MAGNO TÚLIO ANDRADE MEIRELES

UMA ABORDAGEM DE ANÁLISE DE SEGURANÇA DE SOFTWARE PARA CONTROLE DE TEMPERATURA NA PRODUÇÃO DE AÇO

Trabalho de conclusão de curso apresentado ao Curso de Ciência da Computação.

UNIVERSIDADE PRESIDENTE ANTÔNIO CARLOS

Orientador/ Professor Elio Lovisi Filho

BARBACENA
2004

MAGNO TÚLIO ANDRADE MEIRELES

UMA ABORDAGEM DE ANÁLISE DE SEGURANÇA DE SOFTWARE PARA CONTROLE DE TEMPERATURA NA PRODUÇÃO DE AÇO

Este trabalho de conclusão de curso foi julgado adequado à obtenção do grau de Bacharel em Ciência da Computação e aprovado em sua forma final pelo Curso de Ciência da Computação da Universidade Presidente Antônio Carlos.

Barbacena – MG, 26 de Junho de 2004.

____

Prof. Elio Lovisi Filho- Orientador do Trabalho

____

Prof. Gustavo Campos Menezes- Membro da Banca Examinadora

____

Prof. Lorena Sophia C. de Oliveira - Membro da Banca Examinadora

AGRADECIMENTOS

Agradeço a Deus por mais uma etapa vencida, aos meus pais, "Marcos Túlio Meireles e Isabel Maria das Graças A Meireles", irmãos e minha mocinha "Bruna" dedico esta vitória. Aos meus colegas de trabalho e orientador "Elio Lovisi Filho" meus agradecimentos pela colaboração.

RESUMO

Este trabalho apresenta uma Abordagem de análise de Segurança de Software para Controle de Temperatura na Produção de Aço quanto à segurança e confiabilidade de software. Tem por objetivo estruturar um modelo que considere o conceito de segurança no processo de desenvolvimento, especificamente, na fase de análise de segurança de software, alem de definir e utilizar ferramentas para modelagem da segurança do software tais como: SFTA, FMEA e FMECA. A proposição foi aplicada no projeto do sistema de controle de temperatura de aço de uma usina, na cidade de Ouro Branco, estado de Minas Gerais.

Palavras-chave: Segurança de Software, Temperatura, FMECA

SUMÁRIO

FIGURAS

TABELAS

1 INTRODUÇÃO

O conforto e desenvolvimento trazidos pela industrialização produziram um aumento considerável no número de acidentes, ou ainda das anormalidades durante um processo devido à obsolescência de equipamentos, máquinas cada vez mais sofisticadas.

Com a preocupação e a necessidade de dar maior atenção ao ser humano, principal bem de uma organização, além de buscar uma maior eficiência, nasceu a Engenharia de Segurança de Sistemas.

A Engenharia de Segurança de Sistema, surgiu com o crescimento e necessidade de segurança total em áreas como aeronáutica, aeroespacial, nuclear e na área industria, trazendo valiosos instrumentos para a solução de problemas ligados à segurança.

Com a difusão dos conceitos de segurança, falha e confiabilidade, as metodologias e técnicas aplicadas pela segurança de sistema, inicialmente utilizada somente nas áreas militar e espacial, tiveram a partir da década de 70 uma aplicação quase que universal na solução de problemas de engenharia em geral.

1.1 OBJETIVOS

Através da Abordagem de Análise da Segurança de Software para o controle de Temperatura do será possível evidenciar a importância em conhecer os princípios e as correlações existentes no atributo confiabilidade e segurança . Para tal, estruturou-se um modelo que considere o conceito de segurança no processo de desenvolvimento, especificamente, na fase de análise de segurança de software, alem de definir como utilizar ferramentas para modelagem da segurança do software envolvendo o Controle de Temperatura no Aço.

1.2 JUSTIFICATIVAS

Tendo em mente a gravidade de um acidente com Aço Líquido e perdas econômicas elevadas com esse fato, foi elaborada Uma abordagem de Segurança de Software para Controle de Temperatura no Aço, visando seu uso na melhoria da qualidade do processo e segurança do ser humano que é o maior bem de uma organização.

1.3 VISÃO GERAL

Neste capítulo, descreve-se a introdução, o objetivo e a justificativa deste trabalho de pesquisa.

No Capítulo 2, apresenta-se uma pesquisa bibliográfica sobre a Segurança de Software Crítico para Controle de Temperatura do Aço. Nele, descreve-se os principais conceitos sobre Falha de Software, Análise de Segurança do Software e Técnicas para Análise de Segurança de Software.

No Capítulo 3, apresenta-se uma descrição sobre o Processo de Fabricação do Aço; onde serão evidenciados seus principais conceitos e características e os pontos onde é necessário o Controle da Segurança.

No Capítulo 4, apresenta-se a Análise de Segurança para o Sistema de Controle de Temperatura do Aço, onde será analisado e evidenciado a Lista Preliminar de Insegurança, Análise Preliminar de Insegurança, Análise de Insegurança de Subsistemas, Análise de Insegurança de sistemas e a Análise de Insegurança na Operação e Suporte.

No Capítulo 5, descreve-se a conclusão após a elaboração deste trabalho final de curso.

2 SEGURANÇA DE SOFTWARE CRÍTICO

Neste Capítulo, inicialmente, apresenta-se os principais conceitos relacionados com Segurança de Sistemas, Software com Componente de Sistema, Desenvolvimento de Software, Falhas de Software, conceitos sobre as ferramentas SFTA, FMECA e FMEA.

2.1 INTRODUÇÃO

A segurança de um sistema crítico computadorizado deve ser avaliada como um todo, considerando-se inclusive seu ambiente de operação, não sendo conveniente isolar o software e considera-lo à parte . Os erros decorrentes do software ou em conjunção com outros fatores podem levar o sistema a uma condição insegura, ou seja, aquela em que o sistema está exposto a um acidente.

Por outro lado existe o conceito de risco, que tem sido definido de acordo com a seguinte graduação [SCA1999]:

- Probabilidade de se atingir um estado inseguro;

- Probabilidade que um estado inseguro possa levar o sistema a um acidente; e

- Avaliação da pior conseqüência associada ao acidente.

A engenharia de Segurança de Sistemas enfoca, dentro destas perspectivas, a avaliação do nível de risco, considerando-o aceitável ou não, utilizando–se de técnicas de engenharia do ponto de vista gerencial e científico. Dentro desta abordagem destacam-se as técnicas de engenharia de software, combinadas com técnicas apropriadas para sistemas críticos e técnicas especiais para software utilizados em sistemas críticos.

O desenvolvimento de Software Crítico para controle de temperatura do aço exige a garantia de altos níveis de segurança, pois a probabilidade de ocorrência de uma falha nestes Sistemas pode acarretar danos aos equipamentos, financeiros e até mesmo ao ser humano.

2.1.1 SISTEMA CRÍTICO

Um Sistema constitui-se de diversos componentes, que podem ser de diferentes naturezas. Cada um desses componentes realiza uma tarefa específica, normalmente interagindo com os outros para alcançar um objetivo comum. Pode-se ainda considerar cada componente, como um Sistema menor ou um Subsistema [LOV1999].

Um Sistema Crítico apresenta restrições relacionadas a um determinado fator como, por exemplo: financeiro, tempo, capacidade do equipamento e segurança. Sistemas para controle de temperatura , exemplificam Sistemas Críticos quanto a Segurança, onde um defeito na sua execução pode causar um acidente com danos à vida, à propriedade ou ao meio ambiente [LOV1999].

No contexto deste trabalho de pesquisa apresentado apenas os Sistemas Críticos quanto à Segurança. A partir deste ponto, para esta dissertação, qualquer referência a Sistemas Críticos relaciona-se somente a Sistemas Críticos quanto à Segurança.

Até a década de 1970, havia relutância em utilizar-se o Computador para o controle de Sistemas Críticos. Pôr ser o mesmo muito complexo e pouco dominado até então, o computador era ainda pouco utilizado para exercer funções críticas.

O aumento de desempenho; a eficácia na execução de tarefas; a redução de custos; e a versatilidade de sua utilização é os principais argumentos para a agregação de computadores aos sistemas críticos.

2.1.2 SEGURANÇA DE SISTEMAS

Um sistema crítico de segurança ou segurança de sistema pode ser definido como um sistema cuja falha pode resultar em ferimentos, na perda de uma vida ou em grande dano ambiente [LOV1999].

Um exemplo de sistema de segurança crítica é o controle de temperatura do aço, onde se tem como objetivo controlar a velocidade de produção e nível no distribuidor. Ressaltando que velocidade e peso no distribuidor é inversamente proporcional a temperatura; onde um eventual erro do sistema pode proporcionar o transbordamento de aço no distribuidor, o que iria causar sérios danos ao equipamento, alem de colocar em risco a vida dos operadores.

Um acidente geralmente origina-se da combinação de vários fatores. Ele é um mecanismo dinâmico, ativado por um Estado Inseguro ou Perigoso (hazard), que flui pelo sistema como uma seqüência lógica de eventos consecutivos e/ou concorrentes, até que ocorra uma perda. Devido a esse fato, podem existir diversas opções para interromper essa seqüência de eventos, a fim de evitar acidentes [LOV1999].

O grande problema em relação aos sistemas que envolvem segurança, e que têm o software como componente crítico está na falta de objetividade da avaliação de sua segurança e conseqüentemente, do seu reflexo dentro dos níveis aceitáveis de um sistema. Um primeiro aspecto é estabelecer os níveis de risco aceitáveis, que dependem de cada tipo de aplicação, enquanto que o segundo consiste em como demonstrar que estes níveis de risco vêm sendo atendidos, principalmente quando se envolve o software.

Deve-se observar que a incidência de acidentes em um sistema e, conseqüentemente, o seu nível de Segurança, relaciona-se diretamente com a complexidade do mesmo e com o grau de interação de seus componentes [LOV1999].

Deve-se ressaltar que, um Computador somente contribui para a ocorrência de um acidente, quando ele faz parte de um Sistema Crítico, principalmente ao exercer função de apoio ao controle [MOU1996].

2.1.3 O SOFTWARE COMO COMPONENTE DE SISTEMAS

Define-se Software como: uma seqüência de instruções que, quando executadas, produzem os resultados especificados e o desempenho desejado; as estruturas de dados necessárias para a realização destas instruções; e a documentação que descreve toda a operação e o desenvolvimento do mesmo [LOV1999].

Considera-se o Software como um elemento lógico de um Sistema Computadorizado. Sendo assim, ele possui características peculiares em relação aos demais componentes, tais como a escassez de padronização e de metodologias já consagradas para o seu desenvolvimento e Garantia de sua Segurança. Além disso, a inclusão do Software torna o Sistema mais complexo, podendo também adicionar erros sutis e difíceis de localizar-se [MOU1996].

Devido a estes fatores, considera-se o Software o componente de um Sistema que apresenta atualmente maiores dificuldades para avaliação e Garantia de Segurança, a exceção do Peopleware, pois não se tem controle sobre e ele é responsável pela parte operacional do Sistema. [LOV1999, MOU1996].

Um Software componente de um Sistema pode contribuir para a ocorrência de acidentes de duas formas distintas: fornecendo valores errados ou fora do tempo ao Sistema; e falhando ao reconhecer erros de outros componentes, que necessitem de tratamento especial [LOV1999].

O Software componente de um Computador inserido em um Sistema Crítico, denomina-se Software Crítico. Leveson analisa as conseqüências da utilização dos Softwares Críticos da seguinte forma:

"Antes que o Software fosse usado em Sistemas Críticos quanto a Segurança eles freqüentemente eram controlados por dispositivos mecânicos e eletrônicos (não-programáveis) convencionais. Técnicas de segurança de Sistemas são projetadas para lidar com falhas aleatórias nesses Sistemas (não-programáveis). Erros humanos em projetos não são considerados uma vez que todas as falhas causadas por erros humanos podem ser completamente evitadas ou suprimidas antes da entrega e operação" [MOU1996].

A quase totalidade dos Softwares Críticos é também de Tempo Real. Um Software deste tipo deve responder dentro de restrições de tempo bastante precisas, gerando determinada ação de acordo com eventos externos [MOU1996].

Softwares de Tempo Real geralmente realizam funções de controle do Sistema e troca de dados sob rígidas restrições de tempo, confiabilidade e segurança,que será detalhado no item 2.1.4 de maneira que o resultado de suas execuções possa ser aproveitado por outras partes do sistema [LOV1999, MOU1996].

O Software de Tempo Real necessita de um componente de monitoração para coordenar todos os demais e assegurar a execução do Sistema em Tempo Real, para tal será utilizado um medidor de temperatura contínuo Pode-se também empregar técnicas de

Processamento Paralelo e Programação Concorrente para implementar este tipo de aplicação de Software [LOV1999, MOU1996].

2.1.4 FALHAS DE SOFTWARE

Denomina-se defeito de Software, qualquer deficiência existente no código fonte do seu programa. Se alguma entrada do Sistema faz com que uma linha de código contendo defeito seja executada ocorre um erro, que pode constituir-se em uma falha [LOV1999].

Em função da dificuldade de comprovação da não existência de erros na implementação do software em relação à sua especificação, são utilizadas as técnicas de redundância de software e de informação, cujo objetivo é tornar o software mais robusto em relação à segurança, ou seja, tolerante a erros/defeitos porventura ainda existentes. Este aspecto é fundamental principalmente quando se trata de software de aplicação crítica.

A confiança é um atributo essencial dos sistemas críticos e todos os aspectos da confiança (Disponibilidade, Confiabilidade, Segurança e Proteção) podem ser importantes. Atingir um alto nível de confiança é normalmente, o requisito mais importante para os sistemas críticos, como mostra a figura 1 abaixo.

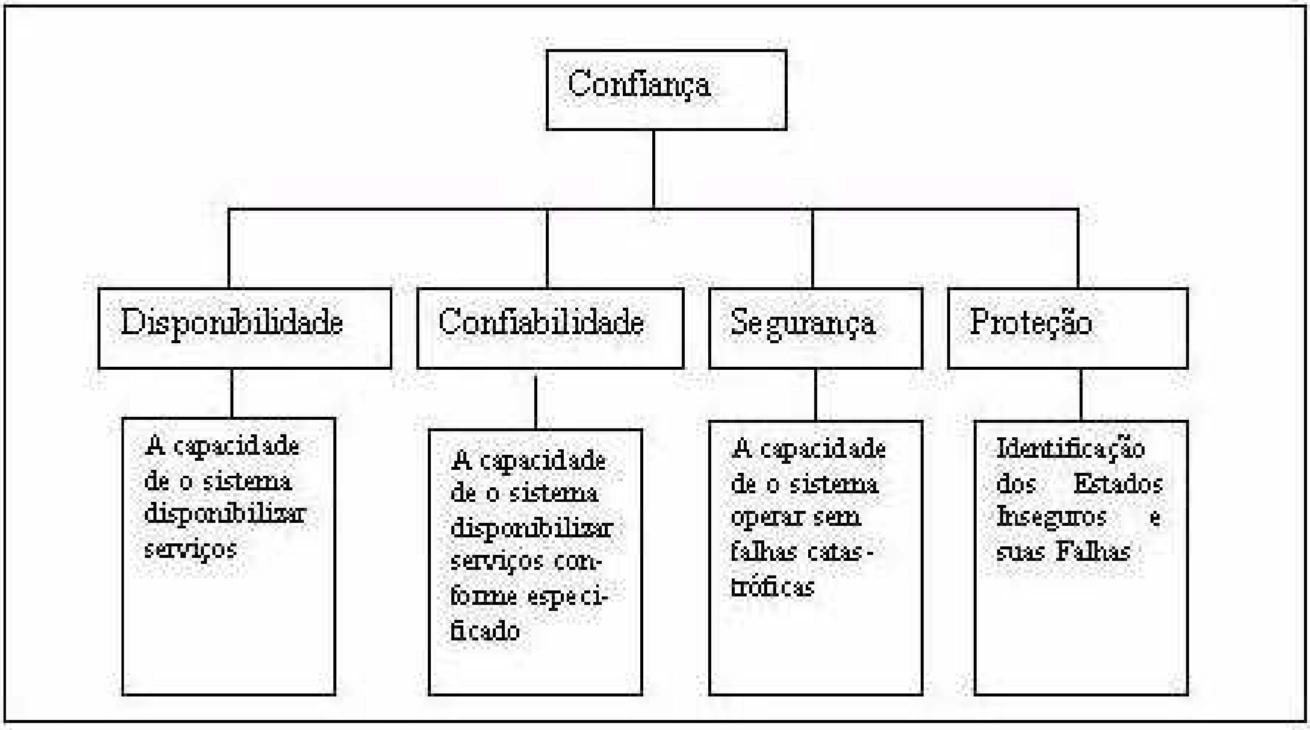


Figura 2-1: Representação da Confiança[VIL2003]

A ocorrência de uma falha pode levar o Software a atingir um Estado Inseguro, de forma que, caso o processamento continue, e não haja um tratamento adequando, ocasione um acidente [LOV1999].Especificamente ao software pode se dizer também que a falta completa de suas especificações em relação ao ambiente de aplicação tem se tornado um grave problema [LEVESON] fazendo com que o sistema atinja situações imprevistas como conseqüências de procedimentos operacionais errados, de mudanças não esperadas no ambiente operacional, ou ainda de modos de falhas do sistema não previstos.

Esse tipo de falha ocorre até que o defeito seja corrigido, pois o mesmo não se caracteriza como um evento aleatório. Isso impede que se apliquem no Software, com propriedade, as técnicas de avaliação e Garantia de Segurança de Sistemas Eletrônicos Analógicos e Mecânicos [MOU1996].

Os defeitos de Software ocorrem devido a imperfeições no seu desenvolvimento, falhas de Hardware, ou ainda a interação do Computador com outros componentes do Sistema [MOU1996].

Define defeito como sendo uma imperfeição existente no código fonte do programa, que caso seja ativada pode produzir erro [COL1995].

A Figura 2.2 a seguir mostra um gráfico da distribuição das principais fontes de defeitos para o Software.

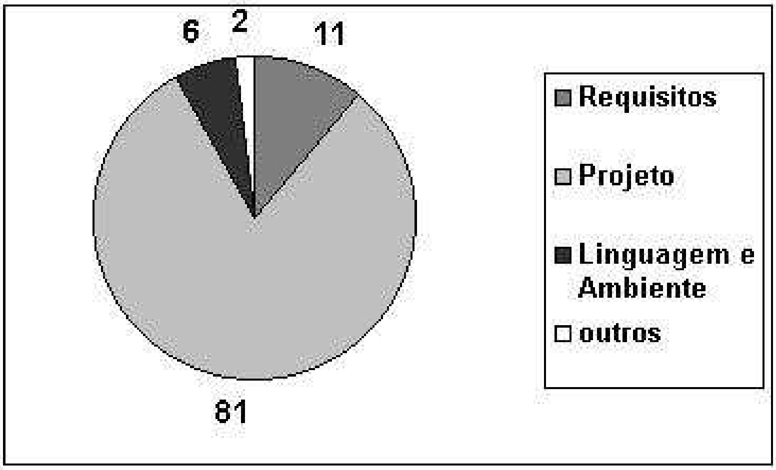


Figura 2-2 Distribuição dos Defeitos do Software[LOV1999]

Observa-se no gráfico da figura 2.2, que grande parte dos defeitos, decorre de da especificação de requisitos de forma incorreta e de imperfeições no projeto. Deve-se notar também que, aumentando-se a complexidade do Software, a quantidade de erros cresce exponencialmente [LOV1999] apud [MOU1996].

A Tabela 2.1 mostra os principais defeitos encontrados no desenvolvimento de um Software, fornecendo uma boa fonte de pesquisa para a identificação de possíveis defeitos.

Tipos de Defeitos	Subtipos
Funcionais	Requisitos Incorretos
	Completeza dos Requisitos
	Verificabilidade dos Requisitos
	Documentação dos Requisitos
	Mudanças de Requisitos
Implementação de Fun-ções	Acertos na Implementação
	Completeza das Características
	Completeza de Condições
	Domínio de Variáveis
	Mensagens
	Condições de exceção
Estruturais	Controle de fluxo do Programa
	Processamento
Dados	Definição, estrutura e declaração de dados
	Tratamento e acesso aos dados
Implementação	Edição de Programas
	Violação de Padrões de Programação
	Documentação de Programas
Integração	Interfaces Internas do Sistema
	Interfaces Externas do Sistema
Execução do Teste	Projeto do Teste
	Execução do Teste
	Documentação do Teste
	Completeza do Teste
Arquitetura do Software	Uso do Sistema Operacional
	Recuperação de Falhas
	Diagnóstico Incorreto
	Partição Incorreta
	Desempenho

Tabela 2.1 Principais Tipos e Subtipos de defeitos no Software[LOV1999].

Ao analisar a Tabela 2.1, pode-se concluir que todos os defeitos funcionais relacionam-se aos requisitos do Sistema, representando grande parte dos possíveis defeitos de seu Software, de acordo com a Figura 2.2. Estes defeitos, bem como os estruturais e de dados, originam-se na fase de Análise do Software.

Os outros tipos de defeitos provêm das demais fases do desenvolvimento do Software, não existindo nenhuma fase que não possa contribuir para a geração de defeitos. Conclui-se então que um processo que vise garantir a segurança de um Software deve consistir em uma abordagem completa do ciclo de desenvolvimento de um Software.

Como a maior parte dos defeitos de Software origina-se na fase de Análise [LOV1999], pode-se concentrar maior parte dos esforços para Garantia de Segurança nas primeiras fases do ciclo de desenvolvimento, procurando detectar o quanto antes os defeitos do Software. Esse procedimento reduz o esforço, o gasto e o tempo necessário para correção dos defeitos.

2.1.5 DESENVOLVIMENTO DE SOFTWARE

O ciclo de desenvolvimento de um Software compreende as fases, que vão desde a concepção até a sua integração. Cabe a uma Metodologia para desenvolvimento de Software, determinar um conjunto de fases do ciclo de desenvolvimento. Em todas estas fases, podem ser utilizados métodos, técnicas e ferramentas necessárias ao seu desenvolvimento, visando otimizar os processos, reduzir a quantidade de recursos necessários, e garantir a qualidade tanto dos seus produtos intermediários, quanto do produto final [LOV1999]

O objetivo deste trabalho é a análise de um sistema crítico para controle de temperatura do aço, utilizando técnicas de segurança de software. Ficará como proposta para trabalhos futuros a implementação do mesmo, devido não haver tempo suficiente para o seu desenvolvimento.

2.2 ANÁLISE DE SEGURANÇA DE SOFTWARE

A Análise de Segurança de Software visa determinar e avaliar as falhas do Software que possam levar o Sistema, que ele faz parte, a um Estado Inseguro. Enquanto a fase de Análise de Sistemas de uma metodologia visa a identificação das funções que o Software deve executar, a Análise de Segurança concentra-se em que o Software não deve executar [LOV1999].

A partir das informações obtidas na Análise de Sistemas, deve-se desenvolver a Análise de Segurança de Software. Para realizar esta análise com sucesso necessita-se de um procedimento composto de métodos e técnicas objetivando a identificação e a avaliação dos Estados Inseguros.

A Análise de Segurança de Software inicia com a preparação de uma Lista Preliminar de Inseguranças. Esta lista também pode ser elaborada na fase de Análise de Requisitos, integrando as restrições de Segurança às requisições do Sistema [LOV1999].

A partir da Lista Preliminar de Inseguranças realiza-se a Análise Preliminar de Insegurança, visando distinguir os Subsistemas Críticos. O produto dessa análise é uma lista contendo os componentes críticos do Sistema e seus possíveis Estados Inseguros.

A Análise de Insegurança de Subsistemas busca identificar novos Estados Inseguros nos Subsistemas Críticos, utilizando as técnicas: Análise de Modos de Falhas, Efeitos e Fatores Críticos (Failure Modes, Effects and Criticality Analysis - FMECA); Análise de Árvore de Falhas (Software Fault Tree Analysis – SFTA).

As atividades pertencentes à Análise de Insegurança de Sistema visam identificar e evidenciar Estados Inseguros relacionados à interação do Computador com outros componentes do Sistema.

Após a identificação dos Estados Inseguros deve-se avaliá-los e classificá-los de acordo com seu Fator Crítico como mostra a tabela 2.3 na subseção 2.2.2.

No final deste procedimento deve-se avaliar os Estados Inseguros não evitados, controlados ou recuperados completamente, visando garantir que os mesmos não representem uma ameaça à Segurança de Software. A Figura 2.3 a seguir mostra uma síntese da Análise de Segurança de Software.

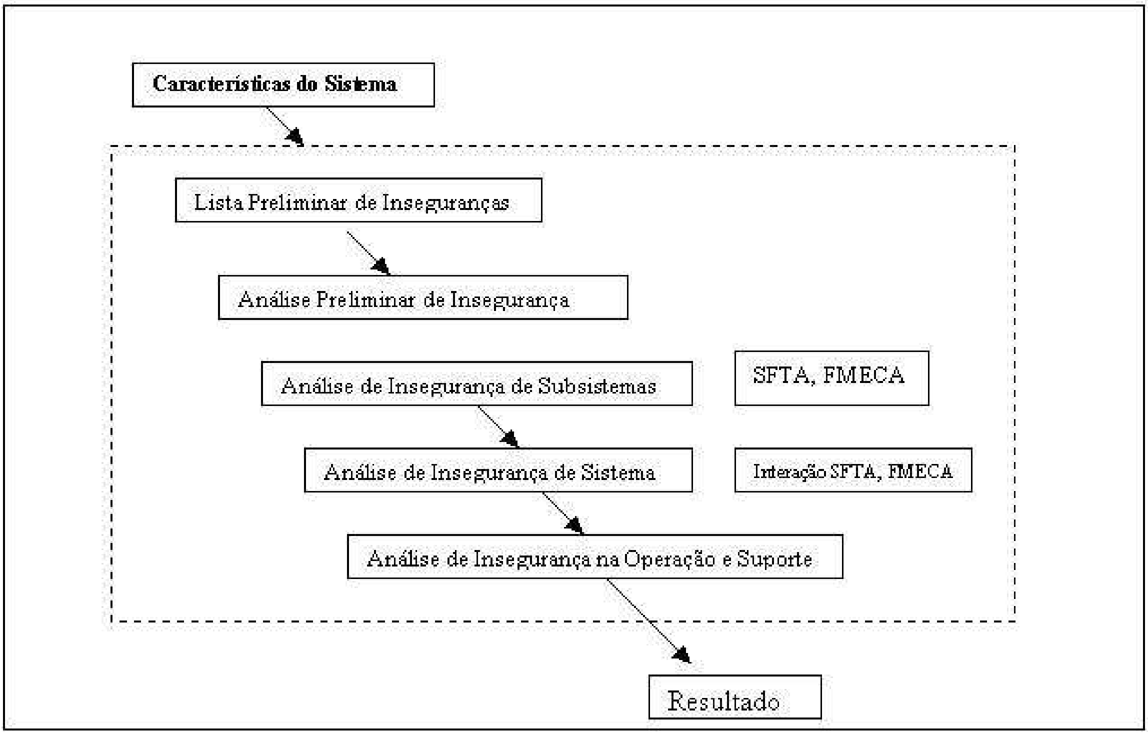


Figura 2-3 Procedimento da Análise de Segurança de Software[LOV1999]

Na Figura 2.3 apresenta-se a atividade do Procedimento de Análise de Segurança de Software, bem como as técnicas para identificação e avaliação de Estados Inseguros, necessários para a realização destas atividades.

O produto final da Análise de Segurança de Software é um documento contendo uma tabela com informações a respeito dos Estados Inseguros, sua falha causadora, seu Efeito, o Fator Crítico do efeito e ação requerida, apresentando-se assim, de forma clara e objetiva.

As atividades componentes do procedimento para Análise de Segurança de Software devem repetir-se iterativamente, à medida que a quantidade de informações aumenta. Logo, não se pode esperar que a seqüência apresentada acima obedeça a uma rígida ordem cronológica [LOV1999]. A realização criteriosa deste procedimento de Análise de Segurança de Software pode reduzir o esforço de detecção e correção de defeitos.

2.2.1 IDENTIFICAÇÃO DOS ESTADOS INSEGUROS E DE SUAS SEQÜÊNCIAS DE FALHAS

No início do procedimento para Análise de Segurança de Software deve-se elaborar uma Lista Preliminar de Inseguranças, a partir de informações existentes sobre Análises de Segurança de Softwares similares.

Outros tipos de análises podem ser feitos, também para identificação dos Estados Inseguros e de suas falhas causadoras [LOV1999, PÔR1997]:

- ✓ A Análise Preliminar de Insegurança (Preliminary Hazards Analysis) visa determinar os Subsistemas críticos, e se possível propor alternativas de controle. Ela baseia-se nos dados existentes e na experiência dos analistas para desenvolver uma análise em alto nível das principais funções e interfaces do Software;

- ✓ A Análise de Insegurança de Subsistemas (Subsystem Hazards Analysis) objetiva a identificação dos Estados Inseguros, e suas falhas, no projeto de cada subsistema e de sua interface. Concentra-se basicamente em aspectos tais como desempenho, degradação no funcionamento e falhas funcionais, procurando determinar os modos de falhas e seus efeitos;

- ✓ A Análise de Insegurança de Sistema (System Hazards Analysis) identifica os Estados Inseguros, e suas falhas, associados às interfaces entre os Subsistemas, incluindo potenciais erros humanos; e

- ✓ A Análise de Insegurança na Operação e Suporte (Operation and Support Hazards Analysis) identifica os Estados Inseguros e suas falhas durante as fases de uso e manutenção do sistema, especialmente aqueles provenientes da interação do homem com o Sistema, ou seja, dos procedimentos operacionais.

É necessário ressaltar que essas análises iniciam logo que a Análise de Sistema apresente-se bem definida. A cada modificação no projeto do sistema deve-se realizar também uma revisão das análises.

Existem algumas técnicas para identificação de Estados Inseguros, utilizadas nessas análises. Como as técnicas para análise de Segurança de Software são adaptadas de outras áreas da tecnologia, torna-se necessária à utilização conjugada das mesmas para a identificação apropriada dos Estados Inseguros [LOV1999].

Portanto, deve-se conhecer as características peculiares do tipo de aplicação em desenvolvimento para determinar-se as técnicas a serem utilizadas nas análises de segurança, observando-se também as suas potencialidades e restrições.

Na Subseção 4.1, apresenta-se à forma de utilização e desenvolvimento dessas técnicas, uma análise detalhada das mesmas, bem como os benefícios e as limitações do seu emprego.

2.2.2 DETERMINAÇÃO DO FATOR CRÍTICO DOS ESTADOS INSEGUROS E AVALIAÇÃO DA ACEITABILIDADE

Após a identificação dos Estados Inseguros realiza-se uma avaliação dos mesmos quanto ao Fator Crítico, considerando a Severidade do acidente ativado pelo Estado Inseguro e sua Probabilidade de ocorrência [LOV1999].

Geralmente, classificam-se os Estados Inseguros quanto à severidade de acordo com a gravidade do acidente que podem causar. A Tabela 2.2 a seguir mostra um exemplo de classificação de acordo com a Severidade.

Categoria do Acidente	**Definição**
Catastrófico	Pode causar muitas mortes
Crítico	Pode causar morte ou muitas lesões graves
Marginal	Pode causar uma lesão grave
Menor	Pode causar lesão leve

Tabela 2.2Uma possível Classificação do Acidente quanto a Severidade [LOV1999].

A classificação apresentada na Tabela 2.2 baseia-se somente nas perdas humanas que podem ser causadas pelos acidentes, não considerando perdas materiais ou ambientais. Determina-se a faixa de valores aceitável para a ocorrência da falha causadora de um acidente, a partir da Severidade do mesmo. Ou seja, uma falha que provoque um acidente de alta severidade deve possuir baixa possibilidade de ocorrência.

Para avaliar-se os Estados Inseguros em função da probabilidade, torna-se necessária à diferenciação entre Falhas. Ocasionais, aquelas causadas por mudanças físicas no Sistema, e Falhas Sistemáticas, as ocasionadas por defeitos na especificação ou no projeto. Como grande parte das Falhas em Software classifica-se como Sistemática, dependendo também das entradas do Sistema e da realização de transições de estados, a previsão da probabilidade de ocorrência de acidentes em função do tempo para o Software torna-se muito difícil. Devido à falta de dados numéricos confiáveis, comumente não se considera a classificação quanto à probabilidade para o Software [LOV1999, MOU1996].

Caso existam dados numéricos, pode-se determinar a probabilidade de cada estado, classificando-os de acordo com as exigências do sistema. De acordo com probabilidade de ocorrência de acidente,os Estados Inseguros do Software pode ser dividido em cinco faixas de valores: Freqüente, Provável, Ocasional, Remota e Improvável [LOV1999].

A classificação do Fator Crítico dos Estados Inseguros deve considerar a severidade e a probabilidade dos mesmos. A Tabela 2.3 mostra uma possível classificação dos estados quanto ao Fator Crítico.

Fator Crítico				
	Tipo de Severidade			
Faixa de valor (Probabilidade)	Catastrófica	Crítica	Marginal	Desprezível
Freqüente	S4	S4	S3	S2
Provável	S4	S3	S3	S2
Ocasional	S3	S3	S2	S2
Remota	S3	S2	S2	S1
Improvável	S2	S2	S1	S1

Tabela 2.3 Uma possível Classificação do Acidente quanto ao Fator Crítico [LOV1999].

Cada nível do Fator Crítico exige um tipo diferenciado de tratamento para garantir a segurança do Software. Pôr exemplo, um estado do nível S4 necessita de atenção especial, exigindo a utilização de estruturas de programação para evitar, controlar ou recuperar sua ocorrência. Um estado do nível S1 pode não demandar tantos cuidados, ou até mesmo nem exigir a utilização destas estruturas.

Após o término da determinação do Fator Crítico, avaliam-se os Estados Inseguros não evitados, controlados ou recuperados completamente, para determinar se os mesmos realmente não representam uma ameaça a Segurança de Software, garantindo assim que o mesmo apresenta um nível aceitável de risco.

2.3 TÉCNICAS PARA ANÁLISE DE SEGURANÇA DE SOFTWARE.

Para utilizar a sistemática de Segurança de Software, comentada na seção 2.2, deve-se determinar a forma de utilização, as potencialidades e as limitações dos métodos, técnicas, ferramentas e métricas necessárias às fases de Análise.

Este trabalho de pesquisa não aborda nenhuma ferramenta computacional específica para Segurança de Software. A referência [LOV1999, COL1995] apresenta um relato sobre a utilização das ferramentas (SFTA, SFMECA E SFMA), alem de Métodos Formais para análise quantitativa e qualitativa que será utilizado na segurança de Software .

Pode-se dizer que a técnica para Análise de Segurança de Software se dá de duas formas diferentes: a Análise Retroativa (Backward Analysis); e a Análise Progressiva (Forward Analysis) [LOV1999, COL1995].

A Análise Retroativa parte de um determinado Estado Inseguro e procura identificar quais eventos ou condições conduziram a ele. Quando o Software possui muitos estados possíveis e já se conhece os Estados Inseguros, recomenda-se esse tipo de análise.

A Análise Progressiva inicia a partir de um conjunto específico de entradas em um determinado estado do Software. A partir desta configuração, simula-se as situações que podem ocorrer com o Software, para determinar a possibilidade de atingir um Estado Inseguro. A análise permite que se verifique se as especificações concordam com a forma prevista de execução do Software.

2.3.1 TÉCNICA DE ANÁLISE DE MODOS DE FALHA, EFEITOS E FATORES CRÍTICOS (FMECA)

A Análise do Tipo e Efeito de Falha, conhecida como FMEA (do inglês Failure Mode and Effect Analysis), é uma ferramenta que busca, em princípio, evitar, por meio da análise das falhas potenciais e propostas de ações de melhoria, que ocorram falhas no projeto do produto ou do processo. Este é o objetivo básico desta técnica, ou seja, detectar falhas antes que se produza uma peça e/ou produto. Pode-se dizer que, com sua utilização, se está diminuindo às chances do produto ou processo falhar, ou seja, estamos buscando aumentar sua confiabilidade.[COL1995], A Tabela 2.4 a seguir apresenta um exemplo de utilização da FMEA.

Componente	Modo de Falha	Efeitos no Sistema
Medidor de temperatura contínuo	Falha na medição temperatura do aço	Estado Inseguro (será utilizado modo manual)

Tabela 2.4 Exemplo de Utilização de FMEA[LOV1999].

Observa-se no exemplo da tabela acima, que os modos de falha representam as reações do Software às possíveis situações em que pode ocorrer a falha. Por meio de raciocínio indutivo, utiliza-se a técnica para determinar o efeito no Sistema da falha de um componente em particular, incluindo instruções de Software [LOV1999].

Para aplicar-se a análise FMEA em um determinado processo, será relacionado todos os tipos de falhas que possam ocorrer, descrever, para cada tipo de falha suas possíveis causas e eventos, relacionar as medidas de detecção e prevenção de falhas que estão sendo, ou já foram tomadas, e para cada causa de falha, atribuir índices para avaliar os riscos. [COL1995]

Um exemplo da utilização da técnica FMECA pode ser visto na Tabela 2.5, abaixo.

Componente	Modo de Falha	Efeitos no Sistema	Fatores Críticos
Medidor de temperatura contínuo	Falha na medição temperatura do aço	Estado Inseguro (será utilizado modo manual)	S3

Tabela 2.5 Exemplo da Utilização da FMECA[LOV1999].

A Tabela 2.5 mostra a utilização da técnica FMECA para o Sistema crítico de controle de temperatura. Caso não se consiga uma amostragem da temperatura do aço no modo contínuo (automático), acontecerá um Modo de Falha com Fator Crítico S3, pois o mesmo pode ocasionar Estado Inseguro, do ponto de vista Gerencial, e ainda, considera-se que ele se encontra na Faixa de Valor "Provável" da Tabela 2.3. Deve-se então definir as formas de prevenção necessárias para eliminar a causa deste modo de falha.

As principais vantagens da FMECA consistem na possibilidade de revelar Estados Inseguros imprevistos, na determinação do Fator Crítico do efeito de uma falha, e no fato dela apresentar-se bem definida e sistematizada [LOV1999].

No entanto, a FMECA apresenta algumas limitações, a saber: não considera falha múltipla; consome muito tempo; e aumenta muito o esforço necessário para avaliação do Software, mesmo qualitativa [COL1995].

2.3.2 TÉCNICA DE ANÁLISE DE ÁRVORE DE FALHAS DE SOFTWARE (SOFTWARE FAULT TREE ANALYSIS – SFTA).

A Técnica de Análise de Árvore de Falhas de Software (Software Fault Tree Analysis - SFTA), é uma técnica analítica de análise de confiabilidade e de segurança amplamente utilizada para sistemas complexos. Sua utilização, visa à identificação de pontos para a introdução de melhorias ou de modificações para tornar o processo mais robusto, através de abordagem sistêmica, traçando a rota entre os sintomas percebidos pelo software e as causas das anomalias dentro da arquitetura do processo, não necessitando aplicá-la em todo Sistema, mas somente naquela parte considerada crítica.[COL1995]

Um SFTA, na sua fase inicial de elaboração, é basicamente uma representação gráfica da relação seqüencial e paralela da causa e efeito, obtida quando a falha de um sistema é estabelecida, através da pesquisa de diferentes causas de sua origem. determinando o Estado Inseguro [COL1995].

Pode utilizar-se a SFTA para calcular a probabilidade de ocorrência de um Estado Inseguro, conhecendo os valores das probabilidades de seus eventos causadores. Isto ajuda na determinação das partes mais críticas do Software, e que portanto, requerem cuidados especiais para Garantia de Segurança [LOV1999].

Para desenvolver essa análise utiliza-se uma Árvore de Falhas, que tem como raiz, ou evento inicial, o Estado Inseguro que se pretende avaliar. A partir da raiz expande-se a árvore, por meio de um exercício de lógica indutiva, até os eventos básica causadores do estado Inseguro. Os símbolos usados na SFTA podem ser vistos na Tabela 2.6 a seguir.

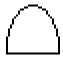
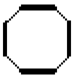
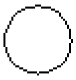
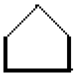
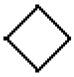
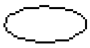
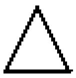
	Módulo ou comporta "E"
	Módulo ou comporta "OU"
	Módulo ou comporta de inibição. Permite aplicar uma condição ou restrição à sequência
	Identificação de um evento particular, topo ou contribuinte
	Falha primária de um ramo ou série. Evento básico
	Normalmente um evento que sempre ocorre, a menos que ocorra falha
	Evento não desenvolvido. Falta de informação ou de consequência suficiente.
	Indica ou estipula restrições
	Símbolo de conecção a outra parte da árvore

Tabela 2.6 Simbologia de uma Árvore de Falhas [LOV1999].

Na Tabela 2.6 percebe-se a existência de simbologia específica para eventos sobre o qual não se tem controle. Um evento desse tipo, para este Trabalho de Pesquisa, significa, por exemplo, erros do usuário na utilização do Sistema, pois esses eventos não pertencem à Abrangência do Trabalho.

Resumidamente, o processo de elaboração de uma Árvore de Falhas envolve as seguintes atividades:

A partir de um modelo representativo do Sistema desenvolve-se uma lista categorizada de Estados Inseguros;

- Selecionam-se os Estados Inseguros para análise;
- Assume-se o Sistema esteja em um Estado Inseguro e com base nas informações existentes retrocede-se à procura das causas do evento. O Estado Inseguro é a raiz da árvore que se expande até chegar aos eventos básicos ou àqueles sobre os quais não se tem informação suficiente;
- A Análise da Árvore de Falhas realiza-se por meio de um conjunto de cortes mínimos ou por análise de causa comum; e
- A determinação de alternativas possíveis para evitar as falhas.

No capítulo 4 será construída uma SFTA para ilustrar os Estados Inseguros de um sistema crítico de controle de temperatura do aço, alem de falar um pouco mais dessa ferramenta.

2.3.3 USO INTEGRADO DE SFTA E FMECA

Pode-se considerar as técnicas SFTA e FMECA como básicas no processo de Análise de Segurança de Software. Para suprir as restrições dessas técnicas, expostas nas subseções 2.3.1.1 e 2.3.1.2, Moura e Santellano propõem a utilização conjugada das mesmas [LOV1999, COL1995]. A Tabela 2.7 a seguir resume as principais diferenças entre as técnicas.

Característica	SFTA	FMECA
Tipo de Análise	Seletiva	Exaustiva
Eventos Pesquisados	Apenas os relacionados a determinada falha	Todos os modos de falha em potencial
Forma de Condução	Retroativa	Progressiva
Análise do Seqüenciamento dos Eventos	Evidencia situações em que um Estado Inseguro só ocorre a partir de um encadeamento de eventos anteriores	Não mostra encadeamento de eventos
Análise da Relação entre os Eventos	Evidencia os inter-relacionamento entre os eventos	Não mostra inter-relacionamentos
Esforço	Menor que FMECA	Demorado e caro

Tabela 2.7 Diferenças Básicas entre SFTA e FMECA [LOV1999].

Pode-se observar na Tabela 2.7 acima, a complementaridade das características das técnicas FMECA e SFTA. Como a SFTA é conduzida de forma progressiva e a FMECA de forma retroativa, a combinação de seus resultados auxilia a determinação dos principais aspectos relacionados a uma falha, relevantes para a Análise de Segurança de Software. A SFTA permite a identificação de eventos encadeados, causadores de um Estado Inseguro, complementando os resultados da FMECA.

Como citado anteriormente, as técnicas para Análise de Segurança de Software provêm de outras áreas do conhecimento, exigindo a utilização conjugada das mesmas para obter-se resultados apropriados.

A utilização conjunta das técnicas deve seguir os seguintes procedimentos [LOV1999]:

- Realização de uma análise do Sistema visando a identificação dos Estados Inseguros mais críticos, os quais agrupa-se em classes de equivalência, de acordo com seu Fator Crítico, denominadas Configurações de Pane (Failure Conditions);
- Utilizando a SFTA determina-se a causa relacionada ao Software de cada Estado Inseguro denominada panes funcionais;

- Desenvolvimento da FMECA para obtenção dos modos de falha de cada componente do Sistema, representadas graficamente por uma árvore de eventos;

- Conjugação da árvore de eventos com SFTA, para obtenção de um gráfico da relação causa - efeito dos Estados Inseguros mais críticos; e

- Elaboração de um Sumário de Modos de Falha e Efeitos (SMFE), ordenados objetivando-se relacioná-los às panes funcionais.

Essa técnica permite a agregação dos resultados das Análises de Segurança em nível do Sistema e em nível do Software. Pode-se investigar as causas e/ou as conseqüências de uma falha em diferentes níveis do Sistema. A técnica permite ainda observar com bastante clareza a interação do Software com o Sistema [LOV1999].

3 DESCRIÇÃO DO PROCESSO DE FABRICAÇÃO DO AÇO

Será apresentado neste capítulo como é o processo de fabricação do aço, alem de apresentar Tabela de Velocidade X Temperatura, relatar qual a Importância do controle de temperatura, definir Estratégias para controle de temperatura do aço e apresentar o Diagrama de equilíbrio das ligas ferro–carbono.

3.1 INTRODUÇÃO.

A Aciaria possui instalações para o recebimento do Ferro Gusa, Pré Tratamento do Gusa e fabricação do aço onde ocorre a transformação do gusa líquido em aço líquido através do Convertedor. Para atingir a qualidade desejada o aço passa por um processo de refino secundário podendo ser tratado no Forno Panela, Estação de Borbulhamento e no Desgaseificador RH. Logo após este processo o aço segue para o Lingotamento, onde esta operação pode ser feita pelo método contínuo ou convencional como mostra a figura 3.1 abaixo. A proposta apresentada para esse trabalho de fim de curso enfatiza o controle de temperatura do aço no Lingotamento Contínuo que posteriormente será detalhado.

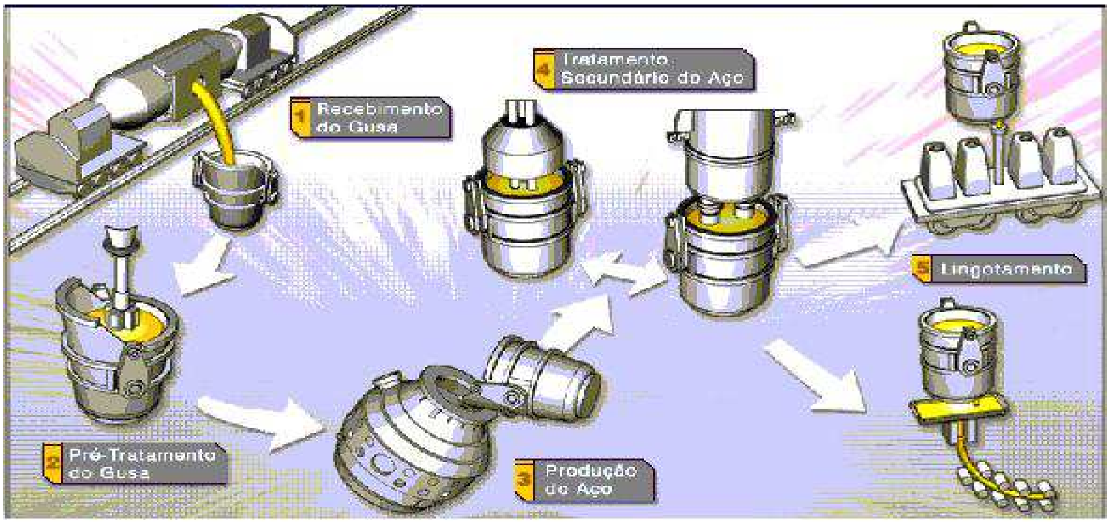


Figura 3-4 Estágio da Transformação do Gusa em Aço [CD HIPERMÍDIA]

Nos próximos itens será definida cada etapa do processo de Fabricação do Aço como mostra a figura 3.1.

3.2 RECEBIMENTO DO GUSA

O processo de produção inicia se no recebimento do ferro gusa produzido no Alto Forno e transportado pelo carro torpedo (Figura 3.2) que é um equipamento preparado para o transporte do Gusa Líquido do Alto Forno para a Aciaria, mantendo a temperatura necessária ao processo, cujo capacidade é de 350T.

Figura 3-5 Carro Torpedo [CD HIPERMÍDIA]

Chegando na Aciaria o material é transferido para a panela de gusa (Figura 3.2), também chamada de panela pelicano, que é o equipamento responsável pelo carregamento do gusa líquido no Convertedor. Este recipiente é fabricado com chapas de aço, a panela é revestida internamente com tijolos refratários, que recebe o aço líquido do Convertedor. Possui um sistema ou vazamento composto de uma válvula no fundo da panela, que controla o fluxo de aço, tendo capacidade de 220T.

Em seguida a panela será içada e transferida até a estação de Pré Tratamento do Gusa, seguindo todos os padrões e normas de segurança da Empresa .

Figura 3-6 Panela de Gusa [CD HIPERMÍDA]

3.3 PRÉ TRATAMENTO DO GUSA

O Pré Tratamento do gusa é um produto obtido no Alto Forno a partir da redução dos óxidos de ferro contidos nos minérios de ferro, sinter, e pelotas. É a principal matéria _ prima para a fabricação do aço no Convertedor, correspondendo de 80% a 85% da carga metálica.

O processo de Dessufuração do gusa consiste na redução do enxofre por meio de agitação mecânica e adição de cal , florita, e óxido de alumínio. Após a Dessufuração remove-se a escória liberando a panela para o carregamento do Convertedor.

3.4 PRODUÇÃO DO AÇO

O Convertedor (Figura 3.4) é o equipamento responsável pela transformação do gusa em aço. Este processo consiste no refino de uma carga metálica composta de Gusa Líquido e sucata sólida. Composto de um vaso metálico o Convertedor é revestido com tijolos refratários, em função das altas temperaturas do processo. A carcaça é sustentada por um anel que possui dois munhões para permitir seu basculamento em 360 graus.

Figura 3-7 Visão do Convertedor Internamente e Externamente[CD HIPERMÍDA]

A sucata é carregada no Convertedor e em seguida carregada o gusa líquido. Após o carregamento da carga metálica inicia - se o processo de sopro de oxigênio. Durante esta etapa, fundentes e materiais refrigerantes são adicionados.

Momentos antes do término do sopro são feitos medições no banho metálico através da sub-lança, possibilitando a realização de ajustes finais para garantir a qualidade do aço. Finalizando o sopro o Convertedor inicia o vazamento do aço líquido na panela, onde são adicionadas as ligas para ajuste a composição química desejada. A escória formada durante o processo é vazada pela boca do Convertedor no pote de escória.

3.4.1 PREPARAÇÃO DA SUCATA

O uso de sucata no Convertedor é necessário para o controle da temperatura do aço líquido e para o aumento da produção. A sucata utilizada na aciaria é gerada na área interna da usina e adquirida externamente.

3.4.2 PROCESSO DE FABRICAÇÃO DO AÇO

O processo de transformação de gusa líquido em aço ocorre no Convertedor. Após o carregamento das sucatas e do gusa o Convertedor é basculado para a posição vertical (Figura 3.5). A lança de oxigênio é então abaixada e o sopro é iniciado.

O oxigênio entra em contato com o banho metálico (gusa líquido + sucata sólida) e fundentes adicionados no início do processo começando o refino. Alguns elementos contidos na carga como: Carbono, Manganês, Enxofre, Silício e Fósforo são oxidados formando os gases e a escória .

A lança de oxigênio é composta de três tubos de aço concêntricos. O tubo central é por onde passa o oxigênio e os outros dois de entrada e saída de água de refrigeração. Na extremidade deste conjunto de tubos, é soldado um bico de cobre, com furos, por onde o oxigênio é soprado para reagir quimicamente com os componentes da carga.

A sub-lança é o equipamento utilizado para coletar amostras para análise química, medir o teor de carbono e oxigênio, nível do banho, altura da sola e temperatura do aço.

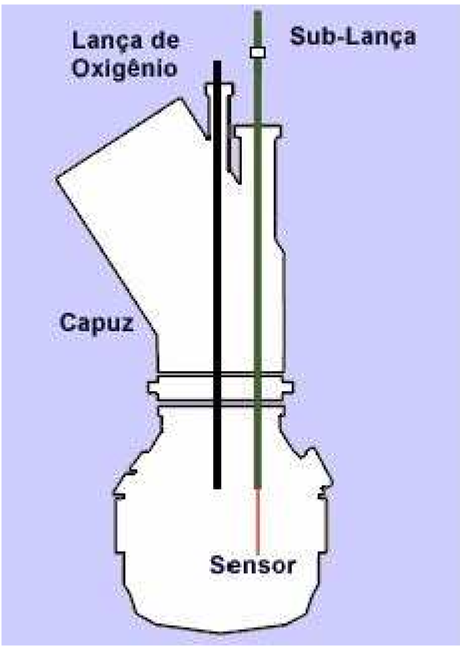


Figura 3-8 Convertedor na Posição Vertical, Lança de Oxigênio e Sub – Lança Submersas no Material Líquido

3.4.2.1 Calcinação

Principal escorificante do processo de Fabricação do Aço em Convertedores a Cal obtida da calcinação do calcário que é a forma que se encontra na natureza. Durante o processo de transformação do aço, a Cal (CaO) reage com elementos químicos contidos na carga como o fósforo, e o enxofre, cujos compostos vão fazer parte da escória. Estes elementos são considerados impurezas na maioria das aplicações do aço.

3.4.2.2 Sistemas de limpeza e recuperação de gás

O gás de Aciaria GAC gerado no processo são recuperados para uso na geração de energia elétrica. Para isto o sistema de limpeza e recuperação que faz a captação, refrigeração, lavagem e envio destes gases para o gasômetro, que é o recipiente de estocagem.

3.5 TRATAMENTO SECUNDÁRIO DO AÇO

O aço vazado na panela passa por uma série de ajustes de composição química e temperatura antes de ser lingotado, podendo receber outros elementos de liga dando novas propriedades no aço de acordo com o produto final pretendido.

No Forno e Panela são feitos ajustes de composição química e temperatura através da adição de ligas e aquecimento elétrico. As principais funções do Forno e Panela são:

1. Ajustar a composição química e homogenizar o aço.

2. Aquecer as corridas com baixa temperatura.

O Desgaseificador RH é feita à retirada de gases como nitrogênio e hidrogênio, aumentando o grau de pureza e garantindo a qualidade do produto final.

Conhecido originalmente para remoção de hidrogênio do aço, o sistema RH de desgaseificação a vácuo permite tratamentos adicionais, tais como a descarburação adição de ligas e elevação da temperatura e controle de alumínio do aço.

A Estação de Borbulhamento são feitos ajustes de composição química e correção da temperatura, através do sopro de nitrogênio e oxigênio. Geralmente o aço vem com temperatura mais elevadas, com intuito de ser corrigidas nesta fase de processo.

3.6 LINGOTAMENTOS

Após o refino secundário, o aço segue destino para o ligamento convencional ou contínuo.

No Lingotamento Convencional (Figura 3.6) através do auxílio da ponte rolante, o aço é vazado em formas especiais denominada lingoteiras formando lingotes. Após solidificação do material, os lingotes serão retirados das lingoteiras e enviados para a laminação onde serão conformados.

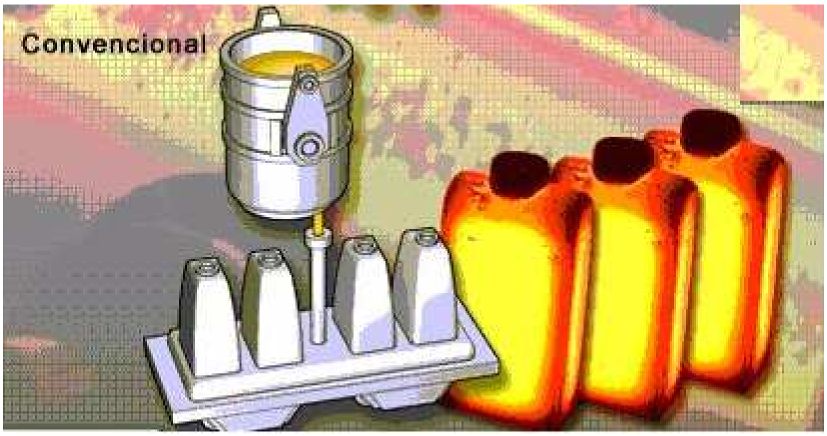


Figura 3.9 Lingotamento Convencional [CD HIPERMÍDA]

No Lingotamento Contínuo (Figura3.6) a panela de aço será acoplada na torre giratória, colocando a panela na posição de lingotamento Após a abertura da válvula da panela o aço será vazado no distribuidor que garante a alimentação contínua do material nos moldes. No molde o aço inicia se o processo de solidificação tomando a forma do material final.

O tarugo formado é continuamente extraído e resfriado por sprays de água na câmara de resfriamento, sendo cortado em comprimento desejado para serem transferidos para o acabamento Este trabalho abordará especificamente estudos sobre um sistema crítico envolvendo o controle de temperatura do aço no Lingotamento Contínuo.

No Lingotamento Contínuo é produzido o tarugo (produto final) de duas maneiras: Utilizando o jato abeto ou jato protegido contendo diâmetros de 130X130, 140X140 e 160X160 cm. A diferença entre eles é que o jato protegido o aço não tem contato com o ar, utiliza válvula submersa, sendo então classificado como um aço de melhor qualidade e conseqüentemente mais caro. No jato abeto despensa - se a utilização da válvula submersa .

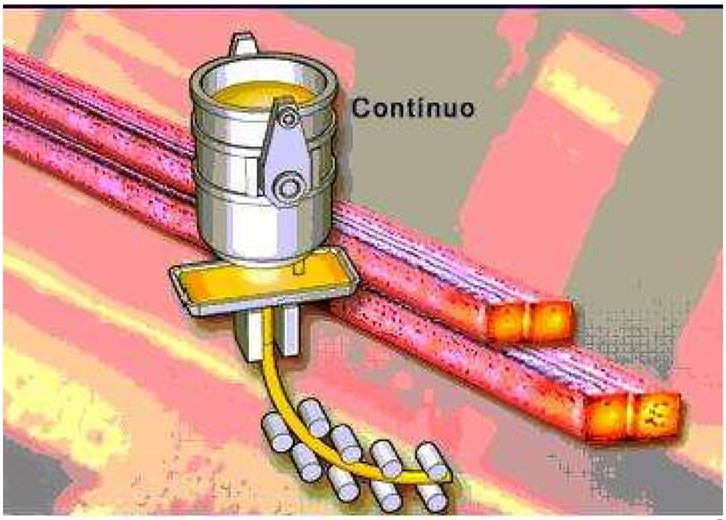


Figura 3-10 Lingotamento Contínuo de Tarugos [CD HIPERMÍDA]

3.7 IMPORTÂNCIA DO CONTROLE DE TEMPERATURA

O desejo de obter controle preciso de temperatura do aço líquido desde o Forno de fusão/refino até o molde de Lingotamento Contínuo tem três grandes razões: [FEL1992]

(I) aumento de produtividade;

(II) melhoria de qualidade e;

(III) uso eficiente de energia.

Em termos de produtividade, a perda precoce de temperatura no distribuidor poderá causar entupimento da válvula submersa [FEL1992] ou mesmo uma interrupção do processo de Lingotamento com uma certa tonelagem de material sendo retornada para o forno ou desviada para produção de lingotes. Temperatura baixa facilita também a formação de cascão na panela e no distribuidor. Quando a temperatura de Lingotamento é muito alta, tende a ocorrer redução de espessura solidificada no molde para uma dada velocidade de

Lingotamento, podendo causar acidentes tais como furos de veio ("breakouts") ou forçar redução das velocidades de Lingotamento.

Quanto à qualidade do produto lingotado continuamente, ela é muito influenciada pela parcela de superaquecimento do metal líquido no distribuidor. O efeito prejudicial de uma temperatura de Lingotamento excessiva é evidenciado por uma região de grãos colunares, que pode se estender desde a borda até o centro da seção do tarugo. Tal zona extensiva de cristais colocares é indesejável devido a seu efeito prejudicial sobre as propriedades mecânicas, maior freqüência de trincas e aumento de porosidade e segregação central de impurezas e elementos de liga.

Já a baixa temperatura de Lingotamento, por outro lado, é prejudicial com relação às melhores condições para flotação de inclusões [fel1992], tornando se difícil à separação das mesmas no molde, o que resulta em deterioração da qualidade do produto.

Uma vez removido do forno, seja de fusão (Convertedor) ou de refino (Forno Panela, Borbulhamento ou RH), o aço líquido deve conter toda a energia (Tempertutra) necessária para compensar as perdas de calor durante as operações subseqüentes O uso eficiente dessa energia requer um conhecimento dos mecanismos e magnitudes das perdas de calor na panela.

Existem três estratégias para controle de temperatura do aço líquido no Lingotamento (rotas):

(1) Via Estação de Borbulhamento

Vazar a corrida com temperatura elevada, porem com menor superaquecimento possível, que permita o tratamento da corrida na estação de metalurgia secundária até atingir a temperatura e composição química desejada para Lingotamento.

(2) Direto do Convertedor

Vazar cada corrida com o mínimo superaquecimento possível que permita obter a temperatura de liberação objetivada sem resfriamento ou aquecimento na panela.

(3) Via Forno e Panela

Vazar toda corrida a uma temperatura pré- determinada e reaquecer ou resfriar o aço na panela ou no distribuidor conforme necessário.

A estratégia (1) é típico onde o controle de tempo é realizado por acúmulo de corridas. Isso leva a longos tempos médios de corrida e uma alta freqüência de resfriamento na panela. Além disso, pode dificultar a abertura normal das válvulas das panelas e as altas temperaturas de vazamento reduzem a vida do refratário das mesmas.

A estratégia (2) é um método de otimização de energia no processo de fabricação do aço já que a energia térmica mínima necessária para manuseio do aço líquido é calculada para cada corrida para se determinar à temperatura de vazamento Esta estratégia requer um completo conhecimento das perdas de temperatura no processo do vazamento ao Lingotamento, o que requer um programa de computador “on line” que calcula uma temperatura de vazamento do forno baseada nos tempos da operação de Lingotamento. Esta estratégia não permite erros e o sistema é inflexível em termos de problemas operacionais

A estratégia (3) maximiza a vida do forno e simplifica sua operação, mas requer entrada subseqüente de energia na panela ou no distribuidor. O maior problema deste método é a instalação e custo de operação dos dispositivos de aquecimento na panela. Entretanto, uma vez que o aquecimento suplementar é disponível, a freqüência de corridas abortadas e desviadas pode ser minimizada e a temperatura realmente ser controlada.

A operação ótima depende da mistura de produtos a serem lingotados. Se for necessário controle muito estreito de temperatura (Exemplo: em aços sensíveis à segregação), uma combinação das estratégias (2) e (3) fornecerá os melhores resultados. Neste caso, as temperaturas de vazamento são calculadas e o metal é reaquecido ou resfriado na estação de metalurgia na panela apenas quando necessário e não como rotina.

Independentemente da estratégia de operação, é necessário calcular ou uma temperatura de vazamento ou uma temperatura de saída da estação de tratamento na panela.

Para atender este objetivo, porém, as perdas térmicas que irão ocorrer devem ser conhecidas e avaliadas.

Será apresentada a tabela de temperatura X velocidade que é utilizada pelo operador da sala de controle para fazer o pedido de temperatura do aço que se deseja produzir; como mostra a tabela 3.1 abaixo.

Sigla			Bitola	Liquidus :
a95			160	1451
Temperatruras para liberação				
Jato Aberto	Direto	Borbulh.	F. Panela	RH
Partida	1539	1529	1519	1512
Seguencia	1534	1524	1514	1509
J. Protegido	Direto	Borbulh.	F. Panela	RH
Partida	1539	1529	1514	1510
Seguencia	1534	1524	1509	1506
Velocidade m/min	Estrip. Negativo Máx. 190 cpm	Temperatura °C	Delta T	
2,70	34	1470	19	Mínimo
2,60	34	1473	22	
2,50	34	1476	25	Visado
2,40	34	1479	28	
2,30	34	1482	31	
2,20	34	1485	34	
2,10	34	1488	37	Máximo

Tabela 3.1 Velocidade de Lingotamento X Temperatura[CD HIPERMÍDIA]

3.8 DIAGRAMA DE EQUILÍBRIO DAS LIGAS FERRO – CARBONO

Este diagrama é obtido experimentalmente por pontos e apresenta as temperaturas em que ocorrem as diversas transformações dessas ligas, em função do seu teor de carbono como mostra a figura 3.8.

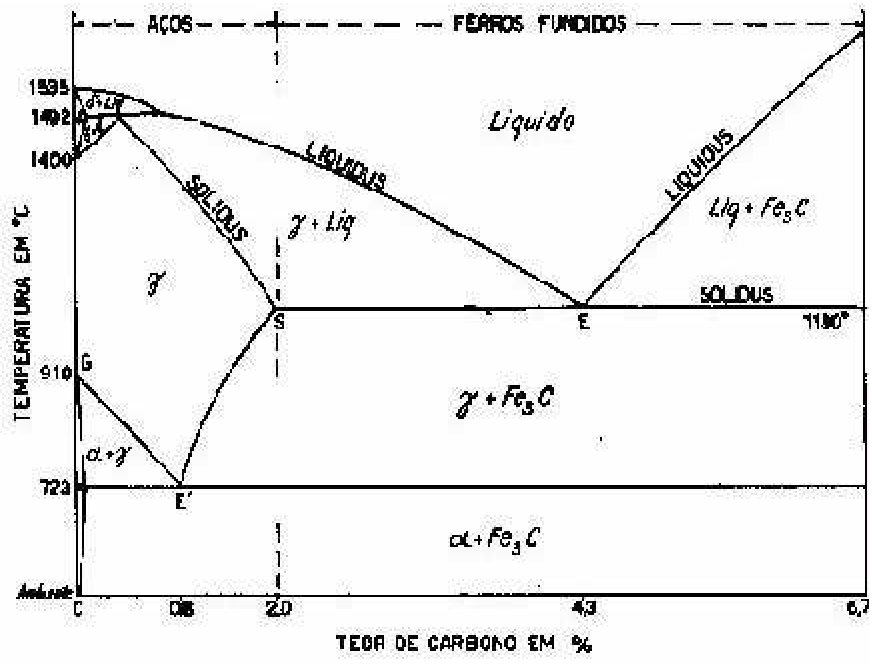


Figura 3-11 Diagrama de Equilíbrio das Ligas Ferro–Carbono [FEL1992]

Será abordada apenas à parte deste diagrama que diz respeito aos aços, isto é, a parte compreendida entre 0 e 2% de carbono.

Os componentes fundamentais dos aços carbono (Também chamada dos aços comuns ou ordinários) são o ferro e o carbono. Este último combina-se como uma parte do ferro, formando o carboneto de ferro, Fe3C, que contem 6,7% de carbono.

Quando o aço está no estado de fusão, o referido arboneto se encontra inteiramente dissolvido na massa líquida, constituindo, com o ferro, uma solução homogênia.

Ao esfriar-se, verifica-se que existe para cada aço de acordo com seu teor de carbono, uma certa temperatura à qual começa a solidificação, que depois prossegue, à medida que a temperatura cai. O lugar dos pontos de início de solidificação chama-se linha do liquidus, porque acima dela o aço está completamente líquido. O lugar dos pontos de fim de solidificação intitula - se linha dos sólidus, porque abaixo dela o aço está inteiramente sólido. Entre essas duas o aço está portanto , nem líquido nem sólido. Quando o aço atingir uma temperatura de 723 º (linha detransformação) pode - se dizer que a transformação está completa; sendo então assinalada no diagrama por uma linha horizontal.

A linha GE´S e a horizontal de 723ºC chamam - se de transformação, (porque marca o início e o fim das transformações no estado sólido), e a região delimitada por essas linhas denomina - se zona crítica. A solubilidade do carbono no ferro gama é limitada e depende da temperatura. A 1130º a temperatura é máxima e corresponde a 2,0% de carbono; a temperaturas mais baixas, a solubilidade decresce segundo a curva SE` assinalada.

4 ANÁLISE DE SEGURANÇA DA TEMPERATURA

Será apresentado neste capítulo a utilização dos conceitos apresentados no capítulo 3 aplicando-se aos conceitos estudados no capítulo 2.

4.1 LISTA PRELIMINAR DE INSEGURANÇA

A Análise de Segurança de um Software objetiva a determinação e a avaliação das possíveis falhas de Software que podem acarretar um Estado Inseguro no Sistema. Esta Análise é o primeiro procedimento específico para a Garantia de Segurança aplicado no desenvolvimento de um sistema. Deve-se ressaltar que se pretende analisar os Estados Inseguros que acionem acidentes que possam causar perdas.

Desenvolveu-se, uma Lista Preliminar de Inseguranças para o Protótipo de Software para o controle de temperatura, observando a Análises de Sistemas semelhantes apresentado na referência [LOV1999]. Os seguintes Estados Inseguros compõem a Lista Preliminar de Inseguranças deste Protótipo:

- Erro na abertura da Panela;
- Erro na Amostragem da Temperatura;
- Indicação de temperatura Baixa;

4.2 ANÁLISE PRELIMINAR DE INSEGURANÇA

A partir desta lista realizou-se a Análise Preliminar de Insegurança para identificar os Subsistemas críticos e seus possíveis Estados Inseguros, que foram agrupados em uma lista da seguinte forma:

- **Temperatura**
  - Alta;
  - Baixa.

- **Água de Refrigeração do material**
  - Erro ao digitar vazão de água

- **Unidade Hidráulica**
  - Vazamento de óleo.

4.3 ANÁLISE DE INSEGURANÇA DE SUBSISTEMAS

Serão apresentados nessa sessão exemplos de um SFTA da Lista Preliminar de Insegurança como mostra as figuras abaixo.

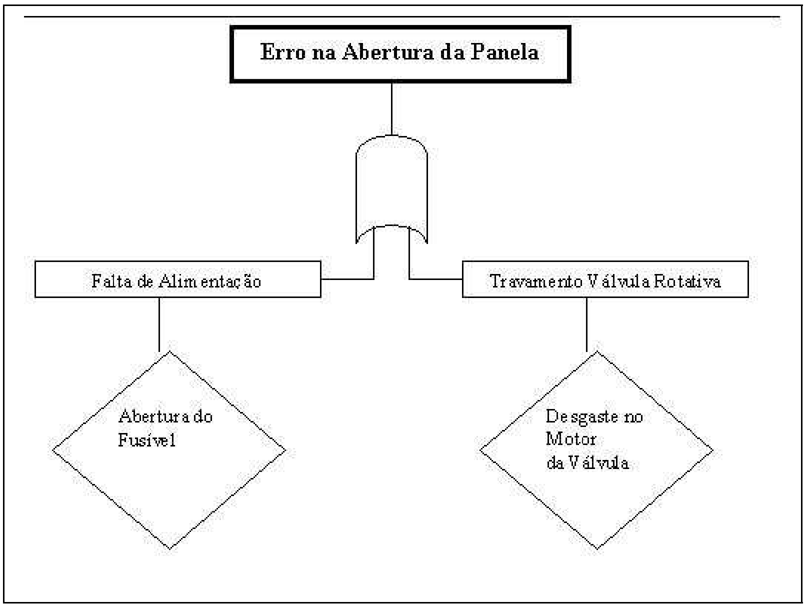


Figura 4-12 Exemplo de SFTA

A figura 4.1 apresenta uma árvore de falha conhecida como SFTA, onde tem como topo da árvore o Estado Inseguro Erro na Abertura da Panela. A causa do Erro na Abertura da Panela pode ter sido originado pela Falta de Alimentação devido a Abertura do Fusível ou pelo Travamento da Válvula Rotativa que tem como responsável o Desgaste no Motor da Válvula.

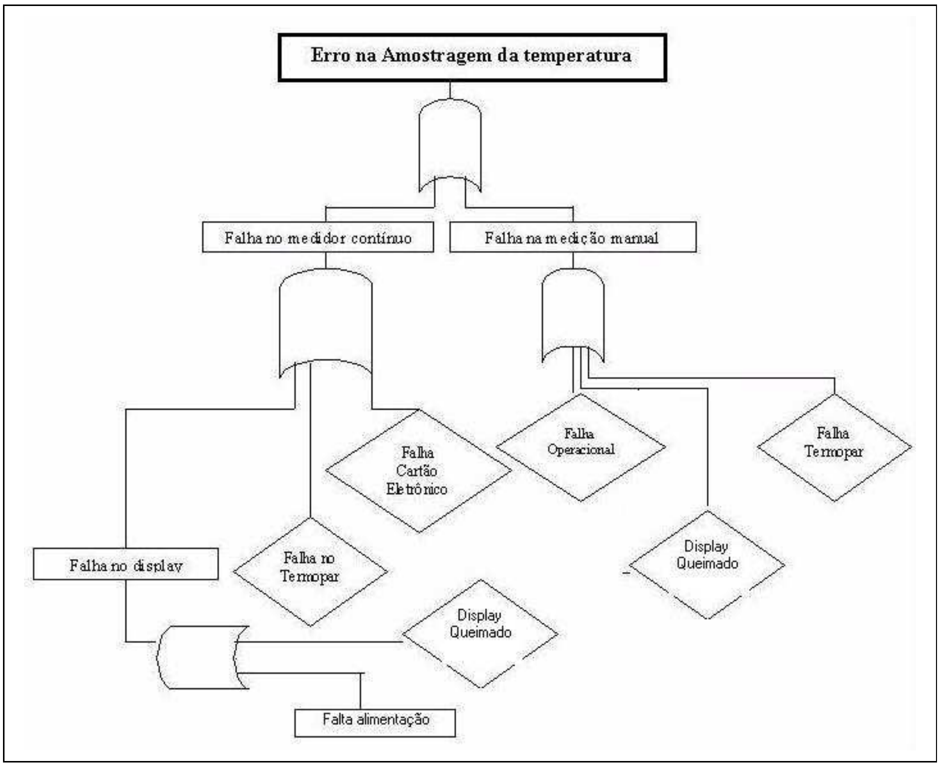


Figura 4-13 Exemplo de SFTA

A figura 4.2 tem como topo da Árvore o Estado Inseguro Erro na Amostragem da Temperatura. Este Erro pode ter sido causado pela Falha no Medidor Contínuo que apresenta as possíveis causas: Falha no Cartão Eletrônico, Falha no Termopar, Falha no Display, a qual pode ser originado ou pela Queima do Display ou por Falta de Alimentação; ou ainda por Falha na medição Manual que tem como causadores: Falha operacional, Falha do Termopar e Display Queimado.

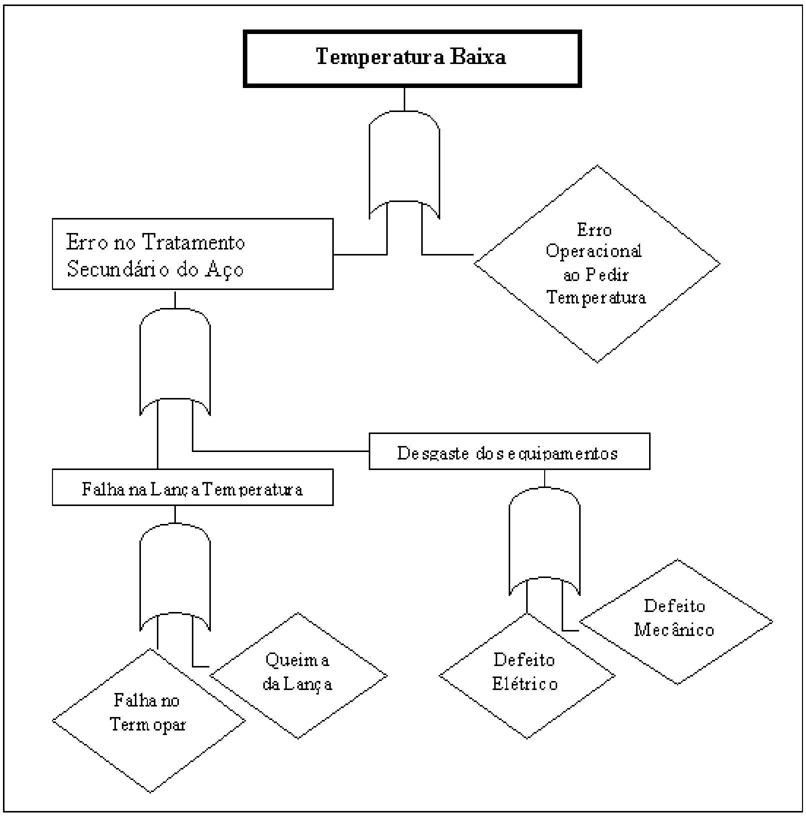


Figura 4-14 Exemplo de SFTA

A figura 4.3 tem como topo da Árvore o Estado Inseguro Temperatura Baixa. Esta Falha pode ter sido causado por um Erro Operacional ao pedir a Temperatura ou Erro no Tratamento Secundário do Aço apresentando pelas possíveis falhas: Falha na Lança de Temperatura ocasionado : Falha no Termopar ou Queima da Lança ou pelo Desgaste dos Equipamentos que compõe o Tratamento secundário do Aço que pode ser por Defeito Elétrico ou Defeito Mecânico.

A Análise de Insegurança de Subsistemas iniciou a partir dos Estados Inseguros determinados na análise anterior, buscando identificar novos Estados Inseguros nos Subsistemas críticos.

Aplicou-se primeiramente, a técnica de Análise de Modo de Falha, Efeito e Fatores Críticos (Failure Modes, Effects and Criticality Analysis - FMECA) que auxiliou na identificação dos novos Estados Inseguros e de seu fator Crítico. A Tabela 4.1 a seguir mostra um fragmento da aplicação da técnica para alguns Subsistemas Críticos do trabalho abordado.

Classe componente	**Modo de Falha**	**Efeitos**	**Fator Crítico**
Controlador de Temperatura	Temperatura alta Temperatura baixa	Perfuração do veio e possível acidente com funcionários Perda da produção devido obstrução.	S3 S2
Controlador de vazão Água	Erro ao digitar vazão	Danifica equipamento Deformação no material	S3 S2
Controlador da Unidade Hidráulica	Vazamento de óleo	Perda da produção Verificar causa do vazamento	S3 S2

Tabela 4.1 Um Fragmento da Aplicação da FMECA

Deve-se ressaltar que, na Tabela 4.1 considerou-se para efeitos de Estudo de Caso que todos os Modos de Falha encontra-se na Faixa de Valor “Provável”. Após avaliar-se a Severidade dos Estados Inseguros, empregou-se a Tabela 2.3 para determinar-se o Fator Crítico dos mesmos.

4.4 ANÁLISE DE INSEGURANÇA DE SISTEMAS

Com a realização da Análise de Insegurança de Sistema objetivou-se a identificação dos Estados Inseguros relacionados à interação do Computador com outros componentes do Sistema, utilizando-se para tanto da técnica de integração da SFTA com a FMECA como mostra a tabela 4.2.

Nível	Do Sistema	Do Software
Falhas	Erro ao digitar vazão	Defeito na Rotina de segurança. Defeito na Rotina de tratamento de Falhas.
	Vazamento de óleo	Defeito na Rotina de tratamento de Falhas Defeito na Rotina de Detecção

Tabela 4 2 Interação da SFTA com a FMECA

Observa-se na Tabela 4.2 que a integração dos resultados da FMECA com SFTA permite uma fácil identificação das conseqüências das falhas ocasionadas pelo Software, principalmente aquelas relacionadas à iteração do mesmo com os outros componentes do Sistema.

4.5 ANÁLISE DE INSEGURANÇA NA OPERAÇÃO E SUPORTE

Será mostrada uma lista constando os eventos sobre o qual não se tem informação ou controle conforme a tabela abaixo.

Classe componente	Modo de Falha	Operação e suporte	Ação
Abertura da Panela	Falta de Alimentação	Abertura do fusível	Trocar Fusível
	Travamento da Válvula Rotativa	Desgaste no Motor	Inspeção semanal no motor
Amostragem da Temperatura	Contínuo	Falha no Termopar Falha cartão Eletrônico Queima do Display	Verificar Estufa Inspeção semanal Substituição
	Manual	Falha operacional	Treinamento
Temperatura Baixa	Erro na Lança de Temperatura	Falha no termopar Queima da Lança	Verificar Estufa Reparar Lança

Tabela 4. 3 Lista dos Eventos que não se tem Informação ou Controle

Pela falta de valor probabilístico para os Estados Inseguros do Software, convencionou-se que todos se encontravam na Faixa de Valor “Provável”, permitindo assim a classificação dos Fatores Críticos destes estados.

5 CONCLUSÃO

No Capítulo, descreve-se a Revisão, Conclusão e a Proposta após a elaboração deste trabalho final de curso.

5.1 REVISÃO

No capítulo 1, descreram - se a introdução, o objetivo e a justificativa deste trabalho de pesquisa.

No Capítulo 2, apresentou – se a pesquisa bibliográfica sobre a Segurança de Software Crítico para Controle de Temperatura do Aço. Nele, descreveu - se os principais conceitos sobre Falha de Software, Análise de Segurança do Software e Técnicas para Análise de segurança de software.

No Capítulo 3, apresentou - se uma descrição sobre o Processo de Fabricação do Aço, principais conceitos e características.

No Capítulo 4, foi proposta uma análise de segurança sistema de controle de temperatura do aço, onde foram analisados alguns exemplos de SFTA e FMECA além de evidenciar a Lista Preliminar de Insegurança, Análise Preliminar de Insegurança.

Neste capítulo, descreve–se a conclusão e a proposta para continuação deste trabalho final de curso.

5.2 CONCLUSÃO

Todos nós estamos familiarizados com os problemas das falhas de sistemas computacionais. Por esta razão foi feito estudos sobre as ferramentas SFTA, FMEA e FMEC para que as informações mantenham sua integridade; conseqüentemente evitando graves conseqüências econômicas e humanas.

É de fundamental importância uma correta e precisa avaliação do software que desempenhe funções de segurança em sistemas críticos, de modo a evitar, em primeira instância, que se atinja um estado potencialmente inseguro, e em segundo lugar que ocorra um acidente.

Tendo este quadro em mente, torna-se óbvia a importância de se avaliar, em um software os fatores de insegurança.

5.3 PROPOSTA

Através da capacidade de se poder medir certos fatores do software, aliada às técnicas aqui mencionadas, podem e devem ser atingidos melhores níveis de segurança, diminuindo, desta forma, os riscos de acidentes fatais.

É essencial para o desenvolvimento do Software Crítico para controle de Temperatura conhecimento sobre Probabilidade e Estatística, pois com isso é possível estimar as possíveis falhas do Software.

Um trabalho importante a se realizar é a aplicação deste conjunto de fatores que foram abordados neste trabalho, visando como proposta para trabalhos futuros a implementação deste software utilizando Redes Baysianas, que é uma dos métodos abordado na Inteligência Artificial.

REFERÊNCIAS BIBLIOGRÁFICAS

[COL1995] COLPARERT HUBERTUS. “Metalografia dos Produtos Siderúrgicos Comuns”.(SP): Editora Edbard Blücher LTDA, Agosto de 1995.

[FEL1992] FELIX ARNALDO MOREIRA .”Influência das Variáveis de Processo de aciaria sobre a Temperatura Final do Aço no Lingotamento Contínuo”: Belo Horizonte (MG): Escola de Engenharia, setembro 1992.

[LOV1999] LOVISI FILHO E., CUNHA A. M. ”Uma Abordagem para o Desenvolvimento de Software Crítico Embarcado Aeroespacial com Garantia de Segurança”. São José dos Campos (SP): Anais do Simpósio sobre Segurança em Informática, Setembro de 1999, p. 57 - 66.

[MOU1996] MOURA C. A. T. e SANTELLANO J. "Integração de técnicas para Análise de Segurança de Software". Curitiba: Anais da Conferência Internacional de Tecnologia de Software: Qualidade de Software, Junho de 1996, p. 187 - 201.

[MOU1996] MOURA C. A. T. "Uma Estratégia de Análise de Segurança de Software para Aplicações Críticas". S. José dos Campos: ITA, 1996 (Dissertação de Mestrado).

[PÔR2003] PÔRTO I. J., DE BORTOLI L. A. "*Sistemas Tolerantes a Falhas*". Disponível por meio da WWW no endereço http://www.inf.ufgrs.br/gpesquisa/tf/portugues/ ensino/ lisangela/segsoft.html, Julho de 1997.

[SCA1999] SCAPIN CARLOS ALBERTO. “Análise Sistêmica de Falhas”. Belo Horizonte (MG): Editora de Desenvolvimento Gerencial,1999

[VIL2003] VILLE IAN SOMMER “Engenharia de Software 6ª Edição”. São Paulo (SP): Editora ABDR, 2003.