UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
DEPARTAMENTO DE ENGENHARIA ELÉTRICA
PROJETO DE GRADUAÇÃO

FERNANDO LUIZ SOSSAI MARTINELLI

SISTEMÁTICA DE SÍNTESE DE SISTEMAS ELETRÔNICOS DE ALTA CONFIABILIDADE APLICADA AO CONTROLADOR SEMAFÓRICO

VITÓRIA – ES
MARÇO/2016

FERNANDO LUIZ SOSSAI MARTINELLI

SISTEMÁTICA DE SÍNTESE DE SISTEMAS ELETRÔNICOS DE ALTA CONFIABILIDADE APLICADA AO CONTROLADOR SEMAFÓRICO

Parte manuscrita da Projeto de Graduação do aluno **Fernando Luiz Sossai Martinelli**, apresentada ao Departamento de Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para aprovação na disciplina "ELE08553– Projeto de Graduação II".

_____

Prof. Dr. rer. nat. Hans Jorg Andreas Schneebeli
Orientador

Professor
Convidado 1

Professor
Convidado 2

Professor
Convidado 3

VITÓRIA – ES
MARÇO/2016

AGRADECIMENTOS

Agradeço primeiramente a Deus pela oportunidade de ter estudado Engenharia Elétrica na UFES, por ter conferido a mim as habilidades, disciplina, e as forças necessárias durante toda minha trajetória acadêmica e profissional. Agradeço também a toda a minha família, em especial a meus pais, Luiz Fernando Martinelli e Ana Rosa Sossai Martinelli e meu irmão, Stefano, por valorizarem a educação, reconhecendo minhas conquistas e me apoiando em todos os momentos de cansaço e dificuldade. Sei que o modo como fui criado e incentivado a aprender e todo o amor de vocês durante a minha vida foram essenciais para esta conquista. Agradeço a Paula Sarmenghi, minha namorada, e sua família que, com amor e carinho, me apoiaram durante grande parte do meu percurso universitário, compreendendo de perto todas as dificuldades e desafios de um futuro engenheiro eletricista. Obrigado, gente, sem vocês esta conquista não seria possível. Agradeço ao meu orientador, Prof. Hans por compartilhar todo seu conhecimento e experiência. Como orientador, obrigado por aceitar compartilhar comigo objetivos e sonhos de projetos, encorajando-os e, talvez mais importante, desencorajando-os da maneira certa para que meus objetivos fossem alcançados. Nas disciplinas, certamente todas as histórias contadas, que não serão esquecidas, serviram de motivação para a carreira escolhida. Agradeço ao professor Evandro Ottoni, por aceitar o convite para compor a banca e por nos ensinar todas as dificuldades e encantos da Eletrônica. Agradeço aos professores Heliomar Guzzo e José Borba por aceitarem o convite para compor a banca e por compartilharem com tanta disposição seu conhecimento e experiência adquiridos por longos de anos de trabalho na Indústria. Obrigado, Profa. Raquel Frizera, por sua tutoria em diversas oportunidades, desde o começo do curso de graduação, essencial para minha formação profissional e pessoal. Obrigado, Prof. Alessandro Mattedi, pela incansável dedicação aos seus alunos, se preocupando, antes de mais nada, com a educação e formação das futuras gerações de engenheiros, ofício tão importante para mim, assim como para nosso país. A todos os professores aqui homenageados e a tantos outros não citados, obrigado por compartilharem suas experiências e exigirem dos alunos da UFES do curso de Engenharia Elétrica altos padrões, propondo desafios tão necessários para nosso amadurecimento pessoal e profissional. Agradeço aos amigos de turma, pelo companheirismo e as amizades formadas, vocês são também responsáveis por esta conquista. Em especial aos amigos Vitor Roriz, pela

genialidade e empenho em todos os muitos trabalhos e projetos desenvolvidos em conjunto, Renan Botan, Victor Grobberio e Danilo Marquesini por todas as discussões enriquecedoras e auxílios nos momentos de maior necessidade. Agradeço a todos os amigos do PET/PND pelo apoio e por representarem um jeito de pensar que marca nossa amizade. Agradeço, finalmente, aos colegas e amigos da Sinales, tão importantes e valiosos enquanto pessoas e profissionais em todos os projetos passados, presentes e, seguramente, projeto futuros, dentre os quais cito, especialmente, Leonardo e Anderson pela paciência e ensinamentos de Eletrônica quando os nomes “amplificador operacional” ou “microcontrolador” ainda não tinham nenhum significado para mim.

RESUMO

Este documento trata-se de um projeto de graduação que propõe uma sistemática de síntese de sistemas eletrônicos de alta confiabilidade. A apresentação da sistemática será por meio do desenvolvimento de um controlador semafórico a prova de falhas. O equipamento desenvolvido aplica, em seu projeto conceitual e em sua implementação, métodos de redundância e segurança de sistema para verificação e correção de erros de funcionalidades, software e hardware. Embora a acessibilidade a plataformas abertas tenha facilitado cada vez mais a utilização de sistemas embarcados de baixo custo nas mais diversas aplicações, seu estudo e desenvolvimento com metodologias para aplicações de alta confiabilidade ainda se encontra muito deficiente. Neste contexto, a proposta do projeto inclui a utilização de ferramentas de desenvolvimento robustas que permitem garantir o desenvolvimento de um sistema a prova de falhas, cujos erros, tanto de hardware quanto de software, possam ser detectados, rastreados e tratados. Enquanto estudo de caso, a aplicação de controle de tráfego foi escolhida por se tratar de um sistema que demanda um pleno funcionamento de alta confiabilidade, justificando a utilização de tais ferramentas, da mesma forma que se trata de uma aplicação próxima da realidade acadêmica e do Mercado.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

CCO	Centro de Controle de Operações
BRT	*Bus Rapid Transit*
FMEA	*Failure Mode and Effects Analysis*
SEV	Severity
OCCUR	Occurrence
RPN	Risk priority number
DETEC	Detection
DfR	Design for Reliability
ARQ1	Arquitetura 1
ARQ2	Arquitetura 2
ARQ3	Arquitetura 3
PCRP	Parts Count Reliability Prediction
PSAP	Parts Stress Analysis Predicition
DD-FMEA	Detailed-Design FMEA
VLT	Veículo Leve Sobre Trilhos
ITS	*Intelligent Traffic Syste*ms
PCB	*Printed-Circuit Board*
PF-FMEA	Product Function-FMEA
PI-FMEA	Product Interface-FMEA
GS	Grupo Semafórico
RTOS	*Real-Time Operating System* (Sistema Operacional de Tempo Real)
UFES	Universidade Federal do Espírito Santo

INTRODUÇÃO

1.1 Controle de Tráfego

Na Engenharia de Tráfego, o semáforo é o elemento mais básico. Constituído de 3 focos luminosos acionados de modo sequencial, é possível, adequando os tempos de cada cor, alocar o fluxo de tráfego conforme as necessidades e limitações de cada cruzamento. O acionamento desses focos luminosos é realizado pelo controlador semafórico.

O desenvolvimento da tecnologia de transporte, desde o começo do transporte ferroviário até a consolidação da Indústria Automobilística, transformou imensamente o ambiente urbano. A difusão do automóvel como meio de transporte pessoal contribuiu para um vertiginoso aumento de acidentes de trânsito graves. Nesta nova realidade, a implantação de sistemas de controle de tráfego através de sinalização semafórica, originalmente restrita a sistemas de transporte ferroviário, tornou-se essencial, conferindo ao controlador semafórico, como ilustrado na Figura 1.1.1, grande importância.

Figura 1.1.1- Controlador semafórico Siemens ST900.

FONTE: SIEMENS, 2009

A confiabilidade do sistema de sinalização semafórica reside, assim, tanto no pleno funcionamento do controlador semafórico, responsável pela sequência lógica correta de grupos semafóricos, quanto na confiabilidade de cada um dos focos luminosos acionados [2].

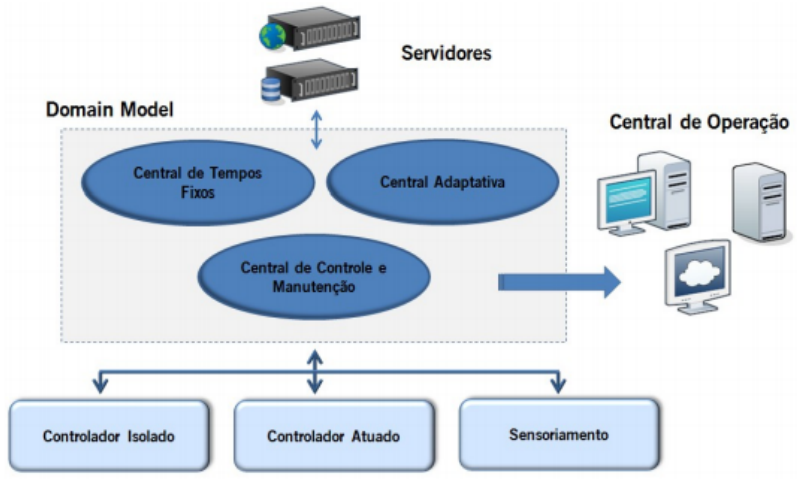
Um controlador semafórico deve ser um equipamento eletrônico de alta confiabilidade, pois o funcionamento de toda a malha semafórica depende da disponibilidade de cada um dos controladores do sistema. O mal funcionamento ou falha de um controlador pode gerar desde uma redução na fluidez do trafego até acidentes fatais.

Quando há falha de acionamento dos semáforos, acidentes ocorrem porque o usuário da via se comporta de modo não determinístico. Entre os modos de falha, a situação de maior risco está em um acionamento defeituoso de dois focos semafóricos verdes de fluxos de tráfego conflitantes. Essa situação normalmente implica em acidentes fatais, já que o usuário não tem conhecimento da falha no controle semafórico e não age com a prudência requerida em caso de falha. Esse comportamento é chamado de "falha de verde conflitante", e trata-se de uma falha não-tolerável para um controlador semafórico [2] [3].

O controlador semafórico deve ser, portanto, um sistema eletrônico embarcado a prova de ou tolerante a falhas. Em seu projeto de hardware e software devem ser tomadas medidas para assegurar o pleno funcionamento do sistema, evitando falhas graves e gerenciando falhas menores. Sendo assim, para o desenvolvimento de um controlador semafórico, desde a concepção do projeto devem ser tomadas medidas para aumentar a confiabilidade do produto final.

A Figura 1.1.2 ilustra o diagrama de um Sistema Inteligente de Transporte, ou ITS (*Intelligent Traffic Systems*). Um sistema de controle de tráfego inteligente pode ser composto por uma rede de controladores semafóricos e uma rede de sensores na malha viária. Esses equipamentos normalmente são controlados por uma central de controle inteligente responsável pela supervisão e otimização do plano de atuação [4]. A rede é supervisionada, possibilitando que critérios de otimização e controle sejam definidos - de modo automatizado ou não - no CCO (Centro de Controle de Operações).

Figura 1.1.2 - Sistema Inteligente de Transporte



Fonte: Produção do próprio autor.

FONTE: Produção do próprio autor

Atualmente, devido à ampla difusão de sistemas inteligentes, o controlador semafórico se tornou um equipamento de múltiplas funcionalidades, sendo capaz de executar diferentes métodos de controle semafórico e gerenciar sensores de tráfego. Essas diversas funcionalidades requerem controladores cada vez mais robustos. Gerenciar recursos de hardware garantindo todas as funcionalidades com níveis de confiabilidade adequados constitui-se, assim, como o principal desafio.

Entretanto, em sistemas com alto grau de integração entre centros de controle, controladores e sensores, uma falha de sistema tem seus efeitos amplificados, propagando a falha em múltiplos cruzamentos semafóricos ou outros subsistemas.

A integração dos controladores com sistemas de otimização de cadeias de logística e meios de transportes diversos, como sistemas intermodais urbanos com presença de VLT, programas de BRT, veículos leves, veículos comerciais, bicicletas, pedestres, integração portuário-ferroviária, integração aeroportuária e outros, também implica em crescente risco dessas aplicações. [18]

1.2 Sistema de alta confiabilidade

Assim como o controlador semafórico, uma série de outros equipamentos e aplicações devem possuir altas exigências de confiabilidade. Esse tipo de aplicação é chamado de *Safety-Critical*

Application. Aplicações industriais de alto risco, aviônica, eletrônica automotiva e equipamentos médicos, por exemplo, normalmente implicam um risco iminente à vida humana em caso de falha, havendo, portanto, a necessidade de alta confiabilidade nesse tipo de equipamento.

Embora haja vasta literatura sobre Engenharia de Confiabilidade e Manutenção [5][7][11][12], a teoria divulgada contempla exclusivamente a análise de confiabilidade. Para realizar uma análise de confiabilidade, entretanto, pressupõe-se que os elementos que compõem um equipamento são conhecidos e que seus parâmetros de confiabilidade também o são. Os parâmetros de confiabilidade mais comuns para equipamentos eletrônicos são o MTBF (*Mean-Time-Between-Failure*) e o MTTF (*Mean-Time-To-Failure*), sendo o último considerado quando não há perspectiva de reparos no equipamento. A Equação 1.2.1 descreve a utilização desse parâmetro (MTTF) para a função de confiabilidade quando se pode atribuir uma distribuição de probabilidade exponencial, adequada para componentes eletrônicos segundo [7]. Outro parâmetro utilizado é a taxa de falha λ, constante no tempo para distribuição exponencial, que é por sua vez o inverso do MTTF.

$$R(t) = e^{-\lambda t} = e^{-\frac{t}{MTTF}}$$

Equação 1.2.1-Função da confiabilidade para equipamentos eletrônicos

O elemento mais crítico para analisar a confiabilidade de um equipamento são os dados de confiabilidade de seus componentes. A qualidade e a veracidade dos dados são de extrema importância, sendo que dados empíricos devem ser prioritariamente utilizados. Por esse motivo há uma tendência, neste tipo de aplicação, em utilizar componentes com mais tempo de mercado, pois sua taxa de falha pode ser estatisticamente comprovada.

Dessa forma, torna-se uma tarefa bastante complicada prever a confiabilidade de um equipamento no momento do desenvolvimento. Por essa razão há pouca literatura descrevendo uma sistemática, ou até mesmo princípios para síntese de sistemas eletrônicos, com alta confiabilidade. Ciclos de tentativa e erro baseados em análise de confiabilidade recorrentes podem ser utilizados, entretanto será determinada uma sistemática para que requisitos de confiabilidade tenham maior chance de serem atingidos, desde a concepção do projeto até seu detalhamento.

1.3 Motivação

Diante do exposto, é de grande interesse o estudo do desenvolvimento de equipamentos de alta confiabilidade aplicados ao controlador semafórico. A motivação deste projeto é justificada por questões de preferência pessoal, profissional e pela situação da indústria brasileira de eletrônica e computação. O projeto de graduação foi concebido de modo a exercitar todas as habilidades e conhecimentos técnicos esperados de um profissional da área. Este tipo de experiência é essencial para adquirir conhecimento com a finalidade de supervisionar um processo de desenvolvimento de produto, com uma visão cada vez mais realista e correta sobre as dificuldades e relações de compromisso envolvidas. Deste modo será possível obter maior experiência nos aspectos do desenvolvimento que impliquem em um melhor produto, tanto do ponto de vista técnico quanto comercial, atendendo aos requisitos de mercado e introduzindo novas funcionalidades. Assim, o tema "Sistemas de Alta Confiabilidade" foi escolhido, porquanto desenvolver maior competência nesta área é extremamente necessário para a indústria de eletrônica no Brasil.

Nos últimos anos houve uma explosão da disponibilidade de ferramentas de plataforma aberta de desenvolvimento. Comunidades de desenvolvimento colaborativas têm transformado em commodity o conhecimento referente à grande parte do mercado de eletrônica. Esta tendência tem sido seguida não só no Brasil, mas principalmente em pequenos e médios pólos de tecnologia pelo mundo. Essa nova realidade confere um grande acesso à informação e grande redução do tempo de desenvolvimento de produtos. Apesar disso, pouca atenção ainda é dada para padrões de desenvolvimento bem definidos visando alcançar altos padrões de qualidade e confiabilidade.

A situação mercadológica apresenta-se tecnologicamente favorável para que muitas empresas *startups* desenvolvam produtos complexos e, portanto, de alto valor agregado. Anteriormente grande parte deste tipo de mercado estava restrita a grandes companhias.

Embora o mercado de tecnologia aparente uma maior democratização, existem problemas sobre os quais as Comunidades Acadêmicas e a profissional não parecem estar atentas. Há pouco esforço em adquirir competência para soluções de sistemas para aplicações críticas com altos padrões de qualidade e confiabilidade. Essa tendência pode criar uma grande segmentação de mercado, permitindo somente o crescimento de empresas em nichos com produtos de baixos volumes, não permitindo, assim, economias de escala.

Seria interessante se essa tendência fosse alterada, pois mercados com produtos de baixas exigências tendem a ser dominados por empresas maiores, capazes de obter grandes economias de escala. Por outro lado, o desenvolvimento de competência em produtos de alta confiabilidade

permite igualar as economias de escala de empresas médias com empresas maiores em muitos setores. Isso é devido ao fato de que produtos de alta confiabilidade tendem a ter demanda inerentemente menor, não permitindo grandes economias de escala, mas possuem preços de venda maiores, assegurando maiores margens.

Dentro deste contexto, o controlador de tráfego configura-se como uma aplicação adequada ao conteúdo teórico que se espera abordar alinhada com uma visão de mercado, convergindo o interesse acadêmico e profissional. Este trabalho visa trazer o tema para discussão dentro do curso de graduação em Engenharia Elétrica da UFES, objetivando ser fonte de informação para trabalhos futuros. O objeto da pesquisa é justificado pela vontade de ajudar colegas interessados nos temas e discussões aqui apresentados para que a Universidade se desenvolva enquanto centro de compartilhamento de conhecimento acadêmico e profissional, bem como ambiente de debate construtivo sobre os temas pertinentes à sociedade.

1.4 Definição do problema

Com objetivo de estudar equipamentos para aplicações de alta confiabilidade, dentro das atuais demandas dos sistemas inteligentes de transporte, este trabalho visa propor uma sistematização para síntese de sistemas eletrônicos de alta confiabilidade. Esta sistemática será posteriormente aplicada ao controlador semafórico proposto.

O controlador semafórico deve, portanto, ser capaz de executar um plano de programação para controlar um cruzamento isolado com apenas quatro grupos semafóricos (GS). O plano de programação semafórico nada mais é que a sequência de cores previstas para serem executadas em cada semáforo com instante e duração pré-determinados. É necessário também que o plano de programação semafórico possa ser agendando e armazenado em memória não-volátil, seja localmente no equipamento ou de forma remota através de um CCO (Centro de Controle de operações). Por essa razão, o controlador deve prever alguma interface de comunicação com a Internet.

Neste cruzamento isolado nenhum dos quatro grupos semafóricos pode estar em verde simultaneamente sob hipótese alguma. Esse é o primeiro requisito funcional que implica em exigência de alta confiabilidade. O acionamento de dois focos verdes ao mesmo tempo implica em risco iminente a vida dos usuários das vias e, por essa razão, o projeto deve ser feito para que a probabilidade de ocorrência dessa falha seja irrelevante dentro da vida útil do equipamento.

A falta de execução adequada de um plano de programação semafórico é, por si só, um risco aos usuários das vias. Semáforos apagados ou focos isoladamente apagados podem induzir um comportamento de risco. Para que isso não ocorra, o controlador semafórico deve mitigar esse tipo de falha entrando em modo piscante, tal como proposto em [2]. O modo piscante será aplicado em todos os focos amarelos do cruzamento. Os motoristas e pedestres, por sua vez, têm conhecimento que esse comportamento dos semáforos exige alerta. Com essa medida entende-se que o risco de acidentes será amplamente reduzido em caso de falha da execução de um plano semafórico.

O controlador deverá acionar focos semafóricos, em conformidade com a norma ABNT NBR 15889:2010. Além disso, a confiabilidade deve ser suficiente para que as falhas descritas nesta seção tenham irrisória probabilidade de ocorrência, considerando uma vida útil garantida de dois anos do equipamento. Propõe-se também que a funcionalidade do modo piscante tenha uma vida útil de cinco anos, promovendo grande conforto ao operador, responsável pela programação e manutenção do equipamento, sob o ponto de vista de segurança de trânsito. Com base nessa definição do problema, este projeto deve quantificar a confiabilidade exigida deste equipamento.

Sabendo-se que há pouca literatura sobre sistemas eletrônicos de alta confiabilidade, o problema de propor uma sistematização da síntese para aplicação no controlador de tráfego se faz necessária. No desenvolvimento do projeto são usados modelos de predição de confiabilidade para adequar o hardware aos requisitos de confiabilidade deste tipo de equipamento. Métodos de identificação e análise de falhas são usados e as especificações terão por finalidade desenvolver o melhor projeto possível capaz de identificar e tratar falhas pertinentes. O equipamento cujo desenvolvimento é proposto deve prever, também, sua aplicação em sistemas inteligentes de tráfego. Esses sistemas devem ter conectividade de forma fácil, barata e robusta, além da capacidade de processamento suficiente para aplicações de métodos de controle adaptativos baseados na medição de fluxo e velocidade nas vias.

1.5 Metodologia

A solução do problema proposto implica na utilização da metodologia e da sistemática para avaliar em todas as etapas do projeto fatores que aumentem a confiabilidade intrínseca do equipamento. Além disso, espera-se que este trabalho esteja estruturado para que a sistemática seja descrita de forma clara. Para que esses objetivos pudessem ser alcançados, segue abaixo a metodologia utilizada.

1. ***Estudo da aplicação***

Primeiramente foi realizado um estudo profundo da aplicação através das normativas e manuais regulatórios da área, bem como literatura teórica pertinente. Esta etapa visa adquirir conhecimento profundo dos tipos de equipamento normalmente utilizados e as necessidades e dificuldades do operador/usuário. Espera-se que ao final desta fase de estudo seja possível listar requisitos funcionais mínimos exigidos.

2. ***Levantamento de requisitos funcionais***

Com base no estudo da aplicação é possível definir os requisitos funcionais esperados do projeto que se pretende desenvolver. Nesta etapa devem ser listados os requisitos funcionais mínimos, juntamente com requisitos funcionais que agreguem características/qualidades diferenciadas pretendidas para o projeto segundo os objetivos propostos neste trabalho.

3. ***Aquisição de base teórica de engenharia de confiabilidade***

Estudo da literatura de Engenharia de Confiabilidade para que seja possível listar e avaliar a síntese e a análise da confiabilidade para sistemas eletrônicos. Pretende-se ao final desta fase ter um conjunto de técnicas de análise dominadas.

4. ***Definição de sistemática para síntese de alta confiabilidade***

Definir uma sequência de passos para que seja realizada uma síntese cujas decisões de projeto tendam a aumentar a confiabilidade e a tolerância a falhas do sistema. Devem ser propostos elementos de análise para especificar a confiabilidade quantitativamente e técnicas de análise de confiabilidade para cada fase do projeto, bem como princípios e auxílio para decisões mais detalhadas de projeto.

5. ***Levantamento de requisitos não funcionais***

Através do conhecimento da aplicação deve-se elencar requisitos não funcionais que impliquem em restrições/especificações de projeto assim como a diminuição da confiabilidade geral.

6. ***Definição da especificação de confiabilidade***

   Etapa de análise das características e requisitos não funcionais do equipamento. Deve-se elencar pontos de risco de falha, restrições para componentes, arquitetura possível, concluindo de forma quantitativa a confiabilidade exigida do projeto com relação às funcionalidades exigidas e maiores falhas a se evitar.

7. ***Análise de hipóteses de arquitetura segundo princípios de confiabilidade***

   Devem ser propostas múltiplas arquiteturas para o sistema e criar um método de avaliação e escolha da melhor arquitetura.

8. ***Análise de modos de falha e seus efeitos para uma arquitetura***

   Deve-se definir o modo de análise de confiabilidade e de possíveis falhas, prós e contras da arquitetura escolhida para prosseguir com o detalhamento do projeto.

9. ***Detalhamento de projeto da arquitetura com análise de confiabilidade quantitativa inicial***

   Detalhar o funcionamento da arquitetura esboçando diretrizes de detalhamento de projeto.

10. ***Detalhamento de circuito com boas práticas de projeto com análise de confiabilidade quantitativa parcial***

    Detalhar os circuitos e propor método de análise de confiabilidade e práticas de projeto.

11. ***Análise de confiabilidade precisa final***

    Esta etapa deve validar o atendimento dos requisitos funcionais, não funcionais e das especificações de confiabilidade.

1.6 Estrutura do Trabalho

O Capítulo 1 contextualiza o leitor sobre a área de controle de tráfego, descrevendo o controlador de tráfego e os riscos de sua aplicação. Além disso, contém breve descrição sobre a área de Engenharia de Confiabilidade é sua aplicação no desenvolvimento de sistemas eletrônicos embarcados. Ao final, o texto explana a motivação seguido pela definição do problema a ser

resolvido, relacionando a área de Engenharia de Confiabilidade e o projeto do equipamento eletrônico.

O Capítulo 2 detalha o controlador semafórico e define requisitos funcionais, apresentando ao final um diagrama de blocos conceitual, que representam os elementos de hardware que devem estar presentes neste tipo de equipamento. Explicitados os problemas, está inserido no Capítulo 3 a apresentação sistemática de síntese para alta confiabilidade. Nessa parte, estão presentes todas as sugestões para tomada de decisão e análises de confiabilidade, correlacionando técnicas observadas na literatura e princípios de projeto propostos neste trabalho.

A partir do Capítulo 4 a sistemática de síntese é aplicada ao controlador semafórico. Neste capítulo é definida a especificação quantitativa de confiabilidade, escolha e detalhamento da alocação de confiabilidade da arquitetura. Também é realizada a análise de modos de falha de maneira a permitir um detalhamento de projeto e análise quantitativa de confiabilidade inicial. Com a definição de arquitetura segue no Capítulo 5 todo o detalhamento de hardware até a análise de confiabilidade final mais precisa.

O projeto final de hardware é, então, apresentado. São explanadas as pequenas decisões de projeto, assim como aplicação de passos da sistemática do Capítulo 3 no projeto detalhado do circuito. Também é feita uma análise quantitativa final do projeto. A análise crítica dos resultados e etapas do projeto está presente no Capítulo 6, contendo as conclusões e trabalhos futuros. Este capítulo contém uma análise geral do andamento do projeto, atendimento ou não dos objetivos propostos, assim como a sugestão de trabalhos futuros com relação à sistemática de síntese e o controlador semafórico.

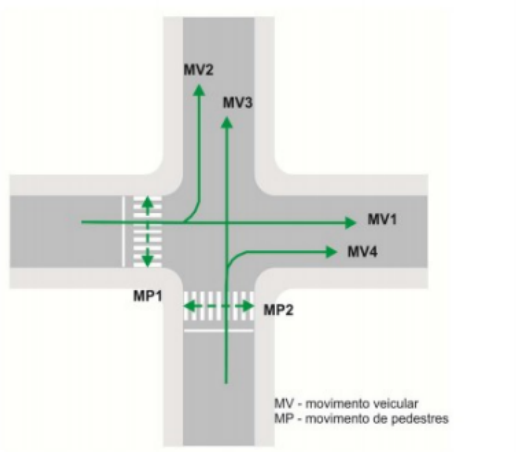
O CONTROLADOR SEMAFÓRICO

2.1 Caracterização da aplicação

Em uma interseção viária com dois fluxos de entrada, Figura 2.1.1, quando o intervalo de tempo entre veículos em um dos fluxos é sempre menor que o tempo necessário para que um veículo do outro fluxo passe pela interseção, é necessário a utilização semafórica. O intervalo de tempo entre veículos, em um determinado fluxo, é chamado de *headway*.

É necessário assegurar a existência de um *headway* tal que sejam criados *gaps* que permitam a passagem de um fluxo conflitante. O número de *gaps* criados é proporcional ao número de veículos de um determinado fluxo que de fato realizam a passagem pela interseção. Essa quantidade de veículos deve ser suficiente para que não haja saturação do cruzamento, ou seja, crescimento de fila nas entradas maior do que a capacidade da infraestrutura [3].

Figura 2.1.1 – Croqui de cruzamento isolado de duas entradas



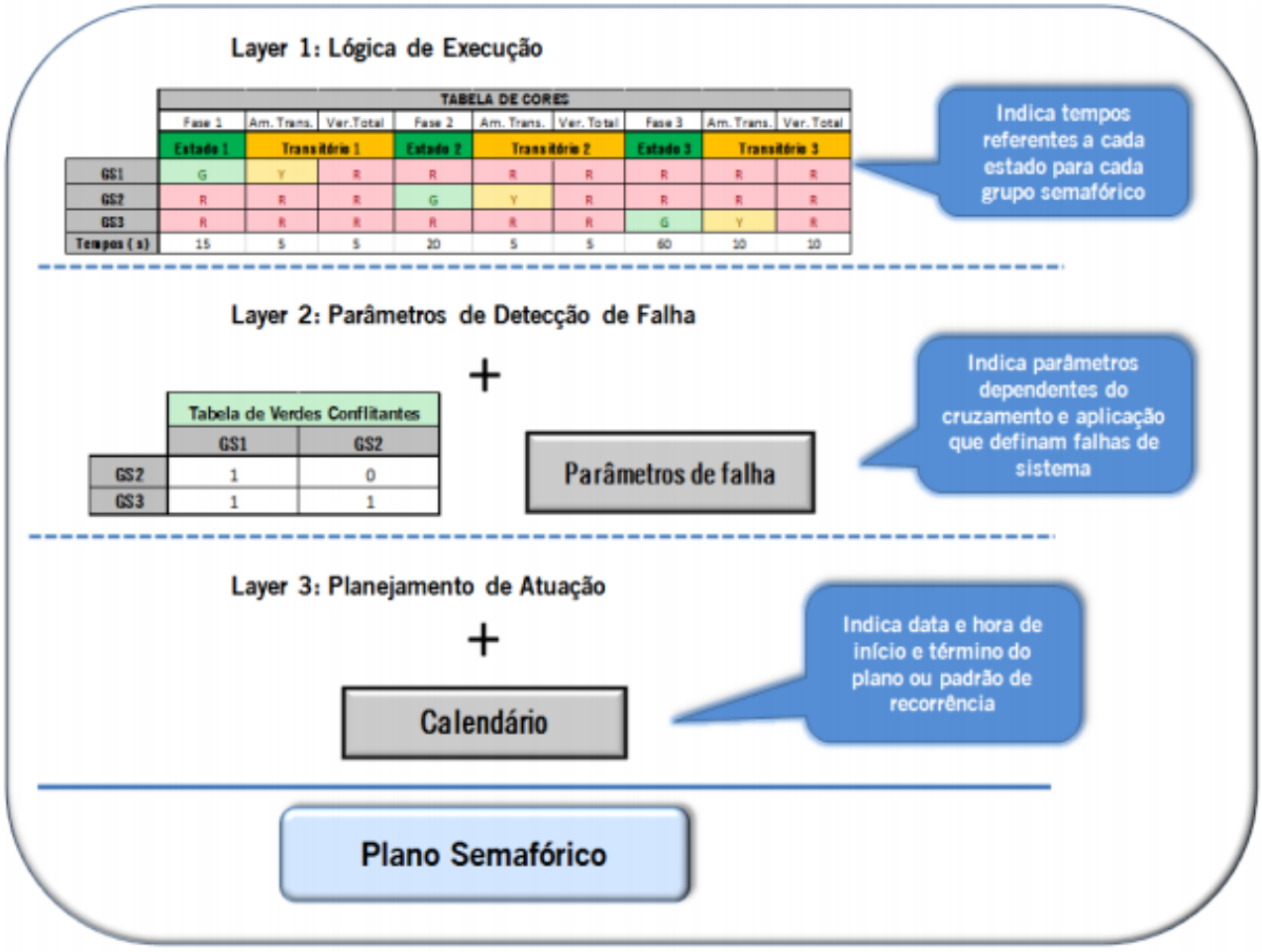
Fonte: MANUAL BRASILEIRO DE SINALIZAÇÃO DE TRÂNSITO, 2014.

FONTE: MANUAL BRASILEIRO DE SINALIZAÇÃO DE TRÂNSITO, 2014

O controlador semafórico proposto neste projeto é o equipamento que deve gerenciar os fluxos de saída de um cruzamento isolado de duas vias de entrada, tal como apresentado na Figura 2.1.2. Entende-se neste trabalho por cruzamento isolado uma interseção que não influencia significativamente nenhuma outra e que não possui qualquer sensoriamento para medição do fluxo

de veículos ou acúmulo de veículos em fila. Dessa forma, o controlador semafórico deve armazenar e executar, para quatro grupos semafóricos, planos de programação semafóricos pré-gravados e agendados. Será considerado que quatro grupos semafóricos serão suficientes para o controle de tráfego desta interseção exemplo escolhida.

Figura 2.1.2 - Plano de programação semafórico



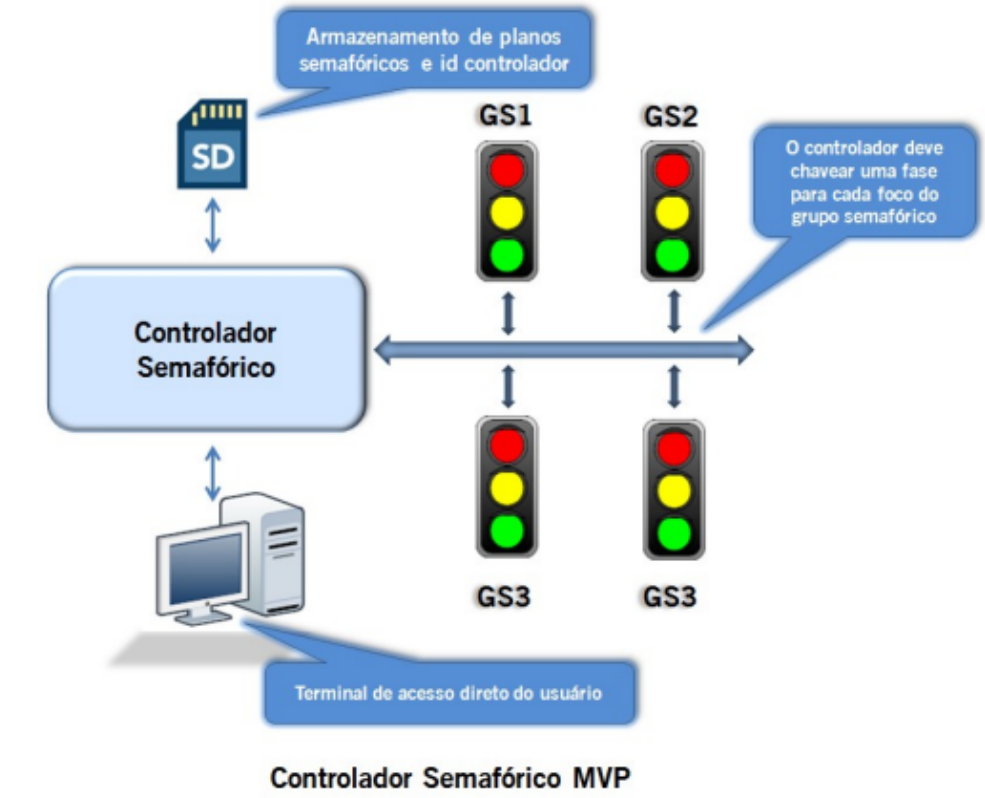
Fonte: Produção do próprio autor.

FONTE: Produção do próprio autor

O plano de programação semafórico, ou simplesmente plano semafórico, é apresentado na Figura 2.1.3 com quantidade de GS inferior à proposta, de modo meramente ilustrativo. Ele é constituído, conceitualmente, de uma tabela de cores, uma tabela de verdes conflitantes e uma tabela de agendamento ou calendário. A tabela de cores indica para todos os GS a duração de cada cor a ser acionada, enquanto a tabela de verdes conflitantes indica quais GS não podem em nenhuma

hipótese estar no estado verde simultaneamente. Parâmetros de falha, como corrente do foco resultante da queima parcial devido a queima de alguns LEDs em um foco com múltiplos LEDs, por exemplo, podem ser incluídos. [2][3]

Figura 2.1.3 - Conceito controlador semafórico



Fonte: Produção do próprio autor.

FONTE: Produção do próprio autor

A Figura 2.1.3 ilustra a atuação do controlador. A partir deste ponto é possível analisar que tipo de recurso de hardware e funcionalidade é necessário para definir os requisitos de sistema para tratamento do problema definido.

O controlador deve armazenar um conjunto de planos semafóricos de forma a compor um plano estratégico de controle que preveja diferentes momentos do dia, visto que há diferentes padrões de fluxo. Cabe ao operador do controlador aplicar normas de engenharia de tráfego para projetar os planos da melhor forma possível, não cabendo ao controlador conferir o impacto dos planos semafóricos na fluidez do trânsito.

Como o acionamento incorreto de qualquer um dos grupos semafóricos implica em um comportamento não previsível do usuário da via, não será permitido o funcionamento parcial do controlador. Quando a sequência lógica acionada pelo controlador está correta, mas existe falha de acionamento em algum dos grupos semafóricos, o controlador é considerado em funcionamento parcial. Para todos os efeitos, qualquer tipo de mal funcionamento que se mantiver após respectivo tratamento de falha deve ser suficiente para colocar o controlador em modo piscante.

Dessa forma, o projeto do controlador semafórico deve ser realizado visando garantir plena capacidade de detectar qualquer falha e acionar o modo piscante durante toda a vida útil do equipamento. Além disso, deve-se garantir a operação em funcionamento pleno com máxima confiabilidade, de forma a não acionar o modo piscante desnecessariamente. [2][3]

Figura 2.1.4 – Diagrama conceitual do controlador

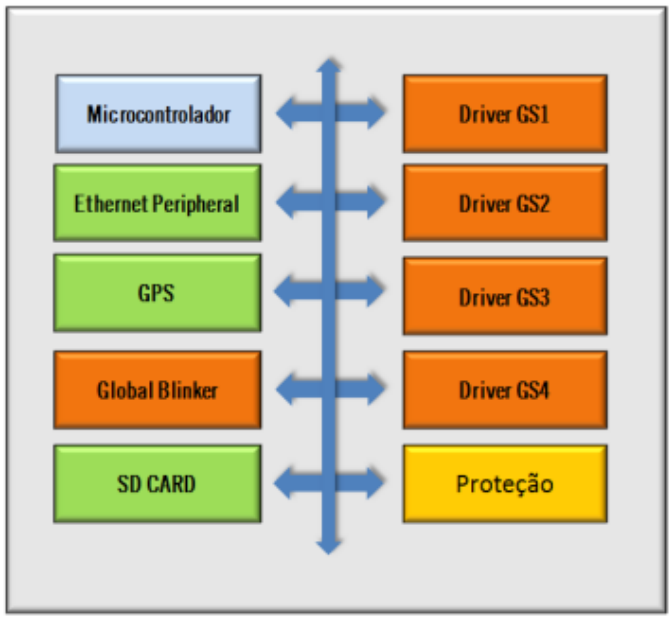


Diagrama Conceitual de Hardware

FONTE: Produção do próprio autor

A Figura 2.1.4 representa o projeto conceitual do controlador semafórico em que cada bloco descreve uma função. *Ethernet Peripheral* é a interface de rede externa, *Global Blinker* gera o sinal para acionamento do modo piscante, e a Proteção corresponde à proteção elétrica do equipamento. Em adequação às necessidades descritas nesta seção, o controlador é definido como um conjunto de módulos de acionamento, chamados no diagrama de DRIVERS, e um conjunto de periféricos digitais

acessados pelo microcontrolador. O projeto deve abranger mínimos recursos de hardware para promover armazenamento não volátil de grande quantidade de planos de programação semafórica, interface de comunicação com computador pessoal, relógio de tempo real e calendário com precisão, verificação de erros de acionamento e circuitos de gerenciamento e supervisão de potência.

2.2 Requisitos funcionais do controlador semafórico

A Seção 2.1 descreve a aplicação para a qual o controlador semafórico deve ser especificado, e através deste conhecimento preliminar pode-se chegar a um diagrama conceitual das necessidades de hardware. Embora tenha-se uma ideia geral das funcionalidades necessárias para o controle de tráfego, é necessário limitar e detalhar cada um dos requisitos que o projeto deverá atender.

Requisito funcional 0
O controlador jamais deve permitir que grupos semafóricos controladores de fluxos de tráfego com sentido e direção conflitantes estejam com seus focos verdes acionados simultaneamente.

Requisito funcional 1
O controlador deve controlar quatro grupos semafóricos. Cada GS também pode ser acionado com as cores amarela e vermelha de forma intermitente.

Requisito funcional 2
O controlador deve armazenar planos de programação semafóricos podendo o controlador ser completamente desligado sem perder essas informações. Ele deve executar o plano devidamente associado com a hora e data corrente.

Requisito funcional 3
O controlador deve possuir recursos de hardware que o permita obter/manter informação de data e hora com precisão e em sincronismo com outros equipamentos deste tipo de aplicação. Prevendo assim utilização do controlador para sistemas mais complexos que um simples cruzamento isolado.

Requisito funcional 4

O controlador deve monitorar focos de cada grupo semafórico, assegurando que todo o sistema esteja funcionando corretamente, mantendo um log de erro do sistema.

Requisito funcional 5

O controlador deve ser capaz de acionar os grupos semafóricos com a potência/tensão/corrente adequadas permitindo que sejam colocados no mínimo dois semáforos por grupo semafórico. Isto deve ser feito, pois é comum que haja um semáforo principal projetado sobre a via e um semáforo menor, chamado repetidor fixado mais baixo.

Requisito funcional 6

O controlador deverá ser capaz de avaliar/monitorar o funcionamento do sistema, armazenando log de erro e informando o operador local ou remotamente em caso de algum erro.

Requisito funcional 7

O controlador jamais poderá deixar de acionar algum foco dos semáforos. Ele deve seguir o plano de programação semafórico e, no mínimo, atuar em modo piscante. Não existe outra opção de lógica de acionamento.

Requisito funcional 8

O controlador deve prever alguma forma de conexão com a internet e deve manter-se sempre conectado para que seja monitorado e/ou sejam gravados novos planos de programação.

Requisito funcional 9

O controlador deve possuir alguma forma de gravação de plano de programação localmente.

Os requisitos funcionais expostos representam o mínimo de funcionalidades de um controlador semafórico competitivo. É importante acrescentar que todas as funcionalidades devem operar com nível de confiabilidade especificado durante toda a vida útil do equipamento, de forma ininterrupta, considerando que o controle de tráfego é uma atividade de natureza contínua. O requisito funcional 0 é, porém, aquele que define a necessidade de alta confiabilidade, pois a inoperância dessa funcionalidade acarreta risco de vida iminente para os usuários das vias. Expostos

esses requisitos funcionais, pode-se propor um primeiro detalhamento da especificação de cada bloco proposto no hardware conceitual da Seção 2.1.

Microcontrolador

Deve ser especificado um microcontrolador com especificações técnicas suficientes para a aplicação proposta assim como ferramentas de desenvolvimento de hardware e software disponíveis.

Periférico Ethernet (*Ethernet Peripheral*)

Deve permitir uma adequação de software que permita ao controlador conectar-se a uma central de controle remota, através de um ponto de rede.

GPS

Para que o relógio do controlador esteja sempre adequadamente preciso, o microcontrolador deve corrigir seu relógio interno através de um GPS.

Memória não volátil (SD Card ou outra)

Os planos de programação semafóricos devem ficar armazenados em cartão SD de modo que seja possível ao usuário retirá-lo e acrescentar novos planos de programação para o controlador, que precisa de, no mínimo, possuir uma memória não volátil suficiente.

Driver GS:

Deve ser projetada uma PCB com circuito de acionamento para um grupo semafórico a fim de ser replicada para mais três grupos. Esta placa de DRIVER deve ser controlada e supervisionada pelo microcontrolador para atender os requisitos de segurança.

Como o objetivo deste trabalho prevê que a síntese do produto seja baseada em métodos que aumente a confiabilidade do equipamento final, qualquer maior detalhamento do projeto deve ser feito baseado em análise de confiabilidade. Os requisitos não funcionais que serão expostos no capítulo 3 deverão ditar as demais especificações e detalhamentos de projeto.

PROJETO DE SISTEMA ELETRÔNICO DE ALTA CONFIABILIDADE

3.1 Fases de Projeto

A síntese de um sistema eletrônico cujos requisitos sejam críticos (*Safety-Critical Applications*) necessita da aplicação concomitante de técnicas de engenharia eletrônica e engenharia de confiabilidade, como estimar e alocar a probabilidade de falha de cada subsistema em cada fase do projeto desde o levantamento de requisitos do equipamento até a produção.

Há uma vasta literatura propondo métodos de análise de confiabilidade, manutenabilidade e disponibilidade. A engenharia de manutenção e confiabilidade tem como objetivo o controle de risco de falhas e indisponibilidade de um equipamento e/ou sistema. Embora haja métodos de análise e predição de confiabilidade com margem de confiança aceitáveis, esses métodos estatísticos normalmente necessitam de dados empíricos que indiquem a probabilidade de falha e distribuição de probabilidade referente a cada falha [MIL-HDBK-338B, Seção 5]. Dessa forma, a teoria de confiabilidade isoladamente não é suficiente para a síntese de um projeto eletrônico capaz de garantir alta confiabilidade de forma sistemática.

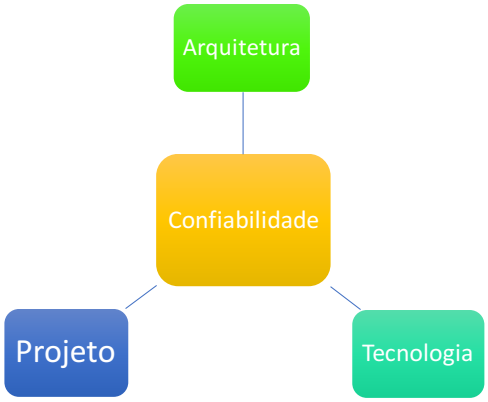
Este trabalho pretende, portanto, apresentar uma sistemática a ser aplicada para projetos eletrônicos de alta confiabilidade, enfatizando os *trade-offs* e técnicas cabíveis. Tal sistemática é demonstrada no projeto exemplo, que é o desenvolvimento de um controlador semafórico. Este tema é designado comumente como *Design for Reliability* e é referido neste trabalho simplesmente com DfR.

A primeira etapa necessária em DfR é detalhar todos os requisitos funcionais e não funcionais do sistema eletrônico que se deseja desenvolver. A especificação detalhada do sistema, correspondente à natureza da aplicação, é de grande necessidade para determinar a especificação de confiabilidade do mesmo.

Em [13] é proposto uma divisão das etapas associadas a DfR. As etapas de síntese estão divididas entre sistema conceitual, projeto preliminar, projeto detalhado, teste de fabricação e suporte de fabricação. Esta divisão proposta pressupõe uma análise e predição de confiabilidade seguida de alteração de projeto em cada uma das etapas. Tendo como base [5][10][11][12][13] pretende-se propor uma sistemática de síntese baseada em três diferentes níveis de abstração, que devem ser detalhados e definidos um a um. A Figura 3.1.1 apresenta os três eixos que determinam de modo geral a confiabilidade de um sistema. São eles a arquitetura do sistema, projeto e tecnologia. Esses

três aspectos contribuem para a confiabilidade final do sistema e, embora sejam apresentados nesta sistemática como três eixos independentes, há correlação intrínseca entre os mesmos.

Figura 3.1.1 - Eixos de Síntese



FONTE: Produção do próprio autor

Com o objetivo de alcançar a confiabilidade exigida pode-se realizar análise e síntese em todos os eixos. No eixo de arquitetura, pode-se modificar o conceito com o qual se deseja implementar as funcionalidades; no eixo de projeto pode-se mudar a implementação da arquitetura utilizando técnicas e componentes que diminuam o risco de falha, mitiguem falhas e eliminem pontos de falhas; no eixo de tecnologia pode-se realizar um controle de qualidade dos componentes e processos de fabricação utilizados no sistema de modo a aumentar a confiabilidade intrínseca do projeto.

Tabela 3.1.1- Fases de Projeto

Fase	Estágio	Objetivo	Descrição	Eixo
1	Idéia	Entender requisitos do sistema	Detalhar especificações do sistema assim como requisitos funcionais, não funcionais, requisitos de confiabiliade e análise de arquitetura do sistema para alocação de confibilidade a nível de sistema e análise de pontos de falhas iniciais através de FMEA (Failure Modes and Effects Analysis)	ARQUITETURA E PROJETO
2	Avaliação	Detalhamento de projeto para confiabilidade e testes	Estudo de mitigação de risco de falha através de FMEA (Failure Modes and Effects Analysis) permitindo detalhar a síntese do sistema assim como teste e avaliação da confiabildiade da tecnologia empregada (HALT e TAAF).	PROJETO E TECNOLOGIA
3	Desenvolvimento	Maturação do projeto e testes	Demonstrar que o projeto de confiabilidade atende os requisitos através de testes significativamente acelerado para comprovadamente a confiabilidade intrínsica normalmente sem introdução de falhas forçadas.	TECNOLOGIA
4	Transição	Validação de produção	Garantir que a produção das unidades seja robusta impedindo diminuição de confiabilidade na produção no começo do ciclo de vida do produto	TECNOLOGIA
5	Produção	Monitoramento da qualidade	Garantir monitoramento constante da qualidade de produção e da confiabilidade apresentada .	TECNOLOGIA

Foi escolhida a proposta de fases de projeto conforme a Tabela 3.1.1, tal como proposto em [12], porém associando os eixos que serão analisados em cada etapa. Deste modo é possível visualizar, baseando-se na descrição atribuída em cada fase, que cada etapa do projeto irá analisar prioritariamente um eixo, visando sistematizar o processo de síntese. Considerando que o projeto desenvolvido neste trabalho não constituirá um produto comercial, assim como há limitações financeiras e de tempo, serão detalhadas somente as três primeiras fases. Com a fase de ideia, avaliação e desenvolvimento será possível apresentar uma sistemática de síntese que englobe cada um dos três eixos.

3.2. Especificação de Confiabilidade

O projeto de qualquer sistema de alta confiabilidade deve ser idealizado e implementado para atingir a confiabilidade requisitada. O objetivo, então, é alcançar a especificação de confiabilidade, que é consequência direta da natureza da aplicação, incluindo requisitos funcionais e não funcionais. Em [5] são elencados os elementos essenciais que devem ser aferidos para que a especificação de confiabilidade de um sistema seja determinada.

ELEMENTOS PARA ESPECIFICAR CONFIABILIDADE:

(1) Sentença quantitativa da confiabilidade requerida.

(2) Descrição completa do ambiente no qual o equipamento/sistema será armazenado, transportado, operado e reparado.

(3) Identificação métrica de tempo clara para descrever a disponibilidade requerida (horas de operação, horas de voo, ciclos, etc.) ou perfil da missão.

(4) Definição clara do que constitui falha.

(5) Descrição dos procedimentos de teste e critérios que serão usados para especificar a confiabilidade.

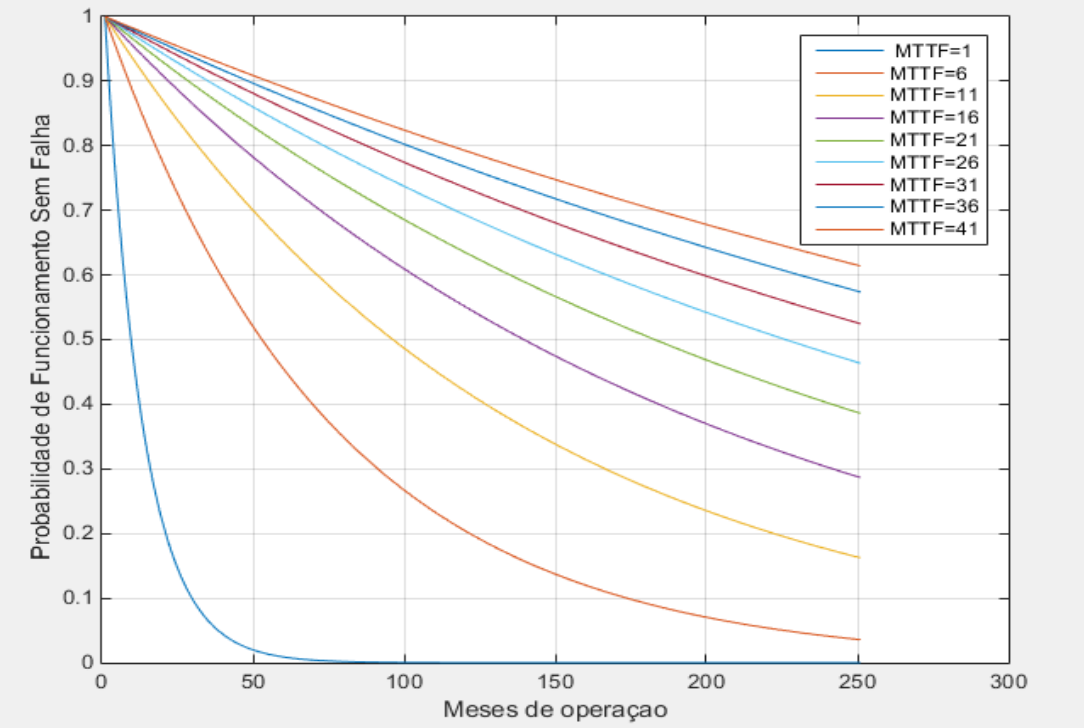
Sabe-se, então, que a especificação de confiabilidade deve ser função da aplicação, que no contexto militar seria declarado como "perfil da missão", e deve ter natureza quantitativa bem definida. Além disso, deve ser claro o momento no qual o equipamento seja considerado em situação de falha ou indisponível. O MTBF (*Mean-Time-Between-Failure*) ou MTTF (*Mean-Time-To-Failure*) são duas medidas comumente utilizadas para avaliar a confiabilidade de um sistema, e são adequadas quando a distribuição de probabilidade da confiabilidade não é crítica, podendo ser considerada exponencial (probabilidade de não ocorrer falha em até um determinado instante cai exponencialmente no tempo). Considerando que a área de aplicação deste trabalho são sistemas eletrônicas de alta confiabilidade, segundo [7], a distribuição de probabilidade exponencial é adequada para descrever a confiabilidade.

$$R(t) = e^{-\lambda t} = e^{-\frac{t}{MTTF}}$$

Equação 3.2.1- Função da confiabilidade para equipamentos eletrônicos

Conhecendo a Equação 3.2.1, pode-se traçar as curvas de probabilidade de funcionamento sem nenhuma falha no tempo, como pode ser observado na Figura 3.2.1, pra escolher um MTTF correspondente à vida útil necessária para um determinado equipamento.

Figure 3.2.1 - Curvas de confiabilidade para MTTF variados



3.3. Métodos de Análise e Alocação de Confiabilidade

3.3.1. Princípios de Projeto

Embora métodos de análise e predição de confiabilidade possam ser utilizados em equipamentos diversos, serão enfatizadas aplicações para sistemas eletrônicos, tendo como foco o controlador semafórico. Sendo assim, os princípios e métodos propostos pelos manuais MIL-HDBK-338B [7], MIL-HDBK-217F [5] e a norma MIL-STD-756B [6] foram julgados como os mais adequados e comumente aceitos para equipamentos eletrônicos. Primeiramente é proposta uma lista de princípios de projeto de circuito com confiabilidade que devem guiar a definição preliminar de

arquitetura. Posteriormente deve ser feita a previsão e análise de confiabilidade da arquitetura com a finalidade de propor alterações de arquitetura e projeto que aumentem a confiabilidade. Para que sejam reduzidos possíveis pontos de falha deve-se ter como base os seguintes princípios:

PRÍNCIPIOS DE PROJETO DE CIRCUITO PARA AUMENTO DA CONFIABILIDADE

(1) Manter componentes em sua região de operação
(2) Reduzir números de conectores
(3) Reduzir complexidade do circuito
(4) Aumentar redundância de subsistemas
(5) Reduzir transitório e problemas de alimentação
(6) Aumentar a qualidade dos componentes
(7) Reduzir o número de sinais paralelos entre conectores
(8) Reduzir o número de interfaces de comunicação de dados em subsistemas

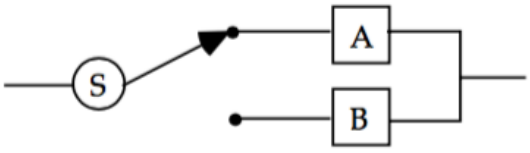
Embora haja princípios que guiem o projeto desde o princípio, cada eixo de confiabilidade pode influenciar o outro. Segue abaixo a descrição de cada princípio e as consequências intrínsecas de se priorizar um ou outro durante o projeto.

(1) As condições ambientais descritas nas especificações de confiabilidade devem guiar o projeto desde o início. Esse princípio inclui o eixo de projeto e tecnologia, pois prevê que os componentes utilizados devem operar nas condições do equipamento com confiabilidade e, além disso, o projeto do circuito deve ser tal que garanta sua correta utilização. Essa é uma premissa básica para qualquer projeto com difícil atendimento em função das restrições ambientais, que são muito críticas.
(2) Conectores são componentes mecânicos altamente sujeitos a defeitos inerentes, falhas de produção, falhas de operação, vibração e até mesmo temperatura em caso de dimensionamento incorreto do conector em função das características dos sinais. Embora o conector possa ser fornecido de forma a atender precisamente as especificações, eles constituem pontos de falhas devido a sua própria natureza, podendo ocasionar circuito aberto ou curto com propagação de falha. Além disso, segundo [14], não existe sistemática de projeto e produção de conectores que consiga mensurar em sua síntese os efeitos da degradação dos conectores em função do tempo e condições ambientais a que são expostos, prejudicando assim a confiabilidade geral deste tipo de componente.

(3) A complexidade do circuito, medida usualmente pela quantidade de componentes necessária para funcionamento correto, implica na introdução de pontos de falha não previstos. Qualquer intervenção no projeto com objetivo de aumentar a confiabilidade pode aumentar demasiadamente a quantidade de componentes do circuito, o que deve ser analisado recorrentemente. A confiabilidade em função da quantidade de componente, ou complexidade, pode ser analisada através do método *Part Count Reliability Prediction,* descrita no apêndice A do MIL-HBDK-217F[5].

(4) Cada subsistema pode ter sua confiabilidade aumentada com introdução de redundâncias. Essa prática, porém, deve ser utilizada com cuidado, pois pode introduzir pontos de falhas, além de aumentar a complexidade do sistema. O grande *trade-off* de projeto de circuitos eletrônicos confiáveis está entre aumentar a confiabilidade através do detalhamento de projeto (redundâncias, intertravamentos e monitoramento de falhas) ou alocar a confiabilidade no eixo de tecnologia, que seria manter o circuito o mais simples e pequeno possível utilizando componentes com alta confiabilidade. Entre os tipos de redundâncias mais comuns estão inclusas a redundância paralela ativa e a redundância paralela *stand-by*. Na primeira supõe-se que a redundância começará a atuar naturalmente em caso de falha; a segunda, porém, trata-se do acréscimo de mais um subsistema, denominado *Voter*, descrito como bloco S na Figura 3.3.1.1. O *Voter* é quem irá monitorar os dois subsistemas redundantes e, assim, decidir qual dos dois deve atuar em um determinado momento. No segundo caso existem várias possibilidades de gerenciar qual subsistema redundante irá atuar, ainda assim, o *voter* torna-se um ponto de falha, assim como aumenta a complexidade do circuito.

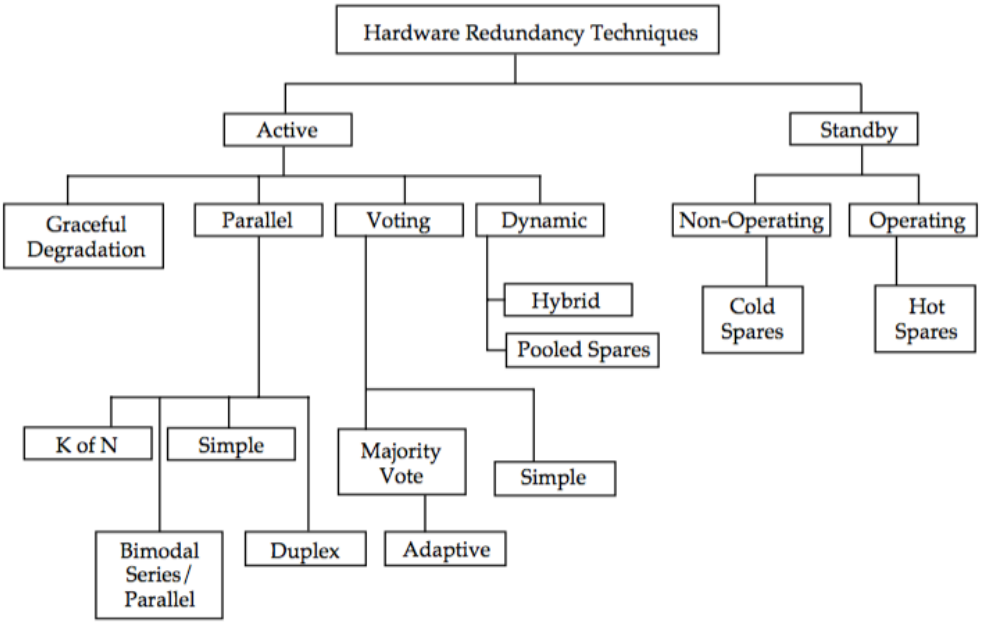
Figura 3.3.1.1 - Redundância com *voter*



FONTE: [7]

Embora as técnicas de redundâncias estejam divididas em dois principais tipos, tal como já exposto, o manual MIL-HBDK-338B [7] apresenta grande variedade de redundâncias. Além dos tipos de redundância já apresentadas, a Figura 3.3.1.2 demonstra as outras variedades, sendo que aquelas que de fato serão utilizadas neste trabalho serão explanadas de forma detalhada durante sua aplicação.

Figura 3.3.1.2- Árvore de técnicas de redundâncias



(5) Considerando causas independentes da aplicação específica, a mais comum causa de falha em equipamentos eletrônicos é a falta de estabilidade e de confiança do sistema de alimentação. Sendo assim, é imprescindível que sejam utilizadas técnicas de circuito para aumentar o isolamento e proteção elétrica do sistema.

(6) O controle de qualidade de componentes eletrônicos é um problema a ser analisado. Componentes de uso genérico como resistores e capacitores podem apresentar altas taxas de falhas, comprometendo assim a confiabilidade geral do sistema. Além disso, é preciso prever o *derating* de cada componente, assim como as demais fontes de aumento de taxa de falha, usualmente analisadas através do método de *Part Stress Analysis Prediction,* descrita em [5]. Esse método deve ser aplicado no circuito em um estágio avançado do projeto quando os componentes já tiverem sido especificados. O resultado dessa análise pode implicar alteração no eixo de tecnologia, ou seja, requerendo alteração da especificação de um componente ou alteração de projeto detalhado inserindo redundância no eixo de projeto do circuito.

(7) Sinais digitais de baixa tensão e corrente, comumente chamados de sinais fracos, estão altamente sujeitos a ruído e má interpretação de nível lógico. Sendo assim, deve-se evitar que esses sinais estejam propensos a má intepretação e ruído, reduzindo sua quantidade ao máximo, principalmente em conectores. Uma boa medida é blindá-los quando possível. Além disso, pode-se combinar um sinal com o seu conjugado lógico, facilitando, assim, a detecção de inconsistências. Essa medida, porém, pode aumentar a complexidade do circuito.

(8) Interfaces de comunicação entre subsistemas digitais constituem pontos de falha críticos, pois falha de transmissão de apenas um *bit* pode comprometer a consistência do dado, gerando uma falha que pode ser difícil de ser detectada. Para mitigar estes problemas deve-se procurar reduzir ao máximo as interfaces de comunicação digital, assim como dimensionar corretamente o canal de comunicação (conectores e/ou trilhas da placa de circuito impresso). A escolha do protocolo de comunicação adequado para as características de hardware e de confiabilidade exigidas é um fator fundamental, tendo como princípio que o mais simples e com menor taxa de falha deve ser o escolhido.

Tendo como base os princípios para confiabilidade aqui propostos, esta etapa da síntese requer a execução dos passos abaixo.

1. Proposição de hipóteses de arquitetura concebidas através dos princípios para confiabilidade
2. Avaliação qualitativa de vantagens e desvantagens de várias arquiteturas
3. Criação de critérios para avaliação de confiabilidade baseados nos princípios para confiabilidade e proposição de notas de zero a dois para cada critério de cada hipótese de arquitetura.
4. Listagem de critérios operacionais de avaliação baseados em características funcionais/complexidade de desenvolvimento e proposição de notas de zero a dois para cada critério de cada hipótese de arquitetura.
5. Quantificação de nota parcial para cada arquitetura, tanto para critérios de confiabilidade quanto operacionais, e equalização de valores, pois cada classe de critérios pode ter quantidades diferentes de critérios listados.
6. Proposição de pesos para critérios de confiabilidade e operacionais. Sugere-se 90% para nota parcial dos critérios de confiabilidade e 10% para nota parcial dos critérios operacionais.
7. Composição da nota final como sendo a soma das notas parciais ponderadas com pesos escolhidos.
8. Escolha da arquitetura com maior nota para prosseguimento da síntese para confiabilidade.

3.3.2. Métodos de Análise

Escolhida a arquitetura deve-se detalhar o projeto para que seja realizada a análise de confiabilidade intrínseca do projeto. Entende-se por confiabilidade intrínseca a confiabilidade do sistema em executar sua função plena sem nenhuma falha após implementadas as redundâncias e

tecnologias de componentes pertinentes. O detalhamento do projeto será norteado portanto, por políticas de mitigação de falha como FMEA (*Failure Mode and Effect Analysis*) [11] e métodos de análise de confiabilidade intrínseca, como *Part Count Reliability Prediction* e *Part Stress Analysis Prediction* [5][7], que serão aplicados respectivamente no começo do detalhamento do projeto e ao final do projeto.

Part Count Reliability Prediction (PCRP)

É aplicado nos estágios iniciais do projeto e pode servir de parâmetro para propor um limite de complexidade/quantidade de componentes de cada escolha de projeto final. Para aplicar este método são necessárias as taxas de falha genéricas para cada tipo de componente, nível de qualidade para cada tipo, condições ambientais sob a qual cada tipo de componente está exposto e quantidade aproximada de cada um dos tipos de componentes. Cada projeto terá um quantitativo e tipos de componentes diferentes, assim como diferentes condições ambientais. As taxas de falha genéricas são propostas por [7], assim como níveis de qualidade de componentes conforme suas tecnologias de fabricação.

$$\lambda_{equipamento} = \sum_{i=1}^{i=n} N_i(\lambda_g \pi_Q)_i$$

Equação 3.3.2.1- Confiabilidade por PCRP

$\lambda_{equipamento}$ = Taxa de Falha geral do equipamento (Falha/10ˆ6 horas)
λ_g = Taxa de falhas genérica do tipo de componente genérico i
π_Q = Fator de qualidade do tipo de componente genérico i
N_i = Quantidade total do tipo de componente genérico i
n = Quantidade dos diferentes tipos de componente genéricos presentes

Para um equipamento em uma determinada condição ambiental, a Equação 3.3.2.1 fornece uma aproximação inicial à confiabilidade do sistema considerado.

Tendo este conhecimento é possível realizar uma análise de confiabilidade assim que o projeto começar a ser detalhado. Sendo assim, serão considerados dez tipos de componentes genéricos:

1. RESISTORES
2. CAPACITORES
3. CIRCUITOS INTEGRADOS 1-100 GATES
4. CIRCUITOS INTEGRADOS MICROPROCESSADORES
5. MEMÓRIAS EEPROM
6. MEMÓRIAS FLASH
7. CONECTORES DIGITAIS
8. CONECTORES DE POTÊNCIA
9. OPTOACOPLADORES/ISOLADORES
10. DISPOSITIVOS SRC/TRIAC

Todos os dez tipos de componentes genéricos propostos possuem equivalentes em [5], permitindo assim a aplicação deste método.

Part Stress Analysis Prediction (PSAP)

Este método é a forma definitiva para mensurar uma aproximação realista da taxa de falha final do equipamento e, assim, a sua confiabilidade intrínseca. Uma posterior análise de confiabilidade intrínseca mais precisa só poderá ser aferida empiricamente através de ensaios de envelhecimento acelerado do equipamento. O resultado dessa análise permite alterações no eixo de tecnologia do produto, podendo indicar corretamente mudanças na especificação dos componentes escolhidos.

A aplicação do PSAP requer que seja conhecida a taxa de falha de cada um dos componentes utilizados no circuito. Esse requisito pode ser difícil de se alcançar, pois a grande maioria dos componentes comercialmente disponíveis não possui relatórios de confiabilidade efetuados com controle de qualidade adequado. A disponibilidade de relatório de confiabilidade de cada componente será um fator determinante para a seleção dos componentes do controlador semafórico, para que seja possível a aplicação de PSAP com dados reais.

O manual MIL-HDBK-217F propõe também taxas de falha genéricas para vários tipos de componentes, porém desta vez com um nível de detalhamento maior. Também é especificado em [5] a divisão dos componentes em seis grupos: Microcircuitos, Semicondutores Discretos, Capacitor de Confiabilidade Estabelecida, Resistores de Confiabilidade Reconhecida, Indutores e Antenas de

Confiabilidade Estabelecida, Relés com Confiabilidade Estabelecida. Cada uma dessas classes possui diferentes classificações de qualidade, sendo que esses descritivos de qualidade são normalmente encontrados na especificação de componentes comerciais.

Há também classificações padronizadas para as condições ambientais e de operação [5]. Considerando o projeto do controlador semafórico, esta classificação pode ser tanto Gb quanto Gf, sendo que a descrição dessas classificações definida em [5]. A condição Gb é mais adequada para o controlador semafórico. Essa classificação já contempla as temperaturas de operação previstas na especificação de confiabilidade do controlador, assim como demais características operacionais do equipamento.

$$\lambda_p = \lambda_b \pi_T \pi_A \pi_R \pi_S \pi_C \pi_Q \pi_E$$

Equação 3.3.2.2-Model de taxa de falha de um componente

λ_p = Taxa de falha do componente no equipamento

λ_b= Taxa de falha base inerente do componente

π_T= Fator de correção para simular o efeito da temperatura

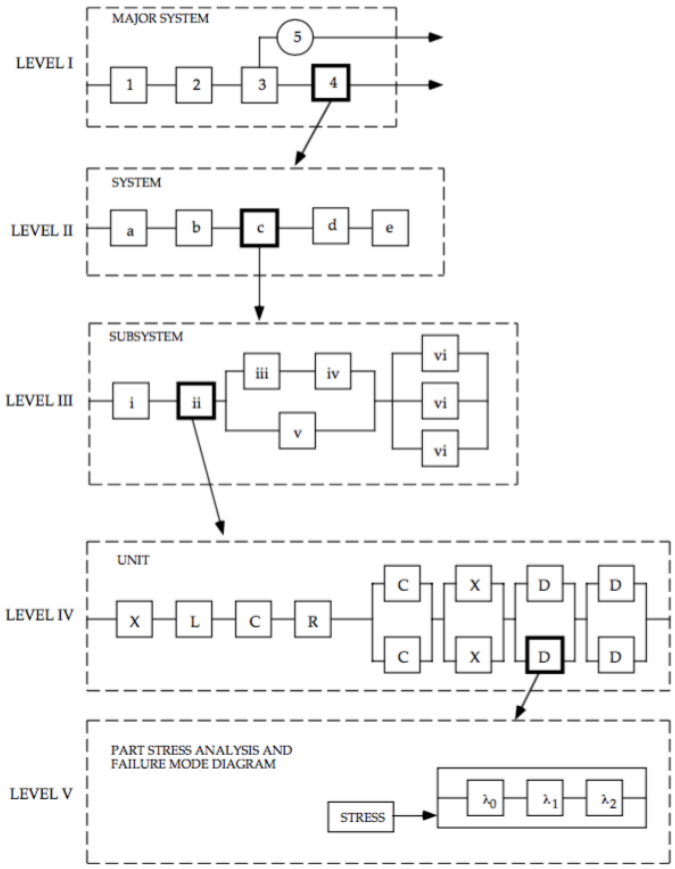
π_Q= Fator de correção devido a qualidade de produção do componente

π_E = Fator de correção devido as condições ambientais do equipamento

A Equação 3.3.2.2 fornece um modelo empírico para calcular a taxa de falha de um único componente quando exposto às condições ambientais de operação do equipamento. Os fatores descritos são os que de fato serão utilizados, pois são considerados suficientes por [5], dado que a taxa de falha base adotada seja realista. No decorrer do projeto são sempre escolhidos componentes que possuem relatórios de confiabilidade empíricos disponíveis, permitindo, assim, que as taxas de confiabilidade adotadas para cada componente sejam as taxas reais.

Calculadas as taxas de falha de cada componente, deve-se dividir o sistema em subsistemas, e cada subsistema em componentes, para que seja montado o diagrama de blocos associado à confiabilidade. Nesse diagrama, subsistemas redundantes e independentes são apresentados como blocos em paralelo, enquanto subsistemas/circuitos com relação de confiabilidade dependentes são apresentados em série. A confiabilidade final do sistema é adquirida através da teoria de confiabilidade, que calcula a confiabilidade global através da taxa de falha de cada bloco [5, Seção 6].

Figura 3.3.2.1-Diagrama de Blocos para confiabilidade



FONTE: [7]

3.4. Análise dos modos de falha e seus efeitos (FMEA)

Definida a arquitetura, deve-se começar a detalhar o projeto eletrônico e a principal ferramenta para garantir confiabilidade durante o processo de síntese é o método FMEA (Failure Mode and Effects Analysis). O método FMEA é abrangido pela norma ABNT NBR 5462 (1994), porém a primeira sistematização desta análise está contida no MIL-STD-1692A, norma militar norte-americana. Considerando que esse mesmo método tem grande aplicação em controle de qualidade total em processos industriais, gerenciais e principalmente em engenharia de manutenção, foi

adotado a abordagem de FMEA apresentada em [11, cap.12], pois entende-se ser a mais adequada para síntese para confiabilidade (DfR).

Failure Mode and Effect Analysis (FMEA) trata-se de sistematicamente analisar os possíveis modos de falha inerentes da aplicação e da arquitetura escolhida, relacionar causa e efeito, bem como propor ações para mitigar a falha ou impedir que ela ocorra. As ações de mitigação de falha irão compor as especificações do projeto detalhado, já que o *hardware* deve estar preparado para executar as ações previstas.

Em [11] é proposta a realização da FMEA para cada uma das fases de projeto descritas na Seção 3.1. O proveito será maior para o escopo deste projeto e para sistematizar a DfR caso seja aplicada uma das variantes da FMEA: *Product Interface* FMEA.

Primeiro deve-se analisar uma arquitetura, escolhida com base nos princípios de confiabilidade da Seção 3.3.1, depois deve ser feita a *Product Interface* FMEA (PI-FMEA) ou *Product Function* FMEA (PF-FMEA) [11]. Propõe-se a aplicação de um misto entre as duas formas de FMEA, que será chamada somente de PI-FMEA. Com o resultado desta análise estarão encerradas as avaliações de síntese no eixo de arquitetura (Seção 3.1). A PI-FMEA irá guiar a síntese inicial do eixo de projeto, possibilitando a definição de um projeto eletrônico preliminar que atenda ao máximo os requisitos funcionais e não funcionais especificados. Segue exemplo de PI-FMEA na Tabela 3.4.1.

Tabela 3.4.1 - PI-FMEA exemplo controlador semafórico

Failure Mode and Effect Analysis (PI-FMEA)																
Sistema: **Controlador Semafórico Arquitetura 3**			Blocos: **N/A**								**Número da FMEA:**	1				
Subsistemas em Análise : **CPU, BLINKER, DRIVER** Componentes: **N/A**																
Item	**Função do Item/ Funcão em falha**	**Failure Mode Potential**	**Efeitos Potências da falha**	**SEV**	**CLASS**	**Causas Potências/Mecanismo da Falha**	**OCCUI**	**DETEC**	**RPN**	**Ação Recomendada**	**Resultados de Ação**					
											Medida de Mitigação	**SEV**	**OCCUR**	**DETEC**	**RPN**	
CPU	Manter a alimentação em níveis adequados para própria operação	Falha na fonte de alimentação	Não acionamento lógico de nenhum GS	9	FHNR	Surtos de tensão e corrente/ sobrecarga/ sobretemperatura	2	2	36	Outra unidade assumir o acionamento lógico	Criar redundância de alimentação através de bateria	9	2	2	36	
		Falha no circuito de regulação da CPU	Não acionamento lógico de nenhum GS e falha não reparável da CPU	6	FHNR	Sobre temperatura/sobre tensão/sobrecorrente/ curto na PCB	4	4	96	Outra unidade assumir o acionamento lógico	Aumentar confiabilidade do circuito de regulação ou realizar redundância da CPU	6	4	4	96	
		Falha no conector	Não acionamento lógico de nenhum GS	6	FHR	Mal contato/ conector danificado/conector aberto	6	2	72	Outra unidade assumir o acionamento lógico	Criar redundância de conector de alimentação	6	6	2	72	

Durante a FMEA foram identificados possíveis pontos de falha e sugestões de projeto para mitigação ou eliminação dos pontos de falha. Com a finalidade de mensurar quais modos de falhas possuem maior gravidade, probabilidade de ocorrência e probabilidade de detecção, foi aplicado um sistema de pontuação abrangendo esses três critérios.

A avaliação de cada critério é subjetiva, baseando-se no conhecimento técnico e experiência da equipe responsável pela formulação da FMEA. O caráter qualitativo de avaliação faz com que a escolha da equipe responsável pela análise seja extremamente importante. Propõe-se aqui, assumido o risco desta escolha, que ao menos três equipes completamente independentes, porém com completa compreensão do equipamento e da aplicação, realizem análises FMEA simultâneas para que seja obtida uma média das avaliações mais próximas da realidade e isentas de vícios de julgamento.

Cada modo de falha recebe uma nota final chamada *Risk Priority Number*, ou simplesmente pela sigla, RPN, calculada segundo a Equação 3.4.1.

$$RPN = SEV * DETEC * OCCUR$$

Equação 3.4.1 - Cálculo do RPN

Risk Priority Number (RPN)

Medida quantitativa que representa a prioridade para mitigar ou eliminar as possíveis causas de falha para um respectivo modo de falha. O valor do RPN deve ser interpretado conforme a Tabela 3.4.2 [11].

Tabela 3.4.2- Interpretação do RPN

Interpretação do RPN	
Rank	**Guideline**
1 < RPN < 18	Pouco risco
18 < RPN < 64	Risco moderado. Este nível requer validação seletiva de componente, validação e teste do projeto assim como caracterização bem definida dos processos envolvidos para minimizar RPN.
64< RPN	Grande risco. Requer revisão contínua de projeto extensivo e análise dpara reduzir RPN

Severity (SEV)

Medida quantitativa que representa a gravidade das consequências das falhas para o sistema e para o usuário/cliente do sistema. O valor do SEV deve ser escolhido conforme a Tabela 3.4.2 [11].

Tabela 3.4.2 - Padrão de avaliação de SEV

Padrão de avaliação de *Severity* (SEV)		
Rating	**Guideline**	**Nota**
Muito alta	Indica um modo de falha potencial potencial que pode causar morte (9 com alarme e 10 sem alarme)	9 ou 10
Alta	Alta insatisfação do cliente em razão da falha, como um subsistema inoperante cuja funcionalidade é um dos requisitos funcionais principais. (ex. Motor de um carro)	8
Alta a moderada	Falha em sistem que pode estar inoperante, que não envolve aspecto imediatos de segurança	7
Moderada	Falha causa alguma insatisfação no cliente	6
Moderada a baixa	Desconforto ao cliente causado pela falha	5
Baixa	Cliente irá perceber a deterioração de um sistema	4
Baixa a muito baixa	Falha causa apenas pequeno desconforto, sendo notado queda de desempenho	3
Muito baixa	Não se espera que a falha cause nenhum problema maior momentâneo ao sistema	2
Insignificante	A maioria dos clientes nem perceberá	1

Occurence Probability (OCCUR)

Medida quantitativa que representa a probabilidade de ocorrência da falha. O valor de OCCUR deve ser escolhido conforme a Tabela3.4.3 [11].

Tabela 3.4.3 - Padrão de avaliação de OCCUR

Padrão de avaliação de *Ocurrence* (OCCUR)			
Taxa de Falha	**Ocorrência da falha**	**Nota**	**Ocorrência por unidade de tempo**
Muito alta	Falha praticamente inevitável	10 ,9	50%/33%
Alta	Falha repetitiva	8 , 7	12.5 %/5%
Moderada	Falha ocasional	6,5,4	1.25%/0.25%/0.05%
Baixa	Relativamente poucas falhas	3,2	666 PPM/ 6.66 PPM
Remota	Falha improvável	1	0.66 PPM

Detection Capability (DETEC)

Medida quantitativa que representa a habilidade do sistema de detectar a causa da falha para ações de mitigação. O valor de DETEC deve ser escolhido conforme a Tabela 3.4.3 [11].

Tabela 3.4.3 – Padrão de avaliação de DETEC

Padrão de avaliação de *Detection* (DETEC)		
Chance de Detecção	**Guideline**	**Nota**
Certeza de não detecção	Monitoramento não conseguirá detectar mecanismo da falha	10
Muito baixa	Monitoramente provavelmente não conseguirá detectar falha	9
Baixa	Monitoramente tem baixa chance de detectar falha	8,7
Moderada	Monitoramento pode detectar falha	6,5
Alta	Monitoramento tem grandes chances de detectar falha	4,3
Muito alta	Monitoramento com certeza irá detectar falha	2,1

Classe (CLASSE)

O sistema de classificação de modos de falha é de escolha da equipe de desenvolvimento no caso de não haver normatização ou sugestão na literatura que melhore as decisões de síntese para que se obtenha maior confiabilidade. Este trabalho propõe, porém,

diferenciar os modos de falha classificando em conformidade com a natureza da falha e com a ação necessária para mitigação ou eliminação a falha.

$$FHNR = Falha\ de\ hardware\ não\ reparável$$
$$FHR = Falha\ de\ hardware\ reparável$$
$$FSNR = Falha\ de\ software\ não\ reparável$$
$$FSR = Falha\ de\ software\ reparável$$

Compreende-se como "não reparável" uma falha tal que o sistema não tenha nenhuma ação apropriada para manter o pleno funcionamento. Falha de hardware não reparável seria, por exemplo, a falha de todas as possíveis fontes de alimentação de um equipamento. Entende-se por "reparável" uma falha que não impede a operação do equipamento como, por exemplo, a queda de um subsistema que possui redundância ou a falha de alimentação momentânea.

Definidos assim os critérios de avaliação, espera-se que a aplicação da PI-FMEA permita reunir os modos de falhas com maior RPN, sugerindo decisões de projeto que mitiguem ou eliminem estas possibilidades de falha. Essas decisões irão guiar o projeto detalhado posteriormente.

3.5. Diretrizes de projeto de circuito para confiabilidade

Realizada a PI-FMEA da arquitetura escolhida, considera-se finalizada a síntese no eixo de arquitetura, como exposto na Seção 3.1. As sugestões para mitigação de falha avaliadas na PI-FMEA servirão de base para a síntese no eixo de projeto. Entretanto, deve-se levar em consideração uma série de fatores causadores de falhas em circuitos. Serão expostas técnicas de síntese de circuito normalmente utilizadas para aumentar a confiabilidade do projeto. Essas técnicas impactam o eixo de projeto, assim como o eixo de tecnologia, respectivamente nesta ordem. Com relação aos fatores de maior risco determinados em [7], pode-se aplicar algumas boas práticas de projeto que aumentam a confiabilidade dos circuitos. São elas:

(1) Derating de Componentes

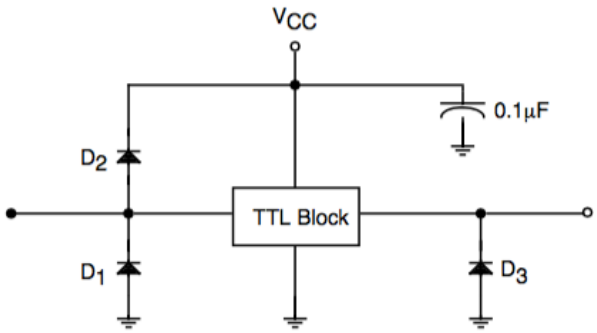
Deve-se levar em consideração que os componentes estarão expostos a condições ambientais que podem fazer com que seus parâmetros mudem significativamente ao longo do tempo. O próprio

envelhecimento natural de um componente operando em condições nominais podem resultar em comportamentos não planejados. Para mitigar essa fonte de falha é necessário dimensionar componentes com menor variação de parâmetro e efeitos de *Derating* possíveis, e que preferencialmente possam operar nas condições do equipamento proposto sem risco. A confiabilidade do circuito aumentará à medida em que a funcionalidade do circuito for menos dependente de parâmetros sujeitos a variações.

(2) Transientes e proteção contra surtos

Praticamente todo componente eletroeletrônico está sujeito a falha quando exposto a transientes não previstos em suas especificações, assim como surtos diversos na alimentação. Sendo assim, é necessário prever mecanismos de circuito para proteção de blocos importantes. Elementos nocivos para a vida útil dos componentes como, por exemplo, a eletricidade estática, devem ser evitados na etapa de fabricação do equipamento e em sua operação.

Figura 3.5.1 - Diodos de Proteção TTL Block

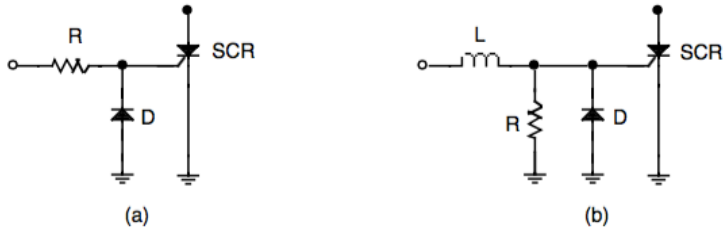


FONTE: [7]

Este fato torna-se ainda mais relevante em semicondutores com alta densidade de integração. Dispositivos com tecnologia MOS/CMOS têm, por exemplo, grande vulnerabilidade a eletricidade estática, baixa capacidade de dissipação de potência e baixas tensões de alimentação. Por essa razão, embora considerada uma tecnologia mais rudimentar, aconselha-se a utilização de circuitos lógicos com tecnologia TTL.É uma boa pratica para confiabilidade utilizar diodos de proteção contra surtos

também para circuitos TTL, conforme Figura 3.5.1, sendo este tipo de tecnologia mais confiável tendo como parâmetro os dados fornecidos em [5].

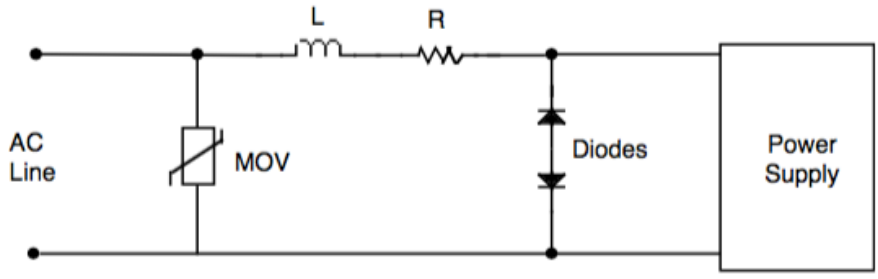
Figura 3.5.2 - Diodos de Proteção para gate de SCR



FONTE: [7]

Tiristores, DIACS e TRIACS também estão sujeitos a mal funcionamento devido a transientes, ruídos e surtos. Seus terminais de *gate* têm grande sensibilidade, podendo ser acionados de modo não intencional, além de poder sofrer danos irreparáveis. Considerando aplicações críticas, essa falha pode causar grande transtorno, já que este tipo de dispositivo normalmente é utilizado para acionar cargas com grande potência. A Figura 3.5.2 indica também uma maneira de reduzir os riscos de falha através de diodos de proteção.

Figura 3.5.3 - Diodos de proteção para alimentação



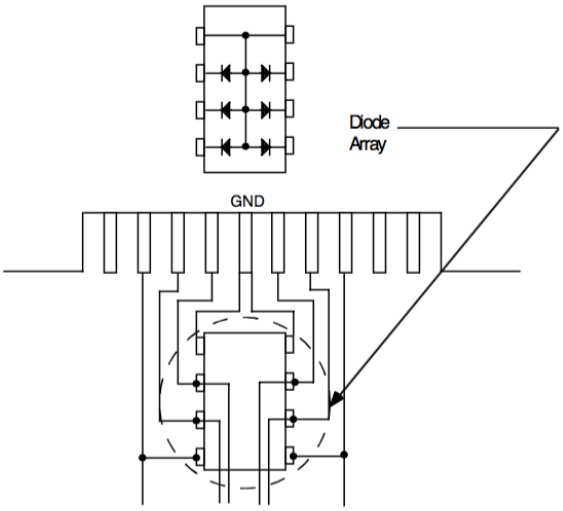
FONTE: [7]

Em um sistema eletrônico, espera-se que os componentes estejam protegidos contra falhas na alimentação pelo subsistema responsável por regular a tensão de alimentação e gerenciar a

potência distribuída. Proteger o restante do sistema dos surtos provenientes da rede de alimentação requer, porém, que o subsistema de alimentação esteja devidamente protegido.

São boas práticas utilizar diodos de proteção, conforme Figura 3.5.3, juntamente com dispositivos MOV (varistores) para supressão de picos de tensão e rede RL para proteção contra grandes variações de corrente. A utilização de fusíveis também deve ser considerada mandatória para que haja confiabilidade no subsistema de alimentação. Sabe-se que há grande oferta de fontes de alimentação industriais com alta confiabilidade disponíveis no mercado. Sendo assim, tratando-se da criticidade deste subsistema, é preciso dimensionar corretamente uma fonte com confiabilidade comprovada para utilização em aplicações específicas.

Figura 3.5.4 - Diode Array



FONTE: [7]

Diode Arrays, Figura 3.5.4, têm ampla distribuição no mercado e são utilizados em funções de proteção, como apresentado nesta Seção.

(3) Aterramento e alimentação isolada

Embora seja comum a prática de conectar a malha de terra de vários subsistemas para manter os níveis lógicos com referências iguais, nem sempre isso implica em maior confiabilidade. Na verdade, tendo como premissa que um dos grandes problemas enfrentados em sistemas embarcados

é a estabilidade na alimentação e má interpretação de *bits* devido a ruídos, conectar as malhas de terra de um número grande de subsistemas reduz a confiabilidade.

Subsistemas com alimentações com aterramento único contém no aterramento um grande ponto de falha único, reduzindo assim a capacidade de tolerância a falha, já que a falha na alimentação implicará em falha em todo o sistema. Trilhas compridas em placas de circuito impressas (PCB) , conectores, ou cabos extensos, irão introduzir impedâncias entre as referências de terra de cada um dos subsistemas, tornando o sistema muito mais sensível a ruídos, transientes e surtos. [7][14][15]

A solução comumente utilizada é que cada módulo, ou subsistema, de um equipamento de alta confiabilidade tenha sua fonte de alimentação isolada. Esta medida aumenta a quantidade de componentes, e requer a utilização de fontes mais complexas, mas ajuda significativamente a aumentar a confiabilidade do sistema.

Com fontes de alimentação distribuídas e independentes o sistema torna-se tolerante a este tipo de falha. Por exemplo, em caso de curto na fonte de um subsistema, a redundância deste subsistema, tendo ela uma alimentação isolada, continuará funcionando perfeitamente, mantém o equipamento disponível.

Esta decisão de projeto, altamente aconselhável, implica em adicionar complexidade em termos de comunicação entre subsistemas, porém há vários dispositivos normalmente utilizados para tal. Para sinais digitais há grande disponibilidade de optacopladores, optaisoladores, interface digitais para duas alimentações isoladas. Para sinais analógicos, diferenciais e de alta frequência, optaisoladores analógicos e transformadores de pulso são usados normalmente.

(4) Comunicação digital

Comunicação entre módulos ou subsistemas em geral á candidata a ponto de falha. Para minimizar ocorrência de falha na propagação de sinais digitais deve-se optar por protocolos de comunicação mais robustos, em geral com análise de consistência de dados e/ou que trabalhem com sinais diferenciais.

No caso de sinais digitais de baixa tensão paralelos como alternativa a métodos mais complexos, sugere-se que sempre seja transmitido o sinal nível lógico correto e com o inverso do nível lógico. Esta técnica permite que seja avaliado se o sinal digital esta apresentando um nível lógico estático quando não deveria, além de permitir diagnosticar falha da interface de transmissão separadamente do componente que geral o nível lógico. Por exemplo, se um determinado sinal que

deveria ter nível lógico baixo apresentar um comportamento semelhante a um sinal de *clock* enquanto o segundo conjugado mantém nível lógico alto, pode-se supor que há uma falha no conector pelo qual este sinal passa.

É uma boa prática blindar sinais fracos, seja na placa de circuito impresso quanto em conectores para que eles não tenham seus níveis lógicos contaminados por sinais de potência ou ruídos.

3.6. A sistemática de síntese para sistemas de alta confiabilidade

Tendo apresentado os passos e as análises propostas para projeto de equipamento com alta confiabilidade, pode-se determinar de forma objetiva a sistemática de síntese para sistemas eletrônicos genéricos. Segue abaixo os passos propostos, que serão aplicados.

1. Definição dos requisitos funcionais e não funcionais
   (Requisitos do sistema)
2. Definição e cálculo da especificação de confiabilidade
   (Requisitos do sistema)
3. Elaboração de hipóteses de arquitetura segundo princípios de confiabilidade para sistemas eletrônicos
   (Eixo de arquitetura)
4. Aplicar método proposto de avaliação de arquitetura
   (Eixo de arquitetura)
5. Realizar o PI-FMEA da arquitetura escolhida
   (Eixo de arquitetura)
6. Determinar blocos de hardware na arquitetura necessários através das conclusões do PI-FMEA
   (Eixo de projeto)
7. Realizar análise de confiabilidade via *Part Count Reliability Prediction*
   (Eixo de projeto)
8. Detalhar projeto de hardware propondo técnicas de aumento de confiabilidade segundo o resultado do PCRP.
   (Eixo de projeto)
9. Detalhar o projeto de hardware escolhendo componentes que apresentem menores taxas de falhas
   (Eixo de tecnologia)

10. Realizar análise de confiabilidade completa utilizando *Part Stress Analysis Prediction* e/ou parâmetros específicos de cada componente.
    (Eixo de tecnologia)

O CONTROLADOR DE TRÁFEGO COM ANÁLISE DE CONFIABILIDADE

4.1. Especificação de confiabilidade do controlador semafórico

Diante do exposto, é necessário que o controlador semafórico cujo o desenvolvimento é proposto tenha sua confiabilidade especificada, como determinado na Seção 3.2. Para aferir este parâmetro será seguida a diretriz e os elementos de especificação, já apresentados na Seção 3.2.

Elementos (2) e (3):

O ambiente no qual o controlador deverá operar inclui também outros equipamentos com os quais ele deve interagir. Procura-se projetar o controlador em conformidade com os atuais requisitos de carga, cujas características técnicas estão descritas na norma ABNT NBR 15889:2010. Segundo a normativa, o foco semafórico a LED deve ser um módulo eletrônico único de modo que seu funcionamento seja equivalente a uma lâmpada incandescente, podendo a coloração sendo imposta pela cor do LED ou pela cor da lente em conformidade com ASTM G153 ou ASTM G 155 com ciclo de 2000h.

Como 2000h seria uma vida útil inferior a dos LEDs, entende-se que esta deve ser o MTBF mínimo do controlador. Isto é justificado pois, caso o foco semafórico (conjunto composto de placa de LED, fonte, *case* e lente) apresente alguma falha, obrigatoriamente deverá ser feita alguma intervenção de manutenção no cruzamento. Esta intervenção implica necessariamente em desligar a funcionalidade principal do controlador por algum tempo, que é acionar os focos semafóricos de modo correto. Logo o MTBF do foco semafórico será o mínimo MTBF para um controlador semafórico considerado a prova de falhas. Entende-se também que todas as condições ambientais previstas para o foco semafórico devem ser suportadas também pelo controlador semafórico, visto que ambos os equipamentos operam em condições semelhantes. Logo o controlador deve atender as exigências equivalentes ao foco semafórico descritas abaixo.

- Alimentação: 127 Vca ± 25,4 Vca ou 220 Vca ± 44 Vca , 60 Hz ± 3 Hz
- Potência: menor ou igual 15 W, fp=0,92
- Resistência de isolamento: 2 MΩ
- Condições ambientais: -10°C a 60°C e umidade relativa de até 95%
- Teste 1: funcionar 24hrs a 60°C
- Teste 2: -10 e 60°C com ciclos de 30 min .
- Teste 3: suportar tensões de 2500V, 60Hz durante 1 min.

A partir dos dados da norma é possível detalhar as grandezas elétricas pertinentes ao circuito de acionamento dos focos semafóricos. Considerando o circuito de acionamento de cada foco, pode-se calcular o seu real tempo de utilização de cada foco baseado no ciclo de acionamento previsto em um plano de programação semafórico. Os dados auferidos estão presentes na Tabela 4.1.1.

Tabela 4.1.1 - Características Elétricas e Horas de Operação

Analise Requisitos dos Focos Semaforicos		
Potencia Ativa (W)	15	
Potencia Aparente (VA)	16,30435	
Fator de Potencia	0,92	
Tensao Nominal (Vac)	127	220
Tensao Maxima	152,4	264
Tensao Minima	101,6	176
Corrente Operacao Maxima (mA)	106,9839	61,75889
Corrente Operacao Minima (mA)	160,4759	92,63834
Corrente Operacao Nominal (A)	0,128381	0,074111
Impedancia Conduçao (Ω)	989,2453	2968,533
Resistencia de Isolamento (MΩ)	2	2
Tensao de Ensaio (Vca)	2500	2500
Tempo de Reaçao de Proteçao	1 min	1 min
Ciclo Ensaio Minimo (s)	0,5	0,5
Ciclo Ensaio Maximo (horas)	24	24
Ciclo Medio Operacional (min)	10	10
Ciclos /dia	72	
Horas / dia	12	12
Ciclos/ano	26280	
Horas/ano	4380	

Requisitos de Confiabilidade Controlador			
	HORAS	Ciclos	Fator Piscante
Vida Util Driver 5 anos (hr)	21900	157680	1,2
Vida Util Driver 10 anos (hr	43800	315360	
Vida Util CPU 5 anos (hr)	43800	315360	
Vida Util CPU 5 anos (hr)	87600	630720	
Vida Util CPU 2 anos (hr)	17520		
Corrente 3 GS Max	320,95173		
Tensao (Vac)	152,4		
Potencia Ativa Driver (W)	45		

Os dados foram utilizados posteriormente para projetar de forma devida o acionamento dos focos a LED feito pelo controlador. O projeto do circuito de acionamento parte da análise dos dados referentes ao regime permanente para cálculo da potência a ser dissipada, considerando o maior tempo que um foco requer potência nominal. Analisar os problemas decorrentes do transitório também é muito importante, por isso a taxa de chaveamento foi estipulada considerando que o menor

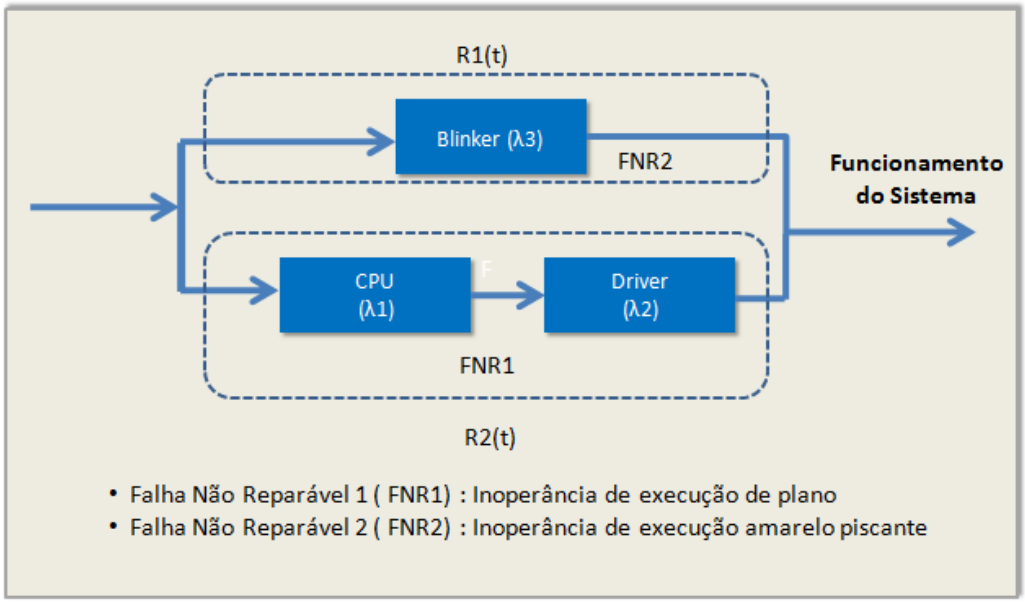
ciclo possível é o do modo piscante, em que o foco ficaria piscante a uma taxa de 0,5 segundos. Foi considerado o ciclo médio operacional para calculo da quantidade de ciclos por dia, acrescido de um multiplicador chamado fator piscante que representa a porcentagem dos ciclos que foram ciclos mínimos. Estima-se razoável acrescentar 20% na quantidade de ciclos para representar a quantidade de ciclos em modo piscante durante períodos sem falhas, correspondente por exemplo a modo piscante noturno previsto no plano semafórico. Estes dados de vida útil baseados na vida útil dos focos de LED representam a mínima vida útil do controlador.

Elementos (1) e (4):

Para determinação da especificação de confiabilidade será adotada uma hipótese de arquitetura na qual o sistema será composto de três subsistemas. Os subsistemas considerados inicialmente serão a CPU, Driver e Blinker. A CPU responsável pelo processamento, o Driver responsável pelo acionamento em condição de pleno funcionamento da CPU, e o Blinker que será responsável pelo acionamento de amarelo piscante em caso de falha da CPU ou do Driver. A Tabela 4.1.2 apresenta a vida útil prevista em horas esperadas dos blocos CPU e Driver conectados em série.

Entende-se por vida útil o intervalo de tempo de operação no qual um equipamento novo terá probabilidade insignificante de apresentar alguma falha. Como o tempo efetivo de utilização de cada subsistema é diferente, assim como o nível de importância de cada um, cada bloco terá um requisito de confiabilidade diferente. Foi decidido através dos dados calculados que o módulo CPU deve funcionar por no mínimo 2 anos sem ocorrência de falhas e o modulo Blinker deve operar por 5 anos sem ocorrência de falha relevante. Esta vida útil sem falhas é maior do que o mínimo 2000 hrs aferido e se adequar melhor à premissa do trabalho de desenvolver sistemas de alta confiabilidade, assumindo-se o risco desta escolha.

Figura 4.1.1 - Diagrama de Confiabilidade

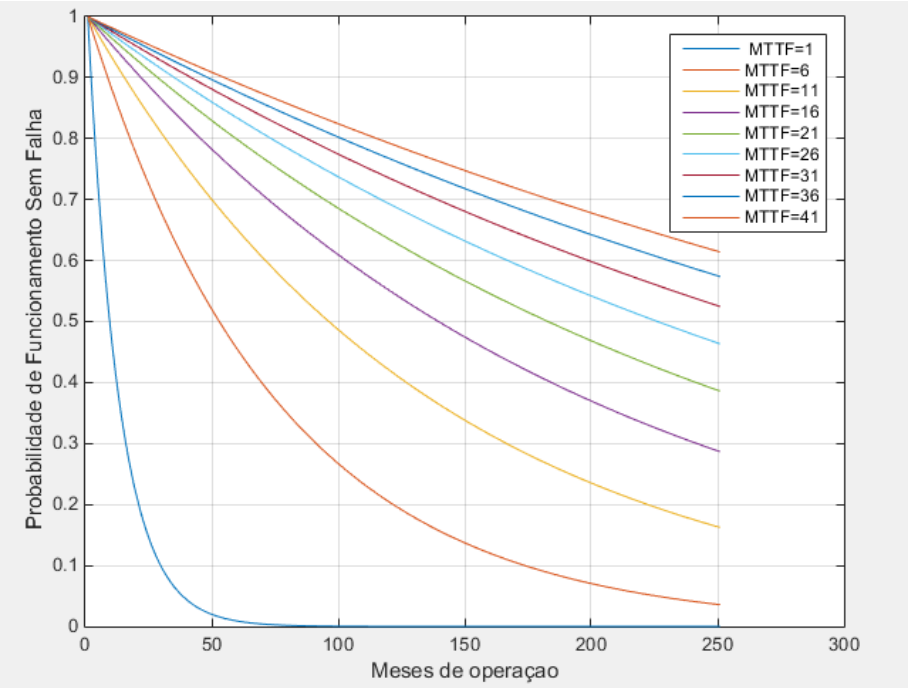


FONTE: Produção do próprio autor

A Figura 4.1.1 ilustra as duas falhas cuja probabilidade de ocorrência se deseja diminuir durante o projeto. A primeira falha não reparável, FNR1, seria a inoperância do conjunto CPU e DRIVER. Falha não reparável é uma falha que não pode ser revertida por tratamento de hardware ou software. Desta forma um controlador que apresente esta falha esta impossibilitado de executar um plano semafórico corretamente e deve, portanto, entrar em modo piscante (função exercida pelo Blinker). A segunda falha não reparável, ou FNR2, trata-se da deficiência do sistema em operar em modo piscante. A ocorrência da FNR2 marca o final da vida útil do equipamento, pois sem a opção de operar em modo piscante o equipamento torna-se inutilizável segundo os padrões de alta confiabilidade propostos neste trabalho.

É sabido que existe uma relação de compromisso entre confiabilidade, manutenabilidade e disponibilidade, e neste trabalho a confiabilidade será priorizada. Deste modo a disponibilidade requerida do equipamento deverá ser atingida através da confiabilidade, não sendo permitido, por exemplo, que um subsistema possua fácil substituição, mas apresente curta vida útil.

Figura 4.1.2 - Curvas de Confiabilidade para MTTF variados



FONTE: Produção do próprio autor

A Figura 4.1.2, assim como a Tabela 4.1.2, foram analisadas para que fosse possível definir as taxas de falhas (λ) associadas a cada um dos blocos do sistema. A taxa de falhas (*Failure Rate*), é o principal parâmetro a ser especificado para o sistema, pois ele será parâmetro da função confiabilidade, Seção 3.2 . Outro importante parâmetro é o inverso da taxa de falhas, o tempo médio até uma falha, MTTF (*Mean-Time- To-Failure*).

Tabela 4.1.2 - Taxa de Falha

Mean Time to Failure	Constant Failure Rate		Probabilidade de Funcionamento Sem Falha		
MTTF (anos)	λ (falhas/ano)	λ (falhas/10^6 hrs)	R(1 anos)	R(2 anos)	R(5 anos)
1	1	0,00876	37%	14%	1%
6	0,166666667	0,00146	85%	72%	43%
11	0,090909091	0,000796364	91%	83%	63%
16	0,0625	0,0005475	94%	88%	73%
21	0,047619048	0,000417143	95%	91%	79%
26	0,038461538	0,000336923	96%	93%	83%
31	0,032258065	0,000282581	97%	94%	85%
36	0,027777778	0,000243333	97%	95%	87%
41	0,024390244	0,000213659	98%	95%	89%
46	0,02173913	0,000190435	98%	96%	90%
51	0,019607843	0,000171765	98%	96%	91%
56	0,017857143	0,000156429	98%	96%	91%
61	0,016393443	0,000143607	98%	97%	92%
66	0,015151515	0,000132727	98%	97%	93%
71	0,014084507	0,00012338	99%	97%	93%
76	0,013157895	0,000115263	99%	97%	94%
81	0,012345679	0,000108148	99%	98%	94%
86	0,011627907	0,00010186	99%	98%	94%
91	0,010989011	9.63E+00	99%	98%	95%

Considerando a distribuição de probabilidade de falha exponencial um bom modelo para representar equipamentos eletrônicos exposta no manual MIL-HDBK-338B [5], temos que a função de confiabilidade R(t) pode ser descrita pela Equação 4.1.1.

$$R(t) = e^{-\lambda t} = e^{-\frac{t}{MTTF}}$$

Equação 4.1.1 - Função de Confiabilidade

Para garantir a confiabilidade, foram analisadas a probabilidade de funcionamento sem falha considerando diferentes MTTF. Desta forma é possível comparar as curvas de confiabilidade possíveis de serem atingidas para cada λ . Esta análise tem como objetivo determinar a taxa de falha máxima de cada bloco. A taxa de falha de cada bloco determinará o MTTF do controlador semafórico, que é o elemento quantitativo final para descrever a especificação de confiabilidade.

É assumido o risco considerando adequada uma expectativa de vida útil de 5 anos para o módulo Blinker e 2 anos para o modulo CPU em cascata com o módulo DRIVER. Desta forma pode-se definir taxas de falhas a serem especificadas para cada bloco. As confiabilidades da CPU e o DRIVER serão inicialmente alocadas de forma igual. Será considerado que cada bloco tenha uma probabilidade de 95% de não apresentar nenhuma falha dentro da vida útil. Desta forma temos:

$$\lambda_{FNR1} = \lambda_{CPU} + \lambda_{DRIVER}$$
$$\lambda_{FNR1} = 0{,}012195 + 0{,}012195 = 0{,}024\ falhas/ano$$

$$\lambda_{FNR2} = 0{,}010\ falhas/ano$$
$$P_{FNR1}(0 < t < 2\ anos) = 5\% \quad P_{FNR2}(0 < t < 5\ anos) = 5\%$$

Conclui-se a especificação de confiabilidade:

$$P_{FNR1}(0 < t < 2\ anos) = 5\% \quad P_{FNR2}(0 < t < 5\ anos) = 5\%$$

A probabilidade de falhas de um equipamento com as taxas de falha tal como proposto é considerada a confiabilidade intrínseca do equipamento considerando funcionamento dentro das condições nominais de operação especificadas.

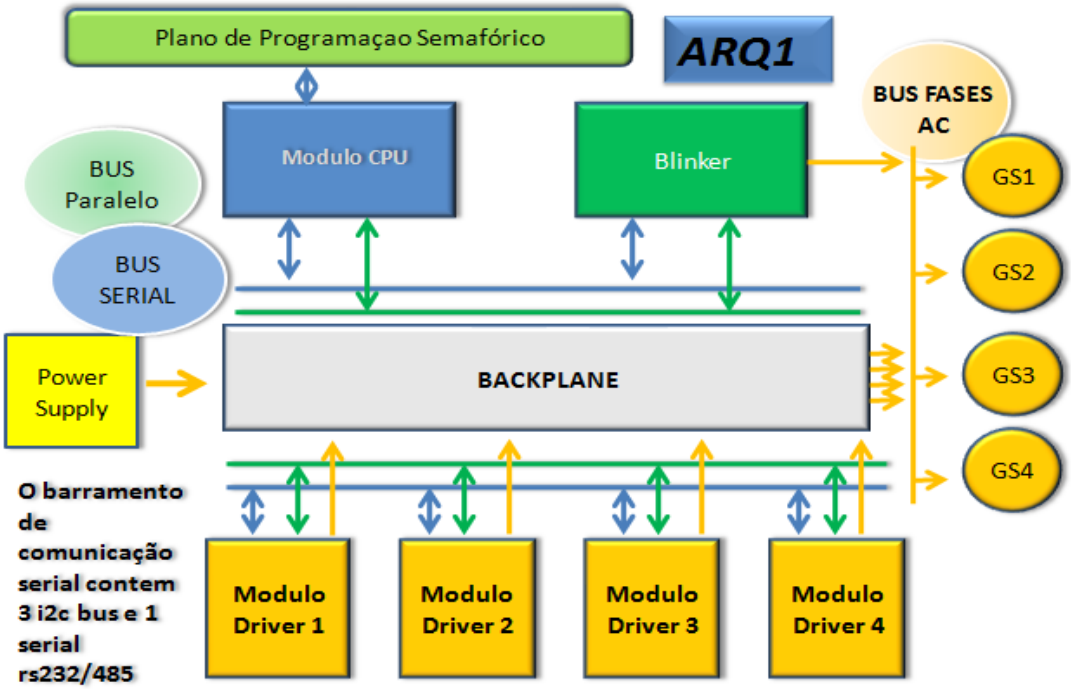
Os requisitos funcionais e as condições de nominais de operação já apresentados, devem ser analisadas em conjunto no decorrer do projeto para que a confiabilidade seja atingida em cada subsistema conforme a necessidade e possibilidade.

4.2. Princípios de confiabilidade e hipóteses de arquitetura do controlador

Considerando as especificações já propostas para o controlador semafórico, as especificações de confiabilidade já apresentadas e os princípios para projeto eletrônico de alta confiabilidade elencados na Seção 3.3.1, foram idealizadas 3 arquiteturas distintas para análise.

Arquitetura 1

Figura 4.2.1 - Diagrama da Arquitetura 1



FONTE: Produção própria do autor

Nesta arquitetura há 6 módulos, Figura 4.2.1, conectados com uma placa mãe chamada de módulo *backplane* responsável pela integração entre módulos. A CPU aciona os 4 módulos *DRIVER* através de um barramento I2C e monitora o funcionamento dos módulos através de um segundo barramento I2C isolado. A CPU também poderia se comunicar com um microcontrolador local do módulo DRIVER, via RS232/485 para diagnostico de falha detalhada e/ou redundância de acionamento e monitoramento. O circuito piscante, também chamado de BLINKER, irá gerar um sinal digital piscante para todos os módulos e também uma fase (127Vca) piscante para que seja chaveada direto para todas os focos semafóricos vinculados ao controlador em caso de falha geral no sistema.

Tabela 4.2.1- Descrição dos módulos Arquitetura 1

ARQ1: Arquitetura com Processamento Distribuido e Isolado		
Modulo CPU	PCB contendo microcontrolador responsavel por todo o processamento, desde a comunicacao, acesso a perifericos, acionamento e tratamento de falhas. Entre perifericos estao incluidos, porta ethernet, gps, cartao sd, barramentos serial e i2c para acionamento e tratamento de falha	Qtd. 1
Modulo Driver	PCB contendo Ics de interface de comunicacao com a CPU. Deve realziar tratamento de falhas em hardware com componentes discretos e/ou com microcontrolador local . Cada modulo deve acionar um grupo semaforico, ou seja, deve acionar as 3 fases de cada semaforo .	Qtd. 4
Modulo Backplane	PCB mae responsavel pela conexao entre as demais placa. Possui os conectores necessarios e os barramentos de comunicacao, alimentacao e demais sianis de controle necessarios para que a CPU e os microcontroladores/modulos Driver se comuniquem	Qtd. 1
Modulo Piscante(Blinker)	PCB contendo circuito oscilador de baixa frequencia capaz de criar uma fase piscante de alta confiabilidade a ser chaveadas para os focos semaforicos em caso de emergencia.	Qtd. 1

A ideia central desta arquitetura, apresentada na Tabela 4.2.1 e na Figura 4.2.1, está em manter cada módulo completamente isolado eletricamente, implicando em uma maior independência entre módulos. O objetivo é que uma falha em um módulo não inviabilize o funcionamento parcial do sistema.

Cada módulo DRIVER também teria seu próprio microcontrolador criando um grau de isolamento lógico entre os módulos. Deste modo, é possível que este microcontrolador local assuma o controle do acionamento do módulo driver em caso de falha da CPU.

O microcontrolador local poderia aprender a atual sequência de estágios vinculadas ao grupo semafórico que ele gerencia e assim manter a mesma duração de estagio em situação de falha da CPU.

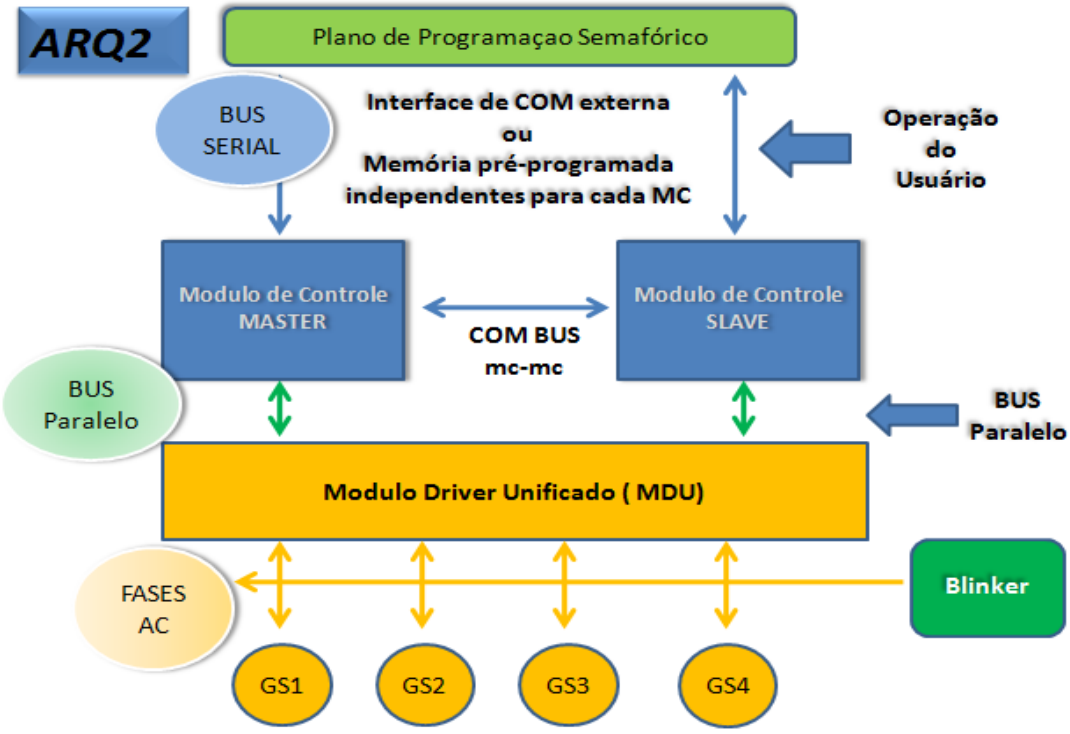
Qualquer um dos 4 microcontroladores locais poderia também assumir controle dos barramentos I2C de acionamento e leitura disponíveis no *backplane* para agir integralmente como CPU no que diz respeito ao tratamento de falhas, principalmente o monitoramento de verde conflitante. O desafio de desenvolvimento é gerenciar o controle dos barramentos I2C para que na falha da CPU e/ou gradual falha dos microcontroladores dos DRIVERs, um dos microcontroladores disponíveis assuma o controle.

A CPU também deve ter uma interface de comunicação serial segura com todos os demais microcontroladores para que ela envie a tabela de verdes conflitantes, permitindo assim a detecção de conflito de verde possa ser feita, também, pelos demais microcontroladores.

Sinais paralelos estariam presentes no backplane e devem estar disponíveis por todos os módulos de forma a indicar a queda ou reconexão de módulos críticos. Este fato diminui também a confiabilidade.

Arquitetura 2

Figura 4.2.2 – Diagrama da Arquitetura 2



FONTE: Produção própria do autor

Nesta arquitetura, Figura 4.2.2, todo o processamento, da comunicação externa ao sistema, controle de periféricos, pré-acionamento, à detecção de falhas é realizado em um mesmo módulo chamado aqui módulo de controle (MC). Esta PCB possui todo o hardware necessário para o acionamento de 4 grupos semafóricos completos, como exceção de recursos para acionamento de potência. Os sinais de saída destes módulos deverão ser os sinais de saída de optoisoladores ou semelhantes.

Desta forma espera-se que dois MC completamente autossuficientes criem redundância de acionamento em uma única PCB chamada Módulo Driver Unificado ou MDU, Tabela 4.2.2,

responsável pelo acionamento de potência de todos os 4 grupos semafóricos. O MDU deve apresentar um banco de TRIACs ou semelhantes para chaveamento das fases dos 3 focos de 4 grupos semafóricos assim como toda a proteção elétrica e leitura de sinais pertinentes para detecção e tratamento de falha pelo MC.

Tabela 4.2.2- Descrição de módulos da Arquitetura 2

ARQ2: Arquitetura de Controle Centralizado com redundancia		
Modulo de Controle (MC)	PCB contendo microcontrolador responsavel pela comunicaçao externa ao sistema, controle de periféricos, pre-acionamento, à detecçao de falhas . Esta PCB possui todo o hardware necessario para o acionamento de 4 grupos semaforicos completos, como exceçao da parte de potencia propriamente dita, ou os sinais de saida deste modulos deverao ser os sinais de saida de optaisoladores ou semelhantes. Desta forma espera-se que dois MC completamente auto-suficientes atuem a criar redudancia	Qtd. 2
Modulo Driver Unificado (MDU)	PCB chamada Modulo Driver Unificado ou MDU, resposanvel pelo acionamento de potencia de todos os 4 grupos semaforicos . O MDU deve apresentar um banco de triac ou semelhantes para chaveamento das fases para cada um dos 3 focos de cada um dos 4 grupos semaforicos assim como toda a proteçao eletrica e leitura de sinais pertinentes para tratamento de falha no MC .	Qtd. 1
Modulo Piscante(Blinker)	PCB responsavel por gerar uma fase piscante de alta confiabildiade para ser chaveada somente em caso de falha geral, ou seja, modo piscante de focos isolados sera gerado pelo MC porem piscante de hardfault sera gerado pelo Blinker de forma a criar um grande grau de isolaçao eletrica e logica entre os dois modulos.	Qtd. 1

A ideia é isolar a parte de potência da parte lógica, sem isolar cada grupo semafórico. diferentemente da Arquitetura 1. Esta escolha deve acarretar em custo muito mais baixo do módulo de potência. Seria possível que o MDU possuísse alguns componentes lógicos que permitisse armazenar algum dado como um *serial number* com a finalidade de realizar um diagnóstico de falha vinculado ao número de registro do módulo, porém tal funcionalidade não interfere no nível de confiabilidade do projeto.

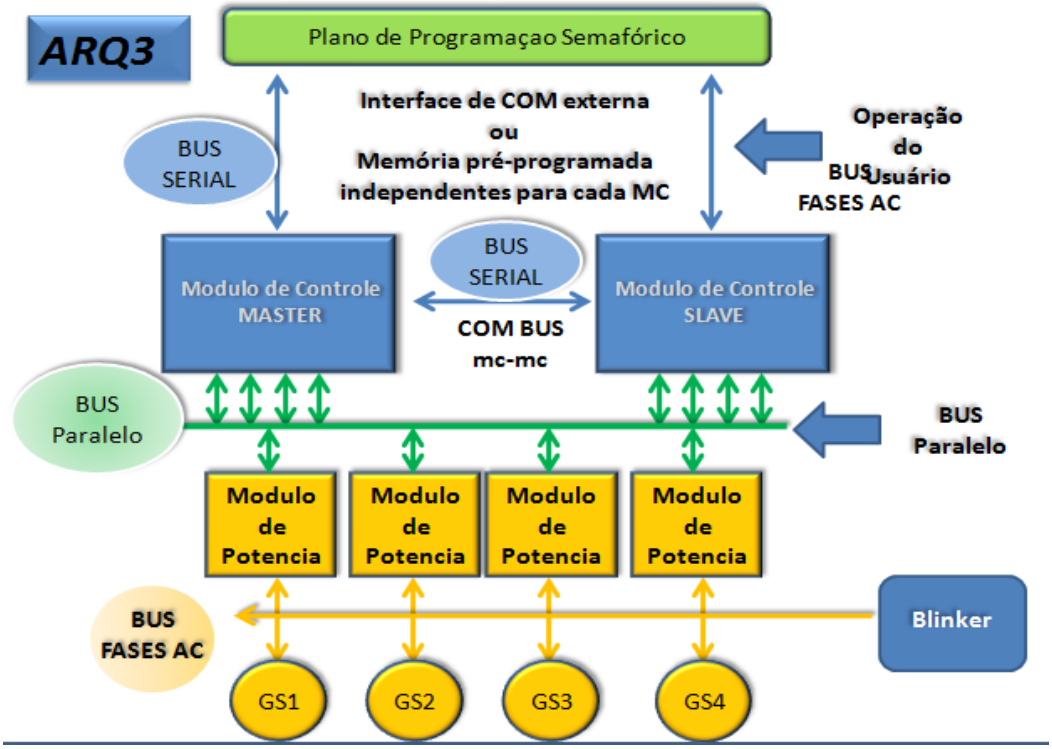
O módulo piscante ou BLINKER será responsável por gerar uma fase piscante de alta confiabilidade para ser chaveada somente em caso de falha geral. O piscante de focos isolados será gerado pelo MC, sendo que em caso de falha, o modo piscante será gerado pelo BLINKER. Espera-se assim, criar um grande grau de isolação elétrica e lógica entre os dois módulos.

Nesta arquitetura também foi abolida a existência de backplane. Assim haverá a necessidade de conectores PCB-Cabo entre os dois MC e de ambos MC para o módulo DRIVER. Cada MC deve ser gravado com um plano de programação semafórico. O operador do sistema deve gravar um MC primeiro atribuindo um grau de prioridade maior ao primeiro e posteriormente gravar o segundo com grau de prioridade menor. O MC com maior prioridade, ou MC MASTER irá então conferir a programação no MC com o qual esta conectado, para validar os dados de tratamento de falha do

outro MC. Logo não será possível usar o sistema com dois MC inicialmente configurado com mesma prioridade.

Arquitetura 3

Figura 4.2.3 - Diagrama da arquitetura 3



FONTE: Produção própria do autor

Na arquitetura 3, Figura 4.2.3, todo o processamento, da comunicação externa ao sistema, controle de periféricos, pré-acionamento, à detecção de falhas é realizado em um mesmo módulo chamada aqui de Modulo de Controle (MC), ou simplesmente CPU. Esta PCB possui todo o hardware necessário para o acionamento de 4 grupos semafóricos completos, como exceção do acionamento de potência. O MC, ou CPU, envia sinais digitais de controle para o módulo de potência (também chamado de DRIVER).

Desta forma espera-se que dois MC completamente autossuficientes atuem, havendo assim uma redundância de processamento e acionamento de cada módulo DRIVER, responsável pelo acionamento de potência de cada grupo semafórico.

O MP deve apresentar um banco de TRIACs ou semelhantes para chaveamento das fases para cada um dos 3 focos de cada um grupo semafórico assim como toda a proteção elétrica e leitura de sinais pertinentes para tratamento de falha no MC. Deste modo aumenta o número de cabos e conectores em relação a ARQ2 pois cada MC deve ter 4 conectores, 1 para acionar cada um dos 4 módulos de potência. A ideia é isolar a parte de potência da parte logica, sem isolar cada grupo semafórico, como no caso analisado na primeira arquitetura. Esta escolha deve acarretar em custo muito mais baixo do módulo de potência.

Tabela 4.2.3 – Descrição de módulo Arquitetura 3

ARQ3: Arquitetura de Controle Centralizado e Acionamento Distribuido		
Modulo de Controle (MC)	PCB contendo microcontrolador responsavel pela comunicaçao externa ao sistema, controle de periféricos, pre-acionamento, à detecçao de falhas . Esta PCB possui todo o hardware necessario para o acionamento de 4 grupos semaforicos completos, como exceçao da parte de potencia propriamente dita, ou os sinais de saida deste modulos deverao ser os sinais de saida de optaisoladores ou semelhantes. Desta forma espera-se que dois MC completamente auto-suficientes atuem a criar redudancia	Qtd. 2
Modulo de Potencia (MP)	PCB chamada MP, responsavel pelo acionamento de potencia de todos os 1 grupos semaforicos . O MP deve apresentar um banco de triac ou semelhantes para chaveamento das fases para cada um dos 3 focos com toda a proteçao eletrica e leitura de sinais pertinentes para tratamento de falha no MC .	Qtd. 4
Modulo Piscante(Blinker)	PCB responsavel por gerar uma fase piscante de alta confiabildiade para ser chaveada somente em caso de falha geral, ou seja, modo piscante de focos isolados sera gerado pelo MC porem piscante de hardfault sera gerado pelo Blinker de forma a criar um grande grau de isolaçao eletrica e logica entre os dois modulos.	Qtd. 1

Seria possível que o MP possuísse alguns componentes lógicos que permitisse armazenar alguns dados como um *serial number* com a finalidade de realizar um diagnostico de falha vinculado ao numero de registro do modulo, porém tal funcionalidade não interfere no nível de confiabilidade do projeto. O Módulo piscante ou Blinker será responsável por gerar uma fase piscante de alta confiabilidade para ser chaveada somente em caso de falha geral, ou seja, modo piscante de focos isolados será gerado pelo MC porem piscante de *hardfault* será gerado pelo Blinker de forma a criar um grande grau de isolação elétrica e logica entre os dois módulos.

Nesta arquitetura também foi abolida a existência de *backplane*, porém desta forma há necessidade de conectores PCB-CABLE entre os dois MC e de ambos MC. Cada MC deve ser

gravado com um plano de programação semafórico. O operador do sistema deve gravar um MC primeiro atribuindo um grau de prioridade maior ao primeiro e posteriormente gravar o segundo com grau de prioridade menor. O MC com maior prioridade, ou MC MASTER irá então conferir a programação no MC com o qual esta conectado para validar os dados de tratamento de falha do outro MC. Logo não será possível usar o sistema com dois MC inicialmente configurado com mesma prioridade.

4.3. Análise de confiabilidade qualitativa

Com base nos princípios de projetos eletrônicos expostos na Seção 3.3.1 foram listadas as vantagens e desvantagens de cada hipótese de arquitetura. Os critérios usados foram prioritariamente os princípios de confiabilidade, porém também foram consideradas como vantagens características que apresentariam melhor desempenho funcional/operacional.

Tabela 4.3.1 - Tabela de vantagens e desvantagens

ARQ.1	**Vantagens**	**Desvantagens**
	Grande modularidade e isolamento elétrico entre módulos	Grande quantidade de componentes
	Permite grande grau de redundancia possibilitando funcionamento parcial	Grande quantidades de conectores PCB-PCB
	Comunicaçao serial I2C para acionamento e leitura de falhas	Necessidade de sinais paralelos para indicar falhas de modulos no backplane
	Diagnostico de falha mais detalhado	Ponto de falha critico pela necessidade de backplane
	Facilidade de expansão	Custo mais alto para produçao
ARQ.2	**Vantagens**	**Desvantagens**
	Reduçao do numero de compoentes	Diminuiçao do isolamento eletrico
	Reduçao do numero de conectores	Não possibilidade de funcionamento parcial
	Reduçao do numero de PCBs	Conectores PCB-CABLE com grande quantidade de sinais paralelos
	Simplifacao do projeto eletronico	Pequena manutenabilidade com grande custo em caso de falha de MC
	Reduçao de custo de produçao	Baixo detalhamento de diagnostico
ARQ.3	**Vantagens**	**Desvantagens**
	Reduçao do numero de compoentes	Diminuiçao do isolamento eletrico
	Reduçao do numero de conectores	Não possibilidade de funcionamento parcial
	Reduçao do numero de PCBs	Conectores PCB-CABLE com grande quantidade de sinais paralelos
	Simplificaçao do projeto eletronico	Pequena manutenabilidade com grande custo em caso de falha de MC
	Reduçao de custo de produçao	Baixo detalhamento de diagnostico

Após a análise qualitativa de vantagens foi criado um padrão de avaliação quantitativo baseado nas premissas de aumento de confiabilidade. O método utilizado levou a um resultado convergente em relação a análise qualitativa. Desta forma é proposto um modo de avaliação

semelhante nos estágios iniciais de projeto quando ainda faltam elementos quantitativos e detalhamento para os métodos PCRP, PSCAP e FMEA.

Tabela 4.3.2 - Avaliação quantitativo dos princípios de confiabilidade

	PREMISSAS PARA AUMENTAR CONFIABILIDADE	ARQ1	ARQ2	ARQ 3	PESO
1	Reduzir numero de conectores	0	2	1	90%
2	Reduzir numero de componentes	0	2	1	
3	Aumentar isolamento eletrico entre modulos	2	0	0	
4	Melhorar protecao eletrica nos modulos	2	0	1	
5	Utilizacao de redundancias para funcionamento pleno	0	2	2	
6	Usar componentes de alta confiabilidade	2	2	1	
7	Reduzir o numero de sinais paralelos em conectores	2	0	1	
8	Reduzir o numero de interface de comunicaçao	0	2	1	
9	Tratamento de falhas em hardware	1	1	1	
10	Tratamento de falhas em software	0	0	0	
11	Utilizacao de redundancias para funcionamento parcial	2	0	1	

A pontuação proposta, por critério, vai de 0 a 2. Para indicar total inadequação do critério, ou inexistência do item avaliado utiliza-se a nota 0. A nota 1 indica a existência ou consideração do critério. A nota 2 permite criar diferenciação entre as arquiteturas, elencando as melhores ou a melhor em cada critério.

Tabela 4.3.3 - Avaliação quantitativa de critérios operacionais

	Demais pontos de avaliaçao	ARQ1	ARQ2	ARQ3	PESO
1	Potencial de Desenvolvimento	0	2	1	10%
2	Testabilidade	0	2	2	
3	Manutenabilidade	2	0	1	
4	Producao	0	2	1	
5	Custo Direto	0	2	1	

Como o objetivo principal deste trabalho é o projeto de equipamento com alta confiabilidade, mesmo que com operação, produção e manutenção dificultada, foram escolhidos pesos para o os critérios de confiabilidade e os demais pontos de avaliação. As notas de cada critério foram somadas, equalizadas e depois considerando pontos provenientes de critérios de confiabilidade ou operacionais, as notas parciais foram ponderadas para gerar uma nota final entre 0 e 1 para cada arquitetura. A avaliação de confiabilidade ficou com 90% da nota e a avaliação operacional 10 % da nota.

Tabela 4.3.4 –Avaliação final das hipóteses de arquitetura

Pts. Confiabilidade:	0.5	0.5	0.454545455	
Pts. Operacional:	0.2	0.8	0.6	
PONTOS TOTAIS	0.47	0.53	0.469090909	Range: 0-1

Todas as arquiteturas conseguiram boa notas de confiabilidade, em termos comparativos, em parte porque todas foram propostas a partir dos princípios de confiabilidade eletrônica. Entretanto, foi possível determinar qual das arquiteturas é mais indicada para confiabilidade. Tendo esta avaliação como premissa, é preciso agora detalhar os requisitos funcionais para mitigação das falhas. Toda arquitetura possui pontos de falha intrínsecos, que devem ser avaliados exercitando hipóteses de falha através de FMEA. Com a aplicação da FMEA ficam claros os pontos de falha e as ações de projeto que podem mitiga-las, permitindo então, que essas ações sejam aplicadas no detalhamento do projeto a nível de circuito.

A Arquitetura 2 com nota final de 0.53, em comparação com 0.47 e 0.46 das demais, prosseguirá para a síntese referente ao eixo de projeto. A nomenclatura que será utilizada para descrever esta arquitetura prevê três módulos, ou subsistemas. A CPU (anteriormente chamado de Módulo de Controle), o DRIVER (anteriormente chamado de módulo de potência) e o BLINKER. A arquitetura conceitualmente é definida, por, uma CPU MASTER e uma CPU SLAVE, acionando o DRIVER em modo redundante, sendo este também acionado pelo BLINKER.

4.4. *Failure Mode and Effects Analysis* da arquitetura escolhida

Foi realizada a PI-FMEA da Arquitetura 2, contida no Apêndice A. Neste processo foram analisados os três subsistemas distintos, a CPU, DRIVER e BLINKER, considerando todas as interconexões descritas na Seção 4.2. Foram levantadas possíveis falhas inerentes à arquitetura, seus diferentes modos de falha, efeitos esperados, causas possíveis e ações previstas pela arquitetura. Cada modo de falha gerou uma sugestão de projeto para mitigar a falha, sugestões que serão implementadas conforme a nota RPN de cada modo de falha.

Os modos de falha com maior RPN foram:

1. Falha de execução do software da CPU
2. Falha de execução do software da BLINKER
3. Plano de programação gravado inconsistente

4. Falha no conector CPU-DRIVER, BLINKER-CPU, CPU-CPU
5. Curto entre focos verdes
6. Trigger não intencional do DRIVER
7. Trigger não intencional dos circuitos lógicos de acionamento no DRIVER ou CPU
8. Curto nos circuitos lógicos de acionamento CPU ou DRIVER
9. Falha generalizada na PCB CPU, PCB DRIVER, PCB BLINKER
10. Perda de sinal de rede durante longos períodos
11. Falha de seleção entre unidade de controle lógica (VOTER) no DRIVER
12. Curto na saída da fase no DRIVER
13. Falha na alimentação CPU, BLINKER, DRIVER

De modo geral, os pontos de falha mais críticos foram resumidos entre falhas de execução de software, falhas generalizadas das placas de circuito impresso, falhas de conectores, falhas nas cargas, falhas de alimentação e acionamento não intencional.

Com base nas sugestões para mitigação de falha provenientes do PI-FMEA pode-se prever blocos pertinentes para cada subsistema. De forma resumida as medidas de mitigação escolhidas serão:

- Redundância de conector de comunicação entre a CPU MASTER e CPU SLAVE.
- Redundância em hardware para monitorar conflito de verde.
- Redundância em hardware para monitorar consistência do plano de programação semafórico.
- Redundância de memória não volátil para preservar consistência do plano de programação semafórico.
- Transmissão de sinais em pares conjugados, comunicados entres os subsistemas através do DRIVER visando indicar FHNR e FSR/FSNR (nomenclatura apresentada na seção 3.5).
- Utilização componentes de alta confiabilidade no VOTER (bloco de hardware que seleciona o subsistema irá assumir o controle lógico do acionamento do DRIVER).
- A CPU capacitada a ler as correntes de todos os focos semafóricos para verificar e corrigir acionamentos não intencionais.
- Fornecimento de opções variadas de conexão com a Internet para diminuir a probabilidade de não conectividade por períodos prolongados.
- Projeto de subsistema com fontes de alimentação e terras isolados para prevenir que falha generalizada da PCB ou curto/surto/falha na alimentação se propague entre subsistemas.
- Aplicação de redundância no acionamento de potência para mitigar falha de curto do dispositivo usado no acionamento.

As medidas de mitigação de falha tendem a aumentar a confiabilidade, permitindo que o projeto começasse a ser detalhado, descrevendo agora blocos de hardware dentro de cada subsistema.

O hardware foi projetado para que as ações previstas no PI-FMEA sejam passíveis de implementação no software, principalmente para que fosse possível a avaliação dos subsistemas elegendo sempre a CPU MASTER, ou a CPU SLAVE, ou BLINKER como unidade responsável pelo controle lógico do DRIVER.

Tabela 4.4.1 - Gerenciamento de unidade de controle lógico

CPU MASTER	CPU SLAVE	BLINKER	Unidade de Controle Lógico
Funcionamento normal	n/a	n/a	CPU MASTER
Qualquer falha	Funcionamento normal	n/a	CPU SLAVE
Sinais Inconsistentes	n/a	Funcionamento normal	BLINKER
n/a	Sinais Inconsistentes	Funcionamento normal	BLINKER
Qualquer falha	Qualquer falha	Funcionamento normal	BLINKER
Qualquer falha	Qualquer falha	Falha de Software	BLINKER
Falha de Software	Qualquer falha	Falha de Hardware	CPU MASTER
Falha de Software	Qualquer falha	Sinais inconsistentes	CPU MASTER
Falha de Hardware	Falha de Software	Falha de Hardware	CPU SLAVE
Falha de Hardware	Falha de Software	Sinais inconsistentes	CPU SLAVE

Como demonstrado na Tabela 14, foram consideradas os efeitos de uma FHNR/FHR ou FSR/FSNR. Estas falhas seriam identificadas por sinais digitais compartilhados entre os subsistemas através do DRIVER. São definidos então os sinais responsáveis por tal:

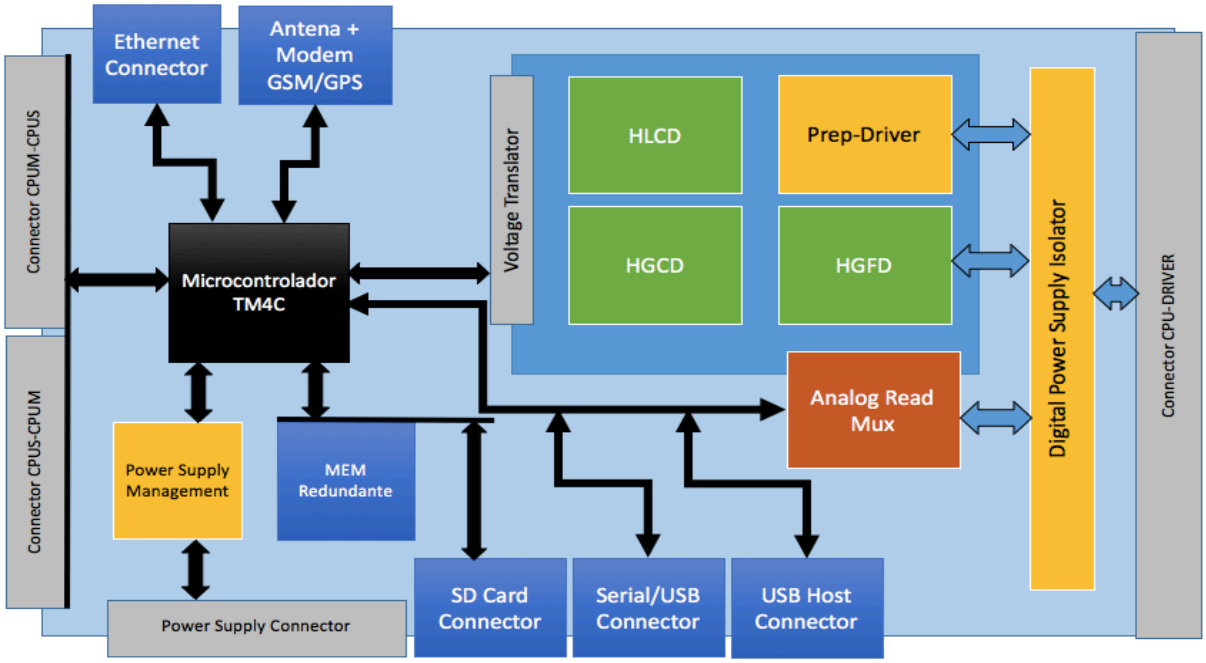
CPU_MASTER_EN *= Nível lógico baixo indica FHNR/FHR da CPU MASTER*
CPU_MASTER_ALIVE *= Sinal baixo ou alto por muito tempo indica FSNR/FSR da CPU MASTER*
CPU_SLAVE_EN *= Nível lógico baixo indica FHNR/FHR da CPU SLAVE*
CPU_SLAVE_ALIVE *= Sinal baixo ou alto por muito tempo indica FSNR/FSR da CPU SLAVE*
BLINKER_EN *= Nível lógico baixo indica FHNR/FHR da CPU SLAVE*
BLINKER_ALIVE *= Sinal baixo ou alto por muito tempo indica FSNR/FSR da CPU SLAVE*

Estas definições de sinais permitem que as falhas descritas sejam detectadas, e assim seja possível mudar o subsistema responsável pelo controle lógico. Existe um problema que pode ocorrer com sinais digitais fracos. Ao invés de simplesmente assumirem um valor lógico errado, falha que pode ser resolvida com um resistor de pull-up ou pull-down, o sinal pode simplesmente ficar alterando seu nível lógico de forma aleatória. Neste caso não é possível indicar se há uma condição de erro/falha no subsistema associado ou uma falha de conector. Quando isto ocorrer o sinal será considerado inconsistente, ou seja, inconclusivo. Definidas as condições da Tabela 4.4.1, para todas as situações de falha levantadas há um subsistema para exercer a função de controle lógico do DRIVER. Basta então detalhar ainda mais o projeto para que cada subsistema possa de fato interpretar os sinais de falha escolhidos.

Determinada a interface entre os subsistemas, e as funcionalidades necessárias em cada subsistema é detalhado o projeto. O resultado desta etapa pode ser chamado projeto conceitual, tal como pode ser observado nas Figura 4.4.1, 4.4.2 e 4.4.3. Nesta fase do projeto todas as funcionalidades possíveis de serem atribuídas e implementadas para cada subsistema já foram avaliadas no PI-FMEA, cabendo agora descrever blocos de hardware internos de cada subsistema para que o hardware possa atender os requisitos funcionais e não funcionais.

Com esta etapa finalizada deve-se detalhar o projeto de circuito de cada bloco de hardware planejado. A síntese no eixo da tecnologia deve ser analisada a partir deste ponto, pois as funcionalidades estão bem definidas, redundância de subsistemas e blocos de hardware funcionais também. O crescimento da confiabilidade neste ponto em diante depende da escolha apropriada da tecnologia de cada componente assim como projeto de circuito seguindo boas práticas de confiabilidade propostas na seção 3.5.

Figura 4.4.1 – Projeto conceitual CPU



FONTE: Produção do próprio autor

Projeto conceitual CPU:

Seguindo a proposta do PI-FMEA, a CPU deve ter um microcontrolador com alto desempenho e capacidade de executar rotinas de monitoramento de falha e acionamento entre intervalos determinísticos. Por questões de segurança e confiabilidade normalmente é indicado a utilização de um *Real-Time-Operating-System* (RTOS). A família TM4C da *Texas Instruments Inc.* apresenta grandes vantagens, por isso sua utilização será mantida para este trabalho.

Haverá dois conectores redundantes entre a CPU MASTER e a CPU SLAVE permitindo a comunicação serial entre o conjunto de CPU redundantes com segurança e confiabilidade. Este canal de comunicação pode ser utilizado para realizar um diagnóstico de falha preciso, mudanças de posse do

controle lógico do DRIVER, checagem de consistência dos sinais de falhas transmitidos para subsistema DRIVER e até mesmo para que a conectividade a Internet de uma das CPU possa ser utilizada pela outra. Diante do exposto cabe ao projeto de software implementar e gerenciar estas múltiplas funcionalidades para mitigação de falha e aumento de confiabilidade. De modo geral, essa decisão de projeto segue as conclusões obtidas pelo PI-FMEA, reafirmando assim sua importância.

Também estão previstas duas formas de conexão com a Internet. Uma interface ethernet será projetada, assim como a utilização de um modem GSM com funcionalidade de GPS, visando garantir data e hora precisas.

Além do cartão SD servindo como um grande espaço de endereçamento de memória não volátil, será previsto também um banco de memória não volátil redundante. Este fato é importante, pois o cartão SD tem natureza removível, podendo uma falha de operação (remoção involuntária) prejudicar o funcionamento do sistema. Ele é também baseado em memória FLASH cujas especificações de condições ambientais suportadas atendem no limite as especificadas para este projeto. Com objetivo de garantir que os requisitos não funcionais sejam atendidos, com relação a memória redundante, espera-se utilizar uma tipologia de memória com maior confiabilidade.

As interfaces de comunicação local serão: conector USB-Host e USB-Serial. A possibilidade de escolha entre interfaces aumenta a disponibilidade deste tipo de comunicação. Além disso, possuir um USB-Host, permite a utilização de modens 3G comercias que implementam ethernet em cima do protocolo USB. Esta opção, que deve ser implementada em software, garante ainda outra alternativa de conectividade remota.

Com todas as formas de conectividade à Internet previstas, considerando a CPU MASTER e CPU SLAVE funcionando simultaneamente, seria possível utilizar até quatro provedores de internet diferentes simultaneamente. Esta configuração permitiria alta confiabilidade de conectividade sem fio.

Também foram conceituados três blocos lógicos de detecção e mitigação de falhas em hardware. Aumentando a confiabilidade referente a várias falhas críticas apontadas pelo PI-FMEA. São elas:

Hardwired Logical Consistency Detector (HLCD):

Este bloco é responsável por ler os sinais de controle, proveniente do microcontrolador, de cada um dos focos semafóricos, para determinar se há inconsistência no plano de programação. Ele monitora se não há tentativa de acionar duas ou mais cores em um mesmo instante para um mesmo grupo semafórico. Caso isto ocorra, pode ser um indicativo de perda de consistência na memória, ou falha na análise do plano de programação no momento da gravação. Este bloco gera um sinal de erro, cuja nomenclatura utilizada será ERROR1.

Hardwired Green Conflict Detector (HGCD):

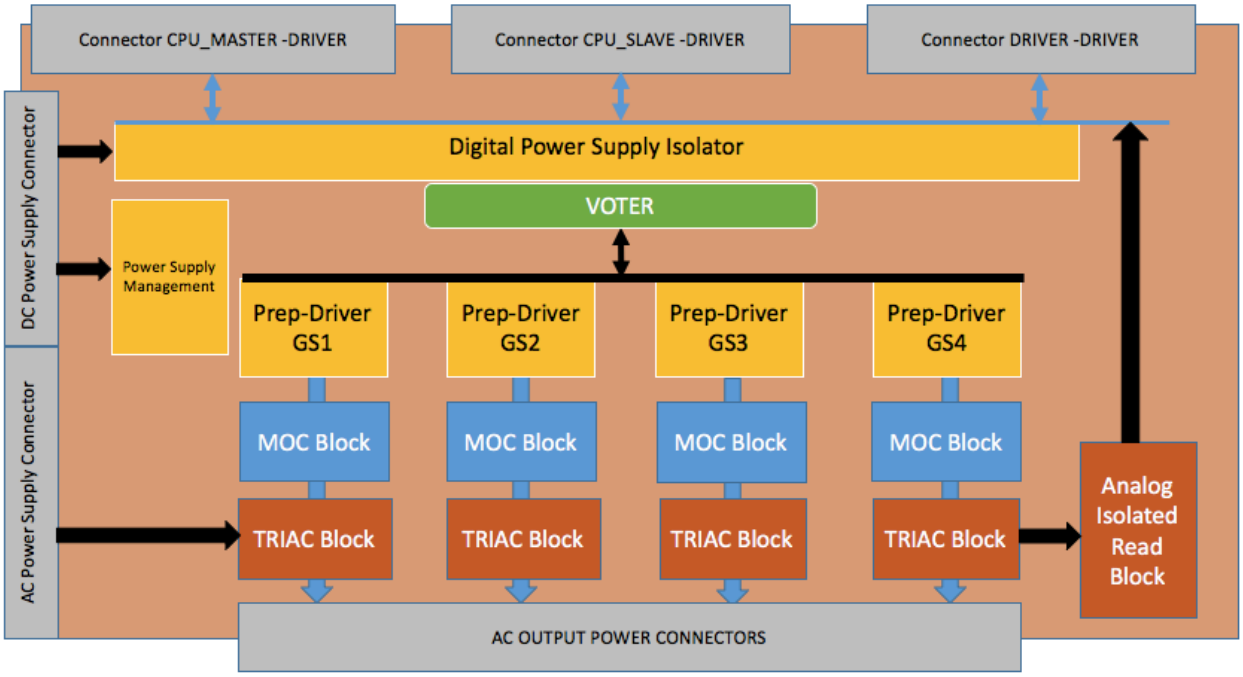
Este bloco realiza a detecção de conflito de verde em hardware através dos sinais dos focos verdes lidos pela interface analógica. Este procedimento de mitigação de falha é uma redundância em hardware que o microcontrolador também deverá executar em software. O modo *default* deste bloco, que confere a maior nível de confiabilidade, considera que nenhum dos quatro grupos semafóricos pode estar em verde simultaneamente. Mantendo o controlador neste nível de mais alta confiabilidade, o operador não poderá desabilitar este modo *default* , tendo que utilizar um novo conjunto de CPU redundantes se houver necessidade de utilizar maior conjunto de verdes simultâneos . Desconsiderando este nível de mais alta confiabilidade, é previsto que a CPU possa desabilitar este bloco, mantendo somente a detecção de conflito de verde em software. Esta capacidade de ser desabilitado permite que a CPU funcione parcialmente em caso de falha do HGCD. O sinal de erro gerado por este bloco será chamado ERROR2.

Hardwired General Fault Detector (HGFD):

Este bloco age diretamente no conector CPU-DRIVER e é responsável por indicar aos demais subsistemas a condição de funcionamento da CPU através dos sinais transmitidos para o DRIVER. Os sinais ERROR1 e ERROR2 são analisados, em conjunto com os sinais CPU_EN, CPU_ALIVE, BLINKER_EN e BLINKER_ALIVE. Neste bloco o sinal BLINKER_EN é usado para desabilitar o CPU_EN, de modo que, o BLINKER tenha poder de retirar o controle lógico da CPU, caso seja identificada falha pelos sinais CPU_EN e CPU_ALIVE.

Todos os blocos lógicos na CPU já estão previstos para que seja utilizada lógica TTL com níveis de tensão de alimentação e nível lógico mais altos a tensão de alimentação do microcontrolador. Para que isto aconteça estão previstos *Voltage Translators* para adequar os níveis lógicos dos pinos de entrada e saída do microcontrolador.

Figura 4.4.2 – Projeto conceitual DRIVER



FONTE: Produção do próprio autor

Projeto conceitual Driver

O DRIVER é o subsistema capaz de acionar os focos semafórico chaveando fases 127/220 Vca. Isto requer a utilização de semicondutores de potência tal como um TRIAC, ou SRC comum. Sendo assim, foi previsto no mínimo um TRIAC por foco semafórico, totalizando 12 TRIACs no mínimo. Estes componentes por sua vez são acionados por dispositivos opta-acopladores, como um MOC (componente optoacoplador) , que é responsável por isolar eletricamente o sinal lógico de controle, proveniente do *Voter*, Figura 4.4.2, e o acionamento de potência realizado pelo TRIAC.

Três conectores do DRIVER, são referentes a conexão com os demais subsistemas. CPU MASTER, a CPU SLAVE e BLINKER enviam sinais digitais de controle, sendo o *Voter* responsável por analisar os sinais indicadores de FHNR/FHR (CPU_MASTER_EN, CPU_SLAVE_EN, BLINKER_EN). O *Voter* então escolhe qual subsistema assumirá o controle lógico do DRIVER, permitindo assim que os sinais lógicos pertinentes controlem os blocos de hardware referentes ao acionamento de potência.

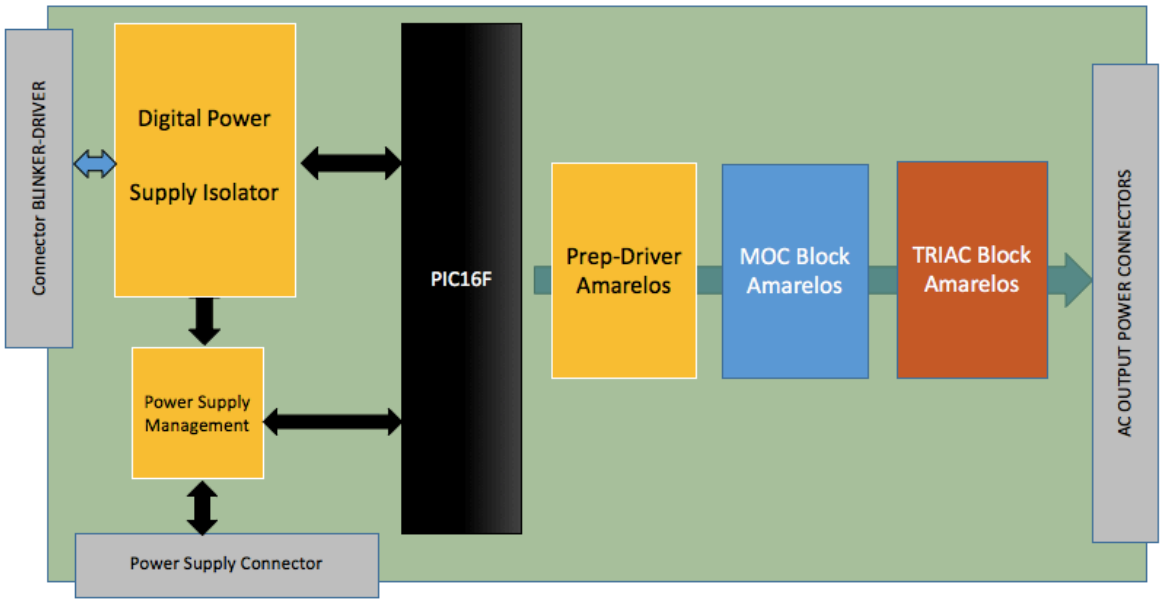
É possível também que a leitura de corrente de cada foco seja realizada. Esta funcionalidade do DRIVER é essencial para detecção e mitigação de uma grande quantidade de falhas identificadas no PI-FMEA. É necessário, entretanto, que o componente responsável pelo bloco *Analog Isolated Read* promova isolamento elétrico, fornecendo para os demais subsistemas, sinais de tensão

analógicos, análogos às correntes dos focos, compatíveis com os níveis de tensão dos demais subsistemas.

O bloco Prep-Driver contém simplesmente transistores integrados capazes de fornecer a corrente necessária para acionamento dos fotodiodos internos dos dispositivos MOC.

Projeto conceitual BLINKER

Figura 4.4.3 – Projeto conceitual BLINKER



FONTE: Produção do próprio autor

O BLINKER, segundo a especificação de confiabilidade (Seção 4.1), deve ser o subsistema de maior confiabilidade. Este subsistema, porém, é o mais simples em termos de funcionalidade. Levando em conta os princípios de confiabilidade, pretende-se tornar o circuito do BLINKER o mais simples possível.

Este subsistema deve simplesmente gerar um sinal de acionamento de modo piscante que por sua vez também funcionaria como indicação de funcionamento do software do microcontrolador do BLINKER. Este sinal é o BLINKER_ALIVE. O sinal BLINKER_EN indica a ocorrência de falha em algum outro subsistema, sendo este sinal usado pelo BLINKER para tomar o controle lógico do DRIVER em caso de falha de ambas CPUs.

Espera-se utilizar um microcontrolador da família PIC da *Microchip Inc*, pois sua arquitetura *Harvard* já sofreu ampla validação de confiabilidade por se tratar de uma arquitetura antiga e com

muitos anos de aplicação . Não sendo necessário um alto desempenho de processamento e/ou grande espaço de endereçamento para a aplicação no BLINKER, não se faz necessário a utilização de microcontroladores mais modernos e muitas vezes menos confiáveis. A utilização de um microcontrolador se faz também necessária para que se reduza a quantidade de circuitos integrados necessária para executar as funcionalidades de lógica combinacional e sequencial exigidas.

Este subsistema também deve conter um acionamento de potência para os focos amarelos, executando assim o modo piscante, para o caso de falha aparente do DRIVER.

4.5 *Part Count Analysis Prediction* para o controlador semafórico.

O nível de detalhamento de projeto obtido permite a avaliação quantitativa preliminar da confiabilidade aproximada obtida até então. Esta análise quantitativa aplicável é a PCRP, descrita na seção 3.3.2. Sendo assim, é possível determinar taxas de falhas genéricas para cada um dos tipos de componentes descritos abaixo, utilizando-se os dados empíricos apresentados em [5].

Deve-se observar que o PCRP promove uma percepção pessimista da confiabilidade, pois considera a utilização de componentes genéricos. Por esta razão, para alcançar a especificação de confiabilidade já nesta etapa de projeto necessita-se de pouca alteração conceitual de projeto ou seletividade na escolha da tecnologia dos componentes. Se tratando de componentes genéricos, a qualidade de fabricação destes componentes deve ser ponderada por um fator de qualidade, que é multiplicado pela a taxa de falha genérica uma taxa de falha mais realista do componente realmente utilizado.

O grande problema deste método é que a própria classificação dos componentes propostapela norma [5] é muitas vezes subjetiva, assim como a escolha do fator de qualidade. Foram utilizadas para essa análise um fator de qualidade igual a 1, para componentes considerados de baixa complexidade de fabricação e densidade de integração, e 2 para os demais componentes, visto que nesta etapa do projeto os critérios mais detalhados propostos por [5] ainda não podem ser avaliados.

Tabela 4.5.1 – Tabela de PCRP considerando todos os componentes em série

CONFIABILIDADE PCAP CONTROLADOR SEMAFÓRICO	Failure Rate Gb (Failures/ 10^6 hours)	Failure Rate Gf (Failures/ 10^6 hours)	Quality factor	Qtd.	Reliability Contribution Gb	Reliability Contribution Gf
RESISTORES	0.0012	0.0027	2	65	0.156	0.351
CAPACITORES	0.0036	0.0074	2	35	0.252	0.518
CIRCUITOS INTEGRADOS 1-100 GATES	0.0036	0.012	2	32	0.2304	0.768
CIRCUITOS INTEGRADOS MICROPROCESSADORES BIPOLAR	0.028	0.061	2	0	0	0
CIRCUITOS INTEGRADOS MICROPROCESSADORES MOS	0.048	0.089	2	2	0.192	0.356
MEMÓRIA EEPROM	0.0049	0.018	2	5	0.049	0.18
MEMÓRIA FLASH	0.0079	0.022	2	2	0.0316	0.088
CONECTORES DIGITAIS	0.0054	0.021	1	7	0.0378	0.147
CONECTORES DE POTÊNCIA	0.011	0.14	1	3	0.033	0.42
OPTO-ACOPLADORES/ISOLADORES	0.027	0.07	1	16	0.432	1.12
DISPOSITIVOS SRC/TRIAC	0.4	1.2	1	16	6.4	19.2
Total failure rate do controlador (Falhas / 10 ^6 horas)					7.8138	23.148
Total failure rate do controlador (Falhas / ano)					0.0684489	0.2027765

Tabela 4.5.2 - PCRP para o BLINKER

CONFIABILIDADE PCAP BLINKER	Failure Rate Gb (Failures/ 10^6 hours)	Failure Rate Gf (Failures/ 10^6 hours)	Quality factor	Qtd.	Reliability Contribution Gb	Reliability Contribution Gf
RESISTORES	0.0012	0.0027	2	5	0.012	0.027
CAPACITORES	0.0036	0.0074	2	5	0.036	0.074
CIRCUITOS INTEGRADOS 1-100 GATES	0.0036	0.012	2	2	0.0144	0.048
CIRCUITOS INTEGRADOS MICROPROCESSADORES BIPOLAR	0.028	0.061	2	0	0	0
CIRCUITOS INTEGRADOS MICROPROCESSADORES MOS	0.048	0.089	2	1	0.096	0.178
MEMÓRIA EEPROM	0.0049	0.018	2	0	0	0
MEMÓRIA FLASH	0.0079	0.022	2	0	0	0
CONECTORES DIGITAIS	0.0054	0.021	1	1	0.0054	0.021
CONECTORES DE POTÊNCIA	0.011	0.14	1	1	0.011	0.14
OPTO-ACOPLADORES/ISOLADORES	0.027	0.07	1	4	0.108	0.28
DISPOSITIVOS SRC/TRIAC	0.4	1.2	1	4	1.6	4.8
Total failure rate do controlador (Falhas / 10 ^6 horas)					1.8828	5.568
Total failure rate do controlador (Falhas / ano)					0.0164933	0.0487757

Tabela 4.5.3 - PCRP para o DRIVER

CONFIABILIDADE PCAP DRIVER	Failure Rate Gb (Failures/ 10^6 hours)	Failure Rate Gf (Failures/ 10^6 hours)	Quality factor	Qtd.	Reliability Contribution Gb	Reliability Contribution Gf
RESISTORES	0.0012	0.0027	2	30	0.072	0.162
CAPACITORES	0.0036	0.0074	2	10	0.072	0.148
CIRCUITOS INTEGRADOS 1-100 GATES	0.0036	0.012	2	10	0.072	0.24
CIRCUITOS INTEGRADOS MICROPROCESSADORES BIPOLAR	0.028	0.061	2	0	0	0
CIRCUITOS INTEGRADOS MICROPROCESSADORES MOS	0.048	0.089	2	0	0	0
MEMÓRIA EEPROM	0.0049	0.018	2	0	0	0
MEMÓRIA FLASH	0.0079	0.022	2	0	0	0
CONECTORES DIGITAIS	0.0054	0.021	1	3	0.0162	0.063
CONECTORES DE POTÊNCIA	0.011	0.14	1	1	0.011	0.14
OPTO-ACOPLADORES/ISOLADORES	0.027	0.07	1	12	0.324	0.84
DISPOSITIVOS SRC/TRIAC	0.4	1.2	1	12	4.8	14.4
Total failure rate do controlador (Falhas / 10 ^6 horas)					5.3672	15.993
Total failure rate do controlador (Falhas / ano)					0.0470167	0.1400987

FONTE: Produção do próprio autor

Tabela 4.5.4 - PCRP para a CPU

CONFIABILIDADE PCAP CPU	Failure Rate Gb (Failures/ 10^6 hours)	Failure Rate Gf (Failures/ 10^6 hours)	Quality factor	Qtd.	Reliability Contribution Gb	Reliability Contribution Gf
RESISTORES	0.0012	0.0027	2	30	0.072	0.162
CAPACITORES	0.0036	0.0074	2	20	0.144	0.296
CIRCUITOS INTEGRADOS 1-100 GATES	0.0036	0.012	2	20	0.144	0.48
CIRCUITOS INTEGRADOS MICROPROCESSADORES BIPOLAR	0.028	0.061	2	0	0	0
CIRCUITOS INTEGRADOS MICROPROCESSADORES MOS	0.048	0.089	2	1	0.096	0.178
MEMÓRIA EEPROM	0.0049	0.018	2	5	0.049	0.18
MEMÓRIA FLASH	0.0079	0.022	2	2	0.0316	0.088
CONECTORES DIGITAIS	0.0054	0.021	1	3	0.0162	0.063
CONECTORES DE POTÊNCIA	0.011	0.14	1	1	0.011	0.14
OPTO-ACOPLADORES/ISOLADORES	0.027	0.07	1	0	0	0
DISPOSITIVOS SRC/TRIAC	0.4	1.2	1	0	0	0
Total failure rate do controlador (Falhas / 10 ^6 horas)					0.5638	1.587
Total failure rate do controlador (Falhas / ano)					0.0049389	0.0139021

FONTE: Produção do próprio autor

Nas Tabelas 4.5.1 ,4.5.2 ,4.5.3 e 4.5.4 estão apresentados as PCRP para cada subsistema, e também considerando cada subsistema em série com os demais (Tabela 4.5.1). A análise da Tabela 4.5.1 contabilizou os componentes de todos os subsistemas, sendo assim indica a confiabilidade de que nenhum subsistema tenha nenhuma falha.

A Tabela 4.5.2 indica a análise isolada do BLINKER, que representa a confiabilidade em se manter o modo piscante funcionando. A Tabela 4.5.3 e 4.5.4 apresentam a análise da CPU e do DRIVER.

Foi possível ter uma noção quantitativa valiosa sobre a alocação de confiabilidade no sistema. O quantitativo de cada componente foi feita assumindo-se um risco da aproximação. Assim como no detalhamento de arquitetura, estas premissas poderão mudar.

As taxas de falhas obtidas estão bem abaixo da especificação de confiabilidade proposta, porém sabe-se que a taxa de falha individual de cada componente, usadas nesta análise se encontra exacerbadamente alta. Muitos componentes podem ser encontrados em suas versões de alta confiabilidade ou EP (*Enhanced Products*) que devem ser selecionados para que a confiabilidade cresça no eixo de tecnologia (Seção 3.1).

A concepção da arquitetura já contribuiu significativamente para a confiabilidade. Foi possível também definir diretrizes no eixo de projeto para aumento a confiabilidade, assim como necessidade de análise no eixo de tecnologia.

Todas as análises foram realizadas considerando dois padrões de condição ambientais Gb e Gf definidos em [5]. Para as condições Gb a confiabilidade da CPU já se encontram em níveis aceitáveis, assim como a confiabilidade do BLINKER esta em níveis próximos dos requeridos. Para a condição Gf, que é a mais severa, a confiabilidade ainda está baixa. Observa-se os dispositivos de estado sólido com o MOC e o TRIAC como responsáveis pela alta taxa de falhas no DRIVER e no BLINKER. O projeto detalhado deve prever algum tipo de redundância destes componentes para minimizar os efeitos de suas altas taxas de falha. Considerando inicialmente a redundância de CPU prevista na arquitetura como uma redundância paralela ativa, a taxa de falha do conjunto de CPUs ainda cairia 0.001 em média, demonstrando que o gargalo de confiabilidade esta nos dispositivos de potência. Relacionando a probabilidade de falhas, tal como proposto na Seção 3.1, com os resultados do PCRP foram obtidos os seguintes valores:

Ambiente Gb:

$$\lambda_{FNR1} = \lambda_{CPU} + \lambda_{DRIVER}$$
$$\lambda_{FNR1} = 0.0049 + 0.0470 = 0.0519\ falhas/ano$$

$$\lambda_{FNR2} = 0.0164\ falhas/ano$$
$$P_{FNR1}(0 < t < 2\ anos) = 9\% \quad P_{FNR2}(0 < t < 5\ anos) = 7\%$$

Ambiente Gf:

$$\lambda_{FNR1} = \lambda_{CPU} + \lambda_{DRIVER}$$

$$\lambda_{FNR1} = 0.013 + 0.141 = 0.154\ falhas/ano$$

$$\lambda_{FNR2} = 0.048 falhas/ano$$

$$P_{FNR1}(0 < t < 2\ anos) = 28\% \quad P_{FNR2}(0 < t < 5\ anos) = 21\%$$

PROJETO HARDWARE DETALHADO

Realizada a definição da arquitetura e a análise de confiabilidade através de *Part Count Reliability Prediction*, foram implementados cada um dos blocos descritos no Capítulo 4. Nesta etapa de detalhamento do projeto de circuito, sabe-se o projeto tem impacto na confiabilidade, entretanto devido a sua complexidade foi considerada uma implementação de acordo com o domínio de eletrônico adquirido.

Foram atingidos todos os requisitos funcionais com a implementação proposta, cabendo agora a análise no eixo de tecnologia o fator de maior contribuição para a confiabilidade. Técnicas de projeto de circuito para aumento da confiabilidade, propostas no Capítulo 3, foram utilizadas pontualmente. Maior esforço de projeto foi dedicado a blocos/componentes genéricos que apresentaram alta taxa de falha na análise do Capítulo 4.

Com relação a maioria dos componentes, espera-se que a escolha de componentes e tecnologias de montagem adequadas permitam que a taxa de falha real do sistema seja suficientemente superior em comparação com a taxa de falha calculada com componentes genéricos e quantitativos aproximados.

Este capítulo pretende esclarecer decisões de projeto e característica específicas da implementação, assim como a análise de confiabilidade para avaliar a confiabilidade teórica do circuito previsto. Os esquemáticos completos dos subsistemas estão no Apêndice B, e layout das placas de circuito impresso no Apêndice C.

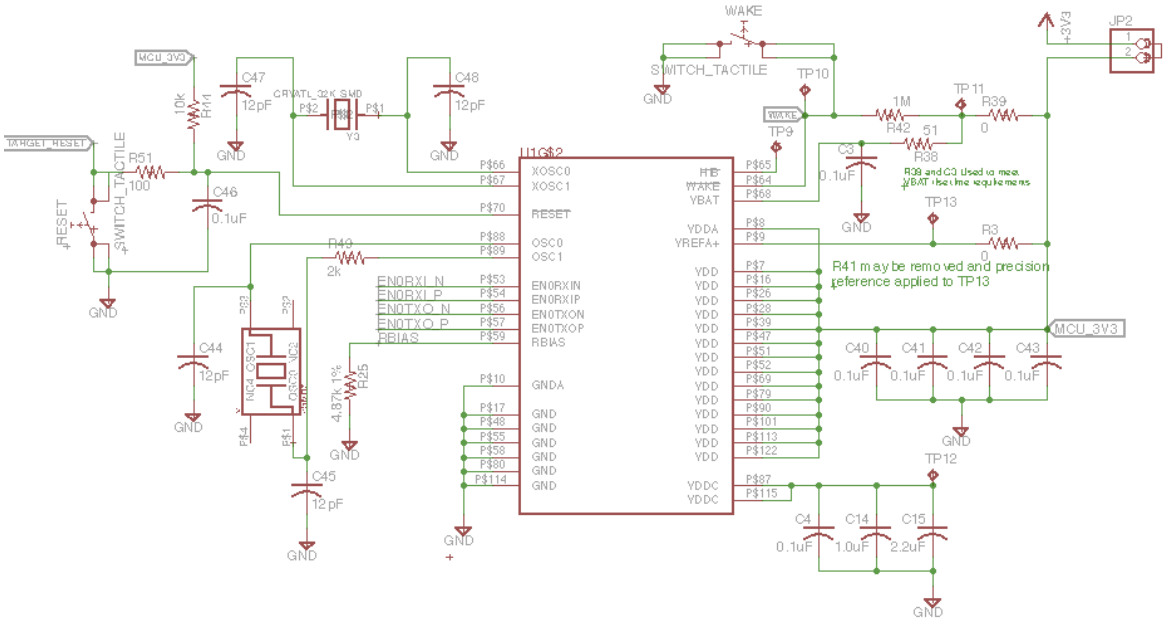
5.1. Projeto da CPU

5.1.1. O Microcontrolador e Alimentação

O microcontrolador escolhido foi o TIVA TM4C129ENCPDT, Figura 5.1.1.1, da *Texas Instruments*. Devido a sua arquitetura ARM-Cortex M4 e vasta quantidade de periféricos acreditava-se que ele possuiria quantidade de pinos de uso geral, assim como periféricos em abundância. Posteriormente averigou-se uma limitação de pinos, Figura 5.1.1.2, que impossibilitou, por exemplo, a criação de redundância de conectores entre a CPU MASTER e CPU SLAVE. Entretanto sua utilização foi mantida para preservar o conceito do projeto. Entendia-se que a conexão redundante deveria utilizar um periférico de comunicação, UART, isolado, diferente da outra conexão, porém não houve disponibilidade de pinos com funcionalidade da UART. Considerando os valores de taxa

de falhas obtidas para a CPU no Capítulo 4, que representariam inicialmente um pior caso de confiabilidade, acredita-se não ser necessário utilização de redundância de conectores.

Figura 5.1.1.1 – Hardware mínimo TM4C129ENCPDT



FONTE: Produção do próprio autor

A escassez de pinos que possibilitassem a conexão redundante, foi porque, esperava-se que a CPU possuísse conectores com compatibilidade de pinos com a EK-TM4C129NCPDT. Esta por sua vez é uma placa de desenvolvimento de baixo custo, que permitiu assim uma validação mais rápida dos demais periféricos da CPU.

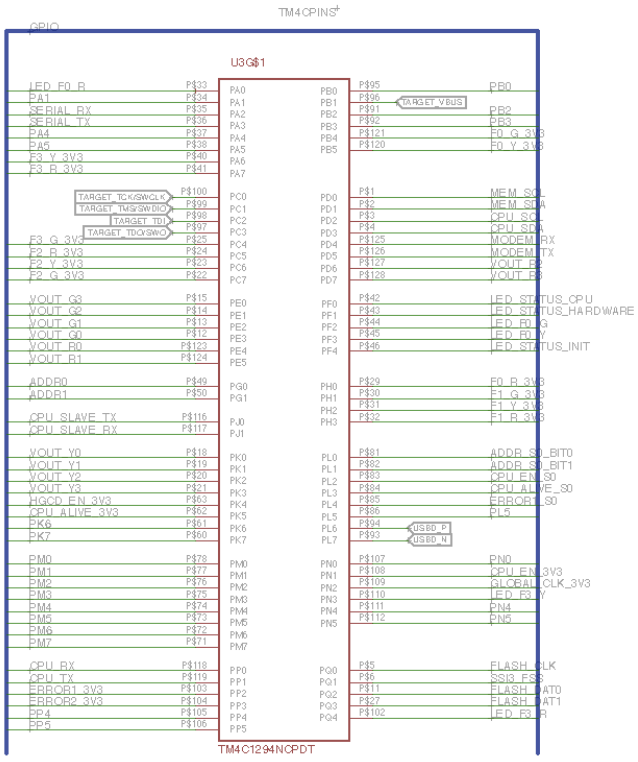
Além disso há dois grandes motivos para sua escolha. Primeiramente, este microcontrolador foi planejado para aplicações industriais, possuindo relatório de confiabilidade detalhado, fornecidos pelo fabricante, apresentando uma taxa de falhas extremamente mais baixa em comparação com os valores apontados em [5] para componentes genéricos desta natureza.

A segunda grande razão é que este microcontrolador também possui um periférico ETHERNET nativo, permitindo conexão com alta taxa de transmissão com este tipo de enlace. Esta característica não é comum em grande parte dos microcontroladores, permitindo assim ampla conectividade de forma segura, rápida e facilitada.

Os mais conhecidos, e confiáveis, sistemas operacionais de tempo real (RTOS) também tem versões de seus kernels compiladas para este microcontrolador, como o MicroC/OS(Micrium)

e o FreeRTOS. A segurança e a confiabilidade do software, tão necessária e difícil neste tipo de aplicação torna-se muito mais próxima de ser alcançada, visto que um sistema operacional de propósito geral é altamente não recomendável para aplicações de alto risco.

Figura 5.1.1.2 - Pinos funcionais do microcontrolador



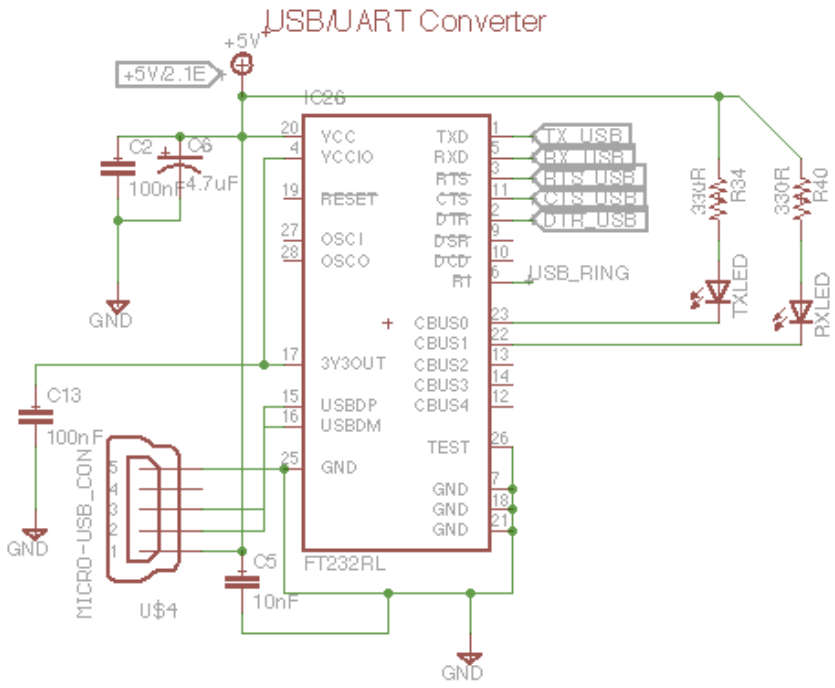
FONTE: Produção do próprio autor

Para o sistema de alimentação foram previstas uma alimentação 24V fornecida por uma fonte industrial isolada e externa ao equipamento, necessitando assim de reguladores de alta confiabilidade para adequar os níveis de tensão para 5V e 3.3V, que serão utilizados pelos circuitos lógicos da placa. A escolha dos componentes foi baseada na disponibilidade de relatórios de confiabilidade, assim como dados de referência para projeto do circuito. Além da escolha de componentes de confiabilidade conhecida utilizou-se o *Diode Array* contra surto na alimentação proveniente do conector USB, assim como não alimentação principal.

5.1.2. Memória e Interfaces USB e Ethernet

O circuito integrado FT232RL, Figura 5.1.2.1, foi utilizado para que houve-se um conector USB que emulasse uma porta serial. Esta escolha é importante, criando assim uma interface de comunicação local segura entre o controlador semafórico e computador de propósito geral.

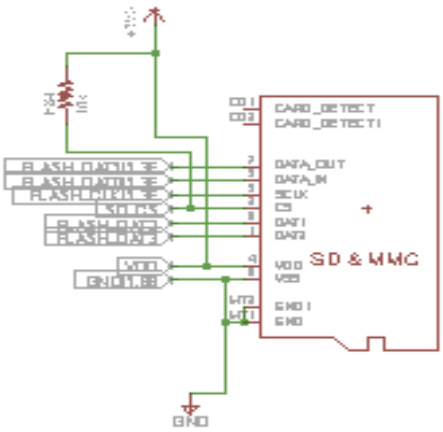
Figura 5.1.2.1 - Interface USB/SERIAL



FONTE: Produção do próprio autor

Deste modo a gravação de planos de programação semafórica poderá ser feita localmente através de conversor USB/Serial ou com a inserção de um cartão SD com os planos de programação anteriormente armazenado, Figura 5.1.2.2.O cartão SD, embora forneça capacidade de armazenamento muito além da necessária para este tipo de aplicação, é considerado um ponto de risco. O gerenciamento de sistemas de arquivos vinculado a um cartão SD assim como a natureza tecnológica da memória FLASH constitui um ponto de falha, não suportando normalmente altas temperaturas (maiores que 60•C).

Figura 5.1.2.2 - Slot SD card

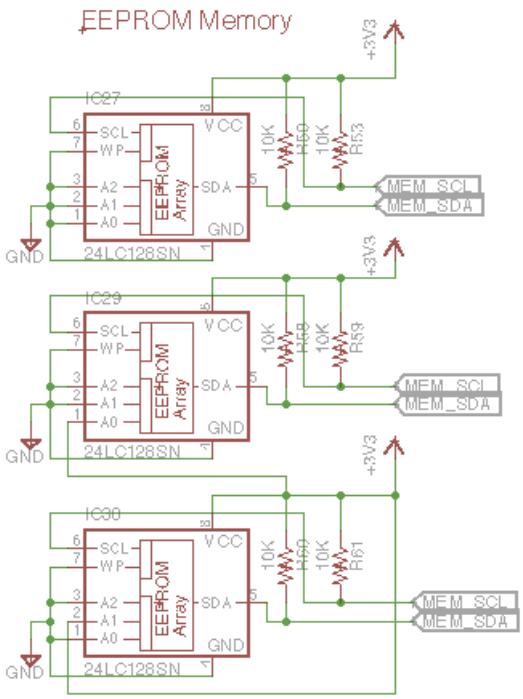


FONTE: Produção do próprio autor

Com o objetivo de criar redundância de armazenamento em memória não volátil, foi prevista uma rede de memórias EEPROM, prevendo inicialmente um conjunto de 128Kbits por memória. Este tipo de memória apresenta alta confiabilidade normalmente, podendo suportar temperaturas bem mais elevadas.

A utilização de memórias menores, o estritamente suficiente, também aumenta a confiabilidade, observando-se que os dados de confiabilidade genéricos propostos em [7] indicam maior taxa de falha para memórias maiores. Esta escolha de memória permite armazenar um plano semafórico com grande confiabilidade, ajudando a mitigar falhas de consistência de memória, mantendo o sistema funcionando corretamente.

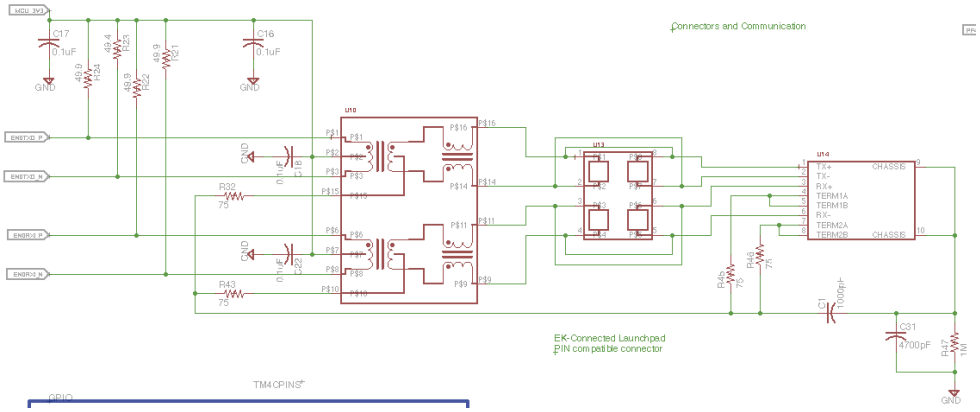
Figura 5.1.2.3 – Memória redundantes



FONTE: Produção do próprio autor

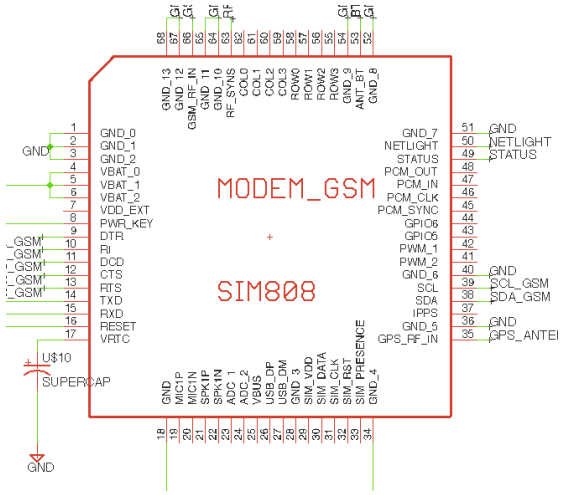
A conectividade com a Internet foi alcançada de duas formas: um enlace Ethernet MAC 10/100 nativo, Figura 5.1.2.3, e um modem GSM SIM808, Figura 5.1.2.4. Foi previsto um conector RJ45 seguido de um *Diode Array*, SLVU2.8-4.TBT, para prevenção de surtos nos conectores. Se tratando o protocolo Ethernet, de uma comunicação com sinais diferenciais, e considerando que a comunicação deve ser sempre entre equipamento com alimentação supostamente isoladas foi utilizado um transformador de pulso para manter o isolamento dos sinais, HX1198FNL.

Figura 5.1.2.4 - Enlace Ethernet 10/100 MAC



FONTE: Produção do próprio autor

Figura 5.1.2.5 - Modem GSM/GPS SIM808



FONTE: Produção do próprio autor

Foi escolhido o modem SIM808, pois já havia o domínio deste componente. Ele fornece conectividade através de comandos seriais AT, tornando assim a implementação da comunicação remota entre o controlador semafórico e o CCO facilitada. Este modem também funciona como GPS, atendendo aos requisitos funcionais propostos sem acréscimo de componentes.

Um ponto crítico do projeto com modem é o projeto das antenas necessárias, tanto para o GSM quanto para o GPS. Para tal foram utilizados padrões de antena fornecidos pelo fabricante, sendo previstos na PCB resistores e capacitores caso seja observada a necessidade de casamento de impedâncias devido ao roteamento da PCB.

Há presente na placa, o LED 16 e 17, para debug do status de conexão do modem, com as seguintes indicações respectivamente.

- 64ms on/ 800ms off: SIM808 sem registro na rede GSM
- 64ms on/ 3000ms off: SIM808 registrado na rede GSM
- 64ms on/ 300ms off: comunicação GPRS estabelecida

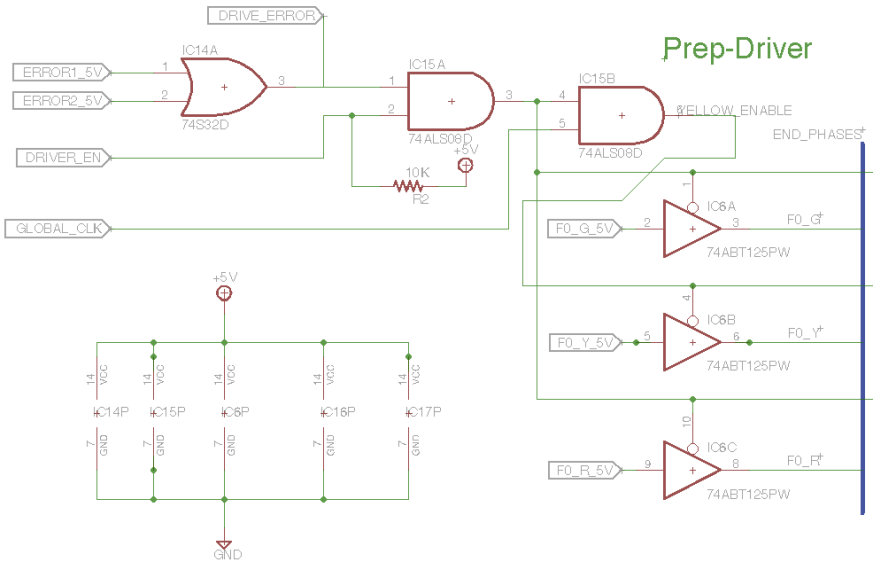
5.1.3. HGCD, HLCD, Prep Driver

Os blocos de hardware HGCD e HLCD foram implementados com portas lógicas discretas seguindo a mesma descrição e nomenclatura de sinais apresentada no Capítulo 4. Espera-se que o HGCD tenha como entrada os valores de tensão referentes as leituras dos focos semafóricos, que irão variar entre 0 e 3.3V, sendo que com o acionamento de um foco verde este sinal seja suficiente para acionar a entrada de uma porta lógica TTL. Os componentes utilizados foram especificamente SN74HC125PWR, SN74AS08D, SN74LS32D, pois possuem relatório de confiabilidade disponível e versão de maior confiabilidade a venda (*Enhanced Product* , versão de uso militar e aeroespacial.)

Para analisar os erros, descritos como ERROR1 e ERROR2, foi implementado um bloco chamado Prep-Driver, Figura 5.1.3.1. Este bloco faz com que o controlador entre em modo piscante automaticamente em caso que qualquer uma das falhas. Este modo piscante determinado a partir da CPU, diferentemente do piscante acionado pelo BLINKER, trata-se a situação em que a CPU está funcionando corretamente, sem travamento de software, porém existe falha aparente nos demais subsistemas. O sinal DRIVER_EN permite assim que a CPU tente avaliar a persistência da falha do restante do sistema, reenviado sinais de controle relacionados ao plano de programação semafórica. Isso permite que falha reparáveis ou que não durem sejam identificadas.

No caso de erro de consistência do plano de programação semafórico, o controlador pode por exemplo localizar a falha de consistência e repará-la, mantendo-se assim fora do modo piscante.

Figura 5.1.3.1 - Prep-Driver



FONTE: Produção do próprio autor

5.1.8 Análise de Confiabilidade utilizando PSAP

A CPU apresenta uma grande quantidade de componentes e assim as relações de causalidade entre os sinais internos da CPU é de difícil avaliação. Por isso, na CPU foi considerando que falha em qualquer componente significaria FNR2 (falha não reparável). Esta suposição é apropriada pois a PCB é um ponto de falha comum a todos, logo qualquer falha de componente que implique em deterioração de sinais importantes na PCB implicaria em falha generalizada.

Grande parte dos componentes escolhidos, possuem relatórios de confiabilidade fornecidos pela sua fabricante, fornecendo assim taxas de falha realistas a disposição. Todos os relatórios de confiabilidade observados tinham condições ambientais de teste superiores mais restritivas do que as condições ambientais nominais. Por esta razão foram utilizadas as taxas de

falha fornecidas pela fabricante ao invés de modelos teóricos imprecisos fornecidos por [5] através de *Part Stress Analysis Prediction* (PASP).

Ao final foram contados os componentes de cada tipo, considerando que, cada tipo recebeu então uma taxa de falha atribuída utilizando o método de PSAP proposto em [5], Seção 3.3.

Deve-se ressaltar que a equação apresentada na seção 3.3 para cálculo da taxa de falha via PSAP, é uma equação genérica, sendo que [5] indica variações específicas para cada tipo de componente. Foram desta forma considerados os componentes em maior quantitativo e de maior relevância para a confiabilidade. Segue a análise de confiabilidade utilizando PSAP e relatórios de confiabilidade de fabricantes, na Tabela 5.1.8.1.

Tabela 1 - Análise de confiabilidade final CPU

CONFIABILIDADE PCAP CPU	Failure Rate base (Failures/ 10^6 hours)	Environment factor Gb	Environment factor Gf	Quality factor	Temperature factor 65 graus	Qtd.	Failure rate Gb PSAP	Failure rate Gf PSAP	Reliability Contribution Gb	Reliability Contribution Gf
Dados genéricos para Part Stress Analysis MIL-HDBK-217F										
RESISTORES	0.0014	1	3	0.1	1	80	0.00014	0.00042	0.0112	0.0336
CAPACITORES	0.0021	1	2	3	1	43	0.0063	0.0126	0.2709	0.5418
CIRCUITOS INTEGRADOS MICROPROCESSADORES MOS	0.048	0.5	2	2.4	0.42	1	0.024192	0.096768	0.024192	0.096768
MEMÓRIA EEPROM	0.0034	0.5	2	2.4	1.6	3	0.006528	0.026112	0.019584	0.078336
CONECTORES DIGITAIS	0.0111	1	1	2.4	1	2	0.02664	0.02664	0.05328	0.05328
PCB CPU SMD	0.00552	1	2	20	3	1	0.3312	0.6624	0.3312	0.6624
DIODE ARRAY	0.003	1	6	2.4	3.3	3	0.02376	0.14256	0.07128	0.42768
TRANSISTOR MOSFET	0.012	1	6	2.4	2.1	1	0.06048	0.36288	0.06048	0.36288
TRANSFORMADOR DE PULSO	0.003	1	6	5	1	1	0.015	0.09	0.015	0.09
SWITCH E PSUH BUTTONS	0.0027	1	3	1	1	5	0.0027	0.0081	0.0135	0.0405
SUBTOTAL failure rate do controlador (Falhas / 10 ^6 horas)									0.870616	2.387244
SUBTOTAL A failue rate do controlador (Falhas / ano)									0.007626596	0.020912257
Dados de relatório de confiabilidade Texas Instruments inc.										
TM4C129ENPDCT	2.11084E-05	1	2	1	1	1	2.1108E-05	4.2217E-05	2.11084E-05	4.22169E-05
TPS2052BD	1.8758E-06	1	2	1	1	1	1.8758E-06	3.7516E-06	1.8758E-06	3.75161E-06
LM2596S-5.0	2.45653E-05	1	2	1	1	1	2.4565E-05	4.9131E-05	2.45653E-05	4.91307E-05
TPS73733DCQ	1.8758E-06	1	2	1	1	1	1.8758E-06	3.7516E-06	1.8758E-06	3.75161E-06
TPD4S012DRYR	9.06738E-06	1	2	1	1	1	9.0674E-06	1.8135E-05	9.06738E-06	1.81348E-05
ISO7221CD	1.8758E-06	1	2	1	1	18	1.8758E-06	3.7516E-06	3.37645E-05	6.75289E-05
SN74HC125PWR	2.84416E-06	1	2	1	1	4	2.8442E-06	5.6883E-06	1.13766E-05	2.27532E-05
SN74AS08D	3.2809E-06	1	2	1	1	4	3.2809E-06	6.5618E-06	1.31236E-05	2.62472E-05
SN74LS32D	6.39416E-06	1	2	1	1	6	6.3942E-06	1.2788E-05	3.8365E-05	7.67299E-05
SUBTOTAL B Total failure rate do controlador (Falhas / ano)									0.000134014	0.000268028
TAXA DE FALHA (FALHAS/ANO) CPU									0.00776061	0.021180285

FONTE: Produção do próprio autor

A Tabela 5.1.8.1 mostra que as especificações de confiabilidade foram atendidas plenamente com uma única CPU para a condição ambiental Gb (determinada em [5]), considerando a margem de aproximação. Para a condição ambiental Gf a redundância prevista para CPU é capaz de fazer com a confiabilidade geral do sistema de CPU redundante fique dentro do especificado.

$$R_{CPU_{MASTER}}(2) = R_{CPU_{SLAVE}}(2) = e^{-0.021*2} = 0.9585 \ (confiabilidade\ em\ 2\ anos)$$

$$Sabendo - se\ \ pela\ redundância\ paralela\ [7]\ \ que$$

$$Q_{SISTEMA_{CPU}}(2) = Q_{CPU_{MASTER}}\ Q_{CPU_{SLAVVE}} = 0.00172$$

Logo conseguimos atender os requisitos:

$$R_{SISTEMA_{CPU}}(2) = 0.0998\ \ implicando\ \lambda =\ 0.008\ falhas/ano$$

Com este resultado pode-se perceber que a alocação de confiabilidade inicial, relacionada com a falha do conjunto CPU e DRIVER, descrita na Seção 4.1, foi modificada. Originalmente previu-se uma alocação de confiabilidade igual entre os subsistemas, com taxa de falha 0.012 falhas/ano, entretanto o bom desempenho de confiabilidade da CPU permitiu que o DRIVER apresente uma confiabilidade menor do que a prevista inicialmente.

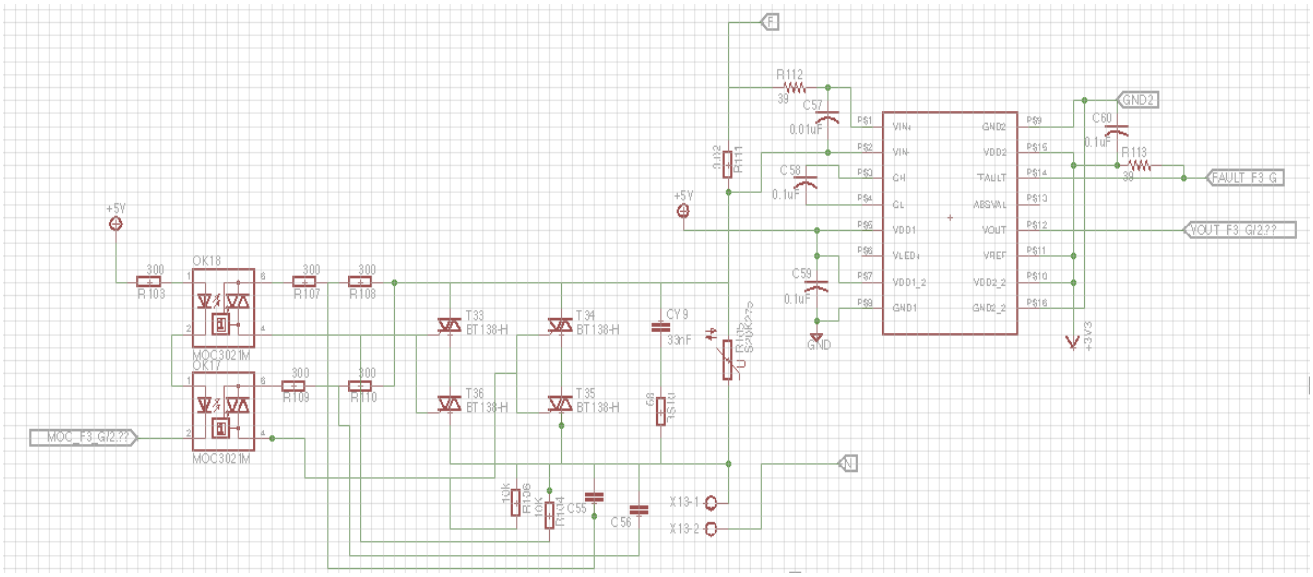
5.2. *Projeto do DRIVER*

5.2.1 Banco de TRIACS e leitura analógica

O DRIVER foi o subsistema que apresentou a maior disparidade entre a confiabilidade objetivo e a apresentada na *Part Count Reliability Prediction* (PCRP). Os componentes que apresentaram a maior taxa de falha foram optoacopladores e dispositivos TRIAC, sendo assim os esforços e a complexidade resultante do circuito tiveram objetivo de aumentar a confiabilidade intrínseca deste bloco de hardware.

Em [7], é fornecido estatísticas de modos de falhas para componentes genéricos, contendo assim dados considerando os modos de falhas de TRIAC. Perto de 80% das falhas de TRIAC são relacionadas a curto entre os terminais principais (MT1 e MT2), 17% das falhas são referente a curto entre terminal de gate e um dos terminais principais.

Figura 5.2.1.1 - Banco de TRIACs



FONTE: Produção do próprio autor

A técnica de circuito utilizada, uma alteração em relação ao proposto em [15], foi a inserção de dois ramos paralelos de dois TRIACs em série operando ambos no terceiro quadrante [16]. Desta forma cria-se uma redundância de TRIAC tanto contra curto quanto em caso de modo falha aberto. Espera-se dessa foram a redução da taxa de falha equivalente.

Há também redundância de MOC em caso de curto entre o gate do TRIAC e um dos seus terminais. Dois optoacopladores MOC foram utilizados, considerando a utilização do MOC3081, pino compatível com o MOC3021, apresentado no esquemático. Ambos não possuem relatório de confiabilidade oficiais disponível, de modo que serão utilizadas taxas de falhas genérica fornecidas por [5].

Foi-se estudada a utilização de CLA60MT, BTA138, MAC223 e BTA330, sendo todos os 4 componentes plenamente em conformidade as especificações de acionamento dos focos. Todos possuem compatibilidade de pinos, sendo o CLA60MT o único com relatório de confiabilidade disponível. Para este trabalho, devido a limitações logísticas foi considerado a

utilização de BTA138 ou MAC223, sendo assim utilizados dados de confiabilidade genéricos [5].

A rede *snubber* foi calculada segundo [16] e [17]. Considerando que não previsão de dimerização da fase para gerenciamento de potência entregue, a rede snubber foi dimensionada para não permitir variações de tensão muito maiores que a necessidade da rede elétrica. Isto serve para proteger o TRIAC de chaveamento demasiadamente rápidos e picos de tensão, muitas vezes responsáveis por falhas deste tipo de componente.

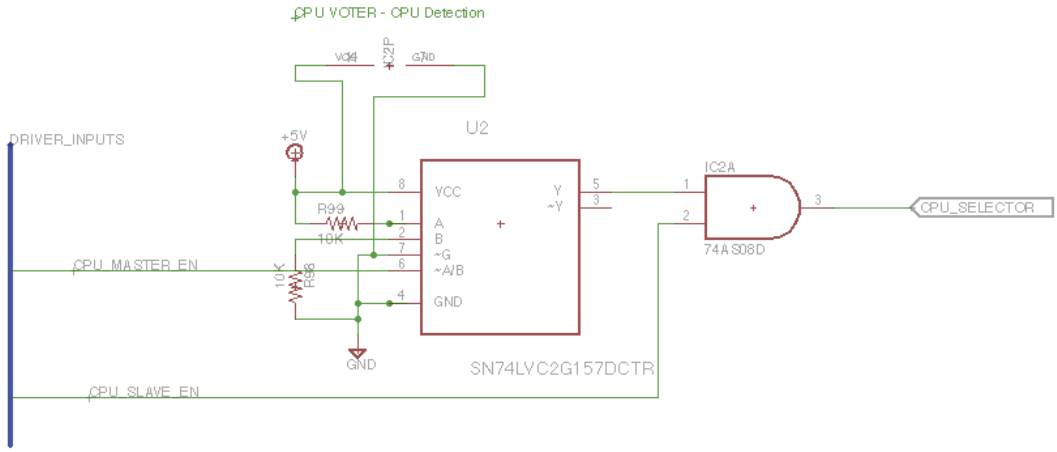
Além disso, para cada grupo semafórico foram utilizados varistores de proteção contra picos na rede de alimentação, e foi previsto um único fusível geral para acionamento dos focos semafóricos. A utilização de fusível local para cada foco semafórico restringiria a utilização de focos semafóricos em paralelo, por esta razão foi descartada esta alternativa.

A capacidade de corrente dos dois ramos combinados, é portanto, superior a 40 A, entretanto, sabe-se esta corrente esta muito acima, conferindo confiabilidade ao sistema. Devido calor, é muito importante que todos os TRIACs estejam termicamente em contato, ou seja, que suas temperaturas tendam ao equilíbrio. Em caso de alta temperatura a resistência interna do TRIAC irá variar, podendo prejudicar mais um ramo. A corrente de *gate* requerida também irá variar podendo prejudicar o TRIAC em série, acionado em conjunto, este TRIAC gêmeo não tenha a mesma temperatura.

5.2.2 Voter

O voter é o bloco de hardware responsável por efetivar a redundância de acionamento proporcionada pela CPU_MASTER, CPU_SLAVE e BLINKER, controlando assim qual subsistema estará atuando como unidade de controle lógico. Tendo isto em consideração o voter é um ponto de falha crítico. Para isto foram escolhidos componentes (multiplexadores) com relatórios de confiabilidade oficiais disponíveis e fornecimento em versões de alta confiabilidade *(Enhanced Products*, versão militar e aeroespacial) .

Figura 5.2.2.1 Voter seletor de CPU



FONTE: Produção do próprio autor

O Voter foi dividido em duas partes, um bloco seletor de CPU, Figura 5.2.2.1, e o bloco seletor de sinais, Figura 5.2.2.2. Os sinais analisados para identificar qual CPU esta ativa são o CPU_SLAVE_EN e CPU_MASTER_EN, desta forma é possível avaliar qual CPU não contém erro de hardware não reparável. O BLINKER é o subsistema capaz de desabilitar estes sinais, podendo assim fazer com que ele assuma o controle, tal como pode-se notar na Figura 5.2.2.2.

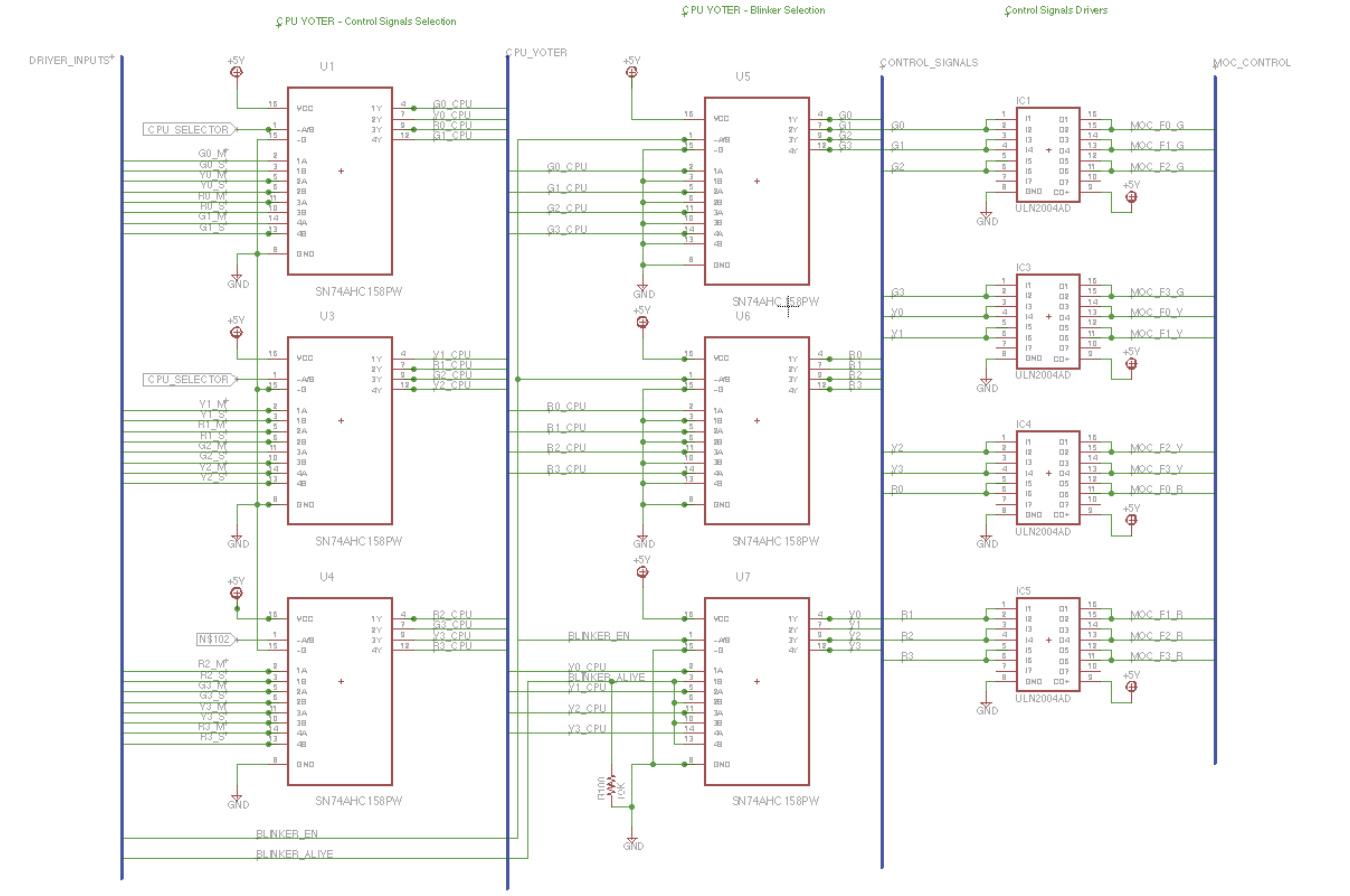
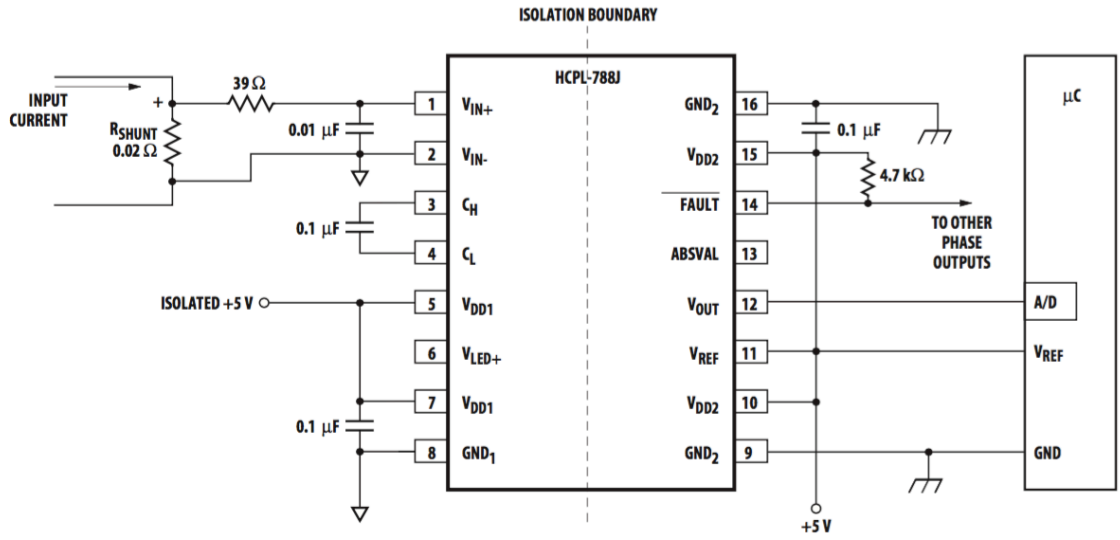


Figure 5.2.2.2 - Voter -Seletor de sinais

O BLINKER_EN é o sinal utilizado para avaliar se, de fato serão utilizados os sinais de controles de uma das CPU ou se o controlador será posto em modo piscante. Em caso de BLINKER_EN corretamente habilitado o modo piscante será acionado através do DRIVER.

5.2.3 Leitura analógica dos focos

Figure 5.2.3.1 - Sensor de corrente



FONTE: ANALOG DEVICES, 2015

A Figura 5.2.3.1 apresenta o sensor de corrente utilizado para leitura de cada um dos focos. O HCPL-788J é um sensor de corrente com isolamento entre duas fontes de alimentação. Através da corrente em um resistor *shunt* de baixa resistência, ele fornece um sinal de tensão analógico adequado ao *range* de leitura definido pelo pino Vref, uma tensão correspondente ao valor absoluto, e um sinal de falta indicando curto. Todos esses sinais de leitura já esta isolados eletricamente da leitura do sinal, permitindo a manutenção da proposta desta arquitetura de ter alimentações isoladas para cada módulo.

5.2.4 Análise de confiabilidade DRIVER

Grande parte dos componentes escolhidos, possuem relatórios de confiabilidade fornecidos pela fabricante, fornecendo assim taxas de falha realistas a disposição. Todos os relatórios de confiabilidade observando tinha condições ambientais de teste superiores mais restritivas que as condições ambientais nominais. Por esta razão foram utilizadas as taxas de falha fornecidas ao fabricante ao invés de modelos teóricos imprecisos fornecidos por [5] através de *Part Stress Analysis Prediction* (PASP).

Ao final foi possível realizar um quantitativo preciso de cada um dos tipos de componentes genéricos. Cada tipo de componente recebeu então uma taxa de falha atribuída utilizando o método de PSAP proposto em [5], Seção 3.3.

Deve-se ressaltar que a equação apresentada na Seção 3.3 para cálculo da taxa de falha via PSAP, é uma equação genérica, sendo que [5] indica variações específicas para cada tipo de componente. Foram nesta forma considerados os componentes em maior quantitativo e de maior relevância para a confiabilidade. Segue a análise de confiabilidade utilizando PSAP e relatórios de confiabilidade de fabricantes, na Tabela 5.2.4.1.

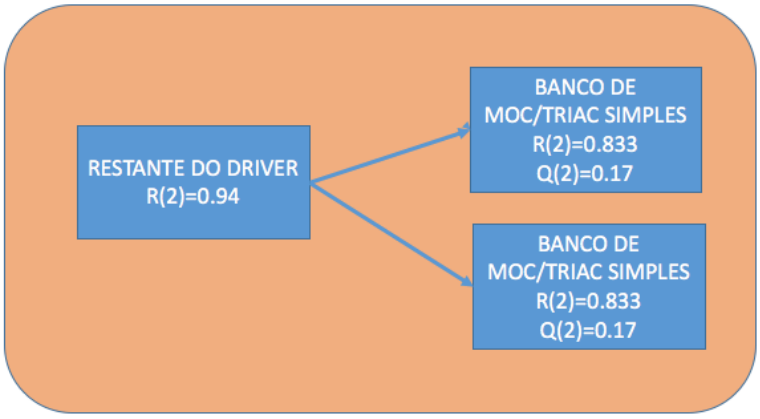
Table 5.2.4.1 - Análise de confiabilidade DRIVER

CONFIABILIDADE PCAP DRIVER	*Failure Rate base*	*Environment factor Gb*	*Environment factor Gf*	*Quality factor*	*Temperature factor 65 graus*	*Qtd.*	*Failure rate Gb PSAP*	*Failure rate Gf PSAP*	*Reliability Contribution Gb*	*Reliability Contribution Gf*
Dados genéricos para Part Stress Analysis MIL-HDBK-217F (Taxa de falha em falhas/10^6 horas)										
RESISTORES	0.0014	1	3	0.1	1	108	0.00014	0.00042	0.01512	0.04536
CAPACITORES	0.0021	1	2	3	1	72	0.0063	0.0126	0.4536	0.9072
CONECTORES DE POTÊNCIA	0.132	1	3	2.4	1	1	0.3168	0.9504	0.3168	0.9504
PCB DRIVER PTH	0.005248	1	3	2	1	1	0.010496	0.031488	0.010496	0.031488
CONECTORES DIGITAIS	0.0111	1	1	2.4	1	4	0.02664	0.02664	0.10656	0.10656
PCB CPU SMD	0.00552	1	2	20	3	1	0.3312	0.6624	0.3312	0.6624
DIODE ARRAY	0.003	1	6	2.4	3.3	3	0.02376	0.14256	0.07128	0.42768
CIRCUITO INTEGRADO 1-100 TRANSISTORES BIPOLAR	0.01	0.5	2	2	1.6	4	0.016	0.064	0.064	0.256
OPTAISOLADOR ANALOGICO	0.0055	1	2	0.7	2.7	12	0.010395	0.02079	0.12474	0.24948
MOV	0.0013	1	6	2.4	3.3	12	0.010296	0.061776	0.123552	0.741312
SUBTOTAL failure rate do controlador (Falhas / 10 ^6 horas)									1.305056	3.131088
SUBTOTAL A failue rate do controlador (Falhas / ano)									0.01143229	0.02742833
Dados de relatório de confiabilidade Texas Instruments inc. (Taxa de falha em falhas/ano)										
LM2596S-5.0	2.45653E-05	1	2	1	1	1	2.4565E-05	4.9131E-05	2.4565E-05	4.9131E-05
SN74HC158PW	2.41989E-05	1	2	1	1	7				
SN74HC157PW	2.84416E-06	1	2	1	1	1				
ISO7221CD	1.8758E-06	1	2	1	1	17	1.8758E-06	3.7516E-06	3.1889E-05	6.3777E-05
SN74HC125PWR	2.84416E-06	1	2	1	1	1	2.8442E-06	5.6883E-06	2.8442E-06	5.6883E-06
SN74AS08D	3.2809E-06	1	2	1	1	1	3.2809E-06	6.5618E-06	3.2809E-06	6.5618E-06
SN74LS32D	6.39416E-06	1	2	1	1	6	6.3942E-06	1.2788E-05	3.8365E-05	7.673E-05
SUBTOTAL B Total failure rate do controlador (Falhas / ano)									0.00010094	0.00020189
TAXA DE FALHA (FALHAS/ANO) DRIVER									0.01153323	0.02763022
TRIAC	0.0055	1	6	5.5	3.4	12	0.10285	0.6171	1.2342	7.4052
MOC	0.0022	1	6	5.5	3.4	12	0.04114	0.24684	0.49368	2.96208
Taxa de falha do Banco de TRIAC e MOC falhas/ano									0.02666946	0.09081737

FONTE: Produção do próprio autor

Pode-se ver que o DRIVER não apresentou a confiabilidade esperada, principalmente em razão do banco de TRIAC e dos MOC. O acionamento de potência de fato tem uma maior probabilidade de falha como previsto na PI-FMEA. Na Tabela 5.2.4.1, entretanto não foi considerada a quantidade utilizada para aplicar o sistema de redundância de TRIAC proposto na Seção 5.2.1. Isto se deve ao fato que ao quando um componente é acrescentado em redundância a sua falha não implica em risco de falha dos demais componentes do sistema.

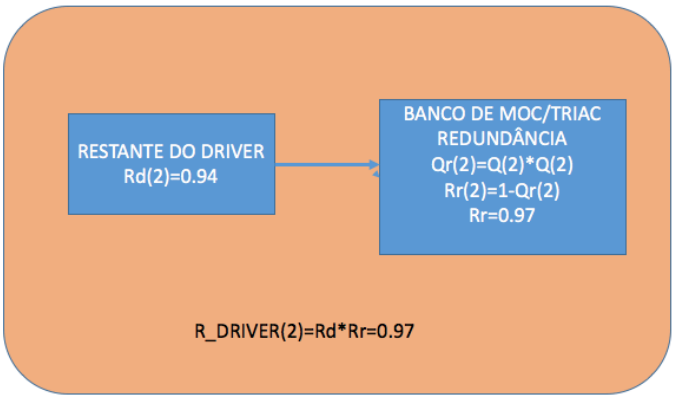
Figure 5.2.4.1 - Confiabilidade da redundância de TRIAC



FONTE: Produção do próprio autor

Embora haja 4 TRIACs acionando cada foco, considerou-se apenas a redundância de 2 TRIAC, como uma medida de segurança pois sabe-se que estatisticamente a chance da redundância em paralelo ser necessária é muito pequena embora permita maior capacidade de corrente, se precavendo de erros de instalação, por exemplo a instalações de mais semáforos do que permitido em paralelo. As Figuras 5.2.4.1 e 5.2.4.2 demonstram, para o pior caso, condição ambiental Gf, que devido a redundância, de TRIAC e MOC foi possível obter uma confiabilidade adequada, que deve ser analisada em conjunto com os demais subsistemas.

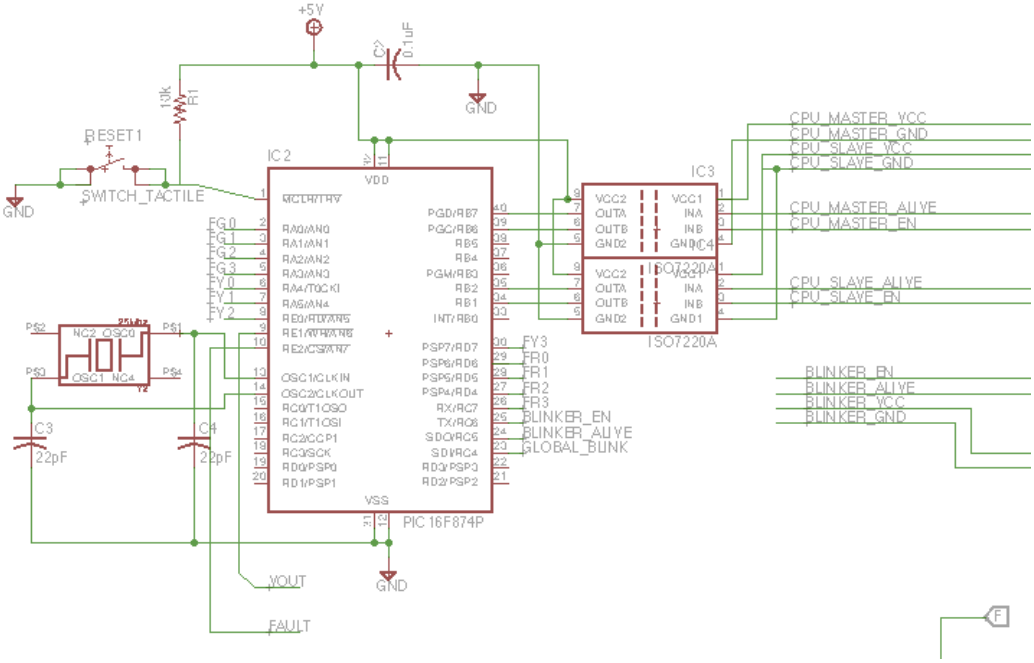
Figure 5.2.4.2 – Confiabilidade DRIVER



FONTE: Produção do próprio autor

5.3. Projeto Blinker

Figure 5.3.1 - Microcontrolador PIC



FONTE: Produção do próprio autor

O BLINKER é o mais simples dos subsistemas, porém ele é responsável também por supervisionar os demais subsistemas. Desta forma possui uma lógica combinacional que justifica um microcontrolador. Foi escolhido, portanto, o PIC16F874, por se tratar de um microcontrolador com arquitetura antiga, simples, e confiável, considerando que seu tempo de mercado garante taxas de falhas estatisticamente comprovadas. Ele também trabalha com níveis de tensão mais elevados em relação a microcontroladores mais modernos, o tornando menos suscetível a falhas, assim como surtos de alimentação e eletricidade estática.

Este subsistema lê todos os sinais indicativos de falha dos demais subsistemas, pois é sua função desabilitar as CPU em caso de falha para ativar o modo piscante. Ele também é responsável por ler os sinais indicadores de curto, rapidamente entrando em modo piscante neste caso. Um banco de TRIACS exatamente como implementado no DRIVER foi previsto, embora a estrutura de acionamento elétrico externa aos subsistemas (junto com o quadro de proteção) de relés/contatores não tenha sido projetada neste trabalho. Esta capacidade do BLINKER permite que em caso de falha do DRIVER, ou inexistência do mesmo, o modo piscante possa ser executado. Segue no Apêndice B o esquemático detalhado.

Tabela 5.3.1 - Análise confiabilidade do BLINKER

CONFIABILIDADE PCAP BLINKER	Failure Rate base (Failures/ 10^6 hours)	Environment factor Gb	Environment factor Gf	Quality factor	Temperature factor 65 graus	Qtd.	Failure rate Gb PSAP	Failure rate Gf PSAP	Reliability Contribution Gb	Reliability Contribution Gf
Dados genéricos para Part Stress Analysis MIL-HDBK-217F										
RESISTORES	0.0014	1	3	0.1	1	10	0.00014	0.00042	0.0014	0.0042
CAPACITORES	0.0021	1	2	3	1	12	0.0063	0.0126	0.0756	0.1512
CIRCUITOS INTEGRADOS MICROPROCESSADORES MOS	0.048	0.5	2	2.4	0.42	1	0.024192	0.096768	0.024192	0.096768
CONECTORES DIGITAIS	0.0111	1	1	2.4	1	1	0.02664	0.02664	0.02664	0.02664
PCB DRIVER PTH	0.005248	1	3	2	1	1	0.010496	0.031488	0.010496	0.031488
TRANSFORMADOR DE PULSO	0.003	1	6	5	1	1	0.015	0.09	0.015	0.09
SWITCH E PSUH BUTTONS	0.0027	1	3	1	1	5	0.0027	0.0081	0.0135	0.0405
OPTAISOLADOR ANALOGICO	0.0055	1	2	0.7	2.7	1	0.010395	0.02079	0.010395	0.02079
MOV	0.0013	1	6	2.4	3.3	1	0.010296	0.061776	0.010296	0.061776
SUBTOTAL failure rate do controlador (Falhas / 10 ^6 horas)									0.187519	0.523362
SUBTOTAL A failue rate do controlador (Falhas / ano)									0.001642666	0.004584651
Dados de relatório de confiabilidade Texas Instruments inc.										
LM2596S-5.0	2.45653E-05	1	2	1	1	1	2.4565E-05	4.9131E-05	2.45653E-05	4.91307E-05
ISO7221CD	1.8758E-06	1	2	1	1	8	1.8758E-06	3.7516E-06	1.50064E-05	3.00128E-05
SUBTOTAL B Total failure rate do controlador (Falhas / ano)									3.95718E-05	7.91435E-05
TAXA DE FALHA (FALHAS/ANO) BLINKER SEM MOC E TRIAC									0.001682238	0.004663795
CONECTORES DE POTÊNCIA	0.132	1	3	2.4	1	1	0.3168	0.9504	0.3168	0.9504
TRIAC	0.0055	1	6	5.5	3.4	1	0.10285	0.6171	0.10285	0.6171
MOC	0.0022	1	6	5.5	3.4	1	0.04114	0.24684	0.04114	0.24684
Taxa de falha Banco de TRIAC e MOC									0.00403652	0.015893618
TAXA DE FALHA (FALHAS/ANO) BLINKER COM MOC E TRIAC									0.005718759	0.020557413

FONTE: Produção do próprio autor

A Tabela 5.3.1 apresenta a análise de confiabilidade do Blinker, demonstrando que ele atendeu as especificações com relação à condição ambiental Gb, mas em função da taxa de falha do conjunto de MOC e TRIAC ele não se adequou à condição ambiental Gf. Entende-se que para operação neste tipo de condição, que não é o caso real, pode-se simplesmente não realizar a montagem destes componentes, não prejudicando os demais componentes do sistema.

5.4. Projeto das placas de circuito impresso

Os projetos das placas de circuito impresso estão presentes no Apêndice C. De modo geral, a PCB representa um grande ponto de falha, porém, com exceção de algumas diretivas com relação a requisitos de impedância em cada trilha, não há um modo sistemático aplicado ao roteamento das placas.

Houve grande dificuldade de roteamento da placa da CPU, tendo sido necessário a utilização de 5 camadas, sendo uma para distribuição exclusiva do Vcc. Devido a restrições de tempo, a solução de roteamento alcançada não confere a máxima confiabilidade ou eficiência, sendo válida para primeira versão de avaliação. Diretivas básicas foram seguidas para posicionamento dos conectores, antenas, e componentes mais críticos como o microcontrolador e osciladores. Seria necessário um roteamento manual detalhista para conferir para confiabilidade ao sistema.

O Blinker não conferiu grande dificuldade de roteamento tendo em vista que tem uma quantidade de componentes reduzida, podendo assim supor que a solução de roteamento atende perfeitamente bem aos requisitos.

Com relação ao projeto de PCB do DRIVER, a escolha de projeto referente a utilização de componentes PTH (*pin through hole*), por terem maior confiabilidade segundo [5], implicou em grande dificuldade de roteamento. Entretanto, devido a criticidade de confiabilidade esta placa teve o posicionamento de seus componentes pensado para que fosse possível um futuro projeto de dissipador de calor capaz de manter todos os TRIACs associados a um mesmo foco semafórico em equilíbrio térmico.

5.5. Predição de confiabilidade do controlador semafórico

Na Seção 4.1 foi definida a especificação de confiabilidade considerando dois modos de falha do controlador semafórico. FNR1 é a falha do sistema em executar corretamente um plano

de programação semafórica, correspondente ao funcionamento da CPU e DRIVER simultaneamente. A FNR2 é a falha na disponibilidade do BLINKER para execução do modo piscante. Considerando estas duas falhas, foram propostos os requisitos de confiabilidade apresentados a seguir.

$$P_{FNR1}(0 < t < 2\ anos) = 5\% \quad P_{FNR2}(0 < t < 5\ anos) = 5\%$$

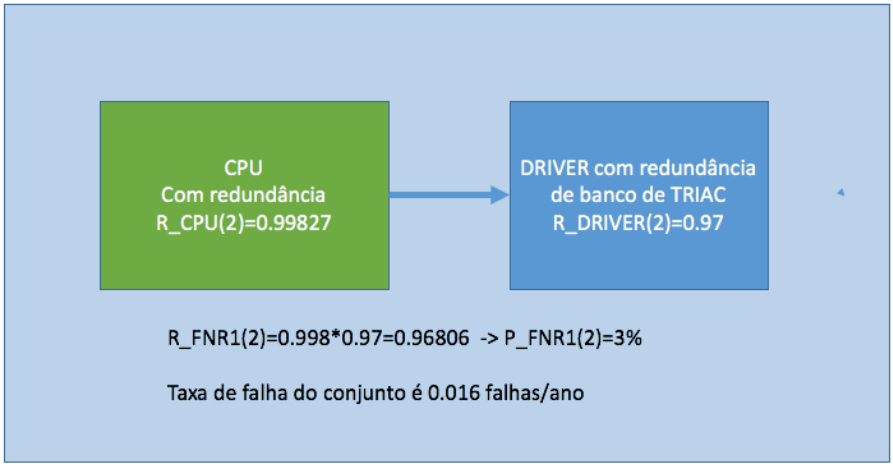
A confiabilidade propriamente dita (probabilidade que não aconteça uma falha em um determinado intervalo da vida útil do equipamento) é função da taxa de falhas, considerando para equipamentos eletrônicos uma distribuição exponencial da confiabilidade. É importante observar que a confiabilidade real do equipamento é representada pela FNR1, pois descreve o atendimento a todos os requisitos funcionais. Para o caso da CPU e BLINKER, no início, desconhecendo as características de cada subsistema, a confiabilidade foi alocada igualmente entre estes subsistemas.

Esta premissa teve que ser abandonada, alocando assim a maior parte da confiabilidade no subsistema CPU, que obteve melhores resultados de confiabilidade dentro da arquitetura escolhida. Esta realocação de confiabilidade é um processo natural para atendimentos das especificações e mitigar falhas onde houver maior possibilidade de tais.

Após a aplicação da sistemática de síntese para confiabilidade, obteve-se uma arquitetura com características que permitirem alcançar níveis de confiabilidade em conformidade com as especificações propostas.

A arquitetura do sistema foi pensada para que haja duas CPU idênticas, denominadas no decorrer do trabalho de CPU_MASTER e CPU_SLAVE, com o objetivo de reduzir a probabilidade de indisponibilidade dos requisitos funcionais do controlador semafórico. Foi obtido um resultado de confiabilidade teórica muito positiva.

Figure 5.5.1 – Confiabilidade do controlador



FONTE: Produção do próprio autor

A Figura 5.5.1 exemplifica o diagrama de blocos de confiabilidade final do controlador semafórico considerando os dois sistemas principais em série, a CPU com confiabilidade já calculada considerando redundância, e o DRIVER considerando o banco de TRIAC com redundância implementado.

Estes dados permitem afirmar que as especificações foram suficientemente atendidas, pois os dados apresentados na Figura 5.5.1 são referentes as condições ambientais mais severas, Gf, previsto em [5]. Para o caso Gb, que é a classificação menos severa de condição ambiental, as taxas de falhas apresentadas neste capítulo, atingiram confortavelmente as metas de confiabilidade.

O subsistema BLINKER, obteve um resultado adequado para a condição Gb, porém abaixo da especificação de confiabilidade inicialmente requerida. Cabe, portanto, uma readequação de projeto, que não poderá ser feita em função das limitações de tempo para este projeto. Já o subsistema CPU e DRIVER, vinculados a FNR1, apresentaram confiabilidade teórica acima da requerida, possibilitando afirma que o controlador semafórico tem alta confiabilidade mesmo para condições de ambientais padrão Gf.

Os resultados obtidos para o pior caso (condição Gf) :

$$P_{FNR2}(0 < t < 2\ anos) = 4\% \quad P_{FNR2}(0 < t < 5\ anos) = 10\%$$

$$P_{FNR1}(0 < t < 2\ anos) = 3\% \quad P_{FNR1}(0 < t < 5\ anos) = 8\%$$

Realizando o mesmo raciocínio do cálculo para o pior caso, apresentado em detalhes anteriormente (condição Gb). Os resultados obtidos para o melhor caso (condição Gb) foram:

$$P_{FNR1}(0 < t < 2\ anos) = 3\ \% \quad P_{FNR1}(0 < t < 5\ anos) = 7\%$$

$$P_{FNR2}(0 < t < 2\ anos) = 2\% \quad P_{FNR2}(0 < t < 5\ anos) = 3\%$$

CONCLUSÕES E TRABALHOS FUTUROS

A sistemática para síntese de sistema eletrônico de alta confiabilidade pôde provar sua utilidade ao ser aplicada ao controlador semafórico, entretanto foi possível detalhar as grandes dificuldades de um projeto com esta concepção.

Percebeu-se que os métodos de análise de confiabilidade realmente não foram originalmente propostos na literatura como instrumentos de síntese. Os métodos de análise apresentam eficiência para entender e prever a alocação de confiabilidade de um determinado circuito, mas não oferecem mecanismos sistemáticos para modificação de projeto. A grande quantidade de critérios subjetivos utilizados, revela a importância de equipes independentes analisando paralelamente.

Análises como FMEA são cruciais para guiar as decisões de projeto em prol da confiabilidade. Além dos métodos encontrados na literatura, o exemplo do controlador semafórico tornou clara a importância da escolha sistemática da arquitetura do sistema. No caso do controlador semafórico, a avaliação de hipóteses de arquiteturas propostas, garantiu o desenvolvimento do projeto com características que permitiram atingir a confiabilidade teórica exigida.

A confiabilidade teórica calculada através dos métodos propostos, é posta em dúvida em relação a grande dificuldade em adquirir dados empíricos reais de componentes. O contraste entre parâmetros de confiabilidade fornecidos por fabricantes e parâmetros teóricos genéricos encontrados na literatura reafirmam essa visão. A utilização de parâmetros de confiabilidade genéricos pode fazer

com que a complexidade do circuito cresça demasiadamente para adequação de confiabilidade. Por outro lado, considerar componentes com alto padrão de qualidade e parâmetros de confiabilidade comprovados estatisticamente, no geral, implica em projeto de circuito mais simplificado.

Projetar um sistema eletrônico simples é sempre mais aconselhável. Desta forma há redução de riscos em todas as etapas de projeto e de produção. A simplificação do sistema permite mitigar o risco que erros de projeto sejam realizados durante etapas de fabricação. Este ponto não foi atendido neste projeto, pois as restrições de disponibilidade de dados definiram escolhas de projeto que determinaram um aumento da quantidade de componentes.

Componentes com qualidade, entretanto, precisam ser aplicados de forma correta, com uma concepção de sistema adequada, explicitando a importância da separação do raciocínio de projeto nos três eixos de síntese propostos na Seção 3.1. O eixo de arquitetura, o eixo de projeto e o eixo de tecnologia, foram analisados prioritariamente separados durante a síntese proposta. Esta abordagem é necessária, pois há grande perda de eficiência no desenvolvimento de um equipamento tentando integrar estes três eixos. É claro, também, que aplicações de alta confiabilidade requer um esforço cíclico de análise e projeto, havendo forte interação entre os eixos na pratica.

Portanto, foi possível definir métodos de análise e avaliação novos, juntamente com técnicas de análise recorrente na literatura, para determinar uma sistemática para desenvolvimento de sistemas eletrônicos de alta confiabilidade. Tal como exposto na Seção 3.6.

Utilizando a sistemática proposta, foi possível projetar uma primeira versão de um controlador semafórico de alta confiabilidade. Em trabalhos futuros é essencial uma revisão completa do projeto tentando simplificar os subsistemas. Isto é importante porque, embora a confiabilidade teórica tenha sido alcançada, há uma grande complexidade de fabricação, podendo introduzir pontos de falha no processo. O projeto das placas de circuito também deve ser refeito, principalmente a placa da CPU, pois embora tenha sido desenvolvida uma primeira versão, é possível que ela seja projetada de maneira mais eficiente confiável.

Mesmo que a sistemática tenha se adequado bem ao projeto de hardware, aplicações de alta confiabilidade tem grandes desafios de software. Não há sistemática definida para implementação de software a prova de falhas. Neste caso, as sugestões de trabalhos futuros são, definir uma sistemática para projeto de software e implementa-lo para o controlador semafórico utilizando técnicas e recursos a prova de falhas. Os recursos previstos pela PI-FMEA e grande parte da mitigação de falha avaliada para o controlador semafórico, requer a implementação de software de tempo real de alta confiabilidade, não tornando assim o software em si, um dos maiores pontos de falha do sistema

ÂPENDICE A – FAILURE MODE AND EFFECT ANALYSIS

Failure Mode and Effect Analysis (PI-FMEA)

Sistema: **Controlador Semafórico Arquitetura 3** | Blocos: **N/A** | **Número da FMEA:** 1

Subsistemas em Análise : **CPU, BLINKER, DRIVER**

Componentes: **N/A**

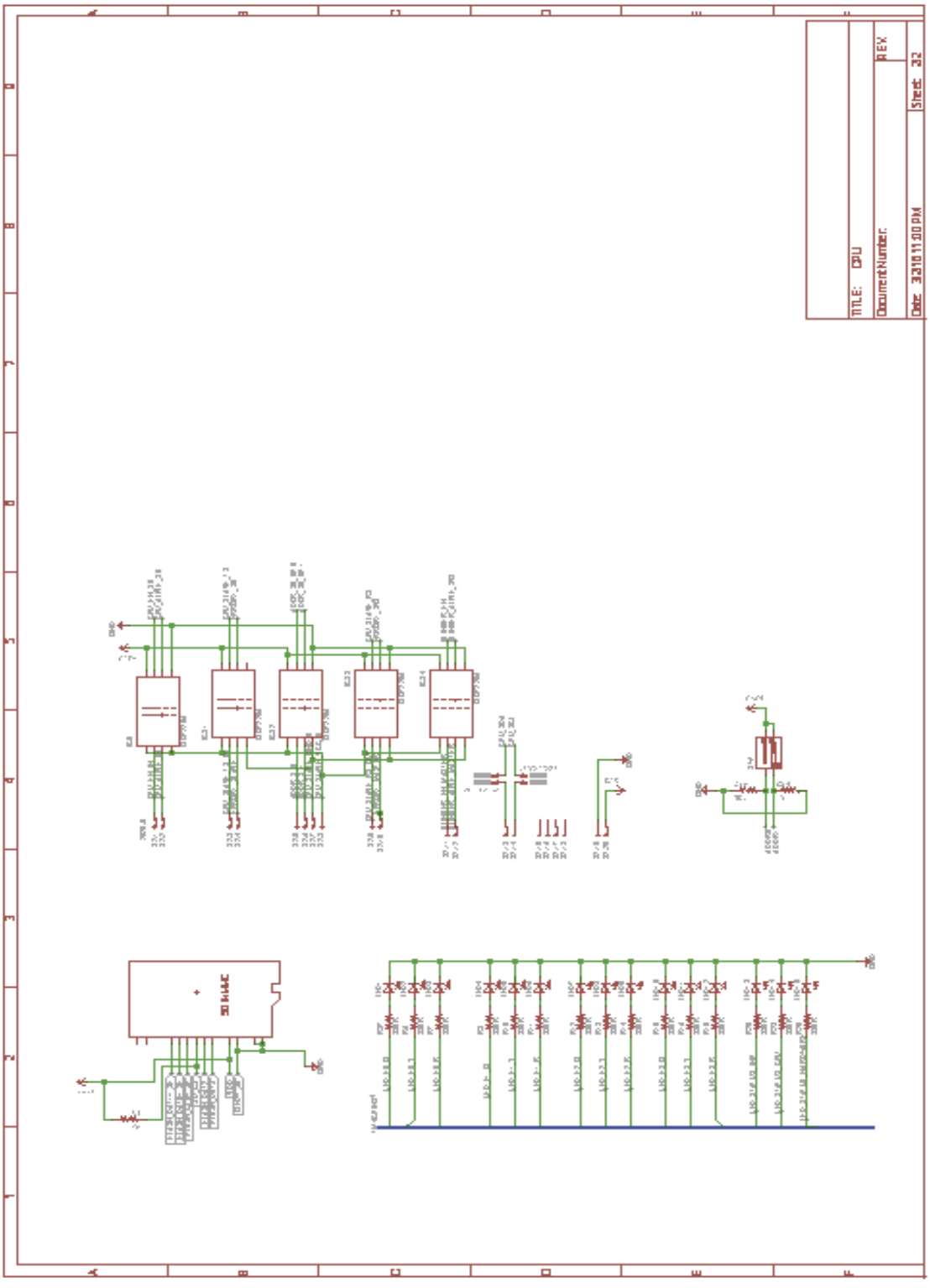
Item	Função do Item/ Funcão em falha	Failure Mode Potential	Efeitos Potências da falha	SEV	CLASS	Causas Potências/Mecanismo da Falha	OCCUR	DETEC	RPN	Ação Recomendada	Resultados de Ação				
											Medida de Mitigação	SEV	OCCUR	DETEC	RPN
CPU	Manter a alimentação em níveis adequados para própria operação	Falha na fonte de alimentação	Não acionamento lógico de nenhum GS	6	**FHNR**	Surtos de tensão e corrente/ sobrecarga/ sobretemperatura	2	2	24	Outra unidade assumir o acionamento lógico	Criar redundância de alimentação através de bateria e isolar fonte de alimentação da CPU dos demais módulos	6	2	2	24
		Falha no circuito de regulação da CPU	Não acionamento lógico de nenhum GS e falha não reparável da CPU	4	**FHNR**	Sobre temperatura/sobre tensão/sobrecorrente/ curto na PCB	4	4	64	Outra unidade assumir o acionamento lógico	Aumentar confiabilidade do circuito de regulação ou realizar redundância da CPU, isolar fonte de alimentação da CPU dos demais módulos	4	4	4	64
		Falha no conector	Não acionamento lógico de nenhum GS	4	**FHR**	Mal contato/ conector danificado/conector aberto	6	2	48	Outra unidade assumir o acionamento lógico	Criar redundância de conector de alimentação, isolar fonte de alimentação da CPU dos demais módulos	4	6	2	48
CPU	Manter os verdes sem conflito	Falha de execução do software da CPU	Risco de vida e falha geral da CPU	9	**FSR**	Falha lógica do software/firmware implementado ou falha intrínseca do microcontrolador	8	3	216	Outra unidade assumir o acionamento lógico e reiniciar a CPU com falha de software	Criar WDT em hardware para reiniciar sistema e realocar controle sobre o acionamento lógico	9	8	3	216
		Plano de programação gravado inconsistente	Risco de vida	10	**FSNR**	Operador não competente ou falha de comunicação	5	4	200	CPU deve tentar entrar em modo piscante	Realizar controle de consistência da programação no momento de gravação do plano assim como checar consistência do plano em hardware para entrar em modo piscante antes de efetivada a falha	10	5	4	200
		Falha no conector	Risco de vida e falha geral da CPU	9	**FHNR**	Mal contato/ conector danificado/conector aberto	6	3	162	Outra unidade deve assumir o controle do acionamento lógico	Selecionar conector com alta confiabilidade , travas, relizar monitoramento periódica do conector, blindar sinais no conector, usar conector redundante e enviar pares conjugados de sinais	9	6	3	162
		Curto entre focos verdes	Risco de vida e falha geral do controlador	10	**FHNR**	Curto entre sinais digitais de controle verde, ou curto nos conectores dos GS	4	4	160	CPU deve tentar colocar verdes em conflito em amarelo piscante	Prever proteções contra curtos em hardware e monitoramento de software conínuo	10	4	4	160
		Trigger não intencional do DRIVER	Risco de vida e falha geral ou parcial do DRIVER	9	**FHR**	Trigger não intecional de fase	3	3	81	CPU deve reeviar sinais de controle corretos e checar realizando leitura das correntes	Aumentar confiabilidade do DRIVER	9	3	3	81
		Trigger não intencional dos circuitos lógicos de acionamento	Risco de vida e falha geral ou parcial da CPU	9	**FHR**	Trigger não intecional de circuitos lógicos devido a ruído e má interpretação de bit, ou falha total do acionamento lógico	4	3	108	CPU deve reeviar sinais de controle corretos e checar realizando leitura das correntes	Redundância na detecção de verdes conflitantes	9	4	3	108
		Curto nos circuitos lógicos de acionamento	Risco de vida e falha geral ou parcial do CPU	9	**FHR**	Falha total de algum circuito lógico	4	5	180	CPU deve reeviar sinais de controle corretos, foco a foco e checar realizando leitura das correntes para identificar em qual foco tem problema e manter este GS em amarelo piscante	Permitir leitura dos sinais acionados	9	4	5	180
		Falha generalizada na PCB	Risco de vida e falha geral ou parcial do CPU	10	**FHNR**	Sobre temperatura e/ou outras condições climáticas	3	4	120	Outra unidade assumir o acionamento lógico e reiniciar a CPU com falha de software	Implementar redundância de CPU	10	3	4	120

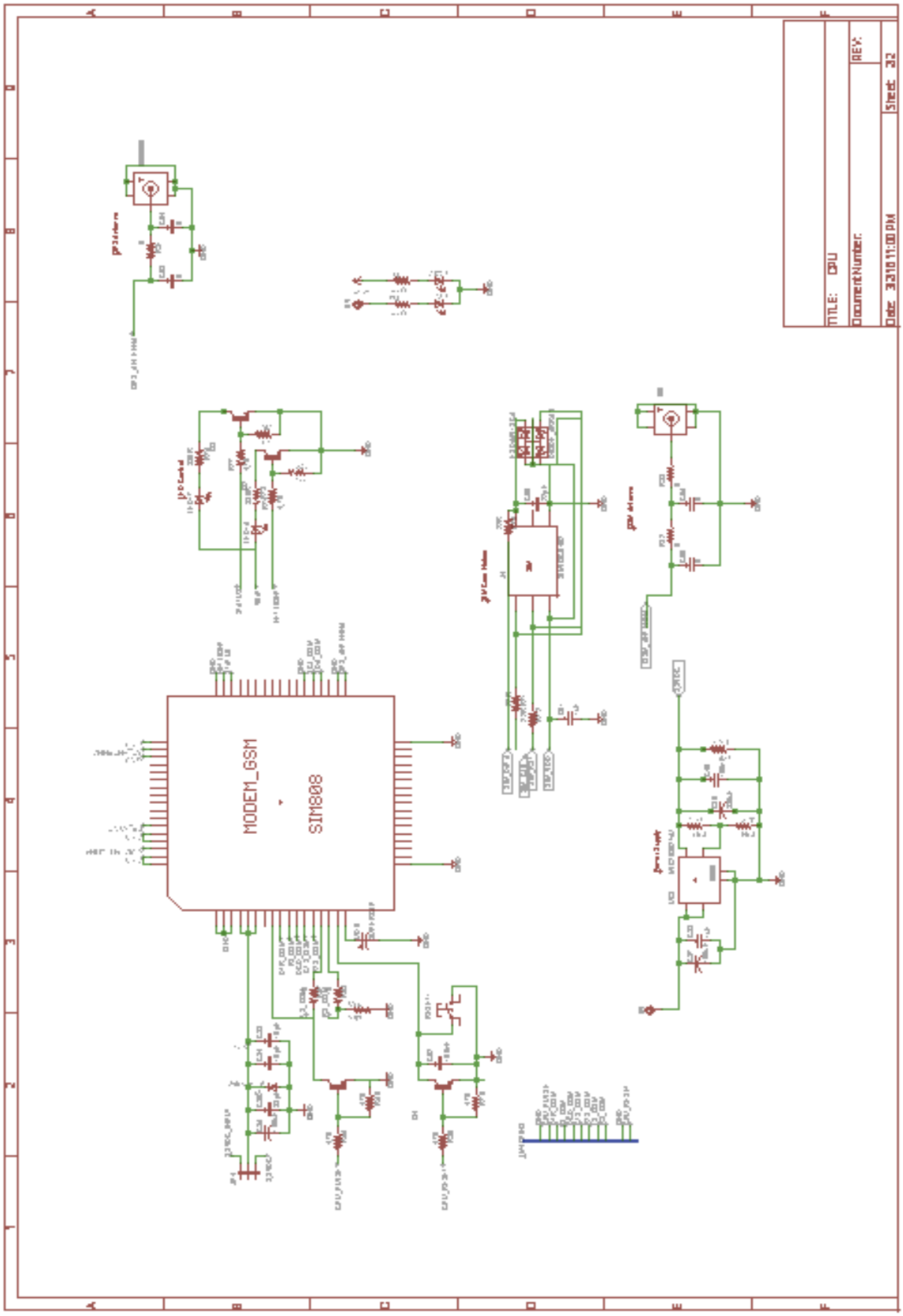
CPU	Acionar plano de programação semafórica conforme programado	Travamento do software da CPU	Insatisfação do cliente, falha geral da CPU	6	**FSR**	Falha lógica do software/firmware implementado ou falha intríseca do microcontrolador	8	3	144	Outra unidade assumir o acionamento lógico e reiniciar a CPU com falha de software	Criar WDT em hardware para reiniciar sistema e realocar controle sobre o acionamento lógico	6	8	3	144
		Plano de programação gravado inconsistente	Insatisfação do usuário, inoperância do controlador	6	**FSNR**	Operador não competente ou falha de comunicação	5	4	120	CPU deve tentar entrar em modo piscante	Realizar controle de consistência da programação no momento de gravação do plano assim como checar consistência do plano em hardware para entrar em modo piscante antes de efetivada a falha	6	5	4	120
		Falha de consistência de memória não volatil	Insatisfação do cliente, Impossibilidade de seguir plano de programação prê gravado	5	**FHR**	Sobre temperatura na memória, falha do sistema de arquivos e/ou escrita de memória por parte do microcontrolador	4	5	100	CPU deve manter o plano de programação semafórico que estiver executando no momento	Redundância de memória não volatil e backuo em RAM do conteúdo da memória não volatil para monitoramento periodico de consistência	5	4	5	100
		Perda de horário e data	Insatisfação do cliente, Impossibilidade de mudar o plano semafórico que estiver sendo executado para o próximo plano agendado	6	**FHR**	Falha do GPS , perda de conectividade durante muito tempo ou falha de comunicação	5	2	60	CPU deve manter data e hora no seu RTC interno	Redudância em hardware e software para manutenção de data e hora	6	5	2	60
		Falha no Conector	Risco de vida e falha geral da CPU	5	**FHNR**	Mal contato/ conector danificado/conector aberto	6	3	90	Outra unidade deve assumir o controle do acionamento lógico	Selecionar conector com alta confiabilidade , travas, relizar monitoramento periódica do conector, blindar sinais no conector, usar conector redundante e enviar pares conjugados de sinais	5	6	3	90
		Trigger não intencional do DRIVER	Risco de vida e falha geral ou parcial do DRIVER	5	**FHR**	Trigger não intecional de fase	3	3	45	CPU deve reeviar sinais de controle corretos e checar realizando leitura das correntes	Aumentar confiabilidade do DRIVER	5	3	3	45
		Trigger não intencional dos circuitos lógicos de acionamento	Risco de vida e falha geral ou parcial da CPU	5	**FHR**	Trigger não intecional de circuitos lógicos devido a ruído e má interpretação de bit, ou falha total do acionamento lógico	4	3	60	CPU deve reeviar sinais de controle corretos e checar realizando leitura das correntes	Redundância na detecção de verdes conflitantes	5	4	3	60
		Curto nos circuitos lógicos de acionamento	Risco de vida e falha geral ou parcial do CPU	6	**FHR**	Falha total de algum circuito lógico	4	5	120	CPU deve reeviar sinais de controle corretos, foco a foco e checar realizando leitura das correntes para identificar em qual foco tem problema e manter este GS em amarelo piscante	Permitir leitura dos sinais acionados	6	4	5	120
		Falha generalizada na PCB	Risco de vida e falha geral ou parcial do CPU	10	**FHNR**	Sobre temperatura e/ou outras condições climáticas	3	4	120	Outra unidade assumir o acionamento lógico e reiniciar a CPU com falha de software	Implementar redundância de CPU	10	3	4	120
CPU	Manter conectividade para monitoramento de falha e determinação de planos de programção	Falha de conector ethernet	Insatisfação do cliente, perda de confiança na conectividade	5	**FHNR**	Mal contato/ conector danificado/conector aberto	2	3	30	Iniciar conexão atraves de outra forma de enlace	Implementar redundância na camada de enlace	5	2	3	30
		Falha de modem sem fio	Insatisfação do cliente, perda de confiança na conectividade	5	**FHNR**	Falha de comunicação ou perda do modem devido a sobretemperatura ou alimentação	5	2	50	Iniciar conexão atraves de outra forma de enlace	Implementar redundância na camada de enlace	5	5	2	50
		Perda de sinal de rede durante curtos períodos	Pequena Insatisfação do cliente	3	**FHR**	Antena inapropriada ou falha do provedor de internet	6	2	36	Tentar reconectar sempre que possível	Tentar reconectar periodicamente checando força do sinal recebido para identificar problemas do provedor, tentar redundâncias com provedores diferentes	3	6	2	36
		Perda de sinal de rede durante longos períodos	Grande insatisfação do cliente	7	**FHR**	Queda de todos os elances	4	3	84	Tentar reconectar sempre que possível	Tentar reconectar periodicamente checando força do sinal recebido para identificar problemas do provedor, tentar redundâncias com provedores diferentes e enlaces diferentes	7	4	3	84
		Falha generalizada na PCB	Grande Insatisfação do cliente e inoperância do sistema todo	10	**FHNR**	Sobre temperatura e/ou outras condições climáticas	3	4	120	Outra unidade assumir o acionamento lógico e reiniciar a CPU com falha de software	Implementar redundância de CPU	10	3	4	120
	Manter interface de comunicação local para gravação de plano de programação	Falha no Conector	Pequena insatisfação do cliente ou outra falha na CPU	4	**FHR**	Mal contato/ conector danificado/conector aberto	3	2	24	Deve ser feita conexão com a CPU remotamente e/ou recolocar o conector com problema	Selecionar conector com alta confiabilidade , travas, relizar monitoramento periódica do conector, blindar sinais no conector, usar conector redundante e enviar pares conjugados de sinais	4	3	2	24
		Falha de comunicação	Pequena insatisfação do cliente ou outra falha na CPU	3	**FSR**	Erro de comunicação de software	4	5	60	Pedir para que o pacote de dados seja reenviado	Implementar protocolo com confiabilidade e checagem de erro assim como utilziar baixas taxas de transmissão	3	4	5	60
CPU	Comunicação com CPU redundante	Falha no conector	Pequena insatisfação do cliente por má interpretação do status da CPU e possível erro de acionamento.	4	**FHR**	Mal contato/ conector danificado/conector aberto	3	4	48	O blinker deve avaliar a consistência dos sinais	Selecionar conector com alta confiabilidade , travas, relizar monitoramento periódica do conector, blindar sinais no conector, usar conector redundante e enviar pares conjugados de sinais	4	3	4	48
		Sinais inconsistentes	Futura ou corrente falha de acionamento da CPU redudante que apresente sinais de comunicação inconsistente	5	**FHR**	Erro de software ou falha de hardware	2	4	40	CPU ccom indicação de pleno funcionamento deve forçar a outra a ceder o controle	Deixar sinais sempre com nível lógico bme definido e padrões de comportamento de facil conferência para identificar rapidamente sinais que indiquem não funcionamento	5	2	4	40
	Comunicação com BLINKER	Falha no conector	Pequena insatisfação do cliente por má interpretação do status da CPU e possível erro de acionamento	4	**FHR**	Mal contato/ conector danificado/conector aberto	3	4	48	A CPU deve manter o controle	Selecionar conector com alta confiabilidade , travas, relizar monitoramento periódica do conector, blindar sinais no conector, usar conector redundante e enviar pares conjugados de sinais	4	3	4	48
		Sinais inconsistentes	Futura ou corrente falha de acionamento do Blinker que apresente sinais de comunicação inconsistente	5	**FHR**	Falha de software ou hardware do blinker	2	4	40	A CPU deve manter o controle	Deixar sinais sempre com nível lógico bme definido e padrões de comportamento de facil conferência para identificar rapidamente sinais que indiquem não funcionamento	5	2	4	40

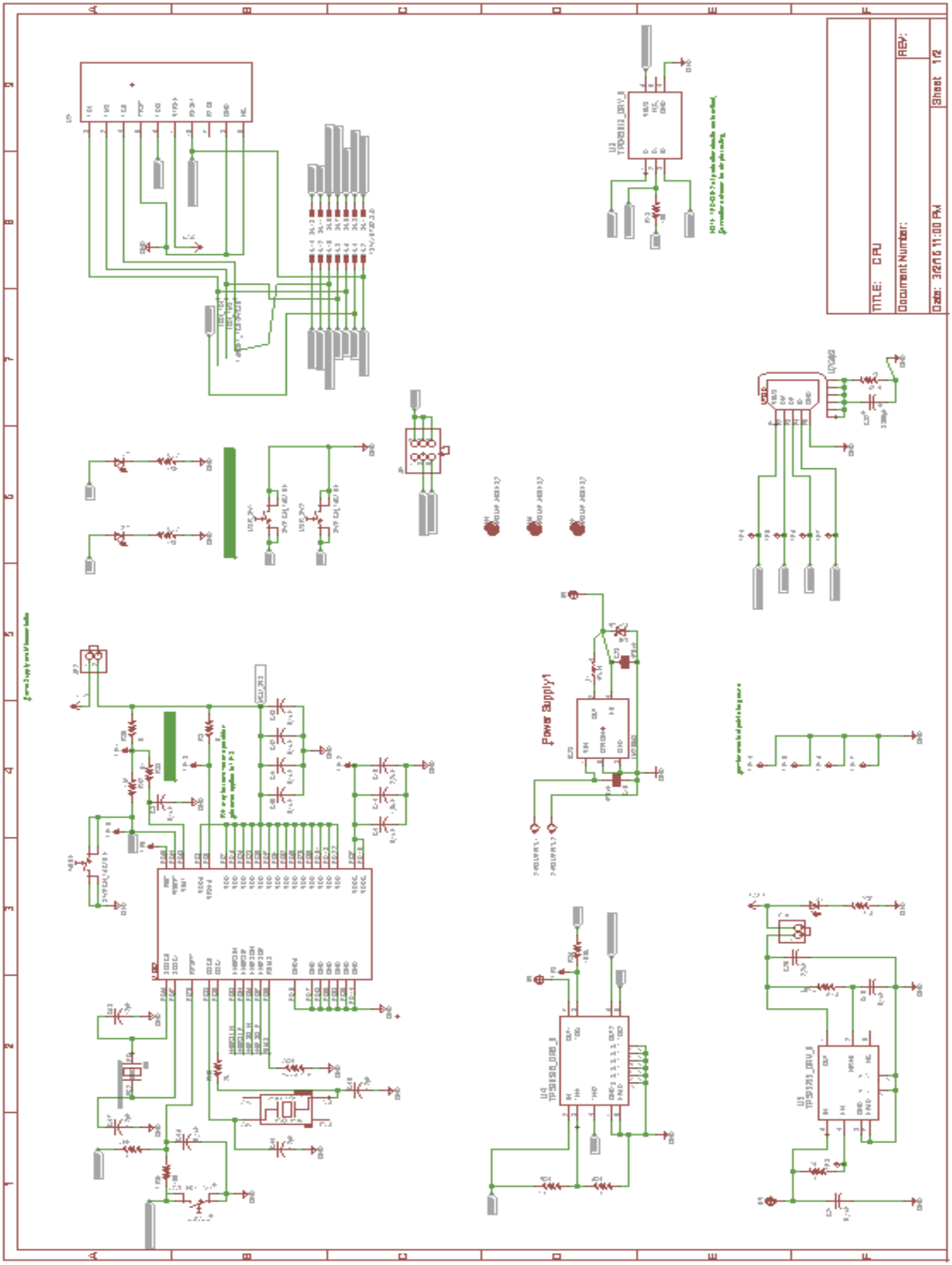
DRIVER	Acionamento de uma fase	Falha do conector CPU/DRIVER	insatisfação do cliente, não acionamento da fase, risco de vida	5	FHNR	Mal contato/ conector danificado/conector aberto	6	3	90	Manter acionamento default adequado em caso de falha	Selecionar conector com alta confiabilidade , travas, relizar monitoramento periódica do conector, blindar sinais no conector, usar conector redundante e enviar pares conjugados de sinais	5	6	3	90
		Falha de seleção entre unidade de controle lógica (VOTER)	Seleção de unidade lógico errada para acionamento, podem impro risco de vida	5	FHNR	Curto nos compoentes do voter devido a sobretemperatura	4	4	80	Manter acionamento default adequado em caso de falha	Aumentar a confibildiade do VOTER assim como manter sinais de controle default adequados para este modo de falha	5	4	4	80
		Trigger não intencional dos circuitos lógicos de acionamento	Risco de vida e falha geral ou parcial da CPU	9	FHR	Trigger não intecional de circuitos lógicos devido a ruído e má interpretação de bit, ou falha total do acionamento lógico	4	3	108	CPU deve reeviar sinais de controle corretos e checar realizando leitura das correntes	Redundância na detecção de verdes conflitantes	9	4	3	108
		Curto na saída da fase	Não acionamento adequado da fase, implicando em risco de vida	8	FHNR	Corrente nominal ou levemente superior durante logos perídos no componente de acionamento, ou acidentes de instalção do controlador e grupos semafóricos	7	2	112	CPU deve ler os focos semafóricos para localizar curtos e informar rapidamente o operador do sistema	Aplicar redundância de acionamento para manter operação em caso de curto em um dos componentes de acionamento, assim como implementar medidas de proteção contra curto	8	7	2	112
		Fase em aberto	Não acionamento adequado da fase, implicando em risco de vida	8	FHNR	Rompimento de cabo ou falha em aberto de componente de acionamento	3	2	48	CPU deve ler os focos semafóricos para localizar os abertos e informar rapidamente o operador do sistema	Aplicar redundância de acionamento para manter operação em caso de aberto em um dos componentes de acionamento, assim como implementar medidas de proteção contra curto	8	3	2	48
		Falha de trigger de componentes de acionamento	Não acionamento adequado da fase, implicando em risco de vida	7	FNHR	Manutenção do sinal de controle por pouco tempo ou falha ocasional de trigger no acionamento	3	2	42	CPU deve reeviar sinais de controle corretos e checar realizando leitura das correntes	Redundância na detecção de verdes conflitantes	7	3	2	42
	Acionamento de todas as fases	Falha de conectores	insatisfação do cliente, não acionamento da fase, risco de vida	8	FNHR	Mal contato/ conector danificado/conector aberto	6	3	144	Manter acionamento default adequado em caso de falha	Selecionar conector com alta confiabilidade , travas, relizar monitoramento periódica do conector, blindar sinais no conector, usar conector redundante e enviar pares conjugados de sinais	8	6	3	144
		Falha de seleção entre unidade de controle lógica (VOTER)	Seleção de unidade lógico errada para acionamento, podem impro risco de vida	9	FHNR	Curto nos compoentes do voter devido a sobretemperatura	4	4	144	Manter acionamento default adequado em caso de falha	Aumentar a confibildiade do VOTER assim como manter sinais de controle default adequados para este modo de falha	9	4	4	144
		Falha generalizada na PCB	Grande Insatisfação do cliente e inoperância do sistema todo	10	FHNR	Sobre temperatura e/ou outras condições climáticas	3	4	120	CPU deve detectar tal falha para através de leitura dos focos semafóricos para informar operador da falha da PCB Driver	Aumentar confiabilidade da PCB através de projeto de dissipação termica correto, mantendo ela simples e com características de projeto seguindo princípios de confibilidade	10	3	4	120
	Manter a alimentação em níveis adequados para própria operação	Falha na fonte de alimentação	Não acionamento lógico de nenhum GS	6	FHNR	Surtos de tensão e corrente/ sobrecarga/ sobretemperatura	2	2	24	Outra unidade assumir o acionamento lógico	Criar redundância de alimentação através de bateria e isolar fonte de alimentação da CPU dos demais módulos	6	2	2	24
		Falha no circuito de regulação da CPU	Não acionamento lógico de nenhum GS e falha não reparável da CPU	4	FHNR	Sobre temperatura/sobre tensão/sobrecorrente/ curto na PCB	4	4	64	Outra unidade assumir o acionamento lógico	Aumentar confiabilidade do circuito de regulação ou realizar redundância da CPU, isolar fonte de alimentação da CPU dos demais módulos	4	4	4	64
		Falha no conector	Não acionamento lógico de nenhum GS	4	FHR	Mal contato/ conector danificado/conector aberto	6	2	48	Outra unidade assumir o acionamento lógico	Criar redundância de conector de alimentação, isolar fonte de alimentação da CPU dos demais módulos	4	6	2	48

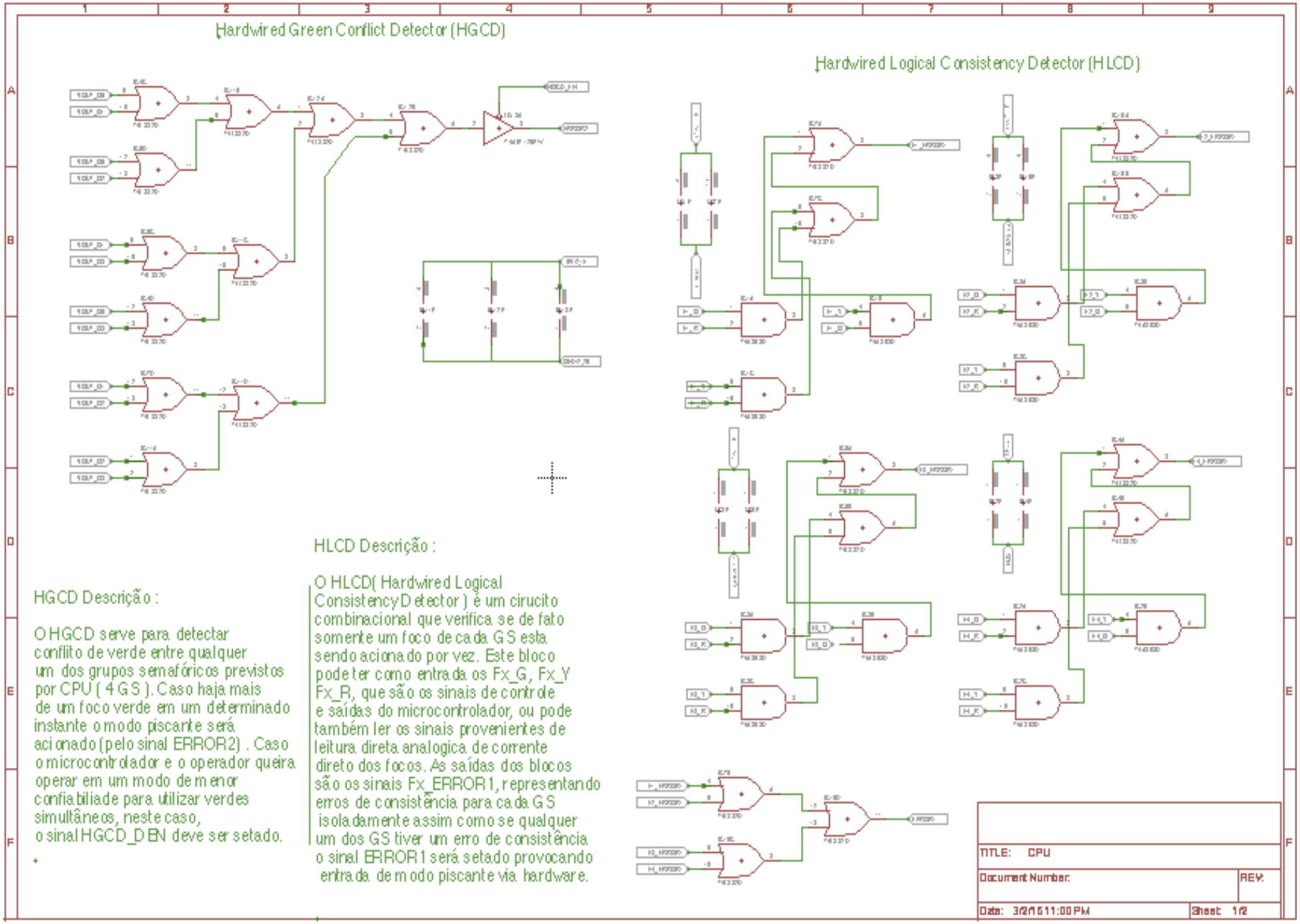
BLINKER	Manter a alimentação em níveis adequados para própria operação	Falha na fonte de alimentação	Não acionamento lógico de nenhum GS	6	FHNR	Surtos de tensão e corrente/ sobrecarga/ sobretemperatura	2	2	24	Outra unidade assumir o acionamento lógico	Criar redundância de alimentação através de bateria e isolar fonte de alimentação da CPU dos demais módulos	6	2	2	24
		Falha no circuito de regulação da CPU	Não acionamento lógico de nenhum GS e falha não reparável da CPU	4	FHNR	Sobre temperatura/sobre tensão/sobrecorrente/ curto na PCB	4	4	64	Outra unidade assumir o acionamento lógico	Aumentar confiabilidade do circuito de regulação ou realizar redundância da CPU, isolar fonte de alimentação da CPU dos demais módulos	4	4	4	64
		Falha no conector	Não acionamento lógico de nenhum GS	4	FHR	Mal contato/ conector danificado/conector aberto	6	2	48	Outra unidade assumir o acionamento lógico	Criar redundância de conector de alimentação, isolar fonte de alimentação da CPU dos demais módulos	4	6	2	48
	Comunicação com CPU redundante	Falha no conector	Pequena insatisfação do cliente por má interpretação do status da CPU e possível erro de acionamento.	4	FHR	Mal contato/ conector danificado/conector aberto	3	4	48	O blinker deve avaliar a consistência dos sinais	Selecionar conector com alta confiabilidade , travas, relizar monitoramento periódica do conector, blindar sinais no conector, usar conector redundante e enviar pares conjugados de sinais	4	3	4	48
		Sinais inconsistentes	Futura ou corrente falha de acionamento da CPU redudante que apresente sinais de comunicação inconsistente	5	FHR	Erro de software ou falha de hardware	2	4	40	CPU ccom indicação de pleno funcionamento deve forçar a outra a ceder o controle	Deixar sinais sempre com nível lógico bme definido e padrões de comportamento de facil conferência para identificar rapidamente sinais que indiquem não funcionamento	5	2	4	40
	Acionamento de fase	Falha do conector CPU/DRIVER	insatisfação do cliente, não acionamento da fase, risco de vida	5	FHNR	Mal contato/ conector danificado/conector aberto	5	3	75	Manter acionamento default adequado em caso de falha	Selecionar conector com alta confiabilidade , travas, relizar monitoramento periódica do conector, blindar sinais no conector, usar conector redundante e enviar pares conjugados de sinais	5	5	3	75
		Falha na leitura de falha da CPU	Seleção de unidade lógico errada para acionamento, podem impor risco de vida	6	FHNR	Curto nos compoentes do voter devido a sobretemperatura	4	4	96	Manter acionamento default adequado em caso de falha	Aumentar a confibildiade do VOTER assim como manter sinais de controle default adequados para este modo de falha	6	4	4	96
		Trigger não intencional dos circuitos lógicos de acionamento	Risco de vida e falha geral ou parcial da CPU	5	FHR	Trigger não intecional de circuitos lógicos devido a ruído e má interpretação de bit, ou falha total do acionamento lógico	3	3	45	CPU deve reeviar sinais de controle corretos e checar realizando leitura das correntes	Redundância na detecção de verdes conflitantes	5	3	3	45
		Curto na saída da fase	Não acionamento adequado da fase, implicando em risco de vida	4	FHNR	Corrente nominal ou levemente superior durante logos perídos no componente de acionamento, ou acidentes de instalção do controlador e grupos semafóricos	4	2	32	CPU deve ler os focos semafóricos para localizar curtos e informar rapidamente o operador do sistema	Aplicar redundância de acionamento para manter operação em caso de curto em um dos componentes de acionamento, assim como implementar medidas de proteção contra curto	4	4	2	32
		Fase em aberto	Não acionamento adequado da fase, implicando em risco de vida	4	FHNR	Rompimento de cabo ou falha em aberto de componente de acionamento	3	2	24	CPU deve ler os focos semafóricos para localizar os abertos e informar rapidamente o operador do sistema	Aplicar redundância de acionamento para manter operação em caso de aberto em um dos componentes de acionamento, assim como implementar medidas de proteção contra curto	4	3	2	24
		Falha de trigger de componentes de acionamento	Não acionamento adequado da fase, implicando em risco de vida	4	FNHR	Manutenção do sinal de controle por pouco tempo ou falha ocasional de trigger no acionamento	3	2	24	CPU deve reeviar sinais de controle corretos e checar realizando leitura das correntes	Redundância na detecção de verdes conflitantes	4	3	2	24
		Falha generalizada na PCB	Grande Insatisfação do cliente e inoperância do sistema todo	10	FHNR	Sobre temperatura e/ou outras condições climáticas	3	4	120	CPU deve detectar tal falha para através de leitura dos focos semafóricos para informar operador da falha da PCB Driver	Aumentar confiabilidade da PCB através de projeto de dissipação termica correto, mantendo ela simples e com características de projeto seguindo princípios de confibilidade	10	3	4	120

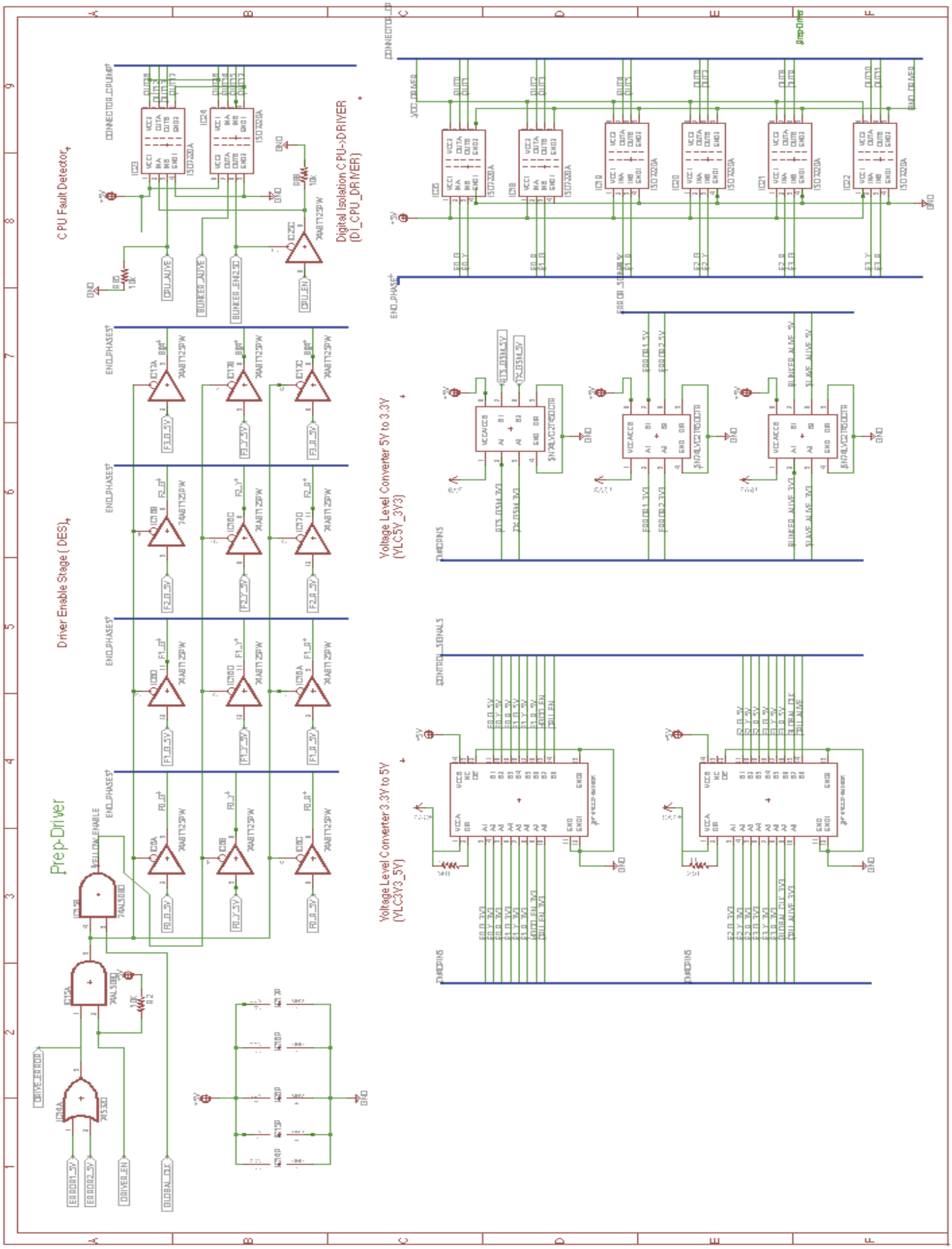
APÊNDICE B – ESQUEMÁTICOS DE PROJETO

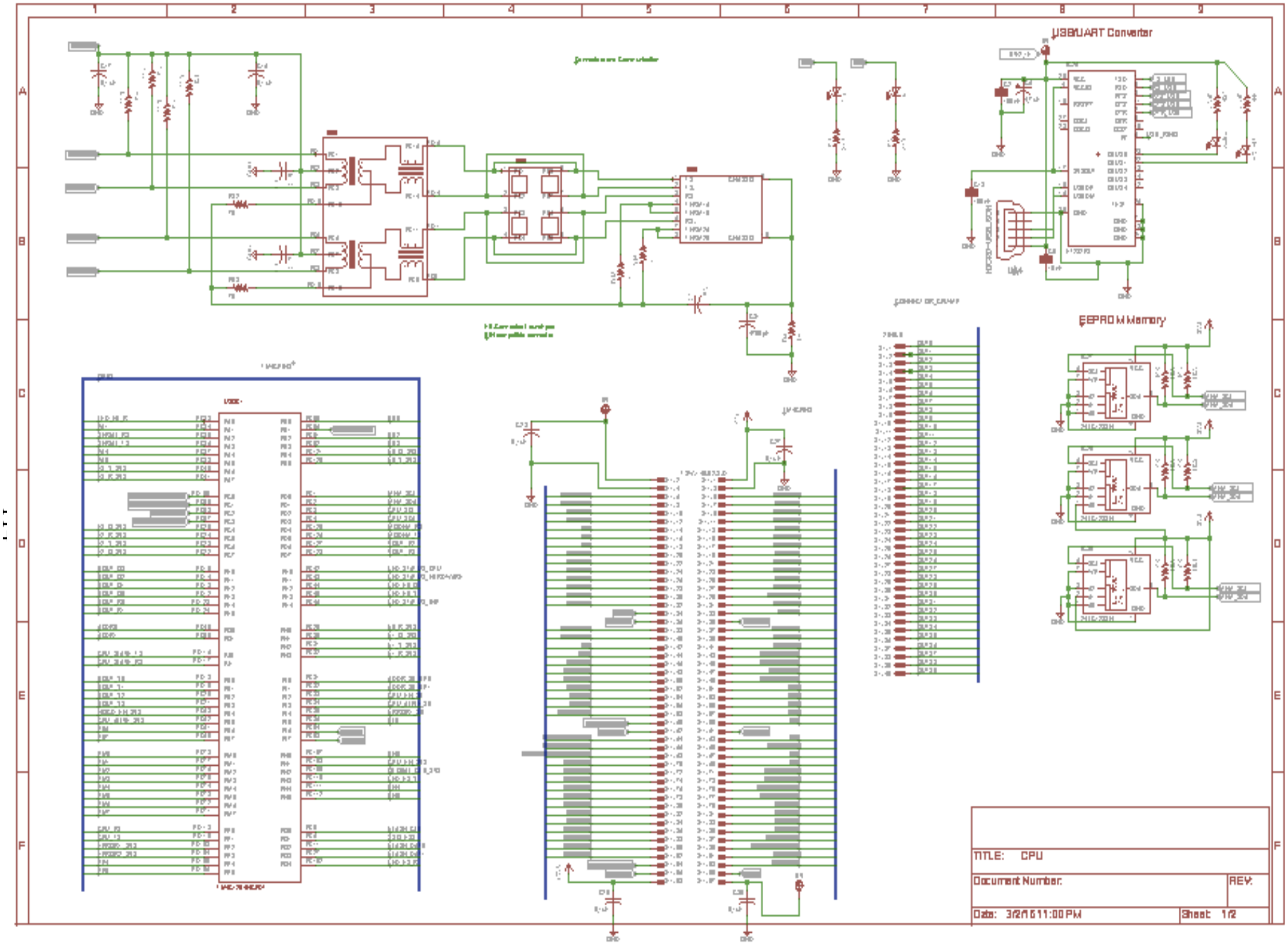


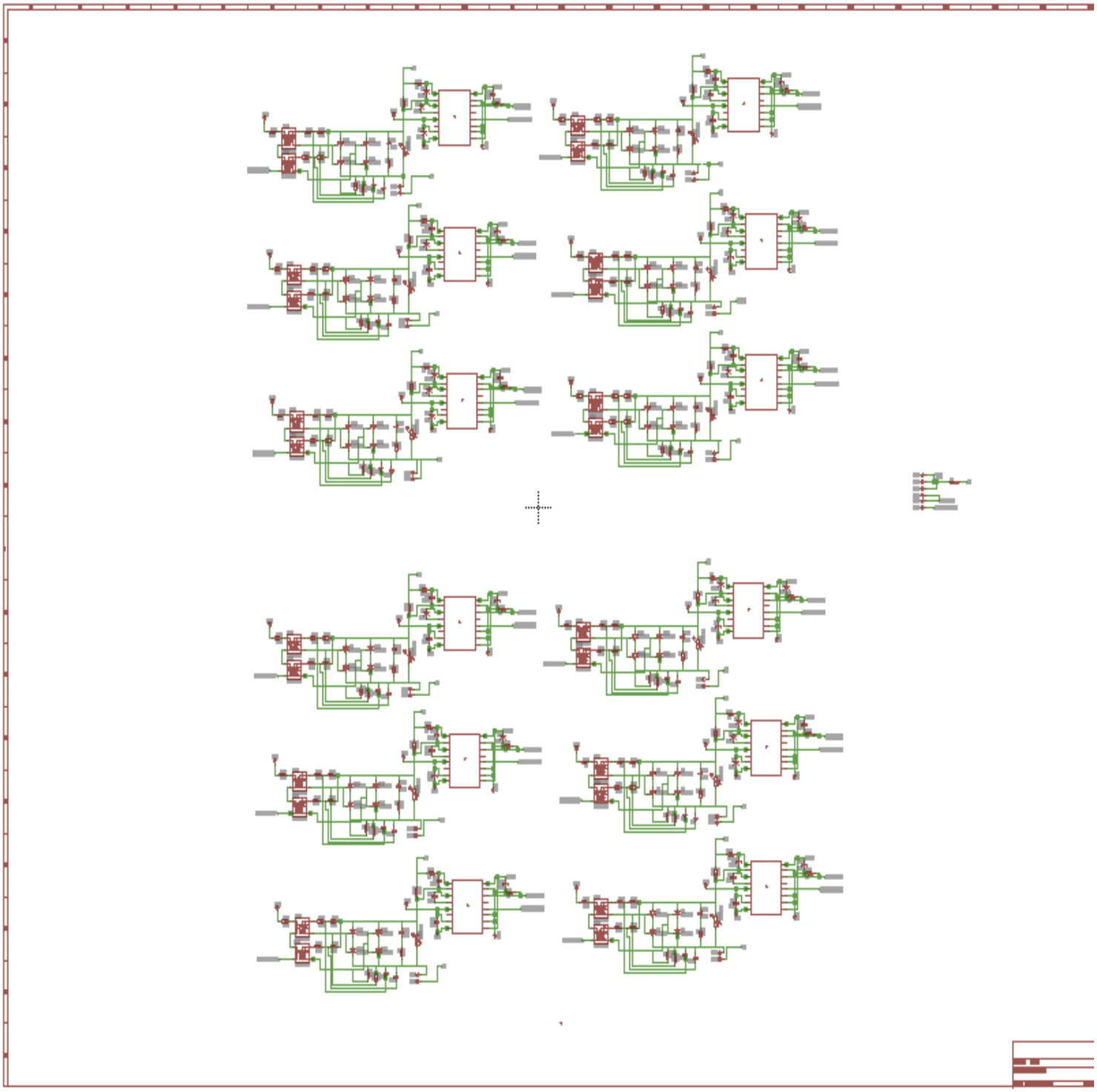

MODEM_GSM
SIM808
TITLE: CPU
DocumentNumber:
REV:
Sheet 32
Date 3/3/10 11:00 PM

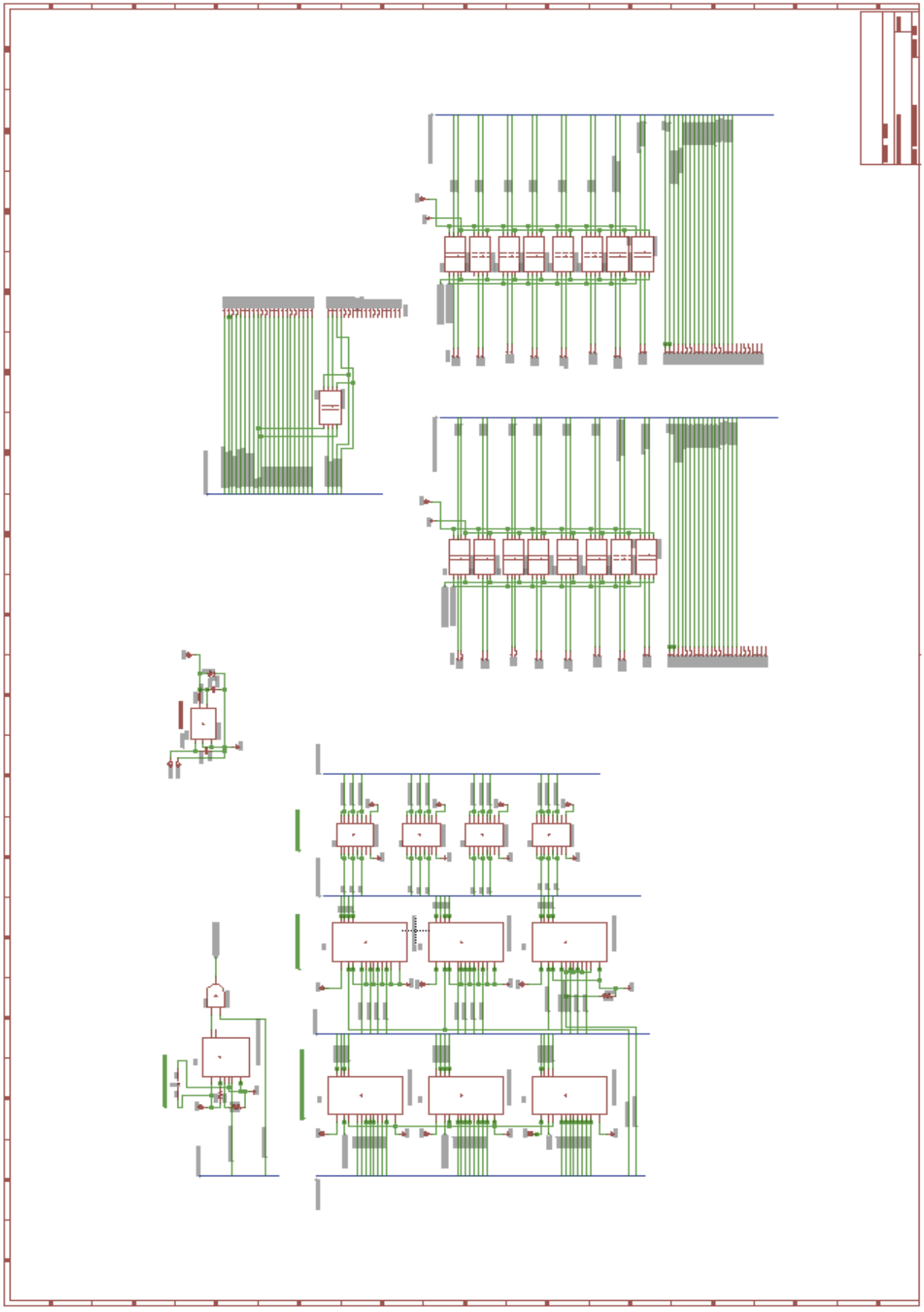

TITLE: CPU
Document Number:
REV:
Sheet 1/2
Power Supply1

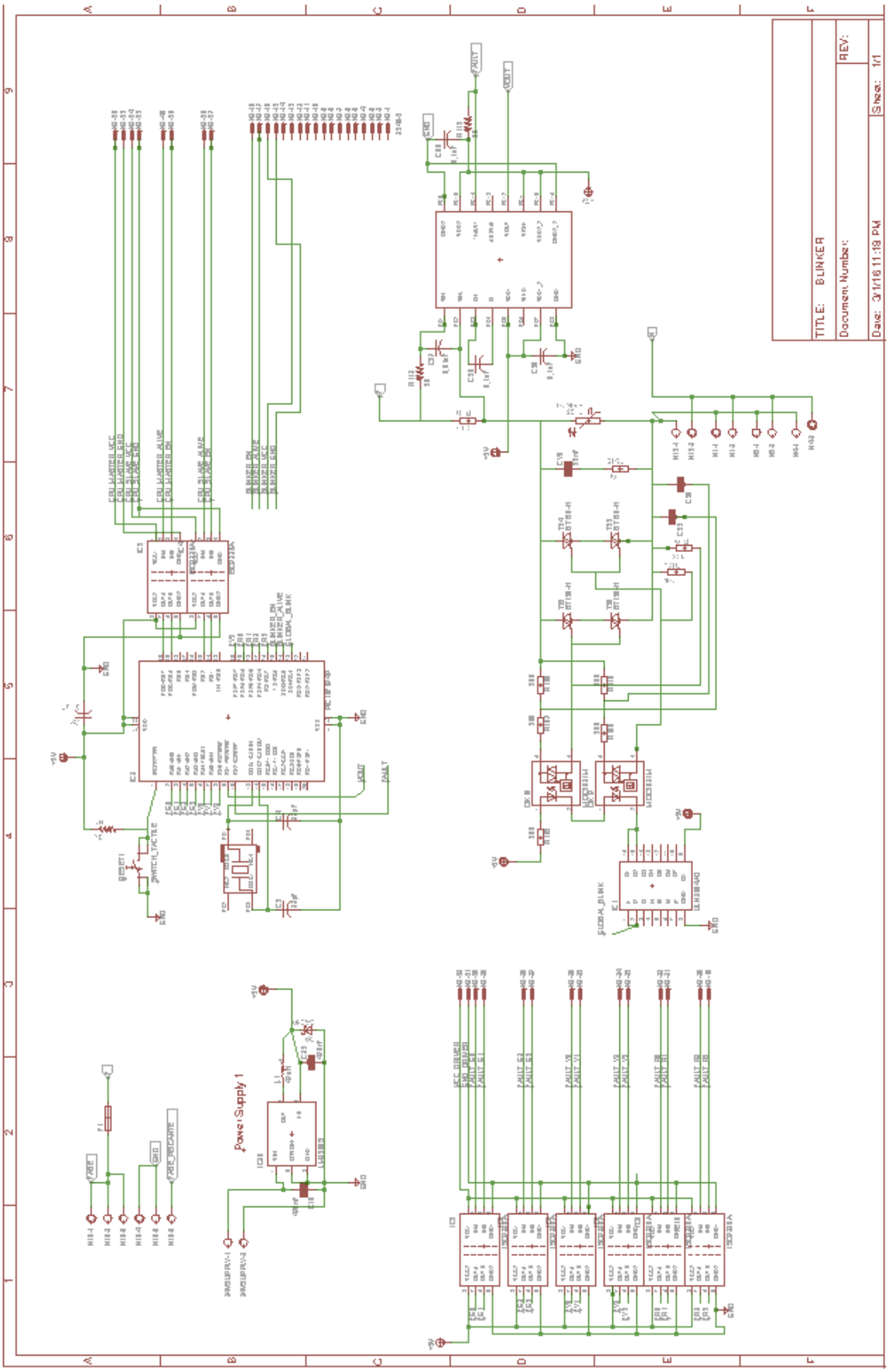

Hardwired Green Conflict Detector (HGCD)
Hardwired Logical Consistency Detector (HLCD)
HGCD Descrição :
O HGCD serve para detectar conflito de verde entre qualquer um dos grupos semafóricos previstos por CPU (4 GS). Caso haja mais de um foco verde em um determinado instante o modo piscante será acionado (pelo sinal ERROR2) . Caso o microcontrolador e o operador queira operar em um modo de menor confiabiliade para utilizar verdes simultâneos, neste caso, o sinal HGCD_DBN deve ser setado.
HLCD Descrição :
O HLCD(Hardwired Logical ConsistencyDetector) é um cirucito combinacional que verifica se de fato somente um foco de cada GS esta sendo acionado por vez. Este bloco pode ter como entrada os Fx_G, Fx_Y Fx_R, que são os sinais de controle e saídas do microcontrolador, ou pode também ler os sinais provenientes de leitura direta analogica de corrente direto dos focos. As saídas dos blocos são os sinais Fx_ERROR1, representando erros de consistência para cada GS isoladamente assim como se qualquer um dos GS tiver um erro de consistência o sinal ERROR1 será setado provocando entrada de modo piscante via hardware.
TITLE: CPU
Document Number:
REV:
Date: 3/2/11 11:00 PM
Sheet 1/2


Prep-Driver
Driver Enable Stage (DES)4
CPU Fault Detector4
Digital Isolation CPU->DRIVER (DI_CPU_DRIVER)
Voltage Level Converter 5V to 3.3V (VLC5V_3V3)
Voltage Level Converter 3.3V to 5V (VLC3V3_5V)
Prep-Driver


USB/UART Converter
EEPROM Memory
TITLE: CPU
Document Number:
REV:
Date: 3/2/11 11:00 PM
Sheet 1/2

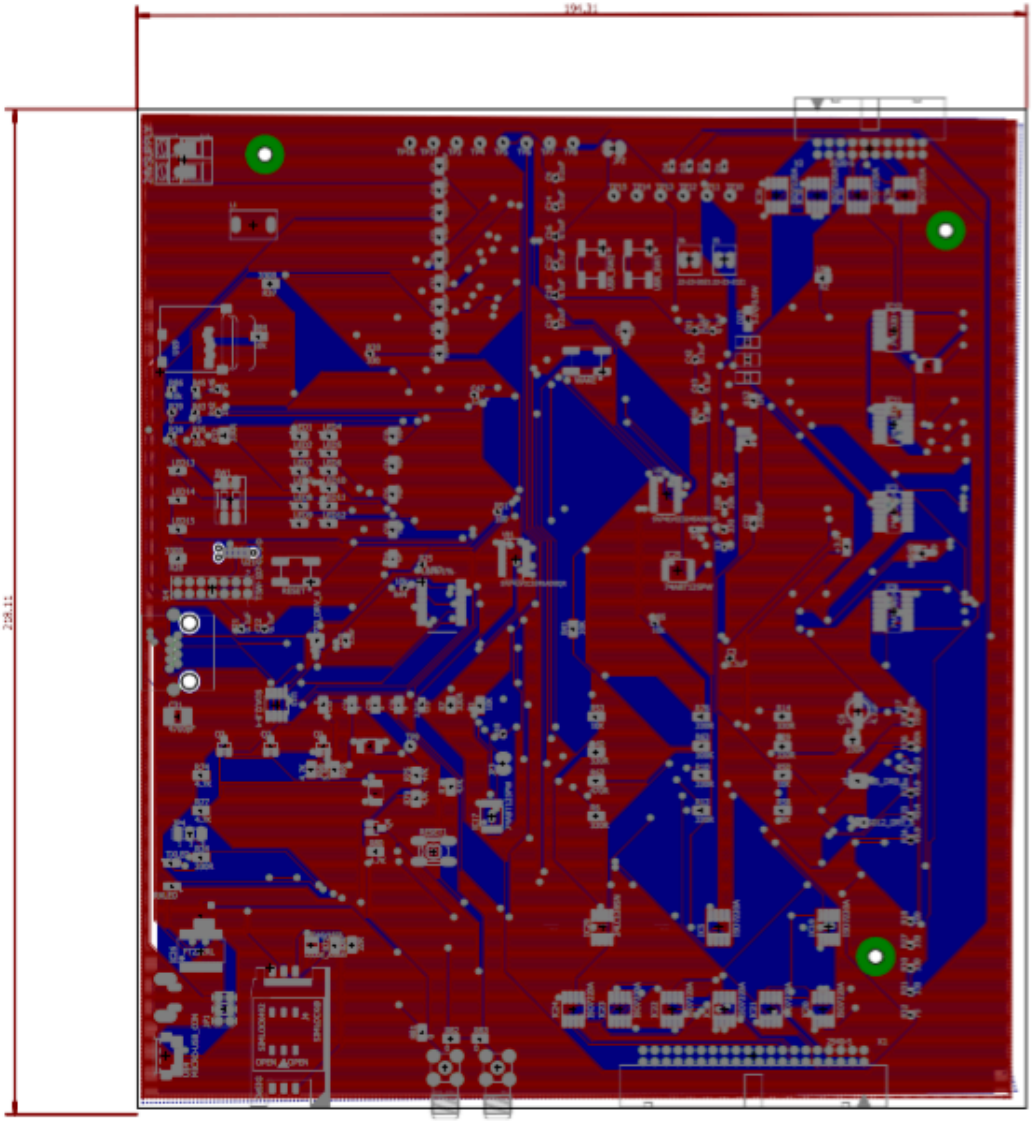


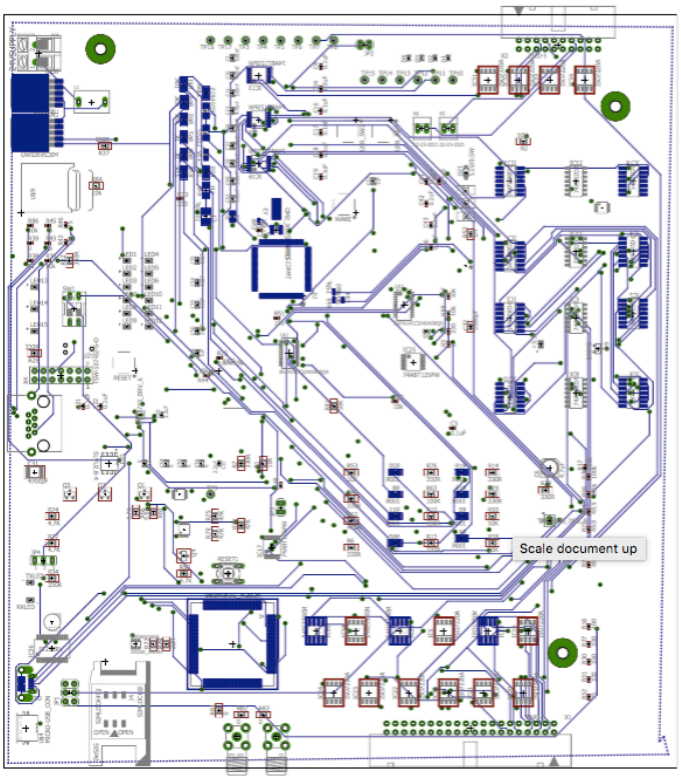



TITLE: BLINKER
Document Number:
REV:
Date: 3/1/16 11:19 PM
Sheet: 1/1
Power Supply 1

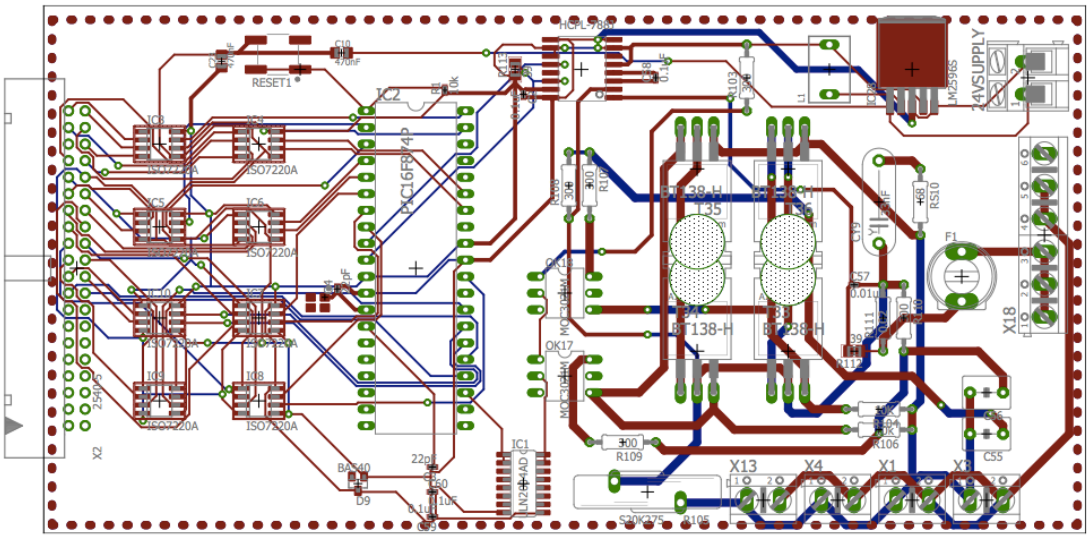
ÂPENDICE C – LAYOUT PLACAS DE CIRCUITO IMPRESSO

PLACA CPU VISÃO COM TODAS AS CAMADAS E VISÃO CAMADA BOTTOM

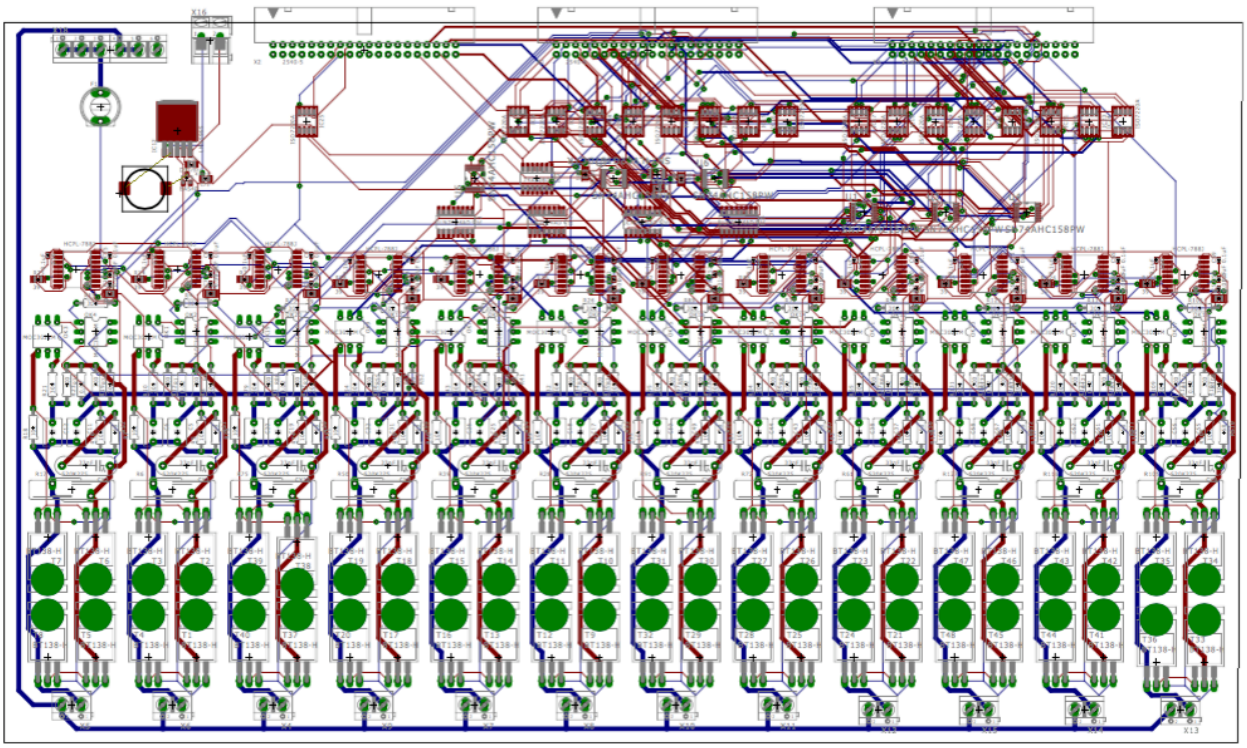




PLACA BLINKER



PLACA DRIVER



GLOSSÁRIO

Para esclarecimentos com relação aos termos que serão utilizados repetidas vezes no trabalho seguem abaixo algumas importantes definições de termos.

***Grupo Semafórico* (GS)**

Conjunto de um ou mais semáforos cujo estado de execução, ou seja, cor a ser exibida, é exatamente a mesma durante o mesmo intervalo de tempo. Também chamado de fase, por sempre ter estado de execução/cor em fase (sincronismo).

Cruzamento Isolado

Cruzamento isolado pode ser definido como um cruzamento semaforizado cuja programação dos semáforos associados com cada liberdade de movimento do cruzamento não depende de nenhum parâmetro externo ao cruzamento em si, como por exemplo, do tempo de verde referente a um segundo cruzamento em cascata (caso de um sincronismo para realização de onda verde).

Controlador semafórico ou controlador de tráfego

O controlador semafórico é um equipamento eletro/eletrônico capaz de acionar grupos semafóricos segundo padrões de estado de cor e tempos configuráveis por um usuário capacitado. Desta forma a funcionalidade inerente do equipamento trata-se do armazenagem de dados inseridos pelo usuário referente aos padrões de acionamento de cada um dos grupos semafóricos.

Plano de programação semafórica

O plano de programação semafórica é o conjunto de informações e regras inseridas em um controlador semafórico que permite que o equipamento siga padrões lógicos e de tempo no acionamento dos grupos semafóricos.

O plano de programação semafórico é composto *layers*, cada *layer* representa um tipo de dado ou informação diferente. A lógica de execução, que seria a primeira camada, é composta da tabela de cores, ou seja, qual a sequência de cores de cada grupo semafórico e seus respectivos tempos de duração, sendo estes os dados primordiais para a operação. Em seguida deve-se fornecer dados para que o controlador

semafórico realize rotinas de prevenção e detecção de falhas ou erros, que inclui uma tabela que indica quais pares de grupos semafóricos não devem estar ligados em verde simultaneamente. Outros parâmetros que caracterizem falhas podem ser inseridos nesta camada, porém tais informações dependem da capacidade de hardware do controlador semafórico em questão, como regras de analise de erros, ou configuração de correntes máximas e mínimas para acionamento dos focos.

É importante deixar claro que podem ser associadas várias lógicas de execução diferentes, embora de fato só seja seguro alterar os tempos de estado de cor, pois para um mesmo cruzamento a tabela de cores depende da infraestrutura das vias e regras de conversão, que tende a ser estática.

Na terceira camada deve-se inserir de fato um calendário com o planejamento do controlador, ou seja, em que dias e em que intervalo no dia uma determinada lógica será executada ou não. Este fato é importante pois permite diferentes métodos de controle conforma a natureza do fluxo de veículos no cruzamento em cada faixa horário ou data.

Controlador Semafórico Isolado

Controlador semafórico que aciona grupos semafóricos em um cruzamento isolado, sendo assim sua tabela de cores e tempos são fixos pois independem do efeito de outros cruzamentos.

Controlador Semafórico em modo atuado

Controlador semafórico capaz de adaptar automática seus tempos conforme fluxo no cruzamento, porém ser considerar fluxos em cruzamento próximos.

Central de controle semafórico/ Software Servidor

Software responsável pela comunicação remota entre vários controladores semafóricos servindo também de interface de supervisão e controle para os operadores da malha semafórica. A central pode prever métodos de otimização e controle para alterar planos semafóricos de toda a rede de controladores de modo colaborativo, automaticamente ou não.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] SIEMENS. **ST900 Intersection Controller Technical Specifications**. Disponivel em:
< http://www.siemens.co.uk/traffic/pool/documents/brochure/st900.pdf>. Acesso em: 5 jun. 2013.

[2] DENATRAN. **Manual Brasileiro de Sinalização de Trânsito Volume V- Sinalização Semafórica**. Disponivel em:

<http://www.denatran.gov.br/download/resolucoes/resolucao4832014_anexo.pdf>.

Acesso em: 7 jun 2015.

[3] DAGANZO, F. C. **Fundamentals of Transportation and Traffic Operations**. 1. ed. Bingley,UK: Emerald 1997

[4] MASHUR, A.C; SADEK, A. **Fundamentals of Intelligent Transportation Systems Planning** 1. Ed. ,Norwood,US: Artech House 2003

[5] U.S. DEPARTMENT OF DEFENSE. **MIL-HDBK-217F Military Handbook Reliability Prediction of Electronic Equipment.** Revision-F :Washington DC 1991. Disponível em: <http://www.weibull.com/mil_std/mil_hdbk_217f.pdf> . Acesso em: 25 fev 2016

[6] U.S. DEPARTMENT OF DEFENSE. **MIL-STD-756B Military Standard Reliability Modelling and Prediction.** Revision-B :Washington DC 1981. Disponível em: <http://www.weibull.com/mil_std/mil_std_756b.pdf> . Acesso em: 25 fev 2016

[7] U.S. DEPARTMENT OF DEFENSE. **MIL-HDBK-338B Military Handbook -Electronic Reliability Design Handbook.** Revision-B :Washington DC 1998. Disponível em: <http://www.weibull.com/mil_std/mil_hdbk_338b.pdf> . Acesso em: 25 fev 2016

[8] U.S. DEPARTMENT OF DEFENSE. **MIL-STD-785B Military Standard Reliability Program for System and Equipment Development and Production.** Revision-B :Washington DC 1980. Disponível em: < http://www.weibull.com/mil_std/mil_std_785b.pdf> . Acesso em: 25 fev 2016

[9] KLEIDERMACHER D. ; KLEIDERMACHER M. **Embedded Systems Security Practical Methods for Safe and Secure Software and Systems Development.** 1.Ed. US: Newnes 2012

[10] BAJENESCU, T.M. ; BAZU,M.I. ; **Component Reliability for Electronic Systems.**
1. Ed. ,Norwood, US : Artech House 2010

[11] CROWE, D.; FEINBERG A. ; **Design for Reliability .** 1. Ed. , Boca Raton, US : CRC Press 2010. Dísponivel em :<http://user.das.ufsc.br/~moreno/seguranca/confiabilidade/Design%20for%20Reliability.pdf>.
Acesso em: 25 fev 2016.

[12] INTERNATIONAL ELECTROTECHNICAL COMMISSION.; **IEC 61508-1 Functional safety of electrical/electronic/programmable electronic safety-related systems .** Version 4. , 1997

[13] U.S. DEPARTMENT OF DEFENSE. **MIL-STD 1629A .** Revision-A :Washington DC 1980. Disponível em: < http://everyspec.com/MIL-STD/MIL-STD-1600-1699/MIL-STD-1629_NOTICE-3_23100/> . Acesso em: 25 fev 2016

[14] ANALOG DEVICES . **Printed Circuit Board Design Issues.**. Disponível em: < http://www.analog.com/library/analogDialogue/archives/43-09/EDch%2012%20pc%20issues.pdf> . Acesso em: 25 fev 2016

[15] TEMPLETON, G. **AN1045/D Series Triacs In AC High Voltage Switching Circuits.**. Disponível em: <http://www.onsemi.com/pub_link/Collateral/AN1045-D.PDF > . Acesso em: 25 fev 2016

[16] NXP SEMICONDUCTORS. **Thyristors & Triacs – Ten Golden Rules for Success In Your Application**. Disponível em: <http://www.nxp.com/documents/application_note/AN_GOLDEN_RULES.pdf> . Acesso em: 25 fev 2016

[17] TEMPLETON, G. . **AN1048/D RC Snubber Networks For Thyristor Power Control and Transient Suppression**. Disponível em: < http://www.onsemi.com/pub_link/Collateral/AN1048-D.PDF> . Acesso em: 25 fev 2016

[18] DALGLEISH , M.; HOOSE, N. **Highway Traffic Monitoring and Data Quality** 1. Ed. ,Norwood,US: Artech House 2008