IHM's conectadas em barramentos com os dois CPL's (cada um dedicado a um cruzamento).

Durante a elaboração do exemplo citado acima, também houveram algumas dificuldades encontradas na configuração de comunicação entre o CLP Mestre e a IHM do Escravo. Isso se deu por causa de um default (padrão) de comunicação que era carregado na IHM mesmo configurando informações diferentes. A solução do problema foi excluir o default para que fosse considerado apenas o configurado de fato. Estes detalhes específicos de configuração são explicados no Apêndice A como forma de complementar o trabalho utilizado como referência INOUE E PALAR.

Além deste material que será disponibilizado em versão PDF aos alunos das disciplinas de CLP, foi elaborada também uma vídeo-aula para tornar as explicações mais ricas e fáceis de entendimento. Ela será entregue ao professor orientador deste trabalho, Daniel Balieiro Silva, que repassará aos seus alunos e interessados no tema. A vídeo-aula está disponibilizada no *Youtube* no seguinte endereço: <<https://www.youtube.com/watch?v=InGn1VZqwj4>> com o nome "Tutorial - Comunicação entre dois CLP's Siemens S7-300 através de rede Profibus".

Conclui-se que, tanto os objetivos específicos, como o objetivo geral, foram alcançados com sucesso. Os autores deste trabalho se orgulham de fazer parte do aprendizado desta tecnologia, tão amplamente utilizada na indústria, pelos próximos

estudantes e interessados. Espera-se que este material sirva de suporte base e contribua ao meio acadêmico e incentive ainda mais a exploração sobre o tema.

REFERÊNCIAS

ALTUS. **Manual de Utilização da Rede PROFIBUS** (Rev. D 01/2013 - Cód. Doc.: MU299026). Disponível em: <http://www.altus.com.br/ftp/Public/Portugues/Produtos/Quark/05%20Rede/QK1405%20-%20Interface%20Profibus%20Mestre/Manuais%20e%20Apostilas/MU299026.pdf> (acesso: 11/09/2016 às 16h49).

AZEVEDO, Paulo Roberto de. – **Implementação de uma Rede Baseada no Padrão RS-485 para Monitorar No-Breaks Remotamente**, 2009.

BAILEY, David; WRIGHT, Edwin. **Practical Scada for Industry**. Elsevier, 2003.

BENTO, E.. **Notas de Aula**. Disponível em: http://www.ppgel.net.br/ebento/Disciplinas/automacao/Aula_x23.pdf (acesso: 02/05/2016)

BETINI, R. C. Semana 9 – **Histórico e Arquitetura de Redes Industriais, 2014 (Notas de Aula da disciplina de Redes Industriais – UTFPR)**. Disponível em: <http://paginapessoal.utfpr.edu.br/betini/Disciplinas/RI/aulas-em-pdf/RI-S9%20-Modo%20de%20Compatibilidade.pdf/view> (acesso: 09/08/2016 às 15h29)

BOYER, Stuart A. **Scada: Supervisory Control And Data Acquisition**. 4 ed. ISA: The Instrumentation, Systems, and Automation Society, 2009.

CAMARGO, Valter Luis Arlindo. FRANCHI, Claiton Moro. **Controladores lógicos programáveis: sistemas discretos**. 2 ed. São Paulo: Érica, 2009.

CAPELLI, Alexandre. **Automação Industrial: controle do movimento e processos contínuos**. 2 ed. São Paulo: Érica, 2007.

ELEC-INTRO. Disponível em: <http://www.elec-intro.com/cms/plus/view.php?aid=86> (acesso: 04/05/2017 às 14h31)

ELIPSE SOFTWARE. **Elipse Scada Tutorial**. 2015.
http://downloads.elipse.com.br/port/download/scada/v2.29/b151/scadatutorial_br.pdf
(acesso: 10/09/2016)

ELIPSE SOFTWARE. <http://www.elipse.com.br/port/e3.aspx> (acesso: 10/09/2016)

FRANCHI, Claiton Moro; CAMARGO, Valter Luís Arlindo de. **Controladores Lógicos Programáveis – Sistemas Discretos**. 2ª ed. Érica; 2009.

GUEDES, Luiz Affonso. **Redes de Comunicação para Aplicações Industriais**. Disponível em: http://www.dca.ufrn.br/~affonso/PET0303/redes_industriais.pdf
(acesso: 13/05/2016)

ICONICS. Disponível em: <http://www.iconics.com/Home/Products/HMI-SCADA/GENESIS32.aspx#.V9dbbGVPPVo>. (acesso: 10/09/2016)

IEC 1331-3. **Programing Languages**.

INOUE, Alberto Akira. PALAR, Piatan Sfair. **Manual Didático Para A Bancada Do Laboratório De Controladores Lógicos Programáveis Da Utfpr**. 2016.

KRUTZ, Ronald L. **Securing Scada Systems**. Wiley Publishing, 2006.

MORAES, Cícero Couto de. CASTRUCCI, Plínio de Lauro. **Engenharia de Automação Industrial: Hardware e Software, Redes de Petri, Sistemas de Manufatura, Gestão da Automação**. Rio de Janeiro: LTC, 2001.

NATALE, Ferdinando. **Automação Industrial**. 2. ed. rev. e atualizada. São Paulo: Érica, 2000.

NORMA EUROPÉIA, **Cenelec EN 50170**.

PADRÃO RS-485. Disponível em>:
 <http://olaria.ucpel.tche.br/autubi/lib/exe/fetch.php?media=padrao_rs485.pdf>.
 (acesso: 08/08/2016 às 17h46)

REGAZZI, Rogério Dias; PEREIRA, Paulo Sérgio; SILVA JR., Manoel Feliciano da.
Soluções práticas de instrumentação e automação: utilizando a programação gráfica LabVIEW. Rio de Janeiro: [s. n.], 2005.

SENAI DN. **Redes Industriais de Comunicação.** Disponível em:
<http://docslide.com.br/documents/redes-industriais-de-comunicacao-senai.html>
 (acesso 15/05/2016).

SHAW, William T. **Cybersecurity for Scada Systems.** Oklahoma: Penwell, 2006.

SIEMENS (1), **Simatic: PLC S7-300, CPU Specifications CPU 312 IFM to CPU 318-2 DP,** disponível em:
https://cache.industry.siemens.com/dl/files/591/8860591/att_105766/v1/HB_CPU312IFM_bis_318-2DP_e.pdf. (acesso: 06/11/2016 às 20h16)

SIEMENS (2), **Simatic S7 300,** disponível em:
<http://w3.siemens.com.br/automation/br/pt/seguranca-de-maquinas/interfaces-de-seguranca/plc-de-seguranca/simatic-s7-300/pages/simatic-s7-300.aspx>. (acesso em 06/11/2016 às 18h41)

SIEMENS (3). Disponível em: <http://w3.siemens.com.br/automation/br/pt/automacao-e-controle/controladores-simatic/simatic-ihm/simatic-wincc/pages/simatic-wincc.aspx>.
 (acesso: 12/09/2016)

SILVA MELLO, M.. **CLP- CONTROLADOR LÓGICO PROGRAMÁVEL.** Disponível em:
http://www.ice.edu.br/TNX/encontrocomputacao/artigos-internos/aluno_marilourdes_silva_clp.pdf (acesso 13/05/2016)

SILVA, Ana Paula Gonçalves da; SALVADOR, Marcelo. **O Que São Sistemas Supervisórios?.** Disponível em:

<http://kb.elipse.com.br/ptbr/questions/62/O+que+são+sistemas+supervisórios%3F>
(acesso: 09/09/2016).

SILVA, Wladimir Lopes. **Diretrizes para Projeto e Instalação de Redes PROFIBUS DP.** <http://www.profibus.org.br/news/julho2009/news.php?dentro=3>. (acesso: 29/10/2016)

SILVEIRA, Paulo Rogério. WINDERSON, dos Santos. **Automação e controle discreto.** 9 ed. São Paulo: Érica, 1998.

STÉFANO, Antonio Geraldo. **Apostilas do curso Básico e Avançado: Ativa Automação Industrial Manual de utilização do programa Master Tool.** 2007.

TEXAS INSTRUMENTS. 2008a. Disponível em:
<http://www.ti.com/lit/an/slla272b/slla272b.pdf> (acesso: 08/08/2016 às 17h26)

THOMAS, Mini S.; MCDONALD, John D. **Power System Scada and Smart Grids.** CRC Press, 2015.

APÊNDICE A

Este apêndice tem por objetivo detalhar a configuração da comunicação via rede Profibus entre dois CLP's Siemens S7-313 disponíveis no laboratório da UTFPR Curitiba.

Para criar um sistema de comunicação mestre-escravo entre dois CLP's Siemens S7-300, seguir os seguintes passos no step7:

Inserir as estações individualmente no programa, neste exemplo denominadas de estações mestre e escravo conforme podemos verificar na figura 31.

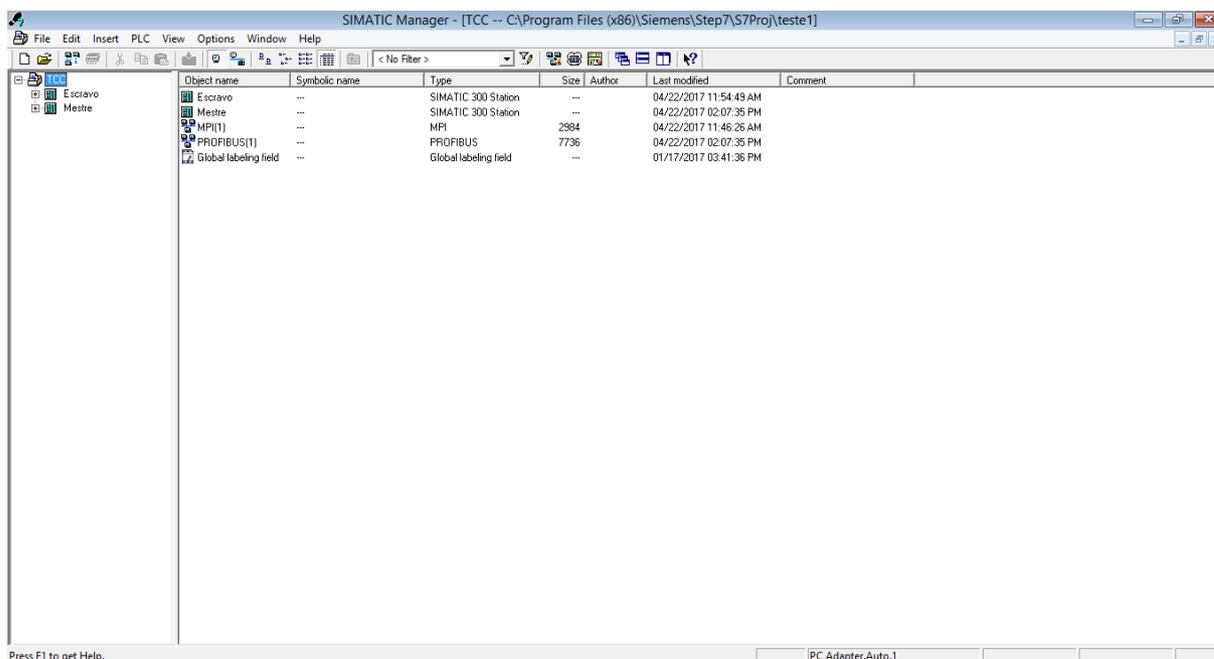


Figura 31 - Tela Inicial com a estações mestre e escravo.

Fonte: Autoria própria, 2017.

Realizar a configuração de hardware, primeiramente para a estação Mestre, conforme mostra a figura 32, clique uma vez no ícone da estação Mestre.

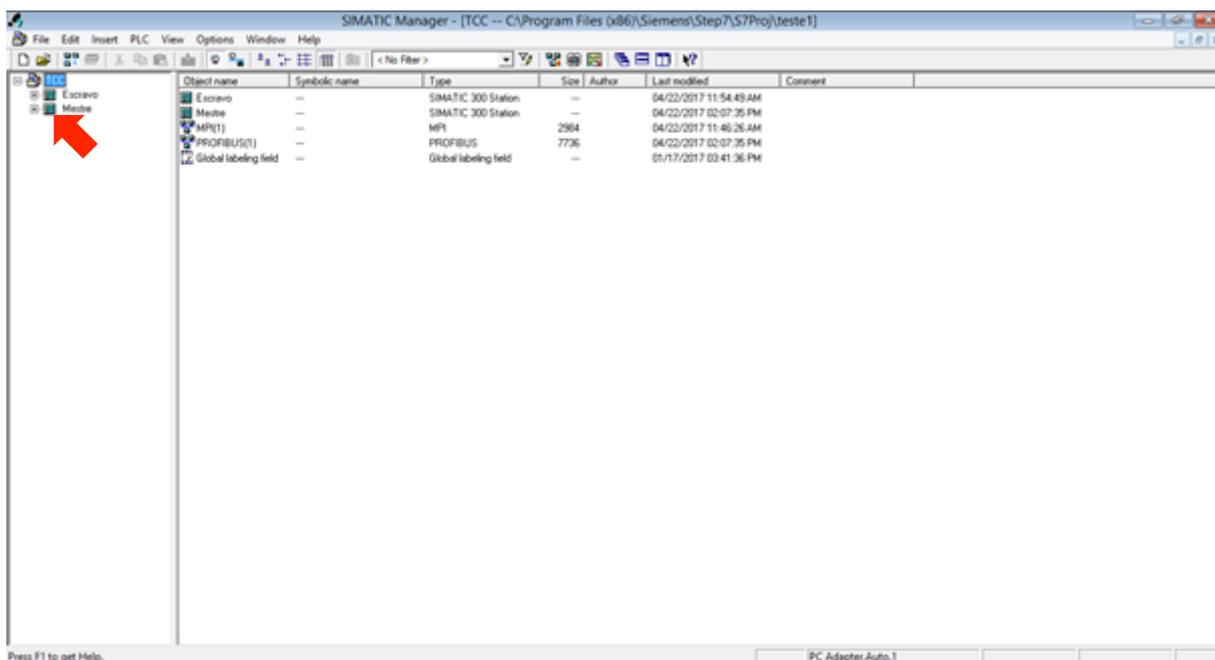


Figura 32 – Abrir projeto estação mestre.
Fonte: Autoria própria, 2017.

Clicar duas vezes no ícone de hardware, conforme a figura 33.

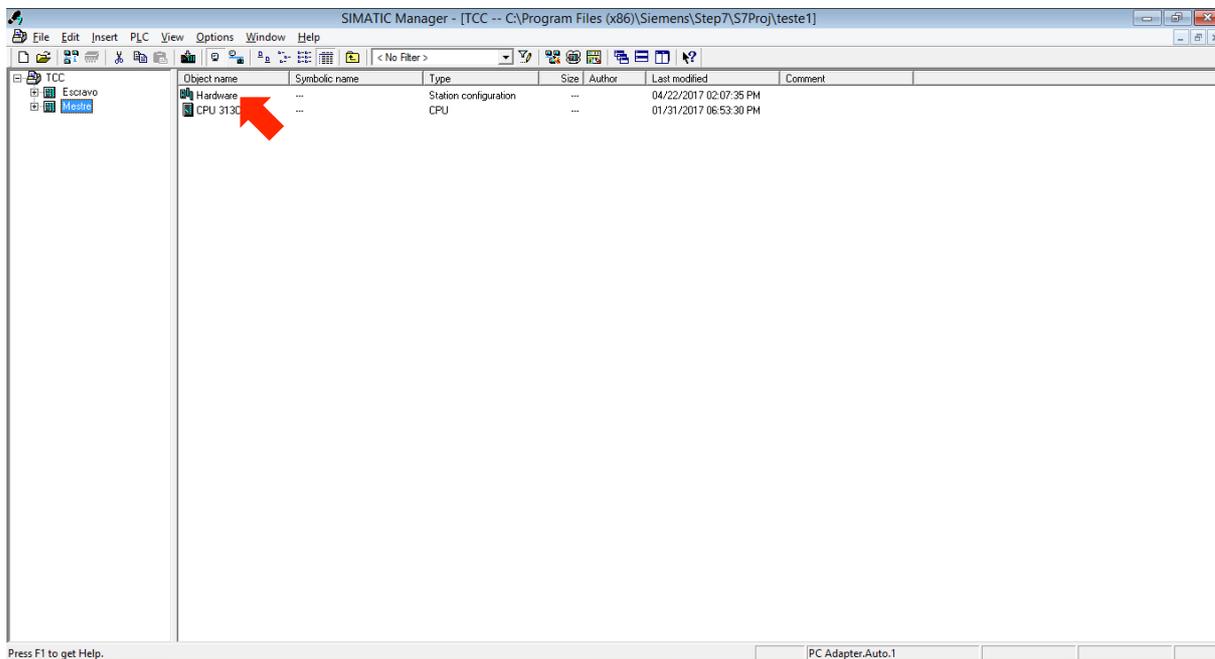


Figura 33 – Abrir configurações de hardware estação mestre.
Fonte: Autoria própria, 2017.

Abrirá a tela de configuração de hardware, conforme figura 34.

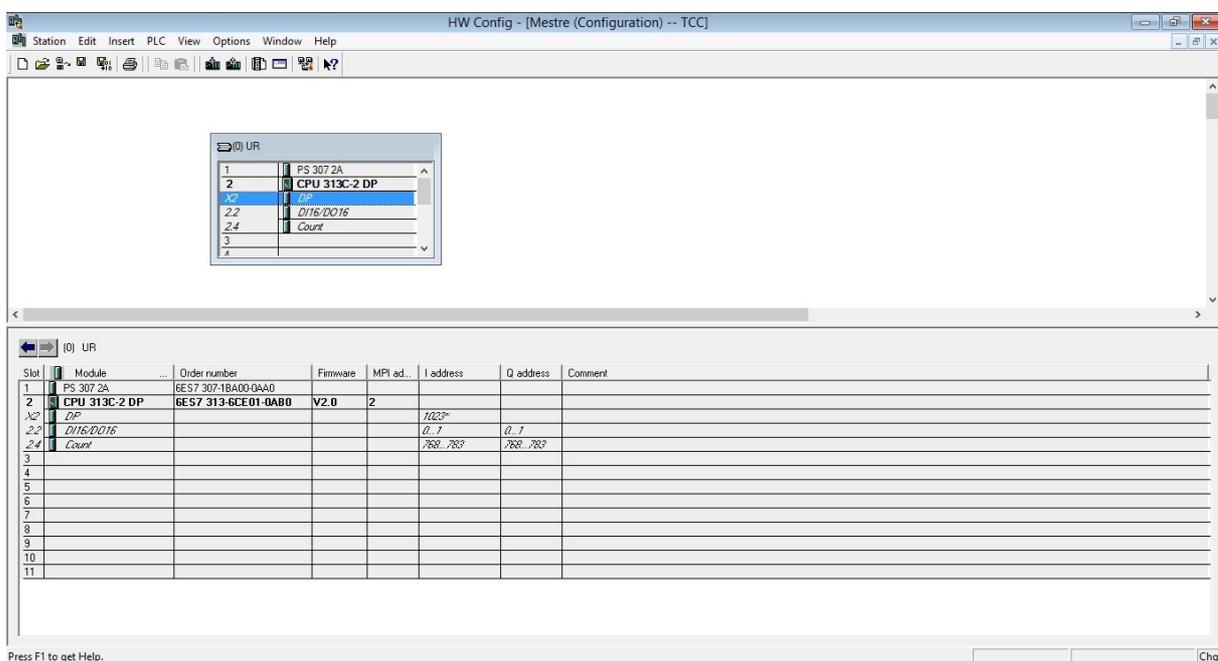


Figura 34 – Tela de configuração de hardware estação mestre.
Fonte: Autoria própria, 2017.

Para criar a rede DP, clicar duas vezes sobre a porta DP, como indica a figura 35.

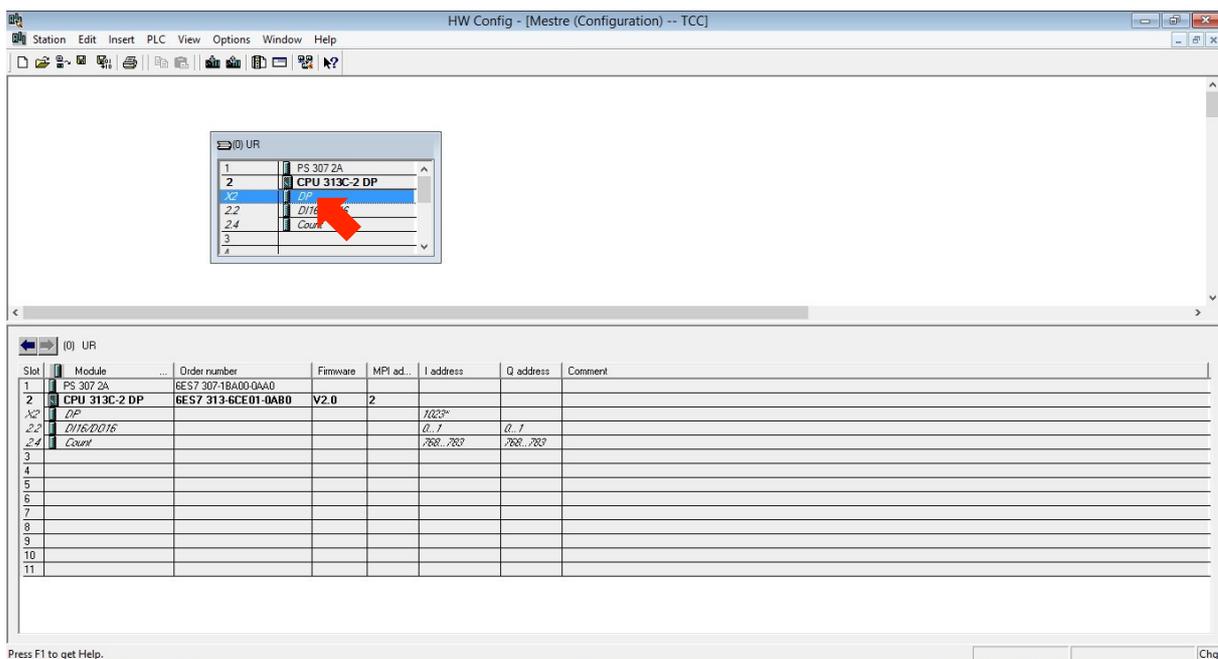


Figura 35 – Abrir tela de configuração de rede DP da estação mestre.
Fonte: Autoria própria, 2017.

Abrirá a janela de Propriedades – DP – (R0/S2.1), conforme figura 36.

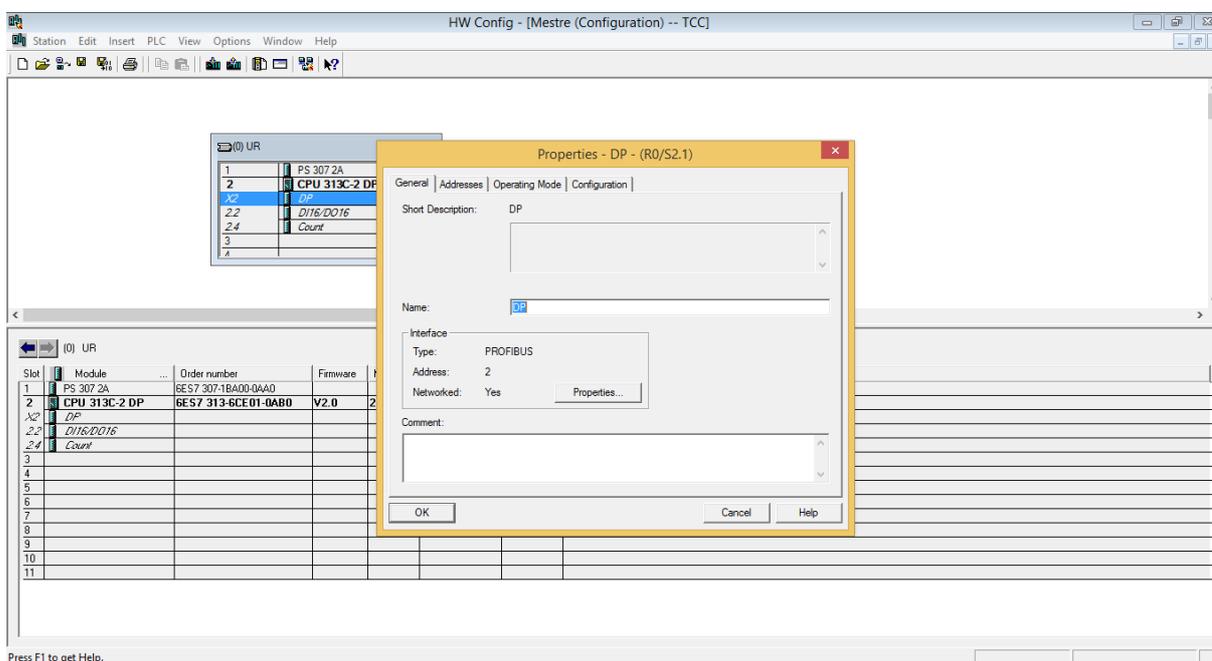


Figura 36 – Tela Propriedades – DP – (R0/S2.1) estação mestre.
Fonte: Autoria própria, 2017.

Na guia Geral, conforme a figura 37, clicar em Propriedades.

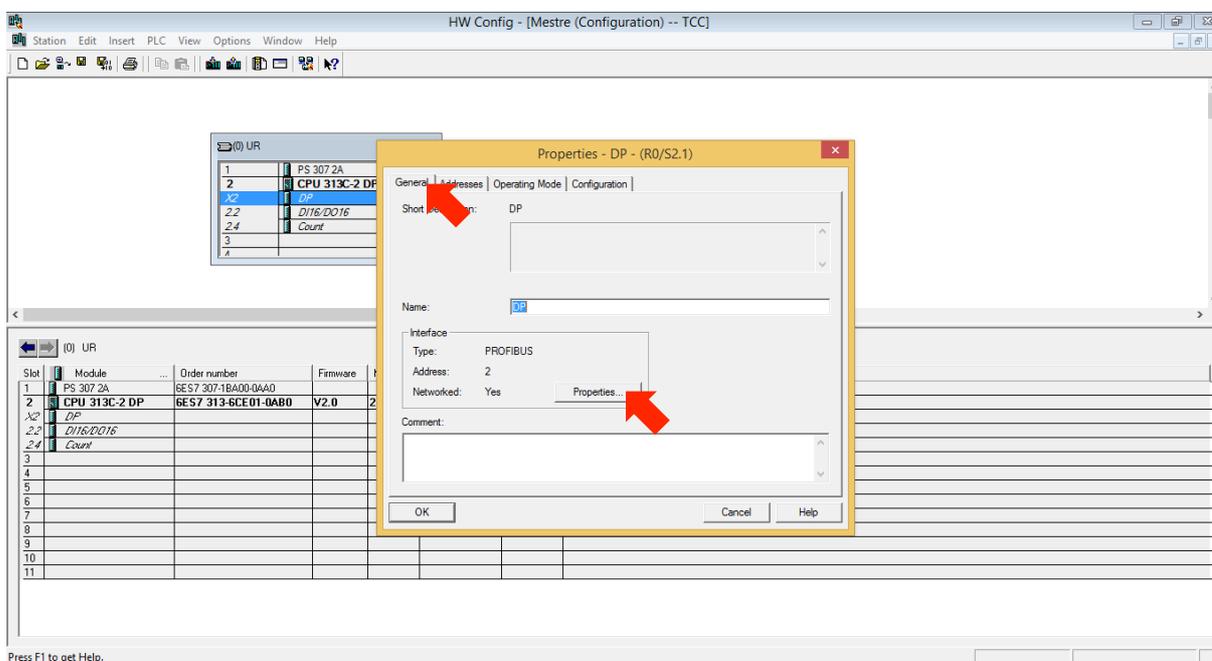


Figura 37 – Abrir tela Propriedades – DP – (R0/S2.1) estação mestre.
Fonte: Autoria própria, 2017.

Abrirá a janela de Propriedades – DP(R0/S2.1), conforme a figura 38, na guia Parâmetros no campo Endereço inserir o valor 2 e clicar no botão NEW...

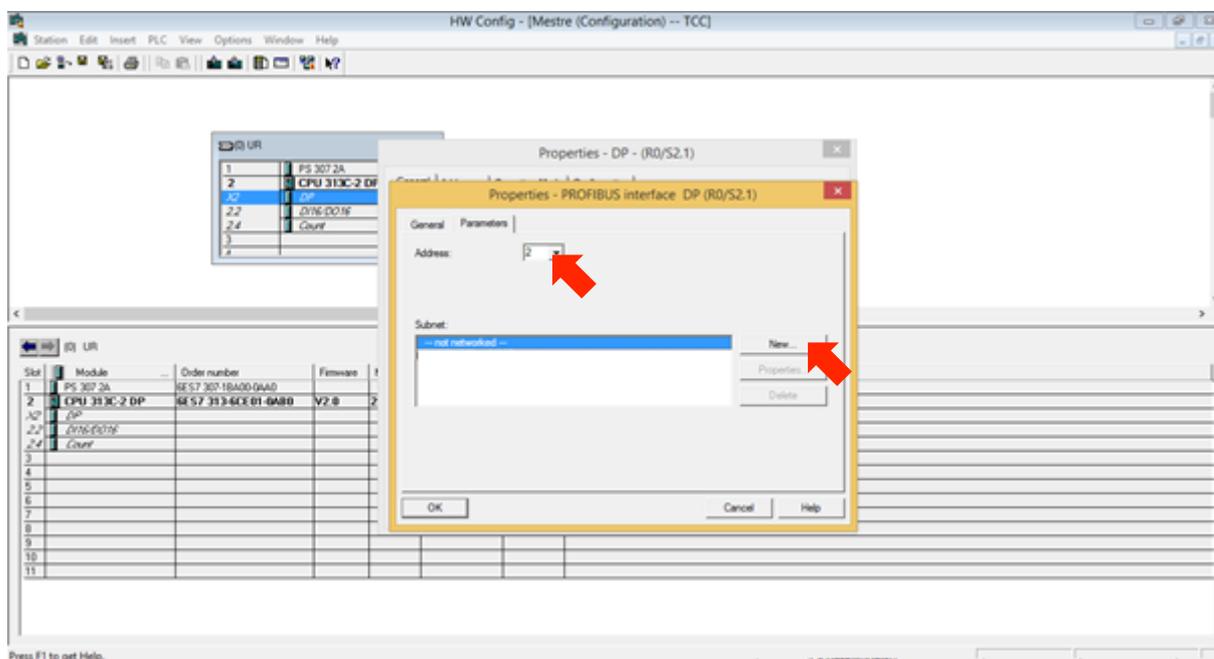


Figura 38 – Criar nova rede de comunicação.
Fonte: Autoria própria, 2017.

Abrirá a janela de Propriedades – PROFIBUS, conforme figura 39, verificar os campos “Name: PROFIBUS(1)” referente ao nome da rede e “S7 subnet ID: 005D – 001D” referente ao endereço de sub rede, estes campos são preenchidos automaticamente não havendo a necessidade de alteração dos mesmos, clique em OK.

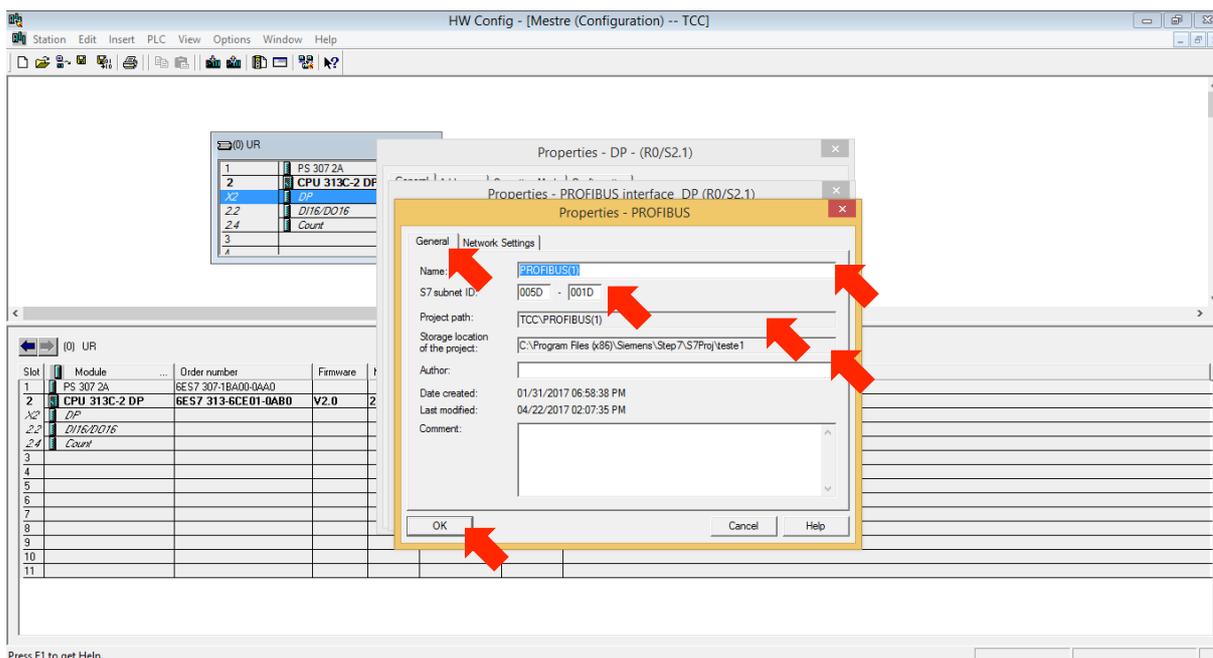


Figura 39 – Configuração da rede PROFIBUS.
Fonte: Autoria própria, 2017.

Retornará a janela Propriedades – DP – (R0/S2.1), conforme a figura 40, verificar que foi acrescentado uma nova linha no campo Subnet, com o nome da rede PROFIBUS criada que neste exemplo é PROFIBUS(1) 1.5 Mbps, clicar novamente em OK.

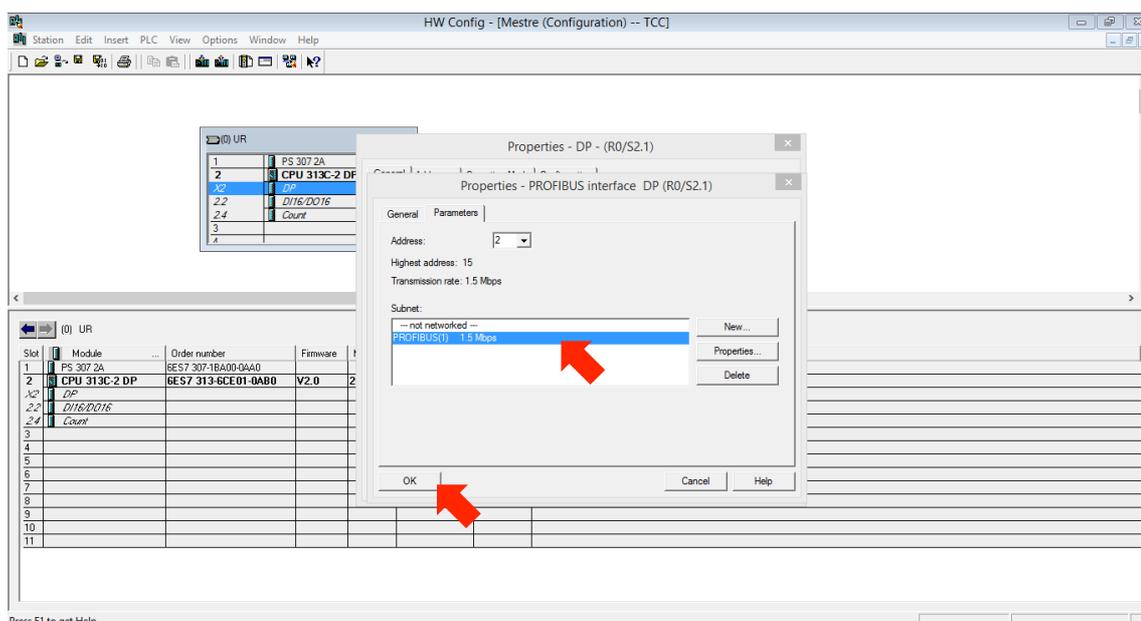


Figura 40 – Tela Propriedades – DP – (R0/S2.1).
Fonte: Autoria própria, 2017.

Na janela Propriedades – DP – (R0/S2.1), como mostra a figura 41, clicar em OK.

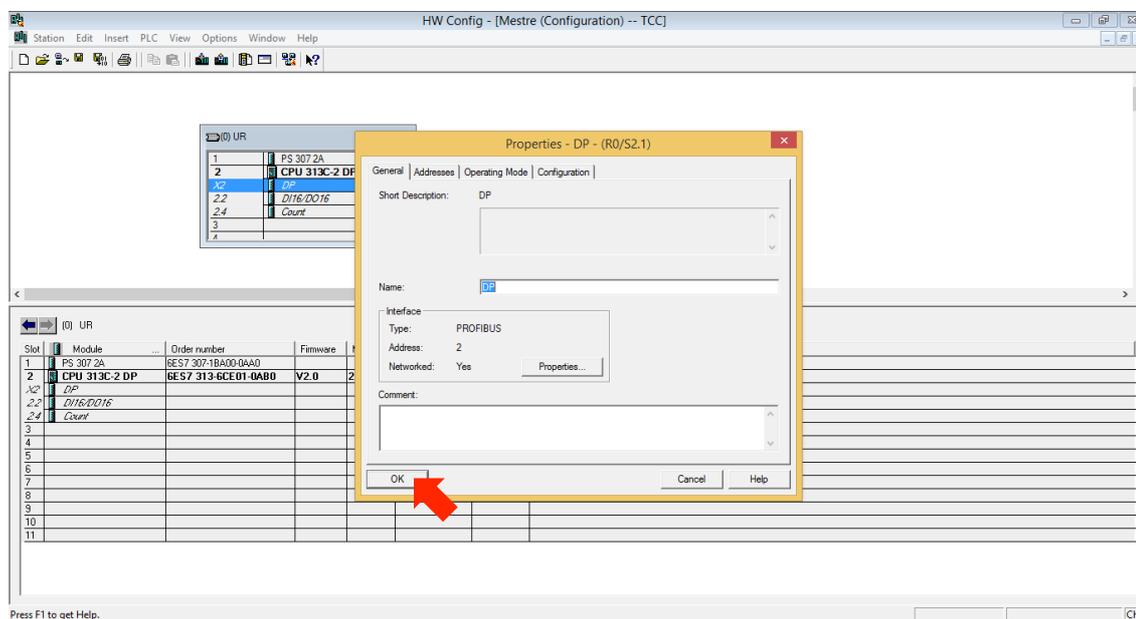


Figura 41 – Confirmar configurações da rede PROFIBUS.
Fonte: Autoria própria, 2017.

Já é possível verificar a rede PROFIBUS criada, conforme sinalizado na figura 42, clique em salvar e fechar para concluir a configuração da rede PROFIBUS.

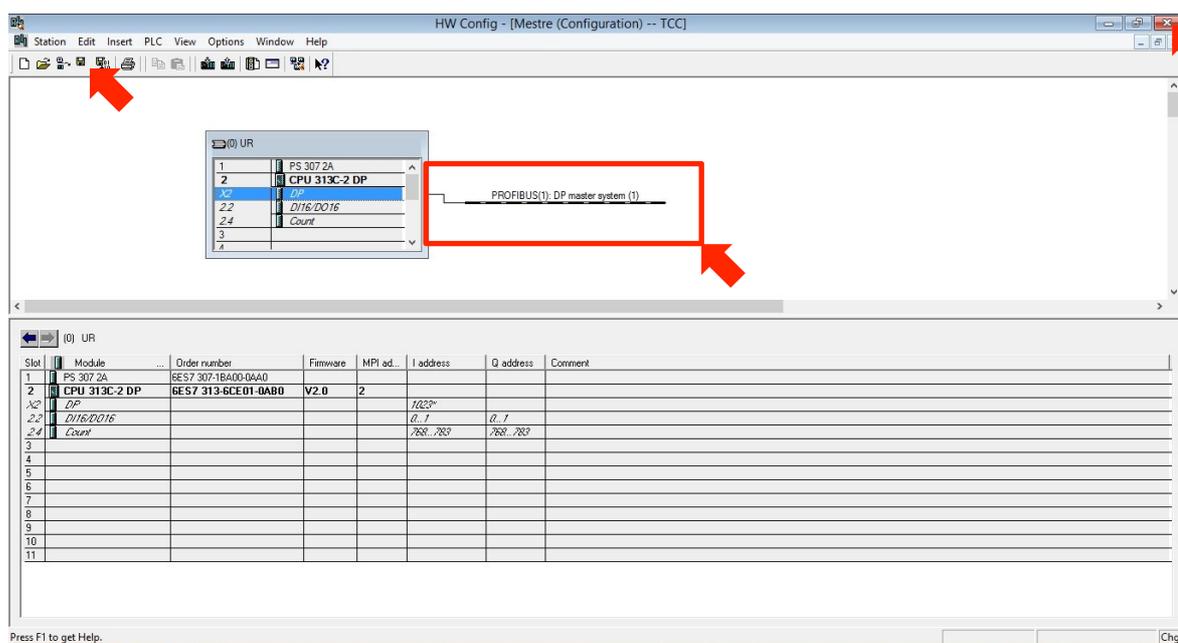


Figura 42 – Salvar configurações da rede PROFIBUS.
Fonte: Autoria própria, 2017.

Configurando a estação Escravo, clique no ícone correspondente a estação Escravo, conforme a figura 43.

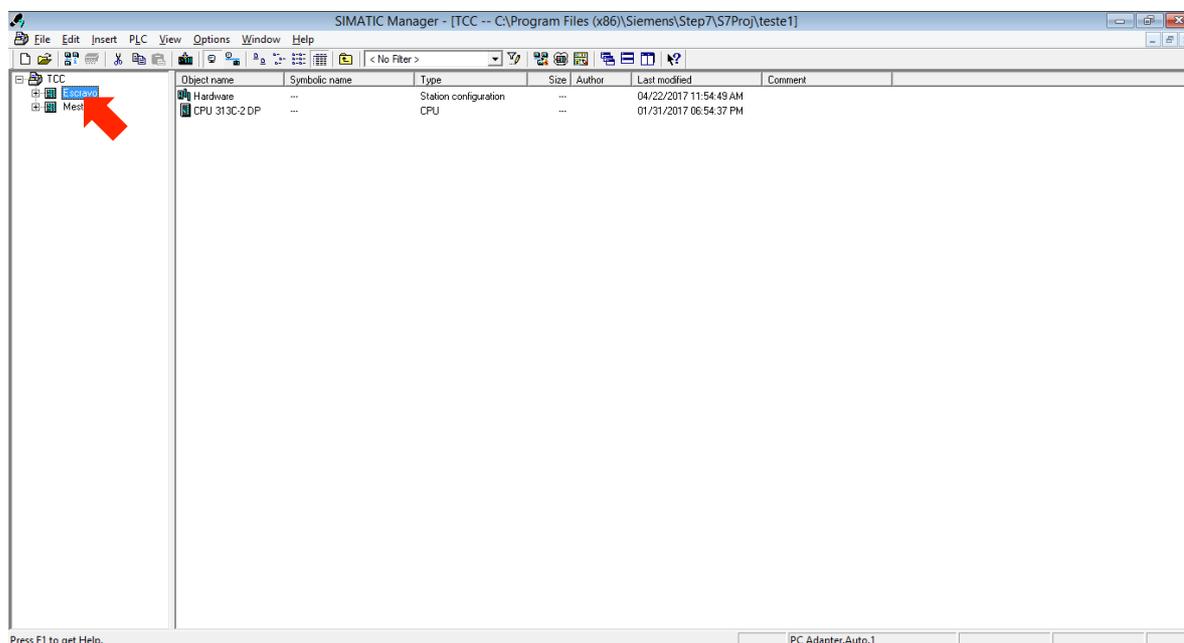


Figura 43 – Abrir projeto referente a estação escravo.
Fonte: Autoria própria, 2017.

Clique em Hardware, conforme mostra a figura 44.

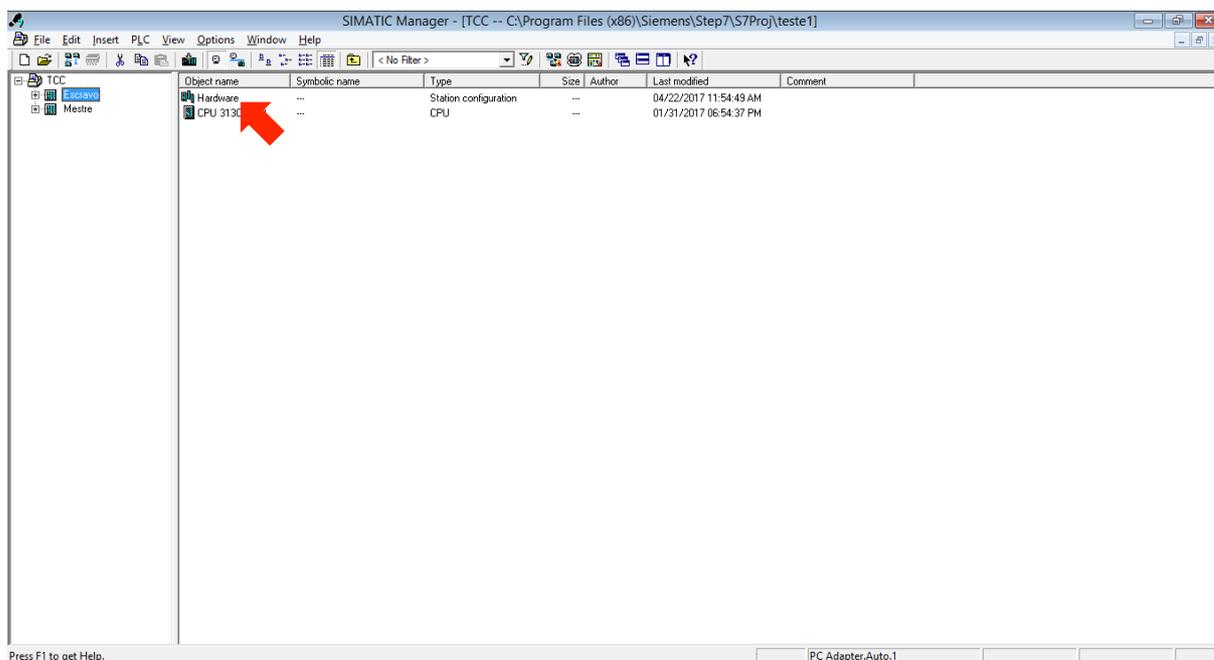


Figura 44 – Configurar o hardware referente a estação escravo.
Fonte: Autoria própria, 2017.

Duplo clique em DP, conforme mostra a figura 45.

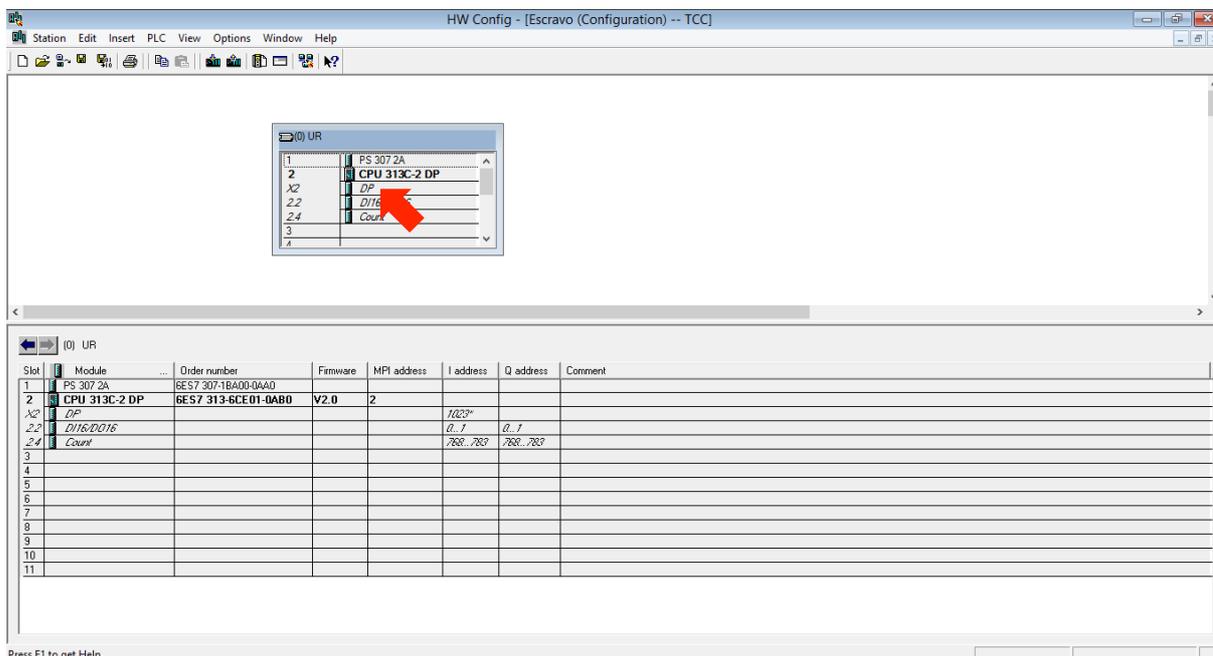


Figura 45 - Abrir tela de configuração de rede DP da estação escravo.
Fonte: Autoria própria, 2017.

Na janela Propriedades – DP – (R0/S2.1), na guia Geral, clique em Propriedades, conforme a figura 46.

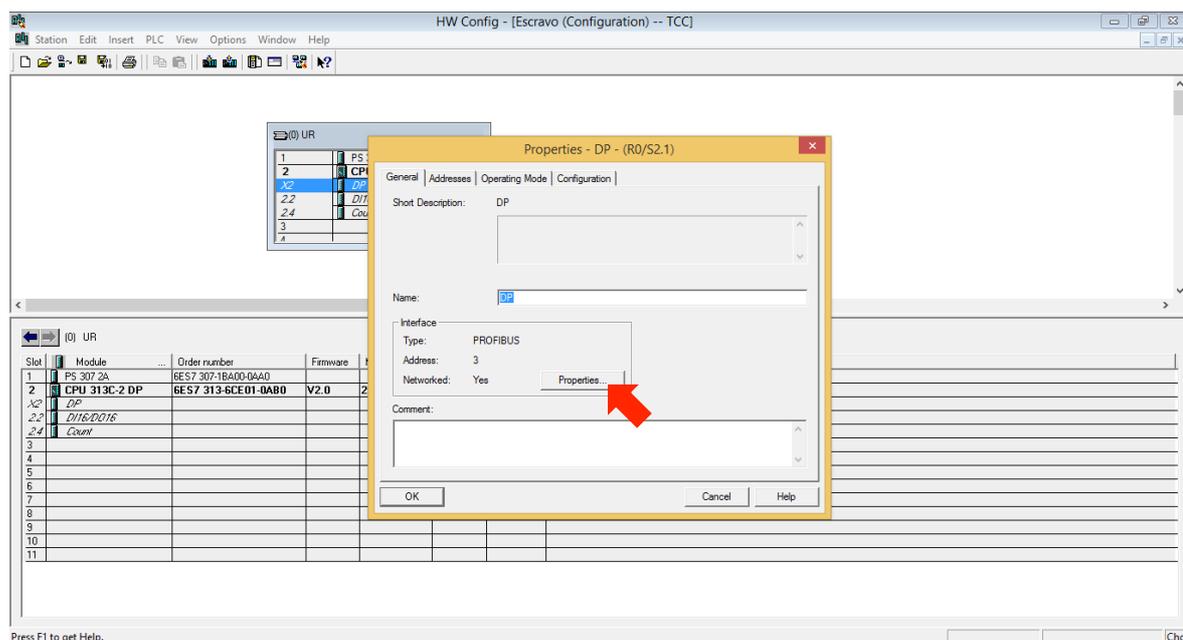


Figura 46 - Abrir tela Propriedades – DP – (R0/S2.1) estação escravo.
Fonte: Autoria própria, 2017.

Na janela Propriedades – PROFIBUS interface DP (R0/S2.1), no campo Endereço insira o valor 3, conforme a figura 47, neste campo deve ser inserido um valor diferente de 2 para evitar conflitos de comunicação, uma vez que este valor foi utilizado para a configuração da Estação Mestre , e clique em OK.

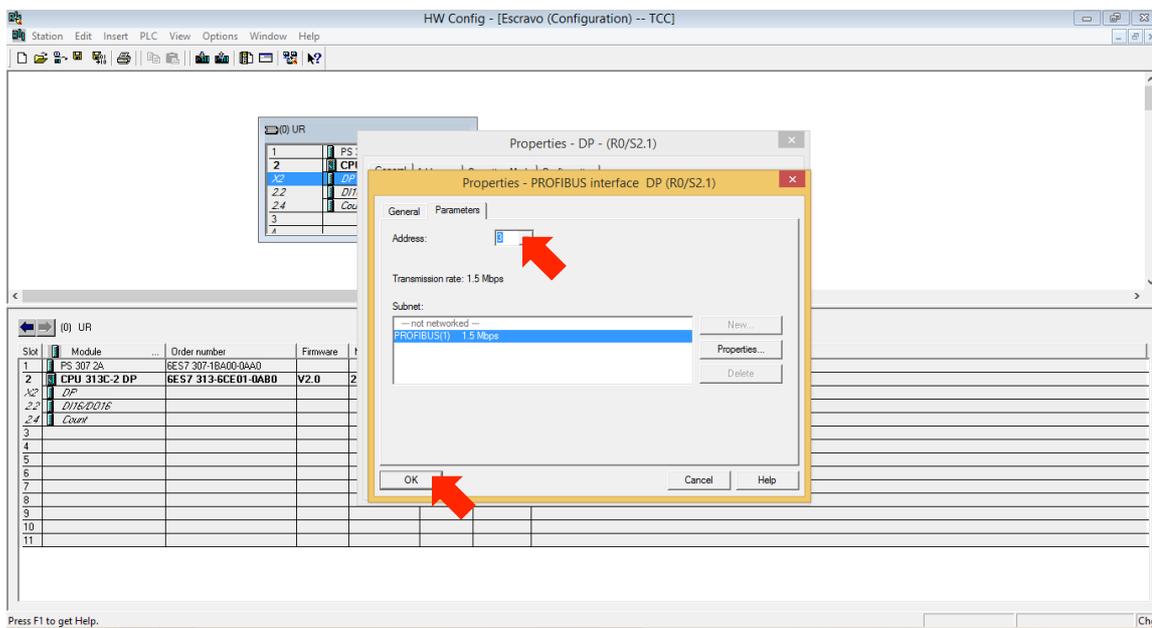


Figura 47 - Tela Propriedades – DP – (R0/S2.1) estação escravo.
Fonte: Autoria própria, 2017.

Na janela Propriedades – DP – (R0/S2.1), na Guia Modos de Operação, selecione DP slave, conforme mostra a figura 48, esta configuração indica que nossa estação irá trabalhar como uma Estação Escravo, clique em OK.

Clique uma vez no ícone da estação Mestre, conforme mostra a figura 50.

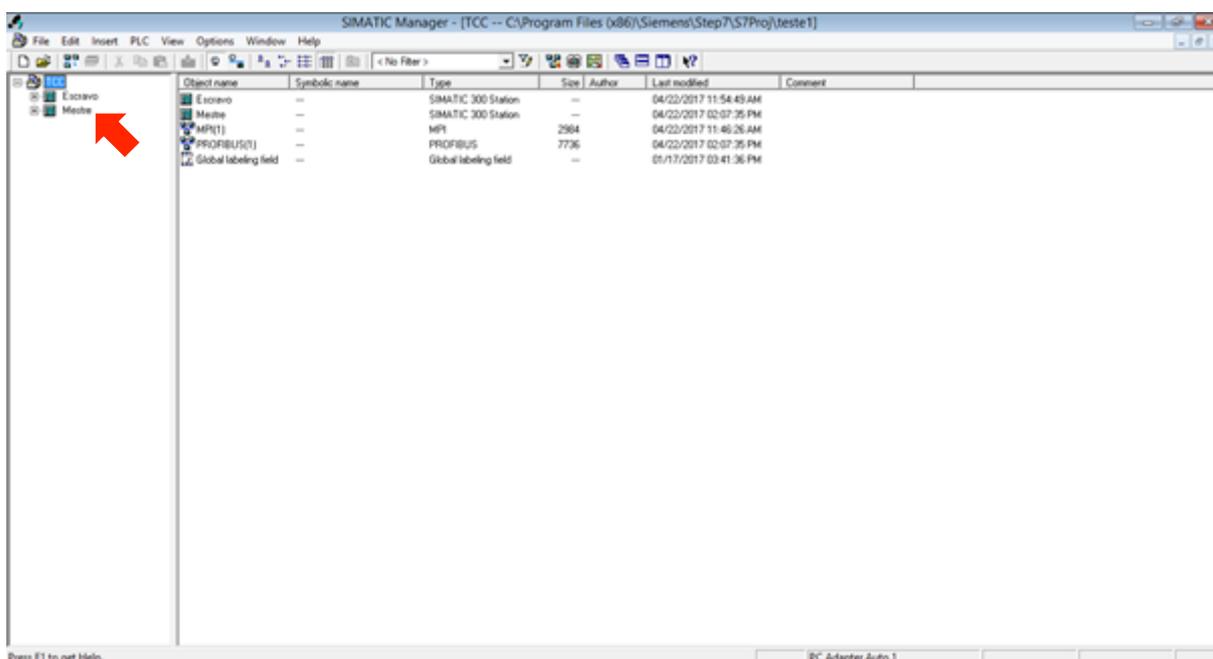


Figura 50 – Abrir projeto da estação mestre.
Fonte: Autoria própria, 2017.

Clicar duas vezes no ícone de hardware, conforme a figura 51.

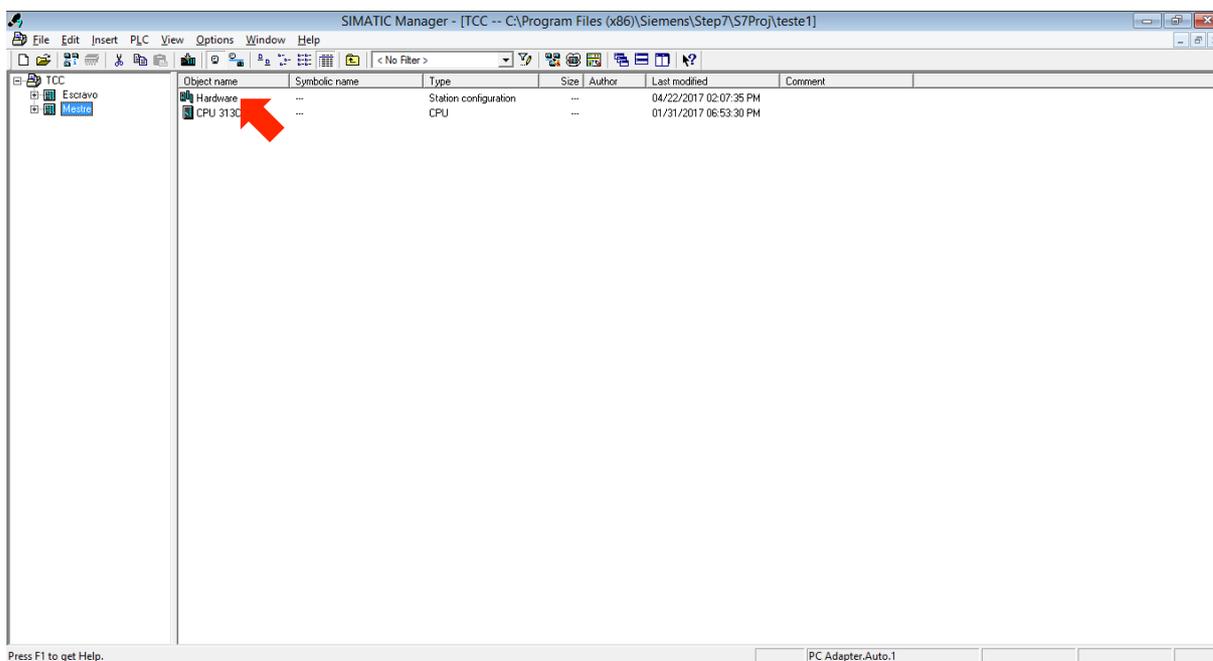


Figura 51 – Abrir configuração de hardware da estação mestre.
Fonte: Autoria própria, 2017.

Clique no ícone Catálogo, conforme a figura 52.

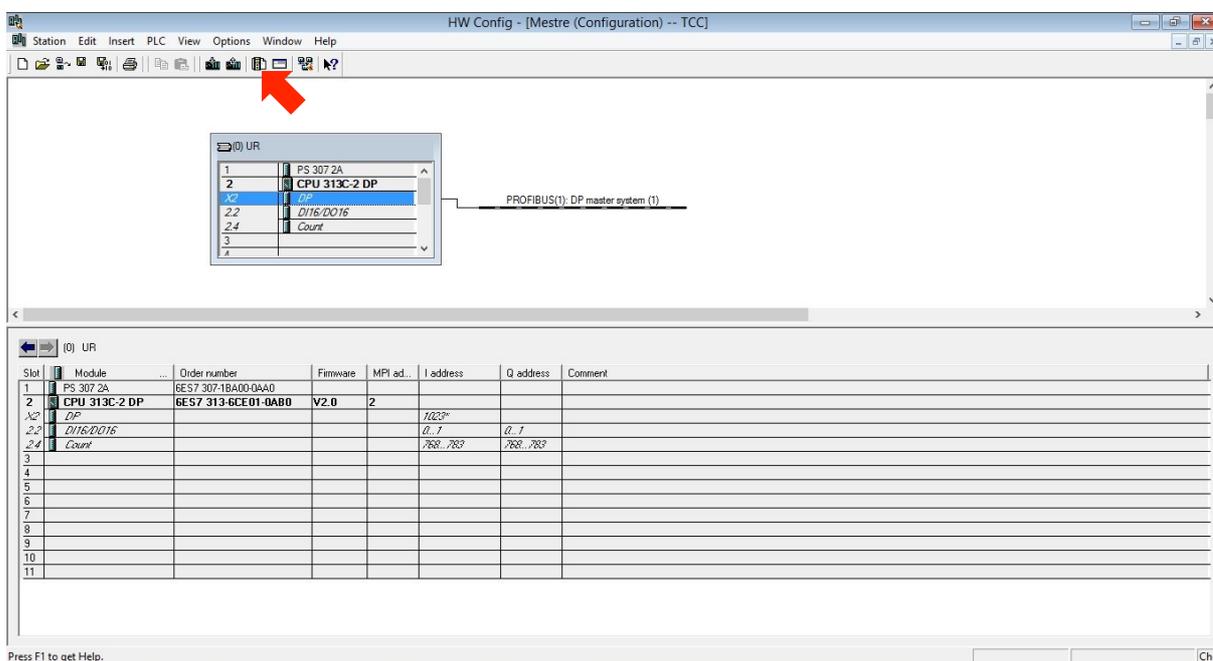


Figura 52 – Abrir o catálogo.
Fonte: Autoria própria, 2017.

Na janela que se abre, clique no símbolo (+) ao lado da palavra PROFIBUS DP, conforme a figura 53.

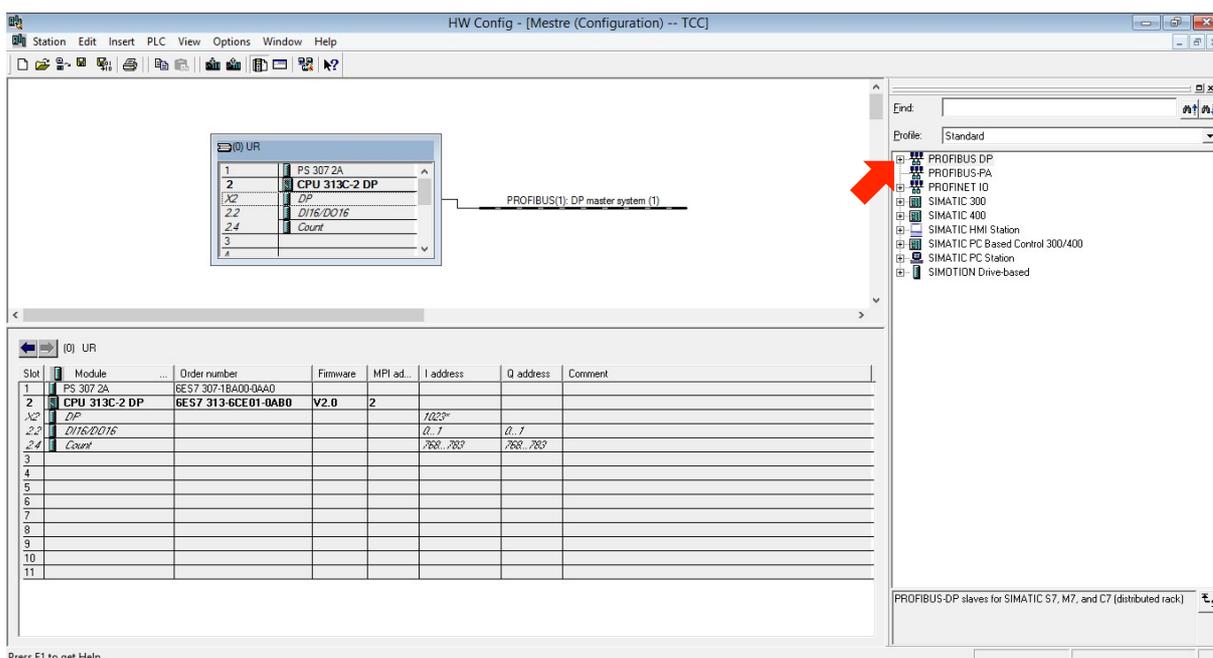


Figura 53 – Expandir a lista PROFIBUS DP do catálogo.
Fonte: Autoria própria, 2017.

Na lista que se abre, clique no símbolo (+) ao lado de Configuração de estações, conforme a figura 54.

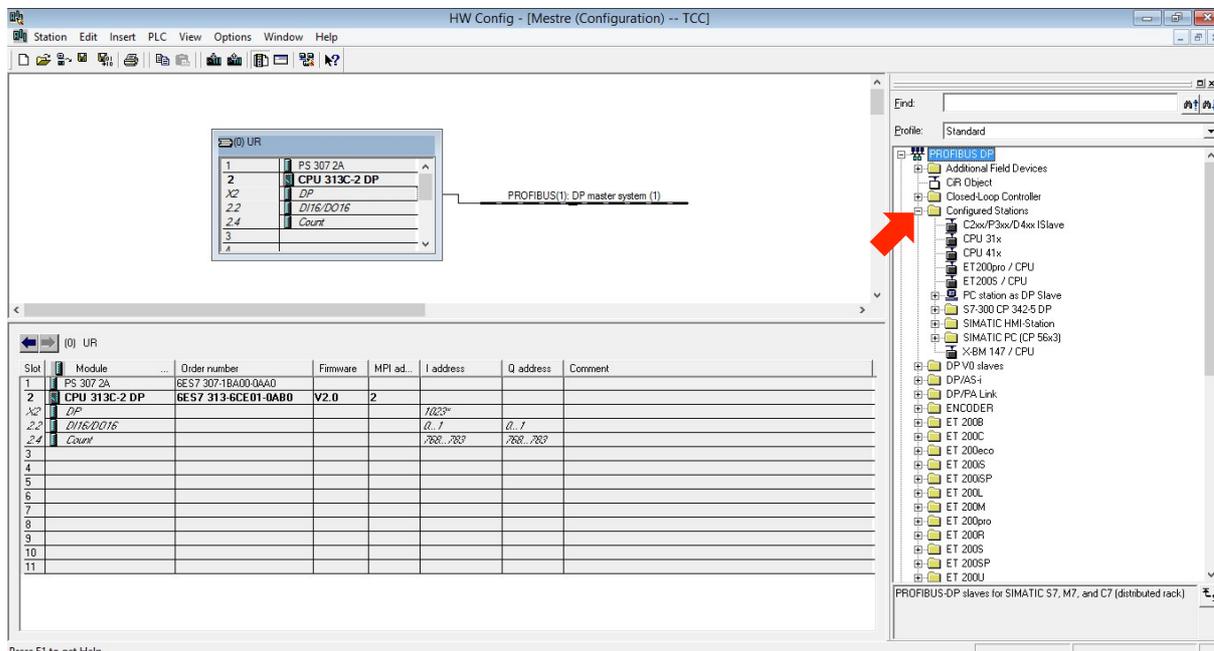


Figura 54 – Expandir a lista de configuração de estações.

Fonte: Autoria própria, 2017.

Clique e segure sobre o ícone CPU 31x, referente ao modelo de nossa CPU da Estação Escravo, e arraste até a rede PROFIBUS criada anteriormente, quando aparecer o sinal de mais solte o botão do mouse, conforme mostra a figura 55, com esta ação estamos acoplando através da rede PROFIBUS que criamos as Estações Mestre e Escravo.

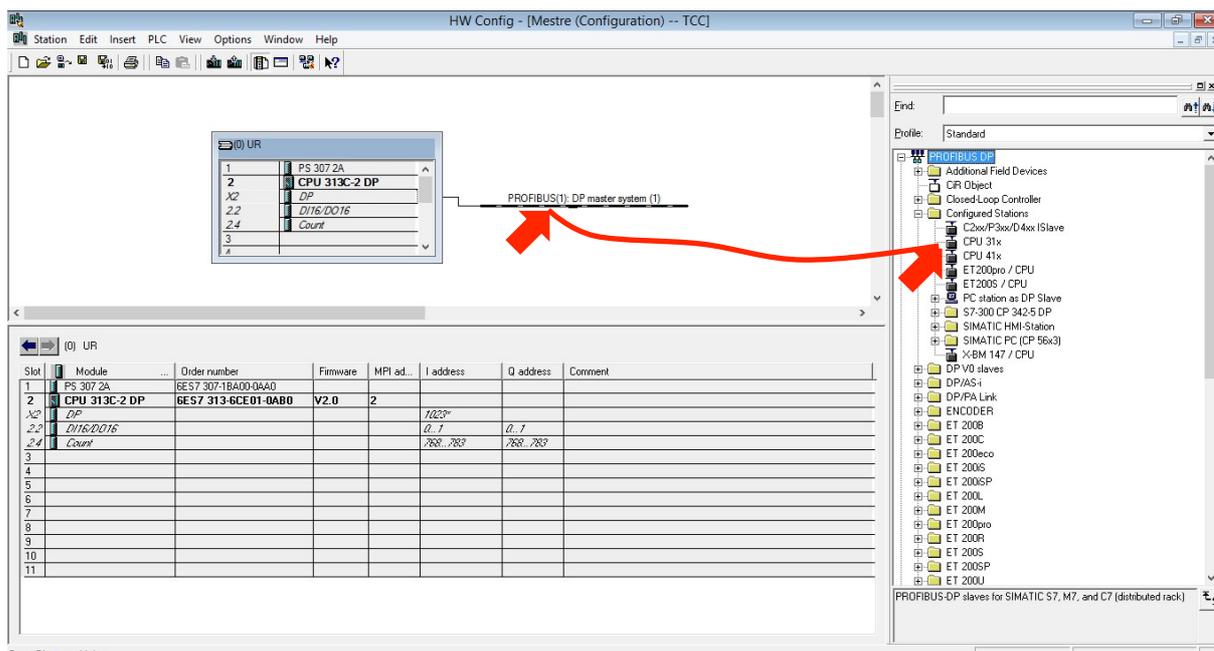


Figura 55 – Inserir a CPU 31x na rede PROFIBUS.
Fonte: Autoria própria, 2017.

Irá abrir a janela DP slave properties, verifique na guia Coupling que aparece a CPU referente a Estação Escravo que já configuramos bem como o endereço 3, clique no botão acoplar, conforme a figura 56.

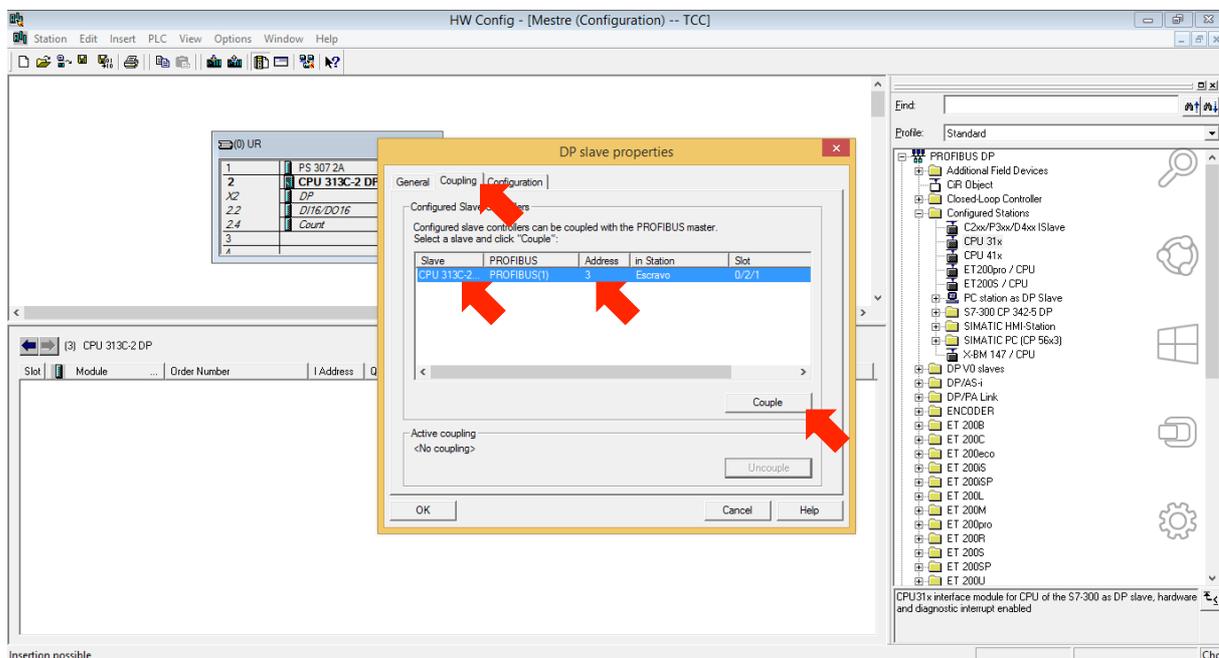


Figura 56 – Acoplar a estação escravo a estação mestre.
Fonte: Autoria própria, 2017.

Clique em OK, conforme a figura 57 e a estação Escravo estará acoplada a rede PROFIBUS.

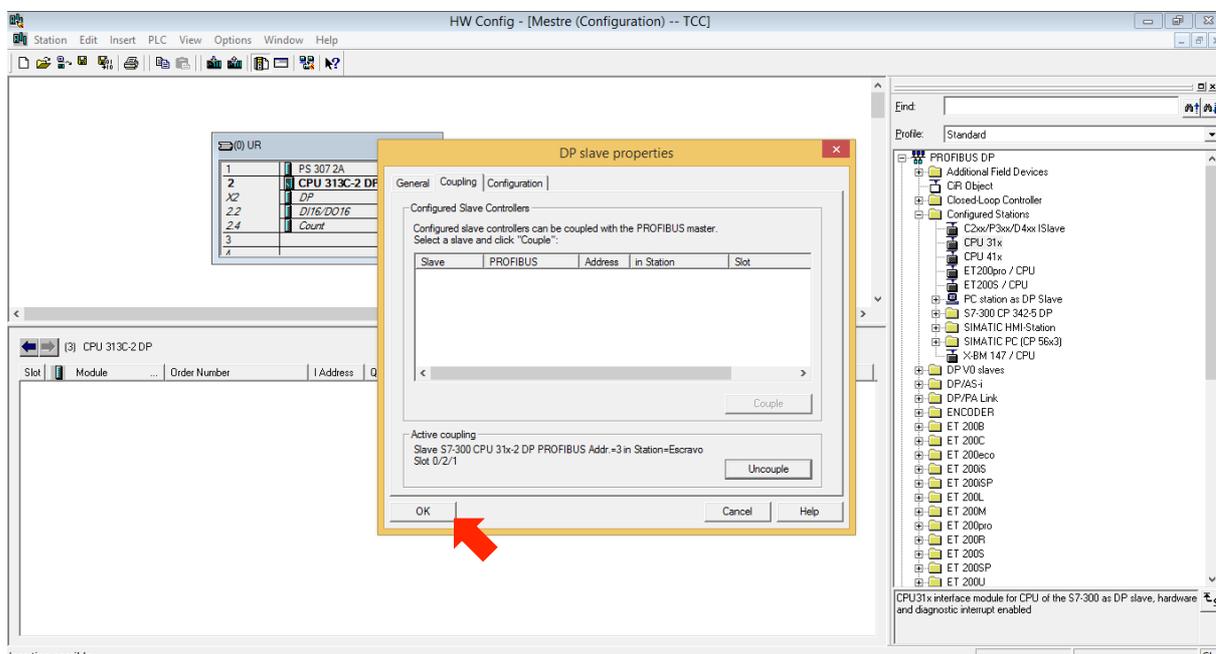


Figura 57 – Confirmar o acoplamento da estação escravo.
Fonte: Autoria própria, 2017.

Clique em salvar/compilar e depois em fechar, conforme mostra a figura 58.

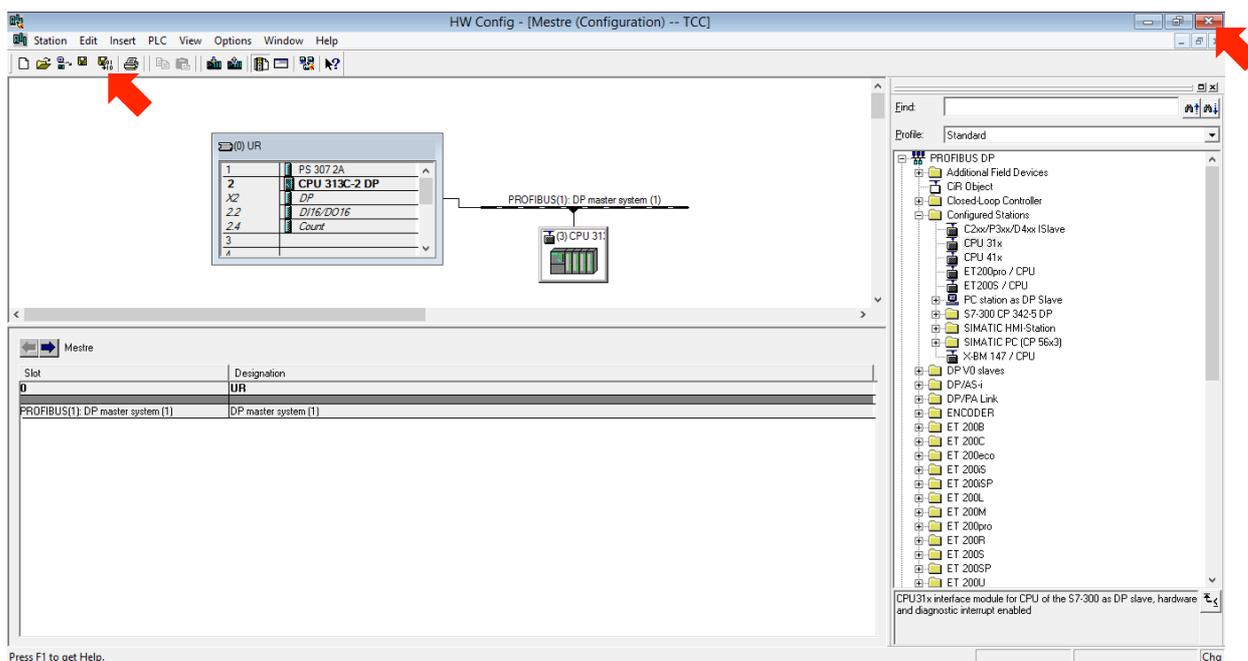


Figura 58 – Salvar e fechar o projeto da estação mestre.
Fonte: Autoria própria, 2017.

Na estação Escravo devemos configurar os frames, que são os endereços “bytes” de comunicação entre uma estação e outra.

Clique no ícone da estação Escravo, conforme a figura 59.

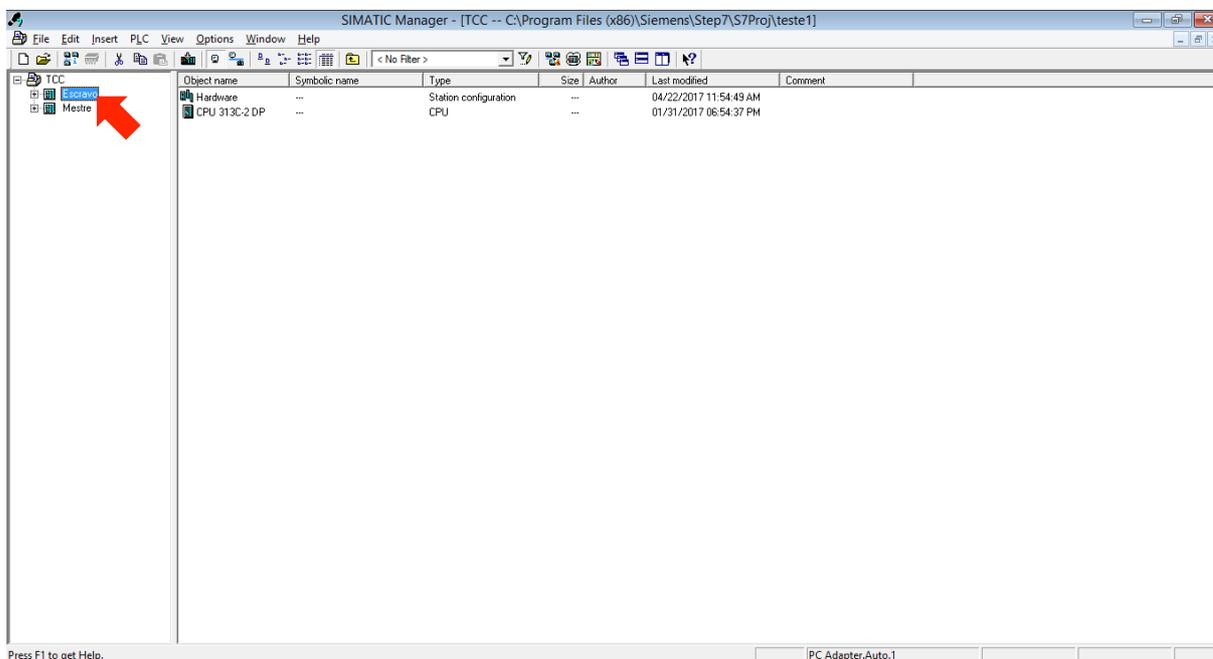


Figura 59 – Abrir projeto da estação escravo.
Fonte: Autoria própria, 2017.

Clique em Hardware, conforme a figura 60.

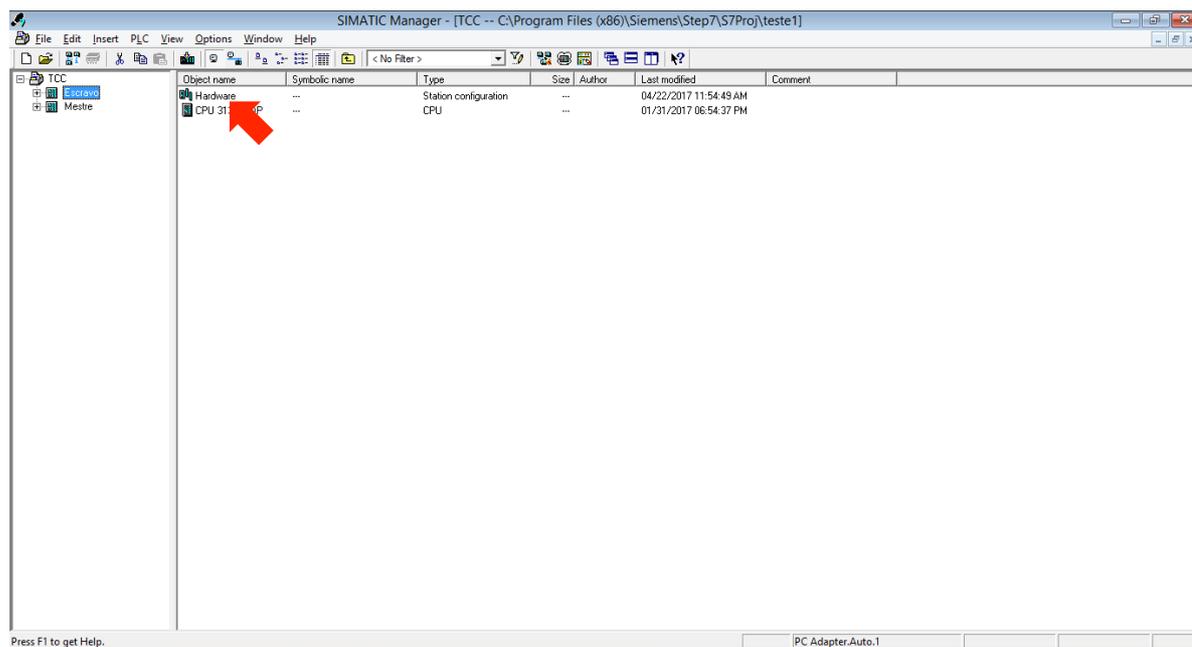


Figura 60 – Abrir configuração de hardware da estação escravo.
Fonte: Autoria própria, 2017.

Duplo clique em DP, conforme a figura 61.

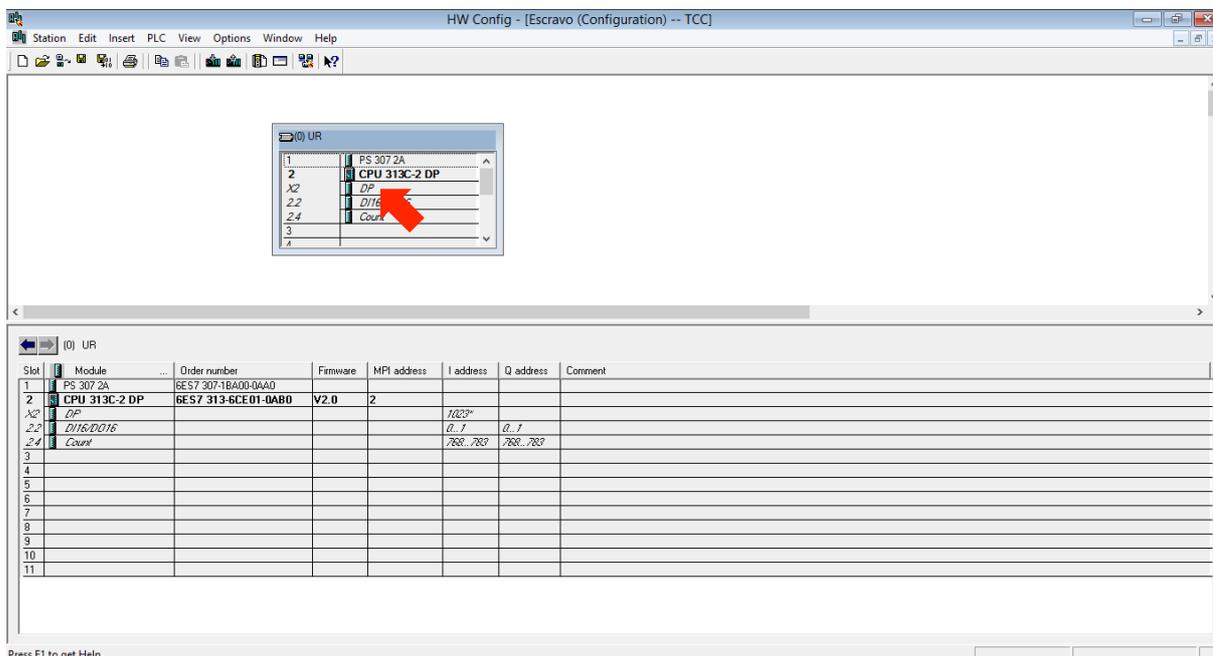


Figura 61 – Abrir configuração DP da estação escravo.
Fonte: Autoria própria, 2017.

Abrirá a janela Properties – DP – (R0/S2.1), na guia Configuração, clique em New, conforme a figura 62.

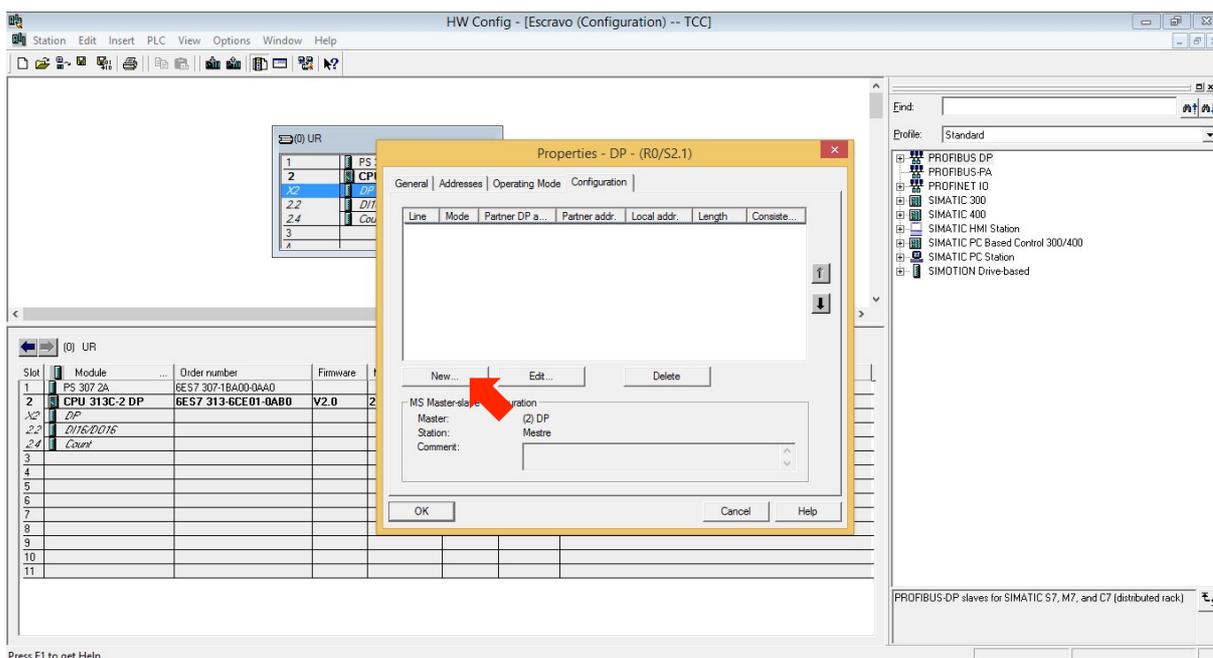


Figura 62 – Abrir nova configuração DP na estação escravo.
Fonte: Autoria própria, 2017.

Abrirá a janela Properties – DP – (R0/S2.1) – Configuration – Line 1. Nesta janela tem-se duas colunas onde a coluna com o nome DP Partner: Master é referente a estação Mestre e a coluna Local: Slave é referente a estação Escravo, conforme a figura 63.

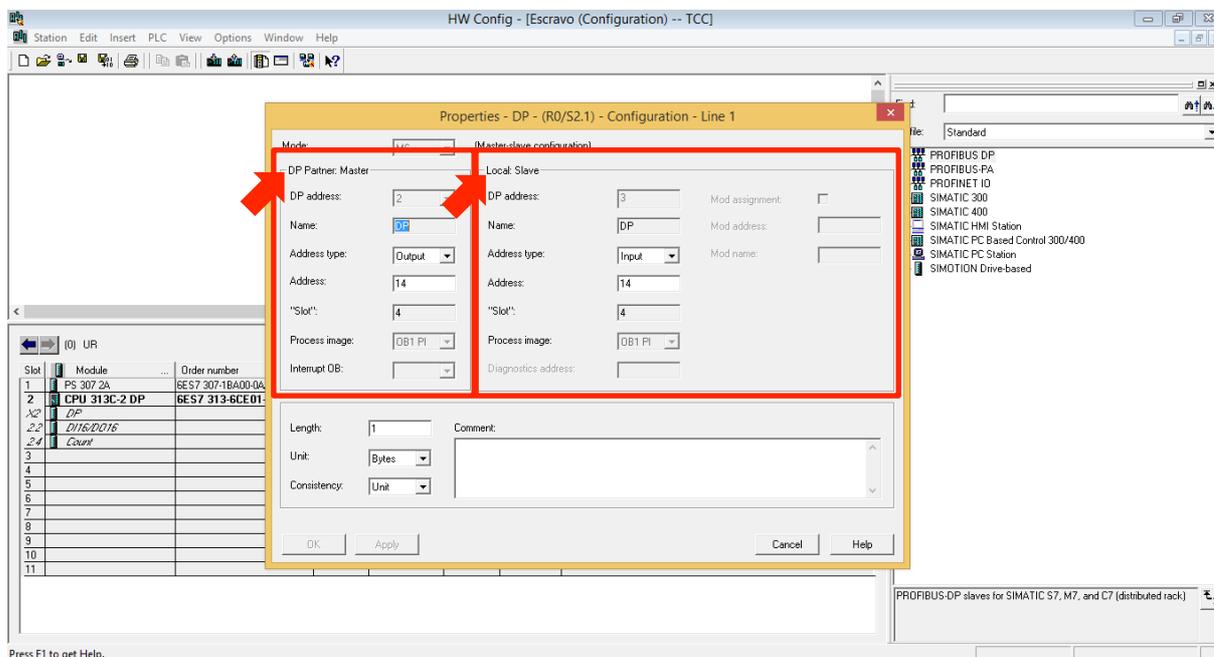


Figura 63 – Tela Propriedades – DP – (R0/S2.1) estação escravo.
Fonte: Autoria própria, 2017.

Neste primeiro frame, configurar a comunicação da estação Mestre para a estação Escravo.

Conforme a figura 64, na coluna referente a estação Mestre neste exemplo configura-se os campos da seguinte maneira:

Address type: Output

Address: 14

O campo Address type indica o tipo daquele endereço, se será uma entrada ou saída de dados, em nosso caso estaremos utilizando o conjunto de Bytes 14 como saídas na Estação Mestre.

Qual o conjunto de Bytes estará sendo utilizado em seu projeto é opcional, a única recomendação é que se utilize um conjunto especificamente para interface entre as estações Mestre e Escravo.

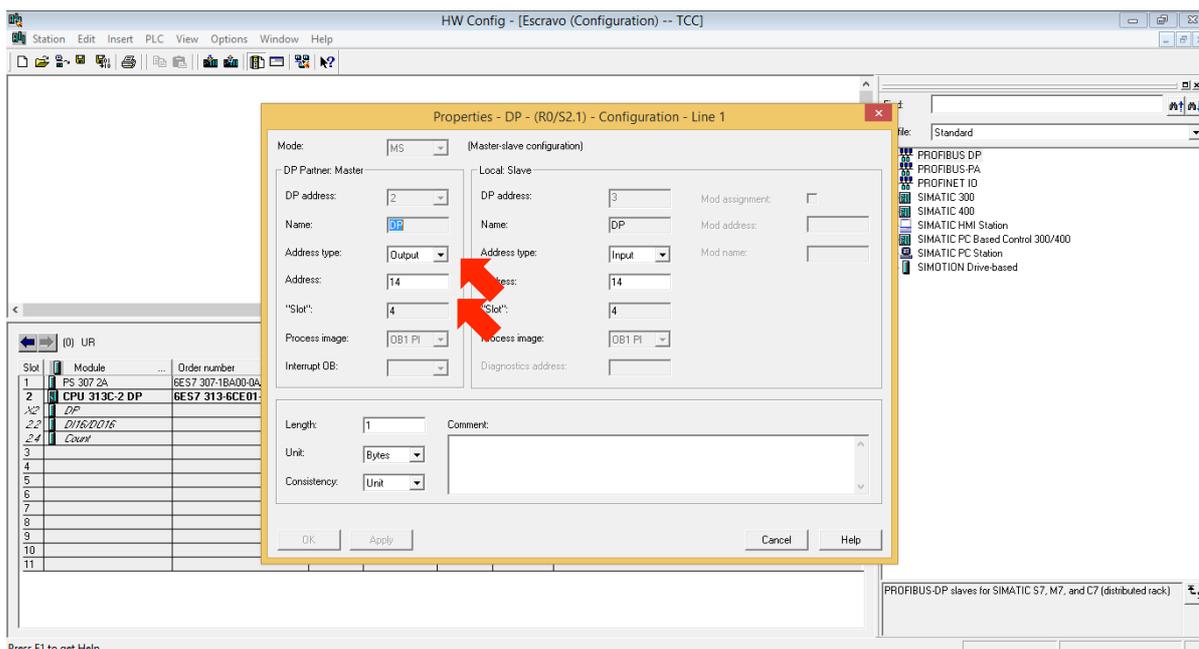


Figura 64 – Frame de comunicação mestre para o escravo.
Fonte: Autoria própria, 2017.

Conforme a figura 65, na coluna referente a estação Escravo neste exemplo configura-se os campos da seguinte maneira:

Address type: Input

Address: 14

O campo Address type indica o tipo daquele endereço, se será uma entrada ou saída de dados, neste caso o conjunto de Bytes 14 serão entradas na Estação Escravo, o que significa, que toda vez que por exemplo o bit Q14.0 estiver no estado lógico 1 na estação Mestre, seu bit correspondente I14.0 na estação Escravo também estará no estado lógico 1, se Q14.0 na estação Mestre estiver em estado lógico igual 0, o bit I14.0 da estação Escravo também estará em estado lógico igual a 0.

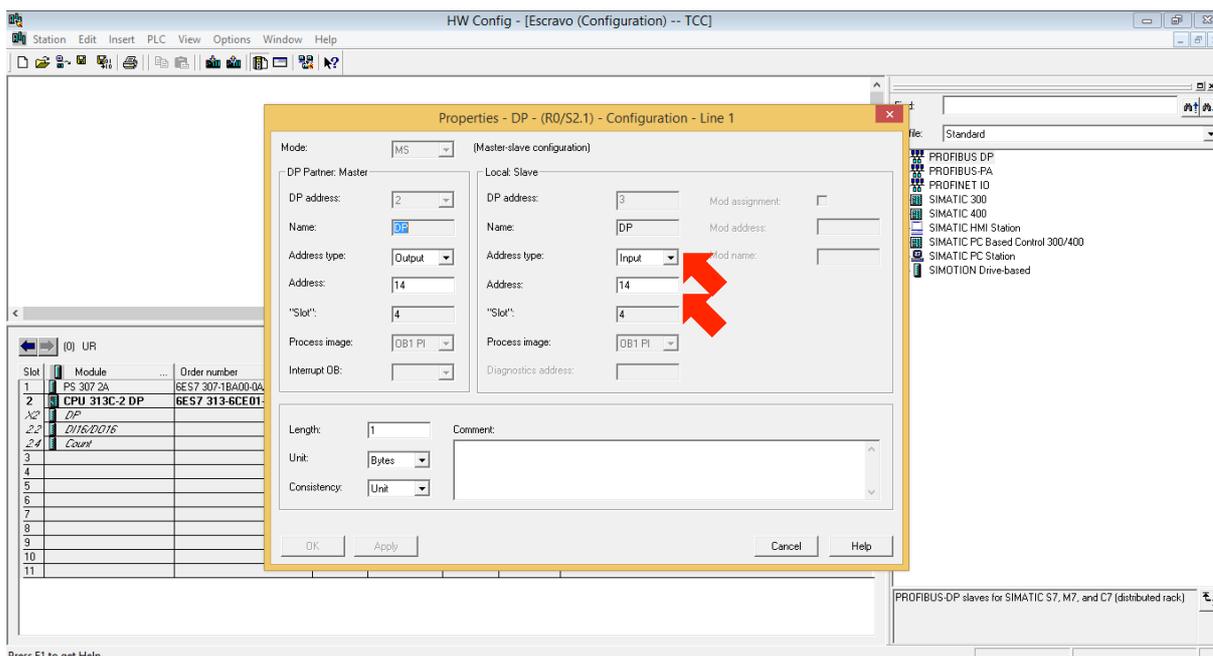


Figura 65 - Frame de comunicação mestre para o escravo.
Fonte: Autoria própria, 2017.

Após as configurações realizadas clicar em OK, conforme a figura 66.

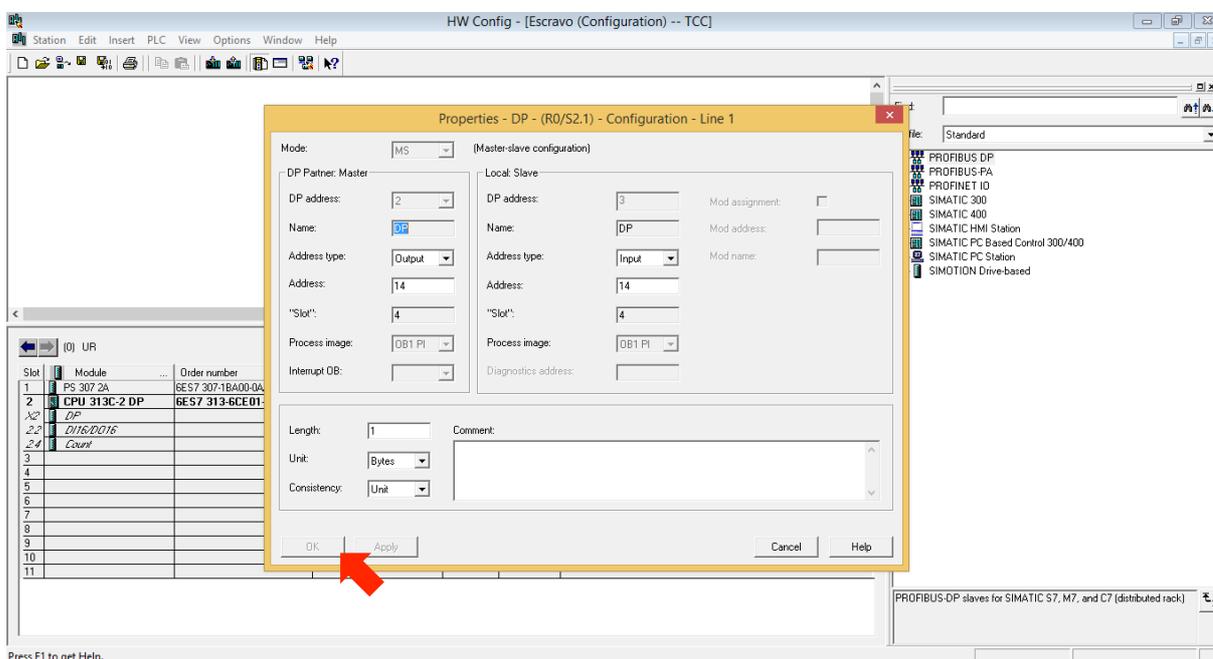


Figura 66 – Confirmando as configurações do frame de comunicação.
Fonte: Autoria própria, 2017.

Conforme a figura 67, verifique que a linha correspondente ao frame criado apareceu na janela Properties – DP – (R0/S2.1), agora deve-se configurar um segundo frame, que será a comunicação da estação Escravo para a Estação Mestre, clique em new.

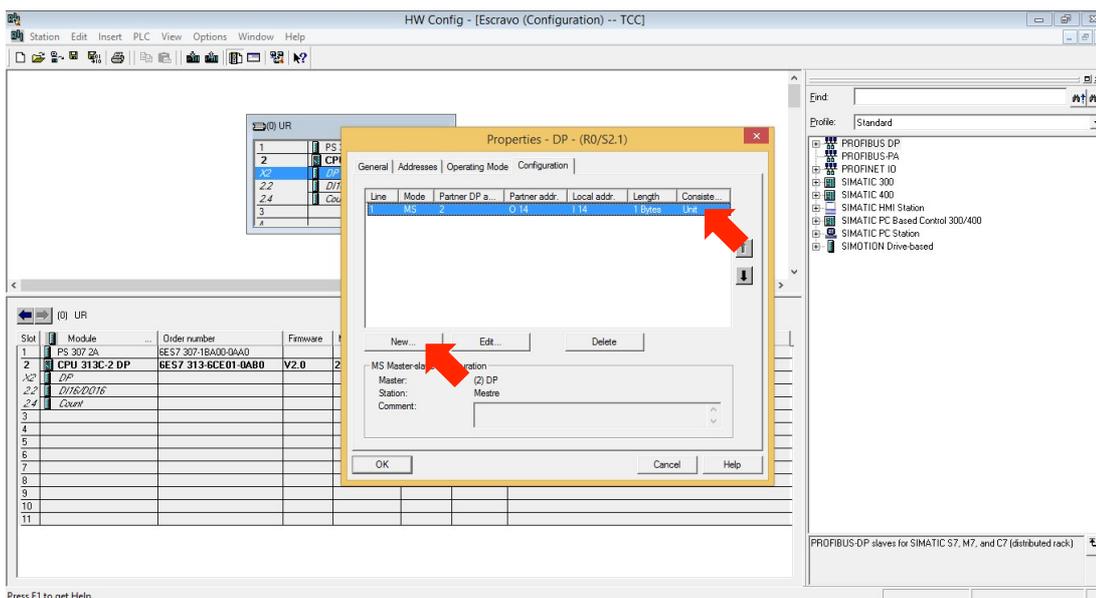


Figura 67 – Criar novo frame de comunicação.
Fonte: Autoria própria, 2017.

Analogamente aos passos anteriores e conforme a figura 68 e 69, deve-se configurar os campos da seguinte maneira:

Address type: Input

Address: 15

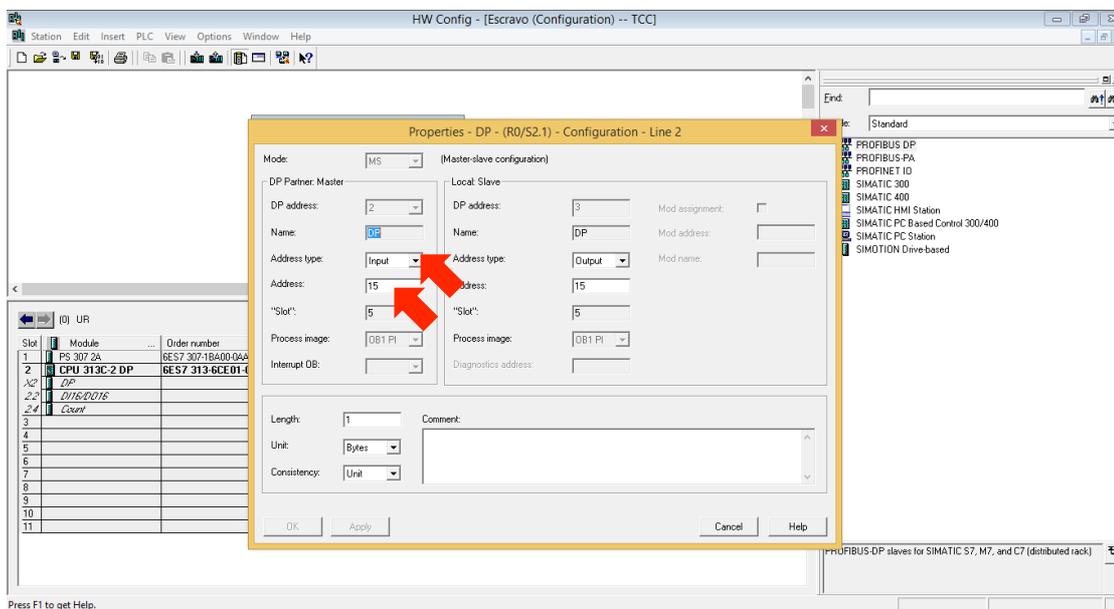


Figura 68 – Frame de comunicação da estação escravo / mestre.
Fonte: Autoria própria, 2017.

Address type: Output

Address: 15

O que significa que os bits referentes ao byte 15 na estação Mestre terão o mesmo estado lógico que os bits referentes ao byte 15 na estação Escravo.

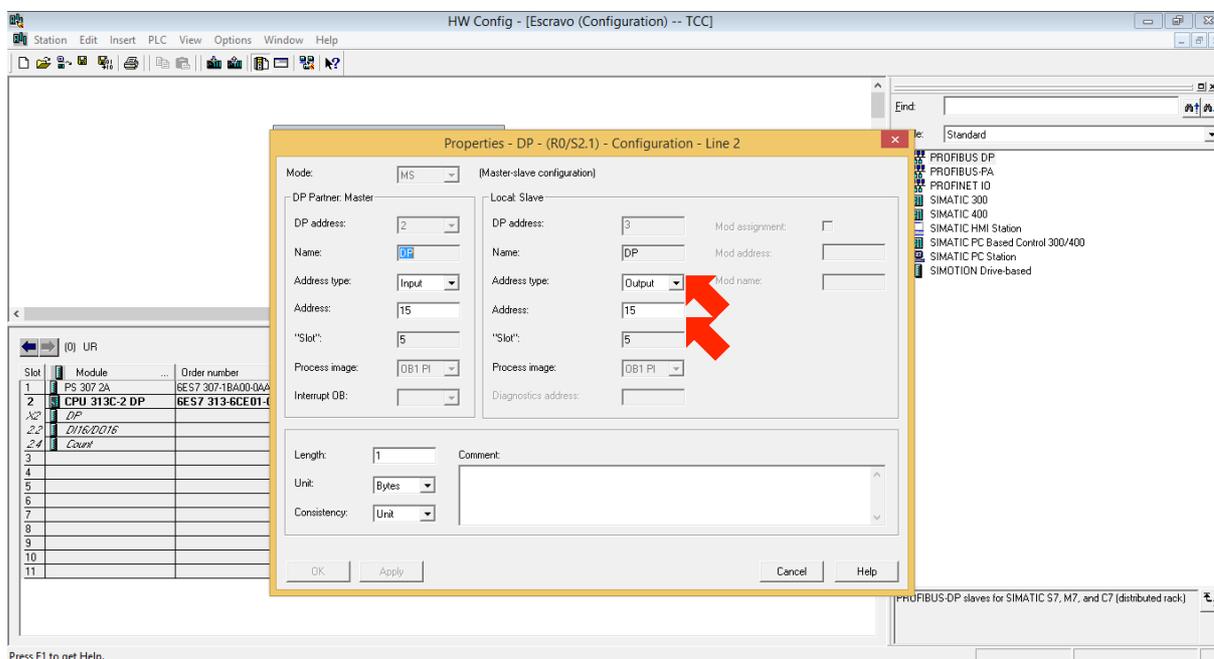


Figura 69 - Frame de comunicação da estação escravo / mestre.
Fonte: Autoria própria, 2017.

Após as configurações realizadas clicar em OK, conforme a figura 70.

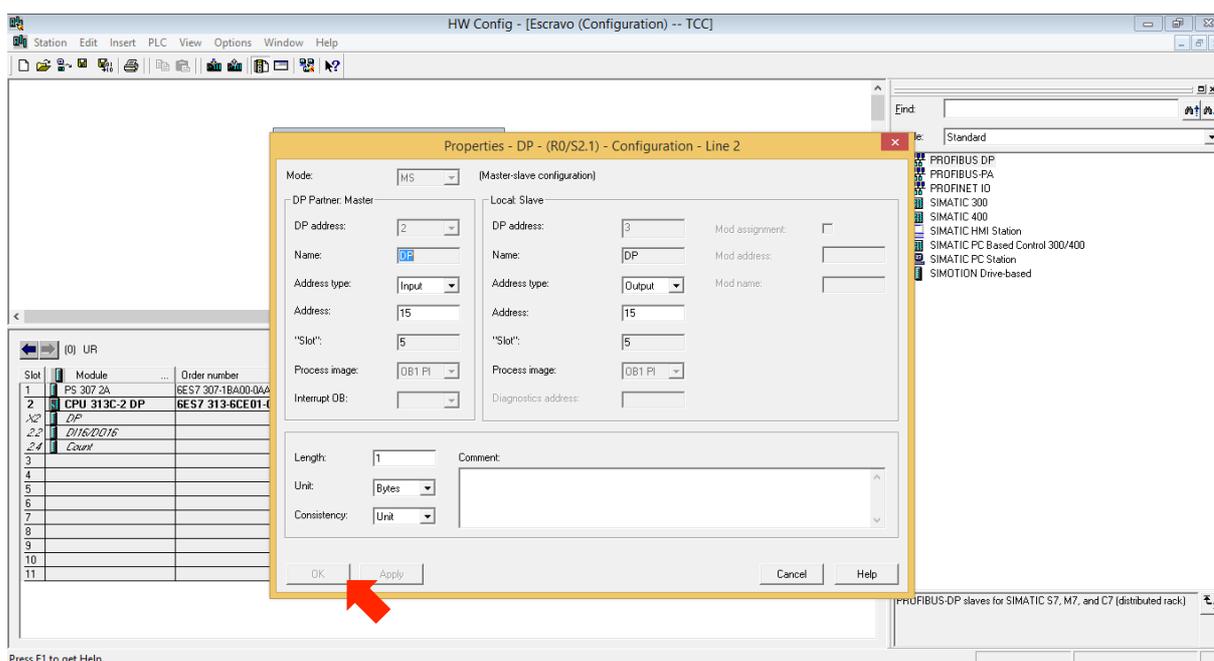


Figura 70 – Confirmar configurações do frame de comunicação.
Fonte: Autoria própria, 2017.

Verifique na janela Properties – DP – (R0/S2.1), figura 71, que existe uma linha 2, referente ao segundo frame criado e clique em OK.

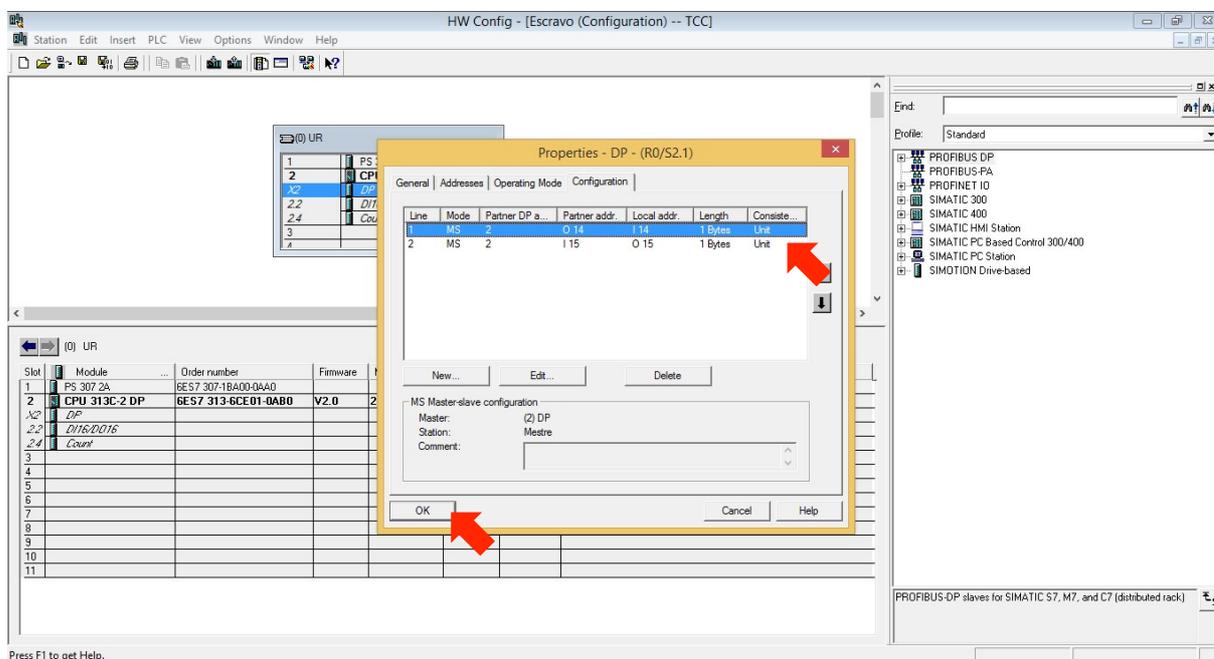


Figura 71 – Verificar segundo frame criado e confirmar configurações.
Fonte: Autoria própria, 2017.

Clique em salva/compilar e fechar, conforme a figura 72.

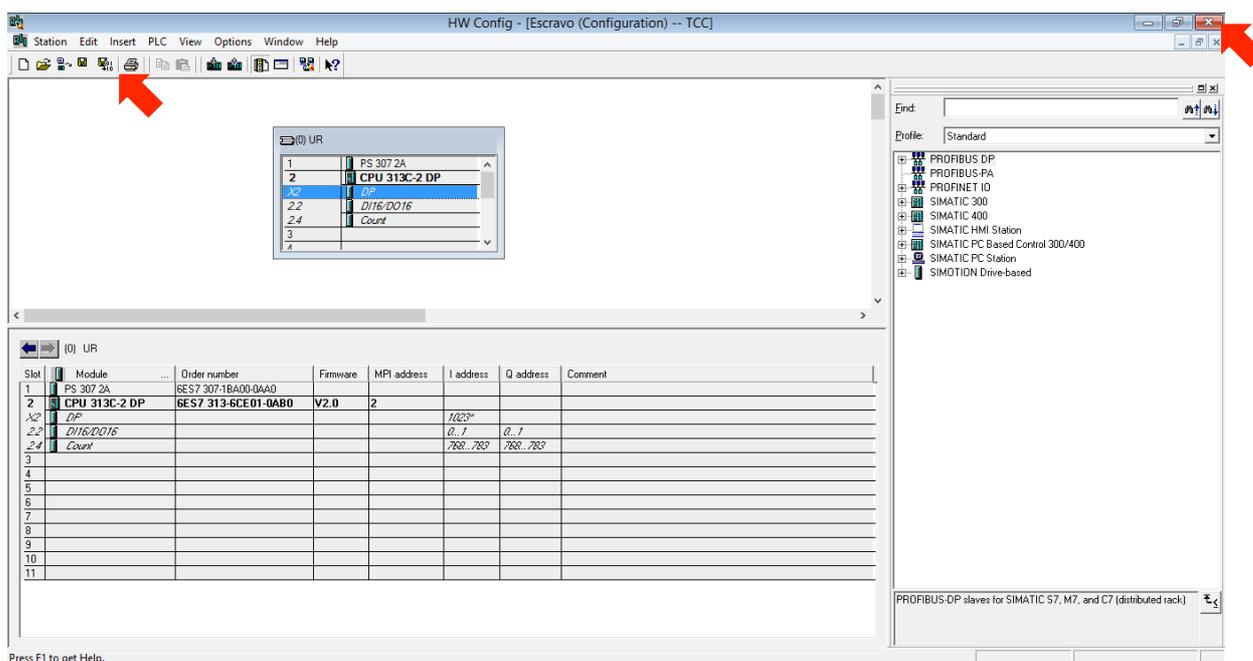


Figura 72 – Salvar e fechar o projeto da estação escravo.
Fonte: Autoria própria, 2017.

Selecione a estação Escravo e realize o download do programa clicando no ícone download, conforme a figura 73.

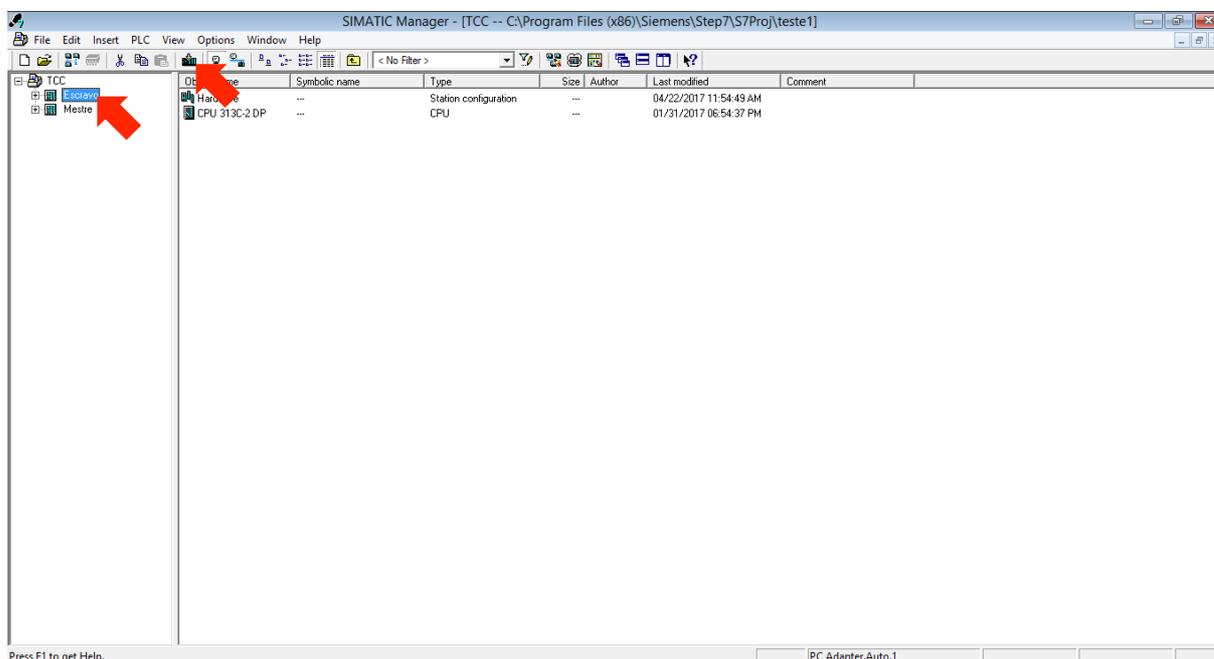


Figura 73 – Download do projeto da estação escravo.
Fonte: Autoria própria, 2017.

Selecione a estação Mestre e realize o download do programa clicando no ícone download, conforme a figura 74.

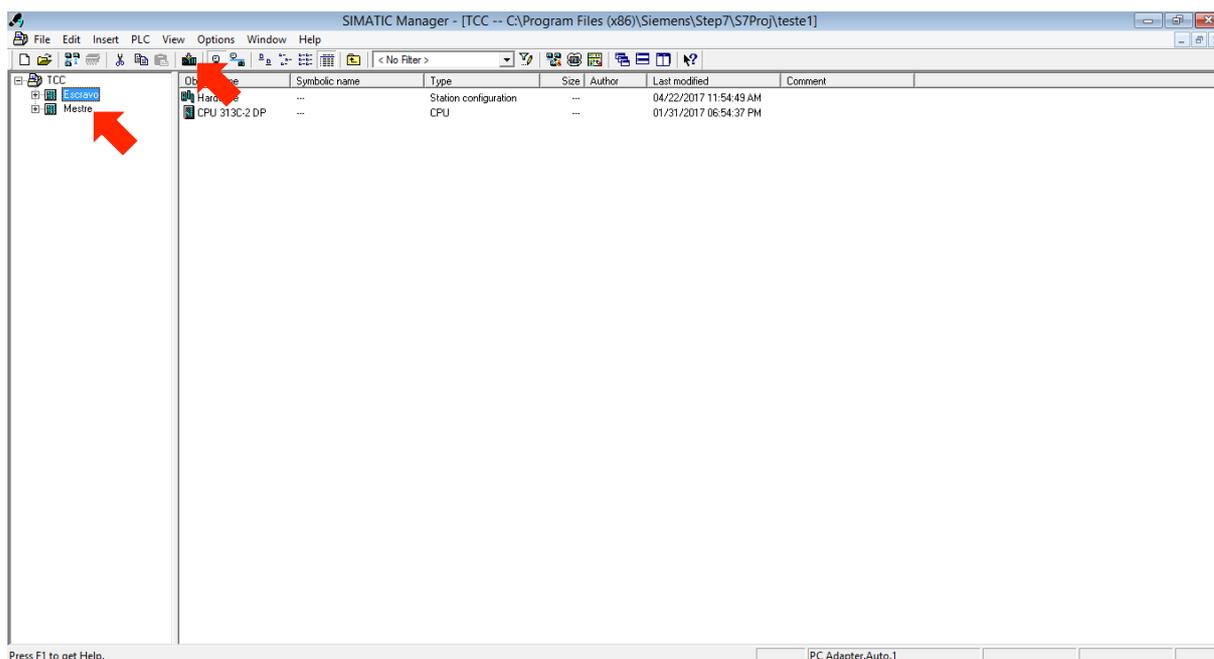


Figura 74 – Download do projeto da estação mestre.
Fonte: Autoria própria, 2017.

Conforme a configuração realizada, toda vez que o byte 14 da estação Mestre for acionado, ou seja, seu estado lógico for igual a 1, o byte 14 da estação Escravo também será acionado, com estado lógico igual a 1 e quando o byte 15 da estação Escravo for acionado, estado lógico igual a 1, o byte 15 da estação Mestre será acionado, com estado lógico igual a 1. A figura 75 indica esta configuração a qual foi realizada nos passos descritos anteriormente.

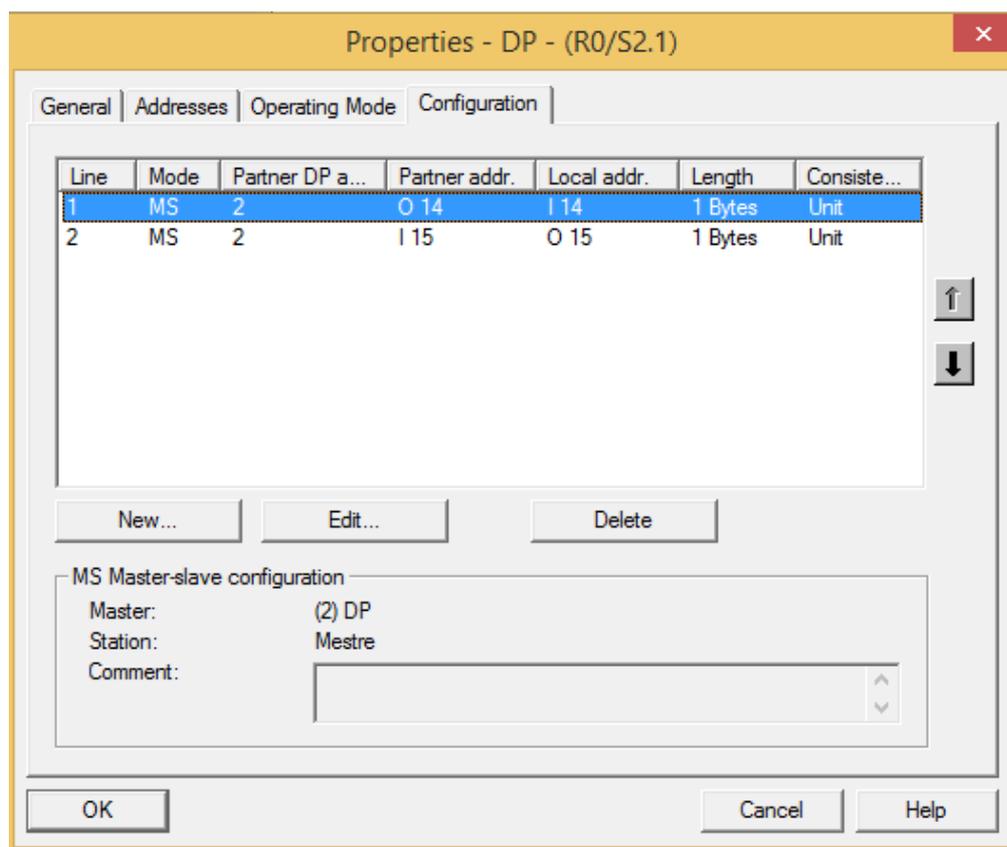


Figura 75 – Tela de configuração dos frames de comunicação.
Fonte: Autoria própria, 2017.

O Quadro 5 mostra a relação entre os bits das estações Mestre e Escravo.

Bytes de Saída			Bytes de Entrada	
Estação	Bits		Bits	Estação
Mestre	14.0	→	14.0	Escravo
...
Mestre	14.7	→	14.7	Escravo
Escravo	15.0	→	15.0	Mestre
...
Escravo	15.7	→	15.7	Mestre

Quadro 5 – Relação entre bits das estações mestre e escravo.

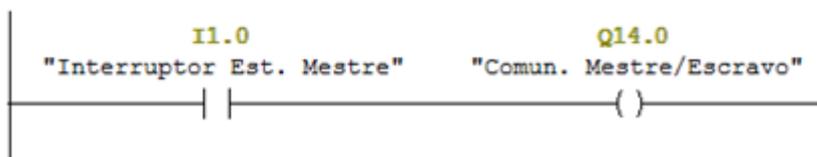
Fonte: Autoria própria, 2017.

As figuras 76 e 77 exemplificam um diagrama LADDER, onde o interruptor I1.0 da Estação Mestre aciona a lâmpada Q1.0 da Estação Escravo e o interruptor I1.0 da Estação Escravo aciona a lâmpada Q1.0 da Estação Mestre.

FC4 : Title:

Comment:

Network 1: Interruptor Estação Mestre aciona Lâmpada Estação Escravo



Network 2: Interruptor Estação Escravo aciona Lâmpada Estação Mestre

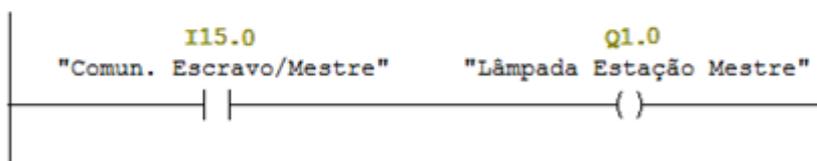


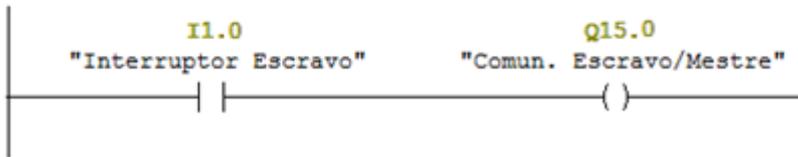
Figura 76 - Programação Software SIMATIC Manager.

Fonte: Autoria própria, 2017.

FC3 : Title:

Comment:

Network 1 : Interruptor Estação Escravo aciona Lâmpada Estação Mestre



Network 2 : Interruptor Estação Mestre aciona Lâmpada Estação Escravo

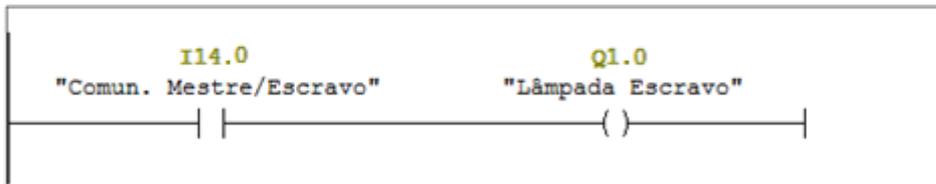


Figura 77 - Programação Software SIMATIC Manager.
Fonte: Autoria própria, 2017.

APÊNDICE B

Este apêndice tem por objetivo detalhar pontos específicos referentes à comunicação via rede Profibus da IHM Eaton 102 que a UTFPR Curitiba possui em seu laboratório de CLP's da SIEMENS. Estes detalhes se dirigem para a comunicação via Rede Profibus que não foram abordados no trabalho utilizado como referência (INOUE E PALAR, 2016).

O primeiro deles é se atentar ao configurar os parâmetros de comunicação via software Galileo. A configuração dos parâmetros da IHM do mestre e do escravo do Semáforo estão detalhados no Quadro 6.

PARÂMETRO	DESCRIÇÃO	IHM DO MESTRE	IHM DO ESCRAVO
Baud rate	Velocidade de Transmissão.	1500k	1500k
Status Refresh [ms]	Intervalo entre atualizações.	10	10
MMI Station No.	Define o número da estação da IHM.	4	5
Default Station No.	Estação em default (não alterado).	2	2
TTR	Taxa de Transmissão (não alterado).	34304	34304
HSA (15, 31, 63, 126)	Máximo número de endereços (número da estação).	126	126
Default Slot	Slot da Estação do CLP (não alterado).	2	2

Quadro 6 - Parâmetros de Configuração das IHM's Mestre e Escravo (exemplo Semáforo)

Conforme citado na página 42 de INOUE E PALAR, deve-se configurar a Velocidade de Transmissão (Baud rate) igual ao dos CLP's que estão em rede.

Além disso, é necessário tomar os devidos cuidados com os números das estações da rede (endereços da Profibus). No exemplo do Semáforo, as estações (MMI Station No.) foram configuradas de acordo com o Quadro 8. Outros números de estação poderiam ser escolhidos, desde que sejam diferentes entre si (isso causaria erro de comunicação). Os números de estações podem ser de 1 a 126 (126 foi configurado no Quadro 6 em HSA).

ESTAÇÃO	NÚMERO DA ESTAÇÃO
CLP Mestre	2
CLP Escravo	3
IHM do Mestre	4
IHM do Escravo	5

Quadro 7 - Número de Estações (exemplo Semáforo)

Os outros parâmetros do Quadro 7 podem ser mantidos como default.

É necessário ter em mente que, se os cabos forem conectados depois da IHM ser ligada, poderá ocorrer um erro crítico de memória. Desta maneira, caso isso aconteça, a IHM deverá ser reiniciada. Para fazer tal processo, deve-se carregar o programa do software Galileo via Ethernet e selecionar a opção Reboot IHM. Isso solucionará o erro.

APÊNDICE C

Este apêndice tem por objetivo detalhar a comunicação entre um CLP Siemens S7-300 com o software WinCC Flexible 2008 através de uma rede MPI.

Após criado um novo projeto, a tela inicial será como demonstrado na figura 78:

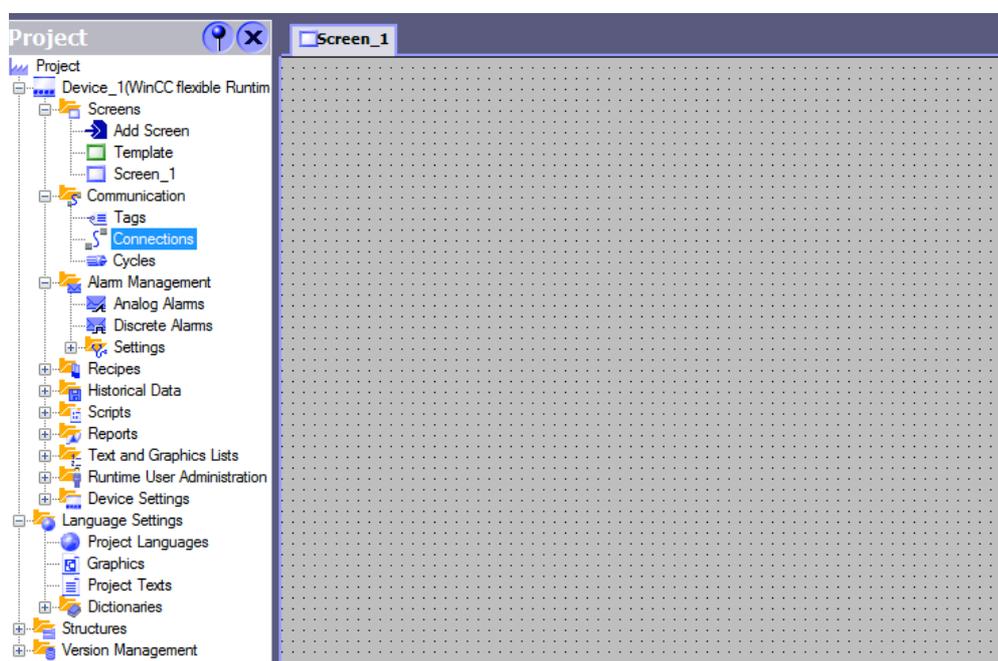


Figura 78 - Tela inicial de um projeto no WinCC.
Fonte: Autoria própria, 2017.

Clique duas vezes em Connections, e uma nova aba se abrirá como aparece na figura 79:

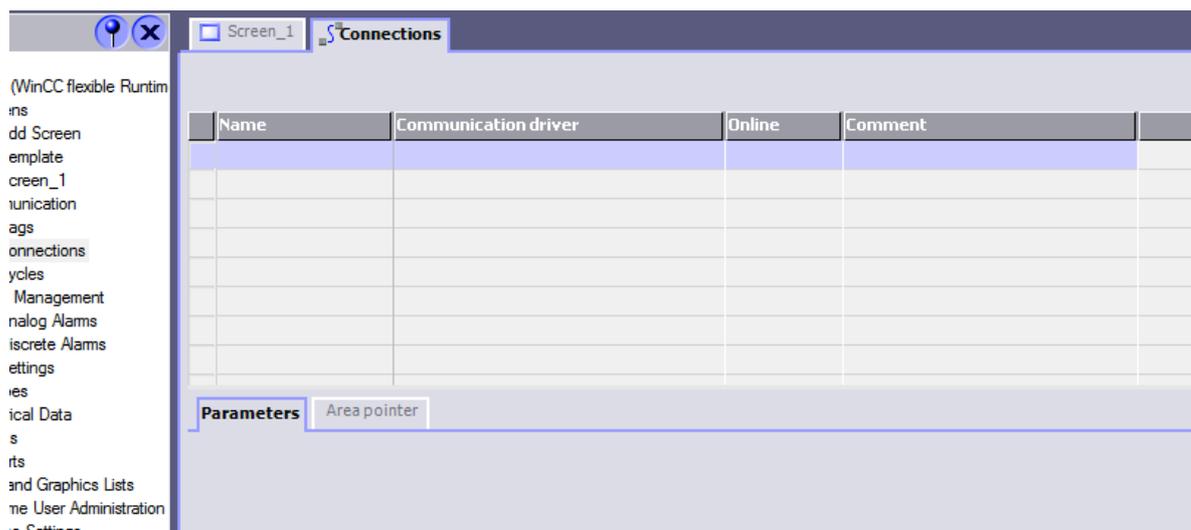


Figura 79 - Aba Connections.
Fonte: Autoria própria, 2017.

Clique duas vezes na primeira linha (linha que está selecionada automaticamente). Será criado automaticamente uma conexão com os parâmetros através para uma rede MPI com um CLP Siemens S7-300 ou S7-400, como pode ser observado na figura 80.

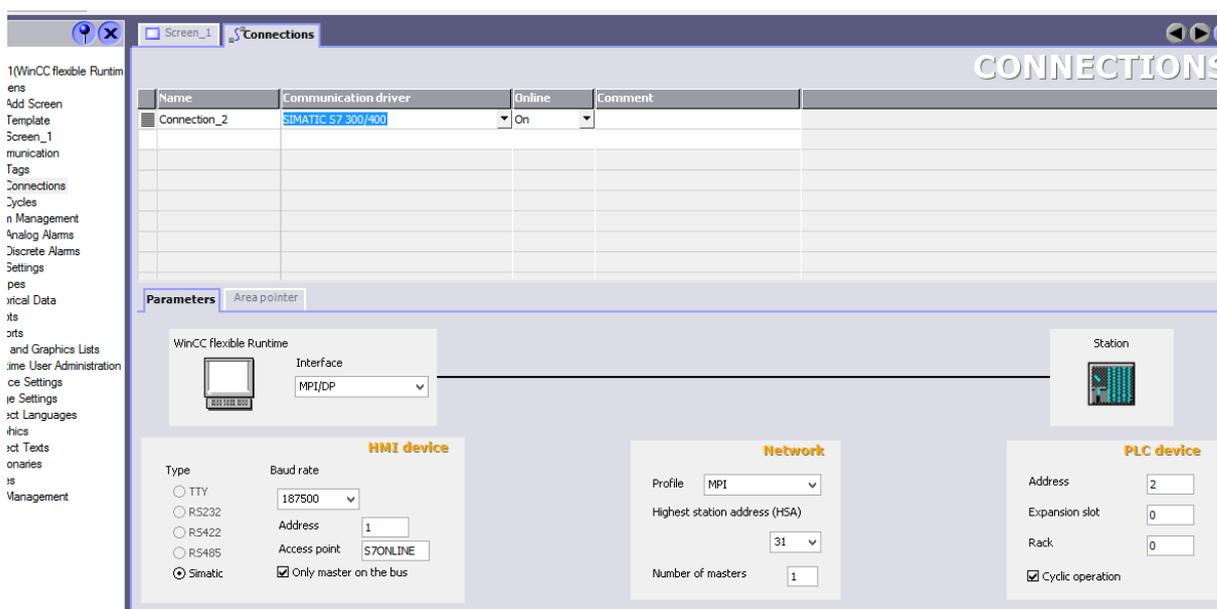


Figura 80 - Parâmetros de conexão.
Fonte: Autoria própria, 2017.

Há outras opções que podem ser alteradas nos parâmetros da conexão, como por exemplo, alterar a Interface para Ethernet, alterar os endereços ou o driver

de comunicação. A figura 81 apresenta os drivers de comunicação que podem ser utilizados com o WinCC Flexible 2008:

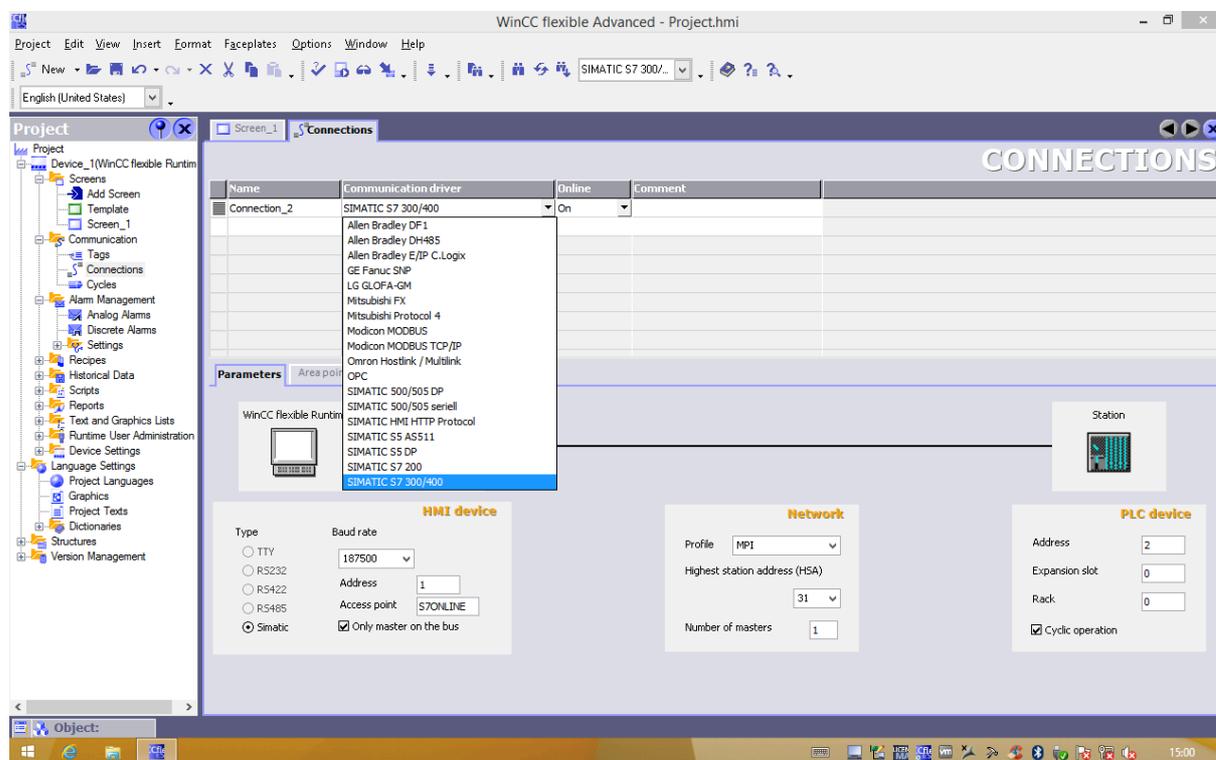


Figura 81 - Drivers de comunicação com o WinCC Flexible 2008.
Fonte: Autoria própria, 2017.