



UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

LUÍS L'AIGLON PINTO MARTINS

**VIRTUALIZAÇÃO DE SISTEMAS DE PROTEÇÃO E CONTROLE DE SUBESTA-
ÇÃO COM UTILIZAÇÃO DE MERGING UNIT E IED VIRTUAL INTEGRADO AO
SAGE DE SUBESTAÇÃO DA REDE BÁSICA**

FORTALEZA

2023

LUÍS L'AIGLON PINTO MARTINS

VIRTUALIZAÇÃO DE SISTEMAS DE PROTEÇÃO E CONTROLE DE SUBESTAÇÃO
COM UTILIZAÇÃO DE MERGING UNIT E IED VIRTUAL INTEGRADO AO SAGE DE
SUBESTAÇÃO DA REDE BÁSICA

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Ceará, como requisito parcial à obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: Sistemas de Energia Elétrica.

Orientadora: Prof^ª. Ph.D. Ruth Pastôra Saraiva Leão.

Coorientador: Prof. Dr. Raimundo Furtado Sampaio.

Fortaleza

2023

LUÍS L' AIGLON PINTO MARTINS

VIRTUALIZAÇÃO DE SISTEMAS DE PROTEÇÃO E CONTROLE DE SUBESTAÇÃO
COM UTILIZAÇÃO DE MERGING UNIT E IED VIRTUAL INTEGRADO AO SAGE

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Ceará, como requisito parcial à obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: Sistemas de Energia Elétrica.

Aprovada em: 22/12/2023.

BANCA EXAMINADORA

Prof.^a Dra. Ruth Pastôra Saraiva Leão (orientadora)
Universidade Federal do Ceará (UFC)

Prof. Dr. Raimundo Furtado Sampaio (coorientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. Lucas Silveira Melo
Universidade Federal do Ceará (UFC)

Prof. Dr. Roberto Feliciano Dias Filho
Universidade de Pernambuco (UPE)

A minha querida esposa Erika e as minhas pequenas Alanis e Alícia.

AGRADECIMENTOS

Ao Grande Arquiteto do Universo que me deu o dom da vida, e iluminou meus pensamentos para que conseguisse chegar no final desse caminho.

Aos meus pais, Luís Ribeiro (*in memoriam*) e Hilda Martins (*in memoriam*), que nunca mediram esforços e sempre me apoiaram em todos os sonhos que tive, tornaram-me a pessoa que sou hoje, com humildade sempre, para trilhar um caminho de conquistas. A vocês toda minha gratidão, meu respeito e muita saudade.

As minhas irmãs, Luiza (*in memoriam*), Luzilda Antônia, Lusinilda Carla, Luilda Maura, Lussilda Cláudia e Luisilda Maria, que sempre estiveram ali incentivando desde as primeiras palavras escritas na infância.

Aos meus avós (*in memoriam*), dos quais lembro dos ensinamentos e da dedicação de querer sempre a família junta, principalmente nas festas dos meses de março, na querida localidade de Amanaiara.

A minha família que, direta ou indiretamente, estava a ouvir os anseios dessa jornada discente, nos nossos encontros e nas palavras de incentivo.

As minhas filhas Alícia Martins e Alanis Martins cujo amor que tenho por elas usei como pilar para me dedicar ao trabalho. A compreensão delas sobre minha ausência em alguns momentos foi essencial para me manter firme no propósito.

A minha amada esposa Erika Barreto que, dedicada à nossa família e a mim, sustentou a nós com seu amor, carinho e sua motivação para lograrmos sucesso nos desafios. Apoiou-me incansavelmente, incentivando-me nos momentos de descontentamento, e terá sempre minha eterna admiração.

Aos meus amigos que me acompanharam na minha vida estudantil, e que de alguns tenho amizade até hoje. Da base forte das escolas: Educandário Casimiro de Abreu e Colégio 7 de Setembro, bem como a que me preparou para a vida acadêmica: Colégio Ari de Sá.

Não podia deixar de homenagear o amigo Daniel “San”, que me apresentou a eletrônica com algumas revistas e ensinamentos, aos onze anos. Fez com que despertasse e me aprofundasse, dando sentido na minha escolha da carreira que eu iria seguir, que era ser engenheiro eletricista.

Aos queridos amigos e amigas da Universidade Federal do Ceará, dos quais tenho muitas boas lembranças e muitas noites de sono mal dormidas, dedicadas aos estudos e aos

encontros estudantis. Não vou citar nomes, para não cometer falhas, mas agradeço a todos de coração.

Aos colegas do GREI, que me apoiaram nas ideias e na concretização do trabalho. Em especial quero destacar Professora Ruth Leão, de quem admiro a paixão pelo curso e pela satisfação em passar seus conhecimentos, desde a Graduação, sendo uma honra para mim tê-la como minha orientadora; compreendeu-me pelos momentos difíceis da minha carreira laboral, e me deu sempre o caminho certo para enfrentar os desafios. Ao Professor Raimundo Sampaio, que conheço desde os tempos de COELCE; admiro-o por sua dedicação e resiliência, que me trouxe alegria para seguir meus propósitos. Aos colegas que me apoiaram no projeto, em especial ao Matheus Souza, que espero vê-lo em breve como colega de profissão – sua dedicação e vontade de aprender me deixaram impressionado.

Aos professores e funcionários do DEE da UFC – tenho o exemplo de muitos que passaram por mim e me tornaram o profissional que sou hoje, tanto da graduação, como das pós-graduações que fiz. Vou saudá-los sempre que os ver, pois a diferença que fez na minha vida e na de vários amigos foi essencial para construção de nosso caráter.

Aos amigos da saudosa COELCE (atual ENEL CE), em destaque a equipe de projetos de subestações e a equipe de comissionamento, na qual tive oportunidade de desbravar o Ceará. Quero destacar meus antigos líderes: Roberto Freire e Gilson Alves, que fez abrir as portas do conhecimento, para que futuramente eu fosse para CHESF, onde não poderia esquecer dos meus amigos da Sede em Recife, e da Regional de Fortaleza. Nessa regional, tive o apoio e a permissão para montagem da bancada do trabalho pelo meu amigo e irmão Temóstenes Nunes. Tive um grande apoio da equipe de técnicos e engenheiros de manutenção e controle de MPCCS, da qual destaco Dalécio e Tadeu, que doaram muito do seu tempo e da sua dedicação.

Ao meu amigo Dennis Lellys, que me incentivou e me ajudou com o fornecimento do equipamento que foi o baluarte desse trabalho, além de todo cabedal de conhecimento que pude receber dele nos últimos anos.

Enfim, peço perdão aos que não nomeiei e que me ajudaram de alguma maneira nos desafios e na concretização desse sonho. Tenham a certeza de que serão lembrados e sempre terão a minha eterna gratidão.

“O homem não é nada além daquilo que a educação faz dele.” (Immanuel Kant)

RESUMO

A norma IEC 61850 e suas atualizações causam um impacto expressivo nos sistemas de proteção, automação e controle (PAC) das subestações ao redor do mundo. Novos avanços são implementados e testados a cada dia, como a virtualização dos dispositivos eletrônicos inteligentes (IED, sigla em inglês), objeto de estudo deste trabalho. A virtualização de IEDs advém da necessidade de reduzir custos de manutenção e substituição de *hardware*, otimizar produtividade e dar suporte à recuperação de problemas com IEDs físicos nas instalações. O objetivo deste trabalho é aplicar IED virtuais (VIEDs), aprimorando os que foram desenvolvidos no Grupo de Redes Elétricas Inteligentes (GREI) da Universidade Federal do Ceará (UFC), no ambiente de plataforma de testes que simulam uma subestação integrante da Rede Básica. Foi testada a performance de VIEDs na rede de barramento de processos, com o tráfego das informações digitalizadas por uma *Merging Unit* (MU), e no barramento de estação, com a troca de mensagens entre o *software* supervisor da Interface Homem-Máquina (IHM) e os *switches* gerenciáveis. Para os testes e a implantação dos VIEDs, foi montada uma plataforma PAC no laboratório estruturado na Companhia Hidroelétrica do São Francisco (CHESF), em Fortaleza-Ceará, com IEDs físicos, MU, caixa de testes hexafásica para relés de proteção, Interface Homem-Máquina (IHM) com o Sistema Aberto de Gerenciamento de Energia (SAGE) e *notebooks* executando as aplicações do VIED e SAGE no ambiente operacional Linux. Na plataforma, foram testadas as respostas do VIED sob situações de falta para as funções de proteção de sobrecorrente 50/50N (instantânea) e 51/51N/51V (temporizada), 67/67N (direcional) e 62BF, comparadas com IEDs físicos presentes na plataforma, com os mesmos ajustes de proteção do VIED. Foram testadas as comunicações IED-VIED usando protocolos GOOSE e *Sampled Values* (SV), e na comunicação VIED-SAGE usando protocolo MMS. Foi validado nesses testes o pleno funcionamento do VIED, através do *software Wireshark*, *AC-MViewer* e o da IHM da MU. O resultado dos testes comprova a eficiência de um VIED, tanto na comunicação vertical e horizontal, bem como na leitura de valores analógicos do SV. Os VIEDs possibilitam um novo aspecto de virtualização de dispositivos físicos, podendo, através da implantação de aplicações embutidas no SAGE, fazer a substituição da proteção física de um vão por uma virtual nas funções de proteção testadas, sendo também uma apropriada fonte de treinamento de pessoal de O & M no entendimento dos protocolos utilizados, qualificando no estado da arte em proteção de sistemas elétricos de potência e aplicado em subestações inteligentes.

Palavras-chave: Virtualização; IEC 61850; IED; VIED; SAGE.

ABSTRACT

The IEC 61850 standard and its updates have had a significant impact on substation protection, automation, and control (PAC) systems around the world. New advances are implemented and evaluated every day, such as the virtualization of intelligent electronic devices (IEDs), the subject of this study. The virtualization of IEDs stems from the need to reduce maintenance and hardware replacement costs, optimize productivity, and support the recovery of problems with physical IEDs in installations. The aim of this work is to apply virtual IEDs (VIEDs), improving those developed at the Intelligent Electrical Networks Group (GREI) of the Federal University of Ceara (UFC), in a test platform environment that simulates a substation that is part of the Basic Grid. The performance of VIEDs was assessed on the process bus network, with the traffic of information digitized by a Merging Unit (MU), and on the station bus, with the exchange of messages between the Human Machine Interface (HMI) supervisory software and the manageable switches. To test and deploy the VIEDs, a PAC platform was set up in the structured laboratory at the Companhia Hidroelétrica do São Francisco (CHESF - Fortaleza) with physical IEDs, MU, a six-phase protection relay tester, a Human Machine Interface (HMI) with the Open Energy Management System (SAGE) and notebooks running the VIED and SAGE applications in the Linux operating environment. On the platform, the VIED's response under fault situations was evaluated for the overcurrent protection functions 50/50N (instantaneous) and 51/51N/51V (timed), 67/67N (directional) and 62BF, compared with physical IEDs present on the platform, with the same protection settings as the VIED. IED-VIED communication was tested using GOOSE and Sampled Values (SV) protocols and VIED-SAGE communication using MMS protocol. These tests validated the full operation of the VIED, using Wireshark software, ACMViewer and the MU's HMI. The results of the tests prove the efficiency of a VIED, both in vertical and horizontal communication, as well as in reading analog values from the SV. VIEDs enable a new aspect of virtualization of physical devices, and through the implementation of applications embedded in SAGE, can replace the physical protection of a span with a virtual one in the protection functions evaluated. They are also an appropriate source of training for O&M personnel in understanding the protocols used, qualifying them in the state of the art in the protection of electrical power systems and applied in intelligent substations.

Keywords: Virtualization; IEC 61850; IED; VIED; SAGE.

LISTA DE FIGURAS

Figura 1 – Arquitetura de um SAS legado antes da norma IEC 61850.....	23
Figura 2 – Arquitetura moderna utilizando a Norma IEC 61850	24
Figura 3 – Arquitetura de Sistema de Automação de Subestações com MU	26
Figura 4 – Estrutura hierárquica dos dados	30
Figura 5 – Modelo de troca de informações de dispositivos no padrão IEC61850.....	31
Figura 6 – Modelo OSI e os serviços da norma IEC 61850.....	33
Figura 7 – Arquitetura de protocolos suportados pelo SAGE	35
Figura 8 – Arquitetura SAGE e seus subsistemas	37
Figura 9 – Visor de Acesso do SCADA SAGE	37
Figura 10 – Visor de Telas e de Alarmes no SCADA SAGE	38
Figura 11 – Visor de Processos no SCADA SAGE	39
Figura 12 – Relacionamentos e processos das bases de dados do SAGE	40
Figura 13 – Diagrama de blocos da MU para aquisição de sinais analógicos.....	43
Figura 14 – Curvas IEC	47
Figura 15 – Diagrama fasorial da proteção 67	47
Figura 16 – Arquivos para criação de uma instância servidora na libIEC61850	52
Figura 17 – Código para iniciar a IS MMS no VIED.....	53
Figura 18 – Código de atualização de LN de sinalização de <i>trip</i> geral	53
Figura 19 – Código de atualização de LN de medições analógicas	54
Figura 20 – Fluxograma de controle.....	54
Figura 21 – Código de habilitação do VIED para publicar e assinar mensagens GOOSE	55
Figura 22 – Código do VIED onde assina mensagens GOOSE	56
Figura 23 – Estrutura de uma APDU com destaque na ASDU e DataSet de valores	57
Figura 24 – Fluxograma da medição de corrente e tensão pelo SV	58
Figura 25 – Fluxograma das funções e interações.....	59
Figura 26 – Fluxograma da lógica da proteção 50 trifásico	61
Figura 27 - Fluxograma da lógica da proteção 50N	63
Figura 28 – Fluxograma da lógica da proteção 51 para fase A	64
Figura 29 – Fluxograma da lógica da proteção 51V para fase A	65
Figura 30 – Fluxograma função 67 fases (representado somente fase A).....	66
Figura 31 – Fluxograma da função 50 BF	68
Figura 32 – Plataforma de testes do VIED na Eletrobrás CHESF	71

Figura 33 – IHMs da plataforma de testes na Eletrobrás CHESF	72
Figura 34 – Esquemático da arquitetura da plataforma de testes	73
Figura 35 – Configuração de Dataset para o IED.....	75
Figura 36 – Dataset do VIED para MU	76
Figura 37 – <i>Software</i> Wireshark com os Datasets	76
Figura 38 – <i>Software</i> de configuração da MU.....	77
Figura 39 – Detalhes da lógica para comando do relé biestável	78
Figura 40 – Mensagem GOOSE do estado do disjuntor	79
Figura 41 – Editor da mensagem GOOSE	79
Figura 42 – Captura da mensagem GOOSE pelo Wireshark	80
Figura 43 – Base de dados referência do SCADA SAGE para SE MND	81
Figura 44 – Base de dados com as pastas de INCLUDE	81
Figura 45 – Tela de interface da planilha de cálculo de tempo de atuação	83
Figura 46 – Tela de alarmes e mímico do SCADA SAGE	85
Figura 47 – Medidas analógicas e disparo de <i>trip</i> para o disjuntor	85
Figura 48 – <i>Status</i> do comando de disjuntor (ACMViewer, SAGE e IHM MU)	86
Figura 49 – Tela do SAGE com o disjuntor fechado após o comando	86
Figura 50 – Mensagens trafegadas na LAN da plataforma	87
Figura 51 – Tela de interface do VIED com função de proteção 50	88
Figura 52 – Tela do SAGE com o resultado do VIED atuando a proteção 50.....	89
Figura 53 – Tela de interface do VIED com função de proteção 51	90
Figura 54 – Cálculo do tempo de atuação com valores ajustados	90
Figura 55 – Tela do SAGE com o resultado do VIED atuando a proteção 51.....	91
Figura 56 – Tela de interface do VIED com função de proteção 51V	92
Figura 57 – Tela de interface do VIED com função de proteção 67	92
Figura 58 – Cálculo do tempo de atuação com valores ajustados	93
Figura 59 – Tela do SAGE com os disparos da função 67	94
Figura 60 – Tela de interface do VIED com função de proteção 67N	94
Figura 61 – Testes de Proteção VIED-IED com Proteção 50 Trifásico	96
Figura 62 – Testes de Proteção VIED-IED com Proteção 51 Trifásico	98
Figura 63 – Testes de Proteção VIED-IED com Proteção 51 Fase B	99
Figura 64 – Testes de Proteção VIED-IED com Proteção 51 Fase C	101
Figura 65 – Tela com as pastas dos VIEDs utilizados no trabalho	156
Figura 66 – Estrutura de documentos da Norma IEC 61850.....	157

Figura 67 – Relacionamentos das Entidades da Base de Dados do SCADA SAGE.....	158
---	-----

LISTA DE TABELAS

Tabela 1 – Documentação do Padrão IEC 61850 2.0.....	28
Tabela 2 – Capítulos e Partes da Norma IEC 61850 Parte 2.1.....	29
Tabela 3 – Tipos de Mensagens.....	33
Tabela 4 – Valores das constantes de curvas.....	47
Tabela 5 – Lista de Pontos com as mensagens no padrão IEC61850.....	62
Tabela 6 – Lista de Dispositivos e seus IPs na Plataforma.....	76
Tabela 7 – Lista de Entidade do SCADA SAGE.....	111

LISTA DE SIGLAS E ABREVIATURAS

A/D	Analógico/Digital
ACSI	<i>Abstract Communication Service Interface</i>
ANSI	<i>American National Standards Institute</i>
APDU	<i>Application Protocol Data Unit</i>
API	<i>Application Programming Interface</i>
API-C	API - Cliente
API-S	API - Servidor
ASDU	<i>Application Service Data Unit</i>
BI	<i>Binary Input</i>
BO	<i>Binary Output</i>
CAPEX	<i>CAPital EXpenditure</i>
CC	Corrente Continua
CDC	<i>Common Data Classes</i>
CEPEL	Centro de Pesquisas da Eletrobrás
CHESF	Companhia Hidroelétrica do São Francisco
CID	<i>Configuration IED Description</i>
CIGRÉ	Conselho Internacional de Grandes Sistemas Elétricos
DA	<i>Data Attribute</i>
DPC	<i>Controllable Double Point</i>
EPRI	<i>Electric Power Research Institute</i>
FC	<i>Functional Constraint</i>
GoCB	<i>GOOSE Control Block</i>
GOOSE	<i>Generic Object Oriented Substation Event</i>
GREI	Grupo de Redes Elétricas Inteligentes
GSSE	<i>Generic Substation Status Event</i>
HAL	<i>Hardware Abstraction Layer</i>
IC	Instancia Cliente
ICD	<i>IED Capability Description</i>
IEC	<i>International Electrotechnical Commission</i>
IED	<i>Intelligent Electronic Device</i>
IP	<i>Internet Protocol</i>
IS	Instancia Servidor

KVM	<i>Kernelbased Virtual Machine</i>
LAN	<i>Local Area Network</i>
LD	<i>Logical Device</i>
LED	<i>Light-Emitting Diode</i>
LN	<i>Logical Node</i>
MAABP	Metodologia Ativa de Aprendizagem Baseada em Projetos
MAC	<i>Media Access Control</i>
MAN	<i>Metropolitan Area Network</i>
MMS	<i>Manufacturing Message Specification</i>
MSVCB	<i>Multicast Sampled Values Control Block</i>
MPCCS	Medição, Proteção, Comando, Controle e Supervisão
MU	<i>Merging Unit</i>
NA	Normalmente Aberto
NF	Normalmente Fechado
OPEX	<i>Operational Expenditure</i>
OSI	<i>Open System Interconnection</i>
PAC	Proteção, Automação e Controle
PD	<i>Physical Device</i>
PDU	<i>Protocol Data Unit</i>
RMS	<i>Root Mean Square</i>
SAGE	Sistema Aberto de Gerenciamento de Energia
SAS	Sistema de Automação de Subestações
SCADA	<i>Supervisory Control And Data Acquisition</i>
SCD	<i>Substation Configuration Description</i>
SCL	<i>Substation Configuration Language</i>
SCSM	<i>Specific Communication Service Mapping</i>
SE	Subestação
SEP	Sistema Elétrico de Potência
SO	Sistema Operacional
SSD	Unidade de Estado Solido
SV	<i>Sampled Values</i>
TC	Transformador de Corrente
TCO	Transformador de Corrente Ótico
TP	Transformador de Potencial

TPAA	<i>Two Party Application Association</i>
TPO	Transformador de Potencial Ótico
UCS	Unidade de Controle de Subestação
UFC	Universidade Federal do Ceara
USB	<i>Universal Serial Bus</i>
USVCB	<i>Unicast Sampled Values Control Block</i>
VID	<i>Virtual Intelligent Electronic Device</i>
VM	<i>Virtual Machine</i>
WAN	<i>Wide Area Network</i>
XML	<i>eXtensible Markup Language</i>

SUMÁRIO

1 INTRODUÇÃO	18
1.1 Motivação	19
1.2 Objetivos	19
1.2.1 Objetivos Gerais	19
1.2.2 Objetivos Específicos	19
1.3 Metodologia	20
1.4 Contribuições científicas	20
1.5 Estrutura da dissertação	21
2 FUNDAMENTAÇÃO TEÓRICA DE AUTOMAÇÃO, CONTROLE E PROTEÇÃO DE SUBESTAÇÕES	22
2.1 Evolução do SAS	22
2.2 Norma IEC 61850	26
2.3 Software EMS/SCADA SAGE	33
2.4 Biblioteca libIEC61850	41
2.5 Merging Unit.....	42
2.6 Redes de Comunicação.....	43
2.7 Proteção de Sistemas Elétricos de Potência	45
2.8 Considerações Finais	48
3 FUNDAMENTOS DO VIED DE PROTEÇÃO E CONTROLE	49
3.1 Criação de Arquivos para os VIEDs.....	51
3.2 Funções de Supervisão e Controle implementadas nos VIEDs	52
3.2.1 Mensagens MMS	52
3.2.2 Mensagens GOOSE	55
3.2.3 Mensagens SV.....	56
3.3 Funções de Proteção Implementadas nos VIEDs	59
3.3.1 Função de Sobrecorrente Instantânea de Fase (50)	61
3.3.2 Função de Sobrecorrente Instantânea de Neutro (50N)	62
3.3.3 Funções de Sobrecorrente Temporizada de Fase (51) e com Restrição de Tensão (51V)	63
3.3.4 Função de Sobrecorrente Temporizada de Neutro (51N).....	65
3.3.5 Função de Sobrecorrente Direcional (67)	66
3.3.6 Função de Sobrecorrente Direcional de Neutro (67N).....	67

3.3.7 Função de Falha de Disjuntor (50/62BF).....	67
3.4 Considerações Finais	69
4 PLATAFORMA DE DESENVOLVIMENTO E TESTES COM VIED.....	70
4.1 Plataforma de Automação e Controle de Testes.....	70
4.2 Configuração dos equipamentos da plataforma.....	74
4.2.1 Definição da Mensagem GOOSE enviada pelo VIED.....	74
4.2.2 Configuração da MU para funcionamento na Plataforma PAC	77
4.2.3 Configuração do SCADA SAGE para funcionamento na Plataforma PAC	80
5 RESULTADOS DOS TESTES DA PLATAFORMA	83
5.1 Execução e validação dos testes e resultados.....	83
5.2 Testes de Comando, Proteção e Supervisão	84
5.2.1 Estado, comandos e medições do disjuntor do vão da MU	84
5.2.2 Função 50	87
5.2.3 Função 51	88
5.2.4 Função 51V.....	91
5.2.5 Função 67	92
5.2.6 Função 67N	94
5.3 Testes do VIED comparados com IED Físico	95
5.3.1 Testes com a função de sobrecorrente instantânea (50)	95
5.3.2 Testes com a função de sobrecorrente temporizada (51)	97
5.3.2.1 Falta Trifásica.....	97
5.3.2.2 Falta monofásica Fase B.....	98
5.3.2.3 Falta monofásica Fase C.....	100
6 CONCLUSÃO E TRABALHOS FUTUROS.....	102
6.1 Conclusão	102
6.2 Trabalhos Futuros	103
REFERÊNCIAS	105
APÊNDICE A	110
APÊNDICE B.....	113
ANEXO A.....	156
ANEXO B	157
ANEXO C	158
ANEXO D.....	159

1 INTRODUÇÃO

Nos últimos anos, com a introdução de equipamentos com microprocessadores e comunicação digital nas subestações de energia elétrica, o sistema de medição, proteção, controle, comando e supervisão ficou sujeito a uma alteração substancialmente positiva, principalmente nos relés de proteção, nas unidades de controle e nos sistemas de monitoramento supervisão. Esses equipamentos, se tornaram mais programáveis, flexíveis, compactos e com maiores precisões e velocidade, são essenciais para a correta atuação das lógicas de proteção, em um sistema que está ficando cada vez mais complexo, que é o Sistema Elétrico de Potência (SEP). Porém, se não utilizados os mais recentes *firmwares* das lógicas de proteção e controle, ou forem feitos testes incipientes para novos equipamentos que adentram no SEP, as falhas em isolar uma linha de transmissão com problemas ou desarmar incorretamente uma saudável podem levar a eventos em cascata e apagões de grandes proporções, como os vistos nos últimos tempos (Wójtowicz, 2018).

Atualmente o monitoramento, a proteção e o controle em subestações (SEs) e redes elétricas se dão, em sua maioria, de maneira local e remota, através dos dispositivos eletrônicos inteligentes (IEDs, sigla em inglês) de diversos fabricantes, sistemas de controle supervisão e aquisição de informações conectados por meio de redes de comunicação de dados baseadas em diversos tipos de protocolos de comunicação. Conforme Ansari (2019), essas tecnologias modernas utilizadas nas SEs permitem mais segurança e flexibilidade na rede para garantir confiabilidade ao fornecimento elétrico.

A interoperabilidade, fruto da norma IEC 61850, tem na comunicação de dados um grande desafio, para conectar com transparência os próprios dispositivos e com rapidez necessária para ter elevadas taxas de amostragem, permitindo medições fasoriais de corrente e tensão (IEC 61850, 2023).

As tecnologias de virtualização do ambiente de execução de aplicações ou de plataformas de *hardware* têm sido objeto da atenção crescente de pesquisadores, fabricantes de *hardware/software*, administradores de sistemas e usuários avançados (Tanenbaum, 2016). A virtualização de recursos é um conceito relativamente antigo, mas os recentes avanços nessa área permitem usar máquinas virtuais (VM, sigla em inglês) com os mais diversos objetivos, como a segurança, a compatibilidade de aplicações legadas ou a consolidação de servidores (Jesus, 2023). Alguns conceitos, como o de contêiner, podem ser incorporados nessas aplicações, visando dar uma maior velocidade no processamento de informações entre as aplicações.

Esse trabalho é um avanço na pesquisa e no desenvolvimento de IED virtual, denominado VIED, desenvolvido no Grupo de Redes Elétricas Inteligentes (GREI) da Universidade Federal do Ceará (UFC), com aplicação voltada ao sistema de distribuição de energia elétrica. Neste trabalho, a aplicação do VIED baseado na norma IEC 61850 está inserido no âmbito do sistema de transmissão de energia elétrica.

1.1 Motivação

Segundo Wójtowicz (2018), a tecnologia de proteção digital foi aprimorada significativamente desde a sua criação em 1969. Os sistemas digitais de automação mais modernos utilizam o padrão de transferência de dados com base na norma IEC 61850, que envolvem arquiteturas de comunicação dos tipos cliente/servidor e editor/assinante, com dispositivos utilizando microprocessadores ligados numa rede local *Ethernet* (LAN). Uma subestação de energia elétrica é composta de vários IEDs que comportam diversas funções de Medição, Proteção, Controle, Comando e Supervisão (MPCCS), conectados em uma LAN. Esses dispositivos para alcançar o *design* atual, consumiram tempo e dinheiro para implantação no mercado, com longos ciclos de desenvolvimento, testes e atualização complexos. Neste trabalho, algumas funcionalidades de proteção e controle dos IEDs físicos são virtualizadas, desacoplando *hardware* e *software*, integrando o SCADA SAGE com a MU por intermédio de VIEDs, facilitando o entendimento de uma transição no futuro, de sistemas de MPCCS tradicionais para os virtualizados.

1.2 Objetivos

1.2.1 Objetivos Gerais

O objetivo principal desse trabalho é apresentar a virtualização de um Sistema de Proteção e Controle com *Merging Units*, VIEDs baseado na biblioteca libIEC61850, e o Sistema SCADA SAGE integrados em uma plataforma de subestações da Rede Básica.

1.2.2 Objetivos Específicos

Os objetivos específicos da proposta são:

- a) Desenvolver as funções de proteção (50/50N, 51/51N, 51V, 50/62BF, 67/67N), implementando nos *softwares* dos VIEDs, sempre de acordo com os padrões adotados pelos protocolos da norma IEC 61850;
- b) Desenvolver o controle de abertura e fechamento remoto, a partir do SAGE integrado ao VIED;
- c) Implantar uma bancada para testes experimentais de proteção e controle, para integrar o VIED com IED físicos, comparando seus resultados;
- d) Integrar o VIED com o SCADA SAGE de forma experimental;
- e) Testar e validar *Sampled Values* (SV) oriundos da MU e os SV sintéticos desenvolvido com a libIEC61850.

1.3 Metodologia

Para alcançar os objetivos do trabalho foi montada uma plataforma PAC no laboratório da Eletrobrás CHESF no Departamento Regional em Fortaleza-Ceará, com IEDs físicos, MU, caixa de testes hexafásica para relés de proteção, relé biestável no papel de disjuntor, Interface Homem-Máquina (IHM) com o Sistema Aberto de Gerenciamento de Energia (SAGE) e *notebooks* executando o sistema operacional Linux CentOS, os VIEDs e a versão SCADA SAGE do CEPEL. Os IEDs físicos e VIEDs usados na plataforma se comunicam com equipamentos de pátio simulados pela mala hexafásica, com supervisão, controle e medição de valores analógicos feitos pelo SCADA SAGE. Teste e validação das diferentes funções de proteção dos VIEDs, como 50/50N/50BF, 51/51N/51V e 67/67N, são realizados, e, para visualização de troca de mensagens, são usados os *softwares* de análise de tráfego da rede de comunicação, *Wireshark* e *ACMViewer*.

1.4 Contribuições científicas

Como contribuições pode-se mencionar:

- a) Desenvolvimento e aprendizado da lógica de funcionamento das proteções dos IEDs físicos de mercado, implantados em VIEDs desenvolvidos em linguagem de programação C/C++ e utilizando as bibliotecas da libIEC61850;
- b) Desenvolvimento e aprendizagem na área de automação, através da criação de código de acordo com a norma IEC 61850 para projeto de VIEDs;

- c) Elaboração de cenários de teste que representam situações reais nas SEs e no SEP, de forma prática e ilustrativa vista no código do VIED, no SAGE, no IED físico e nos analisadores de mensagem de rede.

Somado a isso, este trabalho contribui com o desenvolvimento, a implementação e os testes de VIEDs para ensino e treinamento de tecnologias digitais no domínio da proteção, supervisão, automação, controle de subestações, redes elétricas inteligentes e do protocolo padrão IEC 61850, sendo uma evolução do VIED com características de proteção sobrecorrente instantânea e diferencial do trabalho de Freitas (2022).

1.5 Estrutura da dissertação

O trabalho está dividido em seis capítulos.

O Capítulo 1 contém a introdução, a motivação, os objetivos (geral e específicos), a metodologia, as contribuições científicas e a estrutura da dissertação.

O Capítulo 2 apresenta o contexto histórico do Sistema de Automação de Subestação (SAS), com características e informações das estruturas que fazem a automação de uma SE, com foco na norma IEC 61850, no supervisório SAGE, na biblioteca libIEC61850, nas redes *ethernet*, nos sistemas de proteção utilizados e *merging units*.

O Capítulo 3 apresenta o conceito e a modelagem do VIED, bem como sua estrutura para aplicação em proteção e controle.

O Capítulo 4 contém a descrição da plataforma utilizada para os testes, mostrando as características e os detalhes do seu funcionamento.

O Capítulo 5 apresenta a validação dos testes da plataforma, com as funcionalidades que foram desenvolvidas, o envio e recebimento das mensagens GOOSE, MMS e SV. Além de alguns dados comparativos do funcionamento do VIED e o IED com os mesmos dados e parâmetros.

Finalmente, no Capítulo 6 são apresentadas as conclusões e algumas sugestões para continuidades dos trabalhos com os VIEDs e SAS.

2 FUNDAMENTAÇÃO TEÓRICA DE AUTOMAÇÃO, CONTROLE E PROTEÇÃO DE SUBESTAÇÕES

Neste capítulo é apresentada uma contextualização sobre as funções de proteção e controle de IEDs que compõem o SEP, *softwares* e *hardwares* que são utilizados no desenvolvimento da plataforma. Tem um breve histórico sobre a evolução do SAS, dos protocolos de comunicação e da forma como alterou a arquitetura da rede instalada nas SEs.

Em relação aos *softwares*, são abordadas a utilização do *software* SCADA SAGE do CEPEL, suas características básicas e a forma de modelagem de uma base de dados, bem como a biblioteca libIEC61850 e as suas funcionalidades com códigos abertos. Em seguida, a Norma IEC 61850 com suas estruturas das mensagens de comunicação, modelagem de dados e a linguagem de configuração.

Com relação aos *hardwares*, o capítulo aborda a ampla utilização da rede de comunicação *Ethernet* no SAS e *Merging Units*, que torna possível a utilização do VIED através da captura dos sinais SV na rede de processos.

2.1 Evolução do SAS

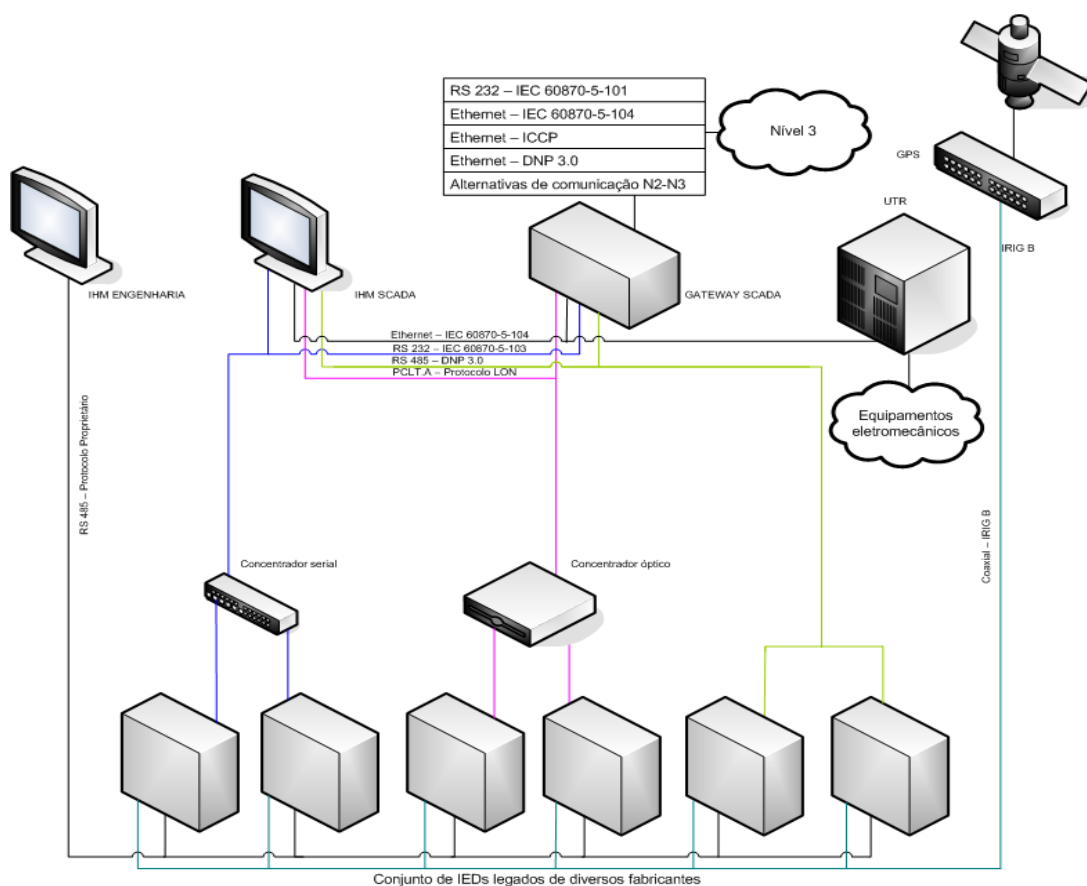
A evolução dos IEDs foi um dos principais responsáveis pelo desenvolvimento da transmissão, geração, distribuição de energia, e principalmente da automação das subestações. Os IEDs modernos são relés digitais numéricos microprocessados que foram substituindo a geração anterior de equipamentos de proteção, controle e medição eletrônicos e eletromecânicos. São responsáveis por funções de medição, monitoramento, registro de faltas e oscilografia, além da função de proteção em si (Alstom, 2011).

O protocolo é a forma de comunicação entre sistemas, feita através de um conjunto de regras que definem a ordem e o tipo de mensagens que irão ser transmitidas entre IEDs. Nos primórdios, cada fabricante de equipamentos de sistemas de automação de subestações desenvolvia seu próprio protocolo de comunicação proprietário. Baseado em seus requisitos de aplicação, possuíam funcionalidades mínimas, informações diferentes e interoperabilidade limitada ou, na maioria dos casos, totalmente impossível (Hitachi, 2023). O *status quo* de um SAS, bem comum de ser encontrado em subestações operacionais nas transmissoras e distribuidoras, antes da publicação da norma IEC 61850 em 2005, é mostrado na Figura 1.

Apesar de parecer bem simples, a arquitetura mostrada na figura citada apresenta diversos aspectos negativos, tanto para os fabricantes como para os usuários. Para os fabrican-

tes, a utilização de protocolos diversos, sendo muitos proprietários, gerava esforços das equipes de desenvolvimento na atualização de recursos com diferentes interfaces, atrasando inclusive futuras inovações e lançamentos de novos modelos. Em alguns casos, os IEDs tinham que alterar totalmente o *firmware*, nessas alterações de interface ou *hardware*, levando a anomalias em partes dos *softwares* que estavam em funcionamento, inclusive em funções críticas, como as de proteção (Hitachi, 2023).

Figura 1 – Arquitetura de um SAS legado antes da norma IEC 61850



Fonte: Elaborado pelo autor.

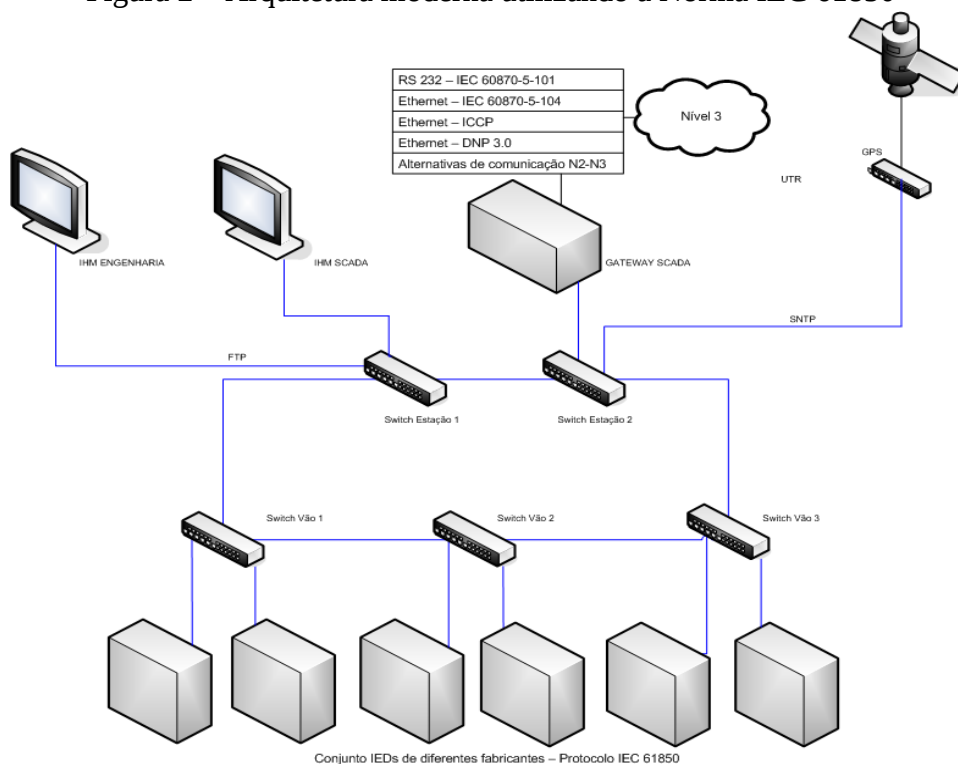
Olhando pelo lado dos usuários, predominantemente concessionárias de energia, a complexidade do gerenciamento de diversos protocolos e tecnologias de comunicação diferentes leva à necessidade de uma equipe com especialistas dedicados a cada uma das tecnologias. Em caso de falhas, nem sempre a equipe estaria disponível para realizar uma manutenção corretiva, fora a dificuldade de encontrar e investigar alguma falha, no meio de tantas interfaces e protocolos diferentes. Isso gera maiores indisponibilidades, que depois da criação da Parcela Variável pela ANEEL, com descontos elevados de remuneração para quaisquer

falhas nos serviços prestados pelas transmissoras, afeta diretamente os negócios da empresa (Jesus, 2008).

Em 1997, foi estabelecido um protocolo mais flexível, o IEC 61870-5-103, que mantinha alguma interoperabilidade entre IEDs e IHMs, mas com o crescente desenvolvimento da automação, suas desvantagens foram aparecendo e a necessidade de conversão de diversos protocolos era necessária para manter a interconexão entre fabricantes.

A disrupção do modelo vigente veio em meados de 2004, com a apresentação da primeira edição da norma IEC 61850, aprofundada no item 2.2. A concepção dos sistemas de MPCCS teve que ser modificada e evoluída para uma nova arquitetura que suportasse o novo padrão. A solução de automação migrou da condição apresentada nos parágrafos anteriores para a ilustrada na Figura 2.

Figura 2 – Arquitetura moderna utilizando a Norma IEC 61850



Fonte: Elaborado pelo autor.

Uma importante vantagem de uma subestação digital, comparada com a convencional, com relés eletromecânicos, relés eletrônicos e UTRs, é que uma grande parte dos cabos de cobre provenientes do pátio e entre painéis de equipamentos de proteção e controle pode ser reduzida, reduzindo os custos em geral de *Capital Expenditure* (CAPEX) e a complexidade de uma obra civil. Com a utilização de um protocolo único para comunicação, po-

de-se aumentar a utilização dos dados, armazenando as informações para utilizações estatísticas de análise ou *business intelligence*, levando à eficiência operacional e a ganhos de produtividade para empresas, reduzindo o *Operational Expenditure* (OPEX).

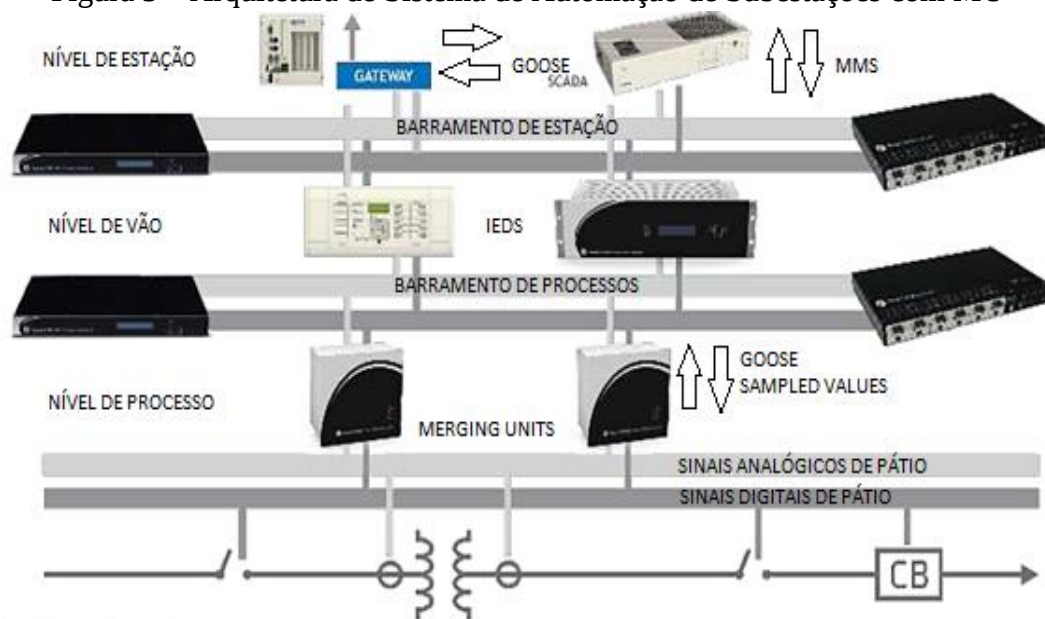
Com a aplicação da norma IEC 61850, estabeleceram-se as funcionalidades dos protocolos de comunicação horizontal com mensagens GOOSE e vertical com mensagens MMS, atrelada a uma rede de alta velocidade, com a troca rápida de múltiplos tipos de mensagens entre os equipamentos de proteção, controle, supervisão e medição da subestação, sempre garantida pela interoperabilidade (Cigré, 2018).

Conforme abordado em Rocha (2020), além dos aspectos operacionais e econômicos, a decisão pela adoção desta nova filosofia das subestações, fazendo uso dos protocolos de comunicação da norma IEC 61850, baseia-se também nos seguintes fatores:

- f) Comunicação de alta velocidade em rede Ethernet;
- g) Interoperabilidade de comunicação entre equipamentos de diferentes fabricantes;
- h) Redução significativa da quantidade de cabos utilizados;
- i) Agilidade no comissionamento e menor probabilidade de falhas;
- j) Alta confiabilidade e disponibilidade do sistema, com projeto mais simples e arquitetura mais eficiente;
- k) Garantia de fácil expansibilidade;
- l) Menor possibilidade de obsolescência no futuro próximo, garantindo o retorno do investimento realizado.

Com a utilização de *Merging Unit*, a arquitetura do SAS se estabeleceu da forma ilustrada na Figura 3, demonstrando a divisão do sistema em três níveis, sendo o Nível 0 (Processo), o Nível 1 (Vão) e o Nível 2 (Estação). Na figura também estão indicados os barramentos de estação e de processo, a direção do fluxo de informações e os protocolos típicos de cada nível. A MU é um módulo concentrador de entradas/saídas analógico/digital, com processamento e interface de comunicação utilizando os protocolos de comunicação. Com o protocolo SV, os sinais analógicos de TCs e TPs são digitalizados e utilizados em medição operacional, qualimetria e proteção, com o protocolo GOOSE e MMS para comunicação horizontal/vertical de sinais digitais.

Figura 3 – Arquitetura de Sistema de Automação de Subestações com MU



Fonte: MU GE (2023, adaptado).

2.2 Norma IEC 61850

O comitê técnico 57 do IEC e o EPRI que criou o padrão UCA se uniram em 1997 para elaborar uma norma que visasse a uma melhoria na comunicação, na performance e na integração dos diversos equipamentos de controle e proteção que estavam surgindo, devido à evolução da microeletrônica e da rede de comunicação (Melo, 2015).

A primeira edição da IEC 61850 foi publicada em 2003/2004 e foi aplicada de imediato nos novos sistemas de proteção, automação e controle. Consolidando a evolução da automação de subestações, que já vinha acontecendo desde os equipamentos primários até os sistemas de Nível 2, o atual estágio dessa evolução é o mais abrangente na maioria dos SAS instalados no SEP das SEs.

A segunda edição da norma foi lançada em 2011 (Sampaio, 2017), com as melhorias propostas devido às mais novas tecnologias, como as PMUs (*Phase Measurement Unit*), gerenciamento de redes, comunicação entre as SEs e Centros de Controle, extensão da área de aplicação para energia renováveis, qualidade de energia e monitoramento *on-line* de equipamentos.

A norma IEC 61850 apresenta novos recursos visando à melhoria da performance e da integração, propiciando redução de custo em vários aspectos, pois simplifica o projeto, reduz os insumos na montagem de painéis, converge vários protocolos legados em um único sistema de PAC e faz a digitalização além das fronteiras da sala de controle, abrangendo os

equipamentos primários no pátio das subestações e ultimamente, nas última edições, as aplicações vão além da rede local da subestação, ou seja, entre subestações que compõem um sistema de uma região mais ampla (Lellys, 2022).

A estrutura da IEC 61850, na sua primeira edição, foi formatada com dez (10) módulos inicialmente, onde as partes têm como base a interoperabilidade, subdividida nos seus três aspectos: Função distribuída, semântica e Sintaxe (IEC, TR 61850-1, 2003). Os temas foram ampliados na segunda edição, sendo distribuídos em partes:

- a) Partes de 1 a 5: contêm os Aspectos do Sistema;
- b) Parte 6: descreve a respeito da Configuração;
- c) Parte 7 e seus subitens: da Estrutura de Comunicação Básica para Equipamentos de SEs;
- d) Partes 8, 80, 9 e 90 com seus subitens: contêm o Mapeamento de Serviços de Comunicação Específicos;
- e) Parte 10: os Testes de Conformidade.

Na Tabela 1, visualizam-se todas as partes mencionadas (Chemin Netto, 2012).

Tabela 1 – Documentação do Padrão IEC 61850 2.0

Parte	Descrição
IEC 61850-1	Introdução e Visão Geral
IEC 61850-2	Glossário
IEC 61850-3	Requisitos Gerais
IEC 61850-4	Gerenciamento de Sistema e Projeto
IEC 61850-5	Requisitos de Comunicação para Modelos de Funções e Dispositivos
IEC 61850-6	Linguagem de Configuração para IED de subestações
IEC 61850-7	Estrutura de Comunicação Básica para Equipamentos de Subestações e Alimentadores
IEC 61850-7-1	Princípios e Modelos
IEC 61850-7-2	Serviços de Comunicação Abstratos (ACSI)
IEC 61850-7-3	Classes de Dados Comuns
IEC 61850-7-4	Classes de Nós Lógicos Compatíveis e Classes de Dados
IEC 61850-7-410	Comunicação para Monitoramento e Controle de Plantas de Geração Hidro-elétrica
IEC 61850-7-420	Nós Lógicos para Fontes de Energia Distribuídas
IEC 61850-7-510	Guia e Conceitos de Modelagem para Plantas de Energia Hidroelétrica
IEC 61850-8-1	Mapeamento de serviço de comunicação específico (SCSM) - Mapeamento para MMS (Manufacturing Message Specification) (International Organization for Standardization - ISO 9506-1 and ISO 9506-2) and to ISO/IEC

IEC 61850-80-1	8802-3. Guia para troca de informações do Modelo da Base de dados CDC usando IEC 60870-5- 101ou IEC 60870.
Parte	Descrição
IEC 61850-9-1	Mapeamento de serviço de comunicação específico (SCSM) - Valores amostrados sobre ISO/IEC 8802-3 (Valores Amostrais sobre Enlace Serial Unidirecional Multidrop Ponto-a-ponto - P2P).
IEC 61850-9-2	Perfil de protocolo de precisão de tempo para automação de concessionária de energia (Mapeamento de Amostra de Valores Analógicos baseado no barramento de processo).
IEC 61850-90-1	Uso do IEC 61850 para comunicação entre subestações
IEC 61850-90-2	Uso do IEC 61850 para transmitir informação de sincrofasores de acordo com IEEEC37.118
IEC 61850-90-3	Modelos de Objeto para conversores de potência em Sistemas de geração de energia distribuída.
IEC 61850-10	Testes de Conformidade

Fonte: Chemin Netto (2012, adaptado).

Os capítulos já foram revisados e integram a última edição da norma (IEC 61850 Ed 2.1), com revisões ocorridas em 2020, nos seguintes aspectos: a integração de PMUs, WAPC (Wide Area Protection & Control), microrredes e geração distribuída. Esses capítulos e partes são mostradas no Anexo 9.2, contam com 44 itens de acordo com a Tabela 2 (IEC 61850:2023).

Tabela 2 – Capítulos e partes da Norma IEC 61850 Parte 2.1

Capítulos e Partes da Norma IEC 61850 Parte 2.1			
IEC TR 61850-1:2013	IEC TR 61850-7-5:2021	IEC 61850-10:2012	IEC TR 61850-90-7:2023 RLV
IEC TS 61850-1-2:2020+AMD1:2022 CSV	IEC TR 61850-7-6:2019	IEC TR 61850-10-3:2022	IEC TR 61850-90-8:2016
IEC TS 61850-2:2019	IEC TS 61850-7-7:2018+AMD1:2023 CSV	IEC TS 61850-80-1:2016	IEC TR 61850-90-9:2020
IEC 61850-3:2013	IEC 61850-7-410:2012+AMD1:2015 CSV	IEC TR 61850-80-3:2015	IEC TR 61850-90-10:2017
IEC 61850-4:2011+AMD1:2020 CSV	IEC 61850-7-420:2021	IEC TS 61850-80-4:2016	IEC TR 61850-90-11:2020
IEC 61850-	IEC TR 61850-7-	IEC TR 61850-90-	IEC TR 61850-90-

5:2013+AMD1:2022 CSV	500:2017	1:2010	12:2020
IEC 61850-	IEC TR 61850-7-	IEC TR 61850-90-	IEC TR 61850-90-
6:2009+AMD1:2018 CSV	510:2021	2:2016	13:2021
IEC 61850-7-	IEC 61850-8-	IEC TR 61850-90-	IEC TR 61850-90-
1:2011+AMD1:2020 CSV	1:2011+AMD1:2020 CSV	3:2016	14:2021
IEC 61850-7-	IEC 61850-8-2:2018	IEC TR 61850-90-	IEC TR 61850-90-
2:2010+AMD1:2020 CSV		4:2020	16:2021
IEC 61850-7-	IEC 61850-9-	IEC TR 61850-90-	IEC TR 61850-90-
3:2010+AMD1:2020 CSV	2:2011+AMD1:2020 CSV	5:2012	17:2017
IEC 61850-7-	IEC/IEEE 61850-9-	IEC TR 61850-90-	IEC TR 61850-90-
4:2010+AMD1:2020 CSV	3:2016	6:2018	27:2023

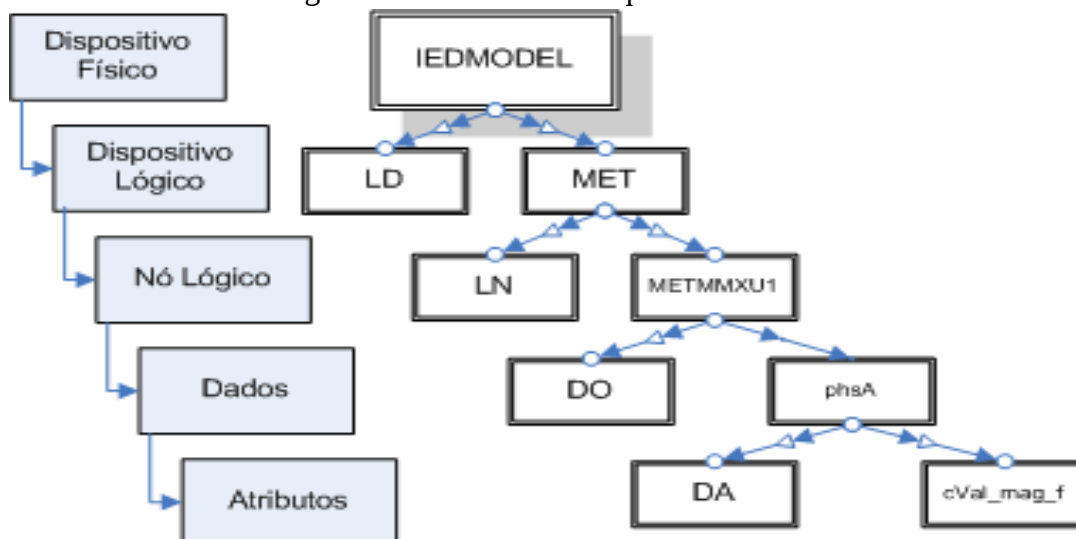
Fonte: International Electrotechnical Commission (2022, adaptado).

Um das características da norma, vista em várias literaturas, que ela é a “prova do futuro”, ou seja, é a independência entre a evolução das tecnologias de comunicação e sua estrutura em relação ao modelo de dados. Com essa independência desses dois aspectos, as evoluções podem se dar em ambos com diferentes velocidades. Quanto à comunicação, que tem maior velocidade nas alterações, pode-se comentar que a norma já acrescentou vários aspectos das suas tecnologias, como a implementação do protocolo PTP ao sincronismo. Outro detalhe é inerente à comunicação do SV e à forma de comunicação em virtude da entrada de equipamentos primários com interfaces digitais, mas o modelo de mapeamento dos dados permanecerá o mesmo. No caso dos dados, pode-se citar um exemplo que, caso seja criado um bloco novo, para um novo equipamento que irá surgir, isso não atrapalha a forma de comunicação, em virtude da utilização da mesma estrutura de modelagem das informações (Hitachi, 2023).

A modelagem dos dados da norma IEC 61850 é baseada em UML (*Unified Modeling Language*), que é muito utilizada para sistemas orientados a objetos (Apostolov, 2010). É utilizando a linguagem XML (*eXtensible Markup Language*) que utiliza um conjunto de *tags* (etiquetas ou marcações) para definir o formato dos dados, como uma marca de identificação. A XML é usada para marcar o documento, estruturado de acordo com a Norma descrevendo os diferentes aspectos da configuração da subestação, a partir da perspectiva do sistema de automação da subestação.

Como exemplo, na Figura 4, observa-se como é a forma de distribuição das informações das medições aquisitadas em SV da corrente primária da fase A de um vão: (IED-MODEL_MET_METMMXU1_A_phsA_cVal_mag_f).

Figura 4 – Estrutura hierárquica dos dados



Fonte: Elaborado pelo autor.

A estrutura de dados, também denominada modelo de dados, define os elementos que compõem um documento, o que se deseja transmitir/coletar de informações e a relação desses elementos entre si. De início é considerado que o IED, um servidor baseado em um dispositivo físico (*Physical Device* – PD) no exemplo *IEDMODEL*, que é a constituição real do equipamento. Este IED tem as diversas funções por ele desempenhadas, agregadas em um mesmo objetivo, chamadas de dispositivos lógicos (*Logical Devices* – LD), na figura *MET*. Os vários nós lógicos (*Logical Nodes* – LN) são as menores entidades que podem permitir o envio dos dados e trocar informações, no exemplo *METMMXU1*. LNs são *containers* que mantêm os dados (*Data Objects* - DO) que remetem a alguma informação de uma função (*phsA* no exemplo) com os seus diversos atributos (*Data Attributes* – DA), que seria a descrição do *status* dessa função, *cVal_mag_f* no exemplo (IEC/TR 61850-7, 2003).

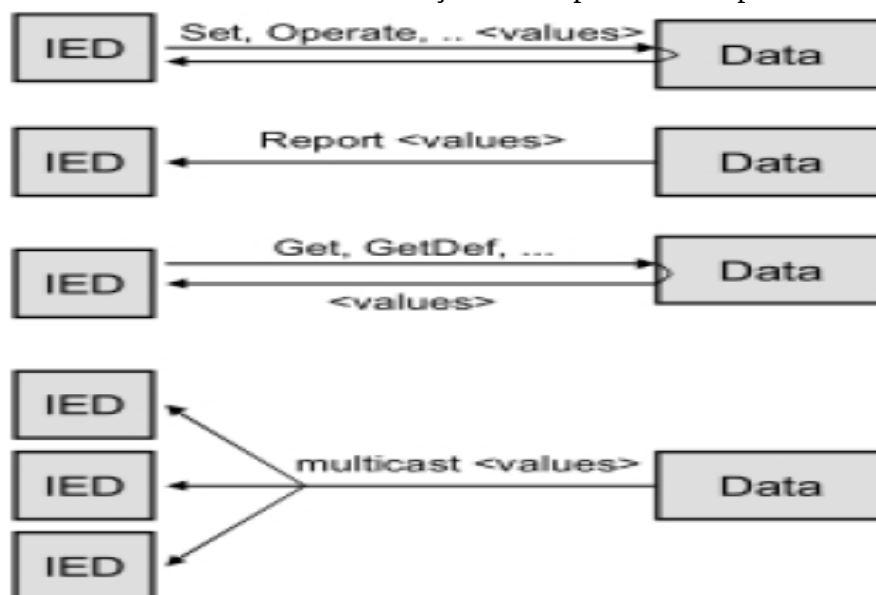
A linguagem de configuração *Substation Configuration Language* (SCL) é a responsável pela configuração dos SAS na norma IEC 61850. Ela está especificada no sexto módulo da norma, baseada no padrão XML, explicado anteriormente, descreve o IED e a sua relação com outros dispositivos da subestação. Os tipos de arquivos gerados são descritos abaixo (IEC/TR 61850-6, 2004):

- a) *Configured IED Description (CID)*: arquivo que é enviado ao IED, gerado pela ferramenta de configuração, com os dados de ajuste e parametrizações;
- b) *IED Capability Description (ICD)*: arquivo que contém as configurações do IED utilizado para a ferramenta de configuração do sistema;
- c) *System Specification Description (SSD)*: arquivo gerado com todos os dados do SAS, configurados a partir do diagrama unifilar.
- d) *Substation Configuration Description (SCD)*: arquivo gerado com toda a configuração da rede de uma subestação, nos quesitos comunicação e fluxo de dados.

Para promover a troca de informações, o item 7.1 da norma, ilustrado na Figura 5, define as seguintes padronizações:

- a) *Get/Set Operate*: serviços responsáveis pelo monitoramento ou controle (*Get*) de um SCADA com os IEDs de uma rede, ou de envio de dados ou comando para esses IEDs (*Set*);
- b) *Multicast*: baseado no envio de informações de um remetente para vários pontos que compõem a rede, sem verificação de erro de transmissão e em alta velocidade;
- c) *Report*: notifica as aplicações tipo cliente acerca de alteração dos dados de um *dataset*, podendo ser do tipo *buffered* ou *unbuffered*, de acordo com os objetivos do que fazer com os dados que virão nos elementos do *dataset*.

Figura 5 – Modelo de troca de informações de dispositivos no padrão IEC61850



Como comentado anteriormente, os cabos estão sendo substituídos por mensagens de protocolo, podendo ser no formato de mensagens do tipo MMS, ou as de alta velocidade que são transmitidas através dos protocolos GOOSE e SV, definidos na IEC 61850-5, e comentados resumidamente (Soares, 2021):

- a) MMS: o protocolo foi incorporado à norma para realizar aplicação voltada à supervisão e ao controle dos dispositivos, sendo a forma de implementar a comunicação vertical (aquisição de dados e disparo de comandos) entre os IEDs e aplicações no barramento de estação. Esse protocolo é orientado à conexão e à troca de informações entre dispositivos, segue o modelo Cliente-Servidor com sua modelagem orientada a objetos. Normalmente os servidores MMS são executados em dispositivos na subestação e os clientes capturam esses objetos fornecidos como servidor. Muitas vezes os dispositivos executam as duas funções (Soares, 2021);
- b) GOOSE: para transmissão de eventos de subestações, através de mensagens rápidas nos barramentos de estação ou de processos, operando na camada 2 (enlace) no modelo OSI. O tempo de envio dessas mensagens ocorre de 3 a 5 milissegundos, considerando a correta implementação da arquitetura e configuração dos equipamentos. As mensagens ou *frames* são enviados periodicamente, em curtos períodos, ou quando um novo evento acontece, como uma mudança de valor de um atributo; funciona de forma assíncrona, na qual não há confirmação de entrega da mensagem;
- c) SV: para envio de informações analógicas de corrente e tensão, de forma rápida, via rede de comunicação, podendo ser digitalizadas por MUs ou através de TIs com a tecnologia ótica. O SV opera na camada 2, como o GOOSE, porém o *trigger* é diferente. No caso do SV é um evento de tempo, o ciclo de 80 amostras enviadas de corrente e tensão, que na frequência de 60 Hz totalizam 4800 amostras por segundo.

Na Tabela 3 estão classificados os tipos de mensagens e suas respectivas prioridades.

Tabela 3 – Tipos de mensagens

Tipo	Classificação	Protocolo	Tempo
1	Mensagens rápidas	GOOSE	20 ms
1 A	<i>Trip</i>	GOOSE	3 ms

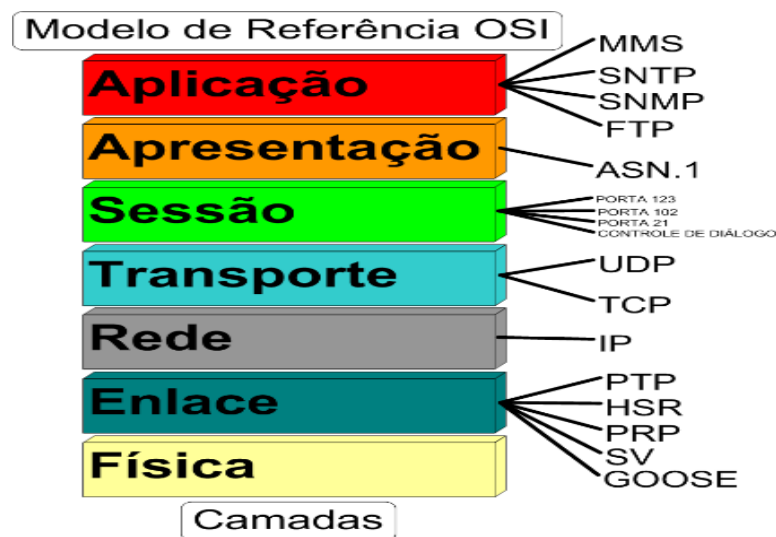
2	Velocidade média	MMS	100 ms
3	Velocidade baixa	MMS	500 ms
4	Dados em rajada (raw data)	SV	3 ms
5	Transferência de arquivo		
6	Sincronização de tempo		

Fonte: Soares (2021, com adaptações).

De fato, o tráfego de rede, em geral, é determinante em termos de tempo. As mensagens de SV são do tipo 4, GOOSE são tipo 1, pois o tempo é crítico, por isso estão mapeadas diretamente na camada de enlace. As mensagens de velocidade média e baixa, por exemplo, tipos 2, 3, 5 e 6 são chamadas de mensagens IP baseadas em cliente/servidor e mapeadas na pilha TCP/IP, nesse tipo estão as mensagens MMS, que estão na camada de aplicação.

As mensagens de sincronização de tempo (tipo 6) são transmitidas usando o *User Datagram Protocol/Internet Protocol* (UDP/IP). Na Figura 6, são apresentadas as mensagens padronizadas pela norma, associada às camadas do modelo OSI.

Figura 6 – Modelo OSI e os serviços da norma IEC 61850



Fonte: Hitachi (2023, com adaptações).

2.3 Software EMS/SCADA SAGE

Há quase três décadas, que o cenário do desenvolvimento de sistemas supervisórios para centros de controle se alterou com a utilização de microcomputadores ligados em uma rede de dados, com uma maior capacidade de processamento e o surgimento de novas plataformas de *hardware* (*workstations*). Essa evolução dos sistemas foi levada também às

subestações, que tiveram essas máquinas, no ambiente da sala de comando, com *softwares* e *hardware* provenientes de alguns fabricantes de sistemas de automação, com soluções proprietárias e outros desenvolvimentos de aplicativos por algumas empresas do setor elétrico, facilitando o gerenciamento do SEP da SE (Hitachi, 2023).

A funcionalidade típica dos *softwares* SCADA, instalados nas subestações e centros de controle, evoluíram de um simples sistema de aquisição de dados para sistema com grande número de aplicações computacionais avançadas, denominado Sistema de Gerenciamento da Energia (EMS, sigla em inglês). Com essa evolução veio a melhoria da qualidade da informação, troca de dados com outros *softwares* corporativos de apoio à gestão de ativos de uma empresa (CHESF, 2011).

Os três principais componentes que fazem um SAS são os computadores que compõem o conjunto de IHMs com o *software* SCADA, o conjunto de IEDs e/ou a UTR (Unidade de Terminal Remota) que compõe o sistema de controle e aquisição de dados e a infraestrutura de comunicação. Nos últimos anos, essas unidades de SAS estão sendo levadas à obsolescência muito rápido, devido ao avanço das tecnologias de *hardware* e às novas versões de *softwares* e seus *drives*, tornando a manutenção com custo proibitivo, e as expansões chegam a exigir trocas de *software*, *hardware* e até de todo o SAS.

Uma das soluções para esse problema, pelo menos na parte da IHM, consiste na utilização de um sistema distribuído em arquitetura aberta, que suporte plataformas de *hardware* de todos os tipos, não comprometendo o nível de desempenho e segurança de todo esse ambiente. Assim, no Centro de Pesquisas da Eletrobrás (CEPEL) foi desenvolvido o SAGE, que é um sistema SCADA/EMS de grande porte e alto desempenho. Em sistemas de supervisão localizados nas subestações, o SAGE exerce a função SCADA, já nos Centro de Operação são utilizados os módulos de gerenciamento de energia, fazendo a função de EMS (CEPEL, 2023a).

O SAGE está sendo utilizado por várias empresas do ramo de energia elétrica, desde concessionárias de geração, transmissão e distribuição no Brasil, até nos centros de controle do Sistema Eletrobrás e do ONS, sendo bem utilizado na maioria das subestações da rede básica. O SCADA SAGE é utilizado em praticamente todas as 168 subestações da Eletrobrás CHESF e o EMS SAGE em todos os seus 4 Centros de Controle Remoto.

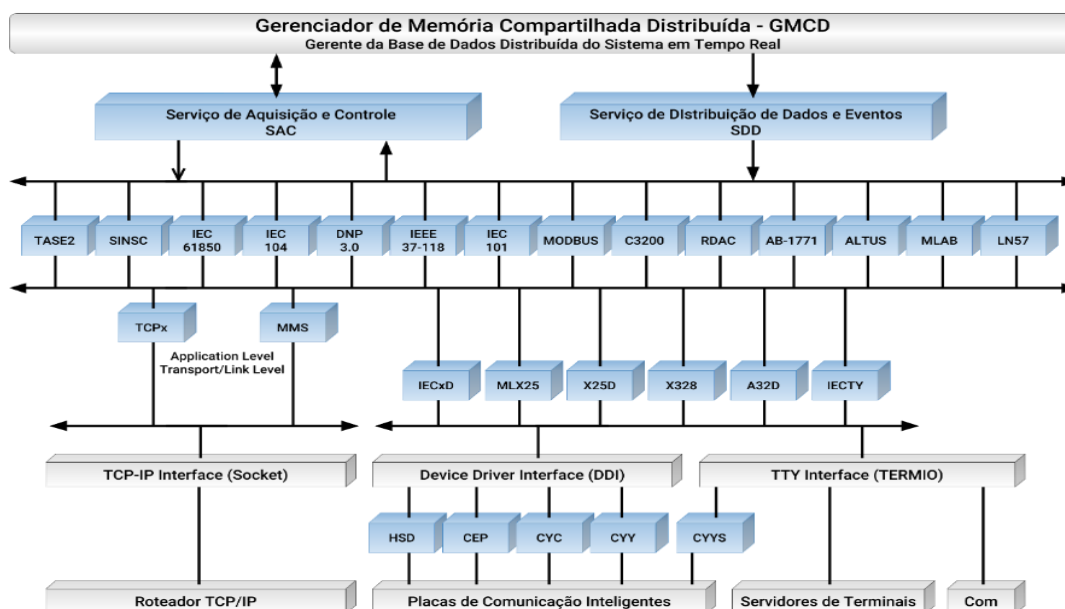
A arquitetura modular permite sua adequada customização para que possa ser utilizado seja como um *gateway* de comunicação, como um concentrador de dados de um sistema de distribuição, como um supervisor local ou regional, como um centro de operação de

sistema ou, ainda, composto por vários centros de controle sincronizados e redundantes (CEPEL, 2023a).

O SAGE implementa as funções de aquisição, tratamento e distribuição de dados do sistema elétrico (SCADA), além de prover suporte a múltiplos protocolos de aquisição e distribuição, conforme mostrado na Figura 7. No tocante à comunicação dos dados que trafegam na rede com a arquitetura do SAGE, existe a presença de três redes distintas (CEPEL, 2023c):

- a) Rede de Processos – utilizada para comunicação com IEDs, CLPs, UTRs etc. Com a função de aquisição de dados e controle, oferecendo a função de distribuição de dados para os Centros de Operação;
- b) Rede de Difusão Confiável – é a rede que trabalha com a responsabilidade de manter a coerência dos dados entre as IHMs, principalmente da Base de Dados de Tempo Real;
- c) Rede Corporativa: pode ser disponibilizada para acessos remotos, possibilitando manutenções a distância do SAS.

Figura 7 – Arquitetura de protocolos suportados pelo SAGE



Fonte: CEPEL (2023c).

Ainda sobre a comunicação, há vários módulos, todos nativos do sistema, que permitem uma boa interação com vários equipamentos, como IEDs e UTRs, centros de controle locais, regionais ou de sistema (COR ou COS) através de protocolos padronizados, tais

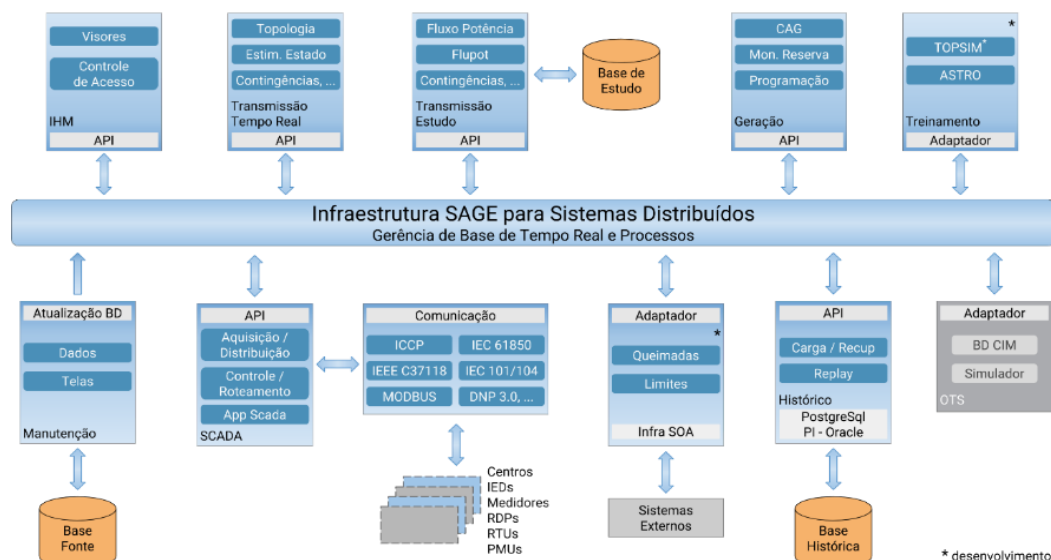
como OPC UA (IEC 62541), IEC 61850, IEC 60870-6 TASE 2), IEEE C37.118, IEC 60870-5-104, IEC 60870-5-101, DNP 3.0 serial ou TCP, Open-ModBus TCP, SNMP, entre outros (CEPEL, 2023c).

O *software* do CEPEL é um sistema desenvolvido com base em conceitos de sistemas distribuídos e expansíveis. O *software* com versões mais atuais, inclusive a última versão do SAGE (SAGE-29), são instalados no Sistema Operacional CentOS, que é uma versão gratuita do *Red Hat Enterprise*, mantido por representantes de várias empresas e voluntários que fazem o time de desenvolvedores. Como tal, o sistema tem sido constantemente expandido, através da adição de novos módulos de *software* e ampliação de *hardware* adicional a qualquer infraestrutura existente e que seja homologada junto ao CEPEL. Diversos outros módulos compõem a solução SAGE, tais como: Sistema de Controle de Geração (CAG), sistema de Armazenamento Histórico de longo prazo (BDH), Interface Gráfica (IHM), suporte a sistemas *multi-site* com facilidades para sincronização de dados e contingenciamento entre centros de controle, interface orientada a serviços (SOA) para conexão com aplicações externas, suporte de múltiplos perfis de usuário, integração com sistemas de segurança eletrônica, entre outros (CEPEL, 2023a).

A arquitetura do SAGE com seus subsistemas é visualizada com detalhes na Figura 8. No tocante às muitas características técnicas do SAGE, pode-se destacar:

- a) Protocolo de comunicação em TCP/IP, com o processo seguindo o modelo OSI, com suporte a milhares de conexões TCP-IP simultâneas, monitoramento e controle, processamento de alarmes e eventos configurável, sequência de eventos;
- b) Sistema Operacional UNIX;
- c) Linguagem de programação (padrão ANSI) Fortran, C e C++;
- d) Tecnologia “*full-graphics*” X Windows e Motif, com visualização de telas em *browser* – tecnologia SVG, visualização em aplicativos para *smartphones* e *tablets* - tecnologia OPC UA, múltiplas janelas, com diagramas unifilares, tabulares, gráficos, listas de alarmes e eventos, camadas, filtragem, *scrolling*, *panning*, *zooming*, *decluttering* e ativação de programas diversos a partir da interface gráfica;
- e) Banco de dados distribuído em tempo real, com informações históricas de dados brutos e consolidados, integração com ambiente corporativo e com versões para banco de dados Oracle, Postgresql, Informix e Osisoft PI.

Figura 8 – Arquitetura SAGE e seus subsistemas



Fonte: CEPEL (2023d).

Acessando às telas e à estrutura interna do SAGE, observa-se que existem três sistemas principais: base de dados, telas e interface gráfica. O Visor de Acesso, mostrado na Figura 9, gerencia e dá acesso às atividades, de acordo com os privilégios do usuário. Na aba de programas tem-se acesso aos botões de inicialização dos visores, sendo que somente com a base de dados com o *status online* dá acesso a todas as aplicações.

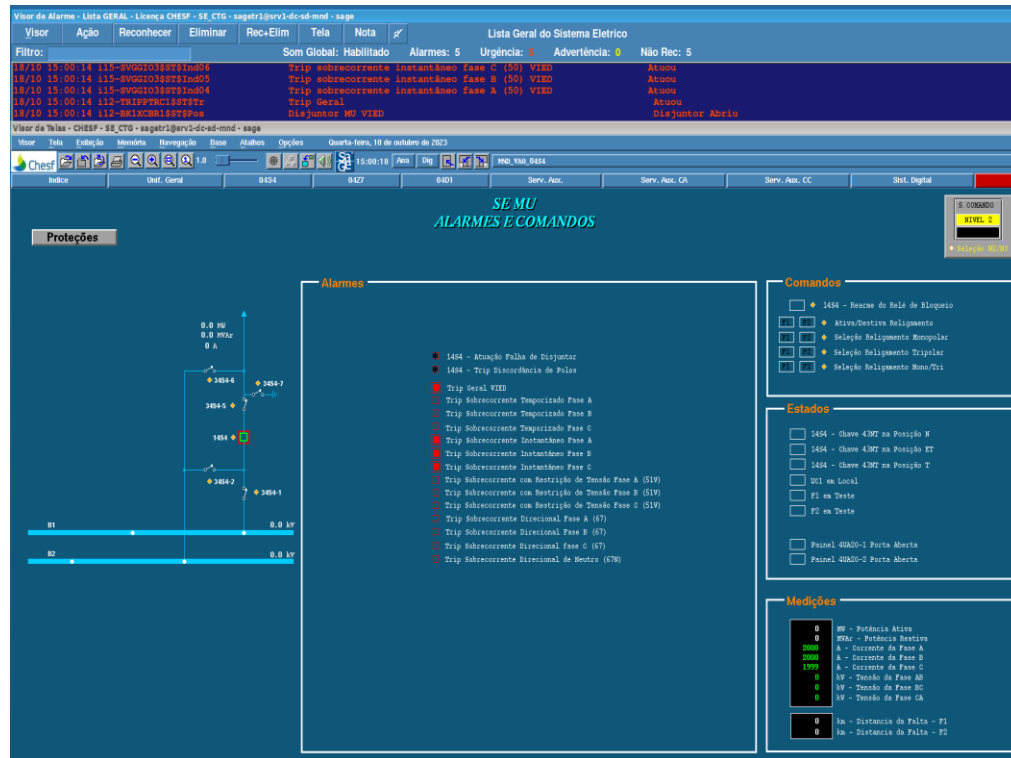
O modo de Visor de Telas (Figura 10), visto em conjunto com o Visor de Alarmes, é uma ferramenta poderosa e simples de configurar, estando todos os objetos bem definidos na base de dados. A indicação de cores dos objetos dá os diagnósticos da aquisição dos pontos (cor verde ou vermelho – normal, cor branca – sem aquisição, cor amarela – inválido na origem, cor azul – aquisição restrita ou manual).

Figura 9 – Visor de Acesso do SCADA SAGE



Fonte: Elaborado pelo autor.

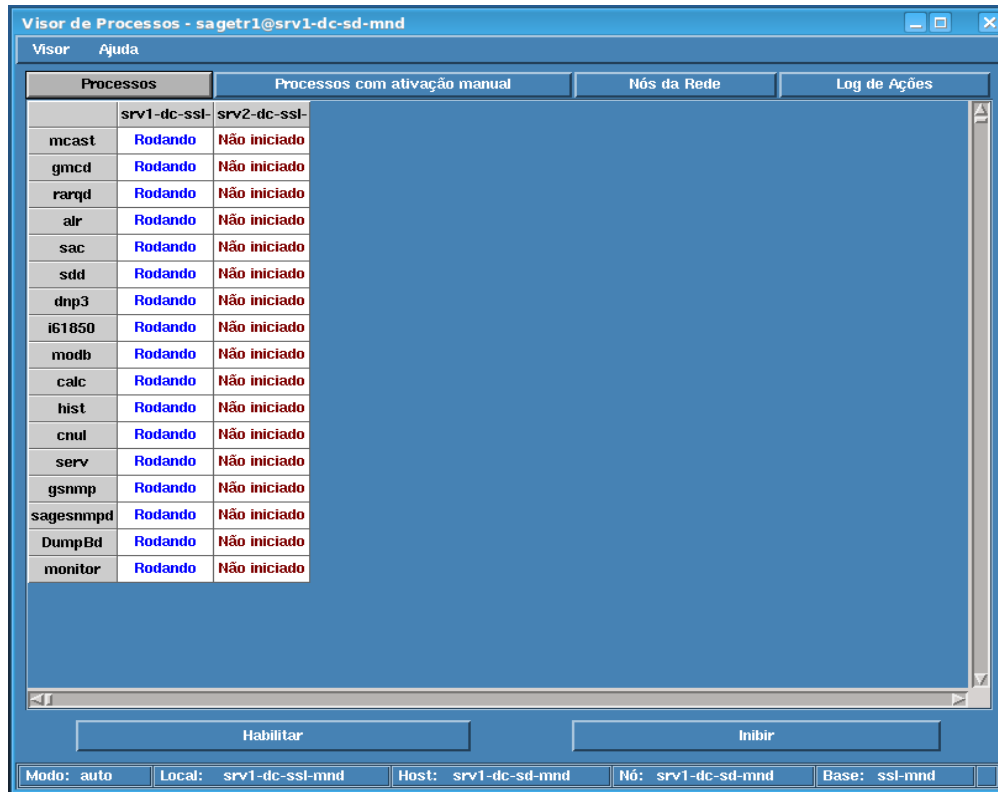
Figura 10 – Visor de Telas e de Alarmes no SCADA SAGE



Fonte: Elaborado pelo autor.

O Visor de Alarmes, o de *Logs* e de Processos, este último mostrado na Figura 11, são importantes para as observações do SAS, tanto no nível das supervisões de equipamentos das instalações, quanto dos processos em cada máquina, denominadas nós, do SAGE. Esse funcionamento correto depende dos subsistemas que contêm as funções SCADA. O SIG (Subsistema de Interface Gráfica) é responsável pelo gerenciamento dos visores, visto anteriormente. O SSC (Subsistema de Suporte Computacional) implementa a Base de Dados Tempo Real, mantendo a coerência entre as cópias dos arquivos nas diferentes máquinas da rede, monitora e controla os processos e trata os alarmes e eventos do sistema. O SCD (Sistema de Aquisição e Comunicação de Dados) é responsável pela interação entre os diferentes dispositivos e o SAGE, configura os protocolos de comunicação, tanto do nível 2 para o nível 1 como também para o nível 3 da arquitetura de uma SAS. Finalmente o STI (Subsistema de Tratamento de Informações) é responsável pela Base Fonte, realiza o seu tratamento e gera a Base Referência que é o conjunto de arquivos-texto com extensão .DAT, define e mantém os vários arquivos de interface com o SSC e também define e mantém a base de dados histórica (CEPEL, 2023f).

Figura 11 – Visor de Processos no SCADA SAGE



Fonte: Elaborado pelo autor.

A Base Fonte mantém todas as configurações do SAGE, reside num banco de dados relacional, pode ser preenchida pelo usuário e os relacionamentos entre as entidades (tabelas) são mostrados com mais detalhes no Anexo 9.1. As entidades são objetos que caracterizam pontos fundamentais para a organização dos dados que vêm do Nível 1 e que carregam toda modelagem de Nível 2 do SAS. A estrutura das tabelas da Base Fonte em relação aos Subsistemas do SAGE é mostrada a seguir:

a) Subsistema do Suporte Computacional (SSC):

- i. Rede de Difusão Confiável (entidades PRO, CXP, CTX, INM, INP, NOH, SXP, SEV);
- ii. Serviço de Alarmes e Eventos (entidades MAP, E2M, OCR).

b) Subsistema de Comunicação de Dados (SCD):

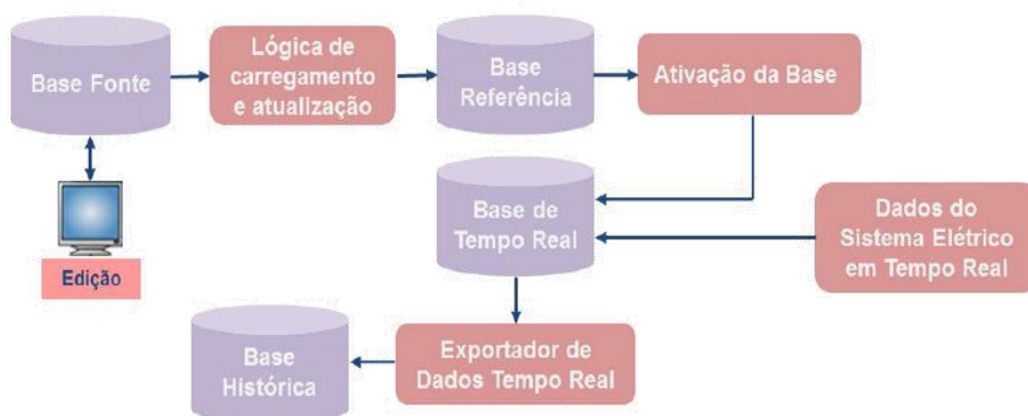
- i. Serviço de Comunicação de Dados (entidades TCV, TTP, GSD, CXU, ENU, UTR, MUL, ENM, CNM);
- ii. Serviço de Aquisição e Controle (entidades LSC, INS, TAC, CGS, PAS, PTS, PDS, RCA, TCL, TCTL);
- iii. Serviço de Distribuição de Dados (entidades TDD, PAD, PTD, PDD);

- iv. Pontos Físicos (entidades CNF, NV1, NV2, TN1, TN2, CGF, PAF, PTF, PDF, RFI).

Cada entidade contém atributos, que carregam suas características e informações armazenadas, podendo ser de preenchimento obrigatório ou opcional. O atributo corresponde a um campo da tabela. As entidades são associadas, através dos seus atributos, podendo se relacionar com alguma regra definida na base de dados relacional. O mapeamento das várias entidades e os atributos definidos na norma IEC 61850 são organizados nessas tabelas registradas com três/quatro letras e números. Essas tabelas são descritas de forma resumida, sem os atributos, no Apêndice 1, fazendo parte do banco de dados que configura e opera em tempo real todas as bases do SAGE (CEPEL, 2023e).

Na Figura 12, pode-se verificar a forma de edição da Base de Dados do SAGE. Primeiro na edição da Base Fonte, através de um programa específico de carregamento, gera-se a Base Referência, ela é atualizada em tempo real, sendo uma interface entre a Base Fonte e a Base Tempo Real. Através da ativação da Base Referência, o sistema começa a adquirir dados para Base de Tempo Real, que está *on-line*, e onde as aplicações do SAGE trocam informações dos dados do sistema elétrico supervisionado e, exportando para uma Base Histórica, que armazena as séries temporais e os históricos de eventos do sistema monitorado (CEPEL, 2023e).

Figura 12 – Relacionamentos e processos das bases de dados do SAGE



Fonte: Elaborado pelo autor.

2.4 Biblioteca libIEC61850

No presente trabalho, foi utilizada a versão 1.5.1 da biblioteca libIEC61850. Segundo Zillgith (2023), essa API é elaborada em código aberto, utilizando linguagem C (padrão C99), podendo ser executada em computadores, sistemas embarcados ou microcontroladores com plataformas Linux, Windows e MacOS. Essa biblioteca tem uma arquitetura para implementar aplicativos, no modelo Cliente/Servidor e Editor/Assinante, padrão IEC 61850. Operando como cliente, solicitando serviços e recursos, e como servidor processando aplicações, prestando serviços e armazenando dados.

A geração do tráfego deve ser compilada e codificada para atender aos objetivos que se propõem, sendo que uma interface gráfica não está presente no escopo da biblioteca. Quanto aos seus recursos, serão elencados os que foram utilizados na proposta desse trabalho:

- a) Cliente/servidor MMS, editor/assinante GOOSE (IEC 61850-8-1);
- b) Editor/assinante de valores amostrados (SV - IEC 61850-9-2);
- c) Serviço de acesso a dados (obter dados, definir dados);
- d) Descoberta e navegação de modelo de dados *on-line*;
- e) Todos os serviços de conjunto de dados (obter valores, definir valores, navegar);
- f) Serviços de conjunto de dados dinâmicos (criar e excluir);
- g) Serviços de arquivo MMS (navegar, obter arquivo, definir arquivo, excluir/renomear arquivo);
- h) GOOSE e manipulação de bloco de controle SV;
- i) Suporte TLS;
- j) API C e C#.NET;
- k) Exemplos de aplicações e implementações.

Na libIEC61850, também tem como intuito a comunicação por mensagens SV, por fazer parte de um dos pontos altos da norma onde envolve virtualização total dos dispositivos de proteção. A biblioteca disponibiliza uma série de exemplos que abordam os protocolos usados pela norma, um desses são as Instância Servidora (IS) e Instância Cliente (IC) de mensagens SV. Com a IS se pode desenvolver aplicações que são capazes de enviar mensagens de SV na rede de valores analógicos “sintéticos” de corrente e tensão elétricas. O algoritmo pode ser modificado para atender as especificações de criação e envio de frames SV, que serão capturados por IEDs físicos ou virtuais com suporte a IC.

As funções e os serviços utilizados na plataforma são apresentados nos capítulos posteriores, sendo descritos na forma de fluxogramas e códigos implementados pelo VIED. Conforme relatado em Zillgith (2023), existem diversas funções na biblioteca, que vão além da parte de comunicação das mensagens dos protocolos da Norma IEC 61850, como, por exemplo, a de autenticação de usuário por meio de senha, para se proteger de ataques cibernéticos, tema bem abrangente hoje em dia. Afinal, a segurança dos dados que trafegam no SAS é tão importante quanto à eficiência na proteção, no controle da automação elétrica.

2.5 *Merging Unit*

Como dito anteriormente, de forma resumida, as MUs são dispositivos de interface que interpretam as várias conexões analógicas, que são sinais oriundos de transformadores de corrente e de potencial, e digitais, que são as supervisões de estados e alarmes dos equipamentos de pátio. A MU executa o processamento de sinais para produzir vários dados em forma de pacotes sincronizados em *multicast* dentro do barramento de processos, respeitando os padrões amostrados transmitidos para os dispositivos de controle, automação e proteção de uma subestação (Lellys, 2020).

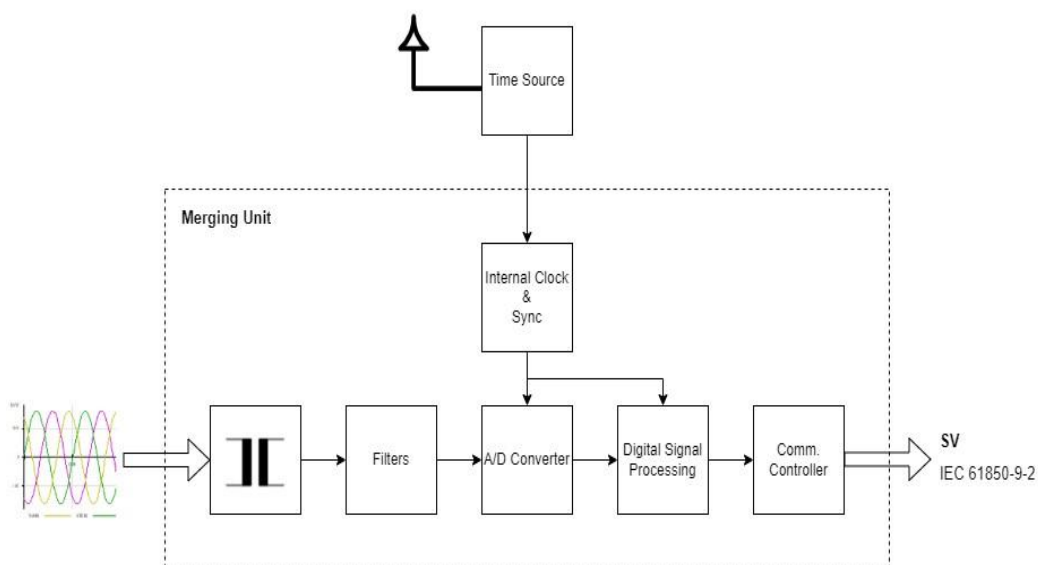
A MU, conforme mostrado na Figura 13, realiza o processamento de sinal analógico para fornecer valores amostrados de correntes de fase (I_a , I_b e I_c), tensões de fase (V_a , V_b e V_c), além da corrente residual (I_{res}) e tensão residual (V_{res}). Aliados a essa amostragem, os quadros de *sample values* são marcados com *timestamp* muito precisos, conforme sistema de sincronização, normalmente baseado em GPS que estabelece a mesma base de tempo com uma alta precisão entre diferentes dispositivos no sistema. A MU tem um formato compacto, podendo ser instalado no pátio da subestação, próximo aos equipamentos primários do vão. Com contatos extremamente rápidos, para comando de disjuntores, tem dois *slots* de placas analógicas com precisão tanto para medição operacional ou de faturamento, com até dois fluxos de SV, sendo 80 amostragens/ciclo para aplicações de proteção e 256 amostragens/ciclo para a qualidade de energia. Na comunicação, integra-se à subestação digital através de 2 portas *Ethernet* em fibra ótica (MU GE, 2022).

As MUs podem ser de tipos diferentes, de acordo com a característica de aquisição de dados. Os tipos referentes ao fabricante utilizado na plataforma são:

- a) RIO (*Remote Input/Output*) - utilizadas somente para aquisição de dados digitais, como as outras são RIO+SAMU;

- b) SAMU (*Stand Alone Merging Unit*) – converte os sinais analógicos dos Transformadores de Instrumentos (TCs e TPs);
- c) PIU (*Process Interface Unit*) – converte tanto analógicos como digitais.

Figura 13 – Diagrama de blocos da MU para aquisição de sinais analógicos



Fonte: CIGRE (2023).

Através dos dados digitais gerados pelas MUs e transmitidos para os IEDs de proteção, as funções de proteção implantadas em dispositivos físicos ou lógicos os utilizam para realizar, além do desligamento diante da detecção de faltas, os cálculos de impedância e outros controles para identificar e isolar problema, com uma grande eficiência.

Essa tecnologia muito contribui para redução dos cabos de cobre presentes ainda hoje nas subestações. Com a consolidação da norma e o avanço das tecnologias, será mais usual e comum as instalações de MUs próximas aos equipamentos do vão, recebendo os sinais analógicos e digitais, publicando em SV e GOOSE e compartilhando, através de fibra ótica, sendo esses dispositivos peças chaves para digitalização das subestações, tornando os sistemas de controle e proteção mais eficientes e confiáveis no SEP.

2.6 Redes de Comunicação

Das tecnologias que são mais relevantes para aplicações de automação de subestações na norma IEC 61850, as redes *Ethernet* evoluíram bastantes, principalmente no quesito de velocidade, sendo muito utilizadas para estabelecer as camadas físicas e de enlace dos pro-

jetos. Dentre as características principais, destaca-se o transporte de dados com a menor taxa de comunicação, sendo o modelo dominante nas arquiteturas das redes LAN. Para aplicações nas subestações de energia elétrica, os equipamentos da rede são diferentes dos utilizados em escritórios, em virtude de interferência eletromagnética, poeira, resíduos metálicos ou químicos em suspensão, vibrações, temperaturas elevadas ou muito baixas, com variações ao longo do dia e umidade em excesso (Hitachi, 2023).

Na rede *Ethernet* de subestações, tem uma seleção de equipamentos (*switches*, roteadores, computadores e cabos), sendo as especificações dos elementos de rede providos pelas normas IEC 61850-3 e a IEEE 1613. O tráfego da rede tem os seguintes atributos (Hitachi, 2023):

- a) As mensagens GOOSE, usadas para transmissão de sinais binários pela rede (alarmes, posição de equipamentos e comandos), e o SV, envio das grandezas elétricas de corrente e tensão, utilizam endereços MAC para identificar suas mensagens em faixas *multicast* definidas na norma IEC 61850;
- b) Na substituição de um IED defeituoso, o novo equipamento utiliza o mesmo mapa *multicast* configurado inicialmente na aplicação;
- c) As mensagens de tempo real são sensíveis a atrasos ou perdas de dados;
- d) Nenhum destes tráfegos tem confirmação de entrega.

Dos equipamentos da rede *Ethernet*, pode-se destacar a forma de atuação do *switch*, mais eficiente, seguro e com uma performance melhor do que o ultrapassado *hub* – ele trabalha com mais informações, pois opera na camada 2, ao invés da 1 do modelo OSI. O *switch* estabelece um filtro de MAC, cria filas para armazenar e enviar dados de acordo com a prioridade de tráfego e separa os domínios de colisão por porta. Esses equipamentos, para operar em ambiente do SAS, são homologados segundo a norma IEC 61850-3. De acordo com Vicente (2011), a característica de criar VLANs independentes na rede do SAS faz com que um IED, no envio de mensagens GOOSE, identificada a VLAN, trafegue somente nas portas identificadas e exclusivas, configuradas pelos *switchs*. Essa mensagem *multicast*, pelo seu formato, teria que trafegar em toda rede, mas irá somente trafegar na VLAN identificada na mensagem, melhorando o tráfego e reduzindo as colisões.

A redundância da *Ethernet* deve ser utilizada para o barramento de processo, evitando falta de mensagens ou mensagens perdidas. A *Ethernet* padrão não permite arquiteturas em anel, para evitar saturação da comunicação. Assim, o RSTP (*Rapid Spanning Tree Protocol*), HSR (*High-availability Seamless Redundancy*) ou PRP (*Parallel Redundance Protocol*) devem ser aplicados para garantir as aplicações nesse barramento de processos, de acordo

com a IEC TR 61850-90-4 (Hoga, 2010). O aspecto de confiabilidade de um sistema determinístico, no caso da rede *Ethernet*, pode ser quase alcançado nessa característica, por uma segregação física e lógica das VLANs, utilização de funções de Qos (Quality of Service) e criação de grupos *multicast* (Hitachi, 2023).

2.7 Proteção de Sistemas Elétricos de Potência

A proteção dos sistemas elétricos tem um papel fundamental para os requisitos de continuidade do fornecimento de energia, confiabilidade e seletividade nas ocorrências. A função dos relés é a retirada imediata de operação de qualquer elemento que sofra uma falta tipo curto-circuito ou opere de forma anormal, abrindo os disjuntores e desconectando o elemento defeituoso.

Os relés, que são equipamentos para cobrir defeitos em alta velocidade e com seletividade, têm várias funções que distinguem corrente e tensão, operando segundo a função de proteção configurada e conseguindo estimar a distância do local da falta. As funções de proteção são definidas pela *American National Standards Institute* (ANSI), sendo padronizados números que simbolizam as funções particulares de cada relé. Nesse trabalho irá resumir algumas características das funções que serão habilitadas e desenvolvidas no VIED, que, apesar da simplicidade do seu funcionamento, ainda são muito utilizadas nas proteções do SEP da Rede Básica. Na CHESF, por exemplo, tem a proteção de sobrecorrente da linha de transmissão, que fica predisposta e sendo habilitada caso a proteção principal, relé de distância, esteja desabilitada ou inoperante. Assim, a função 67 é considerada uma proteção de emergência da linha, com suas coordenações de tempo feitas para não prejudicar as demais proteções da linha (CHESF, 2012).

Recapitulando o conceito da proteção de sobrecorrente, é a denominação de esquema de proteção que atua com determinado aumento de corrente, sempre alinhado a um valor pré-ajustado. Segundo Kindermann (1999) e Silva (2011), tem três termos bem usuais em relés de proteção:

- a) Corrente de *Drop-out* é a maior corrente que desativa o relé, fazendo com que o relé retorne à posição inicial após um *trip* e cessada a falta;
- b) Corrente de *Pick-up* é a menor corrente que ativa o limiar de operação, fazendo com que o relé feche os contatos para enviar sinal para a abertura do disjuntor e o valor de corrente em que atua o relé, denominada corrente de *trip*;

c) TAP ou Faixas de Ajuste trata-se de derivações da bobina do relé (modelo eletromecânico) que permitem o ajuste para as correntes de atuação disponíveis.

A proteção mais simples é a de sobrecorrente instantânea (função 50), que atua instantaneamente para qualquer valor de corrente maior que seu ajuste. Apesar do nome “instantâneo”, a atuação ficará sujeita ao tempo dos algoritmos e lógicas de proteção, tempo de disparo de contatos auxiliares e tempo das bobinas de acionamento dos equipamentos comandados.

A proteção de sobrecorrente temporizada (função 51) atua após um certo tempo, depois que a corrente sobressai o valor de *pick-up*. Ela pode ser de tempo definido, em que uma vez ajustado o tempo de atuação, o relé irá operar diante do valor igual ou maior que o mínimo ajustado para a corrente. O outro caso é de tempo inverso, em que o tempo de atuação do relé é inversamente proporcional ao valor da corrente.

A proteção por tempo inverso, as curvas utilizadas de tempo e corrente são: curva inversa, moderadamente inversa, muito inversa e extremamente inversa. O tempo de operação, de acordo com essas curvas, é definido segundo a equação adaptada de Silva (2011):

$$t = T \left(L + \frac{K}{M^\alpha - \beta} \right)$$

Em que:

t – Tempo de atuação do relé;

T – Múltiplo de tempo da curva;

M – Múltiplo do relé (Corrente real/Corrente de ajuste);

K, L, α e β – Coeficientes da expressão, com valores seguindo a Tabela 4.

Tabela 4 – Valores das constantes de curvas da Norma IEC

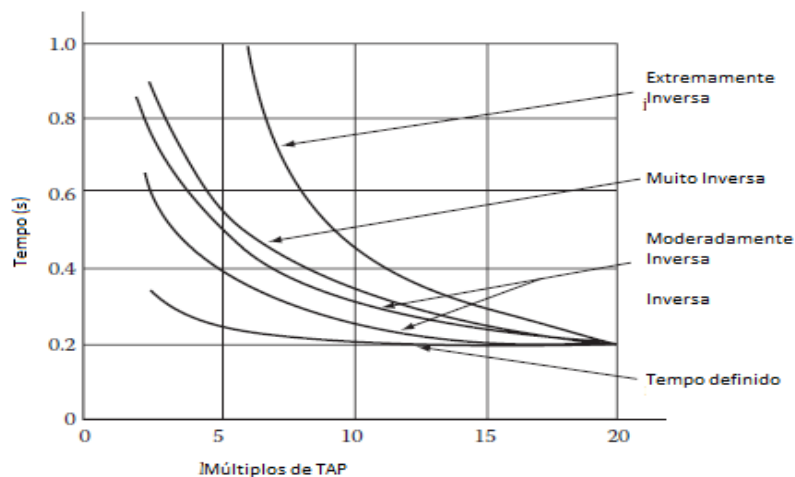
Tipo de Curva	K	α	L	β
Curva Inversa	0,14	0,02	0	1
Moderadamente Inversa	0,05	0,04	0	1
Muito Inversa	13,5	1	0	1
Extremamente Inversa	80	2	0	1

Fonte: Kindermann (1999, com adaptações).

As curvas de tempo inverso são demonstradas na Figura 14 e para o relé digital essas curvas são traçadas pela equação definida acima, que é uma expressão estabelecidas pela IEC, I²t e pela C.37.112-1996 IEEE. A curva de tempo inverso, atrelada à proteção com retardo de tempo (função 51), atua no sentido de que quanto maior a corrente de defeito mais rapi-

damente o relé atua, sendo escolhida a curva de acordo com a coordenação necessária do sistema a ser protegido (Kindermann, 1999).

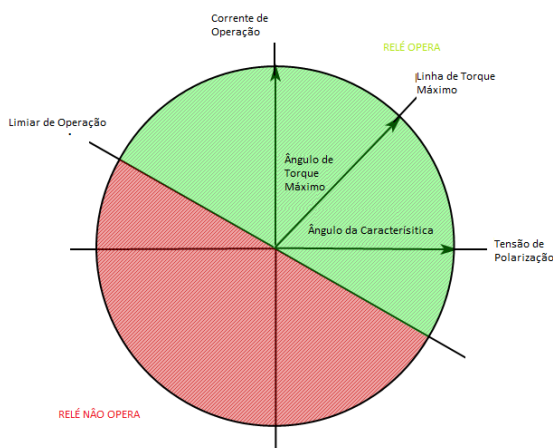
Figura 14 – Curvas IEC



Fonte: Elaborado pelo autor.

A proteção de sobrecorrente direcional (função 67) necessita que o equipamento esteja recebendo sinais analógicos tanto de tensão como de corrente. Esses equipamentos ficam polarizados na direção das falhas que devem proteger, e, algumas vezes, não correspondem à direção do fluxo normal de energia. A direcionalidade é dada pela comparação fasorial das posições relativas de atuação da corrente e a polarização dada pela grandeza de tensão. De acordo com a Figura 15, o ângulo da característica é o próprio direcional em si, pois decide a direção de operação do relé, a tensão de referência é a de polarização, e a linha do limiar define a área que o relé opera ou o seu disparo é inibido.

Figura 15 – Diagrama fasorial da proteção 67



Fonte: Babu (2019).

Quando a proteção atua de maneira correta, porém a corrente de curto-circuito não é extinta, entra em ação a função chamada falha de disjuntor (função 50/62BF), sendo executada por um relé de sobrecorrente (50) associado a um relé de tempo (62). Essa proteção é importante para resguardar o vão ou evento com falha, abrindo os disjuntores mais próximos e até em outras subestações, isolando todo o trecho no qual se alimenta o defeito.

O conceito das proteções e seus cálculos não mudaram com o passar dos anos. As equações dos relés eletromecânicos e eletrônicos vêm sendo utilizadas nos *firmwares* dos IEDs. A implementação desses *chips* mais complexos, multiprocessadores em arquiteturas de processador único, fez com que melhores desempenhos e confiabilidade ficassem evidentes e ainda mais funções fossem desenvolvidas com fácil acesso para parametrizações, o que será fomentado em outros trabalhos nessa área tão importante da engenharia.

2.8 Considerações Finais

Neste capítulo foram introduzidos os principais conceitos das novas redes de comunicação de um SAS, com a utilização expressiva da rede do tipo *Ethernet*, que é a que mais evoluiu no decorrer dos anos, principalmente no quesito velocidade de transmissão de dados. Essa característica facilitou a introdução da norma IEC 61850 como padrão de comunicação e aprimorou os sistemas de automação, com a interoperabilidade e a utilização de protocolos de mensagens GOOSE, MMS e SV entre fabricantes de equipamentos diferentes. No entanto, a crescente dependência do protocolo IEC 61850 gera preocupações sobre a segurança cibernética.

Atualmente, o SCADA SAGE é o *software* amplamente utilizado nas IHMs de SAS do SEP da Rede Básica pelas concessionárias de energia. As MUs estão sendo utilizadas como solução de aquisição de dados de campo para novas instalações, possibilitando muitas melhorias e ganhos de CAPEX, gerando avanços para subestações totalmente digitais. Ainda sobre as MUs, elas são os equipamentos necessários para uma virtualização de proteções de linhas de transmissão, transformadores e reatores. Funções de proteção como as de sobrecorrente instantânea, temporizada e direcional podem ser implementadas, com lógicas de programação baseadas em C e na biblioteca libIE61850, possibilitando trabalhar em conjunto os VIEDs e IEDs, auxiliando inclusive na eliminação de faltas em situações de falha de disjuntor.

3 FUNDAMENTOS DO VIED DE PROTEÇÃO E CONTROLE

Uma tecnologia que vem sendo estudada e ampliada é a virtualização de dispositivos, pois entende-se que os IEDs baseados em *hardware* têm uma tendência de obsolescência, ciclo de vida mais restrito e, em muitos casos, exigem uma grande quantidade de dispositivos para proteger setores de um SEP, o que está se tornando cada vez mais complexo. VMs utilizam um sistema operacional ou aplicativo ambiente que está instalado no *software*, imitando *hardware* dedicado. Uma máquina virtual é um arquivo de computador (normalmente chamado de imagem) que se comporta como um computador de verdade ela pode ser executada em uma janela como um ambiente computacional separado, geralmente para executar um sistema operacional diferente, ou até mesmo para funcionar somente um aplicativo. A VM é particionada do restante do sistema, o que significa que o software dentro de uma VM não pode interferir no sistema operacional primário do computador *host*. As VMs são muito comuns no mundo da Tecnologia de Informação (TI), softwares como o VMWare, ESXi e KVM estão em cerca de 75% dos sistemas de trabalho virtualizados em servidores, no entanto, na automação e controles, menos de 30% dos sistemas são virtualizados, com base em pesquisa da empresa *General Electric* (Jesus, 2023). Outra tecnologia que está em fase de implantação são as baseadas em containeres, utilizado software como o docker, que é uma unidade padrão de software que armazena todos os códigos e dependências para que um aplicativo seja executado de maneira rápida e confiável. Armazenado em pacotes de software leves, o software containerizado sempre será executado da mesma maneira, independente da infraestrutura e do sistema operacional do aplicativo e do *host* (Docker, 2023), essa tecnologia é mais leve que as VMs clássicas, com um bom suporte a escalabilidade.

O SEP com diversas fontes e cargas de características intermitentes, com contribuições que afetam o sentido e características do fluxo de potência, tensões e frequência, gera uma necessidade de mudança de parâmetros – habilitar ou desabilitar determinadas funções de proteção e lógicas de controle em determinados momentos do dia. Com isso, a necessidade de modelos mais adaptáveis tem conduzido ao desenvolvimento dos *Virtual Intelligent Electronic Device*, ou soluções baseadas em plataforma computacional que agregam diversas máquinas virtuais (Wójtowicz *et al.*, 2018).

Com os VIEDs, pode-se desenvolver, através de *software*, IEDs que contém funções de proteção e controle, aquisitando os sinais digitais provenientes de uma MU, e centralizado as tomadas de decisão de forma dinâmica e customizável. Os VIEDs podem, sem troca de *hardware*, ser modificados e adaptados de acordo com a necessidade da instalação ou em

um contexto mais amplo, do próprio SEP. O modelo de dispositivos de controle e proteção baseados em *software* diz respeito a aplicativos que podem ser rapidamente alterados, desenvolvidos e testados, o que pode aumentar a confiabilidade dos processos e as melhores condições de manutenções (Freitas, 2022a). A centralização, que seria uma agregação das funções de múltiplos IEDs, em uma única plataforma computacional, está em estudo para aproveitar as tecnologias de virtualização, utilizando de VMs e tecnologia de *container*, atendendo às necessidades de subestações em constante evolução (Myrda, 2021).

Magalhães (2019) desenvolveu uma plataforma para testes de comunicação com IEDs físicos por meio de mensagens MMS e GOOSE. A comunicação foi feita por meio de *software*, que permitia a leitura de dados e atributos de um *report* enviado via MMS e a troca de mensagens GOOSE com os IEDs. A contribuição do autor consistiu em criar algoritmos em *softwares* capazes de se comunicarem com dispositivos físicos utilizando dois protocolos básicos padrão IEC 61850.

Freitas (2022b) desenvolveu uma plataforma de proteção utilizando IEDs físicos e virtuais. O foco da autora consistiu em validar os dispositivos virtuais e realizar a sua integração com o *software* LAQUIS, que é um sistema SCADA de licença livre. A plataforma do trabalho de Freitas (2022a) consistiu na demonstração da operação dos relés virtuais para a realização das comunicações MMS e GOOSE e na sua integração ao SCADA ELIPSE.

Segundo Kulmala *et al.* (2023), que desenvolveram a partir de um projeto multi-fabricante denominado *constellation*, é demonstrado a utilização de proteção e controle virtualizado em uma arquitetura centralizada. Nesse projeto, diversas soluções utilizam um mesmo hardware e infraestrutura de comunicação, sendo de extrema importância buscar a melhor solução de virtualização, para essas operações de proteção em tempo real que tem requisitos de confiabilidade muito rigorosos. Aliado a essa característica, é necessário o requisito de baixa latência da rede, para que se tenha consistência nos sinais digitalizados de corrente e tensão. Também é importante mecanismos de disponibilidade de todo potencial de processamento de hardware e memória, quando as aplicações estiverem executando os algoritmos de cálculo no tempo necessário para atuação das proteções.

Wójtowicz *et al.* (2022) apresentou uma concepção geral de virtualização utilizando soluções de sistemas de tecnologia de informação convencionais, como uma versão Linux CentOS 7.5 e máquinas virtuais baseadas no KVM v2.10. Utilizando uma plataforma de hardware não muito atual, como servidor, baseada na arquitetura x86, o sistema de proteção virtual e o ambiente desenvolvido em Linux, os testes tiveram resultados satisfatórios na implementação de um sistema de proteção de sobrecorrente simulando uma subestação com

vários vãos, com tempos de resposta muito bons, quando comparado com os mesmos testes em Windows 7, onde se constatou a fraca capacidade de resposta desse sistema gerando maior latência dos dados. As possibilidades para se utilizar um conjunto maior de VMs, aliado a velocidade de cálculo das novas CPUs de servidores atuais, irão permitir o desenvolvimento de funções de proteção para uma subestação bem mais complexa.

Este trabalho demonstrou que os IEDs virtuais desenvolvidos pelo GREI ainda poderiam avançar em sua complexidade e serem aplicados ao ensino, enquanto plataforma PAC.

O VIED proposto teve como base a API da libIEC61850 (ZILLGITH, 2021), seus LNs foram criados a partir de IEDs físicos, utilizados em trabalhos anteriores do GREI, os quais aproveitavam suas funções e integravam o dispositivo a um tipo de plataforma de testes. Foram feitas modificações em códigos-base fornecidos pela biblioteca da API, para criar relés virtuais com funções de proteção e controle, integrando no supervisório SCADA SAGE. Esse supervisório simula uma subestação da Eletrobrás CHESF conectada à Rede Básica

As funções de proteção implementadas e testadas foram instantâneas de sobrecorrente (50), instantânea de sobrecorrente de neutro (50N), sobrecorrente temporizada (51), sobrecorrente temporizada de neutro (51N), sobrecorrente temporizada com restrição de tensão (51V), sobrecorrente direcional temporizada (67), sobrecorrente direcional temporizada de neutro (67N) e falha de disjuntor (50BF). Foram feitos também testes de controle de abertura e fechamento de disjuntor pelos Níveis 1 (VIED) e 2 (SCADA SAGE).

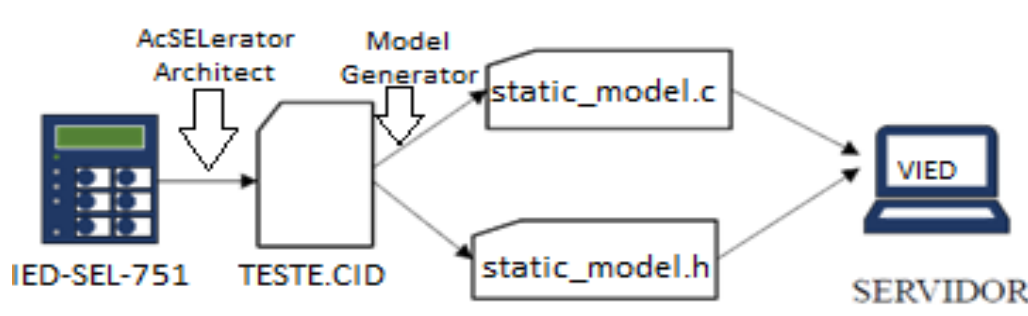
3.1 Criação de Arquivos para os VIEDs

Para aplicação de servidor padrão IEC 61850, a virtualização de um IED é obtida através de um arquivo do tipo: CID, ICD ou SCD extraído de um dispositivo real. Dentre estes arquivos, o formato mais recomendado seria o CID, pois este, quando feito o processo de virtualização, já contempla todas as configurações realizadas no IED (Magalhães, 2019).

Foi utilizado o arquivo denominado TESTE.CID gerado de um IED SEL-751, com o auxílio do *software AcSELerator Architect* da fabricante SEL. O arquivo do tipo CID foi configurado de forma que seja possível utilizá-lo em todas as variações de VIEDs aplicados nesse trabalho. Cada VIED baseado no arquivo CID executa uma função de proteção e controle. Na Figura 16, visualizamos o processo que, junto à biblioteca libIEC61850, utilizando a ferramenta gerador de modelos (*model generator*), cria os arquivos que serão utilizados pelas aplicações em C, em conjunto com a biblioteca. São os arquivos *static_model.c* e

static_model.h, servindo de base de toda virtualização a ser executada (ZILLGITH, 2022). Os arquivos e subpastas que serão utilizadas no decorrer dos testes com VIEDs, estão sendo relacionados no Anexo 9.1, e distintos por cada função de proteção testada.

Figura 16 – Arquivos para criação de uma instância servidora na libIEC61850



Fonte: Freitas (2022a, com adaptações).

3.2 Funções de Supervisão e Controle implementadas nos VIEDs

Com o modelo de dados já definido pela Norma IEC 61850, a troca de informações para supervisão e controle do VIED com a MU e com o SCADA SAGE, de acordo com a parte 7.1, pode ser feita pelas comunicações do tipo cliente-servidor, por meio de mensagens MMS e editor-assinante, por meio de mensagens GOOSE.

3.2.1 Mensagens MMS

O protocolo de comunicação MMS envia e recebe mensagens de maneira vertical com os serviços *Get/Set Operate* e de *Dataset*. As funções de instância servidora (IS), de prefixo *IedServer*, são voltadas à alteração/atualização dos valores de atributos de dados, conforme o tipo (inteiro, decimal, *string*, entre outros). A conexão do servidor está disponível e os clientes podem realizar aquisição/alteração de dados da estrutura interna, além de leitura e monitoramento de *datasets*. Os RCB (*Report Control Block*) cadastrados atualizam os dados disparando *reports*, sendo utilizados nos clientes que estiverem monitorando os *datasets*.

Na Figura 17, é apresentada a estrutura de um VIED como uma IS MMS, e os pontos que precisam ser ativos no SCADA SAGE, para que a IC MMS consiga fazer requisições de *reports* ao servidor. As requisições utilizadas nesses pontos são os estados de atuação das funções de proteção identificadas por fase, comandos de controle da IC MMS para IS MMS e medições analógicas.

Figura 17 – Código para iniciar a IS MMS no VIED

```

IedServer_start(iedServer, 102);

IedServer_setControlHandler(iedServer, IEDMODEL_CON_RBGGIO1_SPCS001, (ControlHandler) controlHandlerForBinaryOutput,
IEDMODEL_CON_RBGGIO1_SPCS001);

IedServer_setControlHandler(iedServer, IEDMODEL_CON_RBGGIO1_SPCS002, (ControlHandler) controlHandlerForBinaryOutput,
IEDMODEL_CON_RBGGIO1_SPCS002);

IedServer_setControlHandler(iedServer, IEDMODEL_CON_RBGGIO1_SPCS003, (ControlHandler) controlHandlerForBinaryOutput,
IEDMODEL_CON_RBGGIO1_SPCS003);

IedServer_setControlHandler(iedServer, IEDMODEL_CON_RBGGIO1_SPCS004, (ControlHandler) controlHandlerForBinaryOutput,
IEDMODEL_CON_RBGGIO1_SPCS004);

IedServer_setControlHandler(iedServer, IEDMODEL_CON_RBGGIO1_SPCS005, (ControlHandler) controlHandlerForBinaryOutput,
IEDMODEL_CON_RBGGIO1_SPCS005);

IedServer_setControlHandler(iedServer, IEDMODEL_CON_RBGGIO1_SPCS006, (ControlHandler) controlHandlerForBinaryOutput,
IEDMODEL_CON_RBGGIO1_SPCS006);

IedServer_setControlHandler(iedServer, IEDMODEL_CON_RBGGIO1_SPCS007, (ControlHandler) controlHandlerForBinaryOutput,
IEDMODEL_CON_RBGGIO1_SPCS007);

```

Fonte: Elaborado pelo autor.

Com a criação específica do arquivo CID, os LN atualizados da função de proteção escolhida a sensibilizar o SCADA vai ser responsável por verificar e informar essa atuação, atualizar a base de dados e a tela de alarmes do SCADA SAGE. Essa parte do código é apresentado na Figura 18, onde o LN de sinal de trip geral (TRIPPTRC1) é modificado o seu atributo general do dado *Tr* (verdadeiro ou falso), de acordo com o disparo de alguma função de proteção sensibilizada.

Figura 18 – Código de atualização de LN de sinalização de *trip* geral

```

IedServer_updateBooleanAttributeValue(iedServer, IEDMODEL_PRO_TRIPPTRC1_Tr_general, true);

IedServer_updateBooleanAttributeValue(iedServer, IEDMODEL_PRO_TRIPPTRC1_Tr_general, false);

```

Fonte: Elaborado pelo autor.

Os LNs referentes à medição são padronizados em todos os algoritmos dos diferentes VIEDs. Na Figura 19 são apresentadas as linhas de mensagem que são capturados pelo cliente SCADA SAGE. Elas contêm os LNs (METMMXU1), com os dados de medições analógicas de magnitude de corrente (*A_phsA*, *A_phsB*, *A_phsC*) e tensão de linha (*PPV_phsAB*, *PPV_phsBC*, *PPV_phsCA*). Os atributos para cada fase estão armazenados em *cVal_mag_f*.

Figura 19 – Código de atualização de LN de medições analógicas

```

IedServer_updateFloatAttributeValue(iedServer, IEDMODEL_MET_METMMXU1_A_phsA_cVal_mag_f, corrente_primarioA);
IedServer_updateFloatAttributeValue(iedServer, IEDMODEL_MET_METMMXU1_A_phsB_cVal_mag_f, corrente_primarioB);
IedServer_updateFloatAttributeValue(iedServer, IEDMODEL_MET_METMMXU1_A_phsC_cVal_mag_f, corrente_primarioC);
IedServer_updateFloatAttributeValue(iedServer, IEDMODEL_MET_METMMXU1_PPV_phsAB_cVal_mag_f, tensao_primarioA);
IedServer_updateFloatAttributeValue(iedServer, IEDMODEL_MET_METMMXU1_PPV_phsBC_cVal_mag_f, tensao_primarioB);
IedServer_updateFloatAttributeValue(iedServer, IEDMODEL_MET_METMMXU1_PPV_phsCA_cVal_mag_f, tensao_primarioC);

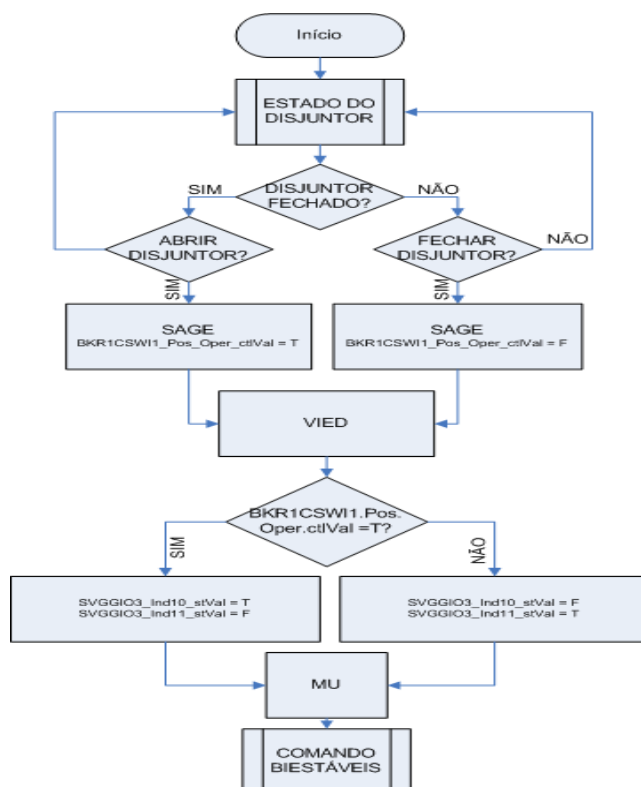
```

Fonte: Elaborado pelo autor.

Para o controle existe o atributo de dado, chamado de *ctlVal* do LN *BKR1CSWI1*, que atua no comando do disjuntor com as ações do SCADA. O tipo de dado é definido como *Controllable Single Point (SPC)*, tendo ações sobre o equipamento, com a mudança do atributo de *true* para *false* (fechamento) e de *false* para *true* (abertura).

Na Figura 20 é mostrado fluxograma referente ao controle, partindo do estado do disjuntor até o comando final nos relés biestáveis. Os estados dos atributos vinculados as mensagens GOOSE e MMS são modificados de acordo com as ações que foram tomadas de comando. Pelo o SCADA SAGE, a mensagem MMS do LN *BKR1CSWI1* com o atributo *Pos_Oper_ctlVal* é diagnosticado pelo VIED, o momento que é alterado, é modificado os atributos de dados da mensagem GOOSE para a MU, que comanda um pulso nas suas saídas digitais, efetuando comandos de abertura e fechamento de disjuntor.

Figura 20 – Fluxograma de controle



Fonte: Elaborado pelo autor.

3.2.2 Mensagens GOOSE

Podem se enviar e receber mensagens de maneira horizontal, utilizando o protocolo GOOSE e os serviços relativos ao GOOSE *ControlBlocks* (SendGOOSEMessage, GetGoCBValues, SetGoCBValuesOperate). Para configurar quais mensagens GOOSE o VIED envia, na elaboração do arquivo tipo CID, os parâmetros de configurações de comunicação IEC 61850 devem ser mencionados no *software* do IED a ser virtualizado. Na Figura 21, pode ser vista a configuração do VIED feita para habilitá-lo a publicar e assinar mensagens GOOSE, esse código também serve para assinar mensagens oriundas da MU.

Primeiramente, é feita a assinatura de mensagens GOOSE da MU por parte do VIED, pois nessa configuração a MU verifica o estado do relé biestável instalado na bancada com uma entrada digital, processa esse dado e transmite, em forma de mensagem GOOSE, a posição ao disjuntor simulado ao VIED.

Figura 21 – Código de habilitação do VIED para publicar e assinar mensagens GOOSE

```
IedServer_enableGoosePublishing(iedServer);

GooseReceiver receiver = GooseReceiver_create();

GooseReceiver_setInterfaceId(receiver, "p5p1");

GooseSubscriber subscriber = GooseSubscriber_create("MU320CTRL/LLN0$GO$FastGOOSE1", NULL);

GooseSubscriber_setlistener(subscriber, gooseListener, iedServer);

GooseReceiver_addSubscriber(receiver, subscriber);

GooseReceiver_start(receiver);
```

Fonte: Elaborado pelo autor.

O VIED assina essa mensagem que retorna o *status* do equipamento, que de acordo com a parte 7-3 da norma IEC 61850, é o atributo de dado *stVal*. Ele é definido do tipo *Double Point Status (DPS)*, com a informação de quatro possíveis estados (aberto, fechado, em transição e inválido). Esse atributo de dado está presente no LN (BK1XCBR1), sendo mostrado no código da Figura 22, a sequência dessa informação do VIED para o SCADA SAGE.

Figura 22 – Código do VIED onde assina mensagens GOOSE

```

if(b == 116){
    IedServer_updateDbposValue(iedServer, IEDMODEL_PRO_BK1XCBR1_Pos_stVal, DBPOS_OFF);
    IedServer_updateBooleanAttributeValue(iedServer, IEDMODEL_ANN_SVGGI03_Ind11_stVal, false);
}

else{
    IedServer_updateDbposValue(iedServer, IEDMODEL_PRO_BK1XCBR1_Pos_stVal, DBPOS_ON);
    IedServer_updateBooleanAttributeValue(iedServer, IEDMODEL_ANN_SVGGI03_Ind10_stVal, false);
}

```

Fonte: Elaborado pelo autor.

3.2.3 Mensagens SV

A MU fornece mensagens SV na rede de comunicação e o VIED precisa capturar essas mensagens e interpretá-las de forma específica, para que no final possam ser fornecidos valores de módulo e ângulo para cada fase de corrente e tensão. Na norma IEC 61850-9 parte 2, foi definido o que são essas mensagens SV e, a partir desses documentos, pode ser criada a lógica para traduzir esses *bits* em valores reais. A mensagem SV, como a GOOSE, possui requisitos críticos de tempo, na geração e no envio de mensagens, e uma rede de comunicação *Ethernet* é a melhor solução para suportar esse tráfego. A diferença entre o GOOSE e o SV é que os pacotes de mensagens SV não têm repetição de envio. Essas mensagens SV são responsáveis pela principal parte do tráfego nessa rede de uma subestação digital (Eggers,2019).

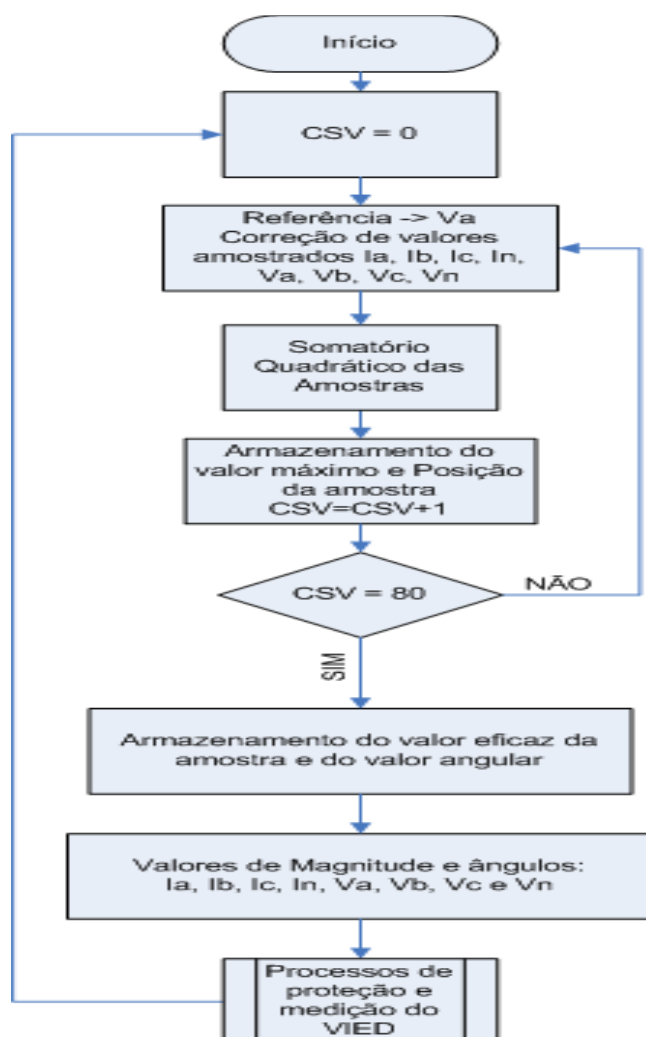
Cada *frame* da mensagem SV contém 4 amostras de corrente (I) e 4 de tensão (V). Divididos em 4800 amostras por segundo, contabilizadas por um atributo chamado sample Count (*smpCnt*). Para a proteção, são enviadas 80 mensagens (amostras) por ciclo (16,6 ms). Na medição ou para oscilografias, uma melhor resolução se faz necessária, logo são realizadas 256 amostras por ciclo. Na Figura 23, é vista a estrutura básica de dados SV que é chamada de *Application Service Data Unit* (ASDU), contém dados importantes para sincronização como o contador de amostras (*smpCnt*) e a identificação da mensagem (*svID*).

A norma permite a concatenação de ASDU em um *Application Protocol Data Unit* (APDU), podendo um DataSet conter de 1 ou 8 ASDU. Alocada na APDU, que armazena nos *buffers* de transmissão ou de recepção da mensagem, a estrutura ASDU identifica o que está presente em cada *byte* dessa mensagem, os valores instantâneos de corrente e tensão

mazenados os valores instantâneos de magnitude de corrente de cada fase e de neutro, em somatórios quadráticos. Depois de verificado o maior valor das 80 amostras, é armazenada o *smpCnt*. Todas essas variáveis que serão utilizadas na segunda parte do código.

Na etapa seguinte, verifica-se onde capturamos novamente a amostra que tem o máximo valor e se compara com a que foi armazenada anteriormente. Caso o valor máximo da amostra seja maior, é capturado o contador dessa mensagem SV. Após terminar o ciclo de cada fase, é verificada a diferença dos contadores das outras fases com a fase de referência, e o resultado é multiplicado por 4,5. Esse valor de 4,5 é o resultado do ângulo de 360° de um ciclo, dividido pelas 80 amostras aquisitadas. Com os valores de magnitude e ângulo calculados, são utilizados num vetor que é utilizado em todas as outras funções do VIED. Essa lógica funciona de maneira ininterrupta e podem ser observadas de forma mais específica, com expressões matemáticas e caracterização dos vetores, no apêndice 8.2.

Figura 24 – Fluxograma da medição de corrente e tensão pelo SV



Fonte: Elaborado pelo autor.

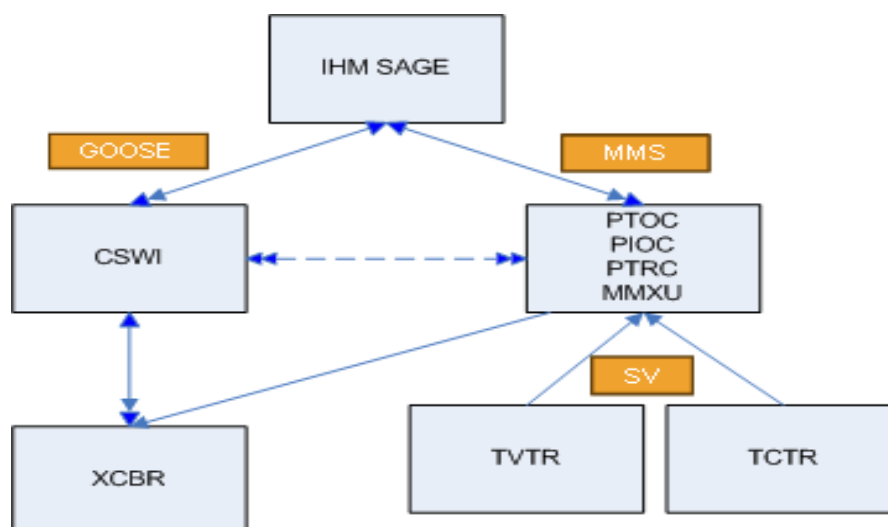
3.3 Funções de Proteção Implementadas nos VIEDs

O VIED proposto foi configurado de forma que seja possível fazer disparos de *trip* e gerar sinalizações para que o SCADA SAGE receba informações de qual proteção atuou, nas mesmas condições que faria um IED físico.

O IED físico no qual utiliza-se como base para criação do VIED não tem suporte à sinalização de todas as proteções via LN específicos, definindo qual fase atuou no último disparo. Isso fere os requisitos do Procedimento de Rede do ONS, Item 8 do Submódulo 2.7, que relata o sequenciamento de eventos, indicando a necessidade da distinção por fases de um evento cuja proteção foi sensibilizada (ONS, 2023). No entanto, com alguns ajustes na programação do VIED, sua configuração de comunicação IEC 61850 o tornou capaz de seguir a regra exigida pelo ONS e ter esse funcionamento adequado e demonstrado em conjunto com o SCADA SAGE.

São demonstradas, no fluxograma da Figura 25, as interações entre os LNs e as mensagens presentes na plataforma de testes. Os LNs PTOC, PIOC, PTRC e MMXU são habilitados para serem servidor MMS, enviando mensagens a IHM do SCADA SAGE. O VIED assina e edita mensagens GOOSE, também recebe controles de abertura e fechamento do nível 2 (CSWI) para o disjuntor XCBR. O suporte a mensagens SV é feito com o recebimento das medições de corrente (TCTR) e tensão (TVTR) via rede de comunicação enviada pela MU. A diferença entre as funções de proteção se dá nos LN específicos de cada uma.

Figura 25 – Fluxograma das funções e interações



Fonte: Elaborado pelo autor.

O arquivo CID que foi utilizado para gerar o VIED, também é utilizado para elaboração da base de dados do SCADA SAGE, incluindo os novos pontos a serem supervisionados. Na ferramenta Tabular digital do SAGE ficam expostos todos os pontos criados, que vão além das nossas necessidades para os testes que serão executados. Para esses testes, foram utilizados os pontos cadastrados na Tabela 5, com o endereço e a descrição da mensagem.

Tabela 5 – Lista de pontos com as mensagens no padrão IEC61850

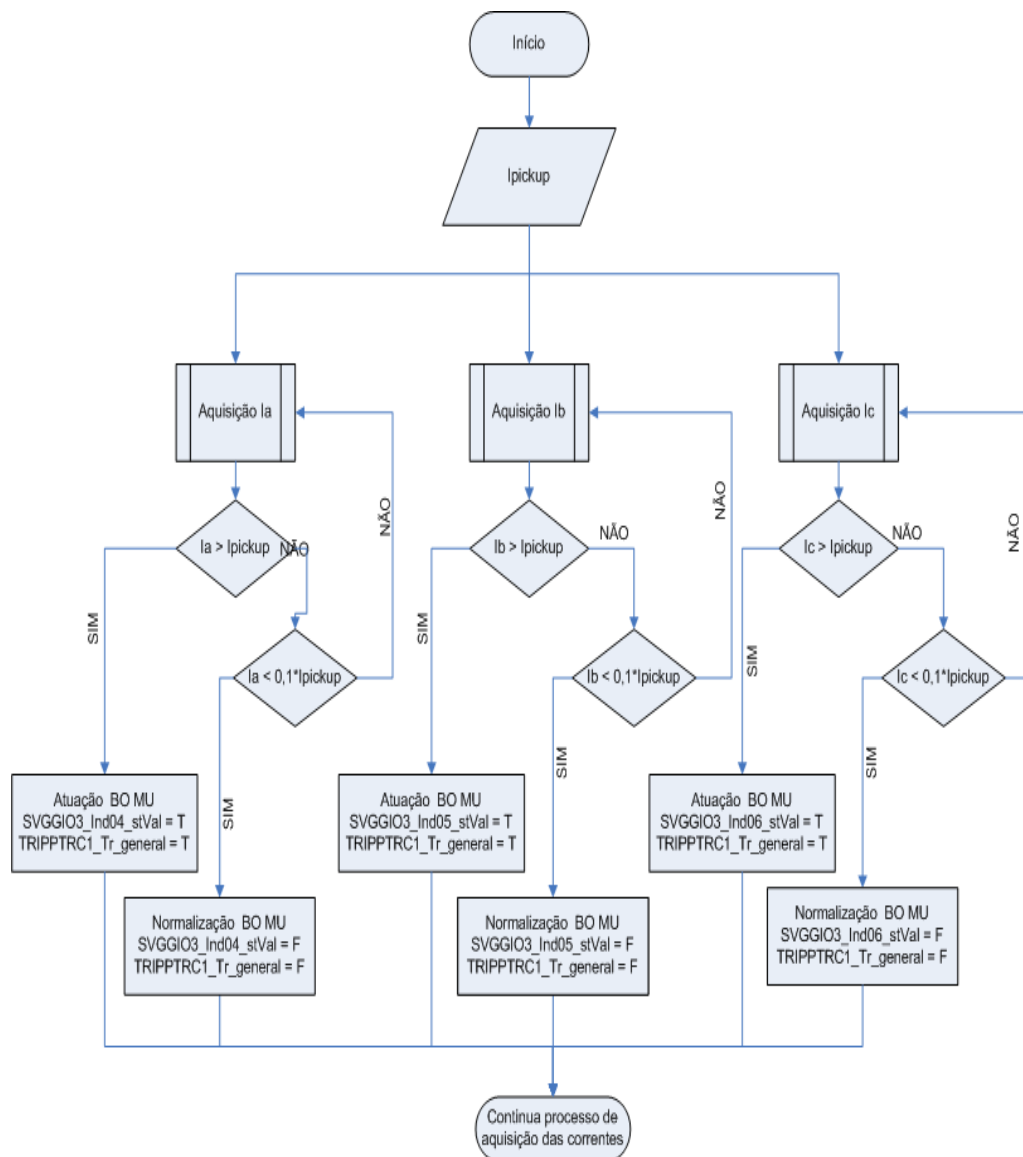
Item	Mensagem MMS/GOOSE/SV	Descrição
1	i12-TRIPTRC1\$ST\$TR	<i>Trip</i> Geral
2	i12-BK1XCBR1\$ST\$POS	Posição Disjuntor
3	i12-CTPTOC20\$ST\$OP\$GENERAL	<i>Trip</i> 51 Fase A
4	i12-CTPTOC21\$ST\$OP\$GENERAL	<i>Trip</i> 51 Fase B
5	i12-CTPTOC22\$ST\$OP\$GENERAL	<i>Trip</i> 51 Fase C
6	I15-SVGGIO3\$ST\$IND01	<i>Trip</i> 51V Fase A
7	I15-SVGGIO3\$ST\$IND02	<i>Trip</i> 51V Fase B
8	I15-SVGGIO3\$ST\$IND03	<i>Trip</i> 51V Fase C
9	I15-SVGGIO3\$ST\$IND04	<i>Trip</i> 50 Fase A
10	I15-SVGGIO3\$ST\$IND05	<i>Trip</i> 50 Fase B
11	I15-SVGGIO3\$ST\$IND06	<i>Trip</i> 50 Fase C
12	I15-SVGGIO3\$ST\$IND07	<i>Trip</i> 67 Fase A
13	I15-SVGGIO3\$ST\$IND08	<i>Trip</i> 67 Fase B
14	I15-SVGGIO3\$ST\$IND09	<i>Trip</i> 67 Fase C
15	I15-SVGGIO3\$ST\$IND10	Estado disjuntor aberto
16	I15-SVGGIO3\$ST\$IND11	Estado disjuntor fechado
17	I15-SVGGIO3\$ST\$IND12	<i>Trip</i> 67N
18	I15-SVGGIO3\$ST\$IND13	<i>Trip</i> 51N
19	I15-SVGGIO3\$ST\$IND14	<i>Trip</i> 50N
20	i13-METMMXU1\$MX\$PPV\$phsAB-M	Fase AB de Tensão
21	i13-METMMXU1\$MX\$PPV\$phsBC-M	Fase BC de Tensão
22	i13-METMMXU1\$MX\$PPV\$phsCA-M	Fase CA de Tensão
23	i13-METMMXU1\$MX\$A\$phsA-M	Fase A de Corrente
24	i13-METMMXU1\$MX\$A\$phsB-M	Fase B de Corrente
25	i13-METMMXU1\$MX\$A\$phsC-M	Fase C de Corrente

Fonte: Elaborado pelo autor.

3.3.1 Função de Sobrecorrente Instantânea de Fase (50)

Através da estrutura padrão IEC 61850, os arquivos *static_model.c* e *static_model.h* foram gerados e alterado para as funcionalidades da proteção 50. O fluxograma de funcionamento do código do VIED é demonstrado na Figura 26. O código foi editado no *software Visual Code* e esse arquivo será utilizado como base em todos os testes com os diferentes VIEDs. O algoritmo elaborado na Linguagem C se encontra na íntegra no Apêndice 8.2.

Figura 26 – Fluxograma da lógica da proteção 50 trifásico



Fonte: Elaborado pelo autor.

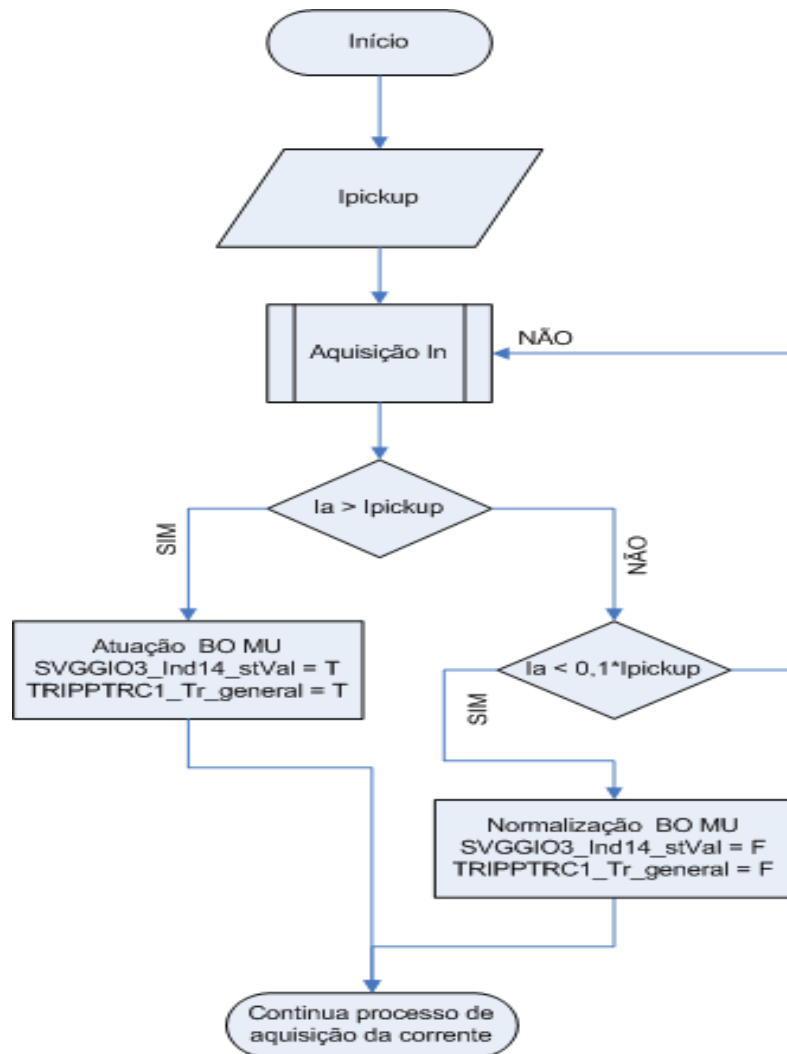
Para a lógica da função de proteção 50, a informação principal é o valor dado pela variável *pick_up*, essa função inicial é executada apenas uma única vez, auxiliada pela variável booleana denominada *resposta*, pois quando o algoritmo do VIED é iniciado e após a informação de *pick up*, ele entra em um *loop* infinito onde ficará sendo executada a lógica de proteção. Os valores de corrente são processados pelas outras funções internas, já caracterizadas, quando abordadas as mensagens SV. Os valores são testados e a atuação vai depender, se um dos valores das fases no primário estão maiores que a variável *pick_up*. Com isso, ocorre a atualização dos atributos de dados booleanos, é modificado para *true* o atributo de dados *Tr_general* do LN PTRC. Na sequência o LN SVGGIO3, que tem os atributos de dados Ind04, Ind05 e Ind06, para o disparo das fases A, B e C, respectivamente, é também modificado para *true*. Esses atributos de dados estão vinculados a um *Dataset* que, por sua vez, está vinculado a um *Report*, sendo informado ao SCADA quando ocorrer uma mudança de estado, levando a alarmar com um sonoro e sinalizar na tela de alarmes do SCADA SAGE.

Por fim, mesmo depois de acontecer o *trip*, o relé continua seu *loop* infinito, checando os valores aquisitados de corrente. Na última parte do algoritmo, tem a sessão pós atuação da função de proteção. É checada se a corrente presente nas fases é inferior a 10% do valor da variável *pick_up*, esse valor foi estabelecido para que o VIED faça a função de *drop-out*, que é quando o *trip* já foi cessado e o VIED retorna seus atributos de dados dos LNs informados anteriormente para o estado de *false*, consequentemente o SCADA SAGE será informado que a falta foi cessada.

3.3.2 Função de Sobrecorrente Instantânea de Neutro (50N)

Da mesma forma que a função anterior, o valor de *pick-up* de neutro deve ser informado. A Figura 27 apresenta o fluxograma da lógica de programação. Quando o programa é compilado, ele entra em *loop* infinito e executa a lógica de proteção para o 50N. O LN SVGGIO3 tem o atributo de dado Ind14 que serve para sinalizar, no SCADA SAGE, o disparo dessa proteção.

Figura 27 - Fluxograma da lógica da proteção 50N

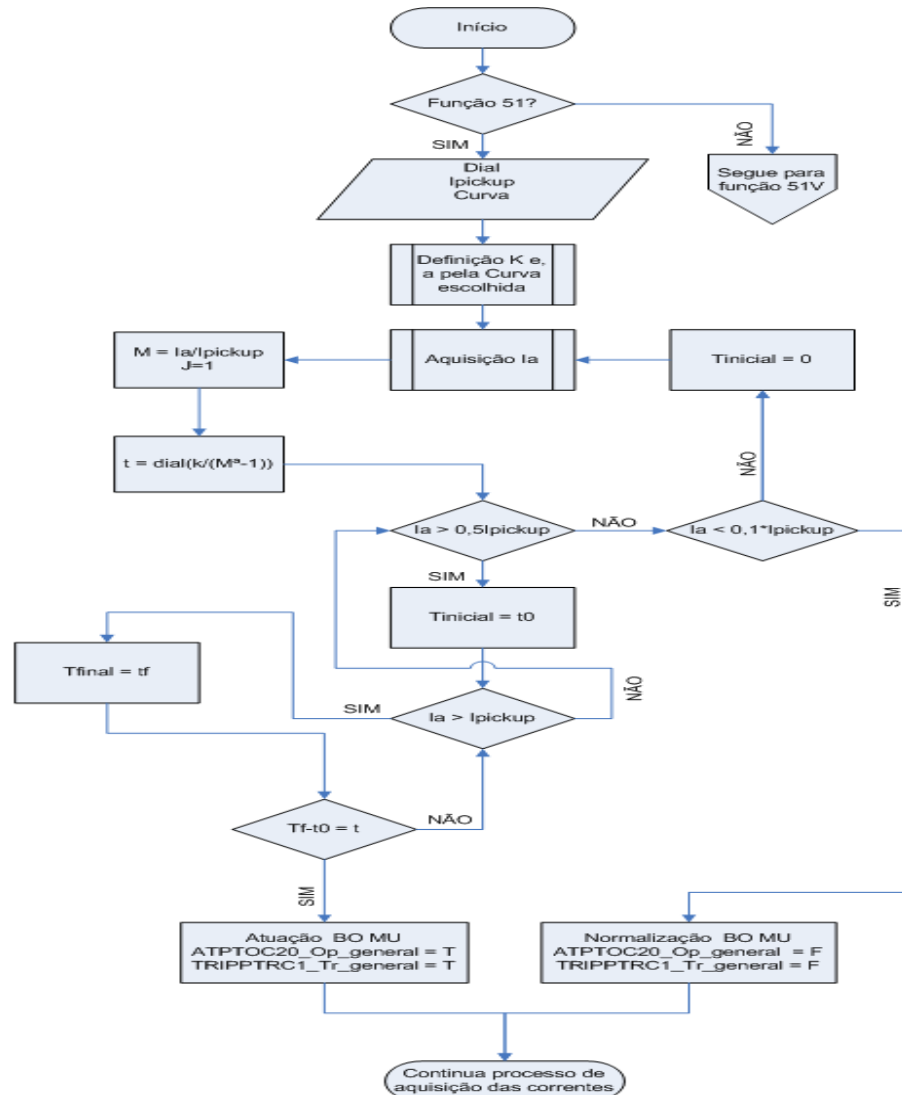


Fonte: Elaborado pelo autor.

3.3.3 Funções de Sobrecorrente Temporizada de Fase (51) e com Restrição de Tensão (51V)

A Figura 28 apresenta o fluxograma para a lógica de proteção 51, sendo mostrado somente a fase A, pois as demais têm o fluxo repetitivo, somente sendo modificadas algumas variáveis de controle para cada fase. Visualiza-se, na primeira parte, qual será a função a ser escolhida, 51 ou 51V, e armazenada a opção na variável *funcao*. Depois são solicitados os valores para o cálculo do tempo, de acordo com as variáveis: *dial*, variável *T*, a corrente de disparo, variável *pick_up* e qual tipo de curva, variável *curva*. Os valores das constantes K e a são agregados de acordo com a curva IEC escolhida, conforme a Tabela 4.

Figura 28 – Fluxograma da lógica da proteção 51 para fase A



Fonte: Elaborado pelo autor.

A proteção do VIED é sensibilizada quando a corrente atual do sistema é 50% superior ao valor do *pick-up*. O programa começa a analisar o tempo inicial da partida de proteção e armazena a diferença entre esse tempo e o de disparo, que foi calculado pela fórmula da curva. A diferença fica sendo calculada 60 vezes por segundo em um *loop* infinito, até que o tempo calculado seja maior que o tempo determinado de atuação pela função da IEC. O algoritmo irá dar continuidade, atualizando os atributos de dados ATPTOC20, BTPTOC21 e CTPTOC22., utilizados para fase A, B e C, para *true*, que estão vinculados a um *Dataset* e esse a um *Report* que será informado ao SCADA SAGE.

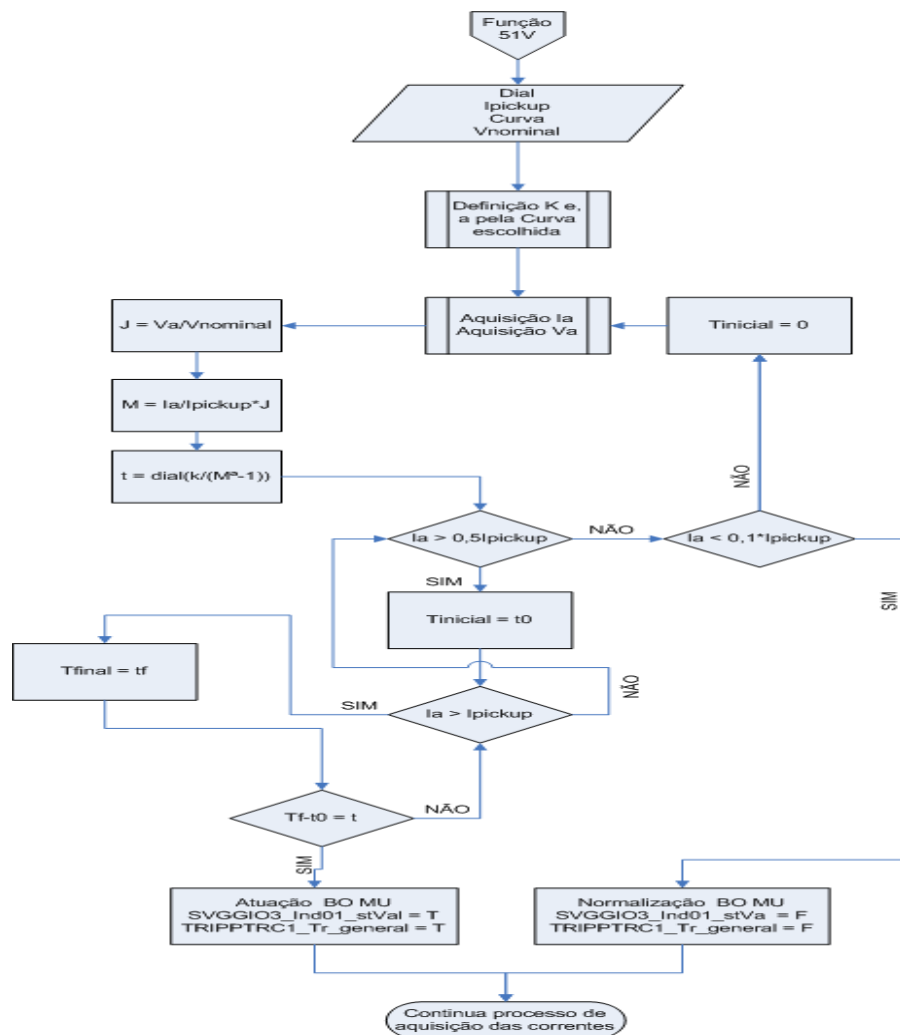
Da mesma forma que as anteriores, quando a corrente presente é menor que 10% da corrente de *pick-up*, todos os atributos de dados são levados para o estado *false*. Caso seja escolhida a função 51V, o fluxograma está demonstrado na Figura 29 e difere um pouco do

fluxograma anterior, pois é acrescentada a tensão nominal que será anexada a fórmula do cálculo de tempo de atuação. Também tem a diferença do LN de atuação da proteção, o SVGGIO3, terá os atributos booleanos modificados em Ind01, Ind02 e Ind03 que são exclusivos para a supervisão do SCADA SAGE.

3.3.4 Função de Sobrecorrente Temporizada de Neutro (51N)

Os códigos em Linguagem C são muito parecidos com o VIED da função de sobrecorrente de fase (51). Assim, para não ser repetitivo, se destaca que no *trip* da função 51N, se encontra no LN SVGGIO3, que será modificado no seu atributo de dados Ind13, para supervisão do SCADA SAGE. O *trip* geral vem do atributo de dados *Tr_general* do LN PTRC.

Figura 29 – Fluxograma da lógica da proteção 51V para fase A

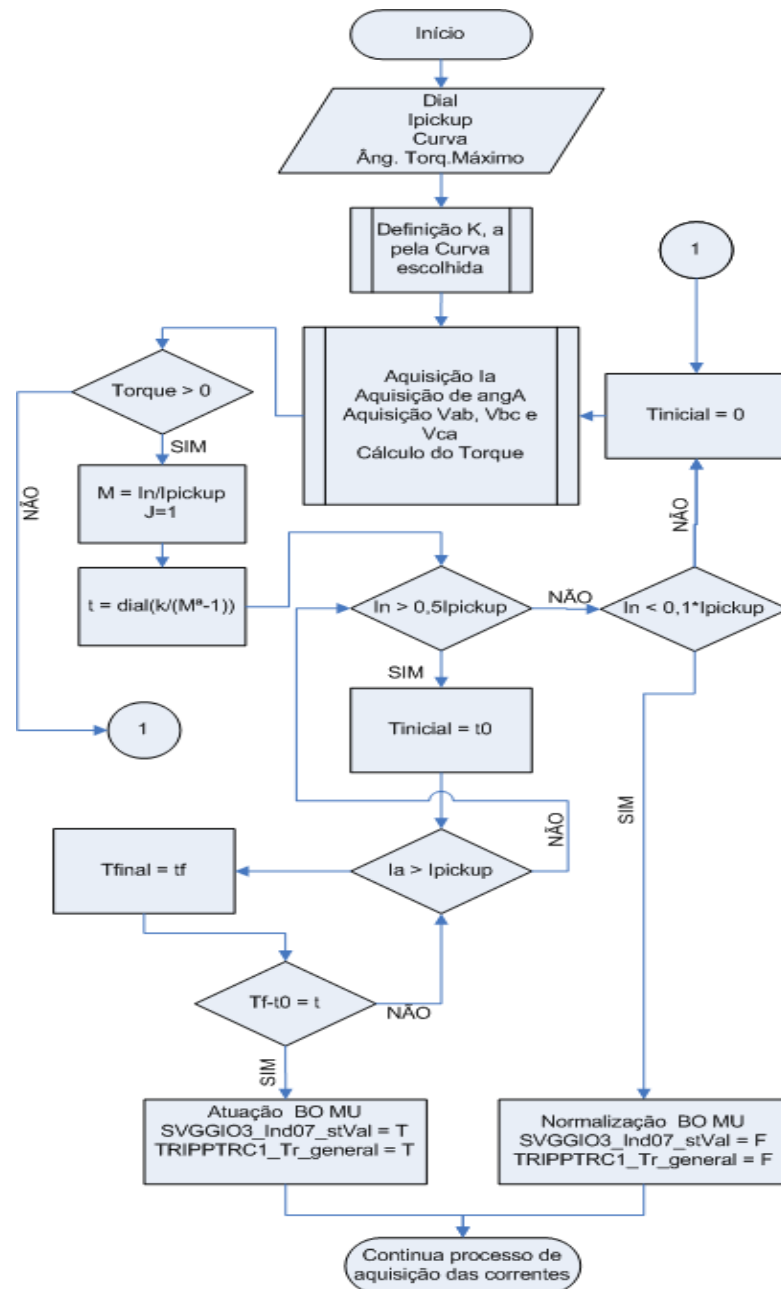


Fonte: Elaborado pelo autor.

3.3.5 Função de Sobrecorrente Direcional (67)

A Figura 30 apresenta o fluxograma representando a função de proteção 67 para fase A, sendo repetitivo para as outras fases, com alteração somente em algumas variáveis e atributos de dados do LN referente à supervisão do SCADA SAGE. A definição dos parâmetros para a atuação é igual a função de proteção 51, adicionado do ângulo de torque máximo.

Figura 30 – Fluxograma função 67 fases (representado somente fase A)



Fonte: Elaborado pelo autor.

Após definir os dados de parametrização do VIED, o próximo passo são os processos já abordados de aquisição de medidas analógicas por parte do SV.

Após é feita a construção do algoritmo, de acordo coma abordagem da filosofia de proteção para a função 67 no item 2.7 do capítulo 2. É construído o número complexo das tensões de linha V_{ab} , V_{bc} e V_{ca} , para depois determinar o módulo e o ângulo de cada tensão de linha mencionada. Os ângulos estão presentes nas variáveis aV_{ab} , aV_{bc} e aV_{ca} e os módulos estão presentes nas variáveis mV_{ab} , mV_{bc} e mV_{ca} . Com esses dados, são feitos três cálculos; o primeiro é a diferença angular entre a I_a e a tensão V_{bc} e esse valor armazenado na variável $teta$, o segundo, a diferença angular entre a I_b e a tensão V_{ca} e esse valor armazenado na variável $teta1$, o terceiro, a diferença angular entre a I_c e a tensão V_{ab} e esse valor armazenado na variável $teta2$.

Enfim, é feito o cálculo de torque que seria aplicado no disco do relé, responsável pela direcionalidade da função de proteção 67, o valor é armazenado em $torque$, $torque1$ e $torque2$, respectivamente para fases A, B e C – se o valor delas forem superiores a zero, é entendido que a função 67 foi sensibilizada, a partir dessa polarização. A partir desse ponto, o funcionamento é o mesmo da função 51 no item 3.3.5. O único ponto de diferença é que o LN *SVGGIO3* tem os atributos de dados *Ind07*, *Ind08* e *Ind09*, sendo, respectivamente, para a atuação das fases A, B e C da função 67.

3.3.6 Função de Sobrecorrente Direcional de Neutro (67N)

Da mesma forma que a função do item 3.3.5, os valores de parametrização devem ser informados como no fluxograma da Figura anterior. A diferença do 67 é o dado do disparo da função, o LN *SVGGIO3* tem o atributo de dado *Ind12* e no *trip* é atualizado para *true*.

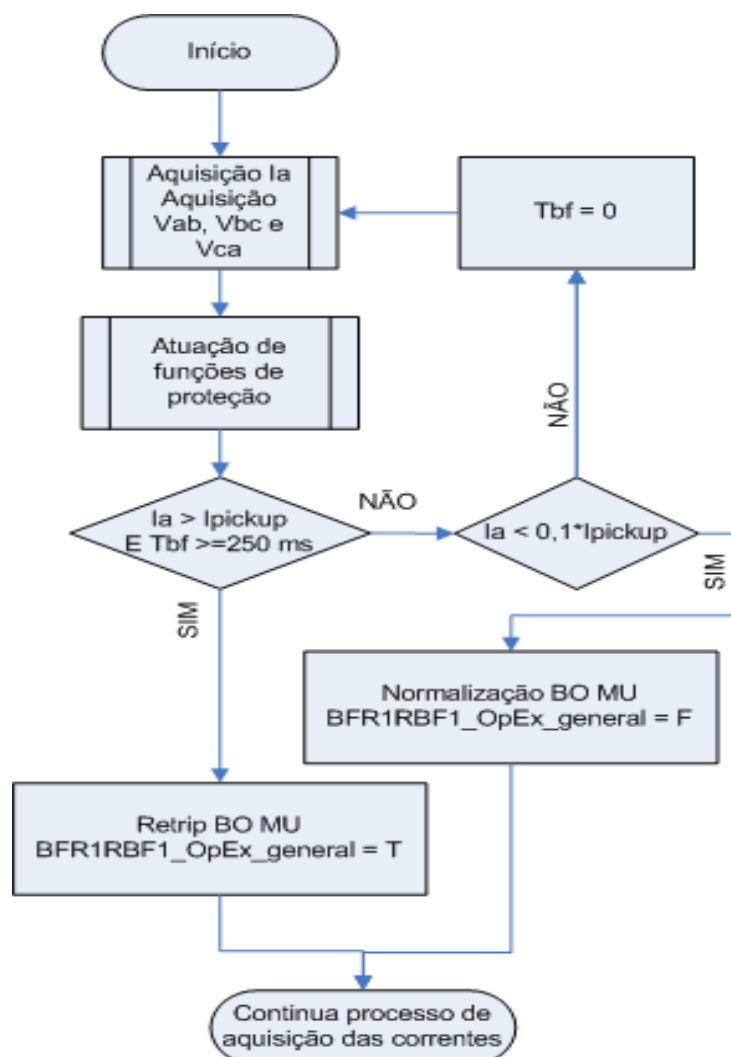
3.3.7 Função de Falha de Disjuntor (50/62BF)

Uma das funções de proteção que estão implementadas em todos os VIEDs é a que informa para o sistema, a falha do disjuntor, com o código ANSI 50BF. Essa lógica opera de forma complementar a todas as lógicas de proteção criadas, aumentando assim a confiabilidade do dispositivo virtual desenvolvido. Pode ser verificada a sua vinculação em todos os algoritmos dos VIEDs que estão no Apêndice 8.2 – o conceito geral da função é que após 250 milissegundos do disparo do *trip* do VIED, se ainda permanecem as correntes da falta, a fun-

ção 50BF tem que atuar e comunicar no SCADA SAGE uma falha do disjuntor que está associado, isso é feito no VIED por meio de mensagens GOOSE e por *report* MMS.

A lógica, retratada no fluxograma da Figura 31, inicia com o armazenamento do tempo inicial em uma função que será executada apenas uma vez, em sequência é armazenado o tempo seguinte a esse tempo inicial. Depois é calculada a diferença de tempo em segundos entre o inicial e o tempo seguinte, e, caso a falta permaneça, a variável de tempo Tbf for maior ou igual a 250 milissegundos, dispara a função falha de disjuntor, o atributo de dados do LN BFR1RBRFa, *OpEx_general* é alterado para o estado de verdadeiro. Da mesma forma que as outras funções, se a falta foi cessada, o atributo mencionado terá seu estado atualizado para falso.

Figura 31 – Fluxograma da função 50 BF



Fonte: Elaborado pelo autor.

3.4 Considerações Finais

As funções de proteção, que são elaboradas em linguagem C e apresentadas no VIED, podem ser alteradas via *software* de maneira bem dinâmica, utilizando a biblioteca libIEC61850. A biblioteca trabalha com algumas linguagens, mas se optou pelos códigos em Linguagem C. Apesar de não ter uma apresentação visual para o usuário, foram elaborados *prompts* que são bem dinâmicos e ilustrativos para executar os VIEDs.

As funções de proteção possuem algumas simplificações de parametrizações, visto que não foi escopo do trabalho abordar os vários itens de parametrização que cada IED possui. Inclusive IEDs de fabricante diferente, tem algumas parametrizações e dados distintos, ficando bem mais complicado podermos simular essas situações e implementá-las nos códigos da linguagem C.

A utilização da planilha CURVAS RELES, para os cálculos de tempo de disparo das proteções de sobrecorrente temporizada, foi bastante útil e norteou o cálculo matemático empregado na lógica do VIED. Todos os códigos dos arquivos utilizados e algumas explicações das lógicas de cada função de programação se encontram no Apêndice 8.2.

4 PLATAFORMA DE DESENVOLVIMENTO E TESTES COM VIED

Para o desenvolvimento e testes com o VIED, foi elaborada uma plataforma experimental, montada no laboratório de proteção na Eletrobrás CHESF, onde se realizou os testes de validação da integração do VIED, MU e SCADA SAGE, checando o tráfego da utilização dos protocolos MMS, GOOSE e SV resultado das simulações de atuações de proteção, verificação das supervisões e os comandos de controle.

4.1 Plataforma de Automação e Controle de Testes

A plataforma de testes utilizada era composta de equipamentos de *hardware* e *softwares* específicos, com as seguintes características:

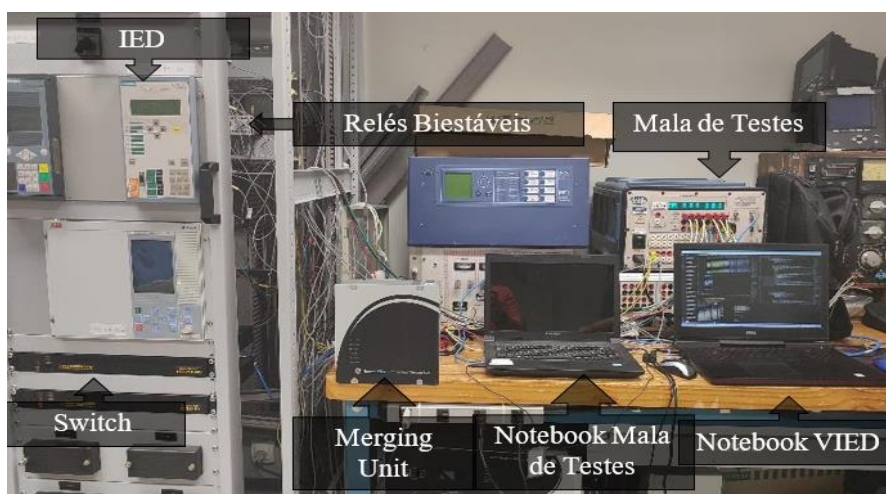
- a) *Notebook* DELL, com processador i7, frequência de 3,6 GHz, 16 GB de memória RAM, com SSD (Unidade de estado sólido em português) de 240 GB, com sistema operacional Linux 3.10.0-1062.12.1.el7.x86_64_x86_64 (CentOS Linux release 7.7.1908) e no qual estão instalados os VIEDs responsáveis pelas funções de proteção;
- b) *Notebook* DELL, com processador i3, frequência de 3,3 GHz, 8 GB de memória RAM com sistema operacional Windows 10 Enterprise. Dispositivo utilizado com o *software* de controle da mala de testes;
- c) Computador Industrial com processador Intel Core 2 Duo, frequência de 2,93 GHz, 8 GB de memória RAM com sistema operacional Windows 10 Enterprise. Dispositivo utilizado para configurar a Merging Unit, o relé 7SJ6 da SIEMENS e com os *softwares* de análise do tráfego de rede;
- d) Computador Industrial com processador Intel Core 2 Duo, frequência de 2,93 GHz, 8 GB de memória RAM com sistema operacional Linux 3.10.0-1062.12.1.el7.x86_64_x86_64 (CentOS Linux release 7.7.1908). Dispositivo que serve de IHM que utiliza o *software* do SCADA SAGE;
- e) Um IED do fabricante SIEMENS da linha Siprotec 4, modelo 7SJ62-11-5EB92-1FE0-Z/FF com as proteções 50/51, 50N/51N, 49, 46, 37, 47, 59N/64, 50BF, 74TC, 86, 27/59 e 81O/U;
- f) Mala de testes do fabricante DOBLE modelo F6150, hexafásica com 6 terminais de corrente e 6 de tensão configuráveis, entradas digitais configuráveis, utilizada para teste de dispositivos e relés eletromecânicos, eletrônicos e mi-

croprocessados tradicionais, e foi utilizado para aplicar as correntes e tensões analógicas e verificar a abertura dos disjuntores simulados na bancada;

- g) *Switch* gerencial do fabricante RUGGEDCOM, modelo RSG2100NC, para ambientes industriais severos e exigentes, com 19 portas *Ethernet* modulares, é utilizado como barramento de processos e LAN para conexão de todos os equipamentos;
- h) Uma *Merging Unit* do fabricante General Electric, modelo MU320E, com interfaces analógicas e digitais para modelagem, controle e digitalização de equipamentos de pátio. Na nossa aplicação, é utilizada para digitalização dos sinais analógicos provenientes da Mala de Testes em SV e envios de comandos aos relés biestáveis através dos GOOSEs recebidos.;
- i) Um relé biestável da SIEMENS, modelo 3RH1422-1BG40 da linha Sirius, utilizado para simular o disjuntor do vão protegido pelo IED;
- j) Um relé biestável da SCHNEIDER ELECTRIC, modelo CAD32-LAD6K10 da linha Tesys, utilizado para simular o disjuntor do vão protegido pelo VIED;
- k) Acessórios: Cabos *ethernet*, cabos das conexões elétricas da mala de testes, fibras óticas multimodo com conectores tipo LC e ST.

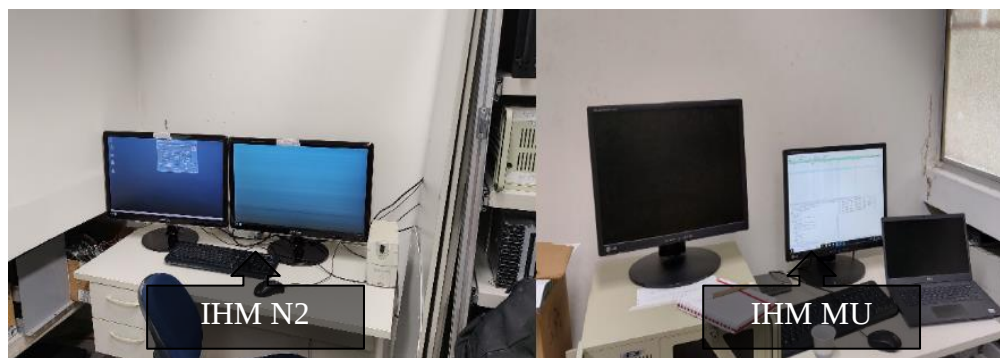
A plataforma utilizada, com seus componentes e equipamentos, pode ser visualizada nas figuras 32 e 33.

Figura 32 – Plataforma de testes do VIED na Eletrobrás CHESF



Fonte: Elaborado pelo autor.

Figura 33 – IHMs da plataforma de testes na Eletrobrás CHESF



Fonte: Elaborado pelo autor.

Os *softwares* utilizados foram os seguintes:

- a) No *Notebook* da Mala de Testes, o *software F6 Test* da DOBLE, que contempla a configuração das etapas e cenários para injeção de valores analógicos para os testes de proteção;
- b) No *Notebook* do VIED, o *Visual Code Studio*, versão 1.63.0 para CentOS, para programação dos VIEDs;
- c) No *Notebook* de apoio, o *AcSElarator Architect*, por meio do qual se configura os IEDs do fabricante SEL em sua comunicação padrão IEC 61850. No nosso caso o SEL 751, fornecendo seu arquivo do tipo CID;
- d) Na IHM MU, o *software* de configuração e supervisão da MU (*MU Software Configurator*), o *software* DIGSI versão 4.98 de parametrização dos IEDs SIEMENS, Wireshark para análise de tráfego de rede e ACMViewer que também funciona como *sniffer*.
- e) Na IHM N2, o *software* SCADA SAGE.

O diagrama que melhor exemplifica a plataforma está apresentado na Figura 34. As conexões entre os equipamentos estão distinguidas por cor, sendo que os cabos de rede tipo UTP estão representados pela cor azul e são curtos, fazendo com que a interferência eletromagnética seja desprezível.

- d) Verificação junto ao supervisor SAGE das listas de alarmes e atuações, e verificação com o *sniffer* sobre as mensagens do protocolo que estavam trafegando na rede.

Na Tabela 6, estão mostrados os endereços IP configurados para os equipamentos conectados à rede *Ethernet*:

Tabela 6 – Lista de Dispositivos e seus IPs na Plataforma

Dispositivo	Função	Endereço IP
IED Relé 7SA6	Proteção e Controle	192.168.110.90
Notebook VIED	Proteção e Controle	192.168.110.220
IHM N2	Supervisor SAGE	192.168.110.1
		192.168.110.2
		192.168.110.4
IHM MU	Configuração MU, IED, tráfego de comunicação	192.168.110.200
Merging Unit	Aquisição sinais analógicos e digitais	192.168.110.199

Fonte: Elaborado pelo autor.

4.2 Configuração dos equipamentos da plataforma

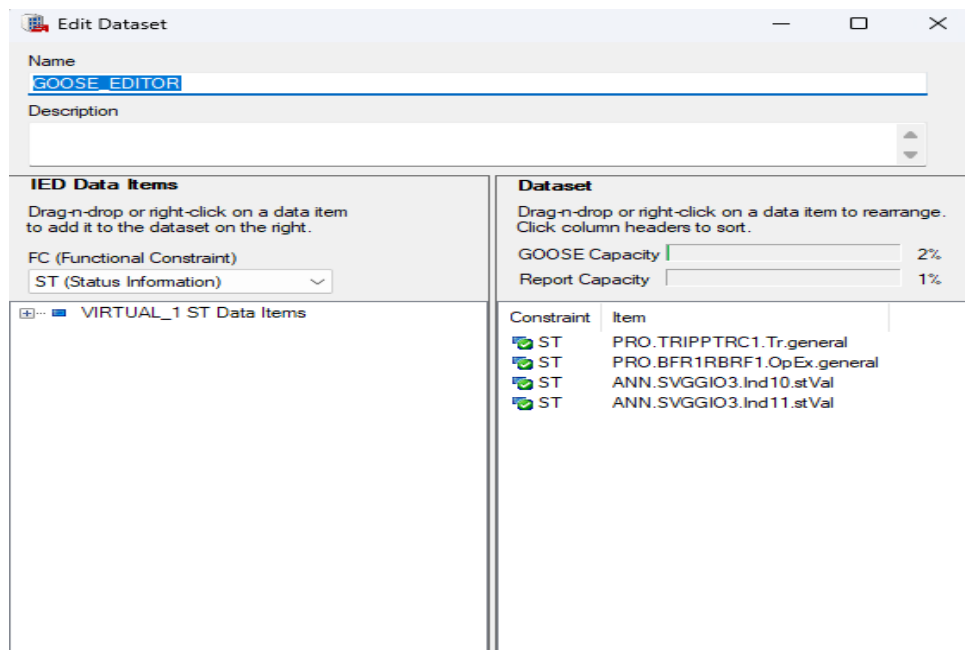
Para todos os algoritmos utilizados do VIED existe a funcionalidade de supervisão e controle integrado ao SCADA SAGE na plataforma de testes utilizada. Esse funcionamento é possível a partir da integração do VIED a MU, onde se utiliza mensagens GOOSE para tal integração. Essa integração é peça fundamental para o funcionamento da plataforma, porque a partir dessa comunicação é possível simular, nos relés biestáveis, o estado de disjuntores e fazer comando de fechamento e abertura. As atuações são realizadas a partir da integração do VIED ao SCADA via mensagens MMS, assim serão retornados os estados dos dispositivos que vêm da plataforma e onde o *software* SCADA SAGE possa fazer o controle.

4.2.1 Definição da Mensagem GOOSE enviada pelo VIED

Na Figura 35 é demonstrado o *software* *AcSELerator Architect* do fabricante de IEDs SEL. Nela podem realizar a configuração do envio de mensagens GOOSE que será feita pelo VIED. Foi criado um *Dataset* de nome GOOSE_EDITOR que é a fonte de quatro atributos de uma mensagem GOOSE, sendo:

- a) O *PRO.TRIPPTRC1.Tr.general*, responsável por informar a atuação do *TRIP* de qualquer das funções de proteções parametrizadas no VIED;
- b) O *PRO.BFR1RBRF1.OpEx.general* responsável pela função de proteção de falha de disjuntor (50BF) que está disponível em todos os algoritmos de funções de proteções disponibilizados nos VIEDs;
- c) O *ANN.SVGGIO3.Ind10.stVal* sendo um atributo GOOSE genérico que será utilizado posteriormente para fechamento do disjuntor na plataforma, pelo SCADA SAGE;
- d) E por último, o *ANN.SVGGIO3.Ind11.stVal* sendo um atributo GOOSE genérico que será utilizado posteriormente para abertura do disjuntor na plataforma.

Figura 35 – Configuração de Dataset para o IED



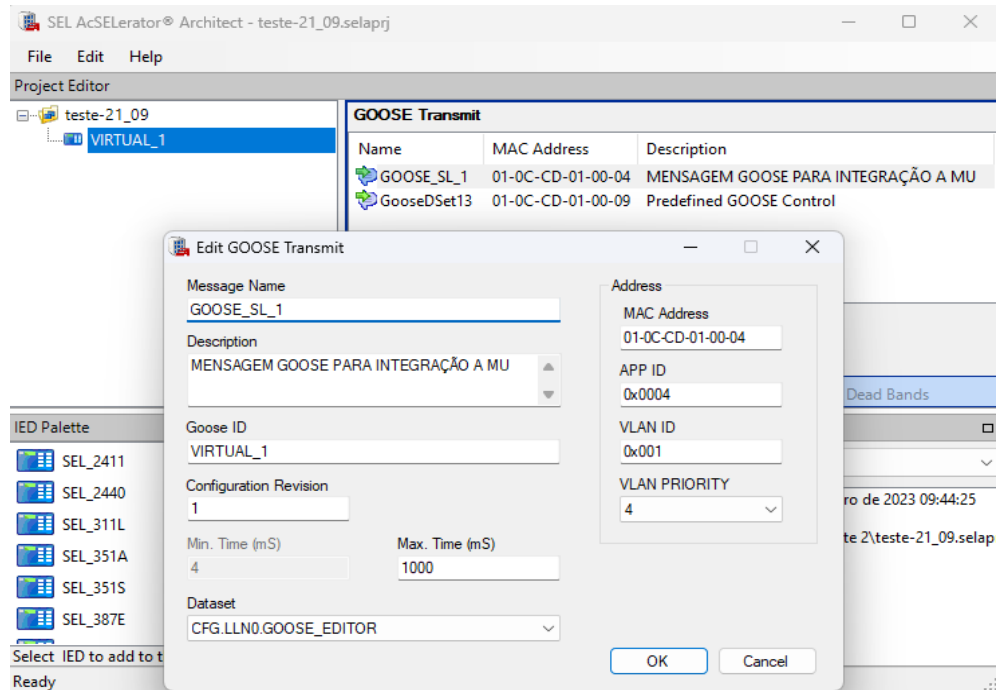
Fonte: Elaborado pelo autor.

Na Figura 36, ainda no *software* da SEL, tem-se a atribuição do *Dataset* criado a uma mensagem GOOSE que será transmitida pelo VIED na rede e assinada pela MU, fechando assim a arquitetura cliente-servidor. A Figura 37 nos mostra um retrato da rede local capturado pelo *software Wireshark*. Este *software* funciona como um analisador de rede e consegue perceber e registrar todos os dados que trafegaram pela rede selecionada.

Nessa captura percebe-se que a mensagem GOOSE enviada pelo VIED está na rede disponível para os dispositivos nela presentes e que estão incluídos os 4 atributos de da-

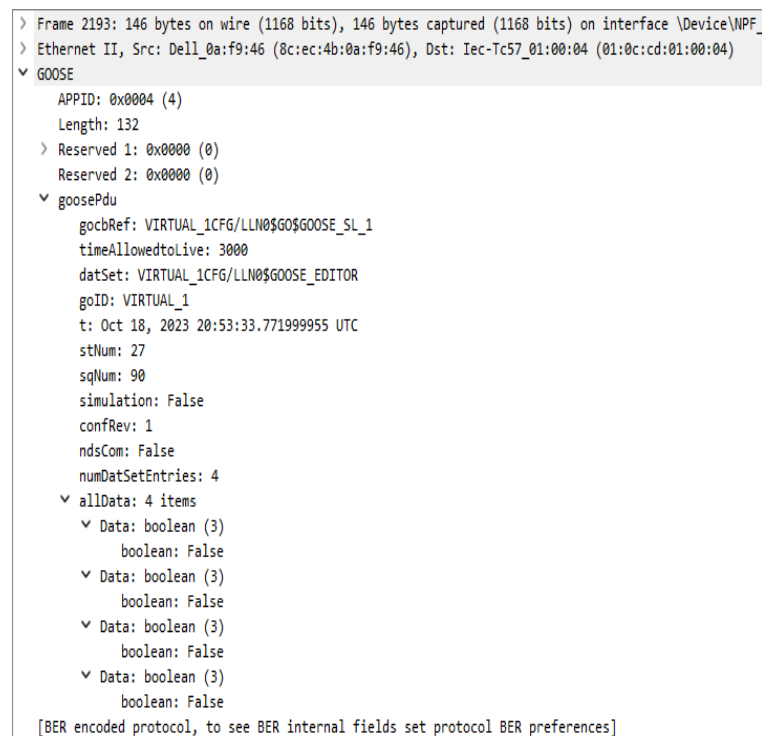
dos mencionados anteriormente, que têm relação ao comando de *TRIP*, abertura, fechamento e falha de disjuntor.

Figura 36 – Dataset do VIED para MU



Fonte: Elaborado pelo autor.

Figura 37 – Software Wireshark com os Datasets

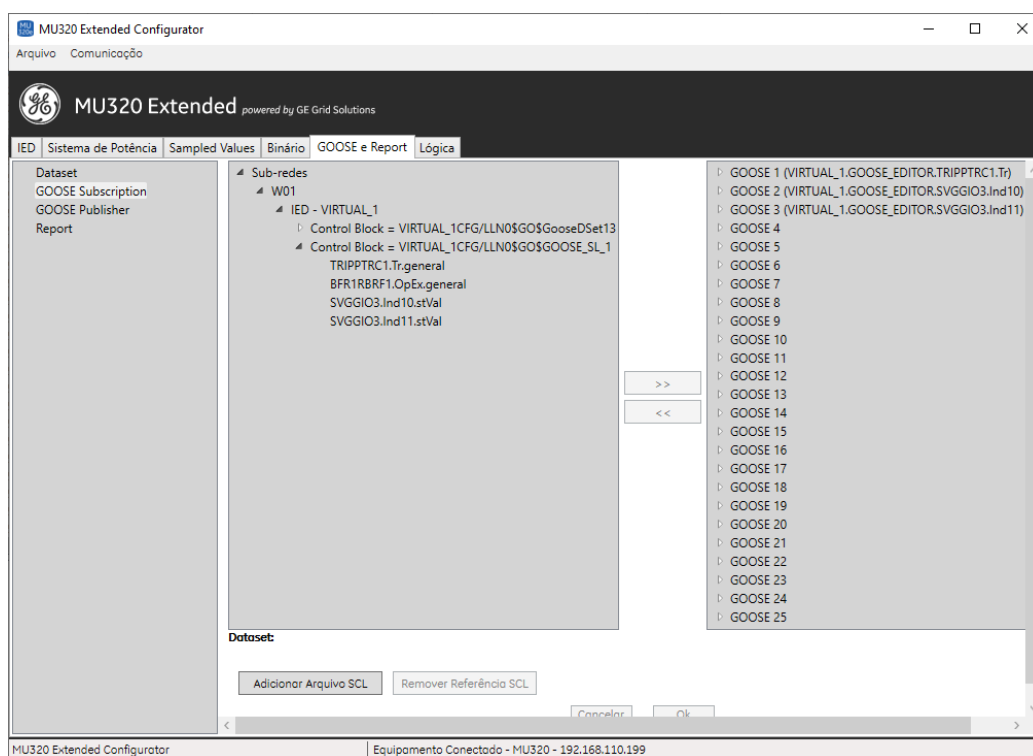


Fonte: Elaborado pelo autor.

4.2.2 Configuração da MU para funcionamento na Plataforma PAC

A Figura 38 destaca a tela do *software MU320 Extender Configurator*, onde se consegue elaborar uma interface com a MU para configurá-la e, nessa tela em específico, tem a configuração necessária para que a MU assine a mensagem GOOSE do VIED. Podem destacar que cada atributo da mensagem GOOSE é destinado a uma variável interna que será utilizada nas lógicas para controle da PAC. Dessas lógicas é que tem a simulação de abertura e o fechamento do relé biestável, simulando um disjuntor do vão protegido pelo VIED.

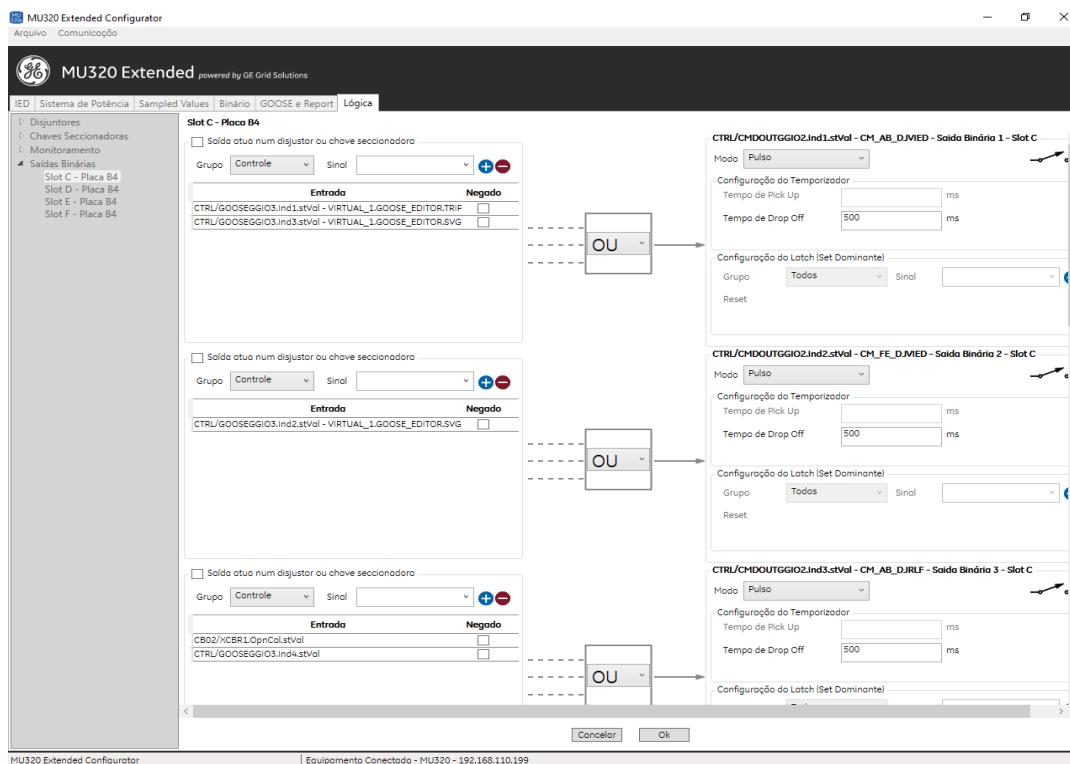
Figura 38 – *Software* de configuração da MU



Fonte: Elaborado pelo autor.

Na sequência da configuração da MU, visualiza-se, na Figura 39, a configuração necessária para atribuir a variável interna para uma saída digital do dispositivo, finalizando desse modo o ponto final do controle. É necessário configurar uma pequena lógica interna, bem intuitiva, que, de acordo com os estados de cada instância GOOSE mencionada para comando, resulta na ativação de algumas das BOs configuradas. Nos testes estamos utilizando, no *Slot C* da MU, a saída binária 1 para o comando de abertura e a saída binária 2 para o comando de fechamento.

Figura 39 – Detalhes da lógica para comando do relé biestável



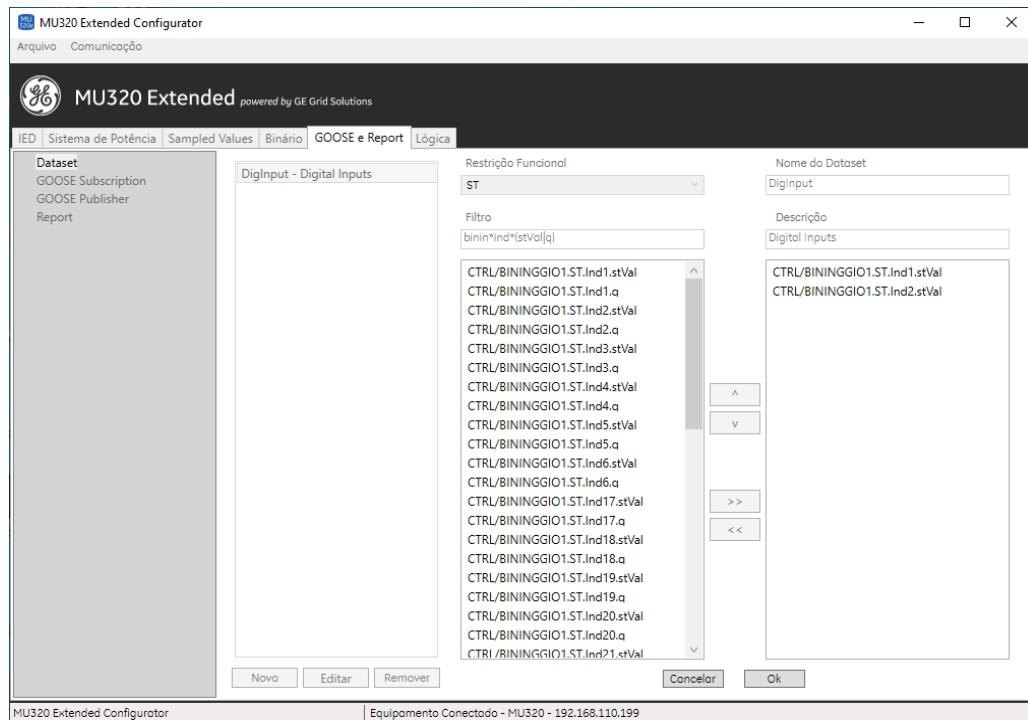
Fonte: Elaborado pelo autor.

O relé virtual utiliza do cartão de entradas e saídas binárias da MU para realizar as ações de comando através da energização das BOs e recebe os feedbacks desses comandos pelas ativações das BI. Essa configuração via GOOSE está de acordo com o item 3.2.2 do Capítulo 3, onde é configurado a assinatura de mensagem GOOSE da MU pelo VIED, essa mensagem retorna o estado do relé biestável presente na PAC que simula um disjuntor, informando, assim, para o sistema, de acordo com os contatos auxiliares, se o disjuntor está aberto ou fechado.

Na Figura 40, é feita a configuração do Dataset que é fonte dos atributos de mensagens GOOSE que informa o estado de cada entrada binária, onde a Ind1 verifica se está aberto e a Ind2 verifica se está fechado.

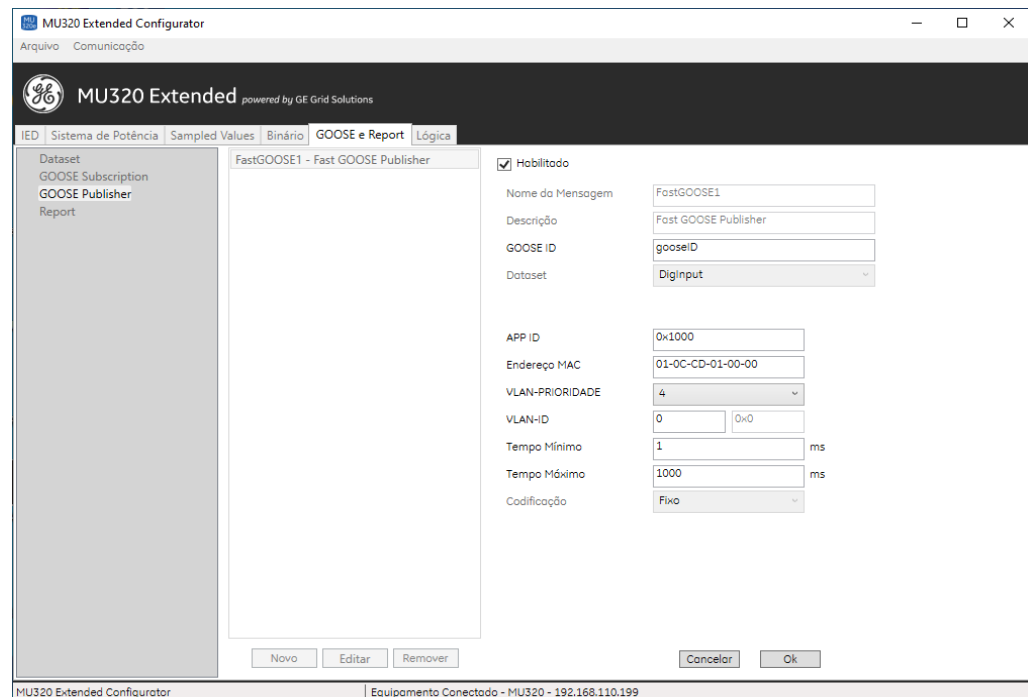
Na sequência, verificamos que na Figura 41 retrata a criação da mensagem GOOSE que será responsável por transmitir esses atributos na rede onde o VIED assina essa mensagem, e que consegue captar o estado do equipamento.

Figura 40 – Mensagem GOOSE do estado do disjuntor



Fonte: Elaborado pelo autor.

Figura 41 – Editor da mensagem GOOSE

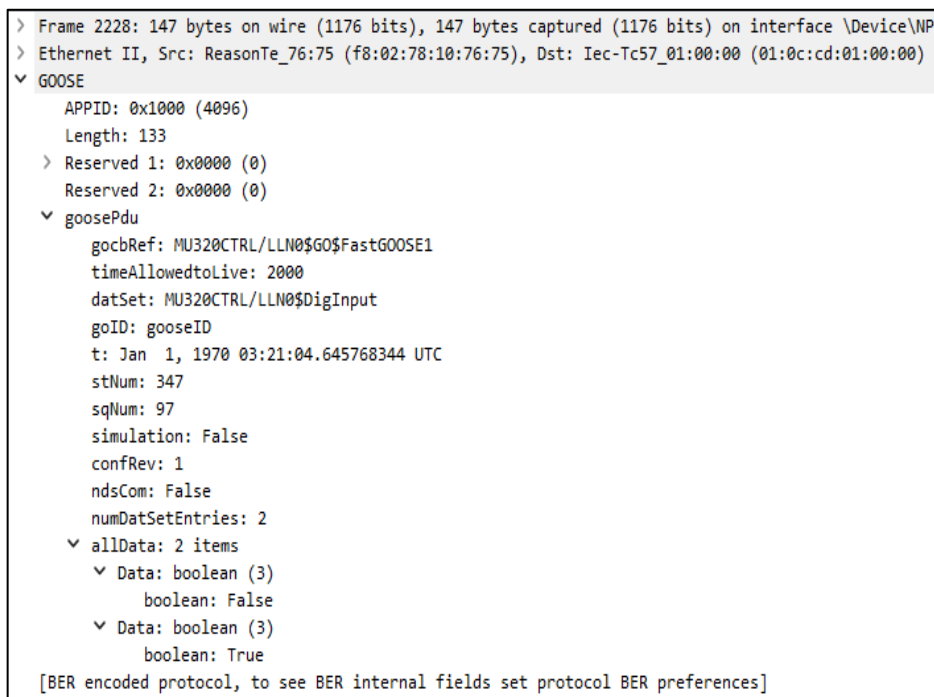


Fonte: Elaborado pelo autor.

Através do *software de sniffer Wireshark*, consegue-se uma visualização da rede local, com os dados que trafegam como mostrado na Figura 42. Nessa demonstração se con-

segue uma captura na qual a mensagem GOOSE é enviada pela MU e está na rede disponível para os dispositivos nela presentes – dois atributos estão incluídos, que são os dos estados das binárias que simulam o disjuntor.

Figura 42 – Captura da mensagem GOOSE pelo Wireshark



Fonte: Elaborado pelo autor.

4.2.3 Configuração do SCADA SAGE para funcionamento na Plataforma PAC

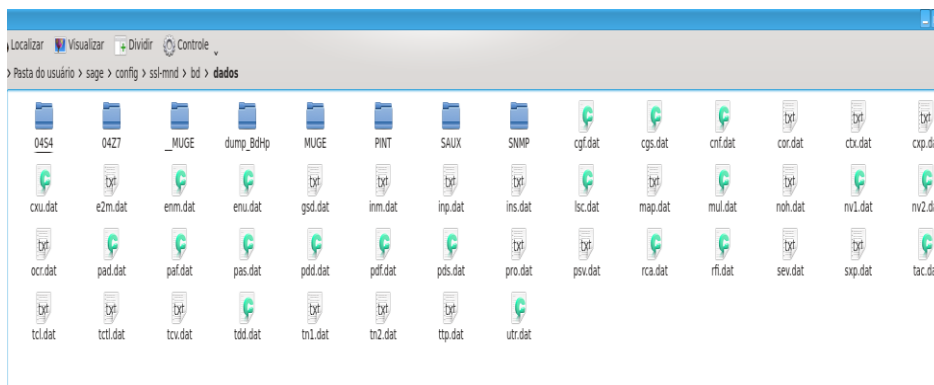
A partir da plataforma existente de IHMs com o SCADA SAGE no laboratório de proteção da Eletrobrás CHESF, foi configurado a inclusão dos equipamentos do restante da plataforma desse trabalho: A UM, o VIED e IED. As informações desses equipamentos foram acrescentadas à base de dados referência da Subestação Maracanaú II (SSL-MND), onde as pastas que a configuram são mostradas na Figura 43.

Com os pontos digitais e analógicos importados do arquivo CID do VIED, forma alimentadas as entidades da base fonte do SCADA SAGE: CGF, CGS, CNF, ENM, LSC, MAP, MUL, NV1, NV2, PAF, PAS, PDF, PDS, PTF, PTS, TAC e TDD. Elas são configuradas, em referência ao VIED de acordo com cada subsistema:

- a) As entidades “CNF”, “ENM”, “LSC”, “MUL” e “TAC” configura o subsistema de comunicação com SAGE com o VIED;

- b) As entidades “CGF”, “CGS”, “MAP”, “NV1”, “NV2”, “PAF”, “PAS”, “PDF”, “PDS”, “PTF”, “PTS” e “TDD” modelam os dados dos pontos digitais e analógicos.

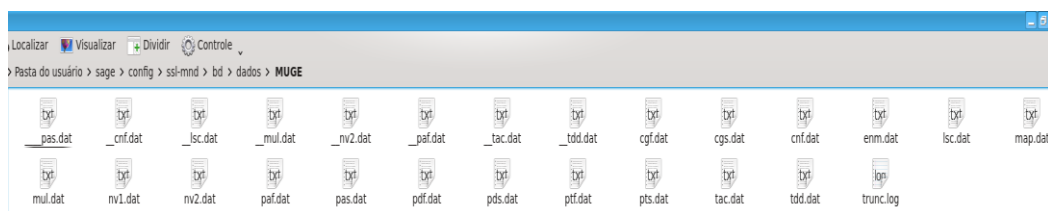
Figura 43 – Base de dados referência do SCADA SAGE para SE MND



Fonte: Elaborado pelo autor.

Como se está utilizando uma base de dados de uma subestação existente, utilizará uma estrutura de dados do SAGE com segmentação para incorporar o vão da MU/VIED para os testes. A segmentação significa que os arquivos, do tipo extensão “DAT”, ficam divididos em dois grupos: os originais na raiz do diretório \$BD/dados e os que estão sendo agregados, referente a MU e VIED na pasta \$BD/dados/MUGE. Na Figura 44, são observados, nas duas pastas informadas, arquivos com mesmo nome, porém nas entidades da pasta principal, irá fazer referência, através da chamada *INCLUDE*, à continuação da leitura de dados dessa entidade, nas subpastas criadas para cada IED ou conjunto de informações que foram criadas posteriormente. Por exemplo, a entidade que está no arquivo tac.dat pertencente a pasta MUGE, complementa os dados do arquivo tac.dat da pasta principal e original “dados”, que contém a linha “#include <MUGE>/TAC.dat.

Figura 44 – Base de dados com as pastas de INCLUDE



Fonte: Elaborado pelo autor.

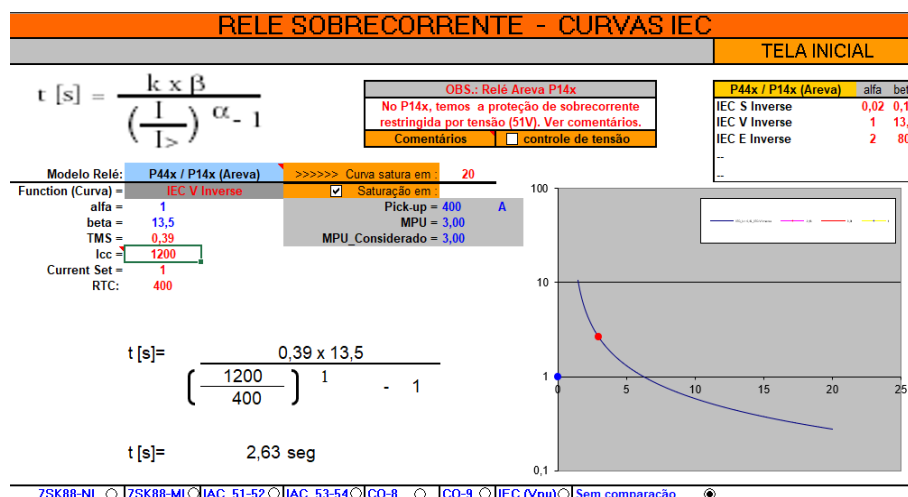
Após a inclusão de informações e pontos novos na base fonte, gera-se a nova base referência, habilita a base de tempo real e reinicia o SCADA SAGE, que trará as alterações que foram feitas, nas telas. Outras informações úteis podem ser mais amplamente verificadas no Manual de Configuração da Base Fonte, no Anexo de Configuração da aquisição do protocolo IEC61850 (CEPEL, 2019), (CEPEL, 2021).

5 RESULTADOS DOS TESTES DA PLATAFORMA

5.1 Execução e validação dos testes e resultados

Foram simulados dois vãos de uma subestação, uma para ser protegido pelo VIED e outro para o IED. Nesses testes foram modelados os disjuntores de pátio e seus sinais analógicos de corrente e tensão oriundos dos transformadores de instrumentos (TCs e TPs). Os relés biestáveis fazem o papel do disjuntor, representando suas bobinas de fechamento e abertura, recebendo o comando vindo do SCADA SAGE através da MU e do IED, e recebendo o *feedback* dos seus estados aberto ou fechado para supervisão de acordo com seus contatos normalmente aberto (NA) e fechado (NF). Na mala de testes, através de cenários elaborados com o *software F6 Test*, são simulados os valores de corrente e tensão. Em uma subestação real, os sinais analógicos são originados nos TIs de cada vão, os valores são rebaixados conforme sua relação de transformação, em caixas de interligação secundárias, finalizando nos IEDs de proteção e controle nos painéis instalados na sala de comando ou cabana de relés. Nos testes, o VIED e IED são obrigados a tomar decisões de acordo com a função de proteção e suas parametrizações. São injetados valores analógicos de pré-falta e falta, devidamente calculados de acordo com uma planilha de cálculo dos tempos de atuação das proteções de sobrecorrente temporizadas, denominada CURVAS RELES. Essa planilha com interface gráfica está demonstrada na Figura 45, sendo bastante utilizada pelas equipes de manutenção de proteção da Eletrobrás CHESF. Os valores mostrados na planilha servem de orientação nas parametrizações de ajustes do VIED e IED.

Figura 45 – Tela de interface da planilha de cálculo de tempo de atuação



Fonte: Elaborado pelo autor.

Na parte traseira da MU e do IED foram conectados os cabos de corrente e tensão elétrica, bem como os cabos de controle que se ligavam aos relés biestáveis, com as funções de levar potencial para simular o comando de abertura e fechamento do disjuntor, além de receber o *feedback* do estado depois do comando executado. Os valores analógicos que chegam na MU são digitalizados e enviados a LAN do barramento de processos. Esses valores são capturados pelo VIED que toma as decisões de acordo com a parametrização da função de proteção que foi selecionada.

Como parte da observação da rede de processos e de subestação, foram elaborados diversos testes e análises dos protocolos GOOSE, SV e MMS, checando a comunicação e os comandos dos equipamentos, oriundos do SAGE. Foi testado o algoritmo da MU sintética que substituiu a MU da General Electric em alguns momentos dos testes, consolidando valores analógicos no *frame* SV enviados ao VIED com atuações de *trip*.

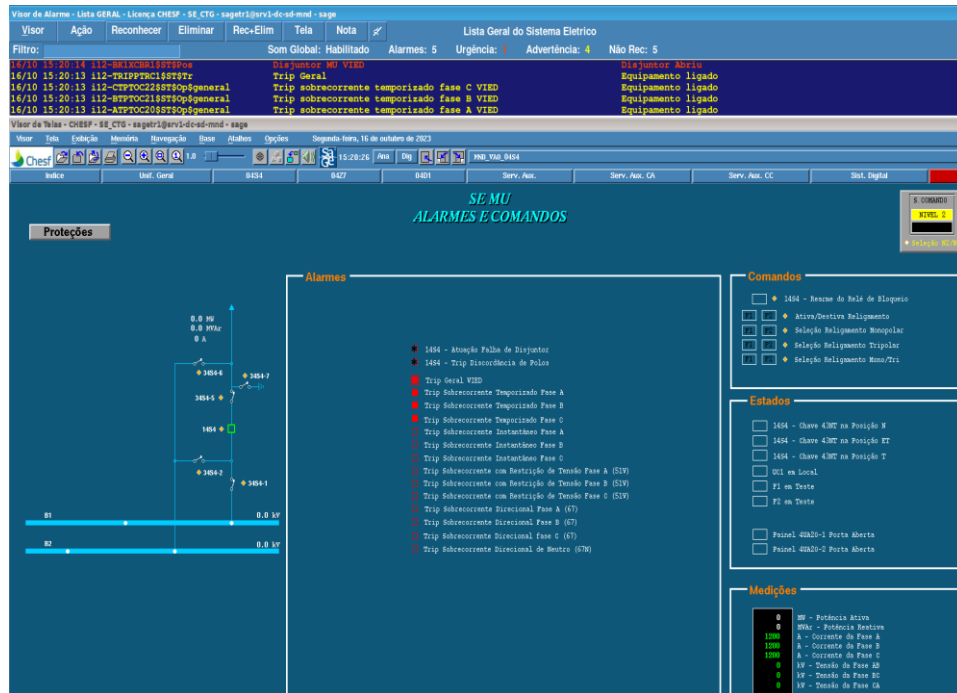
Por fim, foram observados também os *trips* das funções de proteção do VIED, comparando o funcionamento correto, de acordo com as curvas da IEC, também os disparos do IED físico e o VIED, no mesmo teste de proteção que simulava uma falta de uma linha de transmissão, tanto monofásica como trifásica. Todos esses testes foram supervisionados pelos pontos de MMS que se apresentavam na IHM com o SCADA SAGE.

5.2 Testes de Comando, Proteção e Supervisão

5.2.1 Estado, comandos e medições do disjuntor do vão da MU

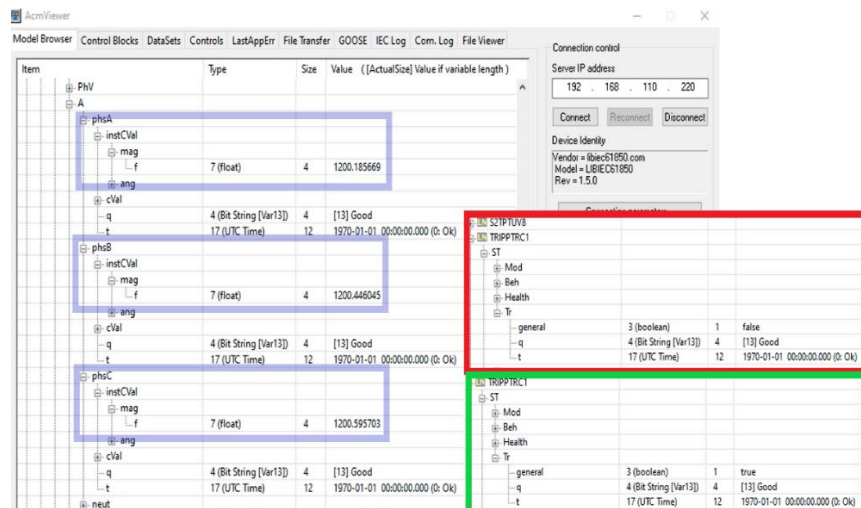
Na tela capturada do SCADA SAGE, visto na Figura 46, são demonstradas duas janelas sobrepostas: uma com o Visor de Alarmes e a outra com o Quadro mímico, que tem o diagrama unifilar do vão, anunciadores que contêm algumas informações em formato de tabela, e outro para medições. Nos alarmes da lista têm os registros dos eventos, que podem ser de proteção ou comandos. Nesta figura se destaca a atuação da proteção e qual o tipo de proteção atuou. Também pode ser observado o disjuntor aberto (quadrado verde vazado), de acordo com o estado do relé biestável, os alarmes atuados em vermelho com as proteções que atuaram e o valor eficaz da corrente de falta que sensibilizou essas funções.

Figura 46 – Tela de alarmes e mímico do SCADA SAGE



Fonte: Elaborado pelo autor.

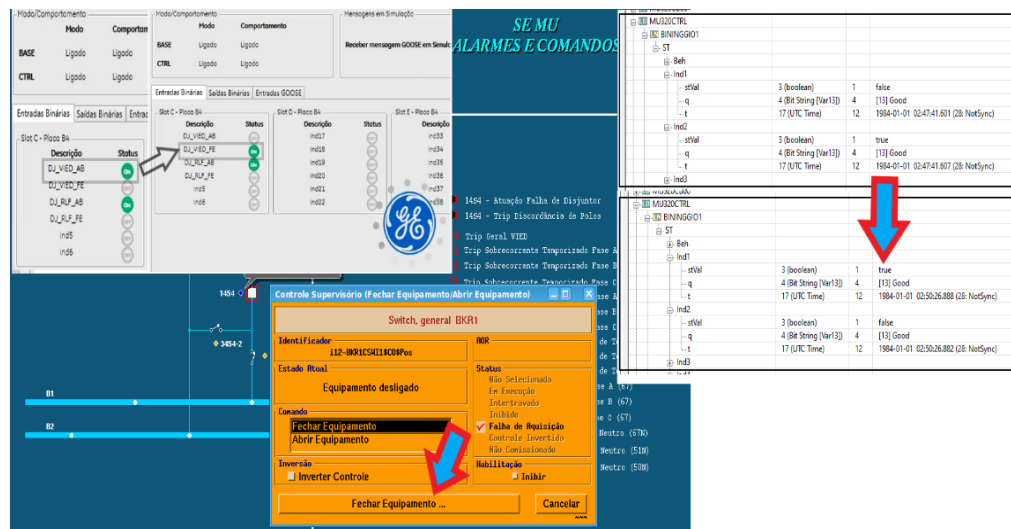
As medidas analógicas por fase de 1200A vistas na tela da IHM SAGE, no momento da falta, são capturadas na rede pelo *software* ACMViewer, e estão mostradas em seus LNs de acordo com a Figura 47 (destaque em azul). Também nessa figura vemos os LN responsáveis pela abertura do disjuntor, que carrega o atributo de *trip* geral, sendo verificado no quadro vermelho o valor *false* para a variável *general* e no disparo da proteção, a atualização para *true* (quadro verde).

Figura 47 – Medidas analógicas e disparo de *trip* para o disjuntor

Fonte: Elaborado pelo autor.

Na Figura 48 está sendo observado que o comando no SCADA SAGE de fechar equipamento, gera a mudança dos atributos da mensagem GOOSE na rede LAN, verificado pelo *software* ACMViewer. Além de que na IHM da MU é mostrado a mudança do estado da mensagem de DJ_VIE_AB para DJ_VIED_FE para ON.

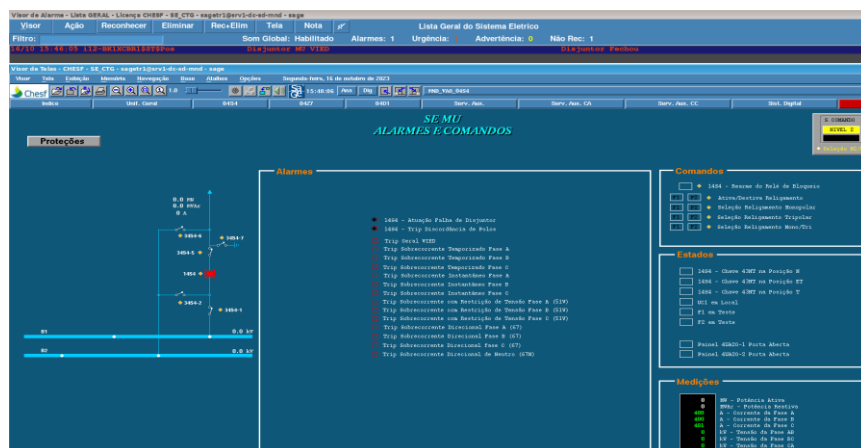
Figura 48 – Status do comando de disjuntor (ACMViewer, SAGE e IHM MU)



Fonte: Elaborado pelo autor.

Com o *trip* já normalizado e disjuntor aberto, após o comando, o disjuntor passou para o estado fechado (quadrado vermelho sólido); também pode se notar que as sinalizações de alarme desaparecem. O disjuntor anteriormente comandado pela IHM, modifica o seu estado para fechado (vermelho). As medidas retornaram ao estado normal, no caso da corrente em 400 A, injetadas pela mala de testes (Figura 49).

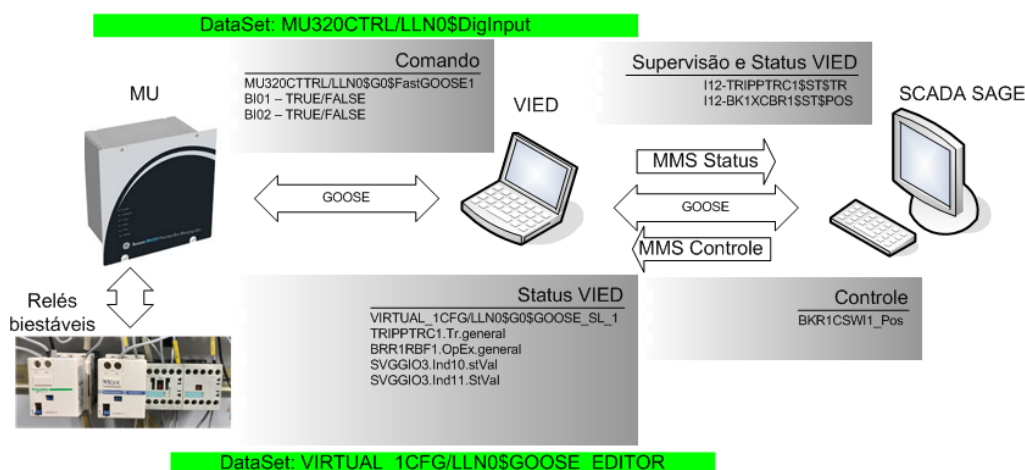
Figura 49 – Tela do SAGE com o disjuntor fechado após o comando



Fonte: Elaborado pelo autor.

Na Figura 50, tem-se um resumo de todas as mensagens da norma IEC 61850, que trafegam pela LAN e que são referentes ao controle de abertura e fechamento que se originam, Comando feito pela interação humana na IHM, originado de forma lógica e chega de forma elétrica nos relés biestáveis. Observamos que são mostradas as mensagens de supervisão que servem de *feedback* para algumas lógicas internas do supervisor SCADA SAGE.

Figura 50 – Mensagens trafegadas na LAN da plataforma



Fonte: Elaborado pelo autor.

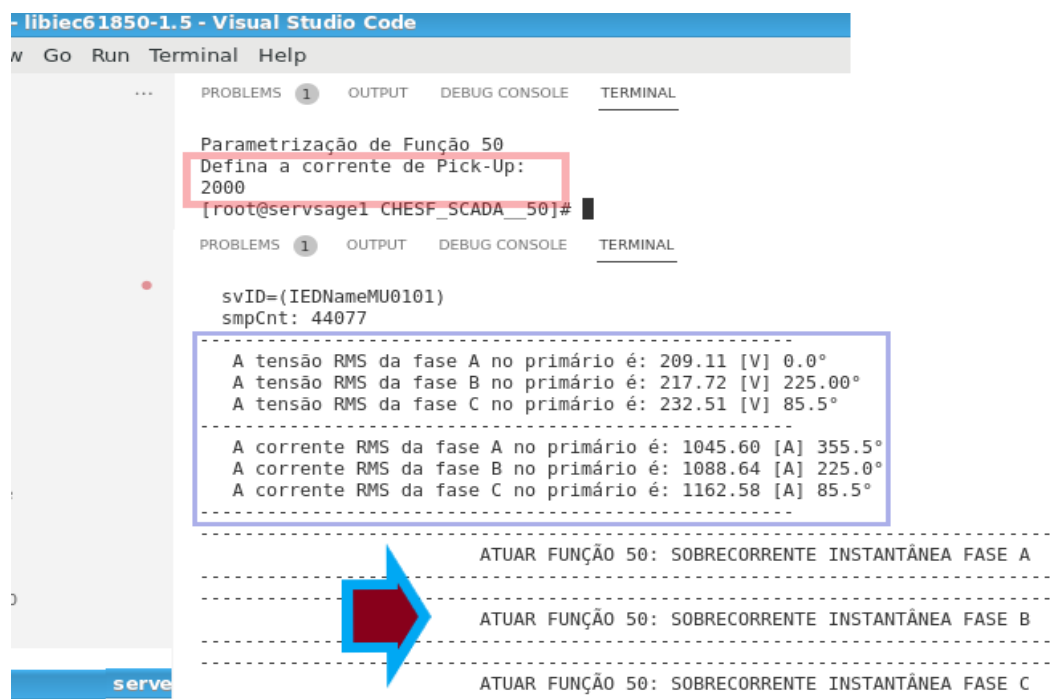
5.2.2 Função 50

Esse teste feito para a função de sobrecorrente instantânea visa observar o comportamento do VIED, de acordo com a programação em C simulando essa função. Foi criado o VIED, a partir do arquivo CID do IED SEL-751, e na plataforma foi fechado o disjuntor, simulado pelo relé biestável, pelo SCADA SAGE e preparado o teste na caixa de testes DOBLE. Após compilado o VIED, retratado na Figura 51, a interface com o usuário solicita a corrente de atuação (*pick-up*) destacado em vermelho. As correntes elétricas analógicas são aplicadas pela caixa de testes na MU, que transmite digitalizando os sinais ao VIED (destaque em azul) e na execução da lógica de proteção, quando o valor de *pick-up* é alcançado, é vista a atuação trifásica da proteção destacada pela seta azul.

Em outro teste, utilizou-se uma corrente parametrizada de 1200 A, que na configuração interna da MU foi definida a relação de transformação de proteção de 1:400. Portanto, com a aplicação de 3A na caixa de testes, foi o suficiente para sensibilizar a função de proteção instantânea de sobrecorrente configurada no VIED, que estava configurado com o *pick-up* de 1200A.

Conforme já abordado no item 3.3.1, que contextualiza a lógica de programação do VIED, foi criado um *dataset* chamado "VIRTUAL_1CFG" para enviar a mensagem GO-OSE dos seguintes LNs: *PRO.P1TPIOC1.Str.general*, *PRO.TRIPPTRC1.Tr.general* e *PRO.BRF1RBRF1.OpEx.general* (definidos necessariamente nesta ordem no dataset). São relacionados ao valor lógico da função 50; o sinal de *trip* e o valor lógico da função falha de disjuntor, respectivamente.

Figura 51 – Tela de interface do VIED com função de proteção 50



Fonte: Elaborado pelo autor.

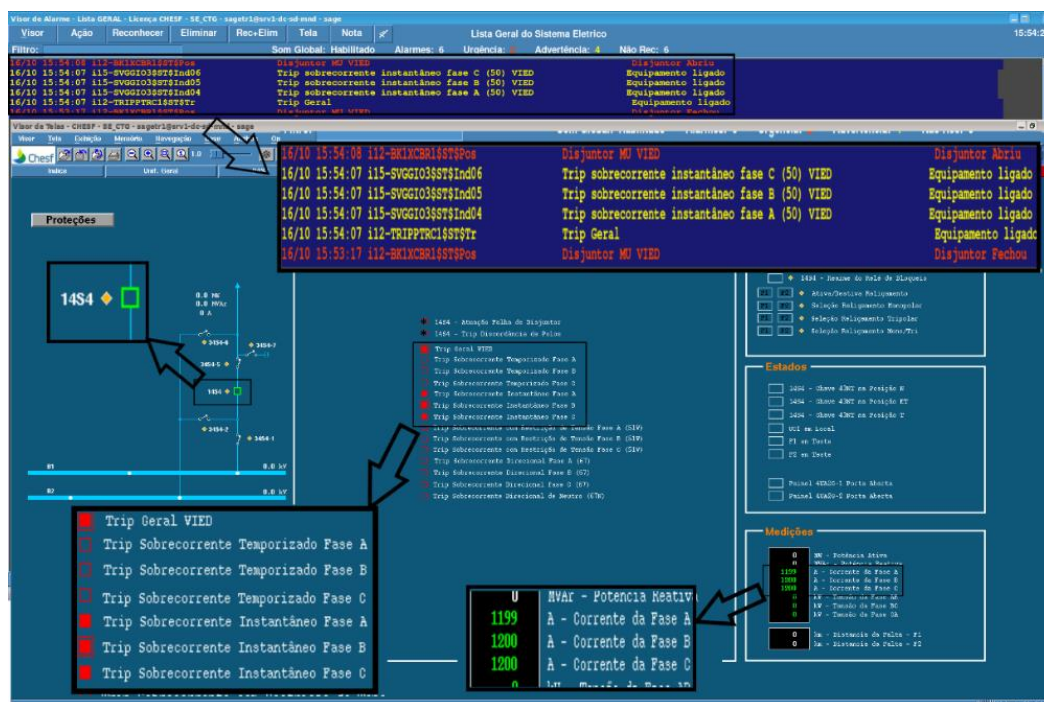
Esses pontos são assinados pelo SCADA SAGE e destacados na tela do vão (Figura 52) quando da atuação dessa proteção de sobrecorrente instantânea. Os pontos de destaque da tela são o Visor de Alarmes com os alarmes em ordem cronológica. No Unifilar a sinalização do disjuntor aberto, no Anunciador os alarmes destacando um Trip Geral e as fases referentes a proteção testada. Finalizando que na parte de Medições, os valores de corrente que antecederam a falta são visualizados.

5.2.3 Função 51

Esse teste será feito para a função sobrecorrente temporizada de fase, visando observar o disparo coordenado do VIED de acordo com os parâmetros da função escolhida. Para

começar, deve-se, com o auxílio da planilha CURVA RELES, calcular o tempo de atuação da proteção. Os parâmetros para a realização dos testes foram a corrente de *pick-up* (I_{cc}) de 1200A, a relação de transformação de corrente (RTC) de 400:1, a curva IEC que será do tipo MUITO INVERSA.

Figura 52 – Tela do SAGE com o resultado do VIED atuando a proteção 50

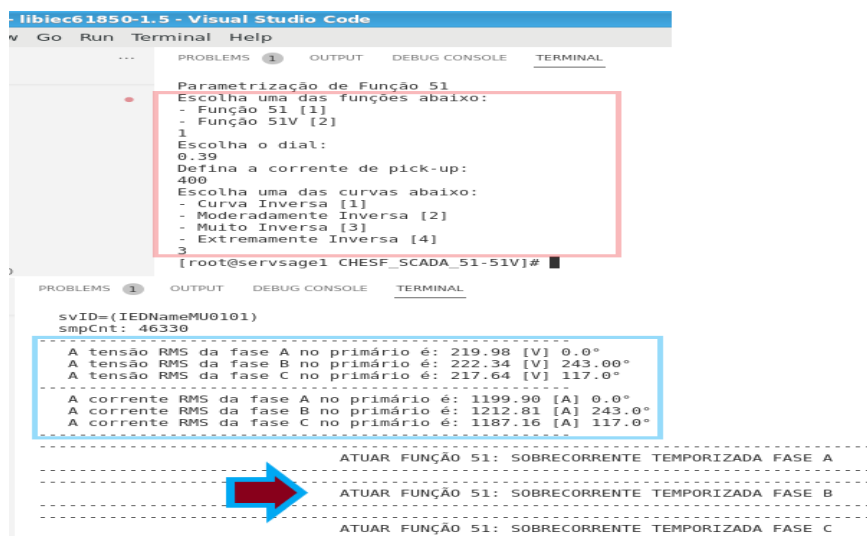


Fonte: Elaborado pelo autor.

Com os parâmetros informados, o tempo de atuação da proteção foi estabelecido em 2.63 s. Os mesmos valores também são agregados aos parâmetros solicitados, quando se começa a executar o VIED. A execução do relé virtual na interface do programa é mostrada na Figura 53, com destaque em vermelho das informações de parametrização, em azul as medições que estão sendo analisadas pelo VIED e a seta indica o disparo do *trip* trifásico da função de proteção.

Esse teste feito para a função de sobrecorrente temporizada, visa observar o comportamento do VIED quanto do tempo de atuação, se realmente está de acordo com o que foi simulado na planilha.

Figura 53 – Tela de interface do VIED com função de proteção 51

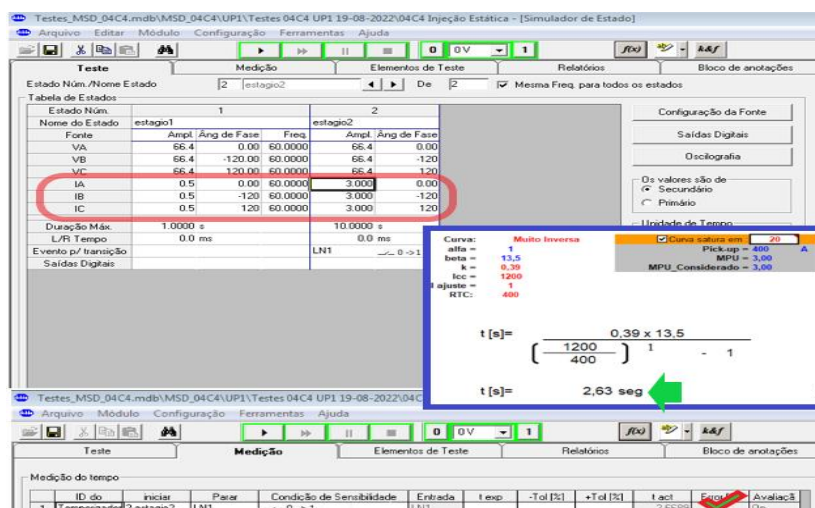


Fonte: Elaborado pelo autor.

Na Figura 54, as correntes elétricas analógicas são injetadas pela caixa de testes de acordo com a sequência destacada em vermelho. A MU digitalizou os dados e enviou para o barramento de processos em SV, para leitura do VIED e execução dos algoritmos. Com a atuação de *trip*, devido ao *pick-up* que estava configurado de 1200A, o tempo calculado pela planilha (destaque em azul) e o verificado pela caixa de testes (2,6689s) foram considerados praticamente idênticos e satisfatórios.

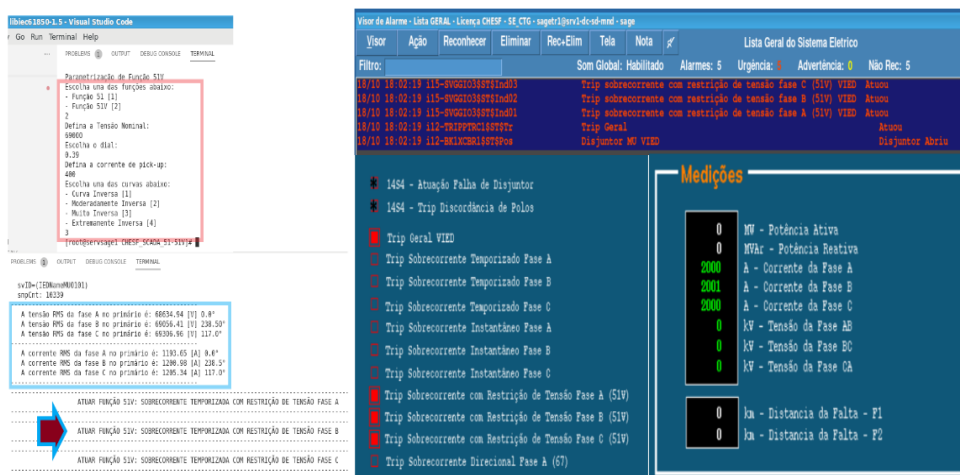
No SCADA SAGE, podem ver as atuações no Nível 2, demonstrado na Figura 55. Em destaque, da mesma forma que foi feito para o teste anterior, o Visor de Alarmes, o Anunciador os alarmes com a falta que foi testada e as medições e os valores de corrente.

Figura 54 – Cálculo do tempo de atuação com valores ajustados



Fonte: Elaborado pelo autor.

Figura 56 – Tela de interface do VIED com função de proteção 51V



Fonte: Elaborado pelo autor.

5.2.5 Função 67

Para a função sobrecorrente direcional, o teste é muito parecido com o da função 51, porém estamos observando a direcionalidade que é fornecida pelo vetor de tensão, de acordo com os parâmetros escolhidos. Para começar, deve-se com o auxílio da planilha CURVA RELES calcular o tempo de atuação da proteção. Na Figura 57, podemos ver em destaque vermelho os parâmetros para a realização dos testes. Primeiro, o ângulo de torque máximo definido em 30°, o dial de 0,39, a corrente de *pick-up* (I_{cc}) de 400 A e a curva IEC que será a MUITO INVERSA. Com os parâmetros informados, o tempo de atuação da proteção foi estabelecido em 5,27s, os valores que foram medidos pelo VIED estão destacados em azul e o disparo das proteções, no caso um *trip* trifásico, indicado pela seta.

Figura 57 – Tela de interface do VIED com função de proteção 67

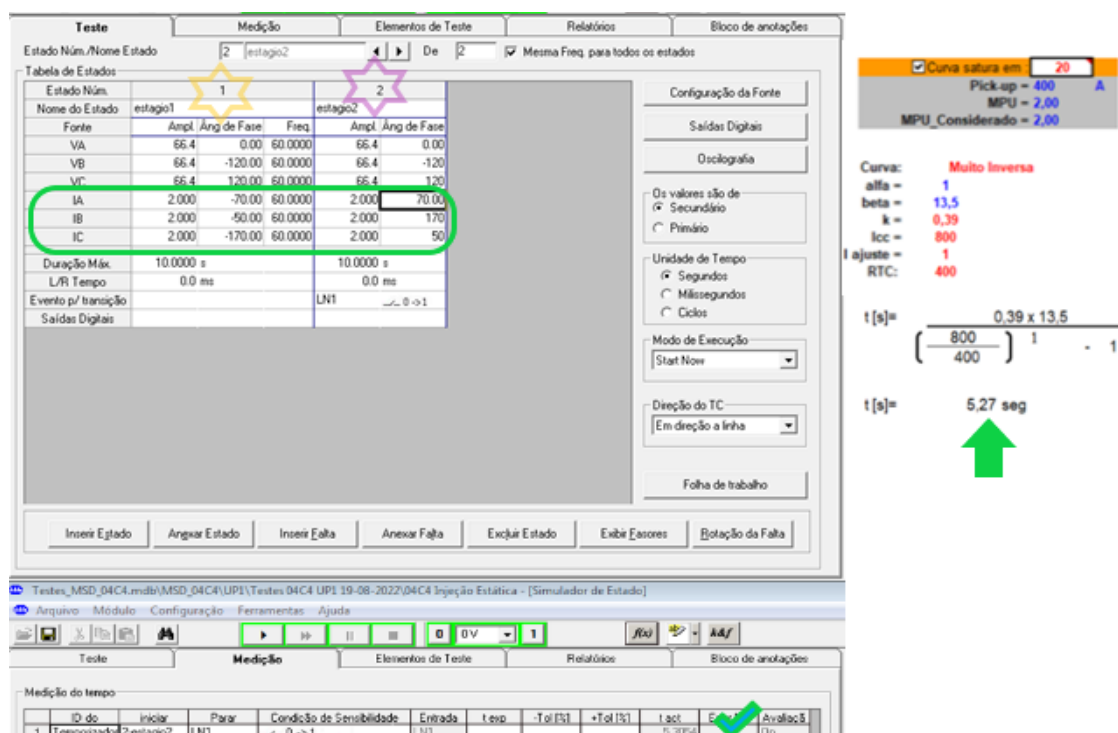


Fonte: Elaborado pelo autor.

Novamente, foi elaborado um novo teste, modificando somente a corrente de *pick-up* para 800A, com a intenção de testar a lógica de programação do VIED. O teste foi elaborado com dois estágios, de forma que se tivéssemos um relé sem direcionalidade, o disparo acontecia no primeiro estágio. Porém, como um validador do teste é a verificação da direcionalidade, os dois estágios diferenciavam nos ângulos de fase, dando uma característica da mudança do fluxo da corrente na falta. Na Figura 58, é mostrado o cenário do teste na mala de injeção de medidas analógicas, destacado em verde, o tempo de atuação da proteção calculado com a seta verde e o resultado alcançado na seta (5,3054s). A diferença é mínima, dando confiabilidade nos testes e na programação do VIED.

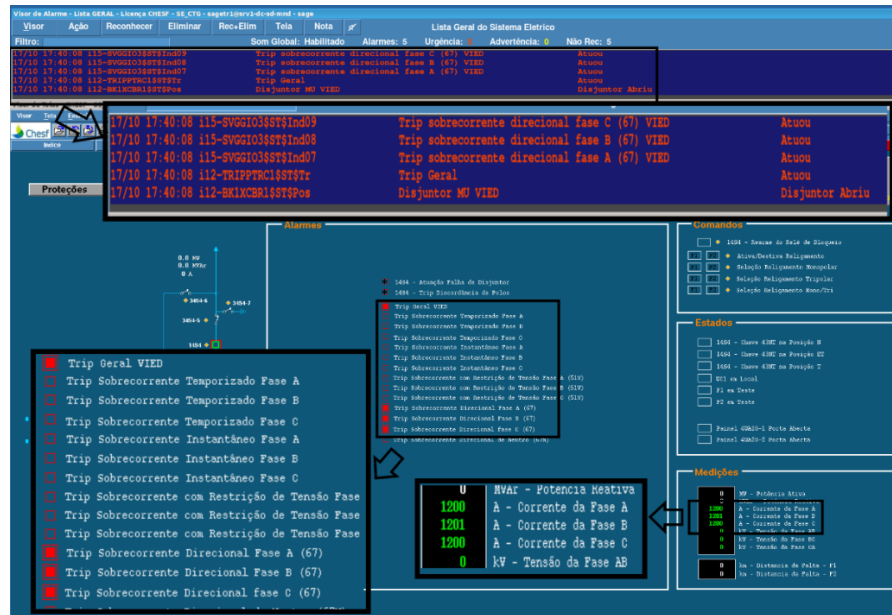
No SCADA SAGE se pode ver a atuação dessa proteção no Nível 2 (Figura 59), onde destacamos novamente os alarmes na lista, no Quadro e as medições de corrente. Nesse último caso, utilizamos um cenário parecido com o da figura anterior, sendo modificado somente o valor do *pick-up*, que passou para 1200A.

Figura 58 – Cálculo do tempo de atuação com valores ajustados



Fonte: Elaborado pelo autor.

Figura 59 – Tela do SAGE com os disparos da função 67

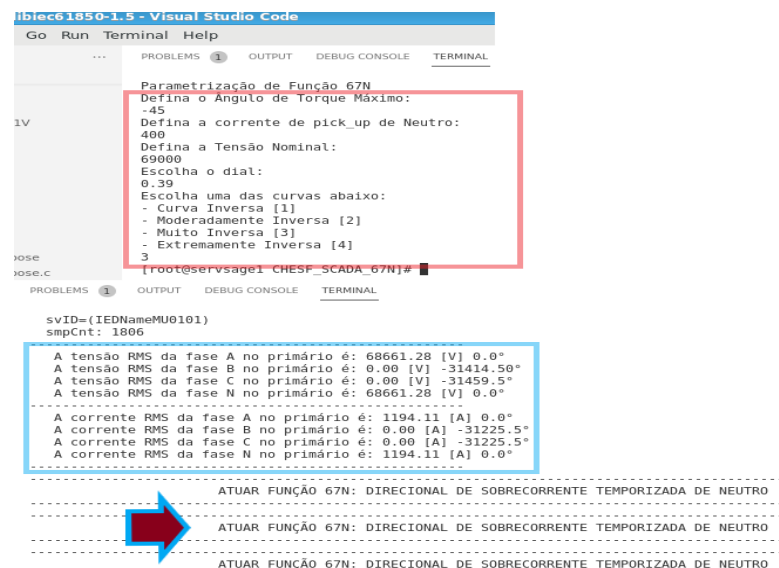


Fonte: Elaborado pelo autor.

5.2.6 Função 67N

Para a função sobrecorrente direcional temporizada de neutro, foi repetido o processo idêntico do teste anterior, da função 67. A interface do VIED é mostrada na Figura 60, com os destaques em vermelho para a inclusão dos dados da parametrização, em azul as medições feitas em tempo real e na seta o disparo quando alcançado a corrente de *pick-up*.

Figura 60 – Tela de interface do VIED com função de proteção 67N



Fonte: Elaborado pelo autor.

Os resultados foram bastante satisfatórios, sendo que é importante realizar um cenário comparativo com o IED físico. Esses testes serão demonstrados a seguir.

5.3 Testes do VIED comparados com IED Físico

Na plataforma de testes, foram utilizados um IED do fabricante SIEMENS e o VIED, na intenção de compararmos a atuação de proteção desses equipamentos, devidamente parametrizados, verificando os seus resultados. A mala de testes foi configurada no seu ambiente gráfico F6Test, onde foram utilizadas 3 saídas de tensão, simulando o barramento principal dos vãos de uma subestação e as 6 saídas de corrente. São 3 correntes de fase que passam por um vão protegido pelo IED, e as outras 3 correntes que vão para MU que digitaliza as medidas e envia ao VIED que protege outro vão.

O *trip* enviado pelos equipamentos testados são disparados pelas BOs do IED e da MU, acionando os relés biestáveis (simulando os disjuntores dos vãos). As informações de abertura desses relés biestáveis são critérios para parada da mala. Os contatos acionam duas entradas digitais na mala de testes, configurada para desligar a injeção dos valores analógicos, cronometrando o tempo do disparo da falta até a extinção do problema, que é exatamente o tempo necessário para que o evento protegido fosse desligado depois de um curto-circuito. Por fim, os testes foram no âmbito das proteções de corrente instantânea e temporizada.

5.3.1 Testes com a função de sobrecorrente instantânea (50)

O teste comparativo, com a utilização da lógica que simula a função de sobrecorrente instantânea foi elaborado com o seguinte cenário, destacando que o disparo da falta será trifásico. O teste tem as seguintes características, que são visualizados na Figura 61.

Figura 61 – Testes de Proteção VIED-IED com Proteção 50 Trifásico

Teste Saídas Digitais Medição Relatório Bloco de Anotações

No. do Estado/Nome Estado: 2 Postfault De 2 ☒ Mesma frequência para todos os estados

Visualização da Tabela

No. do Estado	Nome do Estado	Fonte	Ampl.	Âng de Fase	Freq.	Ampl.	Âng de Fase
1	Prefault						
2	Postfault						
		VA	69.00	0.00	60.000	69.00	0.00
		VB	69.00	-120.00	60.000	69.00	-120.00
		VC	69.00	120.00	60.000	69.00	120.00
		IA	0.500	0.00	60.000	5.000	0.00
		IB	0.500	-120.00	60.000	5.000	-120.00
		IC	0.500	120.00	60.000	5.000	120.00
		I1	0.500	0.00	60.000	5.000	0.00
		I2	0.500	-120.00	60.000	5.000	-120.00
		I3	0.500	120.00	60.000	5.000	120.00
		Duração Máx.	2 s			5 s	
		L/R Tempo	0 ms			0 ms	
		Transição do Disp.					
		Saídas Digitais					

Configuração da Fonte: 3 Voltages and 6 Currents ☐ Mostrar Configuração

Unidade de Tempo: ☒ Segundos ☐ Milissegundos ☐ Ciclos

Modo de Execução: Start Now

Inserir Estado Anexar Estado Inserir Falta Anexar Falta Excluir Estado Exibir Fatores Botão da Falta

Teste Saídas Digitais Medição Relatório Bloco de Anotações

Visualização da Medição

Temporizador	Iniciar	Parar	Deteçãoção Condição	Input No	Temp s	Tol s	Tol s	Tact s	Temp	Aval.
SIEMENS	2-Postfault	Trip B	0->C	2				0.0868	0.0452	Op
VIED	2-Postfault	Trip A	0->C	1				0.0502	0.0102	Op

Inserir Anexar Excluir Limpar Resultados

Fonte: Elaborado pelo autor.

- A relação de transformação de corrente de 2000/5 A para todas as saídas de corrente analógicas da mala de testes;
- Tensão trifásica equilibrada, com os ângulos equilibrados defasados de 120°, simulando a medição secundária por fase de um barramento de 230 kV (destaque em vermelho);
- No 1 estado (destacado com estrela verde), que é a pré-falta, a corrente injetada de 0,5 A equilibrada durante 2 segundos, simulando as medições primárias de um TC em 200 A no primário;
- No 2 estado (destaque com estrela vermelha), que é o curto-circuito, permanece por 5 segundos ou até quando o estado dos relés biestáveis for modificado, em virtude do disparo de *trip*. Tão logo termine o 1 estado de injeção pela mala

de testes, o valor de 5 A é injetado nas entradas analógicas do IED (destaque em preto) e na MU (destaque em azul). Esse valor corresponde a uma medição primária de 2000 A no primário do TC.

Os valores encontrados de tempo, desde o disparo do *trip* dos equipamentos de proteção até o *feedback* dos contatos dos biestáveis, simulando a abertura do disjuntor e consequentemente a extinção do curto-circuito, são de 0,0868 s para IED e 0,0502 s o VIED. Na Figura 61 está destacado esses tempos com as setas vermelhas. Esses pequenos valores se referem a uma atuação instantânea apesar dos tempos maiores que zero segundos. Foi constatado que os valores comparados eram um pouco diferentes, devido às características das lógicas de programação do VIED e do *firmware* do IED e os aspectos construtivos dos relés biestáveis, que são de fabricantes distintos.

5.3.2 Testes com a função de sobrecorrente temporizada (51)

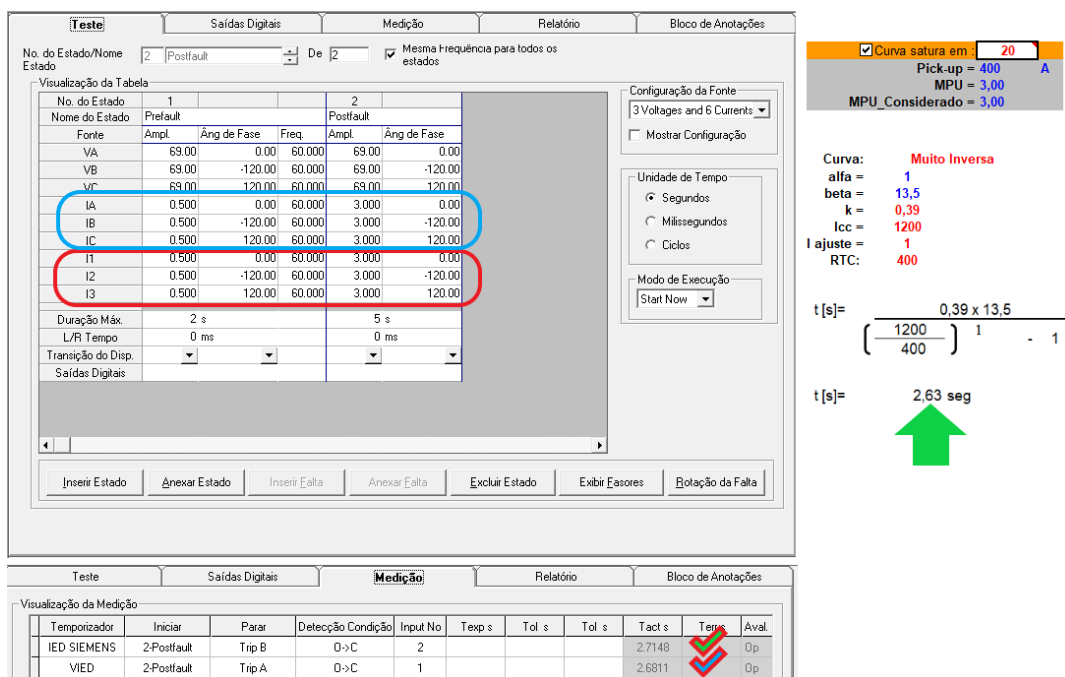
5.3.2.1 Falta Trifásica

A função de sobrecorrente temporizada trifásica foi elaborada de acordo com as características abaixo e visualizada na Figura 62:

- a) A relação de transformação de corrente é de 400/1 A (ou 2000/5 A, proporcionalmente) como informado na planilha de cálculo. Esse valor será utilizado para todas as saídas de corrente analógicas da mala de testes;
- b) Os cálculos para o tempo de atuação do relé (destacado com a seta verde na figura) foram elaborados com os dados de uma curva IEC do tipo Muito Inversa, o múltiplo sendo o valor de 1200/400 A e os demais dados são constantes que variam com o tipo da curva. O valor de tempo de atuação temporizada ficou em 2,63 s, que significa que se a corrente de *pick-up* permanecer durante esse tempo, o relé irá executar o *trip*;
- c) Tensão trifásica equilibrada, simulando a medição secundária de um TP por fase em 230 kV;
- d) No primeiro estágio da injeção de corrente. o da pré-falta, é feito de forma que as correntes e tensões estejam equilibradas, simulando as medições secundárias de um TC por fase em 0,5 A, no primário seria de 200 A;

- e) Após 2 segundos, a falta trifásica é simulada com a injeção de 3 A no conjunto de terminais analógicos do IED (destaque em azul) e do VIED (destaque em vermelho), com a relação de transformação escolhida – é como se estivesse simulando um fluxo de 1200 A no primário dos TCs dos vãos.

Figura 62 – Testes de Proteção VIED-IED com Proteção 51 Trifásico



Fonte: Elaborado pelo autor.

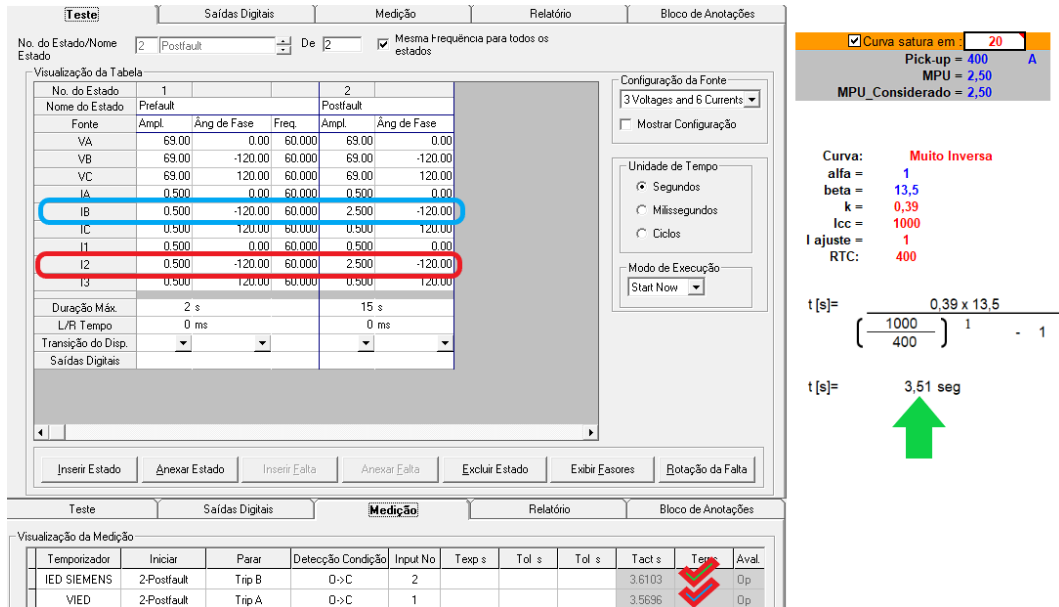
Os valores encontrados de tempo de atuação nos testes do VIED e do relé da SIEMENS são muito parecidos e compatíveis com o que foi calculado na planilha CURVA RELES. Conseguiram-se 2,7148s para IED e 2,6811s o VIED. Na Figura 62 estão destacados esses tempos com as setas vermelhas. Esses valores são bem parecidos com os da planilha, pois temos o *feedback* muito rápido dos relés biestáveis. Caso precisássemos usar disjuntores em campo, poderíamos ter o incremento de mais 50 milissegundos no total desse tempo medido. Esse acréscimo seria o tempo de acionamento das bobinas de abertura dos equipamentos de pátio real.

5.3.2.2 Falta monofásica Fase B

Foi testada a função de sobrecorrente temporizada para somente uma das fases, porém foram modificadas algumas características, no intuito de cobrir pontos diferentes de

tempo de atuação, nas curvas IEC. A falta monofásica na fase B está bem representada na Figura 63 e foi elaborada com o seguinte cenário.

Figura 63 – Testes de Proteção VIED-IED com Proteção 51 Fase B



Fonte: Elaborado pelo autor.

- A relação de transformação de corrente é a mesma do teste anterior, de 400/1 A. Esse valor será utilizado para todas as saídas de corrente analógicas da mala de testes;
- Os cálculos para o tempo de atuação do relé (destacado com a seta verde na figura) foram elaborados com os dados de uma curva IEC do tipo Muito Inversa; o múltiplo desta vez será calculado para um *pick-up* de 1000A, com isso se verifica outro ponto da curva e checa-se o algoritmo dos relés, ficando o valor de 1000/400 A. Os demais dados são constantes do tipo de curva. O valor de tempo de atuação temporizada ficou em 3,51 s (destacado pela seta verde), que significa que, se a corrente de *pick-up* permanecer durante esse tempo, os relés irão atuar;
- Tensão trifásica equilibrada, simulando a medição secundária de um TP por fase em 230 kV;
- No primeiro estágio da injeção de corrente, o da pré-falta, é feito de forma que as correntes e tensões estejam equilibradas, simulando as medições secundárias de um TC por fase em 0,5 A, no primário seria de 200 A;

- e) Após 2 segundos, a falta monofásica é simulada com a injeção de 2,5 A na fase B, deixando a fase A e C com os mesmos valores de corrente de pré-falta. A mudança dos valores de corrente da fase B no teste é destacada em azul, na Figura 60, para o IED e em cor vermelha para o VIED. Com a relação de transformação escolhida, é como se estivesse simulando um fluxo de 1000 A no primário da fase B dos TCs dos vãos;

Os valores encontrados de tempo para os disparos de *trip* são 3,6103s para o IED e 3,5696s no VIED, conforme destacado nas setas vermelhas. Esses valores estão compatíveis com os cálculos feitos de 3,51s da planilha. Vale ressaltar que os ajustes para proteção são os parecidos com os encontrados no teste anterior, somente o item de corrente de *pick-up* foi modificado e o disparo que se escolheu a fase B. Pela característica da curva escolhida, nota-se que o comportamento dos relés está correto, relacionando um tempo maior de atuação para uma falta com corrente de magnitude mais baixa.

5.3.2.3 Falta monofásica Fase C

Utilizando o mesmo cenário de testes anterior, fazendo pequenas modificações para se cobrir uma gama maior de testes, foi testada a função de sobrecorrente temporizada monofásica para fase C, desta vez com as seguintes características:

- a) A relação de transformação de corrente é a mesma do teste anterior, de 400/1 A;
- b) Os cálculos para o tempo de atuação do relé, destacado com a seta verde (Figura 64), foram elaborados com os dados de uma curva IEC do tipo Muito Inversa, o múltiplo desta vez será calculado para um *pick-up* de 2000A, ficando com o valor de 2000/400A. De acordo com os cálculos, o valor de tempo de atuação temporizada ficou em 1,32 s (destacado pela seta verde);
- c) Tensão trifásica equilibrada, simulando a medição secundária de um TP por fase em 230 kV;
- d) No primeiro estágio da injeção de corrente, o da pré-falta, é feito de forma que as correntes e tensões estejam equilibradas, simulando as medições secundárias de um TC por fase em 0,5 A, no primário seria de 200 A;
- e) Após 2 segundos, a falta monofásica é simulada com a injeção de 5 A na fase C, deixando a fase A e B com os valores constantes iguais o da pré-falta. Na Figura 64 se verifica a mudança dos valores de corrente da fase C, onde está

6 CONCLUSÃO E TRABALHOS FUTUROS

6.1 Conclusão

A plataforma do VIED proposta neste trabalho aprimora outros que o antecederam, fortalecendo a ideia de que a construção de um ambiente virtual com VIEDs e IEDs físicos poderão em breve estar trabalhando em conjunto, em algumas subestações de concessionárias. A plataforma concebida abordou questões importantes sobre as configurações de um servidor físico com seu respectivo VIED, a especificação do código necessário à implantação das lógicas de proteção e controle, as funções de proteção testadas e validadas.

Os vários testes de proteção e controle desenvolvidos mostraram que o VIED se comunica com outros dispositivos da plataforma, através de mensagens GOOSE, MMS e SV. Nos testes de simulação de faltas monofásicas e trifásicas, os VIEDs atuaram satisfatoriamente. Quando comparados com um IED físico do fabricante SIEMENS, as pequenas disparidades de tempo encontradas foram atribuídas aos relés biestáveis que eram de fabricantes diferentes e de algumas lógicas de controle e processamento de sinais que estão contidos nos algoritmos dos relés físicos. Analisando as possibilidades e os testes que foram realizados, pode-se constatar que as plataformas com VIEDs serão mais bem estudadas e que a virtualização de relés físicos, poderá, em alguns casos, substituir os IEDs, com a mesma confiabilidade em várias aplicações no setor elétrico e nas subestações digitais.

A integração do VIED com o sistema SCADA SAGE demonstrou um bom potencial de virtualização no sistema operacional onde o supervisório está instalado, mostrando o potencial do dispositivo proposto para usos nas empresas do setor elétrico. Pode-se ainda relacionar os seguintes itens:

- a) Foi possível validar a comunicação por MMS, agregar os alarmes e controle através de telas desenhadas de um vão e lista de eventos. O SCADA SAGE, que já constitui em uma solução unificadora para os diversos níveis hierárquicos em que se organiza a operação em tempo real de sistemas elétricos, permite tornar uniforme o processo de expansão da automação e de manutenção dos sistemas existentes, agregando funções dos VIEDs, o que poderá ser um salto qualitativo na prestação de serviços e manutenção dos sistemas de potência;
- b) Com a MU da GE consegue-se simular a digitalização das grandezas elétricas no pátio, que são trazidas para sala de comando aos IEDs físicos, através do SV. Da mesma forma foi dotada essa função ao VIED, garantindo excelentes

simulações de faltas reais de sobrecorrente no SEP, com a mala de testes da OMICRON, em um primeiro momento, e depois a DOBLE. Assim, concluímos que foram excelentes os resultados de aquisição dessas grandezas analógicas, devido ao bom desempenho dos testes de proteção, inclusive, quando comparado o VIED com o IED da Siemens;

- c) O sincronismo de tempo, realizados por PTP ou SNTP, redundantes, garante em uma subestação com MUs, a simultaneidade necessária para a correta referência temporal, na qual depende os fasores de corrente e tensão digitalizados. Pelo fato de termos uma bancada de forma experimental, não foi utilizado nenhum método para garantia do sincronismo para a arquitetura proposta. Esse aspecto deve ser um pré-requisito para nortear futuros trabalhos;
- d) Na elaboração da plataforma e realização dos testes, nota-se que os conceitos de proteção, controle e da própria norma IEC 61850 foram bastante discutidos. Com isso concluímos que o arranjo feito, de maneira permanente, em laboratórios tanto na Eletrobrás CHESF como nas Universidades pode ser aplicado a treinamento das equipes e ao ensino dos alunos, pois apresenta muitos ganhos em aulas expositivas de PAC nas disciplinas teóricas de Sistemas Elétricos de Potência.;
- e) Constata que um bom computador para simular o VIED suporta a geração e transmissão de tráfego MMS, GOOSE e SV, sem muita latência nas comunicações da rede e na execução dos códigos. O uso da libIEC61850, com códigos mais elaborados das funções de proteção, faz parte da evolução constante do SAS, podendo trazer em breve uma solução para VIED agregado a uma instalação.

6.2 Trabalhos Futuros

Como exposto, foram alcançadas algumas melhorias baseadas nos trabalhos anteriores, representando uma boa evolução nos aspectos de proteção, controle e supervisão. Alguns pontos podem ser melhorados, no contexto da Norma IEC 61850, e aprofundados nesse mesmo tema do trabalho desenvolvido:

- a) Uma GUI mais interativa, via *web*, para que os usuários tenham maior agilidade, na parametrização de VIEDs, agregando funções às plataformas de testes, sem necessidade dos conhecimentos de programação;

- b) Aumentar o número de VIEDs em testes da plataforma, em vários computadores, ou num único servidor. Isso gera mais complexidade dos testes feitos neste trabalho. Exigiria um melhor conhecimento de VMs e *containers*, agregados a uma sincronização de tempo para todos na rede;
- c) Utilização de serviços de autenticação e senhas de acesso aos VIEDs, de acordo com os conceitos de cibersegurança, prevenindo ataques cibernéticos;
- d) Desenvolver VIEDs com funções de proteção mais robustas, como a proteção de distância (21), diferencial de linha (87L), diferencial de barra (87B) e outras, nos quais se avaliaria o seu desempenho e agregaria mais parâmetros de inicialização, tornando bem mais versáteis esses equipamentos virtuais;
- e) Inclusão das funcionalidades de verificação, edição e mudança de grupos de ajustes na plataforma;
- f) Ferramenta de gerenciamento remoto para diversas instâncias espalhadas dos VIEDs, verificando o tráfego de mensagens, elaborando VLANs para melhorar os aspectos de velocidade;
- g) Elaboração de uma GUI para MU sintética, com isso se tem mais facilidade para simulações com protocolo SV;
- h) Enfim, o trabalho pode ser disponibilizado às equipes de estudos e pesquisas do CEPEL, como insumos para homologação de futuros VIEDs e desenvolvimento de aplicações e ferramentas para o SAGE.

REFERÊNCIAS

ALSTOM GRID. **Network Protection & Automation Guide**. 1 ed., [S.I]: Alstom Grid, 2011. ISBN 978-0-9568678-0-3.

ANSARI, S. *et al.* Towards virtualization of operational technology to enable large-scale system testing. *In: IEEE EUROCON 2019 - INTERNATIONAL CONFERENCE ON SMART TECHNOLOGIES*, 18, 2019, Novi Sad. **Proceedings [...]**. Piscataway: IEEE, 2019, p. 1-5. DOI: 10.1109/EUROCON. 2019.8861980. Disponível em: <https://ieeexplore.ieee.org/document/8861980>.

APOSTOLOV, A. P. UML and XML use in IEC 61850. *In: IEEE PES TRANSMISSION AND DISTRIBUTION CONFERENCE AND EXPOSITION*, 2010, New Orleans. **Proceedings [...]**. Piscataway: IEEE, 2010, p. 1-6. DOI: 10.1109/TDC. 2010.5484325. Acesso em: 22 jan. 2022.

BABU, P. Directional Over Current Relay: 67. **Electrical Engineering Materials**, 2019 Disponível em: <https://electengmaterials.com/directional-over-current-relay-67/>. Acesso em: 22 jan. 2022.

CEPEL. **Visão Geral**. 2023. Disponível em: <https://sage.cepel.br/index.php/pt/sage/visao-geral>, 2023a. Acesso em: 11 set. 2023.

_____. **Características Técnicas**. 2023. Disponível em: <https://sage.cepel.br/index.php/pt/sage/caracteristicas-tecnicas>, 2023b. Acesso em: 10 set. 2023.

_____. **Protocolos**. 2023. Disponível em: <https://sage.cepel.br/index.php/pt/documentacao/protocolos>, 2023c. Acesso em: 10 set. 2023.

_____. **Arquitetura de Software**. 2023. Disponível em: <https://sage.cepel.br/index.php/pt/sage/arquitetura-de-software>, 2023d. Acesso em: 10 set. 2023.

_____. **SAGE: Manual de Configuração da Base Fonte: Revisão 29**, Rio de Janeiro, RJ, Brasil, 2019. 2023e.

_____. **SAGE: Manual de Administração: Revisão 52**, Rio de Janeiro, RJ, Brasil, 2019. 2023f.

_____. **SAGE: Anexo de Configuração 17: Comunicação com IEDs em Protocolo IEC 61850 – Revisão 24**, Rio de Janeiro, RJ, Brasil, 2021.

CIGRÉ. **Aplicações da Norma IEC 61850: Sistemas de Automação Operando com Redes de Comunicação**. 2018. (Brochura Técnica 026). Disponível em: <https://cigre.org.br/brochuras>. Acesso em jun 2022.

_____. **Protection, Automation and Control Architectures with functionality independent of hardware**. 2023. (Technical Brochures) Disponível em: <https://www.e->

cigre.org/publications/detail/891-protection-automation-and-control-architectures-with-functionality-independent-of-hardware.html. Acesso em 5 mar 2023.

CHEMIN NETTO, U. **Determinação de um parâmetro para monitoramento do desempenho de mensagens GOOSE do padrão IEC 61850 utilizadas em subestações de energia elétrica**. 2012. 211f. Tese (Doutorado em Sistemas Elétricos de Potência) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2012.

CHESF. **Instrução Operativa de Proteção da SE Sobral III: IOP SBT**. Fortaleza. 2012. Documento Interno.

CHESF. **Arquitetura SAGE Nível 2 em Sistemas Digitais Areva**. São Paulo. 2011. Apostila de Treinamento.

DOCKER. What's a container? **Docker**. 2023. Disponível em: <https://docs.docker.com/guides/walkthroughs/what-is-a-container/>. Acesso em: 12 mar. 2023.

EGGERS, R. P. **Análise do protocolo IEC-61850 em ambientes com elevado tráfego de informação**. 2019. 56f. Trabalho de Conclusão de Curso (Graduação em Engenharia Elétrica) – Universidade Federal de Santa Catarina, Centro Tecnológico, Florianópolis, 2019.

FREITAS, C. D. *et al.* **Plataforma para ensino de proteção de sistemas elétricos de potência com IEDs reais e virtuais baseados na norma IEC 61850**. Fortaleza: Universidade Federal do Ceará, 2022a.

FREITAS, A. D. S. **Plataforma aberta para ensino de proteção de sistemas elétricos de potência com IEDs físicos e virtuais padrão IEC 61850 2022**. 119 f. Dissertação (Trabalho de Conclusão de Curso) – Universidade Federal do Ceará, Centro de Tecnologia, Curso de Engenharia Elétrica, Fortaleza, 2022b.

GUISE, L. **IEC 61850: Overall Documents Structure**. 2022. IEC Mapping plataforma. Disponível em: <https://mapping.iec.ch/#/maps/21>. Acesso em: 19 jan. 2023.

HITACHI, ABB POWER GRIDS. **Treinamento de Manutenção: Process Bus**. Zurique, Suíça, 2023, Apostila de Treinamento da Manutenção.

HOGA, C. **High Availability Network Solutions and Their Usability in Substation Applications**. 2010. Disponível em: <https://www.pacw.org/previous-magazines-2010-flippable>. Acesso em: 3 jan. 2023.

INTERNATIONAL ELECTROTECHNICAL COMMISSION. **IEC 61850-1: Communication networks and systems in substations: Part 1: Introduction and overview**. 2003a. Disponível em: <http://webstore.iec.ch/webstore/webstore.nsf/artnum/033549!opendocument>. Acesso em: 20 abr. 2022.

_____. **IEC 61850-6: Communication networks and systems in substations: Part 6: Configuration description language for communication in electrical substations related to IEDs**, 2003b. Disponível em: <https://webstore.iec.ch/publication/63319>. Acesso em: 16 jun. 2022.

_____. **IEC 61850-7-1:** Communication networks and systems in substations: Part 7-1: Basic communication structure for substation and feeder equipment - Principles and Models. 2003c. Disponível em: <https://webstore.iec.ch/publication/67536>. Acesso em: 15 jun. 2022.

_____. **ON. IEC 61850-8-1:** Communication networks and systems for power utility automation: Part 8-1: Specific communication service mapping (SCSM) - Mappings to MMS (ISO 9506-1 and ISO 9506-2) and to ISO/IEC 8802-3. 2013. Disponível em: <https://webstore.iec.ch/publication/6021>. Acesso em: 23 jul. 2022.

_____. **IEC 61850-90-8:** Communication networks and systems for power utility automation - Part 90-8: Object model for E-mobility. 2016. Disponível em: <https://webstore.iec.ch/publication/24475>. Acesso em: 25 jun. 2022.

_____. **IEC 61850-90-9:** Communication networks and systems for power utility automation - Part 90-9: Use of IEC 61850 for electrical energy storage systems. 2020. Disponível em: <https://webstore.iec.ch/publication/29365>. Acesso em: 10 nov. 2022.

_____. **IEC 61850:2023 SER Series:** Communication networks and systems for power utility automation - ALL PARTS. 2023. Disponível em: <https://webstore.iec.ch/publication/6028#additionalinfo>. Acesso em: 28 out. 2023.

JESUS, L. S. **Avaliação da parcela variável para apuração da qualidade do serviço de transmissão no Brasil segundo conceitos de confiabilidade.** 2008. 83 f. Trabalho de Conclusão de Curso (Graduação em Engenharia Elétrica) – Departamento de Engenharia Elétrica, Escola Politécnica, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2008.

JESUS, J.; LELLYS, D. Integração de projeto de virtualização para sistemas de controle e proteção. *In: SEMINÁRIO NACIONAL DE PRODUÇÃO E TRANSMISSÃO DE ENERGIA ELÉTRICA*, 27., 2023, Brasília, DF. **Anais [...]**. Brasília, DF: Grupo GPC, 2023.

KULMALA, A. *et al.* Virtualized wide-area protection for distribution networks and effective anti-islanding of distributed energy resources using 5G communication. *In: ANNUAL CONFERENCE FOR PROTECTIVE RELAY ENGINEERS*, 76., 2023, College Station. **Proceedings [...]**. Piscataway: IEEE, 2023. p. 1-4.

LELLYS, D. Perspectivas de aplicações de esquema de proteção e controle de área ampla (WAMS/WACS/WAPC) baseada na norma IEC 61850. *In: ENCONTRO ÍBERO-AMERICANO DO CIGRÉ*, 19., 2022, Foz do Iguaçu. **Anais [...]**. Foz do Iguaçu: CIGRÉ Brasil, 2022. Grupo CE B5.

LELLYS, D. *et al.* Process bus merging unit: conceito arquitetura e impacto na automação de subestações. *In: SEMINÁRIO NACIONAL DE DISTRIBUIÇÃO DE ENERGIA ELÉTRICA*, 24., 2020, São Paulo. **Anais [...]**. São Paulo: Abradee, 2020.

MELO, L.S. **Desenvolvimento de uma plataforma para implantação de sistemas multia-gentes com aplicação para recomposição automática de sistemas de distribuição de energia elétrica**. 2015. 121f. Dissertação (Mestrado em Engenharia Elétrica) – Centro de Tecnologia, Universidade Federal do Ceará, Fortaleza, 2015.

MYRDA, P.; FALK H. **Common substation platform**. 2021. Disponível em: <https://www.pacw.org/common-substation-platform>. Acesso em: 15 jan. 2022.

MU GE. **Manual técnico da merging unit: MU320E_TM_EN_1.6**. 2023. Disponível em: <https://www.gegridsolutions.com/multilin/catalog/mu320e.htm>. Acesso em: 01 jul. 2023.

_____. **Brochura técnica da merging unit: MU320**. 2022. Disponível em: https://www.gegridsolutions.com/products/brochures/grid-ga-l3-reason_mu320-10402017_02-pt_lo.pdf. Acesso em: 20 ago. 2022.

ONS. **Procedimentos de rede**: Submódulo 2.7: Requisitos mínimos de supervisão e controle para a operação. 2019. Disponível em: <https://www.ons.org.br/%2FProcedimentosDeRede%2FM%C3%B3dulo%202%2FSubm%C3%B3dulo%202.7%2FSubm%C3%B3dulo%202.7%202019.08.pdf>. Acesso em: 16 jan. 2023.

ROCHA, G. R. P.; OLIVEIRA, C.; FRANCO, P. Aumento da confiabilidade de esquemas de proteção e controle em IEC 61850 com o uso de message quality. **Schweitzer Engineering Laboratories**, 2020. Disponível em: <https://selinc.com/api/download/129644/?lang=pt>. Acesso em 30 mar. 2022.

SAMPAIO, R. F. **Sistema de automação distribuído**: Uma abordagem baseada em multia-gente aplicada a sistemas de distribuição de energia elétrica em média tensão. 2017. 225f. Tese (Doutorado em Engenharia Elétrica) - Centro de Tecnologia, Universidade Federal do Ceará, Fortaleza, 2017.

SILVA, C. H. C. **Desenvolvimento de um laboratório virtual para capacitação tecnológica a distância em proteção de sistemas elétricos**. 2011. 90f. Dissertação (Mestrado em Engenharia Elétrica) – Centro de Tecnologia, Universidade Federal do Ceará, Fortaleza, 2011.

SILVA, E. C. **Proteção de Sistemas Elétricos de Potência**: Guia Prático de Ajustes. Rio de Janeiro: Qualitymark, 2014. ISBN 978-85-414-0181-4.

SOARES, A. A. Z *et al.* Enabling emulation and evaluation of IEC 61850 networks with TITAN. **IEEE Access**, Piscataway, v. 9, p. 49788-49805, 2021. DOI: 10.1109/ACCESS.2021.3068366.

TANENBAUM, A. S; BOS, H. **Sistemas operacionais modernos**. 4 ed. São Paulo, São Paulo: Pearson, 2016.

UTILITY COMMUNICATIONS ARCHITECTURE. **UCA**: Implementation guidance for digital interface to instrument transformers using IEC 61850-9-2. 2004. Disponível em: <https://tc57wg10.info/downloads/digifspec921er21040707cb.pdf>. Acesso em: 08 nov. 2022.

VICENTE, D. T. **Aplicação dos padrões da norma IEC 61850 a subestações compartilhadas de transmissão/distribuição de energia elétrica**. 2011. 117f. Dissertação (Mestrado em Sistemas de Potência) – Escola Politécnica, Universidade de São Paulo, São Paulo, 2011.

WÓJTOWICZ, R.; KOWALIK, R.; RASOLOMAMPIONONA, D.D. Next generation of power system protection automation-virtualization of protection systems. **IEEE Transactions on Power Delivery**, v. 33, n. 4, p. 2002-2010, ago. 2018. DOI: 10.1109/TPWRD.2017.2786339.

_____. Virtualization of protection systems–tests performed on a large environment based on data center solutions. **IEEE Transactions of Power Delivery**, v. 37, n. 4, p. 3401-3411, ago. 2022. DOI: 10.1109/TPWRD.2021.3128993.

ZILLGITH, M. **libIEC61850**. 2021. Disponível em: <https://libiec61850.com/libiec61850-now-on-github/>. Acesso em: 20 fev. 2021.

APÊNDICE A – DESCRIÇÃO DAS ENTIDADES DO SCADA SAGE

A descrição breve das entidades serve para entender a estrutura do SCADA SAGE, baseado nos Manuais de Configuração da Base Fonte e nos Anexos de Configuração do Protocolo (CEPEL, 2019), (CEPEL, 2021). Na Tabela 7, temos o destaque nas entidades que compõem os Serviços de Comunicação de Dados (itens 1 a 7), Sistema de Aquisição e Controle (itens 8 a 22), finalizando em Sistema de Aquisição e Comunicação de Dados (itens 23 a 34) às entidades que foram implantadas na plataforma para comunicação com a MU e o VIED.

Tabela 7 – Lista de Entidade do SCADA SAGE

Item	Código	Entidade	Função
1	GSD	Gateway Scada	Configurar os gateways do sistema
2	CXU	Conexão de Comunicação com UTRs e Canais	Configurar as conexões de comunicação dos servidores
3	ENU	Enlace de Conexão com UTR	Configura os enlaces (canais de comunicação) de todas as conexões de todos os servidores do sistema
4	UTR	Unidade Terminal Remota	Configura as UTR de todas as conexões de todos os servidores do sistema, tanto as UTR físicas para aquisição, como as virtuais para distribuição.
5	MUL	Multiligação	Configura as multiligações de todos os servidores do sistema
6	ENM	Enlace de Multiligação	Configura os enlaces de todas as multiligações de todos os servidores do sistema
7	CNM	Canal Lógico de Multiligação	Configura os canais lógicos de todas as multiligações de todos os gateways do sistema
8	PSV	Preservação	Configura os dados que serão preservados quando da entrada em operação de uma nova versão da base de dados
9	LSC	Ligação SCADA	Configura as ligações virtuais lógicas de aquisição e controle, de distribuição e mistas existentes no sistema
10	INS	Instalação	Configura as diversas instalações do SAGE. A entidade INS aparece na modelagem de dados apenas para prover a ligação da base de dados SCADA com as funcionalidades, que englobam as funções de Análise de Redes e de Controle Automático de Geração

11	TAC	Terminal de Aquisição e Controle	Os terminais de aquisição e controle definem grupos lógicos de aquisição de uma mesma funcionalidade. A entidade TAC serve também para estabelecer bloqueio de controle, ou seja, apenas um ponto do terminal de aquisição e controle pode estar sendo controlado em um determinado instante
12	CGS	Ponto de Controle Genérico do SAC	Configura os pontos de controle lógicos de todos os terminais de aquisição e controle de todas as ligações SCADA do sistema
13	TCTL	Tipo de Controle	Especifica os tipos que o ponto de controle pode assumir na configuração do diálogo de controle
14	PAS	Ponto de Medição Analógica	Configura os pontos analógicos, adquiridos e calculados, de todos os terminais de aquisição e controle de todas as ligações SCADA do sistema.

continua
continuação

Item	Código	Entidade	Função
15	PIS	Ponto de Informação	Configura os pontos de informação lógicos de todos os terminais de aquisição de dados e controle de todas as ligações SCADA do sistema
16	PTS	Ponto Totalizado Lógico	Configura os pontos totalizadores, adquiridos e calculados, de todos os terminais de aquisição de dados e controle de todas as ligações SCADA do sistema
17	PDS	Ponto Digital	Configura os pontos digitais, adquiridos e calculados, de todos os terminais de aquisição de dados e controle de todas as ligações SCADA do sistema
18	RCA	Relação de cálculos do SCD	Especifica os relacionamentos “n” para “m” entre as variáveis de entrada das entidades PAS, PTS e PDS (pontos lógicos simples) e as variáveis de saída das entidades PAS, PTS e PDS (pontos lógicos calculados)
19	TCL	Tipos de Cálculos para pontos de medição	Configura os cálculos implementados no sistema
20	SITE	SITE	Entidade específica para funções de sincronismo
21	COS	Centro de Controle	Define um centro de controle de um SSC
22	SUBSIS	Subsistema Computacional	Subsistema computacional (base de dados) do SITE
23	CNF	Configuração da Ligação Física do SCD	Configura todas as configurações de aquisição e/ou distribuição associadas a todas as ligações SCADA

			do sistema.
24	NV1	Nível 1 da Configuração Física	Configura todas as entidades de nível 1 de todas as CNF
25	NV2	Nível 2 da Configuração Física	Configura todas as entidades de nível 2 de todas as entidades de nível 1 de todas as CNF
26	TN1	Tipo de Entidade Física do Nível 1	Esta entidade descreve os tipos de entidades de nível 1 que podem existir.
27	TN2	Tipo de Entidade Física do Nível 2	Esta entidade descreve os tipos de entidades de nível 2 que podem existir
28	CGF	Ponto de Controle Físico	Configura todos os pontos de controle físicos de aquisição e de distribuição associados a todas as configurações CNF de todas as ligações SCADA
29	PAF	Ponto Analógico Físico	Configura todos os pontos analógicos físicos de aquisição e de distribuição associados a todas as configurações CNF de todas as ligações SCADA
30	PDF	Ponto Digital Físico	Configura todos os pontos digitais físicos de aquisição e de distribuição associados a todas as configurações CNF de todas as ligações SCADA
31	PIF	Ponto de Informação Físico	Configura todos os pontos de informação físicos de aquisição associados às ligações cadastradas
32	PTF	Ponto Totalizado Físico	Configura todos os pontos totalizadores físicos de aquisição e de distribuição associados a todas as configurações de todas as ligações
33	RFI	Relação de Filtros Simples	Filtro simples é aquele cujas parcelas (pontos físicos) podem participar de apenas um único filtro gerando um único ponto lógico
34	RFC	Relação de Filtros Compostos	Filtro composto é aquele cujas parcelas (pontos físicos) podem participar de vários filtros gerando vários pontos lógicos

Fonte: Elaborado pelo autor.

APÊNDICE B – CÓDIGO EM LINGUAGEM C DOS VIEDS

Algoritmo geral do VIED com função de proteção 67N e 50/62BF, o código inteiro é o mesmo para todos os outros VIEDS apenas as linhas 98 a 128 e 281 a 322 pertencem à lógica responsável pela sobrecorrente direcional de neutro. Assim, após essa lógica, serão disponibilizadas as lógicas, que modificando as linhas mencionadas antes de cada código, habilitam o VIED para funcionar em outras funções de proteção.

```

1.  #include "iec61850_client.h"
2.  #include "goose_receiver.h"
3.  #include "goose_subscriber.h"
4.  #include "iec61850_server.h"
5.  #include "hal_thread.h" /* for Thread_sleep() */
6.  #include <signal.h>
7.  #include <stdlib.h>
8.  #include <stdio.h>
9.  #include <string.h>
10. #include "mms_value.h"
11. #include "goose_publisher.h"
12. #include "sv_subscriber.h"
13. #include <math.h>
14. #include "static_model.h"
15. #include <sys/timeb.h>
16. #include <sys/time.h>
17. #include <complex.h>
18. #define MICRO_PER_SECOND 1000000
19. #define pi 3.14
20.
21. /* import IEC 61850 device model created from SCL-File */
22. extern IedModel iedModel;
23.
24. bool contador = true;
25. bool contador10 = true;
26. bool contador9 = true;

```

```
27.  bool contador11 = true;
28.  bool contador7 = true;
29.  bool contador8 = true;
30.  bool contador1 = true;
31.  bool contador2 = true;
32.  bool contador3 = true;
33.  bool contador4 = true;
34.  bool contador5 = true;
35.  bool contador6 = true;
36.  bool contador12 = true;
37.  bool contador13 = true;
38.  bool resposta = true;
39.  bool comando = false;
40.  int curva = 0;
41.  int funcao = 0;
42.  char* svID2;
43.  static int running = 0;
44.  static IedServer iedServer = NULL;
45.  int contadorSV1 = 0;
46.  int contadorSV2 = 0;
47.  int contadorSV3 = 0;
48.  float Vn = 0, atm = 0, j = 0;
49.  float SVrms_deltaA = 0;
50.  float SVrms_deltaB = 0;
51.  float SVrms_deltaC = 0;
52.  float SVrms_deltaN = 0;
53.  float SVrms_deltaA1 = 0;
54.  float SVrms_deltaB1 = 0;
55.  float SVrms_deltaC1 = 0;
56.  float SVrms_deltaN1 = 0;
57.  float max_corrente_a = 0;
58.  float max_corrente_b = 0;
59.  float max_corrente_c = 0;
60.  float max_corrente_n = 0;
```

```

61. float max_tensao_a = 0;
62. float max_tensao_b = 0;
63. float max_tensao_c = 0;
64. float max_tensao_n = 0;
65. static float corrente_primarioA = 0;
66. static float corrente_primarioB = 0;
67. static float corrente_primarioC = 0;
68. static float corrente_primarioN = 0;
69. static float tensao_primarioA = 0;
70. static float tensao_primarioB = 0;
71. static float tensao_primarioC = 0;
72. static float tensao_primarioN = 0;
73. float teste[80];
74. static float pick_up;
75. static float M, M1, M2, K, a, t, t1, t2, T, B = 1;
76. struct timeval start_time, start1_time, start2_time;
77. struct timeval stop_time, stop1_time, stop2_time;
78. struct timeval tart_time, tart1_time, tart2_time;
79. struct timeval top_time, top1_time, top2_time;
80. static float ang1, ang2, ang3, ang4, ang5, ang6, ang7, ang8;
81. static float time_diff = 0, time_diff1 = 0, time_diff2 = 0;
82. static float ime_diff = 0, ime_diff1 = 0, ime_diff2 = 0;
83. double an[8], ar[6], br[6], teta, teta1, teta2, torque, torque1, torque2, tetaN, torqueN;
84. double complex yr[6], Vbc, aVbc, mVbc, Vca, aVca, mVca, Vab, aVab, mVab;
85.
86. void sigint_handler(int signalId)
87. {
88.     running = 0;
89. }
90.
91. /* Callback handler for received SV messages */
92. static void
93. svUpdateListener (SVSubscriber subscriber, void* parameter, SVSubscriber_ASDU
asdu)

```

```
94.  {
95.      int i;
96.      const char* svID = SVSubscriber_ASDU_getSvId(asdu);
97.      if(resposta == true){
98.          system ("clear");
99.          printf("Defina o Ângulo de Torque Máximo:");
100.         scanf("%f", &atm);
101.         printf("Defina a corrente de pick_up de Neutro:\n");
102.         scanf("%f", &pick_up);
103.         printf("Defina a Tensão Nominal:\n");
104.         scanf("%f", &Vn);
105.         printf("Escolha o dial:\n");
106.         scanf("%f", &T);
107.         printf("Escolha uma das curvas abaixo:\n
108.         - Curva Inversa [1]\n
109.         - Moderadamente Inversa [2]\n
110.         - Muito Inversa [3]\n
111.         - Extremamente Inversa [4]\n");
112.         scanf("%d", &curva);
113.         if (curva == 1){
114.             K = 0.14;
115.             a = 0.02;
116.         }
117.         if (curva == 2){
118.             K = 0.05;
119.             a = 0.04;
120.         }
121.         if (curva == 3){
122.             K = 13.5;
123.             a = 1;
124.         }
125.         if (curva == 4){
126.             K = 80;
127.             a = 2;
```

```

128.      }
129.      resposta = false;
130.      IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind10_stVal, false);
131.      IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind11_stVal, false);
132.  }
133.
134.  if ((strcmp(svID,"IEDNameMU0101"))== 0){
135.
136.      SVrms_deltaA = (SVrms_deltaA + pow((SVSubscriber_ASDU_getINT32 (asdu,
0)*0.001),2));
137.      SVrms_deltaA1 = (SVrms_deltaA1 + pow((SVSubscriber_ASDU_getINT32
(asdu, 32)*0.01),2));
138.      SVrms_deltaB = (SVrms_deltaB + pow((SVSubscriber_ASDU_getINT32 (asdu,
8)*0.001),2));
139.      SVrms_deltaB1 = (SVrms_deltaB1 + pow((SVSubscriber_ASDU_getINT32
(asdu, 40)*0.01),2));
140.      SVrms_deltaC = (SVrms_deltaC + pow((SVSubscriber_ASDU_getINT32 (asdu,
16)*0.001),2));
141.      SVrms_deltaC1 = (SVrms_deltaC1 + pow((SVSubscriber_ASDU_getINT32
(asdu, 48)*0.01),2));
142.      SVrms_deltaN = (SVrms_deltaN + pow((SVSubscriber_ASDU_getINT32 (asdu,
24)*0.001),2));
143.      SVrms_deltaN1 = (SVrms_deltaN1 + pow((SVSubscriber_ASDU_getINT32
(asdu, 56)*0.01),2));
144.
145.
146.      if((SVSubscriber_ASDU_getINT32 (asdu, 32)*0.01)>=0){
147.          teste[contadorSV1] = (SVSubscriber_ASDU_getINT32 (asdu, 32)*0.01);
148.      }
149.
150.      if ((contador1 == true)&&((SVSubscriber_ASDU_getINT32 (asdu,
32)*0.01)>=0)){

```

```

151.         max_tensao_a = (SVSubscriber_ASDU_getINT32 (asdu, 32)*0.01);
152.         contador1 = false;
153.     }
154.     if((contador2 == true)&&((SVSubscriber_ASDU_getINT32 (asdu,
40)*0.01)>=0)){
155.         max_tensao_b = (SVSubscriber_ASDU_getINT32 (asdu, 40)*0.01);
156.         contador2 = false;
157.     }
158.
159.     if((contador3 == true)&&((SVSubscriber_ASDU_getINT32 (asdu,
48)*0.01)>=0)){
160.         max_tensao_c = (SVSubscriber_ASDU_getINT32 (asdu, 48)*0.01);
161.         contador3 = false;
162.     }
163.     if ((contador4 == true)&&((SVSubscriber_ASDU_getINT32 (asdu,
0)*0.01)>=0)){
164.         max_corrente_a = (SVSubscriber_ASDU_getINT32 (asdu, 0)*0.01);
165.         contador4 = false;
166.     }
167.     if((contador5 == true)&&((SVSubscriber_ASDU_getINT32 (asdu,
8)*0.01)>=0)){
168.         max_corrente_b = (SVSubscriber_ASDU_getINT32 (asdu, 8)*0.01);
169.         contador5 = false;
170.     }
171.
172.     if((contador6 == true)&&((SVSubscriber_ASDU_getINT32 (asdu,
16)*0.01)>=0)){
173.         max_corrente_c = (SVSubscriber_ASDU_getINT32 (asdu, 16)*0.01);
174.         contador6 = false;
175.     }
176.
177.     if((contador12 == true)&&((SVSubscriber_ASDU_getINT32 (asdu,
56)*0.01)>=0)){
178.         max_tensao_n = (SVSubscriber_ASDU_getINT32 (asdu, 56)*0.01);

```

```

179.         contador12 = false;
180.     }
181.
182.     if(((contador13 == true)&&((SVSubscriber_ASDU_getINT32 (asdu,
24)*0.001)>=0))){
183.         max_corrente_n = (SVSubscriber_ASDU_getINT32 (asdu, 24)*0.001);
184.         contador13 = false;
185.     }
186.
187.     if(((SVSubscriber_ASDU_getINT32 (asdu, 32)*0.01)>=0)){
188.         if(teste[contadorSV1]>teste[contadorSV1-1]){
189.             if((SVSubscriber_ASDU_getINT32 (asdu, 32)*0.01)>max_tensao_a){
190.                 max_tensao_a = (SVSubscriber_ASDU_getINT32 (asdu, 32)*0.01);
191.                 ang1 = SVSubscriber_ASDU_getSmpCnt(asdu);
192.             }
193.         }
194.     }
195.
196.     if(((SVSubscriber_ASDU_getINT32 (asdu, 40)*0.01)>=0)){
197.         if((SVSubscriber_ASDU_getINT32 (asdu, 40)*0.01)>max_tensao_b){
198.             max_tensao_b = (SVSubscriber_ASDU_getINT32 (asdu, 40)*0.01);
199.             ang2 = SVSubscriber_ASDU_getSmpCnt(asdu);
200.         }
201.     }
202.
203.     if(((SVSubscriber_ASDU_getINT32 (asdu, 48)*0.01)>=0)){
204.         if((SVSubscriber_ASDU_getINT32 (asdu, 48)*0.01)>max_tensao_c){
205.             max_tensao_c = (SVSubscriber_ASDU_getINT32 (asdu, 48)*0.01);
206.             ang3 = SVSubscriber_ASDU_getSmpCnt(asdu);
207.         }
208.     }
209.     if(((SVSubscriber_ASDU_getINT32 (asdu, 0)*0.01)>=0)){
210.         if((SVSubscriber_ASDU_getINT32 (asdu, 0)*0.01)>max_corrente_a){
211.             max_corrente_a = (SVSubscriber_ASDU_getINT32 (asdu, 0)*0.01);

```

```

212.         ang4 = SVSubscriber_ASDU_getSmpCnt(asdu);
213.     }
214. }
215.
216.     if(((SVSubscriber_ASDU_getINT32 (asdu, 8)*0.01)>=0)){
217.         if((SVSubscriber_ASDU_getINT32 (asdu, 8)*0.01)>max_corrente_b){
218.             max_corrente_b = (SVSubscriber_ASDU_getINT32 (asdu, 8)*0.01);
219.             ang5 = SVSubscriber_ASDU_getSmpCnt(asdu);
220.         }
221.     }
222.
223.     if(((SVSubscriber_ASDU_getINT32 (asdu, 16)*0.01)>=0)){
224.         if((SVSubscriber_ASDU_getINT32 (asdu, 16)*0.01)>max_corrente_c){
225.             max_corrente_c = (SVSubscriber_ASDU_getINT32 (asdu, 16)*0.01);
226.             ang6 = SVSubscriber_ASDU_getSmpCnt(asdu);
227.         }
228.     }
229.
230.     if(((SVSubscriber_ASDU_getINT32 (asdu, 56)*0.01)>=0)){
231.         if((SVSubscriber_ASDU_getINT32 (asdu, 56)*0.01)>max_tensao_n){
232.             max_tensao_n = (SVSubscriber_ASDU_getINT32 (asdu, 56)*0.01);
233.             ang8 = SVSubscriber_ASDU_getSmpCnt(asdu);
234.         }
235.     }
236.
237.     if(((SVSubscriber_ASDU_getINT32 (asdu, 24)*0.001)>=0)){
238.         if((SVSubscriber_ASDU_getINT32 (asdu, 24)*0.001)>max_corrente_n){
239.             max_corrente_n = (SVSubscriber_ASDU_getINT32 (asdu, 24)*0.001);
240.             ang7 = SVSubscriber_ASDU_getSmpCnt(asdu);
241.         }
242.     }
243.     contadorSV1 += 1;
244. }
245.

```

```

246.     if (contadorSV1==80)
247.     {
248.         corrente_primarioA = sqrt(SVrms_deltaA / 80);
249.         corrente_primarioB = sqrt(SVrms_deltaB / 80);
250.         corrente_primarioC = sqrt(SVrms_deltaC / 80);
251.         tensao_primarioA = sqrt(SVrms_deltaA1 / 80);
252.         tensao_primarioB = sqrt(SVrms_deltaB1 / 80);
253.         tensao_primarioC = sqrt(SVrms_deltaC1 / 80);
254.         corrente_primarioN = sqrt(SVrms_deltaN / 80);
255.         tensao_primarioN = sqrt(SVrms_deltaN1 / 80);
256.
257.         an[0] = 0;
258.         an[1] = (ang1 - ang2)*4.5;
259.         an[2] = (ang1 - ang3)*4.5;
260.         an[3] = (ang1 - ang4)*4.5;
261.         an[4] = (ang1 - ang5)*4.5;
262.         an[5] = (ang1 - ang6)*4.5;
263.         an[6] = (ang1 - ang7)*4.5;
264.         an[7] = (ang1 - ang8)*4.5;
265.         if(an[1]<0){
266.             an[1] = 360 + an[1];
267.         }if(an[2]<0){
268.             an[2] = 360 + an[2];
269.         }if(an[3]<0){
270.             an[3] = 360 + an[3];
271.         }if(an[4]<0){
272.             an[4] = 360 + an[4];
273.         }if(an[5]<0){
274.             an[5] = 360 + an[5];
275.         }if(an[6]<0){
276.             an[6] = 360 + an[6];
277.         }if(an[7]<0){
278.             an[7] = 360 + an[7];
279.         }

```

```

280.
281.     if(((tensao_primarioN>(vn/10))&&(corrente_primarioN>(pick_up/10))) {
282.         tetaN = an[7] - (an[6]);
283.         torqueN = max_tensao_n*max_corrente_n*cos((atm - tetaN)*pi/180);
284.     }
285.
286.     contadorSV3 ++;
287.
288.     if(torqueN > 0){
289.         M = corrente_primarioN/(pick_up);
290.         t=(T*(K/((pow(M,a))-B)));
291.         if ((t>0)&&(corrente_primarioN>(pick_up*50/100))) {
292.             if (contador == true){
293.                 gettimeofday( &start_time, NULL );
294.                 contador = false;
295.             }
296.             gettimeofday( &stop_time, NULL );
297.             time_diff = (float)(stop_time.tv_sec - start_time.tv_sec);
298.             time_diff += (stop_time.tv_usec -
start_time.tv_usec)/(float)MICRO_PER_SECOND;
299.             if ((time_diff)>=t){
300.                 printf("-----
-----\n");
301.                 printf("          ATUAR FUNÇÃO 67N: DIRECIONAL DE
SOBRECORRENTE TEMPORIZADA DE NEUTRO          \n");
302.                 printf("-----
-----\n");
303.                 IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind12_stVal, true);
304.                 IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, true);
305.                 if (contador9 == true){
306.                     gettimeofday( &tart_time, NULL );
307.                     contador9 = false;

```

```

308.          }
309.          gettimeofday( &top_time, NULL );
310.          ime_diff = (float)(top_time.tv_sec - tart_time.tv_sec);
311.          ime_diff += (top_time.tv_usec -
tart_time.tv_usec)/(float)MICRO_PER_SECOND;
312.          if ((ime_diff)>=0.250){
313.              IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, true);
314.          }
315.      }
316.  }
317.  }
318.  if ((corrente_primarioN<(pick_up*10/100)){
319.      IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, false);
320.      IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind12_stVal, false);
321.      IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, true);
322.      contador9 = true;
323.      contador = true;
324.  }
325.  if(contadorSV3 == 60){
326.      IedServer_updateFloatAttributeValue(iedServer, IEDMO-
DEL_MET_METMMXU1_A_phsA_cVal_mag_f, corrente_primarioA);
327.      IedServer_updateFloatAttributeValue(iedServer, IEDMO-
DEL_MET_METMMXU1_A_phsB_cVal_mag_f, corrente_primarioB);
328.      IedServer_updateFloatAttributeValue(iedServer, IEDMO-
DEL_MET_METMMXU1_A_phsC_cVal_mag_f, corrente_primarioC);
329.      IedServer_updateFloatAttributeValue(iedServer, IEDMO-
DEL_MET_METMMXU1_PPV_phsAB_cVal_mag_f, tensao_primarioA);
330.      IedServer_updateFloatAttributeValue(iedServer, IEDMO-
DEL_MET_METMMXU1_PPV_phsBC_cVal_mag_f, tensao_primarioB);

```

```

331.         IedServer_updateFloatAttributeValue(iedServer, IEDMO-
DEL_MET_METMMXU1_PPV_phsCA_cVal_mag_f, tensao_primarioC);
332.         contadorSV3 = 0;
333.     }
334.
335.     contadorSV1=0;
336.     SVrms_deltaA=0;
337.     SVrms_deltaB=0;
338.     SVrms_deltaC=0;
339.     SVrms_deltaA1=0;
340.     SVrms_deltaB1=0;
341.     SVrms_deltaC1=0;
342.     contador1 = true;
343.     contador2 = true;
344.     contador3 = true;
345.     contador4 = true;
346.     contador5 = true;
347.     contador6 = true;
348.     contador12 = true;
349.     contador13 = true;
350. }
351. }
352.
353. void
354. controlHandlerForBinaryOutput(ControlAction action, void* parameter, MmsValue*
value, bool test)
355. {
356.
357.     if (MmsValue_getType(value) == MMS_BOOLEAN) {
358.         printf("received binary control command: ");
359.
360.         if (MmsValue_getBoolean(value)){
361.             printf("on\n");

```

```

362.         IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind10_stVal, true);
363.         IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind11_stVal, false);
364.     }
365.     else{
366.         printf("off\n");
367.         IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind11_stVal, true);
368.         IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind10_stVal, false);
369.     }
370.
371. }
372.     uint64_t timestamp = Hal_getTimeInMs();
373.
374.     if (parameter == IEDMODEL_CON_RBGGIO1_SPCSO01) {
375.         IedServer_updateUTCTimeAttributeValue(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO01_t, timestamp);
376.         IedServer_updateAttributeValue(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO01_stVal, value);
377.     }
378.
379.     if (parameter == IEDMODEL_CON_RBGGIO1_SPCSO02) {
380.         IedServer_updateUTCTimeAttributeValue(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO02_t, timestamp);
381.         IedServer_updateAttributeValue(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO02_stVal, value);
382.     }
383.
384.     if (parameter == IEDMODEL_CON_RBGGIO1_SPCSO03) {
385.         IedServer_updateUTCTimeAttributeValue(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO03_t, timestamp);

```

```

386.         IedServer_updateAttributeValue(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO03_stVal, value);
387.     }
388.
389.     if (parameter == IEDMODEL_CON_RBGGIO1_SPCSO04) {
390.         IedServer_updateUTCTimeAttributeValue(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO04_t, timestamp);
391.         IedServer_updateAttributeValue(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO04_stVal, value);
392.     }
393.
394.     if (parameter == IEDMODEL_CON_RBGGIO1_SPCSO05) {
395.         IedServer_updateUTCTimeAttributeValue(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO05_t, timestamp);
396.         IedServer_updateAttributeValue(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO05_stVal, value);
397.     }
398.
399.     if (parameter == IEDMODEL_CON_RBGGIO1_SPCSO06) {
400.         IedServer_updateUTCTimeAttributeValue(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO06_t, timestamp);
401.         IedServer_updateAttributeValue(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO06_stVal, value);
402.     }
403.
404.     if (parameter == IEDMODEL_CON_RBGGIO1_SPCSO07) {
405.         IedServer_updateUTCTimeAttributeValue(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO07_t, timestamp);
406.         IedServer_updateAttributeValue(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO07_stVal, value);
407.     }
408. }
409.
410. static void

```

```

411. goCbEventHandler(MmsGooseControlBlock goCb, int event, void* parameter)
412. {
413.     printf("Access to GoCB: %s\n", MmsGooseControlBlock_getName(goCb));
414.     printf("    GoEna: %i\n", MmsGooseControlBlock_getGoEna(goCb));
415. }
416.
417. //Função Listener
418.
419. static void
420. gooseListener(GooseSubscriber subscriber, void* parameter)
421. {
422.     MmsValue* values = GooseSubscriber_getDataSetValues(subscriber);
423.
424.     char buffer[50];
425.
426.     MmsValue_printToBuffer(values, buffer, 50);
427.
428.     char b; char c; char d;
429.
430.     b = buffer[1];
431.     c = buffer[6];
432.     d = buffer[11];
433.
434.     if(b == 116){
435.         IedServer_updateDbposValue(iedServer, IEDMO-
DEL_PRO_BK1XCBR1_Pos_stVal, DBPOS_OFF);
436.         IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind11_stVal, false);
437.     }
438.
439.     else{
440.         IedServer_updateDbposValue(iedServer, IEDMO-
DEL_PRO_BK1XCBR1_Pos_stVal, DBPOS_ON);

```

```

441.     IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind10_stVal, false);
442. }
443. }
444.
445. int
446. main(int argc, char** argv)
447. {
448.     IedServerConfig config = IedServerConfig_create();
449.
450.     iedServer = IedServer_createWithConfig(&iedModel, NULL, config);
451.
452.     IedServerConfig_destroy(config);
453.
454.     SVReceiver receiverSV = SVReceiver_create();
455.
456.     if (argc > 1) {
457.         char* ethernetIfcID = argv[1];
458.
459.         printf("Using GOOSE interface: %s\n", ethernetIfcID);
460.
461.         /* set GOOSE interface for all GOOSE publishers (GCBs) */
462.         IedServer_setGooseInterfaceId(iedServer, ethernetIfcID);
463.         SVReceiver_setInterfaceId(receiverSV, ethernetIfcID);
464.     }
465.
466.     if (argc > 2) {
467.         char* ethernetIfcID = argv[2];
468.
469.         printf("Using GOOSE interface for GenericIO/LLN0.gcbAnalogValues: %s\n",
ethernetIfcID);
470.
471.         /* set GOOSE interface for a particular GOOSE publisher (GCB) */

```

```

472.         IedServer_setGooseInterfaceIdEx(iedServer, IEDMODEL_CFG_LLNO,
"BRep0201", ethernetIfcID);
473.     }
474.
475.     /*Preparando o código para receber mensagens SV*/
476.
477.     SVSubscriber subscriberSV = SVSubscriber_create(NULL, 0x4000);
478.     SVSubscriber_setListener(subscriberSV, svUpdateListener, NULL);
479.     SVReceiver_addSubscriber(receiverSV, subscriberSV);
480.
481.     IedServer_enableGoosePublishing(iedServer);
482.
483.     GooseReceiver receiver = GooseReceiver_create();
484.
485.     GooseReceiver_setInterfaceId(receiver, "p5p1");
486.
487.     GooseSubscriber subscriber = GooseSubscri-
ber_create("MU320CTRL/LLNO$GO$FastGOOSE1", NULL); //Especificação de quem o ied
irá receber as mensagens goose
488.
489.     GooseSubscriber_setListener(subscriber, gooseListener, iedServer);
490.
491.     GooseReceiver_addSubscriber(receiver, subscriber);
492.
493.     GooseReceiver_start(receiver);
494.     SVReceiver_start(receiverSV);
495.
496.     //Thread_sleep(1000);
497.
498.     IedServer_setGoCBHandler(iedServer, goCbEventHandler, NULL);
499.
500.     /* MMS server will be instructed to start listening to client connections. */
501.     IedServer_start(iedServer, 102);
502.

```

```

503.     IedServer_setControlHandler(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO01, (ControlHandler) controlHandlerForBinaryOutput,
504.     IEDMODEL_CON_RBGGIO1_SPCSO01);
505.
506.     IedServer_setControlHandler(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO02, (ControlHandler) controlHandlerForBinaryOutput,
507.     IEDMODEL_CON_RBGGIO1_SPCSO02);
508.
509.     IedServer_setControlHandler(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO03, (ControlHandler) controlHandlerForBinaryOutput,
510.     IEDMODEL_CON_RBGGIO1_SPCSO03);
511.
512.     IedServer_setControlHandler(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO04, (ControlHandler) controlHandlerForBinaryOutput,
513.     IEDMODEL_CON_RBGGIO1_SPCSO04);
514.
515.     IedServer_setControlHandler(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO05, (ControlHandler) controlHandlerForBinaryOutput,
516.     IEDMODEL_CON_RBGGIO1_SPCSO05);
517.
518.     IedServer_setControlHandler(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO06, (ControlHandler) controlHandlerForBinaryOutput,
519.     IEDMODEL_CON_RBGGIO1_SPCSO06);
520.
521.     IedServer_setControlHandler(iedServer, IEDMO-
DEL_CON_RBGGIO1_SPCSO07, (ControlHandler) controlHandlerForBinaryOutput,
522.     IEDMODEL_CON_RBGGIO1_SPCSO07);
523.
524.     IedServer_setControlHandler(iedServer, IEDMODEL_PRO_BKR1CSWI1_Pos,
(ControlHandler) controlHandlerForBinaryOutput, IEDMODEL_PRO_BKR1CSWI1_Pos);
525.
526.     /*GooseReceiver_start(receiver);
527.     SVReceiver_start(receiverSV);*/
528.

```

```

529.     if (!IedServer_isRunning(iedServer)) {
530.         printf("Starting server failed! Exit.\n");
531.         IedServer_destroy(iedServer);
532.         exit(-1);
533.     }
534.
535.     running = 1;
536.
537.     signal(SIGINT, sigint_handler);
538.
539.     while (running) {
540.
541.         Thread_sleep(1000);
542.     }
543.
544.     /* stop MMS server - close TCP server socket and all client sockets */
545.     IedServer_stop(iedServer);
546.
547.     /* Cleanup - free all resources */
548.     SVReceiver_destroy(receiverSV);
549.     IedServer_destroy(iedServer);
550.
551.     return 0;

```

Para a função de proteção 67 e 50/62BF, é preciso substituir as linhas 98 a 128 pela lógica abaixo:

```

1.         system ("clear");
2.         printf("Defina o Ângulo de Torque Máximo:");
3.         scanf("%f", &atm);
4.         printf("Escolha o dial:\n");
5.         scanf("%f", &T);
6.         printf("Defina a corrente de pick-up:\n");
7.         scanf("%f", &pick_up);

```

```

8.      printf("Escolha uma das curvas abaixo:
9.      \n- Curva Inversa [1]
10.     \n- Moderadamente Inversa [2]
11.     \n- Muito Inversa [3]
12.     \n- Extremamente Inversa [4]\n");
13.     scanf("%d", &curva);
14.     if (curva == 1){
15.         K = 0.14;
16.         a = 0.02;
17.     }
18.     if (curva == 2){
19.         K = 0.05;
20.         a = 0.04;
21.     }
22.     if (curva == 3){
23.         K = 13.5;
24.         a = 1;
25.     }
26.     if (curva == 4){
27.         K = 80;
28.         a = 2;
29.     }

```

Por fim, é preciso substituir as linhas 281 a 322 pela lógica abaixo:

```

1.      ar[0] = max_tensao_a * cos(an[0]);
2.      br[0] = max_tensao_a * sin(an[0]);
3.      ar[1] = max_tensao_b * cos(an[1]);
4.      br[1] = max_tensao_b * sin(an[1]);
5.      ar[2] = max_tensao_c * cos(an[2]);
6.      br[2] = max_tensao_c * sin(an[2]);
7.      ar[3] = max_corrente_a * cos(an[3]);
8.      br[3] = max_corrente_a * sin(an[3]);
9.      ar[4] = max_corrente_b * cos(an[4]);

```

```

10.     br[4] = max_corrente_b * sin(an[4]);
11.     ar[5] = max_corrente_c * cos(an[5]);
12.     br[5] = max_corrente_c * sin(an[5]);
13.     yr[0] = ar[0] + br[0]*I;
14.     yr[1] = ar[1] + br[1]*I;
15.     yr[2] = ar[2] + br[2]*I;
16.     yr[3] = ar[3] + br[3]*I;
17.     yr[4] = ar[4] + br[4]*I;
18.     yr[5] = ar[5] + br[5]*I;
19.
20.     Vbc = yr[1] - yr[2];
21.     Vca = yr[2] - yr[0];
22.     Vab = yr[0] - yr[1];
23.     aVbc = atan(cimag(Vbc)/creal(Vbc));
24.     aVca = atan(cimag(Vca)/creal(Vca));
25.     aVab = atan(cimag(Vab)/creal(Vab));
26.     mVbc = sqrt(pow(creal(Vbc),2)+pow(cimag(Vbc),2));
27.     mVca = sqrt(pow(creal(Vca),2)+pow(cimag(Vca),2));
28.     mVab = sqrt(pow(creal(Vab),2)+pow(cimag(Vab),2));
29.     teta = an[3] - aVbc;
30.     teta1 = an[4] - aVca;
31.     teta2 = an[5] - aVab;
32.     torque = max_corrente_a*mVbc*cos((atm - teta)*pi/180);
33.     torque1 = max_corrente_b*mVca*cos((atm - teta1)*pi/180);
34.     torque2 = max_corrente_c*mVab*cos((atm - teta2)*pi/180);
35.     contadorSV3 ++;
36.     if (torque > 0){
37.         M = corriente_primarioA/(pick_up);
38.         t=(T*(K/((pow(M,a))-B)));
39.         if ((t>0)&&(corrente_primarioA>(pick_up*50/100))){
40.             if (contador == true){
41.                 gettimeofday( &start_time, NULL );
42.                 contador = false;
43.             }

```

```

44.          gettimeofday( &stop_time, NULL );
45.          time_diff = (float)(stop_time.tv_sec - start_time.tv_sec);
46.          time_diff += (stop_time.tv_usec -
start_time.tv_usec)/(float)MICRO_PER_SECOND;
47.          if ((time_diff)>=t){
48.              printf("-----
-----\n");
49.              printf("          ATUAR FUNÇÃO 67: SOBRECORRENTE
DIRECIONAL TEMPORIZADA FASE A          \n");
50.              printf("-----
-----\n");
51.              IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind07_stVal, true);
52.              IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, true);
53.              if (contador9 == true){
54.                  gettimeofday( &tart_time, NULL );
55.                  contador9 = false;
56.              }
57.              gettimeofday( &top_time, NULL );
58.              ime_diff = (float)(top_time.tv_sec - tart_time.tv_sec);
59.              ime_diff += (top_time.tv_usec -
tart_time.tv_usec)/(float)MICRO_PER_SECOND;
60.              if ((ime_diff)>=0.250){
61.                  IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, true);
62.              }
63.          }
64.      }
65.  }
66.
67.  if (torque1 > 0){
68.      M1 = corrente_primarioB/(pick_up);
69.      t1=(T*(K/((pow(M1,a))-B)));

```

```

70.         if ((t1>0)&&(corrente_primarioB>(pick_up*50/100))){
71.             if (contador7 == true){
72.                 gettimeofday( &start1_time, NULL );
73.                 contador7 = false;
74.             }
75.             gettimeofday( &stop1_time, NULL );
76.             time_diff1 = (float)(stop1_time.tv_sec - start1_time.tv_sec);
77.             time_diff1 += (stop1_time.tv_usec -
start1_time.tv_usec)/(float)MICRO_PER_SECOND;
78.             if ((time_diff1)>=t1){
79.                 printf("-----\n");
-----\n");
80.                 printf("          ATUAR FUNÇÃO 67: SOBRECORRENTE
DIRECIONAL TEMPORIZADA FASE B          \n");
81.                 printf("-----\n");
-----\n");
82.                 IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind08_stVal, true);
83.                 IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, true);
84.                 if (contador10 == true){
85.                     gettimeofday( &tart1_time, NULL );
86.                     contador10 = false;
87.                 }
88.                 gettimeofday( &top1_time, NULL );
89.                 ime_diff1 = (float)(top1_time.tv_sec - tart1_time.tv_sec);
90.                 ime_diff1 += (top1_time.tv_usec -
tart1_time.tv_usec)/(float)MICRO_PER_SECOND;
91.                 if ((ime_diff1)>=0.250){
92.                     IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, true);
93.                 }
94.             }
95.         }

```

```

96.      }
97.
98.      if (torque2 > 0){
99.          M2 = corrente_primarioC/(pick_up);
100.         t2=(T*(K/((pow(M2,a))-B)));
101.         if ((t2>0)&&(corrente_primarioC>(pick_up*50/100))){
102.             if (contador8 == true){
103.                 gettimeofday( &start2_time, NULL );
104.                 contador8 = false;
105.             }
106.             gettimeofday( &stop2_time, NULL );
107.             time_diff2 = (float)(stop2_time.tv_sec - start2_time.tv_sec);
108.             time_diff2 += (stop2_time.tv_usec -
start2_time.tv_usec)/(float)MICRO_PER_SECOND;
109.             if ((time_diff2)>=t2){
110.                 printf("-----\n");
-----\n");
111.                 printf("          ATUAR FUNÇÃO 67: SOBRECORRENTE
DIRECIONAL TEMPORIZADA FASE C          \n");
112.                 printf("-----\n");
-----\n");
113.                 IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind09_stVal, true);
114.                 IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, true);
115.                 if (contador11 == true){
116.                     gettimeofday( &tart2_time, NULL );
117.                     contador11 = false;
118.                 }
119.                 gettimeofday( &top2_time, NULL );
120.                 ime_diff2 = (float)(top2_time.tv_sec - tart2_time.tv_sec);
121.                 ime_diff2 += (top2_time.tv_usec -
tart2_time.tv_usec)/(float)MICRO_PER_SECOND;
122.                 if ((ime_diff2)>=0.250){

```

```

123.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBFR1_OpEx_general, true);
124.          }
125.      }
126.  }
127.  }
128.
129.      if ((corren-
te_primarioA<(pick_up*10/100))&&(corrente_primarioB<(pick_up*10/100))&&(corrente_pr
imarioC<(pick_up*10/100))) {
130.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, false);
131.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind07_stVal, false);
132.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind08_stVal, false);
133.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind09_stVal, false);
134.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBFR1_OpEx_general, false);
135.          contador = true;
136.          contador7 = true;
137.          contador8 = true;
138.          contador9 = true;
139.          contador10 = true;
140.          contador11 = true;
141.      }

```

Para a função de proteção 51N e 50/62BF, é preciso substituir as linhas 98 a 128 pela lógica abaixo:

```

1.      system ("clear");
2.      printf("Defina a corrente de pick_up de Neutro:\n");
3.      scanf("%f", &pick_up);

```

```

4.     printf("Defina a Tensão Nominal:\n");
5.     scanf("%f", &Vn);
6.     printf("Escolha o dial:\n");
7.     scanf("%f", &T);
8.     printf("Escolha uma das curvas abaixo:\n
9.     - Curva Inversa [1]\n
10.    - Moderadamente Inversa [2]\n
11.    - Muito Inversa [3]\n
12.    - Extremamente Inversa [4]\n");
13.    scanf("%d", &curva);
14.    if (curva == 1){
15.        K = 0.14;
16.        a = 0.02;
17.    }
18.    if (curva == 2){
19.        K = 0.05;
20.        a = 0.04;
21.    }
22.    if (curva == 3){
23.        K = 13.5;
24.        a = 1;
25.    }
26.    if (curva == 4){
27.        K = 80;
28.        a = 2;
29.    }

```

Por fim, é preciso substituir as linhas 281 a 322 pela lógica abaixo:

```

1.     contadorSV3++;
2.     M = corrente_primarioN/(pick_up);
3.     t=(T*(K/((pow(M,a))-B)));
4.
5.     if ((t>0)&&(corrente_primarioN>(pick_up*50/100))){

```

```

6.         if (contador == true){
7.             gettimeofday( &start_time, NULL );
8.             contador = false;
9.         }
10.        gettimeofday( &stop_time, NULL );
11.        time_diff = (float)(stop_time.tv_sec - start_time.tv_sec);
12.        time_diff += (stop_time.tv_usec -
start_time.tv_usec)/(float)MICRO_PER_SECOND;
13.        if ((time_diff)>=t){
14.            printf("-----
-----\n");
15.            printf("          ATUAR FUNÇÃO 51N: SOBRECORRENTE TEM-
PORIZADA DE NEUTRO          \n");
16.            printf("-----
-----\n");
17.            IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind13_stVal, true);
18.            IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, true);
19.            if (contador9 == true){
20.                gettimeofday( &tart_time, NULL );
21.                contador9 = false;
22.            }
23.            gettimeofday( &top_time, NULL );
24.            ime_diff = (float)(top_time.tv_sec - tart_time.tv_sec);
25.            ime_diff += (top_time.tv_usec -
tart_time.tv_usec)/(float)MICRO_PER_SECOND;
26.            if ((ime_diff)>=0.250){
27.                IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, true);
28.            }
29.        }
30.    }
31.

```

```

32.         if ((corrente_primarioN<(pick_up*10/100)){
33.             IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, false);
34.             IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind13_stVal, false);
35.             IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, false);
36.             contador = true;
37.             contador9 = true;
38.         }

```

Para a função de proteção 51-51V e 50/62BF, é preciso substituir as linhas 98 a 128 pela lógica abaixo:

```

1.         system ("clear");
2.         printf("Escolha uma das funções abaixo:
3.         \n- Função 51 [1]
4.         \n- Função 51V [2]\n");
5.         scanf("%d", &funcao);
6.         if (funcao == 2){
7.             printf("Defina a Tensão Nominal:\n");
8.             scanf("%f", &Vn);
9.         }
10.        printf("Escolha o dial:\n");
11.        scanf("%f", &T);
12.        printf("Defina a corrente de pick-up:\n");
13.        scanf("%f", &pick_up);
14.        printf("Escolha uma das curvas abaixo:
15.        \n- Curva Inversa [1]
16.        \n- Moderadamente Inversa [2]
17.        \n- Muito Inversa [3]
18.        \n- Extremamente Inversa [4]\n");
19.        scanf("%d", &curva);
20.        system ("clear");

```

```

21.     if (curva == 1){
22.         K = 0.14;
23.         a = 0.02;
24.     }
25.     if (curva == 2){
26.         K = 0.05;
27.         a = 0.04;
28.     }
29.     if (curva == 3){
30.         K = 13.5;
31.         a = 1;
32.     }
33.     if (curva == 4){
34.         K = 80;
35.         a = 2;
36.     }

```

Por fim, é preciso substituir as linhas 281 a 322 pela lógica abaixo:

```

1.     if(funcao == 1){
2.         j = 1;
3.     }else{
4.         j = tensao_primarioA/Vn;
5.     }
6.
7.     M = corrente_primarioA/(j*pick_up);
8.     t=(T*(K/((pow(M,a))-B)));
9.
10.    M1 = corrente_primarioB/(j*pick_up);
11.    t1=(T*(K/((pow(M1,a))-B)));
12.
13.    M2 = corrente_primarioC/(j*pick_up);
14.    t2=(T*(K/((pow(M2,a))-B)));
15.

```

```

16.         contadorSV3 ++;
17.
18.         if ((t>0)&&(corrente_primarioA>(pick_up*50/100))){
19.             if (contador == true){
20.                 gettimeofday( &start_time, NULL );
21.                 contador = false;
22.             }
23.             gettimeofday( &stop_time, NULL );
24.             time_diff = (float)(stop_time.tv_sec - start_time.tv_sec);
25.             time_diff += (stop_time.tv_usec -
start_time.tv_usec)/(float)MICRO_PER_SECOND;
26.             if ((time_diff)>=t){
27.
28.                 if(funcao == 1){
29.                     IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_ATPTOC20_Op_general, true);
30.                     printf("-----
-----\n");
31.                     printf("                ATUAR FUNÇÃO 51: SOBRECORRENTE
TEMPORIZADA FASE A                \n");
32.                     printf("-----
-----\n");
33.
34.                 }else{
35.                     IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind01_stVal, true);
36.                     printf("-----
-----\n");
37.                     printf("                ATUAR FUNÇÃO 51V: SOBRECORRENTE
TEMPORIZADA COM RESTRIÇÃO DE TENSÃO FASE A                \n");
38.                     printf("-----
-----\n");
39.                 }

```

```

40.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, true);
41.          if (contador9 == true){
42.              gettimeofday( &tart_time, NULL );
43.              contador9 = false;
44.          }
45.          gettimeofday( &top_time, NULL );
46.          ime_diff = (float)(top_time.tv_sec - tart_time.tv_sec);
47.          ime_diff += (top_time.tv_usec -
tart_time.tv_usec)/(float)MICRO_PER_SECOND;
48.          if ((ime_diff)>=0.250){
49.              IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, true);
50.          }
51.      }
52.  }
53.  if ((t1>0)&&(corrente_primarioB>(pick_up*50/100))){
54.      if (contador7 == true){
55.          gettimeofday( &start1_time, NULL );
56.          contador7 = false;
57.      }
58.      gettimeofday( &stop1_time, NULL );
59.      time_diff1 = (float)(stop1_time.tv_sec - start1_time.tv_sec);
60.      time_diff1 += (stop1_time.tv_usec -
start1_time.tv_usec)/(float)MICRO_PER_SECOND;
61.      if ((time_diff1)>=t1){
62.          if(funcao == 1){
63.              IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BTPTOC21_Op_general, true);
64.              printf("-----
-----\n");
65.              printf("          ATUAR FUNÇÃO 51: SOBRECORRENTE
TEMPORIZADA FASE B          \n");

```

```

66.             printf("-----\n");
67.             }else{
68.                 IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind02_stVal, true);
69.                 printf("-----\n");
70.                 printf("          ATUAR FUNÇÃO 51V: SOBRECORRENTE
TEMPORIZADA COM RESTRIÇÃO DE TENSÃO FASE B          \n");
71.                 printf("-----\n");
72.             }
73.             IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, true);
74.             if (contador10 == true){
75.                 gettimeofday( &tart1_time, NULL );
76.                 contador10 = false;
77.             }
78.             gettimeofday( &top1_time, NULL );
79.             ime_diff1 = (float)(top1_time.tv_sec - tart1_time.tv_sec);
80.             ime_diff1 += (top1_time.tv_usec -
tart1_time.tv_usec)/(float)MICRO_PER_SECOND;
81.             if ((ime_diff1)>=0.250){
82.                 IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, true);
83.             }
84.         }
85.     }
86.     if ((t2>0)&&(corrente_primarioC>(pick_up*50/100))){
87.         if (contador8 == true){
88.             gettimeofday( &start2_time, NULL );
89.             contador8 = false;
90.         }
91.         gettimeofday( &stop2_time, NULL );

```

```

92.         time_diff2 = (float)(stop2_time.tv_sec - start2_time.tv_sec);
93.         time_diff2 += (stop2_time.tv_usec -
start2_time.tv_usec)/(float)MICRO_PER_SECOND;
94.         if ((time_diff2)>=t2){
95.             if(funcao == 1){
96.                 IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_CTPTOC22_Op_general, true);
97.                 printf("-----
-----\n");
98.                 printf("                ATUAR FUNÇÃO 51: SOBRECORRENTE
TEMPORIZADA FASE C                \n");
99.                 printf("-----
-----\n");
100.            }else{
101.                IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind03_stVal, true);
102.                printf("-----
-----\n");
103.                printf("                ATUAR FUNÇÃO 51V: SOBRECORRENTE
TEMPORIZADA COM RESTRIÇÃO DE TENSÃO FASE C                \n");
104.                printf("-----
-----\n");
105.            }
106.            IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, true);
107.            if (contador11 == true){
108.                gettimeofday( &tart2_time, NULL );
109.                contador11 = false;
110.            }
111.            gettimeofday( &top2_time, NULL );
112.            ime_diff2 = (float)(top2_time.tv_sec - tart2_time.tv_sec);
113.            ime_diff2 += (top2_time.tv_usec -
tart2_time.tv_usec)/(float)MICRO_PER_SECOND;
114.            if ((ime_diff2)>=0.250){

```

```

115.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, true);
116.          }
117.      }
118.  }
119.
120.      if ((corren-
te_primarioA<(pick_up*10/100))&&(corrente_primarioB<(pick_up*10/100))&&(corrente_pr
imarioC<(pick_up*10/100))){
121.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, false);
122.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, false);
123.          if(funcao == 1){
124.              IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_ATPTOC20_Op_general, false);
125.              IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BTPTOC21_Op_general, false);
126.              IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_CTPTOC22_Op_general, false);
127.          }else{
128.              IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind01_stVal, false);
129.              IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind02_stVal, false);
130.              IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind03_stVal, false);
131.          }
132.          contador = true;
133.          contador7 = true;
134.          contador8 = true;
135.          contador9 = true;
136.          contador10 = true;
137.          contador11 = true;

```

138. }

Para a função de proteção 50N e 50/62BF, é preciso substituir as linhas 98 a 128 pela lógica abaixo do algoritmo inicial:

```
1.      system ("clear");
2.      printf("Defina a corrente de pick_up de Neutro:\n");
3.      scanf("%f", &pick_up);
4.      printf("Defina a Tensão Nominal:\n");
5.      scanf("%f", &Vn);
```

Por fim, é preciso substituir as linhas 281 a 322 pela lógica abaixo do algoritmo inicial:

```
6.      if ((corrente_primarioN)>=pick_up){
7.          printf("-----\n");
8.          printf("          ATUAR FUNÇÃO 50N: SOBRECORRENTE TEMPO-
RIZADA DE NEUTRO          \n");
9.          printf("-----\n");
10.         IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind14_stVal, true);
11.         IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, true);
12.         if (contador9 == true){
13.             gettimeofday( &tart_time, NULL );
14.             contador9 = false;
15.         }
16.         gettimeofday( &top_time, NULL );
17.         ime_diff = (float)(top_time.tv_sec - tart_time.tv_sec);
18.         ime_diff += (top_time.tv_usec -
tart_time.tv_usec)/(float)MICRO_PER_SECOND;
19.         if ((ime_diff)>=0.250){
```

```

20.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, true);
21.      }
22.  }
23.      if (corrente_primarioN<(pick_up*10/100)){
24.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, false);
25.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind14_stVal, false);
26.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, false);
27.          contador9 = true;
28.      }

```

Para a função de proteção 50 e 50/62BF, é preciso substituir as linhas 98 a 128 pela lógica abaixo do algoritmo inicial:

```

1.      system ("clear");
2.          printf("Defina a corrente de Pick-Up:\n");
3.          scanf("%f", &pick_up);

```

Por fim, é preciso substituir as linhas 281 a 322 pela lógica abaixo do algoritmo inicial:

```

1.          if (corrente_primarioA>pick_up){
2.              printf("-----
-----\n");
3.              printf("          ATUAR FUNÇÃO 50: SOBRECORRENTE INS-
TANTÂNEA FASE A          \n");
4.              printf("-----
-----\n");
5.              IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind04_stVal, true);
6.              IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, true);

```

```

7.          if (contador9 == true){
8.              gettimeofday( &tart_time, NULL );
9.              contador9 = false;
10.         }
11.         gettimeofday( &top_time, NULL );
12.         ime_diff = (float)(top_time.tv_sec - tart_time.tv_sec);
13.         ime_diff += (top_time.tv_usec -
tart_time.tv_usec)/(float)MICRO_PER_SECOND;
14.         if ((ime_diff)>=0.250){
15.             IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, true);
16.         }
17.     }
18.     if (corrente_primarioB>pick_up){
19.         printf("-----
-----\n");
20.         printf("          ATUAR FUNÇÃO 50: SOBRECORRENTE INS-
TANTÂNEA FASE B          \n");
21.         printf("-----
-----\n");
22.         IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind05_stVal, true);
23.         IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, true);
24.         if (contador10 == true){
25.             gettimeofday( &tart1_time, NULL );
26.             contador10 = false;
27.         }
28.         gettimeofday( &top1_time, NULL );
29.         ime_diff1 = (float)(top1_time.tv_sec - tart1_time.tv_sec);
30.         ime_diff1 += (top1_time.tv_usec -
tart1_time.tv_usec)/(float)MICRO_PER_SECOND;
31.         if ((ime_diff1)>=0.250){

```

```

32.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, true);
33.          }
34.      }
35.      if (corrente_primarioC>pick_up){
36.          printf("-----
-----\n");
37.          printf("          ATUAR FUNÇÃO 50: SOBRECORRENTE INS-
TANTÂNEA FASE C          \n");
38.          printf("-----
-----\n");
39.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind06_stVal, true);
40.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, true);
41.          if (contador11 == true){
42.              gettimeofday( &tart2_time, NULL );
43.              contador11 = false;
44.          }
45.          gettimeofday( &top2_time, NULL );
46.          ime_diff2 = (float)(top2_time.tv_sec - tart2_time.tv_sec);
47.          ime_diff2 += (top2_time.tv_usec -
tart2_time.tv_usec)/(float)MICRO_PER_SECOND;
48.          if ((ime_diff2)>=0.250){
49.              IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, true);
50.          }
51.      }
52.      if ((corren-
te_primarioA<(pick_up*10/100))&&(corrente_primarioB<(pick_up*10/100))&&(corrente_pr
imarioC<(pick_up*10/100))){
53.          IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_TRIPPTRC1_Tr_general, false);

```

```

54.         IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_PRO_BFR1RBRF1_OpEx_general, false);
55.         IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind04_stVal, false);
56.         IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind05_stVal, false);
57.         IedServer_updateBooleanAttributeValue(iedServer, IEDMO-
DEL_ANN_SVGGIO3_Ind06_stVal, false);
58.         contador9 = true;
59.         contador10 = true;
60.         contador11 = true;
61.     }

```

Para a MU sintética foi utilizado o seguinte código:

```

62.     #include <signal.h>
63.     #include <stdio.h>
64.     #include "hal_thread.h"
65.     #include "sv_publisher.h"
66.     #include "math.h"
67.
68.     static bool running = true;
69.
70.     void sigint_handler(int signalId)
71.     {
72.         running = 0;
73.     }
74.
75.     int
76.     main(int argc, char** argv)
77.     {
78.         float r_transformacao = 2.8868;
79.         char* interface;
80.
81.         if (argc > 1)
82.             interface = argv[1];

```

```
83.     else
84.         interface = "eth0";
85.
86.     printf("Using interface %s\n", interface);
87.
88.     signal(SIGINT, sigint_handler);
89.
90.     SVPublisher svPublisher = SVPublisher_create(NULL, interface);
91.
92.     if (svPublisher) {
93.
94.         /* Create first ASDU and add data points */
95.
96.         SVPublisher_ASDU asdu1 = SVPublisher_addASDU(svPublisher, "IEDName-
MU0101", NULL, 1);
97.
98.         int float1 = SVPublisher_ASDU_addINT64(asdu1);
99.         int float2 = SVPublisher_ASDU_addINT64(asdu1);
100.        int float3 = SVPublisher_ASDU_addINT64(asdu1);
101.        int float7 = SVPublisher_ASDU_addINT64(asdu1);
102.        int float4 = SVPublisher_ASDU_addINT64(asdu1);
103.        int float5 = SVPublisher_ASDU_addINT64(asdu1);
104.        int float6 = SVPublisher_ASDU_addINT64(asdu1);
105.        int float8 = SVPublisher_ASDU_addINT64(asdu1);
106.
107.        int ts1 = SVPublisher_ASDU_addTimestamp(asdu1);
108.
109.        SVPublisher_setupComplete(svPublisher);
110.
111.        float fVal1 = 0;
112.        float fVal2 = 0;
113.        float fVal3 = 0;
114.        float fVal4 = 0;
115.        float fVal5 = 0;
```

```

116.     float fVal6 = 0;
117.     float fVal7 = 0;
118.     float fVal8 = 0;
119.     float freq = 60.00;
120.     float vpico = 1697.06;
121.     float tpico = 97580.7358;
122.     float tempo;
123.     double x;
124.
125.     while (running) {
126.         Timestamp ts;
127.         Timestamp_clearFlags(&ts);
128.         Timestamp_setTimeInMilliseconds(&ts, Hal_getTimeInMs());
129.
130.         /* update the values in the SV ASDUs */
131.
132.         SVPublisher_ASDU_setINT32(asdu1, float1, fVal1);
133.         SVPublisher_ASDU_setINT32(asdu1, float2, fVal2);
134.         SVPublisher_ASDU_setINT32(asdu1, float3, fVal3);
135.         SVPublisher_ASDU_setINT32(asdu1, float7, fVal7);
136.         SVPublisher_ASDU_setINT32(asdu1, float4, fVal4);
137.         SVPublisher_ASDU_setINT32(asdu1, float5, fVal5);
138.         SVPublisher_ASDU_setINT32(asdu1, float6, fVal6);
139.         SVPublisher_ASDU_setINT32(asdu1, float8, fVal8);
140.
141.         SVPublisher_ASDU_setTimestamp(asdu1, ts1, ts);
142.
143.
144.
145.         /* update the sample counters */
146.
147.         SVPublisher_ASDU_increaseSmpCnt(asdu1);
148.
149.         //fVal1 += 1.1f;

```

```

150.         x = 2*M_PI*freq*tempo;
151.         tempo = (tempo + 0.0002083);
152.
153.         fVal1 = vpico*sin(x)*1000;
154.         fVal2 = vpico*sin(x + 4.18879)*1000;
155.         fVal3 = vpico*sin(x + 2.0944)*1000;
156.         fVal4 = tpico*sin(x)*100;
157.         fVal5 = tpico*sin(x+4.18879)*100;
158.         fVal6 = tpico*sin(x+2.0944)*100;
159.         fVal7 = vpico*sin(x)*1000;
160.         fVal8 = tpico*sin(x)*100;
161.
162.
163.         /* send the SV message */
164.         SVPublisher_publish(svPublisher);
165.         printf("-----CORRENTES DE LINHA LADO DE ALTA-----\n");
166.         printf("Corrente da linha A %.6f\n", fVal1);
167.         printf("Corrente da linha B %.6f\n", fVal2);
168.         printf("Corrente da linha C %.6f\n", fVal3);
169.         printf("-----TENSÕES DE LINHA LADO DE BAIXA-----\n");
170.         printf("Tensão da linha A %.6f\n", fVal4);
171.         printf("Tensão da linha B %.6f\n", fVal5);
172.         printf("Tensão da linha C %.6f\n", fVal6);
173.         /*
174.          * For real applications this sleep time has to be adjusted to match the SV sam-
175.          * Platform specific functions like usleep or timer interrupt service routines
176.          * to realize the required time accuracy for sending messages.
177.          */
178.         Thread_sleep(0.20833);
179.     }
180.
181.     SVPublisher_destroy(svPublisher);

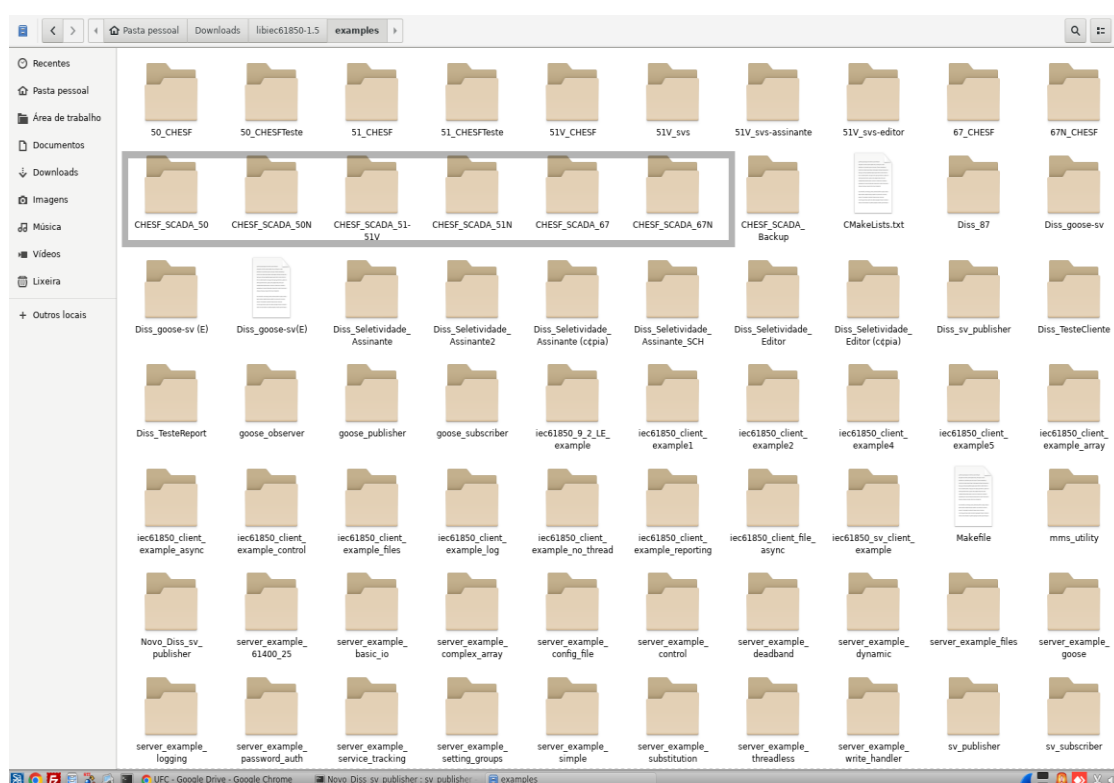
```

```
182.     }
183.     else {
184.         printf("Failed to create SV publisher\n");
185.     }
186.     return 0;
187. }
```

ANEXO A – PASTAS E ARQUIVOS DOS VIEDS

Esse anexo do trabalho apresenta as pastas que contêm os arquivos de exemplo da libIEC61850. De início, foi feito o *download* dos arquivos no *site* oficial da biblioteca (Zillgith, 2021), necessários para desenvolver os códigos com a IEC 61850. Na Figura 70, temos várias subpastas que compõem o trabalho, sendo destacadas as pastas dos VIEDs criados de forma estática, conforme comentado no Capítulo 3.

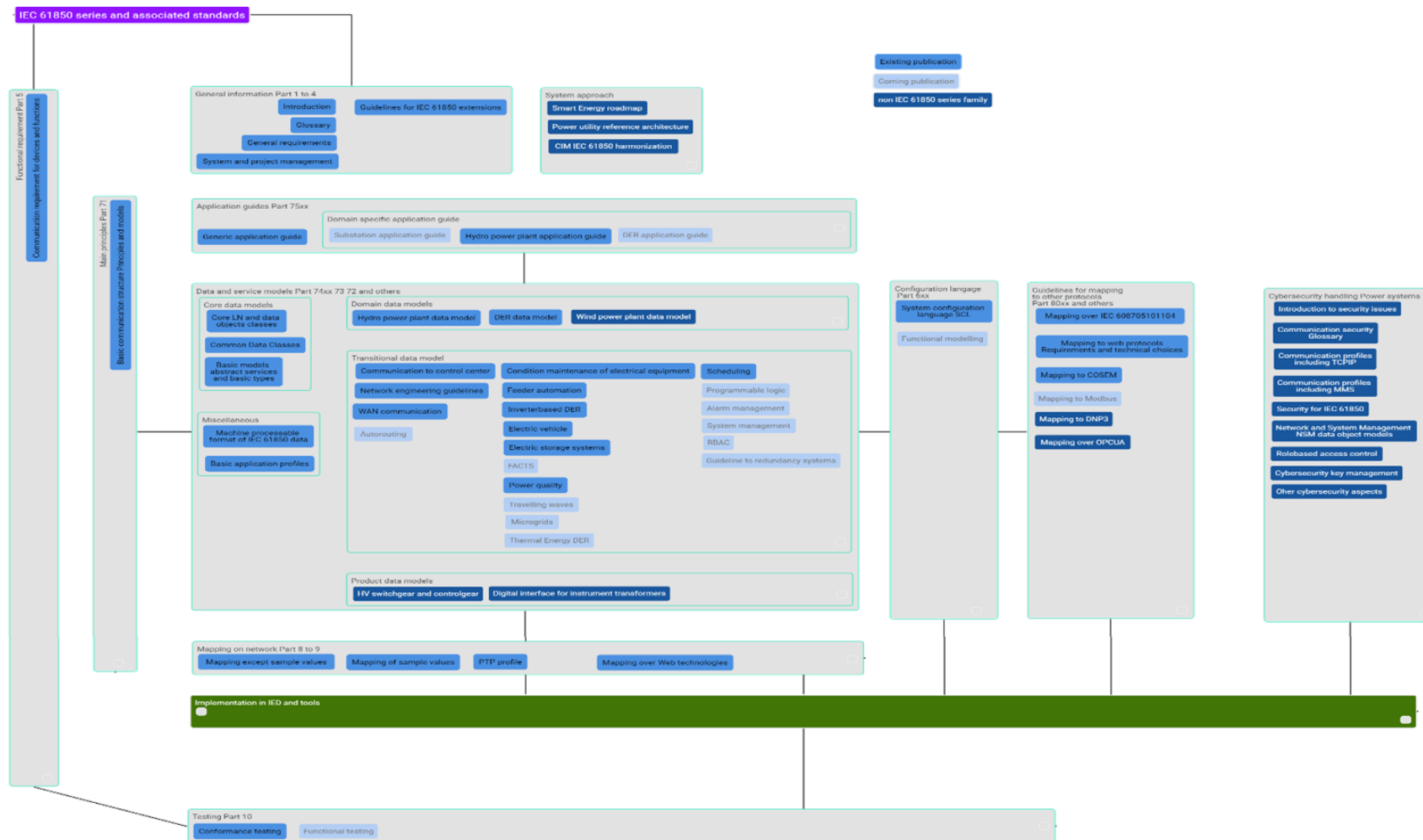
Figura 65 – Tela com as pastas dos VIEDs utilizados no trabalho



Fonte: Elaborado pelo autor.

ANEXO B – PARTES DA NORMA IEC 61850

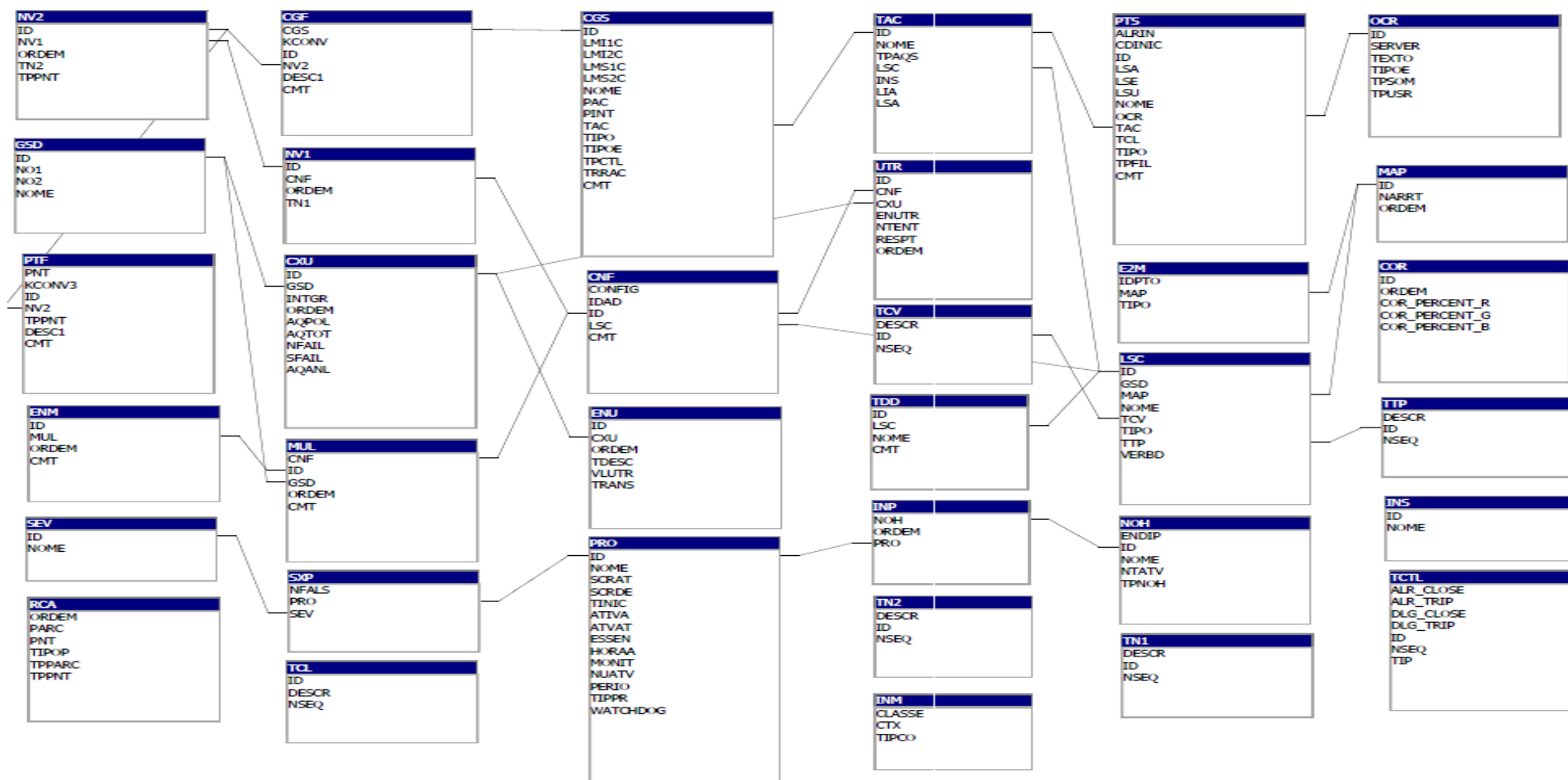
Figura 66 – Estrutura de documentos da Norma IEC 61850.



Fonte: Guise (2022)

ANEXO C – RELACIONAMENTO DAS ENTIDADES DO SCADA SAGE

Figura 67 – Relacionamentos das Entidades da Base de Dados do SCADA SAGE



Fonte: Elaborado pelo autor.

ANEXO D – ATRIBUTOS CONFIGURADOS SCADA SAGE

Entidade PDS

PDS

ALINT = NAO

ALRIN = NAO

CDINIC = NORMAL

ID = i12-ATPTOC20\$ST\$Op\$general

NOME = *Trip* sobrecorrente temporiza-
do fase A VIED

OCR = OCR_NOR_URG01

STINI = A

STNOR = A

TAC = i1

TCL = NLCL

TIPO =

TPFIL = NLFL

-----;

PDS

ALINT = NAO

ALRIN = NAO

CDINIC = NORMAL

ID = i12-BTPTOC21\$ST\$Str

NOME = Start BT

OCR =

STINI = A

STNOR = A

TAC = i1

TCL = NLCL

TIPO =

TPFIL = NLFL

-----;

PDS

ALINT = NAO

ALRIN = NAO

CDINIC = NORMAL

ID = i12-BTPTOC21\$ST\$Str-dg

NOME = Start dir geral BT

OCR =

STINI = A

STNOR = A

TAC = i1

TCL = NLCL

TIPO =

TPFIL = NLFL

-----;

PDS

ALINT = NAO

ALRIN = NAO

CDINIC = NORMAL

ID = i12-BTPTOC21\$ST\$Op\$general

NOME = *Trip* sobrecorrente temporiza-
do fase B VIED

OCR = OCR_NOR_URG01

STINI = A

STNOR = A

TAC = i1

TCL = NLCL

TIPO =

TPFIL = NLFL

-----;

PDS
 ALINT = NAO
 ALRIN = NAO
 CDINIC = NORMAL
 ID = i12-CTPTOC22\$ST\$Str
 NOME = Start CT
 OCR =
 STINI = A
 STNOR = A
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

;-----

PDS
 ALINT = NAO
 ALRIN = NAO
 CDINIC = NORMAL
 ID = i12-CTPTOC22\$ST\$Str-dg
 NOME = Start dir geral CT
 OCR =
 STINI = A
 STNOR = A
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

;-----

PDS
 ALINT = NAO
 ALRIN = NAO
 CDINIC = NORMAL

ID = i12-CTPTOC22\$ST\$Op\$general
 NOME = *Trip* sobrecorrente temporiza-
 do fase C VIED
 OCR = OCR_NOR_URG01
 STINI = A
 STNOR = A
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

PDS
 ALINT = NAO
 ALRIN = NAO
 CDINIC = NORMAL

ID = i12-TRIPPTRC1\$ST\$Tr
 NOME = *Trip* Geral
 OCR = OCR_NOR_URG01
 STINI = A
 STNOR = A
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

;-----

PDS
 ALINT = NAO
 ALRIN = NAO
 CDINIC = NORMAL
 ID = i12-BFR1RBRF1\$ST\$Str
 NOME = Start, timer running BFR1
 OCR =
 STINI = A

STNOR = A

TAC = i1

TCL = NLCL

TIPO =

TPFIL = NLFL

;-----

PDS

ALINT = NAO

ALRIN = NAO

CDINIC = NORMAL

ID = i12-BFR1RBRF1\$ST\$Str-dg

NOME = Start, timer running dir geral

BFR1

OCR =

STINI = A

STNOR = A

TAC = i1

TCL = NLCL

TIPO =

TPFIL = NLFL

;-----

PDS

ALINT = NAO

ALRIN = NAO

CDINIC = NORMAL

ID = i12-

BFR1RBRF1\$ST\$OpEx\$general

NOME = Breaker failure *trip* (external
trip) BFR1

OCR =

STINI = A

STNOR = A

TAC = i1

TCL = NLCL

TIPO =

TPFIL = NLFL

;-----

PDS

ALINT = NAO

ALRIN = NAO

CDINIC = NORMAL

ID = i15-SVGGIO3\$ST\$Ind01

NOME = *Trip* sobrecorrente com res-
trição de tensão fase A (51V) VIED

OCR = OCR_NOR_URG01

STINI = A

STNOR = A

TAC = i1

TCL = NLCL

TIPO =

TPFIL = NLFL

;-----

PDS

ALINT = NAO

ALRIN = NAO

CDINIC = NORMAL

ID = i15-SVGGIO3\$ST\$Ind02

NOME = *Trip* sobrecorrente com res-
trição de tensão fase B (51V) VIED

OCR = OCR_NOR_URG01

STINI = A

STNOR = A

TAC = i1

TCL = NLCL

TIPO =	
TPFIL = NLFL	;-----
	PDS
	ALINT = NAO
	ALRIN = NAO
	CDINIC = NORMAL
	ID = i15-SVGGIO3\$ST\$Ind05
	NOME = <i>Trip</i> sobrecorrente instan-
	tÃ¢neo fase B (50) VIED
	OCR = OCR_NOR_URG01
	STINI = A
	STNOR = A
	TAC = i1
	TCL = NLCL
	TIPO =
	TPFIL = NLFL
	;-----
	PDS
	ALINT = NAO
	ALRIN = NAO
	CDINIC = NORMAL
	ID = i15-SVGGIO3\$ST\$Ind06
	NOME = <i>Trip</i> sobrecorrente instan-
	tÃ¢neo fase C (50) VIED
	OCR = OCR_NOR_URG01
	STINI = A
	STNOR = A
	TAC = i1
	TCL = NLCL
	TIPO =
	TPFIL = NLFL
	;-----
	PDS
	ALINT = NAO
	ALRIN = NAO
	CDINIC = NORMAL
	ID = i15-SVGGIO3\$ST\$Ind04
	NOME = <i>Trip</i> sobrecorrente instan-
	tÃ¢neo fase A (50) VIED
	OCR = OCR_NOR_URG01
	STINI = A
	STNOR = A
	TAC = i1
	TCL = NLCL
	TIPO =
	TPFIL = NLFL
	;-----

PDS
 ALINT = NAO
 ALRIN = NAO
 CDINIC = NORMAL
 ID = i15-SVGGIO3\$ST\$Ind07
 NOME = *Trip sobrecorrente direcional*
 fase A (67) VIED
 OCR = OCR_NOR_URG01
 STINI = A
 STNOR = A
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

;-----
 PDS
 ALINT = NAO
 ALRIN = NAO
 CDINIC = NORMAL
 ID = i15-SVGGIO3\$ST\$Ind08
 NOME = *Trip sobrecorrente direcional*
 fase B (67) VIED
 OCR = OCR_NOR_URG01
 STINI = A
 STNOR = A
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

;-----
 PDS
 ALINT = NAO

ALRIN = NAO
 CDINIC = NORMAL
 ID = i15-SVGGIO3\$ST\$Ind09
 NOME = *Trip sobrecorrente direcional*
 fase C (67) VIED
 OCR = OCR_NOR_URG01
 STINI = A
 STNOR = A
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

;-----
 PDS
 ALINT = NAO
 ALRIN = SIM
 CDINIC = NORMAL
 ID = i15-SVGGIO3\$ST\$Ind10
 NOME = General indication (binary
 input) SV
 OCR =
 STINI = A
 STNOR = A
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

;-----
 PDS
 ALINT = NAO
 ALRIN = SIM
 CDINIC = NORMAL

ID = i15-SVGGIO3\$ST\$Ind11
 NOME = General indication (binary
 input) SV
 OCR =
 STINI = A
 STNOR = A
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

;-----

PDS
 ALINT = NAO
 ALRIN = NAO
 CDINIC = NORMAL
 ID = i15-SVGGIO3\$ST\$Ind12
 NOME = *Trip sobrecorrente direccional*
fase N (67N) VIED
 OCR = OCR_NOR_URG01
 STINI = A
 STNOR = A
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

;-----

PDS
 ALINT = NAO
 ALRIN = NAO
 CDINIC = NORMAL
 ID = i15-SVGGIO3\$ST\$Ind13

NOME = General indication (binary
 input) SV
 OCR =
 STINI = A
 STNOR = A
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

;-----

PDS
 ALINT = NAO
 ALRIN = NAO
 CDINIC = NORMAL
 ID = i15-SVGGIO3\$ST\$Ind14
 NOME = General indication (binary
 input) SV
 OCR =
 STINI = A
 STNOR = A
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

PAF
 PNT = i13-
 METMMXU1\$MX\$PPV\$phsAB-M
 KCONV1 = 1
 KCONV2 = 0
 KCONV3 = MV0

ID = i13-
 METMMXU1\$MX\$PPV\$phsAB-M
 NV2 = i13_AAAQ
 TPPNT = PAS
 DESC1 = Phase to phase voltages fase
 AB MET

;-----

Entidade PAF

PNT = i13-
 METMMXU1\$MX\$PPV\$phsBC-M
 KCONV1 = 1
 KCONV2 = 0
 KCONV3 = MV0
 ID = i13-
 METMMXU1\$MX\$PPV\$phsBC-M
 NV2 = i13_AAAQ
 TPPNT = PAS
 DESC1 = Phase to phase voltages fase
 BC MET

;-----

PAF
 PNT = i13-
 METMMXU1\$MX\$PPV\$phsCA-M
 KCONV1 = 1
 KCONV2 = 0
 KCONV3 = MV0
 ID = i13-
 METMMXU1\$MX\$PPV\$phsCA-M
 NV2 = i13_AAAQ
 TPPNT = PAS

DESC1 = Phase to phase voltages fase
 CA MET

;-----

PAF
 PNT = i13-
 METMMXU1\$MX\$A\$phsA-M
 KCONV1 = 1.0
 KCONV2 = -1
 KCONV3 = CMV0
 ID = i13-
 METMMXU1\$MX\$A\$phsA-M
 NV2 = i13_AAAQ
 TPPNT = PAS
 DESC1 = Phase currents (IL1, IL2, IL3)
 fase A

;-----

PAF
 PNT = i13-
 METMMXU1\$MX\$A\$phsB-M
 KCONV1 = 1.0
 KCONV2 = -1
 KCONV3 = CMV0
 ID = i13-METMMXU1\$MX\$A\$phsB-
 M
 NV2 = i13_AAAQ
 TPPNT = PAS
 DESC1 = Phase currents (IL1, IL2, IL3)
 fase B

;-----

PAF

PNT = i13-	
METMMXU1\$MX\$A\$phsC-M	;-----
KCONV1 = 1.0	;-----
KCONV2 = -1	PAS
KCONV3 = CMV0	ALINT =
ID = i13-METMMXU1\$MX\$A\$phsC-M	ALRIN = NAO
NV2 = i13_AAAQ	BNDMO = -1
TPPNT = PAS	CDINIC = NORMAL
DESC1 = Phase currents (IL1, IL2, IL3)	ID = i13-
fase C	METMMXU1\$MX\$PPV\$phsBC-M
	LIA = -99999
	LIE = -99999
	LIU = -99999
	LSA = 99999
	LSE = 99999
	LSU = 99999
	NOME = Phase to phase voltages fase
	BC MET
	OCR =
	TAC = i1
	TCL = NLCL
	TIPO =
	TPFIL = NLFL
	VLINIC = 0
	;-----
	;-----
Entidade PAS	PAS
ALINT =	ALINT =
ALRIN = NAO	ALRIN = NAO
BNDMO = -1	BNDMO = -1
CDINIC = NORMAL	CDINIC = NORMAL
ID = i13-	ID = i13-
METMMXU1\$MX\$PPV\$phsAB-M	METMMXU1\$MX\$PPV\$phsCA-M
LIA = -99999	
LIE = -99999	
LIU = -99999	
LSA = 99999	
LSE = 99999	
LSU = 99999	
NOME = Phase to phase voltages fase	
AB MET	
OCR =	
TAC = i1	
TCL = NLCL	
TIPO =	
TPFIL = NLFL	
VLINIC = 0	

LIA = -99999	TCL = NLCL
LIE = -99999	TIPO =
LIU = -99999	TPFIL = NLFL
LSA = 99999	VLINIC = 0
LSE = 99999	
LSU = 99999	;-----
NOME = Phase to phase voltages fase	PAS
CA MET	ALINT =
OCR =	ALRIN = NAO
TAC = i1	BNDMO = 0
TCL = NLCL	CDINIC = NORMAL
TIPO =	ID = i13-
TPFIL = NLFL	METMMXU1\$MX\$A\$phsB-M
VLINIC = 0	LIA = -99999
	LIE = -99999
;-----	LIU = -99999
;-----	LSA = 99999
PAS	LSE = 99999
ALINT =	LSU = 99999
ALRIN = NAO	NOME = Phase currents (IL1, IL2, IL3)
BNDMO = 0	ang fase B
CDINIC = NORMAL	OCR =
ID = i13-	TAC = i1
METMMXU1\$MX\$A\$phsA-M	TCL = NLCL
LIA = -99999	TIPO =
LIE = -99999	TPFIL = NLFL
LIU = -99999	VLINIC = 0
LSA = 99999	
LSE = 99999	;-----
LSU = 99999	PAS
NOME = Phase currents (IL1, IL2, IL3)	ALINT =
ang fase A	ALRIN = NAO
OCR =	BNDMO = 0
TAC = i1	CDINIC = NORMAL

ID = i13-
 METMMXU1\$MX\$A\$phsB-M
 LIA = -99999
 LIE = -99999
 LIU = -99999
 LSA = 99999
 LSE = 99999
 LSU = 99999
 NOME = Phase currents (IL1, IL2, IL3)
 fase B ME
 OCR =
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL
 VLINIC = 0

Entidade PNT

PNT = i12-CTPTOC22\$ST\$Mod
 KCONV3 = INC0
 ID = i12-CTPTOC22\$ST\$Mod
 NV2 = i12_ATTA
 TPPNT = PTS
 DESC1 = Mode CT

;-----
 PTF
 PNT = i12-CTPTOC22\$ST\$Beh
 KCONV3 = INS0
 ID = i12-CTPTOC22\$ST\$Beh
 NV2 = i12_ATTA
 TPPNT = PTS

DESC1 = Behaviour CT
 ;-----
 PTF
 PNT = i12-CTPTOC22\$ST\$Health
 KCONV3 = INS0
 ID = i12-CTPTOC22\$ST\$Health
 NV2 = i12_ATTA
 TPPNT = PTS
 DESC1 = Health CT

;-----
 PTF
 PNT = i12-TRIPPTRC1\$ST\$Mod
 KCONV3 = INC0
 ID = i12-TRIPPTRC1\$ST\$Mod
 NV2 = i12_ATTA
 TPPNT = PTS
 DESC1 = Mode *TRIP*

;-----
 PTF
 PNT = i12-TRIPPTRC1\$ST\$Beh
 KCONV3 = INS0
 ID = i12-TRIPPTRC1\$ST\$Beh
 NV2 = i12_ATTA
 TPPNT = PTS
 DESC1 = Behaviour *TRIP*

;-----
 PTF
 PNT = i12-TRIPPTRC1\$ST\$Health
 KCONV3 = INS0
 ID = i12-TRIPPTRC1\$ST\$Health

```

NV2  = i12_ATTA
TPPNT = PTS
DESC1 = Health TRIP

;-----
PTF
PNT  = i12-BK1XCBR1$ST$Mod
KCONV3 = INC0
ID   = i12-BK1XCBR1$ST$Mod
NV2  = i12_ATTA
TPPNT = PTS
DESC1 = Mode BK1

;-----
PTF
PNT  = i12-BK1XCBR1$ST$Beh
KCONV3 = INS0
ID   = i12-BK1XCBR1$ST$Beh
NV2  = i12_ATTA
TPPNT = PTS
DESC1 = Behaviour BK1

;-----
PTF
PNT  = i12-BK1XCBR1$ST$Health
KCONV3 = INS0
ID   = i12-BK1XCBR1$ST$Health
NV2  = i12_ATTA
TPPNT = PTS
DESC1 = Health BK1

;-----
PTF
PNT  = i12-BK1XCBR1$ST$OpCnt

```

```

KCONV3 = INS0
ID   = i12-BK1XCBR1$ST$OpCnt
NV2  = i12_ATTA
TPPNT = PTS
DESC1 = Operation counter BK1

;-----
PTF
PNT  = i12-BK1XCBR1$ST$CBOpCap
KCONV3 = INS0
ID   = i12-BK1XCBR1$ST$CBOpCap
NV2  = i12_ATTA
TPPNT = PTS
DESC1 = Circuit breaker operating ca-
pability BK

;-----
PTF
PNT  = i15-SVGGIO3$ST$Mod
KCONV3 = INC0
ID   = i15-SVGGIO3$ST$Mod
NV2  = i15_ATTA
TPPNT = PTS
DESC1 = Mode SV

;-----
PTF
PNT  = i15-SVGGIO3$ST$Beh
KCONV3 = INS0
ID   = i15-SVGGIO3$ST$Beh
NV2  = i15_ATTA
TPPNT = PTS
DESC1 = Behaviour SV

```

;------
 PTF
 PNT = i15-SVGGIO3\$ST\$Health
 KCONV3 = INS0
 ID = i15-SVGGIO3\$ST\$Health
 NV2 = i15_ATTA
 TPPNT = PTS
 DESC1 = Health SV

;------

Entidade PTS

ALRIN = NAO
 CDINIC = NORMAL
 ID = i12-CTPTOC22\$ST\$Mod
 LSA = 32767
 LSE = 32767
 LSU = 32767
 NOME = Mode CT
 OCR =
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

;------

PTS
 ALRIN = NAO
 CDINIC = NORMAL
 ID = i12-CTPTOC22\$ST\$Beh
 LSA = 32767
 LSE = 32767
 LSU = 32767

NOME = Behaviour CT
 OCR =
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

;------

PTS
 ALRIN = NAO
 CDINIC = NORMAL
 ID = i12-CTPTOC22\$ST\$Health
 LSA = 32767
 LSE = 32767
 LSU = 32767
 NOME = Health CT
 OCR =
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

;------

PTS
 ALRIN = NAO
 CDINIC = NORMAL
 ID = i12-BK1XCBR1\$ST\$Mod
 LSA = 32767
 LSE = 32767
 LSU = 32767
 NOME = Mode BK1
 OCR =
 TAC = i1
 TCL = NLCL

```

TIPO =
TPFIL = NLFL

;-----
PTS
ALRIN = NAO
CDINIC = NORMAL
ID   = i12-BK1XCBR1$ST$Beh
LSA   = 32767
LSE   = 32767
LSU   = 32767
NOME  = Behaviour BK1
OCR   =
TAC   = i1
TCL   = NLCL
TIPO  =
TPFIL = NLFL

;-----
PTS
ALRIN = NAO
CDINIC = NORMAL
ID   = i12-BK1XCBR1$ST$Health
LSA   = 32767
LSE   = 32767
LSU   = 32767
NOME  = Health BK1
OCR   =
TAC   = i1
TCL   = NLCL
TIPO  =
TPFIL = NLFL

;-----

```

```

PTS
ALRIN = NAO
CDINIC = NORMAL
ID   = i12-BK1XCBR1$ST$OpCnt
LSA   = 32767
LSE   = 32767
LSU   = 32767
NOME  = Operation counter BK1
OCR   =
TAC   = i1
TCL   = NLCL
TIPO  =
TPFIL = NLFL

;-----
PTS
ALRIN = NAO
CDINIC = NORMAL
ID   = i12-BK1XCBR1$ST$CBOpCap
LSA   = 32767
LSE   = 32767
LSU   = 32767
NOME  = Circuit breaker operating ca-
pability BK1
OCR   =
TAC   = i1
TCL   = NLCL
TIPO  =
TPFIL = NLFL

;-----
PTS
ALRIN = NAO
CDINIC = NORMAL

```

ID = i15-SVGGIO3\$ST\$Mod
 LSA = 32767
 LSE = 32767
 LSU = 32767
 NOME = Mode SV
 OCR =
 TAC = i1
 TCL = NLCL
 TIPO =
 TPFIL = NLFL

;-----

PTS

ALRIN = NAO

CDINIC = NORMAL

ID = i15-SVGGIO3\$ST\$Beh

LSA = 32767

LSE = 32767

LSU = 32767

NOME = Behaviour SV

OCR =

TAC = i1

TCL = NLCL

TIPO =

TPFIL = NLFL

;-----

PTS

ALRIN = NAO

CDINIC = NORMAL

ID = i15-SVGGIO3\$ST\$Health

LSA = 32767

LSE = 32767

LSU = 32767

NOME = Health SV

OCR =

TAC = i1

TCL = NLCL

TIPO =

TPFIL = NLFL

;-----

Entidade CGF

CGS = i12-BKR1CSWI1\$CO\$Pos

KCONV =

ID = i12-BKR1CSWI1\$CO\$Pos

NV2 = i12_CSIM

DESC1 = Switch, general BKR1

;-----

Entidade CGS

ID = i12-BKR1CSWI1\$CO\$Pos

LMI1C = -9999

LMI2C = -9999

LMS1C = 9999

LMS2C = 9999

NOME = Switch, general BKR1

PAC = i12-BKR1CSWI1\$ST\$Pos

PINT =

TAC = i1

TIPO = PDS

TIPOE = CTCL

TPCTL = CSAC

TRRAC = 35

;-----

Entidade ENM

ID = i12

MUL = i1

ORDEM = 2

;-----

Entidade MAP

ID = VIRTUA

NARRT = Alarmes IEC/61850 IED

VIRTUAL_1

ORDEM = 65

;-----

Entidade NV1

CNF = i1

CONFIG = VIRTUAL_1CFG

ORDEM = 1

ID = i11

TN1 = NLN1

;-----

NV1

CNF = i1

CONFIG = VIRTUAL_1PRO

ORDEM = 2

ID = i12

TN1 = NLN1

;-----

NV1

CNF = i1

CONFIG = VIRTUAL_1MET

ORDEM = 3

ID = i13

TN1 = NLN1

;-----

NV1

CNF = i1

CONFIG = VIRTUAL_1CON

ORDEM = 4

ID = i14

TN1 = NLN1

;-----

NV1

CNF = i1

CONFIG = VIRTUAL_1ANN

ORDEM = 5

ID = i15

TN1 = NLN1

;-----

Entidade PDF

PNT = i12-CTPTOC22\$ST\$Str

KCONV = ACD0

ID = i12-CTPTOC22\$ST\$Str

NV2 = i12_ADAQ

TPPNT = PDS

DESC1 = Start CT

;-----

PDF

PNT = i12-CTPTOC22\$ST\$Str-dg

KCONV = ACD5

ID = i12-CTPTOC22\$ST\$Str-dg

NV2 = i12_ADAQ

TPPNT = PDS

DESC1 = Start dir geral CT

;------

PDF

PNT = i12-

CTPTOC22\$ST\$Op\$general

KCONV = NOR

ID = i12-CTPTOC22\$ST\$Op\$general

NV2 = i12_ADAQ

TPPNT = PDS

DESC1 = Operate CT

PDF

PNT = i12-BK1XCBR1\$ST\$Loc

KCONV = SPS0

ID = i12-BK1XCBR1\$ST\$Loc

NV2 = i12_ADAQ

TPPNT = PDS

DESC1 = Local operation BK1

;------

PDF

PNT = i12-BK1XCBR1\$ST\$Pos

KCONV = DPC0

ID = i12-BK1XCBR1\$ST\$Pos

NV2 = i12_ADAQ

TPPNT = PDS

DESC1 = Switch position BK1

;------

PDF

PNT = i12-BK1XCBR1\$ST\$BlkOpn

KCONV = SPC0

ID = i12-BK1XCBR1\$ST\$BlkOpn

NV2 = i12_ADAQ

TPPNT = PDS

DESC1 = Block opening BK1

;------

PDF

PNT = i12-BK1XCBR1\$ST\$BlkCls

KCONV = SPC0

ID = i12-BK1XCBR1\$ST\$BlkCls

NV2 = i12_ADAQ

TPPNT = PDS

DESC1 = Block closing BK1

;------

PDF

PNT = i12-BKR1CSWI1\$ST\$Pos

KCONV = DPC0

ID = i12-BKR1CSWI1\$ST\$Pos

NV2 = i12_ADAQ

TPPNT = PDS

DESC1 = Switch, general BKR1

;------

PDF

PNT = i12-BKR1CSWI1\$ST\$OpOpn

KCONV = ACT0

ID = i12-BKR1CSWI1\$ST\$OpOpn

NV2 = i12_ADAQ

```

TPPNT = PDS
DESC1 = Operation Open Switch
BKR1

;-----
PDF
PNT = i12-BKR1CSWI1$ST$OpCls
KCONV = ACT0
ID = i12-BKR1CSWI1$ST$OpCls
NV2 = i12_ADAQ
TPPNT = PDS
DESC1 = Operation Close Switch
BKR1

;-----
PDF
PNT = i15-SVGGIO3$ST$Ind01
KCONV = SPS0
ID = i15-SVGGIO3$ST$Ind01
NV2 = i15_ADAQ
TPPNT = PDS
DESC1 = General indication (binary
input) SV

;-----
PDF
PNT = i15-SVGGIO3$ST$Ind02
KCONV = SPS0
ID = i15-SVGGIO3$ST$Ind02
NV2 = i15_ADAQ
TPPNT = PDS
DESC1 = General indication (binary
input) SV

```

```

;-----
PDF
PNT = i15-SVGGIO3$ST$Ind03
KCONV = SPS0
ID = i15-SVGGIO3$ST$Ind03
NV2 = i15_ADAQ
TPPNT = PDS
DESC1 = General indication (binary
input) SV

;-----
PDF
PNT = i15-SVGGIO3$ST$Ind04
KCONV = SPS0
ID = i15-SVGGIO3$ST$Ind04
NV2 = i15_ADAQ
TPPNT = PDS
DESC1 = General indication (binary
input) SV

;-----
PDF
PNT = i15-SVGGIO3$ST$Ind05
KCONV = SPS0
ID = i15-SVGGIO3$ST$Ind05
NV2 = i15_ADAQ
TPPNT = PDS
DESC1 = General indication (binary
input) SV

;-----
PDF
PNT = i15-SVGGIO3$ST$Ind06
KCONV = SPS0

```

ID = i15-SVGGIO3\$ST\$Ind06
 NV2 = i15_ADAQ
 TPPNT = PDS
 DESC1 = General indication (binary
 input) SV

 ;-----
 PDF
 PNT = i15-SVGGIO3\$ST\$Ind07
 KCONV = SPS0
 ID = i15-SVGGIO3\$ST\$Ind07
 NV2 = i15_ADAQ
 TPPNT = PDS
 DESC1 = General indication (binary
 input) SV

 ;-----
 PDF
 PNT = i15-SVGGIO3\$ST\$Ind08
 KCONV = SPS0
 ID = i15-SVGGIO3\$ST\$Ind08
 NV2 = i15_ADAQ
 TPPNT = PDS
 DESC1 = General indication (binary
 input) SV

 ;-----
 PDF
 PNT = i15-SVGGIO3\$ST\$Ind09
 KCONV = SPS0
 ID = i15-SVGGIO3\$ST\$Ind09
 NV2 = i15_ADAQ
 TPPNT = PDS

DESC1 = General indication (binary
 input) SV

 ;-----
 PDF
 PNT = i15-SVGGIO3\$ST\$Ind10
 KCONV = SPS0
 ID = i15-SVGGIO3\$ST\$Ind10
 NV2 = i15_ADAQ
 TPPNT = PDS
 DESC1 = General indication (binary
 input) SV

 ;-----
 PDF
 PNT = i15-SVGGIO3\$ST\$Ind11
 KCONV = SPS0
 ID = i15-SVGGIO3\$ST\$Ind11
 NV2 = i15_ADAQ
 TPPNT = PDS
 DESC1 = General indication (binary
 input) SV

 ;-----
 PDF
 PNT = i15-SVGGIO3\$ST\$Ind12
 KCONV = SPS0
 ID = i15-SVGGIO3\$ST\$Ind12
 NV2 = i15_ADAQ
 TPPNT = PDS
 DESC1 = General indication (binary
 input) SV

 ;-----

PDF
 PNT = i15-SVGGIO3\$ST\$Ind13
 KCONV = SPS0
 ID = i15-SVGGIO3\$ST\$Ind13
 NV2 = i15_ADAQ
 TPPNT = PDS
 DESC1 = General indication (binary
 input) SV

 ;-----
 PDF
 PNT = i15-SVGGIO3\$ST\$Ind14
 KCONV = SPS0
 ID = i15-SVGGIO3\$ST\$Ind14
 NV2 = i15_ADAQ
 TPPNT = PDS
 DESC1 = General indication (binary
 input) SV

 ;-----
 PDF
 PNT = i15-SVGGIO3\$ST\$Ind15
 KCONV = SPS0
 ID = i15-SVGGIO3\$ST\$Ind15
 NV2 = i15_ADAQ
 TPPNT = PDS
 DESC1 = General indication (binary
 input) SV

 ;-----
 PDF
 PNT = i15-SVGGIO3\$ST\$Ind16
 KCONV = SPS0
 ID = i15-SVGGIO3\$ST\$Ind16

NV2 = i15_ADAQ
 TPPNT = PDS
 DESC1 = General indication (binary
 input) SV

 ;-----
 PDF
 PNT = i15-SVGGIO3\$ST\$Ind17
 KCONV = SPS0
 ID = i15-SVGGIO3\$ST\$Ind17
 NV2 = i15_ADAQ
 TPPNT = PDS
 DESC1 = General indication (binary
 input) SV

 ;-----
 PDF
 PNT = i15-SVGGIO3\$ST\$Ind18
 KCONV = SPS0
 ID = i15-SVGGIO3\$ST\$Ind18
 NV2 = i15_ADAQ
 TPPNT = PDS
 DESC1 = General indication (binary
 input) SV

 ;-----
 PDF
 PNT = i15-SVGGIO3\$ST\$Ind19
 KCONV = SPS0
 ID = i15-SVGGIO3\$ST\$Ind19
 NV2 = i15_ADAQ
 TPPNT = PDS
 DESC1 = General indication (binary
 input) SV

```

;-----
PDF
PNT  = i15-SVGGIO3$ST$Ind20
KCONV = SPS0
ID   = i15-SVGGIO3$ST$Ind20
NV2  = i15_ADAQ
TPPNT = PDS
DESC1 = General indication (binary
input) SV

;-----
PDF
PNT  = i15-SVGGIO3$ST$Ind21
KCONV = SPS0
ID   = i15-SVGGIO3$ST$Ind21
NV2  = i15_ADAQ
TPPNT = PDS
DESC1 = General indication (binary
input) SV

;-----
PDF
PNT  = i15-SVGGIO3$ST$Ind22
KCONV = SPS0
ID   = i15-SVGGIO3$ST$Ind22
NV2  = i15_ADAQ
TPPNT = PDS
DESC1 = General indication (binary
input) SV

;-----
PDF
PNT  = i15-SVGGIO3$ST$Ind23

```

```

KCONV = SPS0
ID   = i15-SVGGIO3$ST$Ind23
NV2  = i15_ADAQ
TPPNT = PDS
DESC1 = General indication (binary
input) SV

;-----
PDF
PNT  = i15-SVGGIO3$ST$Ind24
KCONV = SPS0
ID   = i15-SVGGIO3$ST$Ind24
NV2  = i15_ADAQ
TPPNT = PDS
DESC1 = General indication (binary
input) SV

;-----
PDF
PNT  = i15-SVGGIO3$ST$Ind25
KCONV = SPS0
ID   = i15-SVGGIO3$ST$Ind25
NV2  = i15_ADAQ
TPPNT = PDS
DESC1 = General indication (binary
input) SV

;-----
PDF
PNT  = i15-SVGGIO3$ST$Ind26
KCONV = SPS0
ID   = i15-SVGGIO3$ST$Ind26
NV2  = i15_ADAQ
TPPNT = PDS

```

DESC1 = General indication (binary
input) SV

;-----

PDF

PNT = i15-SVGGIO3\$ST\$Ind27

KCONV = SPS0

ID = i15-SVGGIO3\$ST\$Ind27

NV2 = i15_ADAQ

TPPNT = PDS

DESC1 = General indication (binary
input) SV

;-----

PDF

PNT = i15-SVGGIO3\$ST\$Ind28

KCONV = SPS0

ID = i15-SVGGIO3\$ST\$Ind28

NV2 = i15_ADAQ

TPPNT = PDS

DESC1 = General indication (binary
input) SV

;-----

PDF

PNT = i15-SVGGIO3\$ST\$Ind29

KCONV = SPS0

ID = i15-SVGGIO3\$ST\$Ind29

NV2 = i15_ADAQ

TPPNT = PDS

DESC1 = General indication (binary
input) SV

;-----

PDF

PNT = i15-SVGGIO3\$ST\$Ind30

KCONV = SPS0

ID = i15-SVGGIO3\$ST\$Ind30

NV2 = i15_ADAQ

TPPNT = PDS

DESC1 = General indication (binary
input) SV

;-----

PDF

PNT = i15-SVGGIO3\$ST\$Ind31

KCONV = SPS0

ID = i15-SVGGIO3\$ST\$Ind31

NV2 = i15_ADAQ

TPPNT = PDS

DESC1 = General indication (binary
input) SV

;-----

PDF

PNT = i15-SVGGIO3\$ST\$Ind32

KCONV = SPS0

ID = i15-SVGGIO3\$ST\$Ind32

NV2 = i15_ADAQ

TPPNT = PDS

DESC1 = General indication (binary
input) SV

;-----

Entidade LSC

GSD = MND

ID = VIRTUAL1

MAP = GERAL

NOME = IED VIRTUAL_1 proto
 IEC/61850
 NSRV1 = localhost
 NSRV2 = localhost
 TCV = CNVO
 TIPO = AD
 TTP = MMST
 VERBD = SCL_AUTO

;-----

Entidade TAC

TAC
 INS = MND
 NOME = Aquisicao do IED i1
 ID = i1
 LSC = VIRTUAL1
 TPAQS = ASAC

;-----

Entidade TDD

TDD
 ID = i1
 LSC = VIRTUAL1
 NOME = mySPCSO20_229

;-----

Entidade MUL

CNF = i1
 ID = i1
 GSD = MND
 ORDEM = 100

;-----

Entidade CNF

CONFIG = ApTitle= 1 1 10 / 1 1 10
 AeQ= 1 PS= 1 / 1 SS= 1 TS= 1 IDAD=
 600 KEEP= 5 NREP= 3 TOUT= 10
 MPDU= 0 OPMSK= 110 GOOSE= 0
 ID = i1
 LSC = VIRTUAL1

;-----

Entidade NV2

CONFIG = PDF
 ORDEM = 1
 ID = i12_ADAQ
 NV1 = i12
 TN2 = ADAQ
 TPPNT = PDF

;-----

;NV2

;CONFIG = PTF

;ORDEM = 2

;ID = i12_ATTA

;NV1 = i12

;TN2 = ATTA

;TPPNT = PTF

;

;-----

NV2

CONFIG = CGF

ORDEM = 2

ID = i12_CSIM

NV1 = i12

TN2 = CSIM

TPPNT = CGF

```

;-----

```

```

NV2

```

```

CONFIG = PDF

```

```

ORDEM = 1

```

```

ID   = i15_ADAQ

```

```

NV1  = i15

```

```

TN2  = ADAQ

```

```

TPPNT = PDF

```

```

;-----

```

```

NV2

```

```

CONFIG = PAF

```

```

ORDEM = 2

```

```

ID   = i15_AAAQ

```

```

NV1  = i15

```

```

TN2  = AAAQ

```

```

TPPNT = PAF

```

```

;-----

```

```

;NV2

```

```

;CONFIG = PTF

```

```

;ORDEM = 3

```

```

;ID   = i15_ATTA

```

```

;NV1  = i15

```

```

;TN2  = ATTA

```

```

;TPPNT = PTF

```

```

;

```

```

;-----

```

Entidade NOH

```

NOH

```

```

    ENDIP= 200.229.40.1

```

```

    ID= srv1-dc-ssl-mnd

```

```

    NOME= Servidor SAGE Linux 1

```

```

NTATV= 4

```

```

TPNOH= SERVIDOR

```

```

;-----

```

```

NOH

```

```

    ENDIP= 200.229.40.2

```

```

    ID= srv2-dc-ssl-mnd

```

```

    NOME= Servidor SAGE Linux 2

```

```

    NTATV= 4

```

```

    TPNOH= SERVIDOR

```

Entidade GSD

```

GSD

```

```

    ID= MND

```

```

    NO1= srv1-dc-ssl-mnd

```

```

    NO2= srv2-dc-ssl-mnd

```

```

    NOME= Gateway Scada

```