

RICARDO SANTOS PEREIRA

**ESTUDO DA COMUNICAÇÃO ACÍCLICA DENTRO
DE UMA REDE INDUSTRIAL UTILIZANDO
PROFIBUS PA**

**UBERLÂNDIA –MG
2019**

RICARDO SANTOS PEREIRA

**ESTUDO DA COMUNICAÇÃO ACÍCLICA DENTRO
DE UMA REDE INDUSTRIAL UTILIZANDO
PROFIBUS PA**

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

Universidade Federal de Uberlândia – UFU
Faculdade de Engenharia Elétrica

Orientador: Renato Ferreira Fernandes Júnior

**UBERLÂNDIA –MG
2019**

Ricardo Santos Pereira.

Estudo da Comunicação Acíclica dentro de uma Rede Industrial utilizando Profibus PA/ **Ricardo Santos Pereira. - UBERLÂNDIA, 2019**- 100p: il. (algumas color.); 30 cm.

Orientador: Prof. Dr. Renato Ferreira Fernandes Júnior

Trabalho de Conclusão de Curso - Universidade Federal de Uberlândia - UFU
Faculdade de Engenharia Elétrica. **2019**.

Inclui bibliografia.

1. Profibus PA 1. 2. FDT/DTM 2. 2. Sistema de Manutenção 3. I. Orientador.
2. II. Universidade Federal de Uberlândia. III. Faculdade de Engenharia Elétrica. IV. Engenharia de Controle e Automação.

DEDICATÓRIA

Dedico este trabalho aos meus pais Mauricio Alves Pereira e Celma Santos Pereira, pois se não fosse pelos seus esforços, tempo, compreensão, dedicação, todo o investimento e apoio recebido por eles, eu não teria chegado aonde estou hoje.

AGRADECIMENTOS

Aos meus professores do curso de Engenharia de Controle e Automação da Universidade Federal de Uberlândia, por todo o suporte que me deram durante toda esta trajetória.

A minha família e amigos por todo apoio e suporte durante a minha graduação. E por fim ao meu orientador Renato Ferreira Fernandes Júnior, por ter me dado todas as oportunidades e a ajuda necessária para o meu crescimento e amadurecimento, durante todo este período em que trabalhamos juntos, tornando-se um grande amigo.

RESUMO

Nos últimos anos tem se acompanhado que os mercados de instrumentação e automação vêm demandado equipamentos de campo tais como transmissores de pressão e temperatura, conversores, posicionado, atuadores e controladores entre outros com alta performance, confiabilidade, disponibilidade etc., com a intenção de obter melhores resultados como minimizar consumos, reduzir a variabilidade de processos, reduzir custo operacionais e de manutenção, assim como garantir otimização.

Com esse avanço na demanda dos equipamentos de campo com alta performance e na consolidação das redes industriais a proposta do projeto é a implementação de ferramentas de protocolo de padrões abertos para ter acesso aos dados e fazer a parametrização e configuração dos equipamentos de campo (transmissores e sensores), para conseguir obter um plano de manutenção dos equipamentos e analisar a influência que essas ferramentas podem causar na rede pelo excesso de supervisão e informações trafegados na rede.

Para isso foi montado dois sistemas de monitoramento e controle completo utilizando um sistema aberto de manutenção de equipamentos de campo utilizando tecnologia Profibus. O primeiro sistema utilizava o controlador Compact Logix com cartão Prosoft comunicando com quatro instrumentos de campo, supervisão usando o sistema Elipse SCADA e sistema de manutenção usando o software Pactware. O segundo sistema era constituído de um controlador DF95 utilizando os mesmos quatro instrumentos de campo, sistema de supervisão ProcessView e estação de manutenção ProfibusView. Com auxílio de uma ferramenta de escuta de linha foi realizada captura de todos os pacotes de mensagens trocados entre todos os equipamentos, bem como os respectivos tempos da rede. Com isso foi possível fazer um estudo profundo sobre a rede Profibus, além de analisar o desempenho da comunicação cíclica e acíclica entre estes dois sistemas. Os resultados mostraram que a supervisão não apresenta uma mudança significativa nos tempos de scan e a inclusão de novos devices na rede é praticamente linear. Também foi possível notar uma performance melhor do controlador Compact Logix em relação ao DF95.

Palavras chave: Profibus PA, FDT/DTM, Sistema de Manutenção, comunicação acíclica, Mestre Classe 2

ABSTRACT

In recent years it has been observed that the instrumentation and automation markets have been demanding field equipment such as pressure and temperature transmitters converters positioning actuators and controllers among others with high performance reliability availability etc., with the intention of obtaining better results such as minimizing power consumption, reducing process variability reducing operating and maintenance costs as well as ensuring optimization.

With this advance in the demand of high performance field equipment and the consolidation of industrial networks the project proposal is the implementation of open standards protocol tools to access data and make the parameterization and configuration of field equipment (transmitters and sensors) in order to obtain a maintenance plan for the equipment and to analyze the influence that these tools can have on the network due to over supervision and information carried on the network.

For this, two complete monitoring and control systems were assembled using an open field equipment maintenance system using Profibus technology. The first system used the Prosoft Compact Logix card controller communicating with four field instruments, supervision using the Eclipse SCADA system and maintenance system using the Pactware software. The second system consisted of a controller DF95 using the same four field instruments, ProcessView supervision system and ProfibusView maintenance station. With the aid of a line listening tool, all message packets exchanged between all devices were captured, as well as the respective network times. With this it was possible to make a deep study on the Profibus network, besides analyzing the performance of the cyclic and acyclic communication between these two systems. The results showed that supervision does not present a significant change in scan times and the inclusion of new devices in the network is practically linear. It was also possible to notice better performance of Compact Logix controller compared to DF95.

Keywords: Profibus PA, FDT / DTM, Maintenance System, acyclic communication, Master Class 2

Lista de Figuras

Figura 1- Pirâmide hierárquica de automação	19
Figura 2- Ilustração dos níveis de um processo industrial.	20
Figura 3-Exemplo de rede de campo	22
Figura 4-Modelo de referência ISO/OSI aplicado a rede Profibus	24
Figura 5- Segmentação em Instalações Profibus	25
Figura 6 - Ilustração cabo Profibus.....	26
Figura 7- Pinagem completa em conectores Profibus	27
Figura 8- Caractere Profibus.....	28
Figura 9- Rede Profibus DP e PA.....	28
Figura 10 - Rede Profibus DP e PA, topologia estrela	29
Figura 11- Exemplo de diagnóstico no escravo.....	30
Figura 12- Exemplo de telegrama de diagnóstico no Profibus.....	30
Figura 13- Mensagens de Diagnósticos	31
Figura 14- Configuração Bus Parameter Profibus DP/PA	32
Figura 15- Exemplo de perda de comunicação com um escravo	33
Figura 16- Tempo de envio do frame	34
Figura 17- Tipos de Dispositivos rede Profibus	35
Figura 18- Mensagem DPV1 Profibus tipo SD2	36
Figura 19- Exemplo e endereçamento Profibus	38
Figura 20- Serviços da camada de segurança e dados (Data Link Layer).....	38
Figura 21- Ilustração de inicialização da rede Profibus	39
Figura 22- Mensagem DPV1 Profibus tipo SD2	40
Figura 23- Mensagem DPV1 Profibus tipo SD2	41
Figura 24- Tipos de Comunicação Profibus PA: MS0, MS1 e MS2.....	42
Figura 25- Estrutura interna do bloco Transdutor	43
Figura 26- Exemplo Bloco de Entrada Analógico.....	44
Figura 27- Modo de configuração AI	44
Figura 28- Exemplo Bloco de Saída Analógica	45
Figura 29- Configuração Suportada pelo AO	45
Figura 30 - Esquema básico sistema supervisório.....	47
Figura 31- Esquema básico de uma estrutura de um sistema de supervisão	47

Figura 32-Especificações padrão OPC	49
Figura 33- Analogia entre um driver de impressora com a tecnologia FDT/ DTM.....	50
Figura 34- Ilustração da disposição física do projeto	52
Figura 35 - Entrada para termoresistência ou OHM a 4 fios.....	54
Figura 36- RTD 4 fios e OHMs.....	55
Figura 37- Arquitetura do sistema Profibus PA do laboratório.....	56
Figura 38- Configuração da rede de campo no software da PCB	57
Figura 39- Exemplo de ladder no Studio 5000.....	59
Figura 40 - Bloco PID Studio 5000	60
Figura 41- Rede montada no laboratório a partir do sistema SMAR	61
Figura 42- Tela configuração no Software Syscon	62
Figura 43- Tela da configuração dos equipamentos na rede pelo software Syscon.....	62
Figura 44- Parâmetros de configuração FI303	63
Figura 45- Parâmetros de configuração TT303	63
Figura 46- Blocos Funcionais utilizados no projeto	64
Figura 47-Controle AI-PID-AO construída no Syscon	65
Figura 48-Tela de Configuração do PACTware.....	67
Figura 49- Ferramenta SMAR.....	68
Figura 50 -Tela do Supervisório usando Process View.....	69
Figura 51- Exemplo de captura do dado pelo Sniffer.....	70
Figura 52- Tela de Livelist do Software Profitrace	70
Figura 53- Frames Profibus PA.....	71
Figura 54- Frames Profibus DP.....	71
Figura 55- Comparação entre SCADA e FDT/DTM	72
Figura 56- Tela de configuração do PACTware.....	73
Figura 57- Mensagem de Escrita no bloco Display TT303.....	74
Figura 58- Captura dos dados da Rede com 2 devices	75
Figura 59- Diferença de tempo de escrita em 2 parâmetros simultaneamente	75
Figura 60- Captura através do software FBview	76
Figura 61- Captura depois de ser decodificada pelo código.....	76
Figura 62- Tempos da rede para 1 device configuração Rockwell	78
Figura 63- Tempos da rede para 1 device configuração SMAR	79
Figura 64- Boxsplot dos dados da rede com 1 <i>device</i>	80
Figura 65- Gráfico em forma de boxplot comparando o tempo de rede para 2 <i>devices</i>	81

Figura 66- Gráfico em forma de boxsplot comparando o tempo de rede para 3 <i>devices</i>	82
Figura 67- Gráfico em forma de boxsplot comparando o tempo de rede para 4 <i>devices</i>	83
Figura 68- Gráfico comparando a diferença de tempo da rede com supervisão e sem supervisão com 1 <i>device</i>	84
Figura 69-Gráfico comparando a diferença de tempo da rede com supervisão e sem supervisão com 2 <i>devices</i>	85
Figura 70- Comparação do tempo de scan da rede no ambiente SMAR e ambiente Rockwell	86
Figura 71-Captura da rede quando ocorreu uma falha no device com endereço 102	87
Figura 72-Comparação do tempo de scan da rede com diagnóstico e sem diagnóstico com 2, 3e 4 <i>devices</i>	88
Figura 73-Gráfico comparando o tempo real pelo tempo simulado.....	89

Lista de Tabelas

Tabela 1- Serviços SAPs existentes no Profibus	37
Tabela 2- Parametrização dos dispositivos Profibus PA	58

LISTA DE SIGLAS

FDT - *Field Device Tool*

DTM - *Device Type Manager*

SACADA - *Supervisory Control and Data Acquisition*

CLP – *Controle Lógico Programável*

RS-485- *Recommended Standard 485*

OLE - *Object Linking and Embedding*

DCOM - *Distributed Component Object Model*

OPC – *OLE for Process Control*

DP – *Decentralized Peripherals*

PA - *Process Automation*

Sumário

1 INTRODUÇÃO	16
1.1 Justificativa	17
1.2 Objetivo.....	18
2 REFERENCIAL TEÓRICO	19
2.1 Redes Industriais	21
2.2 Protocolo Profibus	23
2.2.1 Meio Físico	25
2.2.2 Topologia	28
2.2.3 Diagnósticos	29
2.2.4 Temporização Profibus	31
2.3 Características Profibus DP	34
2.4 Características Profibus PA	40
2.4.1 Blocos PA	42
2.5 Sistema de Supervisão	46
2.5.1 Sistema Supervisório.....	46
2.5.2 Tecnologia OPC	48
2.5.3 Tecnologia FDT/DTM	49
3 PROCEDIMENTOS METODOLÓGICOS	52
3.1 Descrição dos Equipamentos de Campo	53
3.1.1 TT303 – Transmissor de Temperatura (SMAR).....	53
3.1.2 FI303 – Transmissor de Corrente (SMAR).....	54
3.1.3 644 – Transmissor de Temperatura (Rosemount).....	54
3.1.4 2051 – Transmissor de Pressão(Rosemount)	55
3.1.5 Siemens Simatic DP/PA Coupler CLP S7-300.....	55
3.2 Sistema de Controle e Gateway	55
3.2.1 Cenário 1 – Configurando Controlador Rockwell Compact Logix	56

3.2.2 Controlador PID <i>RSlogix</i>	59
3.2.3 Cenário 2 – Configurando Ambiente SMAR.....	60
3.2.4 Controlador PID <i>Syscon</i>	64
3.3 Nível de Supervisão	65
3.3.1 Tecnologia FDT/DTM	66
3.3.2 Ambiente Rockwell com PACTware	66
3.3.3 Ambiente SMAR <i>ProfibusView</i>	67
3.3.4 Sistema SCADA	68
3.4 Metodologia de Validação	69
4 RESULTADOS E DISCUSSÕES	72
4.1 Comparação entre as ferramentais SCADA E FDT/DTM	72
4.1.1 Estação de Manutenção com o software PACTware	73
4.1.2 Análise de Rede através das ferramentas de escuta de linha <i>Profitrace</i>	73
4.1.3 Análise de Rede através das ferramentas de escuta de linha <i>FBView</i>	76
4.2 Análise do tempo de <i>scan</i> da Rede	77
4.2.1 Sistema Rockwell com 1 <i>device</i>	79
4.2.2 Sistema Rockwell com 2 <i>devices</i>	80
4.2.3 Sistema Rockwell com 3 <i>devices</i>	81
4.2.4 Sistema Rockwell com 4 <i>devices</i>	82
4.2.5 Sistema SMAR com 1 <i>device</i>	83
4.2.6 Sistema SMAR com 2 <i>devices</i>	84
4.2.7 Comparação entre Sistema SMAR e Sistema Rockwell	85
4.3 Diagnóstico na Rede	86
4.4 Simulação de tempo da rede	89
4.5 Discussões.....	90
5 CONCLUSÃO.....	91
6 REFERÊNCIAS BIBLIOGRÁFICAS	93

APÊNDICE A	97
-------------------------	-----------

1 INTRODUÇÃO

Atualmente, a tecnologia da informação tornou-se uma ferramenta muito útil e até indispensável na indústria devido as características da chamada indústria 4.0. Na automação industrial moderna, os dados precisam se tornar parte integrante do sistema operacional e de controle. São necessários tecnologias que forneça uma interface ideal para os programas de planejamento e manutenção executados na fábrica (Data Center Dynamics, 2017).

Dentro do conceito atual de indústrias 4.0, as tecnologias de comunicação modernas procuram convergir o nível gestão com o nível de chão de fábrica como uma forma de realizar a integração dos sistemas de tecnologia da informação (TI) com os sistemas existentes no chão de fábrica, chamados de sistemas de tecnologia operacional (TO) (Data Center Dynamics, 2017). Esta integração visa melhorar o desempenho, reduzir custos e minimizar riscos. A TI engloba todos os componentes de software e hardware que trabalham no nível de gestão da informação indo desde o planejamento, execução até a verificação e alteração das várias etapas que compõem o processo industrial (YAQOOB; AHMED; HASHEM; AHMED; GANI; IMRAN; GUIZANI, 2017).

Atualmente, é notória a importância de redes de automação digital pelo motivo da rede estar presente nos mais diferentes níveis no ambiente industrial. As redes digitais se comunicam entre si desde o chão de fábrica até o ambiente corporativo.

No contexto das redes de automação industrial, a algumas décadas, os protocolos evoluíram para os chamados “fieldbuses”, onde são protocolos digitais padronizados que permitem inúmeras vantagens entre elas a interoperabilidade e a intercambiabilidade entre os equipamentos. Existem diferentes protocolos de acordo com os perfis de comunicação. Entre eles, podem citar para aplicações de controle de processos os protocolos Profibus, Foundation Fieldbus, Profinet e para os perfis de controle de manufatura os protocolos Devicenet, ASI, Controlnet e Ethernet/IP (Kumar,2014).

Ainda nesse contexto das redes de campo, existe um aumento no número de transmissores e atuadores simples e complexos e todos estes instrumentos precisam ser integrados ao sistema e monitorados por diferentes pontos do sistema seja através de acesso loco ou remoto. A integração eficiente e economicamente viável de dispositivos requer tecnologia padronizada e multiprotocolo para que as informações sobre o dispositivo possam ser disponibilizadas em diferentes fabricantes. As soluções atuais - EDDL (*Electronic Device*

Description Language) e FDT (*Field Device Technology*) e mais recentemente o FDI possui soluções multiprotocolos que permite várias ações em diferentes tempos de vida dentro do *device* como configuração, comissionamento, diagnostico e calibração de um instrumento seja ele simples ou complexo (FDI Cooperation,2016).

Com o surgimento de equipamentos de campo inteligentes, uma grande variedade de dados são gerados por esses equipamentos como: dados de configuração através de ferramentas de manutenção e controle através das ferramentas de supervisão, esses dados podem ser gerados para usuários ou mesmo para outras aplicações.

Com isso surge os sistemas PIMS (*Plant Information Management Systems*) e MES (*Manufacturing Execution Systems*), eles proporcionam uma visão unificada do processo solucionando o problemas de fragmentação de dados nas plantas.

O sistema PIMS são softwares que adquirem informações contínuas de processos (temperaturas, pressões, vazões, etc.) de diversos sistemas, os armazenam em banca de dados de históricos, e geram esses dados através de relatórios dos mais diversos tipos.

O sistema MES é similar ao PIMS são softwares que adquirem dados discretos de diferentes fontes e os armazenam em bancos de dados e geram esses dados através de relatórios, ele proporciona uma visão unificada de todo o processo.

1.1 Justificativa

O tema foi escolhido baseado no conhecimento adquirido através do curso na disciplina de Redes Industriais onde foi adquirido o conhecimento do protocolo Profibus e toda parte de equipamentos como o CLP e os sensores e atuadores que serão abordados nesse projeto, e de sistema supervisórios onde foi estudado o protocolo OPC e de sistemas supervisórios SCADA.

O padrão Profibus PA que cobre a automação de processos é rica em parametrização e diagnostico onde cada instrumento consegue trazer várias informações pertinentes sobre a variável do processo que está sendo medida.

O laboratório de Redes industriais e Supervisório possui vários equipamentos de campo no protocolo Profibus PA, porém ainda não existe uma forma amigável de parametrização destes instrumentos. Atualmente, a única forma de parametrização dos equipamentos é através de uma chave magnética atuando diretamente no transmissor.

Existem vários padrões de supervisão e manutenção de equipamentos de campo, como a tecnologia FDT/DTM utilizada por ferramentas de manutenção e as ferramentas de supervisão SCADA. No ambiente industrial essas ferramentas estão em uso simultaneamente então é

necessário um estudo que mostre a influência no tempo da rede com o aumento no nível de informação trafegada na rede devido a essas ferramentas.

1.2 Objetivo

O objetivo principal que caracteriza este trabalho é o estudo e a análise da comunicação acíclica dentro de uma rede Profibus PA. Para isso será estudado tecnologias de mercado que utilizam o protocolo para fazer manutenção dos equipamentos de campo remotamente.

Como objetivos específicos do projeto serão realizados os seguintes tópicos:

- Estudo sobre o Protocolo Profibus DP e PA, formato de mensagens e tempos de rede.
- Estudo dos transmissores profibus PA do laboratório.
- Estudo das ferramentas de Supervisão e Manutenção seguindo os padrões FDT/DTM e OPC
- Montagem de estação de Supervisão e controle utilizando diferentes transmissores PA com equipamentos do laboratório
- Análise da influência da comunicação acíclica em comparação da comunicação cíclica em diferentes aspectos dentro de uma rede Profibus PA.

O projeto está dividido em 5 capítulos que serão apresentados a seguir:

O capítulo 1 é apresentado uma rápida introdução explicando a motivação do projeto.

O capítulo 2 irá apresentar toda teoria necessária para ter um bom entendimento do projeto.

O capítulo 3 é apresentado ao leitor toda metodologia utilizada para montagem dos experimentos para coleta dos dados e toda configuração das ferramentas que foram utilizadas no projeto.

Para o capítulo 4 será trazido os resultados feitos através de análises e gráficos sobre os temas abordados durante o texto. Por fim o capítulo 5 mostra ao leitor as conclusões finais sobre o projeto.

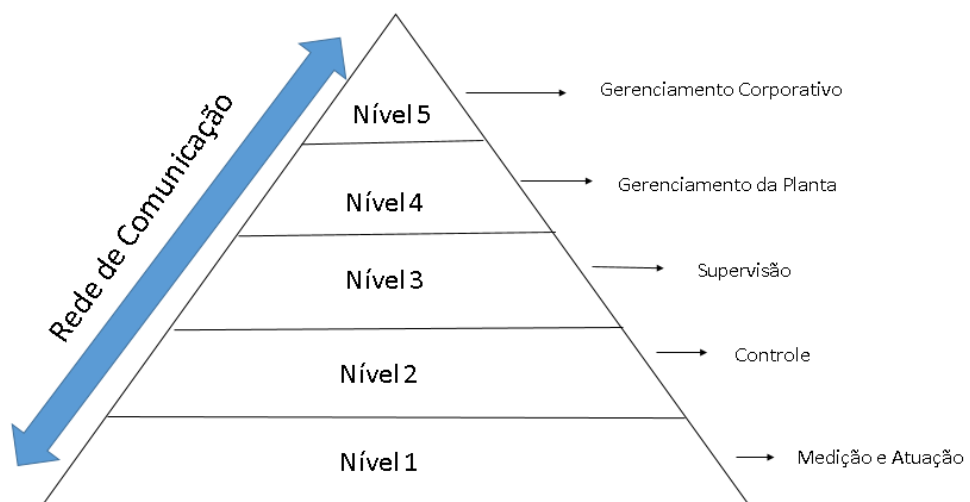
2 REFERENCIAL TEÓRICO

Atualmente as redes fazem um papel importante na comunicação entre os diferentes níveis de uma pirâmide industrial, e a quantidade de dados trafegados nestas redes continuam a aumentar devido a quantidade de equipamentos conectados e a quantidade de informações que podemos tirar destes equipamentos.

Os protocolos modernos de rede são baseados no padrão Ethernet e nos protocolos derivados deste, como TCP/IP. E estes padrões permitem a divisão lógica da rede em subredes, o que torna possível implantar redes distribuídas de controle de campo, aumentando a flexibilidade como os dados podem ser roteado entre as redes lógicas sem sobrecarregar o controladores. Neste caso, ao invés do controlador, a própria rede pode ser considerada como o centro da comunicação, possibilitando configurar os dispositivos diretamente, sem necessidade de passar pelo controlador, e também permitir múltiplos mestres se comunicando com o dispositivo de campo (*Kjellsson; Vallestad; Steigmann; Dzung, 2019*).

Com isso as tarefas de um sistema de automação aparecem separadas em vários níveis hierárquicos, baseado nas características funcionais e temporais específicas com cada nível. O níveis hierárquicos podem ser representados em um pirâmide hierárquica de automação industrial apresentada na Figura 1.

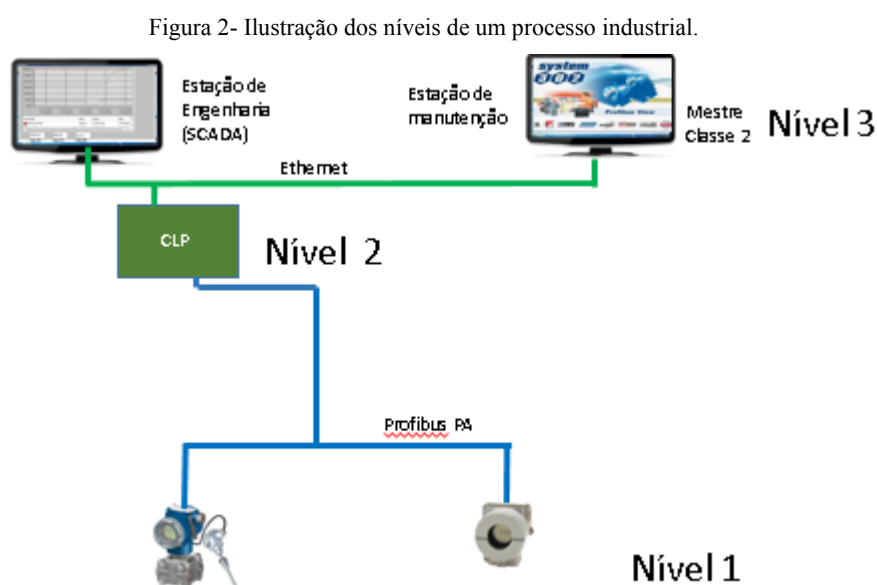
Figura 1- Pirâmide hierárquica de automação



Fonte: Autor, 2019

O nível 1 é onde estão os equipamentos que executam o controle automático e distribuído da planta como sensores, transmissores, motores, etc. O nível 2 é onde se encontramos equipamentos que realizam o controle automático das atividades da planta, como por exemplo os CLPs (Controle Lógicos e Programáveis)

O terceiro nível encontra-se o nível de supervisão, é formado pelas ferramentas SCADA e as ferramentas de manutenção dos equipamentos de campo, esse nível é responsável por monitorar as variáveis do processo e fornecer esses dados aos operadores em tempo real e de forma otimizada. A figura 2 ilustra cada nível mostrado acima.



Fonte: Autor,2019.

O quarto nível 4 é responsável pela programação e planejamento da produção, como realizando o controle, agendamento e a logística de suprimentos (Marchioro Bertelli, G. P, 2015, p.7).

O quinto e último nível é onde fica a administração dos recursos da empresa, neste nível estão os softwares para gestão de venda entre outros para auxiliar nas decisões que influenciam toda a empresa. (Marchioro Bertelli, G. P, 2015, p.7)

E a comunicação entre todos esses níveis é feito através de um protocolo de comunicação industrial, esses protocolos possibilita por exemplo que controladores, sensores e atuadores troquem dados de forma confiável e ágil para supervisionar ou controlar plantas indústrias com isso à possibilidade de comunicação entre dispositivos do mesmo nível hierárquicos ou em níveis diferente. A grande maioria das instalações industriais ainda tem um legado de redes não ethernet como Profibus DP e PA, e que a tendência é perdurar por um longo

tempo ainda e desta forma é importante analisar o comportamento destas redes com os métodos modernos de Supervisão.

Baseado nas tendências tecnológicas em automação industrial, aqui serão apresentados conceitos relacionados ao sistema de supervisão, redes industriais e também as tecnologias de supervisão nestes ambiente, os quais servirão de base para a compreensão deste trabalho.

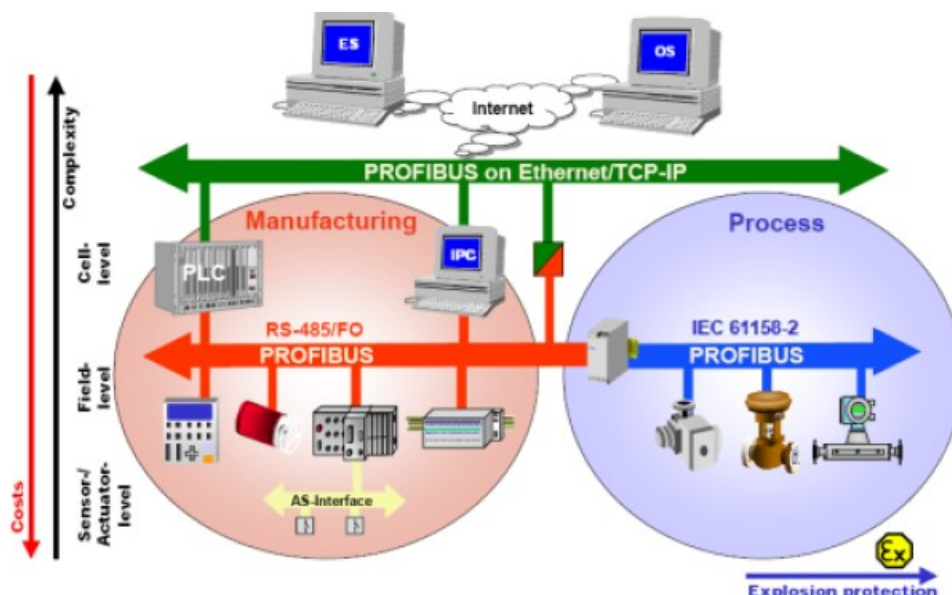
2.1 Redes Industriais

Atualmente nas fábricas, é importante saber o que se produz, quanto se produz e a disponibilidade dos recursos de produção. O primeiro passo para a automação industrial foi proporcionado através da utilização de equipamentos inteligentes em máquinas ou processos para controle ou na aquisição de dados. Isso levanta um novo ponto, como seria feito a interligação entre os equipamentos de diferentes níveis, então a partir dessa aquisição teve início a construção de um sistema de aquisição de dados que acompanhasse em tempo real as informações da produção dos equipamentos envolvidos ou do estado do processo em funcionamento (PESSA,2006, p2.7).

Uma rede de comunicação para sistemas de automação industrial é um conjunto de sistemas independentes, autônomos e interconectados de forma a permitir a troca de informações entre si. Uma rede oferece os meios físicos e lógicos que permitam a integração do sistema através da troca de informações. As redes para sistemas de automação podem ser classificadas, conforme sua finalidade em: Redes de Campo; de Controle e de Gerenciamento de Fábrica. Neste momento daremos uma ênfase maior nas redes de campo (PESSA,2006, p2.8).

A figura 3 a seguir ilustra a interligação dos equipamentos através da rede de campo.

Figura 3-Exemplo de rede de campo



Fonte: Adaptado de (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA,2012)

Atualmente existe vários padrões de redes industriais que pode ser classificada em indústria de processo e indústria de manufatura (KUMAR,2014).

A indústria de processos lida com processos contínuos e mais complexos, que utilizam variáveis contínuas como Pressão, Vazão, Temperatura, Densidade, PH. A rede para esse tipo de processo de automação é de baixa velocidade e geralmente são em áreas classificadas, alguns exemplos deste tipo de rede são Profibus PA, *Foundation Fieldbus*. Por outro lado a indústria de manufatura atuam em processos que possuem eventos discretos como sensores de posição, drivers de motores, válvulas ON/OFF, entre outros. A rede utilizada em processos de manufatura possui características mais rápida por possuírem menos dados trafegando na rede. Como os principais exemplos deste tipo de rede são *Devicenet*, *ControlNET*, ASI, Profibus DP, Profinet, Ethernet/IP.

As tecnologias mais modernas de automação industrial convergem para a rede ethernet como espinha dorsal na comunicação entre os equipamentos. Dentre estes protocolos, a comunicação confiável em tempo real, usando os protocolos PROFINET e Ethernet/IP, tem sido o estado-da-arte dos protocolos de rede em aplicações determinísticas e de tempo crítico como em controle de movimento.

Nos tópicos seguintes são apresentadas as principais características e funcionalidades dos protocolos Profibus DP e PA pois serão importantes para o entendimento deste trabalho.

2.2 Protocolo Profibus

O Profibus é um dos protocolos que pertencem a ao grupo dos “fieldbuses” abertos e independentes de fornecedores, que fazem a integração de equipamentos de diversos fabricantes na mesma rede ou seja possui interoperabilidade, graças a padronização do protocolo pode ser interligado equipamentos de diversos fabricantes com uma comunicação perfeita entre eles e intercambialidade que é a possibilidade de colocar o mesmo equipamento de um outro fabricante na rede capaz de funcionar da mesma forma que o primeiro. . Possui uma série de vantagens em relação à tecnologia 4-20 mA, onde resumidamente pode-se citar, dentre elas (VIVACE,2016, p.5):

- Fácil cabeamento com redução de custos;
- Simples operação;
- Aplicações em área de risco;
- Altas taxas de comunicação no Profibus-DP;
- Poderosas ferramentas de configuração/parametrização e gerenciamento de ativos;
- Tecnologia aberta.

O padrão Profibus está dentro das normas IEC61158 e EN50170 e, conta com 3 tipos de tecnologias: FMS, DP, PA e mais recentemente incorporado o Profinet. O Profibus DP opera com alta velocidade e conexão de baixo custo, utilizado para comunicação de sistemas de controle de automação e seus respectivos I/O's. O Profibus PA possui todas as características do Profibus DP e acrescenta os parâmetros e blocos de função para dispositivos de automação e processo, como transmissores, válvulas e posicionadores que podem trabalhar em áreas classificadas, que ocorrem risco de explosão (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012).

Dedicada à área de processos e amplamente utilizada em comunicação digital bidirecional, permitindo a implementação de sistemas de controle de processos tecnologicamente avançados. Para comunicação entre equipamentos complexos, como PLCs foi criado o Profibus FMS, porem ele caiu em desuso. O protocolo mais recente Profinet (2005) é um padrão de rede Ethernet e visa a integração das redes distribuídas DP, PA com os outros níveis dentro da pirâmide de automação. (CAMARGO, P. R. ,2012) (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012).

Existem três versões do Profibus chamadas DP-V0, DP-V1 e DP-V2. A versão DP-V0 define as funções básicas do protocolo e é utilizada em ambos os padrões DP e PA. Ela define a troca de dados cíclicas, configuração e parametrização de dispositivos e diagnostico padrão.

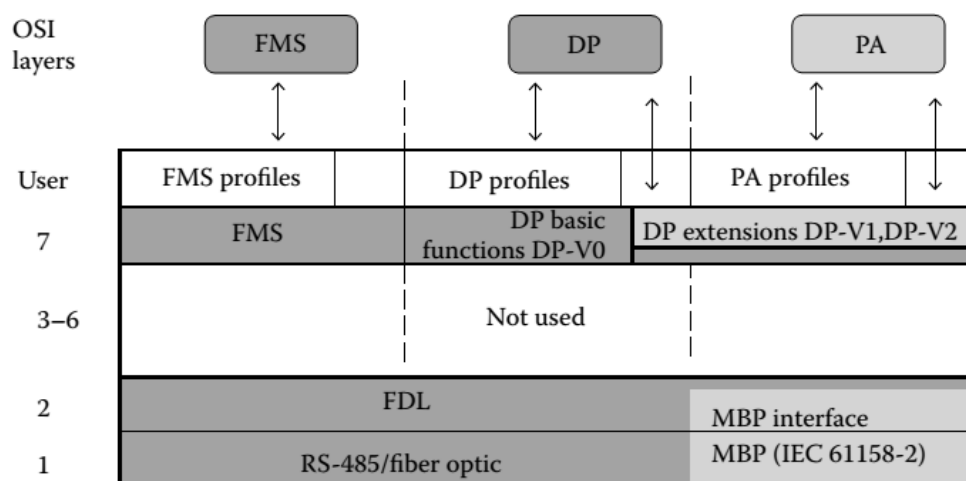
A extensão DP-V1 foi criada incorporada na norma EN50170 em 1998 e foi criada principalmente para atender as características do Profibus PA. Por fim o DP-V2 introduziu melhorias adicionais que são utilizadas no Protocolo Profinet para comunicação com o Profibus DP e PA.

O Profibus é um sistema multi-mestre e permite a operação conjunta de diversos sistemas de automação, engenharia ou visualização, com seus respectivos dispositivos periféricos (por ex. I/O's) (CASSIOLATO; TORRES; CAMARGO, 2012, p.10).

O protocolo Profibus difere seus dispositivos entre mestre que fazem a comunicação de dados nos barramento, o mestre pode enviar mensagens, sem uma requisição externa, sempre que possuir direito de acesso ao barramento, e dispositivos escravos que são dispositivos remotos como I/O , válvulas, transdutores, próprios equipamentos Profibus PA entre outros, diferente do mestre eles não têm direito de acesso ao barramento em outras palavras só podem enviar mensagens ao mestre ou reconhecer mensagens recebidas quando solicitados (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012).

A arquitetura da rede Profibus é baseada em um protocolo de rede que segue o modelo ISO/OSI. No Profibus DP são utilizadas as camadas 1 e 2 juntamente com interface do Usuário. Por outro lado, no Profibus PA, além dessas, é utilizada também a camada 7, com isso essa arquitetura garante uma transmissão de dados eficiente e rápida. Abaixo, será apresentado uma breve explicação sobre cada camada:

Figura 4-Modelo de referência ISO/OSI aplicado a rede Profibus



Fonte: Adaptado de (KUMAR, 2014)

- A camada 1 engloba o meio físico onde a mensagem é transportada, tipicamente um cabo blindado e par trançado. Ela descreve a tecnologia de transmissão de dados, a pinagem

dos conectores e os parâmetros técnicos e elétricos que devem ser cumpridos (*ProfibusInstallation Guideline*,2006). É nesta camada que ocorre o transporte dos dados como um conjunto serial de bits entre dois equipamentos terminais, através dos meios físicos RS-485 ou fibra ótica (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012).

- A camada 2 é a camada de Enlace, é onde são formados os telegramas de mensagem. É feito o controle de quando e por onde cada mensagem irá trafegar, isso é feito para evitar colisões de equipamentos que queiram transmitir ao mesmo tempo (BERGE,2002).

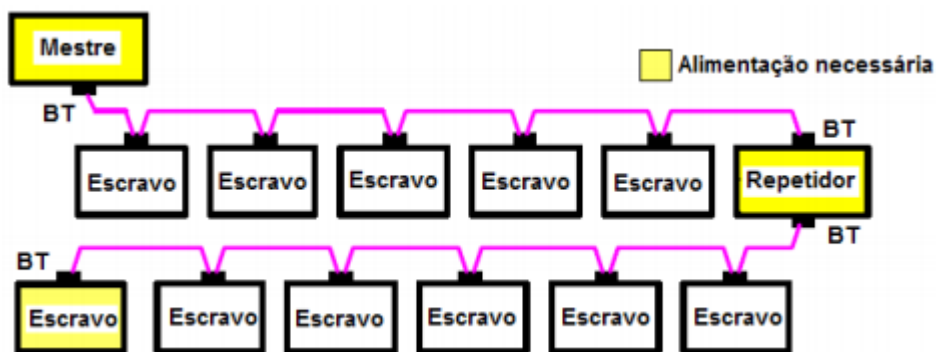
- A camada 7 é a responsável pela interface homem máquina. Acima da camada 7 é onde fica a funcionalidade do instrumento tal como medição, atuação, controle ou a interface de operação de um configurador (BERGE,2002).

2.2.1 Meio Físico

Um sistema de comunicação industrial é muito influenciado pela escolha do meio de transmissão disponível. É preciso que atendam aos requisitos de uso genérico tais como alta confiabilidade de transmissão, grandes distâncias a serem cobertas e alta velocidade de transmissão e também as exigências específicas da área de automação de processos, tais como operação e área de classificação, transmissão de dados e alimentação no mesmo meio físico, entre outros (VIVACE,2009, p.54).

Um sistema com protocolo Profibus aceita conectar até 126 estações. Para ser possível funcionar com este número de estações, o sistema deve ser dividido em segmentos individuais, conectados por repetidores (VIVACE,2009, p.55).

Figura 5- Segmentação em Instalações Profibus

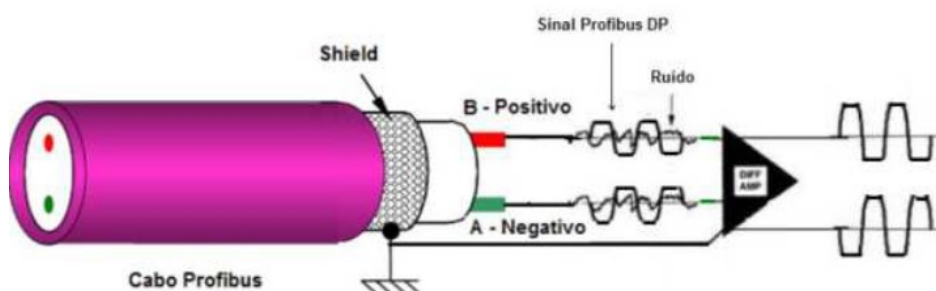


Fonte: Adaptado de (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012).

Existem três tipos básicos de meio físicos de comunicação disponível no Profibus: RS485, Fibra Óptica e MBP.

No Profibus DP o padrão RS485 é a tecnologia de transmissão mais frequentemente utilizada, em especial em sistemas de automação da manufatura. O sinal é diferencial half-duplex e trabalha com taxas de comunicação de 9600 a 12Mbps. Os fios de transmissão de dados em par trançado blindado conforme mostrado na figura 6.

Figura 6 - Ilustração cabo Profibus

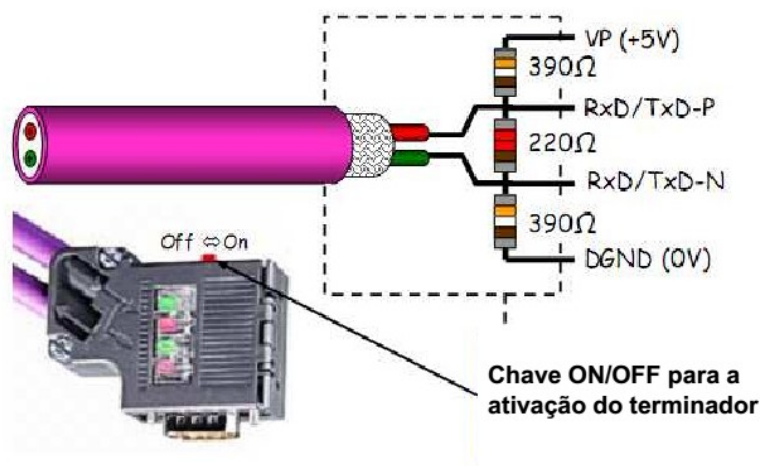


Fonte: Adaptado de (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA,2012)

No barramento RS485, os dispositivos são conectados a uma estrutura de tipo barramento linear. Em um único segmento poder ser conectados até 32 estações (mestre ou escravos). As distancias máximas de um segmento depende do baudrate utilizado variando de 100m até 1000 metros para baudrate mais baixos. No barramento é necessário que exista um circuito terminador ativo no início e no fim de cada segmento (VIVACE,2009, p.55).

Na maioria das vezes esses terminadores são encontrados nos próprios conectores de barramento ou nos dispositivos de campo. São utilizados repetidores quando exista a necessidade de se conectar mais de 32 estações, ou quando a distância total entre estações passar um determinado limite (VIVACE,2009, p.55). Na figura 7 é o conector padrão Profibus DP e o terminador ativo.

Figura 7- Pinagem completa em conectores Profibus



Fonte: Adaptado de (VIVACE,2009)

Ainda no Profibus DP, pode ser utilizado também a Fibra Ótica para aplicações em sistemas que demandam grande imunidades às interferências e grandes distâncias. Cabo ótico de fibra plástica ou de fibra de vidro, pode ser utilizado para transmissões em alta velocidade, livre de interferência elétrica e com isolamento elétrico. Os baudrates na fibra ótica são os mesmos do RS485. As distancias máximas alcançadas neste caso são 3Km para fibra multi-modo e 50Km para redes mono-modo (VIVACE,2009).

Para a comunicação Profibus PA é utilizado o mecanismo MBP (*Manchester Bus Powered*) que é baseado no padrão IEC61158-2. O MBP é uma transmissão em par trançado blindado que suporta até 32 estações por segmento. Alimentação e dados no mesmo cabo.

A transmissão IEC 61158-2 é uma transmissão síncrona em conformidade com a norma utiliza uma taxa de transmissão definida em 31,25 Kbits/s. Além de proporcionar segurança intrínseca permite que os dispositivos de campo sejam energizados pelo próprio barramento. As opções e limites do PROFIBUS com tecnologia de transmissão IEC 61158-2 para uso em áreas potencialmente explosivas são definidas pelo modelo FISCO (Fieldbus Intrinsically Safe Concept) que foi desenvolvida pelo instituto alemão PTB- Physikalisch Technische Bundesanstalt (Instituto Tecnológico de Física).

Em termos de modulação é suposto que uma corrente básica de pelo menos 10 mA consumida por cada dispositivo no barramento. Os sinais de comunicação então são gerados pelo dispositivo que os envia, por modulação de +/- 9 mA, sobre a corrente básica.

A conexão do segmento Profibus DP e Profibus PA é geralmente utilizado um acoplador (Coupler) DP/PA. O acoplador pode ser somente um conversor de meio físico RS485 para MBP

que faz com que a taxa de comunicação da rede profibus DP diminua devido ao baudrate PA ser fixo em 31.25Kbps. Outra opção é a utilização de um Linker que permite ser acoplado ao Coupler e trabalhar como um mestre da rede Profibus PA e um escravo na Rede Profibus DP, aumentando a velocidade da rede DP (CASSIOLATO; TORRES; CAMARGO, 2012).

O caractere Profibus é formado por um start bit, que é sempre “0”, 8 bits de informação, que pode ser “0” ou “1”, um bit de paridade, que pode ser “0” ou “1” e um stop bit, que é sempre “1”, representada na figura 12 (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012).

Figura 8- Caractere Profibus

Start Bit	D0	D1	D2	D3	D4	D5	D6	D7	Paridade	Stop Bit
"0"	0	1	2	3	4	5	6	7	even	"1"
←	LSB	←	←	←	←	←	←	MSB	←	←

Fonte: Adaptado de (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012)

2.2.2 Topologia

Quanto a topologia de rede, existem várias opções suportadas onde as mais indicadas são as topologias do tipo daisy chain para o Profibus DP e barramento com derivações para o Profibus PA. Na figura abaixo, mostra um exemplo de uma rede Profibus DP /PA.

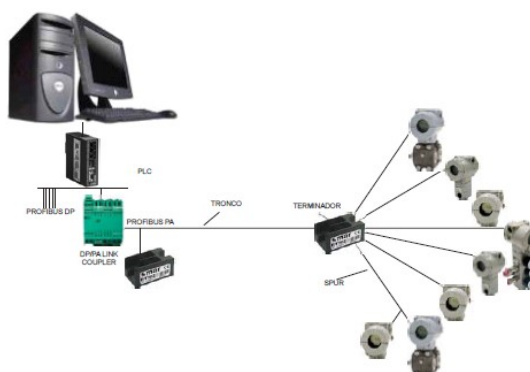
Figura 9- Rede Profibus DP e PA



Fonte: Adaptado de (SMAR,2009, p.1.8)

Em uma rede Profibus PA deve ser colocado dois terminadores de barramento, um na saída do *coupler* DP/PA e outro no último equipamento da rede, dependendo da topologia adotada conforme ilustrado nas figura acima na topologia de barramento e na topologia ponto-a-ponto (SMAR,2009).

Figura 10 - Rede Profibus DP e PA, topologia estrela



Fonte: Adaptado de (SMAR,2009, p.1.8)

Se a topologia adotada for a topologia estrela ou seja se na distribuição dos cabos houver uma caixa de junção no final do tronco principal com vários braços, o terminado de campo deve ser colocado neste ponto (SMAR,2009, p.1.7).

Os terminadores de rede têm basicamente duas funções:

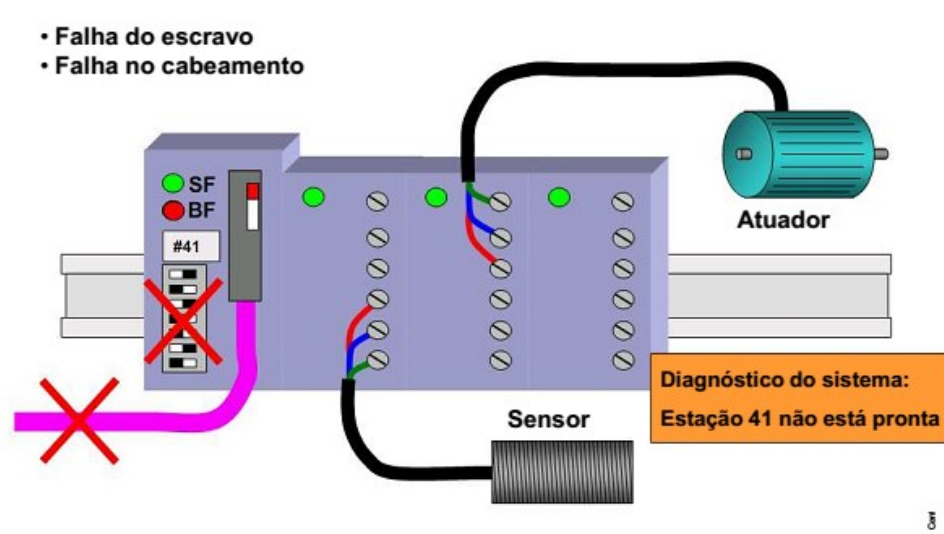
- Proteger a rede contra reflexões de sinal;
- Desviar a corrente referente ao sinal de comunicação.

A falta de terminadores causa a intermitência da comunicação, já que não há casamento de impedância (SMAR,2009, p.1.7).

2.2.3 Diagnósticos

Todo dispositivo Profibus fornece um bloco de diag padrão, que fornece informações básicas sobre o equipamento que são úteis principalmente para detectar falhas na comunicação: o estado das comunicações, se a parametrização e a configuração realizada pelo mestre está correta. Este diagnostico básico é realizado na partida do equipamento, antes de entrar em troca de dados cíclicos.

Figura 11- Exemplo de diagnóstico no escravo



Fonte: Adaptado de (BRANDÃO & SESTILO, 2016)

Além do diagnóstico padrão o equipamento pode enviar um diagnóstico estendido. Diagnósticos estendidos são enviados juntamente com o diag padrão, na mesma msg Diag estendido podem fornecer informações sobre falhas de periféricos. Como a falha em um determinado modulo ou canal.

Para isso, quando o equipamento está em troca de dados (data Exchange), e um escravo tem dado de diagnóstico ele responderá com o código de função FC igual a 0x0A.

Na próxima mensagem o mestre requisita o pedido de diagnósticos dos escravos e ele responde conforme a figura 12. Os escravos respondem com pelo menos 6 bytes mandatórios de diagnósticos e pode conter até 244 bytes.

Figura 12- Exemplo de telegrama de diagnóstico no Profibus

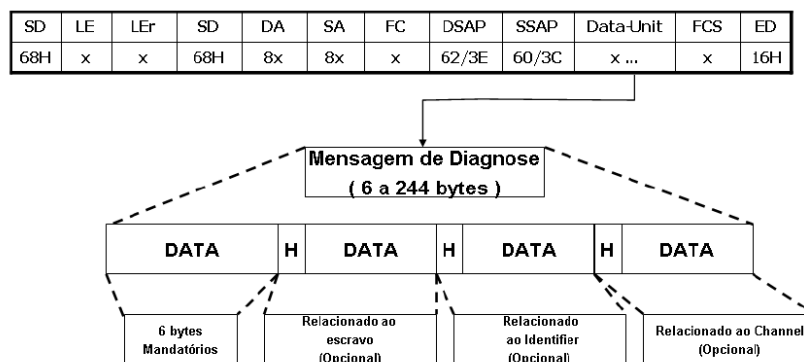
2210	29-Oct-2019 15:57:01.435	SD4	1->1	Token pass	Pass token						
2211	29-Oct-2019 15:57:01.441	SD2	1->4	SRD HIGH	Data Exchange	Req	5	42 C8 00 00 80			
2212	29-Oct-2019 15:57:01.450	ACK			Short acknowledge	Res					
2213	29-Oct-2019 15:57:01.455 Sync	SD2	1->102	SRD HIGH	Get Diagnostics	Req	62->60	0			
2214	29-Oct-2019 15:57:01.472	SD1	1->2	SRD HIGH	Data Exchange	Req					
2215	29-Oct-2019 15:57:01.474	SD2	1<-2	DL	Data Exchange	Res	10	41 40 00 00 80 00 00 00 4F			
2216	29-Oct-2019 15:57:01.483	SD1	1->83	FDL Status		Req					
2217	29-Oct-2019 15:57:01.499	SD4	1->1	Token pass	Pass token						
2218	29-Oct-2019 15:57:01.505	SD2	1->4	SRD HIGH	Data Exchange	Req	5	42 C8 00 00 80			
2219	29-Oct-2019 15:57:01.514	ACK			Short acknowledge	Res					
2220	29-Oct-2019 15:57:01.519 Sync	SD2	1->102	SRD HIGH	Get Diagnostics	Req	62->60	0			
2221	29-Oct-2019 15:57:01.536	SD1	1->2	SRD HIGH	Data Exchange	Req					
2222	29-Oct-2019 15:57:01.538	SD2	1<-2	DL	Data Exchange	Res	10	41 40 00 00 80 00 00 00 4F			
2223	29-Oct-2019 15:57:01.547	SD1	1->84	FDL Status		Req					
2224	29-Oct-2019 15:57:01.563	SD4	1->1	Token pass	Pass token						
2225	29-Oct-2019 15:57:01.569	SD2	1->4	SRD HIGH	Data Exchange	Req	5	42 C8 00 00 80			
2226	29-Oct-2019 15:57:01.578	ACK			Short acknowledge	Res					
2227	29-Oct-2019 15:57:01.583 Sync	SD2	1->102	SRD HIGH	Get Diagnostics	Req	62->60	0			
2228	29-Oct-2019 15:57:01.600	SD1	1->2	SRD HIGH	Data Exchange	Req					
2229	29-Oct-2019 15:57:01.602	SD2	1<-2	DL	Data Exchange	Res	10	41 40 00 00 80 00 00 00 4F			
2230	29-Oct-2019 15:57:01.611	SD1	1->85	FDL Status		Req					
2231	29-Oct-2019 15:57:01.627	SD4	1->1	Token pass	Pass token						

Fonte: Autor, 2019

As mensagens de diagnósticos são divididas em três níveis:

- Mensagens de diagnósticos de estação que estão relacionadas com o estado operacional geral da estação.
 - Mensagens de diagnósticos de módulo representam uma falha em um I/O específico de uma estação.
 - Mensagens de diagnósticos de canal são mensagens que indicam um erro de bit no I/O.
- A figura 13 abaixo mostra os significados dos bytes de diagnósticos.

Figura 13- Mensagens de Diagnósticos

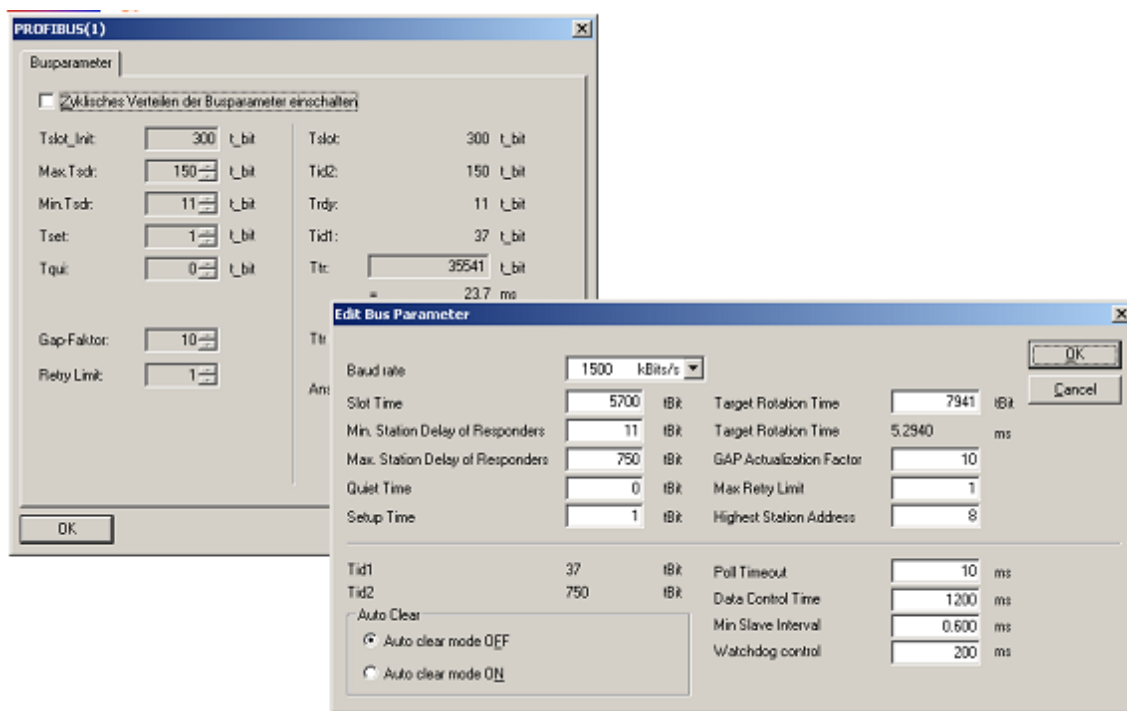


Fonte: Adaptado de (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012, p.56)

2.2.4 Temporização Profibus

Temporização na rede profibus determinam os corretos tempos da rede. Estes tempos são atualizados automaticamente pelos mestres, porem alguns ajustes podem ser necessários. A figura abaixo mostra a configuração de parâmetros do Profibus DP/PA.

Figura 14- Configuração Bus Parameter Profibus DP/PA



Fonte: Autor,2019

Os principais tempos são detalhados abaixo:

O tempo de ciclo da rede é o tempo que o token leva para circular todos os mestres. Ele pode ser considerado como o período de varredura (data exchange) de cada escravo

O Target rotation time, TTR, representa o tempo máximo de ciclo, não o real ou o atual!

O Tempo de Watchdog é configurado para cada escravo na parametrização. Ele Determina o máximo tempo entre duas requisições a este escravo antes de ele ir ao modo de falha segura. Normalmente é configurado como um múltiplo do target rotation, tipicamente entre 5 e 10 vezes o TTR.

O Slot Time indica o tempo máximo aguardado pelo mestre por um reconhecimento ou resposta, após transmissão da mensagem. Se este tempo se expirar, o mestre deve repetir o pedido, até o número máximo de retransmissões.

O parâmetro Retry limit é o máximo número de tentativas de request de um mestre antes de detectar falha. A figura abaixo mostra um exemplo de utilização do SlotTime e Retry limit.

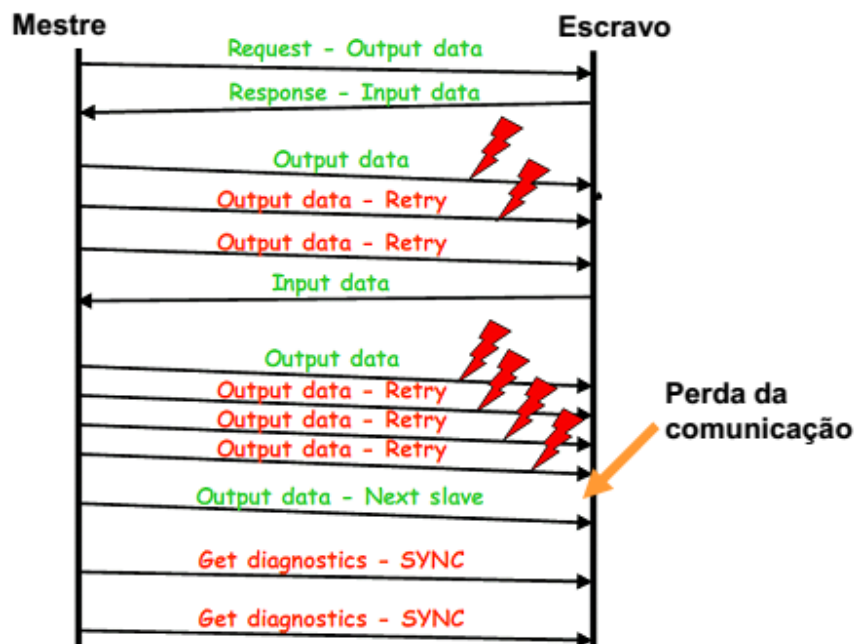
Cada vez que um escravo detecta um erro de transmissão ao receber um pedido o mestre, ele não responde e depois de espera por um tempo (slot time), o mestre reenviará novamente o pedido (retry). Do mesmo jeito se o mestre detectar uma falha na resposta do escravo, também enviará novamente o pedido. A quantidade de vezes em que o mestre tentará obter sucesso na

comunicação com o escravo dependerá da taxa de comunicação sendo (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012):

- 9.6kbits/s a 1.5Mbits/s – retry = 1
- 3.0 Mbits/s – retry = 2
- 6.0 Mbits/s – retry = 3
- 12.0 Mbits/s – retry = 4

Quando todos retries estiveram acabados, o mestre marca o escravo, indicando um problema e faz o log out dele. Nos próximos ciclos se o mestre consegue sucesso, ele realiza a sequência do startup novamente (4 ciclos para trocar dados novamente).

Figura 15- Exemplo de perda de comunicação com um escravo



Fonte: Adaptado de (BRANDÃO & SESTILO, 2016)

O parâmetro G indica o número de rotações de token entre cada envio de "fdl_request".

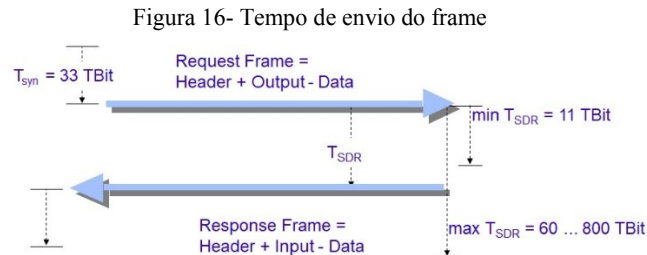
- G=1 indica que um fdl_request será enviado a cada token,
- Altos valores para G implicam em demora para a detecção de novos mestres na rede.

O parâmetro HSA indica o endereço da maior estação para FDL_Req. Como boas práticas de configuração têm-se: HSA baixo permite a detecção rápida de mestres. HSA alto é mais adequado para comissionamento.

O cálculo de tempo de uma mensagem Profibus pode ser determinado por:

Um byte é transmitido como 11 Bits (8 bits de dados + 1 start + 1 stop bit + 1 bit de paridade PAR). O telegram header e trailer (Overhead) consiste de 11 Bytes ou 9 Bytes. Desta forma para um baudrate de 1.5Mbps:

$$1 \text{ TBit @ } 1.5 \text{ MBaud} = 0.6667 \text{ ms (1 byte = 1.33 ms)}$$



Fonte: Adaptado de (BRANDÃO & SESTILO, 2016)

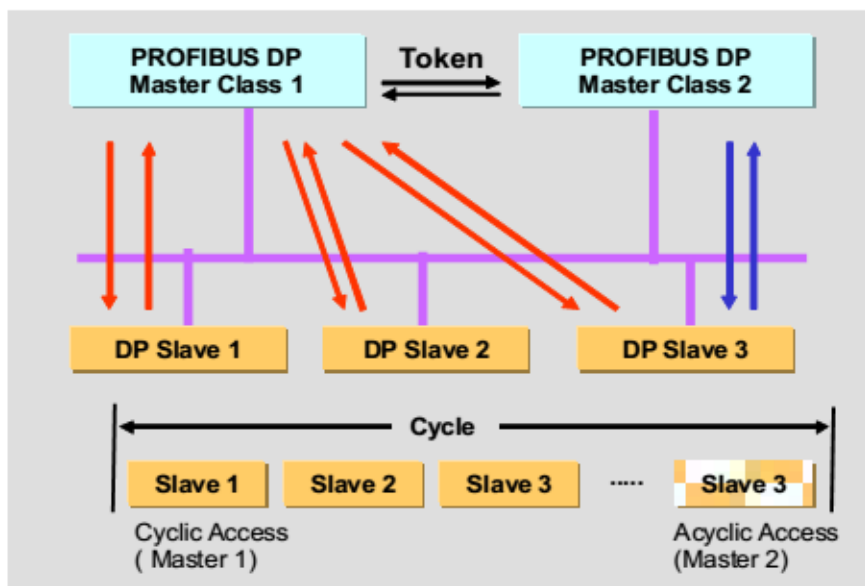
2.3 Características Profibus DP

O PROFIBUS-DP permite sistemas mono e multi- mestre oferecendo um alto grau de flexibilidade na configuração do sistema. A rede PROFIBUS suporta até 126 dispositivos (mestres e escravos). A configuração baseia na definição do número de estações, dos endereços das estações e de seus I/O's, do formato dos dados de I/O.

Cada sistema Profibus pode possuir três tipos de dispositivos diferentes:

Mestres Classe 1-DP que são dispositivos tais como os controladores programáveis (PLCs), que é responsável por fazer a lógica de controle da planta, cada rede possui um mestre classe 1. O Mestre Classe 2 – DP são representados pelas ferramentas de manutenção, dispositivos de configuração, terminais de engenharia entre outros, são utilizados para configuração do sistema DP e manutenção dos equipamentos. Escravo DP são os equipamentos de campo (dispositivos de I/O, drivers, IHM, válvulas, etc.) são responsáveis por coletar informações de entrada e enviarem informações de saída ao controlador (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012).

Figura 17- Tipos de Dispositivos rede Profibus



Fonte: Adaptado de (BRANDÃO & SESTILO, 2016)

Um sistema mono-mestre possui somente um mestre ativo na rede durante toda fase de operação. Por outro lado a configuração multi-mestre possui vários mestres na mesma rede.

A comunicação no Profibus é do tipo mestre-escravo ou polling ou seja, de posse do token, cada mestre pergunta para o seu escravo e aguarda pela resposta. Na maioria da comunicação de dados com os dispositivos distribuídos é feita de forma cíclica. Além da execução destas funções cíclicas, funções de comunicação acíclicas também estão disponíveis para dispositivos de campo inteligentes, tornado assim possível a configuração de diagnósticos e manipulação de alarmes (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012).

As camadas superiores do protocolo Profibus acessa o meio através da camada 2 chamada de Fieldbus Data Link (FDL) do modelo de referência da OSI. O Controle de Acesso ao meio (MAC) determina o procedimento quando uma estação tem a permissão para transmitir dados. O MAC deve garantir que uma única estação transmita dado por vez em um determinado momento. O protocolo PROFIBUS foi construído para anteder os dois requisitos básicos do Controle de Aceso ao Meio (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012):

- Durante a comunicação dos mestres, deve garantir que cada uma destas estações detém tempo suficiente para executar suas tarefas e comunicação dentro de um tempo determinado e preciso de tempo.
- A transmissão cíclica de dados em tempo real deverá ser implementada tão rápida e simples quanto possível entre a comunicação entre Mestre e escravos I/O's.

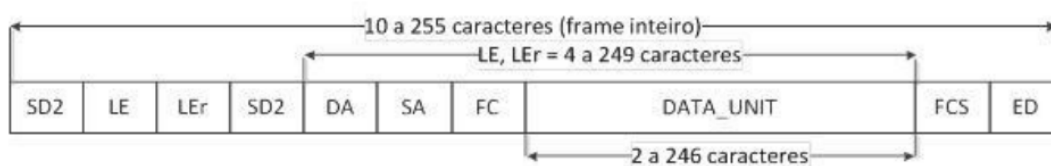
Para realizar este acesso ao meio, o protocolo PROFIBUS inclui o procedimento de passagem de Token, que é utilizado pelos Mestres para fazer a comunicação com os outros dispositivos, e o procedimento de mestre-escravo que é usado pelos Mestres para comunicarem com os escravos (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012).

A passagem de Token dá o direito de acesso ao barramento a cada mestre que possui o Token dentro de um intervalo de tempo. A mensagem de Token é um telegrama especial para passar direitos de acesso de um mestre ao próximo mestre da rede de comunicação, deve ser distribuída no anel lógico de Token pelo menos uma vez para cada mestre dentro de um intervalo de tempo máximo chamado de tempo de rotação de Token (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012).

O procedimento mestre-escravo permite que na hora que o mestre detém o Token possa acessar seus escravos. O mestre pode tanto enviar mensagens como ler mensagens de seus escravos.

Com relação ao formato da mensagem, existem 4 tipos de frames no Profibus. A diferenciação entre os tipos é realizada pelo cabeçalho e depende da natureza do campo de dados: Mensagens de tamanho fixo sem campo de dados, mensagem de token, mensagem de resposta curta (Short ack) e mensagens de campo de dados variável (SD2). Um exemplo desta última mensagem é mostrado na figura abaixo (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012).

Figura 18- Mensagem DPV1 Profibus tipo SD2



Fonte: Adaptado de (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012, p.52)

De acordo com a Figura 18, SD2 é o *Start Delimiter* (68H para o frame SD2), LE indica o *information length*, então se repete o LE e o SD2 novamente, DA é o *Destination Address*, SA significa o *source address*, FC é o *frame control*, DSAP é o *Destination Service Access Point*, SSAP é o *Source Service Access Point*, PDU é o campo de dados que pode variar até 246 bytes, FCS é o *Frame Check Sequence* e o ED é o *end Delimiter* (16H) (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012).

Cada perfil de comunicação PROFIBUS utiliza subconjuntos específicos dos serviços da camada 2. Os serviços são acionados por camadas mais elevadas via pontos de acesso de

serviço (SAP's). No protocolo PROFIBUS-DP a cada função definida é associado um ponto e acesso de serviço. É feita uma distinção entre fonte (SSAP-*Source SAP*) e destino dos pontos de acesso de serviço (DSAP- *Destination SAP*). Para troca de mensagens cíclicas (data Exchange) não existe SAP na mensagem. A tabela abaixo mostra os principais SAPs do protocolo DPV1.

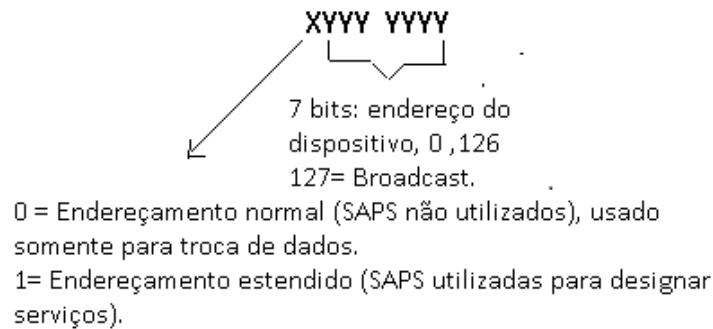
Tabela 1- Serviços SAPs existentes no Profibus

Serviço	SAP mestre Hex (Dec)	SAP escravo Hex (Dec)
Troca de Dados	SAP não usado	SAP não usado
Ajuste de endereço de escravo	0x3E (62)	0x 37 (55)
Leitura de entradas	0x 3E (62)	0x 38 (56)
Leitura de saídas	0x 3E (62)	0x 39 (57)
Controle Global	0x 3E (62)	0x 3A (58)
Obter configuração	0x 3E (62)	0x 3B (59)
Diagnóstico de escravo	0x 3E (62)	0x 3C (60)
Parametrização	0x 3E (62)	0x 3D (61)
Verificação e configuração	0x 3E (62)	0x 3E (62)
DPV0 Master to Master functions	0x 36 (54)	0x 36 (54)
DPV1 Funções de Mestre Classe 1	0x 32 (51)	0x 33 (51) ou 0x32 (50)
DPV1 Gerente de Recursos	0x 32 (50)	0x 31 (49)
DPV1 Mestre Classe 2	0x 32 (50)	0 (0) ... 30(48)

Fonte: Adaptado de (BRANDÃO, D; SESTILO, G, S. 2016)

A inserção de um DSAP ou SSAP em um pedido ou resposta é feita setando o bit mais significativo no campo DA ou SA, portanto somente os 7 bits menos significativos que conterão realmente o endereço. O bit mais significativo nos bytes de endereço de origem e destino é usado para indicar “Endereço Estendido” conforme ilustrado na figura 19 abaixo.

Figura 19- Exemplo e endereçamento Profibus



Fonte: Adaptado de (BRANDÃO, SESTILO. 2016)

A camada 2 do PROFIBUS de tem uma importante tarefa de segurança de dados, ou seja, ela formata frames que asseguram a alta integridade de dados. Todos os telegramas têm Hamming Distance HD=4, alcançada através do uso de telegramas especiais delimitadores de início/fim, bit de paridade e byte de check.

Essa camada opera num modo denominado “sem conexão”. Além de transmissão de dados ponto-a-ponto, proporciona também comunicação multi-ponto (Broadcast e Multicast).

- Comunicação Broadcast é quando um mestre envia uma mensagem sem confirmação para todos os escravos.
- Comunicação Multicast é quando um mestre envia mensagem sem confirmação a um grupo pré determinado de escravos.

Figura 20- Serviços da camada de segurança e dados (Data Link Layer)

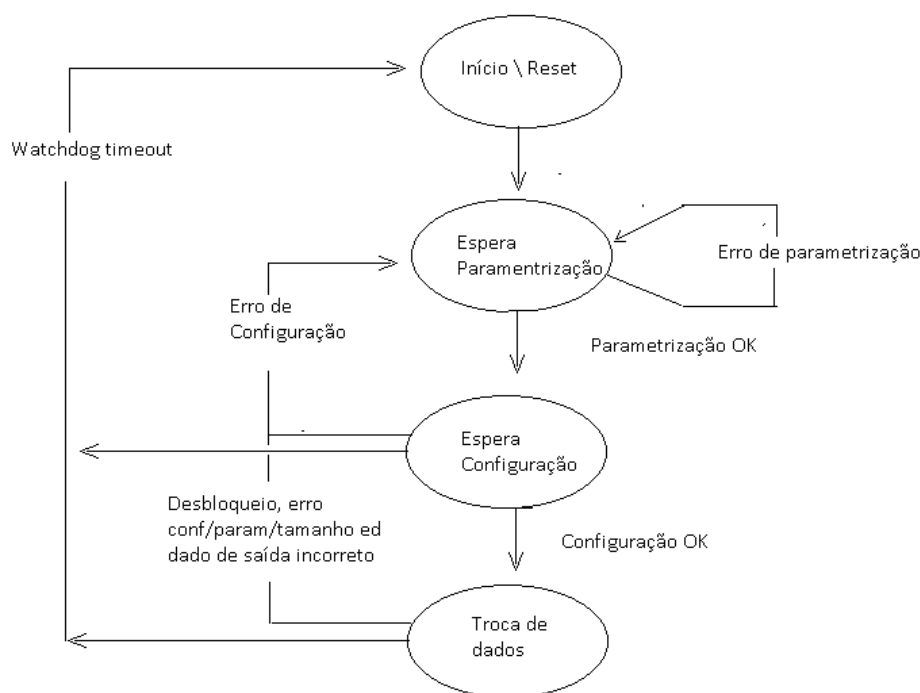
Serviço	Função	DP	FMS	PA
SDA	Send Data with Acknowledge Envia dados com reconhecimento		•	
SRD	Send and Request Data with replay Envia e requisita dados com resposta	•	•	•
SDN	Send Data with No acknowledge Envia dados sem reconhecimento	•	•	•
CSRD	Cyclic Send and Request Data with replay Envia e requisita dados ciclicamente com resposta		•	

Fonte: Adaptado de (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012, p.37)

Na inicialização da rede, o mestre de controle executa várias verificações antes de iniciar a troca de dados cíclica (Data -Exchange) com o escravo. O algoritmo de inicialização consiste basicamente de 5 passos (BRANDÃO; SESTILO. 2016):

1. **Diagnóstico:** O mestre primeiro verifica se o escravo está presente na rede e se este não é controlado por outro mestre, através de um pedido de diagnóstico.
2. **Ajustar Parâmetros:** Após isso, o mestre verifica se o dispositivo é do tipo correto, verificando seu número de ID e definindo alguns parâmetros do dispositivo.
3. **Checar Configuração:** O mestre verifica se os canais de I/O atribuídos no projeto estão presentes e disponíveis no escravo.
4. **Diagnóstico:** Um pedido de diagnóstico verifica se está tudo ok.
5. **Data Exchange:** O escravo só entra em troca de dados se todas as checagens estiverem corretas.

Figura 21- Ilustração de inicialização da rede Profibus

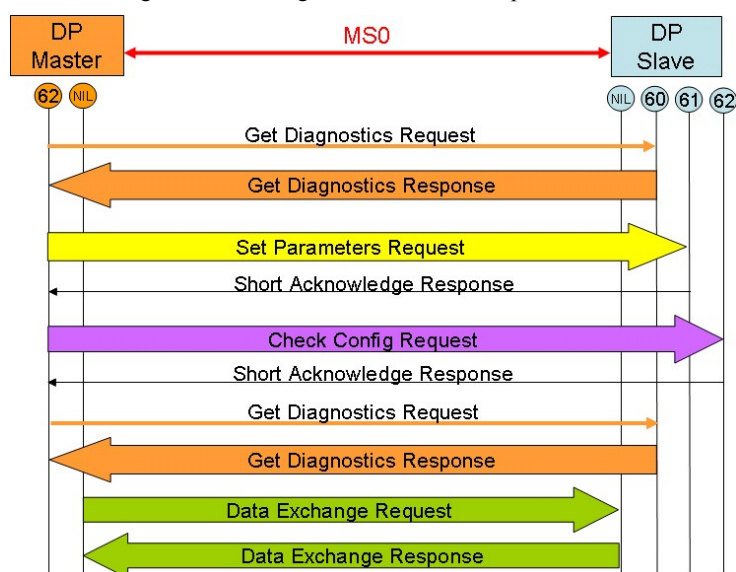


Fonte: Adaptado de (BRANDÃO; SESTILO. 2016)

A figura 21 mostra uma diagrama de inicialização da rede Profibus, nela é mostrado todos passos e condições que um mestre faz para até a troca de dados com seus escravos.

O mestre Profibus envia também um broadcast “Global_Control” em intervalos regulares para indicar que está em modo de operação. Este controle é utilizado pelo mestre para informar aos escravos o estado do mestre. O protocolo não permite que os mestres troquem dados entre si.

Figura 22- Mensagem DPV1 Profibus tipo SD2



Fonte: Adaptado de (BRANDÃO & SESTILO, 2016)

2.4 Características Profibus PA

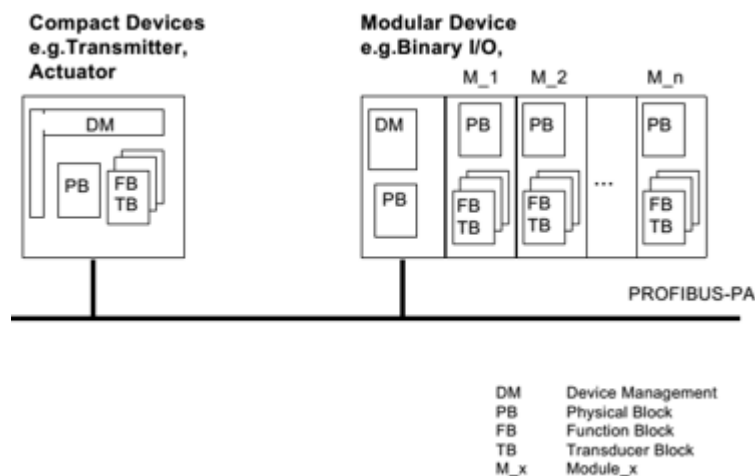
O Profibus PA foi desenvolvido em cooperação com os usuários da Indústria de Controle e Processo (NAMUR), satisfazendo as exigências especiais dessa área de aplicação apresentadas a seguir:

Desenvolvido para aplicação para automação de processo e interoperabilidade dos equipamentos de campo dos diferentes fabricantes; Comunicação transparente através dos acopladores de segmento entre o barramento de automação do processo Profibus PA e o barramento Profibus DP; Alimentação e transmissão de dados sobre o mesmo par de fios baseado na tecnologia IEC 61158-2 e possibilidade de uso em áreas intrinsecamente seguras (VIVACE, 2016).

O Profibus PA suporta os mesmos serviços cíclicos e acíclicos descritos para o Profibus DP. Os serviços cíclicos são utilizados para transmissão de dados pertinentes às medições e comandos de atuação com informações de valor/status. Os serviços acíclicos são utilizados pelas ferramentas de configuração, manutenção e diagnósticos, é utilizada para transmitir comandos de leitura e escrita, tais como alarmes entre mestre e escravos, independente da comunicação cíclica de dados (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012).

O modelo de dispositivo Profibus PA é estruturado em blocos. Cada dispositivo tem um conjunto de blocos padrões de acordo com o Perfil do equipamento. O conceito de Perfil de aplicação PA e os blocos padrões visa facilitar a interoperabilidade entre os equipamentos de campo. Os blocos são acessados através de um gerenciador de dispositivo padrão (Device Manager). A figura abaixo mostra um exemplo de dispositivo compacto e um device modular profibus PA.

Figura 23- Mensagem DPV1 Profibus tipo SD2



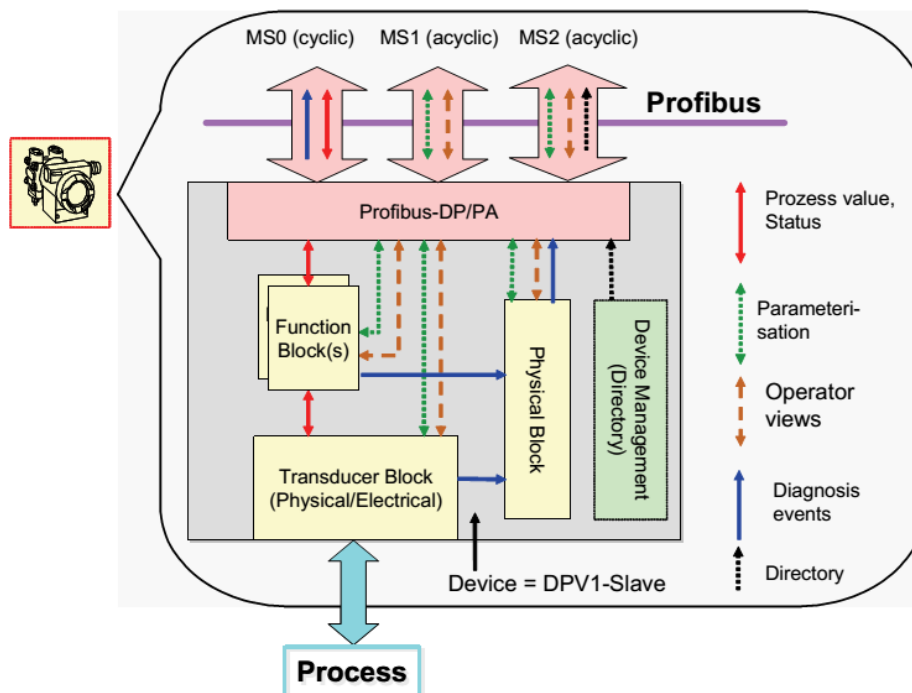
Fonte: Adaptado de (Profibus,1999)

Na troca de dados cíclicos a comunicação é feita pelo mestre classe 1 através do serviço de data-exchange do DPV0 descrito anteriormente para o Profibus DP. No profibus PA o data Exchange deve ser feito através de parâmetros de entrada e saída de blocos funcionais (FB) normalmente utilizada para a variável primária do instrumento ou variável manipulada do atuador (BRANDÃO, D; SESTILO, G, S, 2016).

A troca de dados acíclicos é feita com o mestre classe 1 ou com o mestre classe 2 que são os programas de configuração, manutenção e supervisão como por exemplo o PACTware.

Na troca de dados acíclicos a comunicação é realizada por meio de funções DPV1. Os dados são acessados pelo seu slot e índice através do “*Device Manager*” (BRANDÃO, D; SESTILO, G, S, 2016). A figura 24 mostra um diagrama desta comunicação com um *device*.

Figura 24- Tipos de Comunicação Profibus PA: MS0, MS1 e MS2



Fonte: Adaptado de (BRANDÃO & SESTILO, 2016)

A figura 24 mostra os tipos de comunicação com o equipamento PA: MS0 – Troca cíclica com um mestre classe 1 no modo normal; MS1- Troca acíclica com o mestre classe 1; MS2- Troca acíclica com um mestre classe 2.

O *Device Manager* diz quais blocos existem no dispositivo, onde os blocos estão alocados (slot/índice) e quantos objetos existem em cada bloco. O *Device Manager* sempre é alocado no slot1 / índice 0 (BRANDÃO, D; SESTILO, G, S, 2016).

2.4.1 Blocos PA

Os equipamentos na rede Profibus PA são implementados segundo o modelo de blocos, onde um agrupamento de parâmetros garante acesso uniforme e sistemático das informações. Os blocos representam diferentes funções de usuários, como entradas analógicas e saídas analógicas (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012).

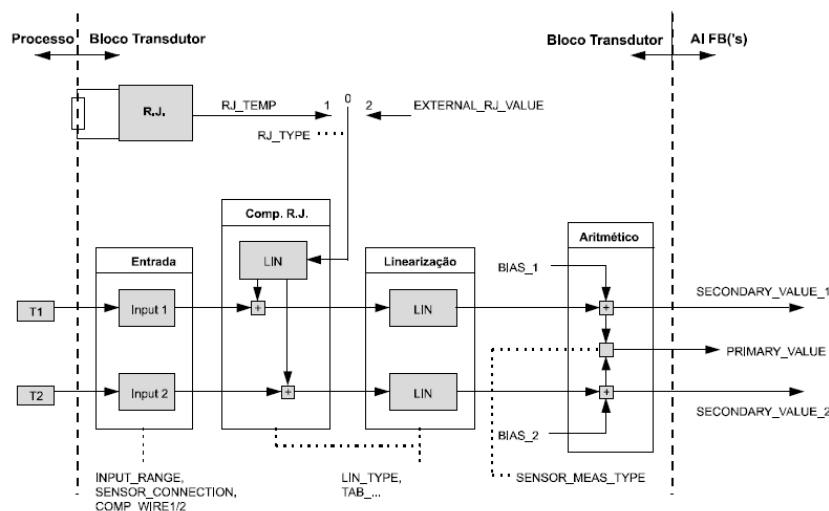
De acordo com a figura 28, cada device deve ter no mínimo um bloco físico (PB), um transducer block (TB) e um function block (FB) além do gerenciador de dispositivo (DM). Cada um dos blocos possuem um conjunto de parâmetros padrões que representam a forma do usuário acessar as funções dentro do bloco e do dispositivo. Alguns parâmetros são obrigatórios, outros são opcionais. Os blocos são:

Bloco Físico (Physical Block): Contém informações gerais do dispositivo, tais como: nome, fabricante, versão e número de série do dispositivo. (CASSIOLATO; PADOVAN; TORRES; OLIVEIRA, 2012).

Bloco Transdutor (TransducerBlock): O bloco transdutor contém as características específicas do sensor ou atuador. Ele é a ligação para o elemento físico do device, responsável pela transdução do sinal do sensor e na conversão no sinal digital ou vice-versa. Além disso fornece opções para o usuário de escolha entre diferentes sensores, diagnósticos avançados, e funções de calibração do instrumento, entre outras (SMAR,2012).

Por exemplo, para o transmissor TT303 de temperatura (SMAR), o bloco transdutor possui duas entradas de temperatura T1 e T2. O usuário pode configurar o tipo de sensor de temperatura: Pt100, Termopar, ohms, etc. Pode fazer uma compensação de junta fria, e pode escolher diferentes saídas para interligar com o bloco funcional (FB) de entrada analógica (AI). O bloco transdutor tem um total de 82 parâmetros configuráveis pelo operador.

Figura 25- Estrutura interna do bloco Transdutor

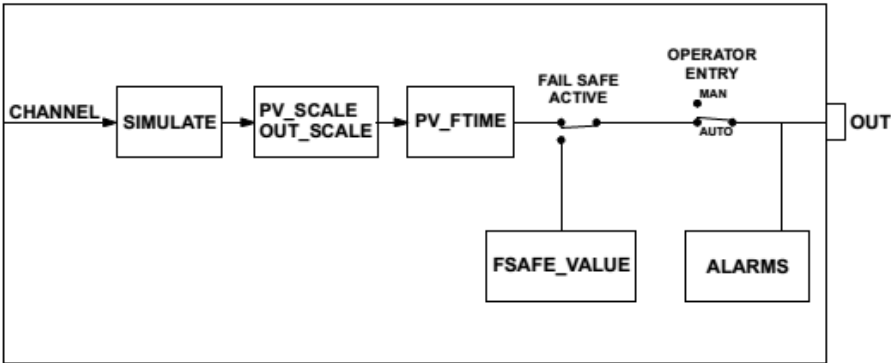


Fonte: Adaptado de (Profibus,1999)

Bloco de Função (Function Block – FB): o bloco de função contém todos os dados para processamento final do valor medido antes da transmissão para o sistema de controle. Os principais blocos funcionais na especificação são: entrada analógica (AI), saída analógica (AO), totalizador (TOT), entrada digital (DI), saída digital (DO). Abaixo será detalhado somente os blocos AI e AO que serão utilizados neste trabalho (SMAR,2012).

Bloco de Entrada Analógica (“Analog Input Block”) – AI: o bloco de entrada analógica utiliza os dados de entrada do bloco transdutor, selecionada por um canal (*channel*), e disponibiliza estes dados em sua saída (*out*) para outros elementos do controle (SMAR,2012).

Figura 26- Exemplo Bloco de Entrada Analógico



Fonte: Adaptado de (SMAR,2012)

De acordo com a figura 26, o bloco AI possui características como simulação, caso não tiver o transdutor funcionando, conversão de escala, filtro de primeira ordem, condições de segurança, limites e alarmes. O modo de operação do bloco pode ser fora de serviço, Manual e Automático. Segundo a especificação Profibus PA, o bloco AI possui 35 parâmetros editáveis pelo usuário.

Em relação a comunicação de troca de dados cíclica (data Exchange), o AI somente possui o parâmetro OUT que pode ser configurado conforme mostrado na figura abaixo:

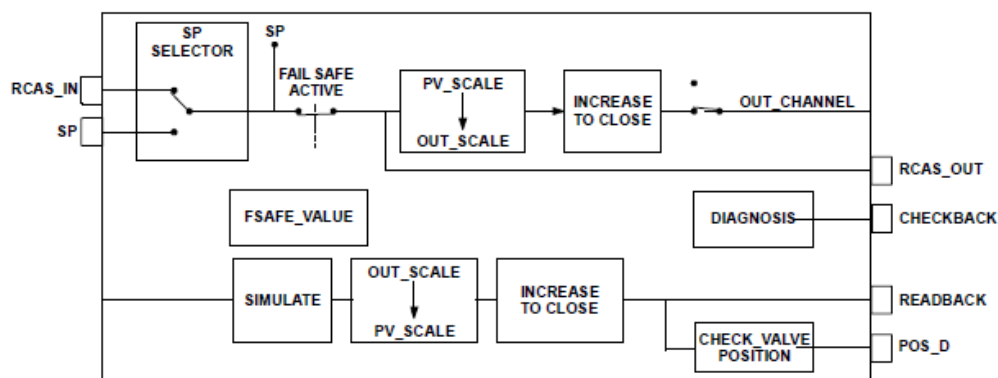
Figura 27- Modo de configuração AI

Configuração suportada	Config Curto (Byte Identificador)	Config Longo (formato Identificador Estendido)
OUT	0x04	0x42,0x84,0x08,0x05

Fonte: Adaptado de (SMAR,2012)

Bloco de Saída Analógica (“Analog Output Block”) – AO: O bloco de saída analógica é utilizado por equipamentos que funcionam como elementos de saída em uma malha fechada, por exemplos, válvulas, atuadores, posicionadores, etc. O bloco AO recebe o sinal de outro bloco funcional no Mestre Classe 1 e transmite o valor para um transdutor de saída através de um canal interno de referência (SMAR,2012).

Figura 28- Exemplo Bloco de Saída Analógica



Fonte: Adaptado de (SMAR,2012)

De acordo com a figura 28, o bloco AO possui características como seletor da entrada SP, conversão de escala, filtro de primeira ordem, inversão do sinal, retorno do processo, simulação caso não tiver o transdutor funcionando, condições de segurança, limites e alarmes. O modo de operação do bloco pode ser fora de serviço (OOS), Manual (MAN), Automático (AUTO), Cascada Remota (RCAS) ou Atuação Local (LO). Segundo a especificação Profibus PA, o bloco AI possui 35 parâmetros editáveis pelo usuário(SMAR,2012).

Em relação a comunicação de troca de dados cíclica (data Exchange), o AO possui dois parâmetros de entrada (RCAS_IN e SP) e quatro saídas (RCAS_OUT, CHECKBACK, READBACK e POS_D) que pode ser configurado conforme mostrado na figura abaixo:

Figura 29- Configuração Suportada pelo AO

Configuração suportada	Config Curto (Byte Identificador)	Config Longo (formato Identificador Estendido)
RCASIN / RCASOUT	0xB4	0xC4,0x84,0x84,0x08,0x05,0x08,0x05
RCASIN / RCASOUT / CHECKBACK	0x97,0xA4	0xC5,0x84,0x87,0x08,0x05,0x08,0x05,0x0A
SP	0xA4	0x82,0x84,0x08,0x05
SP / CHECKBACK	0x92,0xA4	0xC3,0x84,0x82,0x08,0x05,0x0A
SP / READBACK / POSD	0x96,0xA4	0xC6,0x84,0x86,0x08,0x05,0x08,0x05,0x05,0x05
SP / READBACK / POSD / CHECKBACK	0x99,0xA4	0xC7,0x84,0x89,0x08,0x05,0x08,0x05,0x05,0x05,0x0A
SP / READBACK / RCASIN / RCASOUT / POSD / CHECKBACK	0x9E,0xA9	0xCB,0x89,0x8E,0x08,0x05,0x08,0x05,0x08,0x05,0x08,0x05,0x05,0x0A

Fonte: Adaptado de (SMAR,2012)

2.5 Sistema de Supervisão

Na década de 1990, surgiram os novos sistemas de supervisão e controle, desenvolvidos para obter maior produtividade, qualidade e competitividade para esta nova realidade. Com as continuas evoluções que aconteceram ao longo do tempo, os sistemas de supervisão atuais passaram a oferecer três funções básicas: supervisão, operação e controle (PINHEIRO,2006).

Função de Supervisão está relacionado com o monitoramento das variáveis de processo em tempo real. Função de Operação possibilita substituir o controle manual, otimizando os procedimento de controle e modos de operação dos equipamentos do processo. Por fim a Função de Controle possibilita a ação de controle sem a dependência de processo e sistema supervisório, utilizando dinâmico, de acordo com o comportamento global do processo (PINHEIRO,2006).

As tecnologias utilizadas para integração dos sistemas são baseadas em padrões abertos como por exemplo para fazer a comunicação do sistema supervisório com o CLP está sendo utilizado o padrão OPC (*OLE for ProcessControl*) e para um sistema de manutenção é utilizado o padrão aberto FDT/DTM (*Field Device Tool / Device Type Manager*). Esses padrões serão detalhados a seguir.

2.5.1 Sistema Supervisório

Hoje os sistemas SCADAS (*SupervisoryControland Data Aquisition*) tem um importante papel na automação industrial utilizam tecnologias de computação e comunicação para automatizar a monitoração e controle de processos industriais. O operador terá em suas mãos uma tela gráfica representado todo o processo ou parte dele seus parâmetros e os valores de campo reais, ao processar as variáveis o operador pode por exemplo montar gráficos e monitorar valores das variáveis de processo (DA SILVA; SALVADOR,2004).

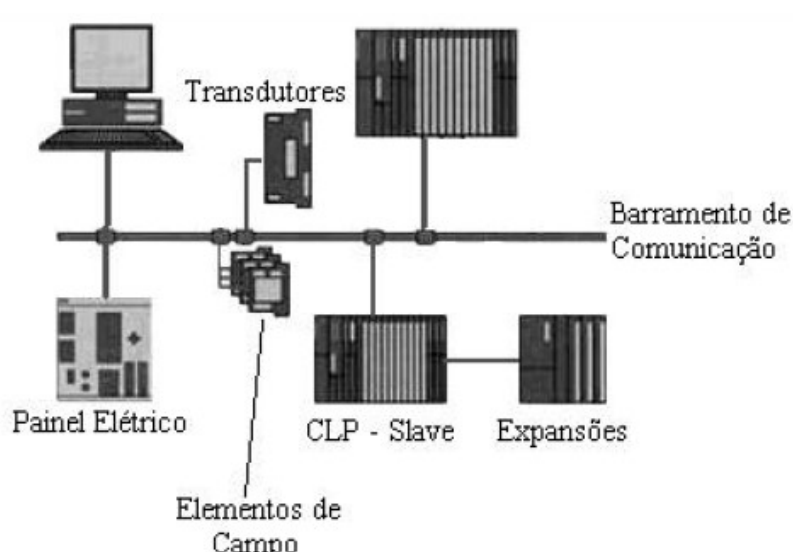
De acordo com Da Silva e Salvador (2004):

O sistemas SCADA identificam as tags, que são todas as variáveis numéricas ou alfanuméricas envolvidas na aplicação podem executar funções computacionais ou representar pontos de entrada/saída de dados de processo.

Os sistemas supervisórios apresenta as seguintes funcionalidades: aquisição de dados desde os equipamentos de campo até as estações centrais de monitoração; visualização de dados através de interface homem máquina e telas animadas; processamento de alarmes para informar anormalidades; tolerâncias as falhas (PINHEIRO,2006).

A figura 30 mostra um esquema básico de um sistema supervisório.

Figura 30 - Esquema básico sistema supervisório

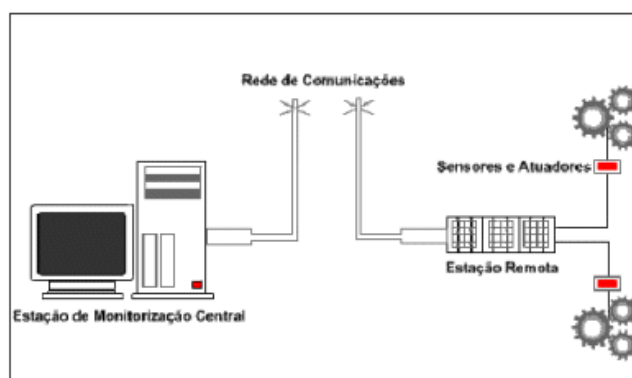


Fonte – Adaptado de (JURIZATO)

A parte física de um sistema de supervisão é composta por sensores e atuadores que estão conectados aos equipamentos de campo controlados e monitorados pelos SCADAS; rede de comunicação é por onde as informações são trafegadas para o controle central, estações remotas que são apresentadas pelos CLPs (Controles Lógicos e Programáveis); estações de monitoração central que são responsáveis recolher as informações geradas pelas estações remotas e realizar atividades conforme os eventos detectados (Da Silva e Salvador,2004).

A figura 31 apresenta uma ilustração de uma estrutura de um sistema de supervisão.

Figura 31- Esquema básico de uma estrutura de um sistema de supervisão



Fonte – Adaptado (PINHEIRO,2006).

De uma forma geral o funcionamento de um sistema SCADA inicia nos processos de comunicação com os equipamentos de campo, onde as informações são enviadas para o núcleo principal do software. O núcleo fica responsável por distribuir e coordenar o fluxo das informações para os demais módulos, até que cheguem na forma esperada para o operador do sistema, na interface gráfica (DA SILVA; SALVADOR,2004).

Essas informações podem chegar ao usuário de forma gráfica, animações, relatórios, etc, permitindo informar erros, propor algumas soluções ou reagir automaticamente (DA SILVA; SALVADOR,2004).

2.5.2 Tecnologia OPC

No ano de 1995, empresas se uniram e gerenciadas através do *OPC Foundation* criaram um padrão baseado na tecnologia OLE/DCOM da Microsoft, para que fosse possível acessar os dados por meio do sistema operacional Windows (Carrijo,2011, p.22).

A tecnologia OLE (*ObjectLinkingandEmbedding*) foi criada pela Microsoft em meados de 1990 para integração entre diferentes aplicações dentro da plataforma do Windows (Carrijo,2011, p.22).

O DCOM (Distributed Component ObjectModel) surgiu com o Windows NT é um conjunto de definições que permite a implementação de aplicações distribuídas no modelo de arquitetura cliente-servidor (Carrijo,2011, p.22).

O padrão OPC define especificações para o desenvolvimento de interfaces entre clientes e servidores, bem como servidores e servidores. Esse padrão tem como objetivo abstrair protocolos específicos de CLP (como o Profibus etc.) em uma interface padronizada, possibilitando que sistemas como HMI e sistemas SCADAs possam acessar os dados em tempo real, fazer monitoramento de alarmes e acessar os dados pelo histórico. Quando o padrão OPC foi lançado ele era restrito ao sistema operacional Windows denominado como OPC Classic. (OPC Foundation,2019).

Segundo a *OPC Foundation* as especificações do OPC Classic são:

Acesso a dados OPC (OPC DA) – A especificação OPC DA define a troca de dados, incluindo valores, tempo e informações de qualidade.

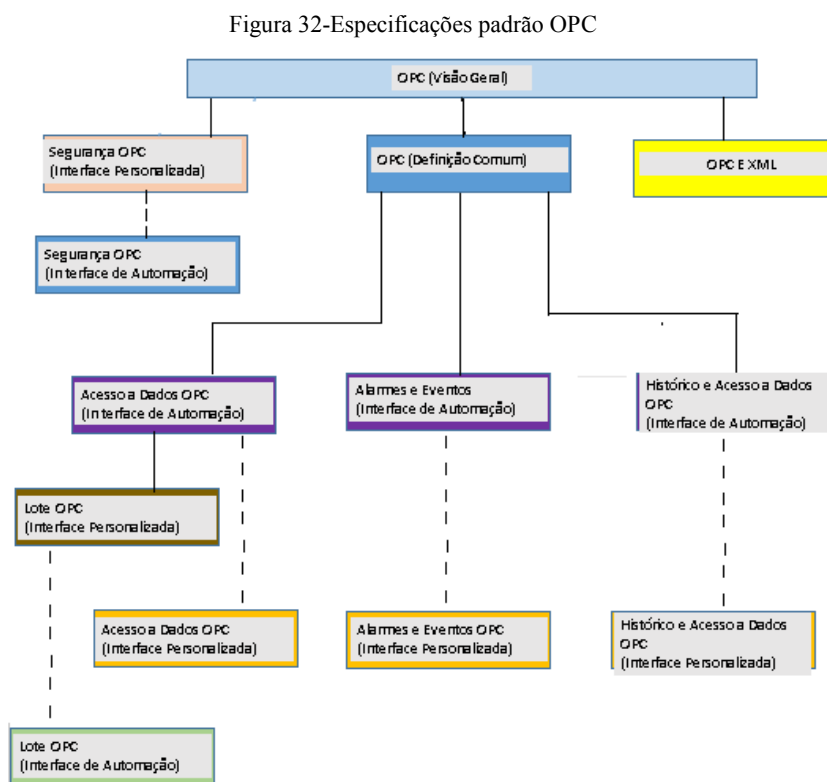
Alarmes e eventos OPC (OPC AE) - A especificação OPC A&E define a troca de informações de mensagens de alarme e tipo de evento, bem como estados variáveis e gerenciamento de estado.

Acesso a dados históricos do OPC (OPC HDA) - A especificação OPC HDA define métodos e análises de consulta que podem ser aplicados a dados históricos com registro de data e hora.

Com o desenvolvimento de arquiteturas orientadas a serviços em sistemas de manufaturas, vieram também novos desafios em segurança e modelagem de dados. A *OPC Foundation* desenvolveu as especificações do OPC UA e ao mesmo tempo que atendia essas

especificações, construiu uma arquitetura orientada a serviços independente de plataforma (OPC Foundation,2019).

A figura 32 apresenta uma visão das especificações do padrão.



Fonte: Adaptado de (OPC Foundation, 2010)

Estas especificações finalidades tem a função de ajudar os desenvolvedores na implementação das aplicações cliente e servidor. Os usuários finais não precisam conhecer a fundo as especificações, sendo suficiente conhecer os aspectos práticos para utilização do padrão.

2.5.3 Tecnologia FDT/DTM

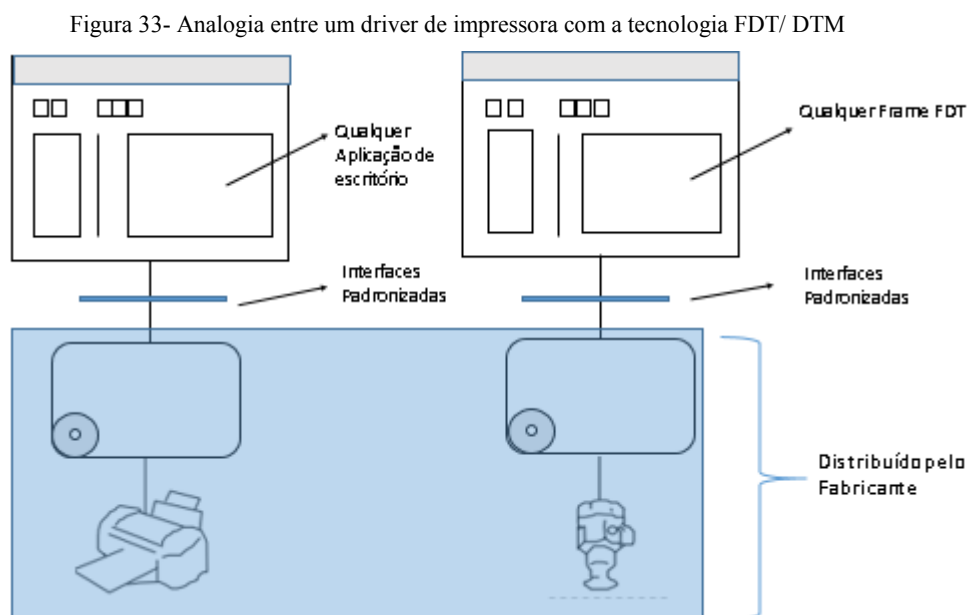
A tecnologia FDT foi desenvolvida pelo Grupo FDT *AISBL* (*Association Industrielle Sans But Lucratif*), esse grupo é uma corporação internacional sem fins lucrativos que consiste em empresas membros globais líderes que atuam na fabricação de automação industrial (FDT GROUP).

O grupo FDT tem como principal objetivo disponibilizar uma interface padronizada aberta e não proprietária para a integração de dispositivos de campo com sistemas de engenharia, automação e gerenciamento de ativo (FDT GROUP).

FDT(Field Device Tol) é uma tecnologia padronizada de integração de ativos e entrega de dados. Essa tecnologia especifica interfaces de software para integração de equipamentos de campo em sistemas de engenharia , automação, permitindo a interoperabilidade entre a IHM (Interface Homem Máquina)(CASSIOLATO, PADAVON, TORRES, OLIVEIRA ,2012).

Para entender mais sobre a tecnologia FDT/DTM é possível fazer uma analogia do DTM com um *driver* de uma impressora para um sistema operacional. Quando um fabricante de impressora vende seu produto ele te fornece juntamente um *driver* para que o usuário possa instalar a impressora em qualquer sistema operacional, e com esse *driver* instalado qualquer software pode imprimir sem que tenha que conhecer o processo. Da mesma forma quando um fabricante de equipamento vende seu produto ele fornece junto com ele um *driver* que nesse caso são os DTMs dos instrumento de campo, com isso qualquer ferramenta FDT pode fazer uso desses DTMs instalados na máquina e acessar todos blocos disponíveis no instrumentos (CARRIJO,2011, p.30).

A figura 33 ilustra a analogia explicada acima entre a tecnologia FDT e um driver de impressora.



Fonte: Adaptado (FDT Group, 2007)

O FDT possibilita que qualquer dispositivo seja de acessado de qualquer host através de qualquer protocolo, ele fornece um ambiente para a operação do DTM.

A tecnologia FDT é baseada em três componentes principais apresentadas a seguir:

- O DTM (Device Type Manager) é usado para a configuração e manutenção de dispositivos de campo, acionamentos, etc.
- O contêiner FDT implementa os FDT *pecification*. É uma interface entre FDT e uma aplicação superior. Ele usa os DTMs para obter acesso a dispositivos.
- FDT aplicativo de quadro é uma ferramenta de configuração que possui o recipiente FDT. A interface do usuário dos DTMs é exibida aqui.

A comunicação DTM (Device Type Manager), fornece uma comunicação padronizada, fazendo assim a interface entre os fornecedores dos dispositivos DTM e o driver específico do sistema host que lida com passagem comunicações do PC até a placa de interface fieldbus (ProSoft Technology, 2008).

O fornecedor do sistema host fornece um DTM de comunicação (comDTM) para cada protocolo fieldbus suportado. Isso garante que os detalhes do PC, rede, placas de interface e protocolos de passagem do sistema host, são transparentes para o DTM do fornecedor do dispositivo (ProSoft Technology, 2008).

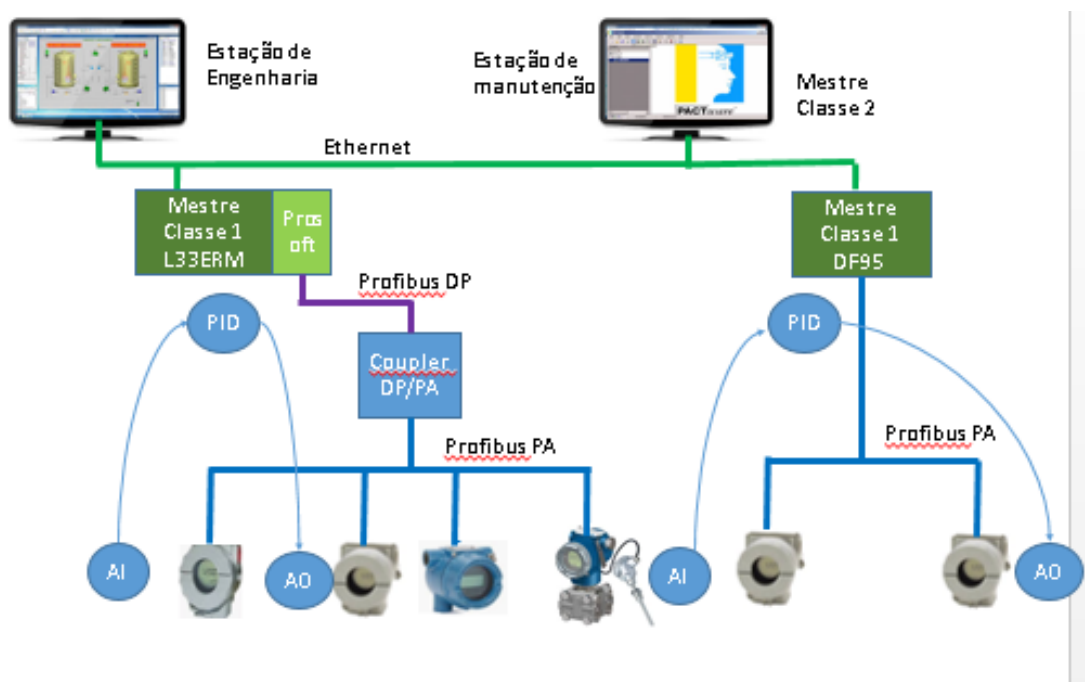
Os fornecedores dos dispositivos DTM pode criar uma interface gráfica de usuário que faz virtualmente qualquer coisa possível criando assim uma facilidade de visualização do processo entre o usuário e o sistema.

Existem 3 tipos de DTMs para configurar nas ferramentas FDT eles são: DTMs de Comunicação eles representam o elemento de comunicação física por exemplo o PLCs; DTMs de *Gateway* eles são responsáveis por representar elementos que fazem a interligação de protocolos diferentes; DTMs de *Devices* eles representam o instrumento de campo propriamente dito (CARRIJO,2011, p.30).

3 PROCEDIMENTOS METODOLÓGICOS

Neste capítulo será descrito a proposta de implementação das ferramentas de padrão aberto que serão utilizadas neste trabalho. A arquitetura proposta é formada por três níveis: Supervisão, comunicação e nível dos sensores e atuadores. A figura 34 mostra a arquitetura proposta para este trabalho.

Figura 34- Ilustração da disposição física do projeto



Fonte: Autor, 2019.

No **nível de supervisão** estão os sistemas de supervisão e controle usados na operação da planta para monitorar as variáveis do processo e também o sistema de gerenciamento de ativos que seria a supervisão voltada para a manutenção da planta. No projeto será abordado os dois sistemas de supervisão como um critério de comparação.

O **nível de controle** é responsável pela gestão do sistema de controle Distribuído (DCS) e também pela comunicação entre os níveis de campo e o nível de supervisão. Neste nível encontra-se os controladores lógicos programáveis e as interfaces de gateways de comunicação.

No **nível de campo** está a rede de campo formada pelos sensores e atuadores que interfaceiam diretamente as variáveis do processo industrial. Como mostrado anteriormente, atualmente, os sensores e atuadores podem comunicar em diferentes protocolos de redes industriais.

Esse trabalho pretende buscar novas ferramentas que utilizem padrões abertos que possibilite acessar os dados e configurar os transmissores e o estudo sobre o Protocolo Profibus DP e PA, em relação ao seus formatos de mensagens e tempo de rede, para isso será montado um sistema conforme mostrado na figura 34.

3.1 Descrição dos equipamentos de Campo

O nível de equipamentos de campo é formado pelos sensores e atuadores que fazem contato direto com as variáveis do processo.

De acordo com o tipo de leitura e armazenamento das variáveis do processo elas podem ser digitais com valores de 0 ou 1, ligado ou desligado, como contatos, chaves, reles e também analógicas, como vazão, pressão, temperatura, nível, PH, entre outras, onde a característica desta variável é um valor analógico inteiro ou ponto flutuante. As variáveis de interesse neste trabalho serão principalmente as variáveis analógicas que tem a ver com controle de processo contínuo.

Com relação ao protocolo de comunicação, neste trabalho será utilizada a comunicação com todos os transmissores do laboratório que seguem as especificações do Profibus-PA ou seja utilizam o modelo de blocos funcionais entre eles estão os transmissor de temperatura, transmissor de pressão/vazão e transmissor de corrente de diferentes fabricantes.

Os transmissores envolvidos no trabalho são:

3.1.1 TT303 – Transmissor de Temperatura (Smar)

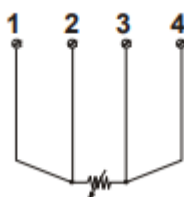
Esse transmissor contém 2 blocos transdutor que contém os dados reais de medição de temperatura.

Esse transmissor possui também 2 blocos de entrada analógicas AIs (Analog Input), esse bloco processa a medição e a torna disponível ao segmento Profibus. O modelo de operação foi selecionado como Automático, isso permite que se o bloco produzir quaisquer saídas, estas continuarão a ser atualizadas

Além desses blocos o transmissor TT 303 possui um bloco físico e um bloco de display que é usado para configurar a exibição local, caso seja necessário configurar algum parâmetro no *display*.

Como todo projeto foi feito em um ambiente simulado foi ligada uma caixa de resistência no transmissor para simular a temperatura, foi feito a configuração nos parâmetros do bloco transdutor como o tipo de sensor que para esse projeto foi selecionado um sensor do tipo OHMs, a escala do transmissor foi modificada para 0 a 2000 ohms e a forma de ligação do sensor no transmissor que foi a 4 a figura 35 mostra ligação do sensor no transmissor.

Figura 35 - Entrada para termoresistência ou OHM a 4 fios



Fonte: Adaptado de (SMAR,2009)

3.1.2 FI303 – Transmissor de Corrente (Smar)

O Transmissor FI 303 possui três blocos transdutores de saída, um bloco físico, um bloco transdutor de display e três blocos de saídas analógicas.

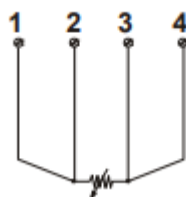
Para configuração desse transmissor foi preciso alterar alguns parâmetros do bloco transdutor que é o responsável pelos dados reais de medição do sensor. No bloco transdutor foi mudada a escala de 4 a 20 mA para 0 a 100%.

Esses blocos de saída analógica são blocos funcionais usado pelo equipamento para que funcionem como elementos de saída como válvulas, atuadores entre outros. Esse bloco foi selecionado como modo de operação Automático isso faz com que a saída do bloco seja calculado usando a entrada do bloco transdutor e usando o valor de *setpoint* fornecido por um servidor ou um operador.

3.1.3 644 – Transmissor de Temperatura (Rosemount)

O transmissor de temperatura 644 permite o uso de blocos funcionais, possui dois blocos transdutores que para esse projeto foi configurado o modo do bloco que foi selecionado como automático isso permite que se o bloco produzir quaisquer saídas, estas continuarão a ser atualizadas. O tipo do sensor selecionado foi OHMs já que foi utilizado uma caixa de resistência para simular a temperatura e o tipo de ligação do sensor no transmissor que foi a 4 fios conforme mostrado na figura a seguir.

Figura 36- RTD 4 fios e OHMs



Fonte: Adaptado de (SMAR,2009)

Possui 2 blocos de entrada analógica que processa as medições do sensor e as disponibiliza para outros blocos de função e pode ser monitorado o valor lido pelo sensor aqui foi ajustado a escala para leitura do sensor que passou a ser de 0 a 2000 Ohms.

Esse transmissor possui também um bloco físico que contém as informações do dispositivo e do fabricante.

3.1.4 2051- Transmissor de Pressão (Rosemount)

O transmissor de pressão permite o uso de blocos funcionais, ele possui um bloco físico, 2 blocos de entrada analógica, 2 blocos transdutores e um bloco transdutor de display.

Nesse transmissor não foi possível ligar nenhum sensor pois não tinha nenhum no laboratório, então os teste neles ficaram limitados em apenas leituras cíclicas e acíclicas da variável, não foi realizado nenhum ajuste de escala ou sensor utilizado.

3.1.5 Siemens Simatic DP/PA Coupler CLP S7-300

O coupler DP/PA é uma passagem entre Profibus DP e Profibus PA, eles são conversores de sinal que adaptam os sinais RS485 ao nível de sinal IEC 61158-2.

3.2 Sistema de Controle e Gateway

Neste capítulo será mostrado os passos realizados para construção da rede Profibus e configuração das ferramentas de padrão aberto realizados através de dois cenários diferentes:

- Cenário com o Controlador Rockwell Compact Logix
- Cenário com o Controlador Smar DF95

A diferença entre o Cenário 1 e o Cenário 2 é somente a mudança do controlador, mantendo os mesmos elementos do nível 1 e nível 3. Desta forma será possível uma comparação

entre as diferenças de performance entre os dois cenários. O trabalho inicialmente também contemplava um terceiro cenário com o Controlador da Siemens S7315. Porém o controlador apresentou problemas durante os testes e não pode ser validado neste trabalho.

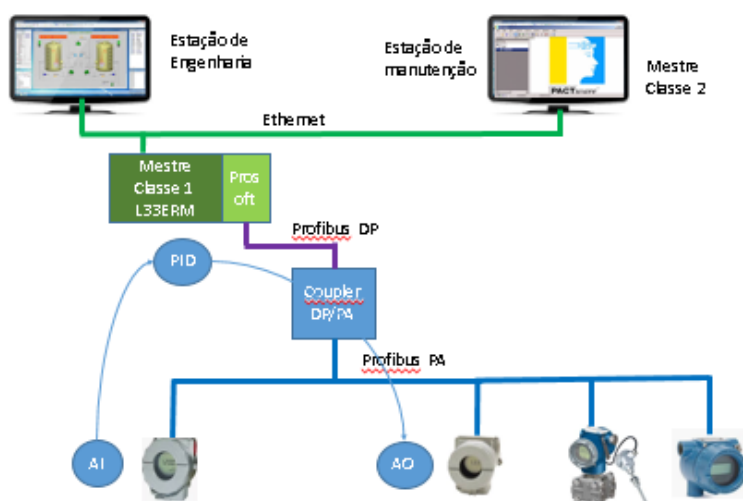
3.2.1 Cenário 1 - Configurando Controlador Rockwell Compact Logix

Como primeiro cenário para os testes foi escolhido o sistema da Rockwell para implementação das ferramentas de padrão aberto e estudo das mensagens acíclicas.

Primeiramente foi construído um sistema Profibus de acordo com a figura 37. O sistema consiste de uma estação de Supervisão e Engenharia onde possui o software de configuração Studio 5000 além do software de Configuração de rede Profibus chamado *Prosoft Configurator Builder* (PCB).

O controlador Mestre classe 1 é formado pela CPU Compact Logix L33ERM e também um cartão scanner Profibus chamado Prosoft MVI69. O Mestre Classe 1 é responsável por realizar a comunicação cíclica da rede de campo. No ambiente da Rockwell o cartão Prosoft é o mestre da rede e responsável por fazer toda a comunicação Profibus com os escravos. E o cartão L33ERM é responsável por ler a informação do cartão scanner Prosoft e executar a lógica de controle em linguagem *Ladder*. No PCB é feito a configuração da rede de campo Profibus.

Figura 37- Arquitetura do sistema Profibus PA do laboratório

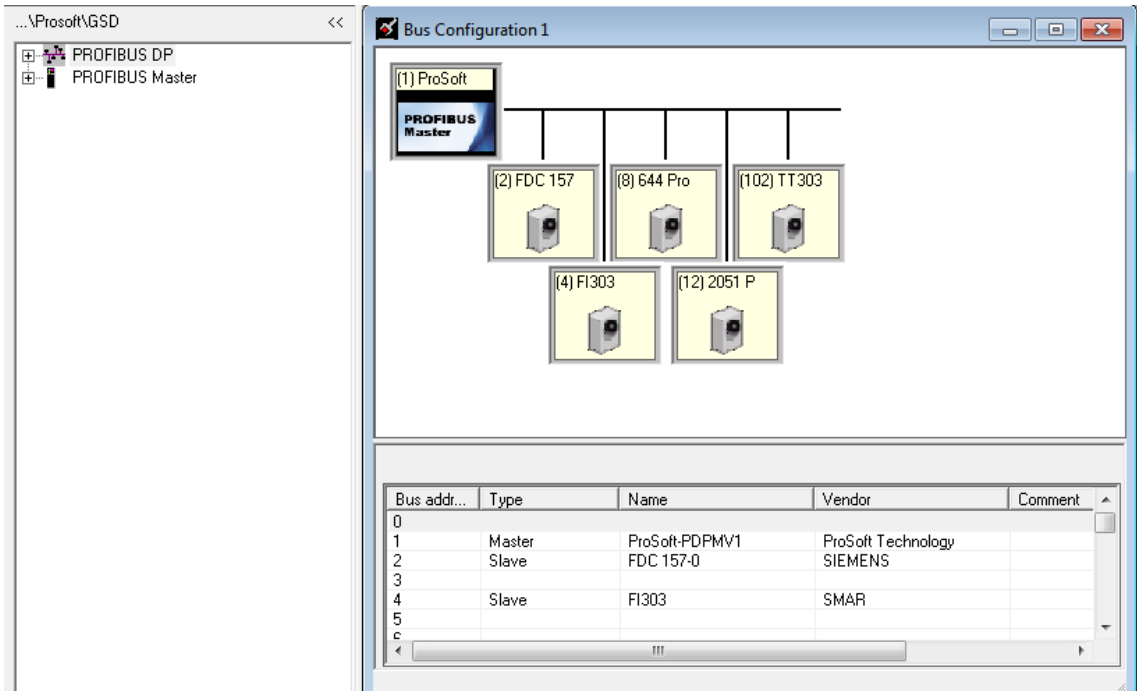


Fonte :Autor, 2019

Para este trabalho, a rede foi configurada adicionando o Mestre Profibus e os escravos da comunicação conforme mostrado na Figura 37. Neste caso é necessário obter o arquivo GSD

de cada equipamento. A partir do arquivo é possível configurar todos os equipamentos de campo. Para o Software PCB não existe distinção entre rede profibus DP e PA conforme mostra a figura. O baudrate da rede deve ser 45.45kbps devido ao Coupler DP/PA.

Figura 38- Configuração da rede de campo no software da PCB



Fonte: Autor,2019

Os parâmetros utilizados para configuração dos escravos estão apresentados na tabela

2.

Tabela 2- Parametrização dos dispositivos Profibus PA

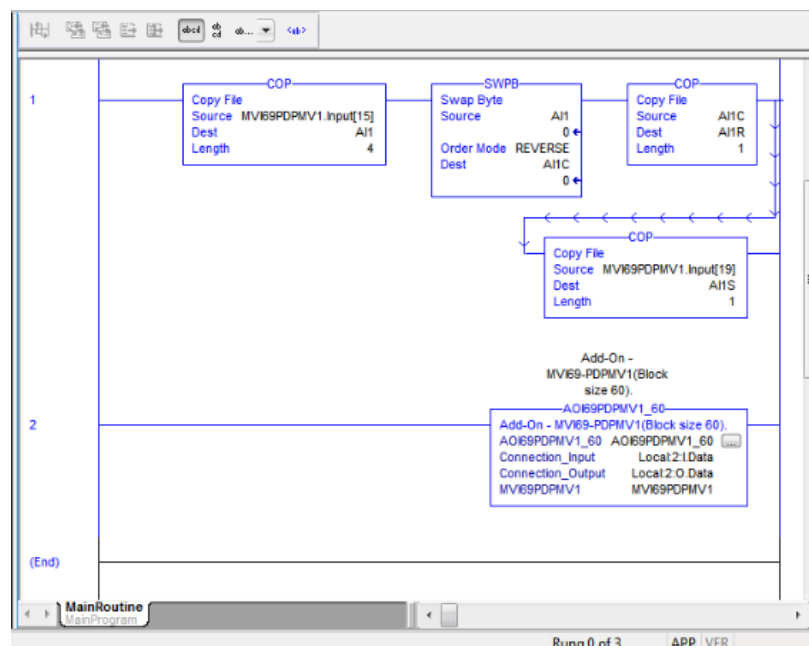
Nome	Endereço	Número do Pedido	Input address	Output Addrees
FI303	4	SP Empty_Module Empty_Module		0...4
TT303	102	Analog Input(short) Analog Input(long)	15...19 10...14	
644	8	Analog Input (short)	25...29	
2051	12	Analog Input (short)	20...24	
Coupler DP/PA	2	Voltage Current	0...4 10...14	

Fonte: Autor,2019

Após realizada a configuração no software PCB, deve ser feita a associação da configuração do Prosoft para o ambiente RSlogix 5000 da Rockwell para que ele possa enxergar os endereçamentos e então fazer a lógica de controle. A figura 39 mostra uma parte da linguagem ladder montada no controlador. Desta forma, a estratégia de controle é basicamente a leitura da saída do bloco Analog Input do dispositivo com endereço 102 que representa a PV do bloco PID. O bloco PID processa o valor da PV com o SP local e então atua na saída. A saída do bloco PID do controlador é ligado na entrada SP do bloco AO do dispositivo 4.

Aqui existe uma dificuldade no sistema da Rockwell para trabalhar com o Profibus. Como mostrado no capítulo 2, a leitura da saída do bloco AI é uma estrutura formada pelo Valor (float) e status (unsigned 8). E o sistema Profibus utilizada o padrão big endian no formato dos dados. E a CPU da Rockwell é little endian, ou seja, é necessário converter os dados na memória para ser possível ler. E também o Prosoft não decodifica o dado.

Figura 39- Exemplo de ladder no Studio 5000



Fonte: Autor,2019

De acordo com a figura 39, a lógica representa as transformações necessárias para leitura do parâmetro de saída do bloco AI do transmissor de temperatura da Smar (TT303).

Na linha 2 representa um bloco funcional da Prosoft para fazer o scan do Cartao e trazer para a memória do Controlador L33ERM. Na linha 1, o primeiro bloco COP copia o valor da área do Prosoft para variável de 4 bytes chamada AI1. O segundo bloco SWPB faz a conversão de cada byte da variável AI1 para a variável AI1C também de 4 bytes transformando de *big endian* para *little endian*. Por fim no bloco COP converte a variável AI1C em um número para REAL (4 bytes) e armazena na variável AI1R. Por fim o último bloco COP da primeira linha copia o status da área do Prosoft para a variável AI1S de 1 byte. Desta forma, é possível agora ligar a variável AI em um bloco de controle PID.

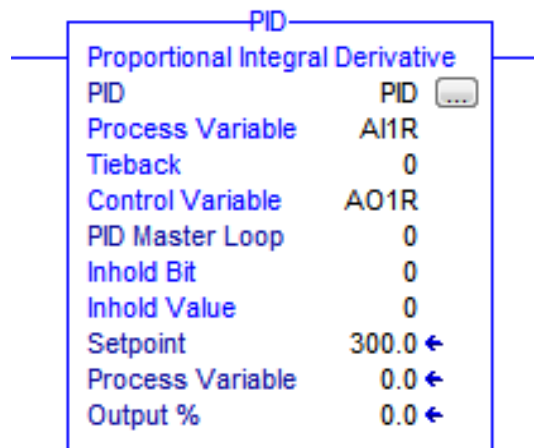
3.2.2 Controlador PID *RSlogix*

A lógica montada no *RSlogix* 5000 funciona da seguinte forma, ao transmissor TT303 está ligado uma caixa de resistência simulando um sensor, no bloco PID foi ajustado o valor de *setpoint* que é o valor onde a temperatura deve permanecer, o PID compara o valor de temperatura vindo do transmissor TT303 e compara com o valor de *setpoint* caso haja alguma

diferença nos valores comparados o PID atua no transmissor de corrente FI303 tentando colocar a temperatura de volta no valor de *setpoint*.

A figura 40 mostra o bloco PID montado no Studio 5000.

Figura 40 - Bloco PID Studio 5000

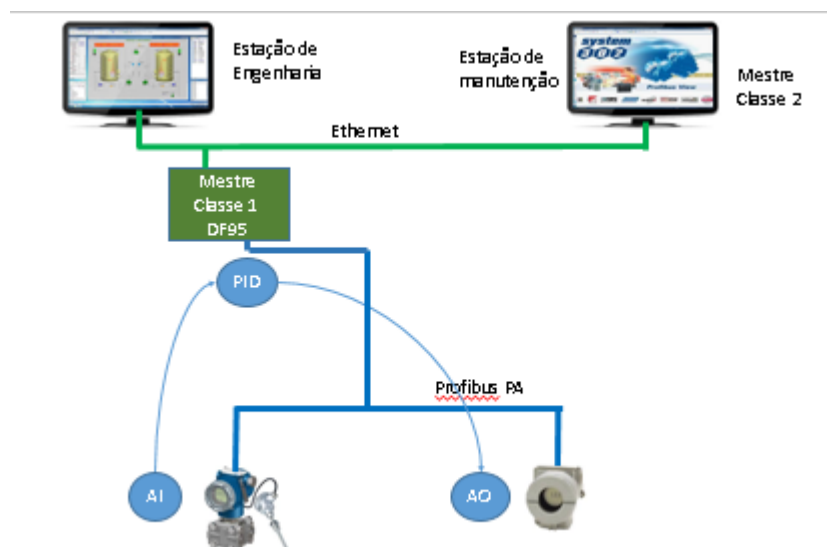


Fonte: Autor, 2019

3.2.3 Cenário 2 - Configurando ambiente Smar

No sistema da Smar a rede Profibus é configurada usando o *software* configurador Syscon que é uma ferramenta que configura, supervisiona e opera equipamentos *Fieldbus* através da Estação de Engenharia. A arquitetura da rede Profibus montada no laboratório é mostrada na figura 41.

Figura 41- Rede montada no laboratório a partir do sistema SMAR



Fonte : Autor,2019

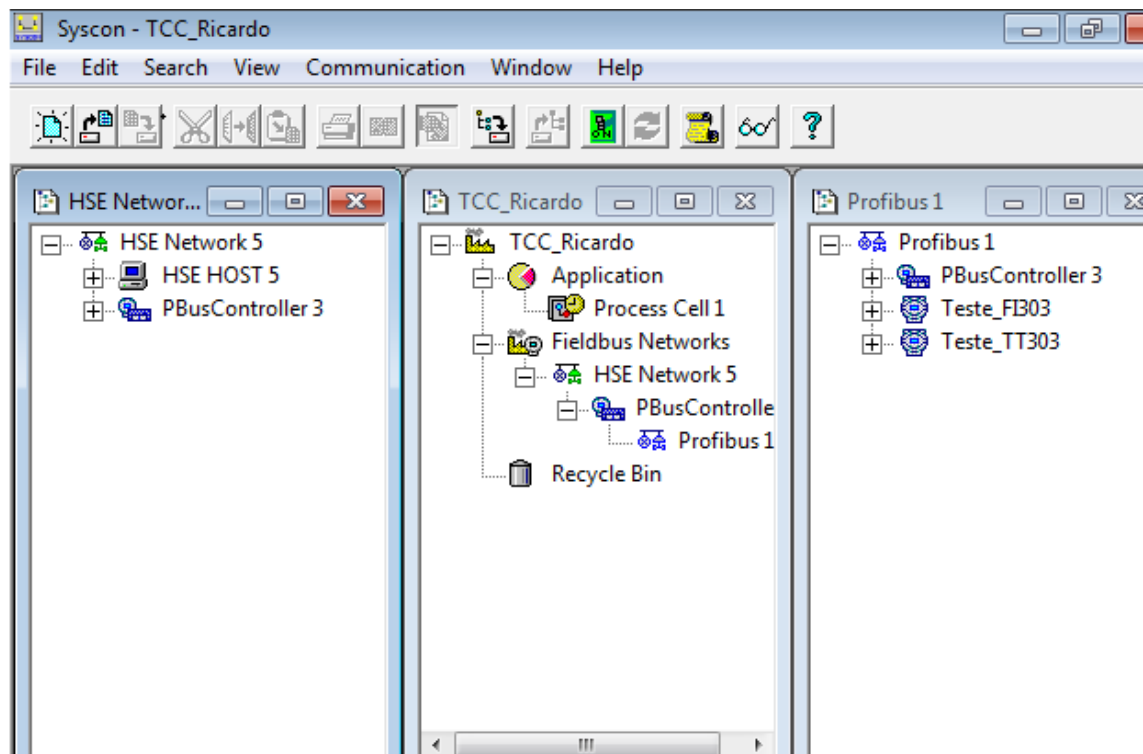
De acordo com a figura 41 o Mestre Classe 1 do cenário 2 foi utilizado o controlador HSE/DF95. O DF95 não precisa de Coupler DP/PA pois ele possui dois canais Profibus PA e um canal Profibus DP.

Com a rede Profibus configurada no Syscon a configuração é dividida em duas partes:

- *Application* – Nesta seção são inseridas as estratégias de controle.
- *Fieldbus Networks* – Nesta seção são adicionados os equipamentos e blocos funcionais.

A figura 42 mostra as telas de configurações da Aplicação do projeto enquanto que a figura 43 mostra a tela de configuração do Fieldbus Network.

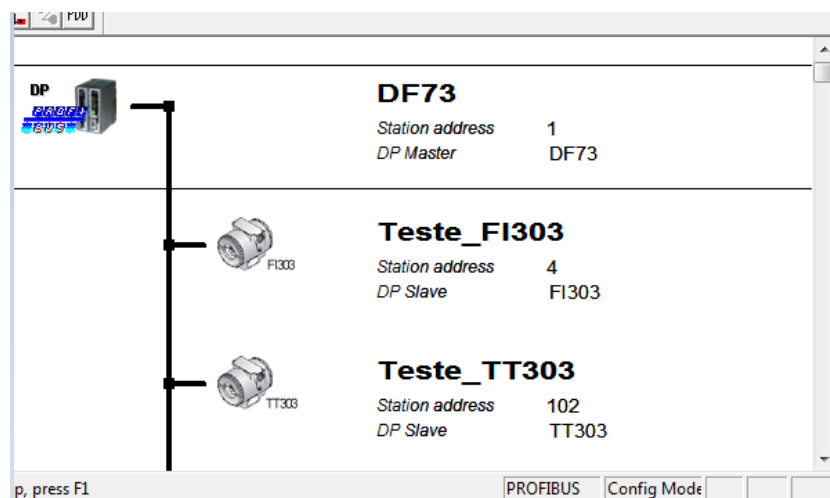
Figura 42- Tela configuração no Software Syscon



Fonte : Autor,2019

Área *Fieldbus Networks* : Aqui foi adicionado os instrumento na rede de comunicação com um nome (*tag*) para diferenciar os instrumentos e o endereço do escravo para identificação do equipamento na rede os dispositivos adicionados foram o TT 303 (Transmissor de Temperatura) e o FI 303 (Transmissor de Corrente) conforme mostrado na figura 43.

Figura 43- Tela da configuração dos equipamentos na rede pelo software Syscon



Fonte : Autor,2019

A figura 44 mostra as configurações utilizadas nos blocos AOs do transmissor FI 303, foi utilizado como parâmetro de entrada do AO o modo SP(setpoint) e os outros blocos AOs que não foram utilizados foram configurados como EMPTY_MODULE por não terem sido usados na configuração.

Figura 44- Parâmetros de configuração FI303

Slot	Idx	Module	Symbol	Type	I Addr.	I Len.	Type	O Addr.	O Len.
1	1	eSP	Module1				QB	0	5
2	1	EMPTY_MODULE	Module2						
3	1	EMPTY_MODULE	Module3						

Fonte: Autor,2019

A figura 45 mostra a configuração utilizada nos blocos AIs do transmissor TT303, foi utilizado apenas um bloco AI com parâmetro OUT curto e o outro bloco foi configurado como EMPTY_MODULE por não ter sido utilizado na configuração.

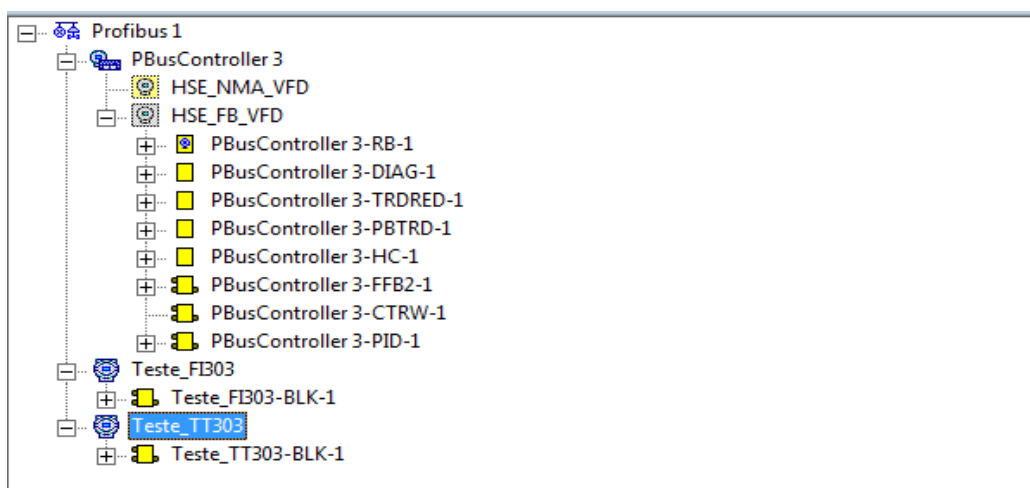
Figura 45- Parâmetros de configuração TT303

Slot	Idx	Module	Symbol	Type	I Addr.	I Len.	Type	O Addr.	O Len.
1	1	Analog	Module1	IB	0	5			
2	1	EMPTY_MODULE	Module2						

Fonte: Autor,2019

Com os instrumentos adicionados na rede Profibus foram adicionados os blocos funcionais na aplicação. Diferentemente da ferramenta da Rockwell, os blocos Criados já mapeiam os valores de entrada e saída em blocos funcionais Profibus e desta forma não é preciso mapear dados de memória como é feito no ambiente da Rockwell. E então para criar uma estratégia AI-PID-AO basta criar os blocos necessários conforme mostrado na Figura 45. De acordo com a figura o bloco AI é representado pelo bloco Teste_TT303-BLK-1, o bloco PID está no PBusController 3 (DF95) com o nome PBusController 3-PID-1. Por fim, o bloco AO é representado pelo bloco Teste_TT303-BLK-1.

Figura 46- Blocos Funcionais utilizados no projeto



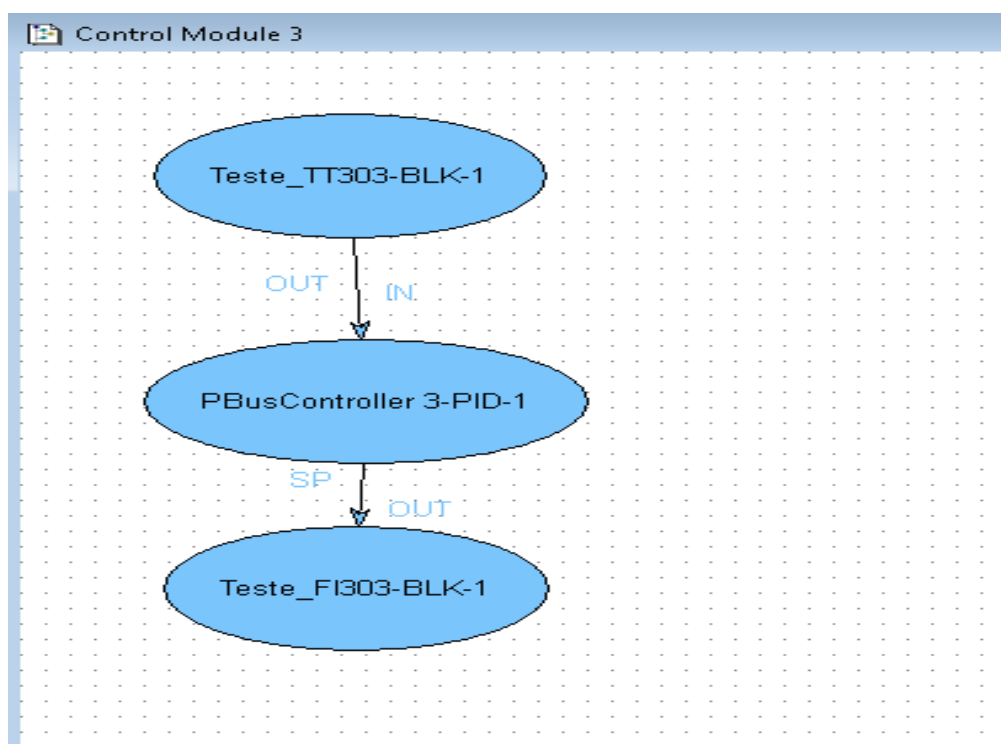
Fonte : Autor,2019

3.2.4 Controle PID Syscon

Por fim, foi criada uma estratégia de controle AI-PID-AO ligando os blocos conforme mostrado na figura 47.

A lógica montada no *Syscon* é similar a montada no *RSlogix 5000* ao transmissor TT303 está ligado uma caixa de resistência simulando um sensor, no bloco PID foi ajustado o valor de *setpoint* que é o valor onde a temperatura deve permanecer, o PID compara o valor de temperatura vindo do transmissor TT303 e compara com o valor de *setpoint* caso haja alguma diferença nos valores comparados o PID atua no transmissor de corrente FI303 tentando colocar a temperatura de volta no valor de *setpoint*.

Figura 47-Controle AI-PID-AO construída no Syscon



Fonte: Autor,2019.

3.3 Nível de Supervisão

Uma das proposta é analisar a influência que um sistema SCADA e uma ferramenta de manutenção pode causar na rede pelo excesso de supervisão e informações trafegados na rede, já que essas ferramentas estão presentes em uma indústria e estão em funcionamento simultaneamente.

A ferramenta de manutenção é baseado em um sistema FDT/DTM é capaz de monitorar a condição dos equipamentos possibilitando a manutenção dos instrumentos, ele é responsável pelas mensagens acíclicas.

O sistema OPC é representado pelo sistema SCADA, responsável pelo monitoramento do processo e por analisar todas as etapas em tempo real ele representa a comunicação cíclica, com isso será testado a influência dele na rede de comunicação.

3.3.1 Tecnologia FDT/DTM

Nesta parte do trabalho foram realizados estudos para determinação de uma ferramenta aberta que para manutenção de equipamentos de campo. Como não existia nenhuma ferramenta Mestre Classe 2 no laboratório de Controle e Automação Industrial, o desafio consistiu em verificar ferramentas que se adequassem aos equipamentos existem sem necessidade de compra de licenças ou outros equipamentos adicionais. Para a comunicação através do FDT/DTM é preciso que o próprio equipamento suporte essa tecnologia pois ele deve responder serviços específicos de comunicação.

As ferramentas implementadas no projeto foram o PACTware e o *Profibus View* que facilita a representação da configuração do sistema através de uma configuração por meio de um gráfico de processamento de aplicativos. Possibilita comparação do plano do projeto com o sistema real disponível ao estabelecer conexões no dispositivo.

O usuário tem uma maior facilidade de configuração do sistema como representação gráfica de todos os gerentes de comunicação e tipo de dispositivo na estrutura da árvore. Usa um código de cores de operação on-line para sinalização de unidades com defeito e operação de simulação. Fazendo ser possível visualizar os parâmetros do dispositivo configurado, monitorar o valor da medição e mostrar o diagnóstico do dispositivo ao mesmo tempo.

3.3.2 Ambiente Rockwell com PACTware

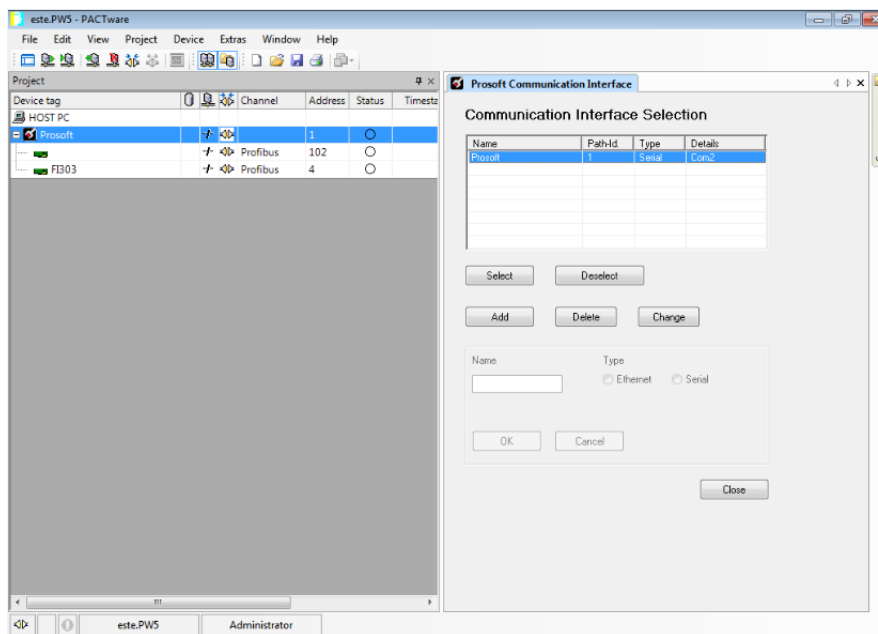
O Mestre Classe 2 é formado por um software de configuração dos equipamentos de campo, neste caso foi utilizado o Software Pactware que é um software opensource e será uma contribuição para o laboratório. O mestre de classe 1 será utilizado o PLC Compact Logix L33ERM da empresa Rockwell Automation. Para fazer a varredura da rede é necessário um cartão no CLP chamado de Prosoft. Também é necessário um coupler DP/PA para conversão do meio físico. Neste caso foi utilizado o Coupler da Siemens. Os dispositivos de campo utilizados foram o transmissor de corrente FI 303 da SMAR, o transmissor de temperatura TT303 da SMAR, o transmissor de temperatura da Emerson 644 e o transmissor de pressão da Emerson 2051.

Primeiramente para que essa ferramenta possa atuar nos parâmetros dos equipamentos é necessário obter os DTMs dos equipamentos disponibilizado no site do fabricante. O fornecedor do sistema host fornece um DTM de comunicação (comDTM) para cada protocolo

suportado. Isso garante que uma transparência para o usuário na configuração de detalhes como PC, rede, placas de interface e protocolos de passagem do sistema hospedeiro (Profibus,2006).

Os DTMs utilizados nesse projeto foram: DTM da Prosoft que é o DTM de comunicação ou de Gateway e os DTM dos dispositivos TT303 e FI303. Estes DTMs são inseridos no PACTware que possui a interface FDT.

Figura 48-Tela de Configuração do PACTware



Fonte: Autor,2019

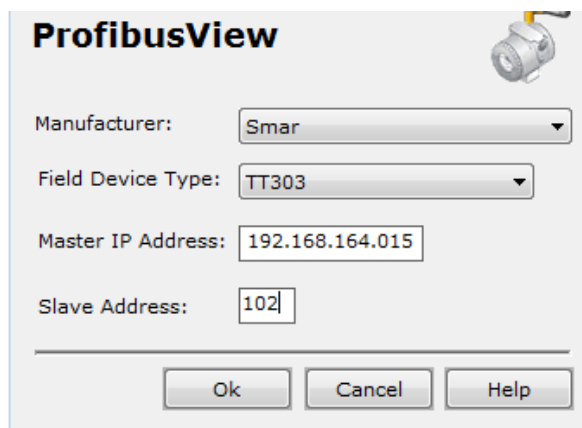
Em uma configuração, primeiramente é inserido o DTM da Prosoft para configurar a interface de comunicação, que no caso deste foi via serial pela porta COM2 conforme mostrado a figura 48. Depois de adicionado o DTM de gateway da Prosoft é adicionado os DTMS dos escravos e os endereços dos dispositivos na rede como ilustrado na figura 49 o transmissor TT303 tem o endereço 102 e o FI303 possui o endereço 4.

3.3.3 Ambiente Smar *Profibus View*

No ambiente da SMAR não foi possível utilizar o PACTware, pois ao conectar o DTM de *gateway* da SMAR apresentou um erro e como o tempo estava curto não foi possível tentar corrigir o erro, então foi utilizada a ferramenta da SMAR *Profibus View*.

Como essa ferramenta é própria para os equipamentos da SMAR, é necessário configurar somente o endereço de IP do mestre e selecionar o equipamento e seu endereço para iniciar a comunicação com a ferramenta. A figura 49 mostra a ferramenta *Profibus View*.

Figura 49- Ferramenta SMAR



Fonte: Autor,2019

3.3.4 Sistema SCADA

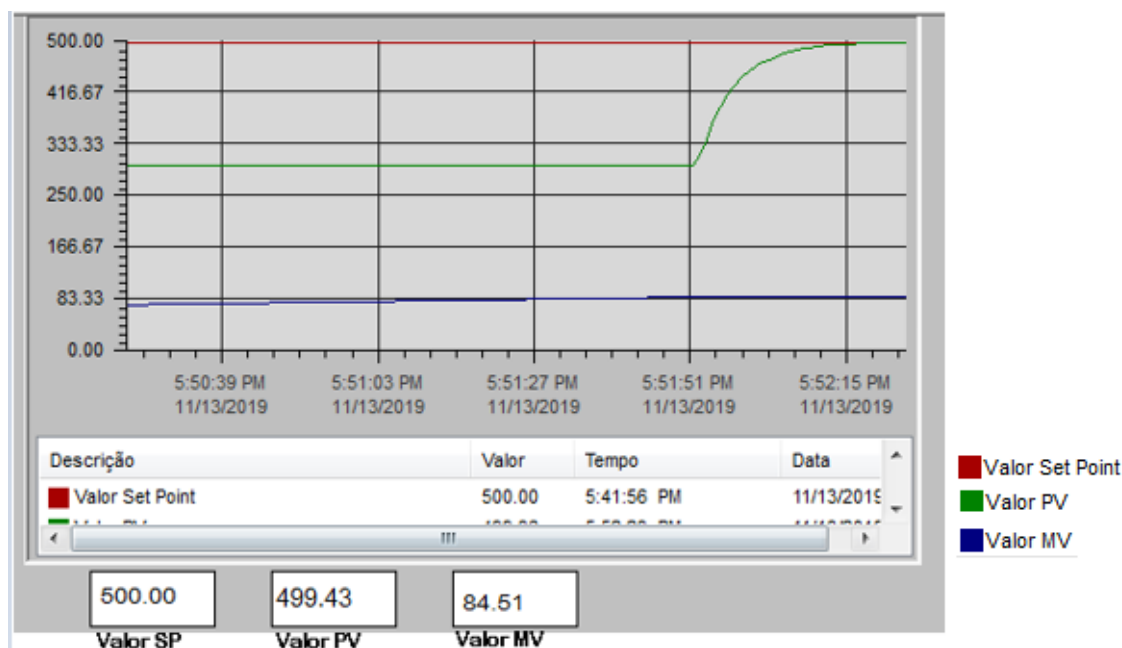
O Mestre Classe 2 é formado por um software de configuração dos equipamentos de campo conforme mostrado na seção anterior. Porém, no Profibus, a Supervisão SCADA é realizada até o nível do Controlador, ou seja, um sistema SCADA não pode supervisionar os Pontos além dos parâmetros do Controlador.

Para o ambiente Rockwell foi feito um supervisório utilizando o elipse E3, que é uma ferramenta SCADA para monitoramento e controle de processos. Para que o Elipse E3 enxergasse as variáveis de processo foi utilizado o *driver* ABCIP que utiliza o protocolo Ethernet/IP.

Para o ambiente da SMAR foi utilizado o software *Process View* que utiliza um driver OPC de comunicação.

A partir dessas ferramentas foi construída a tela para visualização e monitoramento das variáveis de processo as telas. Na tela do supervisório é mostrado os parâmetros SP, PV e OUT do bloco PID e gerado um gráfico em tempo real dos valores das variáveis de processo e telas de *displays* onde é possível monitorar os valores e escrever valor no setpoint, as telas para os dois ambientes foram construídas contendo os mesmo parâmetros a figura 50 ilustra as variáveis monitoradas pelo supervisório.

Figura 50 -Tela do Supervisório usando Process View



Fonte: Autor,2019

3.4 Metodologia de Validação

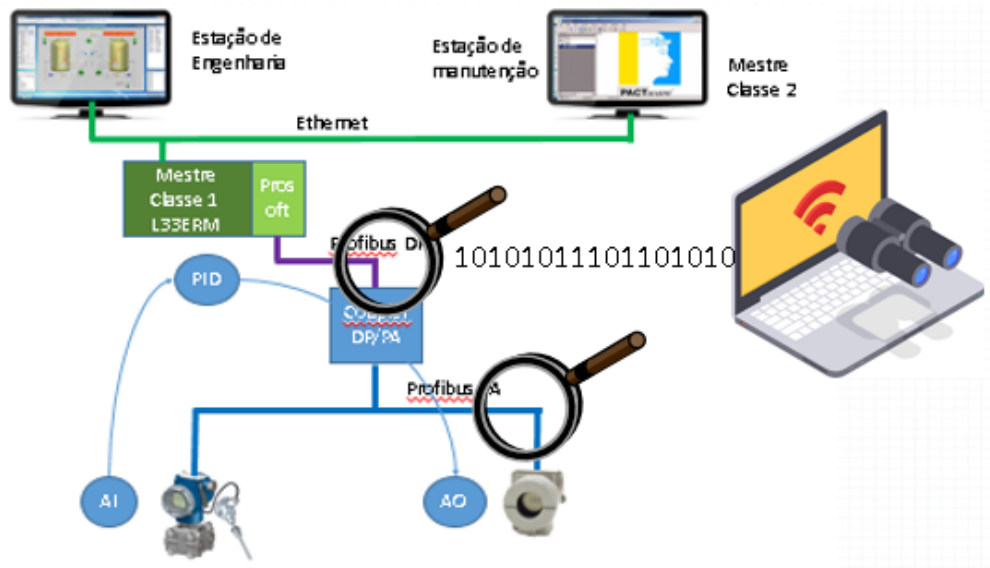
Um dos objetivos desse projeto é analisar a influência da comunicação acíclica no tempo da rede e comparar com o tempo da comunicação cíclica.

Para a análise da performance dos dados acíclicos dentro do ambiente de controle foi montado uma estratégia de controle AI-PID-AO que é muito utilizada no ambiente industrial. Esta estratégia de controle de comunicação cíclica tem prioridade sobre a comunicação acíclica e deve ser determinística, ou seja, deve sofrer o mínimo de *drift* ou variações no período de comunicação. Para isso será feito uma análise de tempo da rede somente com comunicação cíclica e outra análise com a comunicação acíclica.

Esta análise do tempo será feita através da captura da rede. Esta captura é feita através de ferramentas de análise de rede Profibus DP e também captura da latência da rede nas ferramentas de supervisão. Com isso podemos estudar a influência da comunicação acíclica em comparação da comunicação cíclica em diferentes aspectos dentro de uma rede Profibus PA e em diferentes cenários.

Essas ferramentas são conhecidas como *Sniffer* que são os responsáveis por interceptar e registrar tráfego que passa sobre a rede conforme ilustrado na figura 51.

Figura 51- Exemplo de captura do dado pelo Sniffer

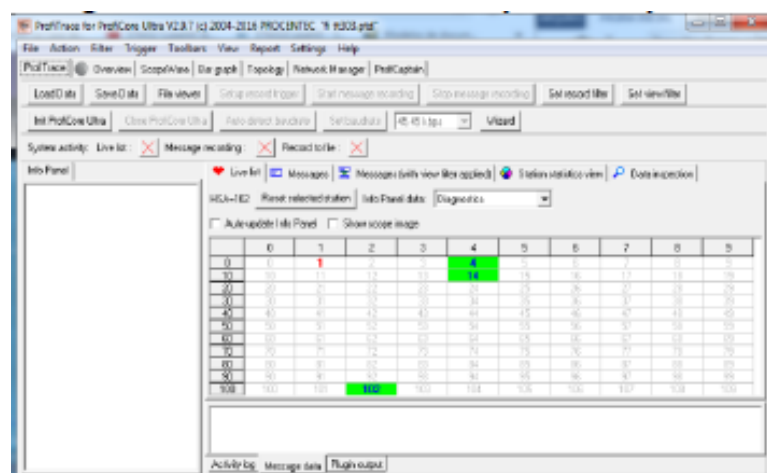


Fonte: Autor,2019

Para análise dos dados no ambiente da Rockwell foi utilizada a ferramenta de escuta de linha ProfiTrace. Ela consegue capturar todos os pacotes de mensagens trocados entre todos os equipamentos, bem como os respectivos tempos da rede.

A figura 52 mostra um exemplo da tela da ferramenta Profitrace, onde mostra os endereços vivos na rede que são: o transmissor FI303 com endereço 4, o transmissor TT303 com endereço 102, e o cooplér DP/PA no endereço 14.

Figura 52- Tela de Livelist do Software Profitrace



Fonte: Autor,2019

No ambiente da SMAR, como não é necessário passar pelo Profibus DP para acessar a rede Profibus PA, pois ele já possui o canal PA direto foi utilizada uma outra ferramenta

chamada *FBView* que consegue pegar somente as mensagens que passam no protocolo Profibus PA.

Como foram feitas capturas de dois meios físicos diferentes, isto não indica que a mensagem Profibus DP é diferente da mensagem Profibus PA. Nas figuras 53 e 54 é possível comparar as mensagens capturadas pelos dois analisadores de linha. Na figura 53 é mostrado um exemplo de captura Profibus PA feito com o software *FBView*, enquanto que na figura 54 foram obtidos um frame de características Profibus DP através do Profitrace.

Figura 53- Frames Profibus PA

7	6:25:19.098	0	0	73	0	0	0	Pass Token	01 10				
8	6:25:19.116	1	4	125	0	0	5	Data Exchange	68 08 08 68 04 01 7D 3F B8 8E DE C0 A1 EC				
9	6:25:19.121	0	0	125	0	0	0	SHORT ACK	E5 14 26				
10	6:25:19.134	1	102	93	0	0	0	Data Exchange Req	10 66 01 5D 37 B9				
11	6:25:19.141	102	1	8	0	0	5	Data Exchange	68 08 08 68 01 66 08 43 95 EC 5A 80 A0 A9				
12	6:25:19.155	1	32	73	0	0	0	FDL Status	10 20 01 49 86 85				
13	6:25:19.196	0	0	73	0	0	0	Pass Token	01 10				

Fonte: Autor,2019

Figura 54- Frames Profibus DP

1	12-Sep-2...	SD4	1->1	DC 01 01
2	12-Sep-2...	SD2	1->4	68 08 08 68 04 01 5D 42 8C 00 00 80 B0 16
3	12-Sep-2...	ACK		E5
4	12-Sep-2...	SD1	1->102	10 66 01 7D E4 16
5	12-Sep-2...	SD2	1<-102	68 0D 0D 68 01 66 08 42 8C 39 60 80 00 00 00 4C A2 16
6	12-Sep-2...	SD1	1->2	10 02 01 7D 80 16
7	12-Sep-2...	SD2	1<-2	68 0D 0D 68 01 02 08 41 C0 00 00 80 00 00 00 4F DB 16
8	12-Sep-2...	SD1	1->85	10 55 01 49 9F 16
9	12-Sep-2...	SD2	1->102	68 09 09 68 E6 81 5C 2F 32 5E 01 81 F0 F4 16
10	12-Sep-2...	ACK		E5
11	12-Sep-2...	SD4	1->1	DC 01 01

Fonte: Autor,2019

4 RESULTADOS E DISCUSSÕES

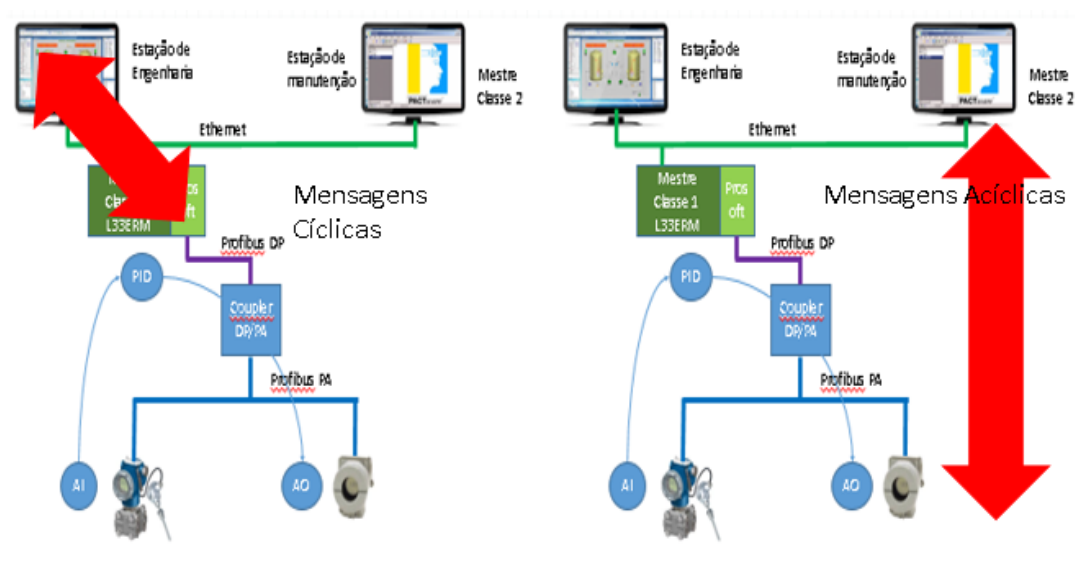
Nessa seção será mostrado os resultados obtidos sobre o estudo das ferramentas abordada nesse projeto e as análise de tempo de *scan* da rede quando se tem somente as mensagens cíclicas e quando se tem as mensagens cíclicas e acíclicas.

4.1 Comparação entre as ferramentas SCADA e FDT/DTM

Este projeto procurou estudar duas ferramentas de supervisão: uma ferramenta SCADA e uma ferramenta de Manutenção. Com base nas duas ferramentas utilizadas nesse projeto pode se observar diferentes características entre elas em termo de supervisão dentro do protocolo Profibus estudado.

No contexto do protocolo Profibus, a ferramenta SCADA que nesse trabalho foram utilizados os softwares Eclipse e Process View, que se destinaria de uma aplicação industrial a supervisionar dados de operação da planta, atua no nível do CLP. Por outro lado, a ferramenta de Manutenção que neste trabalho foi utilizado o PACTware utiliza o padrão FDT/DTM atua no nível de chão de fábrica ou seja ela atua diretamente nos transmissores conforme ilustra a figura 55.

Figura 55- Comparação entre SCADA e FDT/DTM

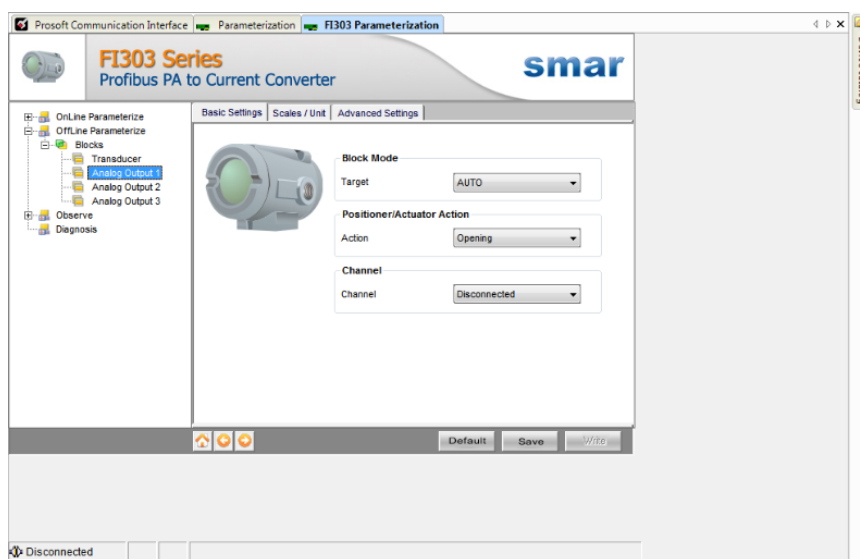


Fonte: Autor, 2019

4.1.1 Estação de Manutenção com o software PACTware

Primeiramente, foi possível já habilitar a manutenção dos equipamentos através do PACTware. Todos os principais parâmetros dos blocos estão disponíveis para o usuário, bem como os métodos de configuração conforme mostrado na figura 56.

Figura 56- Tela de configuração do PACTware



Fonte: Autor,2019

Na figura 56 é mostrado a tela de supervisão do dispositivo FI303. O equipamento FI303 é um transmissor de saída Profibus PA que possui 3 blocos de saída analógica (AO) e três blocos transdutores de corrente, além do bloco display e o *resource*. Foram feitos testes na ferramenta para ver se ela interagia com os transmissores e foi possível alterar todos parâmetros disponíveis, como alterar o modo de operação do bloco, o valor da PV do bloco AO, a escala do transdutor e escrever no display dos transmissores.

4.1.2 Análise de Rede através das ferramentas de escuta de linha Profitrace

Cada um dos blocos tem um conjunto de parâmetros index e subindex que permite o acesso as informações internas do bloco conforme mostrado na figura 57.

Figura 57- Mensagem de Escrita no bloco Display TT303

```

Framestructure: SD2 message
Source address: 1
Destination address: 102
Frametype: Request message

PROFIBUS DP-V1 C2 Message:
Write_Req
Slot=5
Index=27
Length=16
Data :
70 61 00 00 00 00 00 00
00 00 00 00 00 00 00 00

```

Fonte: Autor,2019

A figura 57 é uma captura da ferramenta Profitrace de uma mensagem de escrita no bloco *display* do transmissor TT303 da *string* “pa” obtida através do Pactware. De acordo com a figura 55 o bloco *display* do transmissor TT303 representa o slot 5, o parâmetro alterado no bloco é o 27 e apresenta um tamanho de 16 bytes, onde “70 61” representa o valor ASCII dos caracteres de “pa”.

Na Figura 58 é mostrado uma captura do Pactware no momento de uma leitura de dados dos transmissores TT303 e FI303. Neste caso é possível ver os diferentes tipos de mensagem do protocolo Profibus DP/PA. As mensagens cíclicas são mensagens do tipo *data exchange* e não possuem o campo SAP que é um identificador de funções. Elas somente podem ser enviadas pelo Mestre Classe 1. No exemplo, os pacotes 6, 7, 13,14 são exemplos de mensagens *data exchange*. As mensagens acíclicas podem ser enviadas tanto pelo mestre Classe 1 quanto o mestre Classe 2. Um dos tipos de mensagens acíclicas é a leitura acíclica que possui SAP 50-48. No exemplo, o pacote 10 é uma requisição de leitura, enquanto que o pacote 21 é a resposta.

Figura 58- Captura dos dados da Rede com 2 devices

Frame	Timestamp	Type	Addr	Service	Msg type	Dir	SAPS	Data/L Data
6	19:48.175	SD1	1->14	SRD_HIGH	Data Exchange	Req		
7	19:48.177	SD2	1<-14	DL	Data Exchange	Res		10 41 C0 00 00 80 00 00 00 4F
8	19:48.187	SD1	1->14	FDL Status		Req		
9	19:48.188	SD1	1<-14	Passive		Res		
10	19:48.195	SD2	1->102	SRD_LOW	DPV1_Read_Req	Req	50->48	4 5E 01 81 F0
11	19:48.204	ACK			Short acknowledge	Res		
12	19:48.210	SD4	1->1	Token pass	Pass token			
13	19:48.215	SD1	1->102	SRD_HIGH	Data Exchange	Req		
14	19:48.223	SD2	1<-102	DL	Data Exchange	Res		10 43 95 FC A3 80 00 00 00 4C
15	19:48.233	SD2	1->4	SRD_HIGH	Data Exchange	Req		5 42 C8 00 00 00
16	19:48.242	ACK			Short acknowledge	Res		
17	19:48.247	SD1	1->14	SRD_HIGH	Data Exchange	Req		
18	19:48.249	SD2	1<-14	DL	Data Exchange	Res		10 41 C0 00 00 80 00 00 00 4F
19	19:48.259	SD1	1->15	FDL Status		Req		
20	19:48.274	SD2	1->102	SRD_LOW	DPV1_Poll	Req	50->48	0
21	19:48.283	SD2	1<-102	DL	DPV1_Read_Res	Res	50<-48	8 5E 01 81 04 00 00 00 00
22	19:48.293	SD4	1->1	Token pass	Pass token			
23	19:48.299	SD1	1->102	SRD_HIGH	Data Exchange	Req		
24	19:48.307	SD2	1<-102	DL	Data Exchange	Res		10 43 95 FC A3 80 00 00 00 4C
25	19:48.316	SD2	1->4	SRD_HIGH	Data Exchange	Req		5 42 C8 00 00 00
26	19:48.326	ACK			Short acknowledge	Res		
27	19:48.331	SD1	1->14	SRD_HIGH	Data Exchange	Req		
28	19:48.333	SD2	1<-14	DL	Data Exchange	Res		10 41 C0 00 00 80 00 00 00 4F
29	19:48.342	SD1	1->16	FDL Status		Req		
30	19:48.358	SD4	1->1	Token pass	Pass token			
31	19:48.364	SD1	1->102	SRD_HIGH	Data Exchange	Req		
32	19:48.371	SD2	1<-102	DL	Data Exchange	Res		10 43 95 FC A3 80 00 00 00 4C
33	19:48.381	SD2	1->4	SRD_HIGH	Data Exchange	Req		5 42 C8 00 00 00
34	19:48.390	ACK			Short acknowledge	Res		
35	19:48.396	SD1	1->14	SRD_HIGH	Data Exchange	Req		
36	19:48.397	SD2	1<-14	DL	Data Exchange	Res		10 41 C0 00 00 80 00 00 00 4F

Fonte: Autor,2019

Ainda de acordo com a Figura 58, é possível verificar que a latência de um *data exchange* para o endereço 102 é de 1 a 2 ms enquanto que para o endereço 4 é de 3 a 4 ms. Com relação mensagem acíclica de leitura a latência foi de 3 a 4 ms enquanto que a de escrita possui latência de 7 a 8 ms. Porém é possível notar que a mensagem acíclica nem sempre será resolvida em um mesmo ciclo de *scan* pois ela tem menor prioridade que a mensagem cíclica conforme é definido na norma Profibus.

Outro teste realizado foi de alterar 2 parâmetros dos blocos dos transmissores ao mesmo tempo e do mesmo equipamento, com o objetivo de tentar analisar o comportamento do tráfego da rede quando se tem uma sobrecarga-de mensagens acíclicas. A figura 59 mostra um exemplo o pacote 809 e 877 são frames de escrita que foram enviadas simultaneamente. Mesmo o comando de escrita sendo mandado ao mesmo tempo eles não entram no mesmo ciclo o tempo do primeiro comando de escrita para o segundo é de mais de 600 milissegundos.

Figura 59- Diferença de tempo de escrita em 2 parâmetros simultaneamente

809	19-Jun-2019 16:39:52.766	SD2	1->4	SRD_LOW	DPV1 Write Req	Req	50->47	6	5F 01 77 02 00 02
820	19-Jun-2019 16:39:52.855	SD2	1<-4	DL	DPV1 Write Res	Res	50<-47	4	5F 01 77 02
877	19-Jun-2019 16:39:53.316	SD2	1->4	SRD_LOW	DPV1 Write Req	Req	50->47	5	5F 01 79 01 08
898	19-Jun-2019 16:39:53.483	SD2	1<-4	DL	DPV1 Write Res	Res	50<-47	4	5F 01 79 01

Fonte: Autor,2019

Outros resultados obtidos analisando a rede mostraram que mesmo o PACTware sendo o Mestre de classe 2 ele entra na rede através do endereço do mestre de classe 1 na comunicação, e quando era alterado qualquer parâmetro no bloco pelo PACTware todos parâmetros que contidos no bloco eram enviados de uma só vez até mesmo os que não foram alterados.

Por fim um dos problemas da utilização do PACTware nesta análise é que não é possível enviar mais comandos que existe em uma tela o que limita os teste a poucos parâmetros de leitura e escrita por vez.

4.1.3 Análise de Rede através das ferramentas de escuta de linha *FBView*

Para o sistema da SMAR o dado foi capturado através do software *FBView* conforme mostrado na figura 60. Neste software os frames são apresentados em hexadecimal, onde não existe uma decodificação do tipo da mensagem, como existe na ferramenta *Profitrace* dificultando a análise das mensagens.

Figura 60- Captura através do software FBview

1	16:25:19.000	01 10
2	16:25:19.018	68 08 08 68 04 01 5D 3F B8 8E DE C0 D7 82
3	16:25:19.023	E5 14 26
4	16:25:19.036	10 66 01 7D A8 08
5	16:25:19.043	68 08 08 68 01 66 08 43 95 EC 5A 80 A0 A9
6	16:25:19.057	10 1F 01 49 D1 7C
7	16:25:19.098	01 10

Fonte: Autor,2019

Os dados nessa forma são muitos difíceis de analisar, com isso foi feito um código em *Python* que decodifica o frame Profibus para o *FBView* onde colocamos as capturas e ele gera um arquivo em *Excel* para que ficasse melhor para analisar os dados e assim tratá-los, o código é apresentado no apêndice A. O frame tratado é apresentado na figura 61 a seguir.

Figura 61- Captura depois de ser decodificada pelo código

idx	timestam	src	dst	fc	dsap	ssap	lendata	desc	payload			
1	6:25:19.00	0	0	0	0	0	0	Pass Toke	01 10			
2	6:25:19.01	1	4	93	0	0	5	Data Exch	68 08 08 68 04 01 5D 3F B8 8E DE C0 D7 82			
3	6:25:19.02	0	0	93	0	0	0	SHORTAC	E5 14 26			
4	6:25:19.03	1	102	125	0	0	0	Data Exch	10 66 01 7D A8 08			
5	6:25:19.04	102	1	8	0	0	5	Data Exch	68 08 08 68 01 66 08 43 95 EC 5A 80 A0 A9			
6	6:25:19.05	1	31	73	0	0	0	FDL Statu	10 1F 01 49 D1 7C			
7	6:25:19.05	0	0	73	0	0	0	Pass Toke	01 10			

Fonte: Autor,2019

4.2 Análise do tempo de *scan* da Rede

Por fim, foram realizadas várias capturas para analisar o *scan* da rede tanto sem supervisão e com supervisão entre diferentes equipamentos e com diferentes números de equipamentos na rede. O objetivo era analisar os tempos da rede quando se tem mensagens cíclicas e quando se tem mensagens cíclicas e acíclicas devido a ferramenta de manutenção.

Os teste foram divididos em 2 cenários e as coletas de dados foram feitas da seguinte forma:

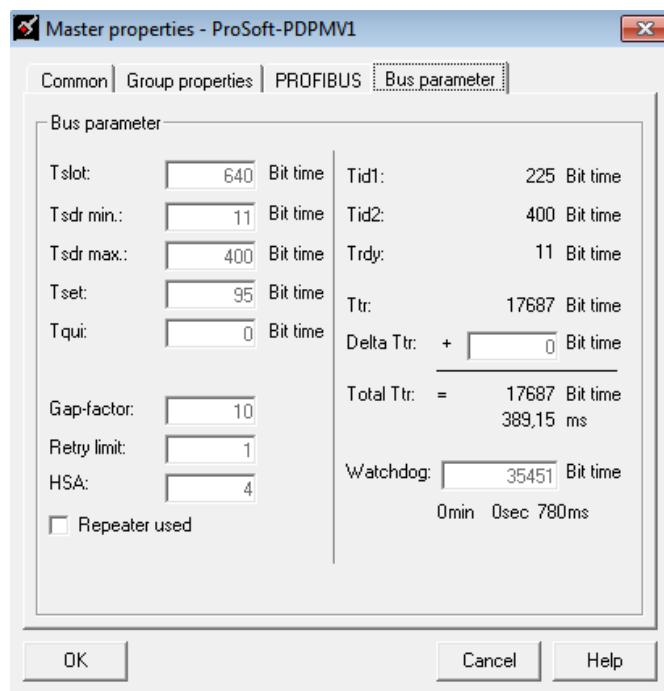
O primeiro cenário foi montado com o CLP da Rockwell Compact Logix, uma estação de supervisão e uma de manutenção. A rede Profibus foi ajustada inicialmente com um *device* e feito a coleta dos dados de comunicação pela ferramenta de escuta de linha, após o final da coleta era acrescentado mais um *device* e a cada acréscimo de equipamento era feita a coleta dos dados, cada configuração da rede é apresentado abaixo:

Conforme mostrado na seção 2.2.4 os tempo são atualizados automaticamente pelo mestre, nenhuma parâmetro foi ajustado manualmente.

- Configuração da rede com 1 *device*: transmissor de corrente FI303 e o Coupler DP/PA da Siemens CLP S7-300.

Os tempos da rede Profibus DP/PA configurados pelo mestre para essa montagem com 1 *device* estão apresentados na figura 62 a seguir.

Figura 62- Tempos da rede para 1 device configuração Rockwell



Fonte: Autor,2019

- Configuração da rede com 2 *devices*: o transmissor de temperatura TT303, transmissor de corrente FI303 e o Coupler DP/PA da Siemens CLP S7-300.

Os tempos da rede Profibus DP/PA configurados pelo mestre para essa montagem com 2 *device* são semelhantes aos mostrado na figura 62 alterando somente o HSA = 8, e teve um aumento do TTR para 426,93 ms e no *Watchdog* para 39087 *Bit Time*.

- Configuração da rede com 3 *devices*: o transmissor de temperatura TT303, transmissor de corrente FI303, transmissor de temperatura 644 e o Coupler DP/PA da Siemens CLP S7-300.

Os tempos da rede Profibus DP/PA configurados pelo mestre para essa montagem com 3 *devices* são semelhantes aos mostrado na figura 62 alterando somente o HSA = 102, e teve um aumento do TTR para 465,92 ms e no *Watchdog* para 42723 *Bit Time*.

- Configuração da rede com 4 *devices*: transmissor de corrente FI303, o transmissor de temperatura TT303, o transmissor de temperatura 644, transmissor de pressão 2051 e o Coupler DP/PA da Siemens CLP S7-300.

Os tempos da rede Profibus DP/PA configurados pelo mestre para essa montagem com 2 *device* são semelhantes aos mostrado na figura 62 alterando somente o HSA = 102, e teve um aumento do TTR para 503,70 ms e no *Watchdog* para 45904 *Bit Time*.

Para o cenário 2 foram feitos os mesmos testes somente foi trocada CPU da Rockwell *Compact Logix E33ERM* pela CPU da SMAR DF95 e foi utilizado somente até 2 *device* na configuração da rede apresentados a seguir.

- Configuração da rede com 1 *device*: transmissor de temperatura TT303.

Os tempos da rede Profibus DP/PA configurados pelo mestre para essa montagem com 1 *device* estão apresentados na figura 63 a seguir.

Figura 63- Tempos da rede para 1 *device* configuração SMAR

Edit Bus Parameter			
Baud rate	500	kBits/s	
Slot Time	200	tBit	
Min. Station Delay of Responders	11	tBit	
Max. Station Delay of Responders	100	tBit	
Quiet Time	0	tBit	
Setup Time	1	tBit	
Target Rotation Time	50000	tBit	
Target Rotation Time	100.0000	ms	
GAP Actualization Factor	10		
Max Retry Limit	5		
Highest Station Address	126		
Tid1	37	tBit	
Tid2	100	tBit	
Poll Timeout	10	ms	
Data Control Time	30000	ms	
Min Slave Interval	25.000	ms	
Watchdog control	5000	ms	
Auto Clear	☒ Auto clear mode OFF ☐ Auto clear mode ON		

Fonte: Autor, 2019

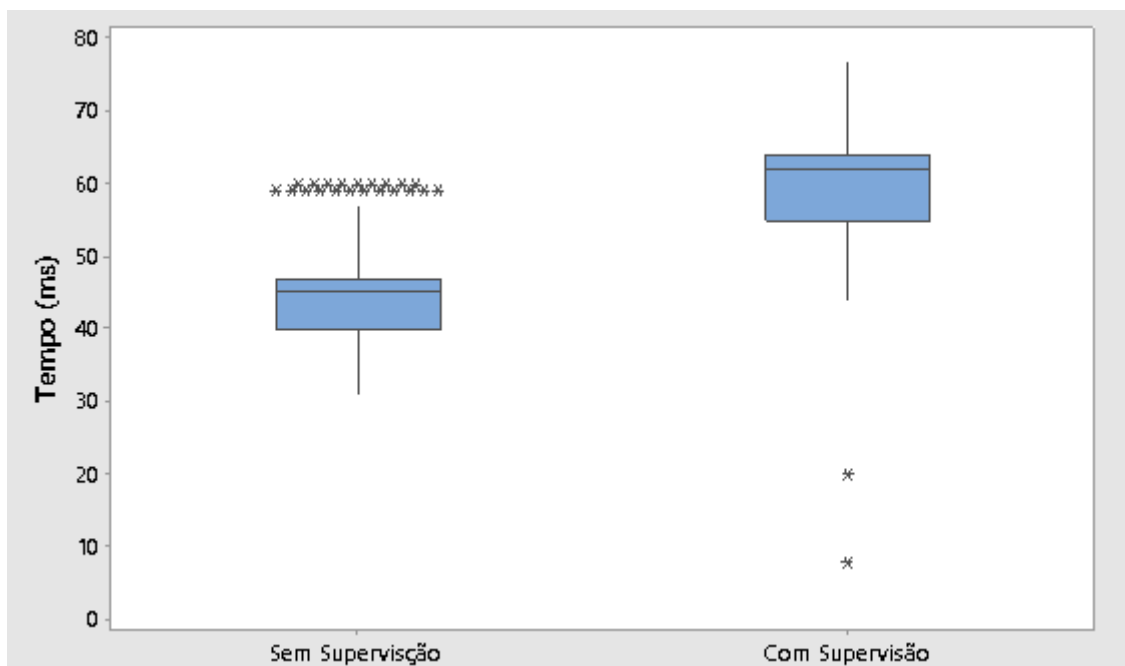
- Configuração da rede com 2 *devices*: transmissor de temperatura TT303 e transmissor de corrente FI303. O tempo da rede configurado pelo mestre é semelhante ao da figura 63.

Foram feitas 3 capturas para cada configuração de rede apresentada pelos dois cenários com cada captura contendo mais ou menos 600 frames.

Os dados foram tratados baseados no tempo de ciclo da rede e foram divididos em dois tempos, os que continham somente mensagens cíclicas e os que continham tanto mensagens cíclicas como também mensagens acíclicas, e com base nesses dois tempos foram montados os gráfico apresentados a seguir.

4.2.1 Sistema Rockwell com 1 *device*

Através dos frames capturados foi criado o gráfico apresentado abaixo que mostra a diferença de tempo da comunicação cíclica da comunicação acíclica.

Figura 64- Boxsplot dos dados da rede com 1 *device*

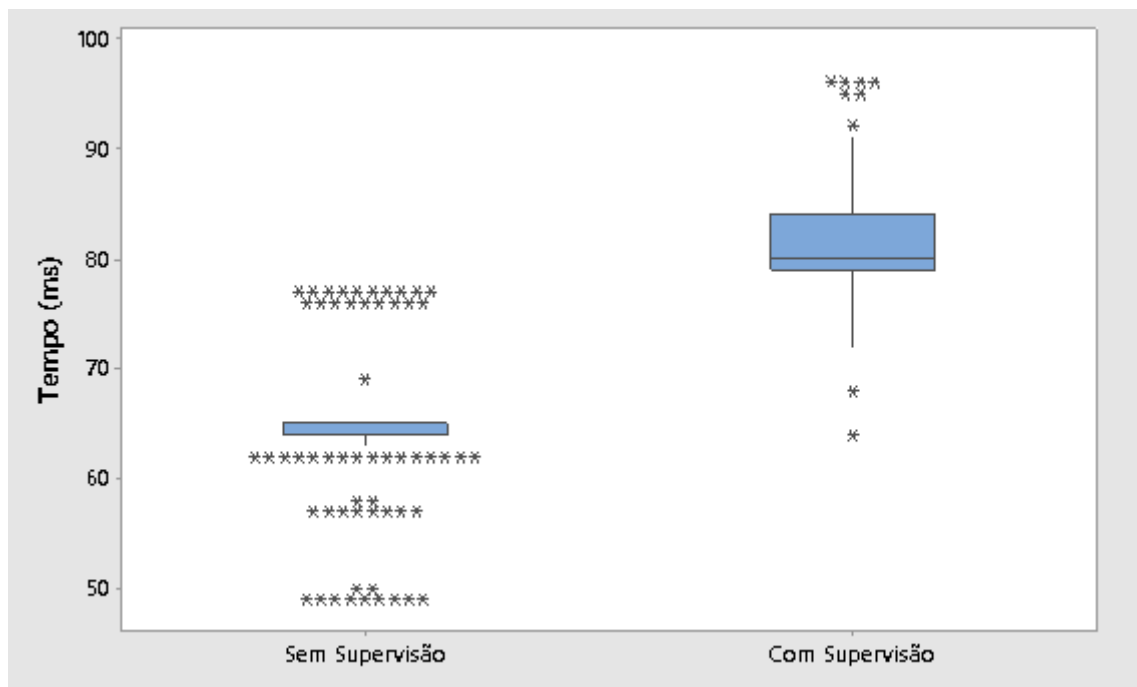
Fonte: Autor, 2019

A análise do boxsplot da figura 64 mostra que o período de *scan* da rede quando existe somente comunicação cíclica (Sem Supervisão) com somente um dispositivo em média de 42,351 ms com desvio padrão de 6,142. Quando existe comunicação acíclica (Com Supervisão) o período de *scan* da rede aumenta para em média de 59,327 ms e também apresenta pouca variação com desvio padrão de 6,893.

4.2.2 Sistema Rockwell com 2 *device*

Através dos frames capturados foi criado o gráfico apresentado abaixo que mostra a diferença de tempo da comunicação cíclica da comunicação acíclica.

Figura 65- Gráfico em forma de boxplot comparando o tempo de rede para 2 devices



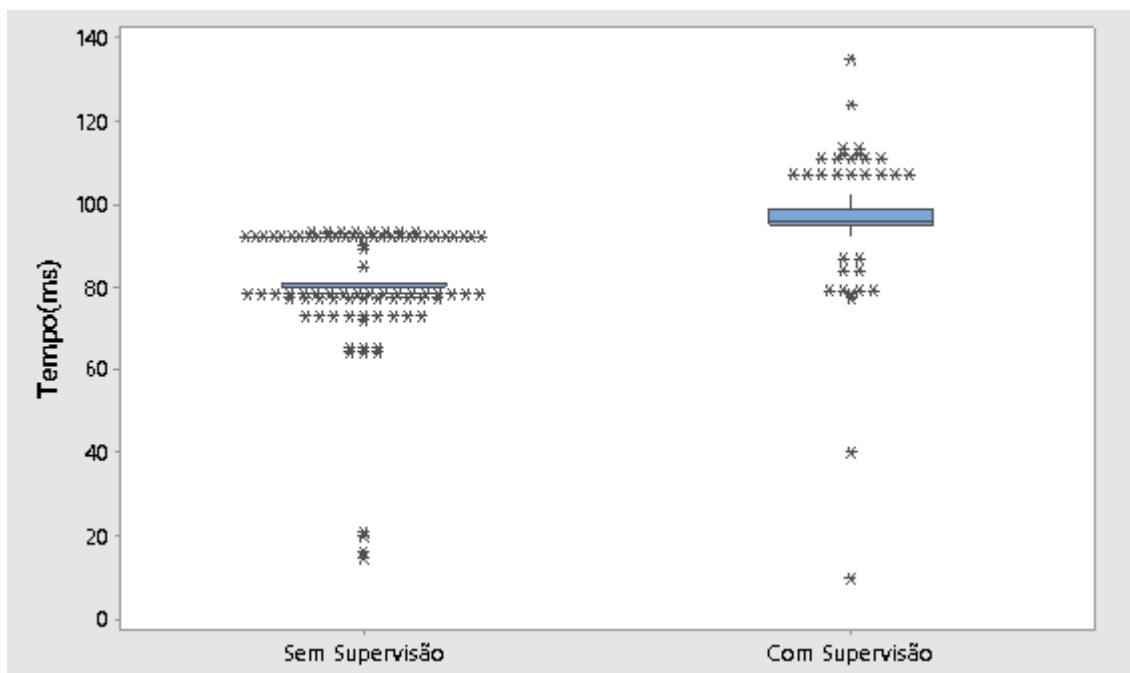
Fonte: Autor, 2019

A análise do boxplot da figura 65 mostra que o período de *scan* da rede quando existe somente comunicação cíclica (Sem Supervisão) com dois dispositivos em média de 64,527 ms com desvio padrão de 2,636. Quando existe comunicação acíclica (Com Supervisão) período de *scan* da rede aumenta para em média de 81,515 ms e também apresenta pouca variação com desvio padrão de 3,293.

4.2.3 Sistema Rockwell com 3 devices

Através dos frames capturados foi criado o gráfico apresentado abaixo que mostra a diferença de tempo da comunicação cíclica da comunicação acíclica.

Figura 66- Gráfico em forma de boxsplot comparando o tempo de rede para 3 devices



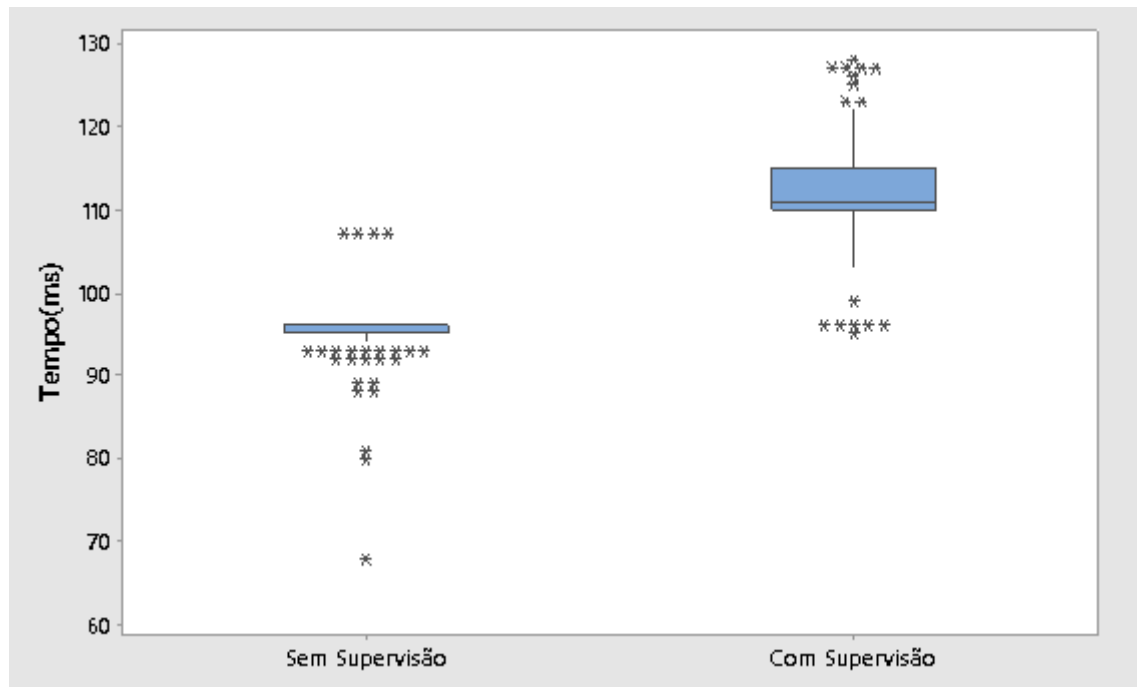
Fonte: Autor, 2019

A análise do boxsplot da figura 66 mostra que o período de *scan* da rede quando existe somente comunicação cíclica (Sem Supervisão) com três dispositivos em média de 80,208 ms com desvio padrão de 4,574. Quando existe comunicação acíclica (Com Supervisão) o período de *scan* da rede aumenta para em média de 96,976 ms e apresenta pouca variação com desvio padrão de 6,321.

4.2.4 Sistema Rockwell com 4 devices

Através dos frames capturados foi criado o gráfico apresentado abaixo que mostra a diferença de tempo da comunicação cíclica da comunicação acíclica.

Figura 67- Gráfico em forma de boxsplot comparando o tempo de rede para 4 devices



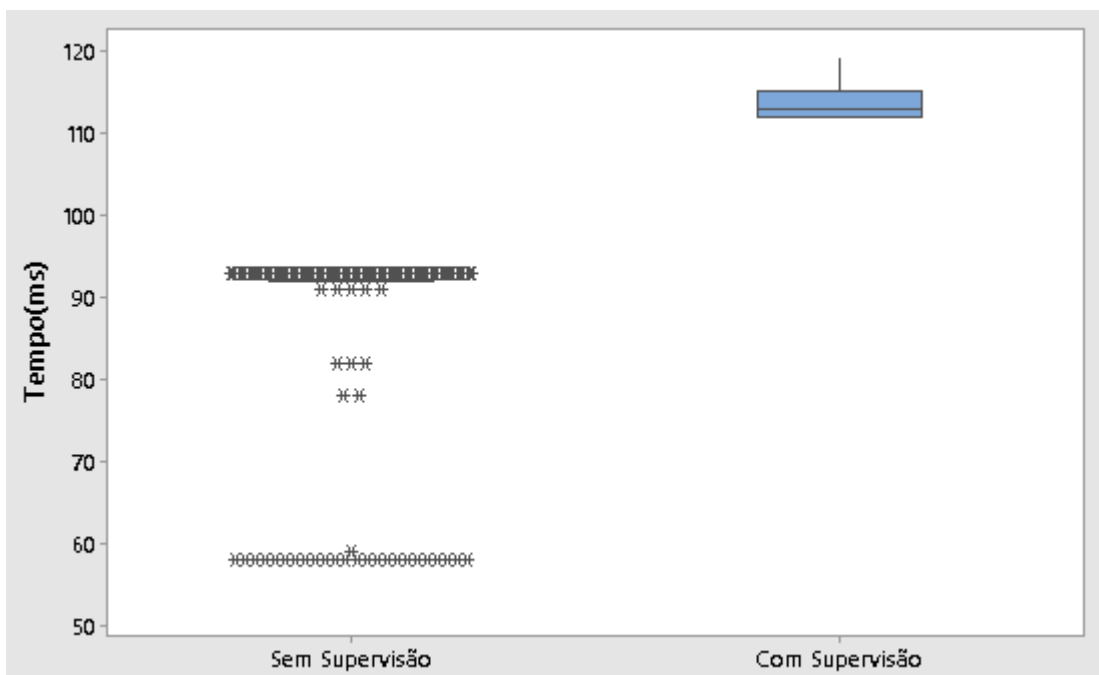
Fonte: Autor,2019

A análise do boxsplot da figura 67 mostra que o período de *scan* da rede quando existe somente comunicação cíclica (Sem Supervisão) com quatros dispositivos em média de 95,496 ms com desvio padrão de 2,370. Quando existe comunicação acíclica (Com Supervisão) o período de *scan* da rede aumenta para em média de 112,50 ms e também apresenta pouca variação com desvio padrão de 4,30.

4.2.5 Sistema SMAR com 1 device

Através dos frames capturados foi criado o gráfico apresentado abaixo que mostra a diferença de tempo da comunicação cíclica da comunicação acíclica.

Figura 69-Gráfico comparando a diferença de tempo da rede com supervisão e sem supervisão com 2 *devices*



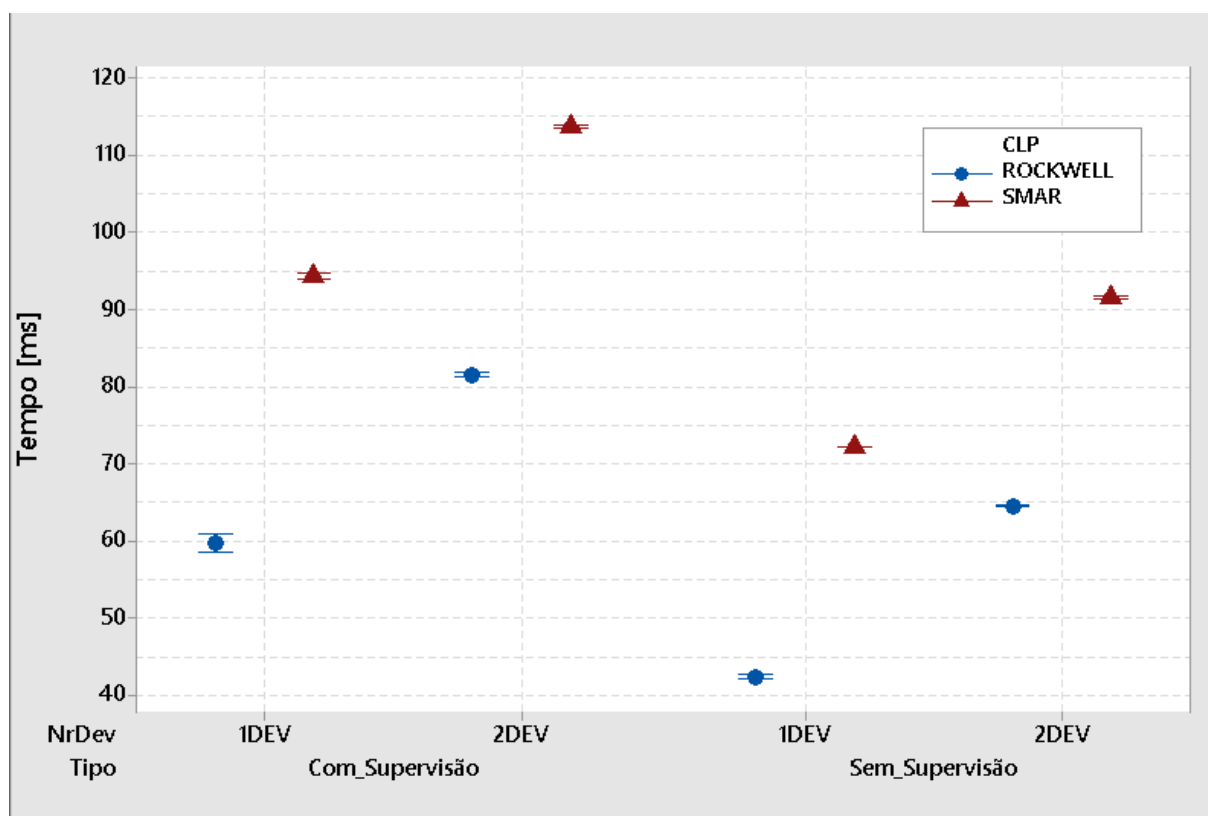
Fonte: Autor, 2019

Através do gráfico acima tem se a comparação de tempo de *scan* da rede com 2 *device*, no lado esquerdo apresenta o tempo sem supervisão ou seja somente com a comunicação cíclica que tem um tempo médio de 91,482 ms e apresenta pouca variação(drift) com desvio padrão de 4,249, já no lado direito apresenta o tempo de *scan* da rede com supervisão com os frame da comunicação acíclica que apresenta um tempo médio de 113,79 ms e apresenta também pouca variação com desvio padrão de 1,87.

4.2.7 Comparação entre Sistema SMAR e Sistema Rockwell

Como foram feitas coletas dos dados em dois cenários diferentes pode ser feito uma análise comparando o tempo de *scan* da rede ajustada pelos mestres de cada cenários, lembrando que nenhum parâmetro foi configurado manualmente, os tempos estão apresentados na figura 70. Essa análise mostra a diferença do tempo de TTR apresentado na seção 2.2.4 fornecida pelos mestres dos dois equipamentos.

Figura 70- Comparação do tempo de scan da rede no ambiente SMAR e ambiente Rockwell



Fonte: Autor,2019

O gráfico acima faz uma comparação no tempo de *scan* da rede entre os 2 sistemas diferentes para um *device* e dois *devices*. O sistema da SMAR apresenta um tempo maior no tempo de *scan* da rede tanto com supervisão quanto sem supervisão. Quando o sistema não tem supervisão o sistema Rockwell é de 28.4 ms em média mais rápido. Quando o sistema tem supervisão o sistema Rockwell é de 33.39 ms em média mais rápido.

Outra análise que pode ser feita em cima das ferramentas de manutenção PACTware e *Profibus View* é o tempo que cada ferramenta afeta na comunicação. No sistema Rockwell a diferença de tempo da rede sem o PACTware e com ele é de 17 ms, por outro lado no sistema da SMARA diferença do tempo da rede sem o *Profibus View* e com ele é de 22 ms.

4.3 Diagnóstico na Rede

Foram feitas capturas para análises sobre o diagnóstico na rede Profibus com o sistema Rockwell, os dados foram capturados retirando um equipamento da rede para observar o efeito que causaria em questão de tempo de *scan* da rede. Era esperado que o tempo de *scan* diminuísse quando um equipamento estivesse com falha, pois como foi explicado na seção 2.2.3

e 2.2.4 no capítulo 2, o Protocolo possui um mecanismo sofisticado de diagnóstico. Porém não foi observado isso na análise das figuras 71 e 72.

Figura 71-Captura da rede quando ocorreu uma falha no device com endereço 102

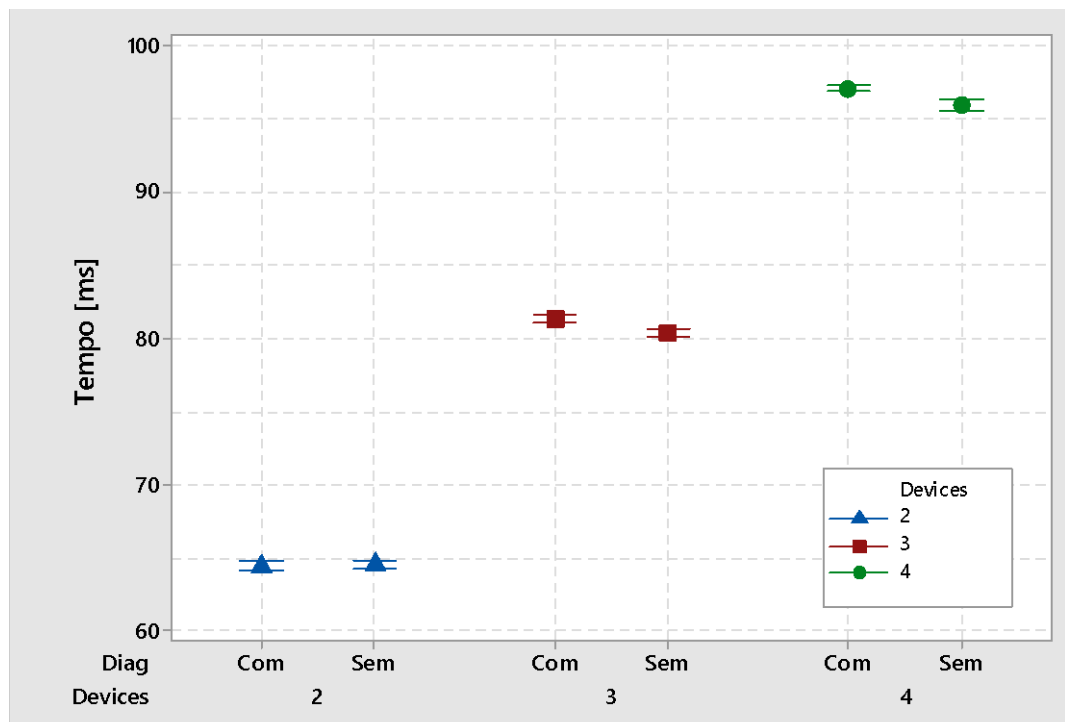
FrameNr	Timestamp	Frame	Addr	Msg type	Req/Res	SAPS	DataLen	Data	FC
2134	7-Oct-2019 12:01:28.670	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2140	7-Oct-2019 12:01:28.736	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2146	7-Oct-2019 12:01:28.802	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2152	7-Oct-2019 12:01:28.868	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2158	7-Oct-2019 12:01:28.934	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2164	7-Oct-2019 12:01:29.000	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2170	7-Oct-2019 12:01:29.066	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2176	7-Oct-2019 12:01:29.132	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2182	7-Oct-2019 12:01:29.198	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2188	7-Oct-2019 12:01:29.264	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2194	7-Oct-2019 12:01:29.331	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2200	7-Oct-2019 12:01:29.397	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2206	7-Oct-2019 12:01:29.463	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2212	7-Oct-2019 12:01:29.529	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2218	7-Oct-2019 12:01:29.595	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2224	7-Oct-2019 12:01:29.661	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2230	7-Oct-2019 12:01:29.727	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2236	7-Oct-2019 12:01:29.793	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2243	7-Oct-2019 12:01:29.860	SD2	1->102	Get Diagnostics	Req	62->60	0		6D
2244	7-Oct-2019 12:01:29.869	SD2	1<-102	Get Diagnostics	Res	62<-60	6	02 05 00 FF 08 9A	08
2251	7-Oct-2019 12:01:29.926	SD2	1->102	Set Parameters	Req	62->61	7	88 56 01 0B 08 9A 00	5D
2259	7-Oct-2019 12:01:29.991	SD2	1->102	Check Confid	Req	62->62	5	42 84 08 05 00	7D
2265	7-Oct-2019 12:01:30.039	SD2	1->127	Global control	Req	62->58	2	00 00	46
2268	7-Oct-2019 12:01:30.069	SD2	1->102	Get Diagnostics	Req	62->60	0		5D
2269	7-Oct-2019 12:01:30.078	SD2	1<-102	Get Diagnostics	Res	62<-60	6	00 0C 00 01 08 9A	08
2276	7-Oct-2019 12:01:30.138	SD1	1->102	Data Exchange	Req	10		00 00 00 00 1F 00 00 00 00 1F	7D
2277	7-Oct-2019 12:01:30.146	SD2	1<-102	Data Exchange	Res	10			08
2284	7-Oct-2019 12:01:30.208	SD2	1->102	Set Parameters	Req	62->61	7	C8 56 01 0B 08 9A 00	6D
2292	7-Oct-2019 12:01:30.277	SD2	1->102	Get Diagnostics	Req	62->60	0		5D
2293	7-Oct-2019 12:01:30.286	SD2	1<-102	Get Diagnostics	Res	62<-60	6	02 05 00 FF 08 9A	08
2300	7-Oct-2019 12:01:30.347	SD2	1->102	Set Parameters	Req	62->61	7	88 56 01 0B 08 9A 00	7D
2308	7-Oct-2019 12:01:30.414	SD2	1->102	Check Confid	Req	62->62	5	42 84 08 05 00	5D
2316	7-Oct-2019 12:01:30.481	SD2	1->102	Get Diagnostics	Req	62->60	0		7D

Fonte: Autor, 2019

De acordo com a figura 71, quando ocorre a falha apesar de já ter decorrido algum tempo o mestre sempre vai enviar mensagem de pedido de diagnostico (Sap 62->60) para o escravo, mesmo o *device* 102 não estando na rede, como é possível notar nos frames 2134 até o frame 2243 da figura 71. E somente quando o *device* responder um diagnóstico de OK (no frame 2293) é que para de se fazer o pedido de diagnostico. E é possível notar ainda na figura que a taxa de pedido de diagnóstico é de aproximadamente 66ms. E não foi possível detectar um parâmetro que fizesse com que esta taxa de pedido de diagnostico fosse mais lenta.

A figura 72 mostra uma análise da influência da falha no ciclo de scan da rede. Foram feitos testes variando o número de device em 2,3 e 4 e analisando a comunicação quando não tem falha (sem diagnostico) e fazendo somente a falha em 1 device (com diagnostico).

Figura 72-Comparação do tempo de scan da rede com diagnóstico e sem diagnóstico com 2, 3e 4 devices



Fonte: Autor,2019

De acordo com a figura 72, quando a rede está operando normalmente (sem diagnóstico) com 2 *devices*, o tempo de *scan* medido foi de 64,55ms com desvio de 2.78, enquanto que quando foi desligado um *device* (com diagnóstico) o tempo de *scan* medido foi de 64,40ms com desvio padrão de 3.608. Para 3 *devices* sem diagnóstico obteve tempo de *scan* 80.415 com desvio de 2.961 enquanto que com diagnóstico obteve tempo de *scan* de 81.415 ms com desvio de 2.926. Por fim, o tempo de *scan* para 4 *devices* sem diagnóstico foi de 96.017 com desvio de 3.858 enquanto que com falha de 1 *device* (com diagnóstico) foi de 97,164 com desvio de 2.441. Resumindo, os tempos com e sem diagnóstico foram praticamente os mesmos quando ocorreu a falha de 1 *device* na rede.

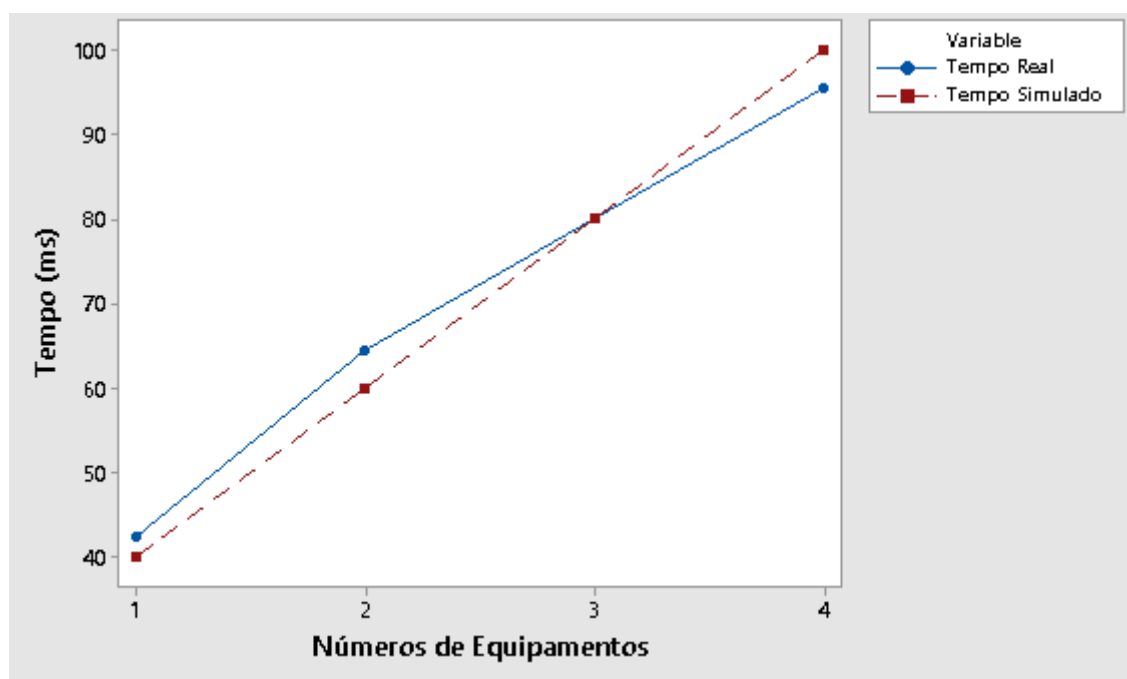
4.4 Simulação de tempo da rede

Analisando os dados obtidos foi observado uma certo determinismo no tempo da rede com o aumento de equipamentos, com isso foi elaborado uma equação para estimar o tempo de *scan* da rede com o acréscimo de escravos apresentada na equação 1 a seguir.

$$T_{scan} = 20 + x.20 \quad (1)$$

Onde, T_{scan} representa o tempo de *scan* estimado com uma variação de 5ms e x o números de equipamentos. Desta forma, podemos prever que uma rede com 50 devices teria um tempo de scan na ordem de 1020ms e para uma rede completa com 126 devices o tempo de scan em média de 2540ms. Afigura 74 apresenta a curva obtida através da equação em comparação com a curva de tempo real.

Figura 73-Gráfico comparando o tempo real pelo tempo simulado



Fonte: Autor,2019

Através da análise do gráfico pode se perceber que o tempo estimado pela equação é bem próximo do tempo real com uma pequena variação apresenta na equação 1.

4.5 Discussões

Um grande desafio encontrado nesse projeto foi em relação ao funcionamento dos DTMs dos equipamentos instalados, eles não apresentavam um bom funcionamento devido ao conflito de versões com alguns arquivos já existente no computador, corrigido esse erro eles funcionaram corretamente e foi possível acessar os parâmetros dos equipamentos.

Ainda em relação ao PACTware foi notado que quando era conectado três equipamentos do mesmo fabricante ou equipamentos de fabricantes diferentes no PACTware a ferramenta começava a apresentar mal funcionamento para fazer ajustes nos transmissores. Então sempre que era realizados os testes era conectado somente um ou dois equipamentos do mesmo fabricante, e quando fosse necessário conectar um terceiro equipamento era necessário desconectar os outros, então para que o PACTware funcionasse bem era conectado um equipamento na ferramenta por vez.

Em relação aos resultados obtidos através das análises dos tempos de *scan* da rede foi observado que existe um aumento no tempo de *scan* da rede quando se tem mensagens acíclicas devido a ferramentas de manutenção. Porém mesmo com um aumento no nível de informação trafegada não é afetado o tempo de *scan*, mostrando que na rede configurada pelos mestres existe uma folga entre cada ciclo de *scan* da rede.

Os *outliers* que apresentam acima dos gráficos ocorrem quando os ciclos de *scan* da rede contém as mensagens de Global_Control enviada pelo mestre. Quando são enviados estas mensagens de broadcast, o tempo de ciclo da rede apresenta um aumento. E os *outliers* que estão abaixo do gráfico ocorrem quando existe mensagens FDL_STATUS para os endereços dos equipamentos que estão na rede. Eles respondem a esse comando com uma mensagem de passivo, quando ocorre essas mensagens os ciclos apresentam um tempo menor.

Como os testes foram feitos em dois ambientes diferentes, pode-se observar que a CPU da SMAR DF95 apresenta um tempo maior de *scan* da rede em relação a CPU da Rockwell Compact Logix. Um motivo que pode causar essa diferença é o fato que o ambiente da Rockwell possui duas CPUs independentes: o controlador L33ERM que é responsável pelo controle AI-PID-AO e o cartão Prosoft que é responsável por ler e traduzir as informações dos pontos AIs e AOs dos equipamentos para o L33ERM. Por outro lado, o controlador DF95 possui mais tarefas em relação ao CLP da Rockwell como por exemplo o tratamento dos protocolos HSE, Modbus e Profibus DP e PA além do tratamento dos cartões de IO local. Então isso pode levar a um aumento no tempo da rede.

5 CONCLUSÃO

O objetivo principal deste trabalho foi o estudo e a análise da comunicação acíclica dentro de uma rede Profibus PA. Também foi a capacitação do laboratório com ferramentas de supervisão e manutenção que ainda não tinha sido testada nos cenários apresentados neste trabalho.

Em relação ao estudo de ferramentas de padrão aberto de manutenção, a ferramenta PACTware possibilitou fazer a configuração e manutenção de todos os equipamentos de campo do laboratório remotamente, onde até então, era possível somente fazer a configuração localmente, através de uma chave magnética.

Em relação aos estudos do Protocolo Profibus PA, foram realizadas análises do desempenho da comunicação acíclica da rede Profibus auxiliada pelo PACTware e o *ProfibusView* que representava o sistema de supervisão da planta e confrontada com a comunicação cíclica feito pelo controle da planta representada na lógica construída no PLC da Rockwell e SMAR. Neste caso, foi possível aprender em detalhes como analisar mais profundamente uma rede de comunicação usando ferramentas de escuta de linha.

Através das ferramentas de escuta de linha observou que há uma diferença de tempo de *scan* da rede entre as duas comunicação mostrando que com a adição da supervisão acarretou um aumento de tempo de *scan* da rede, e mesmo a nossa rede possuindo uma folga entre cada ciclo de *scan* para as mensagens acíclicas nunca é enviado mais de uma mensagem acíclica por ciclo, o que fazem com que as mensagens acíclicas sejam mandadas em ciclos diferentes, concluindo que mesmo com um aumento de nível de informação trafegada na rede não é afetado muito o ciclo de *scan* pois nunca vão existir uma “avalanche” de mensagens acíclicas pois sempre existira somente uma mensagem enviada por ciclo de *scan*. E no caso de considerar a supervisão somente do sistema SCADA, ela não afeta o tempo de ciclo, pois no Profibus existe a centralização de dados no controlador e desta forma somente poderá ser supervisionado os dados que estão no controlador e isso não afeta a comunicação cíclica da rede.

A partir dos gráficos das figuras 64 à 69 é possível analisar o determinismo da rede Profibus, ou seja a rede possui aproximadamente o mesmo tempo de *scan* durante toda comunicação cíclica.

Comparando os 2 sistemas montados neste projeto pode se observar também que o CLP DF95 da SMAR apresenta um tempo de *scan* da rede maior que o do CLP Compact Logix da

Rockwell. Mostrando que equipamentos diferentes trazem tempos de ciclo também diferentes, devido as características de hardware e software de cada equipamento.

Em relação aos resultados sobre os diagnósticos na rede não foram os esperados. Como o Profibus é um protocolo moderno, era esperado que o tempo da rede diminuísse quando ocorresse falha de um equipamento o que não acontece o tempo continua praticamente igual, isso ocorre pelo fato de o mestre mandar um *Get diagnostics* todo ciclo para o escravo com falha até que ele volte a comunicar na rede.

Como o cenário no ambiente Rockwell foi configurado para 1,2,3 e 4 equipamentos a cada adição de um equipamento pode ser notado um determinismo no aumento do tempo na rede. Depois dessa análise foi elaborado uma equação que estimasse o tempo de *scan* da, que aproximou bastante do real possibilitando prever o tempo de *scan* da rede com vários equipamentos.

Por fim, este trabalho permitiu o estudo e um conhecimento maior sobre o Protocolo Profibus. Possibilitou o estudo de ferramentas de manutenção com padrão aberto para agregar ao laboratório de automação, para que fosse possível fazer a manutenção remotamente de todos transmissores Profibus PA do laboratório o que antes era feito através da chave magnética, e através dessas ferramentas foi adquirido um conhecimento maior sobre o funcionamento dos blocos definido no perfil de aplicação do Profibus PA para os transmissores.

E com a ajuda de ferramentas de escuta de linha pode ter um conhecimento maior sobre a comunicação da rede Profibus como os tempos, característica das mensagens e frames, e pode ser observado o impacto causado na rede devido ao excesso de supervisão na rede.

E para trabalhos futuros é possível fazer uma análise comparando os tempos de configuração da rede Profibus feitos pelos mestres de diferentes fabricantes e qual seria a influência com o aumento no nível de informações trafegadas na rede. E pode ser feito um estudo em uma rede mais complexa com um número maior de processos, equipamentos, mestres e estações de supervisão.

6 Referências Bibliográficas

BRANDÃO, D; SESTILO, G, S (2016). *et al* Curso de Engenheiro Certificado Profibus. PI Brasil (Profibus-Profinet) , 2016. Universidade Federal de São Paulo, 2016.

BERGE, J. (2002). *Fieldbuses for Process Control: Engineering, Operation and Maintenance*. ISA – The Instrumentation, Systems and Automation Society, 2002.

CARRIJO, R. (2011). Gerenciamento de ativos aplicado a instrumentos de campo de protocolos abertos: uma abordagem a partir de dispositivos móveis. São Carlos,2011.

CASTRUCCI, P.; MORAES, C. C(2015). *Engenharia de Automação Industrial*. 2ª Edição. São Paulo: LTC, 2007. 358 p *apud* Marchioro Bertelli, G. P. Sistema de Controle e Supervisão da Estação de Processos MPS PA via Plataforma BeagleBone. 2015. 62 f. Trabalho de Conclusão de Curso (Especialização em Engenharia Mecatrônica) – Universidade Federal do Rio Grande do Norte, 2015.

CASSIOLATO, C.; TORRES, L. H. B.; CAMARGO, P. R. (2012). *Profibus – Descrição Técnica*. São Paulo: Associação Profibus Brasil, 2012.

CASSIOLATO, C; PADOVAN, M, A; TORRES, H, B, L; OLIVEIRA(2012), A. *et al* Descrição Técnica Profibus. Associação Profibus Brasil América Latina,2012. P97.

CASSIOLATO, C; PADAVON, M, A; TORRES, H, B, L; RIBEIRO, M, C; OLIVEIRA, A. *et al* Profibus-Descrição Técnica 2009. Associação Profibus Brasil América Latina,2009. P85.

CASSIOLATO, C. Medição de Nível, 2010. Disponível em:
<http://www.smar.com/newsletter/marketing/index39.html>Acesso em: 09/03/2019.

DATA CENTER DYNAMICS (2017) “Convergência entre tecnologia da informação e tecnologia operacional é necessária para garantir a continuidade do negócio “, Disponível em:
<http://www.datacenterdynamics.com.br>. Acesso: 26/jul/2018.

DA SILVA, A. P.; SALVADOR, M.(2004). O que são sistemas supervisórios. Disponível em: http://www.wectrus.com.br/artigos/sist_superv.pdf Acesso em: 18 de Fevereiro de 2019.

EMBARCADOS. Página Institucional. Disponível em: <https://www.embarcados.com.br/redes-de-comunicacao-em-rs-485/> Acesso em: 25/02/2019.

FDI Cooperation (2016), Digital Transformation in the age of IIOT, disponível on: <https://fieldcommgroup.org/technologies/fdi/fdi-documents-and-downloads>. Acesso em: 15/08/2019.

HOLLENDER, M. (2010). Collaborative Process Automation Systems. ResearchTriangle Park: ISA Books, 2010

JURIZATO, L.A (2009). Sistemas Supervisórios. Disponível em: <http://www.centralmat.com.br/artigos/mais/sistemassupervisorios.pdf> Acesso em: 18 de Fevereiro de 2019.

J. Kjellsson, A. E. Vallestad, R. Steigmann and D. Dzung, "Integration of a Wireless I/O Interface for PROFIBUS and PROFINET for Factory Automation," in *IEEE Transactions on Industrial Electronics*, vol. 56, no. 10, pp. 4279-4287, Oct. 2009. doi: 10.1109/TIE.2009.2017098

KUMAR, S. S. Fieldbus and Networking in Process Automation, CRC Press, 2014.

MORAES, C; CASTRUCCI, P (2007). Engenharia de Automação Industrial,2007.

NETO, C. D. (2008). O Profissional de Automação com Nível Superior. III Fórum Internacional de Automação do Setor Sucroalcooleiro e Alimentício. 2008.

OLIVEIRA, A (2010). Modelo de predição para análise comparativa de técnicas neuro-fuzzy e de regressão, 2010. 34f. Programa de Pós Graduação em Informática- Centro Tecnológico, Universidade Federal do Espírito Santo, 2010.

OPC FOUNDATION (2010). Especificação OPC. Disponível em:

<<https://opcfoundation.org>>. Acesso em 06/10/2019.

PANTOJA ENGINEERING & CONSULTANT. Página Institucional. Disponível em:

<<http://www.pantojaindustrial.com/exibir.php?id=48>> Acesso em : 05/03/2019.

PINHEIRO, J. M. (2006). Introdução as Redes de Supervisão e Controle, 2006.

PESSA, R (2006). Instrumentação Básica para Controle de Processo, 2006, p.233. Centro de Treinamento Smar. Disponível em: <www.smar.com.br> Acesso em : 18/10/2019

Profibus Installation Guideline for Cabling and Assembly. Version 1.0.6. May 2006.

PROFIBUS (1999). PROFIBUS-PA Profile for Process Control Devices. Data Sheet Transmitter V3.0, September 15th, 1999

SMAR(2018). Transmissor de Pressão Profibus PA. Disponível em:

< www.smar.com/PDFs/Manuals/LD303MP.PDF> Acesso em: 11/03/2019

_____.(2009). Manual dos Procedimentos de Instalação, Operação e Manutenção Profibus PA. Versão 2. Disponível em: <www.smar.com.br> Acesso em: 04/10/2019

_____.(2012). Manual de Instruções dos Blocos Funcionais. Versão2. Disponível em: <www.smar.com.br> Acesso em: 09/11/2019

_____.(2018). Manual dos Procedimentos de Instalação, Operação e Manutenção. Transmissor de Temperatura. Versão 3. Disponível em: <www.smar.com.br> Acesso em: 09/11/2019

STALLINGS, W. (1997). Data and Computer Communications. Prentice Hall, 1997.

STEMMER, M. R. (2001). Sistemas Distribuídos e Redes de Computadores para Controle e Automação Industrial. 2001. Santa Catarina: Universidade Federal de Santa Catarina.

THEURICH, S.; LEHMANN, R.; WOLLSCHLAEGER, M. (2010) .Network Asset Models in Intelligent FieldDevices. In: Factory Communication Systems (WFCS), 2010 8th IEEE International Workshopon, pp. 161-164, 18-21 May 2010

VIVACE (2016). Manual de instalação, operação e configuração. Disponível em: <http://www.vivaceinstruments.com.br/admin/public/img/manual-profibus-pa.pdf> Acesso em: 18 de Fevereiro de 2019.

YAQOOB, I., AHMED, E.; HASHEM, I. A. T.; AHMED, A. I. A.; GANI, A.; IMRAN, M.; GUIZANI, M. “Internet of Things Architecture: Recent Advances, Taxonomy, Requirements, and Open Challenges.” *IEEE Wireless Communications* 24 (6 2017): 10-16.

Apêndice A

Este Programa faz a decodificacao de uma captura Profibus PA feita pelo programa FBView da Smar.

Ele utiliza o arquivo de entrada -i fileinput que eh uma captura salva pelo FBview no formato texto.

E ele decodifica os frames (baseado no formato do Profitrace) salvando em formato CSV para ser importado pelo excel.

como usar : python FbviewParser.py -i fileinput.txt -o fileoutput.csv

```
import base64
```

```
import time
```

```
import sys
```

```
import datetime
```

```
import struct
```

```
import math
```

```
import os
```

```
import copy
```

```
import argparse
```

```
INFILE = 'fbview2.txt';
```

```
OUTFILE = 'fbview2o.csv';
```

```
def FbviewParser(namefilein,namefileout):
```

```
    idx = {}
```

```
    timestamp = {}
```

```
    desc = {}
```

```
    src = {}
```

```
    dst = {}
```

```
    dsap = 0
```

```
    ssap = 0
```

```
    fc = 0
```

```
    lendata = 0
```

```
    filerd = 0;
```

```

filerd = open(namefilein, 'r');
filewr = open(namefileout, 'w');

#grava o header no arquivo de saida
buf1 = "idx ; timestamp ; src ; dst ; fc; dsap ; ssap ; lendata; desc ; payload \n"
filewr.write(buf1)

file1 = filerd.readlines();
try:
    for x in file1:
        idx = x[0:6]
        timestamp = x[18:30]
        payload = x[30:]
        params = payload.split();
        lenpay = 0
        lendata = 0
        if (params[0] == '10'):
            src = int(params[2],16)
            dst = int(params[1],16)
            fc = int(params[3],16)
            if (fc == 73):
                desc = "FDL Status"
            else:
                desc = "Data Exchange Req"

        elif (params[0] == 'DC') or (params[0] == '01'):
            desc = "Pass Token"
            src = 0 # int(params[2],16)
            dst = 0 # int(params[1],16)
        elif (len(params) > 4) and (params[0] == '68'):
            lenpay = int(params[1],16)
            src = int(params[5],16)
            dst = int(params[4],16)

```

```

    fc = int(params[6],16)
    dsap = int(params[7],16)
    ssap = int(params[8],16)
    if (ssap == 62) and (dsap == 58):
        desc = "Global Control"
        lendata = 2
    elif (ssap == 50) and (dsap == 48):
        desc = "DPV1 Read Req"
        lendata = lenpay - 5
    elif (ssap == 48) and (dsap == 50):
        desc = "DPV1 Read Res"
        lendata = lenpay - 5
    else:
        desc = "Data Exchange"
        lendata = lenpay - 3
        dsap = 0
        ssap = 0

    elif (len(params) < 5) and (params[0] == 'E5'):
        desc = "SHORT ACK"
        src = 0
        dst = 0
    else:
        desc = "Other"
        src = 0
        dst = 0

    buf1 = "{0} ; {1} ; {2} ; {3} ; {4} ; {5} ; {6} ; {7} ; {8} ;
{9}".format(idx,timestamp,src,dst,fc, dsap,ssap,lendata,desc,payload)
    #print buf1
    filewr.write(buf1)

except Exception, e:
    print repr(e)

```

```

        print "Arq leitura:" + namefilein + " Arq. Escrita:" + namefileout + " Nr registros:" +
str(len(file1))
        filerd.close()
        filewr.close()

def check_file(fn):
    if not os.path.isfile(fn):
        print "Invalid file {0}".format(fn)
        return False
    else:
        return True

if __name__ == '__main__':

    parser = argparse.ArgumentParser()
    parser.add_argument('-i', '--input', default=INFILE, help='File Input .txt')
    parser.add_argument('-o', '--output', default=OUTFILE, help='File Output .csv')

    args = parser.parse_args()

    if not check_file(args.input):
        sys.exit(1)

    #if check_file(args.output):
    #    sys.exit(1)

    FbviewParser( args.input, args.output);

```