Marcus Vinicius de Oliveira Bezerra

Projeto de um Sistema de Controle de Mitigação de Falhas Críticas

Brasil

2013

Marcus Vinicius de Oliveira Bezerra

Projeto de um Sistema de Controle de Mitigação de Falhas Críticas

Trabalho de Conclusão de Curso apresentado à Escola Politécnica da Universidade de São Paulo

Universidade de São Paulo – USP

Escola Politécnica - Departamento de Engenharia Mecatrônica

Programa de Graduação

Orientador: Prof. Dr. Diolino José dos Santos Filho

Brasil

2013

FICHA CATALOGRÁFICA

Bezerra, Marcus Vinicius de Oliveira
Projeto de um sistema de controle de mitigação de falhas críticas / M.V.O. Bezerra. -- São Paulo, 2013.
115 p.

Trabalho de Formatura - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos.

1. Sistemas de controle 2. Controladores programáveis 3. Sistemas discretos I. Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos II. t.

Marcus Vinicius de Oliveira Bezerra

Projeto de um Sistema de Controle de Mitigação de Falhas Críticas

Trabalho de Conclusão de Curso apresentado à Escola Politécnica da Universidade de São Paulo

Trabalho aprovado. Brasil, 2013:

Prof. Dr. Diolino José dos Santos Filho
Orientador

Prof. Dr. Arturo Forner Cordero
Membro da Banca Avaliadora

Profa. Dra. Larissa Driemeier
Membro da Banca Avaliadora

Brasil
2013

Dedico esse trabalho à minha família, sem seu incondicional apoio nunca chegaria onde estou.

Agradecimentos

Agradeço primeiramente ao Professor Doutor Diolino José dos Santos Filho, cuja orientação foi essencial para o desenvolvimento do trabalho. Seu conhecimento sobre o tema permitia sempre encontrar o correto direcionamento e sanar com facilidade as dúvidas que sirgiam. Agradeço também ao Doutorando Reinando Squillante Júnior, engenheiro que trabalhou na área e sempre oferece consultoria de grande qualidade para acompanhamento e desenvolvimento do trabalho, principalmente nos momentos iniciais para caracterização da planta e primeiros passos sobre sistemas instrumentados de segurança e mitigação de falhas. Agradecimentos também ao Doutorando Marcosiris Amorim de Oliveira Pessoa, pela disponibilidade e atenção dadas às partes práticas do trabalho e melhor utilização do laboratorio de pesquisa. Esse grupo de pesquisadores permitiu o máximo de aproveitamento no desenvolvimento do trabalho. Agradeço também aos alunos Ricardo Mires e Yuri Capitani Gladek, que desenvolveram projeto de temática semelhante e dividiram durante o ano o mesmo laboratório, cujo compartilhamento de experiências agregou bastante aos dois trabalhos.

"Hindsight is a wonderful thing but foresight is better,
especially when it comes to saving a life,
or some pain!" - William Blake

Resumo

Num contexto de elevada competição, num mundo cada vez mais globalizado, as plantas industriais vêm se tornando um complexo sistema que envolve uma grande quantidade de processos. Esses, por sua vez, podem trazer um conjunto de riscos aos operadores humanos, ao meio ambiente e aos próprios bens, como no caso de plantas chamadas críticas, onde uma falha pode trazer graves consequências; os principais exemplos são usinas termoelétricas e nucleares. Esse Trabalho de Conclusão de Curso tem como principal objetivo projetar um sistema de controle de mitigação de falhas para uma Estação de Compressão de Gás Natural. O sistema de controle em questão age na ocorrência de uma falha, trazendo a planta para um estado de equilíbrio, isolando o problema e garantindo que as consequências da falha não se propaguem para o restante da planta. O estado da arte no projeto de sistemas de segurança baseia-se em uma abordagem orientada a camadas ou níveis de ação de controle, na qual se projetam diversas camadas de controle individuais e autônomas, cada uma agindo na tentativa de garantir que o sistema seja seguro; pode ser sucintamente dividida nas camadas de controle básico, segurança e mitigação, cada uma delas responsável por tratar de uma determinada forma o problema de controlar o sistema, garantindo que a vida humana, o meio ambiente e os bens sejam protegidos em todos os momentos. O presente trabalho considera o projeto do sistema de mitigação em que a modelagem da planta baseia-se no conceito de Sistemas a Eventos Discretos aplicando-se redes de Petri para a construção dos modelos. A partir deste formalismo serão gerados os programas em linguagem para Controladores Programáveis (CPs), de acordo com a norma IEC61131-3. Por fim, implementar-se-á a programação num CP para validação e testes do sistema de controle.

Palavras-chaves: Sistema Instrumentado de Segurança; Sistema a Eventos Discretos; Mitigação de Falhas; IEC 61508; Controladores Programáveis.

Abstract

In a context of increasing competition, in a world greatly globalized, industrial plants have become a complex system that involves lots of processes. These, in turn, can bring a number of risks to human operators, the environment and property, as in the case of so called critical plants, where failures can cause serious consequences; the main examples are power and nuclear plants. This essay has as main objective to design a mitigation control system for a Natural Gas Compression Station. The control system acts in the event of a failure, bringing the plant to a state of equilibrium, isolating the problem and ensuring that the consequences of the failure do not propagate to the rest of the plant. The state of the art in the design of security systems based on a approach with layers or levels of control action, which protrude several layers of independent and autonomous control, each acting in an attempt to ensure that the system is secure; it can be briefly divided into the layers of basic, safety and mitigation control, each responsible for dealing with a particular approach to the safety problem, ensuring that human life, the environment and property are protected at all times. This paper considers the design of mitigation system in which the mod-eling of the plant is based on the concept of Discrete Event Systems by applying Petri nets to the construction of models. From this formalism language programs will be created for Programmable Logic Controllers (PLCs), according to IEC61131-3. Finally, we will implement the pro-gram in a PLC for validation and testing of the control system.

Key-words: Instrumented Safety System; Discrete Event System; Disaster Mitigation; IEC 61508; Programmable Controllers.

Lista de ilustrações

Lista de tabelas

Sumário

Introdução

Ao que tudo indica, o desenvolvimento tecnológico é cada vez mais constante, com novas barreiras sendo superadas a cada dia. Num mundo cada vez mais globalizado, grande ênfase é dada a eficiência dos processos, onde quem produz com os menores custos tem vantagens competitivas estruturais sobre os concorrentes. Ainda mais, grande ênfase é também dada à segurança no processo, de foram a garantir que em situações adversas o processo tenha condições de ser levado para um estado seguro e que volte à operação.

Na Engenharia, os sistemas são projetados para que não falhem dentro de um tempo de vida útil esperado. Contudo, todo sistema é passível de falha, pois está sujeito a fatores humanos e tecnológicos por certas vezes imprevisíveis. Dentre esse conjunto de sistemas há aqueles considerados críticos, nos quais a falha pode trazer graves riscos à vida de seus usuários, ao meio ambiente ou à segurança da uma indústria, como é o caso de aviões, usinas termoelétricos, usinas nucleares e, no caso em questão, em estações de compressão de gás natural. Nesse contexto, grande importância é dada à atuação dos sistemas de controle básico, controle de segurança e de mitigação de falhas.

O Trabalho de Conclusão de Curso apresentado envolve o projeto de um sistema de controle de mitigação de falhas de uma planta industrial existente, modelado como um Sistema a Eventos Discretos (SED). O sistema de controle em questão atua na ocorrência de uma falha crítica, quando o sistema de prevenção desta falha não atua de forma adequada, de modo a minimizar os efeitos causados aos operadores humanos, ao meio ambiente e aos bens. O estado da arte no que diz respeito à engenharia de controle de segurança trabalha num regime de camadas de ações de controle, como apresentado na Figura 1. Cada camada adiciona mais segurança ao sistema. O objetivo final do projeto foi desenvolver o projeto do sistema de controle de mitigação de falhas na planta apresentada como estudo de caso. Foi projetado todo o sistema, incluindo desde o conjunto de sensoriamente necessário para detecção das falhas até o conjunto de atuadores para isolamento do problema. Além disso, foi feita a programação de um CP de segurança para validação do sistema projetado. Um protótipo eletrônico será montado para simular a planta como um sistema a eventos discretos, simulando seu funcionamento normal e os estados de erro que são o foco da mitigação. Este protótipo será desenvolvido utilizando o conceito Hardware in the Loop (HIL).

Figura 1 – Camadas de Proteção

Fonte: (SQUILLANTE, 2011)

1 Revisão da Literatura

1.1 Revisão dos Artigos

A primeira etapa do projeto envolveu o levantamento bibliográfico necessário para melhor compreensão do assunto, com os principais assuntos sendo retratadas abaixo. Os temas que exigiram maior aprofundamento estão discutidos em capítulos próprios da revisão.

Como apresentado em (SUBRAMANIAN et al., 1996), a identificação de ameaças em softwares envolve sua identificação e mitigação. O projeto de software é um processo criativo e não existem guias definitivos sobre sua concepção. O artigo apresenta padrões de segurança e mitigação de falhas que podem ser seguidos, o que permite prevenção apropriada de riscos. É ainda dividido em dois grandes grupos para facilitar compreensão e diferenciação de uso, para projeto de segurança e de mitigação.Sendo que foi possível identificar quais são as principais causas de mitigação em softwares, um ponto importante na relevância do projeto. Além disso, os padrões identificados podem ser aplicados na estrutura do código do Programmable Logic Controller (PLC) de sistema de controle de mitigação de falhas a ser projetado.

Num contexto mais geral e não sobre processos, (GEORGAKOPOULOS et al., 2000) comenta sobre a mitigação de crises em escala municipal ou generalizada, por exemplo, endereçando situações em que um processo de mitigação minimiza os efeitos adversos de uma falha. Processos do tipo devem possuir uma coordenação flexível e mudanças dinâmicas de maneira a viabilizar que especialistas lidem com situações inexperadas. Além disso, deve possuir estrutura o suficiente para que não se chegue num estado caótico e tornar o processo cada vez mais eficiente. Esse artigo introduz o Collaboration Management Infrastructure (CMI) e também descreve sua capacidade em auxiliar os processos de mitigação. O estudo permite que se identifique qual o melhor modelo de trabalho para reverter determinada situação de crise e qual a melhor forma de coordenadas informações, times e recursos. Agora numa abordagem um pouco mais genérica, a análise permite verificar como se comportaria todo o sistema no caso de uma falha crítica, daí fala-se sobre mitigação de situações extremas. Como o projeto atua sobre uma estação de compressão de gás natural, conhecer metolodias para coordenação de uma grande número de pessoas e processos pode tornar a modelagem ainda mais precisa e verossímil, sendo que o assunto é tratado de maneira fundamentada e concisa.

Em continuidade ao assunto, (KHARCHENKO; BREZHNEV; BOYARCHUK, 2011) apresenta uma forma de garantir que toda a malha está segura é o mesmo que

garantir que cada um dos elementos atinge às especificações de segurança estabelecidas, num nível de risco aceitável para a sociedade, seres humanos, bens e ambiente. O artigo fala sobre uma nova abordagem de análise matricial para o desenvolvimento de Smart Grids (SGs). Faz-se um estudo de caso sobre usinas nucleares, onde segurança e confiança são cruciais para a operação.A abordagem matricial é semelhante à apresentada na norma, mas a metolodiga apresentada permite verificar com maior facilidade toda a malha, vendo como os níveis de segurança estão relacionados entre si. O método é aplicado na matriz energética, de maneira a se garantir seu funcionamento. Conhecer o meio externo é importante, a maioria dos sistemas críticos está ligado diretamente a outros, o efeito em cadeia pode ser catastrófico dependendo do caso. Novamente, mostra como uma visão ampla pode ser utilizada para lidar com o assunto, não se limitando a um processo em uma determinada planta.

Em (REICHENBACH; ALME; ENDRESEN, 2009), identifica-se que é crescente a complexidade e uso de computadores em automação de processos e plantsa industriais, em razão disso é necessário uma abordagem sistemática para identificar as falhas que expõem as partes envolvidas à riscos. É feito um estudo de caso sobre um driver de motor, o método é aplicado para identificar os riscos e falhas no projeto e criar mitigações para melhorar o nível de segurança. A metodologia de árvore de falhas é bastante utilizada no projeto desse tipo de sistema. Apresenta-se ainda o modelo em V, também presente na norma e que é muito utilizado em diversas aplicações de análise de riscos. Além disso, o estudo de caso permite ver passo-a-passo qual a proposta do artigo, facilitando sua repetibilidade.

1.2 Norma ISA 5.1

A seguir descreve-se de maneira sucinta a norma ISA 5.1, utilizada para interpretação da estação de gás natural analisada como estudo de caso. Como descrito em (RODRIGUES, 2010) e na redação original (SEARCH TRIANGLE PARK, 2009), a norma ISA (instrument Society of America) 5.1 estabelece um conjunto de diretrizes para padronizar a representação de instrumentos e sistemas de instrumentação usados para medição e controle em equipamentos e processos industriais. Essa padronização é comumente utilizada nos diagramas PEID (Piping and Instrumentation Diagram) de refinarias de petróleo e, no caso desse projeto, em estações de compressão de gás natural. Ela foi criada de forma que qualquer um ao receber os diagramas seja capaz de identificar o processo e como ele é controlado, como discutido no trabalho de McAvinew (2009), retratado inclusive na conhecida máxima “uma foto vale mais do que mil palavras”, no caso, os diagramas representam a foto. Ela não é utilizada para representar fisicamente os equipamentos, tanto que não leva em consideração as dimensões, escalas ou distâncias, é responsável por apresentar a estrutura e conexões existentes, os principais equipamentos são evidenciados,

como filtros, tanques, bombas, compressores, poços, fornos, torres, válvulas, tubulações, etc. Os principais instrumentos e suas definições estão descritos na Tabela 1.

Tabela 1 – Principais Instrumentos

Instrumento	Definição
Detector	Dispositivo responsável por identificar as variações na variável de processo
Transmissor	Converte o sinal obtido no detector para uma forma que possa ser distribuído
Indicador	Instrumento que interpreta o sinal e apresenta o valor medido
Controlador	Componente que compara o valor medido com um valor de referência e, através de uma lógica de controle implementada, controla a variável
Conversor	Converte o sinal enviado pelo controlador para um formato que possa ser interpretado pelo próximo componente na cadeia
Atuador	Dispositivo que efetivamente atua na planta, modificando de forma direta ou indireta a variável de processo estudada

Fonte: (RODRIGUES, 2010)

Além desses, existem outras definições importantes, como: a faixa de medida, sendo o conjunto de valores que a variável de processo pode varia; o alcance, faixa de valores que o elemento detector consegue identificar; o erro, bastante comum em nomenclatura de sistemas de controle, a diferença entre o valor atual e uma referência definida; a zona morta, sendo um mínimo de variação que usualmente não é detectada pelo sistema instrumentado; a sensibilidade, que é a constante que relaciona o nível de variação do sinal detectado com relação à variação real da variável medida, costuma ser melhor quanto maior for. A norma descreve que o nome de um instrumento deve ser dado por no máximo quatro letras, a primeira identifica a variável medida, a segunda as funcionalidades adicionais, seguidos de um número que o identifica dentro da malha de controle. Durante o trabalho será utilizado o padrão da norma para identificar os componentes, sendo que a nomenclatura será é apresentada na Tabela 2.

Os principais exemplos identificados ao longo do trabalho podem ser, por exemplo: TI – indicador de temperatura; LI – indicador de nível; LAH – alarme de nível alto; TAHH – alarme de temperatura muito alta; entre outros. Um trecho dos PEIDs analisados pode ser visto na Figura 2, com exemplos de instrumentos representados de acordo com a norma.

1.3 HazOp

Para identificar as principais falhas expostas no sistema foi utilizado o estudo de operabilidade e riscos (Hazardous and Operability - HazOp), uma metodologia de análise de riscos desenvolvida principalmente em plantas de processos industriais que, apesar de

Tabela 2 – Significado das letras de identificação

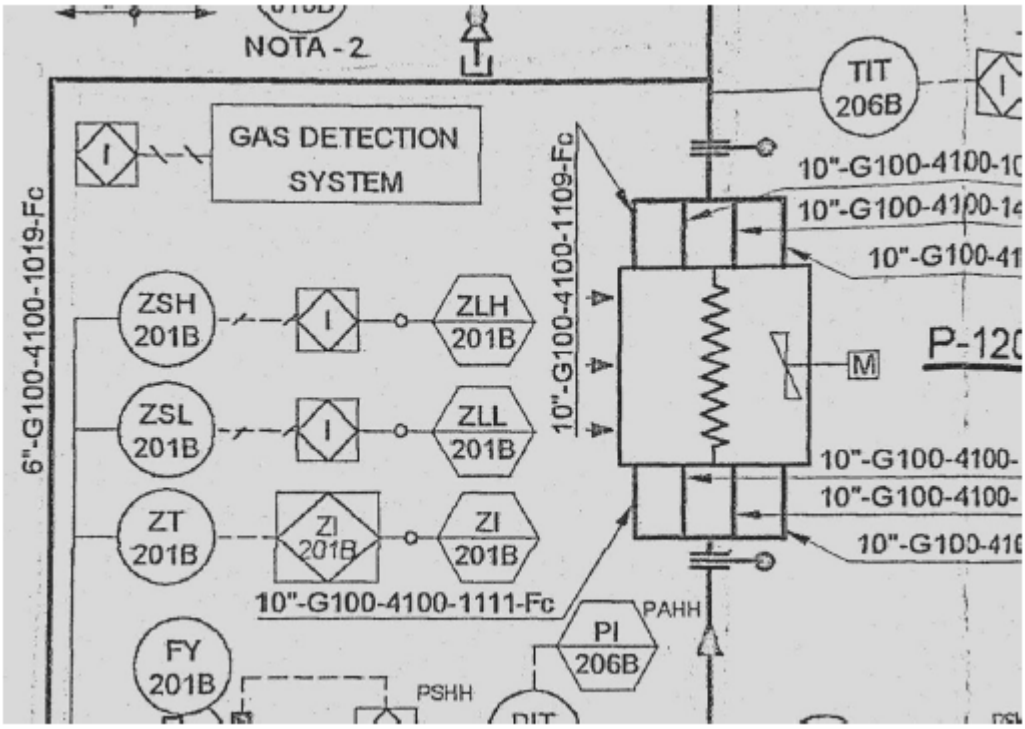
	Variável Medida	**Funcionalidade Adicional**
A	Analisador	Alarme
B	Queimador	Botão de Pressão
C	Condutibilidade Elétrica	
D	Densidade	Diferencial
E	Tensão	
F	Vazão	Relaçao
G	Medida Dimensional	
H	Comando Manual	Entrada Manual
I	Corrente Elétrica	
J	Potência	Varredura
K	Tempo	
L	Nível	
M	Umidade	
N	Vazão Molar	
O	Orifício	
P	Pressão	Percentual
Q	Quantidade	Integração
R	Remoto	
S	Velocidade	Chave de Seguraça
T	Temperatura	
U	Mulitivariável	
V	Vibração	
W	Peso	
X ou Y	Solenóide	
Z	Posição	

Fonte: (RODRIGUES, 2010)

não serem um risco imediato, são intrínsecos e podem comprometer a produtividade e a segurança da planta. Apesar de ser qualitativa, é amplamente utilizada para a caracterização de plantas em diversos estágios de projeto.

Como comentado em (DARAMOLA et al., 2011), a capacidade de identificar ameaças potenciais e problemas na operação e então recomendar os corretos mecanismos de mitigação são vitais para o desenvolvimento de sistemas de controle embarcados em plantas com elevado nível de segurança. A análise Hazard and Operability (HazOp) é muito utilizado para o problema anterior, sendo complexo e com grande atuação humana. O artigo propõe um ferramental para facilitar a identificação de ameaças em estágios prematuros do projeto. O processo HazOp é descrito e suas variações também são apresentadas, uma técnica bastante difundida e que poderá ser também aplicada ao projeto em questão. O ferramental apresentado é modular e tem cada um de seus pontos explicados, facilitando o entendimento e aplicação.

Figura 2 – Captura de imagem de trecho dos PEIDs.



Fonte: (IETEG, 2010)

Descrevendo-se a técnica de maneira sucinta, baseia-se na revisão da planta através de um conjunto de reuniões com um grupo de especialistas, os principais processos são analisados e os principais riscos caracterizados. A experiência de cada um dos participantes colabora pela busca exploratória pelos diversos cenários possíveis, para os quais não pode não haver uma implicação direta advinda do funcionamento da planta, mas de um encadeamento de ações. Dessa forma, ao aplicar-se HazOp, a maioria dos erros viáveis e pertinentes deve ser identificada e, a partir disso, caracterizá-la quanto ao nível de risco e então projetar um meio de impedir que tais falhas ocorram, diminuindo os riscos. Além disso, na ocorrência de uma falha, projetar também a malha que a isole e controle, mitigando-a, levando o sistema para um estado de equilíbrio. Como pode ser visto na Figura 3, a aplicação da metodologia segue de maneira linear. São utilizados alguns termos e conceitos fundamentais, sendo eles:

1 - Nós de estudo: são os pontos do processo, localizados através dos fluxogramas da planta;

2 - Intenção de operação: definido como os parâmetros normais de funcionamento da planta, na ausência de desvios;

3 - Desvios: são os afastamentos das intenções de operação, caracterizados pela constante aplicação de palavras guia;

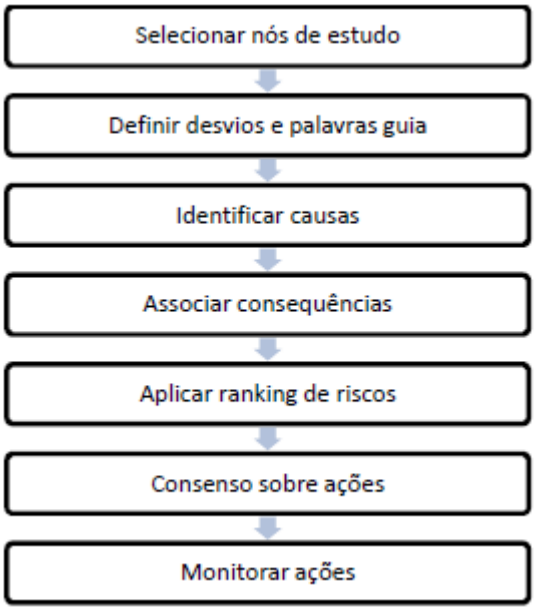
4 - Causas: são os motivos pelos quais os desvios ocorrem, numa relaçõo direta ou indireta de causa e efeito. Elas podem ser, outre outras, por erro no sistema, erro humano e estado não previsto;

5 - Consequências: são os resultados de uma operação com desvio, por exemplo a liberação de gases nocivos para o ambiente;

6 - Parâmetros de processo: são as variáveis físicas do processo, como temperatura e pressão;

7- Palavras-guia: são palavras que servem para classificar os desvios, servindo como guia para o processo de reunião e definição dos riscos. São aplicados sobre os pontos descritos anteriormente, como aos parâmetros de processo, verificando-se, por exemplo, o que aconteceria se ele fosse maior do que o especificado, se ele fosse menor, se o estado canônico não fosse atingido, entre outros.

Figura 3 – O processo HazOp



Uma interessante aplicação foi proposta por (STROM; ANDERSSON; KIVISTO, 2013), onde utiliza ferramentas de UML junto com as técnicas do HazOp para realizar a análise de risco, tornando ainda mais visual e sistêmica a análise, inclusive como forma para documentar e trnasmitir a análise para outros grupos.

1.4 Norma 61508

Para o desenvolvimento do projeto utiliza-se uma metodologia qualitativa presente na norma IEC 61508, muito utilizada em projetos de segurança, sendo inclusive adaptada para aplicações específicas em determinadas indústrias ou empresas. Essa metodologia

não exclui a necessidades de técnicas quantitativas, mas permite caracterizar a planta e identificar quais os pontos mais críticos e que exigem mais atenção da equipe de projetistas.

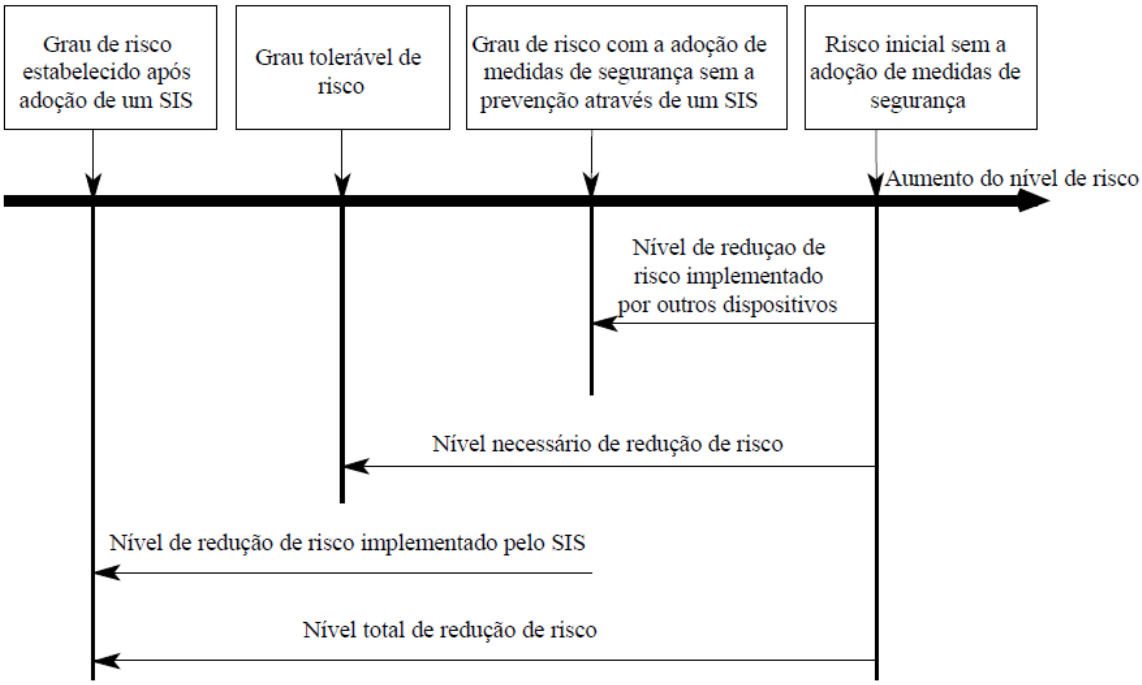
Também discutido em (GALL, 2008), sobre a norma IEC 61508 (Functional Safety of Electrical / Electronic / Programmable Electronic Safety Related System) e a IEC 61511 (Functional Safety for Safety Instrumented Systems for the Process Industry Sector). Tais normas tornaram-se padrão de qualidade em segurança nos mais diversos meios, principalmente em indústrias e sistemas de elevada periculosidade para as partes envolvidas. O principal objetivo é discutir aspectos práticos da norma; benefícios e problemas para os usuários, integradores e fabricantes de equipamentos que atendam à norma. Além disso, discute-se métricas e métodos para correta implementação da norma e vantagens para quem é certificado. Uma releitura direta da norma, essencial para o projeto. Apresenta-se o assunto em pauta em cada uma das partes da referida norma, bem como os principais práticas e lições de cada uma delas. Conhecer os benefícios e malefícios permitirá um maior entendimento e contextualização desde a concepção do sistema de segurança, principalmente por evidenciar a importância do texto para cada uma das partes interessadas, do fabricante ao usuário.

Diversas técnicas podem ser utilizadas para aumentar o nível de segurança de uma planta ou processo: mesmo simples dispositivos mecânicos, gaiolas ou avisos já tornam o ambiente mais seguro. No caso em que mesmo após a implementação de tais iniciativas não é suficiente para baixar os riscos para um grau aceitável, se faz então necessária a utilização de um Sistema Instrumentado de Segurança (SIS), trazendo os riscos de volta para um nível aceitável, como representado na Figura 4.

O artigo apresentado por (SUYAMA, 2003) apresenta um ferramental para análise de integridade e segurança para o controlador de um sistema confiável ou que exija nível de segurança nos padrãos da norma internacional IEC 61508. Esse ferramental permite identificar quais os meios para redução de riscos em sistemas confiáveis e como é possível deixá-los dentro da norma. Como o artigo é bastante fundamentado na norma e ainda aplicado a um controlador, sua aplicabilidade na temática do projeto e direta. Faz-se ligações com a norma e quais as análises quantitativas mais relevantes e como elas devem ser realizadas. Há diversos conceitos e nomenclatura essencial para a correta documentação a ser gerada e validada.

Qualquer projeto deveria passar por análises rigorosas de segurança, sendo que atualmente são poucas as indústrias que não necessitam tomar medidas. Por sua vez, no grupo de sistemas criticos observa-se que existem soluções limitadas, principalmente no que se refere a implementação de sistemas de controle na camada de mitigação. Para a elaboração das análises aqui descritas, usualmente as equipes se utilizam de certos documentos gerados durante a fase de projeto basico e de segurança, entre eles: fluxograma de atividades; relátorio de análise de riscos; matriz de causa e efeito; memorial descritivo

Figura 4 – Redução de Riscos



Fonte: (IEC61508, 2010)

do sistema; e diagramas logicos de intertravamentos. Uma malha de segurança usualmente possui os seguintes elementos:

1 - Iniciadores - linhas de impulso, sensores e transmissores;

2 - Atuadores - elementos finais de atuação e controle;

3 - Executor de Lógica - um CLP, por exemplo;

4 - Sistema de alimentação.

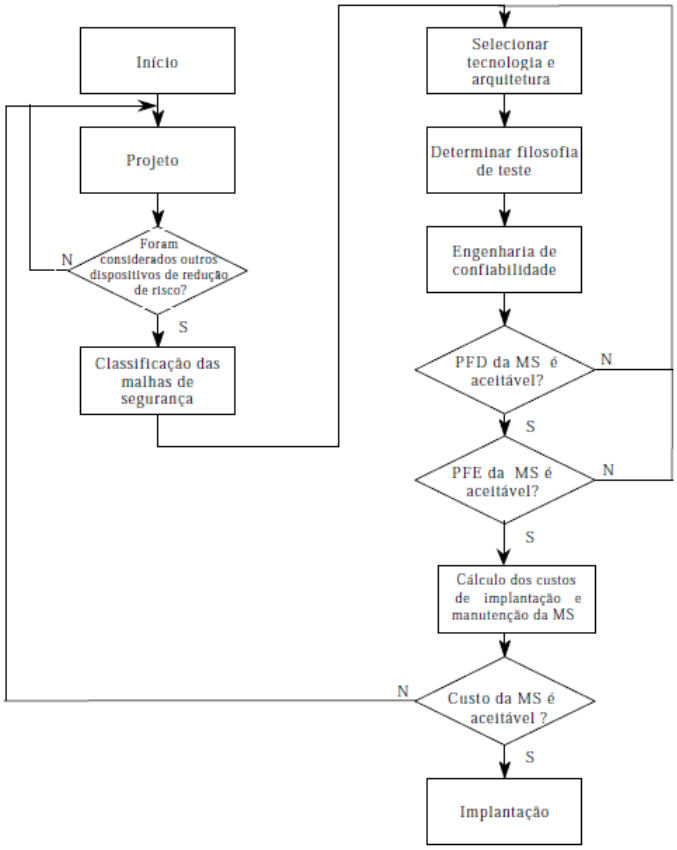
Com o surgimento de normas como a IEC 61508 e a DO-254, padrões de análises qualitativas e quantitativas para sistemas de segurança e mitigação se desenvolveram, ainda mais, tais normas passam por revisão e adaptações constantes. Uma abordagem com Field Programmable Gate Arrays (FPGAs) baseadas em Static Random-Access Memory (SRAM) são plataformas programáveis e de alto padrão tecnológico, sendo bastante utilizados atualmente, como apresentado em (HAYEK; BORCSOK, 2012). O artigo expõe quais as vantagens e desvantagens desse tipo de produto, como utilizá-lo para sistemas de controle de segurança e mitigação e quais as principais métricas para avaliação. Outro sistema de aplicação de prevenção e mitigação de falhas de magnitude diferente da estudada no projeto, contudo, trata-se de mais um estudo de caso, aplicação da norma, caracterização e mitigação de falhas. Utiliza-se novamente o modelo em V, dessa vez para o projeto de um controlador FPGA no lugar de um PLC. Cada um desses controladores

possui vantagem sobre o outro em determinadas aplicações. Será verificado qual a melhor aplicação durante o decorrer do projeto. No caso desse projeto, para o item 3 - Executor de lógica"optou-se por um CLP de segurança.

Cada uma das malhas deve ser classificada de maneira individual e todos os seus elementos devem entrar na analise. No caso de um mesmo conjunto de iniciadores ativar um unico atuador, cada um dos percursos deve ser tratado como uma malha individual. O contrario tambem se aplica, ou seja, quando um unico iniciador ativar diversos atuadores.

Nesse tipo de análise, a classificação mais rigorosa obtida em um dos elementos da malha e entre as malhas deve ser adotada, caso o nivel seja muito elevado, o projeto deve ser refeito para trazer para niveis aceitaveis, como exposto no fluxograma da Figura 5.

Figura 5 – Fluxograma de Projeto de Malha de Segurança



Fonte: (IEC61508, 2010)

A metodologia aplicada separa dois diferentes grupos de classificação:

1 - Para falhas sob demanda - aquelas onde o componente falhou quando nao deveria;

2 - Para falhas espurias - aquelas onde foi enviado um sinal de falha no componente quando ele ainda funcionava corretamente.

Os principais fatores de classificação são os dados que se pode causar, sua frequência e intensidade, tanto nos seres humanos, quanto no meio ambiente e então para os bens, nessa ordem de importância.

No que diz respeito à riscos à segurança pessoal, devem ser levantados os seguintes pontos durante a análise:

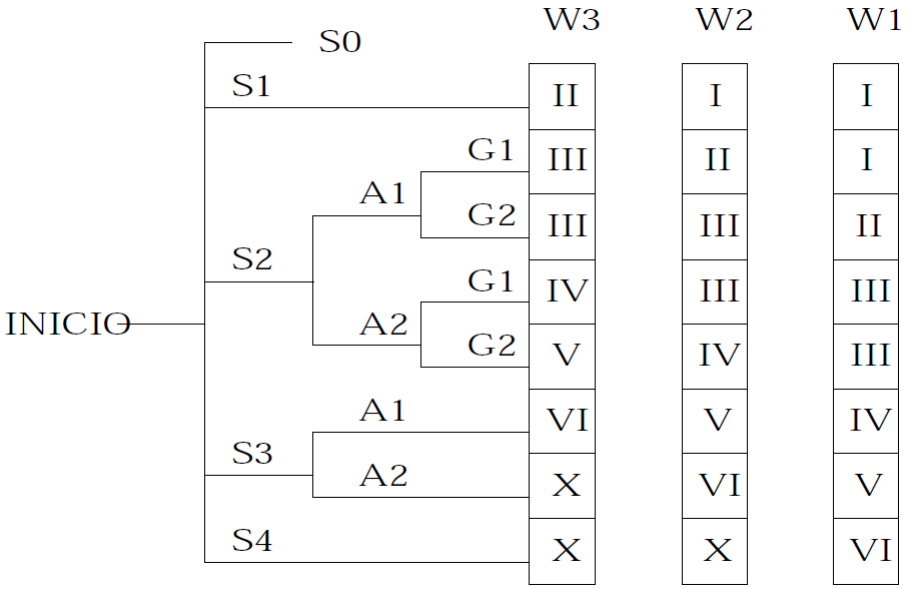
1 - Qual o potencial risco aos seres humanos;

2 - Qual o grau de exposicao que a pessoa esta frente ao risco;

3 - Quão possível é evitar a exposição identificada.

De maneira sucinta, pode-se fazer a classificação como demonstrado na Figura 6.

Figura 6 – Fluxograma de Segurança Pessoal



Fonte: (IEC61508, 2010)

W – frequência de demanda

W0 – não classificada;

W1 – muito baixo (uma a cada 10 100 anos);

W2 – baixo (uma a cada 1 10 anos);

W3 – relativamente alto (mais de um por ano).

S – potencial de risco ao ser humano

S0 – nenhum;

S1 – lesões desprezíveis;

S2 – lesões severas;

S3 – morte de várias pessoas;

S4 – catástrofe.

A – grau de presença humana

A1 – entre rara e frequentemente;

A2 – entre frequente e continuamente.

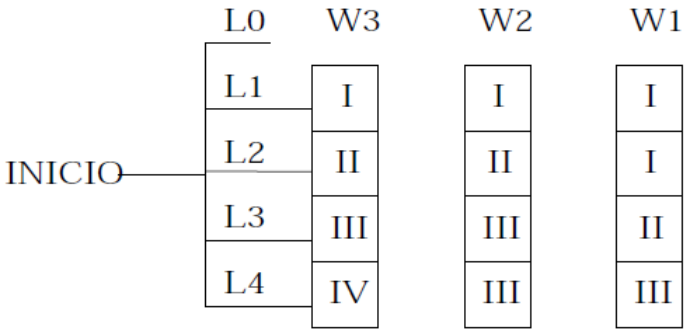
G – possibilidade de evitar exposição

G1 – sobre certas condições;

G2 – dificilmente possível.

Quanto o risco aos bens, Trabalha-se de maneira semelhante à anterior, dessa vez fazendo referência a uma nova tabela de valores e fluxograma da Figura 7.

Figura 7 – Fluxograma de Segurança aos Bens



Fonte: (IEC61508, 2010)

L –potencial de perda de produção e equipamentos

L0 – sem perturbações operacionais ou danos;

L1 – pequenas perturbações operacionais ou danos;

L2 – moderadas perturbações operacionais ou danos;

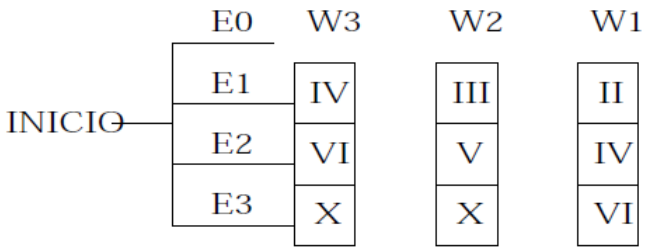
L3 – grandes perturbações operacionais ou danos;

L4 – perda de produção associada a dano em equipamento essencial.

Nesse caso os custos de oportunidade associados ao não funcionamento do equipamento em virtude de falha também é critério para colocação das classificações. De maneira análoga, analisa-se a malha quanto aos possíveis riscos ao meio ambiente, com referência à Figura 8.

E – potencial de danos ao meio ambiente

Figura 8 – Fluxograma de Segurança aos Meio Ambiente



Fonte: (IEC61508, 2010)

E0 – sem liberação ou sem consequências;

E1 – liberação dentro dos limites geográficos da companhia com consequências ambientais conhecidas;

E2 – liberação fora dos limites geográficos da companhia com consequências ambientais conhecidas;

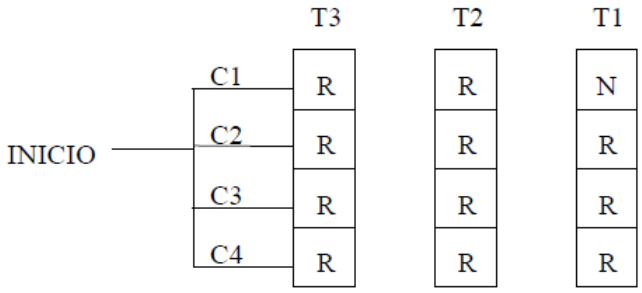
E3 – liberação fora dos limites geográficos da companhia com consequências ambientais desconhecidas.

Deve ser utilizada a maior classificação obtida entre cada um dos modelos utilizados – pessoas, ambiente e bens. Além disso, caso seja atingida uma classificação X, o projeto deve ser refeito para que então se enquadre num valor inferior, o que significa que classe X não possui níveis toleráveis de segurança a um dos critérios utilizados. Por segurança, uma malha entre os níveis II e IV deve possuir um pré-alarme para aviso nas condições em que a planta se aproxima do caso de limite de segurança estabelecido. No estudo de caso apresentado no fim do relatório parcial se apresenta as malhas de segurança de uma planta industrial e sua respectiva classificação, bem como a existência dos pré-alarmes comentados anteriormente. Podem acontecer ainda as chamadas falhas espúrias, um equipamento pode enviar um sinal de falha sem que tenha realmente falhado, pode ser um estado transitório e que pode passar naturalmente. Para analisar a necessidade de implementação de um sistema de controle de prevenção de falhas espúrias analisa-se o custo para sua implementação e qual o custo intrínseco a sua leitura, com o peso de cada um desses parâmetros é possível definir se é viável ou não.

$$CFA > \frac{CTFA}{TFA} \tag{1.1}$$

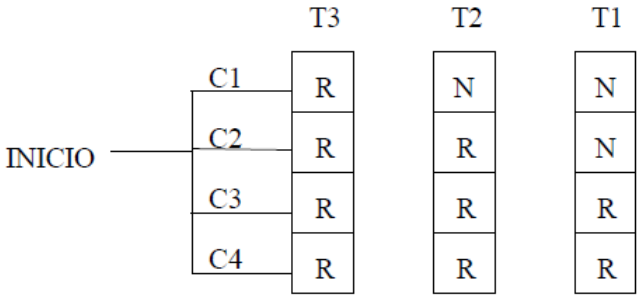
Onde, CFA = custo de falha aparente (espúria), CTFA = custo de implementação de tolerância a falha aparente e TFA = taxa de falha aparente. A **??**, **??** e **??** representam a tomada de decisão com base nos critérios expostos.

Figura 9 – CTFA < 1.000 USD



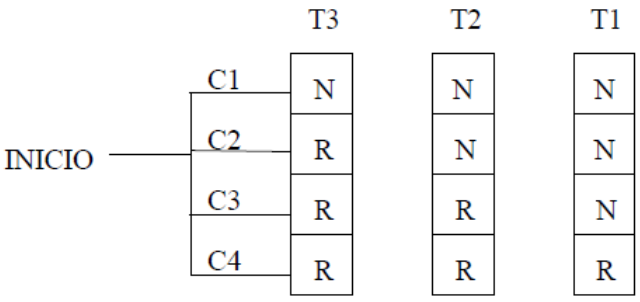
Fonte: (IEC61508, 2010)

Figura 10 – CTFA < 1.000 USD



Fonte: (IEC61508, 2010)

Figura 11 – CTFA < 1.000 USD



Fonte: (IEC61508, 2010)

R – recomendada a implementação de tolerância a falhas espúrias

N – não recomendada a implementação de tolerância a falhas espúrias

C – potencial de perda de produção

C0 – custo < 100.000 USD;

C1 – 100.000 < custo < 1.000.000 USD;

C2 – 1.000.000 < custo < 10.000.000 USD;

C3 – custo > 10.000.000 USD.

T – frequência de falha aparente

T1 – muito baixa (uma em cada 10 100 anos);

T2 – baixa (uma em cada 1 10 anos);

T3 – relativamente alta (mais de uma por ano).

Pode se relacionar as aplicações dessa metodologia com os níveis SIL (Safety Integrity Level) e a AK (DIN V19250), mostrando a correlação entre as diferentes normas, como visto na Tabela 3. Sendo que o PFD (probabilidade de falha sobdemanda) indica a frequência que uma falha pode ocorrer nesse sistema.

Tabela 3 – Significado das letras de identificação

Classe de MS	PFD	SIL	Classe AK
I	$\geq 10^{-1}$	-	-
II	$\geq 10^{-1}$	-	1
III	$\geq 10^{-2}$ - $<10^{-1}$	1	2-3
IV	$\geq 10^{-3}$ - $<10^{-2}$	2	4
V	$\geq 10^{-4}$ - $<10^{-3}$	3	5
VI	$\geq 10^{-4}$ - $<10^{-3}$	3	6
X	$\geq 10^{-5}$ - $<10^{-4}$	4	7-8

Fonte: (IEC61508, 2010)

2 Materiais e Métodos

2.1 Estudo de Caso

2.1.1 Descrição do Problema

O foco do estudo será aplicado sobre uma estação de compressão de gás natural, como por exemplo a da Figura 12. Os chamados gasodutos fazem o transporte de grande quantidade de gases de uma localização para outra e, devido às perdas de carga ao longo da tubulação, é necessária a instalação de estações de compressão intermediária, onde o gás é novamente comprimido, continuando seu trajeto até o destino final, onde poderá ser tratado e consumido.

Figura 12 – Estação de Compressão de Gás Natural

Fonte: http://www.norteng.com.br/

A planta em questão faz o transporte de cerca de 21 milhões de m3 de gás natural diariamente, sendo que possui 4 turbocompressores da Mitsubishi com capacidade efetiva de transporte de 7 milhões de m3 de gás natural diariamente, um dos turbocompressores é mantido desligado em regime de revezamento, para diminuir o desgaste e permitir a manutenção do mesmo. O gás natural é um hidrocarboneto leve e inflamável, toda a planta está sujeita à graves riscos aos operadores, meio ambiente e máquina, mesmo na não ocorrência de uma explosão ou situação semelhante. Além disso, a planta possui já instalados um sistema de controle básico de todos os seus processos e um sistema de controle de segurança, para evitar que as falhas ocorram. Contudo, ainda não possui a terceira camada e que permitirá aumentar ainda mais a robustez da mesma, que é a instalação

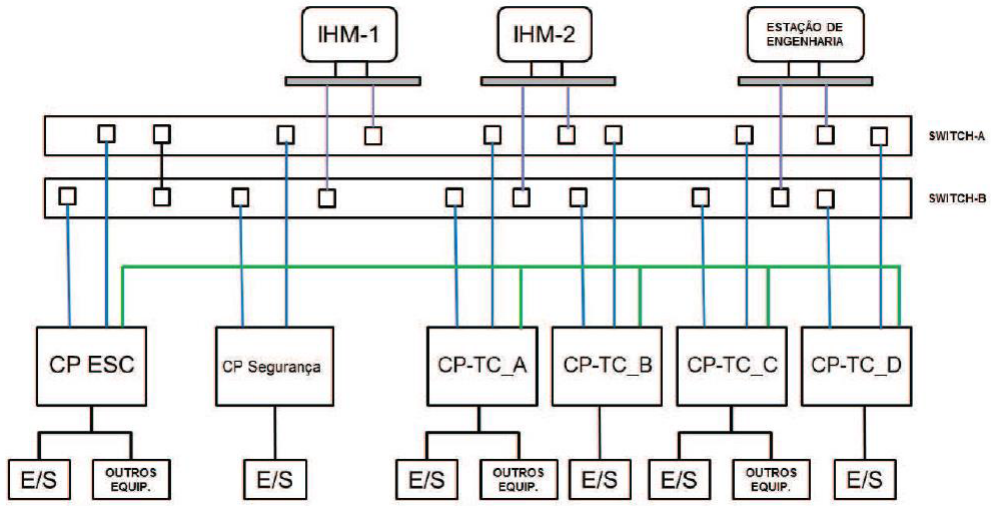
de um sistema de controle de mitigação de falhas. Como essas plantas normalmente se instalam em regiões isoladas, a do exemplo possui uma arquitetura que permite que ela se auto-sustente a partir do próprio gás natural em que trabalha sua compressão. De maneira sucinta, podemos descrever o processo como: sucção inicial do gás natural do gasoduto; parte desse gás passa pelo compressor, onde é comprimido e segue então viagem; outra parte desse gás é utilizada para alimentar geradores movidos a gás, como o exemplo da Figura 13, produzindo energia elétrica para alimentar o restante dos componentes eletrônicos da planta; outra parte do gás é utilizada como combustível para a turbina responsável pela atuação do compressor.

Figura 13 – Exemplo de Gerador a Gás

Fonte: http://image.made-in-china.com/

As diversas linhas para onde o gás caminha diferente do seu destino via gasoduto são as chamadas utilidades, sendo elas: turbina do turbocompressor e geradores de energia elétrica. Quando o gás chega a planta ele é primeiramente filtrado, após essa filtragem que ele se divide nas rotas, sendo que para ir ao gerador ele passa por nova filtragem e para ir ao compressor ele tem sua temperatura alterada para aumentar a eficiência do processo, técnica batante utilizada na engenharia térmica. A arquitetura do controle está segmentada da maneira representada na Figura 14. O controle básico é implementado num CLP, aqui denominado ESC (Estação de Supervisão e Controle). O controle de segurança é implementado num outro CLP de segurança de tripla redundância da fabricante Rockwell. Além disso, cada um dos turbocompressores possui seu próprio CLP, que comanda suas operações básicas. Cada um dos terminais está ligado a entradas e saídas, tanto digitais quanto analógicas, para transmissão de sinais, leitura de sensores, energização de atuadores, etc. Todo o sistema foi projetado para ser seguro, sendo que também existe redundância na transmissão de dados, com duas diferentes vias.

Figura 14 – Topologia do Sistema de Segurança Presente

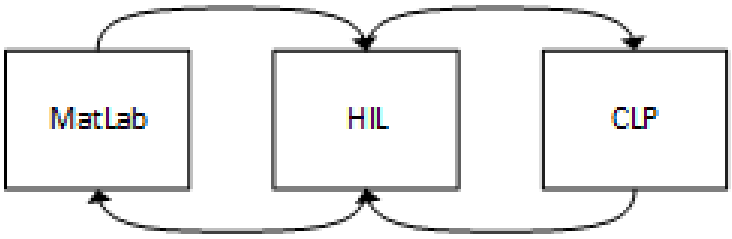


Fonte: (SQUILLANTE, 2011)

2.1.2 Abordagem Utilizada

A abordagem atual tem sua topologia representada na Figura 15.

Figura 15 – Topologia do Sistema de Mitigação Proposto



Será implementado em Matlab um modelo da planta como um sistema a eventos discretos, e para sua implantação será utilizado o algoritmo PFS/MFG apresentado em (MIYAGI, 2007). Esse modelo será enxergado pelo CLP como uma planta real, com todas as relações entre as entradas e saídas definidas.

Duas placas serão utilizadas para a implementação do Hardware in the Loop, apresentadas no Apêndice E - placa de aquisição de sinais que se comunica com o computador e o Matlab, e no Apêndice D - placa de interface que converte os sinais lógicos de nível TTL para os de lógica de relés utilizados pelos CLPs.

No nível do CLP serão implementadas as aplicações de mitigação, com um programa rodando em modo de segurança que permitirá observar os estados da planta e levá-lo para um estado seguro na ocorrência de um problema.

Nas próximas seções serão descritos como foram desenvolvidas e aplicadas as metodologias para a implementação proposta.

2.2 Metodologia PFS/MFG

A metodologia PFS/MFG foi apresentada em (MIYAGI, 2007). A partir de uma sequência de passos e da utilização de ferramentas de modelagem de sistemas a eventos discretos é possível modelar a planta. Pode-se dividir o procedimento em 5 etapas:

1 - Identificação dos principais fluxos de atividades

2 - Detalhamento dos fluxos

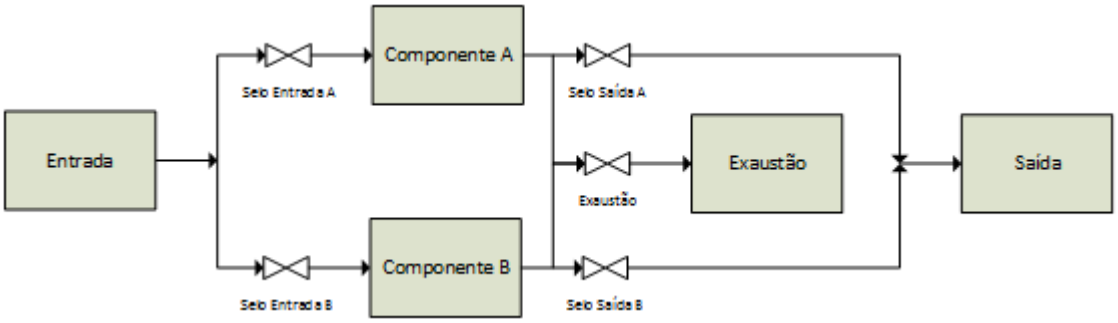
3 - Detalhamento das atividades

4 - Introdução dos elementos de controle de recursos

5 - Indicação dos sinais de controle com a planta

Primeiramente é necessário identificar a topologia do sistema com que se pretende trabalhar. A unidade elementar que se pretende atuar no trabalho será a nível de componente. O processo industrial apresentado pode ser representado como a passagem do gás de um componente para o outro, sendo eles responsáveis por levá-lo ao estado desejado. A topologia proposta pode ser vista na Figura 16. Existem dois de cada componente em cada grupo e o processo é a sequência do gás por dentro desses componentes.

Figura 16 – Topologia de um Componente



As válvulas de selo de entrada e de saída devem possuir interlock lógico ou mecânico, de forma que se uma está aberta a outra também está, o mesmo vale se uma for fechada. Cada um dos circuitos A e B é independente, se houver gás na entrada e o componente não estiver selado, o gás passa pelo componente. Se a exaustão estiver ligada, o gás não vai para a saida e sai para a exaustão, onde será expelido para a atmosfera. Em operação normalizada, o gás segue normalmente para a saida após passar pelo componente.

Os métodos utilizados são melhores descritos no Apêndice C

2.3 Metodologia para Controle de Sistemas Instrumentados de Segurança

A metologia para desenvolvimento do controle de um sistema instrumentado de segurança foi proposto em (SQUILLANTE et al., 2011) e pode ser dividida nas seguintes etapas:

1 - Modelagem do Diagnóstico de Falhas;

2 - Modelagem do Tratamento de Falhas;

3 - Análise dos Modelos;

4 - Geração dos Algoritmos de Controle.

Ao seguir cada uma dessas etapas, ao final teremos um sistema que a partir da observação de determinados sensores consegue determinar qual a situação de falha presente e então atuar em cima da mesma, isso tendo como base o Hazop efetuado anteriormente e que gerou a identificação de 12 diferentes malhas, apresentadas no Apêndice C.

A forma com que cada uma das malhas será tratada em diagnóstico é muito semelhante, portanto será apresentada a metodologia somente para a SIF-01. Do resultado final obtido, a única diferença está na origem de cada um dos sinais utilizados durante a etapa de diagnóstico, coletado através dos sensores, e o atuador que está sendo ativado para trazer a malha para um estado seguro. Os sensores responsáveis por iniciar a malha e os atuadores responsáveis por ativar cada uma das 12 malhas de segurança podem ser vistos no Apêndice C.

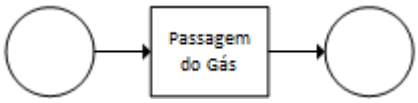
3 Resultados

3.1 Metodologia PFS/MFG

3.1.1 Identificação dos principais fluxos de atividades

Observando a topologia proposta, temos então basicamente o processo de passagem do gás pelo componente, como representado na Figura 17. O segundo caso será tratado como situação emergencial dentro do primeiro procedimento, o que será o caso.

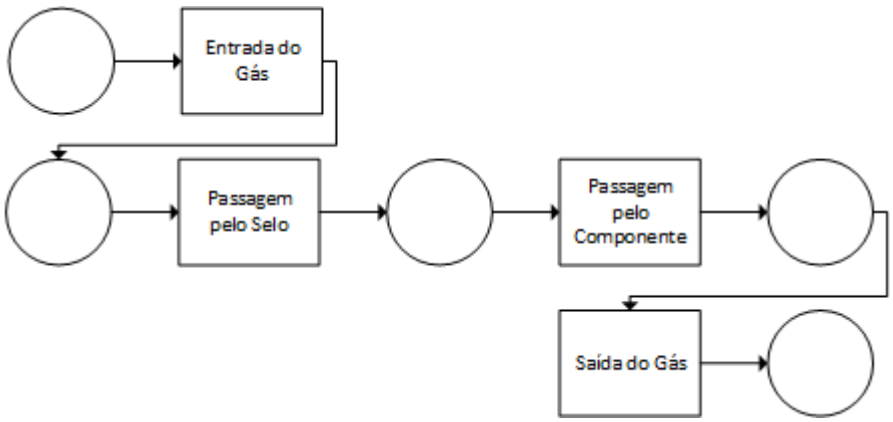
Figura 17 – Fluxo de Atividade Identificado



3.1.2 Detalhamento dos Fluxos

Podemos então expandir a passagem do gás através do detalhamento do fluxo, onde deve-se destacar cada uma das posições pelas quais o gás trafega durante o processo. Como representado na Figura 18, o gás deve estar presente na entrada da planta, em seguida ser capaz de atravessar o selo, ser processado dentro do componente e então novamente sair do grupo, permitindo então que outra etapa do processo seja iniciada.

Figura 18 – Detalhamento do fluxo



3.1.3 Detalhamento das Atividades

Explorando agora as necessidades e especificações de cada um dos blocos, temos a abertura da entrada do gás na Figura 19 e do passagem pelo selo na Figura 20.

No primeiro caso (Figura 19), é importante verificar se há gás na entrada, nessas condições:

1 - caso positivo ele pode continuar, dependendo da situação do selo;

2 - caso negativo o gas não passa, e a situação dentro do componente não é alterada nesse ciclo.

No segundo caso (Figura 20), novas condições são verificads onde:

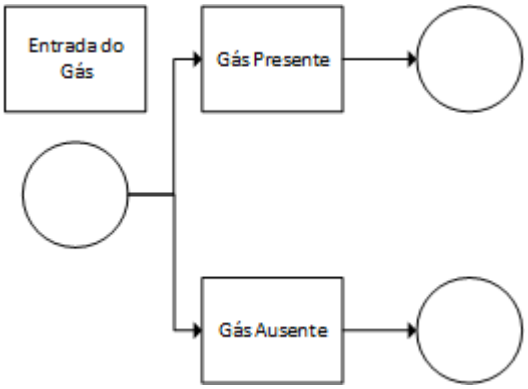
1 - caso não haja gás e se o componente não estiver selado, o gás que havia no componente se acaba e não é reposto;

2 - se não estiver selado ele é reposto e o processo continua normalmente.

Na etapa 5 define-se melhor como o controle vai atuar em cima de cada um desses comandos e levar a planta para um estado degenerado (um estado seguro).

As outras atividades são muito simples e não precisam ser quebradas em novos processos, tratando-se basicamente de um novo local para a passagem do gás.

Figura 19 – Detalhamento da atividade - entrada do gás



3.1.4 Introdução dos Elementos de Controle de Recursos

Pelo componente isolado ser muito simples, não há o compartilhamento de recursos e essa etapa não foi implantada.

3.1.5 Indicação dos Sinais de Controle com a Planta

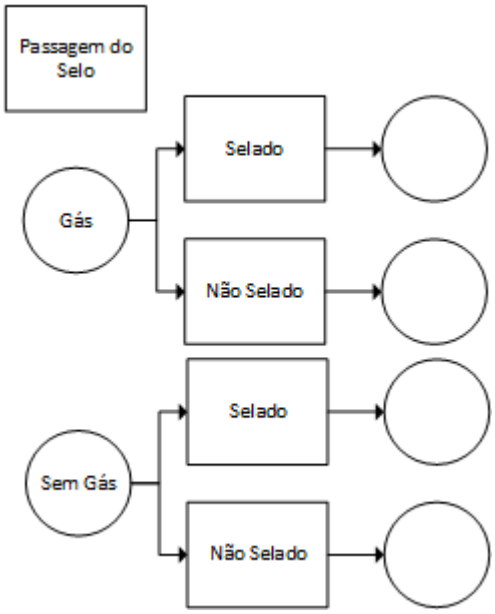
Os sinais de controle serão basicamente 4 para cada um dos grupos de componentes:

1 - Envia sinal para selar componente A

2 - Envia sinal para selar componente B

3 - Envia sinal para exaurir gás do componente A

Figura 20 – Detalhamento da atividade - passagem do selo



4 - Envia sinal para exaurir gás do componente B

O sistema de controle deve ser responsável por gerir as situações de emergência identificadas pelos sensores de forma a levar a planta para um estado seguro. Atuando em cima dessas operações básicas é possivel controlar o fluxo do gás dentro do processo, prendendo-o em determinado ponto e somente liberando quando não houverem riscos, como por exemplo, focos de incêndio.

3.1.6 Implementação da Planta em Ambiente Matlab

Como ferramenta de aplicação, optou-se por implementar o sistema como uma máquina de estados dentro do ambiente SimuLink do Matlab. A máquina de estados funciona com a mesma lógica de transições e posições utilizada na metolodiga. O resultado final pode ser visto no Apêndice A.

Essa máquina sintetiza o funcionamento de um componente, cuja descrição do processo então é a seguinte:

S0 - Verifica se há gás na entrada do componente. Caso negativo segue para S1, caso positivo segue para S4;

S1 - Verifica situação do selo e da exaustão. Caso selado e não exaurido, segue para S3, caso não selado ou exaurido, segue para S2;

S2 - Indica que não há gás no comonente, indica que não há gás sendo enviado para o próximo conjunto. Segue para S8;

S3 - Segue para S8;

S4 - Indica que há gás no componente e verifica situação do selo e da exaustão. Caso selado e não exaurido, segue para S5, caso não selado e não exaurido, segue para S6, caso exaurido, segue para S7;

S5 - Segue para S8;

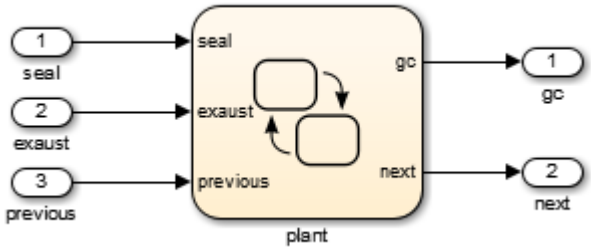
S6 - Indica que há gás sendo enviado para o próximo conjunto. Segue para S8;

S7 - Indica que não há gás no componente e que não há gas sendo enviado para o próximo conjunto.Segue para S8;

S8 - Retorna para o estado inicial.

Cada uma dessas máquinas de estado têm como entrada o indicador de gás presente na entrada e os sinais de controle para selamento e exaustão, como indicados na Figura 21. Esse encapsulamento permite repetbilidade e então essa estrutura poderá ser repetida para todos os conjuntos de componentes presentes na planta.

Figura 21 – Encapsulamento da máquina de estados do componente

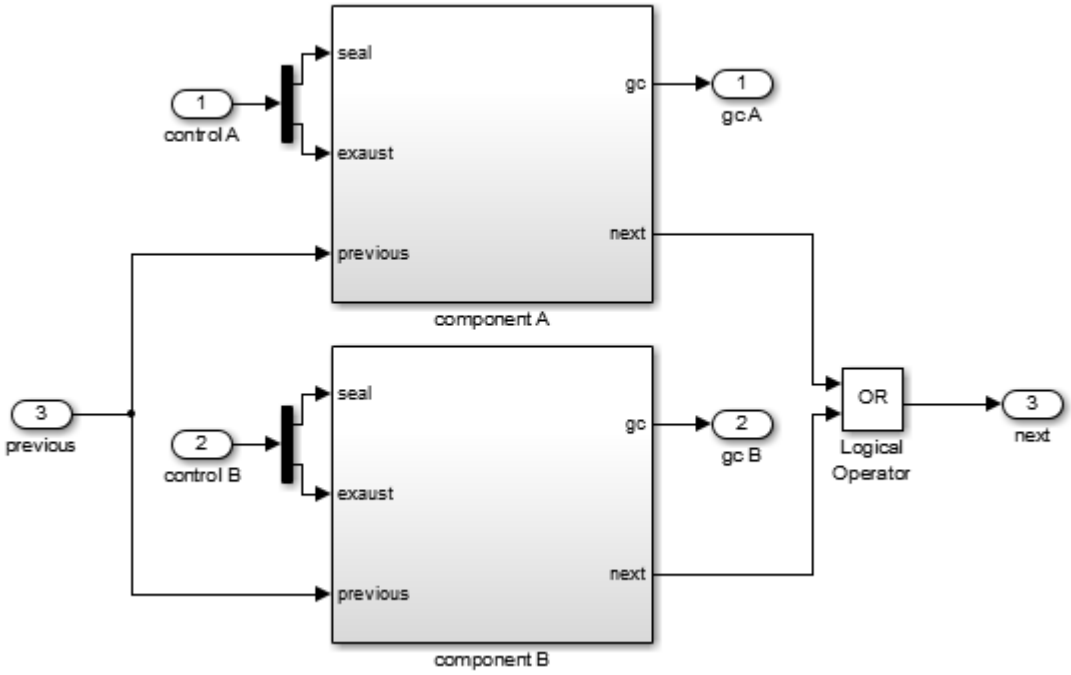


Relembrando a topologia proposta na Figura 16, temos um par de componentes para cada um dos conjuntos, a entrada de gás para os dois é comum, assim como a saída. A nova camada de encapsulamento faz esse controle, sendo que agora eles são alimentados com o indicador de gás na entrada do conjunto e recebem também os sinais de controle individuais para os componentes A e B, como representado na Figura 22, sendo que as saídas desse conjunto são os indicadores de gás dentro do componente A, dentro do componente B na saída do conjunto. Esses indicadores de posição permitirão que se observe o caminho que o gás está fazendo dentro do processo. Note que apesar de o processo ser contínuo, está aqui sendo representado como uma sequência de estados discretos de forma que seja possível a aplicação das técnicas e métodos sem que se perca funcionalidade.

Com esse nível de encapsulamento foi possível então concatenar os grupos de componentes da mesma forma como existe na planta de compressão de gás natural. O gás da entrada a planta é único, ele então é divido por diversas linhas, cada uma delas responsável pelo funcionamento de determinada parte do processo.

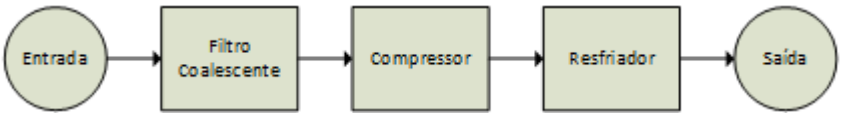
O principal processo e razão de ser da planta é a compressão do gás natural do

Figura 22 – Encapsulamento de um dois componentes, formando um conjunto



duto, sendo então que a maior parte do gás é enviado para os compressores e depois levado para a saída, essa é a chamda linha de compressão representanda na Figura 23.

Figura 23 – Linha de Compressão do Gás



O processo secundário é o de geração de energia para o funcionamento da estação, sendo que a energia gerada é utilizada tanto para girar o compressor quanto para alimentar os controladores, os sensores, a iluminação e os demais componentes, sendo a fonte primária de energia para a planta. Como a turbina é de dois estágios, parte do gás entra na partida e parte da segunda etapa, com a linha representada na Figura 24.

Por fim, para máximo aproveitamento energético e aumentar a eficiência do processo, o gás é queimado e usado no trocador de calor, levando o gás para um estado adequado para entrada nas turbinas e compressores, representado na Figura 25;

O resultado desse sequenciamento e junção de todos os componentes em uma única planta foi implementado no Matlab e pode ser visto no Apêndice B, cuja entrada principal é a presença de gás no duto de coleta, as saídas principais são presença de gás no duto de expedição e pela chaminé de exaustão, por fim, o restante das entradas e saídas são sinais para se verificar a presença de gás em cada um dos componentes e de controle, selando e

Figura 24 – Linha de Energização da Planta

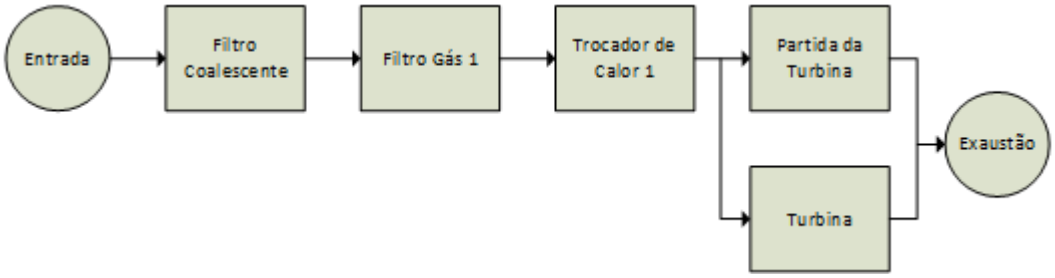
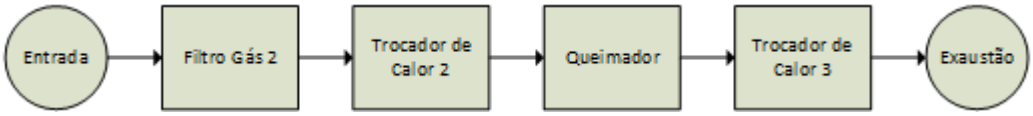


Figura 25 – Linha de Aquecimento do Gás



exaurindo o gás conforme necessidade do algoritmo de controle.

3.2 Metodologia para Controle de Sistemas Instrumentados de Segurança

3.2.1 Modelagem do Diagnóstico de Falhas

Primeiramente, monta-se a matriz de causa-efeito, que relacionam as possíveis causas com a falha efetiva do componente observado na malha. Cada uma das malhas possui 6 sensores, 3 para cada um dos componentes que funcionam por esquema de votação 2oo3, no qual é necessário que pelo menos 2 dos 3 sensores estejam indicando um determinado estado para que seja enviado o sinal para a planta, como indicado na tabela verdade da Tabela 4. Note também que há uma sinal que indica que o sistema SIS já fez sua atuação, se os sinais dos sensores ainda indicarem a situação de gatilho, o sistema de segurança deve ter falhado e o sistema de mitigação deve então atuar.

Condensados os resultados da votação em uma única coluna do sensor A, temos então a tabela de causa e efeito na Tabela 5. Nela já há a discrição entre uma falha somente no componente A ou no componente B, onde ele é selado para controle, se há falha nos dois componentes, o gás deve ser exaurido e toda a planta selada, com o processo interrompido.

A próxima etapa da metodologia ocorre com a construção de uma Rede Bayesiana, que indica as relações de causa-efeito com base na matriz anteriormente construída. O resultado pode ser visto na Figura 26, onde estão relacionados desde o início das causas e o efeito causado e notado. Esse sistema de diagnóstica servirá então de indicador para a falha, servindo de gatilho para os processos do sistema de controle e atuação. Note nas

Tabela 4 – Tabela Verdade de Votação dos Sensores

SA	SA 1	SA 2	SA 3
0	0	0	0
0	0	0	1
0	0	1	0
1	0	1	1
0	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

Tabela 5 – Matriz de Causa-Efeito

Casos	Falha - Exaure	Falha - Sela A	Falha - Sela B	Sinal SIS	SA	SB
1	0	0	0	0	0	0
2	0	0	0	0	0	1
3	0	0	0	0	1	0
4	0	0	0	0	1	1
5	0	0	0	1	0	0
6	1	0	1	1	0	1
7	1	1	0	1	1	0
8	1	1	1	1	1	1

tabelas das malhas de segurança que os iniciadores variam de tipo mas tem nomenclatura parecida, sendo eles então os indicadores do sistema de votação utilizado e que entram na matriz de causa-efeito. Bem como os atuadores, sendo sempre apresentados a válvula responsável por selar os componentes (iniciadas com o indicador 0), e a válvula responsável por exaurir os componentes (iniciados com o indicador 1).

Por fim, para completar a etapa, a rede bayesiana é transformada em uma rede de petri através do seguinte procedimento:

1 - Inserção das transições para cada arco de relacionamento entre as variáveis da Rede Bayesiana;

2 - Inversão no sentido dos arcos, uma vez que o raciocinio diagnóstico é uma relação dos efeitos para as causas.

O resultado pode ser observado, enfim, na Figura 27.

Figura 26 – Redes Bayesianas

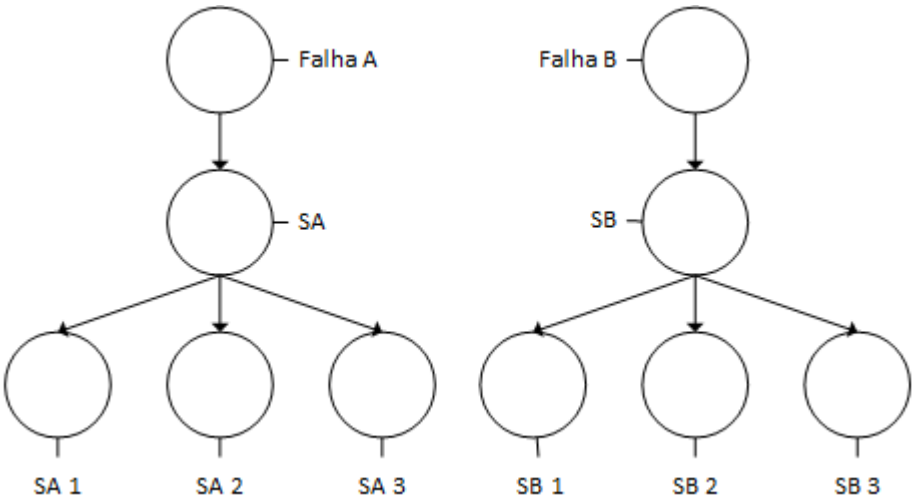
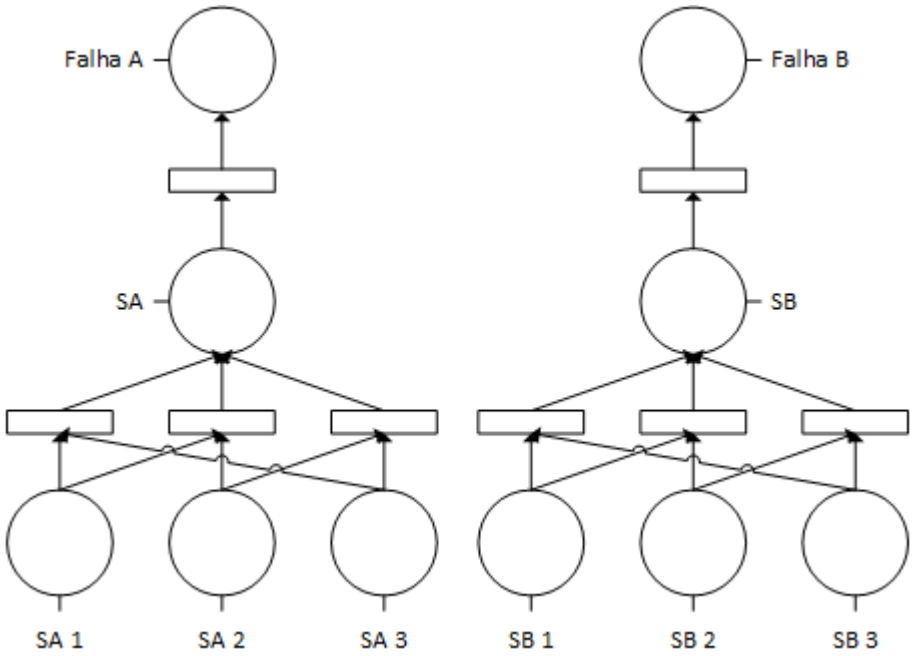


Figura 27 – Passagem da Rede Bayesiana para a Rede de Petri



3.2.2 Modelagem do Tratamento de Falhas

Como levantado anteriormente, temos definidas as malhas no Apêndice C. A próxima etapa para o tratamento é de criação da rede de petri que indica a atuação sobre o sistema com base no resultado do diagnóstico.

Os atuadores terão efeitos muito semelhantes com base nos resultados dos diagnósticos, cada um agindo sobre determinadas condições e de seus respectivos sensores. Portanto, para cada SIF, podemos descrever a atuação como:

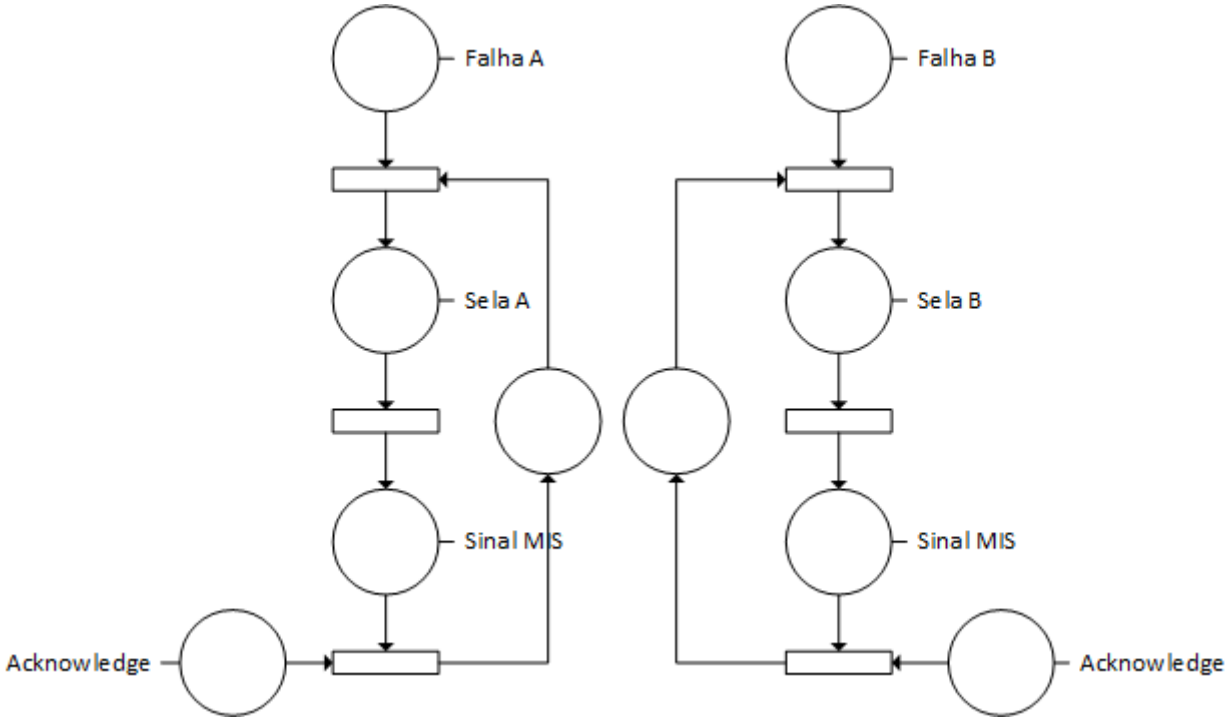
1 - Caso idenfique-se falha em A ou em B, selar o componente respectivo, permitindo que o processo continue e o equipamento seja devidamente condicionado;

2 - Caso identifique-se falha em A e em B, deve-se selar todos os componentes da planta, encerrar completamente o processo e, depois de verificado que a situação é segura, exaurir todo o gás da planta;

3 - Um sensor de incêndio geral vai impedir que qualquer gás seja exaurido até que o foco tenha sido eliminado de maneira automática pela planta;

Traduzindo o descrito acima para uma Rede de Petri, temos o resultado na Figura 28.

Figura 28 – Tratamento da Falha a partir da Coordenação e Diagnóstico



3.2.3 Análise dos Modelos

Os modelos devem ser analisados para que se identifique a não possibilidade de atingir estados com deadlock ou condições espúrias que impeçam todo o processo de mitigação de funcionar. Os sistemas são simples e na maioria dos casos apresentam grande linearidade e são de fácil caracterização.

3.2.4 Geração dos Algoritmos de Controle e Resultado da Metodologia

No que fora discutido em (GERGELY et al., 2008), os Controladores Lógico Programáveis (PLCs) são muito utilizados em processos de automação industrial que em

essência exigem elevada confiabilidade e segurança. Entretando, a prevenção de riscos costuma tornar o sistema de controle mais caro conforme crescem as exigências; o equilíbrio entre riscos e custos cabe ao projetista responsável pelo conjunto. O artigo discute que uma das maneiras de conciliar esses dois parâmetros é através da mitigação dos riscos, assumindo que certas falhas ocorram mais que serão tratadas, dentro de limites aceitáveis. Apresenta-se um ferramental para auxiliar no projeto e verificação de sistemas de segurança que utilizem PLCs. O projeto envolve o desenvolvimento de um sistema de controle de mitigação de falhas, será implementado através de um controlador lógico programável (PLC) e o artigo apresenta uma metologia para deteção e projeto de um sistema do tipo. Além disso, o estudo de caso realizado é semelhante ao que será utilizado, com um sistema de aquecimento de água no lugar de uma planta de compressão de gás natural. Além disso, o projeto também considera detalhes da norma, agregando mais ao conhecimento necessário e padronizando frente aos requisitos. A utilização da metodologia apresentada em (SQUILLANTE et al., 2011) permite que a maioria dos problemas já sejam endereçados de maneira sistemática.

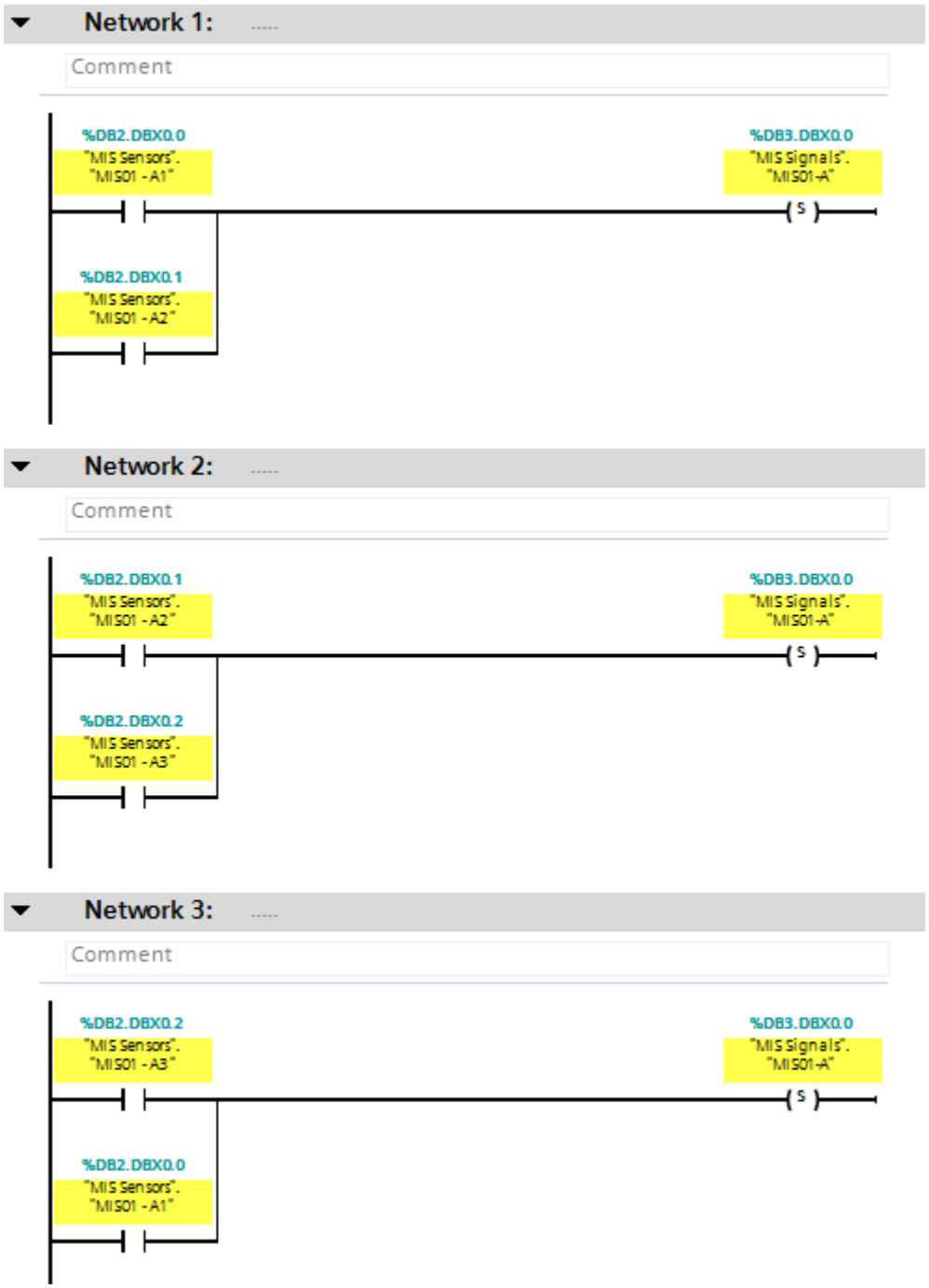
Novamente pensando na repetibilidade e facilidade compreensão, os blocos de programação foram encapsulados e reutilizados para cada uma das malhas. O programa foi basicamente dividido em três diferentes blocos para cada uma das plantas:

1 - Bloco de Coordenação (Figura 29) - Esse bloco vai ler o estado dos sensores e e indicar se foi identificada a falha, servindo então como o diagnóstico da falha, implementado o codigo apresentado na Figura 27. Para que funcione, o CLP de segurança deve receber como sinais de entrada o valor de saída da votação dos três sensores - note que qualquer combinação de dois sensores já é suficiente para mandar o sinal de falha para 1, indicando problemas. É necessário então que o CLP tenha no mínimo 4 entradas para cada uma das malhas de segurança, no estudo de caso apresentado seriam 48 diferentes entradas.

2 - Bloco de Atuação (Figura 30) - Esse bloco vai agir quando o erro for identificado e levantado pelo Bloco de Coordenação, sendo então responsável por implementar o circuito apresentado na Figura 28. Quando o erro for detectado, o MIS deve levar a planta para um estado seguro, selando o gás no interior do componente e travando seu fluxo, além de enviar um sinal de falha que iria para uma interface homem-máquina, notificando o operador. No caso apresentado, esse sinal segue pela porta de saída do CLP, atravessa a interface do Hardware in the Loop e é recebida pela planta simulada em Matlab, que vai agir de acordo com os sinais de controle enviados.

3 - Bloco de Operação (Figura 31) - Esse bloco deve receber o sinal de reconhecimento do operador da planta, indicando que o problema já foi endereçado e está resolvido. Nesse momento o ciclo do CLP de segurança vai novamente rodar e ele vai verificar os sensores, caso o estado ainda seja de perigo ele volta a agir e aguardar novo sinal de reconhecimento do operador. Em suma, ele não vai permitir que o processo continue até

Figura 29 – Nível de Coordenação e Diagnóstico



que de fato a situação tenha sido normalizada.

Foram então criados 12 diferentes conjuntos desses três blocos, um para cada malha de mitigação, ou MIS (Mitigation Instrumented System). Para cada um dos blocos o sensor é diferente, podendo ser de pressão, de nível, de temperatura, o importante é que esteja calibrado para indicar que o processo atingiu níveis fora de sua zona de operação segura. Note que primeiramente o sistema de segurança instalado deve ter agido e falhado em levar a planta para o estado seguro, esses sensores servirão para indicar que chegou a hora

Figura 30 – Nível de Atuação

do sistema de mitigação entre e diminua os estragos que a falha possa trazer.

Os blocos foram encapsulados como na Figura 32, tornando-se então três grandes conjuntos, um para cada tipo de bloco, sendo chamados na função de segurança principal apresentada na Figura 33.

O bloco de programa safety é uma subrotina que funciona através de uma interrupção cíclica de tempo real, fazendo ciclos de varredura num tempo pre-determinado e configurado pelo projetista do controle. No caso, estamos utilizando 100ms, projetos com tempo de resposta mais rápidos podem exigir um tempo de reação mais elevados.

Figura 31 – Nível de Operação

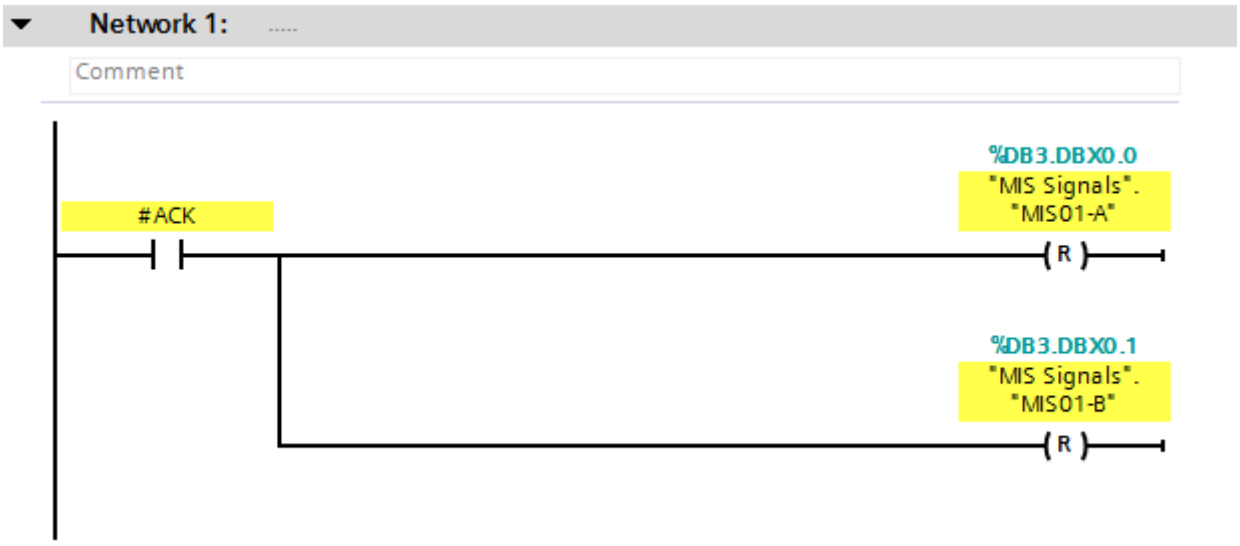


Figura 32 – Grande conjunto, encapsula 12 blocos menores

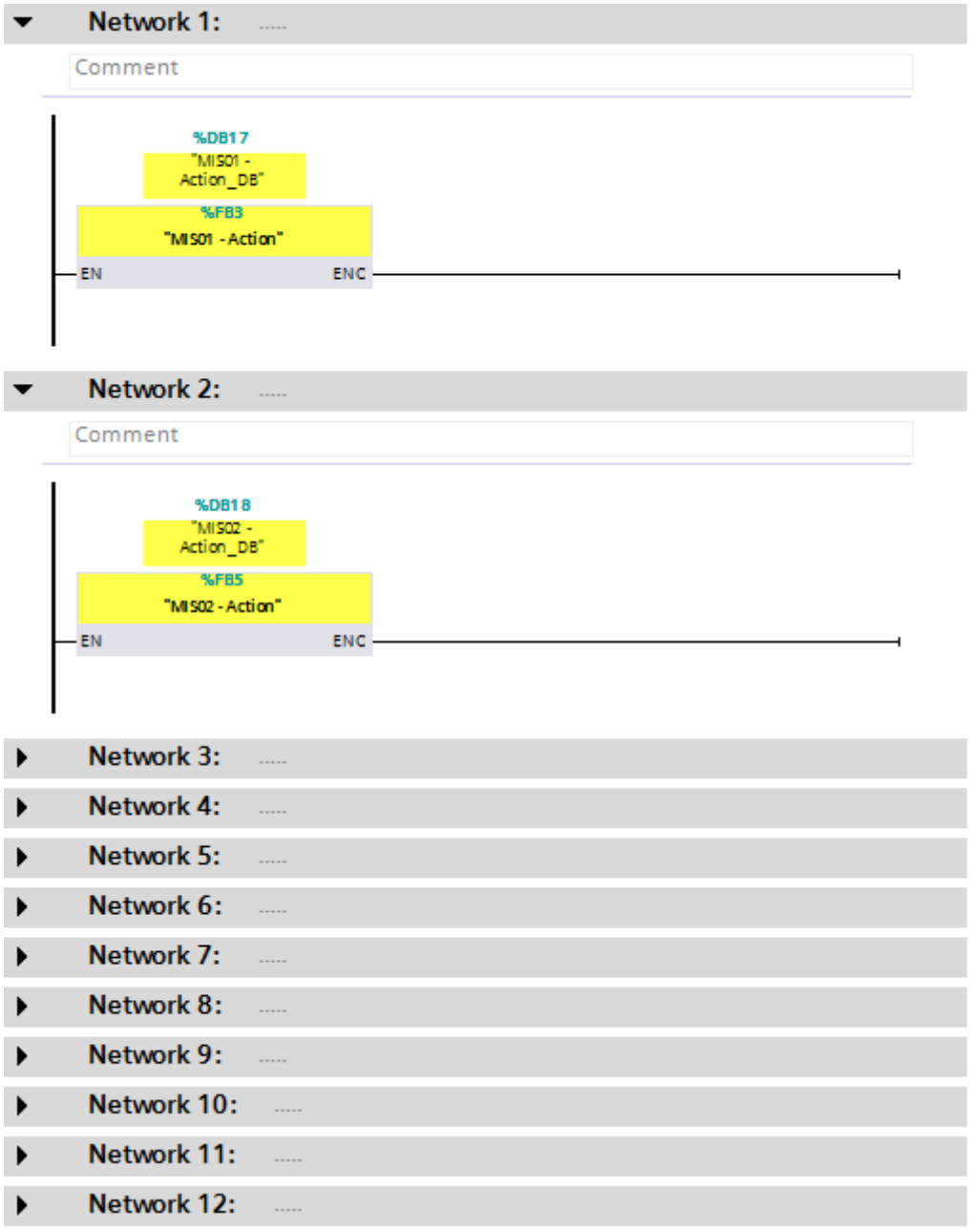
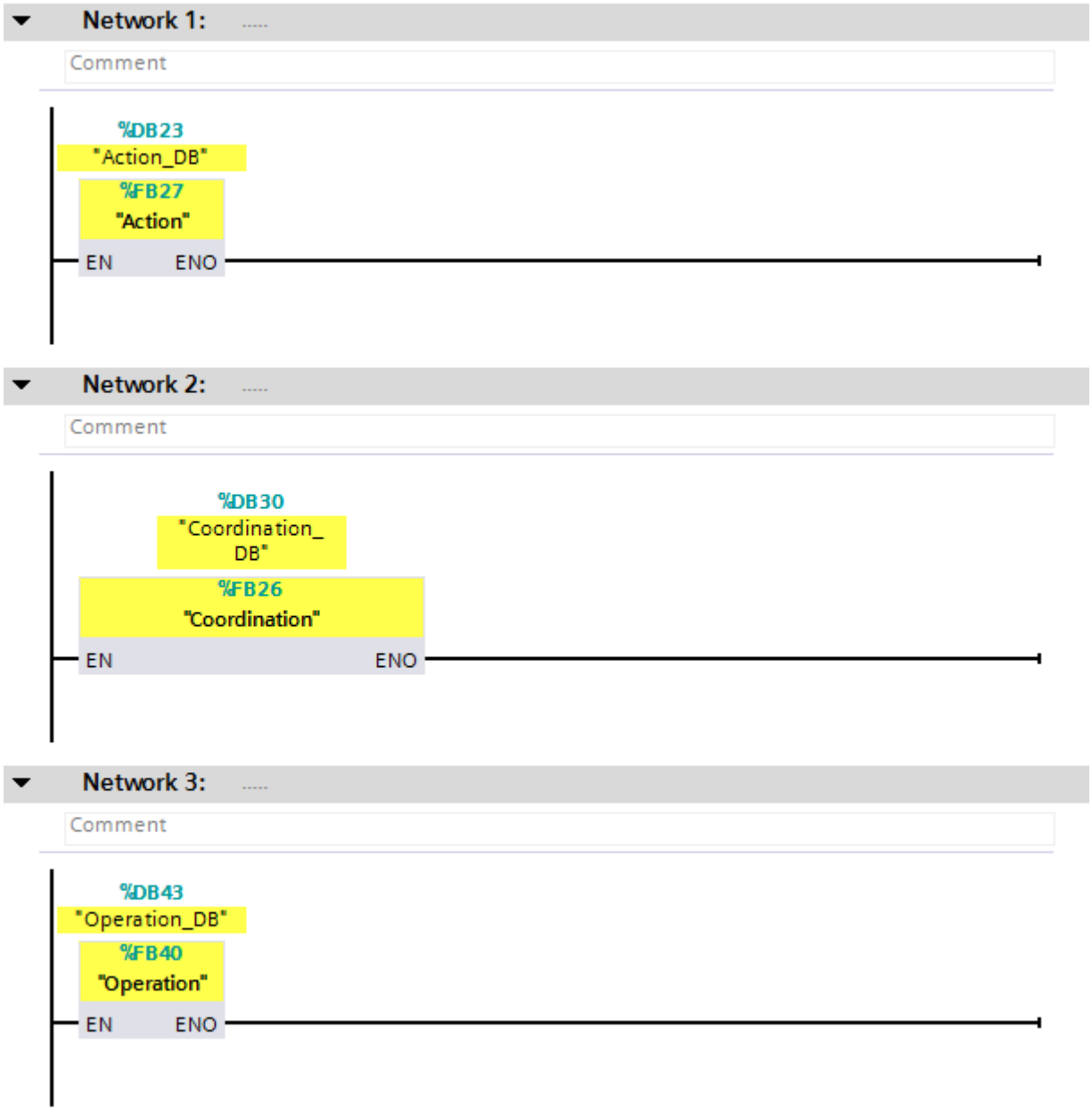


Figura 33 – Bloco Safety



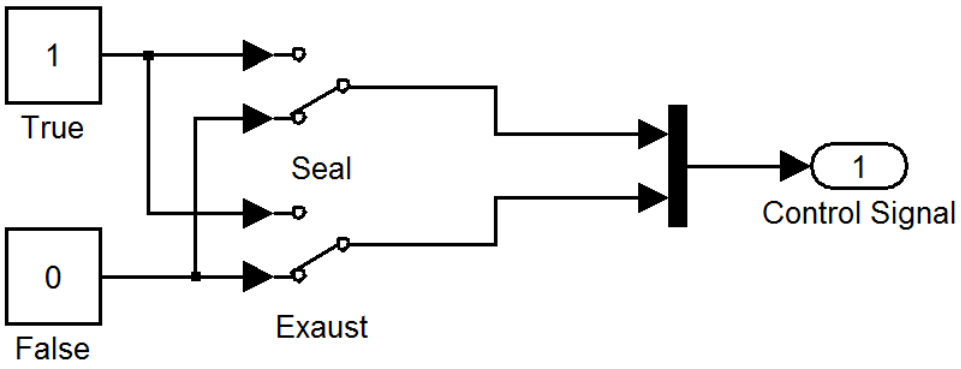
4 Discussão

4.1 Simulação e Testes

Cada um dos ambientes foram testados individualmente.

O ambiente SimuLink do Matlab permite que testes sejam realizados online, enviando os sinais de controle diretamente pelo programa sem que venha de uma fonte externa. Com isso é possível condicioná-lo e já verificar que atinge os estados desejados. Os primeiros testes foram realizados no nível de máquina de estados, com uma estrutura de botões selecionados manualmente pelo usuário como o apresentado na Figura 34.

Figura 34 – Sinal de controle enviado manualmente pelo usuário



Depois de cada uma das máquinas de estado serem testadas individualmente, o mesmo foi feito a nível de processo completo, com todas as saidas sendo visualizadas numa interface para facilitar a visualização do usuário através de LEDs virtuais, como apresentado na Figura 35.

Cada LED indica que um componente está preenchido com gás no sistema, mandando as regiões serem seladas e exauridas é possível acompanhar o caminho que o gás efetuado dentro do processo.

De maneira análoga os testes foram realizados no CLP somente, alterando de maneira manual o valor de leitura dos sensores, como demonstrado na Figura 36.

Ao utilizarmos as metodologias propostas por (MIYAGI, 2007) e (SQUILLANTE et al., 2011), nota-se que o desenvolvimento das aplicações se torna um processo bastante linear e elimina o processo de tentativa e erro. Inclusive utilizando abordagens diferentes para cada uma das partes, mas onde toda a estrutura já fora planejada para a fácil comunicação, implementação e testes.

4.2 Principais Dificuldades

4.2.1 CLP de Segurança

O CLP de segurança utilizado para a implementação desse trabalho de conclusão de curso foi na realidade projetado para uma outra implementação. Dito isso, o número de entradas e saidas disponíveis não são suficientes para a estação de compressão projetada. Seriam necessárias pelo menos 6 entradas para cada uma das 12 malhas - os 6 sensores, 3 para o componente A e 3 para o componente B. Entretanto, somente temos 24 inputs e para facilitar os testes as entradas foram então colocadas manualmente pelo programa no Tia Portal, agindo como se recebessem o sinal de alguma fonte externa (no caso, o próprio usuário). Além disso, para enviar os sinais de controle pelo Hardware in the Loop e para o script Matlab, seriam necessárias 4 saídas para cada uma das 12 malhas - sinais de selamento de A e B, sinais de exaustão de A e B. Contudo, só há 10 sinais de output disponíveis.

Pensando nas limitações de output, um protocolo de comunicação foi desenvolvido para essa aplicação específica, sendo que para controlar todas as saídas seriam necessários então somente 8 outputs. Foi projetado um byte (B0 à B7) de comunicação que tem a seguinte estrutura:

B0 - endereço 0;

B1 - endereço 1;

B2 - endereço 2;

B3 - endereço 3;

B4 - sela A;

B5 - sela B;

B6 - exaure A;

B7 - exaure B.

A lógica é que cada uma das malhas deixa registrado na memória os sinais de selo e exaustão - se as condições forem normais elas devem permanecer em nível lógico 0, indo para 1 conforme o programa solicitar. Um bloco de comunicação varreria cada um dos endereços a cada ciclo, enviando então as informações da malha correspondente - o endereço b0001 diz respeito à malha MIS01, o endereço b0010 diz respeito à malha MIS02, etc. O lado do Matlab que ficou responsável por tratar os endereços, fazendo a verificação e encaminhando o sinal de controle para o componente certo conforme o endereço correspondente.

Por exemplo, no primeiro ciclo ele mandaria b0001 pelos bits de endereço, enquanto

que nos bits de controle ele puxaria da memória os sinais que a malha MIS01 deseja enviar para a planta. No segundo ciclo ele mandaria b0010 pelos bits de endereço, enquanto que nos bits de controle ele puxaria da memória os sinais que a amlha MIS02 deseja enviar para a planta, e assim por diante.

Note que isso foi necessário devido às limitações impostas pelo hardware. Na ocasião da compra de um CLP de Segurança especificamente para o projeto, ele deve ser tal que atenda às especificações de quantidade e saídas necessárias para o seu funcionamento.

Em suma, diversas adaptações para a realização dos testes foram necessárias, mas que não impediram que as metologias fossem implementadas e que os algoritmos de controle e de simulação da planta funcionassem como esperado.

4.3 Trabalhos Futuros

O trabalho tem ainda bastante espaço para desenvolvimento, inclusive na utilização de um CLP de Segurança dedicado para o projeto.

Uma futura implementação envolve a utilização de um CLP com uma quantidade de entradas e saídas suficientes para a comunicação de todos os sensores, além dos testes com a planta real.

A primeira etapa desse novo trabalho é a aquisição dos sinais dos sensores. Como se trata de um sistema de segurança, todos os procedimentos para garantir que quando for colocado online ele haja da maneira como foi planejado. Após condicionamento dos sensores, o programa apresentado já é funcional, bastaria ligar as saídas nas válvulas e novamente garantir que elas funcionarão, o que de acordo com a norma também envolve a utilização de redes seguras para comunicação e fiação.

Figura 35 – Ambiente para simulação e testes

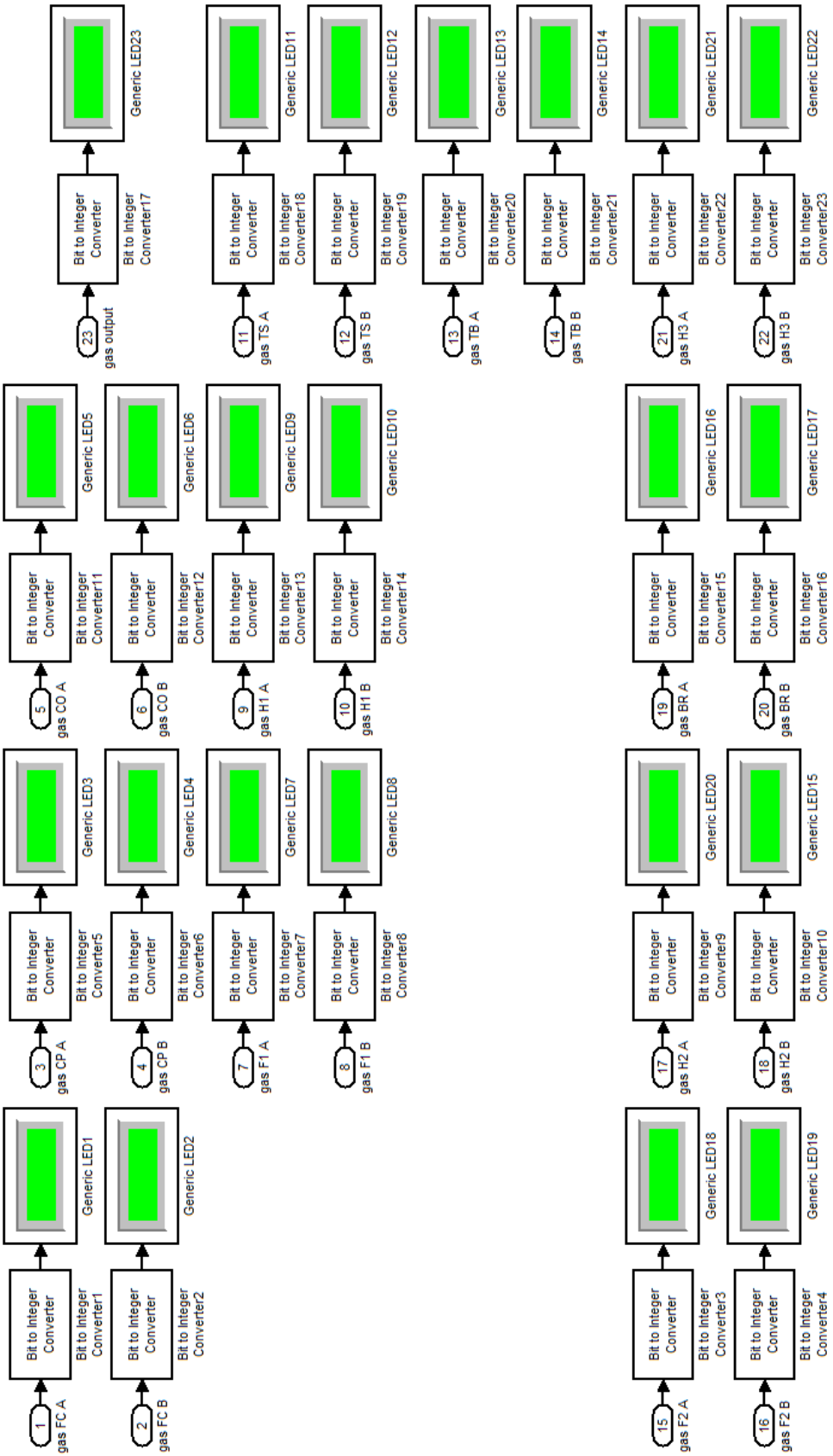
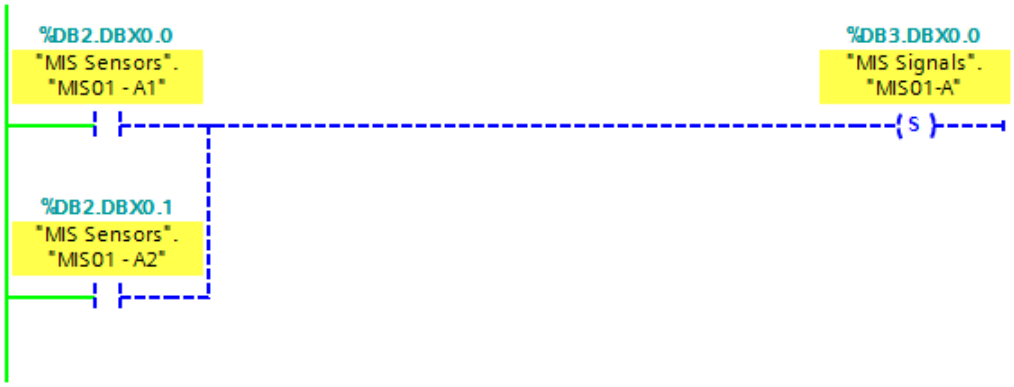


Figura 36 – Ambiente para simulação e testes



Conclusão

O objetivo principal desse trabalho estava no projeto e implementação de um sistema de controle de mitigação de falhas críticas. A engenharia de segurança, como um processo sistemático e gerando um produto autônomo, ainda é recente, ainda mais quando se trata da camada de mitigação. Com o desenvolvimento foi possível demonstrar uma aplicação prática e o resultado da aplicação da metodologia adaptada daquela apresentada por (SQUILLANTE et al., 2011). Outros trabalhos poderão continuar evoluindo nesse tema que é de grande importância, num ambiente que exige cada vez mais a especialização das atividades e a minimização da atividade humana.

Esse Trabalho de Conclusão de Curso serviu como fechamento para uma primeira fase na formação de um engenheiro pleno. Para o seu desenvolvimento foram necessárias diversas competências, entre elas:

1 - busca, interpretação e construção de textos científicos - entender o atual nível do estado da arte e ver de que forma o trabalho pode colaborar com a comunidade acadêmica;

2 - aplicação de metodologia - seguir uma estratégia para a realização das atividades macro e micro permitiram um desenvolvimento mais linear, onde mesmo que cada atividade necessitasse de mais tempo ela era realizada de maneira sistemática e praticamente eliminava tentativa e erro do processo;

3 - documentação - durante toda a extensão do projeto foi cobrado o desenvolvimento de marcos de documentação, através deles foi possivel identificar a situação atual do desenvolvimento e também já permitir que uma extenso arquivo fosse armazenado e evoluísse de maneira contínua.

Diversas dificuldades surgiram durante o desenvolvimento do projeto. As primeiras foram na concepção e abordagem. Apesar da temática de segurança ser amplamente estudada e referenciada em engenharia, as técnicas de mitigação muitas vezes aparecem de forma secundária e a bibliografia era mais limitada. Felizmente, as técnicas utilizadas para o desenvolvimento de sistemas de segurança podem tambem ser utilizadas para sistemas de mitigação, como na utilização das técnicas utilizadas e propostas em (SQUILLANTE et al., 2011).

Mais dificuldades surgiram na utilização do software de programação do CLP de Segurança. Como diversos outros softwares de engenharia, onde normalmente se está sujeito a uma gama muto elevada de possibilidades e configurações, as liberdades que o software disponibiliza muitas vezes se tornam barreiras para o desenvolvimento, com

uma quantidade grande de configurações para o funcionamento de processos simples. A linguagem de programação eram intuitivas e permitiam que modificações fossem feitas com facilidade - linguagens gráficas padrão como ladder e graphcet.

O produto final do projeto possui basicamente duas frentes, a primeira delas é o desenvolvimento do modelo da estação de compressão de gás natural como um sistema a eventos discretos na plataforma SimuLink do Matlab, a segunda delas é o desenvolvimento do algoritmo de controle de mitigação na plataforma Tia Portal da Siemens Automation.

A frente do Matlab foi uma necessidade para realizar as simulações, testes e condicionamento sem a existência de uma planta real, essa necessidade exige adaptações que não existiriam com a existência da planta, mas também são realizadas com um custo muito inferior e sem trazer riscos durante o processo.

A frente do Tia Portal usou a sistemática proposta em Squillante, que permite o desenvolvimento da programação de forma a adequar o código e demais estruturas para a norma IEC 61508, permitindo inclusive a rápida checagem dos blocos na procura de possíveis erros de lógica ou inconsistências.

Por fim, esse projeto serviu como um exercício para os futuros trabalhos como engenheiro, feitos sob elevada demanda, por vezes com cronograma imposto e por vezes definido, mas que na maioria das vezes vai exigir elevada capacidade de organização e a habilidade de encontrar respostas e desenvolver soluções por conta, uma competência cobrada durante a graduação e que dá um diferencial no mercado para o engenheiro politécnico.

Referências

COOPER, G.; HERKSKOVITS, E. A bayesian method for the induction of probabilistic networks from data. 1992. Citado na página 111.

DARAMOLA, O. et al. Enabling hazard identification from requirements and reuse-oriented hazop analysis. August 2011. Fourth International Workshop. Citado na página 19.

DILTS, D. M.; BOYD, N. P.; WHORMS, H. H. The evolution of control architectures for automated manufacturing systems. 1991. Citado na página 107.

GALL, H. Functional safety iec 61508 / iec 61511 the impact to certification and the user. April 2008. IEEE/ACS International Conference. Citado na página 22.

GEORGAKOPOULOS, D. et al. Managing escalation of collaboration processes in crisis mitigation situations. 2000. Proceedings. 16th International Conference. Citado na página 16.

GERGELY, E. et al. Design framework for risk mitigation in industrial plc control. May 2008. IEEE International Conference. Citado na página 43.

HAYEK, A.; BORCSOK, J. Sram-based fpga design techniques for safety related systems conforming to iec 61508 a survey and analysis. December 2012. 2nd International Conference. Citado na página 23.

IEC61508. USA, 2010. Functional Safety of Electrical/Electronic/Programmable Electronic Safety-related Systems. Citado 7 vezes nas páginas 23, 24, 25, 26, 27, 28 e 29.

IETEG. Diagrama de fluxo peid. 2010. Citado na página 20.

KHARCHENKO, V.; BREZHNEV, E.; BOYARCHUK, A. The smart grid's safety: Dynamical hierarchical criticality matrices-based analysis. December 2011. 2nd IEEE PES International Conference and Exhibition. Citado na página 16.

MIYAGI, P. E. Controle programavel: fndamentos do controle de sistemas a eventos discretos. Sao Paulo, September 2007. IEEE Conference. Citado 4 vezes nas páginas 32, 33, 50 e 108.

REICHENBACH, F.; ALME, K. J.; ENDRESEN, J. On the significance of fault tree analysis in practice. September 2009. IEEE Conference. Citado na página 17.

RODRIGUES, W. Instrumentação industrial - simbologia e terminologia da norma isa 5.1. Sao Paulo, 2010. Citado 3 vezes nas páginas 17, 18 e 19.

RUSSEL, S. et al. Artificial intelligence: A modern approach. 2003. Citado na página 111.

SANTOS, F. D. J. Considerações teóricas sobre redes de petri e pfs. 2013. Citado 6 vezes nas páginas 107, 108, 109, 110, 111 e 112.

SEARCH TRIANGLE PARK. Isa 5.1. USA, 2009. Instrumentation Symbols and Identification. Citado na página 17.

SQUILLANTE, J. R. Diagnostico e tratamento de falhas criticas em sistemas instrumentados de seguranca. Sao Paulo, 2011. Citado 2 vezes nas páginas 15 e 32.

SQUILLANTE, J. R. et al. Development of control systems for safety instrumented systems. Sao Paulo, July 2011. IEEE LATIN AMERICA TRANSACTIONS. Citado 4 vezes nas páginas 34, 44, 50 e 55.

STROM, M.; ANDERSSON, R.; KIVISTO, R. Uml modelling concepts of hazop to enhance the ability to identify emerging risks. 2013. Citado na página 21.

SUBRAMANIAN, S. et al. Fault mitigation in safety-critical software systems. June 1996. Proceedings Ninth IEEE Symposium. Citado na página 16.

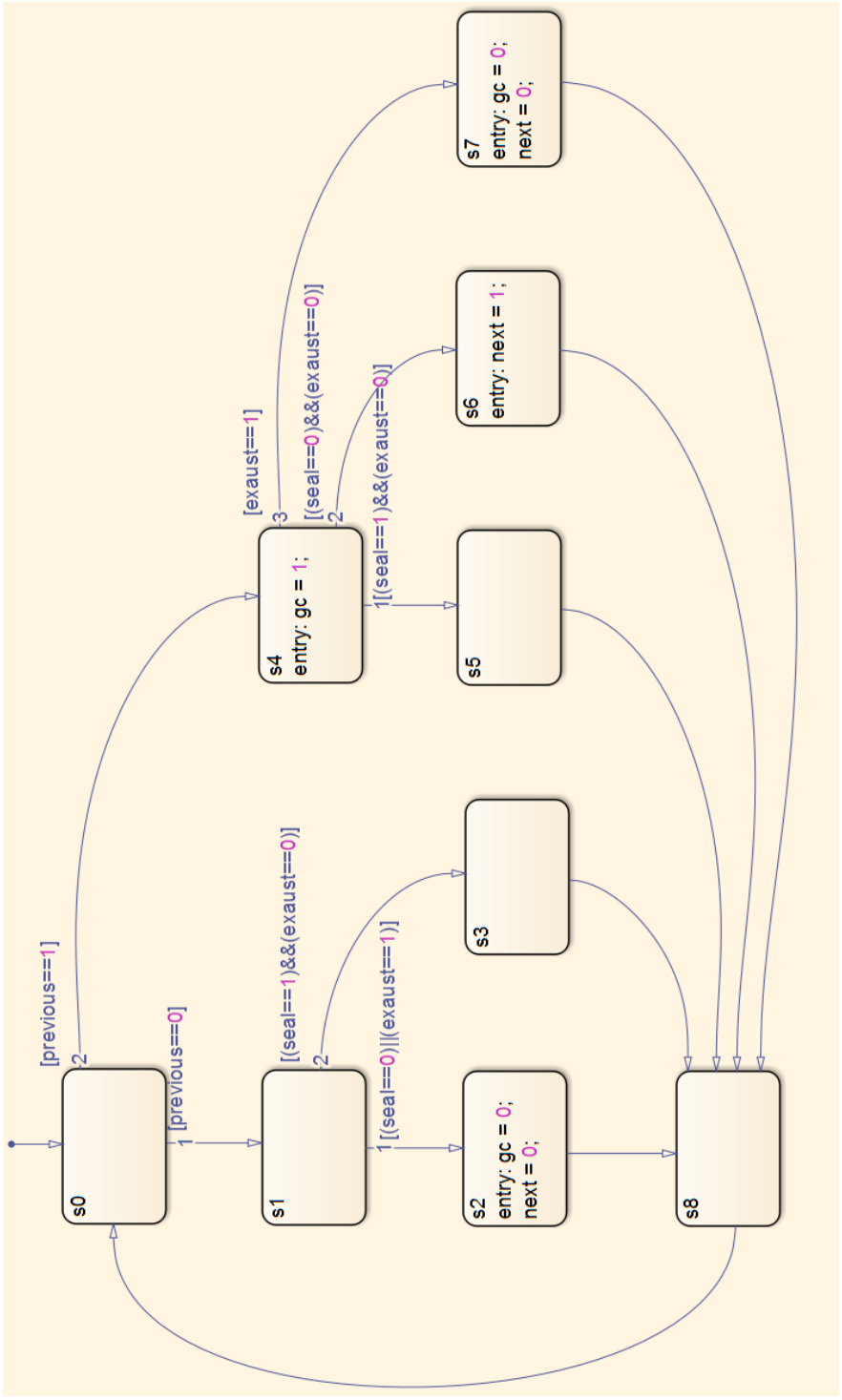
SUYAMA, K. Safety integrity analysis framework for a controller according to iec 61508. December 2003. Proceedings. 42nd IEEE Conference. Citado na página 22.

WIGHTKIN, N.; BUY, U.; DARABI, H. Formal modeling of sequential function charts with time petri nets. 2010. Citado na página 112.

Apêndices

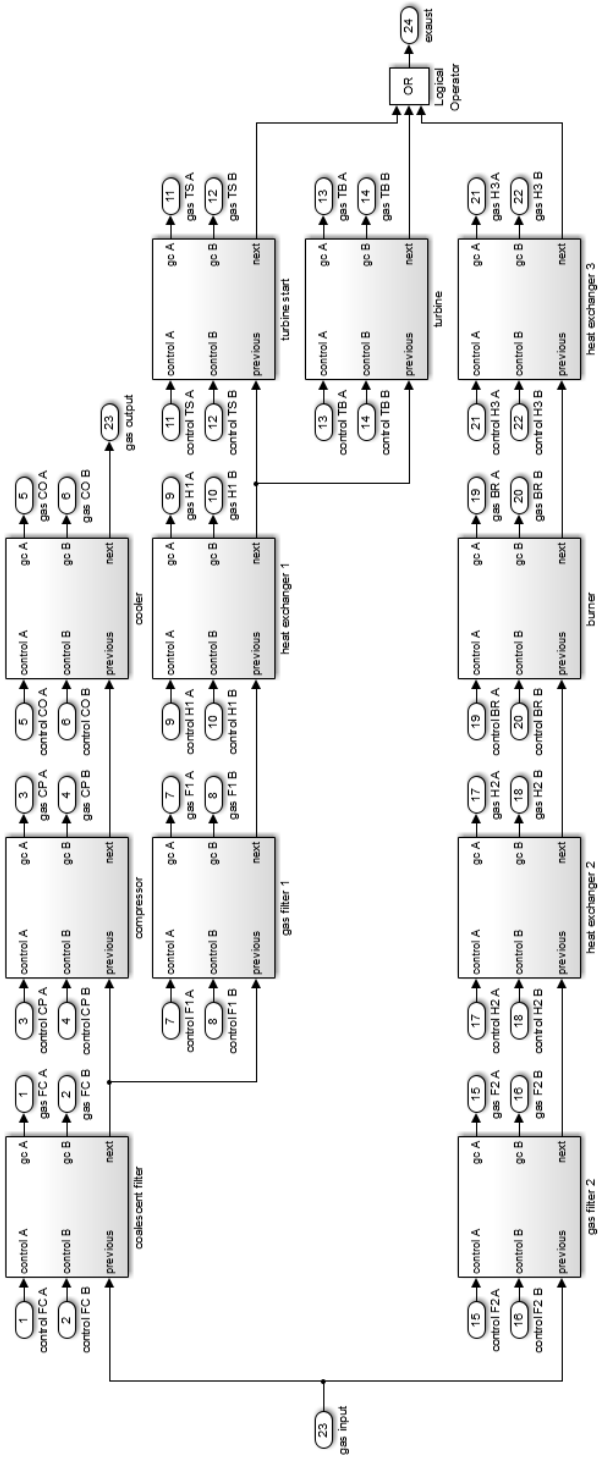
APÊNDICE A – Máquina de Estados

Figura 37 – Resultado da Implementação da Metodologia



APÊNDICE B – Modelo Simulink

Figura 38 – Resultado da Modelagem da Planta no SimuLink



APÊNDICE C – Malhas de Segurança

Tabela 6 – Tabela da Malha de Segurança 1

SIF-01				
Serviço:	Pressão muito baixa na entrada dos filtros coalescedores (FC) por vazão alta			
Identificação dos Iniciadores:	PIT-01A/01B			
Identificação dos Atuadores:	XV-001A/001B; XV-101A/101B			
Demanda:	W2			Classe Associada
Segurança Pessoal:	S2	A1	G2	SIL 1
Perda de Produção	L2			SIL 0
Meio Ambiente:	E1			SIL 1
Classificação Final:	SIL 1			

Tabela 7 – Tabela da Malha de Segurança 2

SIF-02				
Serviço:	Nível muito alto nos filtros coalescedores (FC)			
Identificação dos Iniciadores:	LIT-02A/02B			
Identificação dos Atuadores:	XV-002A/002B; XV-102A/102B			
Demanda:	W2			Classe Associada
Segurança Pessoal:	S0			-
Perda de Produção	L4			SIL 1
Meio Ambiente:	E1			SIL 1
Classificação Final:	SIL 1			

Tabela 8 – Tabela da Malha de Segurança 3

SIF-03				
Serviço:	Diferencial de pressão muito alto nos filtros coalescedores (FC)			
Identificação dos Iniciadores:	PDIT-03A/03B			
Identificação dos Atuadores:	XV-003A/003B; XV-103A/103B			
Demanda:	W2			Classe Associada
Segurança Pessoal:	S0			-
Perda de Produção	L2			SIL 0
Meio Ambiente:	E0			-
Classificação Final:	SIL 0			

Tabela 9 – Tabela da Malha de Segurança 4

SIF-04				
Serviço:	Pressão muito alta na descarga dos compressores (CP)			
Identificação dos Iniciadores:	PIT-04A/04B			
Identificação dos Atuadores:	XV-004A/004B; XV-104A/104B			
Demanda:	W2			Classe Associada
Segurança Pessoal:	S2	A1	G2	SIL 1
Perda de Produção	L2			SIL 0
Meio Ambiente:	E1			SIL 1
Classificação Final:	SIL 1			

Tabela 10 – Tabela da Malha de Segurança 5

SIF-05				
Serviço:	Pressão muito alta no header de descarga dos compressores (CP)			
Identificação dos Iniciadores:	PIT-05A/05B			
Identificação dos Atuadores:	XV-005A/005B; XV-105A/105B			
Demanda:	W2			Classe Associada
Segurança Pessoal:	S2	A1	G2	SIL 1
Perda de Produção	L4			SIL 1
Meio Ambiente:	E1			SIL 1
Classificação Final:	SIL 1			

Tabela 11 – Tabela da Malha de Segurança 6

SIF-06				
Serviço:	Temperatura muito alta na descarga dos compressores (CP)			
Identificação dos Iniciadores:	TIT-06A/06B			
Identificação dos Atuadores:	XV-006A/006B; XV-106A/106B			
Demanda:	W2			Classe Associada
Segurança Pessoal:	S1			SIL 0
Perda de Produção	L1			SIL 0
Meio Ambiente:	E1			SIL 1
Classificação Final:	SIL 1			

Tabela 12 – Tabela da Malha de Segurança 7

SIF-07				
Serviço:	Pressão muito baixa no gás para as turbinas (TB) por vazão alta			
Identificação dos Iniciadores:	PIT-07A/07B			
Identificação dos Atuadores:	XV-007A/007B; XV-107A/107B			
Demanda:	W2			Classe Associada
Segurança Pessoal:	S2	A1	G2	SIL 1
Perda de Produção	L2			SIL 0
Meio Ambiente:	E1			SIL 1
Classificação Final:	SIL 1			

Tabela 13 – Tabela da Malha de Segurança 8

SIF-08				
Serviço:	Pressão muito alta no gás para as turbinas (TB)			
Identificação dos Iniciadores:	PIT-08A/08B			
Identificação dos Atuadores:	XV-008A/008B; XV-108A/108B			
Demanda:	W2			Classe Associada
Segurança Pessoal:	S0			-
Perda de Produção	L4			SIL 1
Meio Ambiente:	E1			SIL 1
Classificação Final:	SIL 1			

Tabela 14 – Tabela da Malha de Segurança 9

SIF-09				
Serviço:	Pressão muito baixa no gás para as turbinas (TS) por vazão alta			
Identificação dos Iniciadores:	PIT-09A/09B			
Identificação dos Atuadores:	XV-009A/009B; XV-109A/109B			
Demanda:	W1			Classe Associada
Segurança Pessoal:	S1			SIL 0
Perda de Produção	L1			SIL 0
Meio Ambiente:	E0			-
Classificação Final:	SIL 0			

Tabela 15 – Tabela da Malha de Segurança 10

SIF-10				
Serviço:	Pressão muito alta no gás para as turbinas (TS)			
Identificação dos Iniciadores:	PIT-10A/10B			
Identificação dos Atuadores:	XV-010A/010B; XV-110A/110B			
Demanda:	W2			Classe Associada
Segurança Pessoal:	S1			SIL 0
Perda de Produção	L4			SIL 1
Meio Ambiente:	E1			SIL 1
Classificação Final:	SIL 1			

Tabela 16 – Tabela da Malha de Segurança 11

SIF-11				
Serviço:	Temperatura muito baixa na saída do queimador (BR)			
Identificação dos Iniciadores:	TIT-11A/11B			
Identificação dos Atuadores:	XV-011A/011B; XV-111A/111B			
Demanda:	W2			Classe Associada
Segurança Pessoal:	S0			-
Perda de Produção	L4			SIL 1
Meio Ambiente:	E1			SIL 1
Classificação Final:	SIL 1			

Tabela 17 – Tabela da Malha de Segurança 12

SIF-12				
Serviço:	Temperatura muito alta no duto de exaustão			
Identificação dos Iniciadores:	PIT-12A/12B			
Identificação dos Atuadores:	XV-012A/012B; XV-112A/112B			
Demanda:	W2			Classe Associada
Segurança Pessoal:	S0			-
Perda de Produção	L1			SIL 0
Meio Ambiente:	E0			-
Classificação Final:	SIL 0			

APÊNDICE D – Malhas de Segurança

Programa desenvolvido para a parte do controlador de segurança no programa Step 7 v12 da Siemens Automation.

Note que o programa é uma versão compacta com somente uma malha de controle implementada. A versão completa do código possui mais de 400 páginas de código e sua estrutura foi descrita durante o texto.

TCC - Tia Side Compact

Table of contents

Program blocks

Main [OB1]

Main Properties					
General					
Name	Main	**Number**	1	**Type**	OB
Language	LAD				
Information					
Title	"Main Program Sweep (Cycle)"	**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Offset
▼ Temp		
OB1_EV_CLASS	Byte	0.0
OB1_SCAN_1	Byte	1.0
OB1_PRIORITY	Byte	2.0
OB1_OB_NUMBR	Byte	3.0
OB1_RESERVED_1	Byte	4.0
OB1_RESERVED_2	Byte	5.0
OB1_PREV_CYCLE	Int	6.0
OB1_MIN_CYCLE	Int	8.0
OB1_MAX_CYCLE	Int	10.0
OB1_DATE_TIME	Date_And_Time	12.0

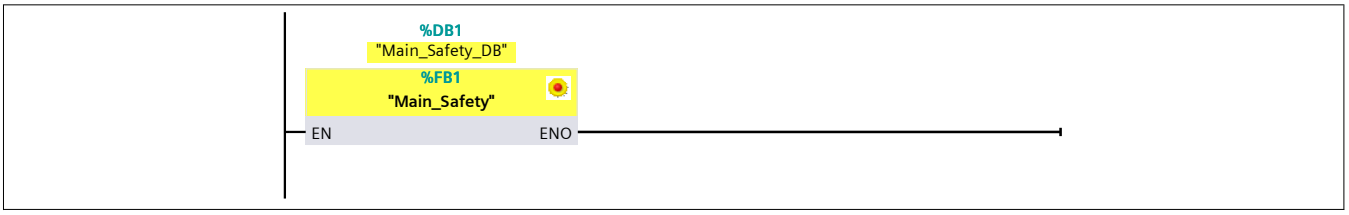
Program blocks

CYC_INT5 [OB35]

CYC_INT5 Properties					
General					
Name	CYC_INT5	**Number**	35	**Type**	OB
Language	LAD				
Information					
Title	"Cyclic Interrupt"	**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Offset
▼ Temp		
OB35_EV_CLASS	Byte	0.0
OB35_STRT_INF	Byte	1.0
OB35_PRIORITY	Byte	2.0
OB35_OB_NUMBR	Byte	3.0
OB35_RESERVED_1	Byte	4.0
OB35_RESERVED_2	Byte	5.0
OB35_PHASE_OFFSET	Word	6.0
OB35_RESERVED_3	Int	8.0
OB35_EXC_FREQ	Int	10.0
OB35_DATE_TIME	Date_And_Time	12.0

Network 1:



Program blocks

Main_Safety [FB1]

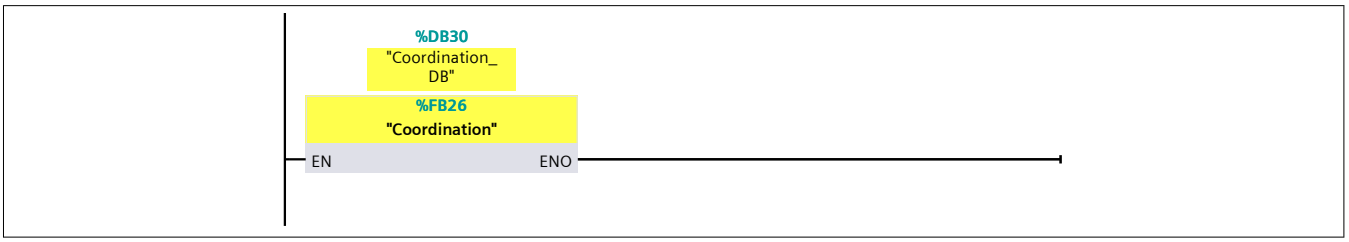
Main_Safety Properties					
General					
Name	Main_Safety	**Number**	1	**Type**	FB.Safety
Language	LAD				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Offset	Default value	Retain
Input				
Output				
InOut				
Static				
Temp				

Network 1:

Symbol	Address	Type	Comment
"Action"	%FB27	Block_FB	
"Action_DB"	%DB23	Block_FB	

Network 2:



Symbol	Address	Type	Comment
"Coordination"	%FB26	Block_FB	
"Coordination_DB"	%DB30	Block_FB	

Network 3:

Symbol	Address	Type	Comment
"Operation"	%FB40	Block_FB	
"Operation_DB"	%DB43	Block_FB	

Program blocks

Main_Safety_DB [DB1]

Main_Safety_DB Properties					
General					
Name	Main_Safety_DB	**Number**	1	**Type**	DB.Safety
Language	DB				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Start value	Retain
Input			
Output			
InOut			
Static			

Program blocks

MIS Sensors [DB2]

MIS Sensors Properties					
General					
Name	MIS Sensors	**Number**	2	**Type**	DB.Safety
Language	DB				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Start value	Retain
▼ Static			
MIS01 - A1	Bool	false	False
MIS01 - A2	Bool	false	False
MIS01 - A3	Bool	false	False
MIS01 - B1	Bool	false	False
MIS01 - B2	Bool	false	False
MIS01 - B3	Bool	false	False
Fire Signal 1	Bool	false	False
Fire Signal 2	Bool	false	False
Fire Signal 3	Bool	false	False

Program blocks

MIS Signals [DB3]

MIS Signals Properties					
General					
Name	MIS Signals	**Number**	3	**Type**	DB.Safety
Language	DB				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Start value	Retain
▼ Static			
MIS01-A	Bool	false	False
MIS01-B	Bool	false	False
MIS01-ACK	Bool	false	False

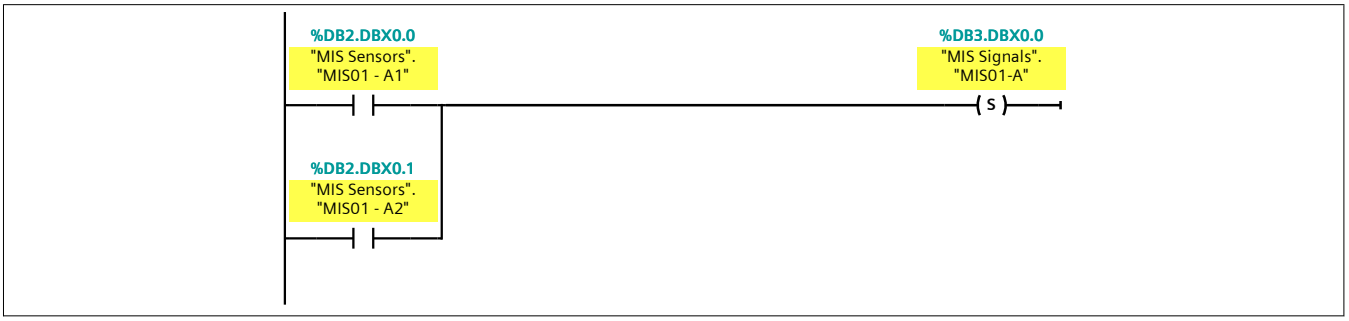
Program blocks

MIS01 - Coordination [FB2]

MIS01 - Coordination Properties					
General					
Name	MIS01 - Coordination	**Number**	2	**Type**	FB.Safety
Language	LAD				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

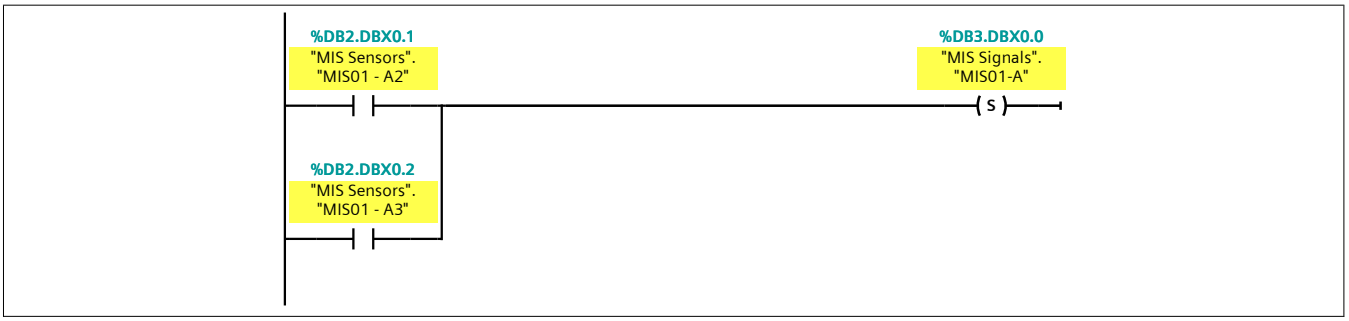
Name	Data type	Offset	Default value	Retain
Input				
Output				
InOut				
Static				
Temp				

Network 1:



Symbol	Address	Type	Comment
"MIS Sensors"	%DB2	Block_DB	
"MIS Sensors"."MIS01 - A1"	%DB2.DBX0.0	Bool	
"MIS Sensors"."MIS01 - A2"	%DB2.DBX0.1	Bool	
"MIS Signals"	%DB3	Block_DB	
"MIS Signals"."MIS01-A"	%DB3.DBX0.0	Bool	

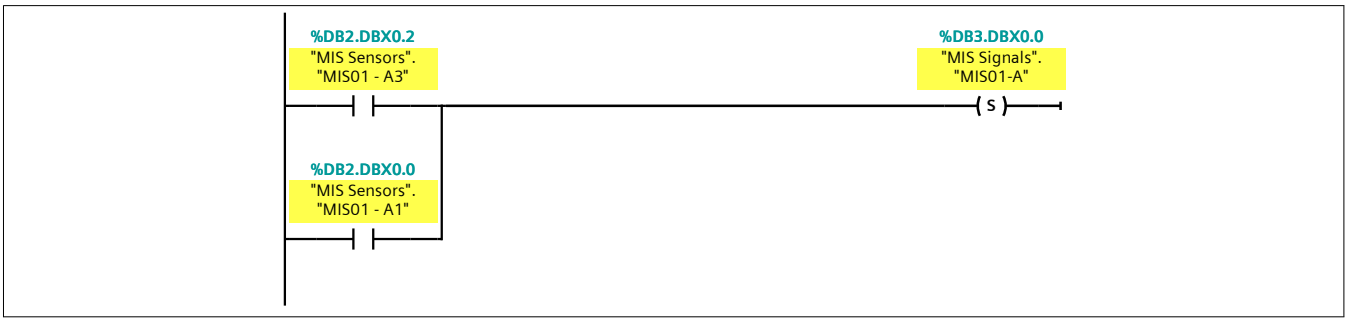
Network 2:



Safety information: E1FA2657 / DA7DE5B3; STEP 7 Safety V12; The safety program is inconsistent.

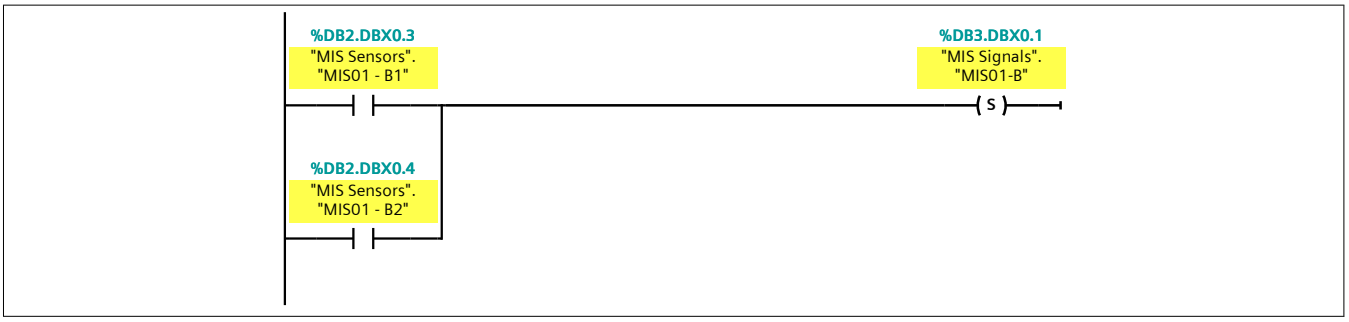
Symbol	Address	Type	Comment
"MIS Sensors"	%DB2	Block_DB	
"MIS Sensors"."MIS01 - A2"	%DB2.DBX0.1	Bool	
"MIS Signals"	%DB3	Block_DB	
"MIS Signals"."MIS01-A"	%DB3.DBX0.0	Bool	
"MIS Sensors"."MIS01 - A3"	%DB2.DBX0.2	Bool	

Network 3:



Symbol	Address	Type	Comment
"MIS Sensors"	%DB2	Block_DB	
"MIS Sensors"."MIS01 - A1"	%DB2.DBX0.0	Bool	
"MIS Signals"	%DB3	Block_DB	
"MIS Signals"."MIS01-A"	%DB3.DBX0.0	Bool	
"MIS Sensors"."MIS01 - A3"	%DB2.DBX0.2	Bool	

Network 4:



Symbol	Address	Type	Comment
"MIS Sensors"	%DB2	Block_DB	
"MIS Signals"	%DB3	Block_DB	
"MIS Sensors"."MIS01 - B1"	%DB2.DBX0.3	Bool	
"MIS Sensors"."MIS01 - B2"	%DB2.DBX0.4	Bool	
"MIS Signals"."MIS01-B"	%DB3.DBX0.1	Bool	

Network 5:

Symbol	Address	Type	Comment
"MIS Sensors"	%DB2	Block_DB	
"MIS Signals"	%DB3	Block_DB	
"MIS Sensors"."MIS01 - B2"	%DB2.DBX0.4	Bool	
"MIS Signals"."MIS01-B"	%DB3.DBX0.1	Bool	
"MIS Sensors"."MIS01 - B3"	%DB2.DBX0.5	Bool	

Network 6:

Symbol	Address	Type	Comment
"MIS Sensors"	%DB2	Block_DB	
"MIS Signals"	%DB3	Block_DB	
"MIS Sensors"."MIS01 - B1"	%DB2.DBX0.3	Bool	
"MIS Signals"."MIS01-B"	%DB3.DBX0.1	Bool	
"MIS Sensors"."MIS01 - B3"	%DB2.DBX0.5	Bool	

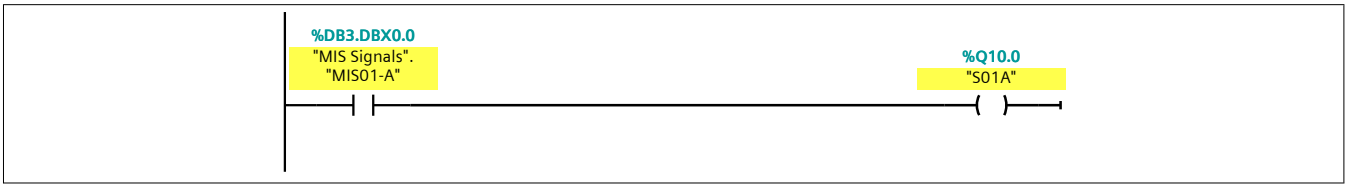
Program blocks

MIS01 - Action [FB3]

MIS01 - Action Properties					
General					
Name	MIS01 - Action	**Number**	3	**Type**	FB.Safety
Language	LAD				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Offset	Default value	Retain
Input				
Output				
InOut				
Static				
Temp				

Network 1:



Symbol	Address	Type	Comment
"MIS Signals"	%DB3	Block_DB	
"MIS Signals"."MIS01-A"	%DB3.DBX0.0	Bool	
"S01A"	%Q10.0	Bool	

Network 2:

Symbol	Address	Type	Comment
"MIS Signals"	%DB3	Block_DB	
"MIS Signals"."MIS01-B"	%DB3.DBX0.1	Bool	
"S01B"	%Q10.1	Bool	

Program blocks

MISAUX [DB4]

MISAUX Properties					
General					
Name	MISAUX	**Number**	4	**Type**	DB.Safety
Language	DB				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Start value	Retain
Static			

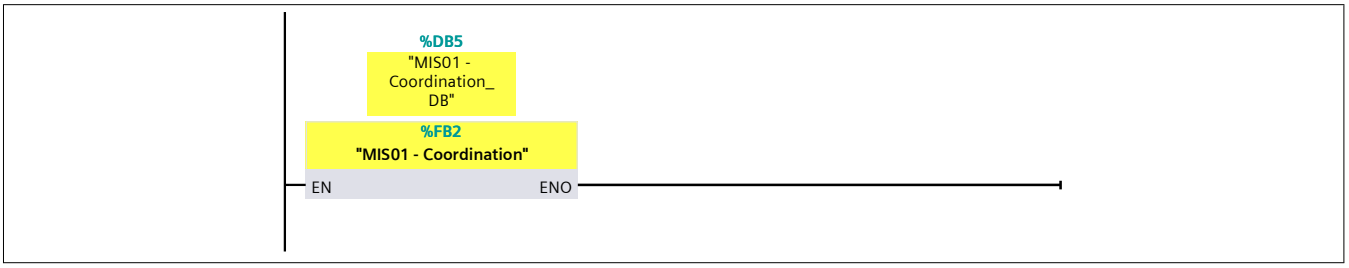
Program blocks

Coordination [FB26]

Coordination Properties					
General					
Name	Coordination	**Number**	26	**Type**	FB.Safety
Language	LAD				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Offset	Default value	Retain
Input				
Output				
InOut				
Static				
Temp				

Network 1:



Symbol	Address	Type	Comment
"MIS01 - Coordination"	%FB2	Block_FB	
"MIS01 - Coordination_DB"	%DB5	Block_FB	

Program blocks

MIS01 - Coordination_DB [DB5]

MIS01 - Coordination_DB Properties					
General					
Name	MIS01 - Coordination_DB	**Number**	5	**Type**	DB.Safety
Language	DB				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Start value	Retain
Input			
Output			
InOut			
Static			

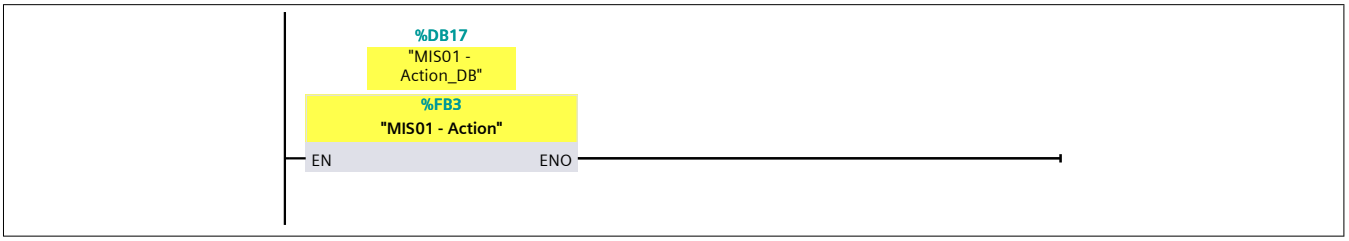
Program blocks

Action [FB27]

Action Properties					
General					
Name	Action	**Number**	27	**Type**	FB.Safety
Language	LAD				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Offset	Default value	Retain
Input				
Output				
InOut				
Static				
Temp				

Network 1:



Symbol	Address	Type	Comment
"MIS01 - Action"	%FB3	Block_FB	
"MIS01 - Action_DB"	%DB17	Block_FB	

Program blocks

MIS01 - Action_DB [DB17]

MIS01 - Action_DB Properties					
General					
Name	MIS01 - Action_DB	**Number**	17	**Type**	DB.Safety
Language	DB				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Start value	Retain
Input			
Output			
InOut			
Static			

Program blocks

Action_DB [DB23]

Action_DB Properties					
General					
Name	Action_DB	**Number**	23	**Type**	DB.Safety
Language	DB				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Start value	Retain
Input			
Output			
InOut			
Static			

Program blocks

Coordination_DB [DB30]

Coordination_DB Properties					
General					
Name	Coordination_DB	**Number**	30	**Type**	DB.Safety
Language	DB				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Start value	Retain
Input			
Output			
InOut			
Static			

Program blocks

MIS01 - Operation [FB28]

MIS01 - Operation Properties					
General					
Name	MIS01 - Operation	**Number**	28	**Type**	FB.Safety
Language	LAD				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Offset	Default value	Retain
Input				
ACK	Bool	0.0	false	Set in IDB
Output				
InOut				
Static				
Temp				

Network 1:

Symbol	Address	Type	Comment
#ACK		Bool	
"MIS Signals"	%DB3	Block_DB	
"MIS Signals"."MIS01-A"	%DB3.DBX0.0	Bool	
"MIS Signals"."MIS01-B"	%DB3.DBX0.1	Bool	

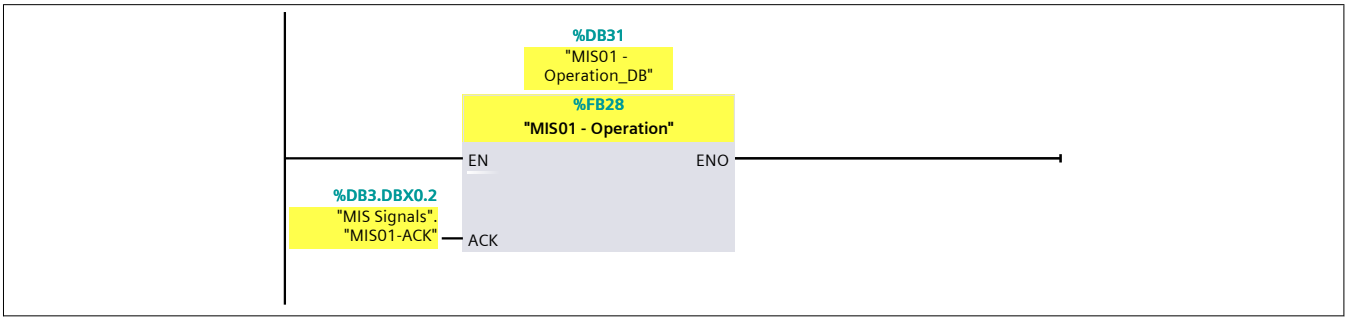
Program blocks

Operation [FB40]

Operation Properties					
General					
Name	Operation	**Number**	40	**Type**	FB.Safety
Language	LAD				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Offset	Default value	Retain
Input				
Output				
InOut				
Static				
Temp				

Network 1:



Symbol	Address	Type	Comment
"MIS01 - Operation"	%FB28	Block_FB	
"MIS01 - Operation_DB"	%DB31	Block_FB	
"MIS Signals"	%DB3	Block_DB	
"MIS Signals"."MIS01-ACK"	%DB3.DBX0.2	Bool	

Program blocks

MIS01 - Operation_DB [DB31]

MIS01 - Operation_DB Properties					
General					
Name	MIS01 - Operation_DB	**Number**	31	**Type**	DB.Safety
Language	DB				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Start value	Retain
▼ Input			
ACK	Bool	false	False
Output			
InOut			
Static			

Program blocks

Operation_DB [DB43]

Operation_DB Properties					
General					
Name	Operation_DB	**Number**	43	**Type**	DB.Safety
Language	DB				
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Start value	Retain
Input			
Output			
InOut			
Static			

Program blocks / System blocks / STEP 7 Safety

F_GLOBDB [DB8001]

F_GLOBDB Properties					
General					
Name	F_GLOBDB	**Number**	8001	**Type**	DB.Safety
Language	DB				
Information					
Title		**Author**		**Comment**	
Family	Safety11	**Version**	0.1	**User-defined ID**	F_GLOBDB

Name	Data type	Start value	Retain
▼ Static			
F_PROG_SIG	DWord	16#0	False
MODE	Bool	false	False
TESTM	Bool	false	False
ERROR	Bool	false	False
VKE0	Bool	false	False
VKE1	Bool	true	False
F_PROG_DAT	Date_And_Time	DT#1990-01-01-00:00:00	False

Program blocks / System blocks / STEP 7 Safety / Compiler blocks

F_IO_CGP [FB3999]

F_IO_CGP Properties					
General					
Name	F_IO_CGP	**Number**	3999	**Type**	FB.Safety
Language	F_STL				
Information					
Title	F_: Driver Block In-Output with Channel Granular Passivation	**Author**	SafeSys	**Comment**	CRC=ece1/IDB=2393 Build: S7FP_CLIB_BUILD_11.00.01.00_01.18.00.00; 20.07.2011 17:42:58
Family	F_DRIVER	**Version**	1.0	**User-defined ID**	F_IO_CGP

Name	Data type	Offset	Default value	Retain
▼ Input				
PASS_ON	Bool	0.0	false	Set in IDB
ACK_NEC	Bool	0.1	TRUE	Set in IDB
ACK_REI	Bool	0.2	false	Set in IDB
IPAR_EN	Bool	0.3	false	Set in IDB
▼ Output				
PASS_OUT	Bool	2.0	TRUE	Set in IDB
QBAD	Bool	2.1	TRUE	Set in IDB
ACK_REQ	Bool	2.2	false	Set in IDB
IPAR_OK	Bool	2.3	false	Set in IDB
DIAG	Byte	3.0	16#0	Set in IDB
QBAD_I_00	Bool	4.0	TRUE	Set in IDB
QBAD_I_01	Bool	4.1	TRUE	Set in IDB
QBAD_I_02	Bool	4.2	TRUE	Set in IDB
QBAD_I_03	Bool	4.3	TRUE	Set in IDB
QBAD_I_04	Bool	4.4	TRUE	Set in IDB
QBAD_I_05	Bool	4.5	TRUE	Set in IDB
QBAD_I_06	Bool	4.6	TRUE	Set in IDB
QBAD_I_07	Bool	4.7	TRUE	Set in IDB
QBAD_I_08	Bool	5.0	TRUE	Set in IDB
QBAD_I_09	Bool	5.1	TRUE	Set in IDB
QBAD_I_10	Bool	5.2	TRUE	Set in IDB
QBAD_I_11	Bool	5.3	TRUE	Set in IDB
QBAD_I_12	Bool	5.4	TRUE	Set in IDB
QBAD_I_13	Bool	5.5	TRUE	Set in IDB
QBAD_I_14	Bool	5.6	TRUE	Set in IDB
QBAD_I_15	Bool	5.7	TRUE	Set in IDB
QBAD_I_16	Bool	6.0	TRUE	Set in IDB
QBAD_I_17	Bool	6.1	TRUE	Set in IDB
QBAD_I_18	Bool	6.2	TRUE	Set in IDB
QBAD_I_19	Bool	6.3	TRUE	Set in IDB
QBAD_I_20	Bool	6.4	TRUE	Set in IDB
QBAD_I_21	Bool	6.5	TRUE	Set in IDB
QBAD_I_22	Bool	6.6	TRUE	Set in IDB
QBAD_I_23	Bool	6.7	TRUE	Set in IDB
QBAD_I_24	Bool	7.0	TRUE	Set in IDB

Name	Data type	Offset	Default value	Retain
QBAD_I_25	Bool	7.1	TRUE	Set in IDB
QBAD_I_26	Bool	7.2	TRUE	Set in IDB
QBAD_I_27	Bool	7.3	TRUE	Set in IDB
QBAD_I_28	Bool	7.4	TRUE	Set in IDB
QBAD_I_29	Bool	7.5	TRUE	Set in IDB
QBAD_I_30	Bool	7.6	TRUE	Set in IDB
QBAD_I_31	Bool	7.7	TRUE	Set in IDB
QBAD_O_00	Bool	8.0	TRUE	Set in IDB
QBAD_O_01	Bool	8.1	TRUE	Set in IDB
QBAD_O_02	Bool	8.2	TRUE	Set in IDB
QBAD_O_03	Bool	8.3	TRUE	Set in IDB
QBAD_O_04	Bool	8.4	TRUE	Set in IDB
QBAD_O_05	Bool	8.5	TRUE	Set in IDB
QBAD_O_06	Bool	8.6	TRUE	Set in IDB
QBAD_O_07	Bool	8.7	TRUE	Set in IDB
QBAD_O_08	Bool	9.0	TRUE	Set in IDB
QBAD_O_09	Bool	9.1	TRUE	Set in IDB
QBAD_O_10	Bool	9.2	TRUE	Set in IDB
QBAD_O_11	Bool	9.3	TRUE	Set in IDB
QBAD_O_12	Bool	9.4	TRUE	Set in IDB
QBAD_O_13	Bool	9.5	TRUE	Set in IDB
QBAD_O_14	Bool	9.6	TRUE	Set in IDB
QBAD_O_15	Bool	9.7	TRUE	Set in IDB
QBAD_O_16	Bool	10.0	TRUE	Set in IDB
QBAD_O_17	Bool	10.1	TRUE	Set in IDB
QBAD_O_18	Bool	10.2	TRUE	Set in IDB
QBAD_O_19	Bool	10.3	TRUE	Set in IDB
QBAD_O_20	Bool	10.4	TRUE	Set in IDB
QBAD_O_21	Bool	10.5	TRUE	Set in IDB
QBAD_O_22	Bool	10.6	TRUE	Set in IDB
QBAD_O_23	Bool	10.7	TRUE	Set in IDB
QBAD_O_24	Bool	11.0	TRUE	Set in IDB
QBAD_O_25	Bool	11.1	TRUE	Set in IDB
QBAD_O_26	Bool	11.2	TRUE	Set in IDB
QBAD_O_27	Bool	11.3	TRUE	Set in IDB
QBAD_O_28	Bool	11.4	TRUE	Set in IDB
QBAD_O_29	Bool	11.5	TRUE	Set in IDB
QBAD_O_30	Bool	11.6	TRUE	Set in IDB
QBAD_O_31	Bool	11.7	TRUE	Set in IDB
InOut				
Static				
▼ Temp				
LB0_TMP	Byte	0.0		
CTRL2_DB_T	Word	2.0		
GLOB_DB_T	Word	4.0		
DIFF_T	DInt	6.0		
I00_07_TMP	Byte	10.0		
I08_15_TMP	Byte	11.0		
I16_23_TMP	Byte	12.0		
I24_31_TMP	Byte	13.0		
I32_39_TMP	Byte	14.0		
I40_47_TMP	Byte	15.0		

Name	Data type	Offset	Default value	Retain
I48_55_TMP	Byte	16.0		
I56_63_TMP	Byte	17.0		
U_INT00_TMP	Int	18.0		
U_INT01_TMP	Int	20.0		
U_INT02_TMP	Int	22.0		
U_INT03_TMP	Int	24.0		
U_INT04_TMP	Int	26.0		
U_INT05_TMP	Int	28.0		
U_DINT00_TMP	DInt	30.0		
U_DINT01_TMP	DInt	34.0		
U_DINT02_TMP	DInt	38.0		
▼ RD_FMODUL	Struct	42.0		
BYTE_00_TMP	Byte	0.0		
BYTE_01_TMP	Byte	1.0		
BYTE_02_TMP	Byte	2.0		
BYTE_03_TMP	Byte	3.0		
BYTE_04_TMP	Byte	4.0		
BYTE_05_TMP	Byte	5.0		
BYTE_06_TMP	Byte	6.0		
BYTE_07_TMP	Byte	7.0		
BYTE_08_TMP	Byte	8.0		
BYTE_09_TMP	Byte	9.0		
BYTE_10_TMP	Byte	10.0		
BYTE_11_TMP	Byte	11.0		
BYTE_12_TMP	Byte	12.0		
BYTE_13_TMP	Byte	13.0		
BYTE_14_TMP	Byte	14.0		
BYTE_15_TMP	Byte	15.0		
▼ WR_FMODUL	Struct	58.0		
I00_07_TMP	Byte	0.0		
I08_15_TMP	Byte	1.0		
I16_23_TMP	Byte	2.0		
I24_31_TMP	Byte	3.0		
I32_39_TMP	Byte	4.0		
I40_47_TMP	Byte	5.0		
I48_55_TMP	Byte	6.0		
I56_63_TMP	Byte	7.0		
BYTE8_11_TMP	DWord	8.0		
BYTE12_15_TMP	DWord	12.0		
RESERVE_BOOL_00_TMP	Bool	74.0		
TIMEOUT_HOST_AKT_TMP	Bool	74.1		
PROFISAFE_TMP	Bool	74.2		
OE_IN_TMP	Bool	74.3		
CHF_PAST_TMP	Bool	74.4		
VKE_TMP	Bool	74.5		
DIAG_KOM_TMP	Bool	74.6		
DIAG_GEH_TMP	Bool	74.7		
▼ INFO	Struct	76.0		
EVENTN_TMP	Word	0.0		
AE_TMP	Byte	2.0		
OB_NO_TMP	Byte	3.0		
TINFO2_TMP	Byte	4.0		

Name	Data type	Offset	Default value	Retain
TINFO1_TMP	Byte	5.0		
INFO1_TMP	Word	6.0		
INFO2_TMP	DWord	8.0		
CRC16_TMP	Word	88.0		
PLZ_LOK_TMP	Int	90.0		
CRC_PLUS_TMP	Int	92.0		
V_RD_FMODUL_CRC16_TMP	Word	94.0		
FMODUL_ANY_T	Any	96.0		
SCHLEIFE_TMP	Int	106.0		
RD_LEBEN_T	Byte	108.0		
TRACER_T	Word	110.0		
CRC_PLUS24_TMP	DWord	112.0		
INFO_IN_INT_T	Struct	116.0		
CH_IN_0	Bool	0.0		
CH_IN_1	Bool	0.1		
CH_IN_2	Bool	0.2		
CH_IN_3	Bool	0.3		
CH_IN_4	Bool	0.4		
CH_IN_5	Bool	0.5		
CRC_IMP16_TMP	Word	118.0		
QUIT_BYTE_TMP	Byte	120.0		
RESERVE_BOOL_01_TMP	Bool	121.0		
LOOP_BOOL_OUT_TMP	Bool	121.1		
NEW_CONS_NUM_TMP	Bool	121.2		
CPU_IS_PLCSIM_T	Bool	121.3		
RESERVE_T_01	Bool	121.4		
RESERVE_T_02	Bool	121.5		
RESERVE_T_03	Bool	121.6		
ACK_REI_GLOB_TMP	Bool	121.7		
Pointer_01_TMP	DWord	122.0		
Pointer_02_TMP	DWord	126.0		
CRC_IMP32_TMP	DWord	130.0		
V_RD_FMODUL_CRC32_TMP	DWord	134.0		
CRC24_TMP	DWord	138.0		
INT_01_TMP	Int	142.0		
INT_02_TMP	Int	144.0		
INT_03_TMP	Int	146.0		
DINT_01_TMP	DInt	148.0		
DINT_02_TMP	DInt	152.0		
DINT_03_TMP	DInt	156.0		
DINT_04_TMP	DInt	160.0		
LOOP_IST_TMP	DWord	164.0		
LOOP_SOLL_TMP	DWord	168.0		
QBAD_MASK_TMP	DWord	172.0		
VKE_TMP_C	Int	176.0		
N_SPL_TMP_C	DInt	178.0		
PROFISAFE_TMP_C	Int	182.0		
OE_IN_TMP_C	Int	184.0		
VKE_2_TMP_C	Int	186.0		
TIMEOUT_HOST_AKT_TMP_C	Int	188.0		
CRC_TMP_C	DInt	190.0		
RESERVE_INT01_TMP_C	DInt	194.0		

Name	Data type	Offset	Default value	Retain
DIFF_T_L	Word	198.0		
DIFF_T_H	DInt	200.0		
RED_WR_FMODUL_V2	Struct	204.0		
CONS_NUM_SEND_TMP	DInt	0.0		
RED_WR_FMODUL	Struct	4.0		
I00_07_TMP	Byte	0.0		
I08_15_TMP	Byte	1.0		
I16_23_TMP	Byte	2.0		
I24_31_TMP	Byte	3.0		
I32_39_TMP	Byte	4.0		
I40_47_TMP	Byte	5.0		
I48_55_TMP	Byte	6.0		
I56_63_TMP	Byte	7.0		
BYTE8_11_TMP	DWord	8.0		
BYTE12_15_TMP	DWord	12.0		
STATUS_T_C	Struct	224.0		
BOOL0_C	Int	0.0		
BOOL1_C	Int	2.0		
BOOL2_C	Int	4.0		
BOOL3_C	Int	6.0		
BOOL4_C	Int	8.0		
BOOL5_C	Int	10.0		
BOOL6_C	Int	12.0		
BOOL7_C	Int	14.0		
INFO_IN_INT_T_C	Struct	240.0		
CH_IN_0	Int	0.0		
CH_IN_1	Int	2.0		
CH_IN_2	Int	4.0		
CH_IN_3	Int	6.0		
CH_IN_4	Int	8.0		
CH_IN_5	Int	10.0		
RESERVE_BOOL_01_TMP_C	Int	252.0		
QUIT_BYTE_TMP_C	DInt	254.0		
I00_07_TMP_C	DWord	258.0		
I08_15_TMP_C	DWord	262.0		
I16_23_TMP_C	DWord	266.0		
I24_31_TMP_C	DWord	270.0		
I32_39_TMP_C	DWord	274.0		
I40_47_TMP_C	DWord	278.0		
I48_55_TMP_C	DWord	282.0		
I56_63_TMP_C	DWord	286.0		
U_INT00_TMP_C	DInt	290.0		
U_INT01_TMP_C	DInt	294.0		
U_INT02_TMP_C	DInt	298.0		
U_INT03_TMP_C	DInt	302.0		
U_INT04_TMP_C	DInt	306.0		
U_INT05_TMP_C	DInt	310.0		
U_DINT00_TMP_L	Word	314.0		
U_DINT00_TMP_H	DInt	316.0		
U_DINT01_TMP_L	Word	320.0		
U_DINT01_TMP_H	DInt	322.0		

Name	Data type	Offset	Default value	Retain
U_DINT02_TMP_L	Word	326.0		
U_DINT02_TMP_H	DInt	328.0		
CHF_PAST_TMP_C	Int	332.0		
LOOP_BOOL_OUT_TMP_C	Int	334.0		
ACK_REI_GLOB_TMP_C	Int	336.0		
QBAD_MASK_TMP_C	DInt	338.0		
QBAD_MASK_INT_TMP_C	Int	342.0		

Program blocks / System blocks / STEP 7 Safety / Compiler blocks

F_FENGDB [DB8000]

F_FENGDB Properties					
General					
Name	F_FENGDB	**Number**	8000	**Type**	DB.Safety
Language	DB				
Information					
Title		**Author**		**Comment**	
Family	Safety11	**Version**	0.1	**User-defined ID**	F_FENGDB

Name	Data type	Start value	Retain
Static			

Program blocks / System blocks / STEP 7 Safety / F-IO data blocks

F00000_F-DI24xDC24V_1 [DB8002]

F00000_F-DI24xDC24V_1 Properties					
General					
Name	F00000_F-DI24xDC24V_1	**Number**	8002	**Type**	DB.Safety
Language	DB				
Information					
Title		**Author**	FRTG	**Comment**	
Family	F_DRIVER	**Version**	1.0	**User-defined ID**	FDRI0000

Name	Data type	Start value	Retain
▼ Input			
PASS_ON	Bool	false	False
ACK_NEC	Bool	TRUE	False
ACK_REI	Bool	false	False
IPAR_EN	Bool	false	False
▼ Output			
PASS_OUT	Bool	TRUE	False
QBAD	Bool	TRUE	False
ACK_REQ	Bool	false	False
IPAR_OK	Bool	false	False
DIAG	Byte	16#0	False
QBAD_I_00	Bool	TRUE	False
QBAD_I_01	Bool	TRUE	False
QBAD_I_02	Bool	TRUE	False
QBAD_I_03	Bool	TRUE	False
QBAD_I_04	Bool	TRUE	False
QBAD_I_05	Bool	TRUE	False
QBAD_I_06	Bool	TRUE	False
QBAD_I_07	Bool	TRUE	False
QBAD_I_08	Bool	TRUE	False
QBAD_I_09	Bool	TRUE	False
QBAD_I_10	Bool	TRUE	False
QBAD_I_11	Bool	TRUE	False
QBAD_I_12	Bool	TRUE	False
QBAD_I_13	Bool	TRUE	False
QBAD_I_14	Bool	TRUE	False
QBAD_I_15	Bool	TRUE	False
QBAD_I_16	Bool	TRUE	False
QBAD_I_17	Bool	TRUE	False
QBAD_I_18	Bool	TRUE	False
QBAD_I_19	Bool	TRUE	False
QBAD_I_20	Bool	TRUE	False
QBAD_I_21	Bool	TRUE	False
QBAD_I_22	Bool	TRUE	False
QBAD_I_23	Bool	TRUE	False
QBAD_I_24	Bool	TRUE	False
QBAD_I_25	Bool	TRUE	False
QBAD_I_26	Bool	TRUE	False

Name	Data type	Start value	Retain
QBAD_I_27	Bool	TRUE	False
QBAD_I_28	Bool	TRUE	False
QBAD_I_29	Bool	TRUE	False
QBAD_I_30	Bool	TRUE	False
QBAD_I_31	Bool	TRUE	False
QBAD_O_00	Bool	TRUE	False
QBAD_O_01	Bool	TRUE	False
QBAD_O_02	Bool	TRUE	False
QBAD_O_03	Bool	TRUE	False
QBAD_O_04	Bool	TRUE	False
QBAD_O_05	Bool	TRUE	False
QBAD_O_06	Bool	TRUE	False
QBAD_O_07	Bool	TRUE	False
QBAD_O_08	Bool	TRUE	False
QBAD_O_09	Bool	TRUE	False
QBAD_O_10	Bool	TRUE	False
QBAD_O_11	Bool	TRUE	False
QBAD_O_12	Bool	TRUE	False
QBAD_O_13	Bool	TRUE	False
QBAD_O_14	Bool	TRUE	False
QBAD_O_15	Bool	TRUE	False
QBAD_O_16	Bool	TRUE	False
QBAD_O_17	Bool	TRUE	False
QBAD_O_18	Bool	TRUE	False
QBAD_O_19	Bool	TRUE	False
QBAD_O_20	Bool	TRUE	False
QBAD_O_21	Bool	TRUE	False
QBAD_O_22	Bool	TRUE	False
QBAD_O_23	Bool	TRUE	False
QBAD_O_24	Bool	TRUE	False
QBAD_O_25	Bool	TRUE	False
QBAD_O_26	Bool	TRUE	False
QBAD_O_27	Bool	TRUE	False
QBAD_O_28	Bool	TRUE	False
QBAD_O_29	Bool	TRUE	False
QBAD_O_30	Bool	TRUE	False
QBAD_O_31	Bool	TRUE	False
InOut			
Static			

Program blocks / System blocks / STEP 7 Safety / F-IO data blocks

F00010_F-DO10xDC24V_1 [DB8003]

F00010_F-DO10xDC24V_1 Properties					
General					
Name	F00010_F-DO10xDC24V_1	**Number**	8003	**Type**	DB.Safety
Language	DB				
Information					
Title		**Author**	FRTG1F47	**Comment**	
Family	F_DRIVER	**Version**	1.0	**User-defined ID**	FDRI000A

Name	Data type	Start value	Retain
▼ Input			
PASS_ON	Bool	false	False
ACK_NEC	Bool	TRUE	False
ACK_REI	Bool	false	False
IPAR_EN	Bool	false	False
▼ Output			
PASS_OUT	Bool	TRUE	False
QBAD	Bool	TRUE	False
ACK_REQ	Bool	false	False
IPAR_OK	Bool	false	False
DIAG	Byte	16#0	False
QBAD_I_00	Bool	TRUE	False
QBAD_I_01	Bool	TRUE	False
QBAD_I_02	Bool	TRUE	False
QBAD_I_03	Bool	TRUE	False
QBAD_I_04	Bool	TRUE	False
QBAD_I_05	Bool	TRUE	False
QBAD_I_06	Bool	TRUE	False
QBAD_I_07	Bool	TRUE	False
QBAD_I_08	Bool	TRUE	False
QBAD_I_09	Bool	TRUE	False
QBAD_I_10	Bool	TRUE	False
QBAD_I_11	Bool	TRUE	False
QBAD_I_12	Bool	TRUE	False
QBAD_I_13	Bool	TRUE	False
QBAD_I_14	Bool	TRUE	False
QBAD_I_15	Bool	TRUE	False
QBAD_I_16	Bool	TRUE	False
QBAD_I_17	Bool	TRUE	False
QBAD_I_18	Bool	TRUE	False
QBAD_I_19	Bool	TRUE	False
QBAD_I_20	Bool	TRUE	False
QBAD_I_21	Bool	TRUE	False
QBAD_I_22	Bool	TRUE	False
QBAD_I_23	Bool	TRUE	False
QBAD_I_24	Bool	TRUE	False
QBAD_I_25	Bool	TRUE	False
QBAD_I_26	Bool	TRUE	False

Name	Data type	Start value	Retain
QBAD_I_27	Bool	TRUE	False
QBAD_I_28	Bool	TRUE	False
QBAD_I_29	Bool	TRUE	False
QBAD_I_30	Bool	TRUE	False
QBAD_I_31	Bool	TRUE	False
QBAD_O_00	Bool	TRUE	False
QBAD_O_01	Bool	TRUE	False
QBAD_O_02	Bool	TRUE	False
QBAD_O_03	Bool	TRUE	False
QBAD_O_04	Bool	TRUE	False
QBAD_O_05	Bool	TRUE	False
QBAD_O_06	Bool	TRUE	False
QBAD_O_07	Bool	TRUE	False
QBAD_O_08	Bool	TRUE	False
QBAD_O_09	Bool	TRUE	False
QBAD_O_10	Bool	TRUE	False
QBAD_O_11	Bool	TRUE	False
QBAD_O_12	Bool	TRUE	False
QBAD_O_13	Bool	TRUE	False
QBAD_O_14	Bool	TRUE	False
QBAD_O_15	Bool	TRUE	False
QBAD_O_16	Bool	TRUE	False
QBAD_O_17	Bool	TRUE	False
QBAD_O_18	Bool	TRUE	False
QBAD_O_19	Bool	TRUE	False
QBAD_O_20	Bool	TRUE	False
QBAD_O_21	Bool	TRUE	False
QBAD_O_22	Bool	TRUE	False
QBAD_O_23	Bool	TRUE	False
QBAD_O_24	Bool	TRUE	False
QBAD_O_25	Bool	TRUE	False
QBAD_O_26	Bool	TRUE	False
QBAD_O_27	Bool	TRUE	False
QBAD_O_28	Bool	TRUE	False
QBAD_O_29	Bool	TRUE	False
QBAD_O_30	Bool	TRUE	False
QBAD_O_31	Bool	TRUE	False
InOut			
Static			

Program blocks / System blocks

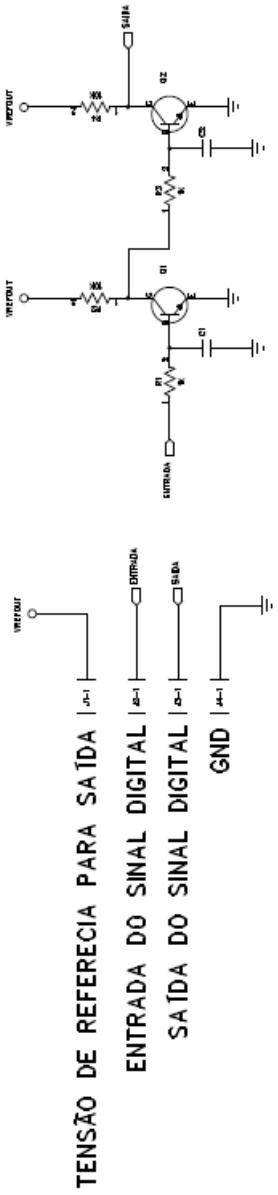
Program resources

This folder is empty.

Anexos

ANEXO A – Circuito Interface IO

Figura 39 – Circuito Interface IO da HIL



Fonte: Equipe Doppler de Desenvolvimento

ANEXO B – Lista de Materiais para Interface IO

Tabela 18 – Lista de Materiais para Interface IO da HIL

Quant.	Descrição
2	Borne vermelho para pino Banana
8	Borne preto para pino Banana
64	Borne amarelo para pino Banana
32	Transistor BC337-25
32	Resistor 1/8w 10K
32	Resistor 1/8w 1K
32	Capacitor cerâmico 100nF 50V
1	Caixa patola PLB-280

Fonte: Equipe Doppler de Desenvolvimento

ANEXO C – Métodos Utilizados

Texto extraído de (SANTOS, 2013), como material complementar e que descreve o funcionamento de algumas das ferramentas utilizadas durante o trabalho, com Redes de Petri e Production Flow Schema.

C.1 Topologia de Sistemas de Controle

Segundo (DILTS; BOYD; WHORMS, 1991), arquitetura de um sistema é definida como a estrutura e organização lógica de funcionamento de um sistema, onde estão determinadas as inter-relações entre os componentes do sistema e onde cada componente possui uma função específica de controle. Desta forma, os sistemas podem ser classificados em dois grupos em função da topologia adotada para efetuar o controle, são elas:

1 - Sistema de controle Centralizado: São sistemas que possuem um único controlador onde, toda a lógica de controle do sistema é processada neste controlador. Este sistema, normalmente, é aplicado a pequenos processos com menor número de entradas e saídas. A Figura 40 apresenta o esquema de uma arquitetura de controle do tipo centralizada;

2 - Sistema de controle Distribuído: Neste tipo de sistema há dois ou mais controladores, envolvidos na execução do controle. Neste contexto, os controladores podem estar organizados de três formas básicas (Figura 40):

Arquitetura distribuída do tipo hierárquica;

Arquitetura distribuída do tipo hierárquica modificada;

Arquitetura totalmente distribuída.

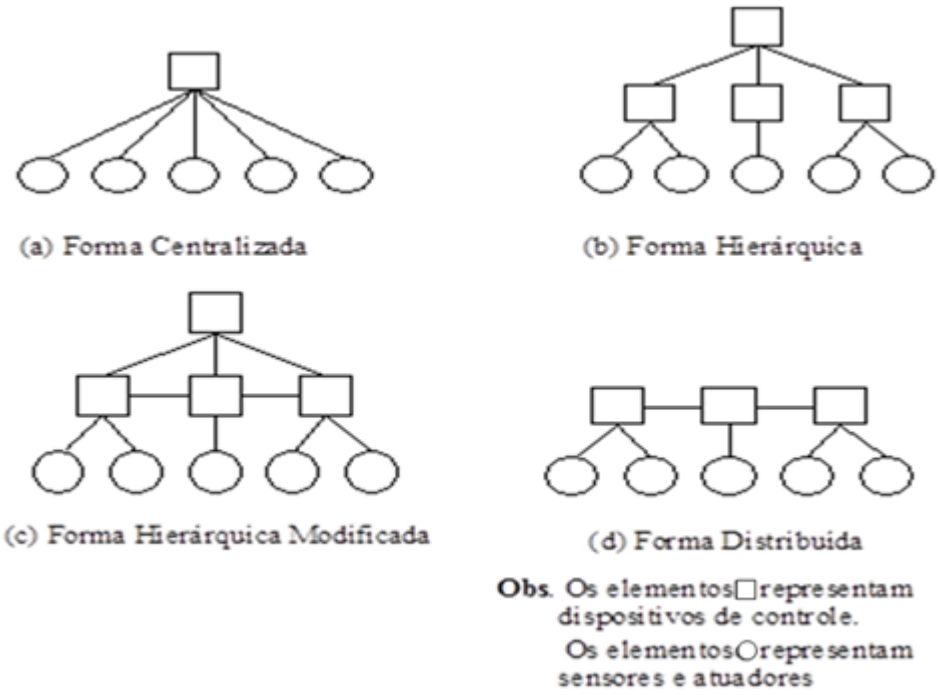
C.2 Modelagem Utilizando o PFS e a Rede de Petri

C.2.1 Rede de Petri

A rede de Petri é um modelo matemático proposto por Carl Petri para a modelagem de sistemas de comunicação. Basicamente uma rede de Petri é representada por um grafo direcionado, bi-partido, ponderado com uma marcação inicial, composto por nós (denominados lugares) e transições onde estes são interligados por arcos orientados. Os arcos possuem um peso k que representa k aros em paralelos. As marcações do da rede são atribuídas aos lugares. Uma rede de Petri é uma sêxtupla (P, T, Ar, K, W, M0), sendo que:

P=p1, p2, p3,... pm é um conjunto finito de lugares;

Figura 40 – Formas básicas de arquiteturas de controle de SP



Fonte: (SANTOS, 2013)

T=t1, t2, t3,... tm é um conjunto finito de transições;

Ar $\subseteq$ (PxT) $\cup$ (TxP) é um conjunto finito de arcos;

K : P $\rightarrow$ N $\cup$ (∞) é a função capacidade;

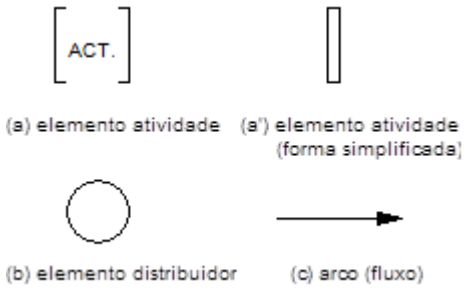
M0 : P $\rightarrow$ N é a função de marcação inicial que satisfaz $\forall$ p $\in$ P : M0(p) $\leq$ K(p).

C.2.2 Production Flow Schema (PFS) e seus elementos

O PFS (MIYAGI, 2007) é utilizado para descrever, graficamente e conceitualmente, os processos relacionados com a produção de itens (peças, produtos, informações, etc.) sob a forma de sequências de etapas de atividades e de distribuição. Como indicado na Figura 41, o PFS é um grafo bi-partido composto de nós de elementos-atividade, nós de elementos distribuidores e arcos de fluxo, que conectam sequencialmente um tipo de nó ao outro. Os elementos-atividade, como ilustrado na Figura 42, podem ser expandidos em duas transições e um lugar (lugar-atividade). Quando é necessária a indicação do início e da conclusão de uma atividade, distinguimos a transição de entrada (como a transição de início) da transição de saída (como a transição final) (Figura 42 - c). Além disso, na Figura 42, m e n são, respectivamente, os números de entradas e saídas simultâneas de um elemento; em rede de Petri tipo lugar/transição - L/T - ou em representações híbridas,

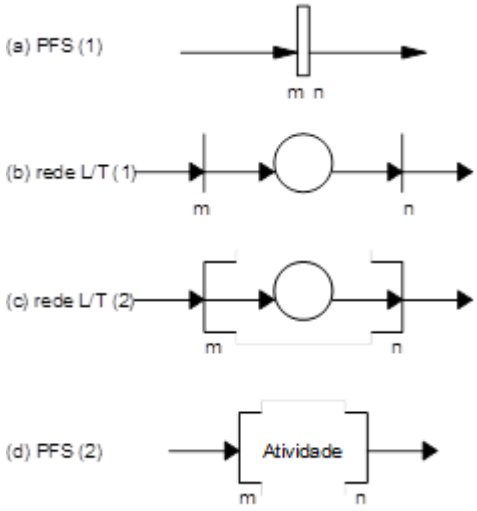
eles compõem os pesos dos arcos de fluxo. Se não houver necessidade, ou m=n=1, eles podem ser omitidos.

Figura 41 – Elementos do PFS



Fonte: (SANTOS, 2013)

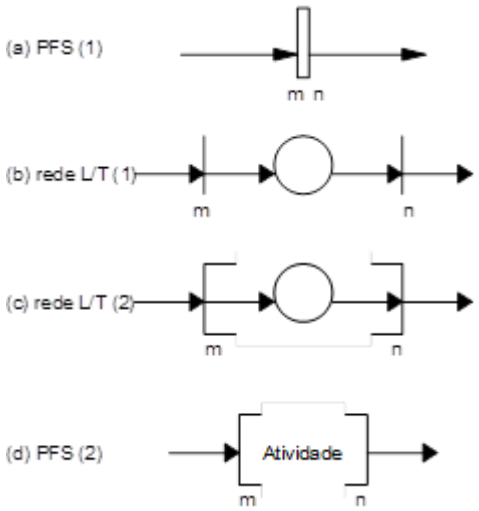
Figura 42 – Elemento atividade em rede L/T



Fonte: (SANTOS, 2013)

Similarmente, um elemento-distribuidor pode ser também expandido em um lugar (lugar-distribuidor) com transições à entrada e à saída, como ilustrado na Figura 43. Ainda, no caso de se expandir dois elementos conectados um ao outro em uma rede L/T, a transição de saída da primeira etapa e a de entrada da etapa seguinte formam uma única transição. Deste ponto em diante, a denominação lugares-etapa é dada tanto para os lugares-atividade como para os lugares-distribuidores

Figura 43 – Elemento distribuidor em rede L/T



Fonte: (SANTOS, 2013)

C.3 Metodologia PFS/Rede de Petri

A rede de Petri pode ser obtida a partir de modelos em PFS **??**. De acordo com a metodologia PFS/rede de Petri, para obter uma rede de Petri a partir de um modelo em PFS devem ser seguidos os seguintes passos:

a) Criar um modelo conceitual e onde cada processo apresente um alto nível de abstração;

b) Representar cada processo como um conjunto de atividades;

c) Definir um conjunto de atributos que pode ser associado a cada marca no modelo em rede de Petri;

d) Detalhar cada atividade do processo descrito em PFS usando modelos em rede de Petri;

e) Mapear os eventos gerados pelos controladores (arcos que conectam os lugares da rede de Petri aos atuadores do processo) e os eventos gerados pelo processo (através de arcos habilitadores e inibidores da rede de Petri);

Portanto, a metodologia PFS/rede de Petri permite desenvolver de forma natural e organizada modelos de controle de processos a partir de sistemas que apresentam alta complexidade e uma variedade de processos simultâneos.

C.4 Formalismo da Rede Bayesiana

A rede Bayesiana e um modelo matemático formal de representação de conhecimento. Esta pode ser representada através de um grafo, onde os nós representam variáveis aleatórias e os arcos direcionados representam relacionamentos causais diretos entre os nós que conectam. Em uma rede Bayesiana, cada nó Xi, representado através de um círculo, é condicionalmente independente de qualquer subconjunto de nós que não são seus descendentes, formando assim um grafo acíclico orientado. Os nós de origem são conhecidos como os nós pais de Xi (representados por pa(Xi)). A associação entre os nós define uma estrutura de nós pais e filhos associados aos pais, e é feita através de arcos orientados e representa as relações de causa e efeito, descritas por P(Xi | pa(Xi)). Normalmente, tais relações são fornecidas em formato de tabela, as quais são chamadas de tabelas de probabilidade condicional (COOPER; HERKSKOVITS, 1992). A probabilidade condicional (PrC) numericamente quantifica a relação causa e efeito. Assim, considerando-se X1, X2,..., Xn como os nós de uma rede Bayesiana, e tomando-se da estrutura desta rede as situações onde se tem independência condicional, permite-se determinar a probabilidade conjunta de todos os nós (RUSSEL et al., 2003).

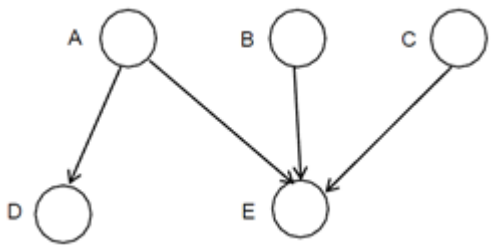
$$P(X1, X2, ..., Xn) = \prod P(Xi|pa(Xi)) \tag{C.1}$$

V = X1, X2, ..., Xn corresponde a um conjunto finito de vértices (nós) da rede, representados por círculos;

F corresponde a um conjunto finito de arcos orientados também chamados de relações de fluxo. Pode-se dizer que G (V, F) é uma estrutura que representa um grafo acíclico orientado;

P é a distribuição de probabilidade entre os nós da rede.

Figura 44 – Estrutura de uma Rede Bayesiana



Fonte: (SANTOS, 2013)

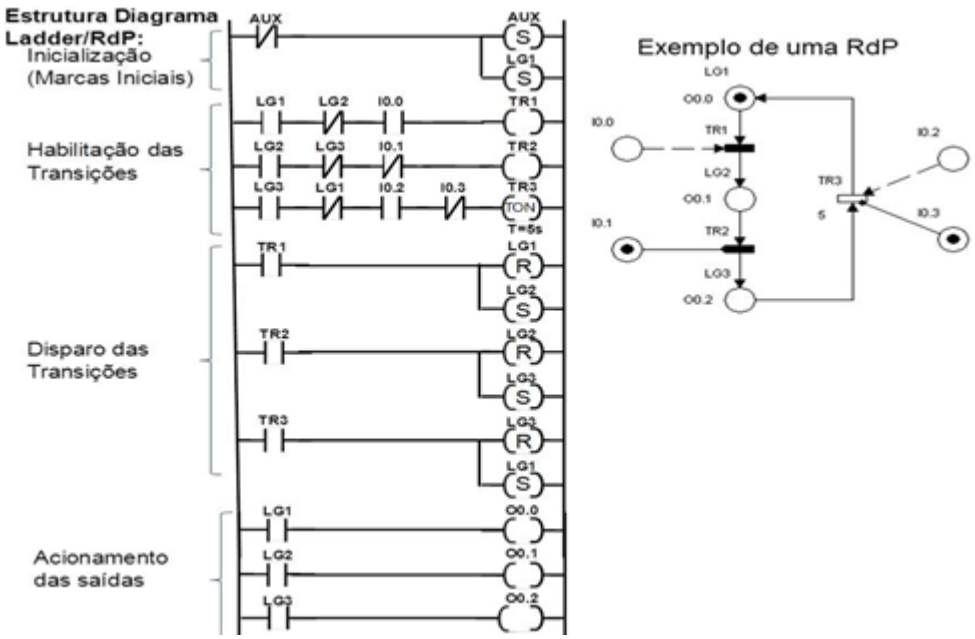
Na Figura 44 é apresentado um exemplo de uma rede Bayesiana. Neste exemplo, a estrutura da rede Bayesiana estabelece a influência causal das variáveis A, B e C (nós

pais) sobre as variáveis D e E (nós filhos). O conjunto V = A, B, C, D, E é o conjunto de variáveis do sistema.

C.5 Conversão de uma rede de petri para linguagem de programação padronizada pela norma IEC61131-3

Uma vez executada a padronização do software de controle, é possível executar a conversão de módulos de controle em rede de Petri Interpretada para permitir a implementação deste em Controladores Programáveis (CPs) que atendam as linguagens de programação estabelecidas pela IEC61131-3. Para efetuar esta conversão de forma isomórfica utiliza-se o procedimento de substituição de cada elemento da rede de Petri Interpretada por elementos do diagrama Ladder conforme apresenta a Figura 45 (WIGHTKIN; BUY; DARABI, 2010).

Figura 45 – Exemplo de um diagrama Ladder obtido a partir de uma rede de petri

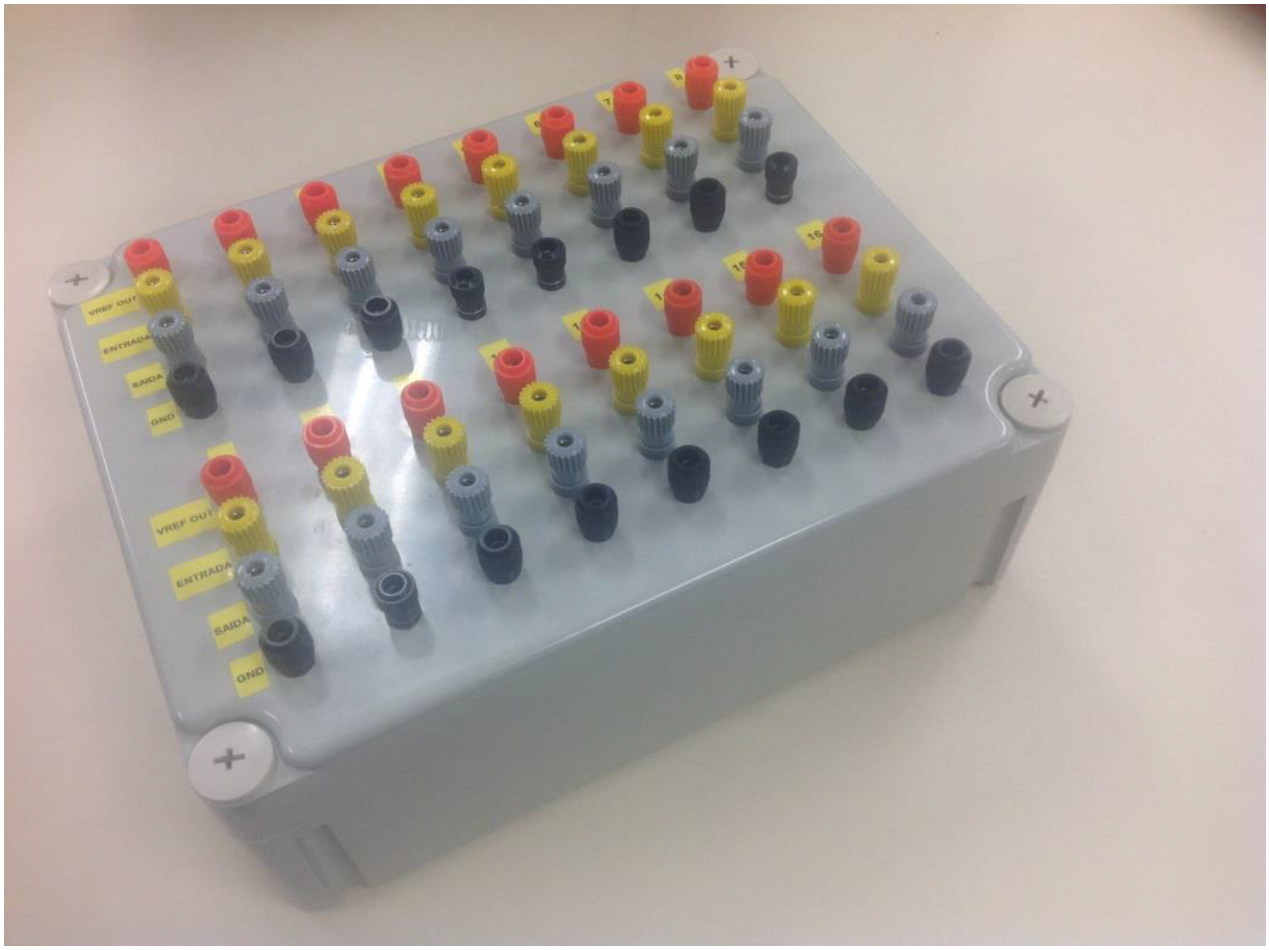


Fonte: (SANTOS, 2013)

Cada etapa do programa deve, necessariamente, estar disposta conforme o seguinte critério: primeiro as transições, depois as etapas e por último as ações. Obedecendo esta regra, obtém-se como resultado um diagrama Ladder que pode ser implementado em qualquer CP que possua a linguagem de programação Ladder padronizada pela IEC61131-3. Devido à criação de um elevado número de elementos Ladder, após a tradução pode ser realizado um agrupamento ou simplificação de funções, tornando assim o programa mais simples para implementação de fato no CP.

ANEXO D – Interface IO para HIL

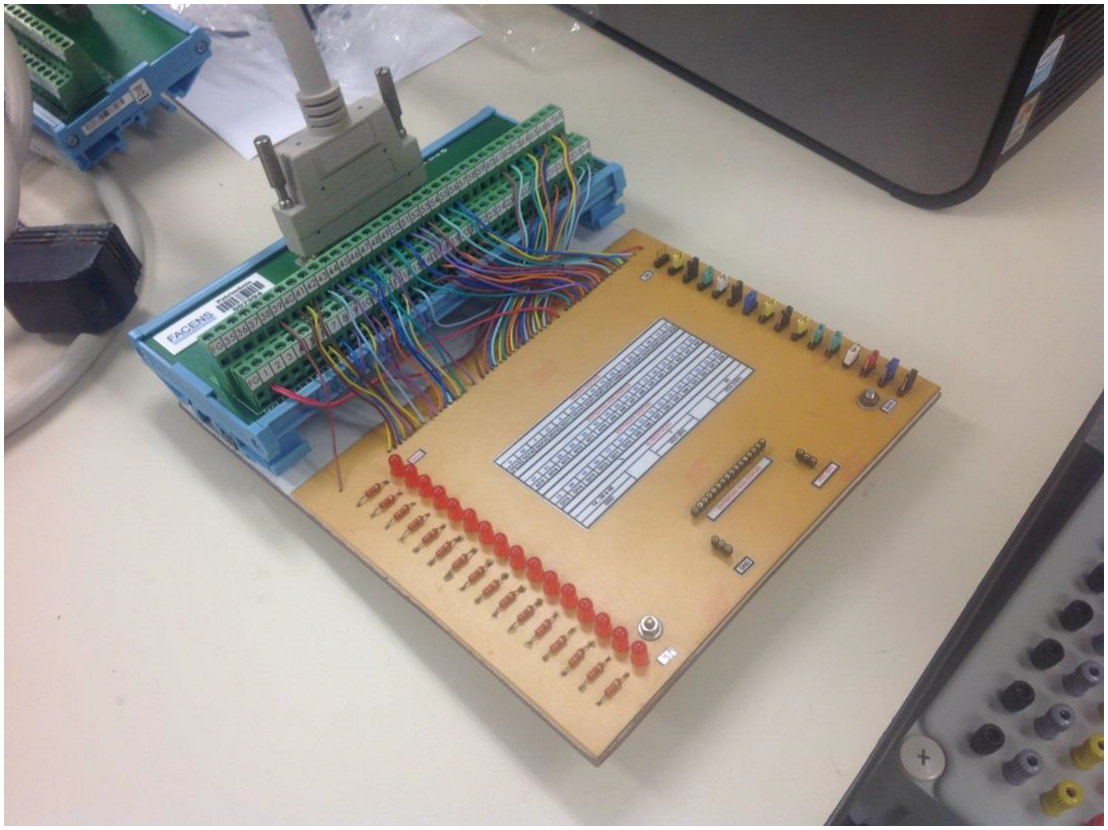
Figura 46 – Interface IO para HIL



Fonte: Equipe Doppler de Desenvolvimento

ANEXO E – Placa de Aquisição de Sinais HIL

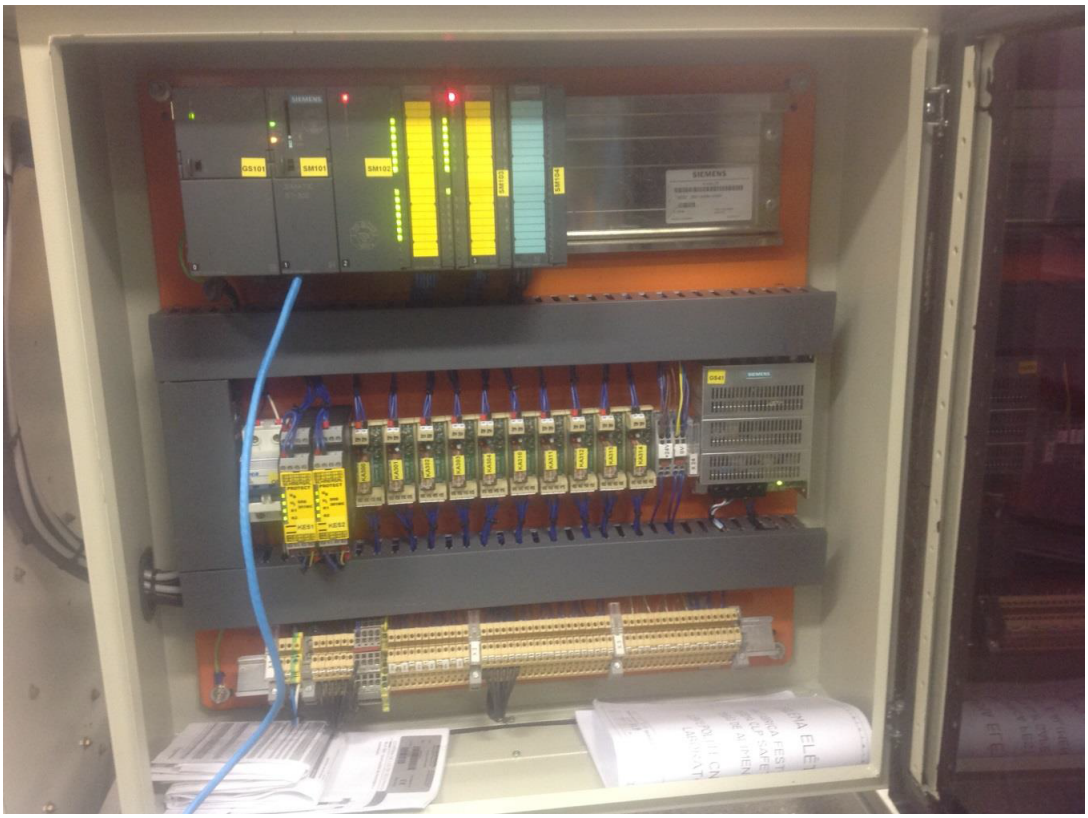
Figura 47 – Placa de Aquisição de Sinais HIL



Fonte: Advantech

ANEXO F – CLP de Segurança

Figura 48 – CLP de Segurança



Fonte: Equipe Doppler de Desenvolvimento